

Вінницький національний технічний університет

(повне найменування вищого навчального закладу)

Факультет інтелектуальних інформаційних технологій та автоматизації

(повне найменування інституту, назва факультету(відділення))

Кафедра комп'ютерних наук

(повна назва кафедри, (циклової комісії))

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Інформаційна технологія створення електронних підручників в системі JetIQ»

Виконав: студент 2-го курсу, групи 2КН-23м
спеціальності 122 - «Комп'ютерні науки»

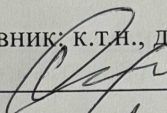
(шифр і назва напрямку підготовки, спеціальності)

221

Власенко Д.В

(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН

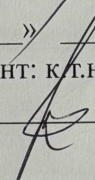


Сілагін О. В.

(прізвище та ініціали)

« 05 » 12 2024 р.

Опонент: к.т.н., професор каф. АІТ



Паламарчук

Є.А.

(прізвище та ініціали)

« 05 » 12 2024 р.

Допущено до захисту

Завідувач кафедри КН

д.т.н., проф. Яровий А. А.

(прізвище та ініціали)

« 06 » 12 2024 р.

Вінниця ВНТУ – 2024 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних наук
Рівень вищої освіти II-й (магістерський)
Галузь знань – 12 «Інформаційні технології»
Спеціальність – 122 «Комп'ютерні науки»
Освітньо-професійна програма – «Системи штучного інтелекту»

ЗАТВЕРДЖУЮ

Завідувач центру диджиталізації
освітнього процесу, к.т.н.,
доц. Паламарчук Є.А.
11.10 2024 року

Завідувач кафедри КН, д.т.н.,
проф. Яровий А. А.
11.10 2024 року

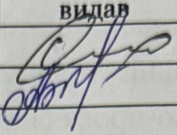
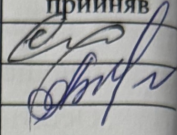
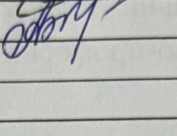
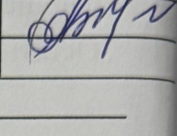
**ЗАВДАННЯ НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ
СТУДЕНТУ**

Власенку Данилу Володимировичу
(прізвище, ім'я, по батькові)

- Тема роботи: Інформаційна технологія створення електронних підручників в системі JetIQ.
Керівник роботи: Сілагін О.В.
Затверджені наказом ВНТУ від «--17 09 2024 року № 310
- Строк подання студентом роботи 11 11 2024 року
- Вихідні дані до роботи: веб-технології створення електронного підручника; база даних типу SQL; кількість колекцій в базі даних - не менше 2 шт; Мова програмування – для створення веб-систем з можливістю маніпулювання даними й управління базами даних
- Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити): інформаційні технології створення електронних підручників; розробка методів створення підручників для платформи дистанційного та змішаного навчання; реалізація інформаційної технології електронного підручника в системі JetIQ; тестування модуля; економічна частина.
- Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень): модель контурів електронної книги; взаємодії користувачів та

сервісів електронної книги; діаграми електронний посібник - викладач-студент в середовищі електронної книги; комунікацій і емоційних повідомлень електронної книги.

6. Консультанти розділів проекту (роботи)

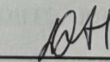
Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Сілагін О.В., к.т.н., доцент		
5	Адлер О.О., к.т.н., доцент		

7. Дата видачі завдання 02.09.2024

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Обґрунтування доцільності розробки	02.09.24	18.09.24	
2	Розробка методів створення підручників для платформи дистанційного та змішаного навчання	18.09.24	02.10.24	
3	Реалізація інформаційної технології електронного підручника в системі JetIQ	03.10.24	17.10.24	
4	Підготовка економічної частини	18.10.24	28.10.24	
5	Апробація результатів дослідження	29.10.24	04.11.24	
6	Оформлення матеріалів до захисту БДР	05.11.24	11.11.24	

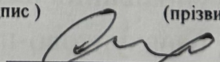
Студент


(підпис)

Власенко Д.В.

(прізвище та ініціали)

Керівник роботи


(підпис)

Сілагін О.І.

(прізвище та ініціал)

АНОТАЦІЯ

УДК 004.052

Власенко Д. О. Інформаційна технологія створення електронних підручників в системі JetIQ. Магістерська кваліфікаційна робота зі спеціальності 122 – комп'ютерні науки, освітня програма – системи штучного інтелекту. Вінниця: ВНТУ, 2024.

На укр. мові. Бібліогр.: 21 назв; рис.: 18; табл. 9.

Дана магістерська кваліфікаційна робота присвячена розробці інформаційної технології створення електронного підручника. В роботі розглянуті різноманітні підходи для створення електронних підручників. Зазначено, що для створення електронного підручника необхідно враховувати педагогічний дизайн, особливості веб-технологій, запровадження інтерактивних комунікацій між учасниками освітнього процесу. Розглянуто багатоконтурна модель освітнього електронного інформаційного середовища та її адаптація до середовища сучасного електронного підручника. Запропоновані моделі взаємодії учасників освітнього процесу в середовищі електронного підручника. Виконана реалізація інформаційної технології за допомогою мов PHP та JavaScript, бази даних SQL.

Графічна частина складається з 5 плакатів.

У економічному розділі розраховано економічну ефективність та внутрішню дохідність інвестицій, що складає 0,38.

Ключові слова: система управління навчанням, система «Електронний університет», електронний підручник, веб-технологія, інформаційна технологія створення електронного підручника, контурна модель електронного середовища.

ABSTRACT

Vlasenko D. AT. Information technology for creating electronic textbooks in the JetIQ system. Master's qualification thesis on specialty 122 - computer science, educational program - artificial intelligence systems. Vinnytsia: VNTU, 2024.

Ukrainian. Bibliography: 21 titles; Fig.: 37; table 9 .

This master's thesis is devoted to the development of information technology for the creation of an electronic textbook. Various approaches to creating electronic textbooks are considered in the work. It is noted that in order to create an electronic textbook, it is necessary to take into account the pedagogical design, the peculiarities of web technologies, and the introduction of interactive communications between the participants of the educational process. The multi-contour model of the educational electronic information environment and its adaptation to the environment of the modern electronic textbook are considered. Proposed models of interaction of participants in the educational process in the environment of an electronic textbook. Implementation of information technology using PHP and JavaScript languages, SQL database. The graphic part consists of 6 posters. In the economic section, the economic efficiency and internal rate of return on investment, which is 0.38, is calculated.

Keywords: learning management system, "Electronic University" system, electronic textbook, web technology, information technology for creating an electronic textbook, contour model of the electronic environment.

ЗМІСТ

ВСТУП	8
1.ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ СТВОРЕННЯ ЕЛЕКТРОННИХ ПІДРУЧНИКІВ.....	
1.1 Стан потреб електронних підручниках в освіті	11
1.2 Особливості інформаційних технологій та платформ для сучасних електронних підручників.....	14
1.3 Постановка задачі розробки інформаційної технології електронного підручника в системі JetIQ	26
1.4 Висновки	27
2.РОЗРОБКА МЕТОДІВ СТВОРЕННЯ ЕЛЕКТРОННИХ ПІДРУЧНИКІВ ДЛЯ ПЛАТФОРМ ДИСТАНЦІЙНОГО ТА ЗМІШАНОГО НАВЧАННЯ.....	
2.1 Відомі підходи створення та використання електронних підручників в системах управління навчанням	28
2.2 Моделі та методи створення та використання електронних підручників в системах управління навчанням	32
2.3 Висновки	59
3.РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ЕЛЕКТРОННОГО ПІДРУЧНИКА В СИСТЕМІ JETIQ.....	
3.1 Можливості запровадження інформаційної технології створення електронного підручника в системі JetIQ	60
3.2 Моделі та алгоритми програмної реалізації електронного підручника.....	62
3.3 Технології PHP та Javascript для створення електронного підручника JetIQ	65
3.4 Висновки	71

4.ТЕСТУВАННЯ МОДУЛЯ «ЕЛЕКТРОННИЙ ПІДРУЧНИК» ДЛЯ СИСТЕМИ JETIQ VNTU	
4.1 Особливості тестування освітніх веборієнтованих модулів.....	72
4.2 Тестування інформаційної технології створення електронного підручника для системи управління навчанням, підтримки методичної, наукової та управлінської діяльності JetIQ VNTU	75
4.3 Пілотний проєкт впровадження інформаційної технології створення електронного підручника в системі JetIQ	84
4.4 Висновки	88
5 ЕКОНОМІЧНА ЧАСТИНА.....	
5.1 Проведення наукового аудиту науково-дослідної роботи	89
5.2 Проведення комерційного та технологічного аудиту науково-технічної розробки	90
5.3 Розрахунок витрат на здійснення науково-дослідної роботи	91
5.4 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором.....	97
ВИСНОВКИ.....	101
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	103
ДОДАТКИ	108
Додаток А (обов’язковий) Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень.....	109
Додаток Б (обов’язковий) Лістинг програми.....	110
Додаток В (обов’язковий) ІЛЮСТРАТИВНА ЧАСТИНА	141

ВСТУП

Актуальність питання створення та використання електронних підручників не зменшується з часом, і набуває своєї значущості для реалізації гасла «Навчання через все життя», створення та реалізацію навчальних курсів дистанційного та змішаного навчання [1].

Відповідно до нормативних документів, кожен паперовий підручник повинен бути доповнений електронним. Але питання його інтерактивності, врахування особливостей окремих предметів, застосування сучасних технологій для якісного навчання та отримання результатів у вигляді визначених компетентностей залишаються дослідженими недостатньо [2].

Електронні підручники є основою для теоретичного навчання та довідниками для виконання практичних завдань. Сучасні технології їх створення дозволяють надати користувачам функціонал, який реалізує педагогічні методики, а також дозволяє покращити самостійну роботу студента в середовищі сучасного електронного підручника. Такі підручники мають SMART функції, а їх середовище складається із функціонального, комунікаційного та емоційного контурів.

Мета магістерської кваліфікаційної роботи – розробка методів, моделей та реалізації технології створення електронного підручника JetIQ.

Об’єкт дослідження – процеси розробки та запровадження інформаційної технології створення електронного підручника в системі управління освітнім процесом підтримки методичної, наукової та управлінської діяльності JetIQ VNTU.

Предмет дослідження – інформаційна технологія створення електронного підручника для платформи «Електронний університет» на основі JetIQ VNTU.

Для досягнення поставленої мети необхідно вирішити такі задачі:

виконати аналіз потреб в електронних підручниках та розробки інформаційної технології для їх створення;

проаналізувати основні поняття інформаційної технології створення електронних підручників;

виконати аналіз аналогів програмного середовища для створення електронного підручника;

виконати постановку задачі для створення інформаційної структури електронного підручника;

розробити інформаційну технологію для створення електронного підручника;

розробити моделі для реалізації інформаційної технології створення електронного підручника;

здійснити програмну реалізацію мікросервісу «Електронний підручник»;

виконати тестування інформаційної технології створення електронного підручника.

Методи дослідження та розробки для створення інформаційної технології формування електронного підручника:

- аналізу – для визначення потреб в технологіях створення електронних підручників; порівняння аналогів; вибору підходів та інструментарію для створення та реалізації інформаційної технології;
- синтезу – для створення логіки моделей інформаційної технології; архітектури застосунку; моделі інтеграції в систему JetIQ;
- моделювання – для візуалізації процесів та режимів роботи в модулі «Електронний підручник»;
- алгоритмізації та програмування – для реалізації мікросервісу «Електронний підручник».

Наукова новизна розробки:

Запропоновано інформаційну технологію створення електронного підручника на основі принципів педагогічного дизайну та контурної моделі електронного середовища, яка, на відміну від існуючих, має вбудовані інструменти підтримки користувача щодо виконання завдань для самостійної роботи, виконання мікротестів до визначених тем, формування бази знань для конкретної дисципліни і поточного користувача з подальшою можливістю використання отриманих балів в загальному оцінюванні результатів навчання студентів викладачем, що підвищує рівень засвоєння знань та ефективність самостійної роботи студента; дає можливість сформувати власну базу знань студента на основі електронного підручника.

Удосконалено інформаційну технологію створення електронного підручника для систем управління навчання, яка, на відміну від існуючих, містить елементи, які дозволяють використовувати електронний підручник як самостійний застосунок з активною взаємодією всіх агентів системи управління навчанням, а також як вбудований інструмент дистанційного та змішаного курсу з визначеної дисципліни, що сприяє покращенню результатів засвоєння та використання знань з дисципліни.

Практична цінність полягає у тестуванні модуля реалізації інформаційної технології для запровадженні електронного підручника в системі JetIQ, оновлення інструкцій та апробацію електронної книги з дисципліни «Алгоритми та структури даних».

Апробація результатів здійснена на конференціях ВНТУ, інтернет-конференціях.

Публікації – за темою досліджень опубліковано три тези доповідей [3-5]; фахова стаття в журналі категорії Б [6].

1. ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ СТВОРЕННЯ ЕЛЕКТРОННИХ ПІДРУЧНИКІВ

1.1 Стан потреб електронних підручниках в освіті

За Законом про освіту [2] кожен паперовий підручник повинен мати електронну версію, яка повинна бути в користування учнів, студентів, в бібліотеках та навчальних ресурсах з визначених предметів. Крім того, в умовах дистанційного та змішаного навчання, актуальними задачами є використання електронних підручників для самостійного навчання, підготовки до лекцій, дискусій актуальних питань.

У Положенні про електронний підручник Міністерства освіти і науки, вказано що «е-підручник – електронне навчальне видання із систематизованим викладанням навчального матеріалу, що відповідає освітній програмі, містить цифрові об'єкти різних форматів та забезпечує інтерактивну взаємодію» [1].

Електронний підручник повинен стати зручним інструментарієм для засвоєння знань. Крім того, він може містити спеціальні інструменти для реалізації відомих педагогічних концепцій.

Методи конструктивізму та коннективізму по-різному використовують електронний підручник. За методом конструктивізму, який означає формування курсу для навчання як модульних складових, електронний підручник може бути використаний як окремий модуль і використовуватись в таких ситуаціях:

- Підготовка до лекцій;
- Повторення матеріалу;
- Засвоєння матеріалу;
- Використання на лекціях та практичних.

За методом коннективізму, який, найбільше використовується у відкритих курсах, електронний підручник є одним з відкритих ресурсів, які використовуються здобувачами та викладачами для засвоєння знань, його використання може бути обумовлене визначеними темами або/і структурою курсу [7].

В будь-якому випадку, електронний підручник повинен враховувати етапи отримання та засвоєння знань.

Так, наприклад, на рис. 1.1 представлена діаграма Блума, яка дозволяє визначити рівні засвоєння знань студентами.

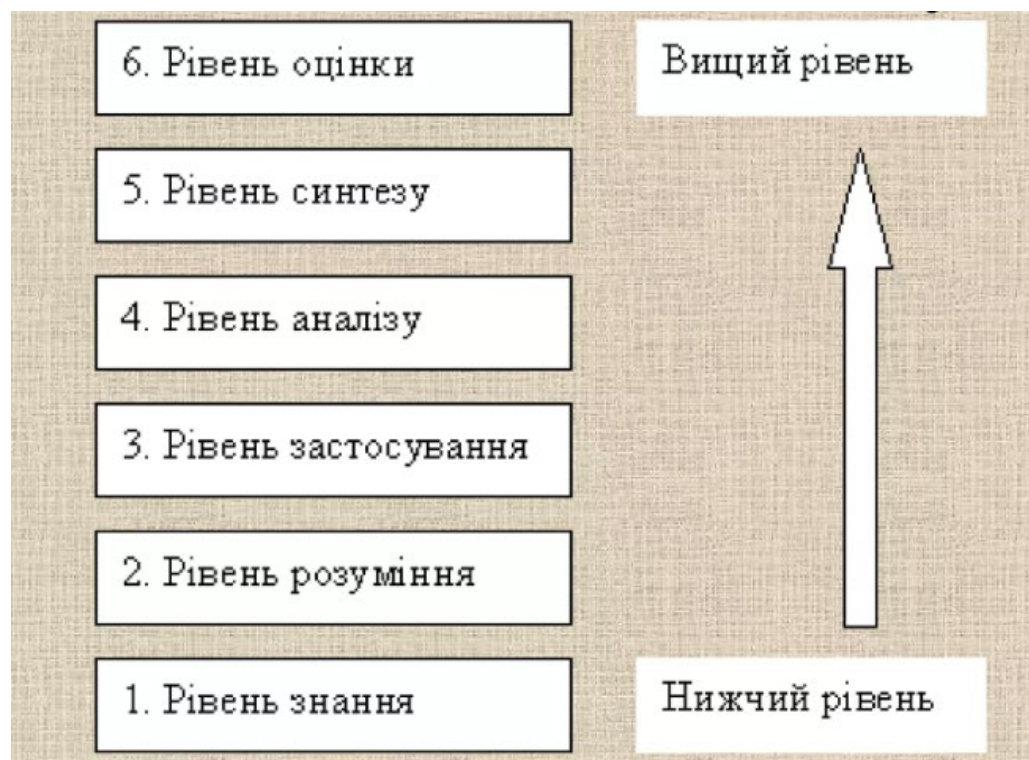


Рисунок 1.1 – Діаграма Блума

Така діаграма може бути основою для формування електронного середовища створення електронного підручника з розвинутим інструментарієм.

Потреба в електронних підручниках пов'язана також з необхідністю внесення змін в контент, запровадження синхронного та асинхронного режимів навчання, запровадження спеціального інструментарію контролю та мотивації для читачів електронного підручника.

Електронні підручники є основою дистанційних та змішаних курсів. Вони можуть бути створені як окремі застосунки, так і бути вбудованими в систему управління навчанням або в платформу дистанційних курсів. Зберігання в pdf форматі використовується для бібліотек. Поєднання з відео матеріалами – для платформ дистанційних курсів.

Електронний підручник може бути додатком до традиційного паперового підручника, а також може бути тільки в цифровому (електронному) вигляді. Останній є більш гнучким, дозволяє швидше виконувати оновлення матеріалу, але також потребує рецензування, професійного редагування тощо.

Багато видавництв пропонують електронні підручники, які можна назвати електронною версією підручників і які, в кращому випадку, мають посилання між розділами, глосарій.

Бажані характеристики е-підручника можна представити у вигляді такої низки факторів:

можливість перенесення на власні засоби); адаптованість (адаптованість до потреб учителів, учнів та їх груп);

ефективність (ефективність та продуктивність за рахунок скорочення часу і коштів, витрачених на створення та майбутні зміни й адаптації);

подовженість (адаптованість до змін технології без витрат на перероблення та перекодування), сумісність (можливість використання в різних навчальних середовищах різними інструментами), багаторазовість (використання частин е-підручника в різних контекстах [7].

Цифрові підручники можуть забезпечити навчальний контент засобами мультимедіа: відео, анімації та віртуальної реальності. Вони є адаптованими та можуть бути доповнені різноманітними технологічними інструментами.

Сучасні цифрові підручники повинні містити інтерактивні елементи, дозволяти активізувати самостійну роботу студента, повинні бути зв'язані з іншими сервісами в системах дистанційного та змішаного навчання.

1.2 Особливості інформаційних технологій та платформ для сучасних електронних підручників

Сучасні електронні підручники створюються в середовищі навчання відкритих ресурсів та окремих університетів [7-9].

Різнноманітні інструменти створення текстових файлів дозволяють формувати регульовану структуру електронного підручника, працювати з текстом, зображеннями, посиланнями, формувати глосарій, оновлювати зміст та формувати бібліографію. Такі технології використовуються в запропонованих редакторах Microsoft Office; Google Документи; Adobe, Latex тощо [10-11].

Різнноманітні сервіси – FlipSnack, Epubbud, Bookemon, Shasoft eBook та вбудовані модулі електронних книг систем управління навчання та спеціальних середовищ для освіти – Moodle, JetIQ VNTU, Microsoft Office 365, Google Workspace for Education – дозволяють працювати в режимах створення, редагування та публікації контенту.

Електронний підручник може бути створений на основі спеціальних редакторів з визначеним форматуванням, посиланнями, глосаріями та вбудованими інструментами контролю, нотаток тощо.

Відомі також спеціальні сайти електронних підручників. Одним з таких прикладів є електронні посібники ВНТУ [12].

Серед видавничих систем виділимо три основних – Adobe InDesign, QuarkXPress, InDesign, Scribus.

Adobe InDesign – це професійний графічний редактор, який широко використовується для створення різноманітних видавничих продуктів, включаючи електронні книги, журнали, брошури, плакати та багато іншого. Його потужні інструменти для верстки, дизайну та підготовки до друку роблять його незамінним помічником для дизайнерів, видавців та інших фахівців, які працюють з друкованою продукцією.

InDesign дозволяє створювати складні макети з використанням колон, таблиць, текстових блоків та інших елементів. Це ідеально підходить для структурування навчального матеріалу.

Програма підтримує експорт в різні формати, включаючи PDF, EPUB, HTML, що дозволяє створювати електронні книги, які можна читати на різних пристроях.

InDesign тісно інтегрується з іншими продуктами Adobe Creative Cloud, такими як Photoshop та Illustrator, що дозволяє легко імпортувати зображення та векторну графіку.

Програма має потужні інструменти для автоматизації рутинних завдань, таких як створення таблиць змісту, нумерація сторінок тощо. InDesign пропонує велику кількість інструментів для оформлення тексту, зображень та інших елементів, що дозволяє створювати візуально привабливі підручники.

Основні функції Adobe InDesign для створення електронних підручників полягають у створенні макетів як багатосторінкових документів з різними розмірами сторінок та орієнтаціями; різноманітному форматуванню, роботі з такими структурами як таблиці, списки, посилання на цитати та примітки; робота з різними зображеннями, інтерактивними елементами, можливостями конвертації готових елементів в PDF, EPUB, HTML та інші формати.

QuarkXPress – це потужний графічний редактор, який вже багато років конкурує з Adobe InDesign. Він призначений для професійної верстки та дизайну різноманітних видавничих продуктів, включаючи книги, журнали, брошури та веб-сайти. QuarkXPress має тривалу історію і є одним з перших професійних інструментів для верстки. Багато дизайнерів та видавців звикли працювати саме в цьому середовищі. QuarkXPress відомий своєю здатністю створювати високоякісну поліграфічну продукцію.

Програма пропонує безліч інструментів для точного контролю над зовнішнім виглядом документів. QuarkXPress легко інтегрується з іншими програмами для обробки зображень та текстовими редакторами.

Scribus – це потужний, відкритий і безкоштовний програмний пакет для верстки, який надає професійні інструменти для створення різноманітних видавничих продуктів. Scribus розповсюджується під ліцензією GNU GPL, що означає, що ви можете вільно використовувати, вивчати, модифікувати і розповсюджувати його. Програма пропонує широкий спектр інструментів для верстки, роботи з текстом, зображеннями та кольорами, підтримує різноманітні стандарти, включаючи PDF, EPS, SVG та інші.

Програма дозволяє створювати складні макети з використанням колон, таблиць, текстових блоків та інших елементів.

Scribus підтримує багато мов, включаючи українську, що робить його зручним для використання в україномовних середовищах.

Розглянемо також відомі редактори електронних книг.

Sigil – це потужний і безкоштовний редактор електронних книг з відкритим кодом, спеціально розроблений для створення та редагування файлів у форматі EPUB. Цей формат є одним з найпопулярніших для електронних книг, оскільки його підтримує більшість сучасних читалок.

Sigil зосереджений виключно на форматі EPUB, що дозволяє йому

пропонувати глибоку інтеграцію з його особливостями.

Інтерфейс Sigil інтуїтивно зрозумілий, що дозволяє швидко освоїти програму навіть новачкам.

Sigil дозволяє редагувати як структуру книги (глави, розділи), так і її вміст (текст, зображення, стилі).

Sigil доступний для безкоштовного завантаження та використання і має відкритий код.

Основні функції Sigil полягають у можливості структурвання книги, редагування та форматування тексту, додавання списків, таблиць, посилань. Також можна працювати зі стилями, зображеннями, метаданими, є інструменти щодо відповідності форматів.

Sigil – це відмінний вибір для тих, хто хоче створювати електронні книги без зайвих витрат і зусиль. Він простий у використанні, потужний і безкоштовний.

Хоча Calibre зазвичай асоціюється з управлінням бібліотекою електронних книг (перетворення форматів, організація, синхронізація з пристроями), він також має потужні інструменти для створення електронних книг. Calibre підтримує велику кількість форматів електронних книг, що дозволяє конвертувати ваші документи в потрібний формат (EPUB, MOBI, PDF тощо).

Calibre має вбудований редактор EPUB, який дозволяє змінювати структуру книги, додавати стилі, зображення та інші елементи.

Calibre дозволяє автоматизувати багато рутинних операцій, таких як перетворення форматів, додавання метаданих, сортування бібліотеки.

Хоча Calibre має функціонал для створення електронних книг, він все ж таки є менеджером бібліотеки, а не спеціалізованим редактором. Деякі складні макети можуть вимагати використання більш спеціалізованих інструментів, таких як Sigil.

Редактор EPUB в Calibre не настільки гнучкий, як спеціалізовані редактори, такі як Sigil.

Calibre – це універсальний інструмент, який дозволяє не тільки керувати бібліотекою електронних книг, але й створювати їх. Він простий у використанні і підходить як для новачків, так і для досвідчених користувачів. Однак, для створення складних електронних книг з багатьма інтерактивними елементами, може знадобитися використання більш спеціалізованих редакторів.

Adobe Digital Editions (ADE) – це програма, перш за все, призначена для читання електронних книг, особливо тих, що захищені DRM (Digital Rights Management). Хоча вона може відкривати і редагувати деякі аспекти EPUB-файлів, її основна функціональність не орієнтована на створення нових електронних книг з нуля.

ADE дозволяє змінювати деякі аспекти форматування тексту, додавати закладки та примітки, але не надає широких можливостей для створення складних макетів, таблиць, інтерактивних елементів тощо.

ADE оптимізований для читання, а не для створення. Це означає, що багато інструментів, які потрібні для професійного створення електронних книг, відсутні або обмежені.

Комплекс тісно інтегрований з системою управління цифровими правами Adobe, що може ускладнювати створення книг.

В системах управління навчанням можливість створення електронних книг та інтерактивних курсів мають майже всі платформи.

Moodle, Canvas та Blackboard – це популярні платформи для онлайн-навчання, які дозволяють створювати та керувати навчальними курсами. Хоча їх основна функція – це організація навчального процесу, вони також надають можливості для створення та розміщення електронних книг.

Створення електронних книг на платформах систем управління навчанням

дозволяє централізувати навчальні матеріали. Усі навчальні матеріали, в тому числі електронні книги, зберігаються в одному місці, що зручно для студентів і викладачів.

Багато платформ дозволяють додавати до електронних книг різноманітні інтерактивні елементи (відео, аудіо, тести тощо).

Викладач може легко керувати доступом до електронних книг, встановлюючи терміни доступу, обмежуючи кількість завантажень тощо.

Платформи часто інтегруються з іншими інструментами, такими як системи управління завданнями, форуми, що дозволяє створювати більш комплексні навчальні курси.

Розглянемо особливості кожної з платформ.

Moodle – гнучка платформа з великою кількістю модулів для створення різноманітних навчальних матеріалів. Підтримує широкий спектр форматів файлів.

Canvas має інтуїтивно зрозумілий інтерфейс, зручний для створення сучасних онлайн-курсів, а також вбудований комплекс інструментів для створення інтерактивних книг.

Blackboard – потужна платформа з широким спектром функцій, включаючи інструменти для співпраці та оцінювання студентів. Підтримує різноманітні формати файлів.

Багато платформ пропонують готові шаблони для створення електронних книг, що дозволяє швидко створити професійно оформлений документ. Важливим є оптимізація зображень, а також тестування книги та її доступність.

Хоча інструменти для презентацій, такі як Microsoft PowerPoint, Google Slides або Canva, не є спеціалізованими для створення електронних книг, вони можуть бути ефективним інструментом для швидкого створення простих електронних книг, особливо якщо вже є готовий матеріал у вигляді презентації.

Презентаційні інструменти дозволяють створювати яскраві та візуально привабливі електронні книги з використанням різних шаблонів, шрифтів, зображень та інших елементів дизайну.

Деякі інструменти для презентацій дозволяють додавати до слайдів анімацію, відео, аудіо та гіперпосилання, що робить електронну книгу більш інтерактивною.

Розглянемо аналоги технології створення електронного підручника. Розглянемо аналоги для побудови електронних підручників.

На рис. 1.2 представлено вигляд сторінки створення електронного підручника в середовищі Canva [13].

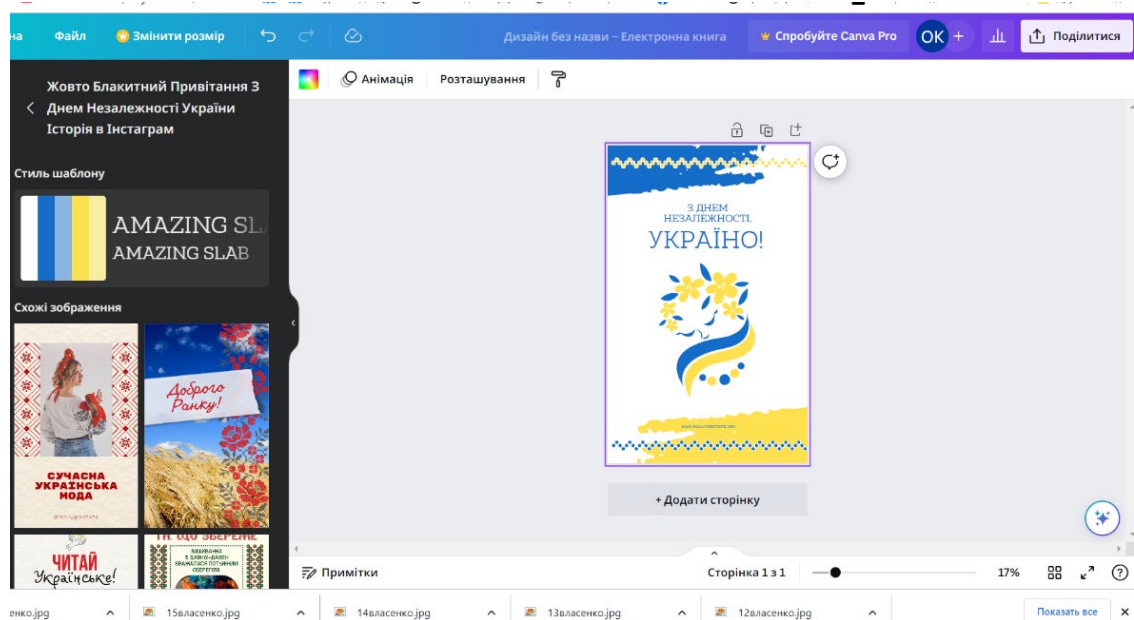


Рисунок 1.2 – Середовище створення електронного підручника Canva

Таке середовище дозволяє сформувати єдиний стиль та об'єднати текст, таблиці, презентації, тести, зображення, анімацію.

Отриманий електронний підручник може мати різноманітні інтерактивні елементи, які представлені на рис. 1.3.

Недоліками запропонованого інструментарію є платність підписки та достатньо складний інтерфейс, необхідність вивчення платформи, неможливість використання власно створених програмних модулів. Але таке середовище дозволяє достатньо легко вбудувати різноманітні технології з відео матеріалами, складними зображеннями, роз'ясненнями тощо.

Для вищої школи електронні підручники є більш гнучкими інструментами, для них визначено більше вимог, серед яких адаптація до активної самостійної роботи студентів.

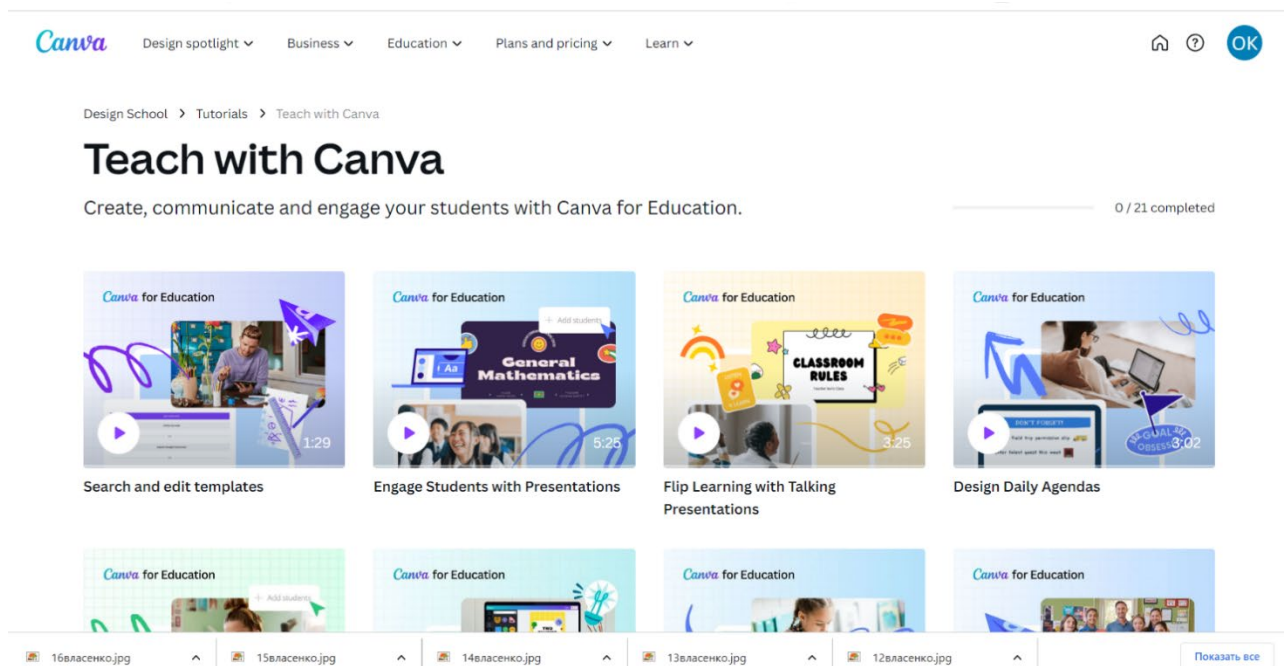


Рисунок 1.3 – Canva Education – Редактор створення електронного підручника

Електронні підручники можна також створювати в таких середовищах як Adobe та Latex. Але вони також потребують багато часу для вивчення функціоналу та є платними в більшій частині функціоналу

В таблиці 1.1 представлені особливості використання платформ для створення електронних підручників. Найпростішими сервісами створення електронних підручників в спеціальних освітніх середовищах є Microsoft Office 365 та Google Workspace for Education. Такі редактори дозволяють використовувати перехресні посилання, формувати бібліографію, зберігати як веб-сторінку тощо.

Таблиця 1.1 – Особливості технології створення електронних підручників на різних платформах управління навчанням та додаткових сервісах

Платформа	Особливості технологій
Microsoft Office 365	Можливість записування відео, використання інструментарію Teams, технологій програм, що входять в склад платформи
Google Workspace for Education – електронна книга на диску, електронна книга в Classroom	Можливість використовувати форми як тести, календар для обмеження часу, спеціальне середовище для дистанційного та змішаного навчання
JetIQ Електронна книга	Посилання на глосарій та тести в модулі TestIQ Можливості встановлення дати і часу проходження тестів
Moodle	Посилання на глосарій та тести, робота з підручником в електронному курсі Елементи геймфікації
Adobe	Технології для використання анімації, посилань
Canva	
JetIQ + Електронна підручник з smart елементами	Реалізація технології за контурною моделлю, використання інтерактивних елементів, можливості роботи з агентами системи управління навчанням

Спеціальні інструменти для створення електронного підручника дозволяють працювати із зображеннями, схемами, а також формувати індивідуальний план роботи з електронними ресурсами.

В таблиці 1.2 представлені результати порівняння інформаційних технологій створення електронного підручника.

Таблиця 1.2 – Порівняння інформаційних технологій для створення електронного підручника

Аналоги	Вбудована в систему управління навчанням	Може бути використана як окремий електронний ресурс	Оцінювання прогресу в навчанні	Вбудовані тести	Емоційна підтримка	Гейміфікація	Планування часової освітньої траєкторії	Загальний бал
Електронна книга Microsoft Office 365	+		-	форми	-	-	календар	3
Сайт електронна книга Google Workspace for Education – електронний підручник в ClassRoom	+	+	-	форми	-	-	календар	3
JetIQ Електронна книга	+	+	-	+ СУ	-	+ СУ	+ СУ	3,5
Електронний підручник Moodle	+	+	+ ДК	+ ДК	+ ДК	+ ДК	+ ДК	4,5
Електронна книга Adobe	спец.	+	-	-	-	+	-	2,5
Електронна книга Canva	спец.	+	-	-	-	+	-	2,5
JetIQ + Електронний підручник	+	+	+ Т. вивч.	+	+	+ СУ	+ СУ календар	4,7

Таке порівняння дозволяє виконати аналіз вимог до електронного підручника за контурною моделлю, яка реалізована у всій інформаційній екосистемі JetIQ та оцінити очікувані результати від розробки та впровадження інформаційної технології створення електронного підручника.

Загальновідоме визначення інформаційної технології визначає що це сукупність методів і засобів, що використовуються для збору, зберігання, обробки і поширення інформації. Таке визначення можна доповнити процесами генерації та обміну інформації. Інформаційна технологія для створення електронного підручника повинна враховувати також наявність контролюючих елементів, емоційної підтримки, гейміфікації, персоналізованих варіантів використання електронних ресурсів.

На рис. 1.4 представлено вікно створення електронного підручника на платформі Moodle.

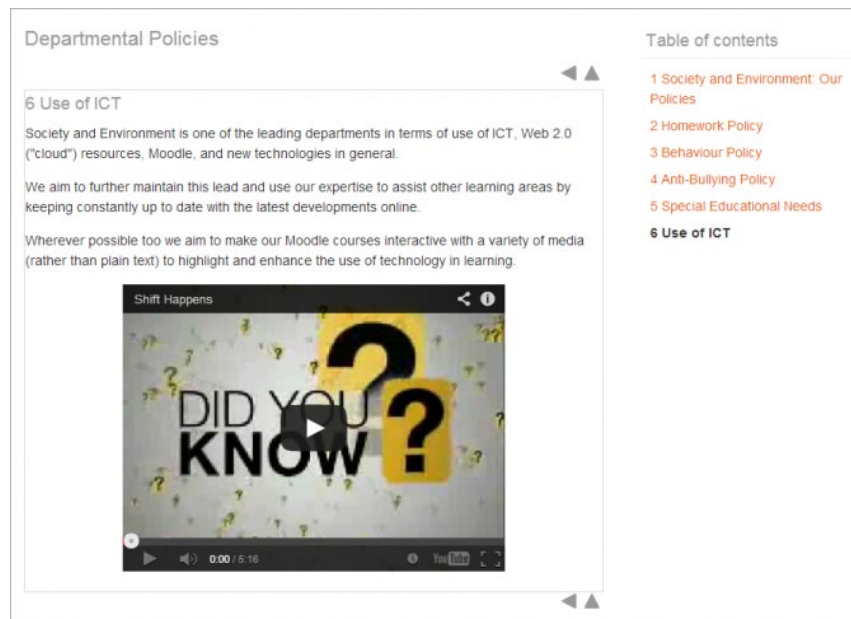


Рисунок 1.4 – Вікно створення електронного підручника Moodle

Аналогічні показники як у платформи Moodle мають відомі платформи дистанційного навчання Coursera, EdEra [14-16] тощо. Але існує думка, що електронні підручники для дистанційних курсів не потрібні. Буде достатньо конспектів лекцій та презентацій. Крім того, стверджується, що дистанційний курс здійснює оцінювання, а електронний підручник – ні. Це дійсно так в

традиційних електронних підручниках. Але сучасна технологія може бути створена так, щоб електронний підручник мав мікроінструменти дистанційного курсу.

Зазвичай підручники не потрібні для проходження курсу та отримання сертифікатів. Відеофрагментів, конспектів і завдань буде достатньо. Однак якщо здобувачам необхідні фундаментальні знання, їх засвоєння, то електронний підручник є найкращим вибором.

На рис. 1.5 представлено базовий прототип створення електронного підручника в системі JetIQ.

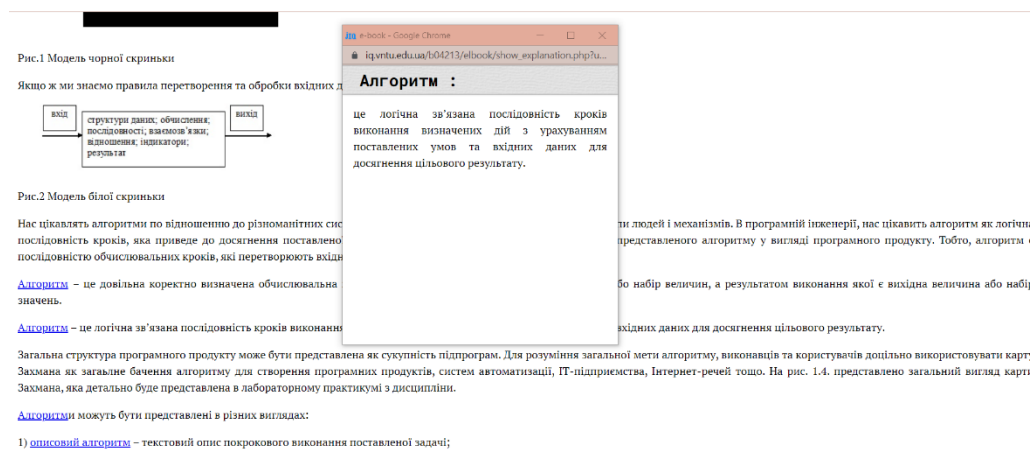


Рисунок 1.5 – Електронна книга JetIQ Jet Book. Глосарій

За результатами оцінювання найкращі реалізації – електронна книга Moodle та розробка, яка розпочата для системи JetIQ.

Багато з авторів визначає, що електронний підручник повинен бути відкритим, самостійним, середовищем, де можна вносити зміни [17].

Але занадто велика відкритість електронного підручника по-перше порушує авторські права, а по-друге не завжди сприяє достовірності фактів, які представлені в підручнику. Прикладом є Вікіпедія та непідтверджені статті в ній.

Крім того, важливо також враховувати педагогічні принципи викладання та навчання.

Можна розділити електронні підручники на статичні – створені в форматах без змін та у відповідності до паперових аналогів; з елементами мультімедіа; інтерактивні.

Універсальним електронним підручником є інтелектуальний підручник, який використовується не тільки як підручник, а ще як мікро курс, основа для створення контенту уроків, баз знань з дисципліни.

1.3 Постановка задачі розробки інформаційної технології електронного підручника в системі JetIQ

Інформаційна технологія створення електронного підручника в системі JetIQ VNTU – це сукупність методів та моделей для створення та використання ресурсів електронного підручника разом з іншими веб-сервісами та агентами системи управління навчання, підтримки методичної, наукової та управлінської діяльності. Для розробки інформаційної технології створення електронного підручника необхідно виконати такі завдання:

1. Визначити особливості та педагогічні методики інформаційної технології створення електронного підручника.
2. Розробити загальні методи та моделі інформаційної технології та вимоги до її реалізації.
3. Запропонувати нову технологію створення електронного підручника з використанням сучасних методів педагогічного дизайну, контурних моделей та інтерактивних елементів.
4. Удосконалити те
5. Визначити технології для реалізації запропонованих методів.

6. Обґрунтувати засоби для реалізації інформаційної технології створення електронного підручника.
7. Розробити моделі агентної взаємодії в середовищі електронного підручника
8. Проаналізувати можливості інтеграції створеного електронного підручника в систему JetIQ.
9. Обґрунтовано вибрати інструменти для реалізації інформаційної технології.
10. Виконати реалізацію інформаційної технології створення електронного підручника для системи JetIQ.
11. Розробити план пілотного проєкту запровадження електронного підручника на прикладі дисципліни «Алгоритми та структури даних».

1.4 Висновки

Отже, в першому розділі виконано аналіз відомих інформаційних технологій, інструментів та платформ створення електронних підручників, проаналізована потреба в таких технологіях, виконано постановку задачі для створення інформаційної технології для системи управління навчанням, підтримки методичної, наукової та управлінської діяльності JetIQ VNTU.

2. РОЗРОБКА МЕТОДІВ СТВОРЕННЯ ЕЛЕКТРОННИХ ПІДРУЧНИКІВ ДЛЯ ПЛАТФОРМ ДИСТАНЦІЙНОГО ТА ЗМІШАНОГО НАВЧАННЯ

2.1 Відомі підходи створення та використання електронних підручників в системах управління навчанням

Електронні підручники в системах управління навчанням використовуються як складові курсу дистанційного та змішаного навчання. Електронний підручник також може бути використаний для індивідуального самостійного навчання. Особливо це важливо для студентів, які мають індивідуальний графік.

Така технологія є інструментом, який можна використовувати в синхронном та асинхронному режимах.

Електронні підручники стали невід'ємною частиною сучасного освітнього процесу, особливо в рамках систем управління навчанням (LMS). Вони пропонують безліч переваг перед традиційними друкованими виданнями: інтерактивність, мультимедійність, доступність та можливість адаптації до індивідуальних потреб кожного студента.

Модульний підхід є одним зі зручних підходів, коли сам підручник ділиться на невеликі логічні модулі, кожен з яких присвячений окремій темі. Кожен модуль містить теоретичний матеріал, завдання для самоперевірки, мультимедійні елементи та посилання на додаткові ресурси. Цей підхід дозволяє студентам легко орієнтуватися в матеріалі та вивчати його в зручному темпі.

Проектний підхід полягає в тому, що електронний підручник будується навколо проєктів, які студенти мають виконати. Теоретичний матеріал подається в контексті проєкту, а завдання спрямовані на його реалізацію. Цей підхід

розвиває у студентів навички самостійної роботи, творчого мислення та вирішення проблем.

На рис. 2.1 представлено модель використання електронного підручника в навчальному процесі на прикладі створення електронного підручника в LaTeX.



Рисунок 2.1 – Модель використання електронного підручника на основі LaTeX

Підхід з використанням гейміфікації дозволяє вводити елементи гри в процес навчання. Це бали, рівні, змагання тощо. Це підвищує мотивацію

студентів та робить навчання більш цікавим та захоплюючим. Ігровий підхід може бути використаний для симуляції роботи алгоритмів (предмет «Алгоритми та структури даних», для дослідження та удосконалення різноманітних електронних систем (наприклад, предмет «Банківська справа».

Інтерактивні елементи дозволяють включати в підручник низку різних інтерактивних елементів: тестів, симуляцій, вікторин, відеороликів. Це дозволяє студентам активно взаємодіяти з матеріалом та закріплювати знання. Такий підхід формує мотиваційно-емоційний контур і дозволяє підвищувати мотивацію студентів. Загальна системи гейміфікації системи управління навчанням повинна бути зв'язана з системою гейміфікації в електронному підручнику.

Підручник як веб-сервіс пропонує різні варіанти використання відповідно до рівня знань та темпу навчання кожного студента. Це дозволяє створити більш персоналізований навчальний досвід. Для цього використовують вхідний замір знань і варіанти завдань.

Електронні підручники інтегруються в LMS за допомогою спеціальних технологій, що дозволяє інтегрувати отримані результати в середовищі електронного підручника, відстежувати прогрес студентів, надавати зворотний зв'язок та автоматизувати деякі процеси навчання.

Електронні підручники можуть бути основою для створення онлайн-курсів, які студенти можуть проходити самостійно або з підтримкою викладача.

LMS дозволяють використовувати різноманітні формати контенту в електронних підручниках: текст, зображення, відео, аудіо, 3D-моделі тощо.

LMS надають можливості для спільної роботи студентів над проєктами, обговорення матеріалу на форумах та в чатах.

Всі електронні підручники повинні забезпечувати такі властивості як доступність, інтерактивність та мультимедійність.

Студенти можуть отримати доступ до підручника в будь-який час і в будь-якому місці, маючи підключення до інтернету. Електронні підручники дозволяють студентам активно взаємодіяти з матеріалом, що підвищує ефективність навчання. Використання різних форматів контенту робить навчання більш цікавим та наочним.

Електронні підручники є потужним інструментом для сучасного навчання. Вони дозволяють створити більш інтерактивні, персоналізовані та ефективні навчальні середовища.

Створення та використання електронних підручників стало невід'ємною частиною сучасного освітнього процесу. Існує безліч моделей та підходів, які визначають, як саме створюються і використовуються такі підручники.

Методи та моделі створення електронних підручників можуть бути лінійними, розгалуженими, адаптивними, колаборативними, гібридними.

В лінійній моделі матеріал подається послідовно, від простого до складного. Студент рухається вперед за фіксованим маршрутом. Така модель є в традиційних електронних книгах, відеолекціях, навігаторах JetIQ.

Колоборативна модель передбачає роботу над підручником всіх зацікавлених осіб – студентів, викладача, роботодавців, а також використання відкритих освітніх ресурсів. В кінці курсу формується новий відкритий освітній ресурс для навчання.

Гібридний метод створення підручника передбачає використання всіх перелічених моделей з адаптацією під дисципліну, рівень студентів, можливості залучення роботодавців, наявність зовнішніх освітніх ресурсів. Крім того, використовують методи педагогічного дизайну, перевернутого класу, фасилітації тощо.

Педагогічний дизайн для створення електронного підручника передбачає використання методик навчання та формування методики самостійної роботи студента в середовищі електронного підручника.

Перевернутий клас передбачає виконання практичних задач до періоду глибокого вивчення дисципліни.

Фасилітація – передбачає формування карт допомоги, стратегій вивчення дисципліни та реалізації практичних завдань за допомогою спеціальних інструментів.

Крім того, електронний підручник є основою для персоналізації навчання за допомогою вбудованих нотаток для підручника, формування закладок, обговорення дискусійних питань тощо.

2.2 Моделі та методи створення та використання електронних підручників в системах управління навчанням

Крім загальних моделей, представлених в п. 2.1 та в першому розділі магістерської роботи, необхідно звернути увагу на моделі користувачів для створення та використання електронних підручників. Так, на рис. 2.2 представлена загальна модель електронного підручника для навчальних закладів [10]. Така загальна модель може бути використана в різних навчальних закладах і передбачає роботу таких агентів як викладач, студент, адміністратор, а також агенти-веб-сервіси – електронний підручник, база даних. В єдиній сукупності та при умові формування персональних нотаток і закладок, виконання персональних тестів, формується персональна база знань з дисципліни конкретного студента. А викладач має персональну базу знань з дисципліни з варіантами тестів та планів вивчення для окремих груп студентів та результати обговорення у форумі.

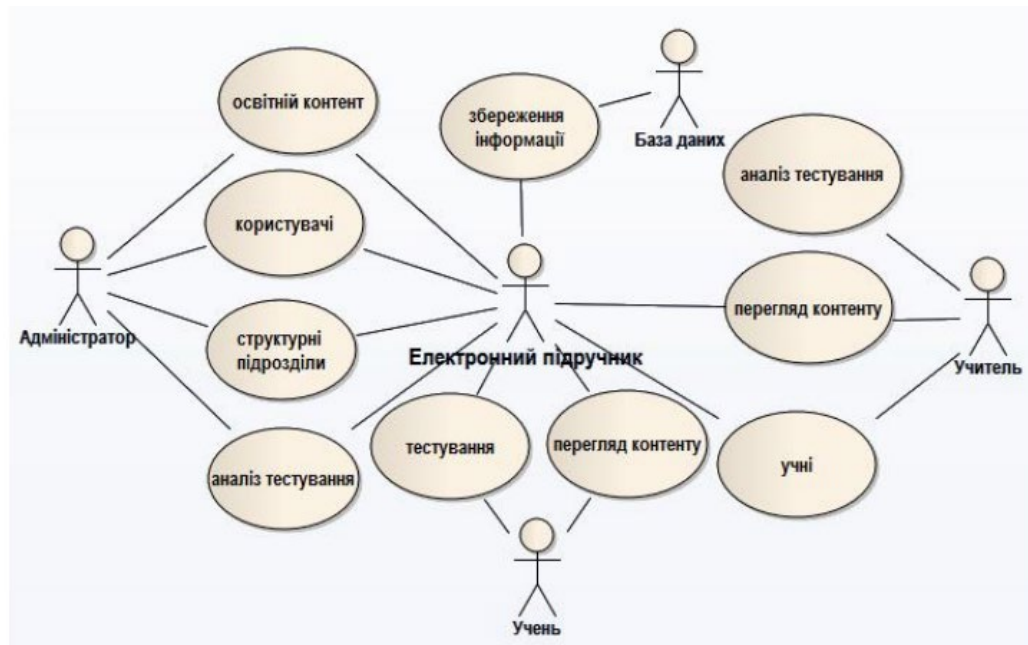


Рисунок 2.2 – Загальна модель електронного підручника [10]

Педагогічний дизайн електронного підручника – це процес створення навчального матеріалу, який враховує не тільки зміст, але й психологічні, когнітивні та соціальні аспекти навчання. Він спрямований на створення ефективного навчального середовища, в якому студент і викладач є активними.

Представлена модель дозволяє виконати адаптацію до різного рівня знань студента та надати йому зворотній зв'язок щодо його прогресу з виконання завдань.

Якщо електронний підручник входить в комплекс навчання студента, то доцільно використовувати ті ж самі інструменти, що і в загальній системі управління навчанням.

Так, при проходженні тестів електронного підручника, необхідно формувати відомість аналогічно тестам системи JetIQ з можливістю перенесення оцінок в електронний журнал. Такий сервіс є одним з мотиваційних

елементів для студентів. Якщо викладач бачить результати активності студента та його статуси, він має можливість врахувати цю активність додатковими балами.

Дослідження були виконані в середовищі системи JetIQ VNTU, на основі модуля електронної книги (Рисунок 2.3).

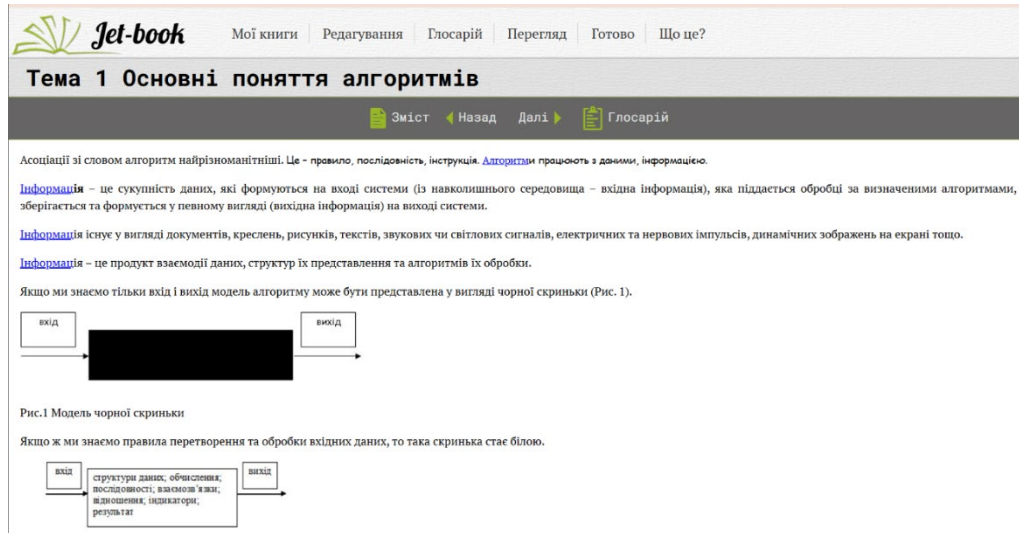


Рисунок 2.3 – Версія для удосконалення електронної книги в системі JetIQ VNTU

Сам електронний підручник має основний формат – текст, але завдяки посиланням на внутрішні та зовнішні мультимедійні ресурси, використання тестів, зображень, анімації такий електронний підручник буде сформований як комплекс різноманітних форм інформації для сприйняття та розуміння студентом.

Математична модель створення інформаційної технології для формування електронного підручника може бути подана на основі теорії множин та концепції Tree-Net [25].

Теорія множин дозволяє подати процедури формалізації та фрагментації контенту, визначення компонент електронного підручника як ієрархічно-мережев модель вебконтенту.

$$K=\{k_1,...,k_n\}, \quad (2.1)$$

де K – контент електронного підручника

Ієрархія може бути представлена у вигляді графу, де є відповідність кожного елементу – підручник – розділи – параграфи – пункти – посилання.

$$I: \{K_1 \rightarrow K_2\}, \quad (2.2)$$

де I – ієрархія контенту для подання.

План вивчення може бути представлений за темами, а може бути змінений за результатами вхідного заміру знань, з пропуском, або узагальненням окремих тем.

Множина

$$T=\{t_1,...,t_n\}, \quad (2.3)$$

де T – ієрархія контенту для вивчення.

Тематичні групи структуруються відповідно до плану і можуть бути представлені як:

$$I: T_1 \rightarrow T_2, \quad (2.4)$$

Кожен елемент контенту пов'язаний з різними темами, тестами, контентом глосарію.

$$TK: K_1 \rightarrow T_2, \quad (2.5)$$

Понятійно-тезисна модель (ПТМ) дозволяє сформулювати різні тези (як окремі фрагменти тексту, елементи контенту (зображення, анімація, посилання, глосарій, тести, поняття, терміни).

Для реалізації моделі формуються зв'язки між такими множинами:

Множина понять $P=\{p_1,\dots,p_n\}$.

Множина термінів $Tr=\{p_1,\dots,p_n\}$.

Множина тез $Tz=\{tz_1,\dots,tz_n\}$.

Множина елементів контенту $K=\{k_1,\dots,k_n\}$.

Множина тем: $T=\{t_1,\dots,t_n\}$

Множина елементів глосарію $G=\{g_1,\dots,g_n\}$.

Множина тестів для перевірки $Q=\{Q_1\dots Q_n\}$.

Множина часових параметрів виконання контролю $D=\{D_1\dots D_n\}$.

Модель дозволяє структурувати всі визначені множини і працювати з ними відповідно до плану та індивідуальної траєкторії студента. Особливо важливим є робота студента самостійно (Рис. 2.3).

Викладач формує контент відповідно до очікуваних результатів, отримання компетентностей студентами. Але навчальний контент електронного підручника поділяється на тези опису досліджуваного об'єкту, основні поняття і терміни (глосарій), підтримка розуміння за допомогою тестів, зображень та анімації, зовнішніх посилань на корисні електронні ресурси. Але такі ресурси повинні бути або створені авторами електронного підручника, або бути перевіреними, з надійних відкритих джерел.

Крім того, важливим є також модуль моніторингу активності студента в середовищі електронного підручника.

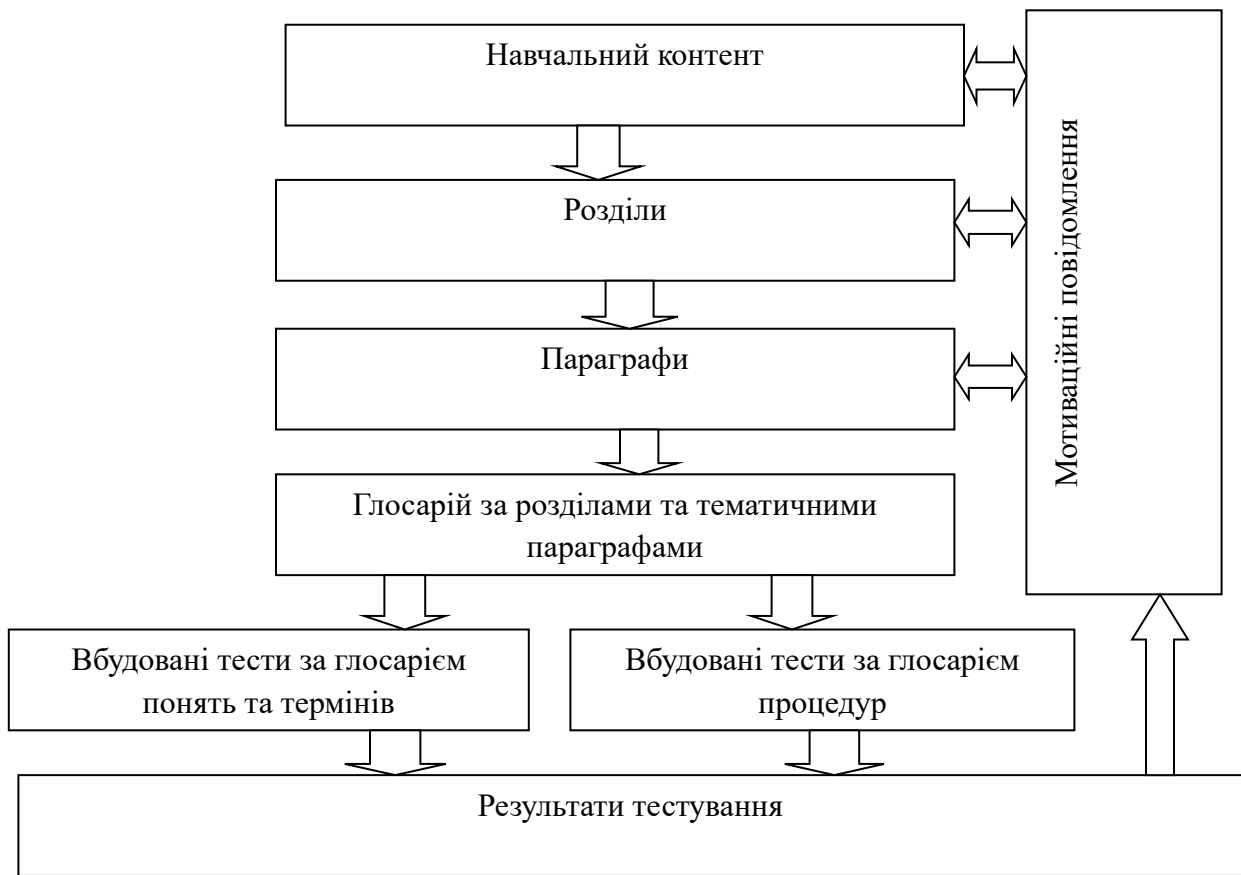


Рисунок 2.4 – Структурна модель електронного підручника Jetbook+

Розумні елементи в середовищі для створення та використання електронних книг дозволяють створювати мікротести для кожного окремого підрозділу, параграфа, нагадувати користувачам про завдання та дати виконання контрольних робіт, містити посилання про необхідність подальшого вивчення теми.

Запропоновану модель можна представити в стислому вигляді. Вона поєднує три профілі.

Функції (викладач – створює шаблони, структуру книги, завантажує контент, створює тести, формує плани на основі дат виконання завдань), (студент

– завантажує книгу, створює нотатки та закладки, виконує тести та завдання за планом, бере участь у форумі).

Комуникатор (нагадування, підтримка, надсилання повідомлень).

Контролер-мотиватор (формує інформацію для виконання завдань, слідкує за виконанням, формує повідомлення щодо виконання).

Дослідження системи JetIQ VNTU як багатоконтурної модульної системи дозволяють зробити висновок, що освітнє електронне інформаційне середовище різного масштабу повинно мати основні контури – функціональний, комунікаційний, управлінський та емоційно-мотиваційний та безпековий контури [2]. Аналогічно до запропонованої контурної моделі для цілісної інформаційної екосистеми JetIQ VNTU, була запропонована система контурів для середовища електронної книги. На рисунку 2.3 подано схему взаємодії (колесо контурів).

1. Перше коло – ядро – функціональний контур – для викладача – створення розділів електронної книги, плану використання відповідно до визначеної методики, формування бібліотеки завдань та термінів виконання, нагадувань щодо виконання, повідомлень для системи моніторингу (СМ).

2. Друге коло – управління, яке охоплює функціонал, має зв'язки впливу та зворотного зв'язку відповідно до дій користувача – відкриття книги, створення нотаток, робота з книгою, створення власної версії книги, формування та надсилання відповідей на вбудовані тести, перехід у форум та систему повідомлень для обговорення книги та уточнення питань з викладачем або/і з групою студентів.

3. Обидва кола пронизують комунікаційні зв'язки (зображено у вигляді наскрізних стрілок), на перетині яких утворюються емоційні кластери (мотиваційний контур).

4. Окремий контур складають елементи безпеки – як на серверному та мережевому рівні, так і на рівні роботи системи і, зокрема, електронної книги.

Аналіз наведеної схеми та опису свідчить про те, що всі вони працюють взаємозв'язано та можуть розглядатись окремо для визначення окремих процесів, функцій роботи з електронною книгою, реалізації методики самостійної та аудиторної або дистанційної (з викладачем та групою) робіт здобувача, а також запровадження елементів комунікації, мотивації, гейміфікації для здобувачів і безпеки для забезпечення доброчесності та авторського права.

Використовуючи теорію множин і гіпотезу контурів інформаційної екосистеми JetIQ VNTU, можна сформувані математичні моделі контурів освітнього електронного інформаційного середовища електронного підручника, на основі шаблону, за яким викладачі можуть створювати різноманітні електронні навчальні ресурси – методичні вказівки або лабораторні практикуми, навчальні посібники та підручники. Запровадження власної методики викладача може бути здійснено за допомогою спеціальних інструментів – плану виконання завдань відповідно до контенту, виконання вбудованих мікротестів; робота з симуляторами тощо [10]

Загальна модель контурів K_elbook може бути подано як об'єднання множин функціональних дій викладача (Рис. 2.5) –

K_f_teach (функції вивчення інструкції роботи з модулем «Електронний підручник» $Learn_instr_teach$;

створення Cr ;

наповнення $Content$;

план управлінських дій $Planm$;

бібліотека повідомлень для системи моніторингу дій студента $Lsms$;

вбудовані тести з визначенням оцінювання $Tests, Val$;

завдання $Tasks$; список зовнішніх посилань $Exterlinks$.

функціональних дій студента –

K_f_st (функції вивчення інструкції роботи з модулем «Електронний підручник» Learn_instr_st;

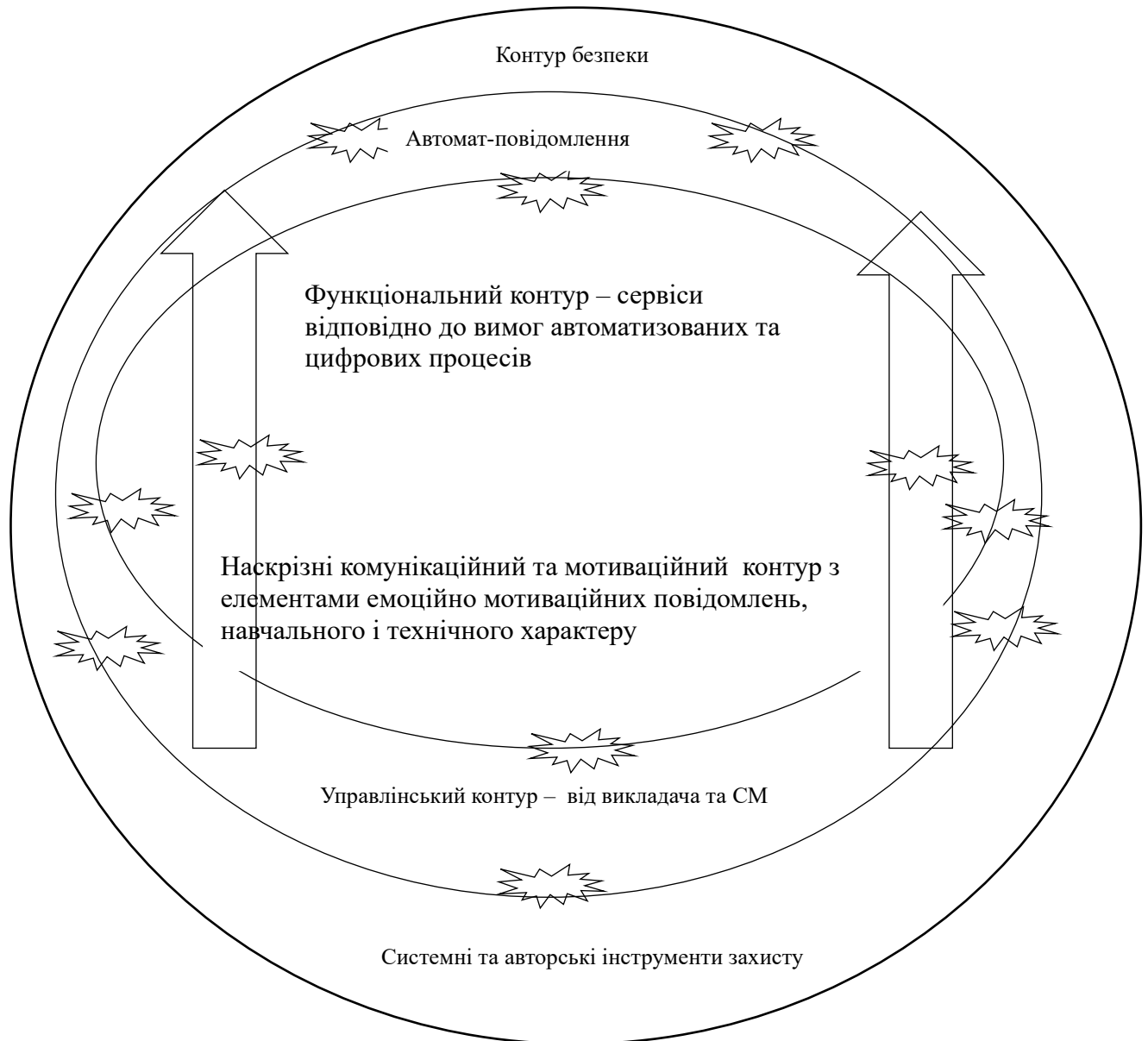


Рисунок 2.5 – Контурна модель середовища «Електронний підручник»

Read_st – читання електронних ресурсів,

Not_st – закладок, нотаток,
 Tasks_St – виконання завдань,
 Tests_st - вбудованих тестів,
 SMS_st – формування повідомлень та переходів до форуму обговорення –
 Forum_st);
 Роботи системи моніторингу –
 K_f_SM функцій системи моніторингу (відстеження дій студента Track_St;
 формування автоповідомлень Auto-SMS,
 зв'язок з системою гейміфікації Gam);
 Інтеграції системи моніторингу –
 K_f_SI функцій системи інтеграції електронного підручника з іншими
 модулями інформаційної екосистеми JetIQ VNTU (інтеграція з навігатором
 дисципліни Integer_navigator;
 формування відомостей результатів тестів; можливості напіваавтоматичного
 занесення оцінок в електронний журнал Integer_test_results;
 зв'язок з системою повідомлень викладачу Integer_communication_teacher;
 надсилання файлів Integer_sms; Integer_sendfile; Integer_forum);
 Система безпеки – контур безпеки підтримується системою входу в
 інформаційну екосистему K_f_Security JetIQ VNTU та власними інструментами
 викладача щодо надання доступу до електронних навчальних ресурсів
 визначеним групам студентів, використання інструментів захисту для різних
 форматів електронних ресурсів.

$$\begin{aligned}
 K_elbook = \{ & K_f_teach (Learn_instr_teach; Cr; Content; Planm; Lsms; Tests, \\
 & Val; Tasks; Exterlinks); K_f_st (Learn_instr_st; Read_st; Not_st; Tasks_St; Tests_st; \\
 & SMS_st; Forum_st); K_f_SM (Track_St; Auto-SMS, Gam); K_f_SI (Integer_navigator; \\
 & Integer_test_results; Integer_communication_teacher; Integer_sms; Integer_sendfile; \\
 & Integer_forum); K_f_Security.
 \end{aligned}
 \tag{2.6}$$

Надана математична модель функціонального контуру містить складові інших контурів, які можуть бути відокремлені та візуалізовані. Всі контури можуть бути активовані діями викладача та студента.

На рисунку 2.6 представлена модель інтерактивного електронного підручника з реалізацією визначених контурів у вигляді фокусування уваги на елементи контурів.

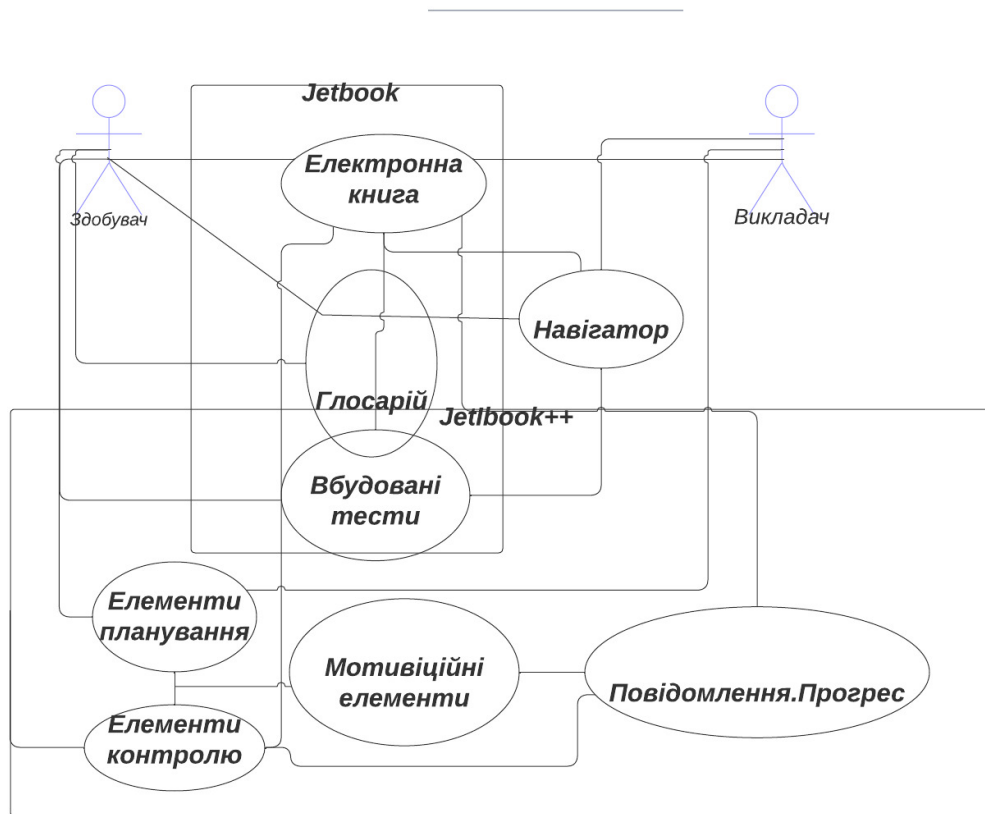


Рисунок 2.6 – Модель середовища «Електронний підручник» з функціональними, комунікаційними, мотиваційними та управлінськими елементами

інформаційних технологій у навчальному процесі [31].
Таблиця 2.1 показує можливості реалізації моделі Carrington Wheel на двох рівнях, де визначено основні технології, результати та цілі розвитку.

Таблиця 2.1 – Проекції моделі колеса Керінгтона для електронного підручника ІТ-напряму (рівні освітньої програми та дисципліни)

Запам'ятовування	(1); тестовий контроль	(2); електронні тести	(3), словник, глосарій, тести дистанційного курсу	(3), словник, глосарій електронної книги; Тест IQ
Розуміння	(4)	(2)	(3); (6); вбудована в дистанційний курс відеоконференція, відео та аудіо подкасти; електронні тести (9)	(3); (6); посилання на відеоконференцію та вбудовані відео-та удаіо подкасти навігатора дисципліни та посилання в електронній книзі, Тест IQ (10)
Застосування	(7), послідовні завдання проєкту (індивідуальні та командні)	(2), Робота в середовищах проєктів	(3), (6), посилання на середовища проєктів, середовища командної роботи (9)	(3), (6), (9), (10), Google диск (11)
Аналіз				
Оцінювання				
Створення				

(*) – дії, що повторюються на різних етапах.

На рівні окремого заняття можна також визначити конкретні дії та процедури, які викладач буде здійснювати за допомогою електронного підручника та інструментів системи управління навчанням.

Так, наприклад, практичне заняття з дисципліни «Основи програмної інженерії» – тема «Виявлення вимог клієнта до програмного продукту» може бути проведено на основі такої сценарної моделі:

- коротка інформація зі стандарту SWEBOOK (10 хв),
- воркшоп з малими групами (формування вимог за простими прикладами – 15 хв, результати – в документи або в чат),
- аналіз результатів (5 хв),
- фасилітація вимог до ІТ продукту за групами (онлайн-дошка Miro – 15 хв), аналіз результатів (5 хв),
- узагальнений аналіз та огляд підходів до виявлення вимог щодо розробки ІТ-проектів з використанням практичних прикладів (20 хв).

Також може бути додано до заняття:

- методика виявлення та визначення цільової аудиторії та подання вимог до ІТ-продукту, що буде реалізований під час навчання (20 хв),
- відповіді на питання та завдання на самостійну роботу (5 хв).

Використовуються: платформа JetIQ з вбудованими сервісами Google, фасилітація – електронна дошка Miro.

На рівні використання електронного підручника для окремих груп студентів електронне освітнє середовище Z_{eos} розглядається як множина лекційного Z_l , аналітично-результативного Z_{anr} модулів та блока самостійної активності Z_{sakt} відповідно до поведінки агентів [32]:

$$Z_{eos} = \langle Z_l; Z_{sakt}; Z_{anr} \rangle \quad (2.6)$$

Навчальний процес організовано в комбінованому синхронно-асинхронному режимі на основі аналізу наявності якісних комунікаційних засобів та можливостей для активного синхронного діалогу між студентами та викладачами.

Для цього рекомендується використовувати матрицю Деніела Стенфорда, яка дозволяє визначити сценарії синхронного та асинхронного форматів класів (рис. 2.8).

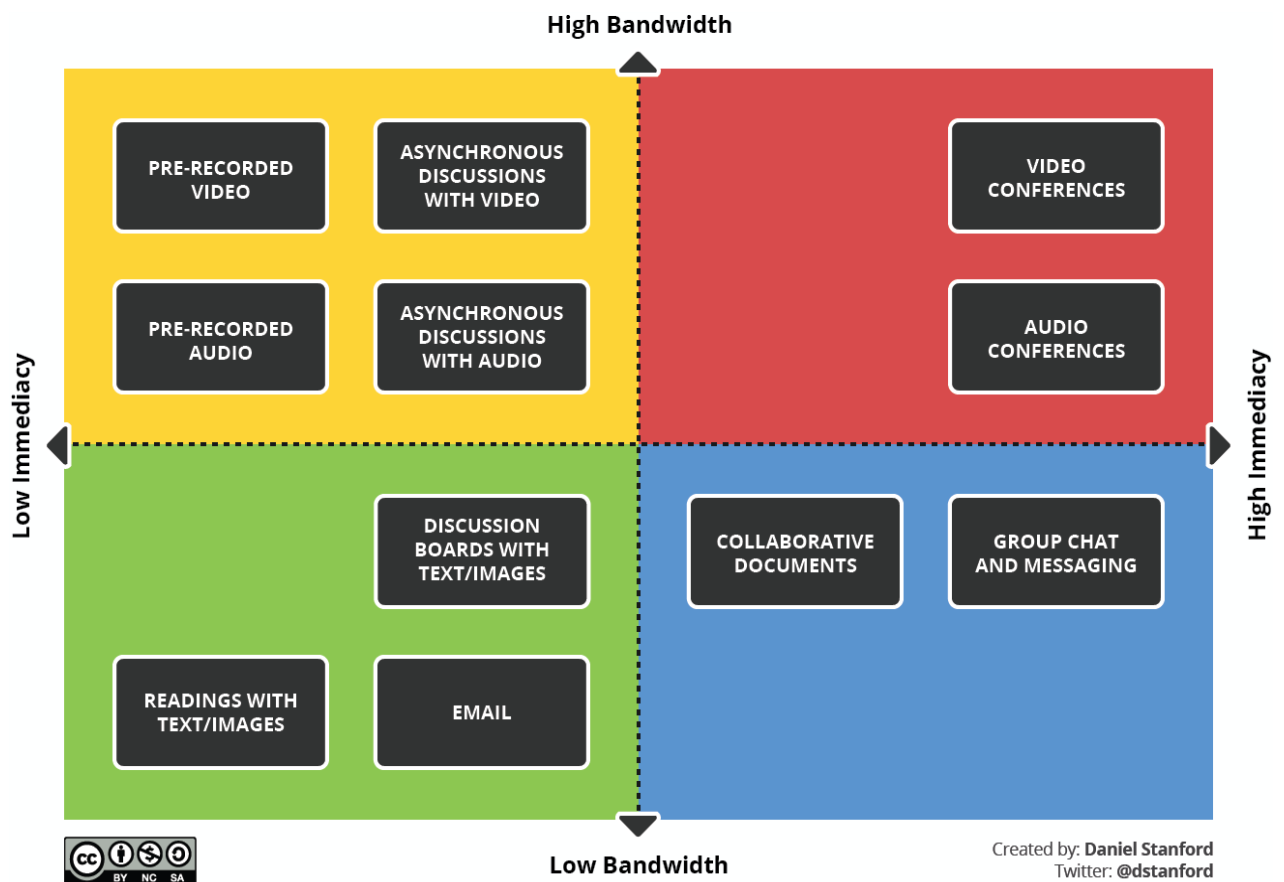


Рисунок 2.8 – Матриця сценаріїв синхронного та асинхронного форматів [33]

Загальний підхід до планування практичного заняття з використанням електронного підручника представлено в таблиці 2.2.

Таблиця 2.2 – Проекції моделі колеса Керінгтона для розділів електронного підручника

Рівень окремого практичного заняття з дисципліни				
Запам'ятовування	(1)*; тестовий контроль, опитування,	(2); електронні тести; відеоконференції, електронні дошки для фасилітації, узагальнення (12)	(3), словник, глосарій, опитування, тести дистанційного курсу	(3), словник, глосарій електронної книги; опитування, Тест IQ
Розуміння	Опитування в малих групах	Робота малих груп в спеціальних середовищах (15)	Виконання практичних завдань, висилання файлів викладачу, підтримка діалогу для розуміння за допомогою інструментів середовища дистанційного навчання (13). Пересилання файлів викладачу, чат (14)	
Застосування	Завдання за прикладом	Створення програм, моделей в спеціальних середовищах за шаблонами та прикладами	(3); (6); (9); (13), (14)	(3); (6); (9); (13), (14), курсова робота (проєкт) – завдання-шаблон
Аналіз	Аналітичний звіт	Аналіз результатів за методикою формульовального оцінювання, електронні дошки	(3); (6); (9); (13), (14)	(3); (6); (9); (13), (14), пояснювальна записка курсової роботи – аналіз результатів - надсилається в файловий архів після закінчення
Оцінювання	Оцінювання роботи та перехресне оцінювання робіт інших команд та студентів			
Створення	Модулю загального проєкту	Створення власних програм, моделей, алгоритмів. Аналіз та оцінювання результатів,		

(*) – дії, що повторюються на різних етапах (дані таблиці 1, 2).

Усі цифрові інструменти дозволяють досягти всіх рівнів таксономії Блума – запам'ятовувати, розуміти, використовувати, аналізувати, оцінювати, створювати в рамках курсу та розвивати їх у подальшій самостійній роботі та інших курсах.

На рис. 2.9 показано модель сценарію впровадження курсу «Основи програмної інженерії» з використанням методів проєктного навчання за темою «Використання програмного забезпечення для управління проєктами та інтелектуальних карт для демонстрації та реалізації проєктів».

Постановка задачі (5 хв). Варіанти завдань проєкту (підготовлене відео) (10 хв), методичні вказівки (текст). Відповіді на питання (асинхронне відео). Робота команди в середовищі проєкту(самостійна робота). Рекомендації - підготовлене відео (5 хв.), Чат . Форум. Надіслані звіти . Контроль результатів - оцінки в електронному журналі. Тести.	Постановка задачі (5 хв). Варіанти завдань проєкту (відеоконференція) (10 хв), методичні вказівки (текст). Відповіді на питання (5 хв.). Контроль наявності учасників команд (5 хв.). Робота команди в середовищі проєкту(45 хв.). Контроль роботи команди (5 хв.), Чат . Синхронний діалог. Контроль результатів (5 хв. - одна команда).
Постановка задачі (5 хв). Варіанти завдань проєкту (підготовлене відео) (10 хв), методичні вказівки (текст). Відповіді на питання (асинхронне відео). Робота команди в середовищі проєкту за шаблонами (самостійна робота). Рекомендації - підготовлене відео (5 хв.), Чат . Форум. Надіслані звіти . Контроль результатів - оцінки в електронному журналі. Тести.	Постановка задачі (підготовлене відео - 5 хв). Варіанти завдань проєкту (відеоконференція) (10 хв), методичні вказівки (текст). Робота команди в середовищі проєкту(45 хв.). Чат . Групове перехресне оцінювання звітів з роботи. Тести.

Рисунок 2.9 – Модель використання електронного підручника за проєктним підходом

На рис. 2.10 представлено загальну сценарну модель, яку може використовувати викладач.

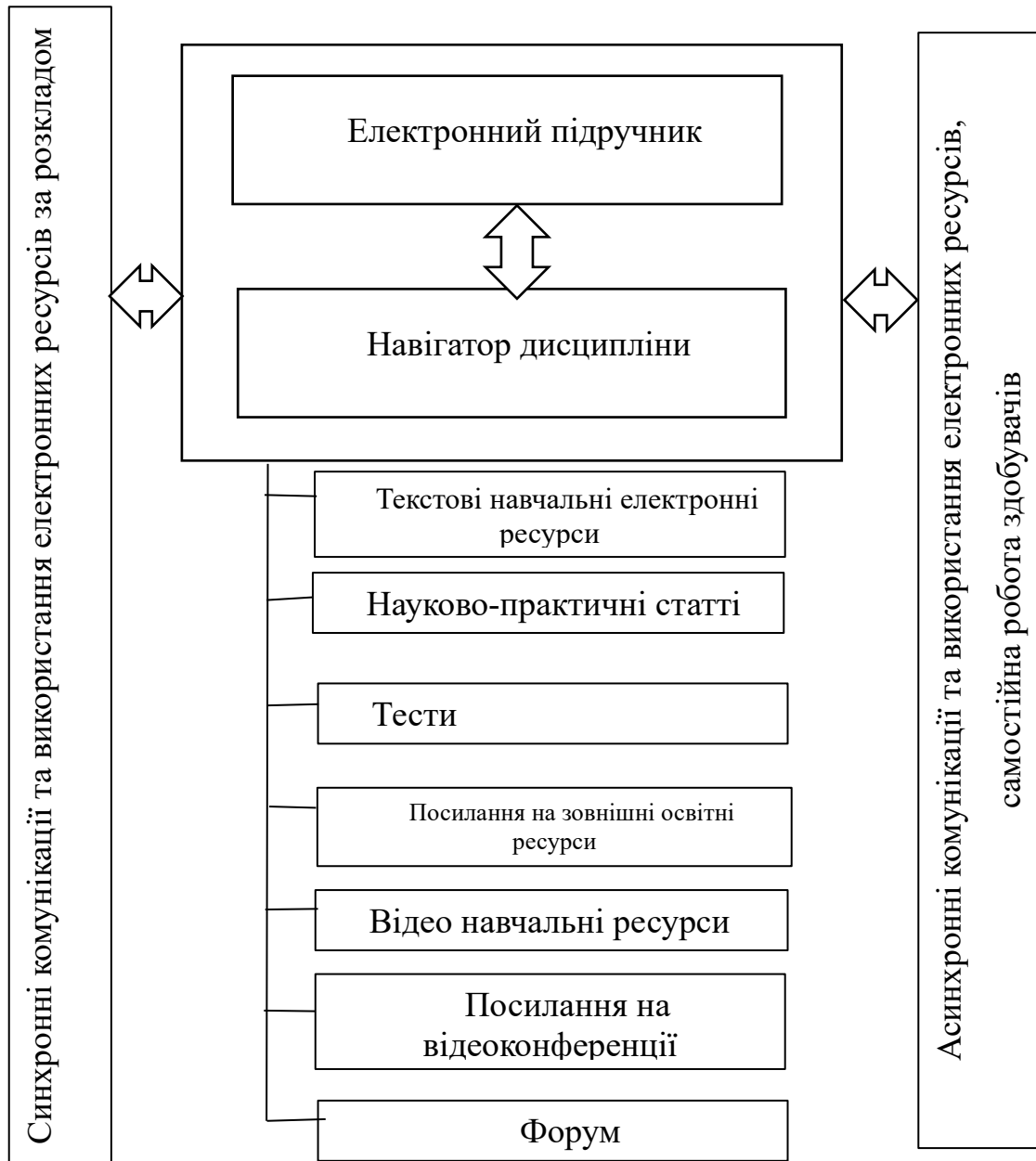


Рисунок 2.10 – Загальна сценарна модель використання електронних ресурсів у форматі гнучкого змішаного навчання

Основою інформаційної технології є метод до створення електронних підручників із вбудованими інструментами для синхронного та асинхронного навчання полягає у формулюванні моделей, сценаріїв та алгоритмів для

використання технологічних можливостей для ефективного впровадження електронних підручників. Спосіб передбачає виконання наступних етапів: Постановка проблеми визначення вимог до мікросервісу «JetIQ Electronic Textbook»;

Визначення загальних функцій мікросервісів;

Формування функціональної структури на основі загальної схеми системи JetIQ та деталей мікросервісу електронного підручника;

Визначені профілі за допомогою математичних і сценарних моделей і алгоритмів;

Технічний аналіз впровадження електронного підручника.

Визначені методики інтеграції складних операцій мікросервісів.

Інтерактивні електронні підручники – це більше, ніж просто електронні версії паперових книг. Це динамічний засіб навчання, який активно залучає читачів до процесу засвоєння матеріалу. Існує багато режимів інтерактивних навчальних матеріалів, кожен режим має свої особливості та переваги. За структурою модель інтерактивного електронного підручника можна поділити на лінійний гілковий та адаптивний типи. Лінійна модель передбачає, що навчальний матеріал викладається послідовно від початку до кінця. Читач рухається за фіксованим маршрутом від однієї ділянки до іншої. Перевагами цієї моделі є те, що її легко створити та використовувати, а структура навчання зрозуміла. Недоліком є те, що діяльність користувача обмежена, і не вистачає адаптивних функцій.

Розгалужена модель передбачає, що матеріал подається в нелінійній формі, а читач обирає наступний розділ на основі своїх знань та інтересів. Ця модель сприяє персоналізації навчання та дозволяє студентам самостійно будувати власний освітній шлях.

Модель може бути складною для навігації та вимагає ретельного планування навчання.

Крім того, можливе використання цільових моделей на основі основних об'єктів навчання. Моделі на основі завдань формують доступ до кожної частини інформації відповідно до завдання. Цей режим активізує пізнавальну діяльність, дозволяючи більш ефективно отримувати зворотній зв'язок.

Однак, якщо студенти не можуть виконати завдання, вони втратять мотивацію.

Модель на основі симуляції забезпечує навчання впровадження через взаємодію з віртуальними моделями реальних процесів або систем. Ця модель дозволяє отримати практичні навички та розвинути критичне мислення. Але це вимагає розробки складних симуляцій, що може бути дорогим і неприйнятним для державних установ.

Моделі, засновані на співпраці, передбачають роботу в групах, розвиток комунікативних навичок, але вимагають спеціальних інструментів.

Ключовими елементами інтерактивних посібників є текст, зображення, відео, аудіо, тести, вправи, симуляції, ігри, засоби навігації та зворотного зв'язку.

Макет електронної книги з мотиваційним планом формується для підвищення мотивації студента. Мотиваційний цикл в електронному підручнику — це набір елементів і механізмів, які стимулюють читача активно вивчати матеріал і просувають його вперед у процесі навчання. Це не просто текст, а інтерактивний досвід, який залучає різні канали сприйняття та заохочує до взаємодії. Інтерактивні елементи роблять навчання веселішим, менш пасивним і сприяють підвищенню залученості. Метод створення електронного підручника з вбудованими інструментами синхронного та асинхронного навчання полягає у формуванні моделей, сценаріїв та алгоритмів використання можливостей

технологій для ефективної реалізації електронного підручника. Такий метод передбачає реалізацію таких етапів:

- формування постановки задачі; виявлення вимог до мікросервісу «Електронний підручник JetIQ»;

- визначення загального функціоналу мікросервісу;

- визначення контурів роботи функціоналу відповідно до загальних контурів системи JetIQ та деталізація у мікросервісі електронного підручника;

- структуризація визначених контурів за допомогою математичних та сценарних моделей та алгоритмів;

- аналіз технологій реалізації електронного підручника;

- інтеграція визначених технологій для комплексної роботи мікросервісу.

Інтерактивний електронний підручник – це не просто електронний аналог паперової книги. Це динамічний інструмент навчання, який залучає читача до активної участі в процесі освоєння матеріалу. Існує кілька моделей інтерактивних підручників, кожна з яких має свої особливості та переваги.

За структурою моделі інтерактивного електронного підручника можна розділити на лінійну, розгалужену, адаптаційну.

Лінійна модель передбачає, що навчальні матеріали подаються послідовно, від початку до кінця. Читач переходить від одного розділу до іншого за фіксованим маршрутом.

Переваги цієї моделі у простоті створення та використання, чіткої структури навчання. Недоліком є обмеження активності здобувача та відсутність адаптивних функцій.

Розгалужена модель передбачає, що матеріал подається в нелінійній формі, читач може вибрати наступний розділ залежно від своїх знань та інтересів.

Така модель сприяє індивідуалізації навчання, дозволяє здобувачу самостійно будувати свій навчальний шлях.

Модель може бути складною в навігації, вимагає ретельного планування структури.

Також можуть бути використані цільові моделі, які базуються на головному об'єкті в навчанні – теми, завдання, симуляція тощо.

Модель, що базується на завданнях, формує доступ до кожного блоку інформації відповідно до завдань. Така модель активізує пізнавальну діяльність, дозволяє більш ефективно отримувати зворотний зв'язок. Але, якщо студент не може виконати завдання, це може бути демотивацією.

Модель, що базується на симуляціях передбачає реалізацію навчання через взаємодію з віртуальними моделями реальних процесів або систем. Така модель дозволяє отримати практичні навички, розвиває критичне мислення. Але вимагає розробки складних симуляцій, що може бути дорогавартісним.

Модель, що базується на співпраці передбачає роботу в групах, розвиває комунікативні навички, але вимагає спеціальних інструментів.

Ключові елементи інтерактивного підручника це текст, зображення, відео, аудіо, тести, вправи, симуляції, ігри, інструменти навігації та зворотного зв'язку.

Моделі електронних книг з мотиваційним контуром дозволяють зробити навчання більш захопливим.

Мотиваційний контур в електронній книзі – це сукупність елементів та механізмів, які стимулюють читача до активного засвоєння матеріалу та рухають його вперед у процесі навчання. Це не просто текст, а інтерактивний досвід, який залучає різні канали сприйняття та заохочує до взаємодії. Інтерактивні елементи роблять навчання більш цікавим і менш пасивним та сприяють збільшенню рівня залученості. Активна участь у навчальному процесі сприяє кращому запам'ятовуванню матеріалу. Почуття досягнення, зворотний зв'язок та можливість вибору шляху навчання мотивують до подальшого навчання.

Мотиваційний контур дозволяє адаптувати темп і складність матеріалу до потреб кожного читача.

Елементи мотиваційного контуру включають в себе повідомлення, тести, симуляції зі зворотним зв'язком, ігри, відео та аудіо, інформація про результати, бали, досягнення, лідери.

Мотиваційний контур робить процес навчання більш ефективним і надає відчуття залучення. Використання інтерактивних елементів, зворотного зв'язку та інших мотиваційних механізмів дозволяє залучити читачів до активної участі в навчанні та підвищити їхню зацікавленість у матеріалі.

Інтерактивні електронні книги відкривають нові горизонти для взаємодії між студентом і викладачем. Це не просто пасивне читання, а динамічний процес, де кожен учасник навчального процесу може активно взаємодіяти.

Існує декілька способів забезпечити ефективну комунікацію між студентом і викладачем в межах електронної книги – прямий діалог з викладачем та групові обговорення, система повідомлень індивідуально та масові розсилання, відеоконференції, застосування проєктного підходу виконання навчального завдання як створення продукту, дослідження у співпраці з викладачем, групою.

Комунікація в електронному підручнику – це важливий елемент сучасного дистанційного та змішаного навчання. Вона дозволяє зробити процес навчання більш інтерактивним, ефективним і приємним. Використання різних інструментів і підходів дозволяє створити комфортне середовище для навчання і взаємодії між студентами і викладачами. Комунікаційний контур перевіряється за рівнем активності всіх учасників комунікацій.

Функціонал нотаток та закладок в електронних підручниках – це потужний інструмент для персоналізації навчання. Він дозволяє кожному читачеві створити власний навчальний досвід, виділяючи важливу інформацію, формулюючи запитання та зберігаючи для подальшого опрацювання.

Нотатки можна реалізувати безпосередньо разом з текстом але в своїй версії книги, в окремому вікні разом з цитатами та посиланнями.

Доцільно також здійснювати маркування важливих фрагментів, швидкий доступ до конкретних розділів, створення персонального змісту на основі закладок. Моделі нотаток і закладок можуть бути організовані в папки та підпапки для зручного доступу, мати синхронізацію нотаток між різними пристроями, можливість експорту нотаток у форматі PDF, DOCX тощо.

Функціонал нотаток та закладок в електронних книгах перетворює їх з пасивного джерела інформації в активний інструмент навчання. Це дозволяє кожному читачеві створити свій власний навчальний досвід, адаптований до його індивідуальних потреб і стилю навчання, сформувати власну базу знань.

Електронний підручник має потенціал стати не просто носієм інформації, а інструментом для всебічного розвитку особистості. Завдяки своїм можливостям вона може сприяти формуванню критичного мислення, розвитку творчих здібностей та навичок самостійного навчання.

При розробці електронного підручника необхідно визначити основні процеси та елементи і зв'язки впливу на його якість. Це можна реалізувати на основі моделі, що представлена на рис. 2.10 [14].

Якість виготовлення інтерактивного електронного підручника (ІЕП), як і будь якого іншого технологічного процесу, залежить від багатьох факторів.

X1 – прототип – чорновий дизайн, який є базою для створення структури ЕП, позначає зв'язки та передбачає деталізацію методології навчання з використанням електронного підручника. Для програмної реалізації необхідно врахувати всі моделі та можливості для взаємодії з електронними ресурсами та викладачем.

X2 – електронний макет – представлення текстової, графічної, аудіо та відео інформації в комплексі, дизайн ЕП. Визначення використання різноманітної

інформації та зв'язок між ними. Так, наприклад, може бути симуляція гри, вбудована в електронний підручник, використання аудіотестів для інклюзії тощо.

X3 – текстовий матеріал – найбільш вагома частина підручника, містить посилання на глосарій, зовнішні посилання, аудіо, відео. Має інструменти стопслів, ключових слів, моніторингу перегортання сторінок тощо. проходить редакторське опрацювання, вичитку та коректурну правку.

X4 – графічний матеріал – ілюстрації, фото, анімації, графіки, схеми, що використовуються в електронному підручнику, тестах. Вони потребують спеціальної обробки та зберігання на сервері.

X5 – аудіоматеріал – зазвичай дублює текстовий матеріал, доповнює його. Елементи гейміфікації дозволяють виконати симуляцію процесів, набирати бали за виконання завдань, тестів.

X6 – відеоматеріал – доповнення до текстової інформації з елементами візуалізації.

X7 – верстання ІЕП – компоновання елементів в межах сторінок, надання їм стильового оформлення.

X8 – інтерактивні елементи – забезпечують взаємодію користувача з підручником, впливають на процес запам'ятовування та засвоєння матеріалу.

X9 – дизайн ІЕП – загальний вигляд підручника. Сукупність елементів таких, як текстовий, графічний, відео та аудіо матеріали, інтерактивні елементи, які формують одне ціле.

X10 – бекенд розробка – підключення бібліотек, компонент, бази даних, побудова логіки додатку.

X11 – фронтенд розробка – формування вигляду електронного підручника для користувача, інтерфейсу, інтерактивних елементів.

X12 – тестування та виправлення.

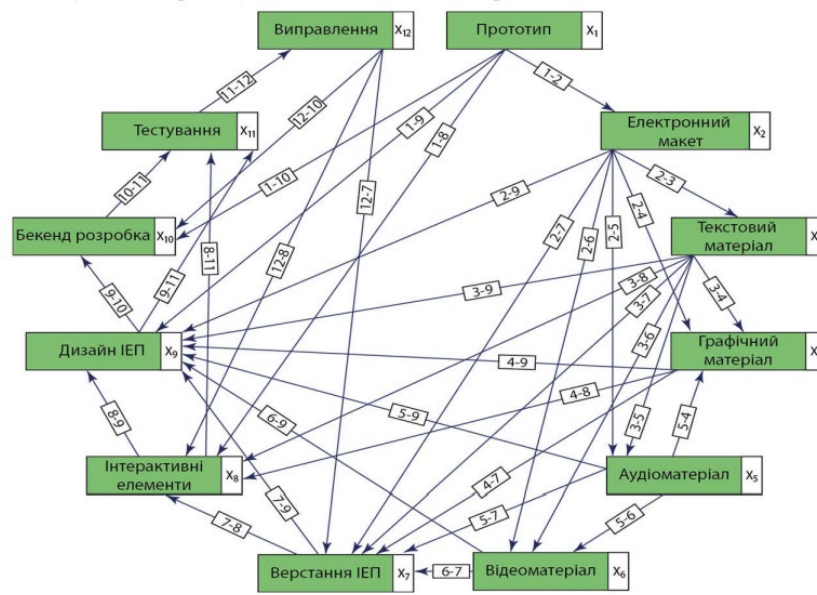


Рисунок 2.11 – Графова модель якості створення інтерактивного електронного підручника

Сучасні електронні підручники передбачають використання методу управління SMART за такими критеріями [14]:

S – Specific; електронний підручник має структурований контент, можливості роботи із закладками, нотатками, мотиваційними повідомленнями та спеціальними тестами;

M – Measurable; вимірювальний – результати читання моніторяться відповідно до змін курсора і часу використання електронного підручника, результати засвоєння матеріалів фіксуються за допомогою виконання вбудованих тестів;

A – Achievable, досяжний – завдання є досяжними, здобувач отримує повідомлення про прогрес та мотивації;

R – Relevant, значущий – структурована та визначена різним дизайном інформація, використання кольорів для закладок. Інформація структурована з атемами, визначення – в глосарії;

T – Time-bound, обмежений в часі – завдання з навчання обмежені в часі, відповідають термінам навчання.

Визначені моделі дозволяють користуватись електронним підручником з навігатора, формувати елементи планування, контролю та мотивації. Контури роботи всієї системи аналогічно реалізуються в електронному підручнику. Крім функціоналу – зі сторони викладача є додавати та редагувати підручники, розділи, формувати методику використання, динамічно задавати терміни виконання завдань, є також мотиваційний контур і можливість – редагувати мотиваційні повідомлення, наповнювати та редагувати вбудовані тести і комунікаційний контур з контролем – визначати завдання, реагувати на виконання, оцінювати результати.

Зі сторони студента є можливості отримувати електронні ресурси у вигляді інтерактивного електронного підручника, виконувати завдання викладача, проходити тести, працювати з електронним підручником самостійно та під час занять.

Запропоновані інформаційна технологія створення електронного підручника на основі принципів педагогічного дизайну та контурної моделі електронного середовища має вбудовані інструменти підтримки користувача щодо виконання завдань для самостійної роботи, виконання мікротестів до визначених тем, формування бази знань для конкретної дисципліни і поточного користувача з подальшою можливістю використання отриманих балів в загальному оцінюванні результатів навчання студентів викладачем, що підвищує рівень засвоєння знань та ефективність самостійної роботи студента; дає

можливість сформувати власну базу знань студента на основі електронного підручника.

В системах управління навчанням запроваджено інформаційну технологію для створення електронних підручників, але запропонована удосконалена інформаційна технологія створення електронного підручника для систем управління навчання, містить елементи, які дозволяють використовувати електронний підручник як самостійний застосунок з активною взаємодією всіх агентів системи управління навчанням, а також як вбудований інструмент дистанційного та змішаного курсу з визначеної дисципліни, що сприяє покращенню результатів засвоєння та використання знань з дисципліни.

2.3 Висновки

Запропонована інформаційна технологія, що включає в себе методи, моделі, технології, інструменти для створення електронного підручника базується на загальній контурній моделі і дозволяє сформувати мікросервіс «Електронний підручник». В процесі наповнення та використання електронний підручник перетворюється в інформаційну базу знань дисципліни.

3. Реалізація інформаційної технології електронного підручника в системі JetIQ

3.1 Можливості запровадження інформаційної технології створення електронного підручника в системі JetIQ

Створення електронного підручника – це складний процес, який вимагає використання відповідного програмного забезпечення. Вибір оптимального інструменту залежить від багатьох факторів, таких як складність підручника, визначені інтерактивні елементи, формат, вимоги до інтерфейсу, використання різноманітних готових рішень програмного забезпечення або створення конструктора для електронного підручника з інтерактивними функціями як самостійного власного програмного забезпечення.

Для системи JetIQ є можливість використовувати електронні підручники вбудованих систем Microsoft та Google, а також створити власний мікросервіс.

В межах дослідження в магістерській роботі буде створена інформаційна технологія як сукупність методів, моделей та алгоритмів на основі власного мікросервісу. Для реалізації була проаналізована існуюча технологія, виявлено відсутність інтерактивних елементів, потреба в удосконаленні зв'язків з сервісами системи, такими як гейміфікація, електронний журнал, тести. Визначено фокус формування електронного підручника як бази знань для викладача та студента.

Електронний підручник перетворюється в базу знань для викладача під час формування тестів, посилок, реєстрації досвіду навчання студентів, зберігання результатів обговорення дискусійних тем, додавання елементів візуалізації, анімації, творчих робіт студентів.

Електронний підручник перетворюється в базу знань для студента під час навчання і формування нотаток, встановлення закладок, виконання тестів та

завдань, групового обговорення та його збереження у форумі.

На рисунку 3.1 представлено модель взаємодії електронного підручника та веб-сервісів системи JetIQ VNTU.

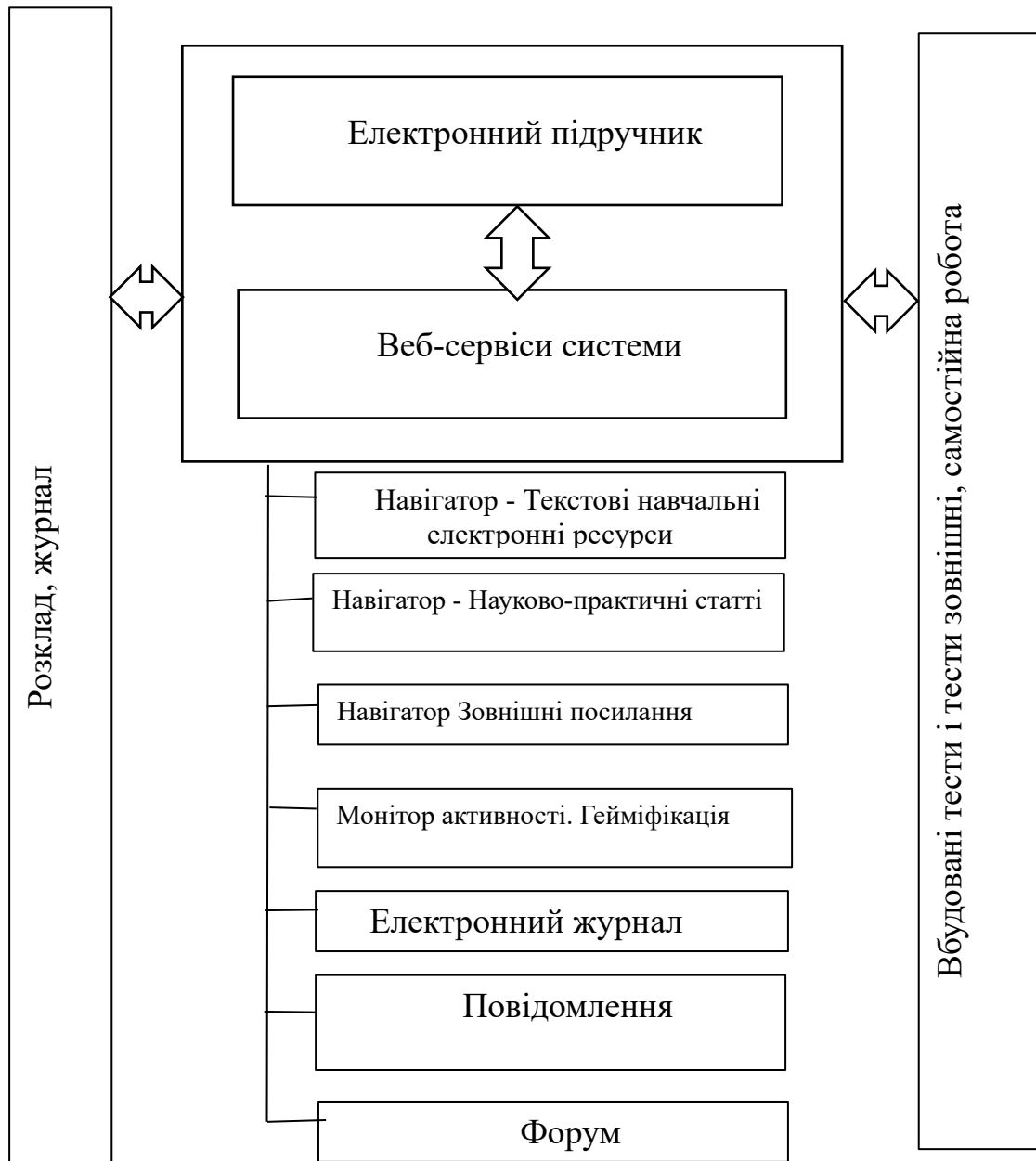


Рисунок 3.1 Модель взаємодії електронного підручника та веб-сервісів

Головна ідея представленої моделі – віддзеркалювати активність студента

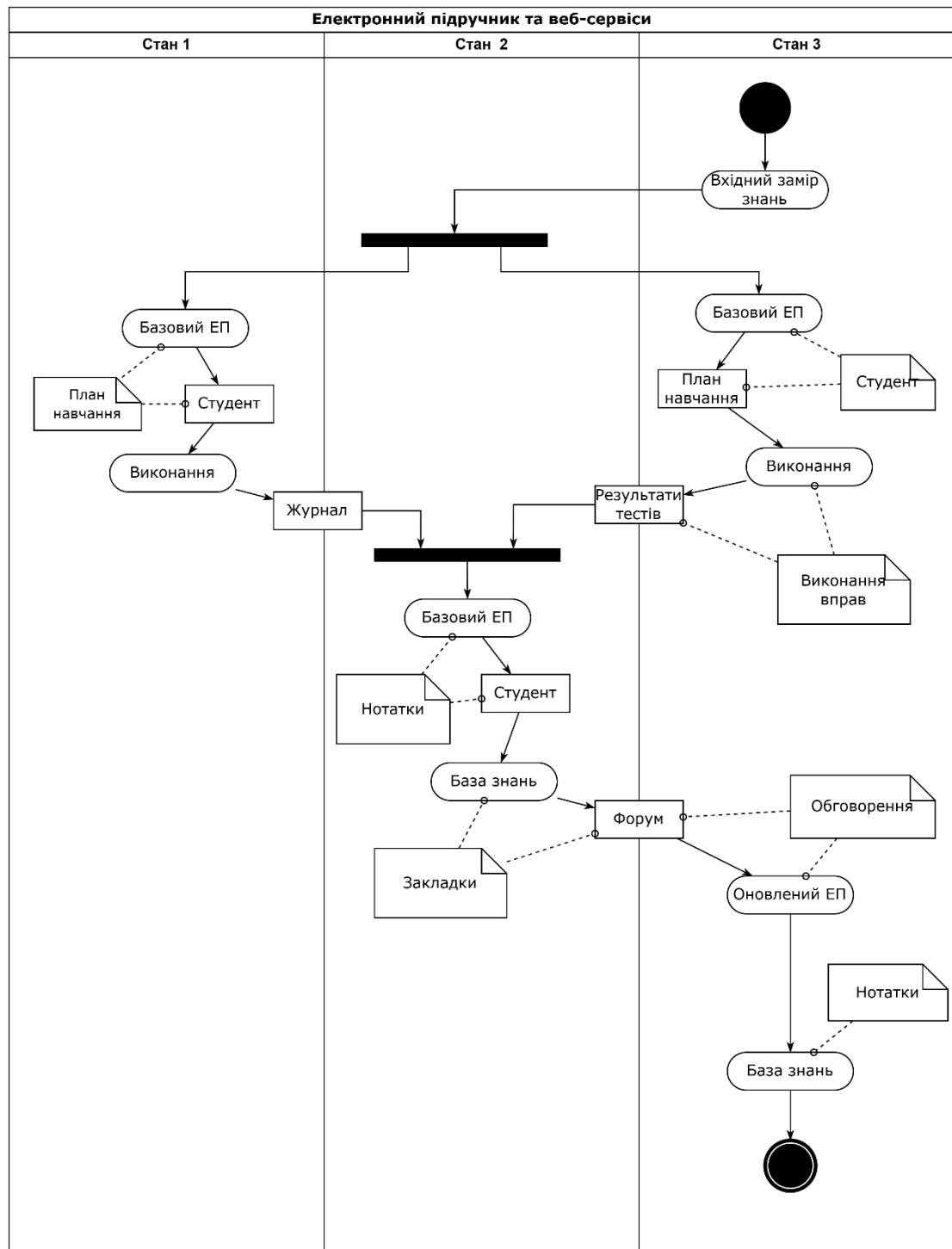
в середовищі електронного підручника. Кількість прочитаних сторінок – отримання повідомлення про виконання, або швидке гортання – мотивація повернутись і прочитати ще раз. Кількісні та якісні результати проходження тестів, спеціальні відмітки в гейміфікації про самостійну роботу і будь-яку активність з електронним підручником, робота з ресурсами за розкладом та самостійно. Головною взаємодією є навігатор, система тестування, а також електронний журнал, відомості тестування, система гейміфікації.

3.2 Моделі та алгоритми програмної реалізації електронного підручника

На рис. 3.3 представлено модель станів формування електронного підручника з врахуванням комунікацій студента з викладачем, а також з іншими модулями системи JetIQ VNTU.

Базовий електронний підручник формується викладачем і завантажується в навігатор. Він містить текст, зображення, посилання на зовнішні та внутрішні ресурси, глосарій.

Студент і викладач можуть додавати нотатки, закладки. Викладач дає завдання, активізує вбудовані або зовнішні тести. Запускає систему моніторингу та фіксації оцінок в журналі та у відомостях тестування. Можливо використання третього стану – замір вхідних даних. Заповнені нотатки та закладки дозволяють сформувати власну базу знань. Результати обговорень також можуть увійти в базу знань. Такий досвід активно використовується у відкритих дистанційних курсах. Оновлений ЕП є основою для подальшого використання та внесення змін в електронний підручник.



На рисунку 3.3 представлено модель емоційно-мотиваційного контуру, що включає в себе генерацію повідомлень, прогресу результатів, моніторинг активностей здобувача, відмітки в статусах і нагородах гейміфікації, перетворення електронного підручника в базу знань.

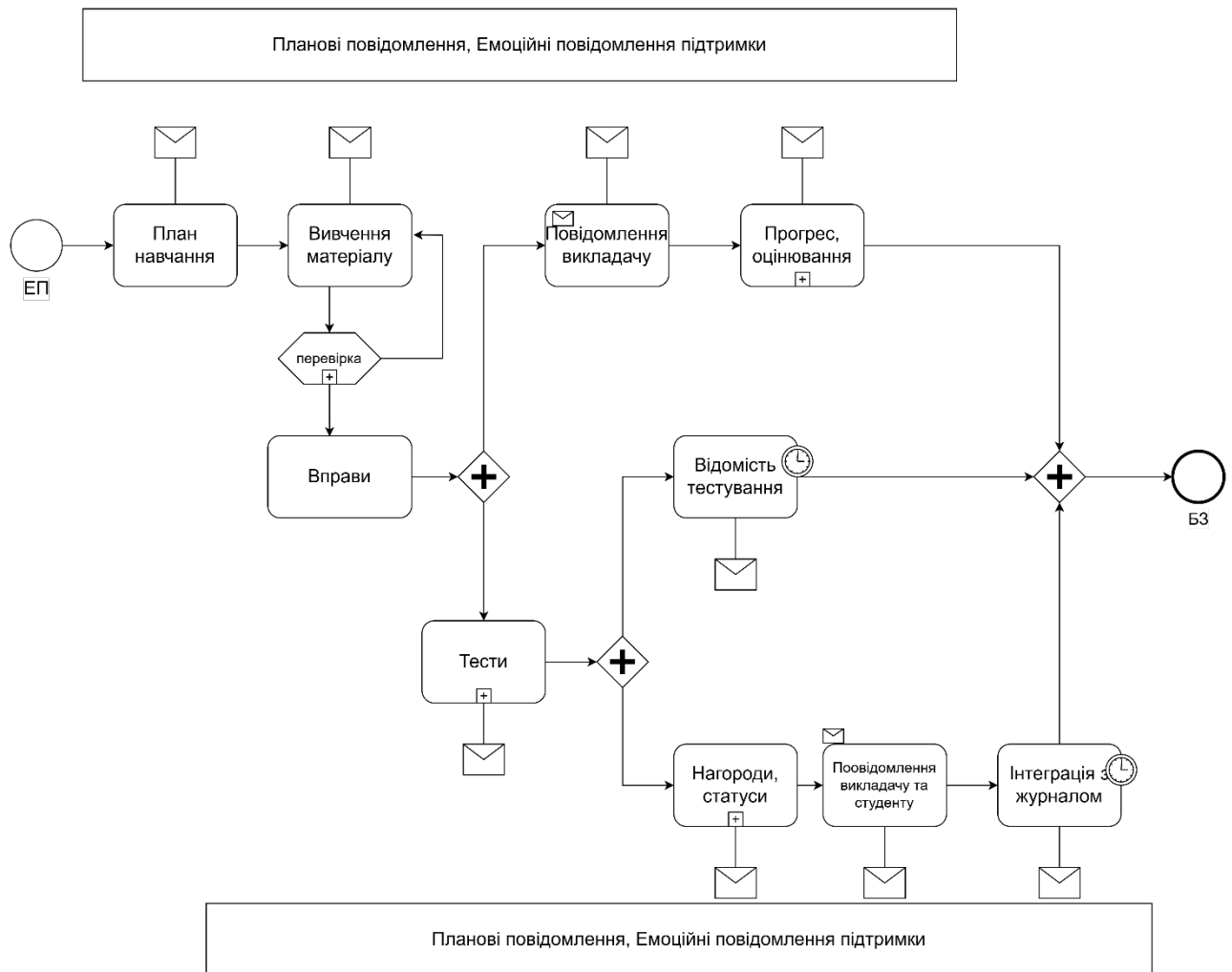


Рисунок 3.3 – Модель емоційно-мотиваційного контуру ЕП -БЗ

Процеси інтеграції результатів самостійної роботи в середовищі електронного підручника з електронним журналом викладача та системою гейміфікації можуть бути виконані за допомогою технологій REST; JSON; з використанням API.

Отже, запропоновані моделі поступово можуть бути запроваджені в систему JetIQ VNTU.

3.3 Технології PHP та Javascript для створення електронного підручника JetIQ

PHP і JavaScript часто асоціюються з веб-розробкою, і техніки цих мов програмування можуть бути потужними інструментами для створення електронних книг, особливо інтерактивних посібників. PHP дозволяє створювати динамічний вміст, який особливо корисний для створення інтерактивних елементів в електронних книгах (наприклад, тести, вправи, ієрархічні розділи).

Електронні книги, створені за допомогою веб-технологій, можна відкрити на будь-якому пристрої за допомогою веб-браузера.

JavaScript дозволяє додавати до електронних книг інтерактивні елементи, такі як анімація, звукові ефекти, слайдери тощо.

Існує велика спільнота розробників PHP і JavaScript, а це означає, що доступно безліч бібліотек, фреймворків і навчальних посібників.

Основні етапи створення електронного підручника включають визначення структури проєкту, папок, створення макетів веб-сторінок, вмісту, збереження та використання зображень, списків і таблиць. Крім того, потрібно вибрати стилі, елементи анімації моделі, модальні вікна, вбудовані тести та навігацію.

Фреймворки PHP і JavaScript, такі як Laravel, Symfony або React, є основою для створення веб-сторінок, а також електронних книг.

Створення електронних книг за допомогою PHP і JavaScript — це комплексний і гнучкий спосіб створення інтерактивних динамічних книг.

PHP і JavaScript — дві популярні мови програмування, які часто використовуються разом для створення динамічних та інтерактивних веб-додатків. Кожен із них має власні сфери застосування, але, працюючи разом, вони можуть досягти синергії.

PHP виконує обробку даних на стороні сервера, взаємодіє з базами даних і генерує HTML-код. JavaScript робить веб-сторінки інтерактивними, реагуючи на дії користувача без перезавантаження сторінки. Завдяки їх взаємодії можна створювати веб-сторінки, які можна змінювати без перезавантаження, покращуючи таким чином взаємодію з користувачем. JavaScript дозволяє додавати різні інтерактивні елементи на ваші веб-сторінки: форми, слайдери, анімацію, графіку тощо. Вибираючи мову програмування для серверної частини (backend), ми розглядали кілька популярних варіантів, таких як Java, PHP, Python тощо. Після детального аналізу було обрано PHP. Це пов'язано з такими критеріями:

Досвід команди розробників системи JetIQ і потреба перенести електронні книги в систему управління навчанням, а також наявність великої спільноти розробників означає, що існує багато готових рішень і підтримки.

PHP дозволяє швидко створити прототип і розширити систему за потреби.

PHP можна легко інтегрувати з різними системами керування базами даних, що дуже важливо для зберігання інформації електронних книг.

Крім того, ця мова також має достатньо високий рівень у рейтингах інших мов програмування і може бути поєднана з JavaScript. На рисунку 3.4 представлено графічне порівняння рейтингів мов програмування [27].

Відповідей: 807!

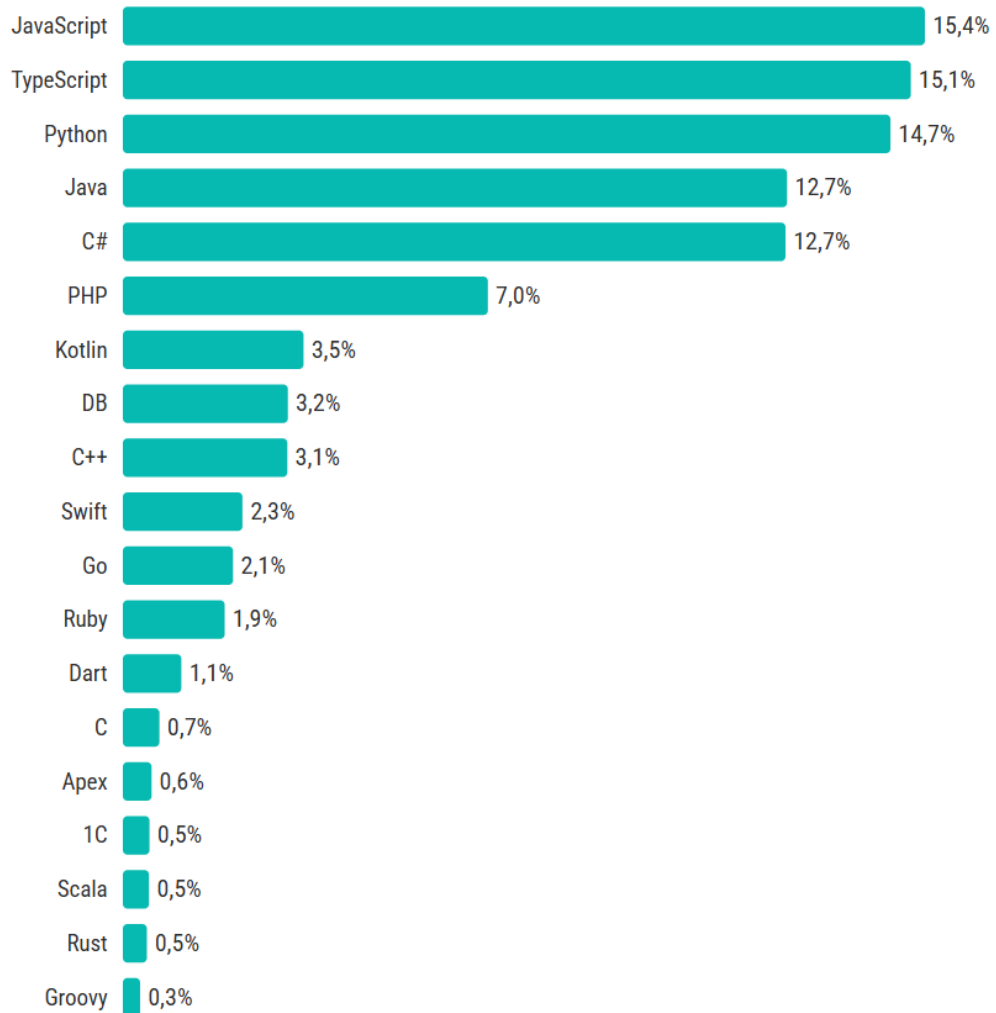


Рисунок 3.4 – Рейтинг мов програмування за опитуванням в 2024 році []

Вибравши PHP як серверну частину для генератора інтерактивних електронних підручників, ви можете створити надійне, гнучке та ефективне рішення. Ця мова програмування пропонує найкращий баланс між простотою використання та функціональністю, що робить її чудовим інструментом для розробки веб-застосунків.

Основні способи інтеграції PHP і JavaScript:

PHP використовує код JavaScript для створення HTML;

PHP створює HTML-код і вставляє в нього сценарії JavaScript.

Ці сценарії виконуються на стороні клієнта (у браузері користувача).

PHP передає дані в JavaScript;

PHP генерує змінні JavaScript, в які передаються дані, отримані з бази даних або іншого джерела.

JavaScript використовує ці дані для подальшої обробки.

Щоб забезпечити двосторонній зв'язок у реальному часі, PHP ініціює з'єднання, а JavaScript обробляє вхідні повідомлення.

Типовими завданнями, що використовують інтеграцію PHP і JavaScript, є динамічні форми, інтерактивні елементи, редагування вмісту, візуалізація даних.

Інтеграція PHP і JavaScript є ключовим аспектом сучасної веб-розробки. Розуміння їх взаємодії дозволяє створювати динамічні, інтерактивні та зручні веб-програми. Вибір технології залежить від конкретних вимог проекту.

Архітектура MVC є однією з популярних архітектур, яку можна реалізувати за допомогою фреймворків JavaScript. Така модель дозволить ефективно реалізовувати мікросервіси як окремі модулі. Для створення мікросервісів рекомендується використовувати шаблон архітектури MVC (Model-View-Controller) [28].

Така модель дозволяє контролеру виконувати завдання розробки серверної частини шляхом взаємодії з моделлю обробки даних (рис. 3.5). Модель MVC може бути деталізована для фронтенд і бекенд розробки і представлена у вигляді окремих модулів серверної та інтерфейсної частин та взаємодії між ними. Це дозволить більш детально проаналізувати сценарні моделі роботи веб-сервісу і більш логічно розподілити функції між модулями.

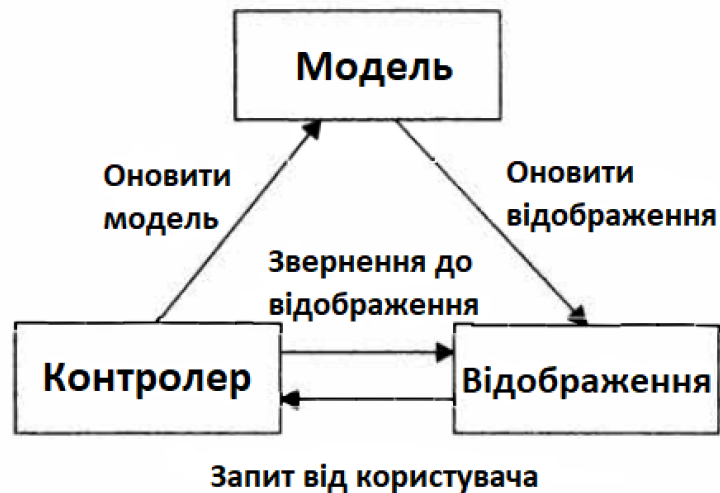


Рисунок 3.5 – Патерн MVC для розробки електронного підручника

Розподіл операційної логіки мікросервісу, даних і представлення дозволяє реалізувати такі процеси:

Користувач запитує необхідні ресурси у сервера.

Запит даних із програми в моделі та передача цих даних у презентацію.

Остаточне форматування даних кінцевого користувача.

Ця модель є стандартом для веб-застосунків. Якщо проаналізувати характеристики інтерфейсної та бекенд-частин, то можна побачити, що інтерфейс відповідає за презентацію, формат дизайну, презентацію динамічних функцій та формат подання інформації.

Бекенд частина бере на себе реалізацію логіки, статичних функцій, наповнення та взаємодію з базою даних, тобто реалізує контролер.

Інтерфейсна та бекенд моделі середовища створення електронних книг показані на рисунку 3.6.

Така модель чітко розділяє фронтенд і бекенд та дозволяє візуально представити зв'язки між ними.

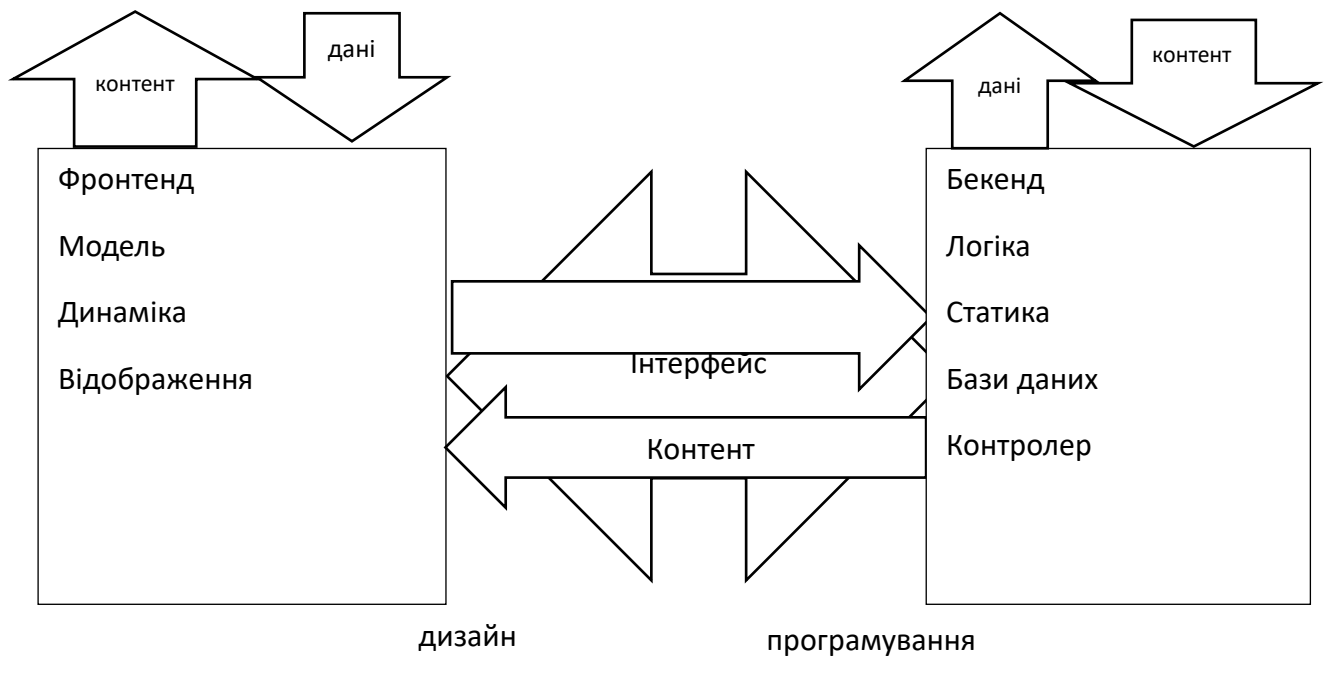


Рисунок 3.6 – Модель фронтенд-бекенд електронного підручника

Для оптимізації роботи вашого коду рекомендується використовувати фреймворки та бібліотеки. Розвиток серверної частини можна представити реалізацією безпосередньо серверних функцій, бази даних та інтерактивного інтерфейсу. Відповідно до концепції клієнт-серверної моделі, сервер відповідає за генерацію запитів і обробку інформації відповідно до запитів.

Бібліотека React [33] використовується для створення інтерфейсної частини. Вона містить функції для параметрів посилань і веб-розмітки. Ця бібліотека використовує спеціальний синтаксис, який містить набір методів і функцій для представлення моделей DOM у Javascript (віртуальний DOM). Модель використовує ієрархічну структуру дерева HTML DOM. Цей синтаксис зручний для користувача та використовує функціональний підхід для створення різних компонентів. Бібліотека – це екосистема, яка діє за принципом від малого

до великого. Для програм є просте модульне тестування, інформаційна підтримка та велика спільнота розробників, підтримка JavaScript і Typescript під час використання React Create App або інших бібліотек React CLI. Реалізація програмних компонентів представлена в додатку.

Інтеграція за допомогою технологій REST, API, JSON дозволяє різним системам безперебійно спілкуватися та обмінюватися даними. REST (Representational State Transfer) – це архітектурний стиль для проектування мережевих програм, JSON (JavaScript Object Notation) – це легкий формат обміну даними. Разом вони утворюють потужну комбінацію для створення гнучких і масштабованих веб-служб. API REST використовують такі методи HTTP, як GET, POST, PUT, DELETE тощо для виконання операцій над ресурсами. Кожен ресурс ідентифікується URI (уніфікований ідентифікатор ресурсу). API REST не мають стану, тобто кожен запит від клієнта містить усю інформацію, необхідну серверу для виконання цього запиту. JSON використовується для форматування даних, які надсилаються та отримуються через REST API. Об'єкти JSON складаються з пар ключ-значення, масивів і вкладених об'єктів.

Кроки для інтеграції полягають в тому, щоб визначитись з кінцевою точкою API, налаштувати автентифікацію, створити та надіслати запити.

Отже, були розроблені основні функціональні модулі для реалізації основних функцій програмного середовища створення та використання електронного підручника.

3.4 Висновки

В третьому розділі розглянуті моделі інформаційної технології електронного підручника для роботи викладача і студента, а також самостійної навчальної діяльності студента. Запропоновані моделі переходу від електронного базового підручника до бази знань на основі електронного підручника.

4. Тестування модуля «Електронний підручник» для системи JetIQ VNTU

4.1 Особливості тестування освітніх веборієнтованих модулів

Тестування веб-системи – це складний процес, який дозволяє виявляти та виправляти помилки в програмному забезпеченні до його випуску. Він є невід’ємною частиною розробки будь-якого веб-проекту та забезпечує його якість, надійність і зручність використання.

Важливо, щоб система керування навчанням (LMS) ефективно працювала на кількох пристроях і з різною швидкістю мережі. Для цього необхідно проаналізувати показники продуктивності, як пам’ять, час відгуку пропускну здатності та переривання, за допомогою тестування навантаження, тестування ємності, стрес-тестування та тестування на витривалість.

По суті, система управління навчанням виконує дві ролі: студент та викладач. Під час тестування функціонального контуру тестувальники повинні пам’ятати про ці дві ролі. Крім того, група контролю якості повинна мати повну документацію щодо вимог до виконання тестування, щоб переконатися, що система управління навчанням відповідає вимогам.

Для перевірки працездатності роботи викладача формуються наступні тестові приклади:

- Робота з контентом підручника;

- Додавання, редагування, структурування контенту;

- Час завантаження контенту;

- Можливості системи моніторингу;

- Можливості завантаження електронного підручника та інших ресурсів на зовнішніх та внутрішніх посиланнях;

Надавати доступ до тестів;

Формувати дати контролю; план навчання;

Інформаційна панель для студентів електронного підручника перевіряється за структурою, роботою всіх кнопок інтерфейсу;

Перевіряється також реакція LMS на різні кліки;

Здатність системи генерувати повідомлення на основі дій користувача.

Тестування мікросервісу системи управління навчанням здійснюється на основі створених кейсів, які базуються на перевірці функціональності, можливості реалізації методів навчання, що входять до моделі формування та використання електронного освітнього середовища. Електронний підручник розроблено з урахуванням особливостей мережевих систем, інтерактивних елементів та вимог універсальної системи JetIQ ВНТУ [1-3]. Протягом усього процесу розробки та тестування можна розробляти або використовувати готові автоматизовані перевірки системи. Це покращує ефективність тестування та забезпечує більш високу якість продукту, який зазвичай використовується в системах управління навчанням, які продаються навчальним закладам і корпоративному навчанню, таке тестування може прискорити вихід на ринок. Автоматизована система тестування – це набір правил і кейсів для створення автоматизованих тестів. Мови та методи автоматизованого тестування узгоджуються під час циклу розробки кожного окремо обслуговуваного проєкту. Якщо мікросервіс має мобільну версію, необхідно провести мобільне тестування на функціональність і зручність використання. Під час тестування мобільних застосунків очікувана функціональність перевіряється на всіх системах і на всіх роздільних здатностях екрана. Необхідно виконати перевірку функцій програмного забезпечення, інтеграції API, графічне тестування та тестування моделювання, з подальшим виправленням застосунку, інтерфейсу користувача.

Для тестування мікросервісу «Електронний інтерактивний підручник» була використана спеціальна інформаційна модель для ролей викладача та студента, яка акцентує увагу на тестуванні всіх функціональних та інтерактивних елементів, контурів відповідно моделей, розроблених в попередніх розділах. Основні цілі тестування можна розділити на: Функціональне тестування передбачає перевірку того, чи відповідає система своїм функціональним вимогам. Таке тестування включає в себе перевірку окремих функцій і сценаріїв користувача. Крім того, необхідно оцінити зручність використання системи. Для тестування електронного підручника були розроблені сценарні кейси.

Кейс1 – перевірка функцій роботи електронного підручника в ролі студента. Завантаження в хмарі. Завантаження на свій комп'ютер. Відкриття сторінок, Формування закладок. Скролінгу сторінок. Відкриття тестів. Проходження тестів.

Кейс 2 – перевірка функцій роботи електронного підручника для викладача. Побудова структури. Завантаження тексту. Редагування тексту. Формування глосарію та посилань. Створення тестів. Створення бібліотеки повідомлень. Створення плану навчання.

Кейс 3 – Перевірка комунікаційного контуру. Отримання завдань від викладача в середовищі електронного підручника (повтор від повідомлення студенту); Отримання повідомлень від автоматизованого модуля. Отримання повідомлень від викладача після автоматичного оцінювання та виконання вправ. Передача повідомлення викладачу. Надання доступу викладачу власних нотаток до підручника. Участь у форумі.

Кейс 4 – Тестування мотиваційних повідомлень – виконується шляхом опитування студентів та оцінювання їх реакцій на повідомлення. В межах магістерської роботи виконується тільки перевірка формування повідомлень.

Загальна перевірка та оцінювання може бути здійснена тільки після повного впровадження.

4.2 Тестування інформаційної технології створення електронного підручника для системи управління навчанням, підтримки методичної, наукової та управлінської діяльності JetIQ VNTU

Перевірка роботи мікросервісу «Електронна підручник» виконано за контурами.

Перший контур – функціональний. Перевіряємо роботу модулів в ролі викладача. На рис. 4.1 представлено середовище створення електронних підручників. Викладач має можливість створити різні види електронних підручників з можливістю навігації, різних розділів, посилань, автопублікації в системі JetIQ та можливості зберігання як локального ресурсу на зовнішніх носіях.

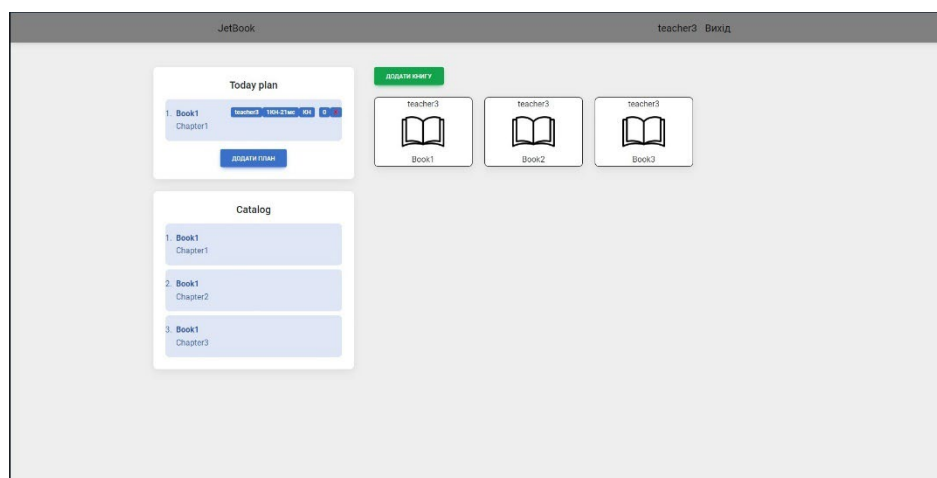
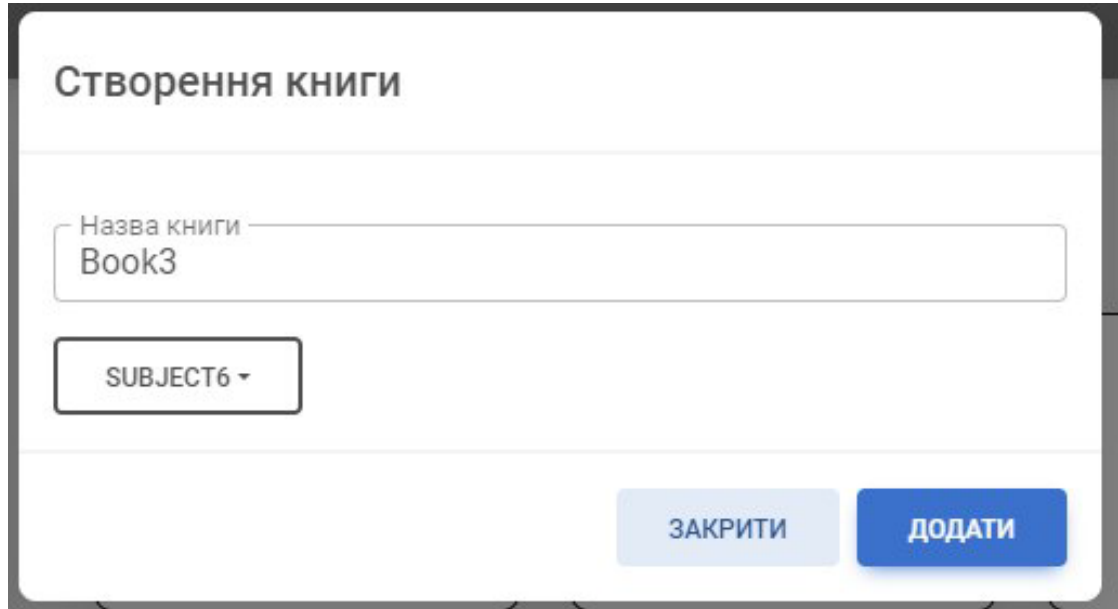


Рисунок 4.1 – Вікно створення електронного підручника

Вікно створення електронного підручника з визначенням назви представлено на рис. 4.2



The screenshot shows a web interface for creating a book. At the top, the title 'Створення книги' is displayed. Below it is a text input field labeled 'Назва книги' (Book Name) containing the text 'Book3'. Underneath the input field is a dropdown menu labeled 'SUBJECT6' with a downward arrow. At the bottom right, there are two buttons: a light blue button labeled 'ЗАКРИТИ' (Close) and a dark blue button labeled 'ДОДАТИ' (Add).

Рисунок 4.2 – Вікно створення книги

Відповідно до структурованих параграфів, викладач може створити вбудовані тести, а також вказати посилання на загальні тести в системі JetIQ. Використання вбудованих тестів рекомендується не тільки для оперативного контролю знань за вивченим навчальним контентом електронної книги, а і для використання електронної книги поза середовища системи JetIQ.

На рис. 4.3 представлено вікно формування ієрархії розділів електронного підручника.

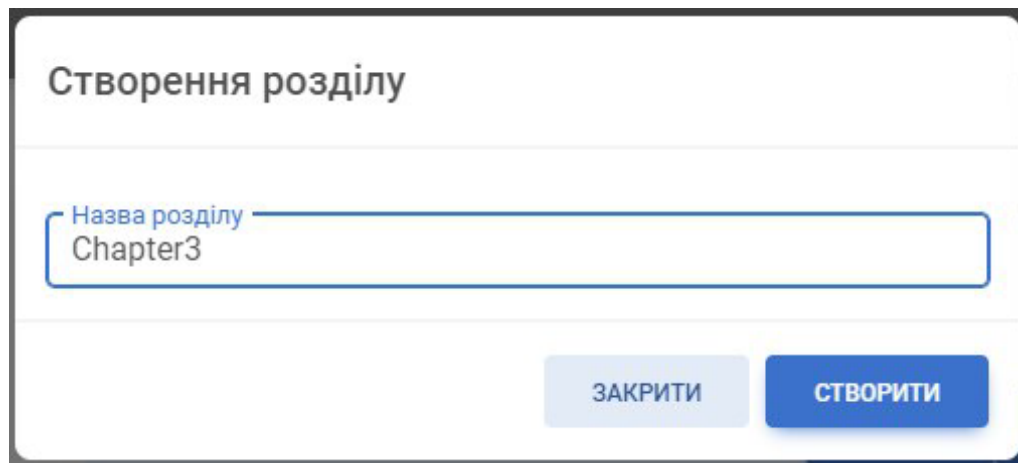


Рисунок 4.3 – Вікно створення розділів за ієрархією

На рисунку 4.4 подано процес формування плану. Аналогічний план створюється також для навчання студента.

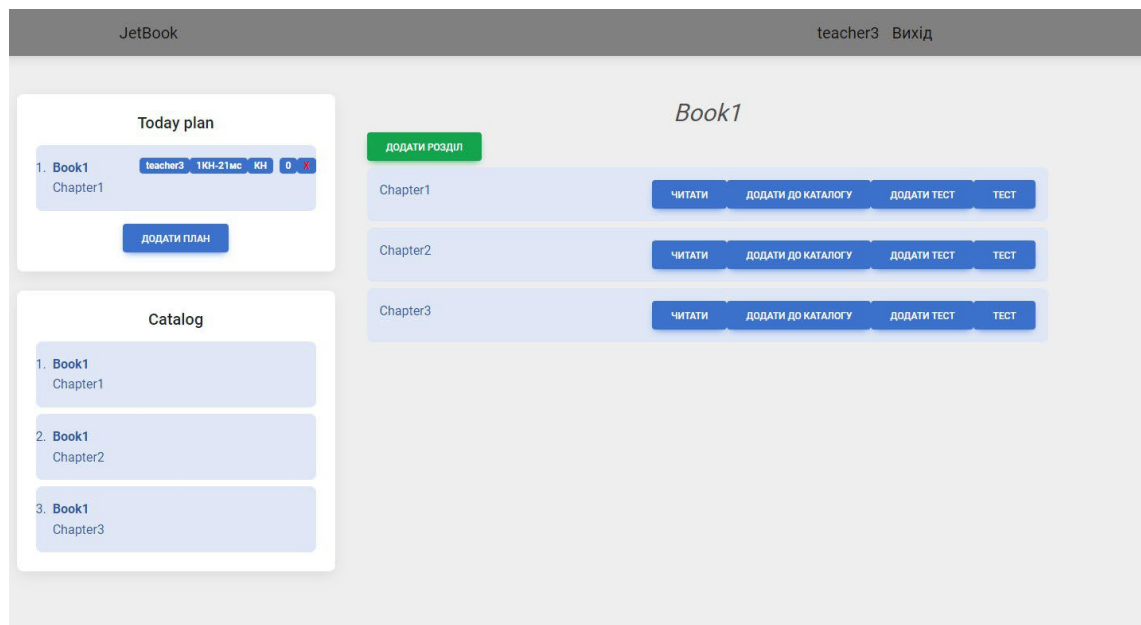


Рисунок 4.4 – Вікно створення розділів книги

На рисунку 4.5 представлені процеси створення контенту електронної книги для конкретної дисципліни та спеціальності. Також визначені дати плану

навчання.

2. **Book1** **teacher3** **1KH-21mc** **КН** **0** **X**
Chapter1

Створення плану

BOOK1 ▾

CHAPTER1 ▾

КН ▾

1KH-21MC ▾

ЗАКРИТИ СТВОРИТИ

Рисунок 4.5 – Вікно плану електронної книги

На рис. 4.6 подано результати визначення дати виконання тесту.

Створення тесту

Питання

ДОДАТИ ВАРІАНТ ВІДПОВІДІ

06/13/2023

What is your favorite day?

< June 2023 >

Su	Mo	Tu	We	Th	Fr	Sa
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1

ЗАКРИТИ СТВОРИТИ

Створення тесту

Питання

ДОДАТИ ВАРІАНТ ВІДПОВІДІ

06/13/2023

ЗАКРИТИ СТВОРИТИ

Рисунок 4.6 – Вікна визначення дат виконання тестів

На рис. 4.7 надано вигляд вікна для створення тесту. Методичні інструкції рекомендують використовувати мікротести як закриті, так і відкриті. Ефективним для електронного підручник є тести розуміння термінів та понять.

Рисунок 4.7 – Вікно створення тесту в середовищі електронного підручника

Для викладача також є вікно пошуку в каталозі електронних книг. За аналогією доступу до репозиторію та тестів, викладач може надати доступ для співавторів або інших викладачів для використання його підручника.

На рис. 4.8 подано вигляд вікна пошуку в каталозі електронних підручників.



Рисунок 4.8 – Вікно пошуку за назвами та авторами електронних підручників

Електронний підручник перетворюється в базу знань в процесі додавання нотаток викладача та студента. Студент може надати доступ до своїх нотаток та закладок. Нотатки свідчать про активну роботу студента з текстом та іншими об'єктами електронного підручника. На рис. 4.9 представлено вікно формування нотаток.

Перевірка взаємодії викладача і студента підтверджує ще один вид комунікації – робота з нотатками до тем електронного підручника, оновлення даних електронного підручника, додавання різноманітних зовнішніх посилань, які не були актуальні під час його написання та завантаження, обговорення власного досвіду за темами підручника студента, викладача та інших студентів.

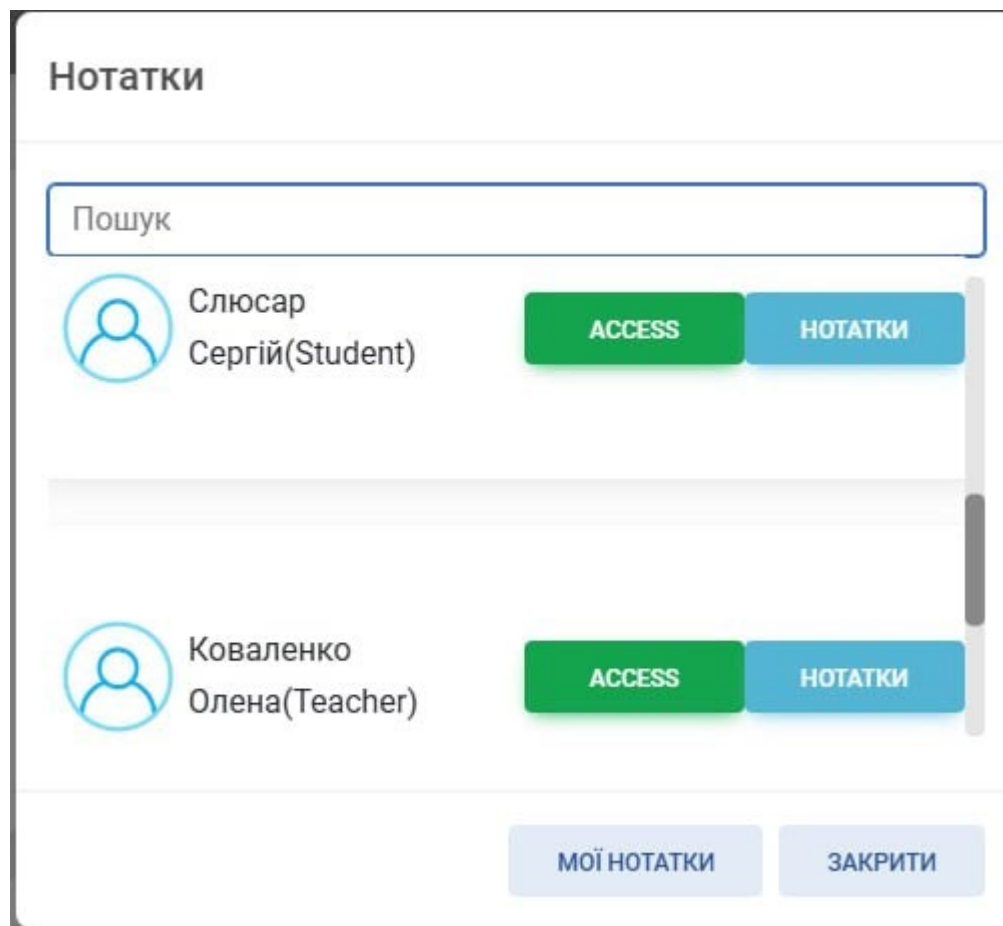


Рисунок 4.9 – Формування нотаток і надання доступу викладачу до нотаток

Комунікаційний контур також перевіряється за допомогою визначення часу отримання повідомлення від викладача з планом навчання, дати виконання вправи, вивчення матеріалу, виконання тесту. На рис. 4.10 подано вид процесу додавання нотаток. Такі відкриті нотатки викладач може використовувати і для планування, і для акцентування важливих питань, тем, тестів, дат виконання. Але, викладач повинен вибрати модель – нотатки для конкретного плану навчання з групою, або нотатки для електронного підручника для всіх груп. Тоді база знань буде формуватись для конкретної групи (що є затратним по часу, але більш персоналізованим до рівня знань студентів та їх зацікавленості) або це буде електронний підручник зі змінами і посиланнями за визначеними темами, які

відбулись за час використання підручника від моменту завантаження.

Нотатки

Текст

Розділ 1 - Типи массивів JavaScript:

15 сторінка - Типи циклів у Javascript: ...

-

-

ДОДАТИ ЗАКРИТИ

Рисунок 4.10 – Вікно додавання нотаток

Комунікації також здійснюються шляхом визначення дати вконання вправ і тестів. Так, на рис. 4.11 представлено вікно нагадування щодо виконання тесту. Таке повідомлення носить функціональний характер. Емоційний характер йому додають знаки оклику. Викладач самостійно може сформувати бібліотеку повідомлень і використовувати її потім для передачі студентам.

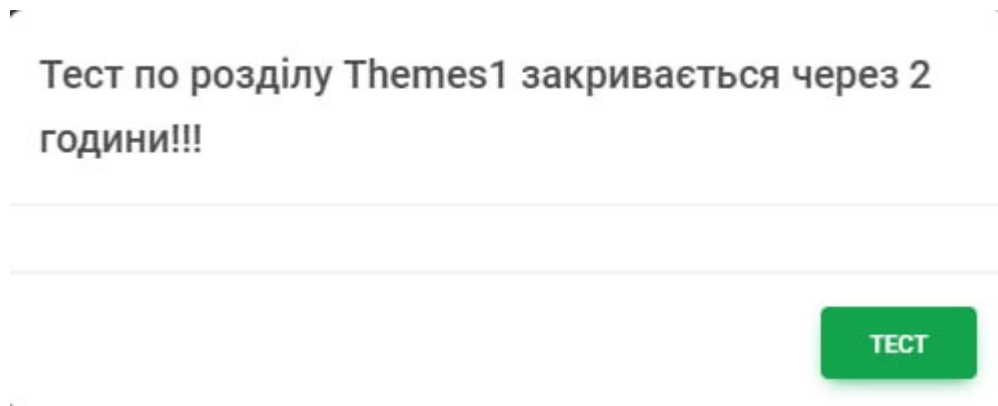


Рисунок 4.11 – Вікно нагадування щодо виконання тесту

Так, наприклад, при перевірці швидкості гортання сторінок, можна виставити час (2 хвилини на сторінку) і, якщо гортання здійснюється дуже швидко зі стартової сторінки, або сторінки, коли закінчили попередній сеанс читання, то студент отримує достатньо емоційне повідомлення і може бути заблокований, поки не пройде тест. Викладач самостійно приймає рішення щодо блокування в такому випадку студента. Але, якщо він визначає це, то студент далі не може вивчати матеріал електронного підручника.

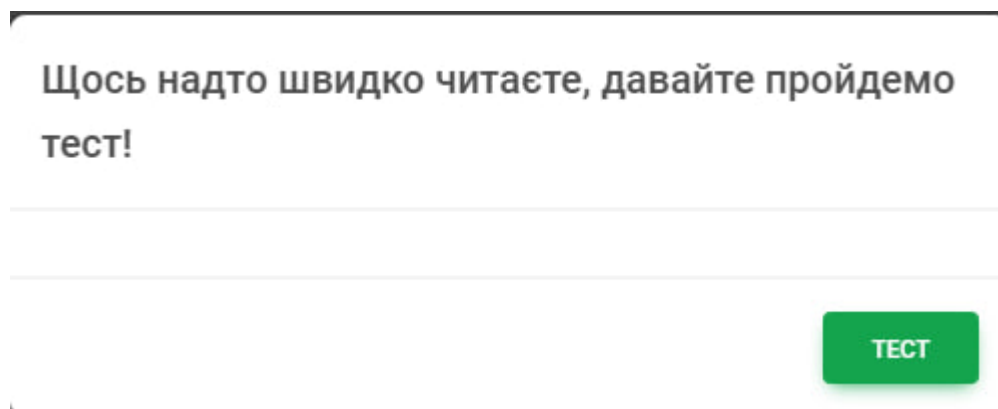


Рисунок 4.12 – Емоційне повідомлення на швидке гортання тексту

Отже, отримані результати свідчать про відповідність функцій створеної інформаційної технології, роботи всіх трьох контурів – функціонального, комунікаційного, емоційно-мотиваційного. Заявлений в загальній моделі контур безпеки працює в системі управління навчанням при авторизації та може бути введений викладачем в електронний підручник інструментами роботи з pdf файлами.

Запропонована інформаційна технологія є основою пілотного проєкту для впровадження в систему JetIQ VNTU.

4.3 Пілотний проєкт впровадження інформаційної технології створення електронного підручника в системі JetIQ

Впровадження інформаційної технології створення електронного інтерактивного підручника у систему управління навчанням (LMS), а саме в систему JetIQ VNTU, дозволить удосконалити процеси роботи студента на заняттях і, саме головне, під час самостійної роботи.

Визначимо цілі пілотного проєкту:

- Забезпечити більш ефективне засвоєння матеріалу завдяки інтерактивним елементам і можливостям персоналізації.
- Збільшити рівень доступності навчальних матеріалів та структурованості інформації з автоматизованою підтримкою;
- Оптимізувати роботу викладачів зі студентами, особливо студентами, що навчаються на індивідуальному графіку
- Підвищення активності студентів за рахунок формування процесу навчання в інтерактивному середовищі

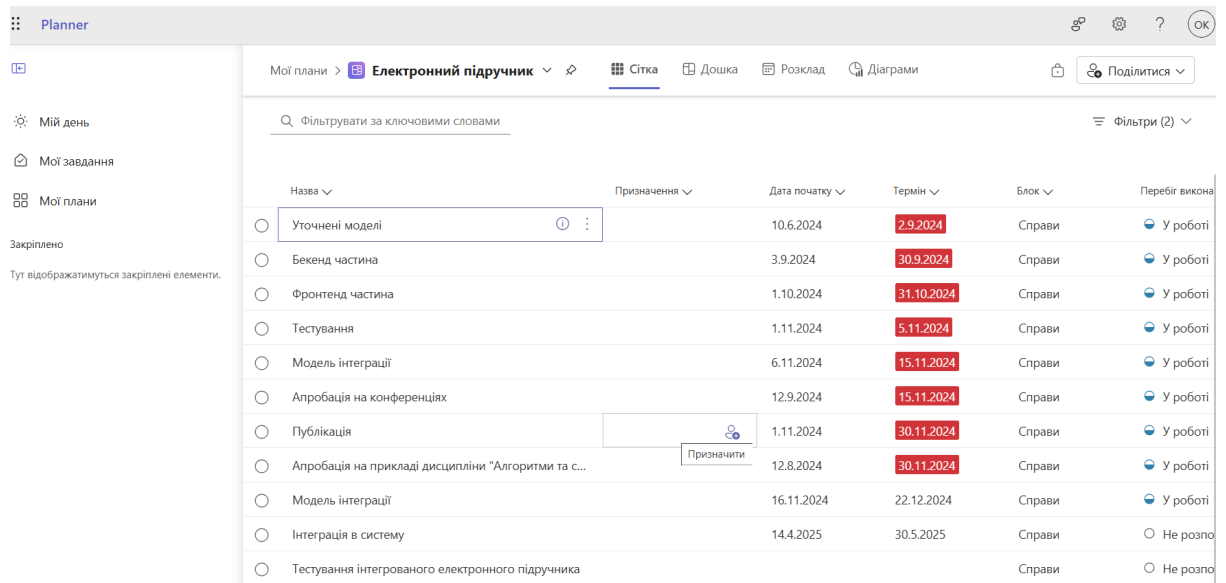
- Створення дистанційних курсів з використанням електронних підручників, навігаторів та інших модулів системи.

На рисунку 4.13 наведено план пілотного проєкту для ВНТУ.

На даний момент виконано базову реалізацію електронного підручника на основі макетів електронних книг, адаптовано бекенд-частину під особливості JetIQ VNTU, виконано перевірку моделей на базовій версії електронного підручника та веб-сервісах системи управління навчанням.

Розроблена інформаційна технологія та програмне забезпечення є основою для інтегрованого інтерактивного електронного підручника.

Реалізація інформаційної технології виконана на прикладі дисципліни «Алгоритми та структури даних».



Назва	Призначення	Дата початку	Термін	Блок	Перебіг викона
Уточнені моделі		10.6.2024	2.9.2024	Справи	У роботі
Бекенд частина		3.9.2024	30.9.2024	Справи	У роботі
Фронтенд частина		1.10.2024	31.10.2024	Справи	У роботі
Тестування		1.11.2024	5.11.2024	Справи	У роботі
Модель інтеграції		6.11.2024	15.11.2024	Справи	У роботі
Апробація на конференціях		12.9.2024	15.11.2024	Справи	У роботі
Публікація		1.11.2024	30.11.2024	Справи	У роботі
Апробація на прикладі дисципліни "Алгоритми та с..."	Призначити	12.8.2024	30.11.2024	Справи	У роботі
Модель інтеграції		16.11.2024	22.12.2024	Справи	У роботі
Інтеграція в систему		14.4.2025	30.5.2025	Справи	Не розпо
Тестування інтегрованого електронного підручника				Справи	Не розпо

Рисунок 4.13 – Пілотний проєкт розробки та впровадження інформаційної технології створення та використання електронного підручника в системі JetIQ

Microsoft Planner – це інструмент керування завданнями та планування проєктів, який є частиною пакету Microsoft 365. Він надає командам простий і наочний спосіб організації роботи, відстеження прогресу та ефективної

співпраці. Користувачі можуть створювати плани, будувати завдання, призначати їх членам команди, встановлювати терміни та оновлювати статуси. Planner пропонує макет дошки для завдань, подібний до дошки Kanban, де завдання представлені у вигляді карток, які можна переміщати по стовпцях, щоб вказати прогрес. Цей візуальний підхід дозволяє легко побачити з першого погляду, які завдання виконуються, завершені або потребують уваги. Для пілотного проєкту запровадження інформаційної технології створення та використання електронного підручника створено план проєкту, працює дошка виконання завдань. Результати магістерської роботи будуть впроваджені в систему після перевірки та інтеграції. На рис. 4.14 представлено дошку та діаграму процесів виконання робіт.

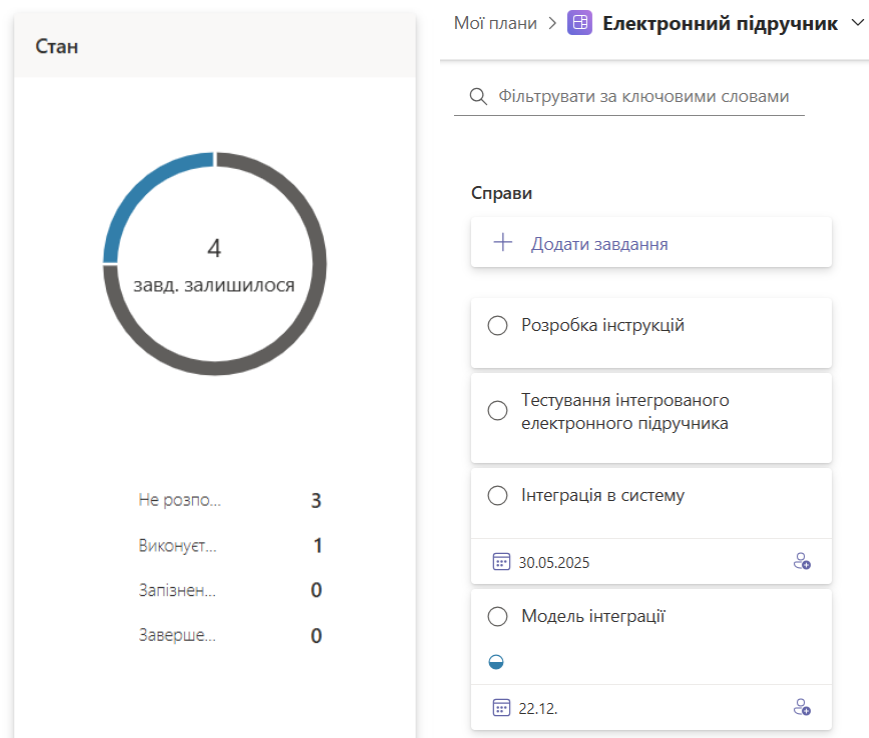


Рисунок 4.14 – Інструменти управління пілотним проєктом в Microsoft Planer

Виконані завдання прийняті до подальшої роботи фахівцями центру диджиталізації освітнього процесу ВНТУ. Інструмент легко інтегрується з іншими програмами Microsoft 365, такими як Teams, Outlook і SharePoint, створюючи згуртоване середовище для командної співпраці. Наприклад, у Microsoft Teams до каналу можна додати вкладку «Планувальник», що дозволить членам команди переглядати та оновлювати завдання, не виходячи з інтерфейсу Teams. Ця інтеграція допомагає спростити спілкування та гарантує, що всі залишаються на одній сторінці.

Planner розроблено таким чином, щоб бути зручним для користувача, що робить його доступним для різних користувачів, від невеликих команд до великих організацій. Це дозволяє класифікувати завдання за допомогою міток, налаштувань пріоритетів і контрольних списків, що дозволяє командам налаштовувати своє планування відповідно до конкретних потреб. Крім того, Planner пропонує функції звітування, які допомагають командам аналізувати рівень виконання завдань і загальну продуктивність, надаючи цінну інформацію для підвищення ефективності робочого процесу.

Отже, Microsoft Planner – це універсальний інструмент, який допомагає командам організовувати завдання, ефективно співпрацювати та підвищувати продуктивність завдяки візуально інтуїтивно зрозумілому інтерфейсу та бездоганній інтеграції з іншими програмами Microsoft 365. А інформаційна система JeIQ VNTU має можливість надавати викладачам та студентам доступ до версії для університетів Microsoft 365. Тому управління цим пілотним проєктом може бути здійснено через даний програмний продукт.

4.4 Висновки

В четвертому розділі магістерської роботи представлено аналіз особливостей тестування веб-орієнтованих електронних підручників. Прийнято рішення щодо тестування інформаційної технології за контурною моделлю та основними контурами – функціональний, комунікаційний, емоційно-мотиваційний. Такий підхід дозволяє перевірити роботу всіх функцій електронного підручника, а також акцентувати увагу на здійсненні комунікацій та формуванні різноманітних повідомлень як автоматично в системі моніторингу, так і викладачем і студентами. Моделі інформаційної технології за допомогою веб-сервісів системи JetIQ VNTU були апробовані на міжнародних та всеукраїнських конференціях, реалізовані під час викладання дисципліни «Алгоритми та структури даних». Крім того, сформовано пілотний проєкт, початкові завдання якого були виконання в межах магістерської роботи. Базова версія електронного підручника перевірена та прийнята до запровадження центром диджиталізації освітнього процесу ВНТУ.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Проведення наукового аудиту науково-дослідної роботи

Ступінь новизни даної науково-дослідної роботи – Нова, оскільки було проведено суттєве вдосконалення системи JetIQ.

Показник, який характеризує науковий ефект, визначається за формулою:

$$E_{\text{нау}} = 0,6 \cdot k_{\text{нов}} + 0,4 \cdot k_{\text{теор}},$$

Підставивши наші значення у формулу отримуємо такі результати:

$$E_{\text{нау}} = 0,6 \cdot 60 + 0,4 \cdot 80 = 68,$$

Отже рівень наукового ефекту – Середній.

При аналізі науково-дослідної роботи на тему «Інформаційна технологія створення електронних підручників в системі JetIQ», була проведена оцінка ефективності роботи, де були виявлені такі переваги:

- Підвищення продуктивності викладачів за рахунок автоматизації обміну інформації за студентами;
- Зменшення часу, витраченого на організаційні питання, завдяки можливостям платформи;
- Оптимізація комунікаційних процесів та інформування студентів у реальному часі.

За рахунок цих переваг і було досягнуто такий рівень наукового ефекту.

5.2 Проведення комерційного та технологічного аудиту науково-технічної розробки

Таблиця 5.1 - Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки

Критерії	Коваленко	Сілагін	Паламарчук
	О.О.	О.В.	Є.А.
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	5	5	5
2. Ринкові переваги (наявність аналогів)	3	3	3
3. Ринкові переваги (ціна продукту)	5	4	4
4. Ринкові переваги (технічні властивості)	4	4	5
5. Ринкові переваги (експлуатаційні витрати)	5	5	5
6. Ринкові перспективи (розмір ринку)	3	4	3
7. Ринкові перспективи (конкуренція)	4	4	4
8. Практична здійсненність (наявність фахівців)	5	5	5
9. Практична здійсненність (наявність фінансів)	5	5	4
10. Практична здійсненність (необхідність нових матеріалів)	5	5	5
11. Практична здійсненність (термін реалізації)	4	5	5
12. Практична здійсненність (розробка документів)	5	5	5
Сума балів	53	54	53
Середньоарифметична сума балів СБс	53,33		

За результатами розрахунків, наведених в таблиці 2.5, робиться висновок що науково-технічний рівень та комерційний потенціал розробки – Високий.

Такий рівень було досягнуто за рахунок:

- суттєве покращення соціальних показників розвитку суспільства;
- суттєве зростання продуктивності, ефективності нової науково-технічної розробки

5.3 Розрахунок витрат на здійснення науково-дослідної роботи

5.3.1 Витрати на оплату праці

Витрати на основну заробітну плату дослідників (Z_o) розраховують відповідно до посадових окладів працівників, за формулою:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p},$$

Підставимо наші значення у таблицю 2.2

Таблиця 5.2 - Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Кількість днів роботи конкретного дослідника, дн.	Середня кількість робочих днів в місяці	Витрати на заробітну плату, грн
Керівник проекту	15000	21	23	13695
Науковий співробітник	17000	22	23	16260
Науковий співробітник	20000	21	23	18260
Лаборант	8000	23	23	8000
Всього	56215			

Витрати на основну заробітну плату робітників (Z_p) за відповідними найменуваннями робіт розраховують за формулою:

$$Z_p = \sum_{i=1}^n C_i \cdot t_i = 103,61 \cdot 240 = 24866 ,$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника на виконання певної роботи, год

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_P \cdot t_{зм}} = \frac{8000 \cdot 1,36 \cdot 1,8}{21 \cdot 9} = 103,61,$$

де M_M – розмір прожиткового мінімуму працездатної особи або мінімальної місячної заробітної плати (залежно від діючого законодавства), грн, Приймаємо 8000;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б), Приймаємо 1,36;

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати. (табл. Б.1, додаток Б), Приймаємо 1,8

T_p – середня кількість робочих днів в місяці, приблизно $T_p = 21 \dots 23$ дні,
Приймаємо 21;

$t_{зм}$ – тривалість зміни, год, Приймаємо 9.

Таблиця 5.3 - – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Розробка веб- сайту	240	5	1,36	103,61	24866
Всього	24866				

Додаткова заробітна плата розраховується як 10- 12% від суми основної заробітної плати дослідників та робітників:

$$Z_{\text{дод}} = \frac{24866 + 56215}{10} = 8\,108,1 ,$$

5.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховується як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$Z_n = \frac{(24866 + 56215 + 8\,108,1) \cdot 22}{100} = 19\,621,602 ,$$

5.3.3 Програмне забезпечення для наукових (експериментальних) робіт

Балансову вартість програмного забезпечення розраховують за формулою:

$$B_{\text{прг}} = \sum_{i=1}^k \text{Ц}_{i \text{ прг}} \cdot C_{\text{прг } i} \cdot K_i = 1200 \cdot 1 \cdot 1,1 = 1320,$$

де $\text{Ц}_{i \text{ прг}}$ – ціна придбання одиниці програмного засобу цього виду, грн,
Приймаємо 1200 грн;

$C_{\text{прг } i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт., приймаємо 1;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1,10 \dots 1,12$), приймаємо 1,1;

k – кількість найменувань програмних засобів, приймаємо 1.

Таблиця 5.4 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
WebStorm	1	1200	1320
Всього	1320		

5.3.4 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо можуть бути розраховані з використанням прямолінійного методу амортизації за формулою:

$$A_{\text{обл}} = \frac{\text{Ц}_6}{T_{\text{в}}} \cdot \frac{t_{\text{вик}}}{12} = \frac{20000}{2} \cdot \frac{1}{12} = 833,33,$$

де Ц_6 – балансова вартість обладнання, програмних засобів, приміщень

тощо, які використовувались для проведення досліджень, грн, приймаємо 20000;

$t_{\text{вик}}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців, приймаємо 1;

$T_{\text{в}}$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років, приймаємо 2.

Таблиця 5.5 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Комп'ютер	20000	2	1	833,33
Всього	833,33			

5.3.5 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію (B_e) розраховують за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{\text{впі}}}{n_i} = \frac{0,5 \cdot 240 \cdot 4,32 \cdot 0,4}{0,3} = 691,2,$$

де W_{yi} – встановлена потужність обладнання на певному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії);

$K_{\text{впі}}$ – коефіцієнт, що враховує використання потужності, $K_{\text{впі}} < 1$;

n_i – коефіцієнт корисної дії обладнання, $n_i < 1$.

Таблиця 5.5 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Комп'ютер	0,5	240	691,2
Всього	691,2		

5.3.6 Інші витрати

Витрати за статтею «Інші витрати» розраховуються як 50...100% від суми основної заробітної плати дослідників та робітників за формулою:

$$I_{\text{в}} = (3_{\text{o}} + 3_{\text{р}}) \cdot \frac{H_{\text{ів}}}{100\%} = (24866 + 56215) \cdot 0,5 = 40\,540,5,$$

де $H_{\text{ів}}$ – норма нарахування за статтею «Інші витрати»

5.3.7 Накладні (загальновиробничі) витрати

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуються як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$V_{\text{нзв}} = (3_{\text{o}} + 3_{\text{р}}) \cdot \frac{H_{\text{нзв}}}{100\%} = (24866 + 56215) \cdot 1 = 81\,081,$$

де $H_{\text{нзв}}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати».

Витрати на проведення науково-дослідної роботи розраховуються як сума всіх попередніх статей витрат.

$$V_{\text{заг}} = 691,2 + 833,33 + 1320 + 81\,081 + 40\,540,5 + 19\,621,602 + 8\,108,1 + 56215 + 24866 = 233\,276,732,$$

Загальні витрати ЗВ на завершення науково-дослідної (науково-технічної) роботи та оформлення її результатів розраховуються за формулою:

$$3B = \frac{B_{\text{заг}}}{n} = \frac{233\,276,732}{0,9} = 259\,196,37,$$

де η – коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи. Так, якщо науково-технічна розробка знаходиться на стадії: науково-дослідних робіт, то $\eta = 0,1$; технічного проектування, то $\eta = 0,2$; розробки конструкторської документації, то $\eta = 0,3$; розробки технологій, то $\eta = 0,4$; розробки дослідного зразка, то $\eta = 0,5$; розробки промислового зразка, то $\eta = 0,7$; впровадження, то $\eta = 0,9$.

5.4 Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором

Розрахунок економічної ефективності науково-технічної розробки за її можливої комерціалізації потенційним інвестором визначається за формулою:

$$\begin{aligned} \Delta\P_i &= (\pm\Delta\P_0 \cdot N + \Pi_0 \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right) = \\ &= (5000 \cdot 20 + (259\,196,37 + 5000) \cdot 2) \cdot 0,8333 \cdot 0,35 \cdot \left(1 - \frac{18}{100}\right) = \\ &= 150\,230,48 \text{ грн} \end{aligned}$$

де $\pm\Delta\P_0$ – зміна основного якісного показника від впровадження результатів науково-технічної розробки в аналізованому році. Зазвичай, таким показником може бути зміна ціни реалізації одиниці нової розробки в аналізованому році (відносно року до впровадження цієї розробки); $\pm\Delta\P_0$ може мати як додатне, так і від’ємне значення (від’ємне – при зниженні ціни відносно року до впровадження цієї розробки, додатне – при зростанні ціни), Приймаємо 5000 грн;

N – основний кількісний показник, який визначає величину попиту на

аналогічні чи подібні розробки у році до впровадження результатів нової науково-технічної розробки, приймаємо 20;

Π_0 – основний якісний показник, який визначає ціну реалізації нової науково-технічної розробки в аналізованому році, $\Pi_0 = \Pi_6 \pm \Delta\Pi_0$;

Π_6 – основний якісний показник, який визначає ціну реалізації існуючої (базової) науково-технічної розробки у році до впровадження результатів, приймаємо загальні витрати на роботу – 259 196,37 грн;

ΔN – зміна основного кількісного показника від впровадження результатів науково-технічної розробки в аналізованому році. Зазвичай таким показником може бути зростання попиту на науково-технічну розробку в аналізованому році (відносно року до впровадження цієї розробки), приймаємо 2;

λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2021 році ставка податку на додану вартість становить 20%, а коефіцієнт $\lambda = 0,8333$, приймаємо 0,8333;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту (послуги). Рекомендується брати $\rho = 0,2 \dots 0,5$, приймаємо 0,35;

θ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2021 році $\theta = 18\%$.

Розраховування приведеної вартості збільшення всіх чистих прибутків ПП, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки визначається за формулою:

$$\begin{aligned} \Pi\Pi &= \sum_{i=1}^T \frac{\Delta\Pi_i}{(1+\tau)^t} = \frac{150\,230,48}{1,05} + \frac{150\,230,48}{1,1025} + \frac{150\,230,48}{1,157} + \frac{150\,230,48}{1,21485} \\ &\quad + \frac{150\,230,48}{1,27} + \frac{150\,230,48}{1,339} + \frac{150\,230,48}{1,4} + \frac{150\,230,48}{1,47} \\ &= 987\,997,004 \end{aligned}$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному з років, протягом яких

виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau = 0,05 \dots 0,15$;

t – період часу (в роках) від моменту початку впровадження науковотехнічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

Розраховування величини початкових інвестицій розраховуються за формулою:

$$PV = k_{\text{інв}} \cdot 3B = 2 \cdot 259\,196,37 = 518\,392,74$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію. Це можуть бути витрати на підготовку приміщень, розробку технологій, навчання персоналу, маркетингові

заходи тощо; зазвичай $k_{\text{інв}} = 2 \dots 5$, але може бути і більшим;

$3B$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, грн.

Абсолютний економічний ефект $E_{\text{абс}}$ або чистий приведений дохід (NPV, Net Present Value) для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{\text{абс}} = \text{ПП} - PV = 987\,997,004 - 518\,392,74 = 469\,604,264$$

де ПП – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, грн, приймаємо 493 856;

PV – теперішня вартість початкових інвестицій, грн, приймаємо 301 395,56.

Внутрішня економічна дохідність інвестицій E_v , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науковотехнічної розробки, розраховується за формулою:

$$E_v = \sqrt[T_{ж}]{1 + \frac{E_{абс}}{PV}} - 1 = \sqrt[10]{1 + \frac{469\,604,264}{518\,392,74}} - 1 = 1,38 - 1 = 0,38$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, грн;

PV – теперішня вартість початкових інвестицій, грн;

$T_{ж}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, роки, приймаємо 10.

Далі визначаємо бар'єрну ставку дисконтування мін τ , тобто мінімальну внутрішню економічну дохідність інвестицій, нижче якої кошти у впровадження науково-технічної розробки та її комерціалізацію вкладатися не будуть.

Мінімальна внутрішня економічна дохідність вкладених інвестицій мін τ визначається за формулою:

$$\tau_{\min} = d + f = 0,1 + 0,05 = 0,15$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2020 році в Україні $d = 0,09 \dots 0,12$;

f – показник, що характеризує ризикованість вкладення інвестицій; зазвичай величина $f = 0,05 \dots 0,5$, але може бути і значно вищою.

Далі розраховуємо період окупності інвестицій Ток (DPP, Discounted Payback Period), які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_v} = \frac{1}{0,38} = 2,63$$

де E_v – внутрішня економічна дохідність вкладених інвестицій.

ВИСНОВКИ

Тема дослідження є актуальною для різних освітніх закладів, які використовують системи управління навчанням та електронні підручники. На сьогоднішній день за Законом про освіту, кожен паперовий підручник повинен бути доповнений електронним підручником. Але це не тільки подання в електронному вигляді тексту та зображень, а розвиток інтерактивних елементів, нових інформаційних технологій, які поєднують в собі педагогічний дизайн – правила роботи з освітніми електронними ресурсами, технічні аспекти інтерактивності та інтеграції.

Запропонована інформаційна технологія базується на методі, що використовує контурну модель і дозволяє реалізувати електронний підручник як веб-середовище з інтерактивними елементами та з'єднаними сервісами підручника і системи управління навчанням.

Електронні підручники є основою самостійної роботи студентів, вони можуть бути використані не тільки при підключенні Інтернет, а і завантажені як локальний ресурс з подальшою публікацією результатів в системі. Запропонована контурна модель складається із функціонального, комунікаційного та емоційного контурів.

В роботі виконані такі завдання:

Проаналізовані потреби в електронних підручниках та розробки інформаційної технології для їх створення; основні поняття інформаційної технології створення електронних підручників; аналоги програмного середовища для створення електронного підручника;

виконана постановка задачі для створення інформаційної структури електронного підручника;

розроблені інформаційна технологія для створення електронного підручника; моделі для реалізації інформаційної технології створення електронного підручника;

здійснена програмна реалізація мікросервісу «Електронний підручник»; виконано тестування інформаційної технології створення електронного підручника.

Наукова новизна розробки полягає в тому, що запропоновано інформаційну технологію створення електронного підручника на основі принципів педагогічного дизайну та контурної моделі електронного середовища, яка має вбудовані інструменти підтримки користувача щодо виконання завдань для самостійної роботи, виконання мікротестів до визначених тем, формування бази знань для конкретної дисципліни і поточного користувача .

Також удосконалено інформаційну технологію створення електронного підручника для систем управління навчання, яка дозволяє використовувати електронний підручник як самостійний застосунок з активною взаємодією всіх агентів системи управління навчанням, а також як вбудований інструмент дистанційного та змішаного курсу з визначеної дисципліни, що сприяє покращенню результатів засвоєння та використання знань з дисципліни.

Практична цінність полягає у запровадженні електронного підручника в системі JetIQ, оновлення інструкцій та апробацію електронної книги з дисципліни «Алгоритми та структури даних».

Апробацію результатів здійснено на конференціях ВНТУ, інтернет-конференціях. За темою досліджень опубліковано тези доповідей; фахова стаття в журналі категорії Б.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про затвердження Положення про електронний підручник.
<https://zakon.rada.gov.ua/go/z0621-18>.
2. Програма діяльності Кабінету Міністрів України. Червень, 2020 р. С. 79-85. URL: <https://www.kmu.gov.ua/storage/app/sites/1/Program/diyalnosti-kmu-20.pdf>.
3. Сілагін О.В., Власенко Д.В. Smart елементи електронного підручника. Матеріали [ЛІІ Науково-технічна конференція факультету інтелектуальних інформаційних технологій та автоматизації \(2023\)](#). URL: <https://conferences.vntu.edu.ua/index.php/all-fksa/all-fksa-2023>
4. Коваленко О.О., Барчук Н.Є., Власенко В.В. Аналіз інструментів для створення електронної книги. Матеріали Всеукраїнської наукової конференції «Молодь в науці». 2023. URL: <https://conferences.vntu.edu.ua/index.php/mn/mn2023/schedConf/>
5. Коваленко О.О., Гаврилюк О.В., Власенко Д.В. Особливості використання сервісів та веб-програмування для створення електронної книги. Матеріали Всеукраїнської наукової конференції «Молодь в науці». 2023. URL: <https://conferences.vntu.edu.ua/index.php/mn/mn2023/schedConf/>
6. Коваленко О.О., Власенко В.В., Пилипенко Д.Ю. Особливості тестування інтерактивного електронного підручника *Електронні інформаційні ресурси: створення, використання, доступ та управління*. Збірник матеріалів Міжнародної науково-практичної Інтернет конференції 20-21 листопада 2024 р. Суми/Вінниця: НІКО / КЗВО «Вінницька академія безперервної освіти», 2024. С.73-74
7. Коваленко О.О., Власенко Д.В., Роботько Д.О. Електронна книга як система знань *Електронні інформаційні ресурси: створення, використання,*

доступ та управління. Збірник матеріалів Міжнародної науково-практичної Інтернет конференції 20-21 листопада 2024 р. Суми/Вінниця: НІКО / КЗВО «Вінницька академія безперервної освіти», 2024. С.74-75

8. Коваленко О., Сілагін О., Власенко Д. Контурні моделі середовища електронної книги в системі 'JETIQ VNTU', *Опт-ел. інф-енерг. техн.*, вип. 48, вип. 2, с. 242–248.

9. Електронні підручники – Освітня технологія "Росток".
<https://rostok.org.ua/medias/elektronni-pidruchniki/>.

10. Найпотрібніше програмне забезпечення для створення електронних книг. Adobe indesign. URL: <https://www.adobe.com/ua/products/indesign/ebook-creator-software.html>

11. Редактор електронних книг Jetbook URL: <http://surl.li/hvisk>

12. Морзе Н.В., Глазунова О.Г., Мокрієв М.В. Методика створення електронного навчального курсу (на базі платформи дистанційного навчання Moodle 3): Навчальний посібник. К.: 2016. 240 с.

13. Веб-посібники ВНТУ URL: <https://web.posibnyky.vntu.edu.ua/>

14. Weber G., Brusilovsky P. ELM-ART – An Interactive and Intelligent Web-Based Electronic Textbook. *Int J Artif Intell Educ* 26, 72–81 (2016).
<https://doi.org/10.1007/s40593-015-0066-8>

15. Розумна система для навчання. URL: <https://academyocean.com/ua>
Academyocean.

16. Грищенко Т.Б., Дейнеко Ж.В., Нікітенко О.М. Використання системи LaTeX під час підготовки наукових публікацій // Поліграфічні, мультимедійні та web-технології: тези доп. IV Міжнар. наук.-техн. конф. (14-17 травня 2019, м. Харків). 2019. Т1. С. 96-99. 96

17. Грищенко Т.Б., Дейнеко Ж.В., Нікітенко О.М. Створення електронних підручників засобами видавничої системи LaTeX

<https://openarchive.nure.ua/server/api/core/bitstreams/18839085-1780-4d9d-944e-e6235ecf3e80/content>

18. IT в освіті. Як створити лекцію з ресурсом «Книга» URL: http://mokriev.blogspot.com/2014/04/blog-post_14.html Moodle

19. Все про електронну книгу та підручник URL: <https://www.sites.google.com/site/vseproeknigutaepidrucnik/servisi>

20. Гриценко С.С. Сучасні технології створення електронних підручників. URL: <http://dspace.pnpu.edu.ua/bitstream/123456789/5182/1/Gricenko.pdf>.

21. Прохорчук О. SMART підручник важливий елемент SMART комплексу. URL: <http://surl.li/hvirq>

22. Колос К. Р. Google Sites (Сайти) як засіб узагальнення професійного досвіду педагогічного працівника : практичний посібник. Житомир : Видавництво «О. О. Євенок», 2016. – 99 с.

23. Все про електронні книгу і підручник. Google Drive: Sign-in. URL: <https://www.sites.google.com/site/vseproeknigutaepidrucnik/servisi> (дата звернення: 15.11.2024).

24. Найпотрібніше програмне забезпечення для створення електронної книги. Adobe. URL: <https://www.adobe.com/ua/products/indesign/ebook-creator-software.html> (дата звернення: 15.11.2024).

25. Паламарчук Є. А., Коваленко О., Бісікало О. В. Інформаційна екосистема «електронний університет». методологія, дослідження, впровадження, результати: монографія. Вінниця : ВНТУ, 2024. 188 с. URL: <https://press.vntu.edu.ua/index.php/vntu/catalog/book/>.

26. Aggarwal C. Reinforcement learning. in: . 2021. URL: https://doi.org/10.1007/978-3-030-72357-6_10. Artificial Intelligence, Springer, Cham. 2021. P. 6–7. URL: https://doi.org/10.1007/978-3-030-72357-6_10.

27. Education 4.0. TVETipedia Glossary UNESCO-UNEVOC. URL: <https://unevoc.unesco.org/home/+TVETipedia+Glossary/lang=en/show=term/term=Education+4.0> (date of access: 15.11.2024).
28. Weber G., Brusilovsky P. ELM-ART – an interactive and intelligent web-based electronic textbook. International Journal of Artificial Intelligence in Education. 2015. Vol. 26, no. 1. P. 72–81. URL: <https://doi.org/10.1007/s40593-015-0066-8> (date of access: 15.11.2024).
29. Magic bookТема 8 Сервіси для створення інтерактивних та електронних книг URL: <http://diagnost.pto.org.ua/index.php/item/78-tem8-cilk>
30. Aggarwal C.C. Reinforcement Learning. In: Artificial Intelligence. 2021. Springer, Cham. https://doi.org/10.1007/978-3-030-72357-6_10
31. Assessment in Higher Education: Professional Development for Teachers. Erasmus University of Rotterdam OnLine Course. URL: <https://www.coursera.org/learn/assessment-higher-education>
32. Tree-net Ієрархічно-мережева модель даних. <https://www.setlab.net/?view=concept:2>
33. Пономаренко В.С., Пушкар П.І., Андрющенко Т.Ю. та інш. Педагогічний дизайн засобів електронного навчання на робочому місці : монографія : [Електронне видання]. за заг. ред. д-ра екон. наук, професора В. С. Пономаренка, д-ра екон. наук, професора О. І. Пушкаря. Харків : ХНЕУ ім. С. Кузнеця, 2017. – 263 с.
34. DOU. Рейтинг мов програмування. <https://dou.ua/lenta/articles/language-rating-2024/>
35. Хирса В. Що таке патерни MVC та MVP <https://highload.today/uk/blogs/shho-take-mvc-ta-mvp-patterni/>
36. E-Commerce Frontend Vs Backend: What's the Difference? URL : <https://fabric>

37. Node Js JavaScript . URL : <https://uk.nodejs.org/>
38. Why we use node js and SQL in the development: URL: <https://www.techmagic.co/blog/why-we-use-node-js-in-the-development/>
39. React – JavaScript-бібліотека для створення користувацьких інтерфейсів. <https://uk.reactjs.org/>.
40. Міні-огляд: що потрібно знати про розробку на React.js у 2022. URL: <https://tresor.tech/ua/mini-oglyad-shho-potribno-znati-pro-rozrobku-na-react-js-u-2022/>.
41. Vasyliiev Oleksii. PHP by Example: A Practical Guide to Creating Web Applications with PHP First Edition. 2024.
42. Трофименко О. Г., Прокоп Ю. В., Логінова Н. І., Копитчук Н. І. Організація баз даних. Київ. 2-ге вид. виправ. і доповн. – Одеса : Фенікс, 2019. 246 с.
43. Яровий А. А., Іванчук Я. В., Озеранський В. С., Василевський В. О. Особливості моделювання мультикомпонентної інформаційної технології автоматизованого тестування. *Інформаційні технології та комп'ютерна інженерія*. 2021. № 3. С. 53-59.
44. Трофименко О. Г., Дика А.І. Тестування та забезпечення якості програмних систем. НУ. Одеса. 2024. 128 с.
45. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 122 «Комп'ютерні науки» [Електронний ресурс] / уклад.: А. А. Яровий, О. К. Колесницький. Вінниця : ВНТУ, 2023. (PDF, 58 с.)

ДОДАТКИ

Додаток А (обов'язковий) Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень

ПЕРЕВІРКА НА ПЛАГІАТ

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ
ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: Інформаційна технологія створення електронних підручників в системі JetIQ.

Тип роботи: магістерська кваліфікаційна робота

Підрозділ: кафедра комп'ютерних наук, ФІТА

ПОКАЗНИКИ ЗВІТУ ПОДІБНОСТІ TURNITIN

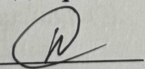
Оригінальність	97,00%	Схожість	3,00%
----------------	--------	----------	-------

Аналіз звіту подібності (відмітити потрібне):

☒ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.

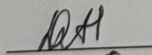
☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.

☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку  Озеранський В.С.

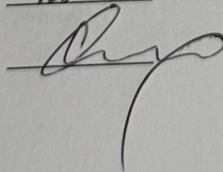
Ознайомлені з повним звітом подібності, який був згенерований системою Turnitin щодо роботи.

Автор роботи



Власенко Д.В.

Керівник роботи



Сілагін О.В

Додаток Б (обов'язковий) Лістинг програми

Index.php

```

<?php
ini_set('session.cookie_lifetime', 3600);
ini_set('session.cookie_secure', false); // Только HTTPS
ini_set('session.cookie_httponly', true); // Защита от XSS

header('Access-Control-Allow-Origin: http://localhost:3000');
header('Content-type: application/json');
header('Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS');
header('Access-Control-Allow-Credentials: true');
header('Access-Control-Allow-Headers: Content-Type, Authorization, X-Requested-With');
if ($_SERVER['REQUEST_METHOD'] === 'OPTIONS') {
    http_response_code(200);
    exit();
}
session_start();
function getData($method)
{
    $data = new stdClass();
    if ($method === "GET")
    {
        $data->body = json_decode(file_get_contents('php://input'));
    }
    $data->parameters = [];
    $dataGet = $_GET;
    foreach ($dataGet as $key => $value) {

```

```

    if ($key = "q")
    {
        $data->parameters[$key] = $value;
    }
    return $data;
}
}

function getMethod(){
    return $_SERVER['REQUEST_METHOD'];
}

$link = mysqli_connect('localhost','root','','jetbook');
if (!$link) {
    echo "Ошибка: Невозможно установить соединение с MySQL." . PHP_EOL;
    echo "Код ошибки error:" . mysqli_connect_errno() . PHP_EOL;
    echo "Текст ошибки error: " . mysqli_connect_error() . PHP_EOL;
    exit;
}

// echo "Соединение с MySQL установлено!" . PHP_EOL;
// echo "Информация о сервере: " . mysqli_get_host_info($link) . PHP_EOL;
$message = [];
$message["users"] = [];
$res = $link->query("SELECT id,email,fullName FROM user");

if (!$res){
    echo "Не удалось выполнить запрос: (" . $mysqli->errno . ") " . $mysqli->error;
}

```

```

    }
else
{
while ($row = $res->fetch_assoc())
{
    $message["users"][] = [
        "id" => $row['id'],
        "email" => $row['email'],
        "fullName" => $row['fullName'],
    ];
}
}

// echo json_encode($_POST);

// spl_autoload_register(function (string $className) {
//     require_once __DIR__ . '/../' . $className . '.php';
// });

// var_dump($_GET['route']);

// $router = new League\Route\Router;

// $router->get('/another-route', function (ServerRequestInterface $request): ResponseInterface {
//     echo 'Another-Route'
// });

$url = isset($_GET['q']) ? $_GET['q'] : "";
$url = rtrim($url, '/');
$urlList = explode('/', $url);

```



```

        else
        {
            echo json_encode($answer);
        }
        break;
    default:
        break;
    }
    break;
case 'POST':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'create':
            $text= $requestData->body->text;
            $taskId = $requestData->body->taskId;
            $correct = $requestData->body->correct;
            $answerInsertResult = $link->query("INSERT INTO answer (text, taskId,correct)
VALUES('$text', '$taskId', '$correct')");
            if(!$answerInsertResult){
                echo "too bad";
            }
            else{
                echo "answer created";
            }
            echo json_encode($requestData->body);
            break;
        default:

```

```

        break;
    }
    break;
default:
    break;
}
}

```

Catalog.php

```

<?php
function route($method,$urlList,$requestData){
    switch ($method) {
        case 'GET':
            $link = mysqli_connect('localhost','root','','jetbook');
            switch($urlList[1]){
                case 'getAll':
                    $userId = $urlList[2];
                    $catalog = $link->query("SELECT * from catalog WHERE userId='$userId')-
>fetch_all(MYSQLI_ASSOC);
                    if (is_null($catalog))
                    {
                        echo "NOPE";
                    }
                    else
                    {
                        echo json_encode($catalog);
                    }
                    break;

```

```

        default:
            break;
    }
    break;
case 'POST':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'create':
            $userId= $requestData->body->userId;
            $chapterId = $requestData->body->chapterId;
            $themeId = $requestData->body->themeId;
            $chapterName = $requestData->body->chapterName;
            $themesName = $requestData->body->themesName;

            $catalogInsertResult = $link->query("INSERT INTO catalog
(userId,chapterId,themesId,chapterName,themesName) VALUES('$userId', '$chapterId', '$themesId',
'$chapterName', '$themesName')");

            if(!$catalogInsertResult){
                echo "too bad";
            }
            else{
                echo "catalog created";
            }

            echo json_encode($requestData->body);
            break;
        case 'delete':
            $catalogId = $urlList[2];
            $catalog = $link->query("DELETE from catalog where id='$catalogId'");
            break;
    }
}

```

```

        default:
            break;
    }
    break;
default:
    break;
}
}

Chapter.php

<?php
function route($method,$urlList,$requestData){
    switch ($method) {
        case 'GET':
            $link = mysqli_connect('localhost','root','','jetbook');
            switch($urlList[1]){
                case 'getAll':
                    $themesId = $urlList[2];
                    $chapter = $link->query("SELECT * from chapter WHERE themesId='$themesId'")-
>fetch_all(MYSQLI_ASSOC);
                    if (is_null($chapter))
                    {
                        echo "NOPE";
                    }
                    else
                    {
                        echo json_encode($chapter);
                    }

```

```

        break;
    default:
        break;
    }
    break;
case 'POST':
    $link = mysqli_connect('localhost','root','','jetbook');
    switch($urlList[1]){
        case 'create':
            $themesId = $requestData->body->themesId;
            $name = $requestData->body->name;
            $chapterInsertResult = $link->query("INSERT INTO chapter (name,themesId)
VALUES('$name', '$themesId' )");
            if(!$chapterInsertResult){
                echo "too bad";
            }
            else{
                echo "chapter created";
            }
            echo json_encode($requestData->body);
            break;
        case 'update':
            $name= $requestData->body->name;
            $chapterId= $requestData->body->chapterId;
            $chapterInsertResult = $link->query("UPDATE chapter SET name='$name' WHERE
id='$chapterId'");
            if(!$chapterInsertResult){
                echo "too bad";
            }
            else{
                echo "chapter updated";
            }
            echo json_encode($requestData->body);
            break;
    }
}

```

```

        }
        else{
            echo "chapter updated";
        }
        echo json_encode($requestData->body);
    break;
    case 'delete':
        $chapterId = $urlList[2];
        $chapter = $link->query("DELETE from chapter where id='$chapterId'");
        break;
    default:
        break;
    }
    break;
    default:
        break;
    }
}
}

```

Glossary.php

```
<?php
```

```

function route($method,$urlList,$requestData){
    switch ($method) {
        case 'GET':
            $link = mysqli_connect('localhost','root','','jetbook');
            switch($urlList[1]){
                case 'getAll':
                    $userId = $urlList[2];

```

```

        $glossary = $link->query("SELECT * from glossary WHERE userId='$userId')-
>fetch_all(MYSQLI_ASSOC);

        if (is_null($glossary))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($glossary);
        }
        break;
    default:
        break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','','jetbook');
    switch($urlList[1]){
        case 'create':
            $title = $requestData->body->title;
            $text = $requestData->body->text;
            $userId = $requestData->body->userId;

            $glossaryInsertResult = $link->query("INSERT INTO glossary (title,text,userId)
VALUES('$title', '$text', '$userId' )");

            if(!$glossaryInsertResult){
                echo "too bad";
            }
            else{

```

```

        echo "glossary created";
    }
    echo json_encode($requestData->body);
    break;

    default:
        break;
}
break;
default:
    break;

```

Plan.php

```

<?php
function route($method,$urlList,$requestData){
    switch ($method) {
        case 'GET':
            $link = mysqli_connect('localhost','root','','jetbook');
            switch($urlList[1]){
                case 'getAll':
                    $plan = $link->query("SELECT * from plan")->fetch_all(MYSQLI_ASSOC);
                    if (is_null($plan))
                    {
                        echo "NOPE";
                    }
                    else
                    {
                        echo json_encode($plan);
                    }
                }
            }
        }
    }
}

```

```

    }
    break;
default:
    break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','','jetbook');
    switch($urlList[1]){
        case 'create':
            $creatorName = $requestData->body->creatorName;
            $chapterId = $requestData->body->chapterId;
            $chapterName = $requestData->body->chapterName;
            $themesId = $requestData->body->themesId;
            $themesName = $requestData->body->themesName;
            $groupName = $requestData->body->groupName;
            $speciality = $requestData->body->speciality;
            $department = $requestData->body->department;
            $views = $requestData->body->views;

            $planInsertResult = $link->query("INSERT INTO plan
(creatorName,chapterId,chapterName,themesName,themesId,groupName,speciality,department,views)

VALUES('$creatorName', '$chapterId', '$chapterName', '$themesName', '$themesId',
'$groupName', '$speciality', '$department', '$views' )");

            if(!$planInsertResult){
                echo "too bad";
            }
            else{

```

```

        echo "plan created";
    }
    echo json_encode($requestData->body);
    break;
    case 'delete':
        $planId = $urlList[2];
        $plan = $link->query("DELETE from plan where id='$planId'");
        break;
    default:
        break;
    }
    break;
default:
    break;
}
}

```

ResultStudent.php

```

<?php
function route($method,$urlList,$requestData){
    switch ($method) {
        case 'GET':
            $link = mysqli_connect('localhost','root','','jetbook');
            switch($urlList[1]){
                case 'getAllStudentResult':
                    $userId = $urlList[2];
                    $resultStudent = $link->query("SELECT * from resultStudent WHERE
userId='$userId'")->fetch_all(MYSQLI_ASSOC);

```

```

    if (is_null($resultStudent))
    {
        echo "NOPE";
    }
    else
    {
        echo json_encode($resultStudent);
    }
    break;

case 'getAllTeacherResult':
    $userId = $urlList[2];

    $resultStudent = $link->query("SELECT * from resultStudent WHERE
userId='$userId'")->fetch_all(MYSQLI_ASSOC);

    if (is_null($resultStudent))
    {
        echo "NOPE";
    }
    else
    {
        echo json_encode($resultStudent);
    }
    break;

case 'getAllTestResult':
    $testId = $urlList[2];

    $resultStudent = $link->query("SELECT * from resultStudent WHERE
testId='$testId'")->fetch_all(MYSQLI_ASSOC);

    if (is_null($resultStudent))
    {

```

```

        echo "NOPE";
    }
    else
    {
        echo json_encode($resultStudent);
    }
    break;
default:
    break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'create':
            $userId= $requestData->body->userId;
            $testId = $requestData->body->testId;
            $grade = $requestData->body->grade;
            $fullName = $requestData->body->fullName;
            $creatorId = $requestData->body->creatorId;
            $chapterName = $requestData->body->chapterName;
            $resultStudentInsertResult = $link->query("INSERT INTO subject
(userId,testId,grade,fullName,creatorId,chapterName)
VALUES('$userId', '$testId', '$grade', '$fullName', '$creatorId', '$chapterName')");
            if(!$resultStudentInsertResult){
                echo "too bad";
            }

```

```

        else{
            echo "resultStudent created";
        }
        echo json_encode($requestData->body);
        break;
    default:
        break;
    }
    break;
default:
    break;
}
}

```

Subject.php

```
<?php
```

```

function route($method,$urlList,$requestData){
    switch ($method) {
        case 'GET':
            $link = mysqli_connect('localhost','root','','jetbook');
            switch($urlList[1]){
                case 'getAll':
                    $subject = $link->query("SELECT * from subject")->fetch_all(MYSQLI_ASSOC);
                    if (is_null($subject))
                    {
                        echo "NOPE";
                    }
                    else

```

```

        {
            echo json_encode($subject);
        }
        break;
    case 'getAllMySubject':
        $speciality = $urlList[2];
        $subject = $link->query("SELECT * from subject WHERE speciality='$speciality'")-
>fetch_all(MYSQLI_ASSOC);
        if (is_null($subject))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($subject);
        }
        break;
    default:
        break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'create':
            $name= $requestData->body->name;
            $speciality = $requestData->body->speciality;

```

```

        $subjectInsertResult = $link->query("INSERT INTO subject (name, speciality)
VALUES('$name', '$speciality')");

        if(!$subjectInsertResult){

            echo "too bad";

        }

        else{

            echo "subject created";

        }

        echo json_encode($requestData->body);

        break;

    default:

        break;

    }

    break;

default:

    break;

}

}

```

Task.php

```

<?php

function route($method,$urlList,$requestData){

    switch ($method) {

        case 'GET':

            $link = mysqli_connect('localhost','root','','jetbook');

            switch($urlList[1]){

                case 'getAll':

                    $tasks = $link->query("SELECT * from task")->fetch_all(MYSQLI_ASSOC);

```

```

        if (is_null($tasks))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($tasks);
        }
        break;
    default:
        break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'create':
            $testId= $requestData->body->testId;
            $name = $requestData->body->name;
            $taskInsertResult = $link->query("INSERT INTO task (name,testId)
VALUES('$name', '$testId')");
            $task = $link->query("SELECT id from task WHERE testId='$testId' AND
name='$name'")->fetch_assoc();
            echo json_encode($task);
            break;
        case 'update':
            $chapterId= $requestData->body->name;
            $date= $requestData->body->name;

```

```

        $creatorId= $requestData->body->name;
        $themesId= $requestData->body->name;
        $themesName = $requestData->body->subjId;
        $chapterName = $requestData->body->creatorName;

        $testInsertResult = $link->query("INSERT INTO test (chapterId, date, creatorId,
themesId,themesName,chapterName) VALUES('$chapterId', '$date', '$creatorId', '$themesId',
'$themesName', '$chapterName')");

        if(!$testInsertResult){
            echo "too bad";
        }
        else{
            echo "test created";
        }

        echo json_encode($requestData->body);

        break;

    default:

        break;

    }

    break;

    default:

        break;

    }
}

```

Test.php

```

<?php

function route($method,$urlList,$requestData){

    switch ($method) {

        case 'GET':

```

```

$link = mysqli_connect('localhost','root','', 'jetbook');
switch($urlList[1]){
    case 'getOne':
        $chapterId = $urlList[2];
        $tests = $link->query("SELECT * from test WHERE chapterId='$chapterId'")-
>fetch_assoc();
        if (is_null($tests))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($tests);
        }
        break;
    case 'getAllThemesTest':
        $themesId = $urlList[2];
        $tests = $link->query("SELECT * from test WHERE themesId='$themesId'")-
>fetch_all(MYSQLI_ASSOC);
        if (is_null($tests))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($tests);
        }
        break;

```

```

    case 'getAllTeachersTest':
        $creatorId = $requestData->body->creatorId;
        $tests = $link->query("SELECT * from test WHERE creatorId='$creatorId'")-
>fetch_all(MYSQLI_ASSOC);
        if (is_null($tests))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($tests);
        }
        break;
    default:
        break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','','jetbook');
    switch($urlList[1]){
        case 'create':
            $chapterId= $requestData->body->chapterId;
            $date= $requestData->body->date;
            $creatorId= $requestData->body->creatorId;
            $themesId= $requestData->body->themesId;
            $themesName = $requestData->body->themesName;
            $chapterName = $requestData->body->chapterName;

```

```

        $testInsertResult = $link->query("INSERT INTO test (chapterId, date, creatorId,
themesId,themesName,chapterName) VALUES('$chapterId', '$date', '$creatorId', '$themesId',
'$themesName', '$chapterName')");

        if(!$testInsertResult){

            echo "too bad";

        }

        else{

            echo "test created";

        }

        echo json_encode($requestData->body);

        break;

    default:

        break;

    }

    break;

    default:

        break;

    }

}

```

Themes.php

```
<?php
```

```

function route($method,$urlList,$requestData){

    switch ($method) {

        case 'GET':

            $link = mysqli_connect('localhost','root','','jetbook');

            switch($urlList[1]){

                case 'getAll':

                    $themes = $link->query("SELECT * from themes")->fetch_all(MYSQLI_ASSOC);

```

```

    if (is_null($themes))
    {
        echo "NOPE";
    }
    else
    {
        echo json_encode($themes);
    }
    break;

case 'getAllTeacherThemes':
    $fullName = $urlList[2];

    $themes = $link->query("SELECT * from themes WHERE
creatorName='$fullName'")->fetch_all(MYSQLI_ASSOC);

    if (is_null($themes))
    {
        echo "NOPE";
    }
    else
    {
        echo json_encode($themes);
    }
    break;

case 'getAllSubjThemes':
    $subjId = $urlList[2];

    $themes = $link->query("SELECT * from themes WHERE subjId='$subjId'")-
>fetch_all(MYSQLI_ASSOC);

    if (is_null($themes))
    {

```

```

        echo "NOPE";
    }
    else
    {
        echo json_encode($themes);
    }
    break;
default:
    break;
}
break;
case 'POST':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'update':
            $name= $requestData->body->name;
            $themesId= $requestData->body->themesId;
            $themeInsertResult = $link->query("UPDATE themes SET name='$name' WHERE
id='$themesId'");
            if(!$themeInsertResult){
                echo "too bad";
            }
            else{
                echo "Theme updated";
            }
            echo json_encode($requestData->body);
            break;

```

```

case 'create':
    $name= $requestData->body->name;
    $subjId = $requestData->body->subjId;
    $creatorName = $requestData->body->creatorName;
    $creatorId = $requestData->body->creatorId;
    $themeInsertResult = $link->query("INSERT INTO themes (name, subjId,
creatorName, creatorId) VALUES('$name', '$subjId', '$creatorName', '$creatorId')");
    if(!$themeInsertResult){
        echo "too bad";
    }
    else{
        echo "Theme created";
    }
    echo json_encode($requestData->body);
break;
case 'delete':
    $themesId = $urlList[2];
    $theme = $link->query("DELETE from themes where id='$themesId'");
    break;
default:
    break;
}
break;
default:
    break;
}
}

```



```

        //json_encode($_SESSION);
        //echo session_status();
        //echo session_write_close();
        //$_SESSION['userId'] = 77;
        $userId = $_SESSION['userId'];
        //echo $_SESSION['userId'];
        //exit;

        $currUser = $link->query("SELECT * FROM user WHERE id='$userId')-
>fetch_assoc();

        echo json_encode($currUser);

        break;
    case 'register':
        $email = $requestData->body->email;
        $user = $link->query("SELECT id from user where email='$email')->fetch_assoc();
        if (is_null($user))
        {
            $password=hash("sha1", $requestData->body->password);
            $email = $requestData->body->email;
            $fullName = $requestData->body->fullName;

            $userInsertResult = $link->query("INSERT INTO user (email, fullName, password)
VALUES('$email', '$fullName', '$password')");

            if(!$userInsertResult){
                echo "too bad";
            }
            else{
                echo "User created";
            }
        }
        echo json_encode($requestData->body);

```

```

    }
    else
    {
        echo "EXIST";
    }
    break;
default:
    break;
}
break;
case 'GET':
    $link = mysqli_connect('localhost','root','', 'jetbook');
    switch($urlList[1]){
        case 'getAll':
            $users = $link->query("SELECT * from user")->fetch_all(MYSQLI_ASSOC);
            if (is_null($users))
            {
                echo "NOPE";
            }
            else
            {
                echo json_encode($users);
            }
            break;
        case 'getAllStudents':
            $users = $link->query("SELECT * from user WHERE role='Student'")-
            >fetch_all(MYSQLI_ASSOC);

```

```

        if (is_null($users))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($users);
        }
        break;
    case 'getAllTeachers':
        $users = $link->query("SELECT * from user WHERE role='Teacher'")-
>fetch_all(MYSQLI_ASSOC);
        if (is_null($users))
        {
            echo "NOPE";
        }
        else
        {
            echo json_encode($users);
        }
        break;
    default:
        break;
}
break;
default:
    brea

```

Додаток В (обов'язковий)**ІЛЮСТРАТИВНА ЧАСТИНА**

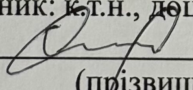
Інформаційна технологія створення електронних підручників в системі JetIQ

Виконав: студент 2-го курсу,
групи 2КН-23м
спеціальності 122 «Комп'ютерні науки»
(шифр і назва напрямку підготовки,
спеціальності)

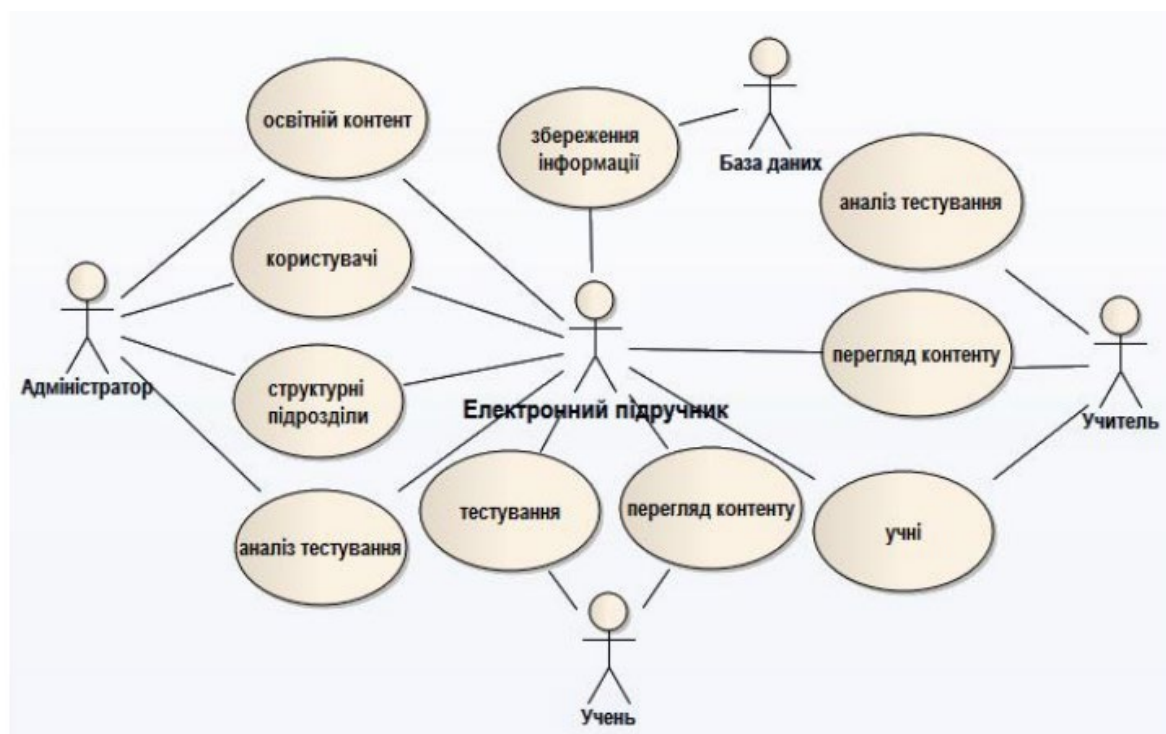


Власенко Д.В.

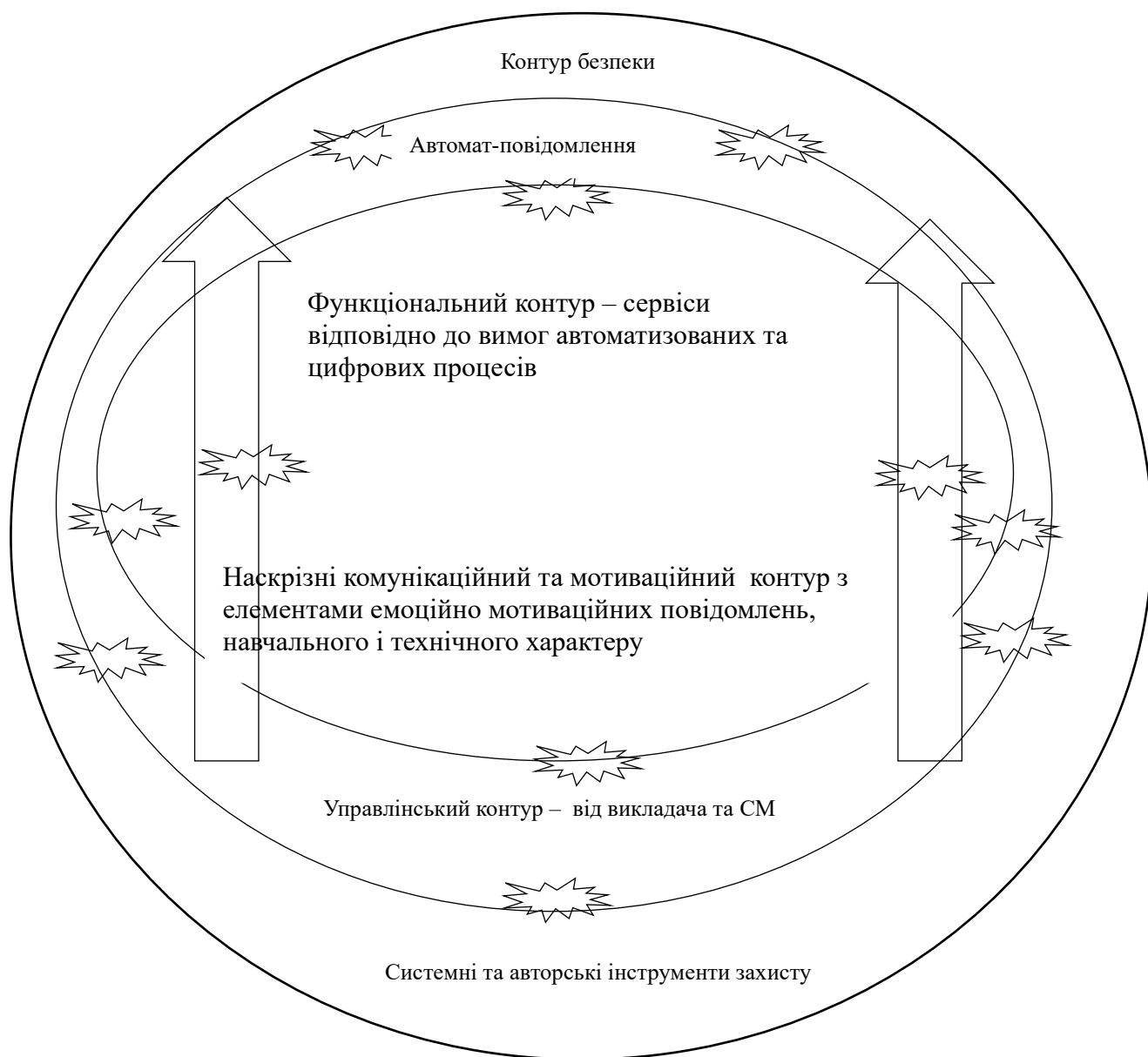
(прізвище та ініціали)

Керівник: к.т.н., доцент каф. КН
 Сілагін О.В.
(прізвище та ініціали)

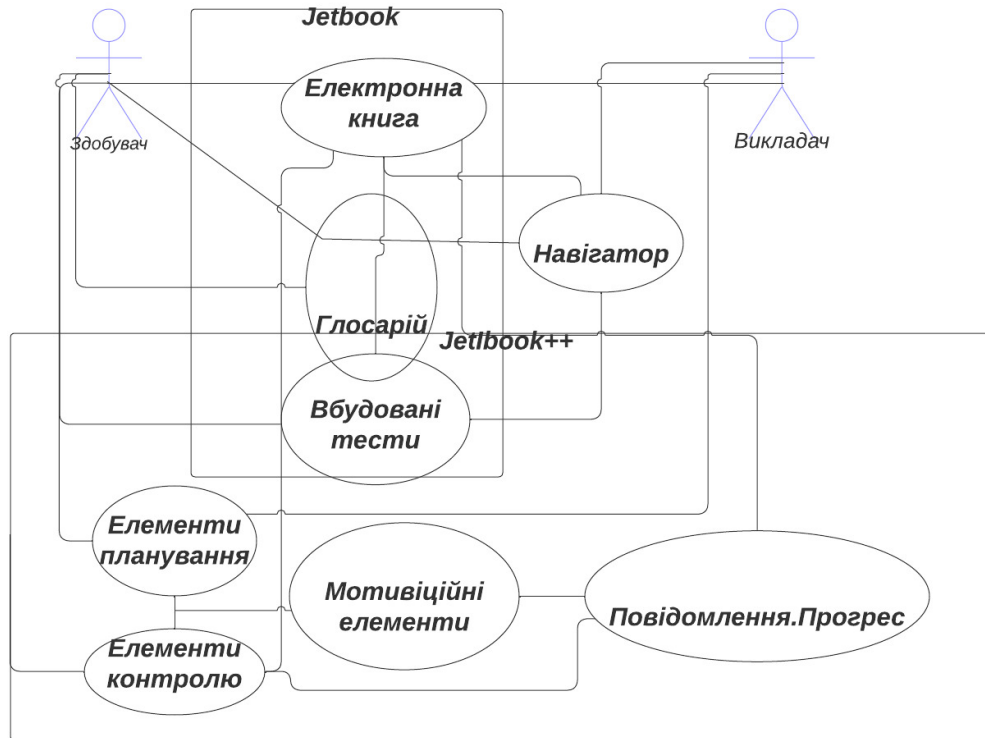
« 05 » 12 2024 р.



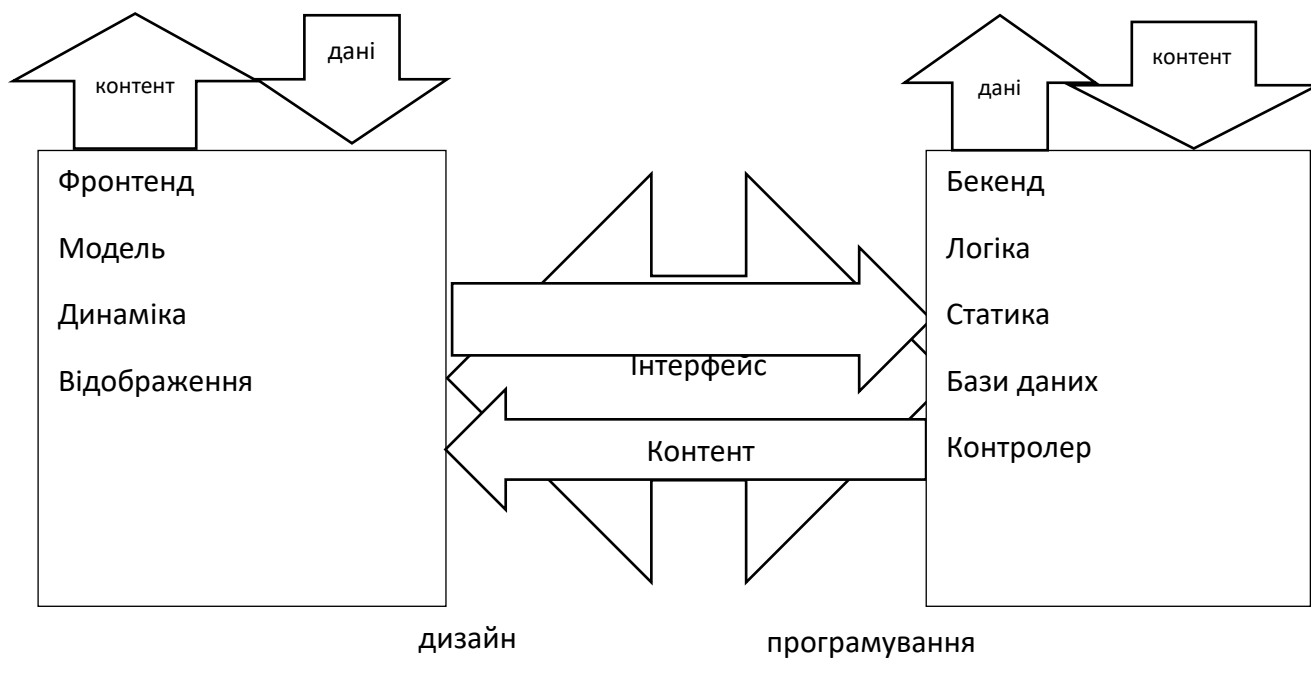
В.1 – Загальна модель електронного підручника



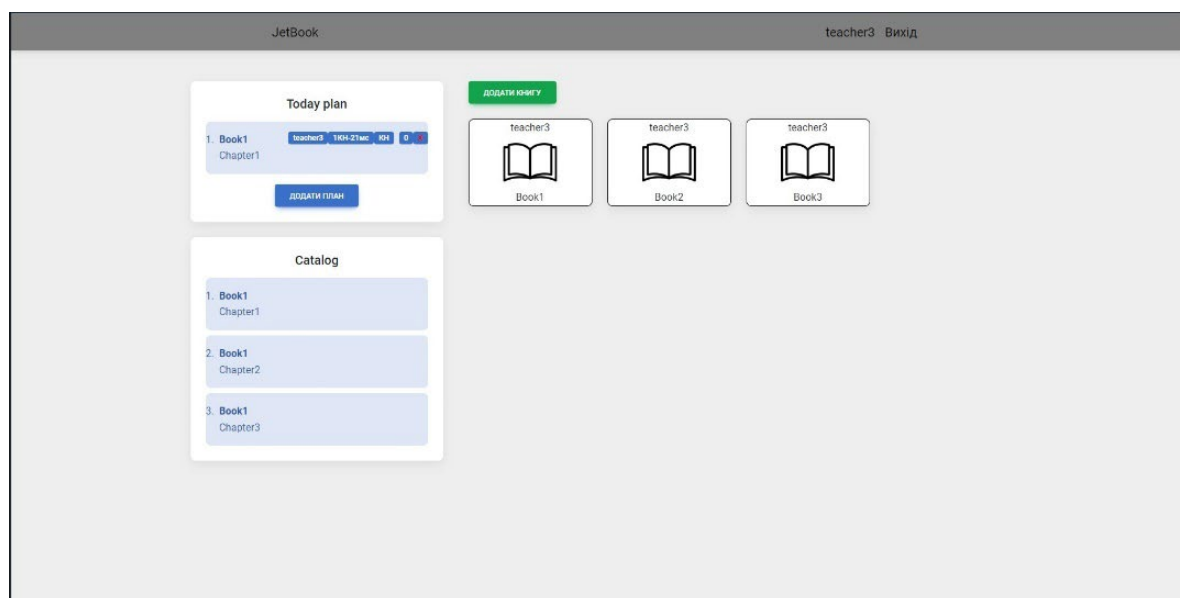
В.2 – Контурна модель середовища «Електронний підручник»



В.3 – Модель середовища «Електронний підручник» з функціональними, комунікаційними, мотиваційними та управлінськими елементами



В.4 – Модель фронтенд-бекенд електронного підручника



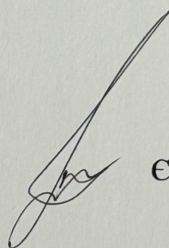
В.5 – Вікно створення електронного підручника

Додаток Г

**ДОВІДКА ПРО ВПРОВАДЖЕННЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕНЬ,
ВИКОНАНИХ В МАГІСТЕРСЬКІЙ КВАЛІФІКАЦІЙНІЙ РОБОТІ НА
ТЕМУ «Інформаційна технологія створення електронних підручників в
системі JetIQ» студента групи 2КН-23м Власенка Д.В.**

Видана студенту Власенку Д.В. про те, що запропонована інформація технологія, реалізована за допомогою фронтент-бекенд моделі електронного підручника з інтерактивними елементами прийнята до впровадження в систему управління навчанням, підтримки методичної, наукової та управлінської діяльності JetIQ VNTU. Запропоновані методи та моделі, програмна реалізація відповідають технічному завданню пілотного проєкту центру диджиталізації освітнього процесу ВНТУ щодо удосконалення веб-сервісу «Електронна книга» та його активного використання в форматі змішаного та дистанційного навчання, самостійної роботи студентів.

Директор центру диджиталізації
освітнього процесу ВНТУ, к.т.н., доцент



Є.А. Паламарчук