

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ  
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

**на тему: «Web-додаток з вивчення дисципліни “Розробка ігрових додатків”»**

за спеціальністю 122 «Комп’ютерні науки»,  
освітньо-професійна програма «Інформаційні технології проектування»

**Виконавець роботи:** студент групи ІТ-82-0 Лютий Дмитро Антонович

**Кваліфікаційна робота бакалавра  
захищена на засіданні ЕК  
з оцінкою**

\_\_\_\_\_ «\_\_» \_\_\_\_\_ 2022 р.

Науковий керівник

\_\_\_\_\_

(підпис)

к.т.н., доц., Неня В.Г.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає  
запозичень з праць інших авторів  
без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Сумський державний університет  
Факультет електроніки та інформаційних технологій  
Кафедра інформаційних технологій  
Спеціальність 122 «Комп'ютерні науки»  
Освітньо-професійна програма «Інформаційні технології проектування»

**ЗАТВЕРДЖУЮ**

Зав. кафедри ІТ

\_\_\_\_\_ В. В. Шендрик  
«\_\_» \_\_\_\_\_ 2022 р.

**З А В Д А Н Н Я**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА СТУДЕНТУ**

*Лютий Дмитро Антонович*

**1 Тема роботи** Web-додаток з вивчення дисципліни “Розробка ігрових додатків”

**керівник роботи** Неня Віктор Григорович, к.т.н., доцент,

затверджені наказом по університету від «27» квітня 2022 р. № 0301-VI

**2 Строк подання студентом роботи** «8» червня 2022 р.

**3 Вхідні дані до роботи** технічне завдання на розробку web-додатку з вивчення дисципліни “Розробка ігрових додатків”

**4 Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)**

Вступ, аналіз предметної області, проектування web-додатку, реалізація, висновки, список використаних джерел, додаток А технічне завдання, додаток Б планування робіт

**5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)** \_\_\_\_\_

## 6. Консультанти розділів роботи:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання \_\_\_\_\_

## КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз предметної області	02.12.2021-10.12.2021	
2	Складання технічного завдання	10.12.2021-14.12.2021	
3	Front-end розробка додатку	01.02.2022-11.02.2022	
4	Back-end розробка додатку	11.02.2022-16.02.2022	
5	Розробка бази даних	17.02.2022-21.02.2022	
6	Наповнення контентом	20.04.2022-02.05.2022	
7	Тестування	03.05.2022-19.05.2022	
8	Реліз	25.05.2022-26.05.2022	

Студент \_\_\_\_\_  
(підпис)

Лютий Д.А.

Керівник роботи \_\_\_\_\_  
(підпис)

к.т.н., доц. Неня В.Г.

## РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Web-додаток з вивчення дисципліни “Розробка ігрових додатків”».

Кваліфікаційну роботу бакалавра присвячено розробці web-додатку з вивчення дисципліни “Розробка ігрових додатків”. Перший розділ був присвячений дослідженню предметної області та її аналізу.

У другому розділі було зроблене структурно-функціональне моделювання та розроблені основні діаграми.

Третій розділ містить у собі детальний опис розробки системи з уточненим описанням розробки усіх модулів. Етапи створення бази даних, а також детально описаний процес тестування.

Результатом проведеної роботи є web-додаток с вивчення дисципліни “Розробка ігрових додатків”

Кваліфікаційна робота бакалавра містить 60 сторінок, 9 таблиці, 39 рисунків, список літератури із 19 найменувань, 2 додатки

Ключові слова: дисципліна, Web-додаток, “Розробка ігрових додатків”, база даних, тестування, тест план, тест кейси, чек-список.

## ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	8
1.1 ОГЛЯД ОСТАННІХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ.....	8
1.2 АНАЛІЗ ІСНУЮЧИХ ПРОДУКТІВ-АНАЛОГІВ.....	9
1.3 ПОСТАНОВКА ЗАДАЧІ.....	18
2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ .....	20
2.1 СТРУКТУРНО-ФУНКЦІОНАЛЬНА МОДЕЛЬ .....	20
2.2 UML-МОДЕЛІ.....	22
2.3 ПРОЕКТУВАННЯ БАЗИ ДАНИХ .....	28
3 ПРОГРАМНА РЕАЛІЗАЦІЯ.....	30
3.1 РОЗРОБКА АРХІТЕКТУРА ПРОГРАМНОГО ДОДАТКУ.....	30
3.2 ПРОГРАМНА РЕАЛІЗАЦІЯ .....	32
3.3 ТЕСТУВАННЯ .....	42
3.4 ХОСТИНГ САЙТУ .....	43
ВИСНОВКИ .....	44
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	45
ДОДАТОК А.....	48
ДОДАТОК Б .....	54

## ВСТУП

На сьогоднішній день розвиток інформаційних технологій є дуже важливим питанням. Саме тому велика кількість ресурсів направлена на їх розвиток. Дуже популярним напрямком є дистанційна форма навчання, яка почала свій розвиток ще у Європі наприкінці XVIII століття, в один час з появою пошти. Суть цього навчання полягала у тому, що учні отримували свої завдання поштою. Тим самим чином вони і відправляли завдання, спілкувались з педагогами та складали іспити.

Після появи інтернету почався значний прорив у розвитку дистанційних методів навчання. Учень міг отримати зворотній зв'язок, міг звернути увагу на свої помилки. Велика кількість матеріалів була розміщена безкоштовно і кожен бажаючий міг цією інформацією скористатися.

Дистанційне навчання – засіб реалізації процесу навчання, основною метою якого є використання інформаційних технологій, що дозволяють навчатися без безпосереднього, особистого контакту між викладачем і учнем. Це метод навчання передбачує індивідуальний процес передання і засвоєння знань, умінь, навичок, який відбувається за взаємодії віддалених один від одного учасників. Дистанційне вивчення певних дисциплін може збільшити попит та аудиторію цієї дисципліни. Кожен зацікавлений користувач може переглянути та зрозуміти чи цікава йому дисципліна, чи цікаво йому буде навчатись. Та виходячи з цього користувач може наприклад пов'язати своє життя чи трудову діяльність з вивчаємою дисципліною.

Сучасне дистанційне навчання включає в себе різні форми та методи передачі інформації, теоретичних та самостійних завдань, така форма навчання передбачає використання сучасних інформаційних технологій. Основною метою дистанційного навчання є розвиток, покращення та вдосконалення знань користувача за допомогою використання інформаційних технологій та інформаційних ресурсів. Виходячи з факторів за допомогою

яких досягаються цілі дистанційного навчання, можна зробити висновок, що реалізовувати такі методи навчання доцільно за допомогою технологій та ресурсів Інтернету.

Дистанційна форма навчання використовує звичайні форми навчання, але трохи змінені. Одною з основних форм навчання є лекція. Ця форма навчання реалізується за допомогою викладання матеріалу вчителем, в той час як учні роблять висновки, після чого можуть сформулювати проблеми та вирішити їх. Така форма навчання може проводитися як у реальному так і нереальному часі. Саме другий метод проведення лекцій передбачає викладання матеріалу вчителем на певний ресурс, після чого учень має можливість ознайомитися з цим матеріалом у зручний для нього час.

Отже, дистанційна форма навчання поширює інформаційно – комунікаційні технології, сприяє поширенню матеріалів та інформації, та залучає використання аудіо та відеоматеріалів. Уся інформація розповсюджується за допомогою певного ресурсу, який розташований у мережі Інтернет.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

## 1.1 Огляд останніх досліджень і публікацій

С реаліями сьогодення актуальність дистанційних форм навчання збільшується з кожним днем. Вона полягає в тому, що результати прогресу суспільства, накопичуються в інформаційній сфері розвитку. В даний момент етап розвитку інформаційної сфери можна охарактеризувати як телекомунікаційний. Дана область являє собою сферу спілкування, зв'язку, поширення інформації та розвиток знань. Можна проаналізувати, що професійні знання здобуті багато років тому с кожним роком старіють, тому необхідне їх безперервне вдосконалення. Дистанційна форма навчання являє собою систему безперервного самонавчання, завдяки масовому обміну інформації незалежно від часу та місця.

Важливою задачею будь-якої освіти є формування у спеціалістів професійного мислення, навичок засвоєння та критичного аналізу нової інформації, вміння планувати та структурувати подальшу роботу. Для вирішення цих завдань необхідною складовою є використання нових інформаційних технологій.

Актуальність саморозвиток та дистанційне вивчення певних дисциплін підвищується з розвитком хвороби коронавірус. Більшість навчальних закладів, шкіл та університетів в будь-який момент можуть бути закриті на карантин, або просто може бути обмежена їх діяльність в зв'язку з унеможливленням великого скупчення людей в одному місці. Тому створення веб-додатку з вивчення дисципліни "Розробка ігрових додатків" є необхідним, як для аудиторії яка вивчає дану дисципліну, так і для просто зацікавлених у самоосвіті осіб. Головною задачею є викладання та вивчення дисципліни без безпосереднього контакту між вчителем та учнем, яка передбачає індивідуальну траєкторію навчання та процес засвоєння навичок і



знань. Будь-який користувач має можливість безкоштовно скористатися ресурсами веб-додатку для вивчення та укріплення певних знань та навичок.

## 1.2 Аналіз існуючих продуктів-аналогів

Сьогодні можна знайти достатню кількість веб-додатків для вивчення певних дисциплін. С кожним днем кількість таких додатків зростає. Переглянути основний функціонал веб-додатку пов'язаного з вивченням дисциплін можна на основі вже існуючих прикладів. Наприклад вебсайт «Google's Python Class» [1]. Головна сторінка сайту зображена на рисунку 1.1.

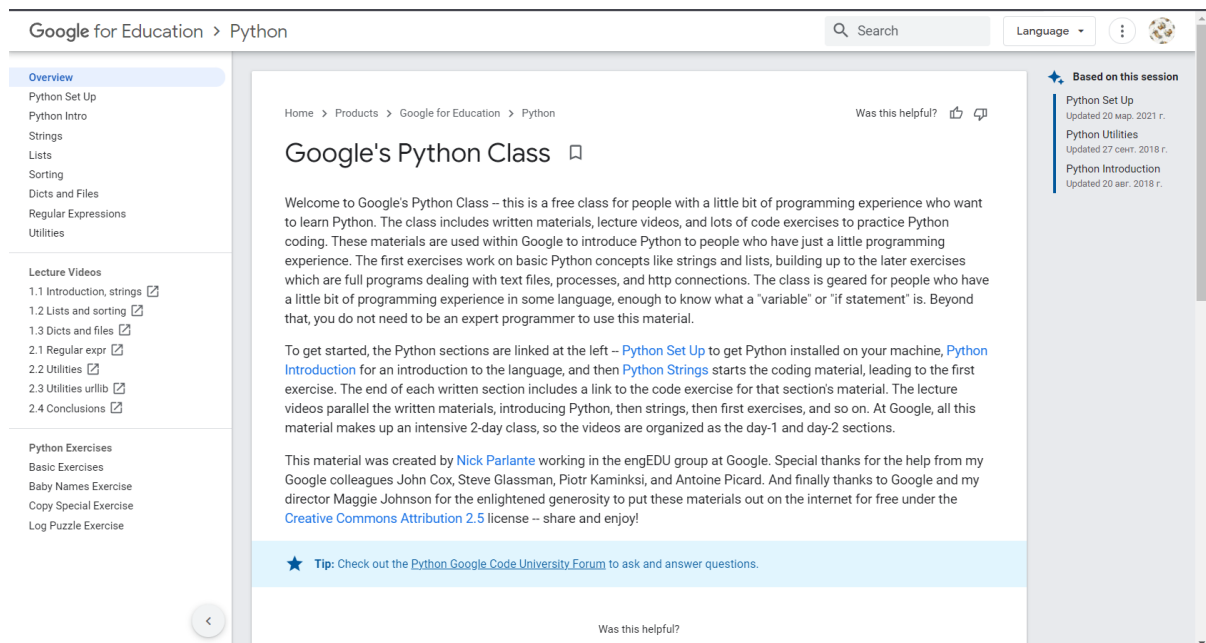


Рисунок 1.1 – Головна сторінка сайту «Google's Python Class» [1]

Зліва на сторінці розташоване навігаційне меню завдяки якому можна пересуватися на сайті та обирати цікавлячи теми для вивчення. Меню має стислий перелік основних тем для вивчення, також воно інформаційно насичене, кожен користувач має можливість знайти потрібну для нього інформацію. При переході на головну сторінку сайту нас зустрічає короткий опис веб-додатку.

Також на головній сторінці присутня можливість пошуку інформації на сайті. Поруч с пошуком є можливість зміни мови відображення сторінки. Дані функції зображені на рисунку 1.2.

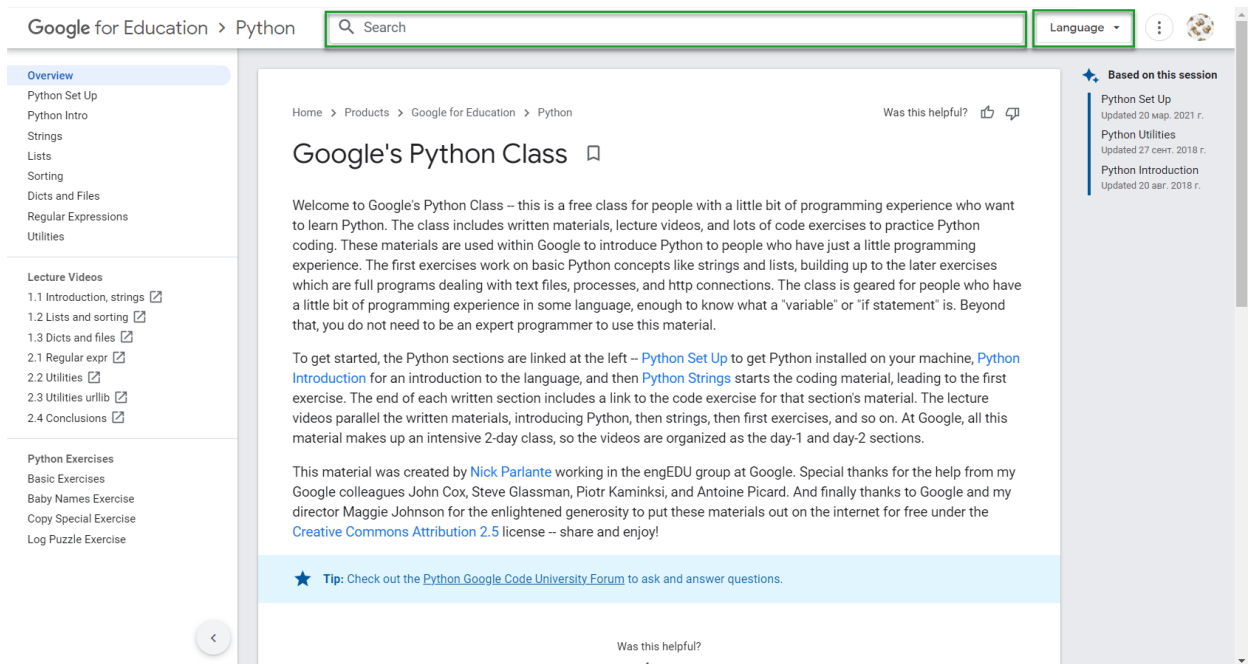


Рисунок 1.2 – Функції пошуку та зміни мови відображення.

Якщо перейти до нижньої частини сайту можна знайти блок для зворотного зв'язку, де розташовані основні медіа посилання та контактна інформація. Даний блок зображений на рисунку 1.3.

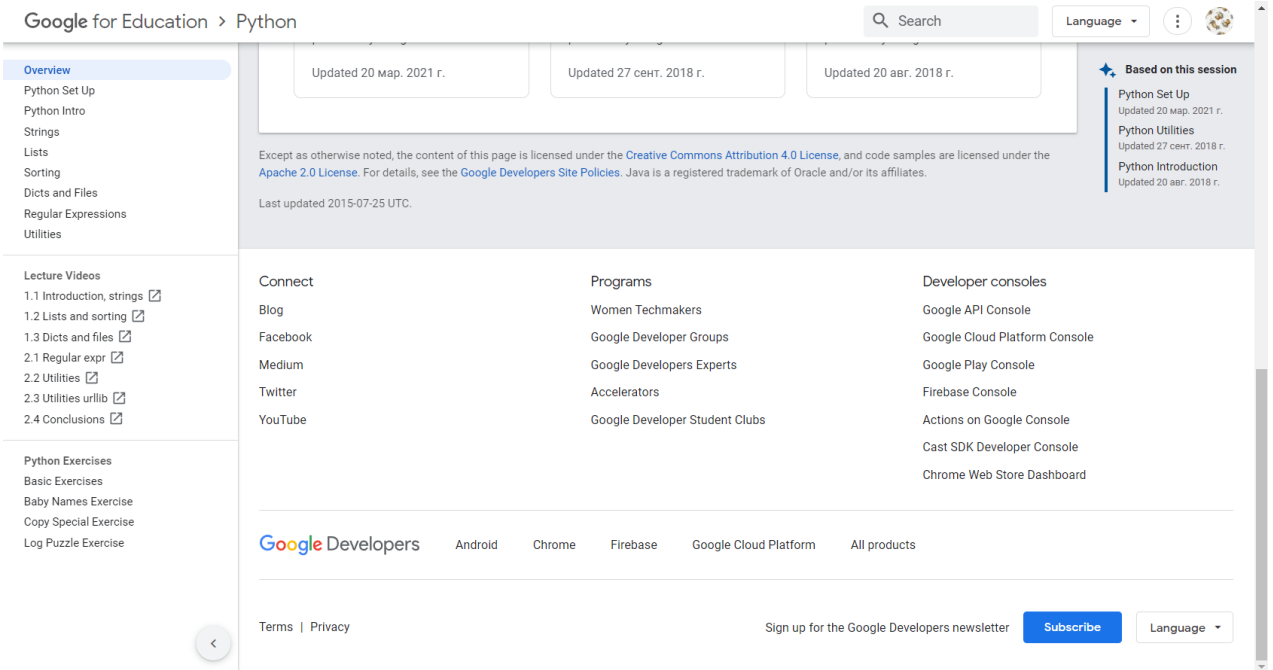


Рисунок 1.3 – Контактна інформація веб-сайту «Google’s Python Class»[1]

При перегляді навігаційного меню можна помітити блок с лекціями, який містить посилання на лекції. Даний блок зображений на рисунку 1.4

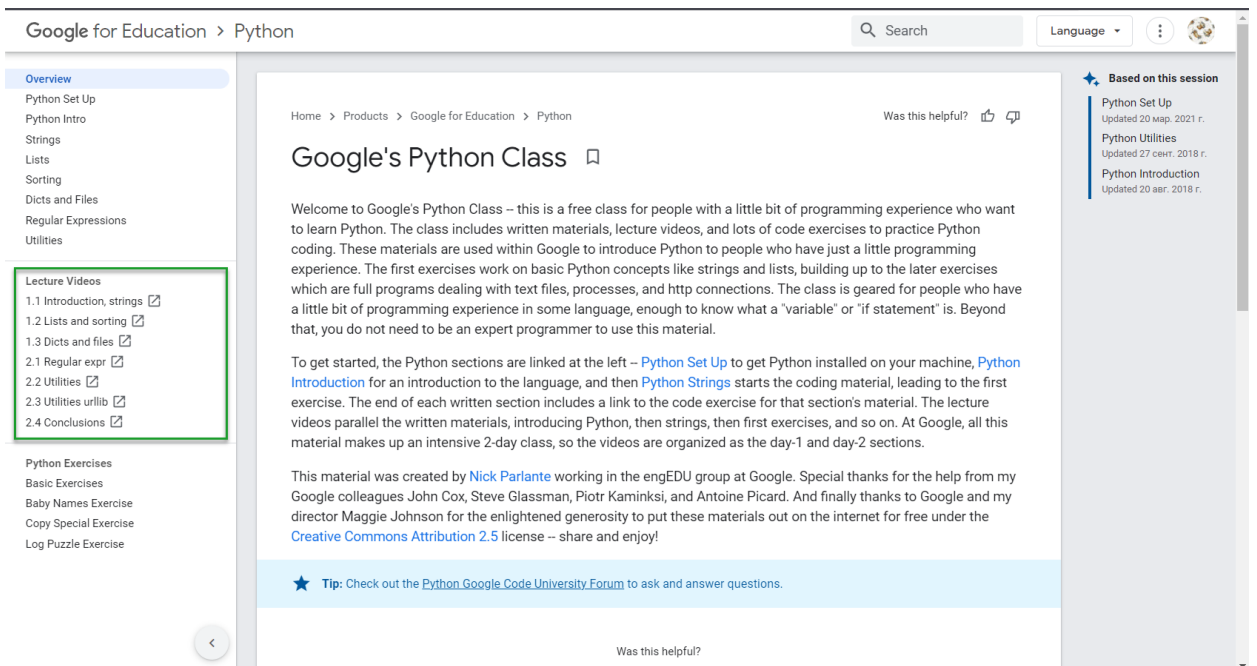


Рисунок 1.4 – Посилання на лекції

В даній веб системі інформація, яка відображається при перегляді певних тем має стисле та насичене наповнення. Вона містить чіткі пояснення та корисні приклади. Приклад зображений на рисунку 1.5.

Рисунок 1.5 – Перегляд тем для вивчення

Як альтернативу можна також розглянути сайт «METANIT.COM» [2]. Головна сторінка сайту зображена на рисунку 1.6.

Рисунок 1.6 – Головна сторінка сайту «METANIT.COM» [2]

Дана веб система також містить навігаційне меню, яке включає в себе основні розділи та теми для вивчення. Також, як і попередній сайт дана веб система на головній сторінці містить інформацію про себе та перелік головний новин сайту, які містяться у нижній частині сайту зображений на рисунку 1.7.

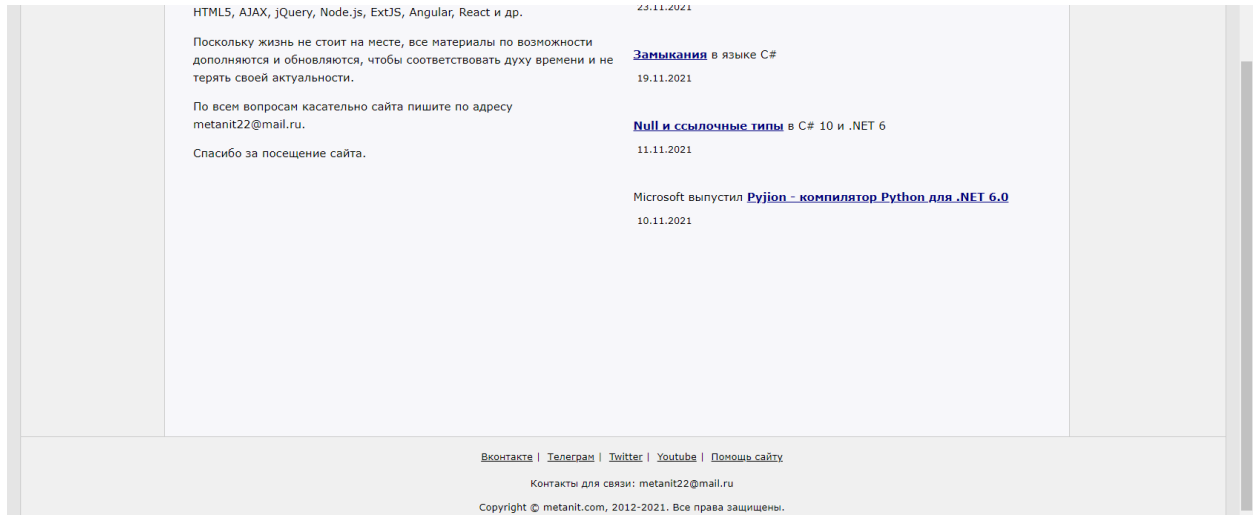


Рисунок 1.7 – Контактна інформація сайту «METANIN.COM» [2]

У нижній частині сайту також присутні головні медіа посилання та контактна інформація, яка дублюється у шапці сайту. На сайті також присутній пошук.

Кожний розділ має ієрархічно побудовану структуру завдяки якій користувач має можливість пересуватися по темам які його цікавлять. Приклад зображений на рисунку 1.8.

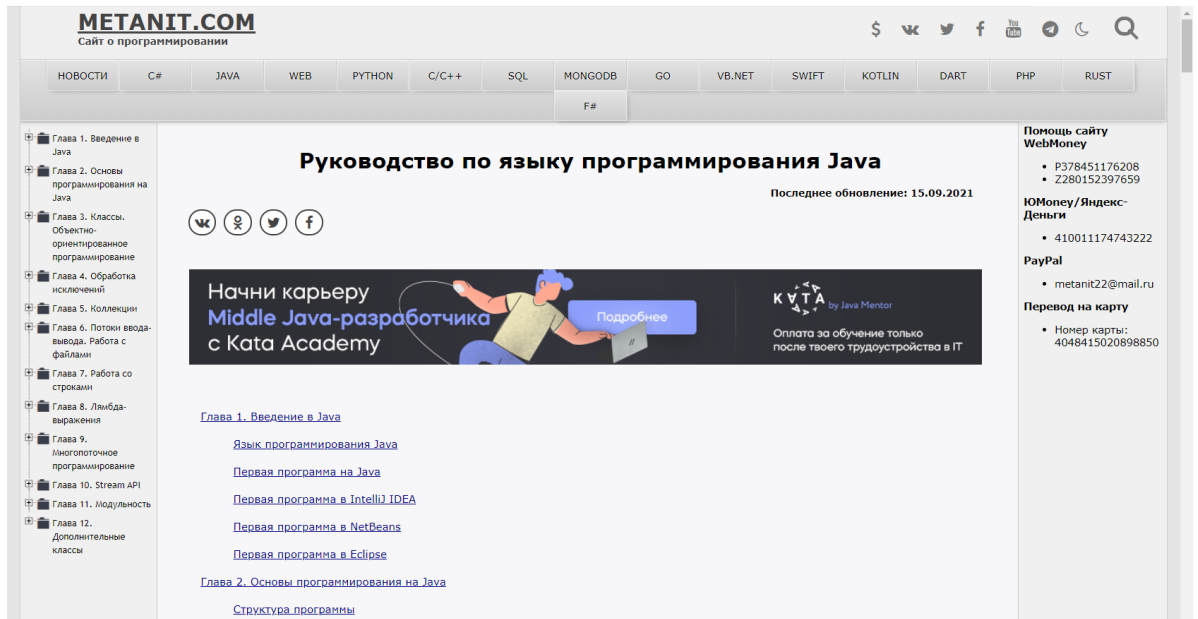


Рисунок 1.8 – Навігация по темам сайта «METANIT.COM» [2]

Даний сайт має стисло викладену інформацію, яка досить легка для розуміння. Також в даній веб системі присутня велика кількість зрозумілих прикладів. Приклад відображення інформації зображено на рисунку 1.9.

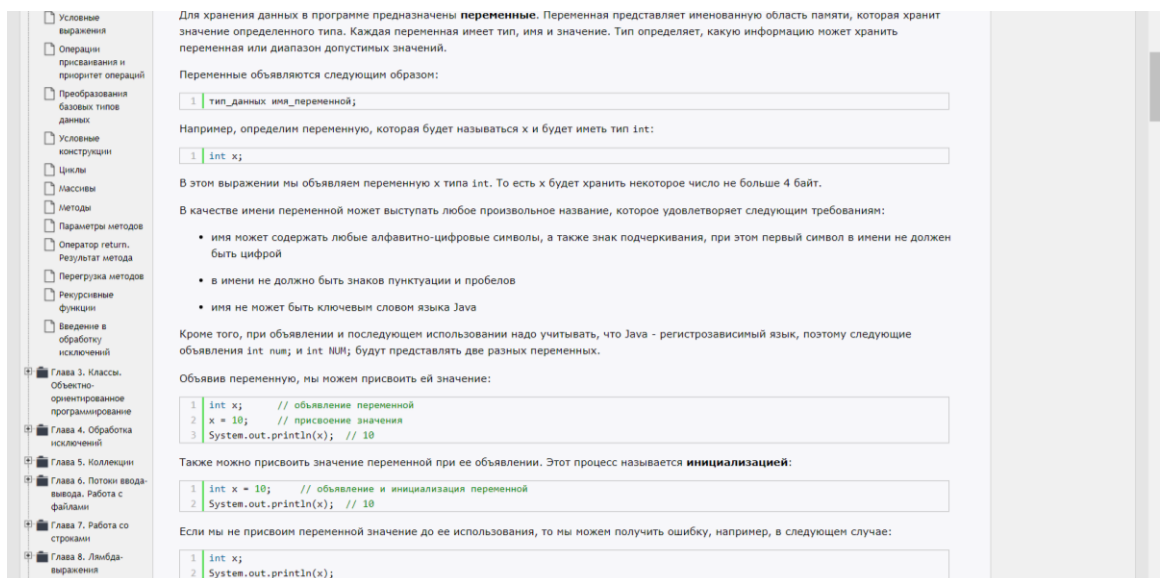


Рисунок 1.9 – Матеріал для вивчення

Також як аналог можна розглянути сайт Udemy [3]. Дизайн даного сайту можна описати, як сучасний та діловий. Інформація відображена чітко та лаконічно. На головній сторінці присутня інформація про портал, та відображені деякі популярні курси. Головна сторінка зображена на рисунку 1.10.

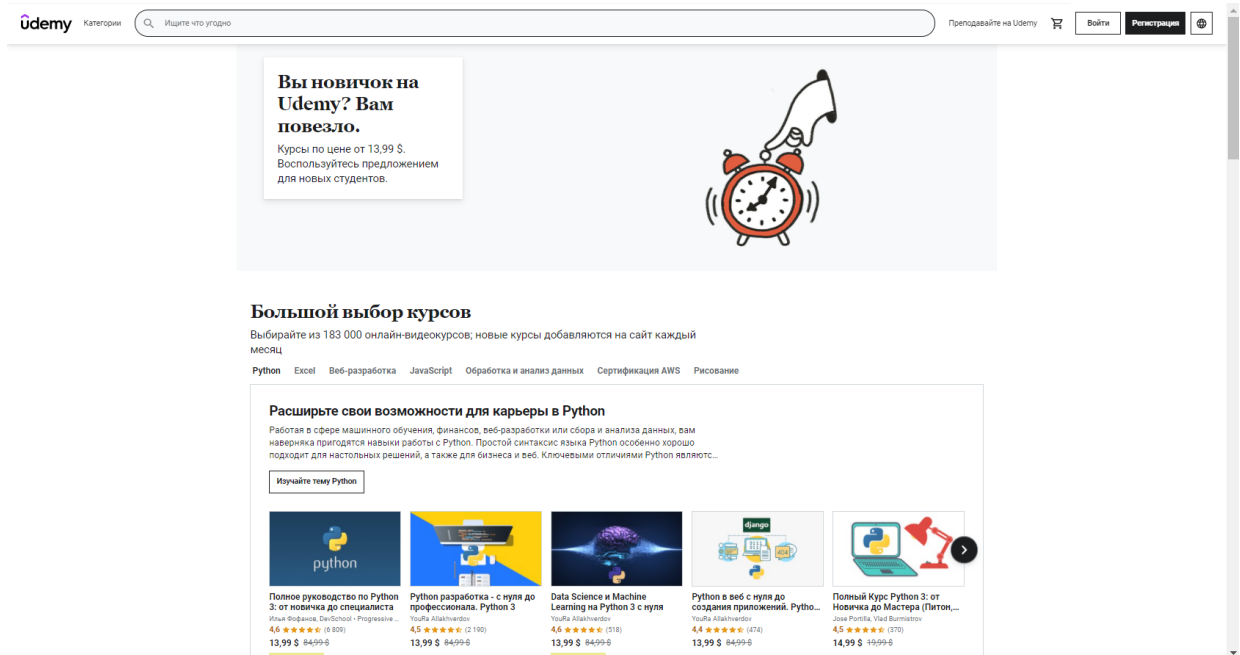


Рисунок 1.10 – Головна сторінка сайту Udemy [3]

Сайт розроблений у вигляді лендингу, це означає, що сайт складається з однієї сторінки, яка при гортанні вниз має анімацію та містить у собі різні блоки з інформацією. Рухаючись далі сайтом можна побачити блок з інформацією розбитою по категоріям, користувач може обрати цікавлячу для нього інформацію та перейти за відповідним посиланням. Даний блок зображено на рисунку 1.11.

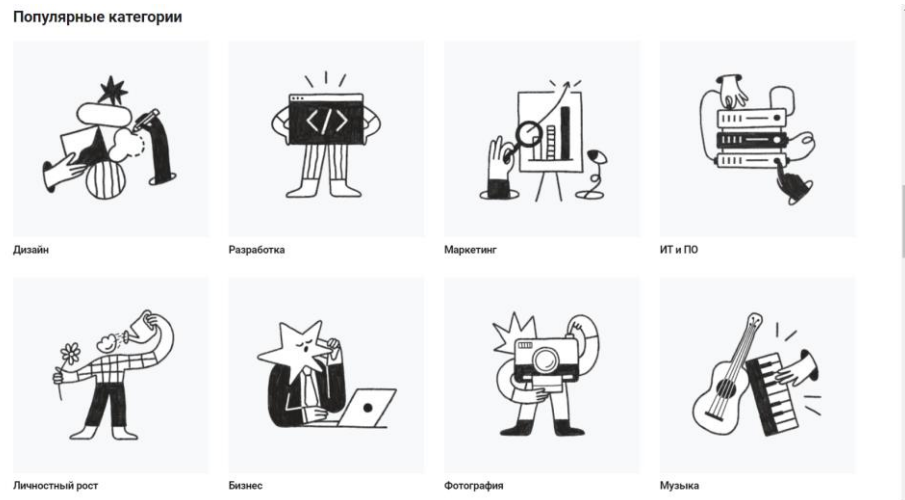


Рисунок 1.11 – Блок з категоріями

Як і у попередніх сайтів, основна контактна інформація знаходиться у нижній частині сайту. Також на сайті є можливість зміни мови. Футер сайту зображений на рисунку 1.12.

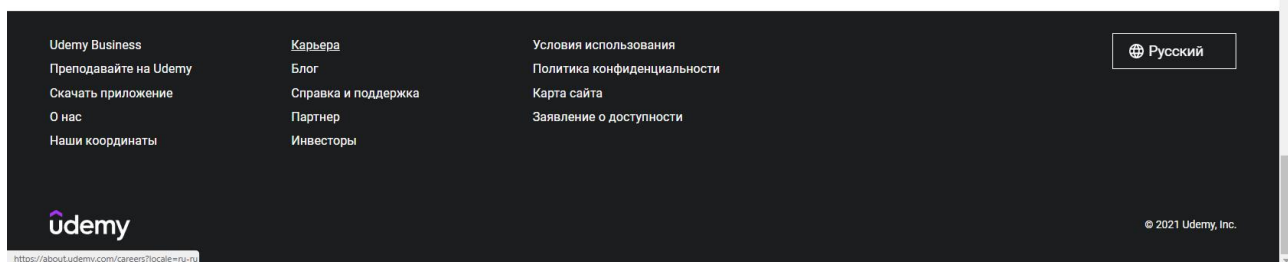
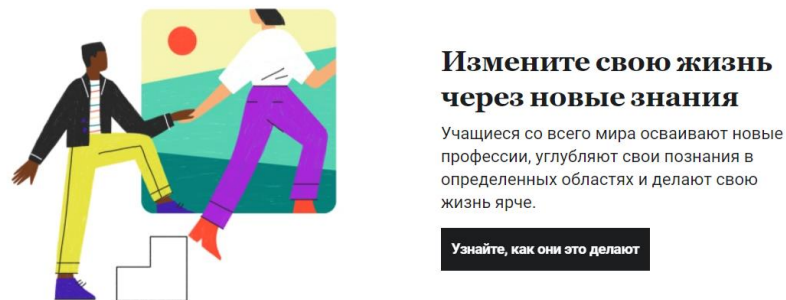


Рисунок 1.12 – Інформаційний блок сайту «UdeMy» [3]

Після проведеного аналізу сайтів аналогів можна побудувати таблицю, в якій відобразити основні переваги та недоліки розглянутих сайтів. Результати аналізу зображені в таблиці 1.1



Таблиця 1.1 – Порівняльний аналіз характеристик веб-сайтів

<b>Характеристика</b>	<b>«Google's Python Class»</b>	<b>«METANIT»</b>	<b>«Udemy»</b>
Можливість роботи на декількох мовах	+	+	+
Сучасний дизайн	+	-	+
Навігація сайтом	+	+	+
Легкість сприйняття матеріалів	+	+	+
Можливість допомоги та консультації	+	+	+

Отже, дані з таблиці порівняння дають змоги виділити основні переваги та недоліки порівнюваних сайтів. За допомогою даної таблиці можна використати корисний функціонал сайту та забезпечити сайт від недоліків.

### 1.3 Постановка задачі

Метою даної роботи є створення веб-додатку з вивчення дисципліни «Розробка ігрових додатків» для збільшення зацікавленої аудиторії та зацікавлення користувачів у даному напрямку. Інтерфейс додатку повинен бути виконаний у сучасному та діловому стилі. Він повинен бути інтуїтивно зрозумілий та простий у використанні.

Проаналізувавши основні переваги та недоліки, та провівши повний аналіз предметної області, були визначені основні вимоги до створення веб-додатку:

- реалізувати можливість допомоги та консультації ;
- реалізувати розподіл інформації по блокам , завдяки чому користувач матиме наочне представлення для подальшої навігації;
- розробити базу даних для зберігання інформації та подальшого оновлення інформації на сайті.

Наведені задачі допоможуть досягти мети проекту:

- дослідити предметну область та провести повний її аналіз;
- виконати аналіз аналогів веб-додатків;
- виконати порівняльний аналіз сайтів аналогів та виділити їх основні переваги та недоліки;
- обрати мову написання додатку;
- створити модель додатку та реалізувати структуру;
- розробити функціональні модулі додатку;
- розробити базу даних;
- провести якісне тестування.

Для детального опису всім функціональних вимог додатку було розроблено технічне завдання (додаток А).

При реалізація веб-додатку будуть використані наступні технології. Мова розмітки гіпертексту HTML, для стилізації буде використаний CSS. JS

буде використаний для реалізації будь-яких анімацій. База даних буде реалізована у сервісі PHPMyAdmin за використанням MySQL.

## 2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

### 2.1 Структурно-функціональна модель web-додатку з вивчення дисципліни «Розробка ігрових додатків»

Найголовніша діаграма, яка будується першою при створенні будь-якого проекту – це контекстна діаграма IDEF0. Опис системи за допомогою IDEF0 називається функціональною моделлю. Функціональна модель призначена для опису існуючих бізнес-процесів. Головним процесом web-додатку є взаємозв'язок між вчителем та учнем, а саме передача та отримання інформації.

Вхідними даними основного процесу виступають:

- інформація з вивчення дисципліни.

Вихідними даними виступають:

- архів,
- звітна документація.

Механізми проекту:

- регламент,
- методи,
- сервер,
- технічне завдання.

На рисунку 2.1 зображено фінальний вигляд контекстної діаграми web-додатку з вивчення дисципліни «Розробка ігрових додатків»:

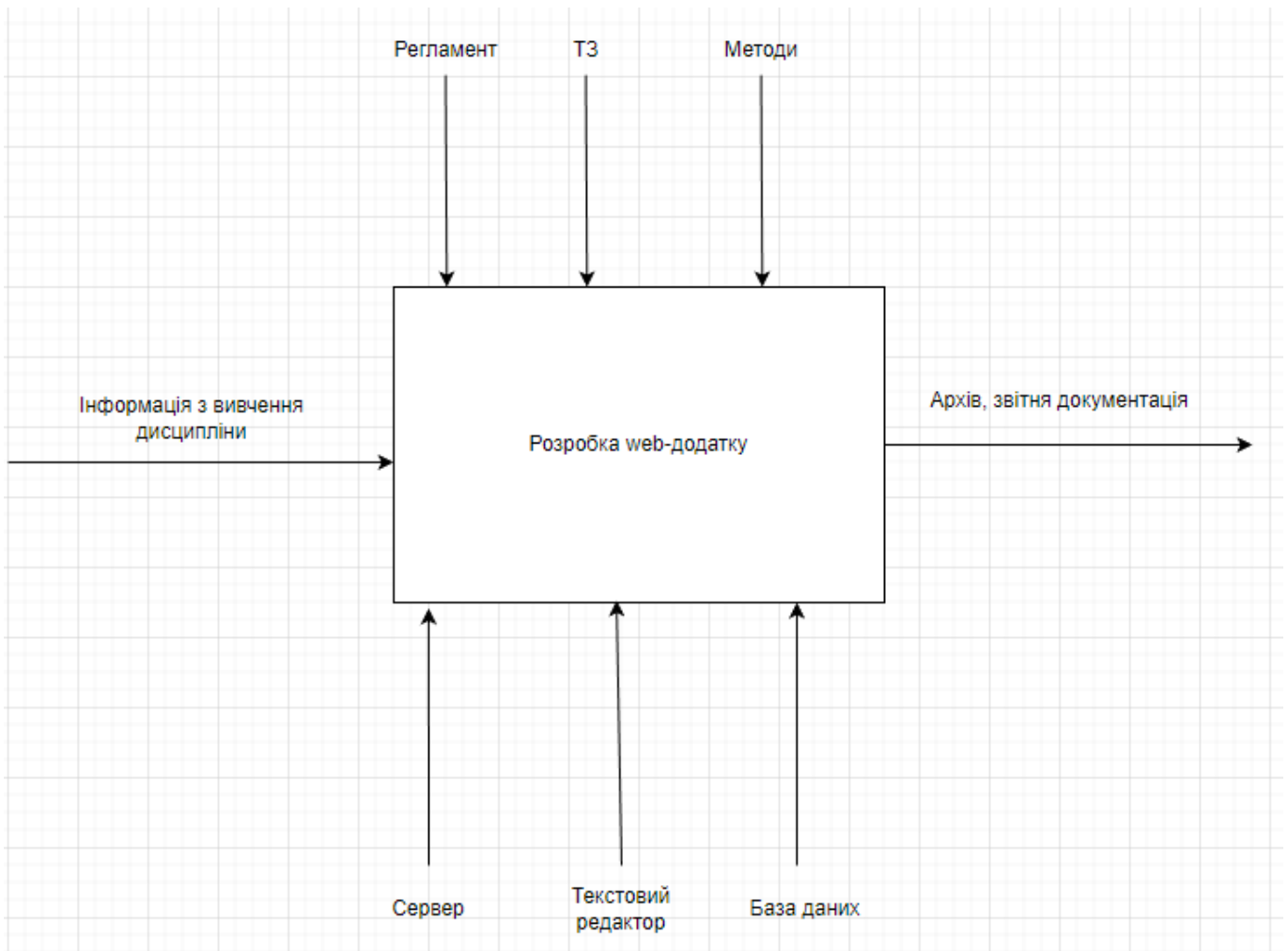


Рисунок 2.1 – Контекстна діаграма процесів Web-додатку

Стандарт IDEF1 це інструмент для аналізу і вивчення взаємозв'язків між інформаційними потоками в рамках діяльності підприємства.

При побудові моделі завжди береться до уваги дві основних глобальні області, кожна з яких відповідає велика кількість характерних об'єктів. Першою з цих областей є сукупність фізичних і інтелектуальних об'єктів, таких, як місця, люди, ідеї, речі і т.д., а також всі залежності між об'єктами та властивості цих об'єктів. Друга область включає в себе існуючі інформаційні відображення об'єктів першої області і їх властивостей. Головною функцією web-додатку є взаємозв'язок між користувачем

та системою, завдяки запиту користувача, який у подальшому обробляється системою та повертає відповідь.

Декомпозиція головної функції додатку представлена на рисунку 2.2

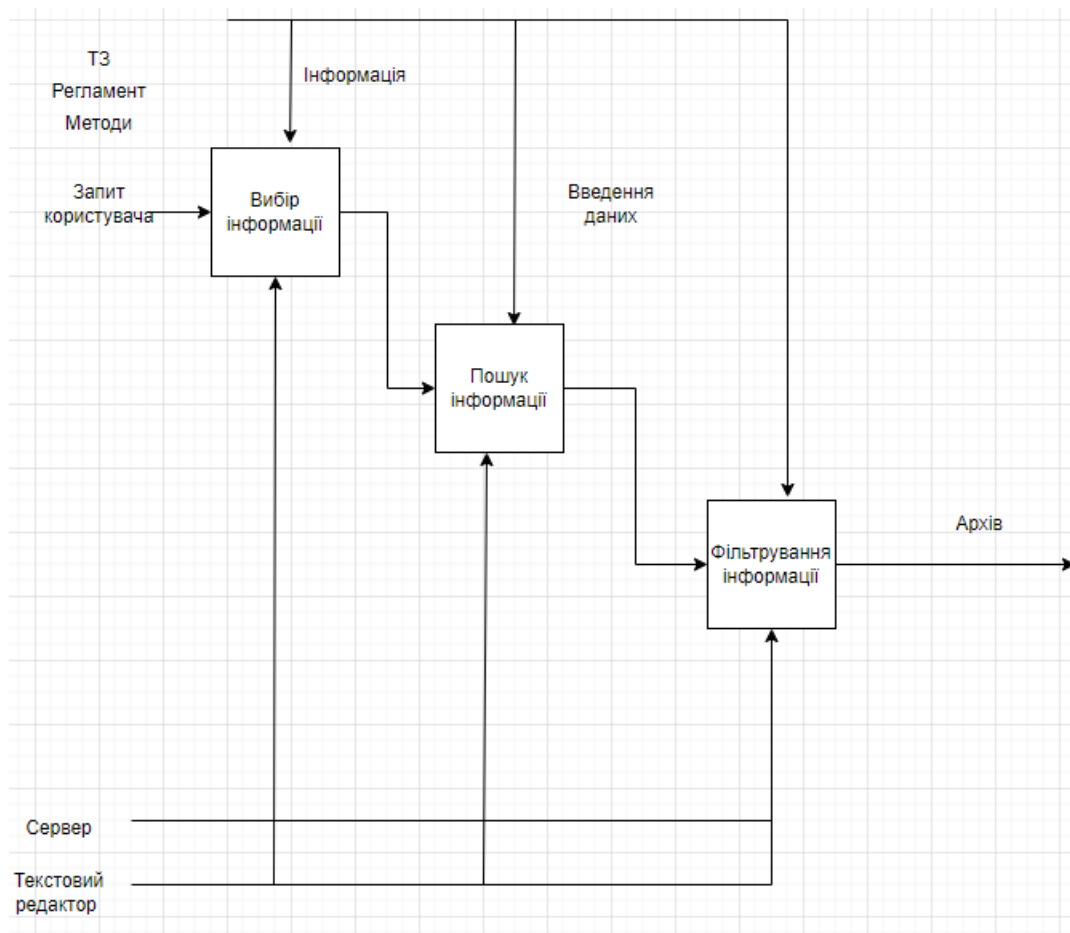


Рисунок 2.2 – Декомпозиція головної функції Web-додатку з вивчення дисципліни «Розробка ігрових додатків»

## 2.2 UML-моделі web-додатку з вивчення дисципліни «Розробка ігрових додатків»

Моделювання складається з пов'язаних між собою етапів. Процес моделювання починається з абстрактної концептуальної схеми, після неї створюються логічна та фізична моделі.

Для досягнення цих цілей спочатку будується модель у формі так званої діаграми варіантів використання (use-case diagram), яка описує функціональне призначення системи. Діаграма варіантів використання є вихідною концептуальною моделлю системи в процесі її проектування та розробки.



Рисунок 2.3 - Діаграма варіантів використання основних функцій web-додатку з вивчення дисципліни «Розробка ігрових додатків»

Основними поняттями об'єктно-орієнтованого підходу, які є фундаментальними є поняття об'єкта і класу, які представлені наборами сутностей. Клас аналізу – абстрактна сутність, та більш абстрактна ніж клас, вона являє собою набір з одного або більше класів. Таким чином, клас аналізу – це укрупнена абстракція, яка на концептуальному рівні (без точного визначення атрибутів і операцій) описує деякий фрагмент системи.

Існує три види класів аналізу:

- граничний;
- керуючий;
- сутності.

Діаграма класів аналізу по суті є прообразом класичної діаграми класів.

Діаграма класів аналізу web-додатку с вивчення дисципліни «Розробка ігрових додатків» представлена на рисунку 2.4:

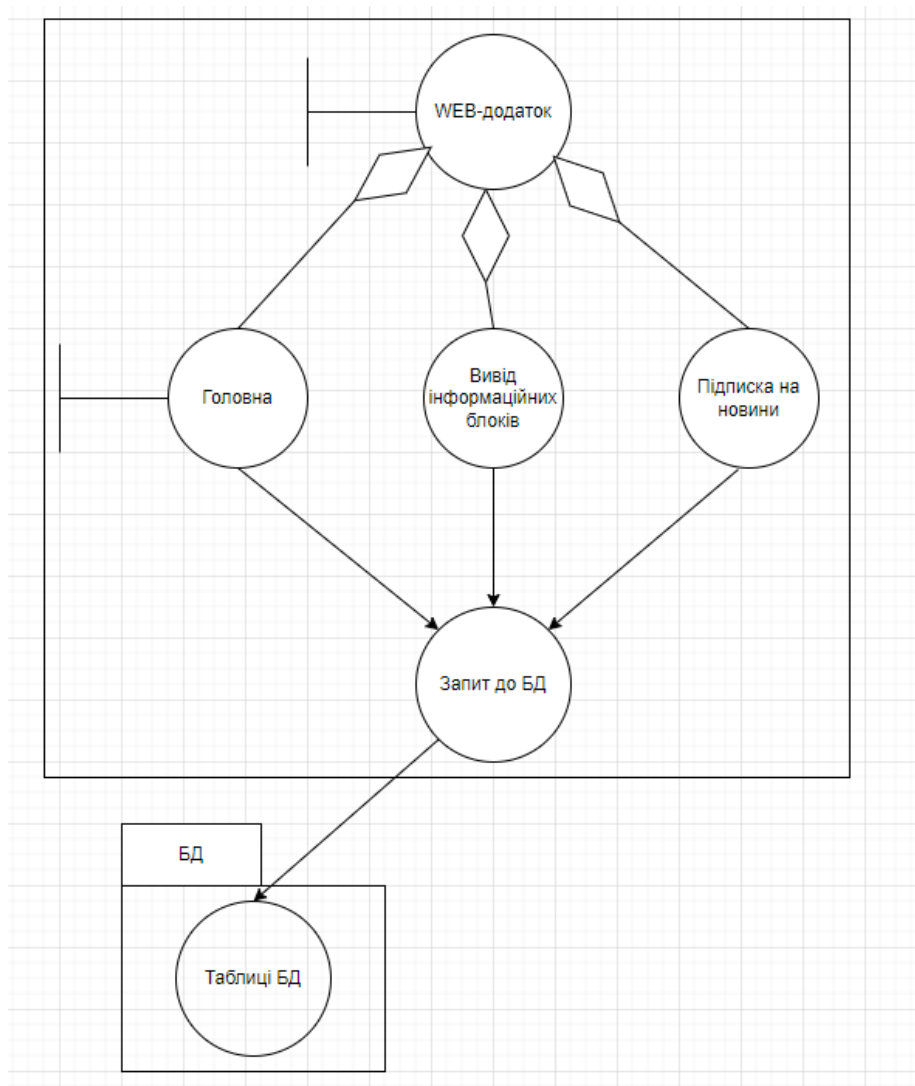


Рисунок 2.4 – Діаграма класів аналізу фрагменту Web-додатку

Діаграма комунікації – це спосіб який описує поведінку, семантично еквівалентний діаграмі послідовності. Фактично, це такий же опис послідовності обміну повідомленнями взаємодіючих примірників класифікаторів, тільки виражене іншими графічними засобами.

Таким чином, на діаграмі комунікації також як і на діаграмі послідовності застосовують один основний тип сутностей - екземпляри взаємодіючих



класифікаторів і один тип відносин - зв'язки. Однак тут акцент робиться не на часі, а на структурі зв'язків між конкретними екземплярами.

На рисунку 2.5 представлена діаграма послідовності додатку:

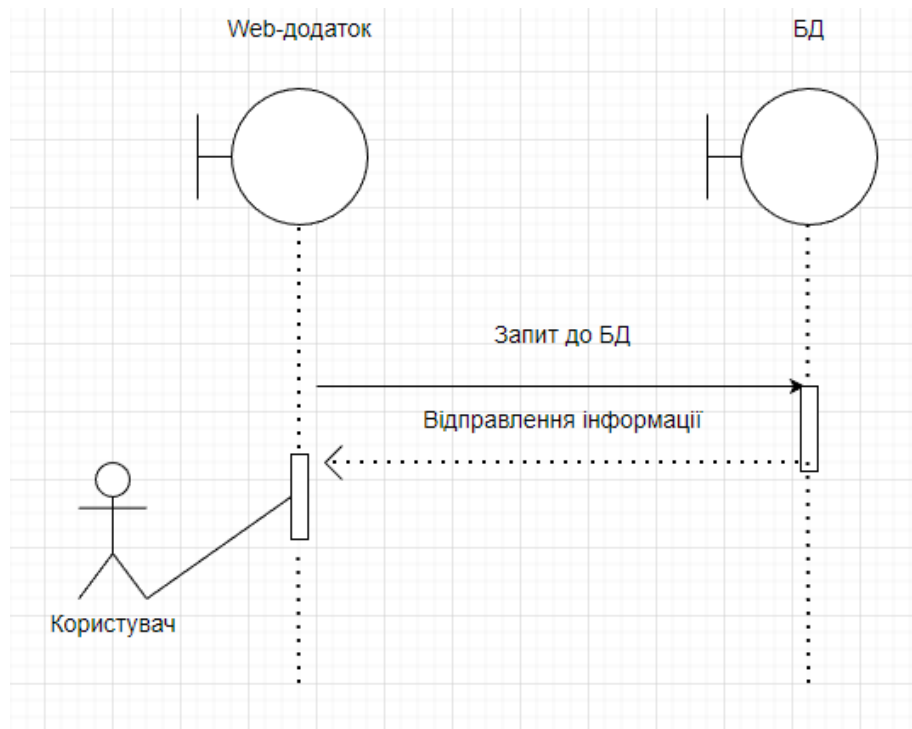


Рисунок 2.5 – Діаграма комунікації елементів web-додатку

Діаграма UML класу - найпоширеніший тип діаграм для документації програмного забезпечення. Оскільки велика кількість програмного забезпечення, створюється на основах об'єктно-орієнтованого програмування, зазначених вище, використання діаграм класів є доцільним для документування програмного забезпечення. Це відбувається тому, що ООП базується на класах та відносинах між ними.

При моделюванні поведінки проектованої або аналізованої системи виникає необхідність не тільки представити процес зміни її станів, але і деталізувати особливості алгоритмічної і логічної реалізації виконуваних системою операцій.

Діаграма діяльності - ще один спосіб опису поведінки користувача, який візуально нагадує блок-схему алгоритму. Використовується для відтворення процесу виконання операцій користувачем.

Процес зосереджений не на тому, що виробляється, а на наборі видів діяльності, які ведуть одна до одної, та те, як вони взаємопов'язані, з чітким початком і кінцем.

Розглянемо приклад діаграми діяльності на функції підписки на новини. В діаграмі присутні кроки відтворення користувачем, а саме заповнення даних про користувача, натискання кнопки підписатися, та в залежності від правильності введених даних запит до БД або повернення повідомлення про помилку.

На рисунку 2.6 представлена діаграма діяльності функції підписки на новини:

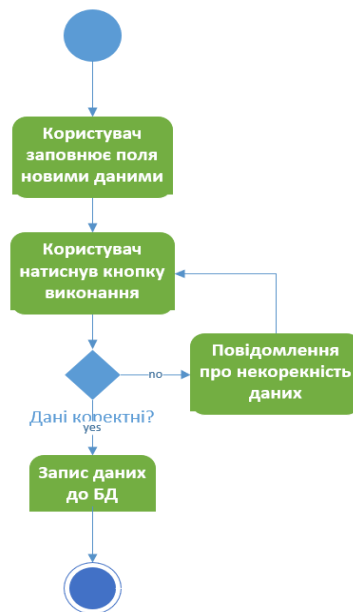


Рисунок 2.6 – Діаграма діяльності функції підписки на новини

Для створення конкретної фізичної системи необхідно реалізувати всі елементи логічного опису в конкретних матеріальних елементах. Для опису таких реальних елементів призначене фізичне подання моделі. У мові UML це означає сукупність зв'язаних елементів, включаючи програмне і апаратне забезпечення, а також

персонал, для виконання спеціальних завдань. Для фізичного подання моделей систем використовуються діаграми реалізації. Діаграма компонентів дозволяє визначити архітектуру системи, що розробляється. Основними графічними елементами діаграми компонентів є компоненти, інтерфейси і залежності між ними.

Діаграми розгортання використовуються для візуалізації зв'язку між програмним та апаратним забезпеченням. За допомогою діаграм розгортання можливо побудувати фізичну модель того, як програмні компоненти (артефакти) розгортаються на апаратних компонентах, відомих як вузли.

Типова спрощена схема розгортання веб-програми включає:

- Вузли (сервер додатків та сервер баз даних)
- Артефакти (клієнт програми та схема бази даних)

Вузли містять артефакти. Схема бази даних працює на сервері бази даних, а клієнт програми працює на сервері додатків.

Схема розгортання вказує, де саме розгортається кожен компонент програмного забезпечення.

На рисунку 2.7 представлена діаграма розгортання:

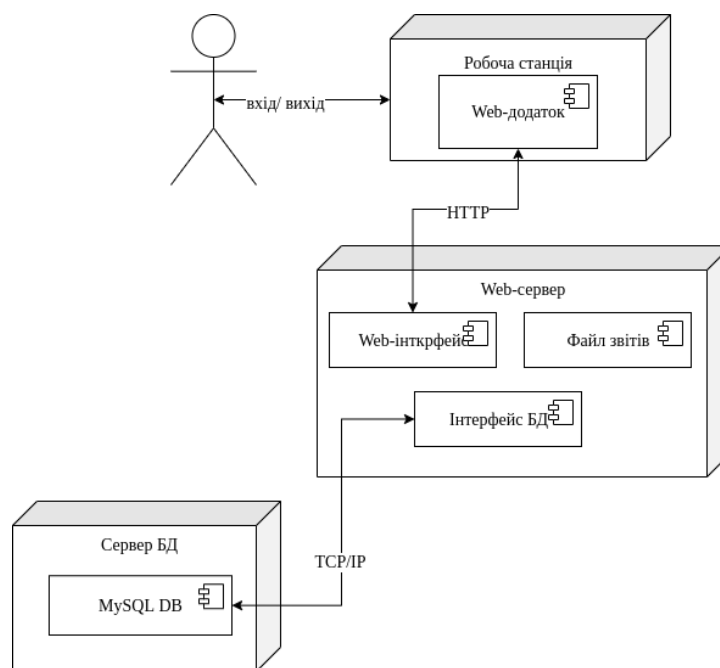


Рисунок 2.7 – Діаграма розгортання компонентів Web-додатку

### 2.3 Проектування бази даних web-додатку з вивчення дисципліни «Розробка ігрових додатків»

Під час дипломного проектування була розроблена база даних, яка містить 2 таблиці: користувачі – для збереження інформації про користувачів для відправки новин та реклами, новини – таблиця, яка містить інформацію про цікаві новини, яка в подальшому під час додавання в базу даних оновлюється на сайті.

Описання полів таблиць бази даних наведено в табл. 2.8 – 2.9.

Таблиця 2.8 – Описання таблиці “User”

Ім'я поля сутності	Тип даних	Інформація, що зберігається
Id	INT	ID користувача
Email	TEXT	Пошта користувача
Name	TEXT	Ім'я

Таблиця 2.9 – Описання таблиці “News”

Ім'я поля сутності	Тип даних	Інформація, що зберігається
Id	INT	ID замовника
Img	TEXT	Посилання на зображення
Name	VARCHAR	Назва блоку
Description	VARCHAR	Описання блоку

Логічна модель бази даних має вигляд (рис. 2.10):

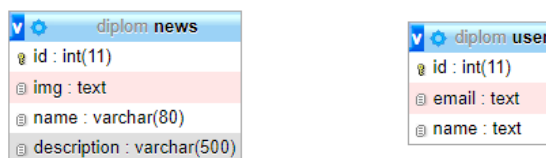


Рисунок 2.10 – Логічна модель бази даних Web-додатку з вивчення дисципліни «Розробка ігрових додатків»

В даній базі даних відсутні ключі для зв'язку таблиць. Це пов'язано с відсутністю необхідності зв'язку між таблицями, тому що вони не залежать один від одного.

## 3 ПРОГРАМНА РЕАЛІЗАЦІЯ

### 3.1 Розробка архітектури програмного додатку

Web-додаток був розробляємо згідно технічному завданню.

Метою створення даного веб-сайту є залучення та зацікавлення нової клієнтської бази до користування системою з вивчення дисципліни “Розробка ігрових додатків”. Також можливе збільшення зацікавленості серед студентів, які вивчають дану дисципліну. Система буде мати усі сучасні критерії та зможе бути конкурентоспроможною на ринку ІТ послуг.

Головна задача Web-додатку це надання можливості ознайомлення з дисципліною, пошуку нових цікавих матеріалів та поліпшення навичок та знань у сфері “Розробки ігрових додатків”. Згідно технічному завданню сайт був розроблений за використанням наступних мов програмування: MySQL, PHP, JavaScript, HTML та CSS.

Інтерфейс додатку був розроблений максимально зрозуміло для користувача, вся інформація та структура сайту інтуїтивно зрозуміла для нового користувача. Основні навігаційні елементи представлені у виді кнопок з посиланнями. Стиль Web-додатку можна описати як сучасний, діловий. Під час розробки сайту були створенні наступні розділи:

- Головна сторінка – сторінка, яка знайомить користувача з системою та дає можливість перейти на сторінки цікавих матеріалів та завдань для самоперевірки.
- Сторінка завдань – сторінка на якій користувач може переглянути за посиланням прості завдання для самоперевірки.
- Сторінка з цікавими матеріалами – сторінка, на якій розташовуються блоки з цікавими новинами, статтями та корисними матеріалами. Блоки динамічно створюються після додавання інформації у базу даних.
- Сторінка контактів – сторінка, яка містить інформацію про розміщення та можливості зв’язку.

- Сторінка зворотного зв'язку – сторінка, на якій відображається інформація про команду, яка може відповісти на цікавлячі питання.
- Сторінка підписки на новини – сторінка, на якій розміщена форма для заповнення користувачем. Після заповнення форми користувачеві будуть приходити новини на пошту.

На рисунку 2.1 зображено макет веб-сайту

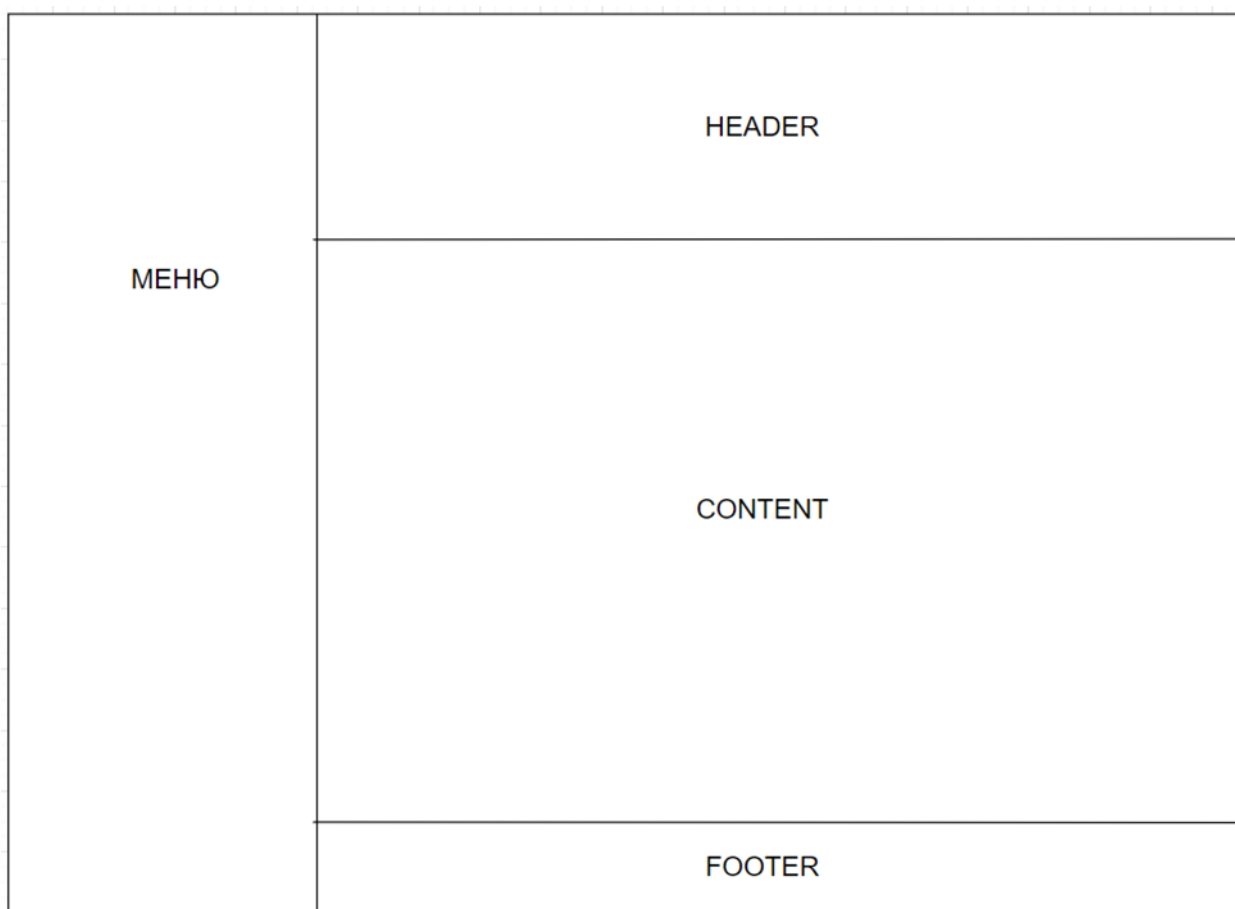


Рисунок 3.1 – Макет Web-додатку

Спираючись на макет зображений на рисунку 3.1 та технічне завдання була розроблена головна сторінка сайту, на якій розташовані основні навігаційні елементи, завдяки яким відбувається основна навігація сторінками сайту. На головній сторінці є можливість перейти на сторінку завдань, сторінку цікавих матеріалів та сторінку зворотного

зв'язку. Навігаційне меню розташоване у лівому блоці сайту та зафіксоване на кожній сторінці. Головна сторінка зображена на рисунку 3.2.



Рисунок 3.2 – Головна сторінка Web-додатку

### 3.2 Програмна реалізація web-додатку

Насамперед перед створенням Web-додатку в першу чергу обираємо зручний редактор коду. Під час створення дипломного проекту була обрана програма Brackets, яка має зручний та зрозумілий інтерфейс. Також програма містить велику кількість доповнень, наприклад Autosave Files on Window Blur, це автоматичне зберігання файлу при згортанні програми або при втраті фокусу на редакторі. Ще одним зручним доповненням є Emmet – плагін, який дозволяє значно прискорити написання коду HTML та CSS, використовуючи скорочення та аббревіатури. Наприклад при після написання знаку (!) та кліку клавіши tab, додається повноцінна html сторінка. Середовище Brackets та робота плагіну зображена на рисунках 3.3-3.5.



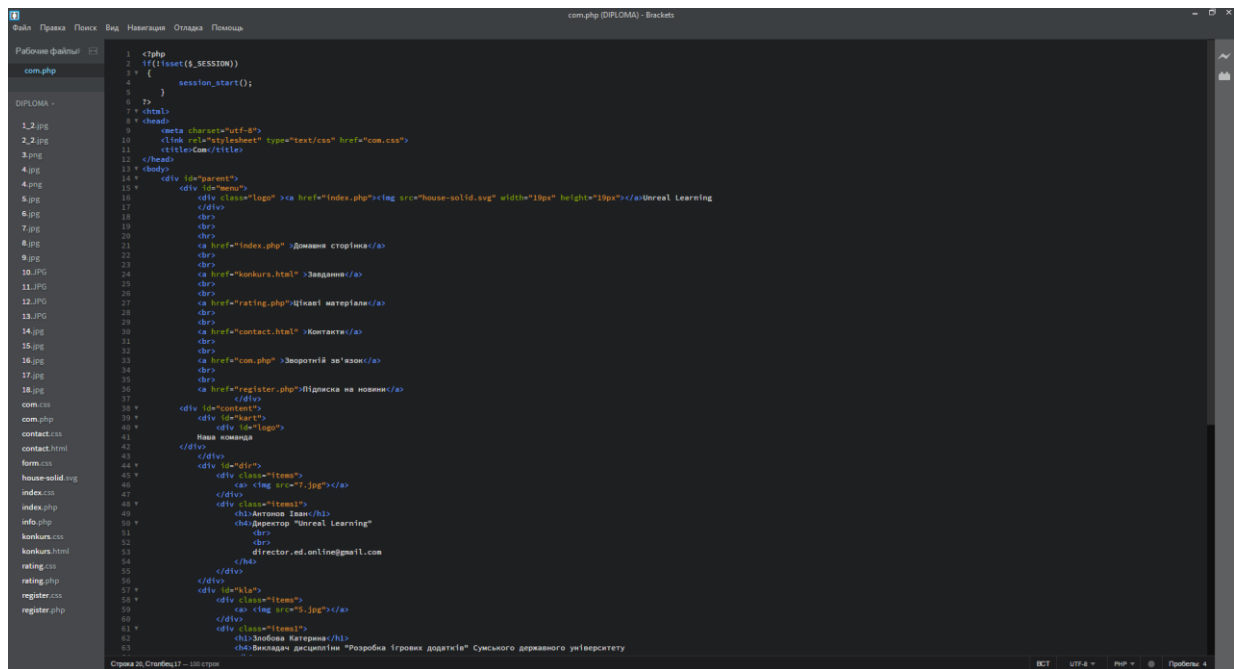


Рисунок 3.3 – Середовище розробки “Brackets”

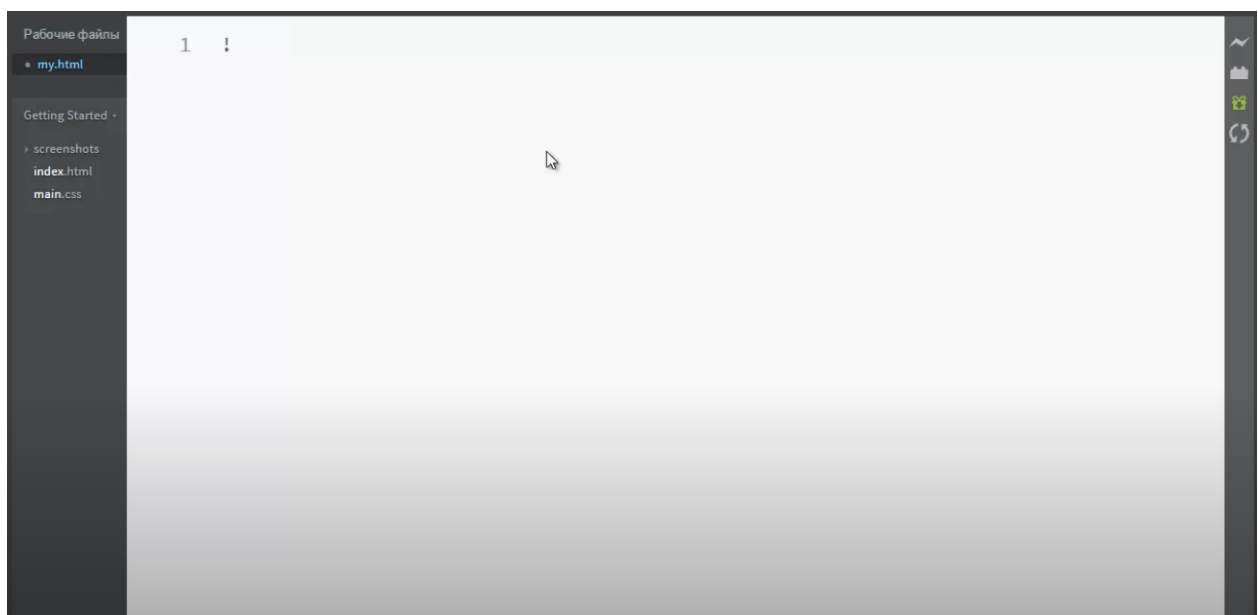
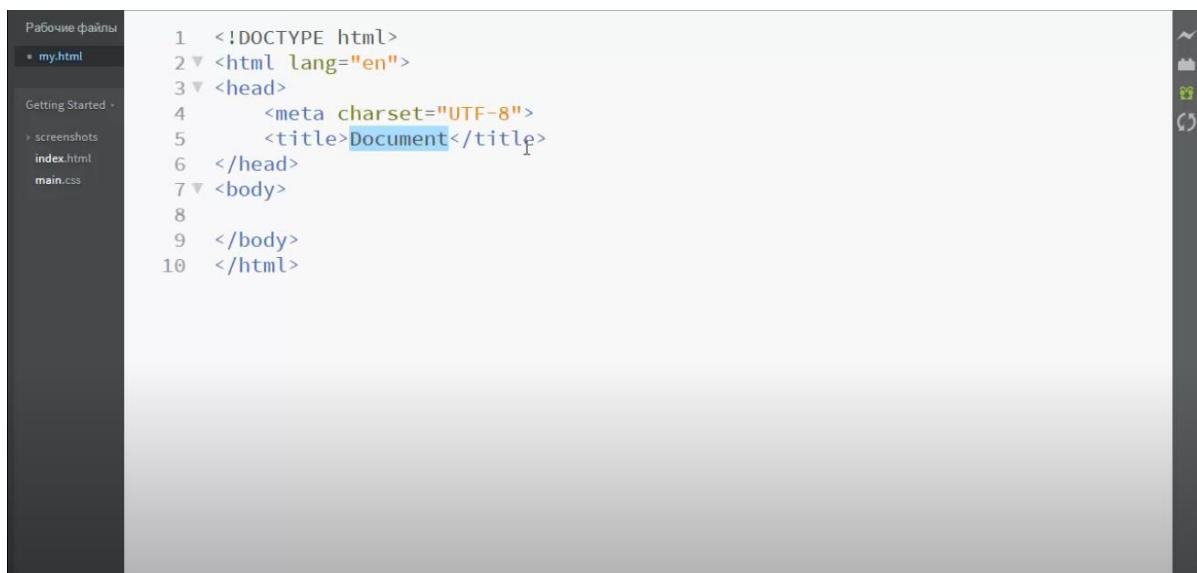


Рисунок 3.4 – Ініціалізація команди плагіну Emmet



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Document</title>
6 </head>
7 <body>
8
9 </body>
10 </html>
```

Рисунок 3.5 – Результат виконання команди плагіну Emmet

Головне навігаційне меню було створене засобами CSS (рис. 3.6). Меню розташоване у лівому блоці сторінки та зафіксоване на всіх сторінках. Воно є інтуїтивно простим та легким для використання. Дизайн можна описати як лаконічний, переважними є світлі кольори. Кожен пункт меню має посилання на відповідну сторінку.

 Unreal Learning

---

[Домашня сторінка](#)

[Завдання](#)

[Цікаві матеріали](#)

[Контакти](#)

[Зворотній зв'язок](#)

[Підписка на новини](#)

Рисунок 3.6 – Навігаційне меню Web-додатку

На головній сторінці також присутні основні блоки сайту та створенні посилання, котрі перенаправляють користувача на відповідну сторінку. Зробити це можна клікнувши на зображення. Також на кожній сторінці розроблений “header” сайту, так звана шапка сайту, яка закріплена для кожної сторінки. Шапка сайту розроблена засобами CSS та HTML. В кодї HTML створені блоки з інформацією, а в CSS цим блокам описані стилі. Підвал сайту, так званий “footer” розроблений за таким же принципом та закріплений на усіх сторінках сайту. Описані вище елементи можна переглянути на рисунку 3.7.



Рисунок 3.7 – Основні елементи сайту (header, footer, зображення з посиланнями)

На сторінці “Зворотній зв’язок” користувач може ознайомитися с членами команди, яка професійно-кваліфікована та може відповісти користувачам на цікавлячи їх питання. Дана сторінка реалізована за допомогою звичайних flex блоків, які за допомогою засобу CSS: `flex-direction: row;` розташовані в рядок та гнучкі при додаванні нових блоків (рис. 3.8).

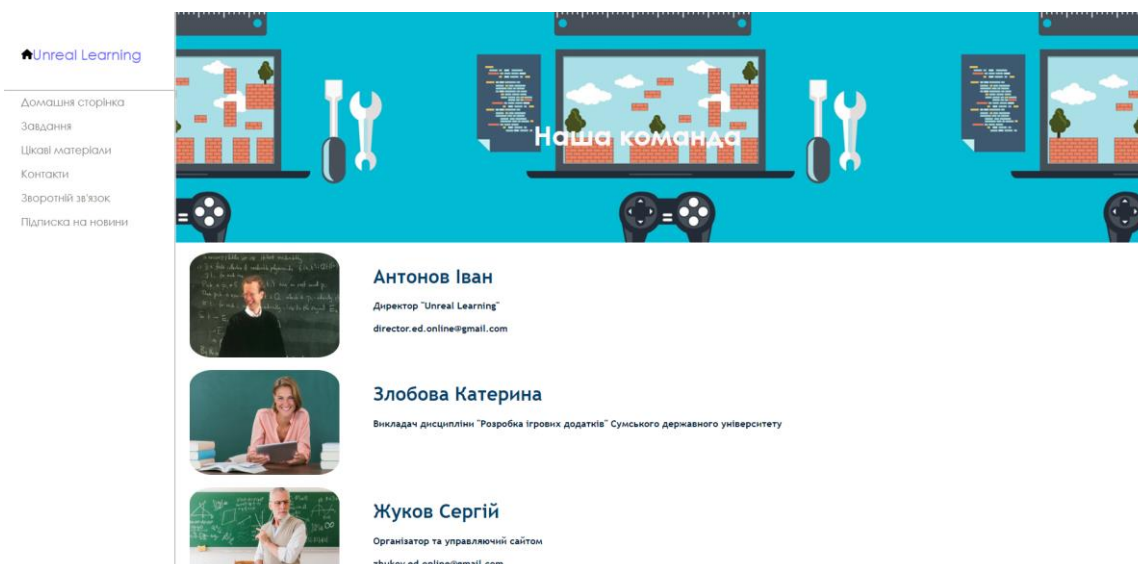


Рисунок 3.8 – Сторінка членів команди для зворотного зв'язку

На сайті є можливість переглядати прості завдання для самоперевірки користувачів. Така можливість є на сторінці “Завдання”, на якій завдання розташовані у flex блоках, які теж створені засобами CSS. Клікнувши на кнопку “Переглянути завдання” користувач потрапляє на фото-зображення завдання, що відкривається у новій вкладці. Це реалізовано за допомогою атрибута тегу `<a href="">`, а саме `“target=_blank”`. Даний функціонал зображений на рисунках 3.9-3.10.

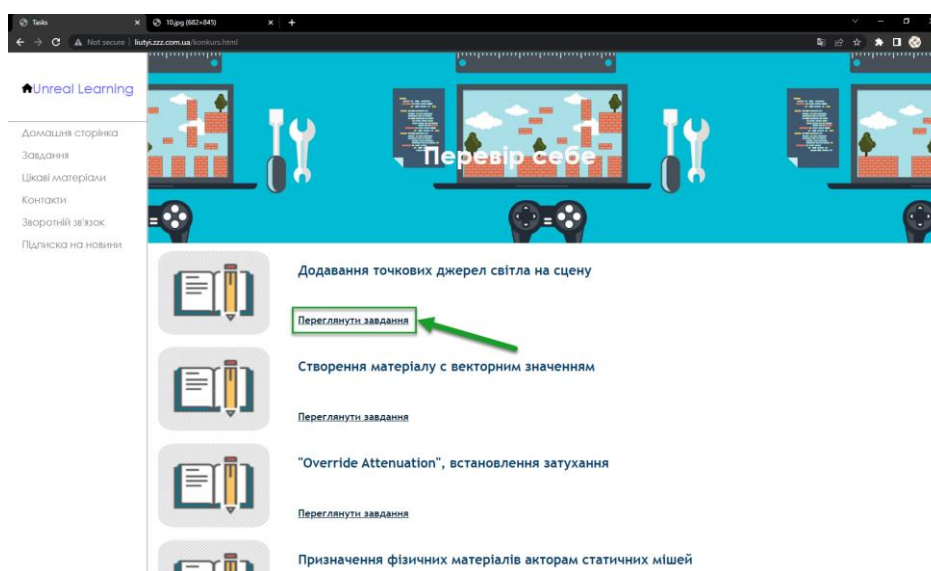


Рисунок 3.9 – Сторінка “Завдання” та посилання на фото-зображення

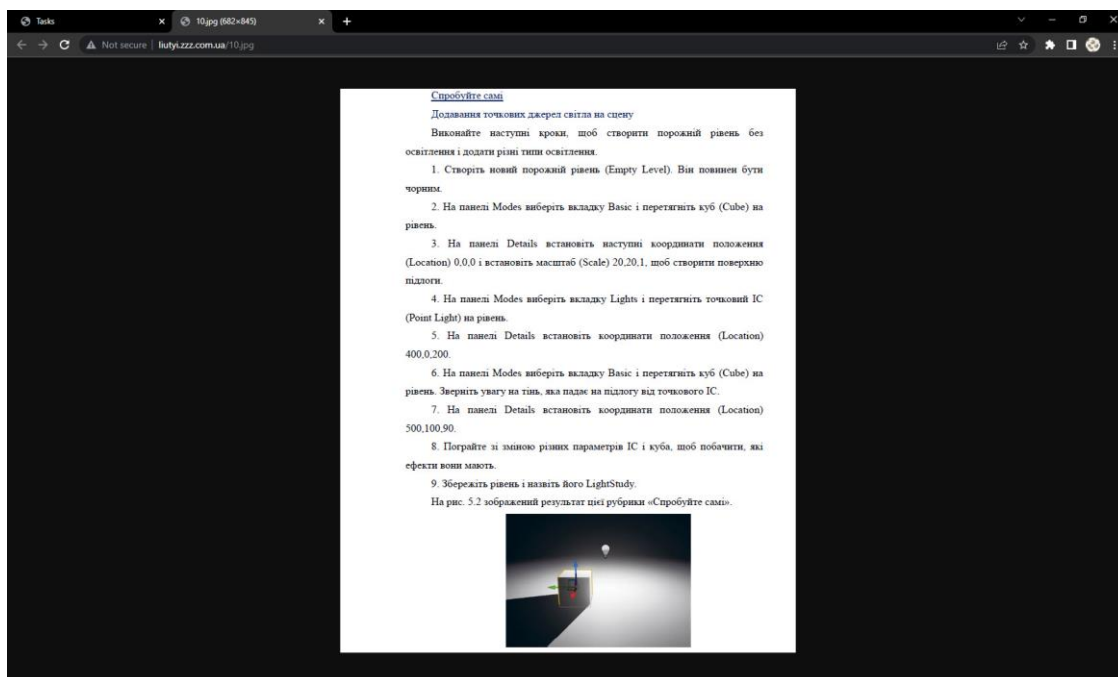


Рисунок 3.10 – Фото-зображення після кліку на кнопку “Переглянути завдання”

Під час дипломного проектування було створено 2 таблиці, завдяки яким відбувалося занесення інформації про користувачів та вивід інформаційних блоків. Таблиці бази даних зображені на рисунках 3.11-3.12.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	int(11)		Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
<input type="checkbox"/>	2	img	text	utf8mb4_general_ci	Нет				Изменить Удалить Ещё
<input type="checkbox"/>	3	name	varchar(80)	utf8mb4_general_ci	Нет	Нет			Изменить Удалить Ещё
<input type="checkbox"/>	4	description	varchar(500)	utf8mb4_general_ci	Нет	Нет			Изменить Удалить Ещё

Рисунок 3.11 – Таблица “News” з якої відбувається додавання інформаційних блоків на сайт

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	int(11)		Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
<input type="checkbox"/>	2	email	text	utf8mb4_general_ci	Нет				Изменить Удалить Ещё
<input type="checkbox"/>	3	name	text	utf8mb4_general_ci	Нет				Изменить Удалить Ещё

Рисунок 3.12 – Таблица “User” в яку додаються дані про користувачів

Цікаві матеріали та інші новини користувач може переглянути на сторінці “Цікаві матеріали”. На сторінці зображені блоки з інформацією, які динамічно додаються на сторінку після додавання інформації у таблицю “News”. Сторінка “Цікаві матеріали” зображена на рисунку 3.13.



Рисунок 3.13 – Сторінка “Цікаві матеріали”

Даний функціонал був реалізованих за допомогою PHP та MySQL. Для початку створено блок для додавання інформації та стилі для нього.

```
<article class="kanovada_casuneg">
  <div class="anesun">
    
  </div>
  <div class="damatsiya">
    <h2>Курс від Unreal Engine</h2>
    <div class="tekstura_novostey">
      <p>Привіт! Це п'ятий випуск геймдев-дайджесту, де ми зібрали найцікавіші знахідки за тиждень. У цьому випуску ви знайдете генератори підземель і будинків,
```

асет з 3D-моделями середовища й реквізиту, інструкцію зі створення Wordle на Unity та багато інших ресурсів.</p>

```
</div>
<div class="esumaku_dupena">
  <a class="novostn_aposled" href="https://gamedev.dou.ua/blogs/gamedev-
digest-5/" target="_blank">Детальніше</a>
</div>
</div>
</article>
```

Після того як блоки були створені завдяки засобам CSS та HTML, підключаємо базу даних для можливості реалізації її можливостей. Це було реалізовано наступним чином.

```
<?php
$a=new
mysqli("mysql.zzz.com.ua","dmitriiliutyi","Lbvflbvfl23","dmitriiliutyi");
$pr1 = $a->query("SELECT name,description FROM `news`");
?>
```

Після підключення до бази даних, було створено запит для подальшого відображення та обробки інформації. Після отримання даних з запиту, була реалізована його обробка та динамічне відображення наступним чином.

```
<article class="kanovada_casuneg">
  <div class="anesun">
    
  </div>
  <div class='damatsiya'>
    <?php
    while($row = mysqli_fetch_array($pr1,MYSQLI_ASSOC)) {

    echo
    "<h2>" . $row ['name'] . "</h2".
    "<div class='tekstura_novostey'>".
    "<p>" . $row ['description'] . "</p".
    "</div>"
```

```

;
}

?>
        <div class="esumaku_dupena">
            <a class="novostn_aposled" href="https://gamedev.dou.ua/blogs/gamedev-
digest-5/" target="_blank">Детальніше</a>
        </div>
    </div>
</article>

```

Після інтеграції даного функціоналу до файлу сторінки “Цікаві матеріали”, блоки почали динамічно створюватися після додавання інформації до бази даних.

На сторінці “Підписка на новини” був реалізований функціонал підписки користувача на новини з подальшою розсилкою новин на пошти, які користувач вказує при заповненні форми. Сторінка зображена на рисунку 3.14.

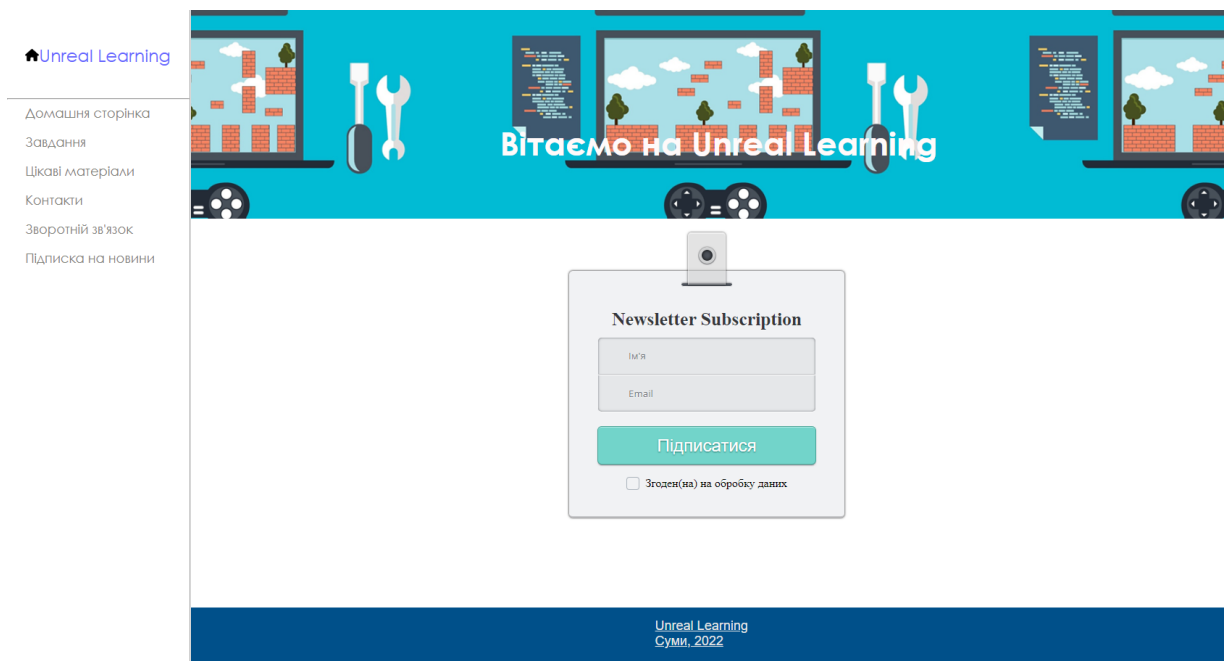


Рисунок 3.14 – Сторінка “Підписка на новини”



Форма була створена звичайними засобами HTML та CSS. Перевірка введеної інформації відбувається завдяки підключенню модулів JavaScript. Валідація даних зображена на рисунку 3.15.

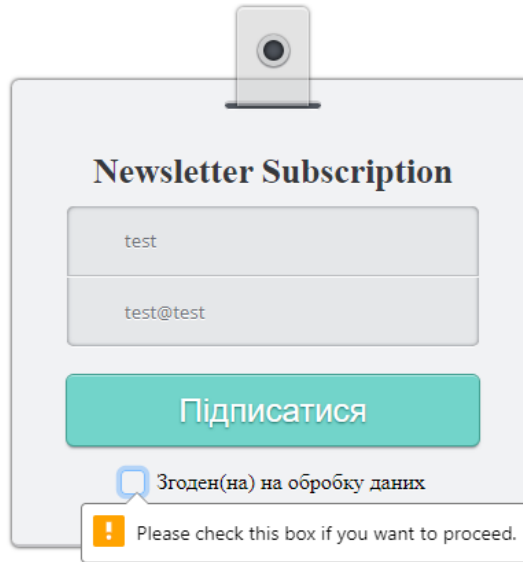


Рисунок 3.15 – Валідація введеної інформації

Помилки адаптуються під мову користувача встановлену у браузері. Також присутня перевірка на можливість повторної підписки на новини. Дана перевірка також реалізована за допомогою JavaScript наступним чином.

```
if ($email!=NULL){
$check = $a->query("SELECT id FROM user WHERE email='$email'");
$myrow = mysqli_fetch_array($check);
if (!empty($myrow)) {
    $errmsg = 'Вибаачте, ви вже підписані.';
    echo "<script type='text/javascript'>alert('$errmsg');</script>";
    exit ();
}
else {
//    $insert=FALSE;
$insert = $a->query("INSERT INTO user(email, name) VALUES('$email', '$name')");
if ($insert=='TRUE')
```

```
{
  $regi= 'Ви успішно підписалися';
echo "<script type='text/javascript'>alert('$regi');</script>";
  header("refresh: 1; url = http://liutyi.zzz.com.ua/index.php"); } } }
```

Після валідації всіх полів та відправки форми користувачем, виконується запит до бази даних та створення нових записів у таблиці. Це реалізовано за допомогою методу POST та запиту INSERT.

```
$email=NULL;
$email = isset($_POST['email']) ? $_POST['email'] : '';
$name = isset($_POST['name']) ? $_POST['name'] : '';
$insert = $a->query("INSERT INTO user(email, name) VALUES('$email', '$name')");
```

### 3.3 Тестування web-додатку

Найважливішим процесом у життєвому циклі створення програмного забезпечення є процес тестування. Саме завдяки процесу тестування, зростає впевненість у якості продукту та стає доступною можливість випуску продукту та працездатність кінцевої версії продукту.

Зважаючи на всі ці фактори, під час дипломного проектування на стадії хостингу проекту було прийнято рішення у детальному тестуванні. Для спрощення процесу тестування та можливості тестування незалежними експертами були створенні тест кейси, які зображені на рисунку 3.16

№	A	B	C	D	E	F
1	Given	Summary	Preconditions	Steps to reproduce	Expected result	Status
1	Given that user is navigated to the "Tasks" page When user is clicked the link Then the task is opened	Verify that the task is opened in a new tab after user is clicking the link on the "Tasks" page	1. The home page is opened	1. Navigate to the "Tasks" page 2. Click the link 3. Pay attention to the result	The task is opened in a new tab after user is clicking the link on the "Tasks" page	Passed
2	And task is displayed in a new tab					
3	Given that user is adding the data into the DB When user is navigated to the "News" page Then the information block is displayed correctly	Verify that the information block is displayed on the "News" page after adding the data into the DB	1. The user is authorized in PhpMyAdmin	1. Add the data into the News table in PhpMyAdmin 2. Navigated to the News page 3. Pay attention to the information block	The information block is displayed on the "News" page after adding the data into the DB	Passed
4	Given that user is navigated to the "Newsletter" page And left the blank "Name" field When user is clicked the "Підписатися" button Then the error message is displayed	Verify that the error message is displayed after leaving the blank "Name" field	1. The "Newsletter" page is displayed	1. Left the blank "Name" field 2. Click the "Підписатися" button 3. Pay attention to the result	The error message is displayed after leaving the blank "Name" field	Passed
5	Given that user is navigated to the "Newsletter" page And left the blank "Email" field When user is clicked the "Підписатися" button Then the error message is displayed	Verify that the error message is displayed after leaving the blank "Email" field	1. The "Newsletter" page is displayed	1. Left the blank "Email" field 2. Click the "Підписатися" button 3. Pay attention to the result	The error message is displayed after leaving the blank "Email" field	Passed
6	Given that user is navigated to the "Newsletter" page And left the blank checkbox When user is clicked the "Підписатися" button Then the error message is displayed	Verify that the error message is displayed after leaving the blank checkbox	1. The "Newsletter" page is displayed	1. Left the blank checkbox 2. Click the "Підписатися" button 3. Pay attention to the result	The error message is displayed after leaving the blank checkbox	Passed

Рисунок 3.16 – Тестові сценарії для перевірки додатку

Після створення основних тест кейсів, вони були адаптовані під мову Gherkin. В сучасних методиках розробки програмних продуктів все більш процесів стає автоматизованими, звичайно це стосується и тестування. Розроблена система в подальшому буде використовувати автоматизацію для виконання регресійного тестування після внесення певних змін в додаток. Тому для спрощення процесу автоматизації використовується Gherkin.

Після тестування додатку, усі тест кейси пройшли перевірку, тому можна вважати, що Web-додаток готовий до релізу та хостингу.

### 3.4 Хостинг web-додатку

Для розміщення Web-додатку в Інтернеті потрібно використовувати ресурси хостингу. Було обрано систему ZZZ.com.ua Hosting. Вона є безкоштовною на певний проміжок часу, після чого можна буде продовжити використання домену за додаткову плату. Система є легкою та зрозумілою для користувача. Для того, щоб розмістити сайт у мережі Інтернет потрібно лише створити базу даних та загрузити файли до FTP репозиторію. Система ZZZ.com.ua Hosting зображена на рисунку 3.17.

The screenshot displays the control panel for a ZZZ.com.ua Hosting account. The interface is in Ukrainian and shows the following details for the 'Огляд PRO #1' account:

- Статус:** Активний (Active)
- Пакет послуг:** Pro (Термін дії закінчується: 2023-05-27)
- Кількість безкоштовних доменів:** 1 / 2 (50%)
- Домен:** lutyi.zzz.com.ua
- IP адреса вашого серверу:** 95.211.16.67
- Акаунти електронної пошти:** 0 / 2 (0%)
- Акаунти FTP:** 0 / 2 (0%)
- Користувачі MySQL:** 1 / 1 (100%)
- Бази даних MySQL:** 1 / 1 (100%)
- Сюп-завдання:** 0 / 1 (0%)
- Програми:** 0 / 2 (0%)
- Захищені розділи:** 0 / 50 (0%)
- Користувачі та групи користувачів захищених розділів:** 0 / 50 (0%)
- Максимальний розмір повідомлення (включно з вкладеннями):** 25 MB

On the right side, there is a circular progress indicator for 'Використання фізичного дискового простору' (Physical disk space usage) showing 0% usage. A legend indicates that the green area represents 'Використаний дисковий простір' (Used disk space) and the grey area represents 'Використана кількість файлів' (Used number of files).

Рисунок 3.17 – Система хостингу ZZZ.com.ua Hosting

## ВИСНОВКИ

Під час виконання дипломного проекту була проаналізована предметна область та визначена актуальність розробки продукту. Було виконане порівняння та аналіз сайту аналогів, відповідно створена таблиця порівняння, яка в подальшому вплинула на розробку. А саме дала проаналізувати основні переваги та недоліки аналогів, і створити більш сучасну систему.

Після детального аналізу предметної області та порівняння аналогів, були розглянуті основні технології для розробки Web-додатків. Було досліджено та виділено основні переваги багатьох мов програмування. Після аналізу яких, було прийнято рішення про обрання HTML, CSS, JavaScript та MySQL, як основних інструментів розробки. Для адміністрування базами даних було обрано систему PhpMyAdmin.

У ході виконання дипломної роботи весь дизайн та функціонал Web-додатку було узгоджено з керівником проекту. Після тестування та проходження основних тест кейсів сайт було розміщено на хостингу ZZZ.com.ua Hosting.

Результатом даного дипломного проекту є Web-додаток з вивчення дисципліни “Розробка ігрових додатків”.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Що таке інформаційні технології [Електронний ресурс] – Режим доступу до ресурсу: <http://apers.kpi.ua/shcho-take-informatsiini-technologii> (дата звернення: 05.03.2022).
2. Готовая CMS или самописный сайт? [Електронний ресурс] – Режим доступу до ресурсу: <https://hyperhost.ua/info/gotovaya-cms-ili-samopisnyiuy-sayt/> (дата звернення: 06.02.2022).
3. ГЛАВА 4 Диаграмма вариантов использования (use case diagram) [Електронний ресурс] – Режим доступу до ресурсу: <http://khpriip.mipk.kharkiv.edu/library/case/leon/gl4/gl4.html#1> (дата звернення: 10.02.2022).
4. Методология IDEF0 [Електронний ресурс] – Режим доступу до ресурсу: [https://studme.org/87184/ekonomika/metodologiya\\_idef0](https://studme.org/87184/ekonomika/metodologiya_idef0) (дата звернення: 10.12.2021).
5. ДИАГРАММЫ АВТОМАТОВ [Електронний ресурс] – Режим доступу до ресурсу: [https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema12/tema12\\_3](https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema12/tema12_3) (дата звернення: 15.05.2022).
6. Рекомендация: Диаграмма последовательности [Електронний ресурс] – Режим доступу до ресурсу: [http://dit.isuct.ru/Publish\\_RUP/core.base\\_rup/guidances/guidelines/sequence\\_diagram\\_AFA76EBB.html](http://dit.isuct.ru/Publish_RUP/core.base_rup/guidances/guidelines/sequence_diagram_AFA76EBB.html) (дата звернення: 11.04.2022).
7. ГЛАВА 5 Диаграмма классов (class diagram) [Електронний ресурс] – Режим доступу до ресурсу: <http://khpri->

- [ip.mipk.kharkiv.edu/library/case/leon/gl5/gl5.html](http://ip.mipk.kharkiv.edu/library/case/leon/gl5/gl5.html) (дата звернення: 18.05.2022).
8. Освітній портал «Академія». URL: <https://academia.in.ua/content/основні-форми-тенденції-та-перспективи-впровадження-дистанційних-технологій-навчання-у> (Дата звернення 26.05.2022).
  9. Мови програмування для розробки сайту. URL: <https://ukrbukva.net/94061-Yazyki-programmirovaniya-dlya-razrabotki-saiyta.html> (Дата звернення 21.04.2022)
  10. Вікіпедія. URL: <https://uk.wikipedia.org/wiki/HTML> (Дата звернення 29.04.2022)
  11. Персональний блог Web-систем дистанційного навчання. URL: <https://igorosa.com/web-cistemy-distancionnogo-obucheniya/> (Дата звернення 23.04.2022)
  12. Айзенк, Г. Супертести IQ / Г. Айзенк. - М.: Эксмо, 2014. - 749 с. (Дата звернення 21.04.2022)
  13. Radu C. Top UI / UX design tools. Medium. [Електронний ресурс] // Режим доступу до ресурсу: <https://uxplanet.org/top-ui-ux-design-tools-818d2faca2a1> (дата звернення: 15.04.2022).
  14. The 8 types of websites and how to design them. 99designs. [Електронний ресурс] // Режим доступу до ресурсу: <https://99designs.com/blog/web-digital/types-of-websites/> (дата звернення: 15.05.2021).
  15. Top 10 education websites to teach you for free. India Today. [Електронний ресурс] // Режим доступу до ресурсу: <https://www.indiatoday.in/education-today/featurephilia/story/free-education-953499-2017-01-06> (дата звернення: 27.05.2021).
  16. The complete guide to understand IDEF diagram | edraw max. Edrawsoft. [Електронний ресурс] // Режим доступу до ресурсу:

<https://www.edrawmax.com/article/the-complete-guide-to-understand-idef-diagram.html> (дата звернення: 10.12.2021).

17. Google for Education [Електронний ресурс] // Режим доступу до ресурсу: <https://developers.google.com/edu/python/> (дата звернення 15.02.2022).
18. METANIT.COM [Електронний ресурс] // Режим доступу до ресурсу: <https://metanit.com/java/tutorial/> (дата звернення 15.02.2022).
19. UdeMy [Електронний ресурс] // Режим доступу до ресурсу: <https://www.udemy.com/> (дата звернення 15.02.2022)
20. Is online education right for you? 5 questions answered [Електронний ресурс] // The Conversation. – 2020. – Режим доступу до ресурсу: <https://theconversation.com/is-online-education-right-for-you-5-questions-answered-130044>.

**Додаток А.**

**ТЕХНІЧНЕ ЗАВДАННЯ**

**на розробку**

**«Web-додаток з вивчення дисципліни "Розробка ігрових додатків"»**



# **1. Призначення й мета web-додатку організації діяльності кондитерського закладу «La Fit Bakery»**

## **1.1 Призначення web-додатку**

Інформаційна система повинна представляти собою веб – систему для вивчення дисципліни «Розробка ігрових додатків» .

## **1.2 Мета створення web-додатку**

Дана веб – система дозволить збільшити зацікавленість до даної дисципліни та відповідно аудиторію. Метою створення веб – системи є поліпшення знань та їх розвиток для зацікавлених осіб.

## **1.3 Цільова аудиторія**

Цільовою аудиторією даного проекту є :

- Користувач зацікавлений у навчанні з даної дисципліни.
- Користувачі які навчаються та мають дану дисципліну в університеті.

# **2 Вимоги до проекту**

## **2.1 Вимоги до проекту в цілому**

### **2.1.1 Вимоги до структури й функціонування**

Web-додаток з вивчення дисципліни “Розробка ігрових додатків” повинен бути реалізований за допомогою web-інструментів та виконувати визначений набір функцій.

Інформаційна система повинна бути реалізована у вигляді веб – сайту.

### **2.1.2 Вимоги до персоналу**

Для експлуатації від користувачів не повинно вимагатися спеціальних технічних навичок, знання технологій або програмних продуктів.

### **2.1.3 Вимоги до збереження інформації**

У системі керування додатка не передбачено механізм пошуку та збереження знайдених особистих даних.

### **2.1.4 Вимоги до розмежування доступу**

Розроблюваний web-додаток має бути загальнодоступним у мережі Інтернет. Інформація є безкоштовною. Користувачі можуть використовувати його повний функціонал.

## **2.2 Структура web-додатку**

### **2.2.1 Загальна інформація про структуру web-додатку**

До структури web-додатку входять усі його web-сторінки, які є загальнодоступними. Web – додаток буде реалізований як багато–сторінковий сайт, який містить відповідні розділи.

Веб – сайт повинен складатися з наступних розділів:

Головна сторінка – яка буде знайомити користувача с сайтом та матеріалами з вивчення дисципліни.

Сторінка контенту, а саме цікава інформація, статті та відео уроки.

У футері сайту присутній блок контактної інформації для уточнення питань.

### **2.2.2 Навігаційне меню**

Для навігації сайтом буде створене меню у лівому блоці сторінки, яке буде зафіксоване на всьому сайті.

Користувацький інтерфейс веб – сайту повинен забезпечувати наочне, інтуїтивно зрозуміле представлення структури розміщеної на ньому інформації.

### **2.2.3 Дизайн web-додатку**

Стиль веб – сайту повинен бути реалізований як сучасний, діловий. Він повинен буди інтуїтивно зрозумілим. Кольори будуть переважно чорний, білий та синій.

Види і розміри шрифтів повинні бути комфортними для перегляду. Інформаційні блоки, графічні матеріали та інші елементи web-сторінок повинні мати зручне і логічне розташування. Приблизний інтерфейс сайту можна переглянути за полсиланням. Мокап сайту створений у програмі Figma.

<https://www.figma.com/file/yIN2Kv2cgwjH3MeU8gzSfT/Diploma?node-id=0%3A1>

## **2.3 Вимоги до видів забезпечення**

### **2.3.1 Вимоги до лінгвістичного забезпечення**

Весь текст у web-додатку має бути виконаний українською або англійською мовами.

### **2.3.2 Вимоги до програмного забезпечення**

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам:

- Браузер “Google Chrome”(версія починаючи з 77 і вище).
- Браузер “Internet Explorer”(версія починаючи з 10 і вище).
- Браузер” Mozilla Firefox”(версія починаючи з 67.0 і вище).

## **2.4 Вимоги до функціонування системи**

### **2.4.1 Системні вимоги**

Проаналізувавши потреби користувачів та персоналу закладу було визначено наступні вимоги:

Система надає можливість ознайомитися з інформацією, цікавими фактами та різними навчальними матеріалами. Користувач має можливість безкоштовно переглядати інформацію та має можливість зв'язку. Для зв'язку можна скористатися контактами, які будуть розташовані в блоці с контактною інформацією.

## **3 Склад і зміст робіт зі створення**

### **web-додатку з вивчення дисципліни "Розробка ігрових додатків"**

Детальний опис етапів створення web-додатку наведено в таблиці А.2.

Таблиця А.2 – Етапи створення web-додатку

<b>№</b>	<b>Склад і зміст робіт</b>	<b>Строк розробки</b>
1	Розробка шаблону сайту	2 дні
2	Верстка сторінок web-додатку	14 днів
3	Реалізація головного функціоналу	10 днів
4	Пошук інформації для наповнення сайту	3 дні
5	Розробка бази даних	7 днів
6	Наповнення контентом сайту	12 днів
7	Тестування	10 днів
8	Розміщення на хостингу	1 день
9	Перевірка працездатності	2 дні
10	Реліз	1 день
	Загальна тривалість робіт	62 дні

**4 Вимоги до складу й змісту робіт  
із введення web-додатку в експлуатацію**

Web-додаток має бути затверджено та розміщено на web-хостингу.

## ДОДАТОК Б

### Планування робіт

Діджиталізація у нашому суспільстві набирає величезних обертів, тому більшість компаній, навчальних закладів, закладів харчування та інших створюють свої веб-додатки. Насамперед це дозволяє підтримувати рівень розвитку та значно збільшує попит. Особливо популярним стає галузь дистанційного навчання. З реаліями сьогодення дуже велика кількість країн обмежує великі скупчення людей, тому люди вимушені вчитися дистанційно. Саме з цієї причини дистанційне навчання набирає популярності. Тому розробка веб-додатку з вивчення дисципліни «Розробка ігрових додатків» є досить популярною та актуальною темою.

**Деталізація мети проекту методом SMART.** Для того, щоб проект був успішним та міг протистояти сучасним аналогам, треба на початковому етапі правильно визначити його мету за допомогою SMART-методу. У таблиці Б.1 наведені результати деталізації проекту методом SMART.

Таблиця Б.1 – Деталізація мети проекту методом SMART

Specific (конкретна)	Створити веб-додаток з вивчення дисципліни «Розробка ігрових додатків».
Measurable (вимірювана)	Розробити веб-додатку в короткий термін з використанням мінімальної кількості ресурсів.
Achievable (досяжна)	Для реалізації задачі необхідні знання HTML, CSS, PHP, JavaScript, MySQL
Relevant (реалістична)	Розроблений веб-додаток збільшить попит на вивчення дисципліни та дозволить зацікавити користувачів.

Time-framed (обмежена в часі)	Мета проекту має часове обмеження. Дати виконання проекту погоджені між замовником та виконавцем.
----------------------------------	---

**Планування змісту роботи.** WBS (Work Breakdown Structure – Ієрархічна структура роботи) – це графічний вигляд елементів проекту, які згруповані ієрархією у єдине ціле з продуктом проекту. Дана методика планування змісту робіт застосовується для досконалого виконання робіт частинами і представляє собою ключову частину проекту, за допомогою якої організовується командна робота. Елементами схеми можуть виступати послуги, дані та продукти. WBS забезпечує можливість для оцінки термінів та контролю за графіком робіт.

На першому рівні даної методології розташовується сам продукт. На другому рівні знаходяться основні дії спрямовані на досягнення мети проекту. Декомпозиція виконується до етапу, поки вони не стануть простими. Тобто не стануть діями, які мають однозначний чіткий результат, та призначені конкретній особі, для якої можна обчислити витрати таких ресурсів, як праця та тривалість. На рисунку Б.1 зображена схема WBS.

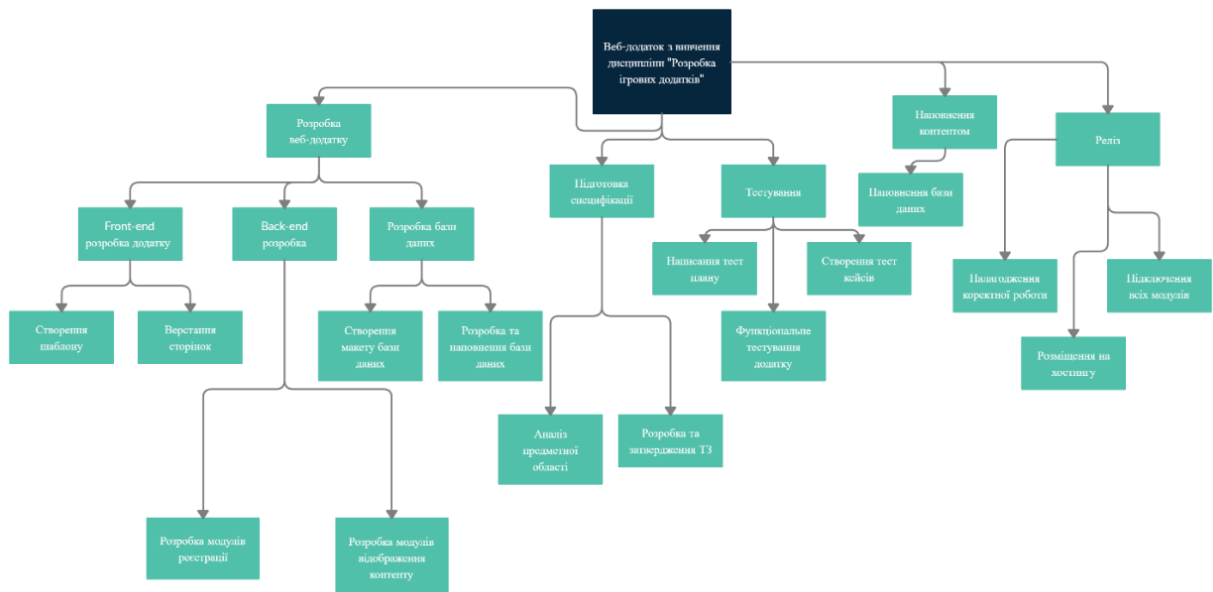


Рисунок Б.1 – WBS – структура

**Планування структури виконавців.** Після створення WBS структури проекту починається наступний етап – розробка структури виконавців або також звана OBS, яка представляє собою графічну структуру відображення учасників або відповідальних осіб, які беруть участь у створенні та реалізації проекту.

На рисунку Б.2 зображено OBS структуру планування проекту. Виконавці проекту зазначені у таблиці Б.2.

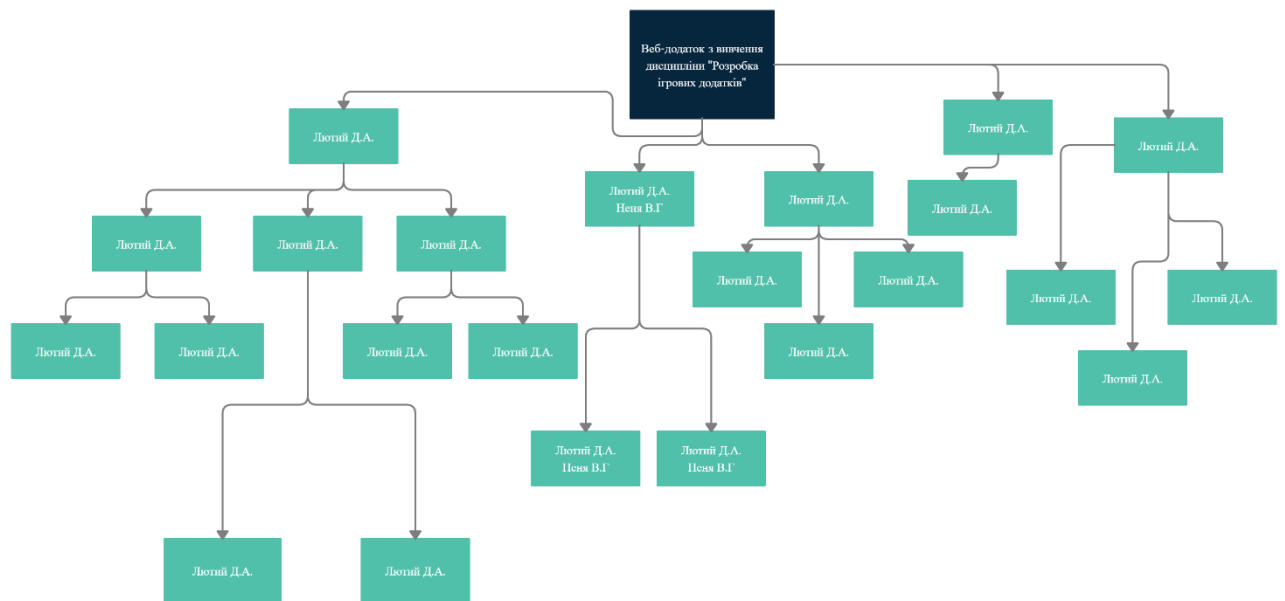


Рисунок Б.2 – OBS структура проекту

Таблиця Б.2 – Виконавці проекту

Роль	Ім'я	Проектна роль
Розробник	Лютий Д.А	Виконання front-end та back-end розробки проекту
Проектувальник	Лютий Д.А	Виконує проектування та розробку бази даних
Тестувальник	Лютий Д.А	Виконує розробку тест плану, тест кейсів та функціональне тестування



Керівник проекту	Неня В.Г.	Формує завдання на розробку проекту, затверджує ТЗ
Менеджер проекту	Лютий Д.А.	Відповідальний за своєчасне виконання проекту та розподіл ресурсів проекту.

**Діаграма Ганта.** Одним з важливих етапів планування проекту є побудова діаграми Ганта, що представляє собою розклад виконання робіт з розподілом часу. Дякуючи діаграмі можна отримати реальні дати закінчення проекту та більш чітко розподілити витрати ресурсів. Діаграма Ганта проекту зображена на рисунках Б.3.



Рисунок Б.3 – Календарний графік проекту

**Управління ризиками проекту.** Під час оцінки ризиків проекту, потрібно визначити ризики, які мають бути усунені якнайшвидше для економії часових та грошових ресурсів. Враховуючи ступень важливості ризику, кожен ризик має свій пріоритет та буде виправлений в залежності від пріоритету. У таблиці Б.3 зображено шкалу для класифікації ризиків за впливом на проект.

Таблиця Б.3 – Оцінювання ризиків за ймовірністю виникнення.

<b>Оцінка</b>	<b>Ймовірність</b>	<b>Вплив</b>	<b>Тип</b>
1	Низька	Низький	Прийнятні
2	Середня	Середній	Виправдані
3	Висока	Високий	Недопустимі

Для зниження негативного впливу ризиків на проект потрібно планувати реагування на них. План включає в себе визначення ефективності розробки та оцінювання наслідків ризиків на проект. Далі була створена матриця ймовірності виникнення ризиків та вплив їх на проект, за допомогою планування.

Відповідно до отриманого значення індексу, створено класифікація ризиків за рівнем, зображену в таблиці Б.4. Описання ризиків та стратегії реагування на них наведені в таблиці Б.5.

Таблиця Б.4 – Шкала оцінювання.

<b>№</b>	<b>Назва</b>	<b>Межі</b>	<b>Ризики</b>
1	Прийнятні	$1 \leq R \leq 2$	1,9
2	Виправдані	$3 \leq R \leq 4$	2,4,6,8,10
3	Недопустимі	$6 \leq R \leq 9$	3,5,7

Таблиця Б.5 – Описання ризиків

№	Опис ризику	Ймовірність виникнення	Назва	Ранг	План А	Тип реагування	План Б
RS_1	Конфлікт між розробником та замовником	Низька	Середній	3	На початку створення проекту налагодити відносини між замовником та розробником та визначитися умовами	Попередження	При виявленні ризику спробувати дізнатися причину виникнення та домовитися про умови вирішення
RS_2	Нерозуміння документації	Низька	Високий	4	Зв'язатися з замовником та вирішити непорозуміння	Прийняття	Спробувати запропонувати свою стратегію розробки
RS_3	Не розрахування часових ресурсів	Висока	Високий	9	Зв'язок з розробником для збільшення часу на розробку. Приведення чітких аргументів	Пом'якшення	Аналіз часу на розробку проекту, та змінення порядку розробки модулів, можливе зменшення часу на виконання
RS_4	Постійна зміна ТЗ	Середній	Середній	4	Чітко описати всі вимоги проекту та встановити умови реалізації	Перенесення	Узгодити умови внесення змін на проєкті та встановити всі положення
RS_5	Помилки в розробці	Висока	Високий	9	Продумувати розробку на етапі проєктування та створення документації	Пом'якшення	Здійснювати проміжне тестування на кожному етапі.
RS_6	Не ефективна технологія розробки	Низька	Середній	3	Провести аналіз методів розробки та обрати найпростішу	Пом'якшення	Проаналізувати основні методи розробки та обрати підходящий.

RS_7	Помилка при створенні документації	Висока	Високий	9	Починати тестування на етапі створення документації	Пом'якшення	
RS_8	Збої в роботі програми	Середня	Середній	4	Регулярне створення резервних копій.	Попередження	Зміна сервера та забезпечення
RS_9	Реалізація зайвого функціоналу	Низька	Низький	1	Слідування раніше створеній та узгодженій документації	Пом'якшення	
RS_10	Недостатня кількість тестового покриття	Середня	Середній	4	Залучення спеціалістів	Попередження	Створення тест плану та тест кейсів на початку розробки