

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
**НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

Кафедра інженерії програмного забезпечення

**Пояснювальна записка**

до бакалаврської роботи  
на ступінь вищої освіти бакалавр

на тему: **«РОЗРОБКА ВЕБ-СЕРВІСУ ДЛЯ ПІДТРИМКИ  
ОРГАНІЗАЦІЙНИХ ПРОЦЕСІВ ДИСТАНЦІЙНОГО НАВЧАННЯ З  
ВИКОРИСТАННЯМ БІБЛІОТЕКИ REACT.JS»**

Виконав: студент 4 курсу, групи ПД-43  
спеціальності

121 Інженерія програмного забезпечення  
(шифр і назва спеціальності/спеціалізації)

\_\_\_\_\_ Виговський Б.С.

(прізвище та ініціали)

Керівник \_\_\_\_\_ Залива В.В.

(прізвище та ініціали)

Рецензент \_\_\_\_\_

(прізвище та ініціали)

Київ –2023

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ**  
**ТЕХНОЛОГІЙ**

Кафедра Інженерії програмного забезпечення  
Ступінь вищої освіти -«Бакалавр»  
Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

**ЗАТВЕРДЖУЮ**  
Завідувач кафедри  
Інженерії програмного  
забезпечення  
Негоденко О.В.  
“ \_\_\_\_\_ ” \_\_\_\_\_ 2023 року

**З А В Д А Н Н Я**  
**НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТА**

**ВИГОВСЬКОМУ БОГДАНУ СЕРГІЙОВИЧУ**

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка веб-сервісу для підтримки організаційних процесів дистанційного навчання з використанням бібліотеки React.JS»

Керівник роботи: Залива В.В. асист.  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від «24» лютого 2023 року  
№26.

2. Строк подання студентом роботи «1» червня 2023 року

3. Вхідні дані до роботи

Методи обробки зображень;

Науково-технічна література з питань, пов'язаних з програмним забезпеченням щодо розпізнавання тексту з зображень;

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

4.1 Системи розпізнавання та вилучення текстової інформації з зображень.

4.2 Вимоги та оцінка якості системи.

4.3 Опис проектування системи.

#### 4.4 Опис використаних технологій.

#### 5. Перелік демонстраційного матеріалу (назва основних слайдів)

1. Актуальність проблеми
2. Існуюче програмне забезпечення та методи розпізнавання
3. Принцип роботи інформаційної системи
4. Розпізнавання та групування даних конкретних документів
5. Архітектура бази даних
6. Логічна діаграма компонентів архітектури програмного забезпечення

6. Дата видачі завдання «25» лютого 2023

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	25.02.23-27.02.23	Виконано
2	Вимоги до системи	05.03.23-08.03.23	Виконано
3	Створення та навчання моделі для вилучення полів	10.03.23-30.03.23	Виконано
4	Створення та навчання моделі для вилучення таблиць	02.04.23-13.04.23	Виконано
5	Концепція та архітектура програмного забезпечення	14.04.23-04.05.23	Виконано
6	Вступ, висновки, реферат	05.05.23-14.05.23	Виконано
7	Розробка обов'язкових демонстраційних матеріалів	16.05.23-21.05.23	Виконано
8	Попередній захист роботи	24.05.2023	Виконано
9	Здача роботи	01.06.2023	Виконано

Студент \_\_\_\_\_

( підпис )

(прізвище та ініціали)

Керівник роботи \_\_\_\_\_

( підпис )

(прізвище та ініціали)





## РЕФЕРАТ

Текстова частина бакалаврської роботи 70 с., 29 рис., 64 джерела.

ІНФОРМАЦІЙНА СИСТЕМА, ВЕБ-СЕРВІС, НАВЧАЛЬНИЙ ПОРТАЛ, САЙТ-ПОРТАЛ, REACT.JS, ОСВІТА, НАВЧАННЯ, ОСВІТЯНИ, ЗАВДАННЯ.

Метою бакалаврської дипломної роботи – це створення веб-ресурсу для організації підтримки організаційних процесів дистанційного навчання з використанням бібліотеки React.js.

Об'єктом дослідження є програмна реалізація систем доступу до даних та інформаційних систем.

Предметом дослідження є проєкт інформаційної системи навчального ресурсу.

Методи дослідження в роботі використані такі: пошуковий по наявній методичній та науковій літературі із аналізом знайденого матеріалу, індукція та дедукція, порівняння, програмне проєктування, з'ясування причинно-наслідкових зв'язків.

В ході розробки інформаційної системи були розроблені наступні елементи проєктованого сервісу: код інформаційної системи; база даних; функціонал інформаційної системи; програмне забезпечення для управління сервісом у форматі фронтенд- та бекенд-проєкту. Система може змінювати, доповнювати, вести пошук і перегляд інформації про користувачів, накладати обмеження доступу до системи, зберігати списки користувачів у вигляді архіву.

*Галузь використання – дистанційне навчання студентів.*

## ЗМІСТ

<b>ВСТУП</b> .....	<b>9</b>
<b>РОЗДІЛ 1 ОГЛЯД ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ПІДТРИМКИ ОРГАНІЗАЦІЙНИХ ПРОЦЕСІВ</b> .....	<b>11</b>
1.1. Аналіз розробки веб-сервісу та його можливостей для підтримки організаційних процесів дистанційного навчання.....	12
1.2. Вибір сучасних інструментів та аналіз методологій для розробки веб-сервісів та бази даних для них.....	17
1.3 Висновки до розділу 1.....	21
<b>РОЗДІЛ 2 ВИМОГИ ДО СИСТЕМИ</b> .....	<b>22</b>
2.1. Аналіз цільової аудиторії програмного продукту.....	22
2.2. Вибір методів моделювання та проєктування процесу розробки веб-сервісу.....	24
2.3 Висновки до розділу 2.....	26
<b>РОЗДІЛ 3. РОЗРОБКА ТА ДИЗАЙН ВЕБ-СЕРВІСУ</b> .....	<b>27</b>
3.1. Сценарії взаємодії.....	27
3.2. Архітектура інформаційної системи.....	28
3.3. Розробка програмних модулів.....	30
3.4. Розробка дизайну.....	37
3.5. Розгортання програмного забезпечення проєктованої системи.....	40
3.6. Тестування інформаційної системи.....	42
3.7 Висновки до розділу 3.....	47
<b>ВИСНОВКИ</b> .....	<b>48</b>
<b>СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ</b> .....	<b>50</b>
<b>ДОДАТОК А</b> .....	<b>56</b>
<b>ДОДАТОК Б</b> .....	<b>61</b>

## **ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ**

ЗМІ – Засоби масової інформації.

SMART – Specific, Measurable, Achievable, Relevant, Time bound.

ІС – Інформаційна система.

HTML – HyperText Markup Language.

CSS – Cascading Style Sheets.

SPA – Single Page Application.

DOM – Document Object Model.

Scss – скриптова метамова, яка інтерпретується в каскадні таблиці стилів (CSS).

CMS – Content Management System.

UML – Unified Modeling Language.



## ВСТУП

В нинішні часи будь-яка галузь, яка хоче бути успішною в своїй справі має слідувати тенденціям сьогодення, а саме зарекомендувати та висвітлити себе в соціальних мережах та мати свою сторінку в світовій мережі.

Підростаюче покоління, яке переживає всі етапи діджиталізації, є найактивнішим користувачем всіх ресурсів на просторах інтернету від спілкування з товаришами до навчання.

Електронне навчання з іноземними однолітками сприяє розширенню світогляду, нехтуванню стереотипами, неформальному мисленню, дозволяє зменшити мовний бар'єр, тому ці якості є найціннішими для тінейджерів.

Для приїжджих громадян найважливішим на просторах Інтернету є доступність до всіх електронних мап, спілкування, навчання, а також як джерело ЗМІ та ін.

**Актуальність теми дослідження.** Дистанційна освіта є важливим аспектом для розвитку та зростання громадського суспільства. На даний момент для українців які потребують підтримки в надзвичайній ситуації в Україні та за її межами є життєво необхідна допомога у навчанні. Дистанційне навчання дозволяє людям робити особистий внесок у вирішення суспільних проблем, брати участь у соціально важливих ситуаціях.

**Метою бакалаврської дипломної роботи** – є створення веб-ресурсу для організації підтримки організаційних процесів дистанційного навчання з використанням бібліотеки React.js.

Задля досягнення поставленої мети необхідно виконати наступні задачі:

1. Визначити актуальність створення веб-сервісу для підтримки дистанційної освіти.
2. Виконати вибір сучасних інструментів та аналіз методологій для розробки веб-сервісів та бази даних для них.
3. Провести дослідження та аналіз цільової аудиторії продукту.
4. Проаналізувати існуючі аналоги на ринку.

5. Виконати моделювання та проектування процесу розробки веб-сервісу.
6. Розробити базу даних ресурсу.
7. Провести реалізацію веб-сервісу та його дизайну.

**Об'єктом дослідження** є програмна реалізація систем доступу до даних та інформаційних систем.

**Предметом дослідження** є проєкт інформаційної системи навчального ресурсу.

**Методи дослідження.** Пошук методичної та наукової місцевої та іноземної літератури з детальним пропрацюванням знайденого матеріалу, індукція та дедукція, програмне проектування.

**Джерельна база дослідження.** Робота побудована на аналізі методично-наукової літератури, наукових публікацій, патентний аналіз, методичних посібників та робіт кваліфікованих ІТ-спеціалістів у цьому напрямку.

**Наукова новизна** роботи полягає у створенні ІС порталу, для підтримки організаційних процесів дистанційного навчання зі зручним інтерфейсом, виконаним на базі React.JS. В дипломному проєкті вперше був застосований комбінований метод проектування до інформаційних систем доступу.

**Теоретична та практична цінність роботи** полягає у використанні сучасної іноземної літератури, яка має високий рейтинг серед наукової спільноти по даній темі. Розроблена ІС система була створена для того, щоб полегшити користувачу даного порталу для підтримки організаційних процесів дистанційної освіти знайти необхідну задачу, вирішити її та отримати зворотній зв'язок.

**Структура роботи.** Дипломна робота містить 70 листів друкованого тексту, 29 рисунки, 3 таблиці та 7 додатків та налічує 64 джерела використаної літератури.

# 1 ОГЛЯД ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ ПІДТРИМКИ ОРГАНІЗАЦІЙНИХ ПРОЦЕСІВ

## 1.1 Аналіз розробки веб-сервісу та його можливостей для підтримки організаційних процесів дистанційного навчання

Для створення успішного веб-сервісу ми повинні орієнтуватись на потреби користувачів а саме:

1. Створити зрозумілий дизайн.
2. Виключити шрифти за типом газети, це не актуально.
3. Написати мінімум тексту для того щоб не відволікати від основного контенту.
4. Інформація яка написана в тексті повинна бути зрозуміла.
5. Зробити можливість зворотного зв'язка.

Для ефективної роботи в 2023 році необхідно створити зручну та інформативну ІС. Була поставлена задача автоматизувати запис на певне завдання, та гнучкий перегляд інформації про портал. Саме ці фактори посприяли вибору актуальності роботи.

Полегшення процесу взяття конкретної задачі, вирішено завдяки створенню власної ІС, так як вона сприяє використанню різних модифікацій та є більш орієнтована на клієнта.

Технологія ІС побудована на тому, що вся програмна логіка розміщена на сервері, а інтерфейс користувача є в загальному доступі для будь-якої людини, яка має доступ до мережі.

Отже, завдання розробити такий веб-сервіс стає все більш популярним та затребуваним, адже його можна підлаштувати під бізнес чи будь-яку діяльність.

Розглянемо можливості досягнення мети методом SMART. SMART є аббревіатурою, розшифровка якої: Specific, Measurable, Achievable, Relevant, Time bound. Кожна літера аббревіатури SMART має своє значення, а саме описує критерій ефективності поставлених цілей.

- Specific – конкретна;
- Measurable – вимірювана;
- Achievable – досяжна;
- Realistic – реалістична;
- Timed – обмежена за часом.
- Specific – Веб-сервіс дистанційного навчального процесу.
- Measurable – Ефективність роботи даного веб-сервісу визначається оцінкою користувачів сайту.
- Achievable – Сайт буде виконано використовуючи HTML, CSS, PHP, JavaScript.
- Realistic – Були використані всі засоби технічного та програмного напрямку.
- Timed – Мінімальний час на опрацювання даних та вчасно виконана робота по розробці веб-сервісу.

Провівши аналіз методом SMART можна зазначити, що розроблення певної ІС навчального порталу відповідає всім вимогам. Також проєкт буде виконано і здано вчасно, а саме до 20 червня 2022 року, що відповідає календарному плану.

Першим етапом створення ІТ-проєкту є планування змісту структури (WBS). WBS ґрунтується на ієрархії завдань задля легкого розуміння, орієнтуванні в проєкті, а також вартості, складу, таймінгу. Процес часто описується як декомпозиція робіт, яка охоплює всі деталізовані складові даної розробки в певному порядку для вдалої організації роботи. WBS може бути представлена в вигляді списків, таблиць, діаграм Ганта.

До головних задач WBS відносять:

- проведення деталізації проєктних робіт та визначення їх ступеню;
- оцінити кількість рівнів;
- визначення структури для кожного рівня;
- опис кожного елемента WBS;
- формування коду для всіх блоків;
- розрахунок витрат для всіх блоків.

Виконання кожного етапу є ключовим у організації робіт по проєкту, адже саме це допоможе з'єднати все в єдину складову без додаткових витрат коштів та часу.

Створення WBS є інтенсивним і довготривалим процесом до якого залучають спеціалістів різних галузей.

Переваги WBS:

- допомагає раціонально оцінити кошти, ризики та час виконання робіт;
- допомагає правильно організувати проєкти;
- допомагає розподілити обов'язки;
- допомагає змістовно описати проєкт;
- видно всі етапи роботи і можна проконтролювати роботу всіх блоків.

Проведемо планування організаційної структури проєкту (OBS).

Відноситься тільки до внутрішньої частини проєкту і не перетинається з головними організаціями чи відносно окремими групами.

OBS процес використовується для найнижчих рівнів – базових робочих груп, а в малих проєктах – до кожного виконавця.

Структура OBS будується наступним чином:

- на першому рівні представляються керівники проєкту;
- на другому – представники всіх етапів проєкту відповідно до розподілених обов'язків.

Поєднання організаційної та робочої структури допомагає оптимально планувати, використовувати та контролювати всі етапи роботи та дає можливість для порівняння з попередніми підрозділами. Кожен відповідальний представник повинен надавати план виконаних робіт, а також повну звітність в межах своєї сфери. За допомогою WBS відбувається розподіл робочого проєкту на основні окремі групи.

В свою чергу OBS використовується для відтворення всіх робіт, які виконуються на найнижчому рівні в WBS. Тобто всі роботи, які знаходяться на найнижчих рівнях в WBS відноситься як до WBS, так і OBS, тобто це основні

частини цих структур. При будіванні схеми OBS керуються наступними заголовками – структура проекту – організаційна структура розробників.

Потреба розробки організаційних структур полягає в тому, що дня виконання будь-якого проекту в ІТ-сфері зазвичай створюється нова команда з фахівців різного спрямування як з боку розробників, так і замовника. Тому для нового персоналу треба чітко прописати проєктні ролі, обов'язки, відповідальність, посади, а також ієрархічну схему підпорядкованості.

Проаналізуємо управління ризиками виконання ІТ-проєкту. Проводячи кількісну оцінку зазвичай вона підтягує і якісну, а також вимагає повну ідентифікацію ризиків. Оцінки якісних і кількісних ризиків оцінюються окремо або разом, в залежності від виділених коштів та вільного часу.

План реакції на ризики – це низка методів і технологій, які допоможуть знизити вплив ризиків на проєкт. Результативність розробки реакції на ризики прямо впливає на ефективність розробки веб-сервісу, адже вони можуть бути позитивними або негативними.

Для оцінки ризиків необхідно знати можливості учасників, мати переліки джерел ризиків і потенційних подій ризиків, вартісні оцінки. Підсумком етапу є поділ ризиків на дві групи:

- події, що вимагають реагування (цим подіям необхідно приділяти особливу увагу);
- події, які не потребують реагування (на ці події поки можна не звертати особливої уваги).

Для аналізу проєктних ризиків використовуються кількісні та якісні методи. Особа, що приймає рішення, може аналізувати невизначеність, пов'язану з кожним з елементів системи. Воно може розглядати ступінь залежності між ризиком і складовими даного об'єкта, оцінювати сукупний вплив ризиків підсистем на процес. Тому велику роль на цьому етапі відіграє досвід і компетентність менеджера.

Таблиця 1.2 – Можливість виникнення і ступінь ризику

Ризик	Ймовірність виникнення	Величина втрат
Поломка ПК	4	0
Не підтримка потрібного софту	1	0
Неправильний розподіл часу	5	2
Непередбачувані витрати	2	1
Некоректне складення ТЗ	1	1
Зміна в ТЗ	4	2
Непередбачувані ситуації	2	3

Ризики проходять оцінку за наступними критеріями:

Ймовірність виникнення:

- 1 Слабо ймовірно
- 2 Малоймовірно
- 3 Ймовірно
- 4 Вельми ймовірно
- 5 Майже можливо

Ступінь втрат:

- 1 Мінімальна
- 2 Низька
- 3 Середня
- 4 Висока
- 5 Максимальна

На рисунку 1.1 представлена матриця ймовірності втрат

Ймовірність виникнення	5				4	
	4			1, 6, 7		2
	3				5	
	2			3		
	1			8		
		1	2	3	4	5

Рисунок 1.1 – Матриця «Ймовірність-Втрати»

Класифікація за ступенем впливу:

- вагомі ( $12 \leq R \leq 19$ );
- ігноровані ( $1 \leq R \leq 4$ );
- критичні ( $20 \leq R \leq 25$ );
- незначні ( $5 \leq R \leq 8$ );
- помірні ( $9 \leq R \leq 11$ ).

Класифікація за рівнем ризику:

- виправданні ризики;
- недопустимі ризики;

- прийнятні ризики.

## **1.2 Вибір сучасних інструментів та аналіз методологій для розробки веб-сервісів та бази даних для них**

1. Мова розмітки гіпертексту HTML. HTML – це мова розмітки, яка визначає структуру вашого вмісту. HTML складається з ряду елементів, які ви використовуєте, щоб укласти або обернути різні частини вмісту, щоб він виглядав певним чином або діяв певним чином. Теги, що вміщують, можуть створювати гіперпосилання слова або зображення на інше місце, можуть виділяти слова курсивом, можуть збільшити або зменшити шрифт тощо. За допомогою HTML, усі ваші об'єкти, такі як інтерактивні форми, можуть бути вбудовані в візуалізовану сторінку. HTML надає засоби для створення структурованих документів, позначаючи їх структурну та семантику тексту, такі як фотографії, списки, таблиці, абзаци посилання та багато інших елементів. У браузерах не показують теги HTML, але вони їх використовують для інтерпретації вмісту сторінки.

2. SCSS - це мова препроцесора, який компілюється або передається в CSS. Він має розширення файлу SCSS. Ми можемо додати кілька додаткових функцій у CSS за допомогою SCSS, включаючи змінні, вкладення та багато іншого. Усі ці додаткові функції можуть зробити напис SCSS набагато довше та швидше, ніж напис стандартного CSS.

Я вирішив використати SCSS, а не стандартний CSS так як у нього є кілька переваг над CSS таких як:

Швидкість написання, писати код і прописувати стилі на SCSS суттєво швидше ніж на CSS.

- Він дозволяє писати чистий CSS-код у стилі мов програмування.
- Scss сумісний з усіма версіями CSS також він може використовувати будь-які доступні CSS-бібліотеки.



- У ньому можна використовувати синтаксис вкладеності та такі корисні функції, як маніпуляція з кольорами, математичні та інші значення.

3. React.js – фреймворк для JavaScript. Бібліотека допомагає створювати веб-програми – сайти, які завантажуються лише один раз. Насамперед він націлений розробку SPA-рішень (Single Page Application), тобто односторінкових додатків. У цьому плані React.js є спадкоємцем іншого фреймворку React.js. У той же час React.js це не нова версія React.js, а новий фреймворк. (SPA) – Single Page Application – підхід, який загрузає весь сайт одразу і динамічно взаємодіє з користувачем, оновлює на сторінці необхідні компоненти, при маніпуляції з сайтом оновлюються тільки необхідні компоненти, що дозволяє не оновлювати сторінку. Переваги технології React.js:

- Велике ком'юніті. React.js має величезне ком'юніті, по ньому проводять велику кількість конференцій, він завжди на слуху і це означає що він завжди буде залишатись у тренді.
- Декларативний стиль коду. При створенні шаблонів React.js застосовується декларативна парадигма програмування. Завдяки цьому код стає легшим, та покращується його читання і підтримку, так як описуються необхідні кінцеві результати а не всі кроки по його досягненню.
- Використання директив. Як мова шаблонів у React.js використовується HTML. Він розширюється за допомогою директив, які додають до коду відомості про потрібну поведінку (наприклад, про необхідність завантажити певний модуль відразу після завантаження сторінки). Директиви дозволяють більше зосередитись на опрацюванні логіки і працювати продуктивніше. Також їх можна використовувати повторно, що в свою чергу підвищує кращу читабельність коду.
- Висока швидкість розробки.
- Модульність. React.js дозволяє організовувати програми з окремих модулів. Такі модулі можуть залежати один від одного, так і бути автономними. Наприклад, в останньому випадку модуль входу через Facebook можна використовувати відразу в декількох частинах програми, скажімо, на

сторінці входу та на сторінці оформлення замовлення. А завдяки вбудованому механізму застосування залежностей React.js сам розпізнає ситуацію, коли потрібно надати допоміжні об'єкти, надає їх і пов'язує об'єкти між собою.

- Наявність готових рішень. Що важливо, для React.js. існує безліч готових рішень, які дозволяють вирішувати досить різноманітні завдання, використовуючи вже готові модулі. Наприклад, існує кілька модулів для роутингу найпопулярніший з яких ui-router, так само є різні модулі для роботи з таблицями ui-grid, ng-table та багато інших.
- Двостороннє зв'язування даних. В React.js. застосовується двостороннє зв'язування: будь-які зміни в інтерфейсі користувача відразу ж відбиваються на об'єктах програми і навпаки. Фреймворк сам стежить за подіями браузера, змінами моделі та діями користувача на сторінці, щоб одразу оновлювати потрібні шаблони. При цьому у коді JavaScript не потрібно зберігати посилання на DOM-елементи та явно ними маніпулювати. Ми просто описуємо необхідний результат у термінах стану моделі, і нам не треба використовувати низькорівневі конструкції.

4. MySQL – це вільна реляційна система яка керує базами даних. Вона є найбільш широко використовуваною базою даних з відкритим вихідним кодом. За допомогою MySQL можна розробити високопродуктивні, масштабовані програми баз даних, незалежно від того, чи ми тільки початківці з невеличкою базою даних, або ж досягнем популярності і в нас вона буде глобальною. MySQL гарантує досягти найвищого рівня масштабованості, безпеки, стабільності та безперебійної роботи. Усім користувачам доступна найповніша колекція розширених функцій, інструментів адміністрування та підтримки експертів якщо виникають якісь питання.

Було прийнято рішення для порівняння двох систем управління базами даних, таких як MySQL і PostgreSQL (табл. 1.3).

Таблиця 1.3 – Порівняльна таблиця MySQL і PostgreSQL

№	Критерії для оцінки	MySQL	PostgreSQL
1	Відкрите джерело	Відкритий код, який належить компанії Oracle	Відкрите джерело
2	Підтримка NoSQL / JSON	Підтримка тільки деяких функцій	Тільки JSON
3	Відповідність вимогам ACID	Частково відповідає	Повністю відповідає
4	Декларативне секціонування	Підтримується	Підтримується
5	Віконні функції	Підтримується	Підтримується
6	Можливість зберігати інформацію тільки в пам'яті	Так	Ні

Важливо використовувати популярні технологічні рішення, адже на знаходження потрібної інформації і вирішення проблем які з'являються під час використання певного ресурсу, відбувається значно швидше ніж на мало відомих системах.

Було проведено опитування більше 65 000 професійних розробників у 2020 році в якому було встановлено, які інструменти найбільш частіше використовуються (Рис. 1.2).

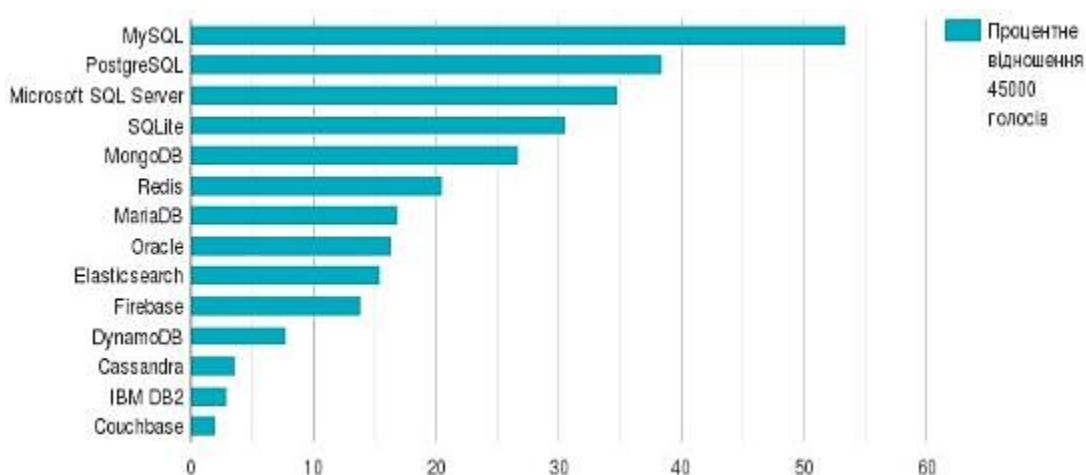


Рисунок 1.2 – Використовування систем для баз даних професійними розробниками

Отже, для розробки проєктованого ресурсу підтримки організаційних процесів дистанційного навчання нами обрано MySQL. Переваги MySQL:

- Безкоштовно та з відкритим кодом.
- Легке встановлення.
- Зрозумілий інтерфейс.
- Періодичні оновлення.
- Різноманітні можливості які включають в себе великий набір інструментів для керування базою даних.
- Безпека яка встановлена вбудованою захисною системою.
- Популярне рішення компаній гігантів таких як: Google, GitHub, Nasa.

### **1.3 Висновки до розділу 1**

Отже, на сьогодні дистанційне навчання є надзвичайно важливим для всієї України. Завдяки розробці веб-сервісу для освітян ми вирішимо такі проблеми як: відстеження історій замовлення, збереження необхідних контактів, групування заявок від освітян, верифікація студентів які використовують основні функції.

З цією метою нами проведено вибір сучасних інструментів та аналіз методологій для розробки веб-сервісів та бази даних для них. Обрано:

- мову розмітки гіпертексту HTML;
- Scss – скриптову метамову, яка інтерпретується в каскадні таблиці стилів (CSS);
- React.js – фреймворк для JavaScript – бібліотека допомагає створювати веб-програми – сайти, які завантажуються лише один раз;
- для розробки проєктованого ресурсу підтримки організаційних процесів дистанційного навчання нами обрано MySQL – це вільна реляційна система яка керує базами даних. Вона є найбільш широко використовуваною базою даних з відкритим вихідним кодом.

## **2 ВИМОГИ ДО СИСТЕМИ**

### **2.1 Аналіз цільової аудиторії програмного продукту**

Насамперед нам потрібно визначитись з головною метою дослідження. Та зрозуміти яку інформацію та послуги ми повинні надати користувачу на проєктованому ресурсі підтримки організаційних процесів дистанційного навчання.

Ідея проєкту – це створення ІС навчального ресурсу. Щоб створений веб-ресурс був максимально актуальний, навчальний портал повинний бути швидкий, зручний. Користувач на створеному порталі повинен вільно та швидко зареєструватися, отримати завдання, мати змогу його виконати та отримати фідбек. А також в будь-який час мати можливість перевірити всі виконані задачі, а також дізнатись про викладення нових. Головні вимоги до створеного веб-сервісу:

1. Адаптивність під засоби з різним розширенням.
2. Вільний перегляд інформації з завдань на головній сторінці.
3. Інформація про ресурс підтримки організаційних процесів дистанційного навчання.
4. Онлайн створення послуг користувачами.
5. Стрічка завдань.

Продуктом проєкту є сайт навчального порталу, який допоможе роботі освітян з усіх регіонів України. Створення даного проєкту потрібне для удосконалення та полегшення дистанційного навчання, а також створення можливості взяти онлайн певні завдання. До цього продукту мають доступ усі користувачі, які підключені до мережі Інтернет.

Створений веб-ресурс включає в себе тільки навчальні елементи. Створена ІС буде мати максимально зручний інтерфейс для роботи.

Веб-ресурс має містити в собі чітко прописані розділи з певними окремими функціями:

1. Інформація про портал.
2. Контактна інформація порталу.
3. Онлайн-запис на певне завдання.
4. Розгорнутий список видів послуг.

Тому, цільовою аудиторією є група користувачів, яка найбільше потребує даної послуги. Цільова аудиторія складається з існуючих та потенційних клієнтів. Аналіз цільової аудиторії – дослідження, у ході якого виникає розуміння, необхідне успішної оптимізації маркетингових процесів і отримання максимальної вигоди від рекламної кампанії та просування загалом. Аналіз цільової аудиторії є невід’ємною частиною будь-якої маркетингової стратегії.

Зробивши аналіз цільової аудиторії, ми зможемо:

- Знайти та сфокусуватися тільки на тій аудиторії, якою справді цікавий проєктований продукт.
- Створювати спеціалізовані рекламні матеріали, орієнтовані специфічну частину аудиторії.
- Розвивати тривалі взаємини та утримувати клієнтів, вирішуючи їх актуальні проблеми через створення корисної їм інформації.
- Отримувати інформацію для більш ефективної стратегії навчання.
- Бути більш конкурентним з допомогою таргетування специфічних ніш.

Цільовою аудиторією проєктованої інформаційної системи підтримки організаційних процесів дистанційного навчання є чоловіки та жінки різного віку, різного роду зайнятості та з різних регіонів. Тобто цільова аудиторія досить широка. На основі цього обов’язковим правилом під час проєктування інформаційної системи є реалізація механізму реєстрації користувачів до парольного доступу до ресурсу.

## 2.2 Вибір методів моделювання та проєктування процесу розробки веб-сервісу

Для створення зручного та зрозумілого інтерфейсу веб-ресурсу були виокремлені такі завдання:

- провести аналіз вже створених аналогів, визначити їх недоліки та підібрати методи проєктування нового веб-сервісу;
- створити план та розробити технічне завдання;
- провести тестування розробленої системи.

Щоб зробити процес роботи користувача на порталі більш простим, та візуально інформативним, він має можливість на сайті проглянути всю потрібну інформація. Після виконання певного запиту, на email користувача прийде інформація про його результат.

Виходячи з проведеного аналізу для створення веб-ресурсу були обрані наступні методи:

- методи аналізу для дослідження предметної області;
- вибір інструментів реалізації завдань;
- методи проєктного моделювання ІС;
- метод синтезу.

Ці методи умовно поділяються на дві групи:

- методи ручного програмування;
- методи автоматизованого програмування;

Один із видів створення сайтів - це ручне написання коду на одній або декількох мовах програмування. Водночас процес може відбуватись як у звичайних(текстових) так і у візуальних програмах для редагування HTML та CSS. Візуальні допомагають створити веб-ресурс в режимі WYSIWYG.

Програма для редагування типу WYSIWYG - це середовище в якому розробник створює сайт, типу Landing page або Web application. Поєднання текстового та графічного редакторів( Adobe, Photopaint, Figma та ін.) дає змогу побачити та протестувати результат одразу під час написання коду. Основна

перевага цього типу розробки це швидкість, можливість об'єднати процес написання коду та створення макету, так як цей метод не передбачає поглибленого вивчення мов програмування( Java, .Net та ін). Основа частина коду пишеться автоматично середовищем розробки. Головним недоліком цих редакторів це низький захист від вірусних атак та зовнішнє втручання злоумисників, так як способи залому програмної частини такого типу сайтів вже давно відомі.

Для створення статичного сайту можливе ручне написання за допомогою HTML та CSS, з використанням JavaScript. Для цього типу розробки дизайн ( графічне оформлення) має створюватись окрема у вигляді макету. Дизайнер може використовувати графічні редактори на свій розсуд. Він також може редагувати вже готові шаблони купивши їх або завантаживши безкоштовні.

Статичний сайт складається в основному з html розмітки, яка складається в єдине ціле. Статичний веб-ресурс- це сукупність HTML файлів, що були розміщені на серверній частині сайту.

Для створення динамічного веб-ресурсу потрібно використовувати серверні мови програмування типу PHP, .NET, NodeJs і т.д. Динамічний тип - це веб-ресурс, який має у своєму складі динамічні сторінки- контент, скрипти, шаблони та ін., у вигляді окремих елементів. Зазвичай для демонстрації будь-якої кількості однакових за типом сторінок, використовують шаблони, в які завантажують необхідний зміст, такий підхід спрощує редагування зовнішнього вигляду( тобто всі сторінки) редагуючи лише шаблон.

Інший тип розробки сайтів це автоматизований метод створення веб-ресурсу, використовуючи спеціалізовані конструктори чи системи керування контентом (CMS). Метод створення сайту за допомогою CMS має найбільшу популярність. Адже цей редактор вже має готову візуальну та програмну частину, яку розробник може заповнити необхідним контентом, або підлаштовувати під свої потреби. Широкий спектр можливостей розробки веб-ресурсів будь якої складності основна перевага цього методу. І тому цей метод один із найбільш зручних та практичних. Різноманіття налаштувань та можливість редагування



окремих компонентів CMS, легкість управління контентом зробило створення сайту за допомогою CMS дійсно ефективною.

## **2.2 Висновки до розділу 2**

Предметна область проектування – інформаційна система по організації підтримки організаційних процесів дистанційного навчання. Система повинна бути створена для того, щоб допомогти користувачам всесвітнього Інтернет-порталу підтримки організаційних процесів дистанційного навчання знайти потрібне завдання, виконати його, переглянути історію завдань і т.д.. Відповідно, під час перегляду завдання користувач повинен бачити короткий опис завдань з посиланням на повний опис на сторінці з відомостями.

В ході розробки інформаційної системи повинні бути розроблені наступні елементи проєктованого сервісу: код інформаційної системи; база даних; функціонал інформаційної системи; програмне забезпечення для управління сервісом у форматі фронтенд- та бекенд-проєкту.

Система повинна змінювати, доповнювати, вести пошук і перегляд інформації про користувачів, накладати обмеження доступу до системи, зберігати списки користувачів у вигляді архіву. Система повинна надати можливість користувачам проходити в деякому значенні «анкети», за підсумками якого можна буде зробити висновки щодо необхідності даної послуги, її параметрів та характеристик.

Інформаційна модель розв'язання задачі створення та розміщення завдань на проєктованому ресурсі підтримки організаційних процесів дистанційного навчання призначена для графічного зображення вхідних інформаційних потоків в систему та вихідних потоків даних. Вхідними потоками є навчальна інформація для діяльності системи та дані про користувача, про завдання та дані про майданчик розміщення завдань для кожного користувача. Вихідними потоками даних слугують всі дані завдань.

## 3 РОЗРОБКА ТА ДИЗАЙН ВЕБ-СЕРВІСУ

### 3.1 Варіанти взаємодії

Створений веб-ресурс є відкритим, а отже ним може користуватись будь-яка людина, яка має доступ до всесвітньої мережі Інтернет.

Вводячи адресу клієнт потрапляє одразу на головну сторінку. Саме через неї починається робота безпосередньо з функціоналом ІС. Переходячи на головну сторінку користувач зверху може побачити наступне:

- 1 Побачити детальну інформацію про портал.
2. Побачити слайдер з фото порталу.
3. Види послуг.
4. Інформація про освітян.
5. Запис на певне завдання.
6. Футер з посиланнями на соціальні мережі, та швидкий макет сайту.

Мною була створена діаграма UML з метою візуального представлення системи разом із її основними користувачами, ролями, діями, щоб краще зрозуміти, змінити, підтримувати або документувати інформацію про систему. Головними можливостями користувача на створеному веб-ресурсі є отримання інформації про створені завдання, можливість їх закріпити за собою, отримати звіт у вигляді електронного листа на пошту. Для підтримання роботи сайту потрібно мати адміністратора (постійне оновлення та редагування) (Рис. 3.1).

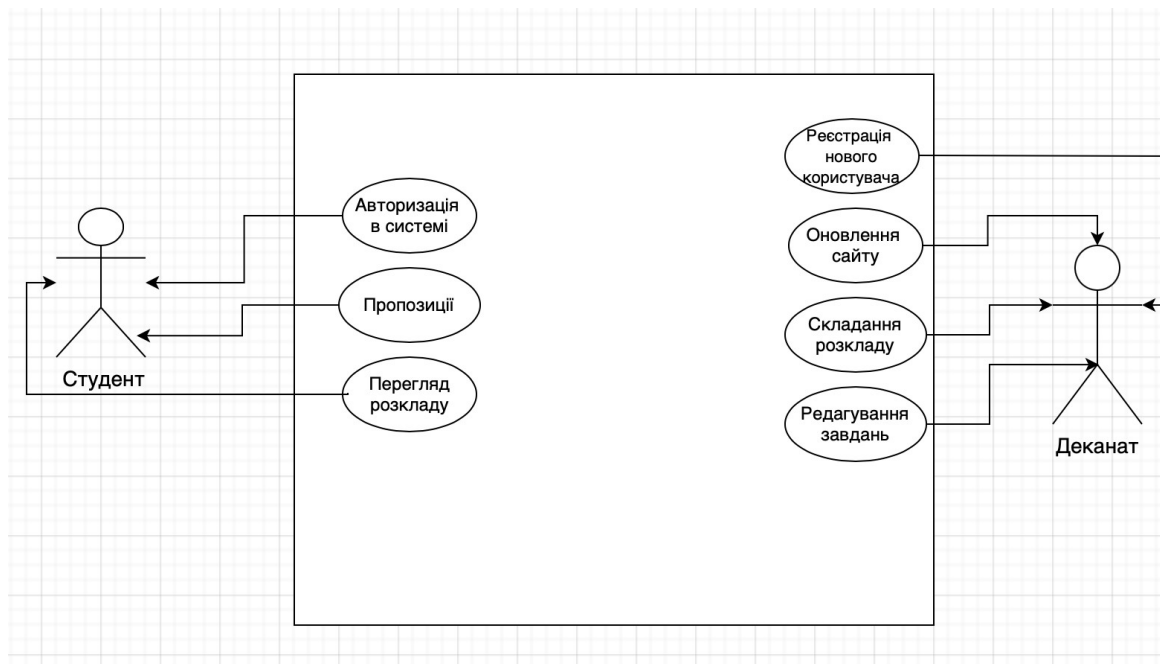


Рисунок 3.1 – UML-діаграма

### 3.2 Архітектура інформаційної системи

Архітектура інформаційної системи (ISA) відноситься до бізнес-процесів і політик, структурі системи, технічній структурі та технологіям продукту, необхідним для бізнесу або інформаційної системи.

Архітектура містить детальний опис дизайну, вмісту, перелік поточного обладнання, програмного забезпечення та мережевих можливостей комп'ютеризованої системи. Ця архітектура включає як апаратне, так і програмне забезпечення, яке використовується для надання рішення клієнту. Він також містить деталі довгострокових планів, як-от оновлення та/або заміна старого обладнання та програмного забезпечення. Архітектура програмного забезпечення клієнт/сервер — це гнучка модульна інфраструктура на основі повідомлень. Він спрямований на покращення зручності використання, сумісності, гнучкості та масштабованості на відміну від централізованих обчислень із розподілом часу на мейнфреймах. Компонентами цієї архітектури є клієнт, машина-запитувач і

сервер. Вони з'єднані один з одним через локальну мережу (LAN) або глобальну мережу (WAN).

Сервер виконує такі функції:

- Обробка запиту.
- Повернути результат клієнту.
- Зверніть увагу на запит клієнта.

Згідно створених завдань, на рисунку 3.2 відображена загальна архітектура системи.

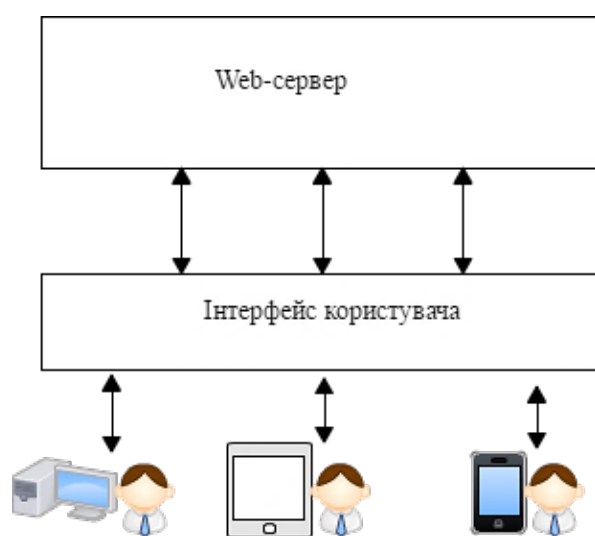


Рисунок 3.2 – Архітектура інформаційної системи

Для візуалізації процесів була створена діаграма IDEF0. IDEF0 використовує спрощені інструменти візуалізації, щоб полегшити залучення експертів галузі та спільне прийняття рішень. Як аналітичний інструмент, IDEF0 дозволяє розробнику моделей визначати функції. Функції включають виконання необхідних завдань, перевірку правильності поточної системи та перевірку помилок існуючої системи. Діаграма IDEF0 «поле та стрілка» відображають функцію як поле, а інтерфейси — як стрілки, що входять або виходять із поля до або з функції.

Діаграма IDEF0 зображена на рисунку 3.3. На вході запити користувача, який на виході має підтвердження отримання.

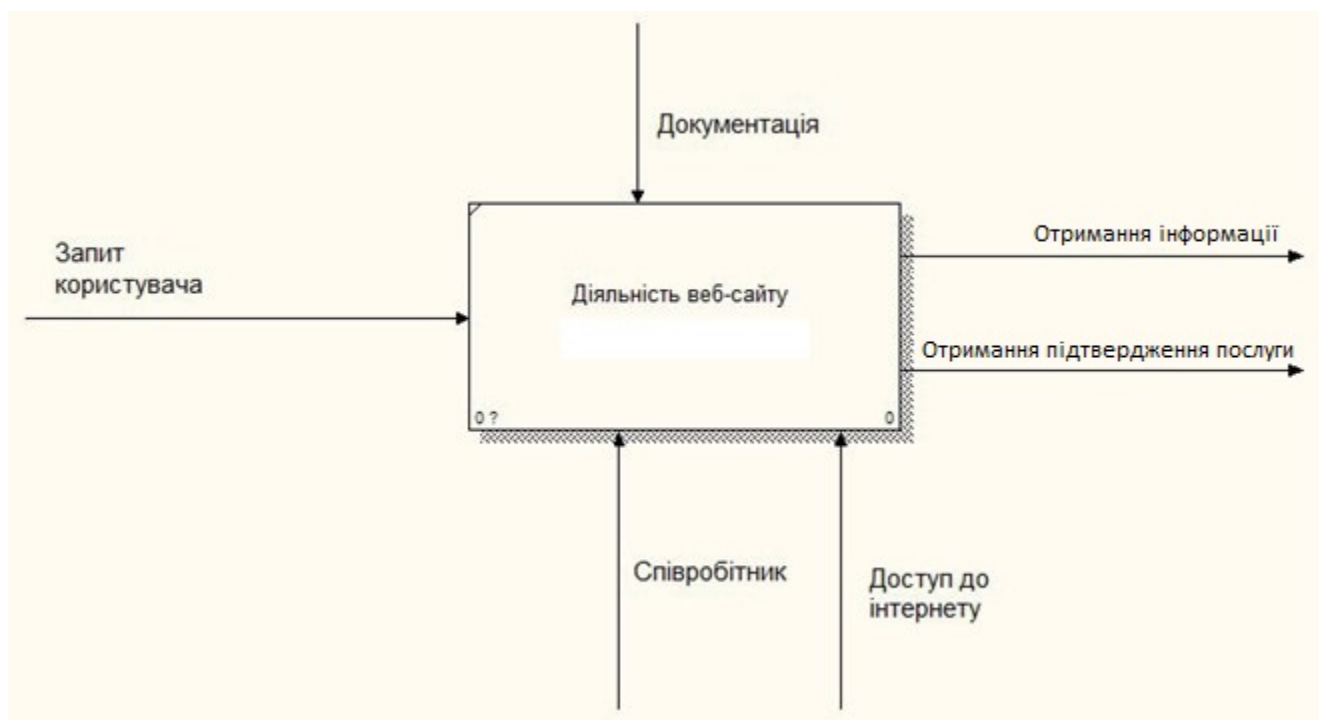


Рисунок 3.3 – Діаграма IDEF0

Основна перевага IDEF0 полягає в тому, що цей підхід виявився успішним в описі операцій рамок рольового моделювання, початкової цілі координації організованого аналізу для IDEF0. Крім того, визначення системних операцій можна швидко оптимізувати все більш детально, доки концепція не стане настільки короткою, наскільки вона підходить для відповідної функції прийняття рішень.

### 3.3 Розробка програмних модулів

У розробці програмного забезпечення існує довга історія розкладання системи на менші модулі. Ця модульність є стандартною технікою для роботи з великими та складними системами. Модулі розділяють дизайн або код системи.

Хоча модулі можуть виглядати як самодостатні елементи, оскільки всі вони є частинами більшого цілого, між ними завжди повинні бути зв'язки, які необхідно враховувати. Ці зв'язки обмежують можливість змінювати один модуль без впливу на інші модулі.

Модуль, який надає послуги іншим модулям, у свою чергу може потребувати використання послуг ще інших модулів. Ці необхідні служби називаються його контекстними залежностями. Контекстні залежності модуля та його інтерфейс, включаючи будь-які вимоги, яким мають відповідати потенційні клієнти, утворюють свого роду контракт з клієнтами.

Коли ваша система програмного забезпечення містить набір добре зрозумілих модулів, кожен із яких має власний інтерфейс і контекстні залежності, ви можете розглянути, чи можна будь-який із них повторно використовувати. Можливо, популярний набір модулів прийнятий як стандарт.

Тому для прив'язки створеної архітектури до середовища зберігання будемо використовувати фізичну модель. Саме вона визначає всі запам'ятовувальні пристрої, методи фізичної організації даних. Ця модель будується також враховуючі всі можливості, що надаються базою даних. Детальний опис структури називається схемою зберігання. Головним завданням даного етапу створення фізичної моделі є визначення співвідносності видів логічної моделі до типів даних, які можуть бути використані в цій базі.

Отже метою створення такої фізичної моделі є надання адміну всієї потрібної інформації для перенесення логічної моделі даних у базу.

Erwin автоматично дозволяє генерувати фізичну модель під конкретну базу даних. Тоді як логічна модель переходить у фізичну за наступними принципами:

- Сутність – таблиця;
- Атрибути – стовпці;
- Ключі – індекси.

Для створеного веб-сервісу файлова структура фронтенд- та бекенд-проєкту відображена на рис. 3.4.

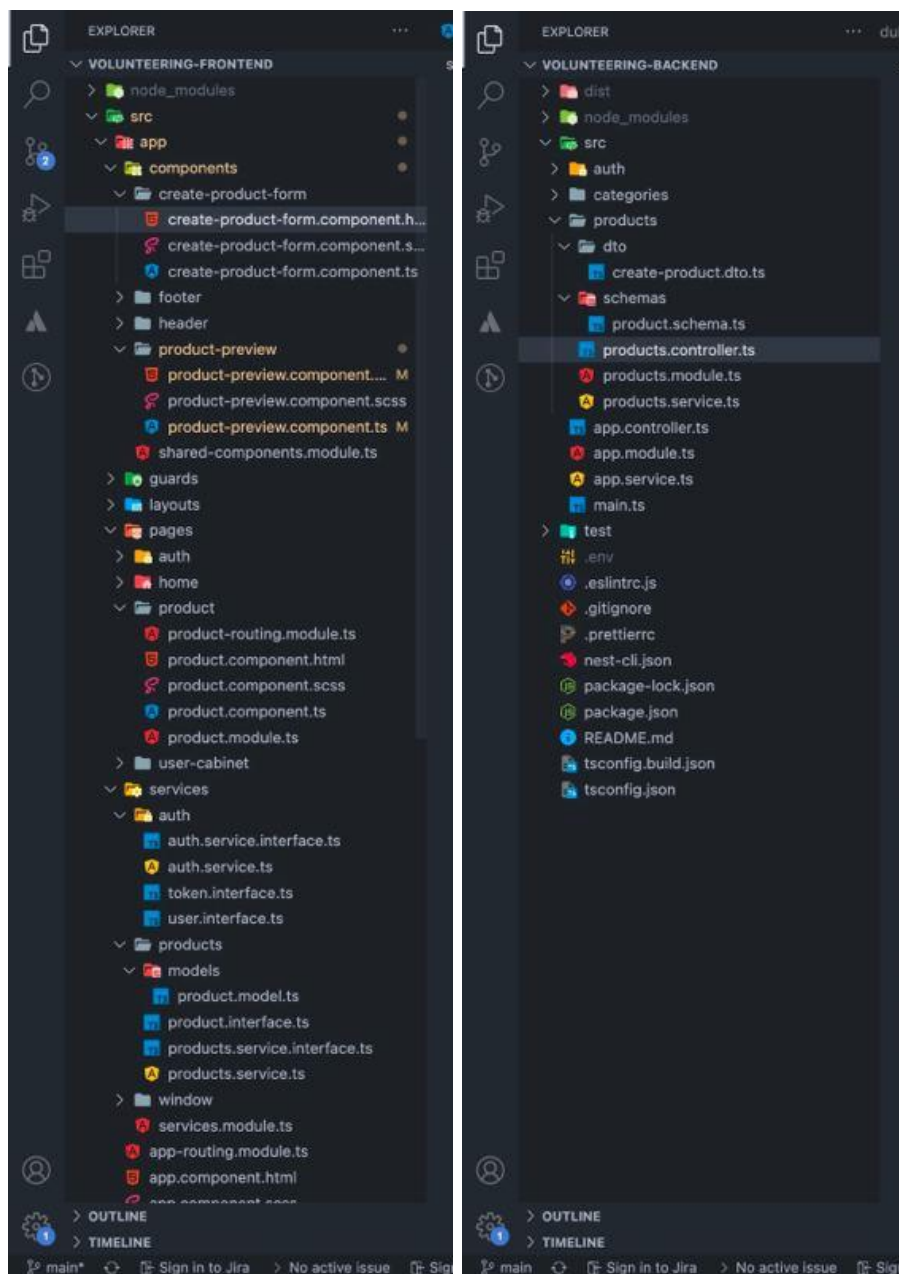


Рисунок 3.4 – Файлова структура фронтенд- та бекенд-проекту

Використовуючи базу даних можна зберігати графічні об'єкти, логотипи. Бізнес логіка ділиться на 2 види: клієнтська та адміністраторська робота з проектом. Користувач може залогінитись, знайти потрібну послугу, подивитись інформацію про сервіс, його параметри та має змогу обрати потрібне для себе. Адміністратор в свою чергу може впроваджувати оновлення чи редагувати наступні елементи: завдань, користувачів інформаційної системи підтримки

організаційних процесів дистанційного навчання та їх облікових даних, всі інші лише переглядати.

Створення акаунта на сайті. Це є складовою системи, яка несе відповідальність за облік користувачів ресурсу підтримки організаційних процесів дистанційного навчання, і не дає можливості помилкового редагування завдання. Для того, щоб клієнт мав доступ до певного спектру послуг на сайті, він повинен створити свій акаунт. Облік користувача також дає адміністраторам інформацію для того, щоб проводити контроль завдань та облікових даних.

```

11  users: 'user',
12  };
13
14  constructor(
15    @Inject('WindowServiceInterface')
16    private windowService: WindowServiceInterface,
17    private router: Router,
18    private http: HttpClient,
19  ) {
20    this._updateUserToken();
21  }
22
23  public get user(): User | null {
24    return this._userSubject.getValue();
25  }
26
27  private set user(users: User | null) {
28    this._userSubject.next(users);
29  }
30
31  public getCurrentUserPassword(): Observable<string> {
32    return this.http
33      .get<password: string>(>
34        `${this.getUserEndpoint}/password/${this.user?.id}`
35      );
36  }
37  .pipe(
38    map(( password ) => password)
39  );
40
41  public getUserById(id: string): Observable<User> {
42    return this.http
43      .get<User>(
44        `${this.getUserEndpoint}/${id}`
45      );
46  }
47
48  public signIn(user: UserloginData): Observable<string | null> {
49    return this._modifyObservable(
50      this.http
51        .post<access token: string>(>
52          this.signInEndpoint,
53          user
54        );
55  );
56
57  public createUser(user: UserRegistrationData): Observable<string> {
58    return this._modifyObservable(
59      this.http
60        .post<string>(
61          this.signUpEndpoint,
62          user
63        );
64  );
65
66  }
67
68  }

```

Рисунок 3.5 – Процес авторизації користувача у фронтенд-частині проєкту

Підсистема надсилає таку інформацію: імена користувача, дата народження та соціальні мережі. Вона оброблює отриману інформацію, та видає коротку характеристику, користувач має змогу редагувати свої дані, в разі потреби.



Спроектований продукт може існувати без інформаційної системи, але лише в якості збирача інформації про користувача, а оброблену інформацію нікуди не відсилає та не зберігатиме як базу даних.

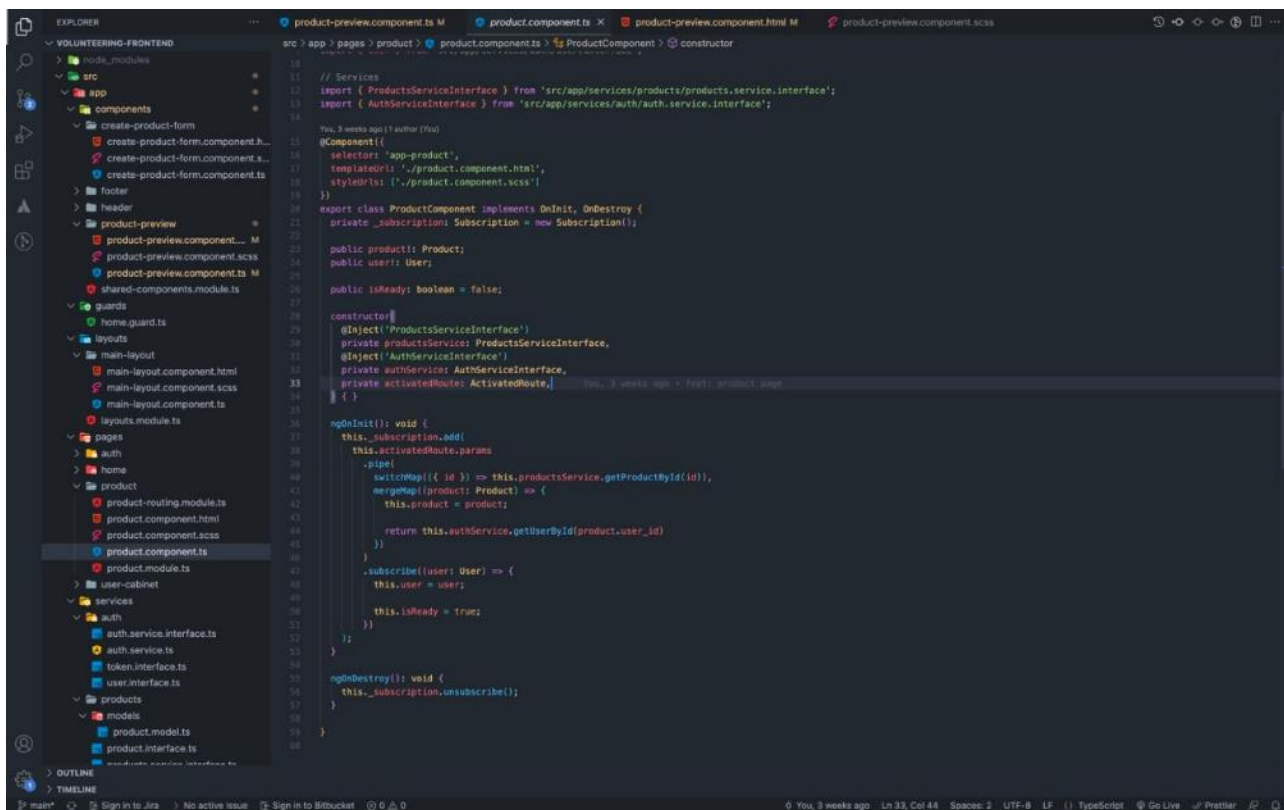


Рисунок 3.6 – Процес редагування завдання у фронтенд-частині проєкту

Елемент "Створення облікового запису" здатен виконувати такі функції:

- введення та редагування даних ;
- збереження та отримання інформації від системи контролю облікового запису;
- створення корзини для завдань;
- перевірки правильності введених даних.

Елемент "Формування корзини для завдань" створює "корзину" зареєстрованих користувачів інформаційної системи підтримки організаційних процесів дистанційного навчання, куди є можливість зберігати нові та запущені завдання для публікації на головній сторінці інформаційної системи.

Корзина може зберігати: номер завдання, ід користувача, вартість продукту та його коротку характеристику, має змогу порівнювати декілька раніше обраних послуг в середовищі інформаційної системи

```

1  import { Component, Inject, Input, OnDestroy, OnInit } from '@angular/core';
2
3  import { Subscription } from 'rxjs';
4
5  // Interfaces
6  import { Product } from 'src/app/services/products/product.interface';
7  import { User } from 'src/app/services/auth/user.interface';
8
9  // Services
10 import { AuthServiceInterface } from 'src/app/services/auth/auth.service.interface';
11
12 You, 1 minute ago | 1 editor (Vue)
13 @Component({
14   selector: 'product-preview',
15   templateUrl: './product-preview.component.html',
16   styleUrls: ['./product-preview.component.scss']
17 })
18 export class ProductPreviewComponent implements OnInit, OnDestroy {
19   private _subscription: Subscription = new Subscription();
20
21   @Input()
22   public product!: Product;
23
24   public user!: User;
25   public isLoading: boolean = false;
26
27   constructor(
28     @Inject('AuthServiceInterface')
29     private authService: AuthServiceInterface,
30   ) {}
31
32   ngOnInit(): void {
33     this._subscription.add(
34       this.authService.getUserId(this.product.user_id)
35         .subscribe((user: User) => {
36           this.user = user;
37           this.isLoading = true;
38         })
39     );
40
41   }
42
43   ngOnDestroy(): void {
44     this._subscription.unsubscribe();
45   }
46
47   public percentOfCollected(): string {
48     const { amount, collected } = this.product;
49     return (collected / amount * 100).toFixed(2);
50   }
51 }

```

Рисунок 3.7 – Створення нового компонента

Контроль валідних даних проводиться за допомогою автоматизованих налаштувань інформаційної системи, для того, щоб визначити актуальність вже зареєстрованих облікових записів. Це дозволяє звільнити місце в інформаційній базі даних, а також допоможе створити її більш оглядною.

Видалені дані можуть зберігатись в архівах, а також бути систематизованими по папках за місяцем або роком.

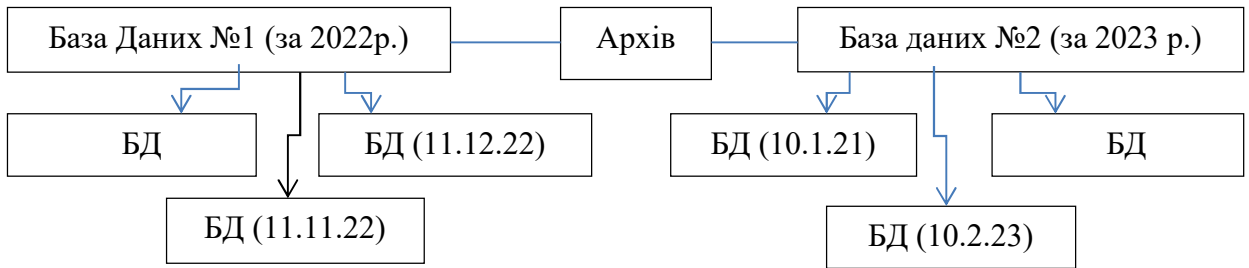


Рисунок 3.8 – Структурна схема підсистеми «Архів»

Визначимо структуру створення навчального завдання на сайті ресурсу підтримки організаційних процесів дистанційного навчання:

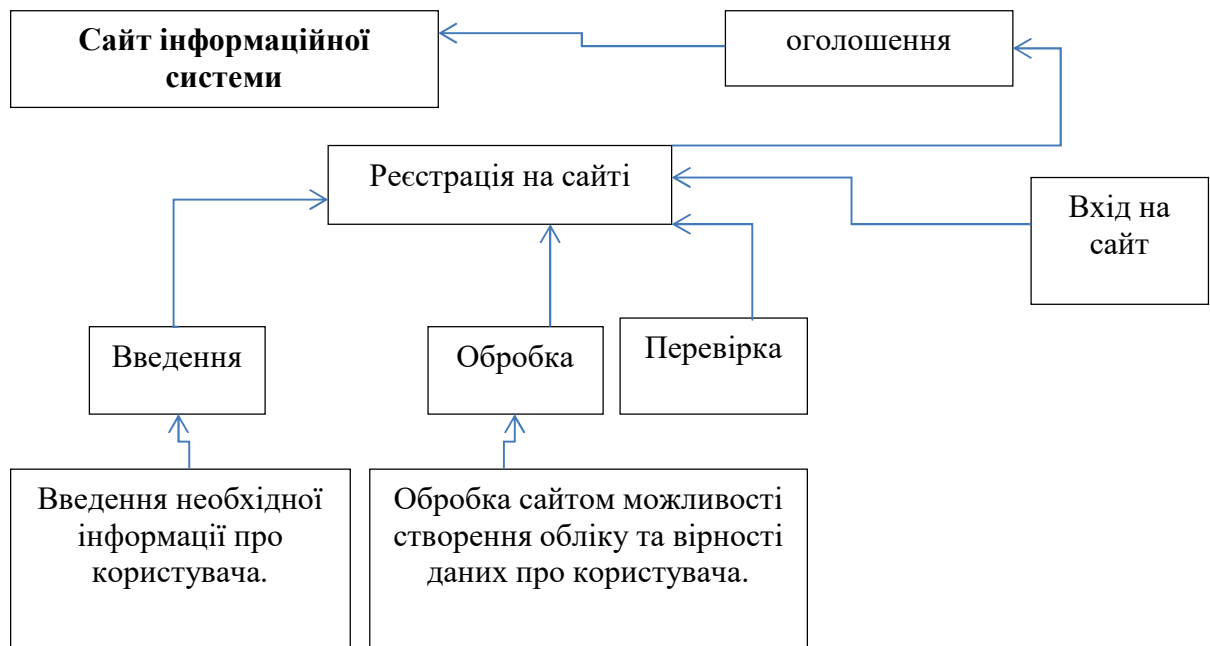


Рисунок 3.9 – Схема створення завдання на порталі

Зробивши все відповідно схемі завдання потрапляє в систему у вигляді віртуального резюме, потім автоматично переноситься до інформаційної бази порталу, а згодом завдання з'явиться на головній сторінці і буде видно для цільової аудиторії.

## 5.4 Розробка дизайну

Розробка макету – важливий етап у розробці сайту. Він дозволяє уявити програмісту загальне положення елементів майбутнього проєкту. Макет являє собою зображення виконане або графічними редактором, або розробкою за допомогою HTML та CSS. Розроблювана ІС повинна забезпечувати наступний функціонал:

- Достовірність та повнота інформації. Постійне оновлення сайту та додавання нових елементів.
- Простий інтерфейс. Для зручного користування необхідно створити інтуїтивно зрозумілий інтерфейс.
- Швидкість роботи та відображення інформації. Необхідно використовувати нові технології для швидкої роботи системи.

Вирішено створити ІС, як web-орієнтований додаток для браузерів. В плані доступності web-технології найбільш зручні, так як надають швидший доступ до даних та не вимагають встановлення додаткового програмного забезпечення на персональний комп'ютер.

Головними дизайнерськими ознаками сучасної ІС підтримки організаційних процесів дистанційного навчання є:

- логотип;
- фірмовий стиль;
- зрозумілий інтерфейс.

Фірмовий стиль передбачає візуальний образ компанії або її продукту. Це набір графічних зображень і смислових форм, об'єднаних однією спільною ідеєю.

За основу було вирішено використовувати відтінки теплого кольору. Для збільшення функціональності використовуємо React.js для роботи з прихованими блоками. Як результат, поєднання HTML5, Scss та React.js дозволили створити яскраву та якісну інформаційну систему, яка не залишить нікого байдужим. На рисунках 3.10–3.14 представлено макет сайту.

The screenshot shows the registration form on the Dii website. The header includes the Dii logo on the left and 'Допомогти' and 'Увійти' with a user icon on the right. The form is titled 'Реєстрація' and has two tabs: 'Вхід' and 'Реєстрація'. The registration form contains the following fields: 'First name', 'Last name', an email address 'qwerty123@gmail.com', and a password field with six asterisks. A 'Підтвердити' button is at the bottom of the form. The footer contains links for 'Умови користування сайту' and 'Політика конфіденційності'.

Рисунок 3.10 – Форма реєстрації нового користувача

Після реєстрації відбувається зворотна автентифікація користувача шляхом перевірки правильності електронної пошти надсиланням листа з посиланням для підтвердження реєстрації.

The screenshot shows the user profile section on the Dii website. The header includes the Dii logo on the left and 'Допомогти' and 'Увійти' with a user icon on the right. The form is titled 'Ваш профіль' and has two tabs: 'Вхід' and 'Реєстрація'. The profile section contains the following fields: an email address 'qwerty123@gmail.com' and a password field with six asterisks. A 'Увійти' button is at the bottom of the form. The footer contains the Dii logo and a list of links: 'Умови користування сайту', 'Мапа сайту', 'Як розмістити оголошення?', 'Як допомогти?', 'Політика конфіденційності', 'Правила безпеки', and 'Мій профіль'.

Рисунок 3.11 – Форма входу користувача в персональну секцію сервісу

Після входу на сайт, відкривається стартова сторінка ресурсу підтримки організаційних процесів дистанційного навчання. Головна сторінка проєктованого ресурсу підтримки організаційних процесів дистанційного навчання має дані про завдання: потреби чи послуги освітян.

The screenshot displays the user profile management interface for the Dii website. The page features a dark header with the 'Dii' logo on the left and 'Допомогти' and 'Профіль' with a user icon on the right. The main content area is divided into three sections: 'User info', 'Change password', and 'Contacts'. The 'User info' section contains input fields for 'Name' (filled with 'Roman'), 'Surname' (filled with 'Nawaz'), and 'Email' (filled with 'qwerty123@gmail.com'). The 'Change password' section includes fields for 'Current password', 'Password', and 'Repeat password'. The 'Contacts' section lists social media handles: '@telegram', '@instagram', '@email', and '@tel', with an 'Add new contact' button below. A footer contains the 'Dii' logo and a list of links: 'Умови користування сайту', 'Політика конфіденційності', 'Мапа сайту', 'Правила безпеки', 'Як розмістити оголошення?', 'Мій профіль', and 'Як допомогти?'.

Рисунок 3.13 – Сторінка редагування персональних даних користувача ресурсу підтримки організаційних процесів дистанційного навчання

Дана секція ресурсу дозволяє редагувати персональні дані користувача, підключені соціальні профілі та пароль доступу до сайту. Пароль зберігається у базі даних в зашифрованому вигляді.

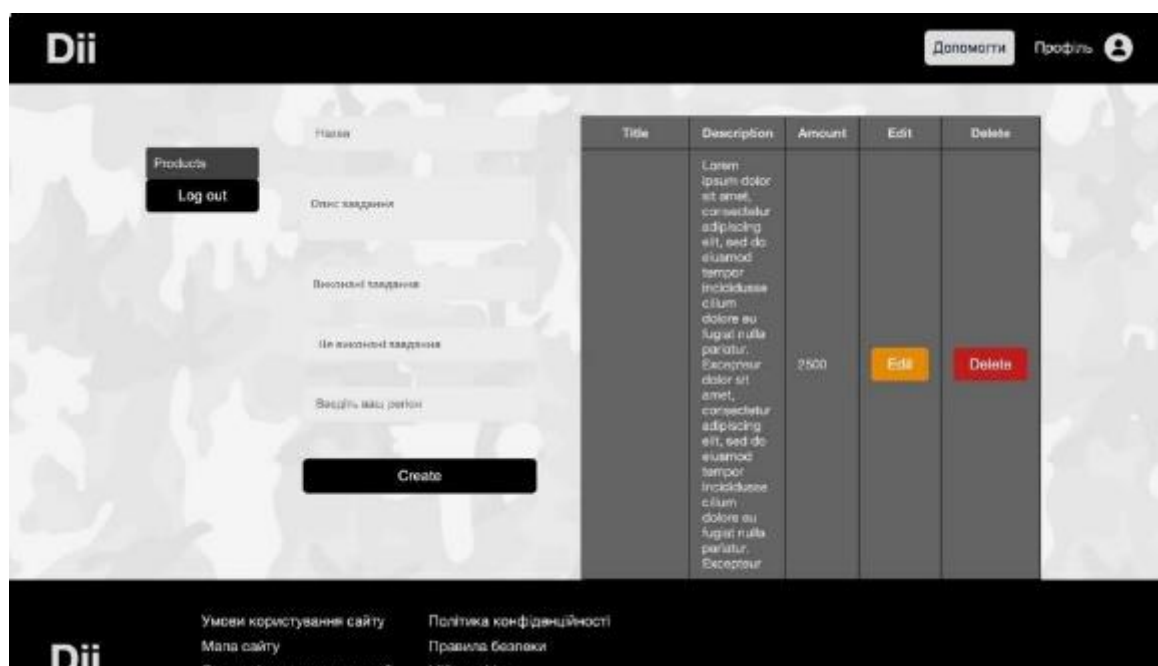


Рисунок 3.14 – Редагування завдання

Дана секція проєктованого ресурсу підтримки організаційних процесів дистанційного навчання дозволяє редагувати опис завдання, видаляти його та змінювати його параметри. Створення нового завдання можливе при натисканні кнопки Create.

## 5.5 Розгортання програмного забезпечення проєктованої системи

Формулювання апаратних рішень щодо розгортання та роботи ресурсу підтримки організаційних процесів дистанційного навчання буде насамперед залежати не від апаратних конфігурацій сервера а від засобів, які оперують виконанням роботи додатку. Тут мається на увазі програмний комплекс роботи веб-сервера від якого залежить якість роботи додатку. Формулювання вимог, щодо апаратного рішення, залежить від багатьох факторів роботи ресурсу підтримки організаційних процесів дистанційного навчання. Правильне функціонування проєктованої інформаційної системи неможливе без

використання відповідного програмного забезпечення. Програмне забезпечення для роботи інформаційної системи являє собою комплекс взаємозв'язаних програмних реалізацій що підтримують функціонування самого сервісу.

Для функціонування роботи проєктованої системи потрібне програмне забезпечення, яке керує його роботою та дозволяє під'єднуватися безпосередньо портативно або стаціонарно до веб-ресурсу. Для цього нами спроектовано фронтенд- та бекенд-частини проєкту. Функціонал роботи додатку створений мовою програмування JavaScript та бібліотеки React.js що безпосередньо вимагає інтерпретатора коду.

Для підключення додатку потрібні повноцінний веб-сервер. Веб-сервер містить в собі встановлене програмне забезпечення, яке відповідає за роботу як протоколів з'єднання так і самого функціонування додатку. Комп'ютер-сервер повинен містити в собі налаштування та підключені програмні засоби, що забезпечують роботу веб-сервісів та встановлення сторонніх додатків на стороні бекенд-ресурсу підтримки організаційних процесів дистанційного навчання. Оскільки проєктований веб-додаток призначений для роботи у мережі Інтернет, тому серверні програми будуть вимагати певних апаратних ресурсів для його обслуговування.

Для оперування роботою веб-сервера, тобто комп'ютера віддаленого доступу, потрібний веб-сервер. Найпоширенішим веб-сервером сьогодення є Apache. У нашому випадку буде використовуватись веб-сервер Apache версії 2.4.54.

Робота додатку неможлива без використання бази даних. Щоб мати можливість збереження даних та їх оперування потрібно щоб додаток мав можливість підтримки доступу до файлової системи. Найпоширенішим рішенням щодо використання бази даних є MySQL-сервер бази даних. Також всі вище наведені компоненти повинні бути налаштовані та функціонувати як єдина система у цілому.

Отже, для оптимальної роботи проєктованої інформаційної системи підтримки організаційних процесів дистанційного навчання потрібна висока



пропускна можливість каналу передачі даних веб-сервера, тобто щоб швидкість завантаження даних із сервера була досить високою при великій кількості одночасних користувачів з усіх регіонів. Це є ключовою умовою при формуванні апаратних характеристик та рішень. У нашому випадку, додаток не в'являє собою надпотужний сервіс, що оперує великими обсягами інформації, тому швидкодія роботи інформаційної системи не вимагає великих обсягів апаратних засобів.

Після написання програмного коду треба створити окремі файли, завантажити їх на портал у відповідні вкладки та почати тестування сторінки. Перевірка сервісу відбувається різними методами на нормальну працездатність. Тестування проводиться як для нового, так і вже існуючого сервісу.

Протягом всього часу розробки проводилось неодноразові перевірки на правильне функціонування та відсутність помилок. Для цього були створені симулятори смартфона з різними видами діагоналей, щоб протестувати розроблений сайт на наявність помилок та на варіант адаптивності. По результатам цього в код можуть вноситись правки.

### **3.6 Аналіз веб-сервісу**

Після створення програмного продукту підтримки організаційних процесів дистанційного навчання виконаємо перевірку, де перевіряємо якість продукту. Перевірка є функціональна та нефункціональна. При функціональній перевірці тестування проходить сайт на коректність роботи, які були прописані в ТЗ. При нефункціональному тестуванні перевіряються характеристики продукту.

Аналіз працездатності системи полягає у комплексному тестуванні усіх функціональних елементів та продукту в цілому.

Доцільність тестування нашого продукту полягає у:

- Робота сесії користувача.
- Тестування з'єднання із базою даних.
- Тестування кросплатформенності сервісу.

- Тестування працездатності технологій.
- Перевірці функціонування таблиць бази даних.

Для тестування окремих модулів ресурсу підтримки організаційних процесів дистанційного навчання роботи у код програми заносять функціональні тести, які обробляють та аналізують базу даних та при виявленні помилки, надсилають запит у реєстр системи. Цей підхід називається як юніт-тест.

```
public class ConectBd : MonoBehaviour
{
    public static string path = "host=127.0.0.1;port=3306;user=root;password=;database=unity";
}
```

Рисунок 3.15 – Тестування з'єднання з БД

В базі при зміні даних проводиться тестування на цілісність (перевірка на відповідність ключів – індексів), після чого при необхідності виводиться запит.

Таблиця 3.1 – Тестування інформаційної системи підтримки організаційних процесів дистанційного навчання

TC_ID	Test Scenario / Test Step Name	Actions	ER	Статус
Prerequisite. Маємо доступ до інтернету та сайт відкрито у вікні браузера Chrome				
TC1.1	Перехід по вкладкам на сторінці	Натиснути лівою клавішею миші на вкладку Головна Натиснути лівою клавішею миші на вкладку Галерея	Перехід на відповідну вкладку	+
TC1.2	Перевірка посилання	На Головній сторінці натиснути на посилання (слово Контакти)	Перехід на вкладку Контакти	+
TC1.3	Відправлення заявки на отримання послуги	В поле вводу Ваше ім'я ввести Ваше ім'я. В поле вводу Ваш email ввести Ваш email адрес. В поле вводу Тема ввести кілька слів, що до теми вашого інтересу. В поле вводу Опис заявки ввести текст, де Ви розкриєте тему вашого запиту Натиснути кнопку Надіслати	Повідомлення про успішне відправлення заявки	+

З розробленим мобільним додатком варто провести такі тести:

- Usability тест (тестування зручності роботи сервісу підтримки організаційних процесів дистанційного навчання). Під час перевірки визначається легкість використання та адаптивність інтерфейсу додатку, також виявляється вірогідність виникнення помилок у структурі. Результат дає можливість вчасно виявити правильність використання користувачем продукту та як швидко він адаптується до роботи системи.
- Виявлення помилок. Кожна функціональна дія аналізується юніт-тестом з метою виявлення помилок, які виникли через невідповідність очікуваних та отриманих даних. Для цього для кожної дії створюється окремий юніт-тест, що піддає тестуванню дію на різні вірні та невірні параметри. При аномальній поведінці доступу до ресурсу підтримки організаційних процесів дистанційного навчання або його збоїв. Аналізувалася поведінка, а виявлені помилки виправлялися.
- Аналіз стійкості до надмірних навантажень. Цей аналіз імітує велике навантаження у вигляді одночасного користування великої кількості користувачів (десятків або тисяч) розробленого веб-ресурсу для визначення життєздатності продукту з підтримки організаційних процесів дистанційного навчання при надмірних навантаженнях або ж інтенсивна тривала робота системи в умовах невеликих ресурсів. Такий тип тестування обов'язковий для розробленого додатку з очікуваною великою кількістю користувачем. У ході аналізу перевіряється не лише продукт, а в цілому комплексна робота апаратної частини серверу, окремих модулів, ядро проекту та інших компонентів підтримки організаційних процесів дистанційного навчання.
- Тестування на коректність отриманих даних. У базу очікувано надсилаються неприпустимі дані у відповідні поля, які можуть не вірно оброблятися продуктом інформаційної системи. Далі аналізується поведінка активності програми під час обробки даних.
- Аналіз сумісності та адаптивності під різні версії Android. Сервіс запускався на пристроях, які працюють на різних версіях Android з метою своєчасного

виявлення тонкощів роботи програми в умовах різних операційних систем. Під час проведення такого типу тесту продукт успішн встановився на пристроях з операційними системами типу: Android 4.4.4 (Kit Kat), Android 5.0 (Lollipop), Android 5.1 (Lollipop), Android 6.0 (Marshmallow).

Для стабільного функціонування бази даних необхідне надійне підключення, Для забезпечення надійного з'єднання із базою даних використовувався xml функція SQLite\_connect. Цей тип функції встановлює безперебійний зв'язок із базою даних, а також при помилці транзакції автоматично відновлює зв'язок. Маніпуляція таблицями забезпечується лише за допомогою програмного коду xml. Під час внесення тестового запису інформації у базу даних зафіксовано гарний результат виконання за часом транзакції, що забезпечує оптимальний алгоритм тесту.

- Аналіз життєздатності технології з обміну даними.

Тест технології полягає у відправленні запиту на серверну частину із передачею параметрів. Параметри помічені як тестові для передачі скриптам серверу. Аналіз швидкості передачі запитів продемонструвало, що всі процеси виконувались швидко та обробились тестовими скриптами на серверній частині продукту. Наш приклад під час відправки часових циклічних запитів на серверну частину локально встановленої машини використовуючи за основу технологію бібліотеки React.js затримку не виявив та відповіді від серверу. Крім цього React.js не створює надмірного навантаження на веб-ресурс, а також просто дає змогу контролювати отриману інформацію від серверної частини.

Робочий процес сесії на ресурсі підтримки організаційних процесів дистанційного навчання. Під час процесу авторизації або реєстрації запускається елемент створення сесії користувача. Під час тестового навантаження процесу реєстрації, сесії автоматизовано присвоєні без присутності будь-якої затримки. Завдяки цьому ми можемо якісно оцінити алгоритми аутентифікації у продукті. Життєздатність роботи алгоритму сесії користувача за вибраними нами показниками була на найвищому рівні.

Перевірка працездатності таблиць бази даних. Під час тестового навантаження та наповнення затримки були відсутні. Таблиці заповнились без проблем та зауважень. Кожна окрема таблиця має режим хешування, через це процес виконання запитів набагато менший.

Перевірка кросплатформенності продукту з підтримки організаційних процесів дистанційного навчання. Під час запуску ресурсу на різних типах операційної системи Android різниця не спостерігалась якості відображення контенту. Усі елементи веб-ресурсу відображались адекватно та злагоджено, що демонструє про адаптивність продукту у різних середовищах сімейства Android.

Першою важливою задачею тестування функціональності інтерфейсу було отримання токенів користувача в кожному з сервісів по-черзі. Для цього, ми повинні створити авторизації, отримати від користувача номер телефону та надіслати його через Rest API до кожного сервісу.

На рис. 3.16 можна побачити графічний інтерфейс авторизації на мобільному пристрої.



Рисунок 3.16 – Форма входу на мобільному пристрої

Далі, після авторизації, нам у відповідь надходить токен користувача, який дозволяє нам робити всі інші запити.

Узагальнюючи, можна сказати, що система виконує свої функції та має можливість до масштабування та додавання нових функцій. Провівши перевірку веб-сервісу не було виявлено критичних помилок у роботі.

### **3.7 Висновки до розділу 3**

Мною було створено інформаційне забезпечення ресурсу підтримки організаційних процесів дистанційного навчання для з дотриманням наступних тезисів: цілісність інформації; достовірність інформації; контроль інформації; захист від несанкціонованого доступу; єдність та гнучкість; стандартизація та уніфікація; адаптивність; мінімізація вводу та виводу інформації.

У веб-сервісу інформаційного забезпечення підтримки організаційних процесів дистанційного навчання зроблений зручний клієнтський інтерфейс та додані наступні функції, а саме при відправленні завдання на перевірку можна додати графічний або текстовий об'єкт та змінити його налаштування, вигляд, опис в будь-який час.

## ВИСНОВКИ

Аналізуючи сьогодні у всіх напрямках життєдіяльності всі галузі намагаються створити всі умови для максимально зручної роботи на просторах Інтернету. Тому розробка порталів, які дозволяють вчитись чи працювати дистанційно є найбільш актуальною в даний час.

Метою дипломного проєкту є створення ІС навчального порталу. Головною проблемою існуючих аналогів є недостатня обізнаність користувача про сам портал. Головним елементом ІС є простота, зручне отримання вичерпної інформації з завдання для навчального процесу та можливість дізнатися детально про портал.

Завдяки розробці веб-сервісу для освітян ми вирішимо такі проблеми як: відстеження історій завдань, збереження необхідних контактів, групування заявок від керівника навчання, верифікація осіб які використовують основні функції. З цією метою нами проведено вибір сучасних інструментів та аналіз методологій для розробки веб-сервісів та бази даних для них.

Обрано: мову розмітки гіпертексту HTML; SCSS – скриптову метамову, яка інтерпретується в каскадні таблиці стилів (CSS); React.js – фреймворк для JavaScript – бібліотека допомагає створювати веб-програми – сайти, які завантажуються лише один раз; для розробки проєктованого ресурсу підтримки організаційних процесів дистанційного навчання нами обрано MySQL – це вільна реляційна система яка керує базами даних. Вона є найбільш широко використовуваною базою даних з відкритим вихідним кодом.

Предметна область проєктування – інформаційна система по організації підтримки організаційних процесів дистанційного навчання. Робота полягає у створенні ІС порталу, для підтримки організаційних процесів дистанційного навчання зі зручним інтерфейсом, виконаним на базі React.JS. Відповідно, під час перегляду завдання користувач повинен бачити: короткий опис послуг з посиланням на повний опис на сторінці з відомостями.

В ході розробки інформаційної системи були розроблені наступні елементи проєктованого сервісу: код інформаційної системи; база даних; функціонал інформаційної системи; програмне забезпечення для управління сервісом у форматі фронтенд- та бекенд-проєкту. Веб-ресурс може редагувати, доповнювати, здійснювати пошук та аналіз інформації про користувачів, надавати обмеження доступу до ресурсу, архівування та збереження даних.

Продукт надає можливість користувачам заповнювати так звані "анкети", за результатами яких можна створити висновок щодо необхідності даного завдання, його параметрів та характеристик. Інформаційна модель розв'язання задачі створення та розміщення завдань на проєктованому ресурсі підтримки організаційних процесів дистанційного навчання призначена для графічного зображення вхідних інформаційних потоків в систему та вихідних потоків даних. Вхідними потоками є юридична інформації для діяльності системи та дані про користувача, про завдання та дані про майданчик розміщення завдань для кожного користувача. Вихідними потоками даних слугують всі дані завдань.

У веб-сервісу інформаційного забезпечення підтримки організаційних процесів дистанційного навчання зроблений зручний клієнтський інтерфейс та додані наступні функції, а саме при відправленні завдання на перевірку можна додати графічний або текстовий об'єкт та змінити його налаштування, вигляд, опис в будь-який час.

Створення цього ІС у дозволило економити час і користувача. Не потрібно зустрічатися, та не буде ситуації, коли в порталі не можуть взяти слухавку і т.д.

Сайт проєктованої інформаційної системи був написаний за допомогою HTML5, SCSS, JavaScript та React.js. Результат виконаної роботи є аналіз: актуальності розглянутого питання, предметної частини та потреб користувача. Також спроєктовано веб-ресурс, розроблено технічне завдання до системи та план тестування продукту.






## ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Азаров О. Д. Комп'ютерні мережі: навчальний посібник / О. Д. Азаров, С. М. Захарченко, О. В. Кадук. Вінниця: Вінницький Національний Технічний Університет, 2013. 371 с.
2. Архангельський А. Я. Програмування на Java. Київ: Біном, 2015. 425 с.
3. Богач І. В., Довгалець С. М., Дубовой В. М., Алгоритми розв'язання задач з програмування. Решебник. Вінниця: ВНТУ, 2017. 119 с.
4. Дизайн сайту. URL: <https://onix-systems.com/> (дата звернення: 20.04.2023)
5. Діаграма IDEF0. URL: <https://uk.wikipedia.org/wiki/IDEF0> (дата звернення: 20.04.2023)
6. Додонов А. Г., Ланде Д. В. Розпізнавання інформаційних операцій. Київ: ООО «Інжинирінг», 2017. 284 с.
7. Дятибратов А. П.; Гудино, Л. П.; Кириченко, А. А. Обчислювальні системи, мережі та телекомунікації; Київ: Фінанси і статистика. 2013. 512 с.
8. Кей С. Хорстманн, Гари Корнелл. HTML. Библиотека профессионала, том 1. Основы. 9-е издание. Киев: Генеза, 2018.
9. Кузь М. В., Соловко Я. Т. Методологія формування узагальненого критерію якості програмного забезпечення в умовах невизначеності. *Вісник Вінницького політехнічного інституту*. 2015. №5. С. 104–107.
10. Ліщинська Л. Б. Основні аспекти автоматизації роботи з клієнтами засобами CRM-систем. *Вісник Хмельницького національного університету. Економічні науки*. 2015. № 5 (1). С. 206–209.
11. Методи розробки веб-сервісів. URL: <http://webstudio2u.net/ua/webdesign/354-site-develop-methods.html> (дата звернення: 20.04.2023)
12. Методи та моделі розроблення комп'ютерних систем і мереж: монографія / В. С. Пономаренко, С. В. Мінухін. Харків: Вид. ХНЕУ, 2016. 316 с.
13. Милорадов К. А. Підвищення ефективності взаємодії з клієнтами за допомогою хмарних сервісів. *Міжнародний журнал експериментальної освіти*. 2015. № 3–1. С. 70–71.

14. Мітчел К. Керман. Програмування та налагодження на Java. Навчальний курс: пер. з англ. Київ: Свічадо, 2020, 1024 с.
15. Оліфер В. Г. Мережеві операційні системи / В. Г. Оліфер. Київ: Свічадо, 2019. 612 с.
16. Онлайн калькулятори. URL: [https:// webcala. net/](https://webcala.net/) (дата звернення: 20.04.2023)
17. Пол К. Створення додатків на Java. Пер. з англ. Київ: Свічадо, 2018. 728 с.
18. Попов В. Б. Java. Підручник. Київ: Світоч, 2020. 522 с.
19. Принципи побудови та етапи проектування баз даних. URL: [https:// stud. com. ua / 35671 / informatika / kontsepsiya\\_baz\\_danih](https://stud.com.ua/35671/informatika/kontsepsiya_baz_danih) (дата звернення: 20.04.2023)
20. Редько В. М.; Басараб, І. А. Базы даних та інформаційні системи; Київ: Знання, 2013. 183 с.
21. Сандизайн. URL: [https:// sandizain. com/](https://sandizain.com/) (дата звернення: 20.04.2023)
22. Управління проектами: Практичні аспекти реалізації стратегій регіонального розвитку. URL: [https:// issuu. com / oleksiiverbytskyi / docs / project\\_management\\_handbook\\_final](https://issuu.com/oleksiiverbytskyi/docs/project_management_handbook_final) (дата звернення: 20.04.2023)
23. Фісун М., Дворецький М., Дворецька С. Побудова моделей для оптимізації структури бази даних вузла у корпоративних інформаційних системах. *ІТКІ*, 2020. vol 48, № 2, С. 52–60.
24. Чмихало Н. В., Оккерт І. Л., Ольховська А. Б., Сотнікова Н. В. Формування ефективної логістичної політики продовольчого підприємства в умовах нестабільного зовнішнього середовища. Харків: ХНЕУ, 2017.
25. Як побудувати календарний графік проекту. URL: [http:// elearn. univector. net / file. php / 58 / Rach / 04 / 32. pdf](http://elearn.univector.net/file.php/58/Rach/04/32.pdf) (дата звернення: 20.04.2023)
26. Acharjya DP, Ahmed K. 2016. A survey on big data analytics: challenges, open research issues and tools. *International Journal of Advanced Computer Science and Applications*. 7 (2): 511–518.
27. Asencio-Cortés G, Morales-Esteban A, Shang X, Martínez-Álvarez F. 2018. Earthquake prediction in California using regression algorithms and cloud-based big data infrastructure. *Computers & Geosciences*. 115:198–210 DOI 10.1016 / j. cageo. 2017.10.011.

- 
28. Beginning HTML. HTML Development by Giulio Zambon. 2012. 436p.
  29. Borodo SM, Shamsuddin SM, Hasan S. 2016. Big data platforms and techniques. Indonesian. *Journal of Electrical Engineering and Computer Science*. 1 (1): 191–200 DOI 10.11591 / ijeeecs. v1. i1. pp191–200.
  30. Bruce Eckel. Thinking in HTML (4th Edition). Prentice Hall PTR. 2006. 425 p.
  31. Bruno Emaus (University of Amsterdam). Guidelines for primary school teachers for integration of ICT in their lessons. 2004. 417 p.
  32. Codd EF, Codd SB, Salley CT Providing OLAP (On-Line Analytical Processing) to User-Analysts: An IT Mandate. – EF Codd & Associates, 2008. 314 p.
  33. Ding W, Zhao Z, Wang J, Li H. 2020. Task allocation in hybrid big data analytics for urban IoT applications. *ACM Transactions on Data Science* 1 (3): 1–22.
  34. Favaretto, M., De Clercq, E., Schneble, C. O., & Elger, B. S. (2020). What is your definition of Big Data? Researchers' understanding of the phenomenon of the decade. *PloS one*, 15 (2), e0228987. <https://doi.org/10.1371/journal.pone.0228987>.
  35. Feng M, Zheng J, Ren J, Hussain A, Li X, Xi Y, Liu Q. 2019. Big data analytics and mining for effective visualization and trends forecasting of crime data. *IEEE Access*. 7:106111–106123 DOI 10.1109 / ACCESS. 2019.2930410.
  36. FPDF Library. URL: <http://www.fpdf.org/> (дата звернення: 20.04.2023)
  37. Furht B, Villanustre F. 2016. Introduction to big data. In: Big data technologies and applications. Berlin, Heidelberg: Springer, 3–11.
  38. Gandomi, A. H.; Chen, F.; Abualigah, L. Machine Learning Technologies for Big Data Analytics. *Electronics* 2022, 11, 421. <https://doi.org/10.3390/electronics163421>.
  39. Gantz, J., & Reinsel, D. (2012). The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. IDC iView: *IDC Analyze the future*, (2012), 1–16.
  40. Györödi CA, Dumșe-Burescu DV, Zmaranda DR, Györödi RȘ, Gabor GA, Pecherle GD. Performance Analysis of NoSQL and Relational Databases with CouchDB and

- MySQL for Application's Data Storage. *Applied Sciences*. 2020; 10 (23): 8524. <https://doi.org/10.3390/app10238524>.
41. Haleem A, Javaid M, Khan IH, Vaishya R. Significant Applications of Big Data in COVID – 19 Pandemic. *Indian J Orthop [Internet]*. 2020;54 (4): 526–8.
42. Han J. OLAP Mining: An Integration of OLAP with Data Mining. IFIP, 2009. 118p.
43. Hook D. W., Porter S. J. 2021. Scaling Scientometrics: Dimensions on Google BigQuery as an Infrastructure for Large-Scale Analysis. *Front. Res. Metr. Anal.* 6:656233. doi: 10.3389/frma.2021.656233.
44. HTML, 2nd Edition by Bryan Basham, Kathy Sierra, Bert Bates. 2018. 842p.
45. James Gosling; Bill Joy, Guy Steele, Gilad Bracha. The HTML Language Specification, Third Edition. Addison-Wesley. 2005. 212 p.
46. Kannan N, Sivasubramanian S, Kaliappan M, Vimal S, Suresh A. 2019. Predictive big data analytic on demonetization data using support vector machine. *Cluster Computing*. 22 (6): 14709–14720 DOI 10.11/s10586-018-2384-8.
47. Kaur P, Sharma M, Mittal M. 2018. Big data and machine learning based secure healthcare framework. *Procedia Computer Science*. 132:76–1059 DOI 10.1016/j.procs.2018.05.020.
48. Klein S. 2017. The world of big data and IoT. In: IoT solutions in Microsoft's azure IoT suite. Berkeley: Apress, 3–13.
49. Kroc, K., Kizun, O., & Skublewska-Paszowska, M. (2020). Performance analysis of relational databases MySQL, PostgreSQL, MariaDB and H2. *Journal of Computer Sciences Institute*, 14, 1–7. <https://doi.org/10.35784/jcsi.1565>.
50. Lu Q, Li Z, Zhang W, Yang LT. 2017. Autonomic deployment decision making for big data analytics applications in the cloud. *Soft Computing* 21 (16): 4501–4512 DOI 10.11/s00500-015-1945-5.
51. Manyika J. Big Data: The next frontier for innovation, competition, and productivity. *McKinsey Global Institute*, June, 2011. McKinsey, 2011. P. 27–31.
52. Mazumder S. 2016. Big data tools, platforms. In: Big data concepts, and theories, and applications. Cham: Springer, 29–128.

53. Mittal S, Sangwan OP. 2019. Big data analytics using machine learning techniques. In: 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence). Piscataway: IEEE.
54. Momento. URL: <http://celsiair.com> (дата звернення: 20.04.2023)
55. Murach's Java Servlets and HTML (2nd Edition) by J. Murach, A. Steelman. 2019. 844p.
56. MySQL is the Database of the Year. January 2020. URL: [https://db-engines.com/en/blog\\_post/83](https://db-engines.com/en/blog_post/83) (дата звернення: 20.04.2023).
57. Nicolalde FC, Silva F, Herrera B, Pereira A. 2018. Big data analytics in IOT: challenges, open research issues and tools. In: World conference on information systems and technologies. Cham: Springer.
58. Oussous A, Benjelloun F-Z, Lahcen A, Belfkih S. 2018. Big data technologies: a survey. *Journal of King Saud University-Computer and Information Sciences* 30 (4): 431–448 DOI 10.1016/j.jksuci.2017.06.001.
59. Parsaye K. OLAP and Data Mining: Bridging the Gap. *Database Programming and Design*. 2003. № 2.
60. Patgiri R, Ahmed A. 2016. Big data: The v's of the game changer paradigm. In: 2016 IEEE 18th international conference on high performance computing and communications; IEEE 14th international conference on smart city; IEEE 2nd international conference on data science and systems (HPCC / SmartCity / DSS). Piscataway: IEEE.
61. Pulgar-Rubio F, Rivera-Rivas AJ, Pérez-Godoy MD, González P, Carmona CJ, del Jesus MJ. 2017. MEFASD-BD: multi-objective evolutionary fuzzy algorithm for subgroup discovery in big data environments-a mapreduce solution. *Knowledge-Based Systems* 117:70–78 DOI 10.1016/j.knosys.2016.08.021.
62. Suntouch. URL: <http://www.suntouch.com> (дата звернення: 20.04.2023)
63. UML – діаграми взаємодії. URL: [http://www.w3eacademy.com/uml/uml\\_interaction\\_diagram.htm](http://www.w3eacademy.com/uml/uml_interaction_diagram.htm) (дата звернення: 20.04.2023)

64. Yun D, Wu CQ, Rao NS, Kettimuthu R. 2019. Advising big data transfer over dedicated connections based on profiling optimization. *IEEE / ACM Transactions on Networking* 27 (6): 2280–2293.

# ДОДАТОК А



**ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ**  
**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ**  
**ТЕХНОЛОГІЙ**  
**КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**



## **Розробка веб-сервісу для підтримки організаційних процесів дистанційного навчання з використанням бібліотеки React.JS**

Виконав студент 4 курсу  
групи ПД-43  
Виговський Богдан Сергійович  
Керівник роботи  
Асистент кафедри, аспірант Залива Віталій Вікторович

Київ – 2023



## МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** - створення веб-ресурсу для організації підтримки організаційних процесів дистанційного навчання з використанням бібліотеки React.js.
- **Об'єкт дослідження** - програмна реалізація систем доступу до даних та інформаційних систем.
- **Предмет дослідження** - проєкт інформаційної системи навчального ресурсу.







# ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Визначити актуальність створення веб-сервісу для підтримки дистанційної освіти.
2. Виконати вибір сучасних інструментів та аналіз методологій для розробки веб-сервісів та баз даних.
3. Провести дослідження та аналіз цільової аудиторії продукту.
4. Проаналізувати існуючі аналоги на ринку.
5. Виконати моделювання та проєктування процесу розробки веб-сервісу.
6. Розробити базу даних ресурсу.
7. Провести реалізацію веб-сервісу та його дизайну.



# АНАЛІЗ АНАЛОГІВ

Властивості	Schedule.kpi	КНУТД	DII (розроблено)
Простий у використанні інтерфейс	-	-	+
Середовище використання	Веб-сайт	Веб-сайт	Веб-сайт
Сучасність дизайну	-	-	+

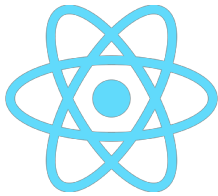


## ВИМОГИ ДО ДОДАТКУ

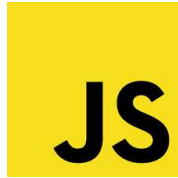
1. Адаптивність під пристрої з різними розширеннями.
2. Вільний перегляд інформації з завданням на головній сторінці.
3. Інформація про ресурс підтримки організаційних процесів дистанційного навчання.
4. Онлайн створення послуг користувачами.
5. Стрічка завдань.



## ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



ReactJS



JavaScript

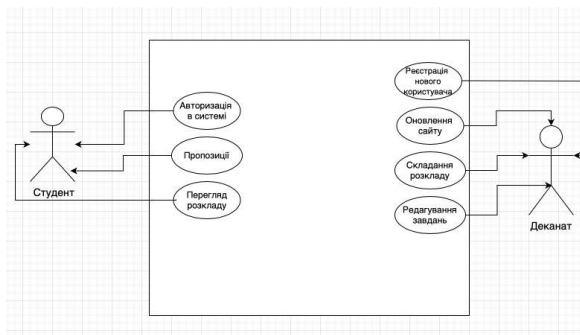


HTML5

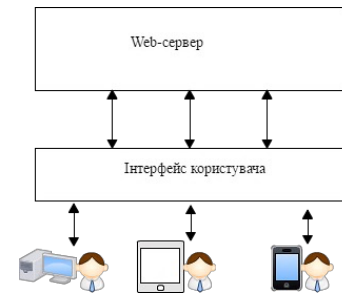


CSS

### UML-ДІАГРАМА

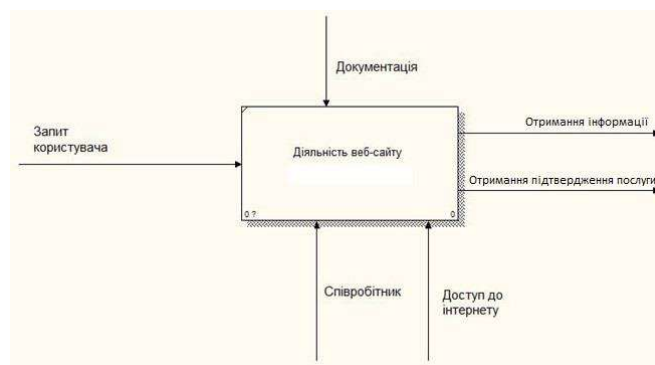


### АРХІТЕКТУРА ІНФОРМАЦІЙНОЇ СИСТЕМИ

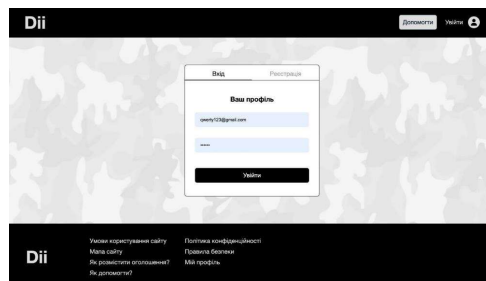




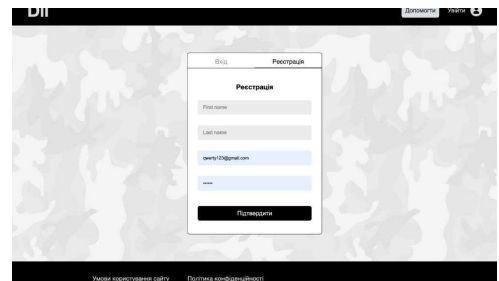
**ДІАГРАМА IDEF0(МЕТОДОЛОГІЯ ФУНКЦІОНАЛЬНОГО МОДЕЛЮВАННЯ)**



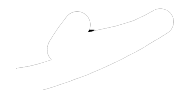
## ЕКРАННІ ФОРМИ



**Форма входу користувача в персональний кабінет**



**Форма реєстрації нового користувача**



## ЕКРАННІ ФОРМИ

The screenshot shows the 'Профіль' (Profile) page in the Dii application. It features a dark header with the 'Dii' logo and navigation links for 'Домашня' (Home) and 'Профіль' (Profile). The main content area is divided into three columns: 'Логін' (Login) with a 'Log out' button, 'Зміна даних користувача' (Change user data) with fields for 'Ім'я' (Name), 'Прізвище' (Surname), and 'Ім'я користувача' (Username), and 'Зміна паролів' (Change passwords) with fields for 'Старий пароль' (Old password), 'Новий пароль' (New password), and 'Підтвердіть' (Confirm). A 'Зберегти' (Save) button is at the bottom right. The footer contains the 'Dii' logo and links for 'Умови користування сайту' (Site usage conditions), 'Політика конфідційності' (Privacy policy), 'Мова сайту' (Site language), 'Телефон' (Phone), 'Як розмістити оголошення?' (How to place an advertisement?), and 'Мій профіль' (My profile).

Сторінка редагування персональних даних користувача

The screenshot shows the 'Редагування завдання' (Task editing) page in the Dii application. It features a dark header with the 'Dii' logo and navigation links for 'Домашня' (Home) and 'Профіль' (Profile). The main content area is divided into two columns: 'Завдання' (Task) with fields for 'Назва' (Name), 'Клас завдання' (Task class), 'Висновок завдання' (Task conclusion), 'Як виконувати завдання' (How to perform the task), and 'Важливість завдання' (Task importance), and 'Деталі' (Details) with a table. The table has columns for 'Назва' (Name), 'Статус' (Status), 'Адреса' (Address), 'Бачити' (View), and 'Деталі' (Details). The first row shows a task named 'Легко зробити проект на тему: "Вплив війни на економіку України"', with a status of '100%', and 'Edit' and 'Details' buttons. A 'Зберегти' (Save) button is at the bottom center. The footer contains the 'Dii' logo and links for 'Умови користування сайту' (Site usage conditions), 'Політика конфідційності' (Privacy policy), 'Мова сайту' (Site language), and 'Профіль' (Profile).

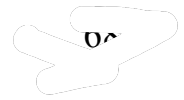
Редагування завдання





## АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

1. Виговський Б.С. Мова програмування для написання web-платформи для українського дистанційного навчання/ Б.С. Виговський, В.В. Залива// Перший крок у науку: Конотопські наукові студії – 2023. Подано до друку.
2. Виговський Б.С. Хмарні технології/ Б.С. Виговський, В.В. Залива// Застосування програмного забезпечення в інфокомунікаційних технологіях. Збірник тез 20.04.23 ДУТ, м.Київ – К.: ДУТ, 2023



# ВИСНОВКИ

1. Метою дипломного проєкту є створення інформаційної системи навчального порталу. Головною проблемою існуючих аналогів є недостатня обізнаність користувача про сам портал. Головним елементом інформаційної системи є простота, зручне отримання вичерпної інформації з завдання для навчального процесу та можливість дізнатися детально про портал.
2. Предметна область проєктування – інформаційна система по організації підтримки організаційних процесів дистанційного навчання. Система створена для того, щоб допомогти користувачам всевітнього Інтернет порталу підтримки організаційних процесів дистанційного навчання знайти необхідне завдання, вирішити його та отримати довідку. Відповідно, під час перегляду завдання користувач повинен бачити: короткий опис послуг з посиланням на повний опис на сторінці з відомостями.
3. Система може надати можливість користувачам проходити в деякому значенні «анкети», за підсумками якого можна буде зробити висновки щодо необхідності даного завдання, його параметрів та характеристик. Інформаційна модель розв'язання задачі створення та розміщення завдань на проєктованому ресурсі підтримки організаційних процесів дистанційного навчання призначена для графічного зображення вхідних інформаційних потоків в систему та вихідних потоків даних. Вхідними потоками є юридична інформації для діяльності системи та дані про користувача, про завдання та дані про майданчик розміщення завдань для кожного користувача. Вихідними потоками даних слугують всі дані завдань.
4. У проєктованого сервісу інформаційного забезпечення підтримки організаційних процесів дистанційного навчання реалізований зручний користувацький інтерфейс та наступні можливості для користувачів: по натиску на кнопку «Додати» користувач, що увійшов у систему може додати текстово-графічний об'єкт завдання, змінити його параметри, опис та інші дані.
5. Сайт проєктованої інформаційної системи був написаний за допомогою HTML5, Scss, JavaScript та React.js. Результатом роботи є: дослідження актуальності питання; аналіз предметної області; аналіз потреб користувачів; проєктування інформаційної системи у; тестування продукту; розробка технічного завдання.

## ДОДАТОК Б

### Додаток А

#### Програмний код головної сторінки проєктованого ресурсу

```
<!DOCTYPE html>
<html lang=«en»>
<head>
<meta charset=«utf - 8» />
<title>VolunteeringFrontend</title>
<base href="/" />
<meta name=«viewport» content=«width=device-width, initial-
scale=1» />
<link rel=«icon» type=«image / x-icon» href=«favicon. ico» />
</head>
<body class=«dark-theme»>
<app-root></app-root>
<script src=«https: // kit. fontawesome. com / 5953287ccc. js»
crossorigin=«anonymous»></script>
</body>
</html>
```

### Додаток Б

#### Програмний код файлу main. ts

```
import { enableProdMode } from '@React.js / core';
import { platformBrowserDynamic } from '@React.js / platform-
browser-dynamic';
import { AppModule } from './ app / app. module';
import { environment } from './ environments / environment';
if (environment. production) {
enableProdMode();
}
platformBrowserDynamic(). bootstrapModule (AppModule)
.catch (err => console. error (err));
```

### Додаток В

#### Програмний код файлу app. scss

```
@import». / ui / reset. scss»;
@import». / themes / light. scss»;
@import». / themes / dark. scss»;
@import». / ui / ui. scss»;
@import». / ui / form. scss»;
@import './ ui / button. scss';
html,
body {
min-height: 100%;
color: var(-text-color);
background: url('../ assets / images / main_backgrounda ');
}
body {
```

```
margin: 0;
font-family: Roboto, «Helvetica Neue», sans-serif;
}
```

## Додаток Г

### Програмний код файлу теми dark.scss

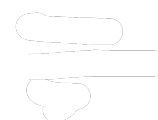
```
body. dark-theme {
  -text-color: #dddddd;
  -background-color: #4e4e4e;
  -header-bg-color: #2e2e2e;
  -primary: #2e2e2e;
  -warning-color: #e68a00;
  -danger-color: #e12424;
  -error-color: #c11e1e;
  -color-grey: #ececec;
```

## Додаток Д

### Програмний код файлу форми form.scss

```
.form__wrapper {
  min-width: 400px;
  padding-bottom: 40px;
  background-color: #fff;
  border: 1px solid #000;
  border-radius: 8px;
  .tabs {
    display: flex;
    .tab {
      cursor: pointer;
      width: 100%;
      padding: 15px 0 7px;
      font-size: 18px;
      text-align: center;
      color: #00000066;
      border-bottom: 1px solid #000;
      &.active {
        color: #000;
        border-bottom: 2px solid #000;
      }
    }
  }
}

form. form {
  padding: 0 30px;
  > * {
    width: 100%;
  }
  h2. form__title {
    margin: 43px 0 27px;
```



```

text-align: center;
font-size: 20px;
color: #000;
}
.form-field {
&:not(:last-of-type) {
margin-bottom: 32px;
}
}
textarea. form__element {
resize: none;
margin-bottom: - 4px;
min-height: 79px;
font-family: Roboto, «Helvetica Neue», sans-serif;
}
.form__element {
width: 100%;
height: 43px;
padding: 13px 16px;
outline: none;
border: none;
border-radius: 6px;
background-color: var(-color-grey);
&:: placeholder {
font-size: 15px;
}
}
button [type=«submit»]. form__element {
cursor: pointer;
margin-top: 46px;
padding: 11px 0;
font-size: 18px;
background-color: #000;
color: #fff;
}
}
.form__label {
margin-bottom: 7px;
font-size: 22px;
}
}

```

## Додаток Е

### Програмний код файлу організації кнопки `button.scss`

```

.button {
cursor: pointer;
width: 100%;
padding: 10px 20px;
outline: none;
border: none;
border-radius: 5px;
color: #fff;
}

```

```

&.color-warning {
background-color: var(-warning-color);
}
&.color-danger{
background-color: var(-error-color);
}
}

```

## Додаток Ж

### Програмний код файлу організації дизайну ui. scss

```

.container {
width: 1100px;
margin: 40px auto 0;
}
.button {
padding: 10px;
font-size: 18px;
color: #ddd;
border: none;
border-radius: 6px;
&.button_dark {
background-color: #000;
}
}
.is-loading {
display: inline-flex;
align-items: center;
justify-content: center;
font-size: 0;
overflow: hidden;
pointer-events: none;
&:before {
content: "»";
position: absolute;
width: 15px;
height: 15px;
border: 3px solid rgba (93, 93, 93, 0.6);
border-left-color: transparent;
border-radius: 50%;
animation: 1.3s loader infinite linear;
}
}
@keyframes loader {
to {
transform: rotate (360deg);
}
}

```

