

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Бердянський державний педагогічний університет

Кваліфікаційна наукова
праця на правах рукопису

ГОРБАТЮК Ілля Анатолійович

УДК 378.147.091.31-051:004.4]:005.8(043.5)

ДИСЕРТАЦІЯ

**ПІДГОТОВКА МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО
ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ**

011 Освітні, педагогічні науки

01 Освіта/Педагогіка

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей,
результатів і текстів інших авторів мають посилання на відповідне джерело

І. А. Горбатюк

Науковий керівник: Кривильова Олена Анатоліївна, доктор педагогічних наук,
доцент

Запоріжжя – 2026

АНОТАЦІЯ

Горбатюк І. А. Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії з галузі знань 01 Освіта/Педагогіка за спеціальністю за спеціальністю 011 Освітні, педагогічні науки. – Бердянський державний педагогічний університет, Міністерство освіти і науки України, Запоріжжя, 2026.

Зміст анотації

У дисертації

вперше: теоретично обґрунтовано, розроблено й експериментально перевірено ефективність структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, яка складається з цільового, методологічного, змістово-процесуального та оцінно-результативного блоків; забезпечує результат, яким є сформована готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; передбачає поетапне запровадження оновленого змісту, форм організації освітнього процесу, методів і засобів навчання; реалізується завдяки створенню організаційно-педагогічних умов (системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проєктної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду);

уточнено поняття «підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління» і «готовність майбутніх бакалаврів з

інженерії програмного забезпечення до проєктного управління», тлумачення яких орієнтоване на предмет дослідження;

удосконалено критерії сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; зміст професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління завдяки впровадженню системи проблемно-орієнтованих та проєктно-практичних завдань з елементами управлінської діяльності;

подальшого розвитку набули наукові ідеї щодо професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення на основі концептуальних підходів (системного, синергетичного і компетентнісного) і специфічних (інтегративного та проєктного).

Підготовку майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління визначено як цілеспрямований, системно організований процес професійної освіти, спрямований на формування готовності здобувачів освіти до здійснення проєктно-управлінської діяльності в ІТ-сфері, що охоплює засвоєння знань, умінь, навичок, практичного досвіду та розвиток особистісних якостей, необхідних для планування, організації, координації та реалізації програмних проєктів в умовах невизначеності, командної взаємодії та змінних вимог сучасної ІТ-індустрії.

Готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління характеризується здатністю до виконання робіт пов'язаних з управлінням проєктами, яка структурно складається з функціональної основи та персональних якостей, а саме знань, умінь, навичок та якостей щодо планування та ретельного визначення проєктних та персональних пріоритетів; максимізації бізнес-цінностей проєкту; використання традиційних і agile інструментів, техніки та методів; пояснення важливих бізнес-аспектів проєкту іншим; керування елементами проєкту; контролю критично важливих технічних елементів проєктного управління.

Визначено структурні компоненти готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління: *функціональний*

компонент – включає здатність до планування та визначення пріоритетів (проектних і персональних); максимізації бізнес-цінності проекту шляхом розроблення ефективної стратегії його реалізації; співпраці зі спонсором проекту, командою та профільними експертами, необхідної для розроблення стратегії реалізації проекту; використання як традиційних, так і Agile-інструментів, технік і методів управління проектами; пояснення іншим важливих бізнес-аспектів проекту; управління елементами проекту, зокрема графіком, вартістю, ресурсами та ризиками; зосередження уваги на критично важливих технічних аспектах проектного управління; *особистісний компонент* – розкривається через сукупність особистісних якостей, а саме: відкритість до нового досвіду, сумлінність, екстраверсію, доброзичливість та емоційну стабільність.

Схарактеризовано критерії, показники та рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління. Когнітивний критерій відображає рівень засвоєння знань щодо методологій, принципів, інструментів і процесів проектного управління в галузі програмної інженерії, здатність аналізувати професійні ситуації та обґрунтовувати управлінські рішення. Операційно-діяльнісний критерій характеризує сформованість практичних умінь і навичок управління ІТ-проектами, зокрема планування робіт, управління вимогами, ризиками та ресурсами, використання традиційних і Agile-підходів, організації командної взаємодії, професійної комунікації та реалізації програмних проектів у межах PBL-діяльності. Особистісний критерій визначає рівень розвитку професійно значущих якостей, зокрема відповідальності, ініціативності, лідерства, здатності до співпраці, рефлексії, прийняття рішень в умовах невизначеності, відкритості до нового досвіду, сумлінності, доброзичливості та емоційної стабільності. Відповідно до визначених критеріїв виокремлено три рівні сформованості готовності: початковий (незадовільний), середній (задовільний) та високий (з відзнакою).

Теоретично обґрунтовано та розроблено структурно-функціональну модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління, яка складається із взаємопов'язаних і взаємозалежних

блоків: цільового, методологічного, змістово-процесуального й оцінно-результативного. Зміст цільового блоку структурно-функціональної моделі містить мету, яка полягає в підготовці майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Методологічний блок відображає підходи та принципи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління: концептуальні (системний, синергетичний і компетентнісний) і специфічні (інтегративний та проєктний). Змістово-процесуальний блок містить форми організації освітнього процесу, методи та засоби навчання дисциплін циклу професійної та практичної підготовки майбутніх бакалаврів з інженерії програмного забезпечення, а також розроблений навчальний курс «Теорія та практика проєктного управління в інженерії програмного забезпечення». Оцінно-результативний блок структурно-функціональної моделі визначає перевірку рівня сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління за визначеними критеріями та показниками.

Реалізація структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління здійснювалась у три етапи: початковий, основний та завершальний. Початковий етап був спрямований на формування мотиваційної основи професійної підготовки, розвиток інтересу до проєктного управління у сфері програмної інженерії та актуалізацію базових знань щодо організації ІТ-проєктів. На цьому етапі здійснювалося ознайомлення студентів зі специфікою проєктної діяльності, сучасними вимогами ІТ-індустрії, основами командної взаємодії та принципами проблемно-орієнтованого навчання. Особлива увага приділялася створенню проблемних ситуацій, розвитку пізнавальної мотивації та готовності до самостійної навчально-професійної діяльності. Основний етап передбачав цілеспрямоване формування функціонального та особистісного компонентів готовності до проєктного управління шляхом інтеграції проблемно-орієнтованого навчання, методу кейсів, командної проєктної роботи та практичних форм діяльності. Студенти залучалися до моделювання та розв'язання реальних професійних

ситуацій, виконання групових ІТ-проектів, аналізу кейсів, планування ресурсів, управління ризиками, комунікації зі стейкхолдерами та використання сучасних інструментів проєктного менеджменту. На цьому етапі забезпечувалося формування навичок прийняття управлінських рішень, командної взаємодії, відповідальності та рефлексії. Завершальний етап був орієнтований на узагальнення, систематизацію та практичне застосування набутих знань, умінь і навичок у контексті комплексної професійної діяльності. На цьому етапі здійснювалися оцінювання рівня сформованості готовності до проєктного управління, презентація результатів командних проєктів, самооцінювання та взаємооцінювання діяльності студентів, аналіз професійних досягнень і труднощів. Завершальний етап також передбачав розвиток здатності до самостійного виконання проєктно-управлінських функцій та готовності до професійної діяльності в умовах сучасної ІТ-індустрії.

Аналіз експериментальних даних засвідчив якісні зміни на початковому рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що зумовлено зменшенням кількості здобувачів вищої освіти цієї категорії впродовж педагогічного експерименту в КГ на 18,2 % та ЕГ на 31,1 %. Середній та високий рівні сформованості означеної готовності також продемонстрували позитивні якісні зміни: збільшення кількості здобувачів вищої освіти у КГ (на 11,3 % і 6,9 % відповідно) та ЕГ (на 12,9 % і 18,2 % відповідно). Отримані дані підтверджено використанням методів математичної статистики, зокрема розрахунку t-критерію Стьюдента ($t_{\text{табл.}}(1,969) < t_{\text{розрах.}}(12,27)$) та непараметричного критерію Пірсона ($\chi^2_{\text{експ.}}(15,998) > \chi^2_{\text{крит.}}(5,99)$).

Практичне значення отриманих результатів дослідження полягає в тому, що створено й упроваджено в процес професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення систему проблемно-орієнтованих та проєктно-практичних завдань з елементами управлінської діяльності (на прикладі змісту дисциплін циклу професійної та практичної підготовки здобувачів вищої освіти за спеціальністю 121 Інженерія програмного забезпечення: «Групова динаміка та командна комунікація» (3-й рік навчання), «Фінансовий менеджмент ІТ-проектів», «Менеджмент проєктів з розробки програмного забезпечення», «Економічне

обґрунтування ІТ-проектів» (4-й рік навчання)) та навчальний курс «Теорія та практика проектного управління в інженерії програмного забезпечення».

Доведено ефективність організаційних форм, активних методів і засобів навчання у підготовці майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління. Зокрема, дискусійні методи сприяють розвитку критичного мислення, аргументованого відстоювання професійної позиції та формуванню навичок командної комунікації; кейс-метод забезпечує формування здатності аналізувати професійні ситуації, приймати управлінські рішення та оцінювати можливі наслідки їх реалізації; проблемно-орієнтоване навчання (PBL) активізує пізнавальну діяльність студентів через розв'язання комплексних професійно спрямованих завдань, сприяє інтеграції теоретичних знань із практичною діяльністю та розвитку навичок роботи в умовах невизначеності. Комплексне застосування зазначених методів і підходів забезпечує формування функціональної та особистісної готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління відповідно до сучасних вимог ІТ-індустрії.

Ключові слова: проектне управління, готовність до проектного управління, інженерія програмного забезпечення, бакалаври з інженерії програмного забезпечення, структурно-функціональна модель професійної підготовки, організаційно-педагогічні умови, проблемне навчання, функціональний компонент, особистісний компонент, менеджмент ІТ-проектів, командна діяльність, професійна освіта.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Gorbatiuk I., Kryvylova O. Ukrainian it students and project management: are they prepared for industry demands? *SWorldJournal*. 2026. № 35-04. С. 142-161. <https://doi.org/10.30888/2663-5712.2026-35-04-102>

2. Горбатюк І. Організація підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління в умовах педагогічного

експерименту. *Інноваційна педагогіка*. 2026. Вип. 92, том 1. С. 220–225.
<https://doi.org/10.32782/ip/92.1.45>

3. Горбатюк І. Особливості проєктного управління в контексті інженерії програмного забезпечення. *Наукові записки БДПУ*. 2022. № 3. С. 267–274.
<https://doi.org/10.31494/2412-9208-2022-1-3-267-274>.

4. Горбатюк І. Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління на основі проблемного навчання. *Вісник КрНУ імені Михайла Остроградського*. Вип. 6/2025(155). С. 32–39.
<https://doi.org/10.32782/1995-0519.2025.6.4>

5. Горбатюк І. Структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Актуальні питання гуманітарних наук*. Вип. 95, том 1, 2026. С. 303–309.
<https://doi.org/10.24919/2308-4863/95-1-43>

6. Горбатюк І. Функціональна основа проєктного управління майбутніх бакалаврів з інженерії програмного забезпечення. *Освіта та розвиток дитини НАПН України*. 2023. № 3 (90). С. 22-26. [https://doi.org/10.32405/2309-3935-2023-3\(90\)-22-26](https://doi.org/10.32405/2309-3935-2023-3(90)-22-26)

Наукові праці, які засвідчують апробацію матеріалів дисертації :

7. Горбатюк І. Характеристика узгодженості підготовки майбутніх бакалаврів з інженерії програмного забезпечення з вимогами стейкхолдерів. *Цифрові технології у професійній діяльності* : матеріали Всеукр. наук.-практ. інтер.-конф., м. Бердянськ, 12-13 травня 2023 р. Бердянськ, 2023. С. 142-144
URL: <https://dspace.bdpu.org.ua/items/597751b9-03dd-47d2-9241-c119f9973eb5>

8. Горбатюк І. Оцінка вартості програмного забезпечення як методологічна проблема ІТ-галузі. *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації* : матеріали III Міжнар. наук.-практ. інтер.-конф., м. Запоріжжя, 30 верес. 2022 р. Запоріжжя, 2022. С. 506-510.
URL: <https://elar.tsatu.edu.ua/server/api/core/bitstreams/1fbc8169-7391-446d-b8f2-8c4720b7ae98/content>

9. Горбатюк І. Зміст підготовки майбутніх бакалаврів з інженерії

програмного забезпечення до проєктного управління. *Наукові засади підготовки фахівців інженерно-педагогічного та технологічного напрямків* : матеріали V Всеукр. наук.-практ. інтер.-конф., м. Запоріжжя, 15 трав. 2024 р. Запоріжжя, 2024. С. 21-24. URL: <https://dspace.bdpu.org.ua/items/138737b1-cba5-4143-ad97-5b07948a38b8>

10. Горбатюк І. А., Кривильова О. А. Trello як засіб формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Наукові підсумки 2024 року*: матеріали XIII наукової конференції. Харків, 2024. С. 44. URL: https://entc.com.ua/download/pidsumki_2024.pdf

11. Горбатюк І. А. Особистісний складник готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Науково-дослідна робота в системі підготовки фахівці-педагогів у природничій, технологічній і комп'ютерній галузях* : матеріали IX Всеукр. наук.-практ. інтер.-конф., м. Запоріжжя, 21-22 верес. 2023 р. Запоріжжя, 2023. С. 52-54. URL: <https://dspace.bdpu.org.ua/items/f1a5d847-7f15-4957-a151-3f2be02d2f7c>

12. Горбатюк І. А., Кривильова О. А. Методологічні підходи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Актуальні проблеми та перспективи технологічної і професійної освіти* : матеріали IX Всеукр. наук.-практ. конф., м. Тернопіль, 23-24 квіт. 2026 р. Тернопіль, 2026. С. 109-112. URL: <https://drive.google.com/drive/folders/1q3Mi3PD5SDTV0thQHZGju3b8vHdGGGgJ>

ANNOTATION

Horbatiuk, I. A. Training of Future Bachelors in Software Engineering for Project Management. – Qualifying scientific work on the rights of the manuscript.

The dissertation for obtaining the degree of the Doctor of Philosophy in the field of knowledge 01 Education/Pedagogy on a specialty 011 Educational, Pedagogical Sciences. – Berdyansk State Pedagogical University, Ministry of Education and Science

Annotation content

The dissertation presents, for the first time, a theoretically substantiated, developed, and experimentally verified structural-functional model for preparing future Bachelor's degree students in Software Engineering for project management. The model consists of target, methodological, content-processual, and evaluation-resultative blocks; ensures the achievement of the intended outcome, namely the formation of readiness of future software engineering bachelors for project management; provides for the phased implementation of updated content, forms of educational process organization, teaching methods, and learning tools; and is implemented through the creation of organizational and pedagogical conditions (systematic implementation of problem-based learning aimed at modeling and solving real professional situations related to project management in software engineering; updating the content of educational components considering the specifics of project management in software engineering and the modern requirements of the IT industry; ensuring pedagogical support and reflection involving consultation, feedback, and analysis of the results of problem-project activities for correcting the educational process and comprehending acquired experience).

The concepts of “preparation of future Bachelor's degree students in Software Engineering for project management” and “readiness of future Bachelor's degree students in Software Engineering for project management” have been clarified, with their interpretation oriented toward the subject of the research.

The criteria for the readiness of future bachelors in software engineering for project management have been improved; the content of professional training for future bachelors in software engineering for project management has been improved by introducing a system of problem-oriented and project-practical tasks with elements of management activity.

Scientific ideas regarding the professional training of future software engineering bachelors based on conceptual (systemic, synergetic, and competence-based) and specific (integrative and project-based) approaches have been further developed.

The preparation of future software engineering bachelors for project management is defined as a purposeful, systematically organized process of professional education aimed at forming students' readiness to perform project-management activities in the IT sphere, encompassing the acquisition of knowledge, skills, abilities, practical experience, and the development of personal qualities necessary for planning, organizing, coordinating, and implementing software projects under conditions of uncertainty, teamwork, and changing requirements of the modern IT industry.

The readiness of future software engineering bachelors for project management is characterized by the ability to perform tasks related to project management, structurally consisting of a functional basis and personal qualities, namely knowledge, skills, abilities, and qualities related to planning and careful definition of project and personal priorities; maximizing project business value; using traditional and Agile tools, techniques, and methods; explaining important business aspects of the project to others; managing project elements; and controlling critically important technical elements of project management.

The structural components of the readiness of future software engineering bachelors for project management have been identified. The functional component encompasses the ability to plan and define priorities (project-related and personal); maximize project business value through the development of an effective implementation strategy; cooperate with the project sponsor, team members, and subject matter experts necessary for project strategy development; use both traditional and Agile project management tools, techniques, and methods; explain important business aspects of the project to others; manage project elements, including schedule, cost, resources, and risks; and focus on critically important technical aspects of project management. The personal component is revealed through a set of personality traits represented in the Big Five model, namely openness to experience, conscientiousness, extraversion, agreeableness, and emotional stability.

The criteria, indicators, and levels of readiness formation of future software engineering bachelors for project management have been characterized. The cognitive criterion reflects the level of acquisition of knowledge regarding methodologies, principles, tools, and processes of project management in software engineering, as well as the ability to analyze professional situations and justify managerial decisions. The operational-activity

criterion characterizes the formation of practical skills and abilities in IT project management, including work planning, requirements, risk, and resource management, the use of traditional and Agile approaches, organization of teamwork, professional communication, and implementation of software projects within PBL activities. The personal criterion determines the level of development of professionally significant qualities, including responsibility, initiative, leadership, cooperation ability, reflection, decision-making under uncertainty, openness to experience, conscientiousness, agreeableness, and emotional stability. According to the identified criteria, three levels of readiness formation were distinguished: initial (unsatisfactory), intermediate (satisfactory), and high (with distinction).

A structural-functional model for preparing future software engineering bachelors for project management has been theoretically substantiated and developed. It consists of interconnected and interdependent blocks: target, methodological, content-processual, and evaluation-resultative. The target block includes the purpose of preparing future software engineering bachelors for project management. The methodological block reflects the approaches and principles of preparation: conceptual (systemic, synergetic, and competence-based) and specific (integrative and project-based). The content-processual block includes forms of educational process organization, teaching methods, and learning tools for the disciplines of professional and practical training for future software engineering bachelors, as well as the developed academic course “Theory and Practice of Project Management in Software Engineering.” The evaluation-resultative block determines the assessment of the level of readiness formation according to the identified criteria and indicators.

The implementation of the structural-functional model for preparing future software engineering bachelors for project management was carried out in three stages: initial, main, and final. The initial stage was aimed at forming the motivational basis of professional training, developing interest in project management in software engineering, and actualizing basic knowledge about IT project organization. At this stage, students were introduced to the specifics of project activities, modern IT industry requirements, fundamentals of teamwork, and principles of problem-based learning. Particular attention

was paid to creating problem situations, developing cognitive motivation, and fostering readiness for independent educational and professional activities.

The main stage involved the purposeful formation of functional and personal components of readiness for project management through the integration of problem-based learning, case-study methods, team project work, and practical forms of activity. Students were involved in modeling and solving real professional situations, implementing group IT projects, analyzing cases, planning resources, managing risks, communicating with stakeholders, and using modern project management tools. At this stage, skills in managerial decision-making, teamwork, responsibility, and reflection were developed.

The final stage focused on generalization, systematization, and practical application of acquired knowledge, skills, and abilities in the context of comprehensive professional activity. During this stage, the level of readiness for project management was assessed, results of team projects were presented, self-assessment and peer assessment of students' activities were conducted, and professional achievements and difficulties were analyzed. The final stage also involved developing the ability to independently perform project-management functions and readiness for professional activity in the conditions of the modern IT industry.

The analysis of experimental data demonstrated qualitative changes at the initial level of readiness formation among future software engineering bachelors for project management, as evidenced by a decrease in the number of higher education students in this category during the pedagogical experiment by 18.2% in the control group (CG) and 31.1% in the experimental group (EG). The intermediate and high levels of readiness formation also demonstrated positive qualitative changes: an increase in the number of students in the CG (by 11.3% and 6.9%, respectively) and in the EG (by 12.9% and 18.2%, respectively). The obtained data were confirmed using methods of mathematical statistics, in particular Student's t-test ($t_{\text{tab.}} (1.969) < t_{\text{calc.}} (12.27)$) and Pearson's nonparametric criterion ($\chi^2_{\text{exp.}} (15.998) > \chi^2_{\text{crit.}} (5.99)$).

The practical significance of the obtained research results lies in the creation and implementation, within the professional training process of future software engineering bachelors, of a system of problem-oriented and project-practical tasks with elements of

managerial activity (based on the content of disciplines within the cycle of professional and practical training for higher education students majoring in Specialty 121 Software Engineering: “Group Dynamics and Team Communication” (3rd year of study), “Financial Management of IT Projects,” “Software Development Project Management,” “Economic Justification of IT Projects” (4th year of study)), as well as the academic course “Theory and Practice of Project Management in Software Engineering.”

The effectiveness of organizational forms, active teaching methods, and learning tools in preparing future software engineering bachelors for project management has been proven. In particular, discussion methods contribute to the development of critical thinking, reasoned defense of professional positions, and the formation of teamwork communication skills; the case-study method ensures the development of the ability to analyze professional situations, make managerial decisions, and evaluate possible consequences of their implementation; problem-based learning (PBL) activates students’ cognitive activity through solving complex professionally oriented tasks, promotes integration of theoretical knowledge with practical activity, and develops skills for working under conditions of uncertainty. The comprehensive application of these methods and approaches ensures the formation of functional and personal readiness of future software engineering bachelors for project management in accordance with the modern requirements of the IT industry.

Keywords: project management, readiness for project management, software engineering, software engineering bachelors, structural-functional model of professional training, organizational and pedagogical conditions, problem-based learning, functional component, personal component, IT project management, teamwork, professional education.

LIST OF PUBLISHED WORKS ON THE THEME OF THE DISSERTATION

Scientific works in which the main scientific results of the dissertation are published:

1. Gorbatiuk, I., & Kryvylova, O. (2026). Ukrainian IT students and project management: Are they prepared for industry demands? *SWorldJournal*, issue 35(04), 142–161. <https://doi.org/10.30888/2663-5712.2026-35-04-102>

2. Horbatiuk, I. (2026). Orhanizatsiia pidhotovky maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektnoho upravlinnia v umovakh pedahohichnoho eksperymentu. *Innovatsiina pedahohika – Innovative pedagogy*, issue 92, Vol. 1, 220–225 [in Ukrainian]. <https://doi.org/10.32782/ip/92.1.45>

3. Horbatiuk, I. (2022). Osoblyvosti proiektnoho upravlinnia v konteksti inzhenerii prohramnoho zabezpechennia. *Naukovi zapysky Berdianskoho derzhavnoho pedahohichnoho universytetu. Seriiia : Pedahohichni nauky – Scientific notes of Berdyansk State Pedagogical University. Series: Pedagogical sciences*, issue 3, 267–274 [in Ukrainian]. <https://doi.org/10.31494/2412-9208-2022-1-3-267-274>

4. Horbatiuk, I. (2025). Pidhotovka maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektnoho upravlinnia na osnovi problemnoho navchannia. *Visnyk Kremenchutskoho natsionalnoho universytetu imeni Mykhaila Ostrohradskoho – Bulletin of Mykhailo Ostrohradskyi Kremenchuk National University*, issue 6(155), 32–39 [in Ukrainian]. <https://doi.org/10.32782/1995-0519.2025.6.4>

5. Horbatiuk, I. (2026). Strukturno-funktsionalna model pidhotovky maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektnoho upravlinnia. *Aktualni pytannia humanitarnykh nauk – Current issues of humanitarian sciences*, issue 95, Vol. 1, 303–309 [in Ukrainian]. <https://doi.org/10.24919/2308-4863/95-1-43>

6. Horbatiuk, I. (2023). Funktsionalna osnova proiektnoho upravlinnia maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia. *Osvita ta rozvytok dytyny NAPN Ukrainy – Education and child development of the National Academy of Educational Sciences of Ukraine*, 3(90), 22–26 [in Ukrainian]. [https://doi.org/10.32405/2309-3935-2023-3\(90\)-22-26](https://doi.org/10.32405/2309-3935-2023-3(90)-22-26)

Scientific works certifying the approval of the dissertation materials :

7. Horbatiuk, I. (2023). Kharakterystyka uzghodzhenosti pidhotovky maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia z vymohamy steikkholderiv. Tsyfrovi tekhnolohii u profesiinii diialnosti : materialy Vseukrainskoi naukovo-praktychnoi internet-konferentsii, Berdiansk, May 12–13, 2023, 142–144 [in Ukrainian]. URL: <https://dspace.bdpu.org.ua/items/597751b9-03dd-47d2-9241-c119f9973eb5>

8. Horbatiuk, I. (2022). Otsinka vartosti prohramnoho zabezpechennia yak

metodolohichna problema IT-haluzi. Rozvytok suchasnoi nauky ta osvity: realii, problemy yakosti, innovatsii : materialy III Mizhnarodnoi naukovo-praktychnoi internet-konferentsii, Zaporizhzhia, September 30, 2022, 506–510 [in Ukrainian].

URL: <https://elar.tsatu.edu.ua/server/api/core/bitstreams/1fbc8169-7391-446d-b8f2-8c4720b7ae98/content>

9. Horbatiuk, I. (2024). Zmist pidhotovky maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektneho upravlinnia. Naukovi zasady pidhotovky fakhivtsiv inzhenerno-pedahohichnoho ta tekhnolohichnoho napriamkiv : materialy V Vseukrainskoi naukovo-praktychnoi internet-konferentsii, Zaporizhzhia, May 15, 2024, 21–24 [in Ukrainian]. URL: <https://dspace.bdpu.org.ua/items/138737b1-cba5-4143-ad97-5b07948a38b8>

10. Horbatiuk, I.A., Kryvylova, O.A. (2024). Trello yak zasib formuvannia hotovnosti maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektneho upravlinnia. Naukovi pidsumky 2024 roku : materialy XIII naukovo konferentsii, Kharkiv, 44 [in Ukrainian]. URL: https://entc.com.ua/download/pidsumki_2024.pdf

11. Horbatiuk, I.A. (2023). Osobystisnyi skladnyk hotovnosti maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektneho upravlinnia. Naukovo-doslidna robota v systemi pidhotovky fakhivtsi-pedahohiv u pryrodnychii, tekhnolohichnii i kompiuternii haluziakh : materialy IX Vseukrainskoi naukovo-praktychnoi internet-konferentsii, Zaporizhzhia, September 21–22, 2023, 52–54 [in Ukrainian]. URL: <https://dspace.bdpu.org.ua/items/f1a5d847-7f15-4957-a151-3f2be02d2f7c>

12. Horbatiuk, I.A. (2026). Metodolohichni pidkhody pidhotovky maibutnikh bakalavriv z inzhenerii prohramnoho zabezpechennia do proiektneho upravlinnia. Aktualni problemy ta perspektyvy tekhnolohichnoi i profesiinoi osvity : materialy IX Vseukrainskoi naukovo-praktychnoi konferentsii, Ternopil, April 23–24, 2026 [in Ukrainian]. URL: <https://drive.google.com/drive/folders/1q3Mi3PD5SDTV0thQHZGju3b8vHdGGGgJ>

ЗМІСТ

ВСТУП	19
РОЗДІЛ 1. ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ОСНОВИ ПІДГОТОВКИ МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ	27
1.1. Особливості проєктного управління в контексті інженерії програмного забезпечення	27
1.2. Структура готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління	43
1.3. Методологічні підходи до підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління	77
Висновки до розділу 1	94
РОЗДІЛ 2. МОДЕЛЮВАННЯ ПІДГОТОВКИ МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ	98
2.1. Організаційно-педагогічні умови підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління	98
2.2. Структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління	116
Висновки до розділу 2	126
РОЗДІЛ 3. ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ЕФЕКТИВНОСТІ СТРУКТУРНО-ФУНКЦІОНАЛЬНОЇ МОДЕЛІ ПІДГОТОВКИ МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ	130
3.1. Організація та проведення експериментального дослідження ефективності структурно-функціональної моделі підготовки	

майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління	130
3.2. Аналіз результатів експериментального дослідження з оцінки ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління	150
Висновки до розділу 3	158
ВИСНОВКИ	161
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	166
ДОДАТКИ	186

ВСТУП

Актуальність теми дослідження зумовлена комплексом соціально-економічних, освітніх і професійних чинників, що визначають сучасний стан та перспективи розвитку ІТ-галузі України. Насамперед це пов'язано зі стрімким розвитком вітчизняної ІТ-індустрії, яка ще станом на 2021 рік налічувала близько 300 тисяч фахівців і демонструвала стабільні темпи щорічного зростання на рівні 25–30 %. ІТ-компанії активно інвестують у власні освітні проекти, водночас потреба ринку праці у фахівцях цієї галузі становить 30–50 тисяч осіб щороку, що майже у два з половиною рази перевищує кількість випускників бакалаврських програм відповідних спеціальностей. Особливої уваги потребує той факт, що близько 80 % працівників ІТ-компаній становить молодь віком від 18 до 32 років, для якої надзвичайно важливими є не лише ґрунтовні професійні знання, а й сформовані навички командної взаємодії, управління ресурсами, організації процесів розроблення програмного забезпечення та реалізації ІТ-проектів.

В умовах повномасштабної війни в Україні, її ІТ-сектор зберігає високий рівень функціональної активності та залишається одним із ключових експортно орієнтованих секторів економіки держави, що забезпечує значні обсяги експорту послуг, які становлять 1,68 млрд доларів. Наявність близько 7000 відкритих вакансій на спеціалізованих платформах працевлаштування станом на 2023 рік засвідчує стійкий попит на висококваліфікованих ІТ-фахівців та посилює потребу в підготовці конкурентоспроможних бакалаврів з інженерії програмного забезпечення, здатних ефективно працювати в умовах швидкозмінного технологічного середовища та високої відповідальності за результати командної діяльності.

Суттєвим викликом для системи вищої освіти залишається необхідність забезпечення високої якості та надійності програмного забезпечення в умовах постійного ускладнення ІТ-проектів, цифровізації суспільства та зростання вимог роботодавців до професійної компетентності молодих спеціалістів. Практика

діяльності ІТ-компаній свідчить про наявність розриву між рівнем теоретичної підготовки випускників закладів вищої освіти та реальними потребами сучасного ІТ-ринку, зокрема щодо сформованості компетентностей у сфері проектного управління, командної комунікації, оцінювання ризиків, планування ресурсів і координації процесів розроблення програмного забезпечення.

Результати аналізу навчально-методичного забезпечення підготовки майбутніх бакалаврів з інженерії програмного забезпечення засвідчили існування дисбалансу між динамічним розвитком ІТ-індустрії та темпами оновлення освітніх програм у закладах вищої освіти. Унаслідок цього сформовані у здобувачів вищої освіти компетентності не повною мірою відповідають актуальним потребам ІТ-галузі та вимогам роботодавців. Незважаючи на ухвалення Закону України «Про державну підтримку розвитку індустрії програмної продукції» (2013), який передбачає сприяння реалізації проєктів дуальної форми здобуття освіти та посилення практичної складової професійної підготовки, зміст більшості навчальних дисциплін, підручників і навчально-методичних матеріалів усе ще недостатньо орієнтований на формування компетентностей з проектного управління в ІТ-сфері.

Теоретичну основу дослідження становлять праці науковців у галузі ІТ (О. Бойко, Б. Жебровський, О. Капітанець, А. Клімова, С. Рибніков, О. Романовський, В. Свистун, Т. Сорочан, С. Тарасов та ін.); у сфері проектного управління (Н. Блага, Л. Довгань, О. Кіріленко, М. Малезик, І. Малик, М. Микитюк, Г. Мохонько та ін.); щодо реалізації компетентнісного, синергетичного, інтегративного та проектного підходів у професійній освіті (С. Гончаренко, М. Гриньова, Р. Гуревич, О. Дубасенюк, В. Захарченко, С. Калашнікова, С. Клепко, І. Козловська, Д. Коломієць, Т. Кристопчук, В. Кушнір, О. Локшина, В. Луговий, Ю. Мальований, Л. Ноздріна, О. Овчарук, О. Олейникова, О. Пометун, Ю. Рашкевич, К. Рихтер, Я. Собко, Ж. Таланова, Т. Якимович та ін.).

Водночас аналіз сучасних наукових праць засвідчує відсутність цілісного

системного дослідження проблеми формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління в умовах сучасного ІТ-середовища. Окремі аспекти цієї проблематики висвітлено у працях А. Бондарчук, Г. Качан, Н. Краснокутської, С. Семерікова, О. Сметанюк та А. Стрюка, однак питання комплексного поєднання професійної підготовки майбутніх ІТ-фахівців із формуванням компетентностей у сфері управління ІТ-проєктами потребує подальшого теоретичного обґрунтування та практичного розв'язання. Це зумовлює необхідність пошуку ефективних шляхів удосконалення професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення відповідно до сучасних викликів цифрового суспільства та потреб ринку праці.

З огляду на результати аналізу соціально-економічних вимог, науково-педагогічної літератури, сучасної освітньої практики з підготовки майбутніх бакалаврів з інженерії програмного забезпечення і тенденцій її розвитку встановлено суперечності, що виникли між:

- запитами сучасної української ІТ-галузі на фахівців, здатних здійснювати ефективне проєктне управління, і обмеженим рівнем інтеграції управлінського компонента у професійну підготовку майбутніх бакалаврів з інженерії програмного забезпечення в закладах вищої освіти

- потребами забезпечення високого рівня готовності майбутніх фахівців з інженерії програмного забезпечення до проєктного управління як обов'язкового компонента їхнього професіоналізму й недостатнім теоретико-методологічним і методичним обґрунтуванням його формування у закладах вищої освіти;

- потребою у цілеспрямованому формуванні готовності майбутніх фахівців з інженерії програмного забезпечення до проєктного управління в процесі професійної підготовки та недостатнім рівнем розробленості методичного забезпечення цього процесу.

Отже, актуальність досліджуваної проблеми, її недостатня теоретико-методична та практична розробленість і виявлені суперечності зумовили вибір

теми дослідження «Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління».

Зв'язок роботи з науковими програмами, планами, темами. Дисертацію виконано відповідно до тематичного плану Бердянського державного педагогічного університету в межах комплексної теми кафедри комп'ютерних технологій та інформатики (2021-2025 рр.) «Розробка методик дуального навчання майбутніх бакалаврів професійної освіти в галузі цифрових технологій» (0121U110532) та кафедри професійної освіти та технологій (2024-2028 рр.): «Теорія та практика підготовки фахівців-педагогів системи професійної та технологічної освіти в умовах інноваційного середовища» (0124U001509).

Тему дисертації затверджено вченою радою Бердянського державного педагогічного університету (протокол № 4/1 від 27.10.2022 року) та узгоджено в бюро Міжвідомчої ради з координації досліджень у галузі освіти, педагогіки і психології (протокол № 1 від 14.03.2023 року).

Об'єкт дослідження – професійна підготовка майбутніх бакалаврів з інженерії програмного забезпечення.

Предмет дослідження – формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Мета дослідження – теоретично обґрунтувати, розробити та експериментально перевірити ефективність структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Відповідно до мети визначено такі **завдання дослідження**:

1. На основі аналізу сучасної теорії та практики розкрити сутність базових понять дослідження та особливості підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

2. Визначити структуру готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, схарактеризувати критерії, показники та рівні.

3. Теоретично обґрунтувати та розробити структурно-функціональну модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

4. Визначити організаційно-педагогічні умови реалізації структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

5. Експериментально перевірити ефективність структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Для досягнення мети, розв'язання завдань обрано сукупність взаємопов'язаних методів дослідження: теоретичних: аналіз (системний, проблемно-цільовий, нормативно-порівняльний, контент-аналіз навчальних планів і робочих програм, підручників, методичних рекомендацій тощо) – для виявлення стану теорії та практики підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; узагальнення й систематизація науково-теоретичних положень – для уточнення сутності основних наукових понять, істотних для розробки проблеми дослідження; моделювання – для розробки й теоретичного обґрунтування структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; емпіричних: діагностика (анкетування, тестування, аналіз результатів діяльності здобувачів вищої освіти та інше) – для оцінки рівнів сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; педагогічний експеримент – для перевірки ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; статистичні методи (кількісна та якісна обробка даних, графічне подання результатів) – для відстеження динаміки рівнів сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління та встановлення наукової достовірності отриманих результатів дослідження.

Наукова новизна отриманих результатів полягає в тому, що:

вперше: теоретично обґрунтовано, розроблено й експериментально перевірено ефективність структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, яка складається з цільового, методологічного, змістово-процесуального та оцінно-результативного блоків; забезпечує результат, яким є сформована готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; передбачає поетапне запровадження оновленого змісту, форм організації освітнього процесу, методів і засобів навчання; реалізується завдяки створенню організаційно-педагогічних умов (системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проєктної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду);

уточнено поняття «підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління» і «готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління», тлумачення яких орієнтоване на предмет дослідження;

удосконалено критерії сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; зміст професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління завдяки впровадженню системи проблемно-орієнтованих та проєктно-практичних завдань з елементами управлінської діяльності;

подальшого розвитку набули наукові ідеї щодо професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення на основі концептуальних підходів (системного, синергетичного і компетентнісного) і специфічних (інтегративного та проєктного).

Практичне значення отриманих результатів дослідження полягає в тому, що створено й упроваджено в процес професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення систему проблемно-орієнтованих та проєктно-практичних завдань з елементами управлінської діяльності (на прикладі змісту дисциплін циклу професійної та практичної підготовки здобувачів вищої освіти за спеціальністю 121 Інженерія програмного забезпечення: «Групова динаміка та командна комунікація» (3-й рік навчання), «Фінансовий менеджмент ІТ-проєктів», «Менеджмент проєктів з розробки програмного забезпечення», «Економічне обґрунтування ІТ-проєктів» (4-й рік навчання)) та навчальний курс «Теорія та практика проєктного управління в інженерії програмного забезпечення».

Матеріали дослідження впроваджено в освітні процеси наступних закладів вищої освіти: Західноукраїнській національний університет (довідка №126.27/1248 від 20 травня 2026 року), Луцький національний технічний університет (довідка №1522/01-14 від 26 травня 2026 року) та Черкаський національний університет імені Богдана Хмельницького (довідка №137/04 від 25 травня 2026 року).

Матеріали дослідження можуть бути використані під час професійної підготовки здобувачів першого (бакалаврського) рівня вищої освіти зі спеціальності 121 Інженерія програмного забезпечення.

Апробація матеріалів дисертації. Основні положення та результати дослідження доповідалися й обговорювалися на науково-практичних конференціях: «Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації» (Запоріжжя, 2022); «Цифрові технології у професійній діяльності» (Запоріжжя, 2023); «Науково-дослідна робота в системі підготовки фахівців-педагогів у природничій, технологічній і комп'ютерній галузях» (Запоріжжя, 2023); «Наукові засади підготовки фахівців інженерно-педагогічного та технологічного напрямків» (Запоріжжя, 2024); «Наукові підсумки 2024 року» (Харків, 2024); «Актуальні проблеми та перспективи технологічної і професійної освіти» (Тернопіль, 2026).

Публікації. Основні теоретичні положення й висновки дисертації

відображено в 12 публікаціях автора, з них: 5 статей у наукових фахових виданнях України з психолого-педагогічних наук, 1 стаття в зарубіжному науковому виданні, 6 тез доповідей у матеріалах конференцій.

Структура та обсяг дисертації. Дисертаційна робота складається з анотацій українською та англійською мовами, вступу, трьох розділів, висновків до них, загальних висновків, списку використаних джерел (176 найменувань, з них – 65 іноземними мовами) і додатків (5 на 22 сторінках). Загальний обсяг дисертації становить 208 сторінок друкованого тексту. Роботу ілюстровано 18 таблицями та 9 рисунками.

РОЗДІЛ 1

ТЕОРЕТИКО-МЕТОДОЛОГІЧНІ ОСНОВИ ПІДГОТОВКИ МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ

1.1. Особливості проєктного управління в контексті інженерії програмного забезпечення

Під час війни ІТ-сектор України став однією з небагатьох галузей економіки, яка продовжує розвиватися, створювати нові робочі місця, впроваджувати нові проєкти та залучати інвестиції. Незважаючи на те, що повномасштабна війна вплинула на всі економічні сфери країни, ІТ-індустрія залишається надійною підтримкою для України та має перспективу стати одним із основних рушіїв її розвитку в майбутньому. Важливими напрямками ІТ-діяльності є інженерія програмного забезпечення та проєктне управління, між якими існує тісний взаємозв'язок. Управління проєктами передбачає розгляд множини сфер прояву життєдіяльності суспільства, що стосуються специфіки процесів предметної області призначення проєкту. Ведення ІТ-проєктів потребує побудови ефективних та прагматичних робочих процесів, включаючи створення та супровід якісного програмного забезпечення. Основними вимогами до якісного проєктного управління в ІТ стають розуміння фахівцями доменної області проєкту, проєктного стеку технології, знання особливостей взаємодії між командами; навички проєктної діяльності, що характеризуються ургентністю виконання, стислими строками та швидким результатом; здатність до врегулювання проєктних обмежень (бюджету, ресурсів, ризиків, дедлайнів та інше) [1]. Наразі існує розрив між рівнем підготовки випускників закладів вищої освіти і вимогами з боку роботодавців, які зауважують, що випускникам бракує умінь формувати мету програмного проєкту, виконувати колективну роботу і підтримувати комунікації, своєчасно завершувати розробку і впровадження програмних продуктів [2].

Отже, зі зростанням популярності ІТ-професій виникає й потреба у

вдосконаленні кадрового потенціалу ІТ-галузі, зокрема готовності фахівців швидко орієнтуватися у змінних умовах праці в системі інженерії програмного забезпечення в контексті проєктного управління.

Розробка програмного забезпечення в сучасних проєктах виконується згідно методологій Waterfall або Agile, що в свою чергу вимагає від учасників проєкту відповідних знань, вмінь та навичок. Окремі автори стверджують, що найбільш популярною методологією в Україні є Scrum [3, с. 234], а деякі великі компанії, як наприклад GlobalLogic, вже адаптували SAFe фреймворк і використовують його для своїх великих проєктів [4, с. 551]. Необхідність оволодіння ІТ-фахівцями сучасними методологіями спонукало окремі компанії України відкрити власні навчальні центри. Це стосується таких великих компаній як: SoftServe, EPAM, GlobalLogic, Intellias, Luxoft. Окрім того, в 2022 року в Україні стартував освітній проєкт IT Generation [5]. Він був спрямований на світчерів – людей, які ніколи не навчалися ІТ-спеціальностей та не працювали в індустрії. Це спільний проєкт Міністерства цифрової трансформації, Vinance Charity та Львівського ІТ Кластеру. За час проєкту успішно завершили навчання 1877 студентів. 147 випускників працевлаштувалися в ІТ. Найчастіше студенти проєкту знаходили роботу за напрямом Project (Product) Management, HR та JavaScript. Офери надали такі компанії та стартапи, як LTIMindtree, BehaviorQuant, Heimdal Data, ChoiceQR, NIX, Netpeak Agency, OmiSoft тощо. А найбільше працевлаштованих – серед випускників шкіл GoIT (20,6%), Source IT (19%) та A-Level (15,7%). Але є й ті, що не завершили навчання, – 323 студенти. Серед причин вони переважно зазначали, що їм на заваді стали проблеми, пов'язані з російською агресією (відключення електроенергії, релокація в інші країни) чи особисті проблеми. Найбільше студентів не довчилися на напрямі Frontend [6].

Загалом розмір ІТ-сектору з точки зору людських ресурсів складає 3,46 млн. спеціалістів, з яких 3,01 млн. мають вищу освіту, а це 87% [7]. Згідно статистичних даних Національного агентства із забезпечення якості вищої освіти, у 2023 році в Україні було 781 286 осіб, які отримували ступінь

бакалавра [8]. За результатами опитування Stack Overflow Developer Survey, у якому взяли участь понад 70 тис. респондентів, 70% мають вищу освіту. Вища освіта на думку організаторів має низку переваг, як-то: сильна теоретична база; можливість потрапити у середовище амбітних розумних студентів, зустріти сильних викладачів, знайти корисні зв'язки як серед одногрупників, так серед менторів, запрошених лекторів та викладацького складу, які підкажуть, які технології та фреймворки зараз варто вивчати та де знайти можливості для стажувань; можливість командної роботи з розробки проєктів, що розвиває не тільки hard-skills, але і soft skills програмістів; знання вищої математики та математичного аналізу без яких не обійтись тим, хто хоче працювати у сфері Machine Learning, Data Science, Big Data; доступ до роботи в окремих компаніях, зокрема в державному секторі; корисні суміжні дисципліни (програми багатьох вишів складені так, щоб ознайомитися з ширшим колом напрямків в ІТ – класичне програмування, робота з інтерфейсами, Machine Learning, Data Science, менеджмент), це спосіб зрозуміти, у якій сфері цікаво надалі розвиватися [9].

Роботодавці очікують, що у процесі навчання у закладах вищої освіти майбутні бакалаври з інженерії програмного забезпечення здобудуть усю необхідну базову підготовку, яку згодом застосовуватимуть у професійній діяльності [10]. Однак, маємо долати труднощі, що виникають при підготовці ІТ-фахівців в умовах закладів вищої освіти, а саме: тривалий період навчання на бакалавраті; відсутність практики та роботи з реальними кейсами та інше.

В умовах стрімкого розвитку інформаційних технологій і зростання складності програмних систем особливої значущості набуває ефективне проєктне управління в галузі інженерії програмного забезпечення. Специфіка цієї сфери зумовлюється високим рівнем невизначеності, динамічністю вимог, необхідністю командної взаємодії та інтеграції різних технічних і управлінських підходів. Саме тому формування у майбутніх бакалаврів з інженерії програмного забезпечення компетентностей у сфері проєктного управління є важливим складником їхньої професійної підготовки. У цьому контексті особливого значення набуває аналіз

ключових характеристик і принципів проєктного управління, адаптованих до умов розроблення програмних продуктів.

Сучасна концепція проєктного управління ґрунтується на ключовому понятті «проєкт», яке розглядається не лише як об'єкт управлінського впливу, наділений специфічними властивостями (обмеженість у часі, ресурсах, унікальність результату), але й як сутнісна характеристика самого процесу управління. У цьому контексті проєкт визначає логіку, структуру та зміст управлінської діяльності, формуючи її особливий, відмінний від інших видів управління характер.

З огляду на зазначене, термін «управління проєктом» певною мірою звужує зміст відповідного явища, оскільки створює уявлення про управління як універсальну діяльність, що лише спрямована на окремий об'єкт – проєкт. Такий підхід не повною мірою відображає специфіку процесів, методів і принципів, притаманних саме проєктно орієнтованій діяльності. Натомість термін «проєктне управління» акцентує увагу на проєктній природі самого управління, підкреслюючи, що його зміст, інструментарій і організація визначаються особливостями проєктів як унікальних систем. Використання поняття «проєктне управління» дозволяє більш чітко окреслити відмінності між цим видом управління та іншими його різновидами, зокрема функціональним чи процесним управлінням, оскільки воно відображає орієнтацію на досягнення конкретного результату в умовах обмежених ресурсів і визначених термінів. Водночас у науковій та практичній площині термін «управління проєктом» залишається більш усталеним і широко вживаним. Зважаючи на це, обидва терміни – «управління проєктом» і «проєктне управління» – можуть розглядатися як семантично близькі та взаємозамінні. Проте в межах даного дослідження доцільним є використання терміна «проєктне управління», оскільки він точніше відображає сутність досліджуваного явища та підкреслює його методологічну специфіку.

Логічним продовженням наведених міркувань є необхідність більш детального розгляду сутності поняття «проєкт», оскільки саме воно становить концептуальну основу проєктного управління. З'ясування змісту цього терміна

дозволяє глибше зрозуміти специфіку проектно орієнтованої діяльності, її відмінності від інших видів управління, а також особливості організації процесів планування, реалізації та контролю. Крім того, уточнення поняття «проект» є важливим для визначення ключових характеристик і ознак, що безпосередньо впливають на формування відповідних компетентностей у майбутніх фахівців з інженерії програмного забезпечення. У зв'язку з цим доцільним є звернення до аналізу наукових підходів до трактування поняття «проект» та виокремлення його основних змістових характеристик.

Поняття «проект» поєднує різноманітні види діяльності, що характеризуються низкою ознак, найбільш загальними з яких є такі: спрямованість на досягнення конкретних цілей; координоване виконання численних взаємозалежних дій; обмежена протяжність у часі, з певним початком та кінцем. Існує ряд визначень терміну «проект», кожне з яких має право на існування, залежно від конкретного завдання, яке стоїть перед фахівцем, як-то: щось, що замислюється чи планується [11]; процес переходу з вихідного стану в кінцевий результат за участю низки обмежень і механізмів [12, с. 8]; деяке завдання з певними вихідними даними та необхідними результатами (цілями), що включає задум, засоби його реалізації (вирішення проблеми) і одержувані в процесі реалізації результати [13]; комплекс спеціально організованих обмежених в часі, бюджеті дій, заходів, чи зусиль скерованих на створення унікального продукту (товари, послуги чи інші результати) визначеної якості [14;15].

Поняття «проект» достатньо чітко визначено у розроблених як за кордоном, так і в Україні стандартах з проектного управління. Згідно стандарту PMI (Project Management Institute) проект – це обмежена у часі спроба створити унікальний продукт, послугу, або будь-який результат [16, с. 5]. Тимчасовий характер проекту визначає те що він має початок та кінець. Кінцем проекту є момент, коли усі цілі проекту були виконані, або коли проект було ліквідовано бо якісь з цілей не були виконані або відмінені, або коли немає сенсу в подальшому існуванні проекту. Обмеженість проекту у часі не означає те, що він є короткостроковим, це також

не означає, що і результат (продукт або послугу) проекту має бути тимчасовим.

Кожен проект створює унікальний продукт, послугу, або інший результат. Деякі елементи можуть повторювати інші проекти, наявність повторів не змінює фундаментальну унікальність самої проектної роботи. Враховуючи унікальну природу проекту, для кожного створюваного проекту, послуги, або іншого результату можуть існувати різноманітні невизначеності [15]. Проектні завдання можуть бути новими або незвичними для проектною команди.

У межах даного дослідження *проектне управління трактується як застосування знань, навичок, інструментів і методів у процесі здійснення проектною діяльності з метою досягнення визначених вимог проекту*, що відповідає підходу Project Management Institute [16, с. 6-7]. Таке розуміння підкреслює комплексний і системний характер проектного управління, орієнтованого на досягнення конкретних результатів в умовах обмежених ресурсів і часових рамок. З огляду на це, доцільним є подальший розгляд особливостей проектного управління у контексті інженерії програмного забезпечення. Реалізація цього завдання передбачає аналіз підходів і методологій розробки програмного забезпечення як важливих складових проектного управління в даній галузі, оскільки саме вони визначають організацію процесів планування, виконання та контролю проектів, а також специфіку взаємодії учасників проектною діяльності.

Серед основних груп методологій розробки програмного забезпечення виокремлюють каскадні та гнучкі підходи, зокрема Waterfall і Agile. Для ілюстрації особливостей організації проектного управління доцільно звернутися до каскадною моделі, яка передбачає послідовне проходження взаємопов'язаних етапів, кожен з яких логічно впливає з попереднього [17]. Класична модель Waterfall охоплює п'ять основних фаз: збір вимог, проектування, реалізацію (виконання), перевірку та підтримку й обслуговування. На етапі збору вимог здійснюється систематизація, структуризація та аналіз усіх вимог до майбутнього програмного продукту, що формує основу для подальших процесів розробки. Фаза проектування включає дві взаємопов'язані підфази – створення логічного та фізичного дизайну [18]. Логічне

проектування базується на результатах аналізу вимог і передбачає розроблення абстрактної моделі системи, незалежної від конкретних програмних і апаратних засобів. Надалі отримані результати трансформуються у фізичний дизайн, який враховує нефункційні вимоги, а також наявні технічні обмеження та ресурси; на цьому етапі також може здійснюватися оцінювання проєкту та його окремих компонентів. Фаза реалізації передбачає безпосереднє втілення розроблених проєктних рішень у вигляді програмного коду відповідно до сформованих специфікацій. На етапі перевірки здійснюється комплексне тестування програмного продукту з метою встановлення його відповідності визначеним вимогам, після чого результати проєкту передаються замовнику. Завершальною є фаза підтримки та обслуговування, яка спрямована на забезпечення працездатності програмного продукту в процесі його експлуатації, зокрема шляхом усунення виявлених помилок і внесення необхідних змін. Таким чином, каскадна модель розробки відображає лінійну логіку реалізації проєкту та чітку структурованість етапів, що дозволяє розглядати її як одну з базових форм організації проєктного управління в інженерії програмного забезпечення.

Серед ключових переваг каскадної моделі Waterfall доцільно виокремити можливість виявлення помилок на ранніх етапах проєктування, зокрема на рівні дизайну та архітектури системи, ще до початку безпосередньої реалізації програмного продукту. Крім того, ваговою характеристикою цієї методології є формування значного обсягу структурованої та якісної документації, яка супроводжує всі етапи життєвого циклу проєкту та виступає важливим результатом проєктної діяльності. Організаційно більшість фаз, за винятком етапу збору вимог, реалізуються інженерно-технічними фахівцями, діяльність яких координується проєктним менеджером у межах визначених процесів проєктного управління. Такий підхід забезпечує чіткий розподіл ролей, відповідальності та підпорядкування, що є характерною ознакою формалізованих моделей управління проєктами. Порівняльний аналіз етапів моделі Waterfall із основними компонентами проєктного управління засвідчує їх значну змістову узгодженість,

оскільки більшість фаз безпосередньо корелюють із процесами планування, виконання та контролю проєкту. Водночас окремі складові, такі як управління ризиками, стратегічне планування чи управління ресурсами, виходять за межі власне процесу розробки програмного забезпечення та належать до ширшого контексту проєктного управління. З огляду на це, їх детальний розгляд у межах аналізу методологій розробки програмного забезпечення не є першочергово доцільним.

У структурі проєктного управління доцільно виокремити такий ключовий компонент, як врегулювання проєктних обмежень, зокрема бюджету, ресурсного забезпечення, ризиків, термінів виконання та показників якості. Саме збалансування цих параметрів значною мірою визначає ефективність реалізації проєкту та досягнення запланованих результатів. У межах каскадної моделі Waterfall зазначений компонент найбільшою мірою корелює з фазою проєктування, на якій здійснюється деталізація рішень, оцінювання обсягів робіт і формування базових характеристик майбутнього продукту [19]. Водночас, як зазначають А. Stellman та J. Greene, у практиці розробки програмного забезпечення проєкти часто ініціюються з метою оперативного розв'язання нагальних проблем. За таких умов стейкхолдери орієнтуються на отримання швидких результатів і чинять тиск на учасників проєкту щодо скорочення термінів його реалізації [20]. Це, своєю чергою, нерідко призводить до заниження оцінок трудомісткості та тривалості робіт ще на етапі проєктування, що може негативно впливати на якість кінцевого продукту та загальну ефективність управління проєктом. Таким чином, такі характеристики, як ургентність виконання завдань, обмежені часові рамки та орієнтація на швидке досягнення результату, доцільно розглядати як специфічні особливості проєктного управління в контексті інженерії програмного забезпечення. Вони зумовлюють необхідність гнучкого балансування між проєктними обмеженнями та підвищують вимоги до професійної підготовки фахівців у сфері проєктного управління.

Поряд із обмеженістю часових рамок, успішність реалізації проєкту значною

мірою залежить від рівня розуміння виконавцями доменної області, у межах якої здійснюється розробка [21; 22]. Зокрема, своєчасне та повноцінне виконання проєкту передбачає не лише наявність технічних компетентностей (володіння мовами програмування, технологічними стеком, методологіями та процесами розробки), але й достатній рівень предметного розуміння бізнес- або прикладної області проєкту. Недостатній рівень або відсутність такого розуміння може призводити до суттєвих похибок в оцінюванні обсягів робіт, складності завдань і ресурсних витрат, що, у свою чергу, негативно впливає на точність планування та збільшує ризики невиконання проєкту в заплановані строки. У крайніх випадках це може стати одним із чинників невдачі проєкту загалом.

У межах формування доказової бази щодо обґрунтування досліджуваної проблематики було здійснено аналіз відкритих вакансій на посади розробників програмного забезпечення. З метою забезпечення репрезентативності та об'єктивності результатів дослідження було обрано два провідні онлайн-платформи для пошуку роботи в ІТ-сфері: DOU (<https://dou.ua>) та Djinni (<https://djinni.co>), які характеризуються значним обсягом актуальних вакансій та широким охопленням роботодавців і кандидатів. Всього було проаналізовано 55 вакансій, з них 10 на стажера, або intern спеціаліста, 18 на молодшого, або junior спеціаліста, 10 на спеціаліста середнього рівня, або middle спеціаліста, 10 на старшого, або senior спеціаліста, та 7 на керівника команди, або lead спеціаліста. Отримані в результаті аналізу дані слугують основою для подальшого визначення та уточнення вимог до професійних компетентностей фахівців з інженерії програмного забезпечення (див. дод. Б). У межах дослідження кожна відібрана вакансія була проаналізована на предмет наявності вимог, що корелюють або повністю збігаються з компетентностями, визначеними у стандарті вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня, який передбачає формування 26 ключових компетентностей [22, с. 7-8]. Такий порівняльний аналіз дозволяє встановити ступінь відповідності між актуальними запитами ринку праці та нормативно

визначеними результатами навчання, а також виявити можливі розбіжності між ними. Це, у свою чергу, створює підґрунтя для подальшого вдосконалення освітніх програм і підвищення рівня підготовки майбутніх фахівців з інженерії програмного забезпечення відповідно до сучасних вимог професійного середовища. Результатом проведеного аналізу є формування ранжованого переліку компетентностей, розташованих за ступенем їх поширеності у вимогах роботодавців – від найбільш до найменш затребуваних. Для побудови зазначеного рейтингу було застосовано бально-оцінювальний підхід, що забезпечує кількісну інтерпретацію якісних даних. Згідно з обраною методикою, кожна компетентність оцінювалася за ступенем її відповідності вимогам, представленим у вакансіях. У разі повного збігу вимоги з визначеною компетентністю нараховувалося два бали; за часткової відповідності – один бал; за відсутності відповідності – нуль балів. Такий підхід дозволив забезпечити уніфіковане порівняння компетентностей та сформувані їх об’єктивне ранжування відповідно до частоти та ступеня релевантності у запитах стейкхолдерів (табл.1.1).

Таблиця 1.1

Матриця ранжування компетентностей на основі частоти та ступеня їх відповідності запитам стейкхолдерів

Position	K01	K02	K04	K05	K06	K07	K13	K14	K15	K16	K17	K18	K19	K20	K23	K24	K25	K26
Intern	8	7	15	9	2	5	0	2	2	3	2	0	3	2	3	2	0	3
Junior	1	11	20	5	3	12	3	8	7	7	8	1	2	4	6	4	0	3
Mid	2	4	13	1	0	12	5	12	11	8	13	5	5	2	10	6	6	0
Senior	3	8	11	3	3	9	9	15	11	6	8	4	7	2	7	9	1	0
Lead	2	4	10	2	2	13	8	11	10	9	6	2	9	2	8	5	3	2

Для позиції intern, або стажера результати виглядають наступним чином: K04 – 15; K05 – 9; K01 – 8; K02 – 7; K07 – 5; K16, K19, K23, K26 – 3; K06, K14, K15, K17, K20, K24 – 2; K03, K08, K09, K10, K11, K12, K13, K18, K21, K22, K25 – 0.

Результати ранжування свідчать, що трьома найбільш значущими

компетентностями для стажерів з точки зору роботодавців є: «здатність спілкуватися іноземною мовою як усно, так і письмово», «здатність вчитися і оволодівати сучасними знаннями» та «здатність до абстрактного мислення, аналізу та синтезу». Натомість до групи найменш затребуваних належать «здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки)» та «здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення». Водночас компетентність «здатність працювати в команді» посідає п'яте місце серед усіх проаналізованих позицій, що свідчить про її достатньо високий рівень значущості та підтверджує очікування роботодавців щодо наявності у фахівців навичок командної взаємодії.

Для позиції junior, або молодшого спеціаліста, результати виглядають наступним чином: K04 – 20; K07 – 12; K02 – 11; K14, K17 – 8; K15, K16 – 7; K23 – 6; K05 – 5; K20, K24 – 4; K06, K13, K26 – 3; K19 – 2; K01, K18 – 1; K03, K08, K09, K10, K11, K12, K21, K22, K25 – 0.

Для позиції молодшого спеціаліста (junior) трьома найбільш значущими компетентностями за результатами ранжування є: «здатність спілкуватися іноземною мовою як усно, так і письмово», «здатність працювати в команді» та «здатність застосовувати знання у практичних ситуаціях». Отримані результати свідчать про підвищений рівень зацікавленості роботодавців у компетентності «здатність працювати в команді». Це дозволяє зробити висновок, що роботодавці очікують від молодших спеціалістів не лише базових технічних знань, але й сформованого розуміння специфіки роботи в проєктній команді з розробки програмного забезпечення, включно з характерними для неї процесами, ролями та формами взаємодії.

Для позиції middle, або спеціаліста середнього рівня, результати виглядають наступним чином: K04, K17 – 13; K07, K14 – 12; K15 – 11; K23 – 10; K16 – 8; K24, K25 – 6; K13, K18, K19 – 5; K02 – 4; K01, K20 – 2; K05 – 1; K03, K06, K08, K09, K10, K11, K12, K21, K22, K26 – 0.

Лідуючими компетентностями для позиції розробника є: «здатність спілкуватися іноземною мовою як усно, так і письмово», «здатність працювати в команді», «здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу», «здатність брати участь у проєктуванні програмного забезпечення, включаючи проведення моделювання (формального опису) його структури, поведінки та процесів функціонування», а також «здатність розробляти архітектури, модулі та компоненти програмних систем».

Отримані результати свідчать, що поряд із такими базовими компетентностями, як володіння іноземною мовою та здатність до командної роботи, роботодавці висувають також більш специфічні технічні вимоги, зокрема щодо проєктування архітектури програмного забезпечення та дотримання встановлених специфікацій і стандартів у процесі його розробки. Це відображає підвищення рівня професійних очікувань у порівнянні з позицією молодшого розробника. Компетентність «здатність працювати в команді» зберігає стабільно високий рівень значущості, оскільки її ранг повністю збігається з відповідним показником для позиції молодшого спеціаліста. Це може свідчити про те, що роботодавці очікують від кандидатів щонайменше незмінного, а в окремих випадках – і більш високого рівня сформованості навичок командної взаємодії, що є критично важливим для ефективної участі в проєктній діяльності з розробки програмного забезпечення.

Для позиції senior, або старшого спеціаліста, результати виглядають наступним чином: K14 – 15; K04, K15 – 11; K07, K13, K24 – 9; K02, K17 – 8; K19, K23 – 7; K16 – 6; K18 – 4; K01, K05, K06 – 3; K20 – 2; K25 – 1; K03, K08, K09, K10, K11, K12, K21, K22, K26 – 0.

Для позиції старшого програміста роботодавці визначають як найбільш значущі такі компетентності: «здатність брати участь у проєктуванні програмного забезпечення, включаючи проведення моделювання (формального опису) його структури, поведінки та процесів функціонування», «здатність спілкуватися

іноземною мовою як усно, так і письмово», «здатність розробляти архітектури, модулі та компоненти програмних систем», «здатність працювати в команді», «здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення», а також «здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення».

Результати ранжування засвідчують незначне зниження рівня вимог до компетентності «здатність працювати в команді», що може бути зумовлено тим, що, з точки зору роботодавців, старший програміст уже має володіти сформованими як теоретичними, так і практичними навичками командної взаємодії. Це свідчить про перехід акценту з базових соціально-комунікативних компетентностей на більш складні професійні та інженерні аспекти діяльності.

Для позиції lead, або керівника команди результати виглядають наступним чином: K07 – 13; K14 – 11; K04, K15 – 10; K16, K19 – 9; K13, K23 – 8; K17 – 6; K24 – 5; K02 – 4; K25 – 3; K01, K05, K06, K18, K20, K26 – 2; K03, K08, K09, K10, K11, K12, K21, K22 – 0.

Для позиції провідного програміста серед найбільш значущих компетентностей виокремлюються такі: «здатність працювати в команді», «здатність брати участь у проєктуванні програмного забезпечення, включаючи проведення моделювання (формального опису) його структури, поведінки та процесів функціонування», «здатність спілкуватися іноземною мовою як усно, так і письмово» та «здатність розробляти архітектури, модулі та компоненти програмних систем».

Високий рівень значущості компетентності «здатність працювати в команді» свідчить про те, що професійна діяльність провідного програміста значною мірою інтегрує елементи проєктного управління. Зокрема, це проявляється у виконанні функцій координації роботи команди, організації взаємодії між її учасниками, а також у частковому здійсненні управлінських завдань, пов'язаних із керуванням підлеглими та забезпеченням ефективної реалізації проєктної діяльності.

Узагальнені результати аналізу свідчать про те, що найбільш поширеними є

наступні компетентності: K04. Здатність спілкуватися іноземною мовою як усно, так і письмово; K07. Здатність працювати в команді; K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування; K15. Здатність розробляти архітектури, модулі та компоненти програмних систем. Проведене моніторингове дослідження виявило розбіжності між підготовкою майбутніх бакалаврів з інженерії програмного забезпечення та вимогами стейкхолдерів. Найбільша узгодженість вбачається між компетентностями та вакантними вимогами до кандидатів, що за змістом пов'язані з елементами проєктного управління, зокрема здатністю працювати в команді; здатністю брати участь у проектуванні програмного забезпечення. Це зумовлює потребу забезпечення високого рівня готовності майбутніх фахівців з інженерії програмного забезпечення до проєктного управління як обов'язкового компонента їхнього професіоналізму.

Отже, майбутні бакалаври з інженерії програмного забезпечення повинні володіти необхідним комплексом знань і навичок, що забезпечують ефективну орієнтацію у масштабних проєктах із залученням численних команд та різноманітних ролей. Така вимога безпосередньо узгоджується із положеннями Закону України «Про освіту» (2017) [23] та Закону України «Про вищу освіту» (2014) [24], у яких закріплено компетентнісний підхід до підготовки фахівців, орієнтований на формування здатності застосовувати знання у професійній діяльності. Вони мають розуміти сучасні підходи до розробки програмного забезпечення, особливості їх застосування, а також усвідомлювати потенційні недоліки кожного підходу, пов'язані з цим ризики та способи їх мінімізації. Зазначені вимоги конкретизуються у наказах Міністерства освіти і науки України щодо затвердження стандартів вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» (2018, 2020) [22; 25], які передбачають формування фахових компетентностей, зокрема здатності застосовувати сучасні методології розробки, управляти якістю програмного забезпечення та оцінювати ризики. Саме

сформованість зазначених компетентностей є ключовою вимогою роботодавців до інженерів, які беруть участь у реалізації проєктів. Це положення корелює із Законом України «Про державну підтримку розвитку індустрії програмної продукції» (2013), який підкреслює необхідність підготовки конкурентоспроможних ІТ-фахівців, здатних ефективно працювати в умовах сучасного ринку та відповідати потребам ІТ-індустрії.

Ефективна реалізація визначених компетентностей у практичній діяльності потребує орієнтації не лише на законодавчі вимоги, а й на визнані у професійному середовищі підходи та інструменти організації роботи. У цьому контексті особливого значення набуває звернення до сучасних стандартів з управління проєктами, які регламентують принципи планування, виконання та контролю проєктної діяльності, а також забезпечують узгодженість дій команд і підвищення якості кінцевого продукту. До таких стандартів відносяться: The Standard for Project Management [26]; The Standard for Portfolio Management [27]; Practice Standard for Project Risk Management [28]; Practice Standard for Earned Value Management [29]; Practice Standard for Project Configuration Management [30]; Practice Standard for Work Breakdown Structures [31]; Practice Standard for Scheduling [32]; Practice Standard for Project Estimating [33]; Project Manager Competency Development Framework [34]; Construction Extension to the PMBOK® Guide, Government Extension to the PMBOK® Guide (A Guide to the Project Management Body of Knowledge) [35]; чотирирівнева система сертифікації IPMA [36], стандарт P2M (Project and Program Management for Enterprise Innovation) [37], ISO 9000 (9001) [38]; ISO 10006 (Quality management systems – Guidelines for quality management in projects) [39]; IPMA (Міжнародної асоціації управління проєктами) National Competence Baseline [40].

Так у Software Extension to the PMBOK® Guide визначено причини, через які управління проєктами у сфері розробки програмного забезпечення може становити складність для керівника проєкту [41, с. 7]. Зокрема, управління проєктами у сфері розробки програмного забезпечення є суттєво ускладненим через сукупність взаємопов'язаних факторів. До них належать нематеріальна природа програмного

продукту, його висока складність та пластичність, а також значна кількість логічних зв'язків і варіантів взаємодії між компонентами системи. Важливим ускладнювальним чинником є динамічність вимог, які змінюються протягом усього життєвого циклу проєкту під впливом накопичення знань, бізнес-процесів організації та взаємодії із зацікавленими сторонами. Це, у свою чергу, ускладнює початкове планування, оцінювання ресурсів і формування точних прогнозів щодо результатів проєкту. Специфіка розробки програмного забезпечення передбачає високий рівень залежності від людського інтелектуального капіталу, інноваційний характер діяльності, а також значну варіативність продуктивності розробників. Сукупність цих чинників зумовлює підвищений рівень ризику та невизначеності проєктів. Додатковими ускладненнями виступають проблеми комунікації та координації в командах, складність інтеграції з продуктами різних постачальників, постійні зміни технологічних платформ, а також необхідність забезпечення безпеки, якості та сумісності програмного забезпечення в умовах обмежених можливостей його повного тестування.

Таким чином, управління проєктами у сфері розробки програмного забезпечення вимагає адаптивних підходів, високого рівня професійної компетентності керівників та використання сучасних методів управління, здатних враховувати складність, динамічність і невизначеність цієї предметної області.

Враховуючи зазначені особливості та складність управління проєктами у сфері розробки програмного забезпечення, особливої актуальності набуває питання формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до здійснення проєктного управління. Така готовність передбачає не лише засвоєння теоретичних знань з методологій управління проєктами, а й розвиток практичних навичок роботи в умовах невизначеності, змінності вимог та високого рівня ризиків. Майбутні фахівці повинні бути здатними ефективно працювати в команді, забезпечувати належний рівень комунікації із зацікавленими сторонами, а також застосовувати сучасні інструменти планування, оцінювання та контролю проєктів. Важливим компонентом такої підготовки є формування

системного мислення, здатності до аналізу складних взаємозв'язків у програмних системах і прийняття обґрунтованих управлінських рішень. Тобто, підготовка бакалаврів з інженерії програмного забезпечення має бути спрямована на інтеграцію знань з програмної інженерії та проєктного менеджменту, що дозволить забезпечити їхню професійну готовність до ефективної роботи в умовах сучасних ІТ-проєктів, які характеризуються високою складністю, динамічністю та інноваційністю.

1.2 Структура готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

У межах нашого дослідження вважаємо за доцільне здійснити теоретичне обґрунтування та визначити структуру готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як кінцевого результату експериментальної підготовки, а також охарактеризувати критерії, показники й рівні її сформованості.

Сформована готовність є однією з ключових передумов успішної професійної діяльності фахівця, оскільки забезпечує його швидку адаптацію до умов роботи, ефективне виконання професійних обов'язків, а також сприяє розвитку навичок самоконтролю, саморегуляції та подальшої професійної самореалізації. Вона виступає інтегральною характеристикою, що визначає здатність особистості діяти впевнено та результативно в умовах професійних викликів [42].

У довідковій та енциклопедичній літературі поняття «готовність до діяльності» визначається як: бажання особистості зробити що-небудь [43]; бажання творити що-небудь [44]; установка на певну поведінку [45], уміння виконувати певні операції і творчо підходити до їх виконання [46]; мобілізація психологічної та психофізіологічної системи до виконання певної діяльності [47], що дозволяє успішно ввійти у професійне середовище, швидко розвиватися в професії [48].

У психологічному аспекті «готовність» трактується як інтегративне утворення, що відображає емоційно-когнітивну та вольову мобілізацію суб'єкта в

момент його включення до діяльності певної спрямованості. Вона формується на основі індивідуального досвіду, що ґрунтується на становленні позитивного ставлення до обраного виду діяльності, усвідомленні мотивів і потреб, а також осмисленні її предмета та способів взаємодії з ним. За таких умов емоційні, вольові та інтелектуальні прояви поведінки особистості виступають конкретним виявом готовності на феноменологічному рівні [49].

Як зазначають дослідники, поняття «готовність до діяльності» розглядається як важлива умова її успішного здійснення, форма вибіркової активності, що орієнтує особистість на майбутню діяльність, виступаючи водночас її передумовою та регулятором [50], а також постає детермінантою рушійних сил діяльності й внутрішнім спонукальним чинником [51]. Як специфічний стан свідомості, «готовність до діяльності» характеризується комплексом взаємопов'язаних компонентів. До них належить система знань про навколишній світ, що формується в процесі пізнавальної діяльності; самосвідомість, яка охоплює усвідомлення, самооцінювання та рефлексію власних дій і поведінки; здатність до цілеспрямованої діяльності, що ґрунтується на постановці мети відповідно до мотивів і потреб, прийнятті вольових рішень та коригуванні дій у процесі їх виконання. Важливим складником є також емоційно-почуттєва сфера, яка відображає ставлення особистості до навколишнього світу та інших людей і проявляється у відповідних емоційних оцінках.

Готовність до трудової діяльності трактується як інтегративне особистісне утворення, що характеризується стійкою спрямованістю на професійну діяльність у певній галузі, сформованістю необхідних знань, умінь і навичок, сукупністю індивідуально-типологічних і соціально-психологічних характеристик, які зумовлюють високу ефективність професійного функціонування. Вона виявляється як цілісне вираження особистості, що охоплює систему переконань, поглядів, ставлень, мотивів, емоційно-чуттєвих, вольових та інтелектуальних якостей, установок і набутого досвіду діяльності [48].

Залежно від предмета дослідження підходи науковців до трактування поняття «готовність до професійної діяльності» відрізняються. Так, І. Коваль

розглядає його, з одного боку, як наявність професійно значущих якостей, а також сформованість знань, умінь і навичок, що забезпечують успішність у обраній сфері діяльності; з іншого – як стан концентрації потенціалу особистості, що передуює виконанню дій, внутрішню налаштованість на професійну діяльність і мобілізацію психофізіологічних систем для її ефективної реалізації [52]; Р. Горбатюк – як інтегративну властивість особистості, систему якостей, які забезпечують здатність до впровадження принципів інженерно-педагогічної культури в освітньому процесі та на виробництві [53]; поліструктурне утворення у складі операційно-діяльнісного, когнітивного, мотиваційного, емоційно-вольового і суб'єктного компонентів, що визначає професійно важливі якості майбутніх інженерів-педагогів і зумовлює рівень їх відповідності вимогам, що висувуються до фахівців у педагогічній діяльності [54]; двосторонній процес, який, з одного боку, пов'язаний із напрацюванням необхідних якостей творчої особистості майбутнього фахівця, а з іншого – із формуванням у нього досвіду творчої діяльності [55]; цілісне, стійке, інтегральне утворення особистості, що підпорядковується цілеспрямованому формуванню сукупності проєктувальних знань та вмінь і проявляється у взаємопов'язаних та взаємозумовлених компонентах [56]; результат процесу професійної підготовки фахівця, що виявляється в його спроможності застосовувати комунікативні технології під час організації освітньої діяльності учнів [57]; стійкий стан налаштованості на успішне розв'язання складних спеціалізованих задач і практичних проблем в професійній освіті та галузі економіки, яка ґрунтується на мобілізації потенційних можливостей (мотиваційних, операційно-діяльнісних, інтелектуальних, емоційно-вольових) і досвіду, з метою отримання очікуваного результату та подальшого особистого розвитку в змінних умовах [58].

У наукових дослідженнях спостерігаються різні підходи до визначення структурних компонентів готовності до діяльності. Готовність до будь-якого виду діяльності передусім передбачає наявність внутрішньої мотивації та прагнення до її здійснення, а також сформованість і здатність ефективно застосовувати необхідні для відповідної галузі знання й уміння, що в сукупності визначає рівень

компетентності особистості. Підкреслюється значущість таких компонентів готовності до професійної діяльності, як позитивне ставлення до праці загалом і до обраної професії зокрема; сформованість рис характеру, темпераментних властивостей, особистісної спрямованості та здібностей; наявність ґрунтовних знань, засвоєних норм, а також сформованих умінь і навичок, гуманістичних цінностей і смислів, що відповідають вимогам діяльності. Важливу роль відіграють також професійно значущі психічні процеси й властивості, зокрема особливості сприйняття, уявлення, уваги, розуміння та мислення, а також специфіка перебігу емоційних, мотиваційних і вольових процесів [59]. Поділяємо позицію науковців, які вважають, що сутність готовності до професійної діяльності розкривається через діалектичну єдність її структурних компонентів, властивостей, зв'язків і відношень. Основними інваріантними та іманентними складовими готовності дослідники вважають взаємозумовлене поєднання особистісного та процесуального компонентів.

Проведений аналіз наукових підходів до трактування поняття «готовність до діяльності» та «готовність до професійної діяльності» дає підстави розглядати її як складне, інтегративне та поліструктурне утворення особистості, що формується в процесі професійної підготовки та забезпечує ефективність майбутньої діяльності. Готовність виступає як передумова, регулятор і внутрішній механізм активізації діяльності, поєднуючи в собі систему знань, умінь і навичок, досвід діяльності, а також мотиваційні, емоційно-вольові й інтелектуальні характеристики особистості. Узагальнення різних наукових підходів дозволяє стверджувати, що структура готовності має подвійний характер. З одного боку, вона ґрунтується на функціональній основі, яка охоплює професійні знання, вміння, навички, способи діяльності та досвід їх застосування. З іншого боку, готовність визначається сукупністю персональних якостей особистості, зокрема мотиваційних установок, ціннісних орієнтацій, емоційно-вольових і когнітивних характеристик, що забезпечують мобілізацію внутрішніх ресурсів і спрямованість на успішне виконання професійних завдань.

Таким чином, готовність майбутніх бакалаврів з інженерії програмного

забезпечення до проєктного управління доцільно розглядати як інтеграцію функціональної складової та індивідуально-особистісних характеристик, що в сукупності забезпечують їхню здатність ефективно діяти в умовах складних, динамічних і невизначених ІТ-проєктів.

Згідно з популярним довідником Project Management Body of Knowledge (PMBOK) [16] можна виділити наступні *функціональні компоненти проєктного управління*: планування та ретельне визначення пріоритетів (проєктних та персональних); максимізація бізнес-цінності проєкту за рахунок правильно створеної стратегії його реалізації; співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту; використання як традиційних так і agile інструментів, техніки та методів для кожного проєкту; пояснення важливих бізнес-аспектів проєкту іншим; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками; увага до критично важливих технічних елементів проєктного управління.

Планування та ретельне визначення пріоритетів (проєктних та персональних). Навички якісного планування та чіткого визначення пріоритетів (як проєктних, так і персональних) безпосередньо пов'язані з обраною методологією, що використовується у проєкті. Наприклад, для успішної реалізації проєкту за методологією Waterfall необхідно розробити детальний план і визначити пріоритети для кожної підфази ще на етапі проєктування, тобто на початку реалізації проєкту.

Це вимагає від майбутніх бакалаврів з інженерії програмного забезпечення відповідних умінь та навичок. Згідно результатів аналізу Професійного стандарту фахівця з розробки програмного забезпечення [10] виявлено наступні навички: відстежувати системні вимоги; визначати здійсненність програмних складових частин відповідно до системних вимог; визначати здійсненність функціонування і супроводу; вибирати моделі життєвого циклу згідно з проєктом; визначати необхідні мови програмування та технології в залежності від сфери застосування, розмірів та складності проєкту.

З позиції трудових функцій ці навички відповідають таким видам діяльності, як «визначення стратегії реалізації програмних засобів відповідно до обраної моделі життєвого циклу, сфери застосування, розміру та складності проєкту» та «оцінювання архітектури системи».

Максимізація бізнес-цінності проєкту за рахунок правильно створеної стратегії реалізації проєкту. Для максимізації бізнес-цінності проєкту майбутні бакалаври з інженерії програмного забезпечення повинні розуміти предметну область проєкту та вплив його артефактів на бізнес замовника. Аналіз функціональних і нефункціональних вимог дає змогу виявити потенційні напрями вдосконалення, що сприяють підвищенню бізнес-цінності проєкту та формуванню обґрунтованої стратегії його реалізації. Важливим чинником також є коректний вибір артефактів (вихідних даних) проєкту, які безпосередньо впливають на досягнення очікуваних результатів.

Для ефективного виконання цієї функції інженер повинен уміти: вибирати моделі життєвого циклу згідно з проєктом; визначати необхідні мови програмування та технології в залежності від сфери застосування, розмірів та складності проєкту; застосовувати на практиці ефективні підходи щодо проєктування програмного забезпечення; вибирати вихідні дані для проєктування, керуючись формальними методами опису вимог та моделювання; знати та вміти застосовувати методи та засоби управління проєктами; здійснювати комунікацію із правовласниками, замовниками та користувачами системи щодо вимог до програмних засобів.

Якщо розглянути ці навички у контексті компетентностей та трудових функцій, то отримаємо наступне: визначення стратегії реалізації програмних засобів відповідно до вибраної моделі життєвого циклу, сфери застосування, розмірів і складності проєкту; взаємодія з правовласниками, замовниками та користувачами системи щодо вимог до програмних засобів; здатність працювати в команді; здатність розробляти архітектуру, модулі та компоненти програмних систем; здатність брати участь у проєктуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури,

поведінки та процесів функціонування.

Співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту. У рамках розробки стратегії реалізації проєкту, майбутньому бакалавру з інженерії програмного забезпечення необхідно мати наступні навички та вміння: уміння комунікувати із замовником або спонсором проєкту, навички правильного визначення артефактів (вихідні дані) проєкту, уміння обирати життєвий цикл проєкту згідно з цілями.

У результаті аналізу стандарту вищої освіти зі спеціальності 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня вищої освіти [22] було виокремлено такі вміння: вибирати вихідні дані для проєктування, керуючись формальними методами опису вимог та моделювання; застосовувати на практиці ефективні підходи щодо проєктування програмного забезпечення; вибирати моделі життєвого циклу згідно з проєктом; визначати необхідні мови програмування та технології в залежності від сфери застосування, розмірів та складності проєкту; здійснювати комунікацію із правовласниками, замовниками та користувачами системи щодо вимог до програмних засобів; мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації; мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.

На підставі отриманих результатів виділено такі трудові функції: визначення стратегії реалізації програмних засобів відповідно до вибраної моделі життєвого циклу, сфери застосування, розмірів і складності проєкту; оцінювання архітектури системи.

З точки зору стандарту вищої освіти, можна виділити наступні компетентності, які відповідають цій функції: здатність брати участь у проєктуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування; здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення; здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі

відповідних моделей і підходів розробки програмного забезпечення; здатність працювати в команді; здатність спілкуватися іноземною мовою як усно, так і письмово; здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.

Використання як традиційних, так і Agile інструментів, техніки та методів для кожного проєкту. Залежно від обраної методології реалізації проєкту, менеджер проєкту може забезпечити різний рівень його гнучкості. Зокрема, використання agile-методологій дає змогу досягти високої адаптивності, за якої замовник має можливість змінювати базові вимоги навіть у процесі реалізації проєкту командою розроблення. Це зумовлює додаткові складнощі та може призводити до зростання вартості проєкту, однак водночас забезпечує можливість оперативного внесення змін, у тому числі до ключових функціональних характеристик. Каскадна модель (Waterfall) передбачає більш ретельне планування всіх етапів проєкту (реалізації, тестування, супроводу) ще на стадії проєктування. У результаті внесення змін під час реалізації є значно дорожчим, проте забезпечується вища швидкість виконання робіт за рахунок оптимального розподілу навантаження між командами розроблення та тестування.

Для коректного вибору методологій, майбутні бакалаври з інженерії програмного забезпечення мають знати, які методи існують, їх сильні та слабкі сторони, підходи до вибору методологій, та ін. В результаті аналізу стандарту вищої освіти та професійного стандарту, було обрано наступні навчання та вміння: мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації; вміти проводити розрахунок економічної ефективності програмних систем; працювати в групі розробки та взаємодіяти з учасниками групи; робота з системами управління задачами; знати та вміти застосовувати методи та засоби управління проєктами.

Як результат, можна виділити наступні трудові функції: взаємодія з командою розробки проєкту при виконанні проєкту.

З точки зору стандарту вищої освіти, можна виділити наступні

компетентності, які відповідають цій функції проєктного менеджера: здатність працювати в команді; здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення; здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.

Пояснення важливих бізнес-аспектів проєкту іншим. Кожен ІТ-проєкт має в своїй основі створення, модифікацію та/або підтримку рішення, яке використовується замовником як частина власного бізнесу. В такому випадку, кожен майбутній бакалавр з інженерії програмного забезпечення має розуміти, як замовник використовує власну систему, та які бізнес-аспекти вона спрощує, або повністю вирішує. Так само, не тільки розуміти, а й поширювати ці знання з усією командою.

В результаті аналізу стандарту вищої освіти та професійного стандарту, було обрано наступні навички та вміння: мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації; вміти проводити розрахунок економічної ефективності програмних систем; здійснювати комунікацію із правовласниками, замовниками та користувачами системи щодо вимог до програмних засобів; проводити передпроєктне обстеження предметної області, системний аналіз об'єкта проєктування.

З точки зору стандарту вищої освіти, можна виділити наступні компетентності, які відповідають цій функції проєктного менеджера: здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення; здатність спілкуватися державною мовою як усно, так і письмово; здатність спілкуватися іноземною мовою як усно, так і письмово; здатність працювати в команді; здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.

Керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками. Для прикладу, фаза дизайну у методології

Waterfall складається зі збору вимог до програмного забезпечення, створення архітектури та прорахування оцінки вартості програмного забезпечення [18]. Зазвичай, цим займаються інженери, а робота менеджера проєкту зводиться до організації та контролю над роботою інженерів з наступною пост-обробкою отриманих даних. Таким чином, інженери напряму керують частиною ризиків (технічні ризики) та вартістю проєкту (наприклад, складність обраної архітектури, або композиція команди розробки) під час цієї фази.

Освітньо-професійна програма має відповідні результати навчання «знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення» та «застосовувати методи та засоби управління проєктами», та відповідну компетентність «здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення».

Професійний стандарт фахівця з розробки програмного забезпечення також має відповідний навичок «вибирати моделі життєвого циклу згідно з проєктом; визначати необхідні мови програмування та технології в залежності від сфери застосування, розмірів та складності проєкту» та трудова функція «визначення стратегії реалізації програмних засобів відповідно до вибраної моделі життєвого циклу, сфери застосування, розмірів і складності проєкту».

Отже, функціональна основа готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління формується як цілісна система взаємопов'язаних умінь, навичок і компетентностей, що забезпечують здатність ефективно реалізовувати ключові функції управління проєктами відповідно до сучасних стандартів. Вона охоплює здатність до стратегічного планування та визначення пріоритетів, орієнтацію на максимізацію бізнес-цінності проєкту, ефективну комунікацію та взаємодію із зацікавленими сторонами, обґрунтований вибір і застосування методологій розроблення, а також управління основними параметрами проєкту – термінами, ресурсами, вартістю та ризиками.

Сформована функціональна основа передбачає інтеграцію теоретичних знань про життєвий цикл програмного забезпечення, методи й засоби проєктного

управління з практичними вміннями їх застосування у реальних умовах, з урахуванням специфіки проєкту, його складності та предметної області. Вона також включає здатність аналізувати вимоги, приймати обґрунтовані рішення щодо архітектури системи, організувати командну взаємодію та забезпечувати узгодженість технічних і бізнес-цілей проєкту. Функціональна основа готовності виступає ключовим компонентом професійної підготовки майбутніх фахівців, оскільки саме вона забезпечує їхню здатність до практичної реалізації проєктного управління в умовах динамічних, складних і невизначених середовищ розроблення програмного забезпечення.

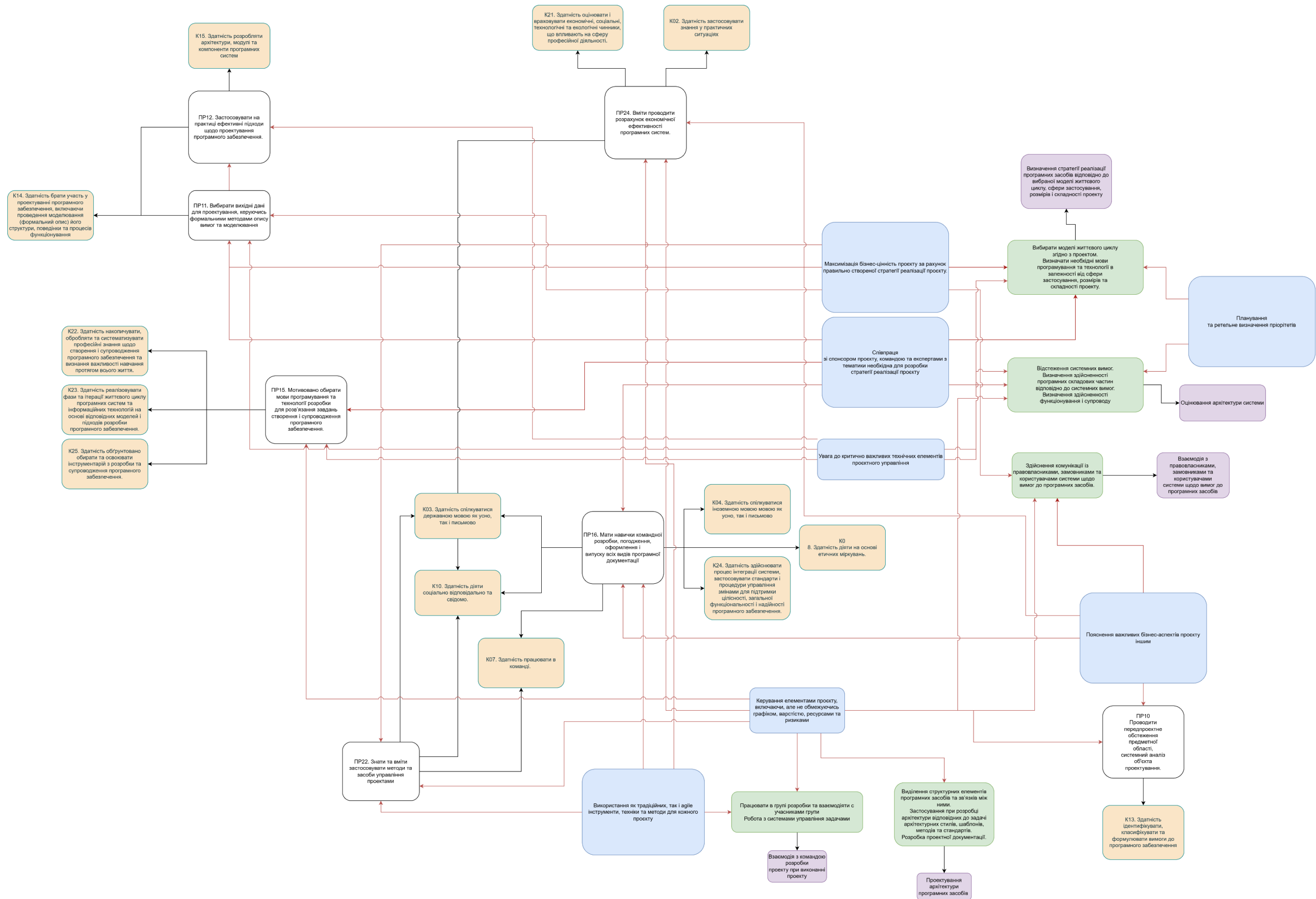


Рис. 1.1 Узгодженість між компетентностями, програмними результатами навчання бакалаврів з інженерії програмного забезпечення та функціями проектного управління

Сформованість функціональної основи готовності не є достатньою умовою ефективного здійснення проєктного управління. Важливу роль відіграють також персональні якості майбутнього фахівця, які забезпечують здатність до реалізації набутих знань і вмінь у реальних професійних ситуаціях.

Персональні якості можна описати за допомогою моделі «Великої п'ятірки» [60]. Прикладом слугує дослідження персональних якостей проєктних менеджерів для публічних та приватних секторів [61]. В рамках дослідження було опитано 244 проєктних керівників у приватних та публічних секторах та у різних сферах бізнесу, таких як інформаційні технології (61%), інженерія (23%), адміністрація (11%) та інші (5%). Опитування проводилось згідно п'ятибальної шкали типу Лайкерта, де керівники мали поставити оцінки наступним ствердженням: Я організований, Я відкритий до нового, Я емоційно стабільний, Я екстравертний, Я привітний. Результати дослідження показали наступну градацію оцінок: відкритість досвіду (4,448), невротизм або емоційна стабільність (4,290), екстраверсія (4,217), доброзичливість (4,217), доброзичливість (4,010). Ці результати напряду свідчать про те що, роль проєктного керівника дуже залежна від відкритості досвіду та емоційної стабільності. Звісно, інші показники мають достатньо високу оцінку та також впливають на персональний портрет керівника проєктів.

Для кращого запам'ятовування в англійській літературі п'ять рис складають в акронім «OCEAN»: Openness – відкритість досвіду; Conscientiousness – сумлінність; Extraversion – екстраверсія; Agreeableness – доброзичливість; Neuroticism – невротизм. Окремі науковці стверджують: «Те, як невизначеність сприймається керівниками проєктів, залежить від особистих навичок, інтуїції та суджень» [62, с. 77], що також підкреслює важливість особистісних складових на рівні з професійними навичками. Згідно отриманих даних, у проєктних менеджерів з великим досвідом найбільш затребуваними є такі якості, як відкритість до нового та емоційна стабільність [61]. З іншого боку, дослідження з перевірки персональних якостей у інженерів програмного забезпечення показали, що

інженери, які займають лідерські позиції (Team Lead, Tech Lead, та інші), або менеджерів мають вищі рівні екстраверсії та conscientiousness [63, с. 17]. Деякі дослідження показують великий вплив поведінкових компетенцій проектного управління на роботу організацій (компаній) [64] та стверджують про можливе підвищення продуктивності, якщо працівники будуть більш обізнані в сфері проектного управління. Інші дослідження ставлять за мету підвищити рівень розвитку всіх софт скілів у майбутніх бакалаврів і, як результат, рівень знань та вмінь [65]. Однак, ці дослідження не відповідають на питання, чи впливають персональні якості на рівень знань та вмінь цих працівників.

Таким чином, виникає необхідність дослідження рівня практичних умінь майбутніх бакалаврів з інженерії програмного забезпечення щодо проектного управління з одного боку та наявністю виражених окремих ознак персональних якостей згідно моделі «Великої п'ятірки». Щоб досягти цього, ми намагалися відповісти на такі запитання дослідження: Наскільки виражені персональні якості «Великої п'ятірки» у майбутніх бакалаврів з інженерії програмного забезпечення? Чи можуть майбутні бакалаври з інженерії програмного забезпечення знайти управлінське рішення в умовних виробничих ситуаціях? Чи існує кореляція між станом персональних якостей майбутніх бакалаврів з інженерії програмного забезпечення та кількістю вірно вирішених завдань? Чи студенти спеціальності «інженерія програмного забезпечення» володіють кращим рівнем знань з проектного управління у порівнянні з іншими спеціальностями?

У структурі особистісного компонента готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління *відкритість досвіду* визначається як індивідуально-психологічна властивість, що характеризує здатність позитивно сприймати нові підходи, технології та управлінські практики, а також бути відкритим до різноманітного професійного досвіду. Вона виявляється у допитливості, готовності до експериментування з інструментами та методологіями проектного управління, сприйнятті інновацій у розробленні програмного забезпечення, а також у прагненні до постійного професійного

розвитку. *Основними показниками є:* допитливість і інтерес до нових ідей; гнучкість та адаптивність. Допитливість та інтерес до нових ідей є складовою комунікативної компетентності у взаємодії із замовниками та користувачами системи, оскільки вони сприяють кращому розумінню предметної області системи або бізнесу замовника. У свою чергу це забезпечує глибше усвідомлення проблем і вимог [66, с. 17]. Гнучкість та адаптивність сприяють відкритості мислення, сприйнятливості до нової інформації та ідей, готовності розглядати альтернативні точки зору, а також здатності швидко пристосовуватися до змін. Однією з ключових властивостей програмного проєкту при використанні Agile-методологій є готовність до постійних змін. Дана методологія забезпечує контрольоване внесення змін до коду проєкту та вимог. Ефективна реалізація змін залежить не лише від обраного підходу, а й від готовності всієї команди до адаптації, тому гнучкість та адаптивність як особистісні якості керівника проєкту є особливо важливими.

Сумлінність свідчить про схильність до організованості, відповідальності та надійності у виконанні професійних завдань. Проявляється у самодисципліні, дотриманні встановлених правил і планів, орієнтації на досягнення результату, а також у перевазі структурованої та спланованої діяльності над спонтанними діями. *Основними показниками є:* організованість, увага до деталей, відповідальність, довгострокове планування. *Організованість* є важливою особистісною характеристикою, що передбачає здатність ефективно впорядковувати як власну діяльність, так і професійне середовище. Здобувачі освіти з високим рівнем сумлінності, як правило, характеризуються структурованим підходом до виконання завдань, а також схильністю до створення систем і процедур, що забезпечують результативність діяльності. Динамічний характер програмного проєкту часто зумовлює формування неструктурованого середовища, що проявляється у недостатній впорядкованості проєктної документації, обмеженій простежуваності проєктних процесів та інших організаційних недоліках. За таких умов керівник проєкту з високим рівнем організованості здатний мінімізувати прояви хаотичності та забезпечити впорядкованість проєктної діяльності. *Увага до*

деталей – це показник, що характеризує здатність майбутніх бакалаврів з інженерії програмного забезпечення зосереджуватися на суттєвих елементах діяльності, ретельно опрацьовувати інформацію та забезпечувати точність виконання завдань. *Відповідальність* передбачає усвідомлення наслідків власних дій і прийнятих рішень. Здобувачі освіти з високим рівнем відповідальності, як правило, не схильні перекладати провину на інших, здатні визнавати власні помилки та готові нести за них відповідальність. Динамічний характер проєктної діяльності та часті зміни короткострокових вимог зумовлюють необхідність постійної реорганізації внутрішніх процесів проєкту. Такі зміни можуть охоплювати, зокрема, перегляд пріоритетів або формування нових вимог. За таких умов керівник проєкту повинен своєчасно вносити відповідні корективи до проєктної документації. Крім того, кожен проєкт, незалежно від обраної методології, передбачає дотримання визначених процесів. Наприклад, для ведення документації може використовуватися спеціалізоване програмне забезпечення, таке як Atlassian Confluence, яке забезпечує створення та організацію проєктної документації. Особи з високим рівнем відповідальності дотримуються встановлених процесів і сприяють їх виконанню іншими членами команди. Одна з ключових функцій керівника проєкту полягає у здійсненні *планування*. Навіть у межах проєктів із коротким життєвим циклом здатність до стратегічного, довгострокового бачення сприяє формуванню конструктивних і стабільних відносин із замовником, що, своєю чергою, підвищує ймовірність подальшої співпраці та залучення до нових проєктів. У контексті функціональної складової проєктного управління довгострокове планування виступає невід’ємним елементом такої функції, як планування та пріоритизація проєктної діяльності.

Екстраверсія демонструє спрямованість на соціальну взаємодію, активну комунікацію та отримання емоційної стимуляції в процесі спілкування з іншими членами команди. Проявляється у комунікабельності, енергійності, позитивному емоційному фоні, схильності до командної взаємодії та активної участі в обговоренні проєктних завдань. *Основними показниками є:* комунікабельність, наполегливість, широка соціальна мережа, перевага до групових активностей. З

позиції сформованості професійної готовності майбутніх бакалаврів з інженерії програмного забезпечення, *комунікабельність* постає однією з ключових персональних якостей, зумовлених специфікою їхньої діяльності, що передбачає постійну взаємодію як із членами проєктної команди, так і із замовником. Недостатній рівень комунікації обґрунтовано розглядається як одна з провідних причин неуспішності проєктів [67]. У контексті функціональних обов'язків у проєктному управлінні комунікабельність виступає невід'ємною складовою професійної діяльності, зокрема у забезпеченні ефективної співпраці зі спонсором проєкту, командою та галузевими експертами з метою розроблення стратегії реалізації проєкту, а також у процесі інтерпретації та донесення ключових бізнес-аспектів проєкту до різних груп зацікавлених сторін. *Наполегливість* як професійно значуща характеристика виявляється у впевненій поведінці в соціальних взаємодіях: фахівці демонструють схильність до чіткого висловлення власної позиції, готовність брати на себе відповідальність у групових контекстах та здатність до конструктивного самоствердження за необхідності. У поєднанні з довгостроковим плануванням ця риса набуває особливої ваги в умовах проєктного управління. У процесі планування та комунікації із замовником бакалаври з інженерії програмного забезпечення мають цілеспрямовано застосовувати наполегливість як інструмент професійної взаємодії. Вона сприяє активізації діалогу із зацікавленими сторонами, зокрема отриманню релевантної інформації щодо поточного стану проєкту, організаційного контексту замовника, а також уточненню його стратегічних намірів і перспектив розвитку. *Наявність розвиненої соціальної мережі* є характерною ознакою фахівців, залучених до проєктної діяльності: вони, як правило, формують широку систему професійних і соціальних контактів та підтримують значну кількість взаємозв'язків. Схильність до встановлення й підтримання різноманітних соціальних зв'язків сприяє підвищенню ефективності комунікації, обміну знаннями та координації дій у межах проєктного управління. *Перевага до групових активностей* є характерною для фахівців, орієнтованих на роботу в умовах проєктного управління: вони віддають перевагу видам активності, що передбачають розвинену групову динаміку, зокрема

командним проектам і спільним професійним зусиллям. Враховуючи, що основою будь-якого проекту є команда, яка колективно забезпечує його реалізацію, така орієнтація набуває особливого значення. З позиції професійної готовності майбутніх бакалаврів з інженерії програмного забезпечення, зазначена характеристика сприяє ефективній організації командної взаємодії та створенню обґрунтованої стратегії реалізації проекту, що, у свою чергу, забезпечує максимізацію його бізнес-цінності.

Доброзичливість виявляє схильність до співпраці, довірливого ставлення до членів команди, емпатії та конструктивної взаємодії із зацікавленими сторонами проекту. Проявляється у прагненні до узгоджених рішень, готовності підтримувати інших учасників команди та орієнтації на досягнення спільного результату. *Основним показником є:* кооперативність. *Кооперативність* є важливою соціально-професійною характеристикою фахівців, залучених до проектної діяльності. Особи з високим рівнем цієї якості ефективно функціонують у командному середовищі, демонструючи здатність до продуктивної взаємодії з іншими учасниками проекту. Вони схильні до пошуку компромісних рішень, встановлення спільних точок узгодження та мінімізації деструктивних конфліктів і протистоянь, що сприяє підвищенню ефективності командної роботи та досягненню проектних цілей.

Невротизм як індивідуально-психологічна характеристика відображає схильність до переживання психологічного напруження, негативних емоційних станів і труднощів у здійсненні емоційної саморегуляції. Він проявляється у підвищеній чутливості до стресових чинників, рівні емоційної стабільності та здатності контролювати імпульсивні реакції. У контексті підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління ця характеристика набуває особливої значущості, оскільки діяльність у ІТ-середовищі часто здійснюється в умовах невизначеності, високої динаміки та обмежених часових ресурсів. Відповідно, рівень невротизму впливає на здатність фахівців ефективно адаптуватися до стресових ситуацій, підтримувати продуктивність та забезпечувати стабільність командної взаємодії. *Основними показниками є:*

емоційний контроль, стресостійкість, адаптованість до змін. *Емоційний контроль* характеризує здатність особистості регулювати власні емоційні реакції відповідно до вимог ситуації. Особи з розвиненим рівнем цієї якості демонструють стійкість до афективних проявів, знижений рівень емоційної лабільності та схильність до усвідомленого стримування імпульсивної поведінки в умовах підвищеної складності. У межах проєктного управління в галузі інженерії програмного забезпечення це забезпечує підтримання конструктивної взаємодії в команді, збереження раціональності у прийнятті рішень та ефективне реагування на виклики, пов'язані з невизначеністю й інтенсивністю робочих процесів. *Стресостійкість* відображає здатність особистості ефективно функціонувати в умовах підвищеного психологічного навантаження та невизначеності. Вона тісно пов'язана з рівнем емоційної стабільності та визначає спроможність адекватно реагувати на стресогенні чинники без істотного зниження продуктивності. Для осіб із високою стресостійкістю характерним є збереження врівноваженості, самоконтролю та раціональності поведінки в ситуаціях тиску або підвищених вимог. У контексті проєктного управління в інженерії програмного забезпечення це сприяє підтриманню стабільності робочих процесів, ефективному прийняттю рішень та узгодженій командній взаємодії. *Адаптованість до змін* характеризує здатність особистості гнучко реагувати на динамічні умови та ефективно перебудовувати власну діяльність відповідно до нових вимог. Особи з високим рівнем емоційної стабільності демонструють спроможність до швидкої адаптації в умовах змін, зберігаючи функціональну ефективність без надмірного рівня тривожності чи емоційного напруження. У контексті проєктного управління в інженерії програмного забезпечення ця якість забезпечує результативне опрацювання нових викликів, інтеграцію змін у процеси розробки та підтримання стабільності командної діяльності в умовах постійної трансформації середовища.

Отже, зміст особистісного компонента готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління доцільно розглядати крізь призму базових особистісних вимірів, зокрема відкритості до досвіду, сумлінності, екстраверсії, доброзичливості та невротизму. Кожна з цих

якостей репрезентується через сукупність відповідних показників, що відображають особливості поведінки, емоційної регуляції та соціальної взаємодії фахівців. Рівень сформованості зазначених характеристик і їхніх проявів визначає здатність майбутніх фахівців ефективно здійснювати проєктну діяльність, адаптуватися до змін, взаємодіяти в команді та приймати обґрунтовані рішення в умовах професійної невизначеності й динамічності ІТ-середовища.

Таким чином, розуміємо готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як здатність до виконання робіт пов'язаних з управлінням проєктами, яка структурно складається з функціональної основи та персональних якостей, а саме знань, умінь, навичок та якостей щодо планування та ретельного визначення проєктних та персональних пріоритетів; максимізації бізнес-цінностей проєкту; використання традиційних і agile інструментів, техніки та методів; пояснення важливих бізнес-аспектів проєкту іншим; керування елементами проєкту; контролю критично важливих технічних елементів проєктного управління.

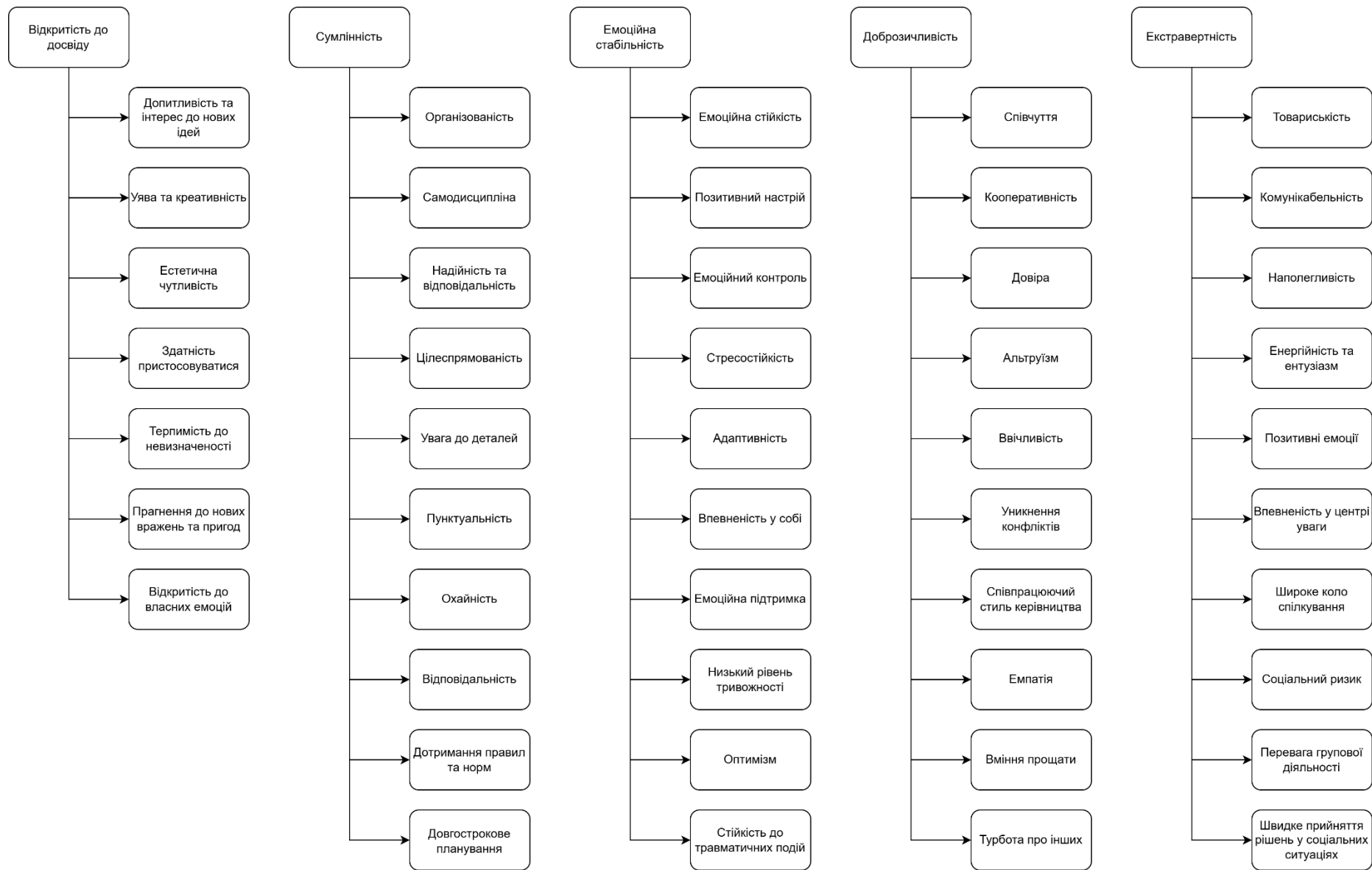


Рис. 1.2 Структура та показники особистісних якостей майбутніх бакалаврів з інженерії програмного забезпечення в рамках проєктного управління

Для підтвердження існування протиріччя між потребами забезпечення високого рівня готовності майбутніх фахівців з інженерії програмного забезпечення до проєктного управління як обов'язкового складника їхнього професіоналізму й недостатнім теоретико-методологічним і методичним обґрунтуванням його формування у закладах вищої освіти було проведено констатувальний експеримент за участі студентів з наступних ЗВО: Вінницький національний технічний університет, Таврійський державний агротехнологічний університет імені Дмитра Моторного та Луцький національний технічний університет.

Загалом в опитуванні прийняли участь 247 студентів I-IV курсів (див. табл. 1.2).

Таблиця 1.2

Кількість і розподіл студентів за спеціальністю та курсом

Спеціальність	Курс	Кількість студентів
121 Інженерія програмного забезпечення	1	52
121 Інженерія програмного забезпечення	2	23
121 Інженерія програмного забезпечення	3	30
121 Інженерія програмного забезпечення	4	10
122 Комп'ютерні науки	1	10
122 Комп'ютерні науки	2	32
122 Комп'ютерні науки	3	7
122 Комп'ютерні науки	4	5
123 Комп'ютерна інженерія	1	25
123 Комп'ютерна інженерія	2	18
123 Комп'ютерна інженерія	3	14
123 Комп'ютерна інженерія	4	13

Констатувальний експеримент було спроектовано у формі анкетування з використанням платформи Google Forms. Структурно опитування включало кілька

змістових блоків: блок організаційних питань; блок самооцінювання індивідуально-психологічних якостей відповідно до моделі «Великої п'ятірки»; а також блок запитань, спрямованих на визначення рівня сформованості функціональної готовності майбутніх бакалаврів з інженерії програмного забезпечення до здійснення проєктного управління.

Блок з організаційних питань містить загальні запитання, зокрема щодо надання згоди на обробку персональних даних, визначення курсу навчання респондента та його спеціальності. Отримані відповіді передбачено використовувати для подальшої обробки та інтерпретації результатів експериментального дослідження.

У блоці самооцінювання індивідуально-психологічних якостей відповідно до моделі «Великої п'ятірки» було використано запитання з відповідями, структурованими за шкалою Лайкерта [68]. Респондентам пропонувалося обрати один із п'яти варіантів оцінки, де значення «1» інтерпретувалося як «повністю не згоден», а «5» – як «повністю згоден». Зміст запитань було сформовано відповідно до основних компонентів моделі «Великої п'ятірки», зокрема: організованість, відкритість до нового досвіду, емоційна стійкість, екстраверсія та привітність (див. табл. 1.3).

Таблиця 1.3

Питання з оцінки персональних якостей

Текст питання	Пояснення до питання
Я організований	Я завжди будую плани і дотримуюся їх. Я старанний і уважний у своїй роботі. Я віддаю перевагу закінчувати завдання, перш ніж розслабитися чи розважитися.
Я відкритий до нового	Мені подобається досліджувати нові культури та ідеї. Я часто думаю про абстрактні концепції та теорії. Я відкритий для того, щоб спробувати нову їжу та заняття.

Я емоційно стійкий	Я рідко відчуваю напругу або нервозність. Я рідко хвилююся через різні речі. Мене важко вивести з емоційної рівноваги.
Я екстравертний	Я відчуваю заряд енергії від спілкування. Мені подобається бути в оточенні інших людей. Я комунікабельний і відвертий у соціальних ситуаціях.
Я привітний	Я співчутливий до інших. Мені легко довіряти людям. Ціную гармонію та співпрацю у стосунках.

Блок запитань, спрямований на оцінювання функціональної складової готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління, включає десять завдань, кожне з яких передбачає можливість вибору до двох варіантів відповіді. Формат завдань є варіативним: частина з них орієнтована на вибір правильних відповідей, тоді як інша – на ідентифікацію неправильних. Кількість допустимих відповідей, а також тип завдання (вибір правильних або неправильних варіантів) зазначено в інструкції до кожного запитання (Q_n). Кожне завдання співвіднесене з окремою функцією проектного управління.

Наприклад: Q1. Замовник звернувся із запитом щодо формування команди у складі п'яти осіб для виконання проекту тривалістю 1,5 року (dedicated team). Йому треба 3 розробники та 2 тестувальника. В умовах до контракту, замовник хоче прописати умови гарантії при яких команда протягом 2 місяців після закінчення проекту має виправляти знайдені помилки (баги). Які будуть ваші дії? *Варіанти відповідей:* 1) Запропонувати інший тип контракту, наприклад fixed price, або time & material. 2) Погодитись на запропоновані умови. 3) Запропонувати додати до команди ще проектного керівника (project manager), який буде контролювати якість роботи команди. 4) Запропонувати розділити контракт на два окремих: на dedicated team та maintenance. *Функції проектного управління:* використання як

традиційних, так і agile інструментів, техніки та методів для кожного проєкту; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками; увага до критично важливих технічних елементів проєктного управління. *Правильна відповідь:* «запропонувати інший тип контракту, наприклад fixed price, або time & material» та «запропонувати розділити контракт на два окремих: на dedicated team та maintenance». *Обґрунтування відповіді.* Особливості контракту типу «dedicated team» полягають в тому, що компанія замовник отримує в своє розпорядження команду працівників і повністю відповідальна за результат їх роботи. Тобто усі функції керування проєктом та очікуваннями лежать на стороні замовника, а працівники в рамках контракту даються в «оренду». Іншими словами, об'єктом контракту є працівники, їх рівень підготовки, кількість, рівень залученості та терміни. При виконанні робіт згідно контрактів fixed price, або time & material, в контракті можуть бути прописані умови, при яких робота вважається виконаною, бо об'єктом контракту є проєкт. У разі, якщо проєктний керівник, або інша людина відповідальна за контракт погодиться на умови, які описані у питанні, то є великий ризик того, що цей контракт буде неприбутковим та/або проблемним.

Q2. За результатами ретро мітингу учасники команди захотіли отримати розуміння стратегічних (майбутніх) планів замовника. До кого, ви, як проєктний менеджер, звернетесь за цією інформацією? *Варіанти відповідей:* 1) До спонсора проєкту. 2) До бізнес аналітика. 3) До архітектора. 4) До одного зі стейкхолдерів. *Функції проєктного управління:* співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту; пояснення важливих бізнес-аспектів проєкту іншим. *Правильна відповідь:* до спонсора проєкту. *Обґрунтування відповіді.* Згідно з функціональними компонентами проєктного керівника «співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту». Одним з елементів співпраці є отримання інформації про стратегічні плани.

Q3. До вас звернувся бізнес-аналітик із проблемою, що два різні стейкхолдери пропонують зміни до системи, внаслідок чого вимоги одного

суперечать вимогам іншого. Усі спроби бізнес-аналітика узгодити ці вимоги та усунути суперечності виявилися безуспішними. Якими будуть ваші дії як керівника проєкту? *Варіанти відповідей:* 1) Створити спільний дзвінок з двома стейкхолдерами, бізнес-аналітиком та виступити самому/самій в якості рефері. 2) Створити спільний дзвінок з двома стейкхолдерами, бізнес-аналітиком та спонсором проєкту в якості рефері. 3) Створити спільний дзвінок з двома стейкхолдерами та вами, щоб спробувати домовитись. 4) Запропонувати бізнес-аналітику «відкласти» ці задачі. *Функції проєктного управління:* співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту; пояснення важливих бізнес-аспектів проєкту іншим; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками. *Правильна відповідь:* створити спільний дзвінок з двома стейкхолдерами, бізнес-аналітиком та спонсором проєкту в якості рефері. *Обґрунтування відповіді.* Згідно з функціональними компонентами проєктного керівника «співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту». Спонсор проєкту – це людина яка знаходиться над стейкхолдерами та також є головним представником бізнесу, який замовляє розробку проєкту. Саме спонсор проєкту може бути рефері у такому випадку і задача проєктного керівника – це залучити спонсора до такого дзвінка у якості рефері.

Q4. Під час аналізу специфікацій SMS-провайдера, з яким планується інтеграція, один із розробників виявив потенційний ризик, пов'язаний тим, що частини SMS-повідомлень може бути не доставлена. Якими будуть ваші дії як керівника проєкту? (Обрати треба одну неправильну відповідь). *Варіанти відповідей:* 1) Змінити SMS-провайдера на іншого. 2) Відмова від інтеграції. 3) Прийняття ризику. 4) Не відправляти в SMS нічого важливого. *Функції проєктного управління:* співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками; увага до критично важливих технічних елементів

проектного управління. *Правильна відповідь:* змінити SMS-провайдера на іншого. *Обґрунтування відповіді.* Згідно з Enterprise risk management підходом [69, с. 4], основними стратегіями роботи з ризиками є наступні: уникнення, зменшення, прийняття та моніторинг. Відмова від інтеграції прибирає ризик того, що деякі SMS-повідомлення не будуть доставлені. Прийняття ризику, фактично, призводить до того, що ми на рівні проекту погоджуємось з тим, що деякі SMS-повідомлення не будуть доставлені. Тактика з «Не відправляти в SMS нічого важливого» частково зменшує ефект від того, що деякі SMS-повідомлення не будуть доставлені. Відповідь «змінити SMS-провайдера на іншого» не зменшує ризик, бо немає гарантії того, що інший SMS-провайдер не має таких самих проблем.

Q5. Замовник звернувся із запитом щодо високорівневої та оперативної оцінки двох масштабних майбутніх функціональних змін проекту: декомпозиції монолітної архітектури на мікросервісну та оновлення мажорних версій використовуваних фреймворків. Які підходи до оцінювання доцільно застосувати за умови використання в проекті ітеративної моделі розробки? *Варіанти відповідей:* 1) T-shirt size estimate. 2) COCOMO modelling. 3) Top-down estimate. 4) Critical path method (CPM). *Функції проектного управління:* планування та ретельне визначення пріоритетів (проектних та персональних); використання як традиційних, так і agile інструментів, техніки та методів для кожного проекту; керування елементами проекту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками. *Правильна відповідь:* «T-shirt size estimate» та «Top-down estimate». *Обґрунтування відповіді.* Саме «T-shirt size estimate» та «Top-down estimate» дозволяють отримати високорівневу оцінку, тобто маючи мінімальну кількість інформації про роботу, яку треба буде виконати. Як зазначено у питанні, замовник просить оцінити швидко та високорівнево, тобто дозволяється мати результатом діапазон часу, який треба на виконання завдань. Самі по собі, дві задачі є великими та мають багато додаткових ризиків. Наприклад, під час підняття мажорних версій фреймворків, може так статись, що буде необхідно змінювати код проекту. Звісно, це напряду залежить від змін, які були зроблені у фреймворках та їх зворотної сумісності з кодом проекту. Прорахувати

такі ризики не витративши додаткового часу на пошук інформації про зміни у фреймворках можна тільки з допомоги правильно підібраних методів оцінки. Підходи «COCOMO modelling» та «Critical path method» дозволяють отримати вищу якість оцінки, але потребують багато додаткової роботи.

Q6. Ви берете участь у плануванні запуску нового проєкту, пов'язаного зі створенням прошивки для медичного обладнання, призначеного для автоматизації процесу хіміотерапії. Відповідно до попередніх оцінок, тривалість розробки становитиме 4 роки, а склад команди налічуватиме близько 100 фахівців (розробників, тестувальників тощо). Який підхід до організації процесу розробки доцільно обрати в даному випадку? *Варіанти відповідей:* 1) Waterfall. 2) Scrum. 3) SAFe. 4) Kanban. *Функції проєктного управління:* використання як традиційних, так і agile інструментів, техніки та методів для кожного проєкту; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками; увага до критично важливих технічних елементів проєктного управління. *Правильна відповідь:* «Waterfall» та «SAFe». *Обґрунтування відповіді.* Саме ці методології можна використати для проєкту такого домену та розмірів. Методологія Agile Scrum дуже зручна у випадку, коли замовник постійно очікує мати зміни у проєкті, тобто команда має бути достатньо гнучкою до постійних змін у вимогах. Такі проєкти, як програмне забезпечення для медичного обладнання мають величезну кількість документації, яка вже пропрацьована різними командами (архітекторами, бізнес аналітиками та ін.), тому очікується, що зміни у ній будуть мінімальними, а у найкращому випадку, повністю відсутніми. З іншого боку, методологія Scaled Agile Framework (SAFe) може бути альтернативою для такого проєкту. Згідно офіційній документації, вона дозволяє отримати усі плюси Agile підходів та, водночас з цим, застосовуватись у великих командах.

Q7. Вас призначили керівником проєкту для стартапу, якому доручено розробити нову революційну платформу соціальних мереж. На початковому етапі планування вам потрібно створити документ, який описує ціннісну пропозицію, інфраструктуру, клієнтів і фінанси проєкту в короткому

односторінковому форматі. Який документ ви підготуєте? *Варіанти відповідей:* 1) Бізнес кейс проєкту. 2) Канва бізнес моделі. 3) Короткий опис проєкту. 4) Статут проєкту. *Функції проєктного управління:* планування та ретельне визначення пріоритетів (проєктних та персональних); максимізація бізнес-цінності проєкту за рахунок правильно створеної стратегії його реалізації; співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками. *Правильна відповідь:* канва бізнес моделі. *Обґрунтування відповіді.* Канва бізнес моделі [70] є одним з артефактів проєкта на рівні з бізнес кейсом проєкту, коротким описом проєкту та статутом проєкту.

Q8. Вас призначили керівником проєкту по розробці нового інструменту підвищення продуктивності для віддалених команд. Проєкт включає кілька етапів розробки та вимагає координації між різними командами. Який документ ви презентуєте командам, щоб показати основні результати та ключові моменти прийняття рішень проєкту? *Варіанти відповідей:* 1) Бізнес кейс проєкту. 2) Бізнес модель. 3) Короткий опис проєкту. 4) Дорожня карта проєкту. *Функції проєктного управління:* планування та ретельне визначення пріоритетів (проєктних та персональних); максимізація бізнес-цінності проєкту за рахунок правильно створеної стратегії його реалізації; співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту. *Правильна відповідь:* дорожня карта проєкту. *Обґрунтування відповіді.* Дорожня карта проєкту є одним з артефактів роботи проєктного керівника. Дорожні карти несуть в собі важливі данні про час та ключові моменти для проєкту.

Усі відповіді було вивантажено з гугл форми у вигляді csv (comma-separated value) файлу. Подальша обробка файлу відбувалась у застосунку Microsoft Excel за допомогою Data Transformation і використанням Power Query Editor. Першим етапом було відфільтровано усі інваріантні відповіді. Це відповіді, в яких студенти не дали дозвіл на обробку їх персональної інформації та відповіді, в яких було неможливо ідентифікувати курс та/або спеціальність. Після цього, були

опрацьовані відповіді на перевірку знань в області проєктного управління. Якщо відповідь на питання правильна – це 1 бал, якщо ні – 0 балів. Якщо питання мало декілька правильних відповідей, а студент не зміг відповісти на всі, або відповідь була частково правильною, то присвоювали 0 балів. Максимальна оцінка, яку могли отримати студенти відповівши на всі питання правильно, дорівнює 8 балам.

Після аналізу результатів опитування, частину респондентів було виключено. Подальшим розрахунком підлягало 239 респондентів. Розподіл оцінок здобувачів освіти щодо сформованості особистісного та функціонального компонентів готовності до проєктного управління зазначено в таблиці 1.4 та рисунку 1.3.

Таблиця 1.4

Сформованість готовності здобувачів вищої освіти до проєктного управління за особистісним компонентом

Критерій	1	2	3	4	5	Середнє значення
Я організований	5	32	85	76	41	3,49
Я відкритий до нового	9	24	58	82	66	3,72
Я емоційно стійкий	23	42	71	55	48	3,26
Я екстравертний	34	48	75	51	31	2,99
Я привітний	13	34	57	83	52	3,53



Рис. 1.3 Діаграма сформованості готовності здобувачів вищої освіти до проектного управління за особистісним компонентом

У частині дослідження персональних якостей здобувачів вищої освіти спеціальності «Програмне забезпечення» встановлено наявність усіх п'яти компонентів моделі «Великої п'ятірки» серед майбутніх бакалаврів. Водночас отримані результати ґрунтуються на самооцінюванні студентів за шкалою Лайкерта, що певною мірою обмежує можливість комплексної та об'єктивної інтерпретації індивідуально-психологічних характеристик у межах моделі «Великої п'ятірки».

За результатами кореляційного аналізу Спірмена статистично значущими виявилися лише такі компоненти, як «відкритість до нового досвіду» та «організованість». Встановлено їхній зв'язок із результатами перевірки якості знань студентів, однак сила такого впливу є невисокою ($Rho = 0,1924492$ та $Rho = 0,1634387$ відповідно). Це свідчить про те, що зазначені персональні характеристики можуть певною мірою впливати на навчальну успішність майбутніх фахівців з програмного забезпечення та формування їхньої готовності до проектного управління, проте не є визначальними чинниками.

Окремо проведений кореляційний аналіз між показниками «відкритості до нового досвіду» та «організованості» продемонстрував наявність статистично значущого взаємозв'язку ($Rho = 0,3108978$; $p\text{-value} = 9,428e-07$). Значення

коефіцієнта кореляції, що перевищує 0,3, дає підстави припускати існування помірного зв'язку між зазначеними персональними якостями. Отримані результати можуть свідчити про те, що вищий рівень організованості студентів потенційно пов'язаний із більшою відкритістю до нових знань, підходів та інструментів, що є важливим аспектом підготовки майбутніх бакалаврів з програмного забезпечення до діяльності в умовах проєктно-орієнтованого середовища.

Додатково прорахована множинна регресія показала статистично значущим відкритість до нового, однак коефіцієнт детермінації (R-squared) має значення всього 0,04922 водночас з низьким значенням P-value – 0,04922.

Таблиця 1.5

Результати множинного регресійного аналізу впливу особистісних рис на досліджуваний показник

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	0.93938	0.38187	2.460	0.0146
O	0.21321	0.08741	2.439	0.0155
C	0.16919	0.09316	1.816	0.0706
E	-0.13831	0.08136	-1.700	0.0904
A	0.05293	0.08692	0.609	0.5432
N	0.05525	0.07509	0.736	0.4626

Такі результати свідчать про те, що на оцінки студентів впливають ще й інші параметри, які не були враховані в моделі.

Таблиця 1.6

Порівняння результатів оцінювання студентів спеціальності «Інженерія програмного забезпечення» та інших спеціальностей

Критерій	Інженерія програмного забезпечення	Інші студенти
Середня оцінка	2,27	2,28
Розмір групи	115	124

Середні значення оцінок у двох групах є дуже близькими, різниця між ними становить лише 0,01. На основі емпіричної функції розподілу було побудовано графіки для обох груп. Як видно з отриманих результатів, розподіли мають подібні характеристики, за винятком незначних відмінностей в інтервалі оцінок 1–2. Однак

ці відмінності не мають суттєвого впливу на загальний результат, а лише підтверджують недостатній рівень знань студентів.

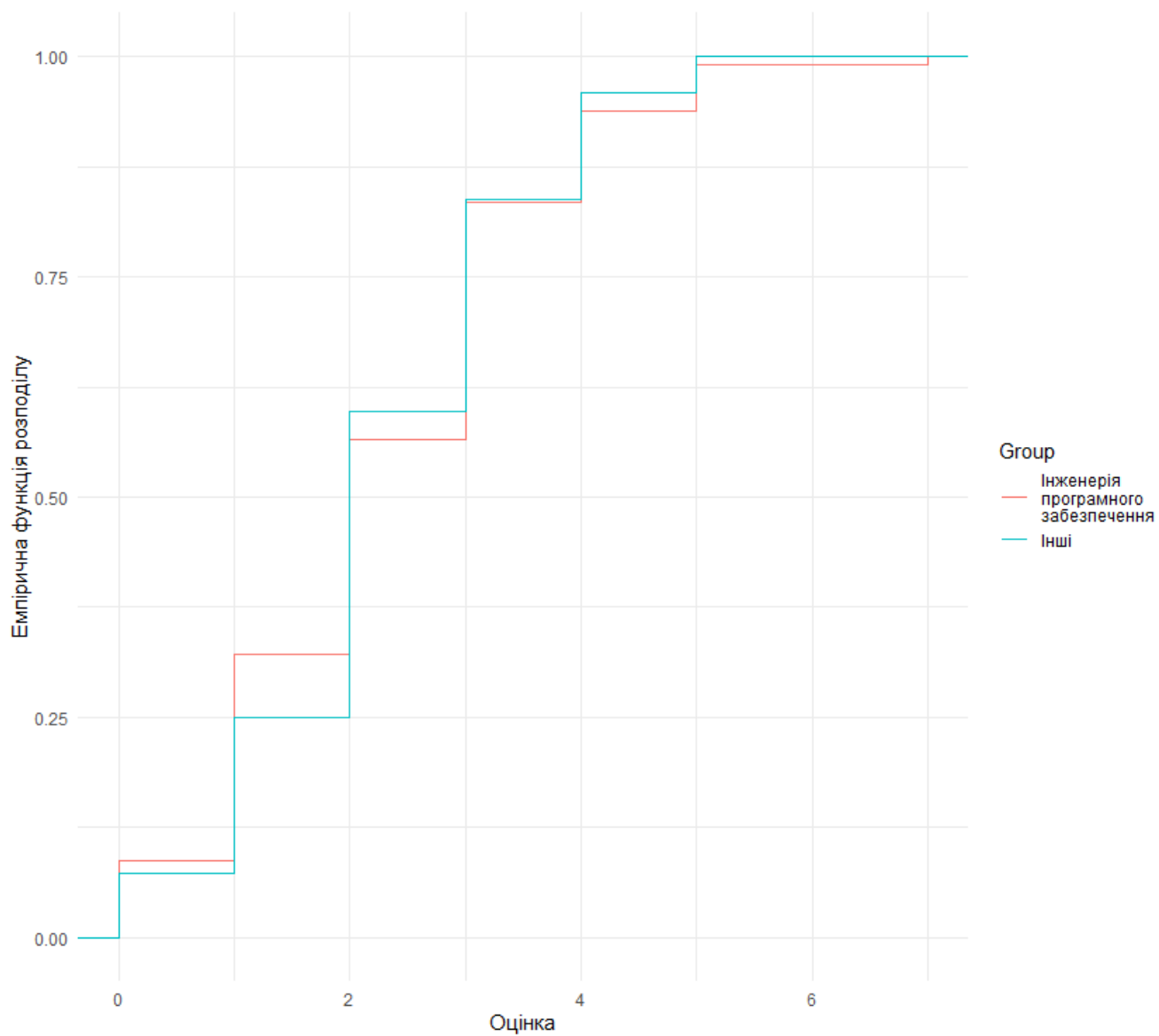


Рис. 1.4 Порівняння розподілів оцінок студентів спеціальності «Інженерія програмного забезпечення» та інших спеціальностей

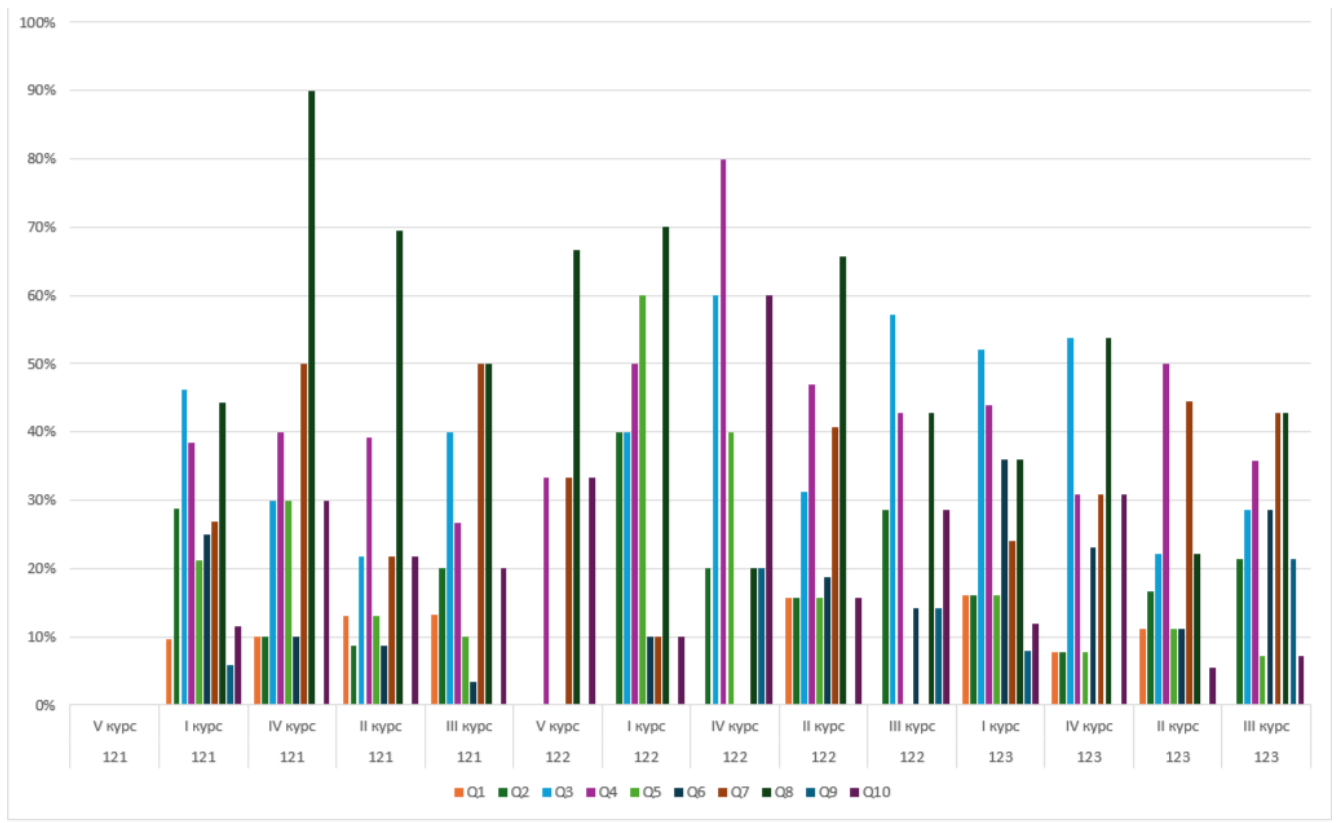


Рис. 1.5 Сформованість готовності здобувачів вищої освіти до проєктного управління за функціональним компонентом

Згідно результатів вимірювання, середній бал за кількістю вірно вирішених завдань серед студентів дорівнює 2,27 з 8 максимально можливих, 68 студентів не отримали жодного балу, тобто їх сумарна оцінка – 0. Тільки 39 студентів отримали 4 бали (50% відсоткова оцінка) та вище, що дорівнює всього 16,31%. За критерієм Шапіро-Вілка, розподілення оцінок студентів не є нормальним ($W = 0.93823$, $p\text{-value} = 1.723e-08$), що обмежує використання деяких методів статистичного аналізу. Відсоткова оцінка рахується за принципом відсотку правильних відповідей, де 100% це максимальний бал, а 0% – це мінімальний бал.

Дослідження показало наявність невеликої кореляції між рівнем прояву окремих компонентів «Великої п'ятірки» та оцінкою, яку студенти отримали. Водночас з цим самі оцінки, показали, що 28% студентів не відповіли правильно на жодне поставлення питання та лише 1 студент отримав відсоткову оцінку вище 65%. Такі низькі показники означають, що такий результат навчання, як «знати та вміти застосовувати методи та засоби управління проєктами» [22] можна та

необхідно покращувати. Судячи з результатів, можна стверджувати, що здобувачі вищої освіти обраних ІТ-спеціальностей не можуть знайти управлінські рішення в умовних виробничих ситуаціях. У доповнення до цього, результати оцінювання здобувачів вищої освіти спеціальності 121 «Інженерія програмного забезпечення» не показали вагомої різниці з іншими учасниками опитування, що може свідчити про більш глобальну наукову проблему в напрямку проєктного управління. Однак, результати оцінювання можуть бути менш точними, оскільки це тестування не впливало на академічні досягнення з дисциплін, і студенти могли діяти безвідповідально, обираючи випадкові відповіді, що позначилося на підсумкових результатах дослідження.

Студенти обирали відповіді, керуючись власними відчуттями, і ці оцінки не піддавалися подальшій перевірці. Як результат, повноцінно стверджувати про вплив компонентів «Великої п'ятірки» на якість результатів навчання студентів не є можливим.

Отже, попри високу значущість проєктного управління як ключового процесу організації діяльності в ІТ-компаніях, отримані результати свідчать про недостатній рівень обізнаності студентів ІТ-спеціальностей у цій сфері. Це дає підстави констатувати недостатню готовність майбутніх бакалаврів з інженерії програмного забезпечення до здійснення проєктного управління, що, своєю чергою, актуалізує необхідність удосконалення підходів до їхньої професійної підготовки.

1.3. Методологічні підходи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

У контексті дослідження проблеми підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління вагоме методологічне значення мають праці таких науковців, як В. Жигір, Р. Горбатюк, О. Тітова, О. Щетиніна, О. Овсянніков та їхніх співавторів. Значний масив розвідок цих дослідників присвячено впровадженню компетентнісного підходу, впливу

педагогічних наук на професійну освіту та інтеграції практико-орієнтованих задач в освітній процес [71; 72]. Для моделювання професійної підготовки майбутніх фахівців (зокрема сфери цифрових технологій) концептуально важливим є використання електронних підручників [73], а також впровадження засобів цифрових, інформаційних та мультимедійних технологій у закладах вищої освіти [74; 75; 76; 77; 78]. Автори окремо наголошують на специфіці ІТ-освіти, обґрунтовуючи дієвість персоналізованого навчання через платформи з відкритим вихідним кодом [79] та методів групових дискусій при вивченні комп'ютерних дисциплін [80]. Особливу цінність для формування проєктно-управлінської компетентності становлять результати досліджень щодо розвитку підприємницької компетентності інженерів [81], а також розуміння сучасних маркетингових комунікацій і таргетованих підходів [82]. Оскільки проєктне управління потребує розвинених гнучких навичок, фундаментальними є розвідки щодо формування здібностей до ділового спілкування в умовах дистанційного навчання [83]. До того ж практичний досвід застосування систем управління завданнями, таких як Trello, виступає дієвим інструментом для розвитку навичок навчання впродовж життя у здобувачів вищої освіти [84].

В. Жигір'єв спільно з колегами обґрунтовує необхідність переходу до практико-орієнтованого навчання та використання активних методів викладання для здобувачів інженерно-технічного профілю [85; 86]. Окрему увагу науковці приділяють організації технологічної практики [87] та розкриттю дидактичного потенціалу методу кейсів [88], що є надзвичайно актуальним інструментом для моделювання реальних ситуацій у сфері ІТ-менеджменту.

О. Тітова та її співавтори обґрунтовують доцільність розвитку підприємницького мислення у здобувачів інженерних спеціальностей через залучення підходу Business Canvas, який є базовим інструментом сучасного продуктового та проєктного менеджменту [89]. Окрім того, успішна реалізація проєктів вимагає високого рівня навичок комунікацій (soft skills), тому дослідники акцентують увагу на ефективності використання ІКТ-інструментів для

цілеспрямованого розвитку здібностей до комунікації та командної співпраці (collaboration skills) безпосередньо у студентів комп'ютерних наук [90]. Для того щоб забезпечити системність такої багатокomпонентної підготовки та об'єктивно вимірювати рівень сформованості необхідних компетентностей, автори пропонують впроваджувати спеціальний таксономічний підхід до оцінювання результатів навчання майбутніх інженерів [91].

Обираючи методологічні підходи професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, виходимо з того, що підхід – це зумовлена певною позицією або ідеєю система взаємопов'язаних принципів, методів, способів і прийомів, які визначають, як саме розглядається об'єкт, здійснюється вплив на нього та формується ставлення до нього, причому всі ці елементи мають спільну спрямованість і функціонують у межах єдиної концептуальної рамки [92].

Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління ґрунтується на поєднанні концептуальних підходів (системного, синергетичного та компетентнісного) і специфічних підходів (інтегративного та проєктного).

Обґрунтування та впровадження методологічних підходів у професійній освіті, зокрема системного підходу, здійснювали такі науковці, як Н. Брюханова [93], І. Васильєв [94], В. Докучаєва [95] та інші.

У науковій інтерпретації система розглядається як упорядкована сукупність взаємопов'язаних елементів, що перебувають у взаємодії та утворюють цілісність, у межах якої об'єкти, їхні властивості й зв'язки функціонують у єдності. Системний підхід визначається як напрям наукової методології, зорієнтований на розроблення методів дослідження і проєктування складно організованих об'єктів як цілісних систем. У педагогіці його застосування передбачає розгляд освітніх явищ як інтегрованих утворень, виявлення різних типів зв'язків між їхніми компонентами та узагальнення їх у цілісну теоретичну модель [96].

В основі *системного підходу* лежить низка принципів. *Принцип багатоплановості* – будь-який об'єкт розглядається в декількох аспектах. Згідно

із загальною теорією систем доцільним вважаємо виділення трьох аспектів: функціонального, елементного та організаційного. Функціональний аспект вивчає та визначає коло функцій, які повинна виконувати система і відповідні підсистеми. Функціональні підсистеми, що складаються з функцій однакової цільової спрямованості, визначають коло завдань та формулюють логіку дії системи. Елементний аспект передбачає дослідження об'єкта і встановлення його елементного складу. Організаційний аспект встановлює структуру системи, визначає та реалізує завдання відповідно до функціонального призначення. *Принцип багатомірності* полягає в тому, що будь-який складний об'єкт характеризується великою сукупністю властивостей, які об'єднані в групи, кожна з яких описує ті чи інші його особливості. *Принцип ієрархічності* полягає в тому, що вивчення складних об'єктів має базуватися на уявленні про ієрархічність їхньої структури, а саме на уявленні про розміщення частин або елементів цілого у порядку від вищого до нижчого. Ієрархічну структуру мають не тільки моделі складу системи, а також властивості якості цих систем та критерії, що використовуються для їх оцінки. *Принцип різнопорядковості* властивостей полягає в тому, що ієрархічність будови системи та її властивостей породжує закономірності різного порядку. *Принцип динамічності* полягає в розгляданні об'єктів у їх розвитку на всіх етапах життєвого циклу. *Принцип людських пріоритетів*, який орієнтує на людину – учасника системи. *Принцип діагностованості*, що передбачає організацію постійного зворотного зв'язку, реалізацію вимірjuвального інструментарію, моніторинг функціонування системи на практиці. *Принцип відповідності*, тобто узгодженості між складовими системи підготовки (наприклад, між змістом підготовки та професійними вимогами).

Системний підхід обрано для визначення загальних системних властивостей та якісних характеристик професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Згідно з принципом багатоплановості категорія готовності є складником інтегративної компетентності майбутніх фахівців спеціальності 121 «Інженерія програмного

забезпечення» першого (бакалаврського) рівня вищої освіти [22] та відповідає шостому кваліфікаційному рівню Національної рамки кваліфікацій [97] і передбачає здобуття особою теоретичних знань і практичних умінь і навичок, достатніх для успішного виконання професійних обов'язків за обраною спеціальністю.

Відповідно до принципу багатомірності, кожен компонент готовності до проєктного управління характеризується комплексом взаємопов'язаних властивостей, функціонування яких забезпечує досягнення очікуваного результату – сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до здійснення проєктного управління. Компоненти готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління перебувають у взаємозв'язку.

Добір критеріїв і показників готовності ґрунтується на розумінні її ієрархічної структури, що охоплює готовність як цілісне утворення, її компоненти та елементи цих компонентів. Такий підхід дав змогу виокремити рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, а саме: початковий, середній і високий.

Відповідно до принципу динамічності, розвиток готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління відбувається протягом усього періоду навчання за першим (бакалаврським) рівнем вищої освіти. Це зумовлює необхідність організації постійного зворотного зв'язку, застосування відповідного діагностичного інструментарію та здійснення моніторингу рівня сформованості такої готовності на всіх етапах професійного становлення здобувачів, що відповідає принципу діагностованості.

Системний підхід було застосовано під час розроблення структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Оскільки підготовка до проєктного управління є складовою професійної підготовки майбутніх фахівців, структура цього процесу охоплює взаємопов'язані компоненти, зокрема мету, мотиваційний

аспект, зміст навчання, форми, методи та засоби навчальної діяльності, а також систему контролю й діагностики результатів підготовки. Системний підхід дає змогу визначити мету та завдання навчальної, практико-зорієнтованої і самостійної діяльності здобувачів освіти відповідно до навчального плану та освітньо-професійних програм закладу вищої освіти. Його застосування сприяє виокремленню структури й змісту предметних та міждисциплінарних знань, що відповідають основним функціям бакалаврів з інженерії програмного забезпечення у сфері проєктного управління, а також забезпечує обґрунтований добір навчальних дисциплін, змісту освітніх програм і навчально-методичних ресурсів. Реалізація системного підходу також передбачає вибір ефективних форм, методів, прийомів і засобів організації навчальної, практико-зорієнтованої та самостійної діяльності студентів, а також визначення критеріїв оцінювання сформованості їхніх знань, умінь, навичок і професійно важливих якостей. Його використання забезпечує організацію дієвого контролю та самоконтролю академічних і особистісних досягнень здобувачів освіти, добір оптимальних шляхів їх удосконалення, а також здійснення якісної діагностики й самодіагностики рівня підготовки з метою подальшого підвищення її ефективності.

Отже, використання системного підходу та врахування відповідних принципів дає змогу розглядати підготовку майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як цілісну педагогічну систему, у межах якої всі компоненти освітнього процесу функціонують у взаємозв'язку та взаємозумовленості. Результатом функціонування такої системи є сформована готовність до проєктного управління, що інтегрує професійні знання, практичні вміння, досвід діяльності та особистісні якості, необхідні для ефективного виконання управлінських функцій у професійній сфері.

Синергетичний підхід вивчали С. Клепко [98], В. Кушнір [99] та інші.

Синергетичний підхід створює можливості для розгляду особистості майбутніх бакалаврів з інженерії програмного забезпечення та освітнього процесу як відкритої системи, здатної до саморозвитку й самоорганізації. У

контексті цього підходу формування готовності до проєктного управління трактується як динамічний процес взаємодії взаємопов'язаних підсистем, між якими відбувається безперервний обмін інформацією, досвідом і результатами діяльності. Застосування синергетичного підходу передбачає поступовий перехід від зовнішнього управління освітньою діяльністю до самоуправління здобувачів освіти, а також від розвитку, зумовленого переважно зовнішніми впливами, до саморозвитку, що ґрунтується на внутрішній мотивації та потенціалі освітнього середовища. Це забезпечує активізацію професійного становлення майбутніх фахівців і підвищує ефективність формування їхньої готовності до проєктного управління.

З позиції синергетичного підходу організація освітньої системи ґрунтується на низці фундаментальних принципів, що визначають її функціонування та розвиток. Одним із ключових є принцип відкритості педагогічної системи до зовнішнього середовища, відповідно до якого система здатна еволюціонувати, ускладнювати власну структуру та розвиватися завдяки постійному обміну інформацією й ресурсами з іншими системами. Важливе значення мають також принципи самоорганізації та цілісності, які забезпечують формування активного й багатокomпонентного освітнього середовища, орієнтованого на підтримання взаємозв'язків між його елементами. У межах синергетичного підходу педагогічна система розглядається як така, що характеризується нестійкістю, динамічністю та здатністю до змін у критичних точках розвитку, що зумовлює її відкритість до нових впливів, варіативності й подальшого саморозвитку. Суттєвими характеристиками освітньої системи є також нелінійність, ієрархічність, когерентність та емерджентність, відповідно до яких цілісна система набуває нових якостей, що не зводяться лише до суми властивостей окремих компонентів. Освітній процес у цьому контексті має ймовірнісний, творчий і самодетермінований характер, передбачає відкритість до експерименту, невизначеності та самоактуалізації особистості. Окрему роль відіграє принцип спостережуваності, який відображає відносність і обмеженість уявлень про педагогічну систему залежно від масштабу її аналізу, умов

функціонування та очікуваних результатів освітньої діяльності [100].

Одним із ключових положень синергетичного підходу є ідея нелінійності розвитку особистості, відповідно до якої професійне становлення майбутніх бакалаврів з інженерії програмного забезпечення не має прямолінійного характеру, а відбувається через закономірне чергування станів стабільності та нестабільності, впорядкованості й невизначеності. Такі стани є природними етапами розвитку особистості як відкритої системи, здатної до самоорганізації та саморозвитку. Періоди нестійкості супроводжуються переосмисленням попереднього досвіду, зміною професійних орієнтирів і пошуком нових способів діяльності, тоді як стан упорядкованості характеризується формуванням нової структури професійної готовності, зокрема готовності до проєктного управління в ІТ-сфері.

У контексті підготовки майбутніх бакалаврів з інженерії програмного забезпечення важливого значення набуває принцип когерентності, який відображає узгодженість і взаємозалежність усіх компонентів системи підготовки. Зміни в одному зі структурних елементів готовності – функціональному чи особистісному – спричиняють трансформації в інших компонентах. Це створює можливість цілеспрямовано впливати на формування окремих складових готовності шляхом опори на вже сформовані професійні якості, знання або практичний досвід здобувачів освіти. Наприклад, розвиток навичок командної взаємодії під час виконання ІТ-проєктів може позитивно впливати на формування управлінських умінь, відповідальності та здатності до прийняття рішень.

У межах синергетичного підходу важливими є також поняття флуктуації та надмалого впливу. Флуктуації розглядаються як незначні зміни або відхилення в системі, які за певних умов можуть стати поштовхом до суттєвих якісних перетворень. У процесі професійної підготовки такими флуктуаціями можуть бути нові навчальні ситуації, участь у командних проєктах, виконання ролі керівника групи, рефлексія власних помилок або взаємодія з професійним середовищем ІТ-галузі. Якщо вчасно застосувати відповідний педагогічний

вплив, навіть незначна освітня подія здатна активізувати внутрішні ресурси особистості та сприяти переходу системи на вищий рівень розвитку.

У зв'язку з цим особливого значення набуває концепція «м'якого управління», яка передбачає використання не інтенсивних зовнішніх впливів, а педагогічно доцільних і резонансних стимулів, узгоджених із внутрішніми потребами та потенціалом здобувачів освіти. Такий підхід орієнтований не на жорстке регламентування професійного становлення майбутніх фахівців, а на створення умов для їхньої самореалізації, самостійного вибору траєкторії розвитку та поступового формування готовності до проєктного управління. Своєчасні й педагогічно виважені впливи дають змогу активізувати внутрішні можливості особистості, забезпечуючи гармонійний розвиток професійних і особистісних якостей без порушення цілісності системи професійної підготовки.

Отже, розвиток готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління доцільно розглядати як процес безперервного саморозвитку та самоорганізації особистості, що зумовлює необхідність реалізації суб'єкт-суб'єктної взаємодії в освітньому процесі з урахуванням закономірностей синергетичного підходу. Наявність різних траєкторій професійного становлення актуалізує проблему вибору ефективних педагогічних впливів, здатних забезпечити гармонійний розвиток функціонального та особистісного компонентів готовності до проєктного управління. Особливого значення в цьому контексті набуває створення умов для розвитку гнучкого й інтегративного мислення майбутніх фахівців, здатності до прийняття нестандартних рішень, командної взаємодії та ефективного управління проєктними процесами. Для майбутніх бакалаврів з інженерії програмного забезпечення важливою є здатність аналізувати проблему з різних позицій, генерувати альтернативні способи її розв'язання та обирати найбільш доцільний варіант відповідно до умов реалізації ІТ-проєкту. За таких умов роль викладача полягає не лише у передачі знань, а й у спрямуванні здобувачів освіти до самостійного пошуку оптимальних рішень, розвитку професійної ініціативності, відповідальності та здатності до самоуправління у процесі

проектної діяльності.

У синергетичному контексті підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління розглядається як процес створення умов для професійного саморозвитку, самостійного конструювання знань і пошуку власних стратегій професійного становлення в ІТ-сфері. Такий підхід передбачає орієнтацію освітнього процесу не лише на засвоєння теоретичних знань, а й на розвиток здатності до самоорганізації, прийняття управлінських рішень, командної взаємодії та відповідальності за результати проектної діяльності. Очікуваним результатом такої підготовки є сформована готовність майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління. Модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління має ґрунтуватися на ідеї розвитку особистості, здатної до самонавчання, самоуправління та професійного самовдосконалення. Її реалізація передбачає партнерську взаємодію між усіма учасниками освітнього процесу, методичне співтворення, взаєморегуляцію та підтримання суб'єкт-суб'єктних відносин. За таких умов здобувачі вищої освіти отримують можливість самостійно обирати траєкторії професійного розвитку, набувати досвіду управління командними ІТ-проектами та формувати власний стиль професійної діяльності.

Синергетичний підхід також забезпечує інтеграцію всіх компонентів освітнього процесу, реалізацію міждисциплінарних зв'язків і набуття знань через практичний досвід розв'язання професійно орієнтованих завдань. При цьому варіативність, невизначеність і необхідність прийняття рішень у процесі реалізації ІТ-проектів розглядаються як чинники розвитку професійних та управлінських якостей майбутніх фахівців. Готовність до проектного управління в такому контексті трактується як інтегративний результат професійної підготовки, що охоплює функціональний та особистісний компоненти й забезпечує ефективне виконання управлінських функцій у професійній діяльності. Важливою умовою реалізації моделі є створення освітнього середовища, яке стимулює внутрішню мотивацію студентів, розвиток

самодисципліни, відповідальності, рефлексії та здатності до самоконтролю. Поєднання професійного й особистісного розвитку сприяє формуванню відкритості до співпраці, здатності до конструктивного розв'язання проблемних ситуацій і розвитку творчого мислення, необхідного для успішного управління ІТ-проєктами.

У вітчизняній педагогічній науці значного поширення набули дослідження, присвячені компетентнісному підходу Н. Бібік, Л. Ващенко, О. Локшина, О. Овчарук, Л. Паращенко, О. Пометун, С. Трубачева [101; 102; 103; 104; 105; 106] та інші.

Компетентнісний підхід дозволяє зовсім по-іншому будувати цілі і зміст освіти. Упровадження цього підходу зумовлює визначення якості підготовки майбутніх бакалаврів з інженерії програмного забезпечення шляхом освоєння ними відповідних загальних і фахових компетентностей [22]. При цьому увагу акцентовано не на змісті, а на результатах освіти, що виявляються у формі компетентностей і відображають здатність фахівця продуктивно діяти в професійній діяльності. Важливим постає відбір змісту навчання, що відображає цілісний досвід вирішення життєвих і професійних завдань і втілення його шляхом використання організаційних форм, дидактичних методів та інших професійних інструментів.

Компетентнісний підхід посилює практичну спрямованість освіти, акцентуючи значення досвіду та здатності застосовувати знання на практиці, що зумовлює підпорядкованість знань умінням. Він також зосереджує увагу на результатах освіти, трактуючи їх не як сукупність засвоєної інформації, а як здатність особистості розв'язувати життєві й професійні завдання та ефективно діяти в різних проблемних ситуаціях. Реалізація підходу, орієнтованого на результати навчання та компетентності, передбачає також трансформацію методів викладання, навчання й оцінювання, що використовуються в освітній програмі.

Компетентнісний підхід у вищій освіті характеризується орієнтацією на визначення компетентностей як очікуваних результатів навчання та їх

цілеспрямоване формування, що супроводжується зміщенням акценту з накопичення знань на здатність їх ефективного практичного застосування. Важливою ознакою цього підходу є оцінювання рівня сформованості компетентностей як ключового результату освітнього процесу, а також впровадження студентоцентрованого навчання з урахуванням потреб подальшого працевлаштування випускників. У науковому розумінні компетентність постає як інтегративне утворення, що об'єднує знання та розуміння об'єкта діяльності, здатність до їх доцільного використання у практичних ситуаціях, а також особистісні характеристики, зокрема переконання, установки на самореалізацію і систему ціннісних орієнтацій. Відповідно, реалізація компетентнісного підходу спрямована на формування високого рівня професіоналізму, що базується на сукупності знань, умінь і ставлень, які забезпечують ефективне виконання професійних функцій відповідно до встановлених стандартів діяльності.

У процесі навчання у майбутніх бакалаврів з інженерії програмного забезпечення відбувається формування та розвиток професійної спрямованості, що проявляється у позитивному ставленні до обраної професії, наявності інтересу й схильності до неї, а також прагненні до вдосконалення власної підготовки та задоволення матеріальних і духовних потреб через професійну діяльність. Посилюються мотиви, пов'язані з майбутньою професією, формується орієнтація на якісне виконання професійних обов'язків, зростає прагнення до самореалізації як компетентного фахівця та досягнення професійного успіху. Водночас підвищується рівень домагань щодо ефективного розв'язання складних навчальних завдань і посилюється почуття відповідальності.

Отже, у контексті підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління компетентнісний підхід має забезпечити формування системи знань, умінь, особистісних якостей і мотивацій, що сприятимуть становленню висококваліфікованого фахівця, створюватимуть передумови для його професійної самореалізації,

забезпечуватимуть затребуваність особистісного потенціалу та усвідомлення власної значущості в професійній діяльності. Досягнення таких результатів можливе за умови усвідомленого засвоєння студентами навчального матеріалу та розвитку здатності самостійно, творчо й нестандартно застосовувати набуті знання у процесі розв'язання прикладних завдань, зокрема у сфері проектного управління. Реалізація зазначеного підходу передбачає дотримання таких вимог: відбір змісту навчання відповідно до кінцевих цілей підготовки; удосконалення методів викладання, навчання та оцінювання; визначення професійної компетентності, зокрема в галузі проектного управління, як ключового результату освітнього процесу.

Відносно нашого дослідження принцип людиноцентризму вважаємо домінуючим у реалізації компетентнісного підходу. Людиноцентризм постає як складне й багатовимірне явище та процес. Його комплексне розуміння передбачає врахування щонайменше чотирьох аспектів. По-перше, він відображає активну реакцію освітнього середовища на динамічні зміни потреб ринку праці. По-друге, виступає як модель розвитку освіти, у межах якої здобувач освіти трансформується з об'єкта в суб'єкт освітньої діяльності, тобто стає активним учасником освітнього процесу. По-третє, передбачає зосередження уваги на індивідуальних особливостях і здібностях здобувачів освіти, формуванні їхніх індивідуальних освітніх траєкторій і, відповідно, персоналізованих профілів компетентностей. По-четверте, характеризується підвищенням рівня відповідальності за створення умов, що забезпечують досягнення високих результатів навчання та формування компетентностей, адекватних вимогам сучасного суспільства.

З позицій людиноцентризму компетентність розглядається як специфічна форма особистісного буття людини в діяльності, що є результатом цілісної, а не суто знаннево орієнтованої освіти. На відміну від традиційної моделі, компетентнісно орієнтована освіта характеризується якісно іншими властивостями: вона має суб'єктну спрямованість, діяльнісний характер і надпредметну природу. У такому освітньому процесі здобувачі освіти

орієнтуються не лише на опанування окремих навчальних дисциплін, а й на власний розвиток, професійне становлення, свідоме проєктування та реалізацію індивідуальної освітньо-професійної траєкторії.

Людиноцентроване компетентісно орієнтоване навчання має забезпечувати розвиток і самореалізацію особистості як майбутнього фахівця, формування її життєвої позиції, готовності та здатності до безперервного навчання, а також ефективної діяльності в умовах динамічних змін. Водночас людиноцентризм доцільно розглядати як важливий ресурс успішної реалізації компетентісного підходу у вищій освіті [107].

В основу студентоцентрованого навчання покладено ідею максимального забезпечення студентам їх шансів на ринку праці [108, с. 16], яка знаходить віддзеркалення в освітніх програмах, орієнтованих на результати навчання та врахування особливостей пріоритетів тих, хто навчається [109, с. 61], що актуалізує проблему досягнення максимальної ефективності цього процесу.

Наукові дослідження, присвячені оцінюванню ефективності студентоцентрованого навчання, засвідчують його високу результативність у порівнянні з традиційними підходами. Зокрема, у здобувачів освіти спостерігається вищий рівень сформованості умінь і глибше розуміння навчального матеріалу, зростає їхня активність та залученість до освітнього процесу, підвищуються показники самомотивації та самооцінки. За умов реалізації такого підходу студенти відчують більше визнання й поваги до власної особистості, демонструють зростання впевненості у власних можливостях, а сам процес навчання набуває більшої змістовності та привабливості [110].

Отже, людиноцентроване навчання відіграє ключову роль у стимулюванні та залученні майбутніх бакалаврів з інженерії програмного забезпечення до активної участі в освітньому процесі. Упровадження інноваційних підходів до викладання і навчання, орієнтованих на досягнення конкретних результатів, сприяє підвищенню ефективності освітнього процесу на якісно новому рівні. Реалізація принципу людиноцентризму у підготовці фахівців до проєктного

управління передбачає врахування потреб ринку праці, активізацію ролі студентів як суб'єктів навчання, урахування їхніх індивідуальних особливостей і здібностей, а також створення умов для досягнення високих результатів навчання та формування необхідних компетентностей.

В рамках цього дослідження реалізація компетентнісного підходу передбачала системний аналіз професійних функцій у сфері проєктного управління, актуальних для стейкхолдерів, а також узгодження відповідних компетентностей і результатів навчання здобувачів вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» з вимогами управлінської діяльності. Компетентнісний підхід забезпечує орієнтацію на результати професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення в контексті формування їхньої готовності до проєктного управління.

Теоретичні та практичні аспекти інтегративного підходу досліджували С. Гончаренко [111], Р. Гуревич [112], Л. Дольнікова [113], І. Козловська [114], Д. Коломієць [115], Я. Собко [116], Т. Якимович [117] та інші.

Інтегративний підхід у підготовці майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління передбачає цілісне поєднання змісту освіти, за якого окремі компоненти професійної підготовки об'єднуються в єдину систему знань, умінь і способів діяльності [118]. Реалізація такого підходу забезпечує формування комплексного уявлення про професійну діяльність у сфері проєктного управління, а також інтеграцію знань із різних освітніх галузей, навчальних дисциплін і практичних напрямів підготовки. Інтеграція в цьому контексті розглядається як процес створення нової системної цілісності, у межах якої взаємодія окремих елементів освітнього процесу забезпечує появу нових професійно значущих якостей і компетентностей. Це сприяє формуванню готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як інтегративного результату професійної підготовки.

Інтегративний підхід також передбачає взаємозв'язок усіх компонентів

освітнього процесу в єдиній дидактичній системі [119], що забезпечує здобувачам вищої освіти можливість набувати комплекс професійних знань, управлінських умінь і практичного досвіду, необхідних для ефективного виконання завдань проєктного управління в ІТ-сфері.

У сучасній педагогічній науці змістова інтеграція нерозривно пов'язується з технологічною інтеграцією, яка передбачає комплексне поєднання методів, форм, засобів і технологій навчання, характерних для різних моделей організації освітнього процесу [120]. Такий підхід забезпечує цілісність професійної підготовки та сприяє узгодженню теоретичної й практичної складових навчання. Дидактична модель у цьому контексті вибудовується на основі логічного та педагогічно обґрунтованого об'єднання знань із різних навчальних дисциплін, що дає змогу формувати інтегровану систему професійних знань і вмінь. Реалізація такої моделі здійснюється через використання інтегрованих форм, методів і технологій навчання, орієнтованих на активізацію пізнавальної діяльності здобувачів освіти та забезпечення практичної спрямованості професійної підготовки.

У межах дослідження інтегративний підхід до підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління розглядається як сукупність взаємопов'язаних форм, методів і засобів навчання, що забезпечують формування готовності до проєктного управління як інтегративного результату професійної підготовки. Така готовність передбачає здатність майбутніх фахівців ефективно розв'язувати складні професійні завдання та практичні проблеми в умовах реалізації ІТ-проєктів.

Інтегративний підхід орієнтує освітній процес на досягнення прогнозованих результатів навчання шляхом поєднання різних видів навчальної, практичної та проєктної діяльності. Його реалізація забезпечує наближення змісту професійної підготовки до реальних умов майбутньої професійної діяльності у сфері проєктного управління в ІТ-галузі. Це зумовлює необхідність розроблення інтегрованого навчально-методичного забезпечення, узгодження змісту освітніх компонентів і

впровадження міждисциплінарних зв'язків.

Важливим етапом реалізації інтегративного підходу є поєднання базових знань, умінь і навичок, сформованих у межах різних навчальних дисциплін, з орієнтацією на професійне становлення майбутніх бакалаврів з інженерії програмного забезпечення як учасників ІТ-проектів. Ефективність такого підходу забезпечується використанням методів проблемного навчання, яке сприяє розвитку аналітичного мислення, здатності до командної взаємодії, прийняття управлінських рішень і застосування знань у професійно наближених ситуаціях.

У процесі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління важливе значення мають міждисциплінарні зв'язки ментально-опосередкованого типу. Їх зміст полягає в узгодженому впливі різних навчальних дисциплін на формування цілісної структури професійної готовності здобувачів освіти. Реалізація таких зв'язків забезпечується через зміст навчальних дисциплін «Економічне обґрунтування ІТ-проектів», «Групова динаміка та командна комунікація», «Фінансовий менеджмент ІТ-проектів» та «Менеджмент проектів з розробки програмного забезпечення». У процесі їх вивчення цілеспрямовано формується єдина система компонентів готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління – функціонального та особистісного. Такий підхід забезпечує інтеграцію теоретичних знань і практичних умінь у цілісну професійну систему, орієнтовану на реальні умови проектної діяльності в ІТ-сфері. Внаслідок цього відбувається не лише накопичення фахових знань, а й розвиток управлінського мислення, комунікативних якостей, здатності до командної взаємодії та прийняття рішень у межах ІТ-проектів, що є визначальними складовими готовності до проектного управління. Найбільш репрезентативними та ефективними способами реалізації інтегративного підходу при структуруванні змісту навчальних дисциплін є застосування проблемного навчання.

Роль проектного підходу в підготовці майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління полягає у

цілеспрямованому впровадженні методів PBL, що забезпечують інтеграцію теоретичних знань і практичної діяльності через розв'язання реальних або наближених до професійної практики завдань. Такий підхід сприяє формуванню здатності до планування, організації та реалізації проєктів, розвитку командної взаємодії, критичного мислення та відповідальності за результати діяльності. Водночас він орієнтує освітній процес на активну участь здобувачів освіти у моделюванні професійних ситуацій, прийнятті управлінських рішень і набутті досвіду проєктної діяльності, що є ключовим для формування їх готовності до проєктного управління.

Отже, підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління має здійснюватися на основі цілісної методологічної системи, що поєднує концептуальні (системний, синергетичний, компетентнісний) та специфічні (інтегративний і проєктний) підходи. Їх комплексне застосування забезпечує узгодженість цілей, змісту, форм і методів навчання, сприяє інтеграції теоретичної та практичної підготовки, а також орієнтації освітнього процесу на досягнення конкретних результатів. Така методологічна основа дозволяє сформувати готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як інтегративну характеристику, що поєднує знання, уміння, досвід діяльності та особистісні якості. Вона передбачає активну позицію здобувачів освіти, їх залучення до розв'язання професійно орієнтованих завдань, розвиток здатності до самостійного мислення, рефлексії та безперервного професійного зростання.

Висновок до розділу 1

У розділі розкрито сутність базових понять дослідження та особливості проєктного управління в контексті інженерії програмного забезпечення; визначено сутність та структуру готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, схарактеризовано критерії, показники та рівні її вияву; обґрунтовано методологічні підходи і принципи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного

управління.

З'ясовано, що специфіка проєктного управління в контексті інженерії програмного забезпечення характеризується високим рівнем невизначеності, зміною вимог, обмеженістю ресурсів і необхідністю ефективної командної взаємодії. Це зумовлює потребу у формуванні в майбутніх бакалаврів з інженерії програмного забезпечення комплексу компетентностей, пов'язаних із проєктним управлінням. Аналіз сучасних методологій розробки програмного забезпечення, зокрема Waterfall та Agile, засвідчив, що ефективна організація ІТ-проєктів потребує поєднання технічних знань із навичками планування, комунікації, оцінювання ризиків, управління ресурсами та координації командної діяльності.

Проведене дослідження вимог роботодавців до фахівців різних рівнів кваліфікації підтвердило високий рівень затребуваності компетентностей, пов'язаних із командною роботою, проєктуванням програмного забезпечення, розробкою архітектури програмних систем та володінням іноземною мовою. Встановлено, що саме компетентності, пов'язані з елементами проєктного управління, найбільшою мірою відповідають сучасним запитам стейкхолдерів ІТ-галузі. Водночас виявлено певні розбіжності між підготовкою майбутніх бакалаврів у закладах вищої освіти та реальними вимогами професійного середовища, що актуалізує необхідність удосконалення освітніх програм і посилення практичної складової професійної підготовки.

Підготовка майбутніх бакалаврів з інженерії програмного забезпечення має бути спрямована на інтеграцію знань із програмної інженерії та проєктного менеджменту, розвиток системного мислення, здатності до роботи в умовах невизначеності, ефективної комунікації та прийняття управлінських рішень. Це забезпечить формування професійної готовності майбутніх фахівців до ефективної діяльності в сучасних ІТ-проєктах і підвищить їхню конкурентоспроможність на ринку праці.

У результаті теоретичного аналізу наукових підходів до трактування поняття «готовність до професійної діяльності» встановлено, що готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління є

складним інтегративним та поліструктурним утворенням, яке поєднує функціональну основу професійної діяльності та особистісні характеристики фахівця. Її структура охоплює систему професійних знань, умінь, навичок і способів діяльності, необхідних для реалізації ключових функцій проєктного управління, а також сукупність індивідуально-психологічних якостей, що забезпечують ефективність професійної взаємодії, прийняття рішень і адаптацію до динамічних умов ІТ-середовища.

Визначено, що функціональна складова готовності передбачає здатність до стратегічного планування, визначення пріоритетів, управління ресурсами, ризиками та змінами, застосування традиційних і Agile-методологій, організації командної взаємодії та забезпечення бізнес-цінності проєкту. Особистісний компонент розкривається через якості моделі «Великої п'ятірки» – відкритість до нового досвіду, сумлінність, екстраверсію, доброзичливість та емоційну стабільність, які визначають рівень готовності майбутніх фахівців до ефективної професійної діяльності в умовах невизначеності та високої інтенсивності проєктної роботи.

Результати констатувального експерименту засвідчили недостатній рівень сформованості готовності студентів ІТ-спеціальностей до проєктного управління. Попри наявність окремих статистично значущих зв'язків між певними особистісними характеристиками та результатами виконання завдань, встановлено низький рівень практичних знань і вмінь у сфері проєктного управління. Це підтверджує існування суперечності між сучасними вимогами ІТ-галузі до професійної підготовки фахівців і недостатнім рівнем теоретико-методичного забезпечення формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління у закладах вищої освіти. У зв'язку з цим актуалізується необхідність розроблення та впровадження педагогічних умов, моделей і методик, спрямованих на цілеспрямоване формування зазначеної готовності в процесі професійної підготовки.

Методологічні підходи до підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління становлять цілісну науково-

педагогічну основу організації освітнього процесу, що забезпечує формування професійної готовності до здійснення управлінської діяльності в ІТ-сфері. Комплексне поєднання системного, синергетичного, компетентнісного, інтегративного та проєктного підходів дає змогу розглядати підготовку майбутніх фахівців як багатовимірний, динамічний і практикоорієнтований процес, спрямований на розвиток не лише професійних знань і вмінь, а й особистісних якостей, необхідних для ефективного управління ІТ-проєктами.

Реалізація системного підходу забезпечує цілісність і структурованість професійної підготовки, синергетичного – створює умови для саморозвитку, самоорганізації та професійного самовдосконалення здобувачів освіти, компетентнісного – орієнтує освітній процес на досягнення результатів навчання та формування професійних компетентностей, інтегративного – сприяє міждисциплінарному поєднанню знань і практичного досвіду, а проєктного – забезпечує набуття досвіду управління проєктною діяльністю в умовах, наближених до реальної професійної практики.

Застосування зазначених підходів у взаємозв'язку сприяє формуванню готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як інтегративної характеристики особистості, що охоплює функціональний і особистісний компоненти. Це забезпечує здатність майбутніх фахівців ефективно здійснювати управлінські функції, працювати в команді, приймати обґрунтовані рішення, адаптуватися до динамічних умов ІТ-галузі та здійснювати безперервний професійний розвиток.

Основні результати розділу відображено в наукових працях автора [121; 122; 123; 124].

РОЗДІЛ 2

МОДЕЛЮВАННЯ ПІДГОТОВКИ МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ

2.1. Організаційно-педагогічні умови підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Галузь програмної інженерії визначається швидкою технологічною еволюцією, зміною ринкових вимог та притаманною проєктам складністю. Історично склалося так, що освіта в галузі програмної інженерії на університетському рівні робила акцент на теоретичних парадигмах програмування та алгоритмічній майстерності, ненавмисно нехтуючи практичними, людиноорієнтованими та управлінськими складнощами розробки програмного забезпечення в реальному світі. Цей постійний педагогічний розрив призводить до того, що випускники мають глибокі технічні навички написання коду, але залишаються погано підготовленими до орієнтування в соціально-технічних реаліях управління проєктами, таких як мінливість вимог, переговори із зацікавленими сторонами та динамічний розподіл ресурсів [125]. Щоб подолати цей розрив, виокремлено низку організаційно-педагогічних умов підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління: *системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проєктної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду*

Процес «підготовки» доцільно розглядати як результат цілеспрямованого

процесу навчання, що охоплює сформовану систему знань, умінь, навичок і практичного досвіду, необхідних для здійснення професійної діяльності [92, с. 417]. Проблемі підготовки майбутніх бакалаврів з інженерії програмного забезпечення присвячено роботи зарубіжних та вітчизняних дослідників, зокрема: Акмальдінова О., Лузік Е. та Теремінко Л. – доведено ефективність змішаного навчання для формування професійної мобільності [126]; Семеріков С., Стрюк А., Стрюк Л., Стрюк М., Шалацька Г. – спроектовано систему загально-професійних компетенцій, спрямованих на формування стійкої професійної компетентності фахівця з інженерії програмного забезпечення [127]; Семеріков С., Стрюк А. – запропоновано ефективні техніки розвитку проектних компетентностей [128]; Назаренко І. – підвищено рівень іншомовної професійної компетентності [129]; Артемчук В., Зубань Ю., Ковач В., Любчак В., Півень А., Попов О., Соколюк О., Шишкіна М., Яцишин А., Яцишин Г. – застосовано технології доповненої реальності для підготовки освітніх проєктів [130] та інші. На думку науковців проблемно-орієнтоване навчання (PBL) входить до низки рекомендованих моделей навчання для викликів XXI століття [131]. PBL забезпечує розвиток критичного мислення, здатності до аналізу та пошуку рішень, творчого підходу до діяльності, умінь командної взаємодії й професійної комунікації. Його застосування сприяє зростанню пізнавального інтересу, формуванню стійкої навчальної мотивації, набуттю професійних умінь, створенню цілісної та гнучкої системи знань, удосконаленню стратегій розв'язання складних завдань і готовності до навчальної, професійної та дослідницької діяльності [132].

PBL забезпечує інтеграцію теоретичних знань із практичною діяльністю в межах освітнього процесу та зарекомендувало себе як результативний дидактичний підхід, що сприяє формуванню відповідальності й ініціативності здобувачів освіти. Цей підхід має конструктивний, самостійний і контекстуально зумовлений характер, що підвищує його потенціал у підготовці студентів до подальшого навчання та майбутньої професійної діяльності.

Зміст PBL зосереджується на розв'язанні реальних, складно структурованих

або міждисциплінарних проблем, навколо яких організовується освітній процес, забезпечуючи цілісне засвоєння навчального матеріалу. Учасники такого навчання активно залучаються до планування, моніторингу й оцінювання власної діяльності, осмислення попереднього досвіду та конструювання індивідуальних систем знань у процесі самостійної й групової роботи.

PBL позиціонує себе як метод, який передбачає евристичний пошук рішення через командні дискусії, де викладач виступає фасилітатором, а не домінує чи керує процесом [133]. Деякі науковці відзначають, що проблемне навчання має певні ризики, які можуть впливати на якість кінцевого результату навчання, зокрема: втрата контролю над вивченням навчальної програми та вразливість до незнайомих тем [134]; поверхнєве засвоєння матеріалу, без глибокого розуміння принципів і алгоритмів [135]; зниження ефективності навчання при роботі з масштабними або комплексними проектами [136]. Ці фактори обмежують корисність підходу та здатність проблемного навчання надавати студентам практичний досвід, що стосується численних критичних концепцій управління проектами.

Отже, попри наявні дослідження та напрацьований досвід застосування проблемного навчання у різних галузях освіти, питання використання проблемних методів навчання у професійній підготовці майбутніх бакалаврів з інженерії програмного забезпечення залишається актуальним та потребує подальшого вивчення.

У контексті реалізації означених завдань одним із провідних дидактичних інструментів виступає метод аналізу конкретних ситуацій (кейс-метод). Затребуваність case-технологій у системі професійної підготовки детермінована їхнім високим потенціалом щодо формування навичок групової інтерактивної взаємодії, що є критично необхідним для професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління. Це сприяє інтенсифікації розвитку аналітико-комунікативних здібностей здобувачів освіти, формуванню міцної культури фахової полеміки, здатності до обґрунтованої аргументації власної позиції, а також консолідації зусиль у проектних командах для оптимізації розв'язання комплексних проблемних ситуацій.

Концептуальна сутність зазначеного підходу ґрунтується на дидактичній трансформації традиційного навчального контенту у формат професійно орієнтованих ситуаційних завдань (проблемних історій, казусів, інцидентів з управлінської практики). У цьому вимірі набуття необхідних компетентностей відбувається не шляхом репродуктивного засвоєння, а через включення студентів в активну пізнавальну та евристичну діяльність. Відповідний процес має системний характер і охоплює акумуляцію релевантної інформації, її багатовимірний спільний аналіз, ідентифікацію наявних суперечностей, висунення науково-практичних гіпотез та проєктування алгоритмів дій.

Окрім того, відповідний формат роботи передбачає критичне оцінювання альтернативних варіантів, формулювання самостійних висновків, прийняття обґрунтованих рішень і здійснення самоконтролю власної діяльності [137; 138]. Емпіричне підтвердження результативності такого дидактичного інструментарію знаходимо у працях сучасних науковців. Зокрема, досвід імплементації case-study у підготовку фахівців споріднених технічних напрямів (на прикладі спеціальності «Професійна освіта: енергетика, електротехніка та електромеханіка») засвідчує, що цей метод генерує унікальне синергетичне освітнє середовище. Воно об'єднує академічну групу навколо розв'язання спільного практико-орієнтованого завдання, що, у свою чергу, каталізує пізнавальну активність особистості [88]. Дослідники констатують, що інтеграція ситуаційних вправ у навчальний процес суттєво інтенсифікує набуття прикладного досвіду та підвищує загальну ефективність формування фахової компетентності. Крім того, систематична взаємодія з проблемними ситуаціями вчить здобувачів здійснювати багатофакторний аналіз, ухвалювати автономні рішення з повним усвідомленням міри відповідальності за них, результативно функціонувати в команді, а також толерантно сприймати критику, зберігаючи при цьому здатність аргументовано захищати власну професійну стратегію [88. с. 11].

Завдання дослідження полягає у комплексному аналізі методів проблемного навчання в системі професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що передбачає: подання

стислої характеристики кожного методу; представлення прикладів його практичного застосування в освітньому процесі; визначення впливу відповідного методу на результати підготовки у контексті формування функціональної та особистісної складових готовності до проєктного управління.

Інтеграція проблемно-орієнтованого навчання (PBL) та методу кейсів занурює студентів у симульоване професійне середовище, де вони повинні організувати технічну реалізацію в умовах постійної невизначеності [139]. Коли ці парадигми активного навчання поєднуються з критеріально орієнтованою трибальною системою оцінювання, освітній фокус зміщується з механічного запам'ятовування та накопичення балів на цілісне оволодіння компетенціями [140]. Ця комплексна архітектура навчальної програми окреслює надійну структуру, призначену для підготовки майбутніх бакалаврів з програмної інженерії до багатогранної галузі управління проєктами. Подальший аналіз синтезує теоретичні основи, структурний дизайн навчальної програми, інтеграцію тематичних досліджень, передові стратегії фасилітації та суворі протоколи оцінювання для створення дієвого плану академічного впровадження.

У контексті програмної інженерії, PBL функціонує як конструктивістська парадигма навчання, де невеликі групи студентів беруть участь у спільному навчанні для вирішення справжніх, погано структурованих проблем [141]. Замість пасивного отримання інформації, студентам пропонується складний проєктний сценарій, такий як розробка модульної платформи електронної комерції, програми відстеження в режимі реального часу для логістичної фірми або інструменту забезпечення доступності транспорту на основі штучного інтелекту, який слугує основним каталізатором для всього подальшого навчання [142].

Когнітивний виклик, властивий PBL, змушує студентів проактивно виявляти прогалини в знаннях, проводити самостійні дослідження та синтезувати технічні рішення, одночасно керуючи складною командною динамікою [143]. Це відображає життєвий цикл гнучкої розробки програмного забезпечення, де міжфункціональні команди повинні ітеративно надавати робоче програмне забезпечення, адаптуючись до змінних вимог та інтегруючи постійний зворотний

зв'язок із зацікавленими сторонами [144]. Педагогічні наслідки такого підходу свідчать про те, що, ставлячи проблему вище за навчання, студенти розвивають значно вищу толерантність до неоднозначності. Ця психологічна стійкість є критично важливою в програмній інженерії, де фахівці працюють у «нестабільному середовищі» – сферах, де бракує чітких шаблонів, встановлених меж або механізмів негайного зворотного зв'язку [145].

Крім того, PBL природним чином розвиває підприємницькі та управлінські навички. Студенти повинні самостійно визначати обсяг системи, розподіляти обмежені ресурси та домовлятися про необхідні компроміси щодо часу, вартості та якості продукту [146]. Метод формування команд «слабкий-сильний», який навмисно об'єднує академічно сильніших студентів з їхніми слабшими однолітками, покращує середовище PBL, сприяючи екосистемам самонавчання, які зменшують залежність від інструктора та імітують динаміку наставництва, що спостерігається в професійних інженерних командах [142].

У той час як PBL керує технічним виконанням та внутрішнім управлінням проектом, метод кейсів розвиває аналітичні навички та навички прийняття стратегічних рішень на виконавчому рівні. Інтеграція методу кейсів в освіту з програмної інженерії відповідає на критичну потребу інженерів у розумінні глибокої бізнес-цінності та стратегічних наслідків їхніх технічних рішень. Аналізуючи випадки, що стосуються невдалих Agile-трансформацій, катастрофічних перевитрат бюджету, порушень нормативних вимог або критичних помилок програмного забезпечення, студенти змушені поставити себе на місце керівників проектів, головних технічних директорів або провідних архітекторів [147]. Вони повинні діагностувати першопричини, пропонувати практичні рішення та захищати свої міркування від критики колег.⁴

Розвиваючим ефектом дискусії на основі конкретних випадків є швидке культивування професійної емпатії та ефективності комунікації. Обговорюючи суперечливі точки зору, студенти дізнаються, що технічна перевага рідко гарантує успіх проекту, якщо очікування зацікавлених сторін неправильно керуються, ризики ігноруються або організаційна культура нехтується. Метод кейсів навчає

студентів «обходити проблему», розглядаючи розробку програмного забезпечення не лише як технічну перешкоду, а й як складне соціально-економічне починання.

Поєднання проблемно-орієнтованого навчання та методу кейсів було реалізовано шляхом розроблення та впровадження в освітній процес майбутніх бакалаврів з інженерії програмного забезпечення експериментального навчального курсу (див. дод. Г) розрахований на період стандартного 15-тижневого академічного семестру. Курс працює за двосторонньою системою: поздовжній груповий проєкт з розробки програмного забезпечення, що діє як компонент PBL, та щотижневі інтерактивні семінари, присвячені аналізу конкретних галузевих кейсів.

Поздовжній проєкт PBL вимагає від студентів концептуалізації, проєктування, створення та розгортання функціонального програмного артефакту від початку до кінця [125]. Наприкінці семестру студенти теоретично проаналізують найкращі практики за допомогою тематичних досліджень та одразу ж застосують їх до власного складного процесу розробки в реальному світі.

Ефективність методу кейсів повністю залежить від якості, складності та актуальності наданих наративів. Щоб підготувати студентів до надзвичайної непередбачуваності сучасного управління проєктами, навчальна програма активно використовує ключовий тематичний аналіз, що досліджує безпрецедентну стійкість та стратегічне маневрування українського ІТ-сектору під час повномасштабної війни, що тривала з 2022 по 2026 рік. Цей кейс слугує майстер-класом з екстремального управління ризиками, децентралізованої оркестрації команд та планування безперервності бізнесу.

Цей кейс знайомить студентів із суворою реальністю української технологічної екосистеми – величезної галузі, що охоплює понад 340 000 технічних спеціалістів, яка у 2024 році згенерувала оборот у розмірі 7,48 мільярда доларів та експортувала 6,45 мільярда доларів, незважаючи на роботу в зоні бойових дій [148]. Студентам доручено проаналізувати дії великих вітчизняних фірм, таких як ІТ-Integrator, та дослідити, як виконавче керівництво перевело організації від початкового хаосу до структурованої стійкості.

У розповіді детально описується впровадження проактивних Планів забезпечення безперервності бізнесу (ПББ), які підтримували такі лідери, як віцепрезидент Надія Омельченко, яка боролася проти небажання виконавчої влади виділяти ресурси на випадок воєнних подій наприкінці 2021 року [149]. Студенти оцінюють логістичні складнощі географічного перерозподілу, оскільки компанії швидко переміщували персонал та обладнання до безпечніших західних центрів, таких як Львів, перетворюючи регіон на зміцнений технологічний кластер. Крім того, у кейсі досліджується стратегічний поворот до військових технологій (MilTech), аналізуючи, як українські фірми швидко розробили пристрої зв'язку, захищені IP-адресою, програмні модулі для автономних дронів та оборонні системи на базі штучного інтелекту для підтримки національної безпеки.

У цьому випадку студенти стикаються з жахливими сценаріями, що включають періодичні відключення електроенергії, раптову військову мобілізацію ключових старших інженерів та абсолютну необхідність надійних асинхронних протоколів зв'язку. Педагогічна цінність цього підходу є надзвичайно високою, оскільки він сприяє деконструкції академічної ілюзії існування «ідеального плану проекту». Студенти дізнаються, що гнучкі методології – це не просто зручні рамки розробки програмного забезпечення, а фундаментальні механізми виживання організації. Цей кейс спонукає студентів оцінити, як саме ці компанії підтримували вражаючий рівень утримання контрактів у 95%, успішно постачаючи складне програмне забезпечення клієнтам по всьому світу, в той час як їхня фізична інфраструктура перебувала під постійною загрозою [148].

У навчанні, що базується на PBL та методі кейсів, роль викладача принципово зміщується на стратегічного посередника навчання [150]. Ефективна фасилітація вимагає делікатного, високочутливого балансу педагогічних, управлінських, соціальних та технічних втручань, які часто контролюються за допомогою технологій оркестрації та панелей інструментів для вчителів, що відстежують прогрес команди [151].

Основним і найпотужнішим інструментом фасилітатора є сократівські питання. Замість того, щоб давати прямі відповіді на глухі кути кодування чи

архітектурні дилеми, фасилітатор використовує цілеспрямовані запитання, щоб виявити основні припущення, виділити логічні суперечності та спрямувати студентів до самокорекції [152]. Врахування закономірностей пізнавальної діяльності та логіки освітнього процесу дозволяє послідовно формувати знання, уміння та навички, необхідні для високого рівня професійної готовності майбутніх бакалаврів з інженерії програмного забезпечення у сфері проєктного управління. Між складовими освітнього процесу існує діалектичний взаємозв'язок, тому його моделювання на будь-якому рівні має враховувати особливості цієї структури. Зокрема, можна виділити такі етапи: сприймання – початковий етап, результатом якого є мотивація пізнавальної діяльності; розуміння – глибоке осягнення сутності явищ і процесів та успішне запам'ятовування, що сприяє формуванню свідомості, розвитку інтелектуальних здібностей і пізнавального інтересу; узагальнення та систематизація – процес виділення і об'єднання окремих властивостей предметів і явищ у цілісну систему, результатом якого стає формування системи знань, умінь і навичок; застосування – реалізація знань, умінь і навичок на практиці, що забезпечує набуття досвіду та підкріплює мотиваційну складову навчальної діяльності.

Серед методів навчання пріоритет віддається методам проблемного навчання, які стимулюють самостійну розумову діяльність студентів та інтегруються з методами навчання за джерелом знань. Методи проблемного навчання сприяють формуванню зацікавленості у майбутніх бакалаврів з інженерії програмного забезпечення до опанування своєї спеціальності, підвищують ефективність засвоєння знань та підтримують ранню професійну орієнтацію. Залежно від рівня розумової активності студентів виділяють кілька основних підходів до проблемного навчання. *Проблемно-інформаційний метод* передбачає активізацію мислення через створення проблемних ситуацій, що дозволяє студентам виділити та прийняти проблемне завдання. *Частково-пошуковий метод* ґрунтується на залученні студентів до розв'язання пізнавальних задач, у результаті чого вони усвідомлюють завдання та набувають відповідного досвіду. *Дослідницький метод* передбачає самостійне виконання пізнавальних завдань із

використанням необхідного обладнання, що сприяє систематизації та узагальненню знань, їх закріпленню і практичному застосуванню, а також розвитку навичок самостійної навчально-пізнавальної діяльності.

Найбільш доступним і водночас ефективним засобом створення проблемної ситуації під час лекційного заняття є використання педагогічно виважених запитань. Їх основною функцією є спонукання студентів до усного чи внутрішнього мисленнєвого реагування. Результативність цього методу залежить від змісту, форми й послідовності формулювання запитань. Вдало поставлені запитання активізують інтелектуальну діяльність студентів, сприяють діалогізації навчального процесу та дають викладачеві можливість отримати інформацію про рівень розуміння матеріалу.

Запитання дозволяють підтримувати динаміку навчальної бесіди, стимулювати обмін думками, аналізувати зміст теми та формувати внутрішню мотивацію до її опанування. Для цього необхідно формулювати їх з урахуванням конкретної дидактичної мети: активізувати увагу, поглибити аналіз, викликати пізнавальний інтерес або забезпечити логічне усвідомлення ключових понять. Доброзичливе ставлення викладача до відповідей студентів сприяє відкритості їхніх висловлювань і дозволяє точніше оцінити особливості їх мислення.

Навідні запитання доцільно використовувати для введення ключових понять і спрямування мислення студентів у потрібному напрямі. Особливо ефективними є запитання, відповідь на які не є очевидною, що пробуджує інтелектуальну зацікавленість. Короткі паузи після запитання також підсилюють увагу аудиторії. Під час аналізу складних положень використовують запитання, що допомагають студентам співвідносити нові ідеї з раніше засвоєними, встановлювати логічні зв'язки та бачити практичну значущість знань, зокрема для майбутньої професійної діяльності.

У процесі роботи важливо враховувати, що відповіді студентів не завжди відображають їхні справжні погляди, тому уточнювальні запитання («Чи згодні Ви з цим?», «Яка Ваша думка?») допомагають з'ясувати реальне ставлення. Крім того, запитання можуть виконувати функцію підсилення певної думки або акцентування

на ключових аспектах матеріалу. Ретельно продумана система запитань є ефективним інструментом виявлення помилок у міркуваннях та корекції пізнавальної діяльності студентів.

Фасилітатори використовують кілька різних, чітко структурованих методів запитань для стимулювання дискусії:

1) Альтернативні досягнення: Змушення команд зважувати конкретні, складні компроміси (наприклад, «Враховуючи наближення терміну виконання спринту, чи є більш стратегічним повністю відмовитися від функції платіжного шлюзу чи взяти на себе технічний борг жорстко запрограмованого, немасштабованого рішення?»).

2) Бумеранг питань: Перенаправлення запитання студента назад до групи для сприяння досягненню колективного консенсусу, а не опори на викладача (наприклад, «Це критичне питання щодо масштабованості бази даних. Які, на думку решти команди, наші обмеження?»).

3) Корекція настрою: Формулювання питань для зменшення захисної реакції під час делікатних перевірок коду або критики дизайну. Замість агресивного запитання «Чому ви обрали саме цю архітектуру?», фасилітатор запитує: «Які конкретні обмеження проекту спонукали команду віддати перевагу монолітному підходу над мікросервісами для цього модуля?».

4) Відгук: Перевірка та впорядкування думок студентів для ненав'язливого впливу на напрямок дискусії (наприклад, «Я чув, як деякі з вас згадували про ризики безпеки API, але, схоже, цю тему було виключено з беклогу спринту. Давайте переглянемо це рішення»).

Отже, використання запитань у проблемному навчанні вимагає дотримання низки методичних принципів: перед поданням важливого положення необхідно поставити запитання, що формує інтерес до його змісту; запитання мають сприяти розумінню логічної обґрунтованості навчального матеріалу та його значущості для майбутньої професійної діяльності; доцільно формулювати запитання, які спонукають студентів не лише відтворювати інформацію, а й висловлювати власні судження та інтерпретації. Фасилітатор має створювати постійну «можливість

зростання», вимагаючи від студентів глибокої інтелектуальної відповідальності за свій проєкт та обґрунтування своїх рішень емпіричними доказами.

У проблемно-орієнтованому навчанні дисфункція команди не є провалом курсу; це очікувана та дуже цінна можливість навчання. Фасилітатори повинні активно контролювати співпрацю в групі, виявляючи випадки, коли сильні особистості домінують у технічному напрямку або коли слабші студенти ухиляються від навчального навантаження [142]. Втручання має бути мінімальним, але рішучим. Якщо команда стикається зі збоєм у своєму Scrum-процесі або страждає від токсичної комунікації, фасилітатор може призупинити технічну роботу, щоб провести керовану ретроспективу, використовуючи методи активного слухання, щоб переконатися, що всі члени команди висловлюють свої занепокоєння та спільно встановлюють нові основні правила.

Крім того, фасилітатори повинні переконливо виступати в ролі «клієнта» під час фаз збору та перевірки вимог. Навмисно вводячи зміни обсягу робіт на пізніх стадіях, розпливчасті специфікації або змінюючи бізнес-пріоритети, фасилітатор достовірно імітує непостійність у вимогах від реальних клієнтів з розробки програмного забезпечення. Це змушує студентів активно практикувати узгодження вимог, адаптацію до ризиків та складне мистецтво протистояння неможливим вимогам [153].

Традиційні 100-бальні шкали оцінювання або стандартні літерні оцінки від А до F часто виявляються неадекватними для оцінювання складних, багатовимірних результатів PBL та курсів, заснованих на конкретних випадках. Надто деталізовані системи оцінювання можуть ненавмисно заохочувати шкідливу поведінку «збирання балів», коли студенти недалекоглядно зосереджуються на задоволенні дрібниць рубрики, а не глибоко занурюються в проблемне поле. Щоб протидіяти цьому явищу, навчальна програма використовує надійну трибальную систему оцінювання, яка відображає рівні компетентності, що орієнтовані на критерії [154]. 3-бальна шкала, яка зазвичай відповідає як «незадовільно», «задовільно» та «відмінно» – ідеально узгоджується з бінарними реаліями розгортання програмного забезпечення: програмне забезпечення або функціонує відповідно до

вимог і забезпечує бізнес-цінність, або ні.

Перехід до 3-бальної системи значно зменшує когнітивне навантаження на викладачів, пов'язане з оцінюванням, та зменшує тривожність студентів. Це заохочує студентів йти на творчий ризик у своєму дизайні програмного забезпечення та виборі архітектури без паралізуючого страху втратити частку відсотка через незначну стилістичну помилку. Для забезпечення повної прозорості та об'єктивності в рамках 3-бальної системи оцінювання суворо регулюється аналітичною рубрикою. Аналітичні рубрики розбивають оцінювання на чіткі, вимірювані критерії, зіставляючи кожен критерій з трьома рівнями досягнень (див. дод. Г).

Щоб уникнути жорсткого оцінювання, яке іноді супроводжує жорсткі рубрики, оцінювання використовує метод медіанного балу. Замість розрахунку суворого математичного середнього значення, яке дозволяє одному поганому результату в меншій категорії непропорційно знижувати оцінку студента, викладач визначає медіанний рівень успішності за всіма критеріями, щоб визначити остаточну цілісну оцінку [155].

Постійною структурною проблемою проблемного навчання є забезпечення справедливого оцінювання у спільних групових проєктах. Викладачі повинні уникати сценаріїв, коли слабші студенти «користуються перевагами» інтенсивних зусиль своїх сильніших однолітків, або коли відмінника тягне вниз дисфункціональна команда. Щоб вирішити цю нерівність у рамках 3-бальної системи, оцінювання використовує розподіл 70/30: 70% остаточної оцінки проєкту базується на колективному результаті команди (програмний продукт, архітектура та документація), тоді як 30% базується виключно на індивідуальному внеску [142].

Індивідуальний внесок визначається за допомогою високоефективного методу взаємної оцінки «Розподіл». Кожному студенту призначається концептуальний пул зі 100 балів, які він повинен розподілити між своїми однолітками на основі свого сприйняття якості, послідовності та кількості внеску своїх товаришів по команді. Щоб запобігти етичним дилемам або

самозвинуваченням, студенти зобов'язані призначити собі рівну, математично визначену частку (наприклад, 25 балів у команді з 4 осіб), але їхня самооцінка повністю виключається з остаточного розрахунку [156]. Сукупні оцінки колег надають викладачеві математично точний механізм для коригування остаточної оцінки окремого співробітника. Цей метод дозволяє викладачеві впевнено знизити оцінку члена команди з «Зараховано» до «Не зараховано», якщо дані колег доводять, що він не зробив жодного внеску, або навпаки, підвищити оцінку виснаженого, але блискучого учасника з «Зараховано» до «Зараховано з відзнакою».

Хоча практичні групові проекти та усні аналізи тематичних досліджень є основою оцінювання, об'єктивне індивідуальне тестування залишається цінним інструментом для оцінки теоретичного засвоєння принципів управління проектами. Однак традиційні іспити з механічним запам'ятовуванням та вибором відповідей принципово суперечать конструктивістському духу управління проектами. Натомість тести повинні бути суворо сценарно-орієнтованими, що вимагає високого рівня застосування знань для вирішення дилем ситуаційного управління.

Тестові питання структуровані таким чином, щоб ідеально імітувати стресові складнощі управління програмною інженерією. Оцінювання цих тестів безпосередньо відповідає 3-бальній шкалі: досягнення точності менше 60% у сценаріях призводить до оцінки «незараховано»; 60%-84% – до «зараховано»; а 85% або вище – до «зараховано з відзнакою» [157]. Приклади питань ілюструють глибину оцінювання, необхідну для оцінки функціональної готовності студента до управління проектами (див. дод. Г). Переплітаючи тести, засновані на сценаріях, із цілісними оцінками рубрик поздовжнього групового проєкту, навчальний курс забезпечує вичерпну, багатогранну оцінку справжньої готовності студента до керівництва складними програмними проєктами.

Завдяки цій комплексній інтеграції проблемно-орієнтованого навчання, методу кейсів та ретельно налаштованої 3-бальної системи оцінювання, університети можуть подолати постійний розрив між академічною теорією та

галузевою реальністю. Така архітектура навчального курсу гарантує, що випускники вийдуть на ринок праці не просто як здібні програмісти, а проєктів, готові орієнтуватися у величезних складнощах сучасного технологічного ландшафту.

Важливою особливістю проблемного навчання є трансформація мотиваційної сфери пізнавальної діяльності студентів: провідну роль починають відігравати пізнавально-спонукальні (інтелектуальні) мотиви. Інтерес до навчання формується у зв'язку з виникненням проблемної ситуації та розгортається в процесі інтелектуальної діяльності, спрямованої на пошук і знаходження способів розв'язання проблемного завдання або їх сукупності. На цій основі формується внутрішня вмотивованість, яка виступає чинником активізації навчального процесу та підвищення його результативності. Пізнавальна мотивація, у свою чергу, стимулює майбутніх бакалаврів з інженерії програмного забезпечення до розвитку власних здібностей і потенційних можливостей.

Підґрунтям для створення навчальної проблеми виступає спеціально організована викладачем проблемна ситуація, у яку включаються студенти в процесі навчання. Сутність такої ситуації полягає в тому, що для її розв'язання здобувачам освіти необхідно відшукати та застосувати нові знання, а також опанувати відповідні способи дій. Продуктивна пізнавальна діяльність майбутніх бакалаврів з інженерії програмного забезпечення в умовах проблемної ситуації реалізується через послідовність таких етапів: опис проблемної ситуації, визначення проблеми, пошук можливих способів її розв'язання, вибір найбільш ефективного шляху вирішення та аналіз отриманих результатів.

У процесі навчання створюються різні типи пізнавальних ситуацій, серед яких центральне місце займають проблемні ситуації як ключовий елемент проблемного викладу. Проблема ситуація визначається як інтелектуальне утруднення, що виникає внаслідок неможливості досягнення навчальної мети за допомогою наявних знань і усталених способів дії. Вона має характер пізнавального конфлікту та слугує потужним стимулом активізації навчальної діяльності студентів. Важливою її особливістю є те, що вона виникає

безпосередньо в процесі навчання, а не пропонується у завершеному вигляді, що сприяє підвищенню пізнавальної активності, формуванню інтересу й особистісного ставлення до навчального матеріалу.

Під час організації проблемного навчання цикл роботи будується за такими кроками [158]: презентація проблеми; ідентифікація проблеми; мозковий штурм та побудова гіпотез; спроба вирішити проблему доступними знаннями; виявлення / пріоритезація навчальних проблем; планування роботи в групі; самонавчання; поширення знань в межах групи; використання нових знань для вирішення оригінальної проблеми; презентація результатів; оцінювання (само- та групове) (рис. 2.1).

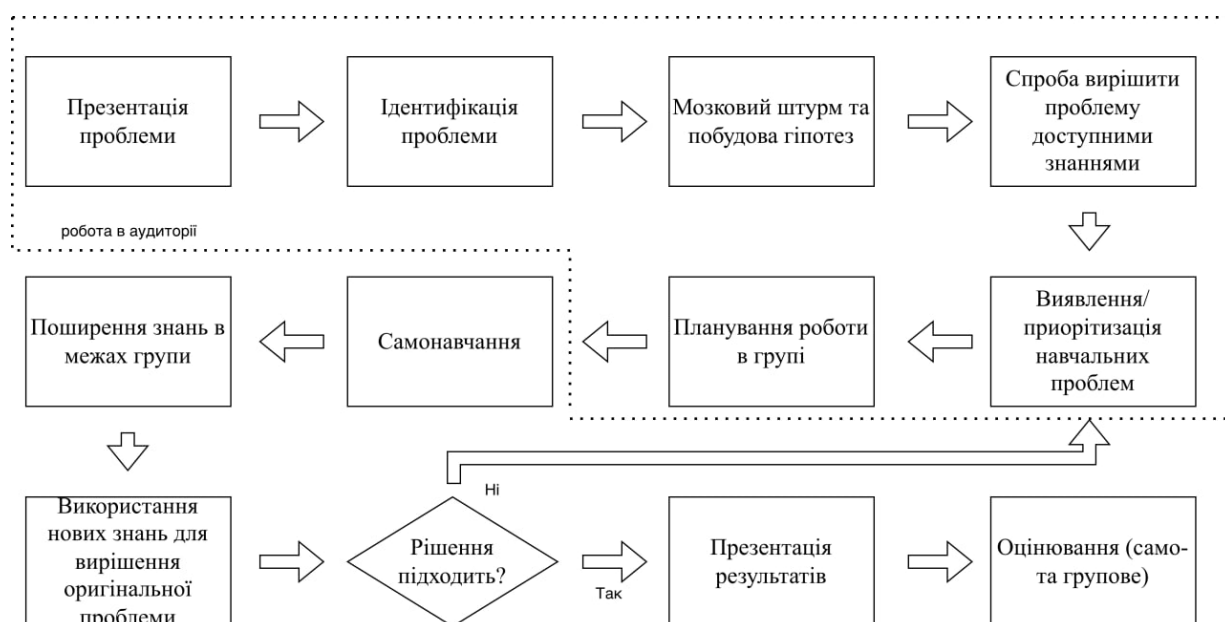


Рис. 2.1 Цикл роботи PBL

Функціональну основу готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління визначено таким чином:

- а) планування та ретельне визначення пріоритетів (проєктних та персональних);
- б) максимізація бізнес-цінності проєкту за рахунок правильно створеної стратегії його реалізації;
- в) співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту;
- г) використання як традиційних, так і *Agile* інструментів, техніки та методів для кожного проєкту;
- г) пояснення важливих бізнес-аспектів проєкту іншим;
- д) керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та

ризиками; е) увага до критично важливих технічних елементів проєктного управління.

Задля забезпечення реалізації функціональної основи необхідним є системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення

Оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії реалізовано на основі таких освітніх компонент: «Економічне обґрунтування ІТ-проєктів» (8 семестр), «Групова динаміка та командна комунікація» (6 семестр), «Фінансовий менеджмент ІТ-проєктів» (7 семестр), «Менеджмент проєктів з розробки програмного забезпечення» (7 семестр). Кожна з освітніх компонент спрямована на формування окремих складових готовності. Застосування проблемно-орієнтованого навчання в межах кожної з них дало змогу продемонструвати майбутнім бакалаврам з інженерії програмного забезпечення типові труднощі та специфічні виклики, притаманні кожному аспекту функціональної складової їхньої професійної готовності.

У процесі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління доцільно застосовувати комплекс форм практичної діяльності, спрямованих на формування проєктно-управлінських компетентностей та розвиток здатності до розв'язання професійних проблемних ситуацій. До таких форм належать: семінари-практикуми, семінари дискусії, практичні заняття, лабораторні роботи, самостійна робота.

Семінари-практикуми, орієнтовані на аналіз і колективне обговорення варіантів розв'язання ситуаційних завдань з управління ІТ-проєктами (розподіл ролей, визначення критичного шляху, оцінювання ризиків, вибір методології), що забезпечує вироблення аргументованих рішень під керівництвом викладача.

Семінари-дискусії, присвячені критичному аналізу методів, підходів і інструментів проєктного управління (Agile-практики, моделі зрілості процесів, методи оцінювання трудомісткості), де студенти засвоюють способи добору та

адаптації управлінських інструментів до конкретних умов розроблення програмного забезпечення.

Практичні заняття, що проводяться як у межах закладу освіти, так і на базі практики, метою яких є оволодіння способами розв'язання типових професійних завдань проєктного менеджера в ІТ-сфері (планування спринтів, створення документації, комунікація з командами та стейкхолдерами).

Лабораторні роботи, спрямовані на моделювання професійних ситуацій та відпрацювання навичок роботи з інструментами управління проєктами (системами трекінгу завдань, засобами контролю версій, програмами для оцінювання ризиків та ресурсного планування).

Самостійна робота студентів, що виступає активною формою індивідуальної та групової діяльності, спрямованої на закріплення знань, оволодіння навичками аналізу проєктних вимог, прийняття рішень в умовах невизначеності та формування готовності до самостійного виконання управлінських функцій у майбутній професійній діяльності.

Комплексне застосування зазначених форм забезпечує поступове формування професійної компетентності у сфері проєктного управління та сприяє підвищенню ефективності підготовки майбутніх бакалаврів з інженерії програмного забезпечення.

Отже, у результаті проведеного дослідження теоретично обґрунтовано доцільність і педагогічну ефективність застосування проблемно-орієнтованого навчання у професійній підготовці майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Доведено, що цілеспрямоване впровадження методів PBL сприяє формуванню інтегрованої професійної готовності, яка поєднує когнітивну, практичну та особистісну складові компетентності, необхідні для успішної діяльності в умовах сучасної ІТ-індустрії.

Встановлено, що проблемне навчання забезпечує подолання розриву між теоретичною підготовкою здобувачів освіти та реальними вимогами професійної діяльності, активізує пізнавальну діяльність студентів, підвищує їхню мотивацію, відповідальність і здатність до самостійного прийняття рішень у складних і

невизначених проєктних ситуаціях. Особливу роль у цьому процесі відіграє використання проблемних ситуацій, евристичних запитань, командної роботи та міждисциплінарних завдань, що моделюють реальні умови проєктного управління.

Обґрунтовано, що системне оновлення змісту освітніх компонент і поєднання різних форм практичної діяльності (семінарів-практикумів, дискусій, практичних і лабораторних занять, самостійної роботи) на засадах проблемно-орієнтованого навчання забезпечує поступове та цілісне формування проєктно-управлінських компетентностей майбутніх фахівців.

Отримані результати підтверджують, що проблемне навчання, за умови його методично виваженого поєднання з іншими педагогічними підходами, є ефективним засобом підвищення якості підготовки бакалаврів з інженерії програмного забезпечення та їх конкурентоспроможності на сучасному ринку праці.

2.2. Структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Підготовку майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління розуміємо як цілеспрямований, системно організований процес професійної освіти, спрямований на формування готовності здобувачів освіти до здійснення проєктно-управлінської діяльності в ІТ-сфері, що охоплює засвоєння знань, умінь, навичок, практичного досвіду та розвиток особистісних якостей, необхідних для планування, організації, координації та реалізації програмних проєктів в умовах невизначеності, командної взаємодії та змінних вимог сучасної ІТ-індустрії.

Ускладнення змісту професійної діяльності майбутніх бакалаврів, поява нових підходів, інструментів і форматів управління проєктами, а також зростання ролі особистісних якостей у проєктній взаємодії зумовлюють потребу в оновленні науково обґрунтованих підходів до організації їхньої підготовки.

У цьому контексті актуалізується необхідність розробки новітньої моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до

проектного управління, здатної забезпечити цілісне формування їхньої готовності до проектної діяльності. Моделювання як науковий метод є об'єктивно зумовленим інструментом педагогічних досліджень, оскільки, з одного боку, воно дозволяє адекватно відобразити ускладнення професійної підготовки, стрімкі зміни змісту діяльності та вимог до майбутніх фахівців, а з іншого – забезпечує системне осмислення й упорядкування трансформацій усєї системи підготовки фахівців у сучасному освітньому й професійному просторі. Саме це зумовлює потребу в розробленні структурно-функціональної моделі, яка інтегрує теоретичні та практичні засади підготовки й відповідає актуальним вимогам проектного управління в галузі інженерії програмного забезпечення.

У наукових джерелах термін «модель» трактується як умовне відтворення у вигляді схеми, образу або опису певного явища чи процесу в природі або суспільстві, що слугує засобом аналізу окремого фрагмента чи аспекту соціальної реальності [159]; спрощений опис складного явища або процесу [43]; смислово представлена і матеріально реалізована система, яка адекватно відображає предмет дослідження [160]. Педагогічна модель розглядається як опис або відтворення певного педагогічного явища чи процесу, що використовується як його аналог. Водночас наукова модель трактується як уявно чи матеріально сконструйована система, яка адекватно відображає предмет дослідження та здатна замінити його в такій мірі, що аналіз моделі дає змогу отримати нову інформацію про досліджуваний об'єкт [161].

Як інструмент наукового пізнання моделі реалізують три ключові функції: описову, яка полягає в упорядкуванні та систематизації емпіричних даних; пояснювальну, що спрямована на встановлення взаємозв'язків між фактами, виявленими в процесі опису, залежностями та вже відомими законами, теоріями й гіпотезами; а також прогностичну, що передбачає прогнозування нових, раніше невідомих властивостей і відношень об'єкта моделювання [162].

Отже, структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління має відтворювати освітній процес у контексті досягнення запланованих результатів, а саме

сформованих властивостей, характеристик і взаємозв'язків, необхідних для здійснення професійно спрямованої діяльності.

Моделювання розглядається як метод наукового пізнання, що ґрунтується на побудові та дослідженні моделей і полягає в заміні об'єкта вивчення його моделлю з метою визначення або уточнення характеристик оригіналу, явищ і процесів, які він відображає [163]. Метод моделювання має наступну структуру: а) постановка завдання; б) створення або вибір моделі; в) дослідження моделі; г) перенесення знань з моделі на оригінал. Означений метод ґрунтується на синтетичному підході, тобто дозволяє виділяти цілісні системи і досліджувати їх функціонування. Головна його перевага – цілісність інформації [161].

За характером моделей виділяють предметне й знакове моделювання. Предметним називається моделювання, у ході якого дослідження може здійснюватися на моделі, що відтворює певні функціональні характеристики об'єкта-оригіналу. При знаковому моделюванні моделями служать схеми [163].

Вітчизняними науковцями розроблено низку моделей підготовки фахівців галузі ІТ: модель організації комбінованого навчання системного програмування бакалаврів програмної інженерії [165]; структурна модель системи підготовки майбутніх інженерів-програмістів до професійної діяльності [166]; модель методичної системи технічної підготовки майбутніх фахівців з інформаційних технологій [167] та інші. Однак зазначені наукові напрацювання переважно зосереджені на формуванні професійних, технічних або предметно-спеціалізованих компетентностей і не в повній мірі враховують специфіку цілеспрямованої підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління як інтегрованого виду професійної діяльності.

Модель дає уявлення про цілісний зміст і структуру процесу підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління, розкриваючи взаємозв'язок і взаємозалежність його складників. У цьому контексті розробка структурно-функціональної моделі здійснюється на основі визначених вихідних положень, що забезпечують системність, логічну

впорядкованість і наукову обґрунтованість процесу підготовки.

Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління полягає у створенні умов для стимулювання та розвитку функціональних та персональних якостей. До першої групи відносимо: планування та ретельне визначення пріоритетів; максимізація бізнес-цінності проєкту за рахунок правильно створеної стратегії його реалізації; співпраця зі спонсором проєкту, командою та експертами з тематики, яка необхідна для розробки стратегії реалізації проєкту; використання як традиційних, так і agile інструментів, техніки та методів для кожного проєкту; пояснення важливих бізнес-аспектів проєкту іншим; керування елементами проєкту, включаючи, але не обмежуючись графіком, вартістю, ресурсами та ризиками; увага до критично важливих технічних елементів проєктного управління. До другої групи: відкритість досвіду; сумлінність; екстраверсія; доброзичливість; невротизм.

Метою створення структурно-функціональної моделі є вдосконалення професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення через формування готовності до проєктного управління. Отже, об'єктом розробки є процес підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, який моделюється в межах структурно-функціональної моделі та оцінюється за результатами сформованості їхньої готовності до здійснення проєктної діяльності.

Структурно-функціональна модель зорієнтована на використання в закладах вищої освіти, що впроваджують освітню діяльність для здобувачів вищої освіти першого (бакалаврського) рівня зі спеціальності 121 «Інженерія програмного забезпечення» [22]. Вона ґрунтується на розумінні людини як ключової цінності суспільства, що є визначальною умовою високотехнологічного й інноваційного розвитку держави та спрямована на задоволення потреб суспільства, ринку праці й державних інституцій у висококваліфікованих фахівцях [24]; орієнтована на забезпечення всебічного розвитку особистості, спроможної до свідомого вибору індивідуальної освітньої траєкторії, навчання впродовж життя, професійного зростання, розвитку підприємницької діяльності та самозайнятості, а також на

підготовку особистості, яка набула освітніх і професійних компетентностей відповідно до власних інтересів, здібностей і можливостей з урахуванням потреб національної економіки та суспільства [168]; здатності учасників освітнього процесу розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій [22].

Розробка структурно-функціональної моделі здійснювалося послідовно та охоплювало комплекс взаємопов'язаних стадій.

Здійснено аналіз підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління як об'єкта розробки. У результаті виявлено основні тенденції розвитку професійної освіти та ІТ-галузі. Окреслено професійне поле діяльності бакалаврів з інженерії програмного забезпечення у сфері проєктного управління та визначено утруднення, що виникають у процесі його реалізації. Проаналізовано ресурсні можливості дисциплін циклу професійної та практичної підготовки, використання яких сприяє формуванню функціонального й особистісного компонентів готовності до проєктного управління. Обґрунтовано методологічні підходи, принципи та методики, що забезпечують формування оновленого змісту підготовки зазначених фахівців. Визначено організаційно-педагогічні умови та методичне забезпечення підготовки до проєктного управління.

Обрано форму розроблення моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Обґрунтовано доцільність використання структурно-функціональної моделі, що відображає цілісні теоретичні уявлення про зазначений процес і охоплює його мету, методологічні підходи та принципи, змістове наповнення, методи, форми й засоби реалізації, умови досягнення поставлених цілей, а також очікуваний результат.

Обґрунтовано теоретичні засади розроблення структурно-функціональної моделі, що дало змогу систематизувати наукові відомості щодо вітчизняного та зарубіжного досвіду, а також результатів створення аналогічних моделей іншими

науковцями. Крім того, проаналізовано теоретичні й емпіричні дослідження впливу педагогічних систем і освітніх процесів на особистість, зокрема в аспекті організаційно-педагогічного впливу на формування функціональних та особистісних якостей.

На стадії методичного забезпечення розроблено рекомендації щодо впровадження організаційно-педагогічних умов формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління (системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проєктної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду); структуру готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; складові структурно-функціональної моделі означеної підготовки; інструментарій контролю й діагностики академічних (особистісних) досягнень з набування функціональних та особистісних якостей).

Просторово-часового забезпечення підготовки до проєктного управління було визначено часові межі реалізації дослідження та окреслено його експериментальну базу. Ураховано особливості організації освітнього процесу. Проаналізовано трудомісткість обраних форм і методів підготовки майбутніх фахівців. Визначено обсяг часу, необхідний для передавання та ефективного засвоєння навчального матеріалу.

Стадія матеріально-технічного забезпечення підготовки до проєктного управління був спрямований на добір технологічного обладнання та програмного забезпечення з урахуванням обраної спеціалізації. Забезпечено використання сучасних інформаційних, комунікаційних і цифрових технологій у процесі підготовки майбутніх фахівців. Відібрано та систематизовано інформаційні

джерела, зокрема спеціальну, психолого-педагогічну літературу й інтернет-ресурси.

Правове підґрунтя розробки структурно-функціональної моделі забезпечувалося дотриманням вимог законодавчих і нормативно-правових актів, що регламентують організацію та реалізацію освітнього процесу.

На стадії добору системоутворювального фактора розробки структурно-функціональної моделі було визначено динамічну стратегію, яка ґрунтується на врахуванні системоутворювальних можливостей суб'єктів і процесів та передбачає обґрунтування цілей, принципів, змісту, методів, засобів і форм, необхідних для їх цілеспрямованого розвитку.

Наступна стадія передбачала виявлення та обґрунтування взаємозв'язків і взаємозалежностей між структурними складниками процесу підготовки до проєктного управління.

Складання календарного плану розроблення моделі та програми педагогічного експерименту з підготовки майбутніх фахівців до зазначеної діяльності передбачало визначення послідовності етапів, термінів їх реалізації, змістового наповнення експериментальної роботи та узгодження основних організаційних заходів, спрямованих на забезпечення ефективності дослідження.

Наступний етап передбачав проведення концептуального (уявного) експериментування підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Він включав попередню оцінку поведінки здобувачів вищої освіти та викладачів у запропонованій моделі освітнього процесу, а також прогнозування очікуваних результатів, що проявляються у формуванні та реалізації індивідуальних функціональних і особистісних якостей майбутніх фахівців.

Здійснено оцінку розробленої форми підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління шляхом її експертизи компетентними фахівцями та потенційними користувачами, зокрема представниками закладів вищої освіти України, з метою визначення відповідності освітнього процесу вимогам професійної діяльності та потребам ринку праці.

Коригування процесу підготовки до проєктного управління здійснювалося на основі аналізу результатів педагогічного експерименту, що дозволяло врахувати виявлені сильні сторони, усунути недоліки та оптимізувати освітні підходи, методи й форми організації навчального процесу для підвищення його ефективності.

Заключна стадія моделювання була присвячена прийняттю рішення щодо застосування структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Після затвердження моделі розпочато її впровадження в освітню практику, що включало інтеграцію розроблених методів, форм і засобів навчання з метою ефективного формування готовності до означеної діяльності.

Отже, структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління базується на поєднанні концептуальних підходів (системного, синергетичного та компетентнісного) і специфічних підходів (інтегративного та проєктного). Вона передбачає формування у здобувача вищої освіти нових функціональних і особистісних якостей, а також компетентностей як суб'єкта проєктної діяльності, одночасно визначаючи комплекс завдань і діяльностей, необхідних для їх ефективного засвоєння та практичного застосування.

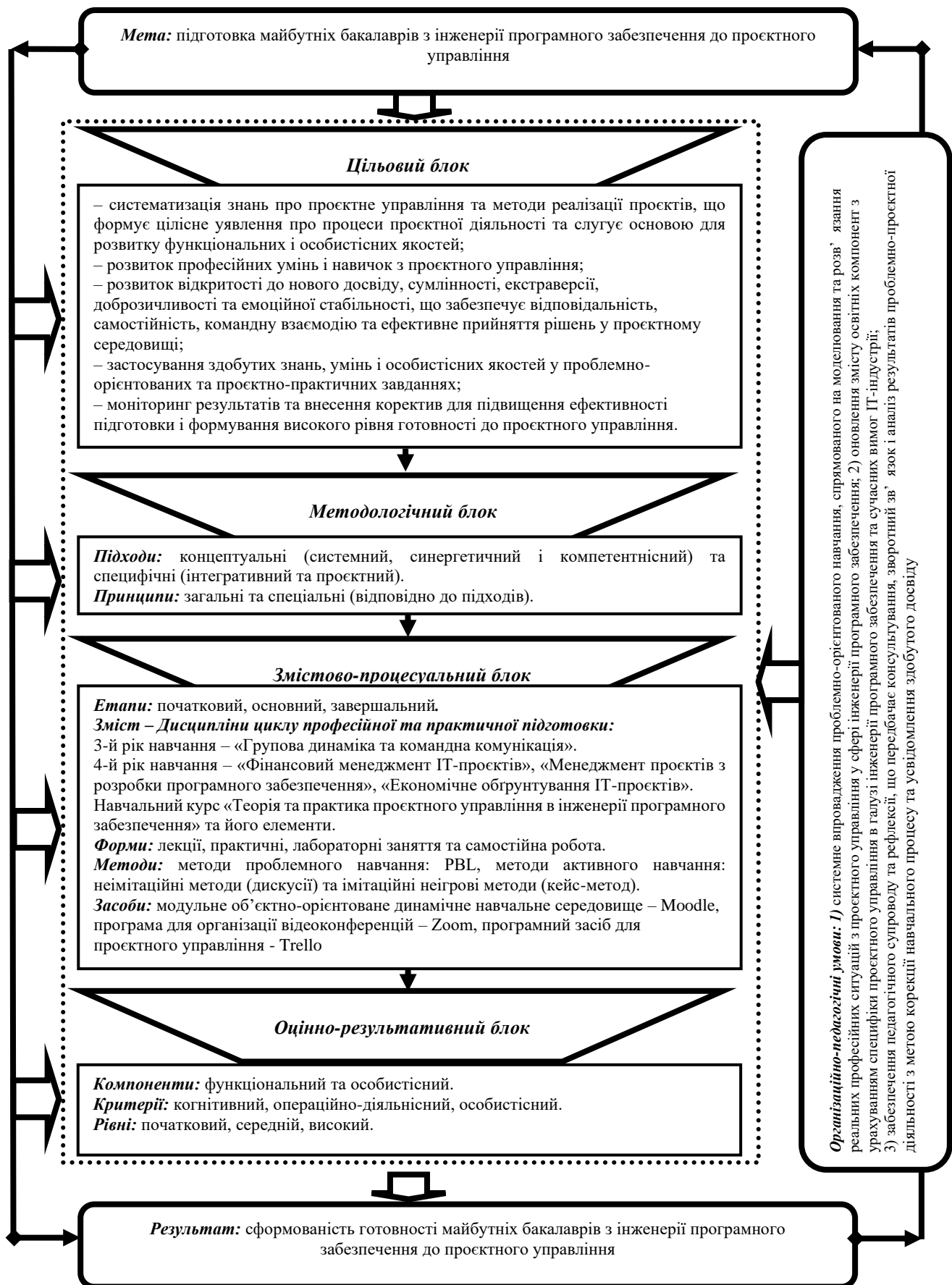


Рис. 2.1 Структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Вважаємо, що майбутні бакалаври з інженерії програмного забезпечення у процесі оволодіння спеціальністю зазнають поступових змін у структурі готовності до проєктного управління. На початковому етапі здійснюється ознайомлення з методами й принципами проєктної діяльності, що відповідає стану адаптації. Далі, у процесі практичної реалізації проєктних завдань, відбувається набуття та відпрацювання практичних умінь і навичок, що характеризує стан інтенсифікації готовності. На етапі професійної інтеграції проєктних якостей здобуті знання та навички впроваджуються у професійну діяльність, формуючи стан ідентифікації, який проявляється через зовнішні та внутрішні зміни у готовності майбутніх фахівців до виконання професійних проєктних функцій.

Розроблена структурно-функціональна модель складається із взаємопов'язаних і взаємозалежних цільового, методологічного, змістово-процесуального та оцінно-результативного блоків, є динамічною і спрямованою на результат з формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Відповідно до поставленої мети було сформульовано комплекс завдань: систематизація знань про проєктне управління та методи реалізації проєктів, що формує цілісне уявлення про процеси проєктної діяльності та слугує основою для розвитку функціональних і особистісних якостей; розвиток професійних умінь і навичок з проєктного управління; розвиток відкритості до нового досвіду, сумлінності, екстраверсії, доброзичливості та емоційної стабільності, що забезпечує відповідальність, самостійність, командну взаємодію та ефективне прийняття рішень у проєктному середовищі; застосування здобутих знань, умінь і особистісних якостей у проблемно-орієнтованих та проєктно-практичних завданнях, моніторинг результатів та внесення коректив для підвищення ефективності підготовки і формування високого рівня готовності до проєктного управління.

Методологічний блок відображає підходи та принципи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Змістово-процесуальний блок містить форми організації освітнього процесу,

методи та засоби навчання дисциплін циклу професійної та практичної підготовки майбутніх бакалаврів з інженерії програмного забезпечення, які розподіляються за етапами підготовки. Оцінно-результативний блок визначає перевірку рівня сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління згідно з когнітивним, операційно-діяльнісним та особистісним критеріями.

Якість формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління залежить від реалізації організаційно-педагогічних умов.

Проведене дослідження створило науково-теоретичне підґрунтя для екстраполяції розробленої структурно-функціональної моделі в систему підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Висновки до розділу 2

У другому розділі дисертаційної роботи «Моделювання підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління» наголошується на тому, що підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління потребує створення цілісної системи організаційно-педагогічних умов, орієнтованих на поєднання технічної, управлінської та комунікативної складових професійної діяльності. Проведений аналіз засвідчив, що традиційна модель підготовки, зосереджена переважно на формуванні алгоритмічних і програмувальних компетентностей, не забезпечує належного рівня готовності студентів до роботи в умовах реальних ІТ-проєктів, які характеризуються високою динамічністю, невизначеністю та необхідністю командної взаємодії.

Встановлено, що найбільш ефективними організаційно-педагогічними умовами підготовки є: системне впровадження проблемно-орієнтованого навчання; оновлення змісту освітніх компонент відповідно до сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу, фасилітації та рефлексії

навчальної діяльності студентів. Організаційно-педагогічні умови забезпечують формування функціональної та особистісної готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, сприяють розвитку професійної мобільності, адаптивності, стратегічного мислення та здатності ефективно діяти в умовах сучасного технологічного середовища.

Використання проблемно-орієнтованого навчання сприяє розвитку критичного мислення, здатності до аналізу складних професійних ситуацій, навичок прийняття рішень, командної взаємодії та відповідальності за результати діяльності. Метод кейсів забезпечує формування стратегічного бачення проєктної діяльності, навичок управління ризиками, комунікації із зацікавленими сторонами та здатності діяти в умовах професійної невизначеності.

Доведено, що ефективність підготовки значною мірою залежить від зміни ролі викладача, який у межах проблемного навчання виступає фасилітатором, консультантом і координатором пізнавальної діяльності студентів. Використання педагогічно виважених проблемних запитань, організація командної взаємодії, підтримка рефлексії та моделювання професійних ситуацій забезпечують активізацію пізнавальної діяльності здобувачів освіти та формування внутрішньої мотивації до професійного розвитку.

Обґрунтовано доцільність застосування трибальної критеріально орієнтованої системи оцінювання, яка дозволяє оцінювати не лише кінцевий результат, а й рівень сформованості професійних компетентностей, індивідуальний внесок студента та готовність до виконання функцій проєктного управління. Інтеграція проблемно-орієнтованого навчання, методу кейсів, фасилітаційних технологій і компетентісно орієнтованого оцінювання створює умови для подолання розриву між академічною підготовкою та реальними потребами ІТ-індустрії.

Обґрунтовано необхідність розроблення структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління відповідно до сучасних вимог ІТ-галузі та тенденцій розвитку професійної освіти. Визначено, що моделювання є ефективним методом

наукового пізнання, який дозволяє системно відобразити процес професійної підготовки, встановити взаємозв'язки між його складниками та прогнозувати результати формування готовності до проєктного управління.

Проаналізовано наукові підходи до трактування понять «модель», «педагогічна модель» та «моделювання», а також функції моделей у педагогічних дослідженнях – описову, пояснювальну та прогностичну. З'ясовано, що існуючі моделі підготовки ІТ-фахівців переважно спрямовані на формування технічних компетентностей і недостатньо враховують специфіку проєктного управління як інтегрованого виду професійної діяльності.

Обґрунтовано доцільність використання структурно-функціональної моделі, яка забезпечує цілісне уявлення про процес підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління та охоплює мету, методологічні підходи, зміст, форми, методи, засоби, організаційно-педагогічні умови й очікувані результати підготовки. Визначено, що модель базується на поєднанні системного, синергетичного, компетентнісного, інтегративного та проєктного підходів.

Встановлено, що підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління спрямована на формування функціональних і особистісних якостей. До функціональних якостей віднесено здатність до планування, управління ресурсами, прийняття рішень, використання agile- та традиційних інструментів проєктного управління, взаємодії з командою та забезпечення бізнес-цінності проєкту. Особистісний компонент охоплює відкритість до нового досвіду, сумлінність, екстраверсію, доброзичливість та емоційну стабільність.

Описано основні стадії розроблення структурно-функціональної моделі: аналіз професійної підготовки як об'єкта моделювання; обґрунтування форми моделі; визначення методологічних засад; розроблення методичного, матеріально-технічного, правового та просторово-часового забезпечення; встановлення взаємозв'язків між структурними компонентами; проведення концептуального експериментування; експертне оцінювання та коригування моделі; впровадження

моделі в освітній процес.

Визначено, що структурно-функціональна модель є динамічною системою, яка складається з цільового, методологічного, змістово-процесуального та оцінно-результативного блоків. Цільовий блок спрямований на формування готовності до проектного управління; методологічний – відображає підходи та принципи підготовки; змістово-процесуальний – охоплює форми, методи та засоби навчання; оцінно-результативний – забезпечує діагностику рівня сформованості готовності за когнітивним, операційно-діяльним та особистісним критеріями.

Окреслено поетапний характер формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління, що включає стадії адаптації, інтенсифікації та ідентифікації. На кожному етапі відбувається поступове засвоєння знань, розвиток професійних умінь і навичок, а також інтеграція функціональних та особистісних якостей у професійну діяльність.

Основні результати розділу відображено в наукових працях автора [169; 170; 171; 172; 173; 174; 175].

РОЗДІЛ 3

ЕКСПЕРИМЕНТАЛЬНА ПЕРЕВІРКА ЕФЕКТИВНОСТІ СТРУКТУРНО-ФУНКЦІОНАЛЬНОЇ МОДЕЛІ ПІДГОТОВКИ МАЙБУТНІХ БАКАЛАВРІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДО ПРОЄКТНОГО УПРАВЛІННЯ

3.1. Організація та проведення експериментального дослідження ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Загальна гіпотеза дослідження ґрунтується на припущенні, що підвищення кадрового потенціалу ІТ-галузі, зокрема в контексті сформованості здатності фахівців ефективно здійснювати проєктне управління, може бути забезпечене цілеспрямованою орієнтацією професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення на формування відповідної управлінської готовності, а також створенням і реалізацією комплексу належних організаційно-педагогічних умов у процесі їх фахової підготовки (системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проєктної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду).

Загальна гіпотеза дослідження конкретизується системою часткових гіпотез, відповідно до яких розроблена структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління буде результативною за умови дотримання низки взаємопов'язаних положень, а саме:

– якщо сформованість готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління визначається як прогнозований і цілеспрямований результат освітнього процесу, організованого з урахуванням динамічної стратегії професійної підготовки та реалізації концептуальних підходів (системного, синергетичного, компетентнісного) і спеціальних методологічних підходів (інтегративного та проєктного);

– якщо між визначеними структурними складниками підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління забезпечується стійкий взаємозв'язок і взаємообумовленість, що сприяє цілісності та узгодженості процесу формування відповідної готовності;

– якщо створено умови для динамічної взаємодії та збалансованого поєднання всіх компонентів готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що забезпечує їх поступальний розвиток і інтеграцію в межах освітнього процесу.

Отже, педагогічний експеримент створює підґрунтя для науково обґрунтованої перевірки ефективності запропонованого педагогічного нововведення – структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Отримані в ході експерименту результати дають змогу оцінити вплив упровадженої моделі на рівень сформованості відповідної готовності здобувачів освіти та підтвердити доцільність її використання в освітньому процесі закладів вищої освіти.

Відповідно до запропонованої моделі визначено чинники, що зумовлюють рівень сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, зокрема організаційно-педагогічні умови освітнього процесу та навчально-методичне забезпечення, а також чинники, які зазнають змін під їхнім впливом, а саме функціональний і особистісний компоненти готовності до проєктного управління. Згідно з логікою перевірки гіпотези дослідження було обрано паралельний експеримент, у межах якого сформовано експериментальну та контрольну групи. Освітній процес в

експериментальній групі здійснювався з урахуванням упровадження визначених чинників, тоді як у контрольній групі він організовувався на засадах традиційної освітньої практики.

З метою підтвердження наявності або відсутності ефекту впливу експериментального чинника було здійснено комплекс взаємопов'язаних дослідницьких процедур. На початковому етапі проведено контрольний зріз у контрольній та експериментальній групах з метою визначення однорідності вихідного рівня готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; у разі виявлення відмінностей порівняння результатів здійснювалося з урахуванням установлених методологічних вимог. Подальший освітній процес у контрольній та експериментальній групах було організовано відповідно до програми педагогічного дослідження, при цьому в експериментальній групі реалізовувався вплив визначеного експериментального чинника, а також здійснювалися проміжні діагностичні зрізи з метою відстеження динаміки змін показників готовності. Після завершення формувального етапу експерименту проведено підсумковий зріз у обох групах, що дало змогу встановити рівень сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Отримані результати вихідних і підсумкових зрізів було порівняно в межах кожної групи та між групами загалом, зокрема щодо змін рівня сформованості готовності та її окремих структурних компонентів. Для забезпечення об'єктивності та наукової вірогідності висновків результати експерименту опрацьовано з використанням методів математичної статистики та визначено їхню статистичну достовірність. На підставі узагальнення кількісних і якісних показників педагогічного експерименту сформульовано обґрунтовані висновки щодо ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Базою експериментального дослідження обрано такі заклади вищої освіти: Західноукраїнській національний університет, Черкаський національний університет імені Богдана Хмельницького та Луцький національний технічний

університет.

Педагогічний експеримент з перевірки дієвості структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. здійснено в три етапи протягом 2022-2026 років.

На першому етапі дослідження (2022–2023 роки) були окреслені та розв’язані такі завдання: з’ясовано особливості професійної діяльності майбутніх бакалаврів з інженерії програмного забезпечення та освітні вимоги до їхньої підготовки; обґрунтовано сутність проєктного управління та вимоги до його реалізації; визначено структуру готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, схарактеризовано критерії, показники та рівні.

На другому етапі дослідження (2023–2025 роки) реалізовано такі завдання: здійснено діагностичні зрізи (констатовано стан сформованості компонентів готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління); конкретизовано гіпотезу дослідження; обґрунтовано, розроблено та реалізовано структурно-функціональну модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; виявлено й обґрунтовано організаційно-педагогічні умови впровадження розробленої структурно-функціональної моделі в освітній процес майбутніх бакалаврів з інженерії програмного забезпечення; виміряно проміжні результати; скориговано випробувану структурно-функціональну модель; зроблено контрольні діагностичні зрізи. Другий етап передбачав проходження констатувального, формувального та контрольного складників педагогічного експерименту.

Метою констатувального експерименту було дослідження фактичного стану професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення та оцінка рівня досліджуваних характеристик на початковому етапі експерименту. Проведення цього етапу дозволяло отримати об’єктивну інформацію про початкові показники готовності студентів до проєктного управління та створити надійну методологічну основу для подальшої реалізації

формувального експерименту, спрямованого на підвищення ефективності освітнього процесу.

Формувальний експеримент проводився з метою дослідження об'єкта безпосередньо в процесі впровадження структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Цей етап експерименту дозволяв оцінити ефективність моделі в реальних умовах освітнього процесу, простежити динаміку змін у рівні готовності студентів до проєктного управління та визначити вплив організаційно-педагогічних і методичних чинників на формування необхідних професійних компетентностей.

Перехід до формувального експерименту здійснено на підставі усвідомлення викладачами експериментальних баз вимог щодо реалізації розробленої структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що проявився у наступних напрямках:

1. Науково-теоретичний – ознайомлення з концептуальною базою дослідження, методологічними підходами, критеріями, показниками та рівнями сформованості готовності майбутніх бакалаврів до проєктного управління; використання діагностичних інструментів та елементів структурно-функціональної моделі підготовки; відстеження специфіки досліджуваного процесу, оцінювання проміжних результатів та коригування організаційно-педагогічних умов професійної підготовки.

2. Науково-методичний – впровадження оновленого змісту практико-орієнтованої складової циклу дисциплін професійної та практичної підготовки у навчальний процес; апробація форм, методів та засобів професійної підготовки відповідно до розробленої структурно-функціональної моделі.

3. Організаційний – укладання згоди на впровадження експериментальної моделі в освітній процес бази дослідження; забезпечення комунікації з учасниками педагогічного експерименту; інформаційний контроль за перебігом експерименту в конкретній групі здобувачів першого рівня вищої освіти; вирішення завдань

організації освітнього процесу відповідно до експериментальної моделі у межах функціональних обов'язків.

4. Інші види діяльності – презентація накопиченого досвіду на засіданнях кафедри, наукових семінарах та конференціях.

Контрольний експеримент проводився з метою порівняння отриманих результатів експериментальної та контрольної груп. Сформульовано висновки стосовно динаміки сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління за всіма критеріями та рівнями.

На третьому етапі (2025-2026 рр.) узагальнено й систематизовано результати експериментального дослідження: обґрунтовано вибір методів математичної статистики; сформульовано основні висновки та рекомендації щодо впровадження та застосування структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Педагогічний експеримент здійснювався на основі розробленої структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління із впровадженням в освітній процес навчального курсу «Теорія та практика проєктного управління в інженерії програмного забезпечення» та його елементів у цикл дисциплін професійної та практичної підготовки.

Головні цілі реалізації запропонованої методики формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління:

1 тиждень: забезпечити занурення студентів у фундаментальні аспекти управління проєктами, створити функціональні команди розробників та проаналізувати катастрофічні наслідки невдалого планування проєктів у реальному світі.

2 тиждень: забезпечити перехід студентів від етапу формування команди до діяльності в умовах хаотичного процесу збору вимог, формування здатності виявляти, класифікувати та пом'якшувати конфлікти між конкуруючими вимогами

зацікавлених сторін з метою запобігання неконтрольованому розширенню сфери проєкту, а також розвиток сумлінності як особистісної якості, необхідної для ретельного та системного документування вимог.

3 тиждень: забезпечити формування здатності інтелектуально оцінювати обмеження проєкту, нормативні вимоги та особливості командної взаємодії для обґрунтованого вибору між прескриптивними підходами (зокрема каскадною моделлю Waterfall) та адаптивними фреймворками (Agile, Scrum, Kanban, SAFe).

4 тиждень: забезпечити формування у студентів розуміння фінансових і ресурсних обмежень проєктної діяльності в інженерії програмного забезпечення, зокрема усвідомлення впливу командної взаємодії та комунікаційних витрат на ефективність реалізації проєкту відповідно до закону Брукса.

5 тиждень: забезпечити перехід студентів від реактивного виправлення помилок до проактивного антикризового управління, формування здатності застосовувати структуру корпоративних ризиків для виявлення технічних, операційних і бізнес-ризиків.

6 тиждень: забезпечити початок фактичного технічного виконання проєкту шляхом написання коду та створення програмного артефакту.

7 тиждень: забезпечити проведення проміжного огляду проєкту та формування стратегій його розгортання в умовах виникнення технічних труднощів і часових обмежень у процесі реалізації проєктної діяльності.

8 тиждень: забезпечити інтеграцію в проєктний цикл суворих стандартів забезпечення якості (Quality Assurance) у процесі реалізації проєктної діяльності.

9 тиждень: забезпечити формування у студентів здатності управляти технічним боргом та застосовувати стратегії його мінімізації в умовах дотримання часових обмежень комерційної розробки програмного забезпечення.

10 тиждень: забезпечити формування здатності застосовувати техніку згладжування ресурсів (Resource Smoothing) для оптимального розподілу робочого навантаження без зміни термінів завершення проєкту.

11 тиждень: забезпечити формування у студентів здатності здійснювати системну інтеграцію програмного забезпечення та забезпечувати корпоративну

кібербезпеку під час роботи зі сторонніми API, управління сесіями та захисту корпоративних і користувацьких даних.

12 тиждень: забезпечити формування у студентів здатності здійснювати закриття проєкту, підготовку документації та планування переходу програмного продукту до етапу технічного обслуговування і підтримки.

13 тиждень: змодельовати стресове середовище перед фінальним релізом шляхом впровадження фінального спринту та жорсткого правила «заморозки коду» (code freeze) для стабілізації продукту й підготовки до випуску.

14 тиждень: підготуватися до захисту проєкту та забезпечити стратегічне узгодження, перетворюючи технічні рішення програмного забезпечення на зрозумілу бізнес-цінність для кінцевих стейкхолдерів.

15 тиждень: підготувати підсумкові презентації, провести оцінювання та ретроспективу як кульмінацію всього семестру, що інтегрує технічні навички, комунікаційні здібності та командну роботу.

Для систематизації процесу підготовки майбутніх бакалаврів з програмного забезпечення до практичного застосування знань у проєктному управлінні запропоновано послідовний підхід до виконання завдань. Алгоритм розв'язання завдань, що моделюють реальні умови проєктного управління індивідуально та у команді представлено у таблиці 3.1

Таблиця 3.1

Алгоритм виконання завдань з проєктного управління для підготовки майбутніх бакалаврів з програмного забезпечення

№	Етап алгоритму	Індивідуальні дії	Командні дії
1	Ознайомлення з завданням та постановка цілей	Аналіз завдання, визначення очікуваного результату, оцінка ресурсів та часових обмежень, формування власних навчальних цілей	Визначення цілей проєкту, узгодження очікуваних результатів, обговорення обмежень, формування навчальних цілей команди

2	Аналіз ресурсів та умов виконання	Оцінка власних знань і навичок, готовності до виконання завдання	Визначення ролей і зон відповідальності, оцінка компетенцій учасників, розподіл ресурсів
3	Планування дій	Складання особистого плану: алгоритм виконання, пріоритети, часові рамки	Формування спільного плану проєкту, розподіл завдань і етапів, узгодження комунікаційних каналів
4	Моделювання сценаріїв і управління ризиками	Передбачення технічних та організаційних проблем, розробка способів їх вирішення	Оцінка ризиків, розробка плану мінімізації, вироблення стратегій адаптації до змін
5	Виконання завдання	Реалізація власних частин проєкту: код, тести, документація, контроль якості	Інтеграція індивідуальних результатів, код-рев'ю, тестування, контроль дотримання графіку
6	Контроль і оцінка результатів	Порівняння результатів із завданням, оцінка власних сильних і слабких сторін	Колективна оцінка проєкту, аналіз командної взаємодії, підготовка підсумкового звіту або презентації
7	Рефлексія та професійне вдосконалення	Аналіз власного досвіду, визначення напрямів розвитку навичок	Спільна ретроспектива, обговорення успіхів і труднощів, вироблення рекомендацій для покращення командної роботи

Успішність реалізації проєкту визначається збалансованим управлінням ключовими параметрами проєктного трикутника, а саме обсягом і якістю результатів, вартістю та часовими обмеженнями, взаємозалежність яких зумовлює

досягнення стратегічних і тактичних цілей проєкту та ефективність використання ресурсів. Система управління проєктами охоплює сукупність взаємопов'язаних і послідовних управлінських дій, серед яких визначення та формалізація вимог до проєкту, постановка чітких і вимірюваних цілей, налагодження ефективної комунікації між усіма зацікавленими сторонами, а також управління проєктними обмеженнями, зокрема бюджетом, ресурсами, ризиками, термінами виконання та якістю результатів. Важливу роль відіграє взаємодія з проєктною командою, врахування її потреб, очікувань і зворотного зв'язку з подальшим коригуванням планів відповідно до отриманої інформації.

Зазначені управлінські процеси структуруються за етапами життєвого циклу проєкту, а саме: ініціація, планування, виконання і контроль, завершення. Саме комплексне планування, ефективна організація завдань і складових проєкту, забезпечення їх необхідними ресурсами та систематичний контроль результативності обраної стратегії становлять основу управління проєктами та забезпечують досягнення поставленої мети.

Ініціація проєкту є початковим етапом його життєвого циклу, під час якого здійснюється первинне формування уявлення про проєкт, визначаються його сутність і цілі, а також здійснюється формування проєктної команди відповідно до поставлених завдань.

Планування є етапом управління проєктом, на якому детально визначаються всі дії, необхідні для досягнення поставленої мети. Проєкт декомпонується на окремі складові та сукупність взаємопов'язаних завдань, формується календарний графік виконання робіт із визначенням термінів для кожного завдання, а також здійснюється ідентифікація та розподіл необхідних ресурсів. Водночас процес планування передбачає можливість його періодичного перегляду й коригування відповідно до змін умов реалізації проєкту.

Етап виконання і контролю тісно пов'язаний з етапом планування та реалізується з ним у взаємодії. В ефективній системі управління проєктом цей процес має циклічний характер і передбачає послідовну постановку завдань, їх виконання, контроль результатів, внесення необхідних коригувань до плану та

подальше формування нових завдань. Під час виконання проєкту застосовуються управлінські інструменти, спрямовані на підвищення ефективності процесів, зокрема делегування повноважень, методи тайм-менеджменту, використання матриці Ейзенхауера для пріоритизації завдань, а також інші прийоми організації та контролю виконання робіт.

На етапі завершення проєкту проводиться підсумкова оцінка виконаних робіт та контроль досягнення поставлених цілей, а також забезпечується збереження всіх вихідних даних, інструкцій і регламентів, що використовувалися під час реалізації проєкту. Сучасне управління проєктами передбачає застосування різноманітних методологій. Окрім класичного підходу, поширене використання Agile, який передбачає розподіл великого проєкту на серію міні-проєктів із поетапною реалізацією; Scrum, що передбачає поділ проєкту на окремі функціональні блоки; Lean, який орієнтований на дроблення проєкту на невеликі робочі пакети; та Kanban, ефективний у проєктах без жорстких дедлайнів завдяки можливості тимчасового призупинення завдань. Кожна з цих методик має власні особливості, переваги та обмеження, а вибір найбільш відповідного підходу визначається специфікою організації та складом команди, що реалізує проєкт.

Розглянемо приклади, що ілюструють досягнення цілей підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління в контексті реалізації організаційно-педагогічних умов. Так під час вивчення дисципліни «Групова динаміка та командна комунікація» (6-й семестр) основна увага була зосереджена на формуванні у майбутніх бакалаврів з інженерії програмного забезпечення усвідомлення власних індивідуальних та особистісних характеристик, емоційних станів і соціально значущих почуттів, а також на розвитку здатності до психічної регуляції, включно зі становленням та відновленням внутрішніх ресурсів у процесі проєктної діяльності в умовах постійних змін. Особливий акцент робився на усвідомленні студентами критичної ролі самостійності, особистої відповідальності та активного прийняття рішень у забезпеченні ефективності командної роботи та досягненні поставлених цілей. Крім того, навчальна діяльність передбачала формування навичок адаптивної

взаємодії у команді, розпізнавання і регулювання конфліктних ситуацій, а також стратегій конструктивної комунікації для підвищення ефективності спільних проєктних процесів.

В процесі підготовки використовувалися неімітаційні методи навчання, зокрема дискусія, що передбачала реалізацію дій адаптації, оцінювання та консолідації. Дія адаптації полягала в підготовці майбутніх бакалаврів з інженерії програмного забезпечення до усвідомлення визначеної проблеми та формування установки на її розв'язання. При цьому викладач, виконуючи функції організатора дискусії, забезпечував формулювання проблеми й цілей обговорення, створення мотиваційних передумов шляхом обґрунтування значущості розв'язання проблеми, виявлення її суперечливих аспектів, визначення очікуваного результату та встановлення правил ведення дискусії.

Так, під час лекційного заняття з теми «Галузеві особливості ефективної командної діяльності» увагу було зосереджено на психологічному та організаційному аналізі командної роботи в ІТ-галузі з використанням методології Agile як сучасного підходу до управління проєктною діяльністю в умовах постійних змін і невизначеності. Викладачем було окреслено ключові характеристики ефективної командної взаємодії майбутніх бакалаврів з інженерії програмного забезпечення, зокрема: роль групової динаміки у формуванні командної згуртованості; значення психологічної безпеки, довіри та відкритої комунікації; особливості розподілу ролей і відповідальності в кросфункціональних Agile-командах.

Особливу увагу приділено усвідомленню здобувачами вищої освіти власних індивідуальних та особистісних характеристик, емоційних станів і соціально значущих почуттів як чинників, що безпосередньо впливають на ефективність командної діяльності. Наголошувалося на важливості розвитку здатності до психічної саморегуляції, зокрема вміння підтримувати й відновлювати внутрішні ресурси під час інтенсивної проєктної роботи, коротких ітерацій, дедлайнів та змін вимог замовника, що є типовими для Agile-середовища.

Акцент робився на критичній ролі самостійності, особистої відповідальності

та активного прийняття рішень у забезпеченні результативності командної роботи. Зазначалося, що в Agile-командах кожен учасник виступає не лише виконавцем, а й активним суб'єктом командної взаємодії, відповідальним за спільний результат. У цьому контексті розглядалися суперечності між високими вимогами до автономності та відповідальності майбутніх інженерів програмного забезпечення і недостатньою сформованістю навичок самоорганізації та командної рефлексії у частини здобувачів вищої освіти.

Виявлені суперечності стали підґрунтям для формулювання проблеми дискусії – необхідності цілеспрямованого розвитку навичок ефективної командної взаємодії, адаптивної комунікації та відповідального прийняття рішень у майбутніх фахівців ІТ-галузі.

Друга дія – оцінки – передбачала активний обмін думками між учасниками заняття, залучення кожного студента до аналізу власного досвіду командної роботи в навчальних або проєктних Agile-форматах. Для активізації дискусії використовувалися професійно зорієнтовані запитання, зокрема: «Що у Вашому розумінні означає ефективна Agile-команда?», «Яку роль відіграє особиста відповідальність у досягненні командних цілей?», «Як впливають емоційні стани учасників на продуктивність спринтів?».

З метою виявлення глибинних установок і ставлень студентів до командної діяльності пропонувалося продовжити речення: «Ефективна командна взаємодія в Agile – це...». Так, здобувач вищої освіти М. зазначав, що це – узгоджена робота команди, де кожен розуміє свою роль і відповідальність; здобувачка А. вважала, що це – постійний обмін зворотним зв'язком і готовність до змін; здобувач С. підкреслював, що це – здатність команди швидко адаптуватися до нових вимог без втрати якості продукту.

На запитання: «Особиста відповідальність у командному проєкті для мене – це...» студенти відповідали, що це – усвідомлення впливу власних рішень на спільний результат; готовність брати ініціативу; здатність визнавати помилки та працювати над їх виправленням. У відповідях на запитання: «Командна комунікація в Agile – це...» здобувачі наголошували на важливості відкритості,

взаємної підтримки та конструктивного вирішення конфліктних ситуацій.

Третя дія – консолідації – передбачала вироблення спільних висновків і узагальнених позицій щодо особливостей ефективної командної діяльності в ІТ-галузі. На цьому етапі здійснювалося співвіднесення поставленої мети заняття з отриманими результатами обговорення, визначення позитивних і проблемних аспектів командної взаємодії, а також формулювання рекомендацій щодо підвищення ефективності роботи Agile-команд.

У результаті спільного обговорення було зроблено висновок, що ефективна командна діяльність у галузі програмної інженерії можлива за умови поєднання високого рівня особистісної усвідомленості, розвитку навичок саморегуляції, відповідального прийняття рішень і конструктивної комунікації. Саме взаємозв'язок індивідуальних психологічних ресурсів і групової динаміки сприяє підвищенню продуктивності командної роботи та досягненню проектних цілей в Agile-середовищі.

Отже, використання дискусійних методів навчання в межах дисципліни «Групова динаміка та командна комунікація» сприяє формуванню готовності майбутніх бакалаврів з інженерії програмного забезпечення до ефективної командної діяльності, адаптивної взаємодії та успішної професійної реалізації в умовах сучасної ІТ-галузі.

На практичних заняттях розв'язувалися кейси, зокрема присвячені середовищу Agile-команди. Наприклад:

1. Усвідомлення індивідуальних та особистісних якостей в Agile-команді. *Кейс:* Під час спринту Agile-команди з розробки програмного забезпечення один із розробників уникає активної участі в командних обговореннях, що ускладнює досягнення спринт-цілей. *Завдання:* Проаналізувати вплив індивідуальних особистісних характеристик на ефективність роботи Agile-команди, визначити власну роль у команді (Developer, Scrum Master, Product Owner) та обґрунтувати, які якості є критично важливими для успішної спринт-діяльності.

2. Усвідомлення емоційних станів і соціально значущих почуттів у процесі спринтів. *Кейс:* Після кількох інтенсивних спринтів у команді спостерігається

зниження мотивації та зростання напруженості під час daily-мітингів. *Завдання:* Визначити основні емоційні стани учасників Agile-команди, оцінити їхній вплив на командну комунікацію та запропонувати інструменти підтримки позитивного психологічного клімату (ретроспектива, командні домовленості, психологічна безпека).

3. Психічна саморегуляція в умовах ітеративної розробки. *Кейс:* Через часті зміни беклогу окремі члени команди відчувають перевантаження та втрату контролю над завданнями. *Завдання:* Запропонувати індивідуальні стратегії саморегуляції та командні практики Agile, спрямовані на зниження стресу та підтримання стабільної продуктивності в межах спринту.

4. Проектна діяльність в умовах постійних змін вимог. *Кейс:* Product Owner змінює пріоритети беклогу безпосередньо під час спринту через нові вимоги замовника. *Завдання:* Проаналізувати допустимість таких змін відповідно до принципів Agile, визначити оптимальні дії команди та обґрунтувати механізми адаптації планування без втрати якості результату.

5. Самостійність і відповідальність у межах Agile-ролей. *Кейс:* Розробник самостійно обирає технічне рішення, яке не відповідає домовленостям команди та ускладнює інтеграцію продукту. *Завдання:* Оцінити межі індивідуальної автономії в Agile-команді, визначити зону відповідальності кожної ролі та запропонувати способи узгодження технічних рішень у процесі спільної роботи.

6. Командна комунікація та взаємодія в Agile-процесах. *Кейс:* Під час спринт-ретроспективи виникає конфлікт між членами команди через різні підходи до оцінювання результатів роботи. *Завдання:* Проаналізувати причини конфлікту, визначити роль комунікаційних практик Agile у його вирішенні та запропонувати заходи для підвищення ефективності командної взаємодії.

Дисципліна «Фінансовий менеджмент ІТ-проектів» орієнтована на формування у здобувачів системи теоретичних знань, прикладних умінь і практичних навичок з економічного обґрунтування ІТ-проектів, а також розвиток компетентностей щодо прийняття проектних рішень на основі аналізу інноваційних підходів до впровадження інформаційних технологій та методів

оцінки економічної ефективності інвестицій у IT-проекти. Оновлення змісту передбачає використання елементів навчального курсу «Теорія та практика проектного управління в інженерії програмного забезпечення». Наприклад, в період практичного етапу PBL, який пов'язан з бюджетуванням та бізнес-моделюванням (4 тижень), команди приступають до розподілу ресурсів для свого симульованого проекту. Вони повинні призначити «сторі-пойнти» (story points) елементам свого беклогу і створити комплексну бізнес-модель. Залежно від обраної раніше методології (SDLC), студенти застосовують різні методи: високорівневі підходи, такі як «розміри футболок» (T-shirt sizing) і оцінка «зверху вниз» для гнучких середовищ з високим ступенем невизначеності, або детерміновані моделі, такі як метод критичного шляху (CPM) та моделювання СОСОМО для зрілих контрактів. Для ілюстрації катастрофічних наслідків помилкової оцінки ресурсів студенти глибоко занурюються у вивчення Бостонського проекту «Центральна артерія/Тунель», відомого як «The Big Dig» – найбільшого, найскладнішого і найтехнологічнішого амбітного інфраструктурного проекту в історії США. Студенти стикаються з шокуючими емпіричними даними: початковий кошторис проекту, розроблений у 1982 році, становив 2,56 мільярда доларів, але на момент фактичного завершення основних робіт у 2007 році (із затримкою майже на 10 років) вартість зросла до приголомшливих 14,8 мільярда доларів (в цінах 2007 року). Аналізуючи цей провал, студенти формують інсайти третього порядку. Вони розуміють, що в оцінці проектів існує різниця між технічною помилкою та системним оптимістичним упередженням (optimism bias). Уроки, винесені з «The Big Dig», вимагають застосування Прогнозування на основі референтних класів (Reference Class Forecasting), коли кошториси базуються на фактичних результатах аналогічних минулих проектів, а не на теоретичних моделях інженерів. Крім того, проект продемонстрував життєву необхідність наявності «єдиного командира» (Single Commander) – уповноваженого керівника з повним контролем над усіма сторонами для усунення агентських конфліктів. Як *метод фасилітації* викладач використовує тактику «Питання-бумеранг». Коли команда не може дійти консенсусу щодо фінансового резерву на випадок ризику інтеграції їхнього

програмного забезпечення, фасилітатор переадресовує запитання всій групі: «Враховуючи, що керівництво The Big Dig виключило інфляцію зі своїх оцінок заради схвалення бюджету, чи не повторюєте ви зараз ту саму помилку, ігноруючи вартість технічного боргу у вашій бізнес-моделі?». Це стимулює критичну самооцінку команди.

Освітній компонент циклу професійної та практичної підготовки «Менеджмент проєктів з розробки програмного забезпечення» спрямований на формування у здобувачів освіти компетентностей, необхідних для розв'язання практичних завдань у сфері управління ІТ-проєктами. У межах дисципліни передбачається опанування методології управління проєктами в ІТ-галузі, методів логіко-структурного аналізу та оцінювання ризиків, підходів до формування концепції проєкту, а також особливостей застосування міжнародних стандартів проєктного менеджменту при створенні програмного забезпечення. З метою посилення впливу на формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління посилено практико-орієнтований зміст. Наприклад, при впровадженні навчального курсу «Теорія та практика проєктного управління в інженерії програмного забезпечення» (перший тиждень) або при вивченні дисципліни «Менеджмент проєктів з розробки програмного забезпечення» (7 семестр). *Етап PBL – формування команди та статуту.* Спочатку організовуємо студентів у невеликі команди розробників по 4–5 учасників. При цьому свідомо застосовується підхід «слабкий–сильний», коли академічно слабших студентів поєднують із сильнішими однолітками. Така структура створює середовище взаємонавчання, зменшує залежність від викладача та моделює реальні умови професійного наставництва. Далі проводяться початкові тимбілдинг-вправи, які дозволяють перевірити ефективність командної взаємодії та підкреслити важливість раннього планування і постановки питань у життєвому циклі проєкту. Після цього командам надається автономія у виборі підходів до розробки, однак обов'язковою умовою є створення офіційного командного статуту. У ньому вони розподіляють ролі, визначають внутрішні обов'язки та встановлюють чіткі протоколи комунікації. Окремо здійснюється моніторинг динаміки роботи в

командах. У разі дисбалансу – коли окремі учасники домінують або, навпаки, усуваються від участі – викладач втручається мінімально, але рішуче, тимчасово призупиняючи технічну роботу для проведення керованої ретроспективи. Це дозволяє вирівняти участь усіх членів команди та оновити базові правила взаємодії.

Застосування методу кейсів – «Невдалий запуск». Представляємо кейс «Невдалий стартап», який демонструє типові проблеми командної дисфункції, відсутності лідерства та нечіткого визначення обсягу проєкту. У цьому кейсі також підкреслюється, що, за дослідженнями, середній життєвий цикл невдалих стартапів часто становить лише близько 28 місяців. Далі проводиться аналіз першопричин невдач. Студенти діагностують фактори провалу, зокрема ситуації, коли керівництво намагається компенсувати слабкі бізнес-рішення надмірними витратами на складні цифрові технології замість того, щоб фокусуватися на реальних потребах клієнтів. Після цього організовується етап прийняття управлінських рішень, під час якого студенти беруть на себе ролі керівників. Вони визначають ключові перешкоди в проєкті, оцінюють необхідні бюджетні обмеження та тренуються обґрунтовувати ресурси, потрібні для успішної реалізації проєкту.

Методи фасилітації (когнітивне скелювання). Передусім для стимулювання обговорення кейсів у PBL утримуємося від надання прямих відповідей на архітектурні дилеми, натомість застосовуючи структуровані методи сократівського запитування. Зокрема використовуємо підхід альтернативного досягнення, змушуючи команди оцінювати складні компроміси, наприклад: чи доцільніше у межах обмеженого часу повністю відмовитися від певної функції, чи прийняти технічний борг. Також застосовується метод «питання-бумеранг», коли запитування студентів перенаправляються назад до групи для формування колективного рішення та досягнення консенсусу. Окремо використовується техніка «розрядки допиту», яка полягає у переформулюванні критичних запитань таким чином, щоб зменшити захисну реакцію студентів і сприяти аналітичному мисленню. Наприклад, замість «Чому ви обрали саме це?» ставиться питання про конкретні обмеження, які вплинули на вибір підходу. Додатково застосовується метод «відкличання», коли думки студентів узагальнюються та структуровано

повертаються в обговорення для корекції або уточнення напрямку дискусії, наприклад через повторне винесення важливих, але недоопрацьованих тем.

При вивченні дисципліни «Економічне обґрунтування ІТ-проектів» (8 семестр) відповідно використовуються дискусійні питання та кейси, які спрямовані формування у здобувачів освіти теоретичних знань, практичних умінь і навичок щодо оцінювання економічної доцільності ІТ-проектів, аналізу ефективності інвестицій у сфері інформаційних технологій та прийняття обґрунтованих управлінських рішень у процесі планування й реалізації ІТ-проектів. Відповідно до навчального курсу «Теорія та практика проектного управління в інженерії програмного забезпечення» (2 тиждень) здійснюється розробка вимог та обсяг проекту – *етап PBL* початкові співбесіди з клієнтами та складання SRS. Пропонується рольова гра «Нестабільний клієнт», в якій студенти повинні переконливо діяти як клієнти проекту. Автентичне моделювання вимагає від них навмисного введення розпливчастих специфікацій, зміни бізнес-пріоритетів і навіть змін обсягу робіт на пізніх стадіях. Мета рольової гри – змусити студентів активно практикувати переговори щодо вимог, адаптацію до ризиків та складне мистецтво протистояння неможливим або суперечливим вимогам. Після співбесід, командам надається допомога у складанні базової Специфікації вимог до програмного забезпечення (SRS). Підкреслюється, що під час виконання цього конкретного завдання потрібна дуже сумлінна поведінка, оскільки увага до деталей тут зменшує властивий хаос у складних середовищах кодування пізніше. В межах застосування *методу кейсів*, спочатку представляємо кейс «Повернення до школи: криза», у якому основна увага зосереджується на ризиках, пов'язаних із неефективним управлінням вимогами до проекту. Далі організовується аналіз першопричин проблемної ситуації, під час якого студенти визначають початкові ризики проекту та наслідки неконтрольованого розширення обсягу робіт. Особлива увага приділяється аналізу помилок у врахуванні очікувань стейкхолдерів та впливу цих помилок на строки реалізації проекту. Після цього студенти переходять до етапу прийняття управлінських рішень, де моделюють діяльність керівника проекту. Вони пропонують практичні підходи до узгодження очікувань

зацікавлених сторін, а також розробляють механізми виявлення й мінімізації конфліктних вимог ще до того, як вони почнуть негативно впливати на реалізацію проєкту. Для супроводу роботи з вимогами та управління обсягом проєкту застосовуємо методи сократівського запитування, адаптовані до відповідних ситуацій. Зокрема використовуємо підхід альтернативного досягнення, моделюючи ситуацію нестабільного замовника, який постійно ініціює нові вимоги. У таких умовах спонукаємо команду аналізувати компроміси між функціональністю, бюджетом, строками реалізації та доступними ресурсами. Також застосовуємо метод «питання-бумеранг», коли запитання студентів щодо зрозумілості або повноти вимог перенаправляються іншим учасникам команди. Це дозволяє виявити приховані припущення, неоднозначності у специфікаціях та покращити спільне розуміння вимог. Окремо використовується техніка «розрядки допиту», що передбачає конструктивне формулювання критичних зауважень до некоректно описаних вимог. Замість прямої критики акцент переноситься на аналіз проєктних обмежень і причин прийняття певних рішень, що сприяє зменшенню захисної реакції та розвитку аналітичного мислення студентів.

Таким чином, підготовка майбутніх бакалаврів з інженерії програмного забезпечення здійснювалася з урахуванням логіки життєвого циклу проєкту – від етапів ініціації та планування до виконання, контролю й завершення проєкту. Особлива увага приділялася усвідомленню студентами взаємозалежності ключових параметрів проєктного трикутника – обсягу та якості результатів, вартості й часових обмежень, збалансоване управління якими визначає успішність реалізації проєкту та ефективність використання ресурсів. Практико-орієнтований характер підготовки забезпечував формування здатності приймати обґрунтовані управлінські рішення в умовах невизначеності, ризиків, змін вимог і необхідності постійної командної взаємодії.

Використання кейсів, рольових ігор, дискусійних методів, фасилітаційних технік і PBL-підходу сприяло розвитку у здобувачів освіти навичок командної комунікації, самоорганізації, адаптивного мислення, управління ризиками, бюджетування, аналізу проєктних обмежень та прийняття рішень у реальних

професійних ситуаціях. Водночас інтеграція сучасних методологій управління проєктами (Agile, Scrum, Kanban, Lean) дозволила наблизити освітній процес до умов функціонування сучасних ІТ-команд та сформувати у студентів готовність до професійної діяльності в динамічному цифровому середовищі.

3.2. Аналіз результатів експериментального дослідження з оцінки ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Для реалізації педагогічного експерименту були сформовані експериментальна та контрольна групи здобувачів вищої освіти спеціальності 121 «Інженерія програмного забезпечення» загальною чисельністю 264 особи. Контрольна група налічувала 132 особи, підготовка яких здійснювалася за традиційною системою навчання, без внесення спеціальних коректив в освітній процес. Експериментальна група також включала 132 особи і відрізнялася тим, що їх професійна підготовка організовувалася із застосуванням розробленої структурно-функціональної моделі, яка передбачала цілеспрямоване формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, інтегруючи практико-орієнтовані завдання, командну взаємодію та проблемно-орієнтовані методи навчання. Така організація педагогічного експерименту дозволяла забезпечити порівняльний аналіз ефективності традиційних і інноваційних підходів у професійній підготовці майбутніх бакалаврів інженерії програмного забезпечення до проєктного управління.

Об'єктивність результатів експериментального дослідження, спрямованого на перевірку ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, забезпечується еквівалентністю двох вибірок здобувачів вищої освіти, тобто однорідністю контрольної й експериментальної груп за рівнем сформованості готовності до проєктного управління на початок експерименту.

З метою визначення початкового рівня сформованості готовності до

проектного управління та перевірки наявності чи відсутності статистичних розбіжностей як рядів прийнято індивідуальні значення тестових відповідей майбутніх бакалаврів з інженерії програмного забезпечення. Кожному рівню присвоєно відповідний бал: початковий – 1; середній – 2; високий – 3. Обчислення середнього бала початкового рівня сформованості готовності до проектного управління наведено в табл. 3.2.

Таблиця 3.2

Обчислення середнього бала початкового рівня сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління

Контрольна група				Експериментальна група			
Бал	Кількість здобувачів вищої освіти	Загальна кількість балів	Середнє значення	Бал	Кількість здобувачів вищої освіти	Загальна кількість балів	Середнє значення
1	71	71		1	64	64	
2	48	96		2	58	116	
3	13	39		3	10	30	
Σ	132	206	1,56	Σ	132	210	1,59

Згідно з табл. 3.2 середній бал у контрольній та експериментальній групах має незначну відмінність, яка складає 0,03. Переважну більшість в обох порівнюваних категоріях складають здобувачі вищої освіти з початковим та середнім рівнями сформованості готовності до проектного управління.

За допомогою розрахунку t-критерію визначили достовірність збігів і відмінностей контрольної та експериментальної груп. Для цього сформулювали дві гіпотези.

Гіпотеза H_0 : відмінності між рівнями сформованості готовності до проектного управління майбутніх бакалаврів з інженерії програмного забезпечення експериментальної і контрольної груп не значні.

Гіпотеза H_1 : відмінності між рівнями сформованості готовності до проєктного управління майбутніх бакалаврів з інженерії програмного забезпечення експериментальної і контрольної груп досить значні.

Значення t -критерію визначили за формулою 3.1:

$$t = \frac{M_1 - M_2}{\sqrt{\frac{S_1^2}{N_1} + \frac{S_2^2}{N_2}}} \quad (3.1),$$

де M_1 та M_2 – середнє значення першої та другої вибірок; S_1 та S_2 – дисперсія (середньоквадратичне відхилення) відповідно для першої та другої вибірок; N_1 та N_2 – кількість оцінок у першій та другій вибірках.

Для обчислення цього показника визначили дисперсію за формулою 3.2:

$$S^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{N-1} \quad (3.2),$$

де $(x_i - \bar{x})^2$ – квадрат відхилень окремих значень ознак від середньої арифметичної; N – кількість ознак.

Розрахунок дисперсії сформованості професійних якостей майбутніх судноводіїв наведено в табл. 3.3.

Таблиця 3.3

Розрахунок дисперсії сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Групи	x_i	\bar{x}	$x_i - \bar{x}$	$(x_i - \bar{x})^2$	$n_i(x_i - \bar{x})^2$	S^2
ЕГ	1	1,59	-0,59	0,3481	0,3481	0,051
	2		0,41	0,1681	0,3362	
	3		1,41	1,9881	5,9643	
КГ	1	1,56	-0,56	0,3136	0,3136	0,052
	2		0,44	0,1936	0,3872	
	3		1,44	2,0736	6,2208	

З табл. 3.3 видно, що розмитість розподілу оцінок щодо середнього арифметичного значення абсолютно незначна.

Отримали дисперсію, розрахували значення t -критерію:

$$t = \frac{1,59 - 1,56}{\sqrt{\frac{0,051}{132} + \frac{0,052}{132}}} \approx 1,075$$

Табличне значення t-критерію Стьюдента більше, ніж розрахункове ($t_{\text{табл.}} (1,969) > t_{\text{розрах.}} (1,075)$). Це свідчить про те, що нульова гіпотеза не відкидається, обидві вибірки відносяться до однієї генеральної сукупності, тобто вони однорідні для рівня достовірності 0,05 (ймовірність 5 %), що, своєю чергою, дозволяє стверджувати про недостатню значущість відмінності рівня готовності майбутніх бакалаврів з інженерії програмного забезпечення експериментальної й контрольної груп і є умовно рівними. Окремо проаналізовано достовірність збігів і відмінностей контрольної та експериментальної груп щодо сформованості компонентів готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління на початку експерименту задля запобігання ймовірних похибок на формувальному етапі.

Метою статистичного опрацювання отриманих даних педагогічного експерименту було оцінювання дієвості й результативності використання розробленої структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

У процесі педагогічного експерименту здійснювалася порівняльна характеристика формування компонентів готовності до проєктного управління за обраними критеріями (див. дод. В, Г).

Узагальнена динаміка рівня сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління представлено в табл. 3.4.

Таблиця 3.4

Динаміка формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління (%)

Рівень готовності	На початок експерименту		На кінець експерименту	
	КГ=132	ЕГ=132	КГ=132	ЕГ=132

Початковий	71	53,8	64	48,5	47	35,6 (-18,2)	23	17,4 (-31,1)
Середній	48	36,4	58	43,9	63	47,7 (+11,3)	75	56,8 (+12,9)
Високий	13	9,8	10	7,6	22	16,7 (+6,9)	34	25,8 (+18,2)

Аналіз експериментальних даних засвідчив якісні зміни на початковому рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що зумовлено зменшенням кількості здобувачів вищої освіти цієї категорії впродовж педагогічного експерименту в КГ на 18,2 % та ЕГ на 31,1 %. Середній та високий рівні сформованості означеної готовності також продемонстрували позитивні якісні зміни: збільшення кількості здобувачів вищої освіти у КГ (на 11,3 % і 6,9 % відповідно) та ЕГ (на 12,9 % і 18,2 % відповідно).

Порівняльний аналіз величин абсолютного середнього значення якісних змін рівня сформованості готовності до проєктного управління в контрольній та експериментальній групах уможливив висновок про те, що позитивна динаміка якісних змін у КГ ($C_p=12,1\%$) зумовлена традиційною професійною підготовкою, а в ЕГ ($C_p=20,7\%$) – результат професійної підготовки здобувачів вищої освіти згідно з експериментальною моделлю. Графічно динаміка якісних змін сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення впродовж педагогічного експерименту представлена на рис. 3.1.

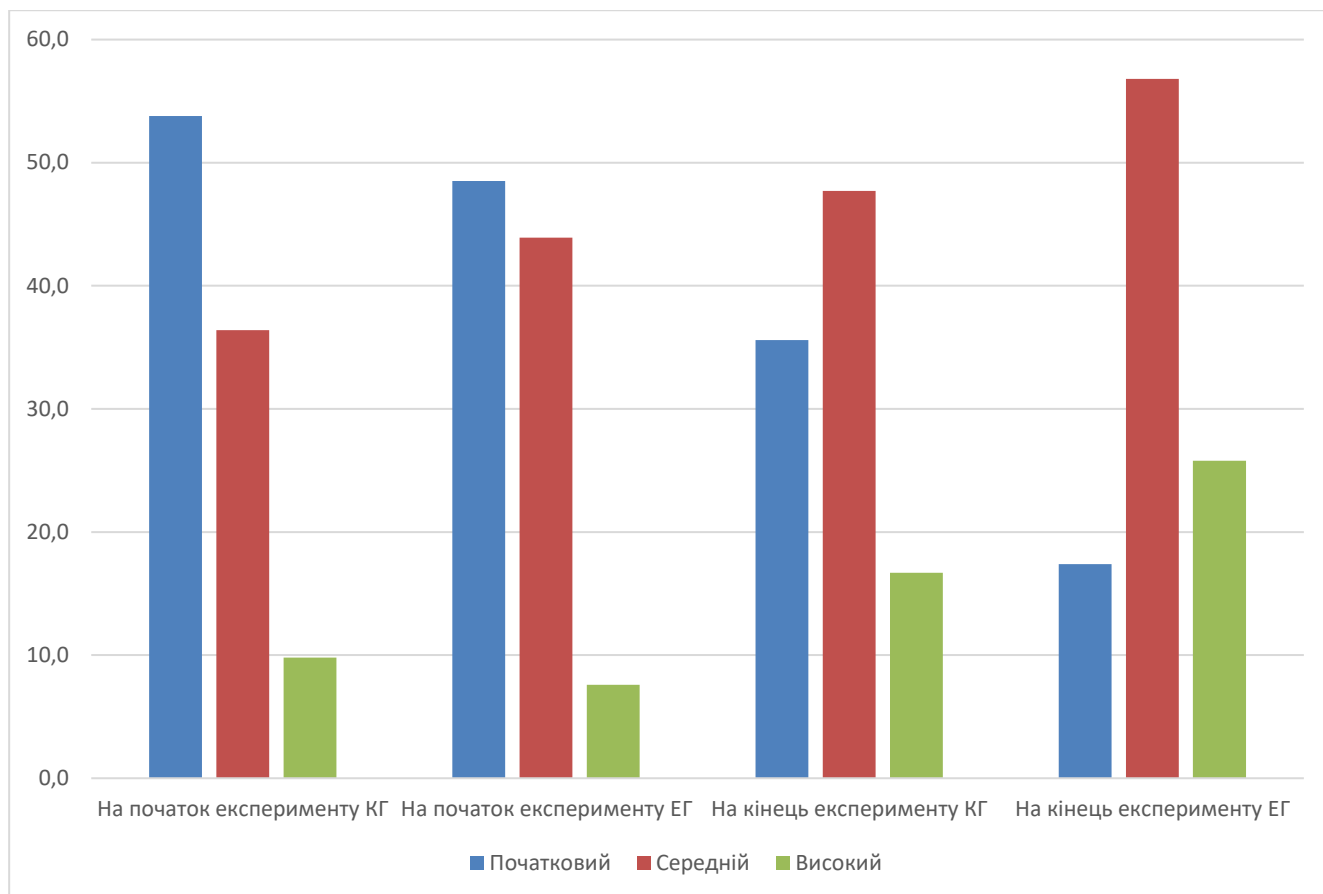


Рис. 3.1 Динаміка якісних змін сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління контрольної та експериментальної груп

Отримані результати експериментального дослідження було підтверджено за допомогою методів математичної статистики, зокрема через обчислення непараметричного критерію Пірсона – χ^2 . Для цього були сформульовані відповідні статистичні гіпотези:

Н₀: експериментальна модель не впливає на якісні зміни рівня сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, а спостережувані результати є випадковими;

Н₁: якісні зміни рівня сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління зумовлені впливом запропонованої експериментальної моделі.

Критерій Пірсона – χ^2 використовується для обчислення значень, отриманих у результаті вимірювання порядковою та інтервальними шкалами, якщо необхідно встановити, чи існує істотна відмінність між рядами показників двох сукупностей.

Ґрунтується метод χ^2 на порівнянні частот, що характеризують розподіл значень. Алгоритм визначення достовірності збігів і відмінностей для експериментальних даних, виміряних за порядковою шкалою, полягає в наступному:

1. Обчислити для порівнюваних вибірок $\chi^2_{\text{експ}}$.
2. Порівняти це значення з критичним значенням $\chi^2_{\text{крит}}$ (табличні дані):
 - якщо $\chi^2_{\text{експ.}} \leq \chi^2_{\text{крит}}$, то зробити висновок: «характеристики порівнюваних вибірок збігаються з рівнем значущості 0,05»;
 - якщо $\chi^2_{\text{експ.}} > \chi^2_{\text{крит}}$, то зробити висновок: «достовірність відмінностей характеристик порівнюваних вибірок складає 95 %».

Непараметричний критерій оцінки χ^2 обчислюємо за формулою 3.3:

$$\chi^2 = \sum \left(\frac{(f'_E - f'_k)^2}{f'_k} \right) \quad (3.3)$$

де f'_E – відносна частота інтервалу ряду експериментальної групи; f'_k – відносна частота інтервалу ряду контрольної групи.

Кількість ступенів свободи для χ^2 розраховується як $k=(R-1)(C-1)$, де R і C – кількість рядків і стовпців у таблиці сполученості 3.8. У нашому випадку $k = (3-1)(2-1) = 2$ (3 – це кількість рівнів, 2 – це кількість груп).

За статистичними таблицями для рівня значення $\alpha = 0,05$ та числа ступеня свободи $k=2$ знайдено критичне значення критерію 5,991.

На основі отриманих даних було визначено відносні частоти та розраховано значення критерію Пірсона. Узагальнені результати подано в таблиці 3.5.

Таблиця 3.5

Робоча таблиця для обчислення χ^2 - критерію

Рівні	$f'_E, \%$	$f'_k, \%$	$f'_E - f'_k$	$(f'_E - f'_k)^2$	$\frac{(f'_E - f'_k)^2}{f'_k}$
Початковий	17,4	35,6	-18,2	331,24	9,304
Середній	56,8	47,7	9,1	82,81	1,736
Високий	25,8	16,7	9,1	82,81	4,958

Розраховане значення критерію Пірсона становить $\chi^2 = 15,998$, що перевищує табличне значення 5,99. Це дає підстави для відхилення нульової гіпотези H_0 та підтвердження альтернативної гіпотези H_1 , згідно з якою вплив експериментальної моделі сприяв формуванню готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління.

Крім того, з метою визначення відмінностей між експериментальною та контрольною групами за рівнем сформованості готовності до проєктного управління після завершення експерименту було здійснено розрахунок t-критерію Стьюдента.

На основі цього сформулювали дві гіпотези:

Гіпотеза H_0 : відмінності рівня сформованості готовності до проєктного управління експериментальної та контрольної груп майбутніх бакалаврів з інженерії програмного забезпечення недостатньо значні.

Гіпотеза H_1 : відмінності рівня сформованості готовності до проєктного управління експериментальної та контрольної груп майбутніх бакалаврів з інженерії програмного забезпечення досить значні.

Середні бали, що характеризують рівень сформованості цієї готовності на завершальному етапі дослідження, подано в таблиці 3.6.

Таблиця 3.6

Обчислення середнього балу рівня сформованості готовності майбутніх бакалаврів інженерії програмного забезпечення до проєктного управління на кінець експерименту

Контрольна група				Експериментальна група			
Бал	Кількість студентів	Загальна кількість балів	Середнє значення	Бал	Кількість студентів	Загальна кількість балів	Середнє значення
1	47	47	1,81	1	23	23	2,08
2	63	126		2	75	150	
3	22	66		3	34	102	
Σ	132	239	1,81	Σ	132	275	2,08

Згідно з отриманими даними середній бал у контрольних та експериментальних групах має значну відмінність, різниця складає 0,27.

Для обчислення t-критерію визначили дисперсію. Розрахунок дисперсії сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління наведено в табл. 3.7.

Таблиця 3.7

Розрахунок дисперсії сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Групи	x_i	\bar{x}	$x_i - \bar{x}$	$(x_i - \bar{x})^2$	$n_i(x_i - \bar{x})^2$	S^2
ЕГ	1	2,08	-1,8	1,1664	1,1664	0,028
	2		-0,08	0,0064	0,0128	
	3		0,92	0,8464	2,5392	
КГ	1	1,81	-0,81	0,6561	0,6561	0,038
	2		0,19	0,0361	0,0722	
	3		1,19	1,4161	4,2483	

$$t = \frac{2,08 - 1,81}{\sqrt{\frac{0,028}{132} + \frac{0,038}{132}}} \approx 12,27$$

Одержане значення t-критерію Стьюдента $t_{\text{розрах.}}=12,27$ ($t_{\text{табл.}} < t_{\text{розрах.}}$ (1,969 < 12,27)) свідчить про відмінність рівня сформованості готовності до проєктного управління експериментальної й контрольної груп з імовірністю 95%.

Отже, за всіма параметрами відбулися статистично достовірні зміни рівня сформованості готовності майбутніх бакалаврів інженерії програмного забезпечення до проєктного управління, що дає підставу зробити висновок про дієвість розробленої моделі підготовки.

Висновки до розділу 3

У розділі «Експериментальна перевірка ефективності структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління» розкрито організацію та проведення експериментального дослідження. Узагальнення результатів педагогічного експерименту дає підстави стверджувати, що розроблена структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного

забезпечення до проєктного управління є педагогічно доцільною та ефективною. Реалізація моделі в освітньому процесі забезпечила позитивну динаміку сформованості готовності здобувачів освіти до проєктного управління, що підтверджується результатами констатувального, формувального та контрольного етапів експерименту, а також статистичною обробкою отриманих даних.

У процесі експериментальної перевірки підтверджено загальну гіпотезу дослідження, відповідно до якої підвищення рівня готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління забезпечується за умов цілеспрямованої професійної підготовки, орієнтованої на формування управлінської готовності, а також комплексного впровадження визначених організаційно-педагогічних умов. Найбільш суттєвий вплив на результативність підготовки справили системне використання проблемно-орієнтованого навчання, оновлення змісту професійної та практичної підготовки відповідно до сучасних вимог ІТ-індустрії, інтеграція навчального курсу «Теорія та практика проєктного управління в інженерії програмного забезпечення», а також педагогічний супровід, спрямований на рефлексію, консультивання та корекцію навчальної діяльності студентів.

Результати дослідження засвідчили, що ефективність структурно-функціональної моделі забезпечується взаємозв'язком і взаємообумовленістю її складників, реалізацією системного, синергетичного, компетентнісного, інтегративного та проєктного підходів, а також створенням умов для динамічної взаємодії функціонального й особистісного компонентів готовності до проєктного управління. Упровадження практико-орієнтованих методів навчання (PBL, кейс-методу, рольових ігор, фасилітаційних технік, дискусій) сприяло формуванню у студентів здатності до прийняття управлінських рішень, командної взаємодії, адаптивної комунікації, управління ризиками, ресурсами, часовими обмеженнями та якістю програмного продукту.

Проведений експеримент підтвердив, що організація професійної підготовки відповідно до логіки життєвого циклу ІТ-проєкту – від ініціації та планування до виконання, контролю й завершення – забезпечує наближення освітнього процесу

до реальних умов функціонування сучасних ІТ-команд. Інтеграція сучасних методологій управління проєктами (Agile, Scrum, Kanban, Lean) дозволила сформувати у здобувачів освіти готовність до професійної діяльності в умовах динамічного цифрового середовища, невизначеності та постійних змін вимог.

Отримані результати експериментального дослідження підтверджують доцільність упровадження структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління в освітній процес закладів вищої освіти та засвідчують її потенціал як ефективного засобу формування професійної готовності майбутніх фахівців ІТ-галузі до управлінської діяльності в умовах сучасної проєктно-орієнтованої практики.

Основні результати розділу відображено в наукових працях автора [172; 174; 176.

ВИСНОВКИ

У дисертації теоретично узагальнено й запропоновано розв'язання наукової проблеми підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, зокрема, обґрунтовано, розроблено й експериментально перевірено структурно-функціональну модель цього процесу. Результати дослідження засвідчили досягнення мети, розв'язання завдань і дали змогу сформулювати такі висновки.

1. На основі аналізу сучасної теорії та практики розкрито сутність базових понять дослідження, зокрема «проєкт», «проєктне управління», «інженерія програмного забезпечення» та ін. Визначено особливості підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що полягають у її комплексному, інтегративному та практикоорієнтованому характері, зумовленому специфікою сучасної ІТ-галузі та вимогами професійного середовища. Підготовка майбутніх фахівців здійснюється в умовах високої динамічності ІТ-сфери, для якої характерні невизначеність, постійна зміна вимог до програмних продуктів, обмеженість ресурсів, потреба в оперативному прийнятті рішень та ефективній командній взаємодії. У зв'язку з цим освітній процес має забезпечувати не лише формування технічних знань у галузі програмної інженерії, а й розвиток компетентностей, пов'язаних із проєктним управлінням. Важливою особливістю такої підготовки є інтеграція знань із програмної інженерії та проєктного менеджменту. Майбутні бакалаври повинні опановувати методології управління ІТ-проєктами (Waterfall, Agile), набувати навичок стратегічного планування, управління ресурсами, ризиками та змінами, координації командної діяльності, оцінювання бізнес-цінності проєкту, а також ефективної професійної комунікації. Практикоорієнтований характер професійної підготовки виявляється у спрямованості освітнього процесу на набуття досвіду, наближеного до реальних умов професійної діяльності. Сучасні вимоги роботодавців засвідчують необхідність посилення практичної складової освітніх програм, оскільки між

рівнем підготовки здобувачів освіти та актуальними потребами ІТ-галузі зберігається певна невідповідність. У зв'язку з цим підготовка майбутніх бакалаврів має передбачати активне застосування проєктних методів навчання, моделювання професійних ситуацій, виконання командних ІТ-проєктів та залучення студентів до діяльності, максимально наближеної до реальної професійної практики.

Зважаючи на предмет дослідження уточнено поняття «підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління» і «готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління».

2. Виокремлено структурні компоненти готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління: *функціональний* – здатність до планування та визначення пріоритетів (проєктних і персональних); максимізації бізнес-цінності проєкту шляхом розроблення ефективної стратегії його реалізації; співпраці зі спонсором проєкту, командою та профільними експертами, необхідної для розроблення стратегії реалізації проєкту; використання як традиційних, так і Agile-інструментів, технік і методів управління проєктами; пояснення іншим важливих бізнес-аспектів проєкту; управління елементами проєкту, зокрема графіком, вартістю, ресурсами та ризиками; зосередження уваги на критично важливих технічних аспектах проєктного управління; *особистісний* – сформованість професійно важливих якостей (відкритість до нового досвіду, сумлінність, екстраверсія, доброзичливість та емоційна стабільність).

Схарактеризовано критерії, показники та рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Когнітивний критерій відображає рівень засвоєння знань щодо методологій, принципів, інструментів і процесів проєктного управління в галузі програмної інженерії, здатність аналізувати професійні ситуації та обґрунтовувати управлінські рішення. Операційно-діяльнісний критерій характеризує сформованість практичних умінь і навичок управління ІТ-проєктами, зокрема

планування робіт, управління вимогами, ризиками та ресурсами, використання традиційних і Agile-підходів, організації командної взаємодії, професійної комунікації та реалізації програмних проєктів у межах PBL-діяльності. Особистісний критерій визначає рівень розвитку професійно значущих якостей, зокрема відповідальності, ініціативності, лідерства, здатності до співпраці, рефлексії, прийняття рішень в умовах невизначеності, відкритості до нового досвіду, сумлінності, доброзичливості та емоційної стабільності. Відповідно до визначених критеріїв виокремлено три рівні сформованості готовності: початковий (незадовільний), середній (задовільний) та високий (з відзнакою).

3. Теоретично обґрунтовано та розроблено структурно-функціональну модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, яка складається з цільового, методологічного, змістово-процесуального та оцінно-результативного блоків; забезпечує результат, яким є сформована готовність майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління; передбачає поетапне запровадження оновленого змісту, форм організації освітнього процесу, методів і засобів навчання; реалізується завдяки створенню організаційно-педагогічних умов.

Розроблено систему проблемно-орієнтованих та проєктно-практичних завдань з елементами управлінської діяльності (на прикладі змісту дисциплін циклу професійної та практичної підготовки здобувачів вищої освіти за спеціальністю 121 Інженерія програмного забезпечення: «Групова динаміка та командна комунікація» (3-й рік навчання), «Фінансовий менеджмент ІТ-проєктів», «Менеджмент проєктів з розробки програмного забезпечення», «Економічне обґрунтування ІТ-проєктів» (4-й рік навчання)) та навчальний курс «Теорія та практика проєктного управління в інженерії програмного забезпечення».

Доведено ефективність організаційних форм, активних методів і засобів навчання у підготовці майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Дискусійні методи сприяють розвитку критичного мислення, аргументованого відстоювання професійної позиції та формуванню

навичок командної комунікації; кейс-метод забезпечує формування здатності аналізувати професійні ситуації, приймати управлінські рішення та оцінювати можливі наслідки їх реалізації; проблемно-орієнтоване навчання (PBL) активізує пізнавальну діяльність через розв'язання комплексних професійно спрямованих завдань, сприяє інтеграції теоретичних знань із практичною діяльністю та розвитку навичок роботи в умовах невизначеності. Комплексне застосування зазначених методів і підходів забезпечує формування функціональної та особистісної готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління відповідно до сучасних вимог ІТ-індустрії.

4. Визначено організаційно-педагогічні умови реалізації структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління: системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проєктного управління в галузі інженерії програмного забезпечення та сучасних вимог ІТ-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проєктної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду.

5. Експериментально перевірено ефективність структурно-функціональної моделі підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Аналіз експериментальних даних засвідчив якісні зміни на початковому рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, що зумовлено зменшенням кількості здобувачів вищої освіти цієї категорії впродовж педагогічного експерименту в КГ на 18,2 % та ЕГ на 31,1 %. Середній та високий рівні сформованості означеної готовності також продемонстрували позитивні якісні зміни: збільшення кількості здобувачів вищої освіти у КГ (на 11,3 % і 6,9 %

відповідно) та ЕГ (на 12,9 % і 18,2 % відповідно). Отримані дані підтверджено використанням методів математичної статистики, зокрема розрахунку t-критерію Стьюдента ($t_{\text{табл.}}(1,969) < t_{\text{розрах.}}(12,27)$) та непараметричного критерію Пірсона ($\chi^2_{\text{експ.}}(15,998) > \chi^2_{\text{крит.}}(5,99)$).

Викладені в дисертації результати дослідження й висновки не претендують на остаточне й вичерпне розв'язання проблеми підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Перспективи подальших розробок убачаємо у вивченні шляхів підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління в умовах дуального навчання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Горбатюк І. Оцінка вартості програмного забезпечення як методологічна проблема ІТ-галузі. *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації* : матеріали ІІІ Міжнар. наук.-практ. інтер.-конф., м. Запоріжжя, 30 верес. 2022 р. Запоріжжя, 2022. С. 506-510.
URL: <https://elar.tsatu.edu.ua/server/api/core/bitstreams/1fbc8169-7391-446d-b8f2-8c4720b7ae98/content>
2. Жекало Г. І. Цифрова економіка України: проблеми та перспективи розвитку. Науковий вісник Ужгородського національного університету : серія: Міжнародні економічні відносини та світове господарство. 2019. Вип. 26. С. 56–60.
3. Корогод Н. П. Специфіка ефективного управління ІТ-проєктами у компанії / Н. П. Корогод, Є. С. Швець, В. К. Виноградова // Збірник наукових праць за матеріалами V Міжнародної науково-практичної інтернет-конференції (Київ – Дніпро, 23–24 березня 2023 р.). – Дніпро : Український державний університет науки і технологій, 2023. – С. 227–236.
4. Канцевіч Б. Інноваційні ІТ-рішення в управлінні проєктами / Б. Канцевіч // Управління розвитком соціально-економічних систем : матеріали VIII Міжнародної науково-практичної конференції (м. Харків, 21–22 березня 2024 року). – Харків : ДБТУ, 2024. – Ч. 1. – С. 548–550.
5. ІТ Generation [Електронний ресурс] : [сайт]. – Режим доступу: <https://it-generation.gov.ua/> (дата звернення: 22.05.2026)
6. Сабадишина Ю. В Україні завершився освітній проєкт ІТ Generation. Найбільше випускників – Front-end, QA, UI/UX Design // DOU. – URL: <https://dou.ua/lenta/news/it-generation-results/> (дата звернення: 22.05.2026).
7. Digital Tiger: the Power of Ukrainian IT – 2023 [Електронний ресурс] : [дослідження] / IT Ukraine Association, Top Lead. – Електрон. текст. дані. – [Київ], 2024. – Режим доступу: https://itukraine.org.ua/files/ITU_GT.pdf (дата звернення: 22.05.2026).
8. Річна доповідь Національного агентства із забезпечення якості вищої

освіти за 2023 рік [Електронний ресурс] / Нац. агентство із забезпечення якості вищої освіти. – Електрон. текст. дані. – Київ, 2024. – Режим доступу: <https://naqa.gov.ua/wp-content/uploads/2024/05/%D0%94%D0%BE%D0%BF%D0%BE%D0%B2%D1%96%D0%B4%D1%8C-2023-%D1%80%D0%BE%D0%BA%D1%83.pdf> (дата звернення: 22.05.2026).

9. Чи потрібна вища освіта програмісту та які є альтернативи // ІТExpert. – URL: <https://itexpert.work/uk/vstupaty-chy-ne-vstupaty-chy-potribna-vyshha-osvita-programistu-dumky-fahivcziv/> (дата звернення: 22.05.2026).

10. Професійний стандарт «Фахівець з розробки програмного забезпечення» (2014). – URL: <https://mon.gov.ua/static-objects/mon/sites/1/vyshcha/IT-prof-standarty/6-ps-rozrobnik-pz-13.12.2014.pdf> (дата звернення: 22.05.2026).

11. Merriam-Webster Thesaurus [Електронний ресурс]. – Режим доступу: <https://www.merriam-webster.com/thesaurus/project> (дата звернення: 22.05.2026).

12. Пан М. П. Проектний аналіз : конспект лекцій (для студентів денної та заочної форм навчання зі спеціальності 051 – Економіка) / М. П. Пан, В. В. Коненко; Харків. нац. ун-т міськ. госп-ва ім. О. М. Бекетова. – Харків : ХНУМГ ім. О. М. Бекетова, 2019. – 95 с.

13. Shtub A. Project management : engineering, technology and implementation / A. Shtub, J. Bard, S. Globerson. – Englewood Cliffs, NJ : Prentice Hall, 1994. – 656 p.

14. Приймак В. М. Управління проектами : навчальний посібник / В. М. Приймак. – Київ : Київський національний університет імені Тараса Шевченка, 2017. – 464 с.

15. Özkan D. Agile Project Management Tools: A Brief Comparative View / D. Özkan, A. Mishra // Cybernetics and Information Technologies. – 2019. – No. 4. – P. 17–25.

16. A guide to the project management body of knowledge (PMBOK guide) / Project Management Institute. – Sixth Edition. – Newtown Square, PA : Project Management Institute, Inc., 2017. – 756 p.

17. Hughey D. The Traditional Waterfall Approach // University of Missouri-St.

Louis. – 2009. – URL: <https://www.umsl.edu/~hugheyd/is6840/waterfall.html> (date of access: 14.10.2023).

18. Hoffer J. A. Modern Systems Analysis and Design / J. A. Hoffer, J. F. George, J. S. Valacich. – Upper Saddle River, NJ : Pearson Education, Inc., 2008. – 683 p.

19. Van Casteren W. The Waterfall Model and the Agile Methodologies: A comparison by project characteristics / W. Van Casteren // ResearchGate. – 2017. – No. 2. – P. 1–6.

20. Greene J. Applied Software Project Management / J. Greene, A. Stellman. – [S. l.] : O'Reilly Media, 2005. – 322 p. – ISBN 978-0596009489.

21. Karhu K. Outsourcing and Knowledge Management in Software Testing / K. Karhu, O. Taipale, K. Smolander // 11th International Conference on Evaluation, Assessment in Software Engineering (EASE). – Göteborg, 2007. – P. 11

22. Про затвердження стандарту вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня вищої освіти : наказ МОН України від 29.10.2018 р. № 1166. – 2018. – URL: <https://mon.gov.ua/storage/app/media/vishcha-osvita/zatverdzeni%20standarty/12/21/121-inzhener.programn.zabezp.bakalavr-1.pdf> (дата звернення: 22.05.2026).

23. Про освіту : Закон України від 05.09.2017 р. № 2145-VIII. – URL: <https://zakon.rada.gov.ua/laws/show/2145-19> (дата звернення: 23.05.2026).

24. Про вищу освіту : Закон України від 01.07.2014 р. № 1556-VII. – URL: <https://zakon.rada.gov.ua/laws/show/1556-18> (дата звернення: 23.05.2026).

25. Про затвердження стандарту вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» для другого (магістерського) рівня вищої освіти : наказ МОН України від 17.11.2020 р. № 1424. – URL: https://mon.gov.ua/static-objects/mon/sites/1/vishcha-osvita/zatverdzeni%20standarty/2020/11/17/121_inzheneriya_prohramnoho_zabezpechnya_mahistr.doc (дата звернення: 23.05.2026).

26. The Standard for Project Management / Project Management Institute. – Newtown Square, PA : Project Management Institute, Inc., 2021. – 250 p.

27. The Standard for Portfolio Management / Project Management Institute. – Fourth Edition. – Newtown Square, PA : Project Management Institute, Inc., 2017. – 189 p.
28. Practice Standard for Project Risk Management / Project Management Institute. – Newtown Square, PA : Project Management Institute, Inc., 2009. – 116 p.
29. Practice Standard for Earned Value Management / Project Management Institute. – Third Edition. – Newtown Square, PA : Project Management Institute, Inc., 2019. – 162 p.
30. Practice Standard for Project Configuration Management / Project Management Institute. – Newtown Square, PA : Project Management Institute, Inc., 2007. – 73 p.
31. Practice Standard for Work Breakdown Structures / Project Management Institute. – Third Edition. – Newtown Square, PA : Project Management Institute, Inc., 2019. – 138 p.
32. Practice Standard for Scheduling / Project Management Institute. – Third Edition. – Newtown Square, PA : Project Management Institute, Inc., 2019. – 176 p.
33. Practice Standard for Project Estimating / Project Management Institute. – Second Edition. – Newtown Square, PA : Project Management Institute, Inc., 2019. – 134 p.
34. Project Manager Competency Development Framework / Project Management Institute. – Third Edition. – Newtown Square, PA : Project Management Institute, Inc., 2017. – 174 p.
35. Construction Extension to the PMBOK® Guide / Project Management Institute. – Newtown Square, PA : Project Management Institute, Inc., 2016. – 226 p.
36. IPMA Individual Competence Baseline for Project, Programme & Portfolio Management (IPMA ICB) / International Project Management Association. – Version 4.0. – Nijkerk : IPMA, 2015. – 431 p.
37. Project Management Association of Japan (PMAJ). A Guidebook of Project and Program Management for Enterprise Innovation (P2M). – Tokyo : PMAJ, 2017. – 427 p.
38. ISO 9001:2015. Quality management systems – Requirements / International Organization for Standardization. – Geneva : ISO, 2015. – 29 p.

39. ISO 10006:2017. Quality management systems – Guidelines for quality management in projects / International Organization for Standardization. – Geneva : ISO, 2017. – 34 p.
40. ICB : IPMA Competence Baseline, Version 3.0 / International Project Management Association. – Nijkerk : IPMA, 2006. – 200 p.
41. Software Extension to the PMBOK® Guide / Project Management Institute. – Fifth Edition. – Newtown Square, PA : Project Management Institute, Inc., 2013. – 288 p.
42. Пінська О. Л. Формування готовності майбутнього вчителя до творчої педагогічної діяльності // Вісник університету імені Альфреда Нобеля. Серія «Педагогіка і психологія». Педагогічні науки. – 2018. – № 2 (16). – С. 260–265.
43. Великий тлумачний словник сучасної української мови (з дод. і допов.) / уклад. і гол. ред. В. Т. Бусел. – Київ ; Ірпінь : ВТФ «Перун», 2005. – 1728 с.
44. Академічний тлумачний словник української мови. – URL: <http://sum.in.ua/> (дата звернення: 23.05.2026).
45. Копоруліна В. М. Українсько-російський психологічний тлумачний словник. – Харків : Факт, 2006. – 400 с.
46. Професійна освіта : словник : навч. посіб. / уклад. С. Гончаренко та ін. ; за ред. Н. Ничкало. – К. : Вища шк., 2000. – 380 с.
47. Енциклопедія освіти / гол. ред. В. Г. Кремень. – Київ : Юрінком Інтер, 2021. – 1144 с.
48. Словник психолого-педагогічних понять і термінів. – URL: <https://osvita.ua/school/method/psychology/1270/> (дата звернення: 23.05.2026).
49. Словник-довідник з професійної педагогіки / за ред. А. В. Семенової. – Одеса : Пальміра, 2006. – 221 с.
50. Броннікова В. Б. Формування готовності майбутнього педагога професійної школи до організації самостійної роботи учнів : дис. ... канд. пед. наук : 13.00.04 / Інститут педагогічної освіти і освіти дорослих імені І. Зязюна НАПН України. – Київ, 2018. – 313 с.
51. Ємчик О. Г. Розвиток творчого потенціалу магістрів дошкільної освіти у процесі професійної підготовки : дис. ... канд. пед. наук : 13.00.04 /

Східноєвропейський національний університет імені Лесі Українки. – Луцьк, 2018. – 318 с.

52. Коваль І. С. Формування професійної готовності майбутніх рятувальників до діяльності в екстремальних умовах : дис. ... канд. пед. наук : 13.00.04 / Львівський державний університет безпеки життєдіяльності. – Львів, 2017. – 43 с.

53. Горбатюк Р. М. Визначення готовності майбутніх інженерів-педагогів комп'ютерного профілю до професійної діяльності // Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців : методологія, теорія, досвід, проблеми. – 2012. – Вип. 32. – С. 279–283.

54. Бондаренко Т. С. Формування готовності до розробки та використання комп'ютерних навчальних систем у майбутніх інженерів-педагогів : автореф. дис. ... канд. пед. наук : 13.00.04 / Інститут професійно-технічної освіти Національної академії педагогічних наук України. – Київ, 2012. – 22 с.

55. Потапчук О. І. Формування готовності майбутніх інженерів-педагогів до професійної діяльності засобами інформаційно-комунікаційних технологій : автореф. дис. ... канд. пед. наук : 13.00.04 / Тернопільський національний педагогічний університет імені В. Гнатюка, Національний університет водного господарства та природокористування. – Рівне, 2016. – 22 с.

56. Литвиненко О. В. Підготовка майбутніх інженерів-педагогів до професійно-педагогічного проектування : автореф. дис. ... канд. пед. наук : 13.00.04 / Сумський державний педагогічний університет імені А. С. Макаренка. – Суми, 2018. – 22 с.

57. Мефанік М. С. Підготовка майбутніх інженерів-педагогів до використання комунікативних технологій у професійному навчанні учнів професійно-технічних навчальних закладів : автореф. дис. ... канд. пед. наук : 13.00.04 / Українська інженерно-педагогічна академія, Харків ; ВНЗ «Університет імені Альфреда Нобеля», Дніпро. – [Харків ; Дніпро], 2020. – 23 с.

58. Дудукалова О. С. Формування готовності майбутніх інженерів-педагогів економічного профілю до професійної діяльності : автореф. дис. ... канд. пед. наук : 13.00.04 / Бердянський державний педагогічний університет. – Бердянськ, 2020. – 23

с.

59. Шандрук С. К. Роль готовності до професійної діяльності у розвитку творчих здібностей майбутніх практичних психологів // Вісник ХНПУ імені Г. С. Сковороди. Психологія. – 2015. – Вип. 51. – С. 276–284.

60. John O. P., Srivastava S. The Big-Five trait taxonomy: History, measurement, and theoretical perspectives // Handbook of Personality: Theory and Research / ed. by L. A. Pervin, O. P. John. – 2nd ed. – New York : Guilford Press, 1999. – P. 102–138.

61. De Moura R. L., Carneiro T. C. J., Dias T. O. de L., Silva B. Personality traits of project managers: differences in knowledge, experience, and business sector // Revista de Administração da Universidade Federal de Santa Maria. – 2019. – Vol. 12, № 4. – P. 767–784.

62. Perminova O., Gustafsson M., Wikström K. Defining uncertainty in projects – a new perspective // International Journal of Project Management. – 2008. – Vol. 26, № 1. – P. 73–79. – DOI: 10.1016/j.ijproman.2007.08.005

63. Smith E. K., Bird C., Zimmermann T. Beliefs, practices, and personalities of software engineers // Proceedings of the 9th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE '16). – 2016. – P. 107–113. – DOI: 10.1145/2897586.2897596

64. Stojiljković S. The Research of Behavioral Project Management Competencies // European Project Management Journal. – 2022. – Vol. 12, № 1. – P. 70–77. – DOI: 10.56889/lewr1634

65. Development of soft skills in computer science bachelors in the project learning process / O. G. Glazunova, O. P. Kuzminska, I. V. Voloshyna, N. I. Morze // Information Technologies and Learning Tools. – 2022. – Vol. 92, № 6. – P. 111–127. – DOI: 10.33407/itlt.v92i6.5029

66. Broy M. Domain Modeling and Domain Engineering: Key Tasks in Requirements Engineering // Perspectives on the Future of Software Engineering: Essays in Honor of Dieter Rombach / ed. by J. Münch, K. Schmid. – Berlin ; Heidelberg : Springer Berlin Heidelberg, 2013. – P. 15–30. – DOI: 10.1007/978-3-642-37395-4_2

67. Monkhouse P. My project is failing, it is not my fault // Proceedings of the PMI®

Global Congress 2015 – EMEA, London, England / Project Management Institute. – Newtown Square, PA : Project Management Institute, 2015.

68. Likert R. A Technique for the Measurement of Attitudes. New York : Archives of Psychology, 1932. 55 p. (Archives of Psychology ; № 140).

69. Tarantino A., Cernauskas D. Essentials of Risk Management in Finance. – Hoboken, New Jersey : John Wiley & Sons, Ltd, 2010. – 352 p.

70. Osterwalder A., Pigneur Y. Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers. – Hoboken, New Jersey : John Wiley & Sons, 2010. – 288 p.

71. Алексеева Г., Тихонова Л., Антоненко О., Овсянников О. Компетентнісний підхід у професійній освіті та інтеграція практико-орієнтованих задач для розвитку сучасної особистості // New Pedagogical Thought. – 2025. – № 121 (1). – С. 81–88.

72. Семененко А., Алексеева Г., Овсянников О. Вплив педагогічних наук на розвиток професійної освіти // Vzdelávanie a spoločnosť : vedecký zborník. – Prešov : Vydavateľstvo Prešovskej univerzity, 2024. – IX. – С. 94–98.

73. Svyrydiuk V., Luzan P., Svyrydiuk O., Titova O., Popova O. Electronic textbooks as means for developing professional competencies of engineering students // Lecture Notes in Mechanical Engineering : 3rd Grabchenko's International Conference on Advanced Manufacturing Processes / eds. V. Tonkonogyi, V. Ivanov, J. Trojanowska, G. Oborskyi, I. Pavlenko. – 2022. – P. 707–716. – DOI: https://doi.org/10.1007/978-3-030-91327-4_68.

74. Алексеева Г., Кулікова Г., Антоненко О., Овсянников О., Горбатюк Л. Використання мультимедійних технологій у викладанні стретчингу для здобувачів вищої та професійної освіти в умовах дистанційної освіти // Молодь і ринок. – 2025. – № 2 (234). – С. 62–71.

75. Горбатюк Р., Волкова Н., Загородній Р., Бурега Н., Замора Я. Модель підготовки майбутніх магістрів педагогічних спеціальностей засобами цифрових технологій // Наука і техніка сьогодні (Серія «Педагогіка»). – 2026. – № 1 (55). – С. 1193–1205.

76. Горбатюк Р., Кабак В., Тулашвілі Ю. Модель впровадження засобів

інформаційних технологій в підготовку майбутніх фахівців закладу вищої освіти // Освітологічний дискурс. – 2025. – Т. 49, № 2. – С. 15–23.

77. Горбатюк Р., Волкова Н., Кабак В., Проценко Д. Умови формування графічної компетентності майбутніх фахівців сфери цифрових технологій у закладах вищої освіти // Педагогіка безпеки : міжнародний науковий журнал. – 2024. – Т. 9, № 2. – С. 70–79.

78. Горбатюк Р., Замора Я., Сіткар С., Бурега Н. Технологія формування професіоналізму майбутніх фахівців професійної освіти засобами мультимедійних технологій // Молодь і ринок. – 2022. – № 5 (203).

79. Алексєєва Г. М., Щетиніна О. С. Персоналізоване навчання у професійній ІТ-освіті через інтерактивні цифрові ресурси та OPEN SOURCE-платформи // Суспільство та національні інтереси. – 2025. – № 8 (16). – С. 15–29. – DOI: [https://doi.org/10.52058/3041-1572-2025-8\(16\)-15-29](https://doi.org/10.52058/3041-1572-2025-8(16)-15-29).

80. Aliksieieva H., Ostenda A., Shchetynina O., Antonenko O., Ovsyannikov O. Applying the group discussion method in vocational educational institutions on the example of computer subjects // Scientific Notes of the Pedagogical Department. – 2022. – № 51. – P. 120–131. – DOI: <https://doi.org/10.26565/2074-8167-2022-51-14>.

81. Тітова О. А., Теловата М. Т. Розвиток підприємницької компетентності здобувачів вищої освіти інженерних спеціальностей // Вісник Глухівського національного педагогічного університету імені Олександра Довженка. Серія : Педагогічні науки. – 2022. – № 3 (50). – С. 18–25. – DOI: <https://doi.org/10.31376/2410-0897-2022-3-50-18-25>.

82. Aliksieieva H., Ostenda A., Kravchenko N., Antonenko O., Ovsyannikov O. Efficient targeted advertising: modern marketing for the educational sector // Scientific Notes of the Pedagogical Department. – 2023. – № 53. – P. 6–15.

83. Nagay I., Khalabuzar O., Aliksieieva H., Antonenko O., Ovsyannikov O. Peculiarities of the Formation of Students' Business Communication Skills within the Distance Learning // Education Research International. – 2023. – Article ID 9660270. – 8 p.

84. Shchetynina O., Kravchenko N., Horbatiuk L., Aliksieieva H., Mezhuiev V.

Trello as a Tool for the Development of Lifelong Learning Skills of Senior Students // Postmodern Openings. – 2022. – Vol. 13, № 2. – P. 143–167. – DOI: <https://doi.org/10.18662/po/13.2/447>.

85. Жигірь В., Онищенко С. Застосування активних методів навчання у викладанні дисциплін електроенергетичного циклу в закладах вищої освіти // Молодь і ринок. – 2024. – № 5 (225). – С. 7–11.

86. Жигірь В. І., Забеліна Ю. Практико-орієнтоване навчання здобувачів закладів вищої освіти дисциплінам електроенергетичного циклу // Молодь і ринок. – 2024. – № 9 (229). – С. 18–22.

87. Жигірь В., Перегудова В. Особливості організації технологічної практики бакалаврів професійної освіти за спеціалізацією // Молодь і ринок. – 2024. – № 10 (230). – С. 13–17.

88. Жигірь В., Онищенко С., Гуренко В. Дидактичні аспекти впровадження методу кейсів у викладання дисциплін електроенергетичного циклу // Молодь і ринок. – 2024. – № 6 (226). – С. 7–11.

89. Titova O., Ishchenko T., Yershova L., Bazyl L., Kruchek V. Entrepreneurial Mindset Development in Engineering Students Through a Business Canvas Approach // Advances in Design, Simulation and Manufacturing VI : DSMIE 2023 / eds. V. Ivanov et al. – Cham : Springer, 2023. – P. 367–376. – DOI: https://doi.org/10.1007/978-3-031-32767-4_38.

90. Zaitseva N. V., Symonenko S. V., Titova O. A., Osadchyi V. V., Osadcha K. P. Fostering communication and collaboration skills for computer science students by means of ICT tools // Proceedings of the 2nd Workshop on Digital Transformation of Education (DigiTransfEd 2023). – Ivano-Frankivsk : Vasyl Stefanyk Precarpathian National University, 2023. – URL: <https://ceur-ws.org/Vol-3553/paper9.pdf>.

91. Titova O., Luzan P., Davlatzoda Q. Q., Mosia I., Kabysh M. The Taxonomy Approach for Engineering Students' Outcomes Assessment // Advanced Manufacturing Processes IV : InterPartner 2022 / eds. V. Tonkonogyi et al. – Cham : Springer, 2023. – P. 345–354. – DOI: https://doi.org/10.1007/978-3-031-16651-8_36.

92. Словник української мови : в 11 т. / АН УРСР, Ін-т мовознавства ; за ред. І.

К. Білодіда. – Київ : Наукова думка, 1979. – Т. 10. – С. 441. – URL: <http://sum.in.ua/p/10/441/2> (дата звернення: 23.05.2026).

93. Брюханова Н. О. Теорія і методика проектування системи педагогічної підготовки майбутніх інженерів-педагогів : автореф. дис. ... д-ра пед. наук : 13.00.04 / Східноукраїнський національний університет імені Володимира Даля. – Луганськ, 2011. – 42 с.

94. Васильєв І. Б. Системно-компетентнісний підхід до формування змісту педагогічної підготовки майбутніх педагогів професійної школи // Проблеми інженерно-педагогічної освіти : зб. наук. праць Укр. інж.-пед. акад. – Харків, 2012. – № 37. – С. 18–24.

95. Докучаєва В. В. Проектування інноваційних педагогічних систем у сучасному освітньому просторі : монографія. – Луганськ, 2005. – 299 с.

96. Гончаренко С. У., Мальований Ю. І. Інтегроване навчання. За і проти // Освіта. – 1994. – № 15–16. – С. 5.

97. Про затвердження Національної рамки кваліфікацій : Постанова Кабінету Міністрів України від 23 листоп. 2011 р. № 1341 // Урядовий кур'єр. – 2011. – № 226. – URL: <http://zakon4.rada.gov.ua/laws/show/1341-2011-п> (дата звернення: 23.05.2026).

98. Клепко С. Ф. Методологія пізнання освіти: організувати хаос // Практична філософія. – 2003. – № 1. – С. 183–190.

99. Кушнір В. А. Характеристика особливостей освітніх систем // Педагогіка і психологія. – 1999. – № 4. – С. 83–91.

100. Вознюк О. В. Розвиток вітчизняної педагогічної думки: синергетичний підхід (др. пол. ХХ ст.) : навч.-метод. посіб. – Житомир : ЖДУ, 2008. – 150 с.

101. Бібик Н. М. Компетентнісний підхід : рефлексивний аналіз застосування // Компетентнісний підхід у сучасній освіті: світовий досвід та українські перспективи : Бібліотека з освітньої політики. – Київ : [б. в.], 2004. – С. 47–52.

102. Компетентнісний підхід у сучасній освіті : світовий досвід та українські перспективи : Бібліотека з освітньої політики / Н. М. Бібик, Л. С. Ващенко, О. І. Локшина та ін. ; під заг. ред. О. В. Овчарук. – Київ : «К.І.С.», 2004. – 112 с.

103. Єрмаков І. Г. Життєва компетентність особистості : від теорії до

практики : наук.-метод. зб. – Запоріжжя : [б. в.], 2005. – 640 с.

104. Овчарук О. В. Розвиток компетентнісного підходу: стратегічні орієнтири міжнародної спільноти // Компетентнісний підхід у сучасній освіті: світовий досвід та українські перспективи : Бібліотека з освітньої політики. – Київ : «К.І.С.», 2004. – С. 111.

105. Пометун О. І. Теорія та практика послідовної реалізації компетентнісного підходу в досвіді зарубіжних країн // Компетентнісний підхід у сучасній освіті: світовий досвід та українські перспективи : Бібліотека з освітньої політики / під заг. ред. О. В. Овчарук. – Київ : «К.І.С.», 2004. – С. 15–24.

106. Локшина О. І. Європейська довідкова система як інструмент упровадження компетентнісного підходу в освіту країн – членів Європейського Союзу // Педагогіка і психологія. – 2007. – № 1. – С. 131–142.

107. Студентоцентризм у системі забезпечення якості освіти в економічному університеті : зб. матеріалів всеукр. наук.-метод. конф. за міжнар. участю (Київ, 2–3 берез. 2016 р.). – Київ : КНЕУ, 2016. – С. 204.

108. Рашкевич Ю. М. Болонський процес та нова парадигма вищої освіти : практичний довідник. – Львів : Видавництво Львівської політехніки, 2014. – 166 с.

109. Національний освітній глосарій : вища освіта / авт.-уклад. : В. М. Захарченко, С. А. Калашнікова, В. І. Луговий та ін. ; за ред. В. Г. Кременя. – 2-ге вид., перероб. і доп. – Київ : ТОВ «Видавничий дім «Плеяди», 2014. – 100 с.

110. Lea S. J., Stephenson D., Troy J. Higher Education Students' Attitudes to Student Centred Learning: Beyond «educational bulimia» // Studies in Higher Education. – 2003. – Vol. 28, № 3. – P. 321–334.

111. Гончаренко С. У., Мальований Ю. І. Інтегроване навчання. За і проти // Освіта. – 1994. – № 15–16. – С. 5.

112. Гуревич Р. С. Теорія і практика навчання в професійно-технічних закладах : монографія. – Вінниця : [б. в.], 2009. – 410 с.

113. Дольнікова Л. В. Структурування змісту навчальних дисциплін: системний, інтегративний і диференційований підхід // Педагогіка і психологія професійної освіти. – 2008. – № 1. – С. 73–81.

114. Козловська І. Теоретичні і методичні основи викладання загальнотехнічних і спеціальних дисциплін: інтегративний підхід / І. Козловська, К. Ленік ; Ін-т педагогіки і психології проф. освіти АПН України, Львів. наук.-практ. центр, каф. основ техніки Люблін. політехн. ін-ту. – Львів : Євросвіт, 2003. – 248 с.
115. Коломієць Д. І. Інтеграція знань з природничо-математичних і спеціальних дисциплін у професійній підготовці учителя трудового навчання : автореф. дис. ... канд. пед. наук : 13.00.04 / Національний педагогічний університет імені М. П. Драгоманова. – Київ, 2001. – 20 с.
116. Собко Я. М. Теоретико-методичні основи впровадження інтегративних курсів у професійно-технічній освіті : навч.-метод. посіб. – Львів : Норма, 2014. – 136 с.
117. Якимович Т. Д. Інтеграція теоретичного і виробничого навчання у процесі професійної підготовки фахівців : автореф. дис. ... канд. пед. наук : 13.00.04 / Ін-т педагогіки і психології проф. освіти АПН України. – Київ, 2001. – 21 с.
118. Енциклопедія освіти / голов. ред. В. Г. Кремень ; Акад. пед. наук України. – Київ : Юрінком Інтер, 2008. – С. 356.
119. Бібік Г. В. Міждисциплінарна інтеграція як основа якісної математичної освіти майбутніх учителів фізики // Збірник наукових праць Херсонського державного університету. Серія : Педагогічні науки. – Херсон : ХДУ, 2014. – Вип. 66. – С. 247–253.
120. Опачко М. В. Системний та інтегративний підходи в освіті : метод. посіб. – Ужгород : УжНУ, 2016. – 69 с.
121. Горбатюк І. Оцінка вартості програмного забезпечення як методологічна проблема ІТ-галузі // Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації : матеріали III Міжнар. наук.-практ. інтер.-конф. (м. Запоріжжя, 30 верес. 2022 р.). – Запоріжжя, 2022. – С. 506–510. – URL: <https://elar.tsatu.edu.ua/server/api/core/bitstreams/1fbc8169-7391-446d-b8f2-8c4720b7ae98/content>.
122. Gorbatiuk I., Kryvylova O. Ukrainian IT students and project management: are they prepared for industry demands? // SWorldJournal. – 2026. – № 35-04. – P. 142–

161. – DOI: <https://doi.org/10.30888/2663-5712.2026-35-04-102>.

123. Горбатюк І. Особливості проектного управління в контексті інженерії програмного забезпечення // Наукові записки БДПУ. – 2022. – № 3. – С. 267–274. – DOI: <https://doi.org/10.31494/2412-9208-2022-1-3-267-274>.

124. Горбатюк І. Характеристика узгодженості підготовки майбутніх бакалаврів з інженерії програмного забезпечення з вимогами стейкхолдерів // Цифрові технології у професійній діяльності : матеріали Всеукр. наук.-практ. інтер.-конф. (м. Бердянськ, 12–13 трав. 2023 р.). – Бердянськ, 2023. – С. 142–144. – URL: <https://dspace.bdpu.org.ua/items/597751b9-03dd-47d2-9241-c119f9973eb5>.

125. Zhang J., Chen G., Zhang R., Chen Y., Yang H. Exploration of Software Engineering Teaching Cases Based on the OBE-PBL Model // Proceedings of the 2025 10th International Conference on Modern Management, Education and Social Sciences (MMET 2025). – Paris : Atlantis Press, 2025. – P. 284–291. – DOI: https://doi.org/10.2991/978-2-38476-475-4_34.

126. Luzik E., Akmaldinova O., Tereminko L. Developing software engineering students' readiness for professional mobility through blended learning // Advanced Education. – 2019. – Vol. 6, № 13. – P. 103–111. – DOI: <https://doi.org/10.20535/2410-8286.185230>.

127. Semerikov S., Striuk A., Striuk M., Striuk L., Shalatska H. Soft skills development among software engineering students // Journal of Information Technologies in Education (ITE). – 2021. – № 49. – С. 34–60. – DOI: 10.14308/ite000751.

128. Striuk A. M., Semerikov S. O. Professional competencies of future software engineers in the software design: teaching techniques // Proceedings of Student Workshop CS&SE@SW 2019, Kryvyi Rih, Ukraine, Nov. 29, 2019. – Vol. 2546. – С. 35–57. – URL: <http://ceur-ws.org/Vol-2546/paper02.pdf> (дата звернення: 23.05.2026).

129. Nazarenko I. Teaching speaking to software engineering students within ESP course: methodological, psychological, and motivational aspects // European Science. – 2024. – № 2 (sge28-02). – P. 116–152. – DOI: <https://doi.org/10.30890/2709-2313.2024-28-00-034>.

130. Iatsyshyn A. V., Kovach V. O., Lyubchak V. O. Application of augmented

reality technologies for education projects preparation // CEUR Workshop Proceedings : Vol. 2643 : Selected Papers of the 9th International Conference on Monitoring, Modeling & Management of Emergent Economy (M3E2 2020), Odessa, Ukraine, May 25-27, 2020. – 2020. – P. 62–71. – URL: <https://ceur-ws.org/Vol-2643/paper07.pdf> (дата звернення: 23.05.2026).

131. Anazifa R. D., Djukri D. Project-Based Learning and Problem-Based Learning: Are They Effective to Improve Student's Thinking Skills? // Jurnal Pendidikan IPA Indonesia. – 2017. – Vol. 6, № 2. – P. 346–355. – DOI: <https://doi.org/10.15294/jpii.v6i2.11100>.

132. Matahari D. B., Nurohman S., Jumadi J. Research Trends in Project-Based Learning Models in Facilitating 21st Century Skills: Systematic Literature Review // Jurnal Penelitian Pendidikan IPA. – 2023. – Vol. 9, № 4. – P. 1607–1614. – DOI: <https://doi.org/10.29303/jppipa.v9i4.2544>.

133. Dmitrenko N. The implementation of problem-based learning in Ukrainian higher educational institutions // Advanced Education. – 2016. – № 5. – P. 28–34. – DOI: <https://doi.org/10.20535/2410-8286.61834>.

134. Tardif M. Saberes Docentes e Formação Profissional. – Petrópolis, RJ : Vozes, 2002. – 328 p.

135. O'Grady M. J. Practical problem-based learning in computing education // ACM Transactions on Computing Education (TOCE). – 2012. – Vol. 12, № 3. – P. 10.1–10.15.

136. Pollard C. E. Lessons learned from client projects in an undergraduate project management course // Journal of Information Systems Education. – 2012. – Vol. 23, № 3. – P. 271–282.

137. Сидоренко О., Чуба В. Ситуаційна методика навчання: теорія і практика. – Київ, 2001. – 127 с.

138. Шеремета П. М., Канищенко Л. Г. Кейс-метод: з досвіду викладання в українській бізнес-школі. – Київ, 1999. – 80 с.

139. Richardson I., Delaney Y. Problem Based Learning in the Software Engineering Classroom // 2009 22nd Conference on Software Engineering Education and

Training. – 2009. – P. 174–181. – DOI: <https://doi.org/10.1109/CSEET.2009.34>.

140. Wilkerson J. R., Lang W. S. Rubrics that systematically identify areas for improvement // *Advances in global education and research* / ed. by W. B. James, C. Cobanoglu, M. Cavusoglu. – Vol. 4. – USF M3 Publishing, 2021. – P. 1–12. – DOI: <https://doi.org/10.5038/9781955833042>.

141. Brodie L., Zhou H., Gibbons A. Steps in developing an advanced software engineering course using problem based learning // *Engineering Education*. – 2008. – Vol. 3, № 1. – P. 2–12. – DOI: <https://doi.org/10.11120/ened.2008.03010002>.

142. Delaney D., Mitchell G. G. PBL Applied To Software Engineering Group Projects // *Proceedings of the 16th Conference on Software Engineering Education and Training (CSEET '03)*. – 2003. – P. 66–73.

143. Johnson J. R. Implementing Best Practice in Training Problem-Based Learning Tutors // *Journal of Problem-Based Learning*. – 2021. – Vol. 8, № 1. – P. 24–34. – DOI: <https://doi.org/10.24313/jpbl.2021.00017>.

144. Kalluri R. A human factors study of risk management of complex agile scrum projects in large enterprises // *International Journal of Business & Management Studies*. – 2022. – Vol. 3, № 8. – P. 38–44. – DOI: <https://doi.org/10.56734/ijbms.v3n8a6>.

145. Rational Badger. Harvard Case Method. Interactive Learning for Critical Thinkers // *Medium*. – 2023. – URL: <https://medium.com/@RationalBadger/harvard-case-method-a1d7fb2680e8> (дата звернення: 27.05.2026).

146. El-Khalili N. H. Teaching Agile Software Engineering Using Problem-Based Learning // *International Journal of Information and Communication Technology Education*. – 2013. – Vol. 9, № 1. – P. 1–12. – DOI: <https://doi.org/10.4018/jicte.2013010101>.

147. The Case Method : [проект]. – URL: <https://www.hbs.edu/case-method-project/about/Pages/case-method-teaching.aspx> (дата звернення: 27.05.2026).

148. Ukraine's tech market: 2025 report // N-iX. – 2025. – URL: <https://src.nix.com/uploads/2025/02/25/d3f17988-8246-40ae-9703-3cd14f6163ad.pdf> (дата звернення: 27.05.2026).

149. Kukuruza O., Golovkina N., Omelchenko N. Coping with a protracted crisis:

a case study of IT-Integrator in Ukraine // Emerald Emerging Markets Case Studies. – 2024. – Vol. 14, № 4. – DOI: <https://doi.org/10.1108/EEMCS-07-2024-0286>.

150. Facilitation skills for faculty // Center for Excellence in Teaching and Learning (CETL), Kennesaw State University. – URL: <https://campus.kennesaw.edu/faculty-staff/cetl/teaching-resources/facilitation.php> (дата звернення: 27.05.2026).

151. A checklist for facilitating online courses // Faculty Focus. – URL: <https://www.facultyfocus.com/articles/online-education/online-course-design-and-preparation/a-checklist-for-facilitating-online-courses/> (дата звернення: 27.05.2026).

152. Socratic questioning : [веб-ресурс] // Inside Product. – URL: <https://insideproduct.co/socratic-questioning/> (дата звернення: 27.05.2026).

153. Ambler S. W. Applying Agile Modeling in Software Projects: A Case Study // EIMJ. – URL: [https://eimj.org/uplode/images/photo/Applying Agile Modeling in Software Projects: A Case Study..pdf](https://eimj.org/uplode/images/photo/Applying_Agile_Modeling_in_Software_Projects:_A_Case_Study..pdf) (дата звернення: 27.05.2026).

154. Rubrics // Center for the Advancement of Teaching Excellence (CATE), University of Illinois Chicago. – URL: <https://teaching.uic.edu/cate-teaching-guides/assessment-grading-practices/rubrics/> (дата звернення: 27.05.2026).

155. Rubrics to evaluate student work // Teaching Engagement Program, University of Oregon. – URL: <https://teaching.uoregon.edu/resources/rubrics-evaluate-student-work> (дата звернення: 27.05.2026).

156. Richardson I., Delaney Y. Problem Based Learning in the Software Engineering Classroom // 2009 22nd Conference on Software Engineering Education and Training. – 2009. – P. 174–181. – DOI: <https://doi.org/10.1145/3328778.3366948>.

157. International Grade Equivalencies // IFMA Foundation. – URL: <https://foundation.ifma.org/wp-content/uploads/2019/11/International-Grade-Equivalencies.pdf> (дата звернення: 27.05.2026).

158. Ribeiro L. C. The Pros and Cons of Problem-Based Learning from the Teacher's Standpoint // Journal of University Teaching & Learning Practice. – 2011. – Vol. 8, № 1. – P. 34–51. – DOI: <https://doi.org/10.53761/1.8.1.4>.

159. Гончаренко С. У. Український педагогічний словник / гол. ред. С. Головка. – Київ : Либідь, 1997. – 373 с.
160. Яцик Т. О., Степанюк В. В. Словник коротких термінів з педагогіки. – Луцьк : ФОП Мажула Ю. М., 2022. – 50 с.
161. Словник базових понять з курсу «Педагогіка» : навч. посіб. для студ. вищ. навч. заклад. / уклад. О. Є. Антонова. – 2-ге вид., доп. і перероб. – Житомир : Вид-во ЖДУ ім. І. Франка, 2014. – 100 с.
162. Горбатюк Р. М., Кабак В. В. Підготовка майбутніх інженерів-педагогів до професійної діяльності засобами комп'ютерних технологій : монографія. – Луцьк : ВМА «ТЕРЕН», 2015. – 264 с.
163. Короткий тлумачний словник психолого-педагогічних термінів (для аспірантів і магістрів) / укл. І. В. Козубовська, О. С. Повідайчик. – Ужгород : ДВНЗ «Ужгородський національний університет», 2021. – 41 с.
164. Словник термінів і понять сучасної освіти / уклад. : Л. М. Михайлова, О. В. Пагава, О. В. Проніна ; за заг. ред. Л. М. Михайлової. – Сєверодонецьк : [б. в.], 2020. – 194 с.
165. Стрюк А. М. Теоретико-методичні засади комбінованого навчання системного програмування майбутніх фахівців з програмної інженерії : монографія. – Кривий Ріг : Видавничий відділ ДВНЗ «Криворізький національний університет», 2015. – Т. VI, вип. 1 (6). – 286 с.
166. Круглик В. С. Система підготовки майбутніх інженерів-програмістів до професійної діяльності у вищих навчальних закладах : автореф. дис. ... докт. пед. наук : 13.00.04 / Запорізький національний університет. – Запоріжжя, 2018. – 43 с.
167. Малезик П. М. Теоретичні й методичні засади технічної підготовки майбутніх фахівців з інформаційних технологій : автореф. дис. ... докт. пед. наук : 13.00.02 / Національний педагогічний університет імені М. П. Драгоманова. – Київ, 2020. – 36 с.
168. Про схвалення концепції Державної цільової соціальної програми розвитку професійної (професійно-технічної) освіти на 2022–2027 рр. : Розпорядження КМ України від 09.12.2021 р. № 1619-р // Верховна Рада України. –

URL : <https://zakon.rada.gov.ua/laws/show/1619-2021-p#Text> (дата звернення: 27.05.2026).

169. Горбатюк І. Організація підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління в умовах педагогічного експерименту. *Інноваційна педагогіка*. 2026. Вип. 92, том 1. С. 220–225. <https://doi.org/10.32782/ip/92.1.45>

170. Горбатюк І. Структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Актуальні питання гуманітарних наук*. Вип. 95, том 1, 2026. С. 303–309. <https://doi.org/10.24919/2308-4863/95-1-43>

171. Горбатюк І. Функціональна основа проєктного управління майбутніх бакалаврів з інженерії програмного забезпечення. *Освіта та розвиток дитини НАПН України*. 2023. № 3 (90). С. 22-26. [https://doi.org/10.32405/2309-3935-2023-3\(90\)-22-26](https://doi.org/10.32405/2309-3935-2023-3(90)-22-26)

172. Горбатюк І. Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління на основі проблемного навчання. *Вісник КрНУ імені Михайла Остроградського*. Вип. 6/2025(155). С. 32–39. <https://doi.org/10.32782/1995-0519.2025.6.4>

173. Горбатюк І. А., Кривильова О. А. Trello як засіб формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Наукові підсумки 2024 року: матеріали XIII наукової конференції*. Харків, 2024. С. 44. URL: https://entc.com.ua/download/pidsumki_2024.pdf

174. Горбатюк І. Зміст підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Наукові засади підготовки фахівців інженерно-педагогічного та технологічного напрямків* : матеріали V Всеукр. наук.-практ. інтер.-конф., м. Запоріжжя, 15 трав. 2024 р. Запоріжжя, 2024. С. 21-24. URL: <https://dspace.bdpu.org.ua/items/138737b1-cba5-4143-ad97-5b07948a38b8>

175. Горбатюк І. А. Особистісний складник готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Науково-дослідна*

робота в системі підготовки фахівці-педагогів у природничій, технологічній і комп'ютерній галузях : матеріали ІХ Всеукр. наук.-практ. інтер.-конф., м. Запоріжжя, 21-22 верес. 2023 р. Запоріжжя, 2023. С. 52-54.
URL: <https://dspace.bdpu.org.ua/items/f1a5d847-7f15-4957-a151-3f2be02d2f7c>

176. Горбатюк І. А., Кривильова О. А. *Методологічні підходи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Актуальні проблеми та перспективи технологічної і професійної освіти* : матеріали ІХ Всеукр. наук.-практ. конф., м. Тернопіль, 23-24 квіт. 2026 р. Тернопіль, 2026. С. 109-112.
URL: <https://drive.google.com/drive/folders/1q3Mi3PD5SDTV0thQHZGju3b8vHdGGGgJ>

ДОДАТКИ

Додаток А

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні наукові результати дисертації:

1. Gorbatiuk I., Kryvylova O. Ukrainian it students and project management: are they prepared for industry demands? *SWorldJournal*. 2026. № 35-04. С. 142-161. <https://doi.org/10.30888/2663-5712.2026-35-04-102>

2. Горбатюк І. Організація підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління в умовах педагогічного експерименту. *Інноваційна педагогіка*. 2026. Вип. 92, том 1. С. 220–225. <https://doi.org/10.32782/ip/92.1.45>

3. Горбатюк І. Особливості проєктного управління в контексті інженерії програмного забезпечення. *Наукові записки БДПУ*. 2022. № 3. С. 267–274. <https://doi.org/10.31494/2412-9208-2022-1-3-267-274>.

4. Горбатюк І. Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління на основі проблемного навчання. *Вісник КрНУ імені Михайла Остроградського*. Вип. 6/2025(155). С. 32–39. <https://doi.org/10.32782/1995-0519.2025.6.4>

5. Горбатюк І. Структурно-функціональна модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Актуальні питання гуманітарних наук*. Вип. 95, том 1, 2026. С. 303–309. <https://doi.org/10.24919/2308-4863/95-1-43>

6. Горбатюк І. Функціональна основа проєктного управління майбутніх бакалаврів з інженерії програмного забезпечення. *Освіта та розвиток дитини НАПН України*. 2023. № 3 (90). С. 22-26. [https://doi.org/10.32405/2309-3935-2023-3\(90\)-22-26](https://doi.org/10.32405/2309-3935-2023-3(90)-22-26)

Наукові праці, які засвідчують апробацію матеріалів дисертації:

1. Горбатюк І. Характеристика узгодженості підготовки майбутніх

бакалаврів з інженерії програмного забезпечення з вимогами стейкхолдерів. *Цифрові технології у професійній діяльності* : матеріали Всеукр. наук.-практ. інтер.-конф., м. Бердянськ, 12-13 травня 2023 р. Бердянськ, 2023. С. 142-144
URL: <https://dspace.bdpu.org.ua/items/597751b9-03dd-47d2-9241-c119f9973eb5>

2. Горбатюк І. Оцінка вартості програмного забезпечення як методологічна проблема ІТ-галузі. *Розвиток сучасної науки та освіти: реалії, проблеми якості, інновації* : матеріали III Міжнар. наук.-практ. інтер.-конф., м. Запоріжжя, 30 верес. 2022 р. Запоріжжя, 2022. С. 506-510.
URL: <https://elar.tsatu.edu.ua/server/api/core/bitstreams/1fbc8169-7391-446d-b8f2-8c4720b7ae98/content>

3. Горбатюк І. Зміст підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Наукові засади підготовки фахівців інженерно-педагогічного та технологічного напрямків* : матеріали V Всеукр. наук.-практ. інтер.-конф., м. Запоріжжя, 15 трав. 2024 р. Запоріжжя, 2024. С. 21-24. URL: <https://dspace.bdpu.org.ua/items/138737b1-cba5-4143-ad97-5b07948a38b8>

4. Горбатюк І. А., Кривильова О. А. Trello як засіб формування готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Наукові підсумки 2024 року*: матеріали XIII наукової конференції. Харків, 2024. С. 44. URL: https://entc.com.ua/download/pidsumki_2024.pdf

5. Горбатюк І. А. Особистісний складник готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Науково-дослідна робота в системі підготовки фахівці-педагогів у природничій, технологічній і комп'ютерній галузях* : матеріали IX Всеукр. наук.-практ. інтер.-конф., м. Запоріжжя, 21-22 верес. 2023 р. Запоріжжя, 2023. С. 52-54.
URL: <https://dspace.bdpu.org.ua/items/f1a5d847-7f15-4957-a151-3f2be02d2f7c>

6. Горбатюк І. А., Кривильова О. А. Методологічні підходи підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. *Актуальні проблеми та перспективи технологічної і професійної освіти* : матеріали IX Всеукр. наук.-практ. конф., м. Тернопіль, 23-24 квіт. 2026 р.

Тернопіль,

2026.

С. 109-112.

URL: <https://drive.google.com/drive/folders/1q3Mi3PD5SDTVOthQHZGju3b8vHdGGGgJ>

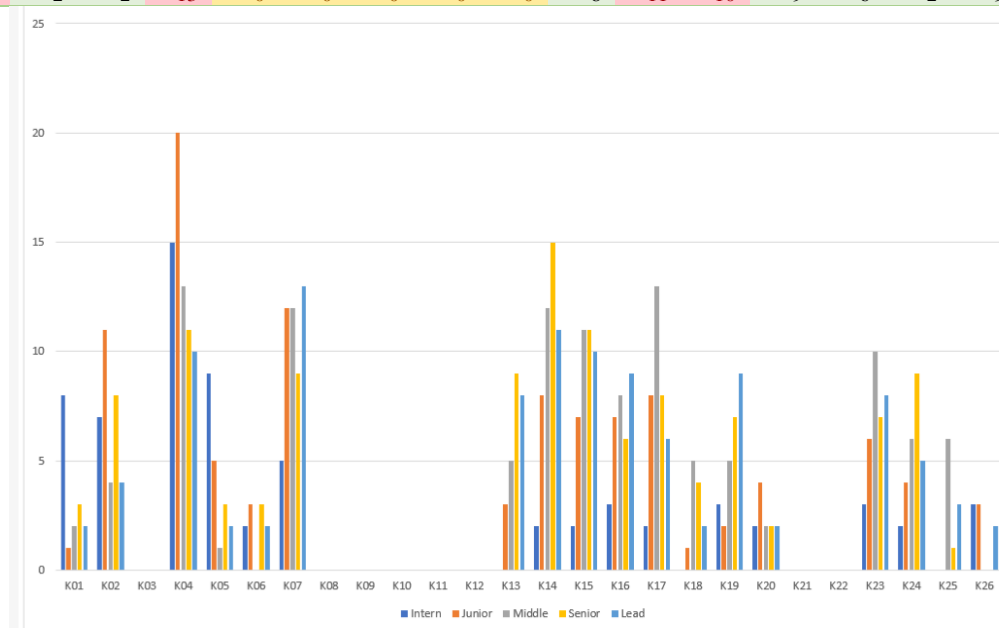
Додаток Б

Порівняльний аналіз вимог стейкхолдерів та компетентностей стандарту вищої освіти спеціальності 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня вищої освіти

Таблиця Б.1

Матриця узгодженості вимог стейкхолдерів і компетентностей стандарту вищої освіти спеціальності 121 «Інженерія програмного забезпечення»

Position	Count	K01	K02	K03	K04	K05	K06	K07	K08	K09	K10	K11	K12	K13	K14	K15	K16	K17	K18	K19	K20	K21	K22	K23	K24	K25	K26
Intern	10	8	7	0	15	9	2	5	0	0	0	0	0	0	2	2	3	2	0	3	2	0	0	3	2	0	3
Junior	18	1	11	0	20	5	3	12	0	0	0	0	0	3	8	7	7	8	1	2	4	0	0	6	4	0	3
Middle	10	2	4	0	13	1	0	12	0	0	0	0	0	5	12	11	8	13	5	5	2	0	0	10	6	6	0
Senior	10	3	8	0	11	3	3	9	0	0	0	0	0	9	15	11	6	8	4	7	2	0	0	7	9	1	0
Lead	7	2	4	0	10	2	2	13	0	0	0	0	0	8	11	10	9	6	2	9	2	0	0	8	5	3	2



URL	Company	Position	Req	K01.	K02.	K03.	K04.	K05.	K06.	K07.	K08.	K09.	K10.	K11.	K12.	K13.	K14.	K15.	K16.	K17.	K18.	K19.	K20.	K21.	K22.	K23.	K24.	K25.	K26.	Have	Comment	
https://jobs.dou.ua/companies/n-ix/vacancies/178623/	N-IX	Middle	N	N	N	N	F	P	N	N	N	N	N	N	N	P	F	F	F	F	F	F	N	N	P	F	F	N	N		SCRUM	
https://jobs.dou.ua/companies/n-ix/vacancies/178624/	N-IX	Senior	Y	P	P	N	F	N	F	P	N	N	N	N	N	P	F	N	P	P	F	P	F	N	N	P	N	P	N	M	TEAM PLAYER	
https://jobs.dou.ua/companies/n-ix/vacancies/178607/	N-IX	Senior	Y	F	F	N	P	F	N	F	N	N	N	N	N	P	F	P	F	F	N	N	N	N	N	P	F	N	N		SCRUM	
https://jobs.dou.ua/companies/n-ix/vacancies/178606/	N-IX	Middle	Y	N	N	N	F	N	N	F	N	N	N	N	N	N	F	F	P	P	F	P	N	N	N	N	F	P	F	N	SCRUM	
https://jobs.dou.ua/companies/n-ix/vacancies/178605/	N-IX	Middle	Y	F	N	N	F	N	N	F	N	N	N	N	N	F	F	F	F	F	N	N	N	N	N	N	F	F	N	N	AGILE	
https://jobs.dou.ua/companies/n-ix/vacancies/178600/	N-IX	Middle	Y	N	N	N	N	N	N	F	N	N	N	N	N	F	F	F	F	F	N	N	N	N	N	N	N	N	N	N	AGILE	
https://jobs.dou.ua/companies/n-ix/vacancies/178587/	N-IX	Lead	Y	N	N	N	F	F	F	F	N	N	N	N	N	F	F	F	F	F	N	N	N	N	N	N	F	F	F	N	ESTIMATES	
https://jobs.dou.ua/companies/sysgears/vacancies/203078/	SysGears	Intern	N	F	F	N	F	F	N	N	N	N	N	N	N	N	N	F	N	N	N	N	N	N	N	N	N	N	N	F	N	
https://jobs.dou.ua/companies/dok/vacancies/227273/	dok.ua	Junior	Y	N	N	N	N	N	N	F	N	N	N	N	N	N	P	P	P	P	N	P	N	N	N	N	P	F	N	N	SCRUM	
https://jobs.dou.ua/companies/abp/vacancies/160410/	ABP	Intern	Y	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/privatbank/vacancies/149205/	PrivatBank	Middle	Y	N	N	N	N	N	N	F	N	N	N	N	N	N	P	N	P	N	N	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/veloxpro/vacancies/110673/	Bookmap	Junior	Y	N	P	N	F	F	P	P	N	N	N	N	N	N	P	N	N	N	N	N	N	N	N	N	N	N	P	N	TEAM PLAYER	
https://jobs.dou.ua/companies/fintech-band/vacancies/202674/	Monobank	Senior	N	N	P	N	F	N	N	N	N	N	N	N	N	N	N	N	N	N	P	N	P	N	N	N	N	N	N	N		
https://jobs.dou.ua/companies/botscrew/vacancies/227233/	BotsCrew	Junior	Y	N	P	N	F	N	N	P	N	N	N	N	N	F	F	N	P	N	N	N	N	N	N	N	P	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/distributedlab/vacancies/227082/	Distributed Lab	Intern	N	P	N	N	F	F	P	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N		
https://jobs.dou.ua/companies/dedicattd/vacancies/217483/	Dedicattd	Junior	Y	N	F	N	N	N	N	P	N	N	N	N	N	N	N	P	N	F	N	N	N	N	N	N	N	P	N	N	B	TEAM PLAYER
https://jobs.dou.ua/companies/futurra-group/vacancies/221556/	Futura	Junior	N	N	P	N	N	N	N	N	N	N	N	N	N	N	P	P	N	P	N	N	N	N	N	N	P	N	N	N		
https://djinni.co/jobs/516259-senior-net-developer-relocate-to-poland/	Luxoft	Senior	Y	N	P	N	N	P	N	P	N	N	N	N	N	P	P	P	P	N	N	N	N	N	N	N	F	F	N	N	AGILE	
https://djinni.co/jobs/492170-devops-engineer/	Cprime Studios	Middle	N	N	P	N	P	N	N	N	N	N	N	N	N	N	P	P	N	N	N	N	N	N	N	N	P	N	F	N		
https://djinni.co/jobs/516249-fullstack-developer-node-js-react/	Sloboda studio	Middle	Y	N	P	N	F	N	N	P	N	N	N	N	N	P	P	N	N	F	N	N	N	N	N	N	F	P	N	N	TEAM PLAYER	
https://djinni.co/jobs/420433-junior-strong-node-js-developer/	Techlexity	Junior	Y	N	P	N	N	N	N	F	N	N	N	N	N	N	P	P	P	P	P	P	N	N	N	N	P	N	N	N	TEAM PLAYER	
https://djinni.co/jobs/402073-middle-node-js-developer/	Techlexity	Middle	Y	N	P	N	N	N	N	F	N	N	N	N	N	P	P	P	P	P	P	P	N	N	N	N	P	N	N	N	TEAM PLAYER	
https://djinni.co/jobs/448871-react-native-developer/	Techlexity	Middle	Y	N	P	N	F	N	N	P	N	N	N	N	N	N	P	N	N	N	P	N	P	N	N	N	P	N	N	N	TEAM PLAYER	
https://djinni.co/jobs/448295-senior-net-react-js-developer/	JustProtect	Senior	Y	N	N	N	F	N	N	F	N	N	N	N	N	N	N	N	P	F	P	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://djinni.co/jobs/516199-senior-net-developer-sql-higher-education/	Luxoft	Senior	Y	N	P	N	F	N	N	P	N	N	N	N	N	P	F	F	N	P	N	P	N	N	N	N	P	P	N	N	AGILE	
https://djinni.co/jobs/516194-senior-net-developer/	Luxoft	Senior	Y	N	P	N	N	N	N	P	N	N	N	N	N	P	F	F	N	P	N	P	N	N	N	N	P	P	N	N	AGILE	
https://djinni.co/jobs/504217-tehnikhii-kerivnik-python-tech-lead/	TECHIIA Holding	Lead	Y	N	P	N	F	N	N	F	N	N	N	N	N	F	F	F	F	F	P	N	P	N	N	N	P	N	N	N	AGILE	
https://djinni.co/jobs/498355-python-team-lead/	PLANEKS	Lead	Y	N	N	N	N	N	N	F	N	N	N	N	N	P	F	F	P	F	N	N	N	N	N	N	P	N	P	N	AGILE	
https://djinni.co/jobs/516080-lead-net-developer/	CHI Software	Lead	Y	N	P	N	F	N	N	F	N	N	N	N	N	F	F	F	F	N	N	P	N	N	N	N	F	F	N	N	AGILE	
https://djinni.co/jobs/516032-react-redux-typescript-team-lead/	Akmeator	Lead	Y	N	P	N	F	N	N	F	N	N	N	N	N	P	P	P	P	N	N	P	N	N	N	N	P	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/relokia/vacancies/165449/?from=first-job	Relokia	Junior	Y	N	P	N	F	N	N	P	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/ajax-systems/vacancies/219347/?from=first-job	Ajax Systems	Junior	Y	N	N	N	N	N	N	P	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	P	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/hys-enterprise/vacancies/226124/?from=first-job	HYS Enterprise	Intern	N	F	N	N	F	N	P	N	N	N	N	N	N	N	N	N	N	P	P	N	N	N	N	N	N	N	N	P		
https://jobs.dou.ua/companies/bart-solutions/vacancies/156895/	bART Solutions	Junior	N	P	N	N	F	N	N	N	N	N	N	N	N	N	N	N	P	F	N	N	F	N	N	N	N	N	N	F	N	
https://jobs.dou.ua/companies/lazy-ants/vacancies/120714/	Lazy Ants	Middle	Y	N	N	N	F	N	N	N	N	N	N	N	N	N	N	N	N	P	N	N	N	N	N	N	N	N	N	N	AGILE	
https://jobs.dou.ua/companies/lambda-team/vacancies/83112/	Lambda Team	Intern	N	N	N	N	F	F	N	N	N	N	N	N	N	N	N	N	N	N	N	P	N	N	N	N	N	N	N	N		
https://jobs.dou.ua/companies/leadsdoit/vacancies/225130/	Leadsdoit	Junior	N	N	N	N	N	N	F	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/kitrum/vacancies/225274/	KitRUM	Junior	Y	N	N	N	F	F	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	PLANNING	
https://jobs.dou.ua/companies/amo/vacancies/206247/	AMO	Junior	Y	N	P	N	F	N	N	P	N	N	N	N	N	N	N	N	N	P	N	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/embrox-solutions/vacancies/226612/	Embrox Solutions	Intern	Y	N	P	N	N	N	N	P	N	N	N	N	N	N	P	N	N	N	N	N	N	N	N	N	N	P	N	N	B	TEAM PLAYER
https://jobs.dou.ua/companies/embrox-solutions/vacancies/204816/	Embrox Solutions	Junior	Y	N	N	N	N	N	N	P	N	N	N	N	N	N	P	P	N	N	N	N	N	N	N	N	N	N	N	N	M	TEAM PLAYER
https://jobs.dou.ua/companies/cloudprinter-com/vacancies/221131/	Cloudprinter.com	Junior	N	N	N	N	F	N	N	N	N	N	N	N	N	N	P	P	N	N	N	N	N	N	N	N	N	P	N	N		
https://jobs.dou.ua/companies/brightech/vacancies/217064/	Brightech	Junior	N	N	P	N	F	N	N	N	N	N	N	N	N	N	P	N	P	N	N	N	N	N	N	N	N	N	N	N		
https://jobs.dou.ua/companies/pushka-studios/vacancies/219129/	Pushka Studios	Junior	N	N	P	N	F	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	P	N	N	N	N	N	N	N		
https://jobs.dou.ua/companies/moduleworks-gmbh/vacancies/224135/	ModuleWorks	Junior	Y	N	N	N	F	N	N	P	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	P	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/appflame/vacancies/187094/	appflame	Junior	N	N	P	N	N	P	N	N	N	N	N	N	N	N	P	N	P	N	N	N	P	N	N	N	N	N	N	N		
https://jobs.dou.ua/companies/serverless-team/vacancies/234677/?from=list_hot	Serverless Team	Intern	Y	N	P	N	P	P	N	P	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	TEAM PLAYER	
https://jobs.dou.ua/companies/hevelian/vacancies/166532/	Hevelian	Intern	Y																													

Compentencies (Стандарт вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня вищої освіти)

- K01. Здатність до абстрактного мислення, аналізу та синтезу.
- K02. Здатність застосовувати знання у практичних ситуаціях.
- K03. Здатність спілкуватися державною мовою як усно, так і письмово.
- K04. Здатність спілкуватися іноземною мовою як усно, так і письмово.
- K05. Здатність вчитися і оволодівати сучасними знаннями.
- K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.
- K07. Здатність працювати в команді.
- K08. Здатність діяти на основі етичних міркувань.
- K09. Прагнення до збереження навколишнього середовища.
- K10. Здатність діяти соціально відповідально та свідомо.
- K11. Здатність реалізувати свої права і обов'язки як члена суспільства, усвідомлювати цінності громадянського (вільного демократичного) суспільства та необхідність його сталого розвитку, верховенства права, прав і свобод людини і громадянина в Україні.
- K12. Здатність зберігати та примножувати моральні, культурні, наукові цінності і досягнення суспільства на основі розуміння історії та закономірностей розвитку предметної області, її місця у загальній системі знань про природу і суспільство та у розвитку суспільства, техніки і технологій, використовувати різні види та форми рухової активності для активного відпочинку та ведення здорового способу життя. Спеціальні (фахові, предметні) компетентності
- K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.
- K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.
- K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.
- K16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.
- K17. Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу.
- K18. Здатність аналізувати, вибирати і застосовувати методи і засоби для забезпечення інформаційної безпеки (в тому числі кібербезпеки).
- K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.
- K20. Здатність застосовувати фундаментальні і міждисциплінарні знання для успішного розв'язання завдань інженерії програмного забезпечення.
- K21. Здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.
- K22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.
- K23. Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення.
- K24. Здатність здійснювати процес інтеграції системи, застосовувати стандарти і процедури управління змінами для підтримки цілісності, загальної функціональності і надійності програмного забезпечення.
- K25. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.
- K26. Здатність до алгоритмічного та логічного мислення.

Додаток В

Діагностика готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Таблиця В.1

Критерії, показники та рівні сформованості готовності майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління

Показники	Рівні		
	Високий	Середній	Початковий
Особистісний компонент			
Відкритість досвіду (допитливість і інтерес до нових ідей; гнучкість та адаптивність)	Проявляється у вираженій готовності активно опановувати нові методології управління проєктами (Agile, Scrum, Waterfall тощо), інструменти командної роботи та сучасні технології розроблення ПЗ, у здатності швидко адаптуватися до змін вимог проєкту, а також у ініціативності щодо впровадження інноваційних рішень у проєктну діяльність.	Загалом позитивно ставиться до нових підходів у проєктному управлінні та розробленні програмного забезпечення, однак застосовує їх переважно за наявності чітких інструкцій або підтримки з боку більш досвідчених членів команди. У таких умовах перевага частково надається вже відомим і перевіреним практикам, хоча можливе поступове залучення нових інструментів.	Спостерігається обмежена готовність до сприйняття інновацій у сфері проєктного управління, орієнтація на традиційні підходи, уникнення невизначених або нестандартних ситуацій, а також труднощі з адаптацією до змін вимог проєкту, технологій чи методологій розроблення програмного забезпечення.
Сумлінність (організованість, увага до деталей, відповідальність, довгострокове планування)	Характеризується чіткою організованістю, високим рівнем самоконтролю, відповідальним ставленням до	Проявляється у загалом відповідальному ставленні до навчально-проєктної діяльності, однак із періодичними	Недостатня організованість, низька послідовність у виконанні завдань, схильність до спонтанної поведінки без належного

	виконання проєктних завдань, системністю у плануванні діяльності та послідовністю у досягненні цілей. Здобувачі освіти демонструють надійність у командній роботі, проте іноді можуть виявляти надмірну принциповість і ригідність у дотриманні встановлених процедур.	коливаннями в організованості та самодисципліні. Такі здобувачі освіти здатні працювати за планом, але за певних умов допускають спонтанні рішення та потребують зовнішньої підтримки для стабільного дотримання термінів і вимог.	планування, а також знижена надійність у дотриманні зобов'язань, що може негативно впливати на ефективність участі у проєктній діяльності.
Екстраверсія (комунікабельність, наполегливість, широка соціальна мережа, перевага до групових активностей)	Характеризується високою комунікабельністю, ініціативністю у встановленні професійних контактів, активною участю в командній роботі та обговореннях, схильністю до лідерських позицій і впливу на групові рішення. Такі здобувачі освіти легко взаємодіють із членами команди та зацікавленими сторонами, однак іноді можуть демонструвати надмірну домінантність у комунікації.	Проявляється у збалансованому поєднанні соціальної активності та індивідуальної роботи. Здобувачі освіти здатні ефективно працювати в команді, брати участь у комунікації та обговореннях, але водночас не завжди ініціюють взаємодію і можуть потребувати додаткової мотивації для активного залучення в групові процеси.	Спостерігається схильність до стриманої комунікації, обмеженої соціальної активності, уникнення надмірної взаємодії в команді, що може сприйматися як замкнутість або дистанційованість. Такі здобувачі освіти частіше орієнтуються на індивідуальну роботу та менш активно залучаються до групових обговорень у проєктному середовищі.
Доброзичливість (кооперативність)	Характеризується вираженою кооперативністю,	Проявляється у загалом позитивному ставленні до	Спостерігається стриманість у міжособистісній

	емпатійністю, відкритістю до співпраці та високим рівнем довіри до членів команди. Такі здобувачі освіти активно сприяють формуванню позитивного соціально-психологічного клімату в проєктній групі, охоче допомагають іншим і прагнуть досягнення консенсусу, однак іноді можуть уникати конфліктних ситуацій навіть тоді, коли вони є конструктивно необхідними.	командної взаємодії та готовності до співпраці, однак за певних умов можливі прояви обережності у довірі до інших або ситуативна орієнтація на власні інтереси. Такі здобувачі освіти здатні працювати в команді, але ступінь їхньої залученості до підтримки інших учасників може варіюватися залежно від ситуації.	взаємодії, підвищена критичність або недовіра до інших учасників команди, схильність до конкурентної поведінки та обмежена готовність до співпраці. Це може ускладнювати командну роботу та знижувати ефективність взаємодії в умовах проєктного управління.
Невротизм (емоційний контроль, стресостійкість, адаптованість)	На високому рівні емоційної стабільності (низькому рівні невротизму) майбутній фахівець характеризується спокійністю, врівноваженістю, стійкістю до стресових ситуацій, здатністю зберігати ефективність діяльності в умовах невизначеності та тиску. Водночас така стабільність іноді може сприйматися як емоційна стриманість або недостатня чутливість до зовнішніх подій.	На середньому рівні спостерігається відносна емоційна рівновага: здобувач освіти загалом здатний контролювати власні емоційні реакції, однак у складних або конфліктних проєктних ситуаціях можливі прояви тривожності, напруження чи зниження впевненості, що потребує додаткових ресурсів саморегуляції.	На початковому рівні емоційної стабільності (високому рівні невротизму) характерними є підвищена емоційна реактивність, схильність до тривожності, дратівливості та стресових переживань, а також труднощі у контролі емоцій у напружених проєктних ситуаціях, що може негативно впливати на ефективність участі в командній роботі та прийняття рішень.
Функціональний компонент			

<p>Знання (повнота, глибина розуміння, усвідомленість, системність, міцність, гнучкість, узагальненість, дієвість)</p>	<p>знання повні, системні та узагальнені; глибина розуміння дозволяє встановлювати причинно-наслідкові зв'язки та закономірності; знання інтегровані в цілісну систему; самостійне та обґрунтоване планування і визначення пріоритетів (проектних і персональних); забезпечення максимізації бізнес-цінності через вибір і реалізацію стратегії проекту; ефективна співпраця з командою у формуванні та реалізації стратегії; гнучке використання традиційних і agile підходів залежно від контексту; здатність чітко пояснювати складні бізнес-аспекти проекту; комплексне керування графіком, вартістю, ресурсами та ризиками; стійка увага до критично важливих технічних елементів проектного управління</p>	<p>знання достатньо повні, охоплюють основні аспекти проектного управління; глибина розуміння дозволяє пояснювати сутність базових процесів; знання частково систематизовані; самостійне планування та визначення пріоритетів у типових ситуаціях; розуміння бізнес-цінності проекту та участь у формуванні підходів до її досягнення; налагоджена співпраця з командою у межах поставлених завдань; використання традиційних і agile інструментів у стандартних умовах; здатність пояснювати основні бізнес-аспекти проекту; самостійне виконання типових завдань з керування графіком, ресурсами та ризиками; увага до ключових технічних елементів проекту</p>	<p>знання базові, охоплюють окремі поняття проектного управління; глибина розуміння обмежена, переважає відтворення без пояснення сутності; знання несистемні; планування та визначення пріоритетів здійснюється за зразком і з допомогою; розуміння бізнес-цінності проекту фрагментарне; участь у співпраці з командою епізодична; використання окремих інструментів і методів (у тому числі agile або традиційних) за інструкцією; пояснення бізнес-аспектів утруднене; виконання окремих дій щодо керування графіком, ресурсами чи ризиками під керівництвом; увага до технічних аспектів нестійка</p>
<p>Уміння та навички (самостійність виконання дій, усвідомленість і обґрунтованість)</p>	<p>самостійно та обґрунтовано планує діяльність і визначає пріоритети з урахуванням цілей</p>	<p>самостійно здійснює планування та визначає пріоритети у типових ситуаціях; узгоджує завдання з</p>	<p>виконує планування та визначення пріоритетів за зразком або з допомогою; виокремлює окремі</p>

<p>дій, результативність виконання, системність організації діяльності, гнучкість і адаптивність, інтегрованість, комунікативна ефективність, інструментальна вправність, контроль і самокорекція діяльності, спрямованість на цінність результату)</p>	<p>проєкту й особистої ефективності; забезпечує максимізацію бізнес-цінності через вибір і реалізацію стратегії; ініціює та організовує співпрацю з командою; гнучко поєднує традиційні та agile підходи відповідно до контексту проєкту; чітко та аргументовано пояснює складні бізнес-аспекти; комплексно керує графіком, вартістю, ресурсами та ризиками; системно враховує критично важливі технічні елементи та впливає на ефективність реалізації проєкту</p>	<p>цілями проєкту; враховує бізнес-цінність при виборі рішень; бере участь у формуванні та реалізації стратегії проєкту; ефективно взаємодіє з командою в межах визначених ролей; доцільно використовує традиційні та agile інструменти у стандартних умовах; пояснює основні бізнес-аспекти проєкту; здійснює керування графіком, ресурсами та ризиками у межах поставлених завдань; враховує ключові технічні елементи проєктного управління</p>	<p>завдання без цілісного бачення проєкту; враховує бізнес-цінність фрагментарно; долучається до реалізації визначеної стратегії без самостійного обґрунтування; взаємодіє з командою та іншими учасниками епізодично; використовує окремі традиційні або agile інструменти за інструкцією; відтворює окремі бізнес-аспекти проєкту без глибокого пояснення; виконує окремі дії з керування графіком, ресурсами чи ризиками під контролем; звертає увагу на окремі технічні елементи без системного врахування</p>
---	---	--	--

Додаток Г

Витяг з експериментального навчального курсу «Теорія та практика проєктного управління в інженерії програмного забезпечення» для другого рівня вищої освіти (авторська розробка)

Таблиця Г.1

Календарний план реалізації навчального курсу

Тиждень	Тематичний фокус	Віха PBL (трек групового проєкту)	Застосування методу кейсів (семінарський напрямок)
1	Вступ до управління проєктами та динаміки команди	Формування команди з використанням методу парної роботи «Слабкі-Сильні»; встановлення статутів групи, ролей та протоколів комунікації.	Кейс: «Невдалий стартап» – аналіз дисфункції команди, відсутності лідерства та катастрофічних наслідків відсутності чіткого визначення обсягу проєкту.
2	Розробка вимог та обсяг	Початкові співбесіди з клієнтом (інструктор виступає в ролі нестабільного клієнта); Розробка базової Специфікації вимог до програмного забезпечення (SRS).	Кейс: «Крутий шлях до школи» – визначення початкових ризиків проєкту, управління агресивним розширенням обсягу робіт та узгодження очікувань зацікавлених сторін.
3	Вибір та методології SDLC	Оцінка предметної області проєкту для вибору відповідної методології (Agile/Scrum проти Waterfall); Визначення поняття «Готово» (DoD).	Кейс: «NASA досягає нових висот» – дослідження проблем масштабування гнучких методологій у критично важливих для безпеки середовищах з високим рівнем дотримання вимог.
4	Розподіл ресурсів та бюджетування	Бюджетування ресурсів для моделювання проєкту; Призначення сторі-	Кейс: «Великі розкопки» – вивчення системних недоліків в управлінні,

Тиждень	Тематичний фокус	Віха PBL (трек групового проєкту)	Застосування методу кейсів (семінарський напрямок)
		пойнтів елементам відкладених завдань; Створення комплексної бізнес-моделі.	масового перевитрату бюджету та наслідків неточної оцінки витрат.
5	Управління ризиками та їх пом'якшення	Визначення технічних та бізнес-ризиків; розробка офіційного плану зниження, моніторингу та управління ризиками (RMMM).	Кейс: «Кризовий менеджмент в українському IT-секторі» – Адаптація до екстремальних геополітичних форс-мажорних обставин та колапсу інфраструктури.
6	Гнучке виконання та планування спринтів	Ініціювання першого спринту розробки; проведення щоденних стендапів; використання канбан-дошок для візуалізації робочого процесу.	Кейс: «John Deere рухається вперед» – застосування гнучких фреймворків у традиційному виробництві та інтеграції апаратного та програмного забезпечення.
7	Огляд та розгортання проєкту в середині проєкту	Оцінка швидкості; Переузгодження вимог з клієнтом на основі початкових технічних перешкод; Оновлення дорожньої карти проєкту.	Кейс: «Міграція Netflix Cloud» – забезпечення безперервної доступності та управління складними переходами інфраструктури без перебоїв у обслуговуванні.
8	Забезпечення якості та відповідність вимогам	Виконання фази тестування; впровадження конвеєра CI/CD; встановлення суворих стандартів експертної перевірки коду.	Кейс: «Проект Tata NSarna» – управління величезним консорціумом університетів, дослідницьких організацій та корпоративних структур для впровадження інновацій.
9	Управління	Рефакторинг кодової бази;	Кейс: «Проблеми

Тиждень	Тематичний фокус	Віха PBL (трек групового проєкту)	Застосування методу кейсів (семінарський напрямок)
	технічним боргом	Визначення скорочень, прийнятих під час ранніх спринтів, та планування їх вирішення для підтримки зручності обслуговування.	команди розробників програмного забезпечення» – вирішення проблем сильного вигорання розробників, плинності кадрів у команді та виявлення дефектів на пізніх стадіях.
10	Розширене згладжування ресурсів	Перерозподіл завдань на основі швидкості роботи окремих розробників; згладжування стрибків ресурсів, щоб уникнути понаднормової роботи та завищення бюджету.	Кейс: «Etsy: Agile в електронній комерції» – Використання безперервного розгортання та швидкої ітерації на висококонкурентних споживчих ринках.
11	Інтеграція та безпека	Вирішення ризиків інтеграції зі сторонніми API; Забезпечення безпеки даних та дотримання правових норм.	Кейс: «Порушення безпеки» – аналіз наслідків поганого управління ризиками в безпеці застосунків для спільного використання автомобілів.
12	Закриття проєкту та документація	Завершення розробки посібників користувача та документації щодо архітектури системи; планування етапів підтримки та технічного обслуговування після розгортання.	Кейс: «Lola Tech та Nexus Framework» – Спостерігаємо за суттєвим зростанням рентабельності інвестицій (ROI) завдяки впровадженню масштабованих практик Scrum.
13	Фінальний спринт	Виправлення критичних помилок; Заморожування розробки нових функцій; Забезпечення відповідності продукту узгодженому визначенню «Готово».	Кейс: «Mavenlink та BTM Global» – Вирішення проблем використання та управління ресурсами шляхом відмови від застарілого відстеження електронних таблиць.

Тиждень	Тематичний фокус	Віха PBL (трек групового проєкту)	Застосування методу кейсів (семінарський напрямок)
14	Підготовка презентації	Структурування фінальної презентації для зацікавлених сторін; синтез технічного шляху у ціннісні пропозиції для бізнесу.	Кейс: «Центр інновацій Merck» – вирішення питань, пов'язаних з агентством, конфліктом цілей та інформаційною асиметрією у взаєминах у проєктах.
15	Заключні презентації та ретроспективи	Постачання кінцевого програмного продукту; офіційна презентація групі «зацікавлених сторін».	Синтез курсу: Сократівський огляд отриманих уроків, оцінка роботи команди та оцінка зростання індивідуальної компетенції.

Таблиця Г.2

Критерії оцінювання результатів PBL-проєкту та командної ефективності

Критерій оцінювання	Незадовільно	Задовільно	Здав з відзнакою
Управління вимогами та обсягом робіт	Не вдається зібрати точні вимоги. Високий ступінь некерованого розширення обсягу робіт. Відсутня або незрозуміла документація SRS.	Документує чіткі функціональні та нефункціональні вимоги. Керує змінами обсягу робіт за допомогою визначеного, логічного процесу.	Передбачає невисловлені потреби клієнтів. Створює високоадаптивний, пріоритетний портфель замовлень, який ідеально відповідає максимізації бізнес-цінності.
Архітектура програмного забезпечення та якість коду	Архітектура нелогічна або надзвичайно крихка. Коду бракує базових стандартів, він погано коментований і не проходить основні модульні тести.	Архітектура належним чином підтримує основні вимоги. Код загалом відповідає галузевим стандартам, є читабельним та проходить тести основної функціональності.	Високомасштабований, модульний та безпечний дизайн. Код елегантний, всебічно задокументований та ефективно використовує передові шаблони проектування.

Критерій оцінювання	Незадовільно	Задовільно	Здав з відзнакою
Планування проектів та зменшення ризиків	Не вдається встановити реалістичний графік. Ігнорує основні технічні або бізнес-ризиками, що призводить до серйозних проблем у проекті або повного провалу.	Встановлює логічні етапи та чіткий графік розвитку. Визначає основні ризики та пропонує стандартні, функціональні стратегії їх пом'якшення.	Створює дуже динамічний план з аналізом варіантів. Проактивно керує складними ризиками, легко адаптує до суттєвих збоїв, не втрачаючи імпульсу.
Командна співпраця та спільне лідерство	Негативно домінує в групі або не робить повного внеску. Генерує невирішені токсичні конфлікти. Пропускає внутрішні дедлайни.	Справедливо виконує свій внесок у робоче навантаження. Ефективно спілкується та вирішує стандартні конфлікти, щоб команда не зосереджувалася на виконанні завдань.	Підвищує продуктивність усієї команди. Сприяє створенню високоінклюзивної атмосфери, наставляє однолітків, які мають труднощі, та бездоганно організовує зусилля.
Аналіз тематичних досліджень та комунікація	Надає поверхневі резюме справ. Не може захищати позиції або застосовувати теоретичні основи до наративу.	Застосовує відповідні рамки для діагностики проблем. Наводить логічні, добре обґрунтовані аргументи під час обговорень у класі.	Синтезує складні дані для виявлення прихованих першопричин. Спілкується переконливою, специфічною для своєї дисципліни мовою керівництва та глибоким розумінням.

1. Незадовільно (< 50% еквівалент): Студент або команда не досягли мінімальних ключових компетенцій. Артефакт програмного забезпечення принципово зламаний, архітектура повністю не підходить для предметної області або відсутня необхідна документація проекту. Аналіз випадків демонструє брак

критичного мислення або сильну залежність від поверхневих припущень.

2. Задовільно (еквівалент 50% - 84%): Студент продемонстрував ґрунтовне розуміння матеріалу. Програмне забезпечення відповідає основним функціональним вимогам, проєкт керувався з використанням відповідних та ідентифікованих методологій, а обговорення кейсів проводилося з логічними аргументами, заснованими на доказах. Цей рівень відповідає базовим очікуванням для висококомпетентного молодшого інженера-програміста, який вступає на роботу.

3. Здав з відзнакою (еквівалент > 85%): Зарезервовано за справді видатні досягнення. Студент продемонстрував глибокий синтез, високоінноваційне вирішення проблем та виняткове лідерство в командній динаміці. Архітектура програмного забезпечення є високомасштабованою, елегантною та тісно узгодженою з бізнес-цінностями. Аналіз кейсів дав глибокі висновки другого та третього порядку, які перевершили безпосередній текст оповіді.

Приклади тестових завдань з оцінювання функціональної готовності до управління проєктами

Сценарій оцінки 1: Договірна стратегія та відповідальність за технічне обслуговування

Деталі сценарію: Клієнт звертається до вашої фірми з проханням надати «виділену команду» з 5 інженерів (3 розробників, 2 фахівці з контролю якості) на термін 1,5 роки. В умовах контракту клієнт наполягає на гарантійному пункті, який передбачає, що команда повинна безкоштовно виправляти будь-які виявлені помилки протягом двох місяців після завершення проєкту. Як керівник проєкту, які стратегічні дії є доречними?

Варіанти:

А) Прийняти умови, щоб забезпечити собі дуже вигідний довгостроковий контракт.

В) Запропонувати додати до команди спеціального менеджера проєкту для жорсткого контролю якості та забезпечення відсутності помилок.

С) *Правильна дія:* Запропонуйте альтернативну структуру контракту (наприклад, «Fixed price» або «Time and Material») або чітко розділіть зобов'язання на два окремі контракти: один для оренди спеціальної команди та окрема Угода про рівень обслуговування (SLA) для післязапускового обслуговування.

Діагностичне обґрунтування: Цей сценарій перевіряє розуміння студентом договірного права та відповідальності в галузі програмної інженерії. Контракти на «виділену команду» передбачають оренду людських ресурсів, а не конкретних результатів. Прийняття фіксованої ціни на обслуговування за контрактом на оренду ресурсів наражає інженерну фірму на величезну, необмежену фінансову відповідальність. Студент, який відповідає неправильно, демонструє небезпечну відсутність навичок оцінки ризиків.

Сценарій оцінювання 2: Вирішення конфлікту між виконавчими зацікавленими сторонами

Деталі сценарію: Ваш провідний бізнес-аналітик повідомляє про критичний глухий кут: два високопоставлених зацікавлених сторони подали взаємовиключні, суперечливі вимоги до основного модуля системи. Усі спроби бізнес-аналітика узгодити ці вимоги на операційному рівні зазнали невдачі. Як керівник проекту, які ваші наступні дії?

Варіанти:

А) Передати рішення провідному архітектору програмного забезпечення для визначення найбільш технічно елегантного та доцільного варіанту.

В) Доручити команді розробників створити обидві функції та дозволити базі користувачів прийняти рішення після запуску за допомогою А/В-тестування в реальному часі.

С) *Правильна дія:* Організуйте спільну зустріч з обома зацікавленими сторонами та запросіть Спонсора Проекту виступити в ролі остаточного арбітра та приймати рішення про прийняття рішення.

Діагностичне обґрунтування: Це питання оцінює розуміння студентом ієрархії зацікавлених сторін та вирішення конфліктів. Конфліктні бізнес-вимоги між рівними в організації не можуть бути вирішені аналітиком латерально, а також

не повинні вирішуватися технічно архітектором. Вони вимагають вертикальної ескалації до спонсора проекту, який має кінцеві фінансові та стратегічні повноваження щодо бізнес-цінності продукту.

Сценарій оцінювання 3: Оцінювання за умов надзвичайної невизначеності

*Деталі сценарію:*Клієнту потрібна негайна, загальна оцінка витрат і часу для переведення масивного, багаторічного застарілого монолітного застосунку на сучасну мікросервісну архітектуру. Команда використовує високоітеративний Agile-підхід. Які методи оцінки найбільш підходять для цієї фази надзвичайної невизначеності?

Варіанти:

А) Метод критичного шляху (CPM) та моделювання СОСОМО.

В)Правильна дія:Визначення розмірів футболок та оцінка «зверху вниз».

С) Планування покером та оцінка «знизу вгору»

*Діагностичне обґрунтування:*У середовищах Agile з високою невизначеністю на ранніх стадіях, детальні детерміновані моделі, такі як CPM або висхідна оцінка, вимагають детальних даних, яких просто ще не існує, що призводить до хибної точності. Визначення розмірів футболок та низхідна оцінка забезпечують швидкі показники відносної складності, які ідеально підходять для початкового визначення обсягу роботи та встановлення прийнятних бюджетних діапазонів.

Сценарій оцінювання 4: Вибір критично важливих для безпеки систем та структури

*Деталі сценарію:*Вам доручено спланувати початок масштабного нового проекту: розробку вбудованої прошивки для медичного обладнання, яка автоматизує процес проведення хіміотерапії. Проєкт триває 4 роки та вимагає команди зі 100 спеціалізованих інженерів та тестувальників. Яка методологія розробки є найбільш доцільною?²

Варіанти:

А)Правильна дія:Водоспад або масштабований Agile Framework (SAFe).

В) Чистий Agile Scrum.

С) Канбан.

Діагностичне обґрунтування: Цей сценарій перевіряє здатність узгоджувати методології з обмеженнями предметної області. Медичне програмне забезпечення є суворо регульованим, критично важливим для безпеки та вимагає вичерпної попередньої документації та відстежуваності для аудитів відповідності. Pure Scrum, який надає перевагу робочому програмному забезпеченню над комплексною документацією та охоплює зміни на пізніх стадіях, створює неприйнятний ризик у цьому контексті. Waterfall забезпечує необхідні суворі фазові обмеження, тоді як SAFe забезпечує структурований спосіб масштабування Agile-практик, зберігаючи при цьому відповідність на рівні підприємства.

...

Додаток Д

Довідки про впровадження результатів дослідження



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

вул. Львівська, 11, м. Тернопіль, 46009; тел./факс +380 (352) 51-75-75;
www.wunu.edu.ua; rektor@wunu.edu.ua; ідентифікаційний код за ЄДРПОУ 33680120

126-27/1248

20 травня 2026

ДОВІДКА

про впровадження результатів дисертаційного дослідження
«Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до проектного управління» на здобуття наукового ступеня доктора філософії
за спеціальністю 011 Освітні, педагогічні науки

Результати дисертаційного дослідження ГОРБАТЮКА Іллі Анатолійовича впроваджено в освітній процес кафедри комп'ютерних наук факультету комп'ютерних інформаційних технологій Західноукраїнського національного університету в межах підготовки здобувачів вищої освіти за спеціальністю Інженерія програмного забезпечення для першого (бакалаврського) рівня вищої освіти впродовж 2022-2026 рр.

Реалізовано авторську методику підготовки бакалаврів з інженерії програмного забезпечення до проектного управління, що передбачає системне впровадження проблемно-орієнтованого навчання, спрямованого на моделювання та розв'язання реальних професійних ситуацій з проектного управління у сфері інженерії програмного забезпечення; оновлення змісту освітніх компонент з урахуванням специфіки проектного управління в галузі інженерії програмного забезпечення та сучасних вимог IT-індустрії; забезпечення педагогічного супроводу та рефлексії, що передбачає консультування, зворотний зв'язок і аналіз результатів проблемно-проектної діяльності з метою корекції навчального процесу та усвідомлення здобутого досвіду. Розроблені навчально-методичні матеріали (діагностичний інструментарій, завдання для практичних занять, критерії оцінювання рівнів готовності) використано під час проведення практичних і лабораторних занять. Методику реалізовано в межах дисциплін циклу професійної і практичної підготовки.

Наукові здобутки мають вагоме теоретичне і практичне значення для модернізації підготовки майбутніх бакалаврів з інженерії програмного забезпечення.

Проректор з науково-педагогічної роботи



Віктор ОСТРОВЕРХОВ

Горбатюк І.М. 0971046422



УКРАЇНА
МІНІСТЕРСТВО ОСВІТИ І НАУКИ
УКРАЇНИ

ЧЕРКАСЬКИЙ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ імені БОГДАНА
ХМЕЛЬНИЦЬКОГО

бульвар Шевченка, 81, м. Черкаси, 18031
тел./факс: (0472) 354463, 372142
e-mail: cic@cdu.edu.ua

UKRAINE
MINISTRY OF EDUCATION AND
SCIENCE OF UKRAINE

BOHDAN KHMELYNYSKY NATIONAL
UNIVERSITY of CHERKASY

81, Shevchenko blvd., Cherkasy, Ukraine, 18031
Phone/Fax: +380(472)354463, 372142
e-mail: cic@cdu.edu.ua

25.05.2026 № 137/04

на № _____

ДОВІДКА
про впровадження результатів дисертаційної праці
ГОРБАТЮКА Іллі Анатолійовича
«Підготовка майбутніх бакалаврів з інженерії програмного
забезпечення до проєктного управління»,
поданої на здобуття наукового ступеня доктора філософії
зі спеціальності 011 Освітні, педагогічні науки

Результати дисертаційного дослідження були апробовані протягом 2022-2026 рр. на базі факультету обчислювальної техніки, інтелектуальних і управляючих систем Черкаського національного університету імені Богдана Хмельницького у процесі підготовки здобувачів другого рівня вищої освіти за спеціальністю 121 «Інженерія програмного забезпечення».

У процесі педагогічного експерименту впроваджено структурно-функціональну модель підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління, спрямовану на формування їхньої готовності до здійснення проєктної діяльності у професійній сфері. Реалізація моделі передбачала поетапне оновлення змісту освітніх компонентів, удосконалення форм організації освітнього процесу, а також застосування сучасних методів і засобів навчання. Ефективність впровадження забезпечувалася створенням відповідних організаційно-педагогічних умов, зокрема: системним використанням проблемно-орієнтованого навчання для моделювання та розв'язання реальних професійних ситуацій з проєктного управління у сфері інженерії програмного забезпечення; оновленням змісту освітніх компонентів відповідно до специфіки проєктного управління та сучасних вимог IT-індустрії; забезпеченням педагогічного супроводу, консультативної підтримки, рефлексії та аналізу результатів проблемно-проєктної діяльності з метою корекції освітнього процесу й усвідомлення здобутого професійного досвіду.

Апробація результатів дисертаційного дослідження підтвердила його актуальність, наукову новизну, а також вагоме теоретичне і практичне значення для професійної підготовки майбутніх бакалаврів з інженерії програмного забезпечення до проєктного управління. Результати впровадження авторських розробок обговорено та схвалено на засіданні кафедри програмного забезпечення автоматизованих систем (протокол №11 від 14.05.2026).

Проректор з наукової, інноваційної
та міжнародної діяльності



Олександр СПРЯГАЙЛО



УКРАЇНА

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ**

вул. Львівська, 75, м. Луцьк, 43018, тел.: +38(0332)74-61-03

e-mail: rector@lntu.edu.ua, web: www.lntu.edu.ua

код ЄДРПОУ 05477296

26.05.2026 № 1522/01-14

на № _____ від _____

ДОВІДКА

**про впровадження результатів дисертаційного дослідження
«Підготовка майбутніх бакалаврів з інженерії програмного забезпечення до
проектного управління» на здобуття наукового ступеня доктора філософії
за спеціальністю 011 Освітні, педагогічні науки**

Результати дисертаційного дослідження ГОРБАТЮКА Іллі Анатолійовича впроваджено в освітній процес кафедри інженерії програмного забезпечення факультету комп'ютерних та інформаційних технологій Луцького національного технологічного університету в межах підготовки здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 121 Інженерія програмного забезпечення впродовж 2022-2026 рр.

У рамках дослідження впроваджено авторську методику підготовки бакалаврів з інженерії програмного забезпечення до проектного управління, орієнтовану на формування готовності до професійної діяльності в умовах сучасної ІТ-галузі. Методика ґрунтується на системному застосуванні проблемно-орієнтованого навчання, що передбачає моделювання та розв'язання професійно спрямованих ситуацій у сфері проектного управління програмними продуктами. Важливим складником методики стало оновлення змісту освітніх компонентів відповідно до специфіки проектного управління в інженерії програмного забезпечення та актуальних вимог ІТ-індустрії. Реалізацію методики забезпечено через педагогічний супровід освітнього процесу, який охоплював консультування здобувачів освіти, надання систематичного зворотного зв'язку, організацію рефлексії та аналіз результатів проблемно-проектної діяльності з метою корекції навчального процесу й усвідомлення набутого професійного досвіду. Для практичної реалізації методики розроблено навчально-методичне забезпечення, зокрема діагностичний інструментарій, завдання для практичних і лабораторних занять, а також критерії оцінювання рівнів сформованості готовності до проектного управління. Запропоновану методику впроваджено в межах дисциплін циклу професійної та практичної підготовки.

Отримані наукові результати мають вагоме теоретичне та практичне значення для вдосконалення системи підготовки майбутніх бакалаврів з інженерії програмного забезпечення.

Завідувач кафедри

інженерії програмного забезпечення

Наталія ЛІЩИНА

Проректор з науково-педагогічної роботи
та досліджень

Олена ЛЮТАК

