

Міністерство освіти і науки України
Тернопільський національний педагогічний університет
імені Володимира Гнатюка

Фізико-математичний факультет
Кафедра інформатики та методики її навчання

Кваліфікаційна робота
МЕТОДИКА ВИКОРИСТАННЯ НАВЧАЛЬНО-ІГРОВИХ ПЛАТФОРМ У
ПРОЦЕСІ ВИВЧЕННЯ ПРОГРАМУВАННЯ УЧНЯМИ СТАРШОЇ ШКОЛИ

Спеціальність 014 Середня освіта (Інформатика)
Освітня програма «Середня освіта (Інформатика, математика, STEM освіта)»

Здобувача вищої освіти
освітньо- кваліфікаційного рівня «магістр»
Семків Руслан Олегович

НАУКОВИЙ КЕРІВНИК:
кандидат технічних наук, доцент кафедри
інформатики та методики її навчання
Вовкодав Олександр Валерійович

РЕЦЕНЗЕНТ:
методист (завідувач) Центру ІКТ
ТОКІППО Кривокульський
Любомир Євстахович

Тернопіль – 2025

Анотація

Семків Р.О. Методика використання навчально-ігрових платформ у процесі вивчення програмування учнями старшої школи ; ТНПУ ім. В. Гнатюка, фізико-математичний факультет, кафедра інформатики та методики її навчання; наук. керівник: кандидат тех. наук, доцент кафедри інформатики та методики її навчання Вовкодав О. В. – Тернопіль 2025 – 61с.

Магістерське дослідження присвячене створенню навчальної платформи і сучасних, новітніх методик у навчанні дітей старших шкіл, розвиток алгоритмічного і аналітичного мислення з реалізацією творчого підходу до її реалізації. Робота викладена у вступі, трьох розділах та висновках, а також додатків і використаних джерел.

У теоретичній частині проаналізовано предметну область дослідження; здійснено порівняльний аналіз програмних середовищ та виділено стек технологій, що йдуть в супроводі зі Spring + Java, як оптимальний варіант для навчання студентів; розглянуті ключові аспекти розробки ігрових платформ та сучасний стан гейміфікації освіти.

Практична частина – розробка онлайн платформи, яка дала спосіб описати методику вивчення мов програмування, математичних складових і фізики, та їх практичного застосування.

Ключові слова: освітній процес; організація навчання; дистанційна освіта; змішане навчання; онлайн-ресурси; освітні платформи; платформне навчання.

ABSTRACT

Semkiv R.O. Methods of using educational gaming platforms in the process of teaching programming to high school students; V. Hnatiuk Ternopil National Pedagogical University, Faculty of Physics and Mathematics, Department of Computer Science and Teaching Methods; scientific supervisor: Candidate of Technical Sciences, lecturer at the Department of Computer Science and Teaching Methods, O.V. Vovkodav – Ternopil 2025 – 61 p.

The master's research is devoted to the creation of an educational platform and modern, innovative methods for teaching high school students, the development of algorithmic and analytical thinking with the implementation of a creative approach to its implementation. The work is presented in an introduction, three chapters, and conclusions, as well as appendices and sources used.

The theoretical part analyzes the subject area of the research; a comparative analysis of software environments is carried out and a stack of technologies accompanying Spring + Java is identified as the optimal option for teaching students; key aspects of game development on these platforms and the current state of gamification in education are considered.

The practical part is the development of an online platform that provides a way to describe the methodology for studying programming languages, mathematical components, and physics, as well as their practical application.

Keywords: educational process; organization of learning; distance education; blended learning; online resources; educational platforms; platform-based learning.

ЗМІСТ

Анотація.....	2
Вступ	5
Розділ 1. Теоретичні засади побудови освітніх платформ дистанційного навчання.....	8
1.1 Актуальність теми в умовах цифровізації освіти та виклики сьогодення	8
1.2 Історичний розвиток дистанційної освіти.....	11
1.3 Теоретико-методологічні основи дистанційного навчання	14
1.4 Аналіз сучасних досліджень та практик	18
1.5 Порівняльний аналіз наявних освітніх платформ.....	20
Висновок до Розділу 1	23
Розділ 2. Принципи та технічні підходи до створення освітніх платформ для дистанційного навчання	24
2.1 Основні дидактичні принципи побудови освітніх платформ	24
2.2 Архітектура та структура сучасної освітньої онлайн платформи.....	27
2.3 Застосування хмарних технологій у реалізації освітніх платформ ...	29
2.4 STEM-орієнтовані компоненти освітніх платформ	31
2.5 Проблеми та виклики реалізації освітніх платформ.....	34
Розділ 3. Практичні аспекти реалізації освітніх платформ для дистанційного навчання у середній школі.....	37
3.1 Педагогічний підхід у розробці методики навчання дітей, з урахуванням ігрових і пізнавальних цілей, для досягнення продуктивнішого навчання	38
3.2 Вибір архітектури програмного забезпечення	46
3.3 Вибрані технології.....	48
3.4 Основні функціональні можливості платформи	50
3.5 Етапи розробки	55
3.6 Висновок до розділу 3	61
ВИСНОВОК	63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	65
ДОДАТКИ	67

Вступ

Стрімкий розвиток цифрових технологій та глобалізація інформаційного простору суттєво змінюють підходи до організації освітнього процесу. Особливої актуальності в цьому контексті набуває дистанційне навчання як форма, що дозволяє забезпечити безперервність та доступність освіти у будь-яких умовах — зокрема в умовах війни, пандемії або соціальних обмежень. На перетині інформаційних технологій, педагогіки та методики викладання формуються нові моделі освітнього середовища, де освітні платформи відіграють ключову роль.

Сучасні дослідження в галузі інформаційних технологій (Биков В.Ю., Шишкіна М.П., Anderson T., Siemens G.) акцентують увагу на переході від традиційної до гнучкої, персоналізованої освіти, яка реалізується завдяки освітнім платформам нового покоління. Проблематика формування ефективного освітнього середовища, орієнтованого на STEM-освіту, потребує критичного переосмислення наукової літератури та введення інноваційних рішень.

Вибір теми обумовлений необхідністю адаптації української системи освіти до сучасних цифрових викликів, а також зростаючою потребою у розробці платформ, здатних забезпечити якісне дистанційне навчання з урахуванням спеціалізації «Інформатика. Математика». Проведений аналітичний огляд джерел свідчить про наявність значної кількості теоретичних напрацювань у сфері електронного навчання, однак спостерігається недостатній рівень реалізації цих підходів у реальному освітньому процесі, зокрема у закладах середньої освіти.

Метою дослідження є теоретичне обґрунтування та аналіз принципів побудови освітніх платформ дистанційного навчання, орієнтованих на потреби учнів середньої школи у контексті STEM-освіти.

Для досягнення поставленої мети необхідно розв'язати такі завдання:

1. Проаналізувати сучасний стан розвитку дистанційної освіти в Україні та світі.
2. Розглянути наукові підходи до побудови освітніх платформ.
3. Виокремити ключові принципи, що лежать в основі ефективних освітніх платформ.
4. Дослідити специфіку застосування платформ у викладанні інформатики та математики.
5. Запропонувати практичні рекомендації щодо побудови освітніх платформ у середній школі.

Мета дослідження – здійснити теоретичний аналіз перспектив розвитку дистанційної освіти в Україні, окреслити найактуальніші проблеми впровадження цього формату навчання, а також дослідити фактори, що стримують його повноцінну реалізацію. Особлива увага приділяється необхідності підвищення цифрової грамотності вчителів та створення сучасної методологічної бази для організації дистанційного навчання. Серед іншого, актуальним залишається формування єдиної державної системи координації та контролю у сфері онлайн-освіти.

Методи дослідження: аналітичний, структурно-функціональний аналіз, феноменологічний підхід, контент-аналіз, філософська рефлексія, а також соціологічні методи — зокрема анкетування й інтерв'ю.

Наукова новизна полягає: у з'ясуванні поточного стану впровадження дистанційної освіти в українських школах та в аналізі особливостей цього процесу. У межах дослідження проаналізовано досвід провідних зарубіжних науковців у галузі дистанційної освіти, а також частково обґрунтовано можливість адаптації їхніх підходів до вітчизняних реалій. На основі проведеного аналізу запропоновано низку інноваційних гіпотез щодо факторів — особистісних, колективних та суспільних — які заважають ефективному впровадженню дистанційного навчання, призводять до його формалізації та

зниження якості, й можуть стати підґрунтям для подальших наукових досліджень у цій сфері.

Розділ 1. Теоретичні засади побудови освітніх платформ дистанційного навчання

1.1 Актуальність теми в умовах цифровізації освіти та виклики сьогодення

У сучасному глобалізованому світі цифровізація стала визначальним чинником розвитку провідних галузей суспільства, і система освіти не є винятком. Технологічні зміни, що відбуваються упродовж останніх років, зумовили необхідність переосмислення традиційних освітніх моделей та пошуку нових форматів організації навчального процесу. Особливо гостро ці питання постали в умовах пандемії COVID-19, що фактично змусила освітні заклади у всьому світі перейти на дистанційний або змішаний формат, а згодом — в умовах повномасштабної війни в Україні. Саме ці події стали каталізатором цифрових трансформацій у сфері освіти, виявивши структурні проблеми, які накопичувалися роками, та водночас продемонструвавши потенціал дистанційних технологій як гнучкого, доступного і сучасного інструменту навчання.

Дистанційне навчання перестало бути допоміжною формою освітньої діяльності та стало фундаментальним елементом освітньої системи, що дозволяє забезпечити безперервність навчання незалежно від зовнішніх обставин. Це особливо важливо в умовах воєнного стану, коли значна частина школярів навчається з інших регіонів або країн, а також у ситуаціях, коли традиційний формат навчання є небезпечним або неможливим. Така гнучкість освітнього процесу дає змогу зберігати цілісність національної системи освіти та підтримувати рівень якості навчання.

Сучасні виклики висвітлили ще одну проблему: традиційна модель навчання, яка спирається на фронтальні методи викладання і єдині стандарти подачі матеріалу, вже не відповідає потребам покоління, що зростає у світі цифрових технологій. Діти, які належать до покоління Z і покоління альфа,

мають інші когнітивні особливості, стиль мислення та спосіб сприйняття інформації. Вони звикли до швидкого доступу до даних, інтерактивності, візуальності, гейміфікації та миттєвого зворотного зв'язку. Внаслідок цього виникає розрив між динамічними потребами учнів і застарілими методами навчання, які нерідко залишаються незмінними попри технологічний прогрес.

Однією з ключових проблем сучасної школи є недостатня індивідуалізація навчального процесу. Традиційна освітня модель, орієнтована на "середнього учня", ігнорує різноманіття типів мислення, інтересів, швидкості засвоєння матеріалу та рівнів когнітивної активності. У контексті розвитку нейropsихології стає очевидним, що індивідуальні освітні траєкторії є значно ефективнішими, ніж уніфіковані підходи. Вони дозволяють врахувати домінантні канали сприйняття інформації, особливості уваги, пам'яті, мислення та мотивації учнів. Проте без належної цифрової інфраструктури та педагогів, які володіють сучасними цифровими інструментами, реалізувати таку персоналізацію у масштабах загальноосвітньої школи надзвичайно складно.

Успішність освітнього процесу безпосередньо залежить від рівня розвитку когнітивних функцій учнів. Наукові дослідження свідчать, що навчання є ефективним тоді, коли воно не лише передає інформацію, а й стимулює розвиток аналітичного, критичного, логічного, системного та дивергентного мислення. Проте в більшості традиційних методик домінують репродуктивні форми діяльності: запам'ятовування, відтворення та механічне виконання завдань. Це стримує розвиток учнів, особливо тих, які орієнтовані на творчість, інженерне мислення чи дослідницьку діяльність. Цифрові освітні платформи, навпаки, створюють умови для активного залучення школярів, взаємодії з матеріалом, самостійного дослідження та проєктної роботи.

В умовах цифровізації змінюється і роль учителя. Від нього очікується не лише передавання знань, а й уміння керувати цифровим освітнім середовищем, інтегрувати інноваційні інструменти, створювати мультимедійні ресурси,

аналізувати цифрові сліди навчальної діяльності учнів та будувати індивідуальні освітні маршрути. Здатність працювати з цифровими технологіями, навчальними платформами, сервісами для відеоконференцій, хмарними сховищами та інтерактивними інструментами стає не додатковою, а базовою професійною компетентністю сучасного педагога. Постійний професійний розвиток учителя — необхідна умова якісної освіти в умовах цифрової трансформації.

Проблематика інформаційної безпеки також набуває дедалі більшої актуальності. Сучасні школярі постійно взаємодіють з великими обсягами цифрового контенту, серед якого є неправдива інформація, маніпулятивні матеріали та ресурси, що несуть загрозу приватності. Тому школа повинна формувати цифрову та медіаграмотність, уміння критично мислити, розпізнавати фейки, аналізувати джерела інформації, дбати про кібергігієну та захищати персональні дані. Ці компетентності входять до переліку ключових навичок XXI століття.

Разом із розвитком дистанційних технологій стає очевидною проблема цифрової нерівності — нерівного доступу до техніки, інтернету та комфортного середовища для навчання. Частина учнів не має якісного інтернет-з'єднання або сучасних гаджетів, що створює ризик освітнього розриву та загрожує порушенням принципу рівності в освіті. Подолання цього виклику потребує системних рішень на державному рівні, модернізації інфраструктури, забезпечення шкіл технічним обладнанням та підтримки соціально вразливих категорій населення.

Значну увагу у наукових дослідженнях привертає питання організації дистанційного навчання у STEM-напрямі. Предмети природничо-математичного циклу, інформатика, інженерія та технології вимагають використання лабораторних робіт, експериментів, моделювання та взаємодії з реальними або цифровими об'єктами. Тому сучасні освітні платформи повинні забезпечувати можливість роботи з віртуальними лабораторіями, 3D-моделями, симуляторами,

інтерактивними тренажерами та програмними середовищами. Це сприяє формуванню дослідницьких, аналітичних та проєктних компетентностей, робить навчання більш змістовним і практикоорієнтованим. Особливо це актуально для предметів «Інформатика» та «Математика», які відіграють центральну роль у формуванні STEM-компетентностей.

Цифрові платформи, орієнтовані на потреби STEM-освіти, дають можливість здійснювати диференціацію та індивідуалізацію навчання, інтегрувати гейміфікаційні елементи, відстежувати динаміку розвитку учнів та формувати об'єктивне формувальне оцінювання. Це дозволяє створити нову модель навчального процесу, де в центрі уваги перебуває учень, його потреби, здібності та інтереси, а не стандартизований підхід.

Таким чином, актуальність теми дослідження обумовлена необхідністю глибокого осмислення процесів цифровізації, їхнього впливу на освітній простір, особливостей організації дистанційного навчання та ролі цифрових платформ у формуванні компетентностей учнів, зокрема в STEM-напрямі. Ці трансформації є не просто тимчасовою реакцією на кризові умови, а стратегічним шляхом розвитку сучасної української освіти, що має забезпечити її конкурентоспроможність і відповідність вимогам XXI століття.

1.2 Історичний розвиток дистанційної освіти

Дистанційне навчання має тривалу та багатогранну історію, яка бере свій початок ще з XIX століття і демонструє поступову еволюцію від навчання за листуванням до сучасних інтерактивних цифрових платформ. Перші спроби організувати процес здобуття освіти на відстані пов'язані з розвитком поштового зв'язку та появою спеціалізованих програм, що дозволяли учням отримувати навчальні матеріали та надсилати виконані завдання викладачам. Цей підхід був затребуваним передусім серед людей, які не мали можливості відвідувати навчальні заклади через віддаленість, стан здоров'я або соціальні бар'єри.

З розвитком телекомунікацій у XX столітті з'явилися нові етапи становлення дистанційної освіти. У 1920–1930-х роках навчання через радіо стало інноваційним способом передавання знань, а з появою телебачення було започатковано перші телевізійні освітні програми. У багатьох країнах світу вони відіграли значну роль у поширенні знань серед широких верств населення та стали підґрунтям для розвитку масової відкритої освіти. У другій половині XX століття поява персональних комп'ютерів й розвиток інформаційно-комунікаційних технологій дали змогу використовувати електронні носії, програмне забезпечення для навчання, CD-диски та мультимедійні ресурси.

Особливий прорив у становленні дистанційного навчання відбувся в 1990-х роках, коли активний розвиток мережі Інтернет створив принципово нові можливості для організації освітнього процесу. На той час з'явилися перші онлайн-курси, які дозволяли студентам взаємодіяти зі змістом навчання в електронному форматі. Це стало фундаментом для появи перших систем управління навчанням (Learning Management Systems — LMS), серед яких найбільш відомими стали Blackboard, Moodle, WebCT. Ці платформи забезпечували структуровану організацію курсів, управління завданнями, проведення онлайн-тестування, обмін файлами, формувальне оцінювання та інтерактивне спілкування між учасниками освітнього процесу. LMS заклали основу для сучасних підходів до електронного навчання, де ключовими елементами стали зручність, доступність та можливість масштабування освітніх програм.

На межі XX–XXI століть дистанційне навчання набуло глобального характеру, а у 2012 році відбувся так званий «рік MOOCs» — масових відкритих онлайн-курсів. Такі платформи, як Coursera, edX, Udacity, FutureLearn, започаткували революцію у сфері безкоштовного доступу до освіти, надавши можливість мільйонам людей із різних країн проходити курси провідних університетів світу. MOOC-платформи не лише розширили доступ до знань, а й

започаткували нові форми сертифікації, інтерактивну взаємодію, автоматизоване оцінювання, а також системи рекомендацій та адаптивне навчання.

В Україні дистанційне навчання до 2020 року розвивалося здебільшого в межах вищої освіти. Багато університетів використовували Moodle або власні LMS для організації онлайн-курсів та підтримки змішаних форм навчання. Проте загальноосвітня школа до пандемії COVID-19 мала мінімальний досвід системної організації дистанційного навчання. Вимушений перехід на віддалений формат у 2020 році виявив неподготовленість значної частини освітніх закладів, брак цифрової інфраструктури та обмеженість цифрових компетентностей педагогів.

У відповідь на виклики пандемії було створено низку державних та приватних платформ, серед яких: «Всеукраїнська школа онлайн», «Моя школа», «Нові знання», «Дія.Освіта». Проєкт «Всеукраїнська школа онлайн» став одним із ключових рішень, оскільки забезпечив доступ до систематизованих відеоуроків, тестових завдань та матеріалів для учнів різних класів. Платформи «Моя школа» та «Нові знання» запропонували системи оцінювання, журналів, календарного планування та комунікації між учнями, вчителями та батьками.

Попри значний прогрес, багато з українських освітніх платформ залишаються фрагментарними: вони забезпечують лише частину функцій, не завжди мають адаптивні алгоритми, інтерактивні лабораторії чи можливості для моделювання, що є критично важливим для STEM-предметів. Викладання інформатики, математичних дисциплін, фізики, хімії та технологій вимагає не лише подачі теоретичного матеріалу, а й взаємодії з віртуальними симуляторами, обчислювальними середовищами, інструментами програмування та лабораторними моделями. Не всі наявні рішення здатні забезпечити ці потреби на належному рівні.

В умовах повномасштабної війни роль дистанційного навчання значно зросла. Мільйони дітей вимушено змінили місце проживання, багато шкіл були зруйновані або працюють в умовах постійної небезпеки. У цих умовах дистанційні технології стали єдиним інструментом, який забезпечує безперервність навчального процесу. Проте це також ускладнило проблему цифрової нерівності, адже доступ до техніки, інтернету, безпечних умов та педагогічної підтримки є неоднаковим для різних категорій учнів.

Загалом історичний розвиток дистанційного навчання демонструє перехід від простого передавання навчальних матеріалів до створення складних, інтерактивних та адаптивних цифрових середовищ. Сучасні платформи поєднують мультимедійний контент, засоби комунікації, інструменти оцінювання, аналітику навчальних даних, елементи гейміфікації та моделювання. Цей розвиток створює фундамент для подальшої трансформації освіти та формування нових підходів до навчання в умовах цифрової реальності.

1.3 Теоретико-методологічні основи дистанційного навчання

Розробка ефективної освітньої платформи дистанційного навчання потребує комплексного теоретико-методологічного підґрунтя, яке враховує особливості цифрової комунікації, специфіку навчальної взаємодії «вчитель — учень» та психолого-педагогічні умови організації сучасного освітнього процесу. У науковій літературі виокремлюють низку теорій та концепцій, що формують фундаментальні принципи побудови онлайн-освіти. Найбільш значущими серед них є теорія транзакційної відстані, коннективізм, соціальний конструктивізм та STEM-педагогіка. Їх використання дозволяє проектувати навчальні платформи, що сприяють розвитку автономності учнів, формуванню мережових компетентностей, забезпечують колаборативне середовище й створюють умови для практико-орієнтованого навчання.

- Теорія транзакційної відстані (М. Г. Мур)

Теорія транзакційної відстані є однією з найбільш фундаментальних концепцій дистанційної освіти. Її ключове положення полягає в тому, що ефективність навчання визначається не фізичною, а психологічною та педагогічною дистанцією, що виникає між учнем та викладачем у процесі опосередкованої взаємодії. Мур виокремлює три головні компоненти навчального процесу:

- Структура — ступінь організованості та передбачуваності навчальних матеріалів. Чим більш жорстка структура, тим менше гнучкості у навчанні та адаптації до індивідуальних потреб.
- Діалог — якість і частота комунікацій між учасниками процесу. Інтерактивні елементи платформи (чат, відеозв'язок, форуми, зворотний зв'язок у реальному часі) зменшують транзакційну відстань.
- Автономія учня — здатність самостійно організовувати своє навчання, приймати рішення, контролювати темп і способи освоєння матеріалу.

У цифровому середовищі важливо підтримувати баланс між цими компонентами: надмірна структурованість без діалогу спричиняє пасивність учня, а недостатня автономія може знижувати мотивацію. Для предметів інформатики та математики це є особливо актуальним, адже учні мають не лише отримувати інформацію, а й активно взаємодіяти з контентом — розв'язувати задачі, експериментувати, застосовувати алгоритми.

- Коннективізм (Дж. Сіменс, Ст. Даунес)

Коннективізм є сучасною парадигмою навчання, яка відповідає реаліям цифрової епохи. У центрі цієї теорії знаходиться уявлення про знання як мережу зв'язків, що формується між людьми, цифровими об'єктами, інформаційними ресурсами та спільнотами. Навчання розглядається як уміння створювати, підтримувати та оновлювати ці зв'язки, постійно адаптуючись до змін інформаційного середовища.

Коннективізм визначає низку ключових принципів:

- важливість доступу до інформаційних ресурсів замість запам'ятовування великих обсягів даних;
- значущість навчання у спільнотах, онлайн-групах, професійних мережах;
- навчання як безперервний процес взаємодії з цифровими технологіями;
- здатність швидко знаходити, аналізувати та критично оцінювати інформацію.
- Для освітніх платформ ця теорія означає необхідність:
- інтеграції елементів соціального навчання (спільні проекти, форуми, відкриті обговорення);
- створення умов для обміну досвідом;
- використання інструментів колаборації;
- підтримки персоналізованих освітніх траєкторій.
- У контексті математики й інформатики коннективізм сприяє розвитку навичок роботи з великими інформаційними масивами, програмними інструментами, алгоритмами та навчальними спільнотами (GitHub, онлайн-курси, STEM-форуми).

- **Соціальний конструктивізм (Л. С. Виготський)**

Соціальний конструктивізм акцентує увагу на ролі соціальної взаємодії у формуванні знань. Виготський стверджував, що навчання є насамперед соціальним процесом, у якому знання конструюються через взаємодію, комунікацію та спільну діяльність. Важливим поняттям є «зона найближчого розвитку», тобто той рівень завдань, які учень здатен виконати з підтримкою педагога або більш досвідчених однолітків.

Для онлайн-платформ ця теорія визначає такі вимоги:

- забезпечення можливості взаємодії між учнями (групові завдання, дискусії, парна робота);
- наявність інструментів наставництва, підказок, методичних рекомендацій;
- створення інтерактивного контенту, який спонукає до активної діяльності, а не пасивного сприймання.

• У контексті навчання інформатики та математики конструктивістський підхід є надзвичайно ефективним, оскільки вимагає виконання практичних задач, моделювання, програмування, аналізу помилок. Саме у спільній діяльності учні краще засвоюють алгоритмічне мислення та математичну логіку.

• **STEM-педагогіка як методологічна основа дистанційного навчання**

STEM-педагогіка (Science, Technology, Engineering, Mathematics) є міждисциплінарним підходом, який орієнтований на розвиток практичних умінь, критичного мислення, творчості та інженерної культури учнів. У дистанційному форматі реалізація STEM-парадигми потребує застосування:

- віртуальних лабораторій (фізика, хімія, інформатика);
- симуляцій та моделювання (математичні моделі, алгоритми, 3D середовища);
- онлайн-платформ для програмування (Code.org, Replit, Scratch, Python-ноутбуки);
- інженерних цифрових інструментів (CAD-програми, симулятори робототехніки);
- проєктного та дослідницького навчання.

STEM-педагогіка органічно інтегрується у дистанційні платформи, оскільки цифрове середовище створює умови для виконання завдань, близьких до

реальних професійних ситуацій. Для учнів спеціальності «Середня освіта: Інформатика, Математика» важливо не просто засвоїти теоретичні знання, а навчитися їх застосовувати у практичних сценаріях, що є центральним принципом STEM-освіти.

1.4 Аналіз сучасних досліджень та практик

У сучасних умовах трансформації освітнього простору тема дистанційного навчання стає однією з ключових у педагогічній науці, методиці навчання та освітній політиці України. Цифровізація суспільства, розвиток інформаційно-комунікаційних технологій та необхідність забезпечення гнучких форматів навчання зумовили інтенсивне зростання кількості досліджень у сфері дистанційної освіти.

В Україні ця проблематика активно досліджується як у теоретичному, так і в практичному аспектах. Значний внесок у розвиток наукового підґрунтя дистанційної освіти зробили такі вчені, як В. Биков, який розробив концептуальні засади створення відкритого освітнього середовища; Н. Думанська, що аналізує дидактичні умови організації дистанційного навчання; Г. Кравцова, яка вивчає педагогічні технології розвитку пізнавальної активності учнів у цифровому середовищі; В. Кухаренко, автор численних праць із методики проведення онлайн-курсів і змішаного навчання;

О. Самойленко, що займається проблемами електронних освітніх ресурсів. Роботи цих дослідників визначають підходи до організації якісного дистанційного освітнього процесу, підкреслюючи важливість інтерактивності, індивідуалізації та науково обґрунтованої структури навчальних матеріалів.

Світовий досвід також є надзвичайно вагомим. Міжнародні дослідження, здійснені Д. Андерсоном, С. Віллером, Т. Едвардом та іншими, спрямовані на аналіз тенденцій розвитку дистанційної освіти у різних країнах та виявлення її переваг і обмежень. У їхніх роботах розглядаються питання ефективності онлайн-платформ, ролі віртуальних комунікацій, цифрової грамотності

учасників освітнього процесу, а також особливості інтеграції дистанційних технологій у традиційні освітні моделі.

Окремий напрям досліджень у світовій науці пов'язаний з технічним та педагогічним забезпеченням дистанційного навчання. Такі автори, як

Д. Мюллер, Р. Філіпс, Н. Хара, аналізують вплив інтерфейсів навчальних платформ, інструментів відеоконференцій, хмарних сервісів, а також засобів автоматизації оцінювання на якість та результативність освітнього процесу. Вони звертають увагу на важливість ергономіки, доступності та адаптивності освітнього середовища, що особливо актуально в умовах підготовки майбутніх учителів.

Проблема професійної підготовки майбутніх учителів інформатики в умовах цифровізації освіти також набула актуальності. Значна кількість українських науковців, серед яких М. Золочевська, А. Єршов, Н. Морзе,

Ю. Рамський, зосереджує увагу на формуванні цифрових компетентностей майбутніх педагогів, розвитку їхньої здатності використовувати інструменти дистанційного та змішаного навчання, створювати інтерактивний освітній контент. У наукових працях підкреслюється необхідність інтеграції сучасних вебтехнологій, хмарних сервісів, віртуальних лабораторій та платформ управління навчанням у підготовку вчителів інформатики й математики.

Швидкий розвиток сучасних комп'ютерних технологій, розширення засобів комунікації та цифрових сервісів створюють сприятливі передумови для масового впровадження дистанційного навчання у закладах освіти різного рівня. Все більше студентів та учнів обирають цей формат завдяки його гнучкості, мобільності та можливості персоналізації. У таких умовах виникає нагальна потреба модернізувати підходи до підготовки педагогічних кадрів, зокрема вчителів інформатики, забезпечуючи їх практичними навичками роботи з дистанційними платформами, інструментами Веб 2.0, системами управління навчанням, такими як Moodle, Google Classroom, Microsoft Teams, а також

сучасними сервісами для створення інтерактивних матеріалів — GeoGebra, Code.org, Quizizz, Kahoot тощо.

Метою даної статті є аналіз підходів та практик упровадження дистанційного й змішаного навчання у процес підготовки майбутніх учителів інформатики, а також представлення досвіду використання платформи Moodle та інструментів Веб 2.0 у створенні сучасного навчального контенту. Окрему увагу приділено питанням ефективності цих інструментів, їх перевагам, а також методичним рекомендаціям щодо інтеграції в навчальний процес педагогічних закладів вищої освіти.

Слід зазначити, що в науковій літературі існує значна різноманітність тлумачень поняття «дистанційне навчання». Це зумовлено багатовимірністю явища, що охоплює одночасно педагогічні, технологічні, психологічні та організаційні аспекти. У статті подано узагальнену таблицю, де зіставлено різні підходи до визначення цього терміна, що дозволяє комплексно оцінити сучасні наукові концепції та погляди на дистанційне навчання.

1.5 Порівняльний аналіз наявних освітніх платформ

Для визначення оптимальних рішень щодо розробки сучасної дистанційної освітньої платформи важливо здійснити комплексний порівняльний аналіз найбільш популярних інструментів, які активно використовуються у вітчизняній та світовій практиці. Порівняння охоплює функціональні, педагогічні, технічні, організаційні та STEM-орієнтовані характеристики:

Таблиця 1. Порівняння основних платформ дистанційного навчання

Платформа	Переваги	Обмеження
Google Classroom	1. Інтуїтивно зрозумілий інтерфейс, дуже легкий старт для вчителя й учня. 2. Глибока інтеграція з Google Workspace (Drive, Docs, Sheets, Meet, Forms). 3. Підтримка авторизації через Google-акаунт, безпека даних. 4. Підходить для щоденної шкільної роботи, домашніх завдань, оголошень. 5. Підтримка мобільних застосунків.	1. Обмежений набір інструментів для створення інтерактивних STEM-матеріалів (симуляцій, моделей, віртуальних лабораторій). 2. Відсутність розвиненої системи модульного курсового наповнення та складних траєкторій навчання. 3. Мінімальні можливості для аналітики та детального відстеження прогресу. 4. Немає SCORM, H5P, адаптивних елементів.
Moodle	1. Повноцінна LMS з модульною структурою та широким потенціалом адаптації. 2. Підтримка SCORM, H5P, гейміфікації, складних тестів, рейтингу, відкритих освітніх ресурсів. 3. Величезна кількість плагінів та розширень, розвинена система ролей. 4. Можливість налаштовувати курси під STEM: інтеграція з GeoGebra, CodeRunner, Virtual Labs. 5. Повністю безкоштовна з відкритим кодом.	1. Висока складність адміністрування, потребує технічних знань та серверних ресурсів. 2. Інтерфейс може здаватися перевантаженим для учнів молодшої школи. 3. Налаштування гейміфікації та складних елементів потребує часу і досвіду. Залежність від якості хостингу й технічного супроводу.
Edmodo	1. Побудована за принципом соціальної мережі, що робить її звичною для учнів. 2. Зручність обміну файлами,	1. Обмежені можливості для створення інтерактивних навчальних курсів і структурованих модулів.

	повідомленнями, спільними групами. 3. Підтримка батьківського контролю та залучення батьків. 4. Добре підходить для гуманітарних предметів і комунікації в класі.	2. Немає широкої аналітики, адаптивного навчання, інструментів STEM. 3. Мало налаштувань системи оцінювання та складних тестів. 4. Більше схожа на соціальну мережу, ніж на LMS повного циклу.
Zoom/ Teams	1. Потужні інструменти синхронної комунікації: відеоконференції, демонстрація екрану, робота в групах (breakout rooms). 2. Teams має інтеграцію з Office 365, планувальниками, SharePoint. 3. Ефективні для проведення онлайн-уроків, семінарів, консультацій.	1. Не є LMS і не мають системи наповнення освітніх матеріалів. 2. Відсутній структурований контент, тести, автоматизоване оцінювання. 3. Не підтримують SCORM, симуляції, гейміфікацію, віртуальні лабораторії. 4. Потребують стабільного Інтернету для відеозв'язку.
Всеукраїнська школа онлайн	1. Створена відповідно до державних стандартів і шкільної програми. 2. Містить відеоуроки, інтерактивні вправи та тести для основних предметів. 3. Орієнтована саме на українських учнів, включаючи учнів із тимчасово окупованих територій. 4. Безкоштовна та доступна без особливих технічних вимог.	1. Недостатня інтерактивність, обмежений набір інструментів для STEM-сфер. 2. Відсутність адаптивного навчання, персональних траєкторій, розширених аналітичних даних. 3. Мало засобів для комунікації між учнем і вчителем. 4. Платформа не дозволяє створювати власні курси повного циклу.

Потреба в платформі, яка поєднувала б функції LMS, інструменти для STEM-навчання та підтримку педагогічної взаємодії, залишається вкрай актуальною.

Висновок до Розділу 1

Порівняльний аналіз засвідчує, що жодна з існуючих платформ не забезпечує повного комплексу можливостей, необхідних для сучасної дистанційної освіти, особливо у сфері STEM, інформатики та математики. Їх можна групувати за функціональним призначенням:

1. **Google Classroom та Edmodo** — інструменти управління завданнями та комунікацією, проте не повноцінні LMS.

2. **Moodle** — найпотужніша LMS, але технічно складна та потребує значних ресурсів.

3. **Zoom/Teams** — забезпечують лише синхронну взаємодію, не маючи освітнього контенту.

4. **Всеукраїнська школа онлайн** — платформа контентного типу, без можливостей гнучкої педагогічної взаємодії.

Отже, залишається гостра потреба у створенні інтегрованої платформи нового покоління, яка поєднувала б:

- функції LMS (модульність, тести, аналітика);
- інструменти для STEM і програмування (симуляції, середовища виконання, інтерактивні моделі);
- засоби синхронної та асинхронної взаємодії;
- елементи адаптивного навчання;
- зручний інтерфейс для шкільного та університетського середовища.

Розділ 2. Принципи та технічні підходи до створення освітніх платформ для дистанційного навчання

2.1 Основні дидактичні принципи побудови освітніх платформ

Створення ефективної освітньої платформи не обмежується лише технічною реалізацією. Для забезпечення високої якості навчального процесу вона має будуватися на сучасних дидактичних засадах, які враховують особливості сприйняття, навчання та мотивації учнів. Дотримання цих принципів забезпечує комплексний підхід до організації дистанційного навчання та сприяє формуванню ключових компетентностей учнів, включаючи аналітичне та критичне мислення, цифрову грамотність, а також здатність до самостійного навчання.

Серед ключових дидактичних принципів можна виокремити такі:

1. Принцип науковості та системності

Освітня платформа повинна надавати достовірну, перевірену інформацію, подану у логічно структурованій формі. Це означає послідовність викладу матеріалу, системність у викладі понять та забезпечення міжпредметних зв'язків. Учень має отримувати цілісне уявлення про предмет, формувати стійкі знання та розуміння логіки розвитку наукових і технічних ідей.

Приклад реалізації: структуровані курси на Moodle, де матеріали розділені на модулі та теми з послідовним підходом до засвоєння теорії, практики та тестування.

2. Принцип доступності та посильності

Інтерфейс платформи, зміст матеріалів та навігація повинні відповідати віковим, когнітивним та психологічним особливостям учнів. Для забезпечення рівного доступу до навчання необхідно інтегрувати адаптивні механізми, мультимедійні підказки та підтримку для учнів з особливими освітніми

потребами. Платформа повинна бути інтуїтивно зрозумілою та зручною як для молодших школярів, так і для старших класів.

Приклад реалізації: використання інтерактивних відеоуроків з текстовими поясненнями та субтитрами, масштабування шрифту, голосові інструкції для учнів з порушеннями зору.

3. Принцип інтерактивності та активності

Сучасна освітня платформа має сприяти не лише пасивному сприйняттю матеріалу, а й активній взаємодії учня з контентом, учителем та однокласниками. Важливе місце посідає зворотний зв'язок, можливості самооцінювання, участь у проєктних та командних завданнях. Інтерактивність підвищує мотивацію та сприяє кращому закріпленню знань.

Приклад реалізації: онлайн-тести з миттєвим зворотним зв'язком, форуми для обговорення завдань, інтерактивні симуляції з фізики або програмування.

4. Принцип диференціації та індивідуалізації

Освітні платформи мають забезпечувати персоналізацію навчального процесу. Це передбачає врахування темпу навчання, попередньої підготовки, інтересів та стилю навчання кожного учня. Адаптивні алгоритми дозволяють формувати індивідуальні траєкторії навчання, виділяти додаткові ресурси для слабших учнів та пропонувати складніші завдання для більш підготовлених.

Приклад реалізації: системи рекомендацій на базі аналізу результатів тестів, адаптивні завдання для вивчення математики або програмування, диференційовані лабораторні роботи.

5. Принцип візуалізації та мультимедійності

Сучасні платформи повинні використовувати мультимедійні ресурси: графіку, анімацію, відео, інфографіку, інтерактивні моделі. Це значно полегшує засвоєння складних понять, особливо в математиці, фізиці, інформатиці та STEM-напрямах загалом. Візуалізація сприяє розвитку просторового та логічного мислення.

Приклад реалізації: інтерактивні 3D-моделі хімічних молекул, віртуальні фізичні експерименти, анімаційні алгоритми програмування.

6. Принцип практико-орієнтованості

Платформа повинна забезпечувати можливість застосування знань на практиці. Це включає розв'язання задач, моделювання процесів, участь у проєктній діяльності та використання віртуальних лабораторій. Практико-орієнтоване навчання допомагає учням зрозуміти зв'язок теорії з реальними процесами, розвиває критичне мислення та навички аналізу.

Приклад реалізації: завдання на створення алгоритмів у програмуванні, лабораторні роботи з фізики у віртуальних середовищах, симуляції економічних процесів або інженерних систем.

7. Принцип адаптивності та гнучкості освітнього середовища

Крім вищезгаданих принципів, сучасні платформи мають бути гнучкими: легко інтегрувати нові курси, технології, інструменти та методики. Це дозволяє швидко оновлювати навчальний контент відповідно до наукових досягнень та освітніх стандартів. Адаптивність забезпечує стійкість платформи до змін у навчальному процесі та потребах учнів.

Приклад реалізації: можливість підключення додаткових плагінів для симуляцій, модулів для онлайн-тестування, інтеграція з зовнішніми ресурсами (Coursera, Khan Academy).

Дидактичні принципи мають інтегруватися у структуру та функціональність освітньої платформи від початкового етапу проєктування. Тільки так можна створити середовище, яке сприяє не просто передачі знань, а формуванню компетентної, самостійної та мотивованої особистості, здатної до саморозвитку та ефективного навчання у цифрову епоху. Впровадження цих принципів особливо важливе у підготовці майбутніх учителів інформатики та STEM-напрямів, де синтез теоретичного знання та практичних навичок має вирішальне значення.:

2.2 Архітектура та структура сучасної освітньої онлайн платформи

З технічного боку освітня платформа являє собою складний програмний комплекс, який поєднує функціональні можливості для організації навчального процесу та відповідає високим вимогам щодо безпеки, масштабованості та зручності користування. Для забезпечення ефективної роботи платформи необхідно враховувати як дидактичні, так і технічні аспекти, оскільки технологічна реалізація без належної педагогічної підтримки не забезпечує повноцінного навчального процесу.

Архітектура освітньої платформи

Сучасна освітня платформа має багаторівневу архітектуру, що дозволяє розділити функціональні компоненти та забезпечити стабільність і гнучкість системи. Типова архітектура включає наступні рівні:

1. Інтерфейс користувача (UI/UX)

Інтерфейс є головним засобом взаємодії учня, учителя та адміністратора з платформою. Він має бути інтуїтивно зрозумілим, адаптивним до різних пристроїв — десктопів, планшетів, смартфонів. Важливо враховувати юзабіліті та психологічні аспекти сприйняття інформації: кольорові схеми, читабельність шрифтів, логіку навігації та швидкість доступу до матеріалів.

2. Серверна частина (бекенд)

Серверна частина відповідає за зберігання і обробку даних, управління користувацькими ролями, доступ до курсів, проведення аналітики та інтеграцію зовнішніх сервісів. Оптимальна архітектура бекенду забезпечує баланс навантаження, безперервну роботу системи та можливість масштабування при збільшенні кількості користувачів.

3. База даних

База даних зберігає інформацію про користувачів, навчальні матеріали, результати тестування, прогрес учнів, завдання та сертифікати. Для підвищення продуктивності рекомендується використання реляційних та нереляційних баз даних у залежності від типу контенту, що зберігається, та частоти запитів.

4. Модуль аналітики та моніторингу

Цей компонент дозволяє відстежувати прогрес учнів, аналізувати результати навчання, виявляти прогалини у знаннях та формувати персоналізовані рекомендації. Аналітика може включати статистику виконання завдань, рівень активності на платформі, ефективність різних методів навчання.

5. Модулі синхронного та асинхронного навчання

Платформа має підтримувати як синхронні (відеоконференції, інтерактивні заняття в реальному часі), так і асинхронні форми навчання (форуми, чати, дошки оголошень, відеоуроки). Таке поєднання дозволяє забезпечити гнучкий та доступний навчальний процес для всіх категорій учнів.

6. Модулі тестування, оцінювання та сертифікації

Ці модулі реалізують механізми формувального та підсумкового контролю знань, включаючи автоматичне оцінювання, тестові завдання різних типів, проєктні роботи та видачу сертифікатів про завершення курсу. Важливим аспектом є можливість інтеграції з системами рейтингу та мотивації.

7. Плагіни та інтеграції

Підтримка сторонніх сервісів та інструментів, таких як Khan Academy, GeoGebra, Code.org, YouTube, а також вбудованих ігрових чи STEM-симуляцій, дозволяє розширити функціональність платформи та підвищити залученість учнів. Інтеграції забезпечують доступ до сучасних освітніх ресурсів без необхідності додаткового встановлення програмного забезпечення.

Додаткові технічні аспекти

- Важливим є масштабованість . Платформа повинна легко адаптуватися до зростання кількості користувачів, нових курсів та додаткових сервісів без втрати продуктивності.
- Безпека та захист персональних даних — критично важливо для роботи з неповнолітніми учнями. Використання сучасних методів шифрування даних, багаторівневих систем авторизації та регулярних аудитів безпеки дозволяє мінімізувати ризики витоку інформації.
- Оптимізація продуктивності — застосування кешування, балансування навантаження та оптимізації запитів до бази даних дозволяє підвищити швидкість роботи платформи та забезпечити стабільну роботу при великому навантаженні.
- Мобільна доступність та кросплатформеність — забезпечення повноцінного функціоналу як на десктопах, так і на мобільних пристроях, що дозволяє учням і вчителям залишатися інтегрованими у навчальний процес у будь-якому місці та часі.

2.3 Застосування хмарних технологій у реалізації освітніх платформ

Хмарні обчислення (cloud computing) стали основою більшості сучасних освітніх систем і відіграють ключову роль у побудові платформ дистанційного навчання. Використання хмарних технологій забезпечує значні переваги, що роблять освітні платформи більш ефективними, доступними та масштабованими:

- Доступність: користувачі можуть підключатися до платформи з будь-якої точки світу у будь-який час, маючи лише інтернет-з'єднання. Це дозволяє реалізувати навчання без прив'язки до фізичного місця, що особливо актуально для учнів, які перебувають у різних регіонах або за кордоном. Постійний доступ до освітніх ресурсів забезпечує гнучкий графік навчання, підтримку дистанційних курсів і самостійного вивчення матеріалу.

- Гнучкість і масштабованість: хмарні платформи дозволяють оперативно масштабувати ресурси залежно від кількості активних користувачів. Це включає горизонтальне масштабування (додавання нових серверів для збільшення обчислювальної потужності) та вертикальне масштабування (збільшення ресурсів окремого сервера, таких як оперативна пам'ять і процесорна потужність). Така гнучкість забезпечує безперебійну роботу платформи навіть під час пикових навантажень, наприклад, під час проведення масових онлайн-уроків або тестувань.

- Підтримка спільної роботи: хмарні технології дозволяють учням і викладачам одночасно редагувати документи, виконувати завдання у групах та створювати спільні проєкти. Це сприяє розвитку командної взаємодії, комунікативних навичок та критичного мислення, що є особливо важливим у STEM-напрямах. Інструменти для колаборації, такі як спільні дошки, чат-кімнати та інтегровані календарі, забезпечують ефективну організацію навчального процесу та контроль за виконанням завдань.

- Оновлення та технічна підтримка: хмарні платформи дозволяють автоматично оновлювати систему, додаючи нові функції та інструменти без потреби у складному локальному налаштуванні. Це забезпечує постійне вдосконалення платформи, адаптацію до сучасних освітніх тенденцій і технологій. Крім того, завдяки хмарним рішенням підвищується надійність роботи серверів та зменшується час простою системи.

- Інтеграція з іншими сервісами: хмарні платформи легко інтегруються з різними освітніми та робочими сервісами, такими як Google Drive, Microsoft OneDrive, Dropbox, а також із віртуальними лабораторіями, STEM-середовищами та освітніми симуляторами. Це дозволяє значно розширити функціональні можливості платформи, забезпечити доступ до різноманітних навчальних ресурсів та підвищити мотивацію учнів через використання сучасних технологій і мультимедійних засобів.

- Ризики та обмеження: одночасно з перевагами хмарних технологій слід враховувати потенційні ризики. До них належить залежність від провайдера послуг, можливі технічні збої, загрози кібербезпеці та питання збереження конфіденційності даних учнів. Тому при використанні хмарних платформ необхідно впроваджувати механізми шифрування, резервного копіювання та багаторівневої авторизації, щоб мінімізувати ризики та забезпечити надійний захист персональної інформації.

- Додаткові переваги: хмарні платформи також дозволяють зберігати великі обсяги даних без необхідності у локальному обладнанні, забезпечують доступ до аналітики та звітності в режимі реального часу, а також дозволяють викладачам проводити персоналізоване навчання та адаптацію навчальних матеріалів під індивідуальні потреби учнів.

Таким чином, використання хмарних обчислень у побудові освітніх платформ створює фундамент для сучасного дистанційного та змішаного навчання, забезпечуючи високу доступність, гнучкість, інтеграцію та безпеку освітнього процесу.

2.4 STEM-орієнтовані компоненти освітніх платформ

Платформи для дистанційного навчання з математичних та інформатичних дисциплін мають включати різноманітні компоненти, що забезпечують активну та глибоку залученість учнів у навчальний процес. Такий підхід дозволяє поєднувати теоретичне засвоєння знань із практичним їх застосуванням, а також стимулює розвиток критичного мислення, логіки, креативності та навичок вирішення проблем. Серед ключових компонентів сучасних освітніх платформ виділяють наступні:

- Інтерактивні середовища програмування: приклади таких середовищ — repl.it, Scratch, Tynker, Code.org. Вони дозволяють учням створювати власні проєкти, писати і тестувати код онлайн, отримувати миттєвий зворотний зв'язок. Такі платформи особливо важливі для розвитку алгоритмічного мислення,

навичок логічного аналізу та структурованого підходу до розв'язання задач. Крім того, інтеграція з системами контролю версій або спільної роботи (GitHub, GitLab) дозволяє учням вивчати принципи колективного програмування та практичної командної взаємодії.

- Віртуальні лабораторії та симуляції: PhET Interactive Simulations, Labster, Algodoo, CoCalc та інші платформи забезпечують моделювання фізичних, математичних та інженерних експериментів у віртуальному середовищі. Учні можуть змінювати параметри, спостерігати наслідки, аналізувати результати і робити висновки. Це особливо важливо для дистанційного навчання, коли фізичні лабораторії недоступні, а також для інтеграції міждисциплінарних практичних завдань, де поєднуються елементи фізики, математики та інформатики.

- Графічні калькулятори та системи динамічної математики: GeoGebra, Desmos та інші подібні інструменти дозволяють вивчати функції, побудову графіків, геометричне моделювання та аналіз математичних моделей. Такі середовища надають можливість візуалізувати абстрактні математичні поняття, що підвищує розуміння та засвоєння складного матеріалу. Вони також сприяють формуванню аналітичного та критичного мислення через активне дослідження властивостей об'єктів і явищ.

- STEM-ігри та гейміфіковані вправи: платформи на кшталт Prodigy Math Game, Mathigon, Brilliant.org та Mangahigh використовують ігрові механіки, сюжетну подачу матеріалу та систему досягнень для підвищення мотивації учнів. Вони створюють умови для самостійного та заохочувального навчання, а також сприяють розвитку компетентностей XXI століття, таких як співпраця, креативність і вирішення комплексних проблем.

- Інтеграція з системами збору даних та сенсорними платформами: використання пристроїв Arduino, Micro:bit, Raspberry Pi дозволяє організовувати практичні завдання, пов'язані з експериментами та збором даних, навіть у

дистанційному форматі. Учні можуть збирати інформацію, аналізувати її та створювати власні моделі чи проєкти. Це стимулює розвиток проектного мислення, навичок інженерного моделювання та практичних компетентностей STEM-напрямку.

- Міжпредметна інтеграція та адаптивність: реалізація всіх цих компонентів у межах єдиної платформи дозволяє створити умови для глибокого міжпредметного занурення, характерного для STEM-підходу. Платформа може підтримувати адаптивне навчання, персоналізуючи завдання та матеріали відповідно до рівня підготовки, темпу навчання та інтересів учнів, що підвищує ефективність засвоєння знань та формування компетентностей.

- Аналітика та моніторинг навчального процесу: сучасні платформи оснащені інструментами відстеження прогресу учнів, оцінювання рівня виконання завдань, аналізу помилок і рекомендацій для покращення результатів. Це дозволяє вчителям оперативно коригувати навчальний процес та формувати індивідуальні траєкторії навчання.

- Використання хмарних технологій та інтеграцій: хмарні обчислення забезпечують доступність ресурсів у будь-який час, підтримку синхронного та асинхронного навчання, інтеграцію з іншими сервісами, такими як Google Drive, Microsoft OneDrive, а також віртуальні лабораторії та STEM-середовища. Це дозволяє створювати єдину екосистему для ефективного та інтерактивного навчання.

Таким чином, сучасні освітні платформи для математичних та інформатичних дисциплін формують комплексне навчальне середовище, де учень є активним учасником процесу, має доступ до широкого спектру інтерактивних інструментів і може застосовувати знання у практичних та проектних завданнях, що повністю відповідає принципам STEM-освіти та вимогам цифрового навчання.

2.5 Проблеми та виклики реалізації освітніх платформ

Незважаючи на значний прогрес у впровадженні цифрових освітніх технологій, сучасні системи дистанційного навчання стикаються з низкою серйозних викликів, які впливають на ефективність освітнього процесу та рівень засвоєння знань учнями.

- **Обмежений доступ до техніки та Інтернету:** значна частина учнів, особливо у сільській місцевості та віддалених регіонах, не має стабільного підключення до мережі Інтернет або сучасних пристроїв (комп'ютерів, планшетів, смартфонів), що робить неможливим повноцінне навчання онлайн. Така цифрова нерівність призводить до освітнього розриву між учнями з різних соціальних та географічних умов, а також обмежує можливості використання інтерактивних та мультимедійних компонентів навчальних платформ.

- **Низький рівень цифрової грамотності:** ефективне використання дистанційних платформ потребує від учнів та вчителів високого рівня цифрової компетентності. Брак навичок роботи з LMS-системами, інтерактивними середовищами, хмарними сервісами та STEM-інструментами може призводити до неефективного засвоєння матеріалу та зниження мотивації до навчання. Це питання особливо актуальне для педагогів старшого покоління та учнів, які не мають досвіду роботи з сучасними технологіями.

- **Психологічна втома та зниження мотивації:** тривала онлайн-активність призводить до когнітивного перевантаження, зниження концентрації уваги та втрати мотивації. Відсутність живої взаємодії, обмеженість соціальних контактів і монотонність дистанційних занять можуть негативно впливати на емоційний стан учнів та педагогів, збільшуючи ризик виникнення психологічного вигорання.

- **Проблеми з безпекою та етикою:** відкритість цифрових середовищ створює потенційні загрози кібербулінгу, витоку персональних даних учнів та педагогів, несанкціонованого доступу до навчальних матеріалів і результатів

тестування. Важливо забезпечувати дотримання правил інформаційної безпеки, впроваджувати шифрування даних, багаторівневі механізми аутентифікації та контроль доступу до навчальної інформації.

- Недостатнє методичне забезпечення: навіть найсучасніша та технічно досконала платформа не буде ефективною без дидактично грамотного створеного контенту, продуманих сценаріїв навчання та інтеграції педагогічних методик. Відсутність якісних навчальних матеріалів, адаптивних завдань та інструкцій для вчителів знижує ефективність використання дистанційних технологій.

Для подолання цих проблем необхідний комплексний підхід, що включає:

- Державну підтримку та інвестиції у розвиток освітньої інфраструктури, забезпечення технікою та стабільним Інтернетом всіх шкіл, особливо у віддалених регіонах.

- Підвищення кваліфікації педагогів через курси цифрової грамотності, тренінги з використання LMS, STEM-платформ та інтерактивних інструментів, що дозволить ефективніше організовувати навчальний процес.

- Розробку національних стандартів якості освітніх платформ, що гарантуватиме відповідність технічних, дидактичних та етичних вимог.

- Залучення міжнародних грантів, фондів та благодійних організацій, що сприятиме покращенню матеріально-технічного забезпечення освітніх закладів та впровадженню сучасних технологій у дистанційне навчання.

- Методичну підтримку учнів та педагогів, включаючи створення рекомендацій, інтерактивних посібників та адаптивних сценаріїв використання цифрових ресурсів у навчальному процесі.

Розширене врахування цих аспектів дозволяє забезпечити рівень цифрової та педагогічної готовності, необхідний для ефективного впровадження дистанційних і змішаних форм навчання. Воно також сприяє подоланню

соціальної та цифрової нерівності серед учнів і створює передумови для впровадження інноваційних STEM-підходів у школах та закладах вищої освіти.

Висновок до Розділу 2

1. Освітні платформи повинні ґрунтуватися на дидактичних принципах, що забезпечують доступність, мотивацію, інтерактивність та практичну спрямованість навчання.

2. Технічна архітектура платформи має підтримувати багаторівневу структуру: користувацький інтерфейс, навчальні модулі, системи аналітики, безпеки та адаптації.

3. Використання хмарних технологій забезпечує гнучкість, масштабованість, доступність та інтеграцію зі STEM-інструментами.

4. Для реалізації STEM-освіти необхідна наявність спеціалізованих компонентів: симуляцій, інтерактивних лабораторій, програмування, гейміфікації та інтерактивних середовищ.

5. Основними викликами залишаються доступність технологій, рівень цифрової компетентності учасників освітнього процесу, психологічна підтримка та методичне забезпечення.

6. Комплексний підхід до проєктування освітніх платформ дозволяє забезпечити інтеграцію дидактики, технологій та STEM-компонентів, що створює передумови для високоефективного дистанційного навчання.

Розділ 3. Практичні аспекти реалізації освітніх платформ для дистанційного навчання у середній школі

У цьому розділі детально описано процес розробки та впровадження онлайн освітньої платформи для дистанційного навчання, яка стала практичним втіленням теоретичних моделей та концепцій, розглянутих у попередніх розділах. Основною метою створення платформи було забезпечення ефективного навчання студентів спеціальності «Середня освіта: Інформатика, Математика» з урахуванням принципів STEM-освіти та сучасних педагогічних підходів.

Розробка платформи здійснювалась із використанням мови програмування Java, що забезпечує високу стабільність та масштабованість системи, а також підтримує інтеграцію з різними бібліотеками та фреймворками для побудови веб-додатків. Для реалізації інтерфейсу використовувалися сучасні вебтехнології, що гарантують адаптивність, інтуїтивність та зручність користування для різних типів користувачів — студентів, викладачів та адміністраторів платформи.

Архітектурний підхід передбачає багаторівневу структуру системи, де окремо виділено модулі для взаємодії з користувачем, серверної логіки, зберігання даних та аналітики навчального процесу. Така структура дозволяє забезпечити ефективну роботу платформи навіть при високому навантаженні та реалізовувати індивідуальні траєкторії навчання, автоматизовану систему оцінювання та підтримку інтерактивних STEM-інструментів.

Особлива увага під час розробки приділялась інтеграції компонентів, що дозволяють студентам виконувати практичні завдання, працювати з віртуальними лабораторіями, програмними середовищами та симуляціями, а також брати участь у колаборативних проєктах. Завдяки цьому платформа

поєднує дидактичні принципи сучасної освіти, педагогічні методи STEM-напрямку та технічні рішення для ефективного дистанційного навчання.

3.1 Педагогічний підхід у розробці методики навчання дітей, з урахуванням ігрових і пізнавальних цілей, для досягнення продуктивнішого навчання

У сучасних умовах стрімкого розвитку цифрових технологій система освіти переживає глибинну трансформацію, яка вимагає переосмислення традиційних підходів до викладання інформатики та, зокрема, програмування. Інформаційне суспільство вимагає від учня не лише засвоєння базових навичок роботи з комп'ютером, а здатності до творчої діяльності, критичного мислення, аналітичних здібностей та готовності до вирішення комплексних технічних проблем. Саме тому вивчення програмування набуває особливого значення, як елемент STEM-освіти та як складова формування цифрової компетентності. Одним із найефективніших способів забезпечити глибоке й мотивоване засвоєння програмування є використання навчально-ігрових платформ, що поєднують педагогічні принципи, інтерактивність, візуалізацію та практико-орієнтований підхід.

У старшій школі навчально-ігрові платформи дозволяють створити навчальне середовище, орієнтоване на діяльність учня, у якому теоретичні знання подаються через практику, дослідження та експеримент. Згідно із сучасними педагогічними концепціями, зокрема конструктивізмом, учень найкраще опановує матеріал тоді, коли він здобуває знання шляхом власної активності та відкриття, а не пасивного сприймання. Програмування саме по собі є діяльністю, що передбачає експерименти, створення продуктів, перевірку гіпотез, роботу з помилками та поступовий рух від простих задач до складніших. Тому інтеграція елементів гри, змагальної, симуляції та проектної діяльності у процес навчання є педагогічно виправданою та методично доцільною.

Використання навчально-ігрових платформ також відповідає віковим та психологічним особливостям старшокласників. У цьому віці учні прагнуть до самостійності, самовираження, визнання у групі, орієнтуються на результат і мають високий рівень потреби у мотивації. Традиційні форми роботи, такі як пояснення нового матеріалу та виконання однотипних завдань, часто не викликають у старшокласників достатньої зацікавленості. Натомість інтерактивні середовища, можливість негайно побачити результат своєї роботи, виконання алгоритмічних вправ у форматі гри або місії стимулюють природну пізнавальну активність учнів і сприяють формуванню навчальної мотивації.

Навчально-ігрові платформи мають виразний інноваційний потенціал, оскільки дозволяють учням не лише виконувати завдання, а й взаємодіяти з навчальним матеріалом у змінних умовах, працювати з віртуальними тренажерами, моделювати поведінку програм і створювати проєктні продукти. Вони забезпечують середовище, де учень отримує миттєвий зворотний зв'язок - одну з найцінніших педагогічних характеристик, що значно прискорює навчання. Завдяки автоматичній перевірці, підказкам, поясненням помилок та адаптивному добору завдань учні відчують, що працюють у середовищі, яке підтримує їхню діяльність і допомагає долати труднощі.

Окремої уваги потребує дидактичне підґрунтя використання таких платформ. Вони ґрунтуються на принципах наочності, системності, активності, індивідуалізації та практичної спрямованості. Принцип наочності реалізується через візуалізацію алгоритмів, демонстрацію роботи програм, графічне представлення структур даних і логічних переходів у виконанні коду. Завдяки цьому учням стає легше зрозуміти абстрактні поняття, такі як цикли, умовні оператори, рекурсія чи стек викликів. Принцип активності реалізується через виконання інтерактивних завдань, де учень не може залишатися пасивним - він повинен експериментувати, редагувати код, спостерігати поведінку програми та

виправляти помилки. Саме така діяльність забезпечує формування ґрунтовних програмістських навичок.

Практична спрямованість є однією з ключових переваг навчально-ігрових платформ. Учень має можливість працювати не лише з абстрактними вправами, а й з симуляціями реальних задач: створенням елементарних алгоритмів керування персонажами, моделюванням фізичних явищ, роботою з об'єктами та подіями, розв'язанням логічних та технічних ситуацій. Таким чином формуються не тільки навички програмування, а й вміння застосовувати програмні засоби у практичному контексті, що є надзвичайно важливим для підготовки учнів до подальшого навчання та професійної діяльності у сфері інформаційних технологій.

Індивідуалізація навчання, якої так часто не вистачає в традиційній шкільній практиці, реалізується завдяки функціоналу платформи. Система може адаптувати складність завдань, пропонувати додаткові матеріали, аналізувати типові помилки учня, формувати рекомендації для подальшого навчання. Це дозволяє працювати з різними рівнями підготовки класу, уникати ситуації, коли сильні учні нудьгують, а слабші - не встигають.

У контексті розробленої платформи "NexusSTEM" особливе значення має інтеграція елементів STEM-освіти. Програмування не може розглядатися ізольовано від математики, логіки, фізики або технологій, адже воно є інструментом для вирішення міждисциплінарних задач. Завдання на платформі можуть передбачати моделювання простих фізичних явищ, створення алгоритмів для математичних обчислень, логічних головоломок, аналізу даних. Завдяки цьому учні навчаються бачити програмування як універсальний засіб опису процесів і явищ.

Методичне використання навчально-ігрової платформи передбачає продуману структуру уроку. У традиційній практиці вчитель часто витрачає

значну частину часу на пояснення нового матеріалу, тоді як набагато ефективніше перенести частину теоретичного пояснення у візуальні модулі платформи. Учитель може розпочати урок з мотиваційного завдання у форматі мінігри або логічної ситуації, що потребує використання певної конструкції програмування. Такий підхід одразу включає учня в активність та дає йому розуміння, навіщо саме потрібна нова тема. У подальшому учитель пропонує учням працювати з симуляціями, які демонструють поведінку алгоритмів у різних умовах, що формує міцні когнітивні зв'язки між теорією та практикою.

Після опанування основного матеріалу учні переходять до виконання інтерактивних вправ на платформі. Ці вправи можуть мати різну форму: від написання коду з нуля до заповнення пропусків, реконструкції алгоритмів, пошуку помилок або створення власного проєкту. Саме завдяки цьому в учнів формується здатність застосовувати знання у нових ситуаціях - один із ключових критеріїв сформованості компетентностей.

Навчально-ігрові платформи також дозволяють організувати роботу над проєктами, що є важливою складовою STEM-підходу. Проєктна діяльність передбачає створення власного програмного продукту: гри, симуляції, анімації або інтерактивної моделі. Учні можуть працювати в командах, розподіляючи ролі, відповідальність і завдання. Це розвиває комунікацію, співпрацю, креативність, самоорганізацію та інші навички XXI століття.

Оцінювання навчальних досягнень учнів на платформі "NexusSTEM" будується на основі аналізу виконання взаємодій із системою, результатів тестових завдань, динаміки успішності та сформованості програмістського мислення. Платформа надає детальну статистику, що дозволяє викладачеві своєчасно виявляти труднощі, коригувати навчальні траєкторії та планувати індивідуальну роботу з учнями. Водночас автоматичне оцінювання звільняє учителя від рутинної перевірки і дає змогу зосередитися на творчих, методичних і консультаційних аспектах навчання.

Важливим компонентом методики є підтримка мотивації. Гейміфікація, яка включає бали, рівні, нагороди, досягнення та сюжетні елементи, допомагає утримувати увагу учня, створює відчуття успіху, підсилює внутрішню мотивацію. Програмування часто викликає труднощі через складність абстракцій, а гейміфіковане середовище знижує рівень тривоги та дозволяє учню сприймати помилки як частину гри, а не як невдачу.

У роботі з платформою важливо також враховувати психологічні аспекти навчання. Старшокласники нерідко бояться припуститися помилки або сумніваються у своїх силах при виконанні програмних завдань. Система, яка надає підказки, пояснення та можливість багаторазово повторювати завдання без покарання, створює безпечне освітнє середовище. Це формує позитивний досвід взаємодії з програмуванням і сприяє розвитку впевненості у власних здібностях.

Інноваційність запропонованої платформи полягає також у тому, що вона поєднує традиційні педагогічні інструменти з сучасними цифровими технологіями, забезпечуючи інтеграцію хмарних сервісів, аналітики, інтерактивних навчальних модулів і засобів гейміфікації. Такий підхід відповідає викликам сучасної освіти та формує у учнів ключові компетентності, визначені державними освітніми стандартами: цифрову, інформаційно-комунікаційну, математичну, логіко-аналітичну, а також компетентності саморозвитку, продуктивної діяльності та критичного мислення.

Застосування платформи у шкільному середовищі також має соціально-педагогічний ефект. Учні, які працюють з інтерактивними системами, стають більш самостійними, дисциплінованими, відповідальними за результат власної діяльності. Вони навчаються аналізувати власні дії, робити висновки, оцінювати свої можливості та прогрес. Платформа сприяє демократизації навчального процесу, оскільки створює рівні умови для всіх учнів незалежно від початкового рівня підготовки та дозволяє кожному рухатися власним темпом.

У підсумку можна стверджувати, що навчально-ігрові платформи є потужним інструментом у викладанні програмування учням старшої школи. Вони поєднують технологічні можливості зі зрозумілою педагогічною логікою, створюють умови для індивідуалізації навчання, сприяють формуванню ключових компетентностей і забезпечують високий рівень залучення учнів до навчального процесу. Розроблена платформа "NexusSTEM" є цілісним педагогічно-технологічним рішенням, що відповідає сучасним вимогам освіти та відкриває широкі можливості для подальшого розвитку методики викладання програмування у школах.

1. Алгоритм формування принципів навчання, що описують методичні аспекти роботи з навчально-ігровими платформами

У процесі створення сучасної методики викладання програмування важливо не лише застосовувати окремі педагогічні підходи, а й вибудовувати послідовний алгоритм, на основі якого формуються дидактичні принципи використання навчально-ігрових платформ. Побудова такого алгоритму дозволяє системно визначити методичні орієнтири, забезпечити узгодженість навчального процесу та відобразити логіку педагогічного впливу у використанні цифрових технологій. Алгоритм формування принципів навчання виконує функцію методологічного "каркасу", на який спирається розробка уроків, вибір інструментів, типів вправ, способів роботи із завданнями й організація діяльності учнів.

Процес формування таких принципів можна описати як інтелектуально-педагогічну послідовність, що ґрунтується на аналізі потреб учнів, специфіки предмету "Інформатика", компетентнісних вимог та можливостей інтерактивного програмного середовища. Цей алгоритм починається з визначення цілей навчання, адже саме вони формують загальний вектор методики. Під час вивчення програмування у старшій школі цілі включають розвиток алгоритмічного мислення, засвоєння базових та проміжних

конструкцій мови програмування, формування компетентностей у галузі аналізу та вирішення проблем, розвиток самостійності, творчості та здатності працювати із складними задачами. Визначивши цілі, педагог переходить до другого кроку - аналізу змісту навчального матеріалу, який має бути представлений на платформі. Важливо оцінити, які теми є найбільш абстрактними, які вимагають візуалізації, де учням найчастіше потрібне додаткове пояснення, а які аспекти можна підсилити за допомогою інтерактивних симуляцій чи ігрових механік.

Наступним етапом алгоритму є визначення специфічних особливостей цільової аудиторії - учнів старшої школи. Це включає оцінку попереднього досвіду, рівня цифрової грамотності, різниці у темпах навчання, мотиваційних факторів та інтересів учнів. Врахування цих факторів є необхідною умовою формування принципів, адже вони забезпечують відповідність навчально-ігрової платформи реальним можливостям і потребам старшокласників. Наприклад, для учнів із сильним математичним підґрунтям важливо передбачити завдання, що вимагають логічного мислення й моделювання.

Для тих, хто краще навчається через наочність, потрібні візуалізовані алгоритми, симуляції та покрокові пояснення.

Після аналізу змісту та аудиторії визначається набір педагогічних принципів, які мають лягти в основу використання платформи. Цей крок передбачає формування концептуальної рамки - узагальнених правил, що відображають ефективні способи подання матеріалу, організації активності та підтримки учнів. Принципи не є випадковим набором порад - вони повинні бути виведені логічно, відповідно до закономірностей навчання та психолого-педагогічних особливостей учнів. До таких принципів можуть належати: принцип візуалізації та симуляції, принцип інтерактивності, принцип поступовості й ускладнення, принцип індивідуалізації навчання, принцип

мотиваційної підтримки через гейміфікацію, принцип практичної спрямованості, принцип рефлексії та досягнення результатів.

Наступний етап алгоритму - операціоналізація цих принципів, тобто їх переведення у конкретні методичні рішення. На цьому етапі принципи перетворюються на конкретні педагогічні інструменти: типи завдань, механізми візуалізації, структуру модулів, підходи до автоматичної перевірки, інтерактивні елементи, логіку підказок. Наприклад, принцип поступовості може проявлятися у тому, що платформа надає учню спочатку алгоритми з готовою структурою, потім - з пропусками, а далі - завдання, які необхідно реалізувати повністю самостійно. Принцип візуалізації може реалізуватися через анімації виконання коду, через графічне представлення станів змінних чи через віртуальних роботів, що рухаються відповідно до написаного алгоритму.

Окремий модуль алгоритму пов'язаний із технічною реалізацією педагогічних рішень. На цьому етапі відбувається адаптація ігрових механік під навчальні цілі: визначення рівнів, системи досягнень, нагород, сюжетних блоків, а також механізмів внутрішньої мотивації. Важливо, щоб технічні інструменти були не просто додатком до навчального процесу, а логічно вбудованим елементом методики. Наприклад, якщо учень отримує бали, то вони мають відображати навчальний прогрес; якщо надаються підказки - вони повинні бути педагогічно вмотивованими, а не автоматичними; якщо рівні гри ускладнюються - це повинно відображати дидактичний принцип розвитку компетентностей.

Заключним кроком алгоритму є перевірка й корекція сформованих принципів на основі педагогічного експерименту. На цьому етапі вчитель або розробник аналізує, наскільки ефективно працюють обрані методичні підходи, чи відповідають вони реальній поведінці учнів, чи сприяють досягненню навчальних результатів. Після проведення уроків або серії занять збирається зворотний зв'язок, аналізуються статистичні дані, виконання завдань, частота

помилки, швидкість проходження рівнів, активність учнів. На основі цих даних відбувається корекція алгоритму: змінюється структура завдань, адаптується складність, уточнюються принципи мотивації, доповнюються або видаляються певні ігрові елементи.

Таким чином, алгоритм формування принципів навчання у контексті використання навчально-ігрових платформ являє собою багаторівневу, циклічну та адаптивну систему, що забезпечує науково-педагогічне підґрунтя для інтеграції інноваційних технологій у процес викладання програмування. Завдяки такому алгоритму методика перестає бути набором інтуїтивних дій і перетворюється на цілісну, теоретично обґрунтовану систему, здатну забезпечити якісне та результативне засвоєння складних понять програмування.

3.2 Вибір архітектури програмного забезпечення

Для побудови сучасної онлайн освітньої платформи було обрано багаторівневу (three-tier) архітектуру, яка є однією з найпоширеніших практик у розробці веб-додатків завдяки своїй гнучкості, масштабованості та можливості окремого розвитку різних компонентів системи. Така архітектура дозволяє чітко розділити функціональні обов'язки між різними шарами, підвищити стійкість платформи та спростити її подальшу підтримку та модернізацію.

1. Презентаційний рівень (Presentation Layer / Frontend)

Цей рівень відповідає за взаємодію користувача з платформою. Він включає веб-інтерфейс, через який студенти та викладачі отримують доступ до курсів, навчальних матеріалів, тестів, форумів та інтерактивних модулів. Інтерфейс розроблено з урахуванням принципів UI/UX-дизайну, що забезпечує інтуїтивну навігацію, адаптивність для різних пристроїв (десктопи, планшети, смартфони) та легкість у користуванні. Особливу увагу приділено інтерактивності: користувачі можуть одночасно працювати з віртуальними лабораторіями, програмними середовищами та іншими STEM-компонентами.

2. Рівень бізнес-логіки (Application Layer / Backend)

Цей рівень обробляє всі запити від користувачів, реалізує механізми авторизації та аутентифікації, управління курсами, тестами та завданнями. Тут же відбувається управління правами доступу користувачів, обробка взаємодії між різними модулями платформи та забезпечується логіка адаптивного навчання. Рівень бізнес-логіки також відповідає за інтеграцію з зовнішніми сервісами (наприклад, Google Drive, GitHub, STEM-симуляції) та автоматизацію оцінювання завдань. Для реалізації цього рівня використовується Java з сучасними фреймворками для веб-розробки, що забезпечує стабільність та ефективність обробки великих обсягів даних.

3. Рівень доступу до даних (Data Layer / Database)

На цьому рівні здійснюється збереження та управління всією інформацією платформи: профілі користувачів, навчальні матеріали, результати тестування, статистика прогресу, історія взаємодії та журнал активності. Використовується реляційна база даних PostgreSQL, що забезпечує надійне зберігання даних, підтримку транзакцій та гнучке формування запитів для аналітики. Рівень доступу до даних також передбачає механізми резервного копіювання, шифрування та безпечної роботи з персональною інформацією учнів та викладачів.

Використання three-tier архітектури дозволяє окремо модернізувати фронтенд, бізнес-логіку або базу даних без порушення роботи всієї системи, а також забезпечує масштабованість платформи при збільшенні кількості користувачів чи впровадженні нових функцій. Такий підхід створює основу для подальшого розширення системи, інтеграції з хмарними сервісами та STEM-інструментами, а також підвищує надійність і безпеку онлайн-навчання.

3.3 Вибрані технології

Для розробки онлайн освітньої платформи було обрано сучасний стек технологій, який забезпечує високу продуктивність, безпеку, масштабованість та зручність у подальшій підтримці системи. Нижче наведено детальний опис вибраних компонентів і фреймворків:

Таблиця 2. Порівняння основних технологій для розробки платформи

Компонент	Технологія Фреймворк	Опис та обґрунтування вибору
Мова програмування	Java 17	Забезпечує стабільність, безпеку, кросплатформеність і підтримку багаторівневої архітектури. Java має великий набір бібліотек і фреймворків, що значно спрощує реалізацію складної логіки бізнес-рівня.
Фреймворк для бекенду	Spring Boot	Дозволяє швидко створювати стійкі веб-додатки з мінімальною конфігурацією. Підтримує REST API, інтеграцію з базами даних, безпеку та масштабування. Забезпечує модульність та легке розширення платформи новими функціями.
База даних	PostgreSQL	Потужна реляційна база даних з підтримкою складних запитів, транзакцій і масштабованості. Підходить для зберігання великого обсягу інформації про користувачів, курси, тестування та статистику.
Веб-інтерфейс (фронтенд)	Thymeleaf / React (опціонально)	Thymeleaf забезпечує інтеграцію з Spring Boot і генерує динамічні HTML-сторінки, тоді як React дозволяє реалізувати сучасний SPA (Single Page Application) з інтерактивними компонентами, що підвищує зручність та швидкість роботи користувачів.

Безпека	Spring Security + JWT	Забезпечує надійну аутентифікацію та авторизацію, контроль доступу до різних ролей користувачів, шифрування даних та захист від несанкціонованого доступу. JWT дозволяє реалізувати безпечну передачу токенів у REST API.
REST API	Spring Web + Jackson	Використовується для організації взаємодії фронтенду та бекенду через стандартизований RESTful інтерфейс. Jackson забезпечує серіалізацію та десеріалізацію даних у форматі JSON.
Тестування	JUnit, Mockito	Дозволяє проводити юніт-тестування та модульне тестування бізнес-логіки, забезпечуючи стабільність системи та виявлення помилок на ранніх етапах розробки.
Збірка проекту	Maven	Забезпечує управління залежностями, автоматизацію процесу збірки та розгортання проекту. Maven спрощує інтеграцію сторонніх бібліотек і підтримку версій компонентів.
Розгортання	Docker, Docker Compose	Використовується для контейнеризації компонентів платформи, що забезпечує просте розгортання на будь-якому сервері, масштабування та ізоляцію сервісів. Docker Compose дозволяє запускати комплексну систему з кількох контейнерів одночасно.

Вибір цього набору технологій забезпечує стабільність і ефективність роботи платформи, підтримку багаторівневої архітектури, інтеграцію з

сторонніми STEM-сервісами та інструментами гейміфікації. Крім того, обрана технологічна база дозволяє легко додавати нові функціональні модулі, забезпечувати безпечну роботу з персональними даними учнів і проводити аналітику навчального процесу в режимі реального часу.

3.4 Основні функціональні можливості платформи

Основні функціональні модулі освітньої платформи

1. Реєстрація та авторизація користувачів

- Забезпечує безпечний доступ до системи та персоналізоване відображення контенту залежно від ролі користувача.
- Передбачає реєстрацію за електронною поштою, через соціальні мережі або інтегровані державні сервіси (наприклад, «Дія»).
- Підтримує багатофакторну автентифікацію та шифрування паролів для безпеки персональних даних.
- Система веде лог активності користувачів для подальшої аналітики та моніторингу.

2. Ролі користувачів: студент, викладач, адміністратор

- Студент: доступ до курсів, навчальних матеріалів, тестів, оцінок і форумів. Можливість формування індивідуальної траєкторії навчання.
- Викладач: створення та редагування курсів, завдань, тестів, відстеження прогресу учнів, організація дистанційних занять та вебінарів.
- Адміністратор: управління користувачами, ролями, курсами, налаштування платформи, моніторинг продуктивності та забезпечення безпеки.
- Система дозволяє гнучко налаштовувати права доступу для різних груп користувачів.

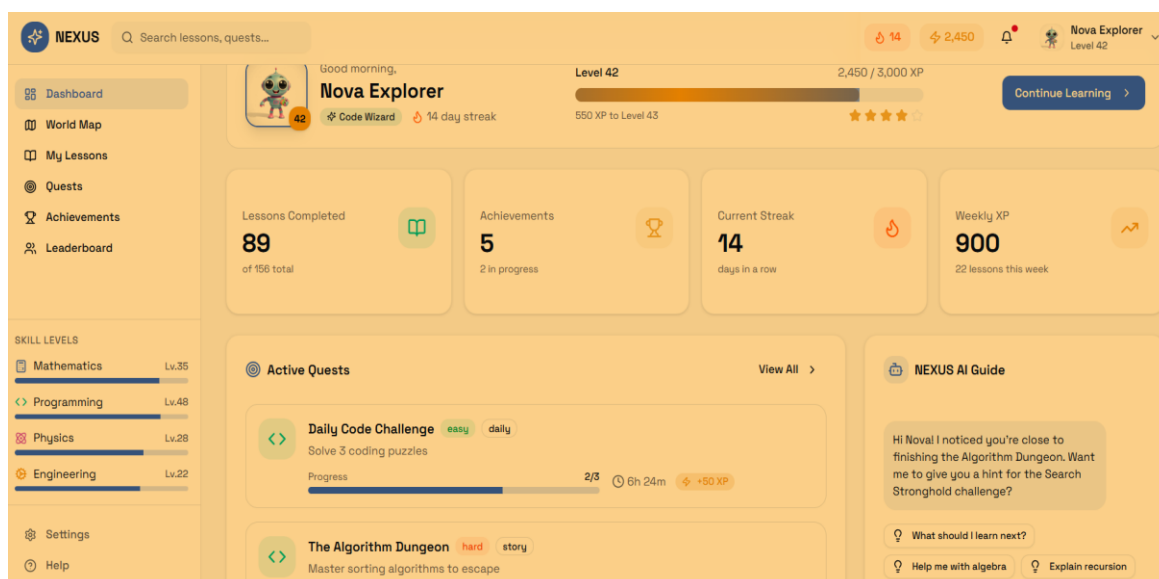


Рис. 1. Головне меню користувача

3. Створення та управління курсами

- Викладачі можуть формувати курси з різних предметів, додавати модулі, лекції, завдання, відеоуроки та інтерактивні елементи.
- Підтримка різних форматів контенту: текст, відео, презентації, симуляції та інтерактивні вправи.
- Можливість організації курсів у серії модулів з визначеною послідовністю проходження.
- Налаштування доступності матеріалів для окремих груп студентів або індивідуальних користувачів.

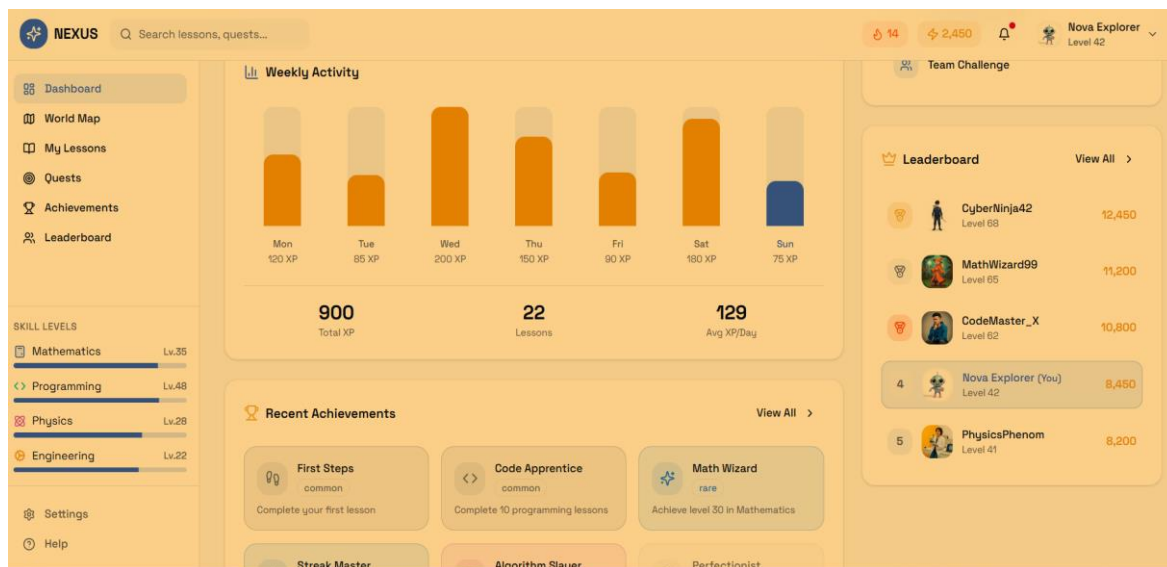


Рис. 2. Меню досягнень користувача

4. Проходження навчальних модулів та тестів

- Студенти можуть виконувати завдання та проходити тести безпосередньо на платформі, отримуючи миттєвий зворотний зв'язок.
- Підтримка різних типів тестових завдань: множинний вибір, відкриті запитання, програмування, інтерактивні завдання STEM.
- Система фіксує час виконання, помилки та успішність проходження для подальшого аналізу прогресу.
- Доступні адаптивні тести, які підлаштовуються під рівень знань студента.

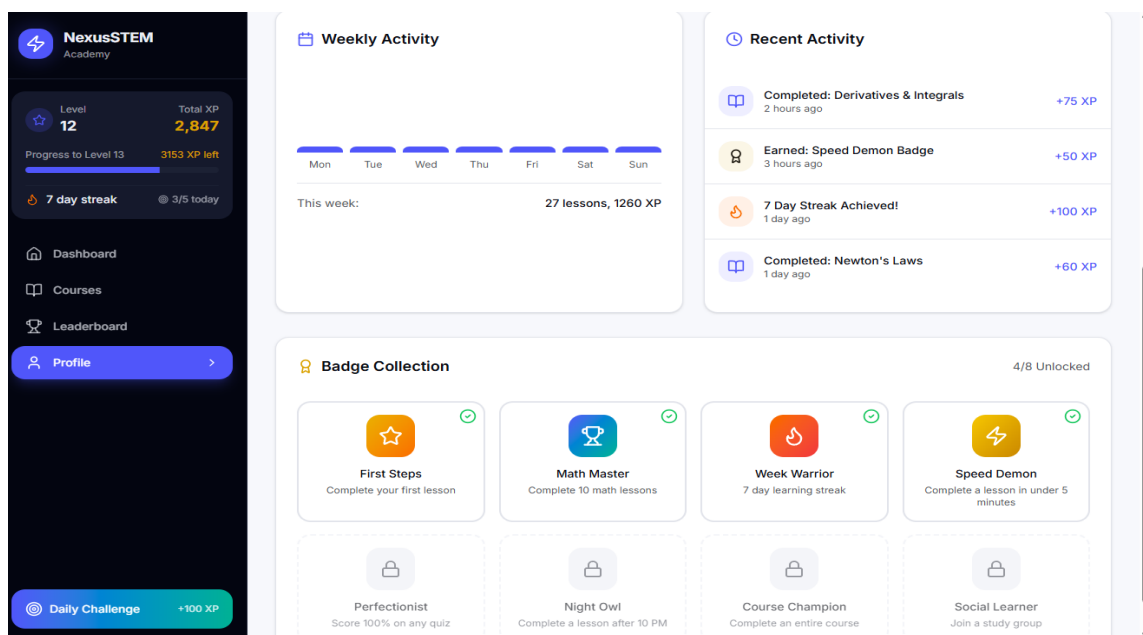


Рис. 3. Профіль користувача

5. Перегляд результатів та статистики

- Інтерактивні дашборди для студентів та викладачів: відображають результати тестів, виконані завдання, середній бал та прогрес у курсі.
- Аналітичні інструменти дозволяють виявляти слабкі теми, прогнозувати успішність та коригувати траєкторію навчання.
- Можливість експорту звітів у форматі PDF або CSV для подальшого використання у звітності та оцінюванні.

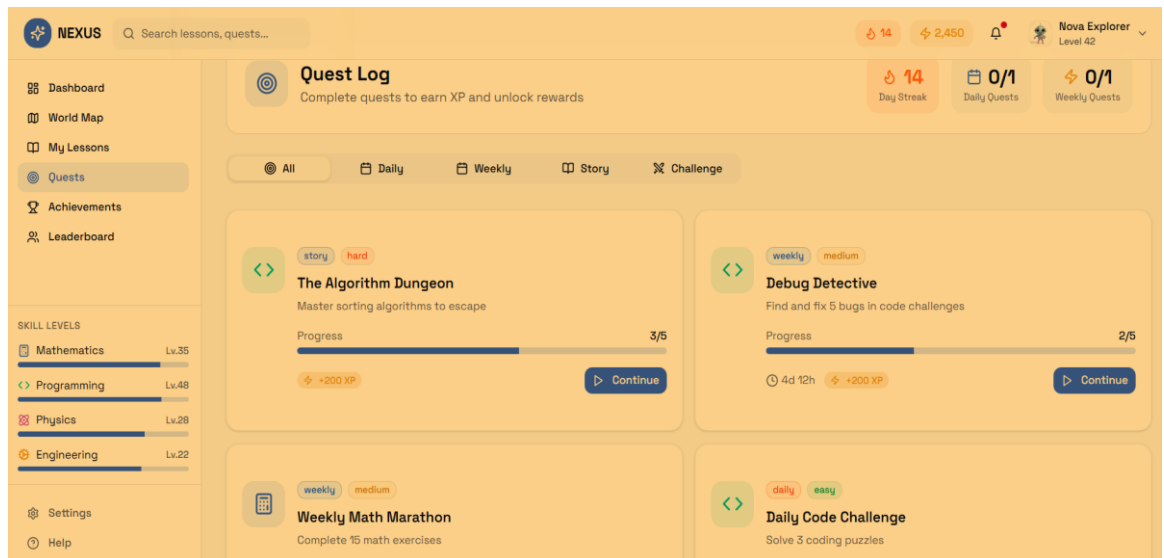


Рис. 4. Вибіркові дисципліни

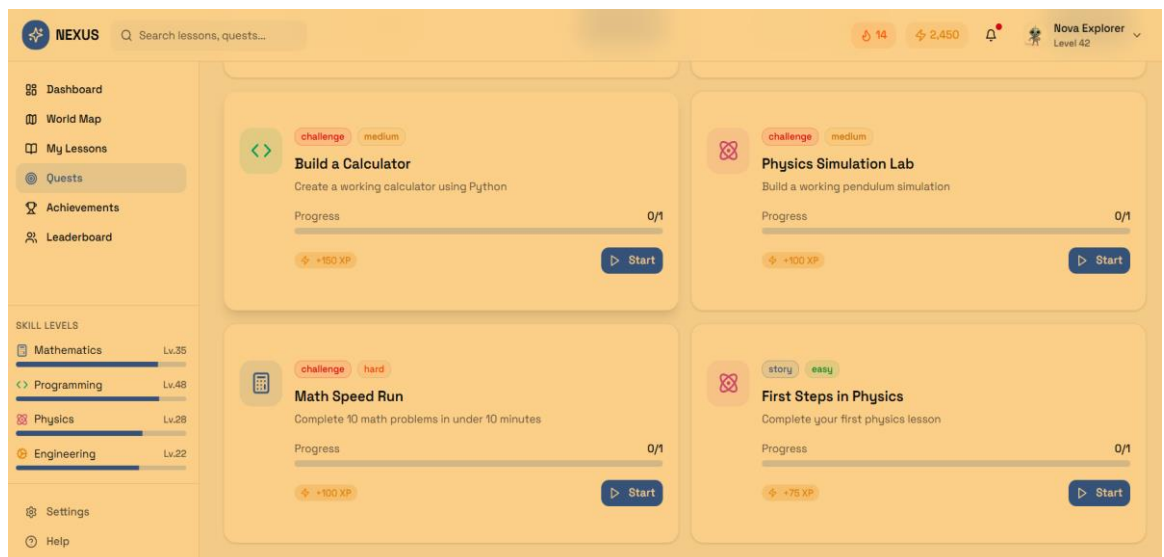


Рис. 4.2 . Вибіркові дисципліни

6. Панель адміністратора

- Центральний модуль управління платформою, що дозволяє контролювати роботу всіх користувачів і курсів.
- Можливість моніторингу серверів, ресурсів, активності користувачів і безпеки системи.
- Інструменти для створення резервних копій, управління оновленнями та інтеграціями зі сторонніми сервісами.

- Адміністратор може налаштовувати політики доступу, модерацію контенту та вести аудит дій користувачів для забезпечення безпеки та дотримання законодавчих норм.

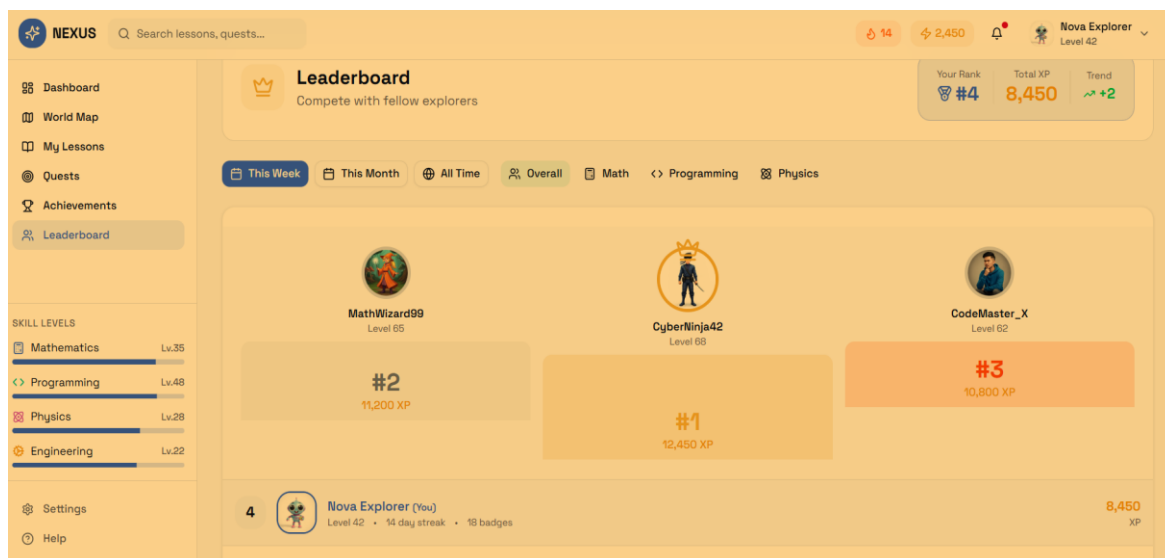


Рис. 5. Дошка успішності серед учнів

3.5 Етапи розробки

Розробка онлайн освітньої платформи є комплексним процесом, який охоплює як планування та проєктування, так і реалізацію, тестування та розгортання системи. Для забезпечення ефективності та послідовності робіт весь процес поділено на кілька основних етапів.

Етап 1: Постановка задачі та проєктування

На цьому етапі визначаються цілі та функціональні вимоги платформи:

- Визначення основних функціональних вимог: описуються ролі користувачів (студент, викладач, адміністратор), ключові модулі (курси, тести, адміністративна панель), необхідність інтеграції з зовнішніми сервісами та вимоги до безпеки.
- Створення діаграм прецедентів (Use Case): демонструється, як користувачі взаємодіють із системою; кожен прецедент описує конкретну дію, наприклад, реєстрація користувача або проходження тесту.

- Розробка UML-діаграм:
- Діаграми класів – показують структуру даних і зв'язки між сутностями.
- Діаграми послідовності – демонструють порядок виконання процесів, наприклад, авторизація та проходження курсу.
- Діаграми компонентів – відображають архітектуру системи, взаємодію між модулем бекенду, фронтенду та базою даних.
- Проектування бази даних (ER-діаграма): визначення сутностей, атрибутів та зв'язків між ними, що дозволяє забезпечити збереження даних користувачів, курсів, тестів та результатів

Етап 2: Створення базового каркасу проєкту

- Ініціалізація Spring Boot-проєкту через Spring Initializr з вибором необхідних модулів.
- Підключення залежностей: Spring Data JPA (для роботи з базою даних), Spring Security (для автентифікації та авторизації), Spring Web (для REST API), драйвер PostgreSQL.
- Конфігурація підключення до бази даних: налаштування властивостей DataSource, Hibernate, JPA та механізму управління транзакціями.

Етап 3: Реалізація реєстрації та автентифікації

- Створення сутностей User та Role: визначення атрибутів користувача (ім'я, електронна пошта, пароль, роль).
- Впровадження Spring Security + JWT:
- Аутентифікація користувачів через токени JWT.

- Авторизація на основі ролей (ROLE_STUDENT, ROLE_TEACHER, ROLE_ADMIN).
- Реалізація REST – ендпоінтів для реєстрації та входу :

```

@Validated
@RestController
@RequiredArgsConstructor
@RequestMapping("/v1/user")
@RateLimiter(name = "defaultLimiter")
@Tag(name = "User", description = "General Resource for managing users")
public class UserController {

    private final UserService userService;
    private final ClientService clientService;

    @GetMapping("/{id}")
    @PreAuthorize("hasAuthority('ADMIN')")
    @Operation(summary = "Endpoint that allows to fetch user by identifier")
    @ApiResponses(value = {
        @ApiResponse(responseCode = "200", description = "Success: returned user related data"),
        @ApiResponse(responseCode = "400", description = "Bad input provided, validation issue"),
        @ApiResponse(responseCode = "401", description = "Unauthorized access"),
        @ApiResponse(responseCode = "403", |
            description = "Access denied. The user does not have permission to use this resource"),
        @ApiResponse(responseCode = "404", description = "User by provided id is not found"),
        @ApiResponse(responseCode = "500", description = "Unexpected server error")
    })
    public UserResponseDto getUser(
        @Parameter(name = "Requested User Identifier")
        @PathVariable("id") @org.hibernate.validator.constraints.UUID @NotBlank String userId) {
        return userService.findUserById(UUID.fromString(userId));
    }
}

```

Рис. 6. Логіка платформи відносно користувача

- Валідація даних користувача: перевірка унікальності електронної пошти, складності пароля, обов'язкових полів.

Етап 4: Реалізація управління курсами

- Створення сутностей: Course, Lesson, Test, Answer, які дозволяють формувати структуру курсів і навчальних модулів.
- Функціонал курсу:

- Створення та редагування курсів викладачем.
- Призначення курсів студентам і групам.
- Перегляд навчальних матеріалів: тексти, презентації, відео, інтерактивні вправи.
- Проходження тестів, відстеження результатів та прогресу.

Етап 5: Відображення інтерфейсу користувача

- Для швидкого прототипу використовується Thymeleaf для генерації серверної HTML-сторінки.
- Опціонально можна застосовувати SPA-фреймворки (React, Angular, Vue.js) для більш динамічного інтерфейсу.
- Взаємодія з API через REST-запити забезпечує отримання даних, надсилання результатів тестування, оновлення контенту та управління курсами.

Етап 6: Адміністративна панель

- Доступна лише для ролі ADMIN.
- Можливості адміністратора:
 - Управління користувачами (створення, редагування, видалення).
 - Контроль за курсами та навчальними матеріалами.
 - Моніторинг прогресу студентів та статистики по курсах.
 - Налаштування безпеки та прав доступу, аудит дій користувачів.

```
@Slf4j
@Configuration
@RequiredArgsConstructor
@EnableWebSecurity(debug = true)
@EnableMethodSecurity(jsr250Enabled = true)
public class SecurityConfiguration {

    private final RateLimitFilter rateLimitFilter;
    private final ApplicationProperties properties;
    private final UserDetailsService userDetailsService;
    private final AnalysisApiUsageFilter analysisApiUsageFilter;
    private final JwtAuthenticationFilter jwtAuthenticationFilter;
    private final CaptchaLimitFilter captchaLimitFilter;

    @Bean
    @ConditionalOnProperty(value = "app.cors.enabled", havingValue = "true")
    public SecurityFilterChain corsFilterChain(HttpSecurity http) throws Exception {
        return filterChain(http, corsEnabled: true);
    }

    @Bean
    @ConditionalOnProperty(value = "app.cors.enabled", havingValue = "false")
    public SecurityFilterChain domainUnrestrictedFilterChain(HttpSecurity http) throws Exception {
        return filterChain(http, corsEnabled: false);
    }

    private SecurityFilterChain filterChain(HttpSecurity http, boolean corsEnabled) throws Exception {
        http.headers(HeadersConfigurer<HttpSecurity> headers -> headers.cacheControl(HeadersConfigurer.CacheControlConfig::disable))
            .csrf(AbstractHttpConfigurer::disable);

        if (corsEnabled) {
            http.cors(Customizer.withDefaults());
        } else {
            http.cors(AbstractHttpConfigurer::disable);
        }

        http.authorizeHttpRequests(AuthorizationManagerRequestMat... registry -> registry
            .requestMatchers(HttpMethod.OPTIONS, @"/**").permitAll()
            .requestMatchers(
                @"/api/score/v1/admin/**",
                @"/api/score/v1/user/**")
            .hasRole(Role.ADMIN.name())
            .requestMatchers(
```

Рис. 7. Налаштування логіки захисту платформи

```

@Bean
@DependsOn("userRepository")
@ConditionalOnProperty(value = "app.cors.enabled", havingValue = "true")
public CorsFilter corsFilter(@Autowired UserRepository userRepository, @Autowired JwtService jwtService) {
    var source = new DynamicUrlBasedConfigurationSource(jwtService, userRepository);
    var config = new CorsConfiguration();
    var cors = properties.getWebSecurityConfig();

    config.setAllowCredentials(true);
    cors.getCorsExposeHeaders().forEach(config::addExposedHeader);
    cors.getCorsAllowedOrigins().forEach(config::addAllowedOrigin);
    cors.getCorsAllowedMethods().forEach(config::addAllowedMethod);
    cors.getCorsAllowedHeaders().forEach(config::addAllowedHeader);
    source.registerCorsConfiguration(pattern: "**", config);

    log.info("Allowed on startup: origins - {}\n headers - {}", config.getAllowedOrigins(), config.getExposedHeaders());
    return new CorsFilter(source);
}

@Bean
@ConditionalOnProperty(value = "app.cors.enabled", havingValue = "false")
public CorsFilter domainUnrestrictedCorsFilter() {
    var source = new UrlBasedCorsConfigurationSource();
    var config = new CorsConfiguration();

    config.setAllowCredentials(false);
    config.addExposedHeader("*");
    config.addAllowedOrigin("*");
    config.addAllowedMethod("*");
    config.addAllowedHeader("*");
    source.registerCorsConfiguration(pattern: "**", config);

    log.info("Allowed origins on startup: {}", config.getAllowedOrigins());
    log.info("Allowed headers on startup: {}", config.getExposedHeaders());
    return new CorsFilter(source);
}

```

Рис. 7.2. Налаштування логіки захисту платформи

Етап 7: Тестування та дебаг

- Юніт-тестування: покриття основних модулів (сервіси, контролери, репозиторії) за допомогою JUnit та Mockito.
- Інтеграційне тестування REST API: перевірка роботи всіх ендпоінтів, обробки запитів та коректності відповіді сервера.
- Тестування безпеки: перевірка JWT-токенів, рольових обмежень, доступу до захищених ресурсів.

- Навантажувальне тестування (опційно): оцінка швидкодії платформи при великій кількості одночасних користувачів.

Етап 8: Контейнеризація та розгортання

- Створення Dockerfile для бекенду: упаковка додатку та всіх залежностей у контейнер для забезпечення портативності та стабільності.
- Створення docker-compose.yml для одночасного запуску бекенду та бази даних PostgreSQL, забезпечення зв'язку між контейнерами.
- Розгортання в середовищі хостингу: Heroku, Render, VPS або хмарні сервіси (AWS, Azure, Google Cloud).
- Налаштування CI/CD (опційно): автоматизація збірки, тестування та деплою при внесенні змін у код.
- Механізми резервного копіювання та масштабування: горизонтальне та вертикальне масштабування контейнерів, забезпечення безперервності роботи платформи.

3.6 Висновок до розділу 3

Розроблена веб-орієнтована освітня платформа забезпечує повний цикл дистанційного навчання та інтегрує ключові функціональні та педагогічні компоненти:

- Простий та зручний інтерфейс

Платформа має інтуїтивну навігацію, адаптивний дизайн для десктопів, планшетів та смартфонів, що забезпечує легкий доступ до курсів, завдань та тестів для учнів і викладачів.

- Гнучка система курсів

Підтримується створення та редагування навчальних модулів, уроків і тестів, можливість призначення курсів студентам та налаштування індивідуальних навчальних траєкторій.

- Надійна автентифікація та авторизація

Реалізована через Spring Security та JWT, що забезпечує безпечний вхід користувачів, контроль доступу за ролями (студент, викладач, адміністратор) та захист персональних даних.

- Можливість масштабування та гнучке розгортання

Використання Docker та docker-compose дозволяє швидко масштабувати систему, підключати додаткові сервери або ресурси, забезпечуючи стабільну роботу платформи при збільшенні кількості користувачів.

- Підтримка STEM-навчання

Платформа інтегрує інструменти для математичних та інформатичних дисциплін: інтерактивні завдання, тести, симуляції, віртуальні лабораторії та графічні середовища (GeoGebra, PhET, Code.org), що дозволяє забезпечити глибоке міжпредметне занурення та практичне застосування знань.

- Аналітика та моніторинг навчального процесу

Система надає можливість відстежувати прогрес студентів, аналізувати результати тестів та виконання завдань, виявляти прогалини у знаннях та формувати звіти для викладачів.

- Інтеграція з зовнішніми сервісами

Підтримуються сторонні ресурси та сервіси (Google Drive, Microsoft OneDrive, віртуальні лабораторії), що розширює функціонал платформи та підвищує її корисність для навчання.

Завдяки реалізації цих компонентів платформа не лише забезпечує технічну основу дистанційного навчання, а й створює комфортні умови для навчальної діяльності, мотивує студентів до активного опанування STEM-дисциплін та забезпечує ефективну взаємодію між учнем і викладачем.

ВИСНОВОК

Проведене дослідження дозволило глибше осмислити роль освітніх платформ у сучасному навчальному процесі, зокрема в контексті дистанційного та STEM-орієнтованого навчання у середній школі. У роботі було детально розглянуто теоретичні та методологічні основи створення цифрових освітніх середовищ, проведено аналіз сучасних платформ, їхніх функціональних і дидактичних характеристик, а також визначено основні педагогічні та технічні вимоги до побудови ефективної та адаптивної системи для дистанційного викладання інформатики та математики.

Дослідження показало, що найбільш ефективні платформи характеризуються такими властивостями:

- Зручний доступ до навчального контенту з будь-якого пристрою та у будь-який час — платформи повинні бути адаптивними для роботи на десктопах, планшетах та смартфонах, що дозволяє учням і викладачам здійснювати навчальний процес незалежно від місця перебування;
- Підтримка різноманітних форматів взаємодії — включаючи відеоуроки, інтерактивні завдання, форуми, чати та симуляції; це сприяє активізації навчальної діяльності, підвищує мотивацію та дозволяє організовувати ефективну комунікацію між учнями та вчителем;
- Інструменти оцінювання, аналітики та зворотного зв'язку — платформа повинна надавати можливості формувального та підсумкового контролю знань,

автоматизованого відстеження прогресу учнів, а також надавати викладачам рекомендації щодо персоналізації навчання;

- Можливості для реалізації міжпредметних зв'язків у STEM-освіті — включення інтерактивних лабораторій, віртуальних симуляцій, середовищ для програмування та математичного моделювання дозволяє забезпечити комплексне, практично орієнтоване навчання.

У результаті практичного блоку було розроблено прототип освітньої платформи «NexusSTEM», що відповідає основним вимогам сучасного дистанційного навчання:

- має модульну архітектуру, що забезпечує гнучке розширення функціоналу та інтеграцію нових сервісів;
- забезпечує простий та інтуїтивний інтерфейс для користувачів з різними рівнями цифрової компетентності;
- реалізує модуль тестування та оцінювання, що дозволяє відстежувати успішність студентів у реальному часі;
- інтегрується з GeoGebra та іншими STEM-інструментами, що сприяє активному засвоєнню математичних та інформатичних понять;
- включає систему авторизації та рольове управління користувачами, що забезпечує безпеку даних та контроль доступу;
- надає панель адміністратора для керування користувачами, курсами та статистикою використання платформи.

Тестування показало, що платформа готова до подальшого розгортання та адаптації в шкільному середовищі, а її функціональні можливості відповідають потребам дистанційного STEM-навчання.

Таким чином, реалізація власної освітньої платформи підтвердила її ефективність як інструменту підвищення доступності та якості освіти в умовах сучасних цифрових викликів. Дослідження засвідчило доцільність розробки та впровадження подібних рішень, орієнтованих на підтримку шкільної STEM-освіти, що відкриває перспективи для подальших наукових розробок, вдосконалення дидактичних методик та практичних впроваджень у сфері цифрової освіти.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Биков В.Ю. Моделі організаційних систем відкритої освіти: монографія. – К.: Атіка, 2008.
2. Шишкіна М.П. Хмаро орієнтоване середовище навчання в школі: теоретико-методологічні засади. – К.: ІПОД НАПН України, 2017.
3. Anderson T., Dron J. Three generations of distance education pedagogy. *International Review of Research in Open and Distance Learning*, 12(3), 2011.
4. Siemens G. Connectivism: A Learning Theory for the Digital Age. – 2005.
5. Moore M.G. The theory of transactional distance. In: Keegan D. *Theoretical principles of distance education*. London: Routledge, 1993.
6. Garrison D.R., Anderson T., Archer W. Critical inquiry in a text-based environment: Computer conferencing in higher education. *The Internet and Higher Education*, 2(2-3), 2000.
7. Власенко К. Огляд освітніх платформ: можливості та обмеження // Освіта.ua. – 2022. – <https://osvita.ua/school/83291/>
8. Міністерство освіти і науки України. Концепція розвитку дистанційної освіти в Україні до 2025 року. – <https://mon.gov.ua>
9. Moodle.org – Офіційний сайт освітньої платформи Moodle. – <https://moodle.org>

10. Google for Education – Classroom overview, Google Inc., 2023. – <https://edu.google.com>
11. Пейперт С. *Мистецтво, розум і комп'ютер: новий стиль навчання*. Київ: Либідь, 2018.
12. Виготський Л. С. *Мислення і мова*. Київ: Радянська школа, 2019 (перевидання).
13. Брунер Дж. *Процес освіти*. Cambridge: Harvard University Press, 2010.
14. Papert, S. *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books, 1993.
15. Wing, J. “Computational Thinking.” *Communications of the ACM*, 49(3), 33–35, 2006.
16. Resnick, M. *Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play*. MIT Press, 2017.
17. Крамаренко Т. Г. *Методика навчання інформатики у школі*. Київ: Академвидав, 2021.
18. Code.org Research Group. *The Impact of Game-Based Learning on K–12 Computer Science Outcomes*. Code.org, 2020.
19. European Schoolnet. *STEM Education Practices in European Secondary Schools*. Brussels, 2021.
20. Галузевий стандарт освіти України. Державний стандарт базової середньої освіти, 2020.
21. Hsin, W.-J., & Wu, H. “The Effect of Game-Based Learning on Students’ Motivation in Programming Education.” *Journal of Educational Technology & Society*, 22(3), 36–49, 2019.
22. Kelleher, C., & Pausch, R. “Lowering the Barriers to Programming: A Survey of Visual Programming Systems.” *ACM Computing Surveys*, 37(2), 83–137, 2005.

ДОДАТКИ

Додаток А. Приклад інтерфейсу учня (макет)

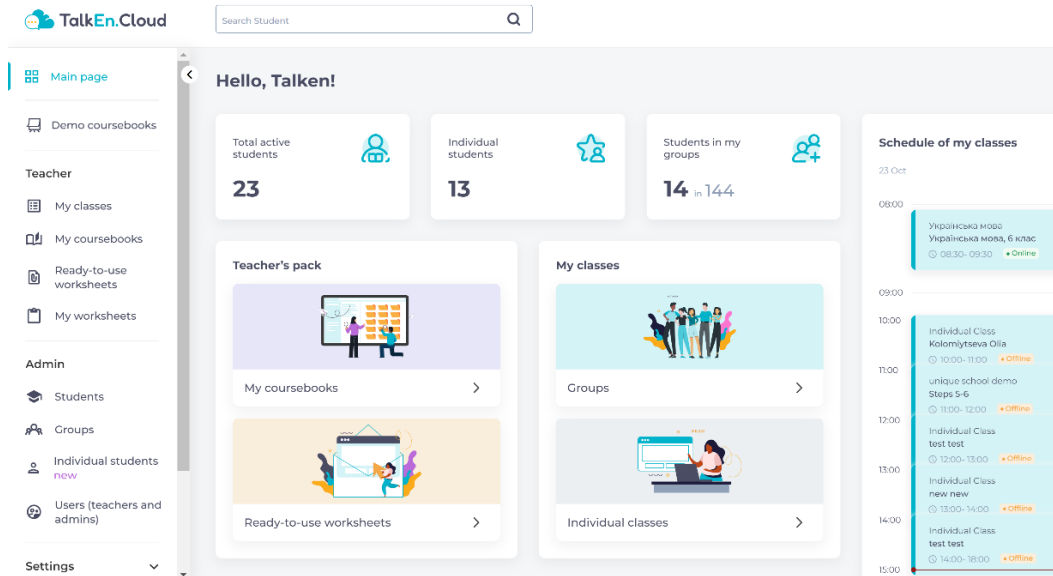


Рис. 1. TalkEn платформа



Рис. 2. Molp платформа

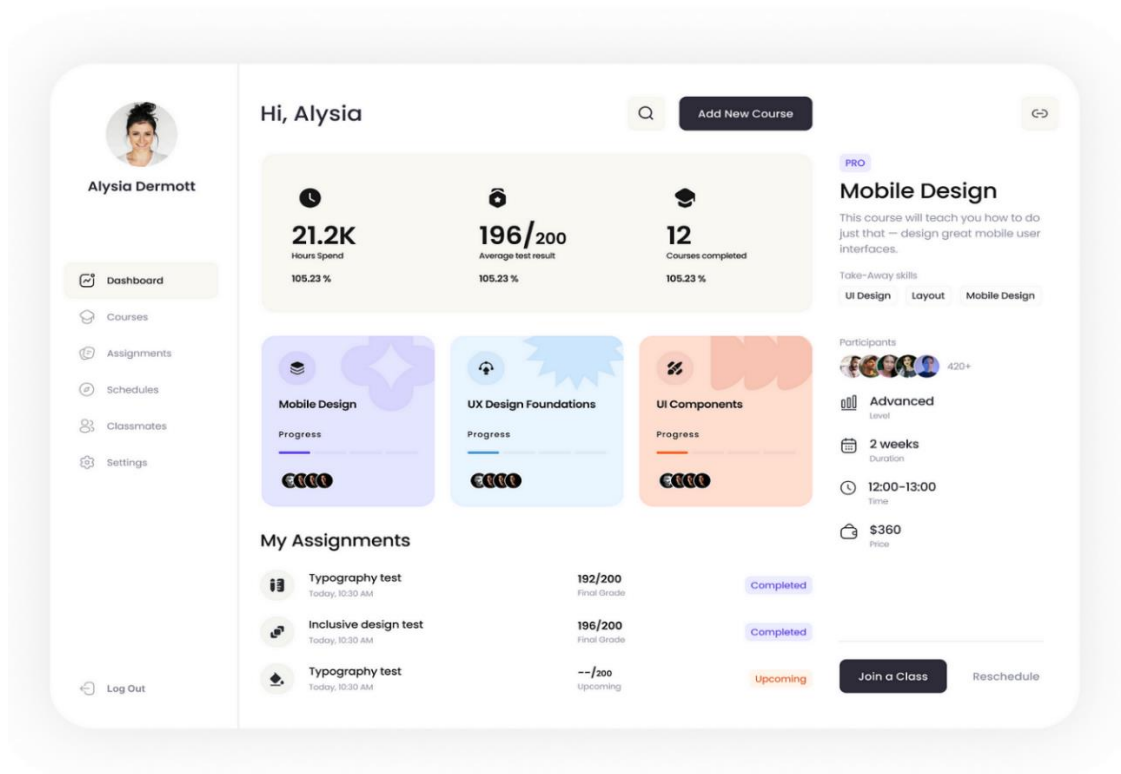


Рис. 3. Приклад успішності користувача на платформі

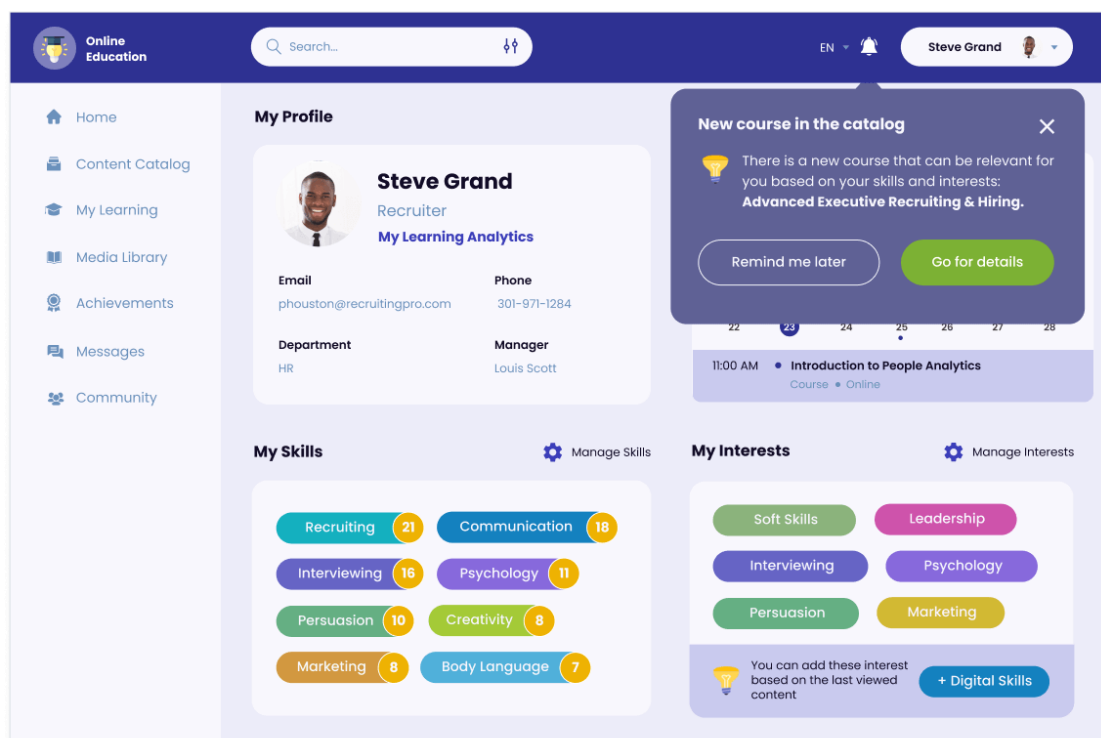


Рис. 4. Приклад профілю користувача на платформі і досягнень

Додаток В. Приклад схеми структури бази даних в sql форматі та фото

```
CREATE TABLE User (  
    UserID INT PRIMARY KEY,  
    Username VARCHAR(255),  
    Email VARCHAR(255),  
    Password VARCHAR(255),  
    UserType VARCHAR(50)  
);
```

```
CREATE TABLE Course (  
    CourseID INT PRIMARY KEY,  
    CourseName VARCHAR(255),  
    Description TEXT,  
    Price DECIMAL(10, 2)  
);
```

```
CREATE TABLE CourseContent (  
    ContentID INT PRIMARY KEY,  
    CourseID INT,  
    ContentType VARCHAR(50),  
    FOREIGN KEY (CourseID) REFERENCES Course(CourseID)  
);
```

```
CREATE TABLE Enrollment (  
    EnrollmentID INT PRIMARY KEY,  
    UserID INT,
```

```
CourseID INT,  
EnrollmentDate DATE,  
CompletionStatus VARCHAR(50),  
FOREIGN KEY (UserID) REFERENCES User(UserID),  
FOREIGN KEY (CourseID) REFERENCES Course(CourseID)  
);
```

```
CREATE TABLE Payment (  
    PaymentID INT PRIMARY KEY,  
    UserID INT,  
    PaymentDate DATE,  
    Amount DECIMAL(10, 2),  
    PaymentMethod VARCHAR(50),  
    FOREIGN KEY (UserID) REFERENCES User(UserID)  
);
```

```
CREATE TABLE Result (  
    ResultID INT PRIMARY KEY,  
    UserID INT,  
    CourseID INT,  
    QuizID INT,  
    Score INT,  
    FOREIGN KEY (UserID) REFERENCES User(UserID),  
    FOREIGN KEY (CourseID) REFERENCES Course(CourseID),  
    FOREIGN KEY (QuizID) REFERENCES Quiz(QuizID)  
);
```

```

CREATE TABLE Quiz (
    QuizID INT PRIMARY KEY,
    CourseID INT,
    QuizName VARCHAR(255),
    Description TEXT,
    TotalMarks INT,
    FOREIGN KEY (CourseID) REFERENCES Course(CourseID)
);

```

