

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук,  
управління та адміністрування  
Кафедра інформаційних технологій

**Бакалаврська кваліфікаційна робота**

на тему: Розробка клієнтської частини Веб-порталу

Виконав студент 4 курсу групи К-41  
Спеціальність 122 комп'ютерні науки  
Обуховський Ілля Юрійович

Керівник \_\_\_\_\_ к.ф.-м.н., доцент  
Козловська Валентина Петрівна

Консультант \_\_\_\_\_  
\_\_\_\_\_

Рецензент: \_\_\_\_\_ к.т.н., доцент  
Великодний Станіслав Сергійович

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук, управління  
та адміністрування  
Кафедра інформаційних технологій

**Комплексна бакалаврська кваліфікаційна робота**

на тему: «Розробка Веб-порталу факультету КНУА ОДЕКУ»

Склад:

**Частина 1** «Розробка серверної частини Веб-порталу»

Виконавець: Ільчов Дмитро Ігорович

Керівник к.ф.-м.н., доцент

Козловська Валентина Петрівна

**Частина 2** «Розробка клієнтської частини Веб-порталу»

Виконавець: Обуховський Ілля Юрійович

Керівник к.ф.-м.н., доцент

Козловська Валентина Петрівна

Староста роботи: Чабанюк Віктор Іванович

Провідний керівник проекту: к.ф.-м.н., доцент Козловська Валентина Петрівна

Рецензент: к.т.н., доцент Великодний Станіслав Сергійович

МІНІСТЕРСТВО ОСВІТИ І НАУКИ  
ОДЕСЬКИЙ ДЕРЖАВНИЙ ЕКОЛОГІЧНИЙ УНІВЕРСИТЕТ

Факультет Комп'ютерних наук, управління та адміністрування  
Кафедра Інформаційних технологій  
Рівень вищої освіти бакалавр  
Спеціальність 122 комп'ютерні науки  
(шифр і назва)

**ЗАТВЕРДЖУЮ**  
Завідувач кафедри \_\_\_\_\_

“ 13 ” квітня 2020 року \_\_\_\_\_

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

\_\_\_\_\_ Обуховському Іллі Юрійовичу  
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка Веб-порталу факультету КНУА ОДЕКУ  
Ч.2 Розробка клієнтської частини Веб-порталу

керівник роботи Козловська Валентина Петрівна, к.ф.-м.н, доцент  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “ 13 ” квітня 2020 року № 37-С

2. Строк подання студентом роботи 30.05.2020

3. Вихідні дані до роботи Технічні вимоги до роботи Веб-порталу  
Зовнішні представлення на ІС груп користувачів Веб-порталу

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) У вступі викладаються мета та задачі дипломної роботи  
У перших розділах проводиться опис та аналіз предметної області, виконується її моделювання

Наводиться огляд та вибір існуючих засобів розробки програмного продукту  
Описується розроблений програмний продукт та наводиться посібник користувача цього продукту

У висновках підводяться підсумки виконаної роботи

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

## 6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання 13 квітня 2020 року**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Оцінка виконання етапу	
			у %	за 4-х бальною шкалою
1	Огляд та аналіз предметної області	13.04- 20.04		
2	Моделювання системи	21.04- 25.04		
3	Вибір засобів розробки Веб-порталу	26.04- 01.05		
4	Проектування сайту	02.05- 10.05		
	Рубіжна атестація	11.05- 16.05		
5	Розробка клієнтської частини Веб-порталу	17.05- 21.05		
6	Оформлення пояснювальної записки	22.05- 29.06		
7	Перевірка дипломної роботи на оригінальність	30.05		
	<b>Інтегральна оцінка виконання етапів календарного плану (як середня по етапам)</b>			

Студент \_\_\_\_\_ Обуховський І.Ю.  
( підпис ) (прізвище та ініціали)Керівник роботи \_\_\_\_\_ Козловська В.П.  
( підпис ) (прізвище та ініціали)

## ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ.....	7
ВСТУП .....	8
1 ПРЕДМЕТНА ОБЛАСТЬ ТА ПОСТАНОВКА ЗАДАЧІ .....	9
1.1 Класифікація веб-ресурсів.....	9
1.2. Аналіз цільової аудиторії веб-ресурсу.....	11
1.3 Аналіз існуючих аналогів.....	12
1.4 Постановка задачі.....	15
2 МОДЕЛЮВАННЯ КЛІЄНТСЬКОЇ ЧАСТИНИ САЙТУ КНУА.....	17
2.1 Визначення груп користувачів та сценаріїв їх роботі з ресурсом .....	18
3 ВИБІР ПРОГРАМНИХ ЗАСОБІВ.....	20
3.1 Фреймворки CSS .....	20
3.1.1 Опис фреймворку Bootstrap .....	20
3.1.2 Опис фреймворку Foundation.....	21
3.1.3 Опис фреймворку Bulma .....	22
3.2 Мультипарадигмна мова програмування Java Script.....	23
3.3 JS бібліотека JQuery .....	24
3.3.1 Опис бібліотеки JQuery .....	25
3.3 Мова гіпертекстової розмітки документа HTML .....	26
3.4 Мова каскадних стилів CSS .....	27
3.5 Вибір середовища розробки.....	28
3.5.1 Visual Studio Code .....	29
3.6 Структура веб-порталу .....	30
3.7 Вибір системи для управління версіями проекту .....	33
3.7.1 Версіонування.....	33
3.7.2 Git.....	34
3.7.3 GitHub .....	35
4 ПРОЄКТНІ РІШЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ.....	37
4.1 Розробка дизайну веб-порталу.....	37
4.1.1 Основна область сайту.....	37
4.1.2 Основна область сайту.....	38
4.1.4 Футер .....	40

4.1.5 Фон сайту .....	40
4.2 Опис інтерфейсу користувача .....	41
ВИСНОВКИ.....	46
ПЕРЕЛІК ПОСИЛАНЬ .....	48
ДОДАТОК А.....	49

## ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ І ТЕРМІНІВ

БД – база даних

ІС – інформаційна система

ОС – операційна система

ПЗ – програмне забезпечення

AJAX – Asynchronous JavaScript And XML

API – Application Programming Interfaces

DOM – Document Object Model

CSS – Cascading Style Sheets – каскадні таблиці стилів

HTML – HyperText Markup Language – мова гіпертекстової розмітки

Framework – платформа, яка визначає структуру програмної системи.

JSON – JavaScript Object Notation

SASS – Syntactically Awesome Stylesheets

SPA – Single Page Application

SQL – Structured Query Language MEAN – MongoDB, Express, Angular,

UI – User Interface

UX – User experience design

## ВСТУП

В результаті розвитку Веб-технологій комп'ютер став інструментом інформаційної підтримки. Великий пласт соціальних груп, які підключаються до мережі Інтернет та шукають там інформацію, весь час розширюється, оскільки інформація стає доступною людям, та наявність зручного, сучасного Веб-сайту є необхідністю, як для малих, так і для великих організацій.

Метою цього дипломного проекту є розробка клієнтської частини Веб-порталу для факультету КНУА ОДЕКУ.

Frontend (фронтенд або клієнтська частина програми) виконується в браузері користувача. Ця частина написана на мові програмування Javascript, мова гіпертекстової розмітки HTML, каскадні таблиці стилів CSS.

Створення Веб-порталу для факультету дозволить студентам ознайомитися, та отримати необхідну інформацію. Велика кількість корисної інформації про навчальний заклад, зможе знайти абітурієнтів, які бажають вступити на навчання у закладі, або студентів, які цікавляться життям кафедри.



# 1 ПРЕДМЕТНА ОБЛАСТЬ ТА ПОСТАНОВКА ЗАДАЧИ

## 1.1 Класифікація веб-ресурсів.

Перш ніж приступити безпосередньо до створення сайту, потрібно визначитися з його типом. Нижче приведена зразкова класифікація інтернет-сайтів, умовно розбита на групи. На практиці сайти дуже часто є комбінованими, і можуть бути приписані до різних видів. Проте, пропонована класифікація сайтів буде корисна для розуміння того, які типи сайтів існують. Вид сайту практично завжди визначається його цілями і завданнями<sup>[1]</sup>.



Рисунок 1.1 – Класифікація веб-ресурсів

Сайт візитка - найпростіший вид сайту. Сайт такого типу будується на простому HTML, без використання системи управління сайтом. Зазвичай сайт-візитка містить від 1 до 5 сторінок. Сайти цього виду як правило включають в себе тільки загальну інформацію про власника сайту і його контактні дані.

<sup>1)</sup> [1] Браун Бредлі, Річард Дж. Ніемік, Джозеф С. Трезза Oracle Database. Створення Web-додатків: - Санкт-Петербург, Лорі, 2007 р.- 722 с.

Простота розробки такого виду сайту робить вартість його створення порівняно дешевою, що є очевидною перевагою для замовника.

Корпоративні сайти - це повнофункціональні представництва компаній в інтернеті. Цей тип сайту найкраще підходить для серйозних середніх і великих фірм. Корпоративні сайти містять повну інформацію про компанію та її діяльність. Такий тип сайту іноді називають віртуальним офісом, так як відвідування такого сайту можна порівняти зі спілкуванням з менеджером по роботі з клієнтами.

Корпоративні сайти потрібні, в першу чергу, для формування іміджу компанії і надання відвідувачам і клієнтам найбільш повної інформації.

Інтернет-магазин або інтернет-каталог товарів - це вид сайтів, основне завдання яких - продавати. На таких сайтах розміщується інформація про товари і контакти, зазвичай телефони, за якими слід дзвонити бажаним придбати пропонований товар. На таких сайтах розміщуються технічні характеристики товарів, відгуки, рекомендації експертів і т.д., а також є можливість замовити пропонований товар прямо через сайт.

Промо-сайти призначені для розкрутки і просування будь-якого товару або бренду.

Тематичні сайти характеризуються тим, що містять інформацію з будь-якої конкретної тематики. Сюди ж можна віднести інтернет-енциклопедії.

Портали - це тип сайтів, що містять велику кількість різноманітної інформації. Як правило, портали схожі за структурою з тематичними сайтами, але мають більш розвинений функціонал і більшу кількість сервісів і розділів. Також на порталах часто бувають розділи для спілкування користувачів: чати, блоги і форуми.

Блог - це тип сайтів, на яких власник або редактор блогу пише пости зі своїми новинами, ідеями або інший постійно надходить інформацією. Відмінною особливістю блогів є актуальність інформації, що публікується інформації

## 1.2. Аналіз цільової аудиторії веб-ресурсу

Аналіз цільової аудиторії сайту - це один з етапів створення структури сайту. Даний етап відрізняється тим, що його виконання здійснюється не тільки на стадії створення структури сайту, але і по закінченню створення сайту, тобто процес аналізу цільової аудиторії можна вважати процесом безперервним<sup>[2]</sup>.

Однак на попередньому етапі на відміну від наступних етапів, аналіз складається, з ймовірними параметрами. Тобто вебмастер планує майбутню цільову аудиторію сайту на своїх припущеннях і проведених дослідженнях.

Для більш точного визначення і аналізу цільової аудиторії сайту слід виконати ряд заходів і провести наступні дослідження:

- Проаналізувати передбачуваний вік цільової аудиторії. Вік цільової аудиторії безпосередньо залежить від обраної теми сайту. Тому буде дуже корисно використовувати проведений аналіз сайтів зі схожою темою. Варто уважно і в повній мірі провести аналіз цільової аудиторії за віковими параметрами.
- Проаналізувати географічну приналежність цільової аудиторії. Якщо розбирати вищевказане вплив географічної приналежності цільової аудиторії, то тут варто зупинитися на мовному факторі. Це дозволить заздалегідь передбачити можливість надання інформації і на інших мовах.
- Проаналізувати статеву приналежність цільової аудиторії. Даний фактор, так само визначається тематикою сайту. Однак вплив теми, не завжди виявляється вирішальним. Тому аналіз статі передбачуваних відвідувачів обов'язково проводиться при складанні структури сайту. Одним із способів провести такий аналіз може бути банальне звернення за статистикою до адміністрації сайтів схожої тематики. Аналіз цільової аудиторії за статевою ознакою, робить свій вплив на дизайн

---

<sup>2)</sup> [2] Браун Д. Розробка веб-сайту. Взаємодія з замовником, дизайнером і програмістом: - Санкт-Петербург, Пітер 2009 р.- 336 с.

сайту, на спосіб надання інформації, на додаткові матеріали та посилання, які безсумнівно з'являться в процесі розвитку сайту.

- Проаналізувати матеріальний достаток передбачуваної цільової аудиторії. Даний фактор розглядається в основному для комерційних сайтів і впливає на перелік наданих товарів і послуг, на спосіб надання інформації про фірму або компанії, на цінову політику сайту. Використовуючи даний аналіз при розробці структури сайту, можна врахувати рекламне навантаження на сайт, мобільність зміни інформації на сайті, а також безпосередньо передбачувано розміри сайту і параметри хостингу для нього
- Таким чином, слід виділити фактори за якими буде проводитися аналіз веб-ресурсу:
- Віковий показник (від 14 до 18, від 18 до 25, від 25 до 40, від 40 до 55, старше ...);
- Статеву приналежність (чоловік, жінка);
- Рівень освіти (середнє, слухач, особливе, вища);
- Сферу зайнятості (держпідприємство, приватна фірма, підприємець);
- Матеріальний достаток (високий, низький, середній);
- Періодичність користування інтернетом (кожен день, раз на тиждень, на місяць);
- Доступ до інтернету (домашній інтернет, на роботі, в інтернет кафе).

Веб-портал КНУА буде цікавим для потенціальних абітурієнтів та студентів, які вже вступили до цієї спеціальності. Більша частина користувачів матиме середню освіту. Періодичність користування буде епізодичною, в загальному для перегляду новин, або іншої необхідної інформації.

### **1.3 Аналіз існуючих аналогів**

На сьогоднішній день майже кожний навчальний заклад має власний веб-сайт, проте більшість побудовано на застарілих технологіях, не адаптивні та

мають застарілий дизайн. Як приклад можна взяти сайт Київський національний університет імені Тараса Шевченка (рис 1.2).



Рисунок 1.2 – Сайт КНУ імені Тараса Шевченка

На перший погляд сайт не має явних недоліків, проте на малих екранах я у смартфонів, можливо побачити відсутність адаптивності, що робить цей сайт менш зручним (рис 1.3). Маленький розмір навігаційного меню, та дрібний шрифт може стати перешкодою для людей з поганим зором.

Також треба відмітити відсутність посилань соціальні мережі, розміщення занадто великої кількості інформації на одній сторінці, застарілий дизайн.

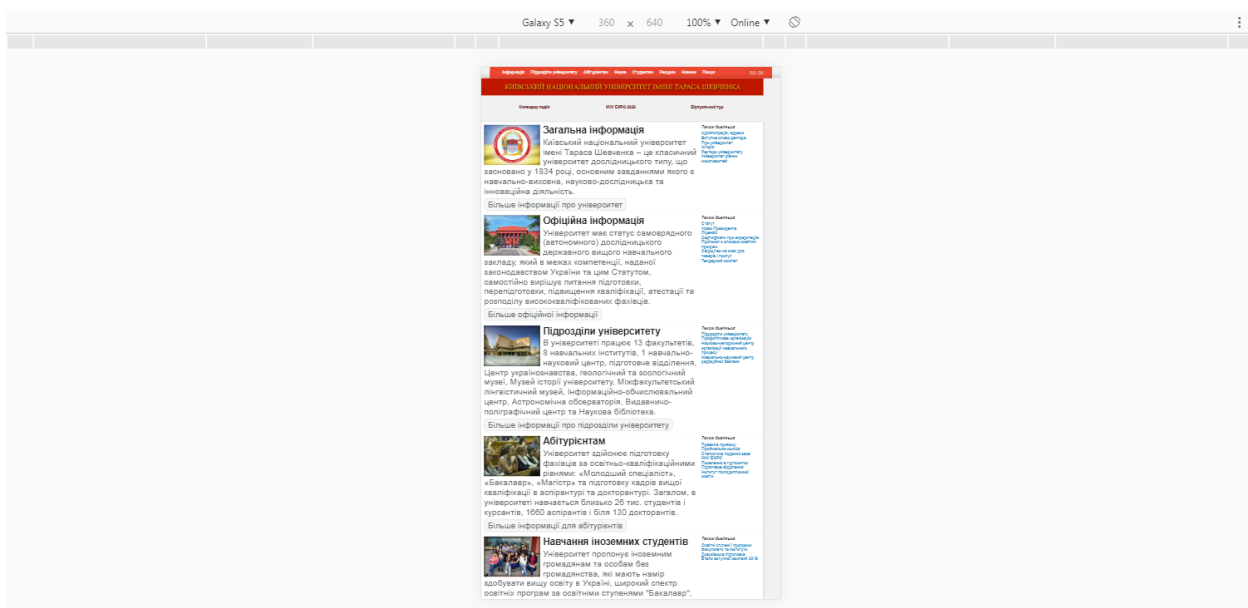


Рисунок 1.3 – Мобільна версія сайту КНУ імені Тараса Шевченка

Іншим прикладом є Талліннський університет (рис 1.4). Цей приклад має більш сучасний дизайн, має посилання на соціальні мережі, зручне навігаційне меню.

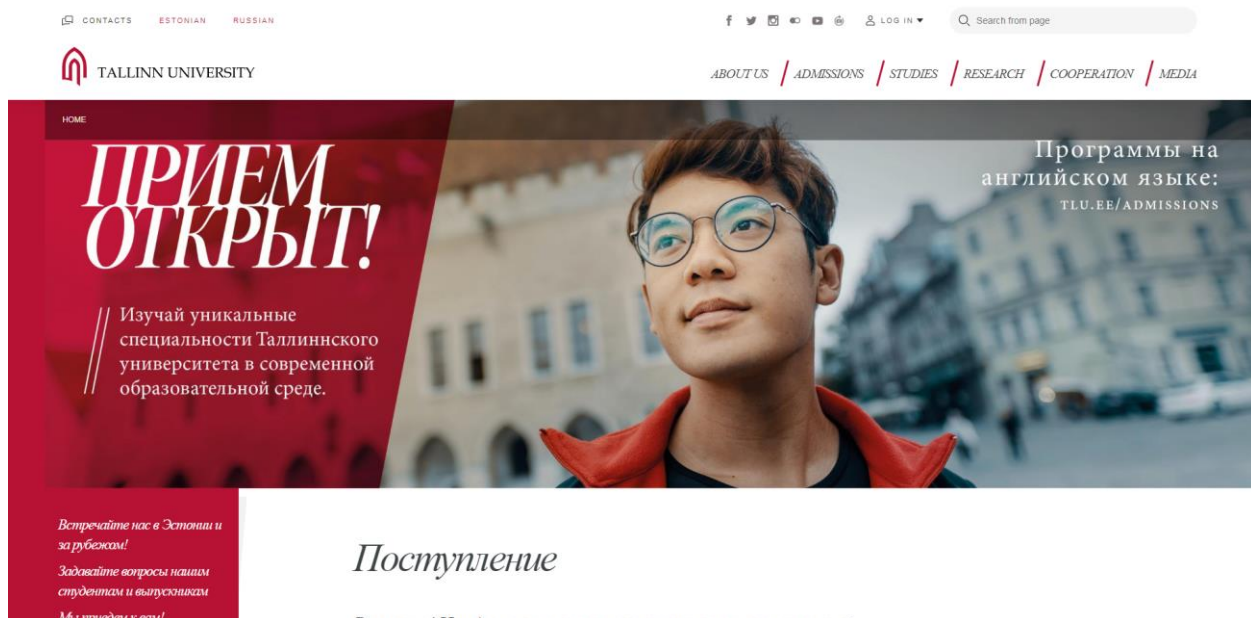


Рисунок 1.4 – Сайт Талліннського університету

Також цей сайт адаптується під малі розширення, що є великим плюсом у порівнянні з іншим аналогом (рис 1.5). Велике навігаційне меню сховано у меню "бургер", що дозволяє зберегти корисний простір під решту корисного контенту. У додаток, таке рішення залишає навігацію зручною, чіткою, без шкоди для решти сторінки.

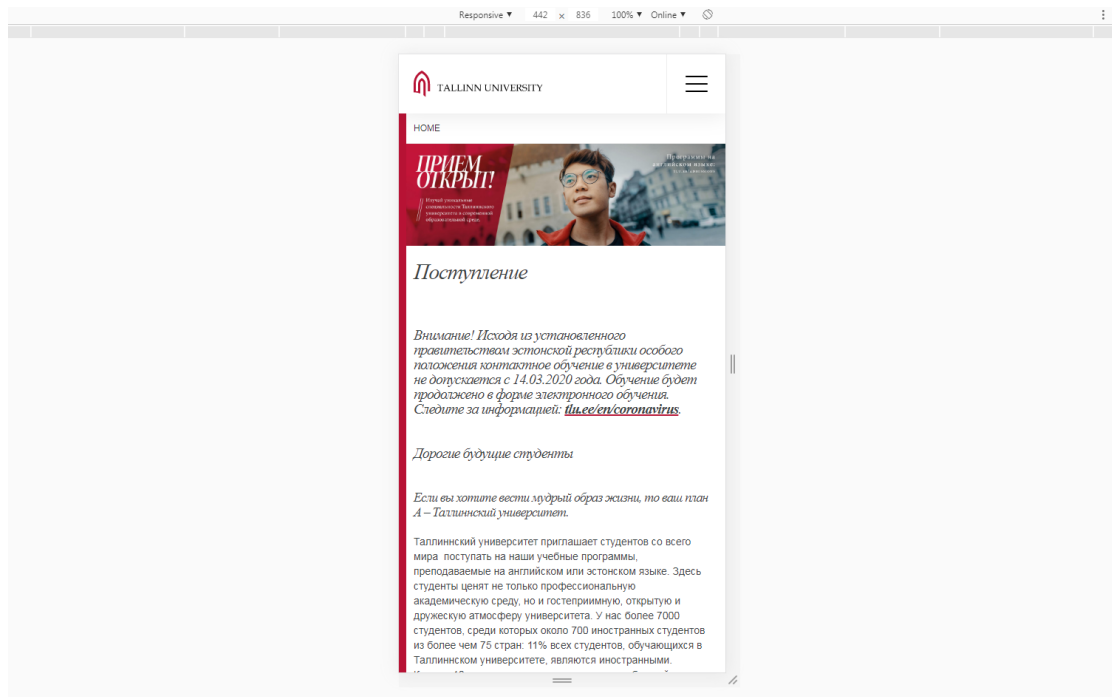


Рисунок 1.5 – Мобільна версія сайту Талліннського університету

## 1.4 Постановка задачі

Метою дипломної роботи є придбання теоретичних знань і практичних умінь з розробки клієнтської частини веб-сайтів із застосуванням Bootstrap та бібліотеки JQuery.

Для реалізації мети дипломної роботи необхідно вирішити наступні завдання:

- вивчити теоретичні складові створення Web-сайтів.
- провести аналіз ПЗ для створення Web-сайтів.
- розглянути мови програмування, що використовуються в Web технологіях.

- провести аналіз засобів розробки клієнтської частини для Web-сайтів.
- створити структуру клієнтської частини.
- створити адаптивний дизайн сторінок веб-сайту.
- провести аналіз працездатності створеного сайту.
- результатом виконання поставлених завдань повинен стати веб-портал факультету КНУА ОДЕКУ.

У технічному завданні повинні бути відображені наступні вимоги:

- Назва сайту - «Інформаційний портал КНУА»
- Призначення і цілі створення сайту - Сайт призначений для поширення інформації про факультет.
- Цільова аудиторія сайту - Цільову аудиторію сайту, складають користувачі, які бажають отримати інформацію про спеціальність до якої бажають поступити.
- Вимоги до системи адміністрування:
- Редагування інформації - Редагування інформації повинно проводитися за адміністрації.
- Мовні версії сайту - Сайт призначений для українськомовних користувачів, тому повинен містити одну мовну версію.
- Дизайн сайту - Дизайн сайту повинен відповідати сучасним стандартам і бути зручний для користувача.
- Структура сайту - Структура сайту повинна являти собою ієрархію, засновану на логічних зв'язках між його компонентами.
- Навігація по сайту:
- Основне навігаційне меню - Сайт повинен містити основне навігаційне меню.
- Функціонал сайту - Сайт повинен містити модулі реєстрації, авторизації.
- Контент і наповнення сайту - Сайт повинен містити:



- Інформацію про факультет та викладачів.
- Новини кафедри, досягнення.
- Загальну інформацію про навчання

## 2 МОДЕЛЮВАННЯ КЛІЄНТСЬКОЇ ЧАСТИНИ САЙТУ КНУА

Для розробки моделі клієнтської частини потрібно спочатку визначити групи користувачів, що будуть працювати з нею. Потрібно виявити, у яких випадках користувачі різних груп будуть звертатись до даної інформаційного ресурсу, та за якими сценаріями вони будуть з нею працювати.

### 2.1 Визначення груп користувачів та сценаріїв їх роботі з ресурсом

Основною групою користувачів веб-порталу кафедри КНУА будуть студенти з викладачами закладу. Крім них цей ресурс можуть використовувати люди, які зацікавлені у цьому закладі, з метою вступу, або ознайомлення.

У групи користувачів, які прийшли на сайт з метою ознайомлення, повинні бути такі варіанти використання як:

- перегляд загальної інформації стосовно факультету
- перегляд інформації стосовно кафедри
- перегляд інформації о досягненнях
- перегляд новин
- У групи «студентів» повинні бути такі варіанти:
- перегляд загальної інформації стосовно факультету
- перегляд інформації стосовно кафедри
- перегляд інформації о досягненнях
- перегляд новин
- можливість ознайомлення з розкладом занять
- можливість залишати коментарі стосовно новин
- У групи «викладачів» повинні бути такі варіанти:
- перегляд загальної інформації стосовно факультету
- перегляд інформації стосовно кафедри
- перегляд інформації о досягненнях
- перегляд новин
- можливість ознайомлення з розкладом занять

- можливість залишати коментарі стосовно новин
- можливість залишати інформацію о собі
- можливість робити оголошення для студентів

Як видно з порівняння наведених варіантів використання веб-порталу вказаними групами користувачів, групи мають багато спільних варіант використання. Таким чином ці варіанти використання повинні відноситись до загального користувача, а перераховані групи користувачів є спадкоємцями цього загального користувача, що мають додаткові варіанти використання системи.

На рис. 2.1 наведена діаграма варіантів використання (use case diagram) веб-порталу для груп користувачів.

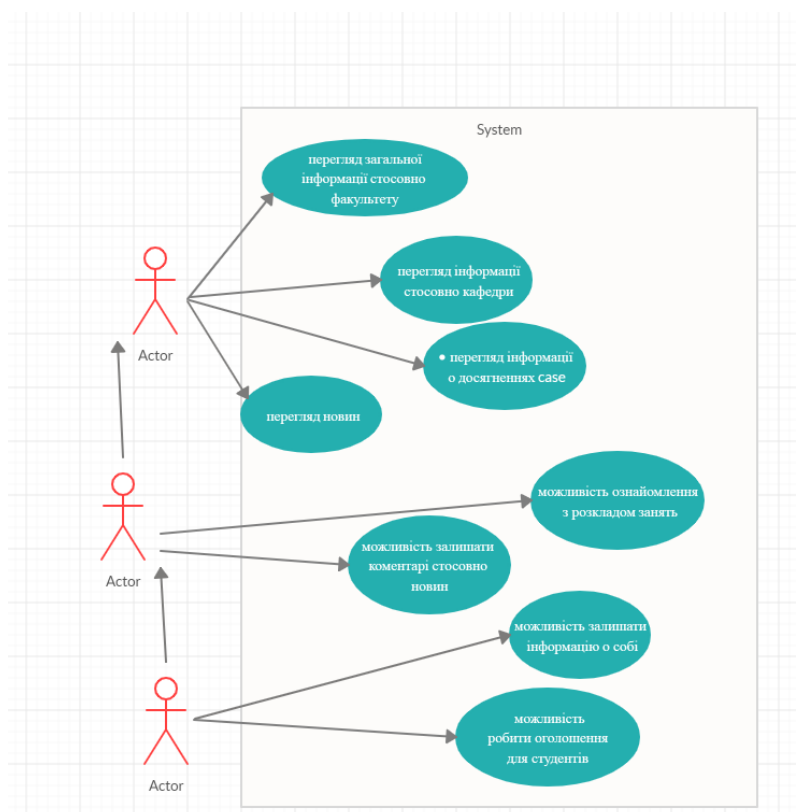


Рисунок 2.1 – Діаграма варіантів використання

## 3 ВИБІР ПРОГРАМНИХ ЗАСОБІВ

### 3.1 Фреймворки CSS

CSS Framework надає веб-розробникам базову структуру, яка включає сітку, інтерактивні схеми інтерфейсу користувача, типографіку веб-сторінок, підказки, кнопки, елементи форми. Ця структура допомагає веб-розробникам швидко та ефективно почати розробку веб-сайту або веб-додатку. Це означає, що розробникам не потрібно починати все з нуля. Структура CSS створить для них міцну основу. Крім того, розробники також можуть повторно використовувати код у всіх проектах, над якими працюють.

Існує багато фреймворків для роботи з каскадною таблицею стилів CSS, розглянемо декілька з них.

#### 3.1.1 Опис фреймворку Bootstrap

Bootstrap - одна з найпопулярніших фреймворків CSS. Поточна версія цього фреймворку - Bootstrap 4, яка вийшла в 2018 році. У цьому випуску було представлено багато значущих особливостей, таких як нові кольорові схеми, нові модифікатори, нові класи утиліти тощо. Крім того, версія 4 Bootstrap побудована з SASS, а це означає, що Bootstrap тепер підтримується як LESS, так і SASS[3]<sup>3</sup>.

Bootstrap забезпечує чутливий дизайн, використовуючи систему Grid. Він простий у використанні, і ви можете швидко створити сітку, яка буде добре працювати у всіх браузерах.

Bootstrap надає великі бібліотеки для розробників, наприклад, макети веб-сайтів, шаблони веб-сайтів, теми Bootstrap, панелі адміністратора та велику

---

<sup>3</sup>) [3] Офіційний сайт фреймворку. URL: [bootstrap.https://getbootstrap.com/docs/4.5/getting-started/introduction/](https://getbootstrap.com/docs/4.5/getting-started/introduction/)

колекцію компонентів інтерфейсу. До компонентів входять кнопки, форми, картки, смужки виконання, сповіщення.

Фреймворк Bootstrap хороший для нових веб-розробників.

Використовуючи цей інструмент, ви можете приєднатись до поля розробки.

Існує маса корисної документації та навчальних посібників, на які можна покластися, коли виникають питання.

Інші можливості Bootstrap:

- Використовує Flexbox
- Хороша документація
- Включає компоненти HTML та JavaScript

### 3.1.2 Опис фреймворку Foundation

Foundation та Bootstrap - це широко використовувані фреймворки CSS, але Foundation більш складний. Цей фреймворк дуже гнучкий та легко налаштовується. Є корисним інструментом для створення сучасних, чутких веб-сайтів та веб-додатків, особливо для підприємства. Facebook, eBay, Mozilla, Adobe, HP, Cisco та Disney використовують Foundation у своїх продуктах<sup>[4]</sup>.

Подібно до Bootstrap, Foundation - це також дуже гнучкий фреймворк, який допомагає веб-розробникам створювати веб-сайти, додатки та електронні листи, які чудово виглядають на будь-якому пристрої.

Окрім веб-сайту та додатків, Foundation також може бути використаний для створення чудових чуйних HTML-листів. Ви можете створювати HTML-адреси електронної пошти, використовуючи Foundation для електронних листів. Не потрібно використовувати складну розмітку таблиці або інші речі. Це чудова допомога для компаній, що займаються маркетингом електронної пошти.

---

<sup>4)</sup> [4] Офіційний сайт фреймворку foundation. URL:

<https://get.foundation/showcase/about.html>

Foundation може бути великою підмогою, якщо ви знаєте, як ним користуватися, але досягти професійного рівня у ньому може бути важко. Ось чому Zubr (компанія, яка розробила Foundation) відкрила онлайн-семінар та професійне консультування, щоб навчити вас та вашу команду цінним навичкам, проте навчання не безкоштовне.

Фонд набагато гнучкіший, ніж Bootstrap. Front-end розробник має повний контроль над інтерфейсами користувача. Однак саме через це фреймворк Foundation може опинитися важким для людей які не працювали з ним.

Інші можливості Foundation:

- Вертикальний макет часової шкали
- Чуткі HTML-шаблони та компоненти інтерфейсу
- Корисні інструменти, які дозволяють вирішити багато проблем, пов'язаних з передовим досвідом

### 3.1.3 Опис фреймворку Bulma

Bulma - це безкоштовна та відкрита платформа CSS, заснована на моделі компонування Flexbox. Це легкий, чуйний, чистий CSS. Усі ці функції зробили Bulma однією з найпопулярніших фреймворків CSS разом з Bootstrap та Foundation<sup>[5]</sup>.

Bulma надає розробникам читабельні назви класів CSS та готові до використання компоненти для створення зручних для мобільних пристроїв інтерфейсів. Розпізнати та запам'ятати назви класів CSS неймовірно просто, тому що всі вони названі логічно.

Bulma створена на чистому CSS. Щоразу, коли ви використовуєте фреймворк, все, що вам потрібно, - це один файл .css, і не потрібно .js. Крім того, розробники можуть легко змінювати усі компоненти за допомогою класів модифікаторів та змінних.

---

<sup>5)</sup> [5] Офіційний сайт фреймворку foundation. URL:

<https://bulma.io/> - офіційний сайт фреймворку foundation.

У Vulma є велика громада. Розробники можуть спілкуватися між собою, задавати питання та отримувати відповіді.

Крива низького рівня навчання - ще одна перевага Vulma. Це відмінна основа для новачків.

Інші можливості Vulma: На основі Flexbox

- Побудований із Sass
- Легкий у навчанні, простий у користуванні

### **3.2 Мультипарадигменна мова програмування Java Script**

Спочатку JavaScript був створений, щоб «зробити веб-сторінки живими». Програми на цій мові називаються скриптами. Вони можуть вбудовуватися в HTML і виконуватися автоматично при завантаженні веб-сторінки.

Скрипти поширюються і виконуються, як простий текст. Їм не потрібна спеціальна підготовка або компіляція для запуску.

Сьогодні JavaScript може виконуватися не тільки в браузері, а й на сервері або на будь-якому іншому пристрої, який має спеціальну програму, що називається «движком» JavaScript.

У браузера є власний движок, який іноді називають «віртуальна машина JavaScript». Різні движки мають різні «кодові імена». наприклад: V8 - в Chrome і Opera. SpiderMonkey - в Firefox.

Сучасний JavaScript - це «безпечний» мову програмування. Він не надає низькорівневий доступ до пам'яті або процесору, тому що спочатку був створений для браузерів, які не потребують цього.

Можливості JavaScript сильно залежать від оточення, в якому він працює. Наприклад, Node.JS підтримує функції читання / запису довільних файлів, виконання мережевих запитів<sup>[6]</sup>.

---

<sup>6</sup>[6]Дронов В.А. JavaScript в WEB-Дизайне. - СПб.: БХВ-Петербург, 2003. – 354 с.: ил. ISBN 5-304-00489-X

У браузері для JavaScript є все, що пов'язано з маніпулюванням веб-сторінками, взаємодією з користувачем і веб-сервером.

Наприклад, в браузері JavaScript може:

- Додавати новий HTML-код на сторінку, змінювати існуючий вміст, модифікувати стилі.
- Реагувати на дії користувача, клацання миші, перемістити вказівник, натискання клавіш.
- Відправляти мережеві запити на віддалені сервера, завантажувати і завантажувати файли (технології AJAX і COMET).
- Отримувати і встановлювати куки, задавати питання відвідувачеві, показувати повідомлення. Запам'ятовувати дані на стороні клієнта ( «local storage»).

Коли інтерпретатор JavaScript вбудовується в веб браузер, результатом являється клієнтський JavaScript. Це, безумовно, найбільш поширений варіант JavaScript, і більшість людей, згадуючи JavaScript, зазвичай подразумеють саме клієнтський JavaScript.

Документи можуть містити JavaScript сценарії, які в свою чергу можуть використовувати модель DOM для модифікації документа або управління способом його відображення.

Іншими словами, можна сказати, що клієнтський JavaScript дозволяє визначити поведінку статичного вмісту веб сторінок. клієнтський JavaScript є основою таких технологій розробки веб додатків, як DHTML, і таких архітектур, як Ajax.

### **3.3 JS бібліотека JQuery**

jQuery - набір функцій JavaScript, фокусируючийся на взаємодії JavaScript та HTML. Бібліотека jQuery допомагає легко отримати доступ до будь-якого елементу DOM, звертатися до атрибутів та вмісту елементів DOM,



маніпулюючи ними. Також бібліотека jQuery надає зручний API для роботи з AJAX. Розробка jQuery ведеться командою добровольців на пожертвування.

### 3.3.1 Опис бібліотеки JQuery

jQuery це популярна бібліотека JavaScript. Вона була створена Джоном Резігом в 2006 році з метою полегшити розробникам використання JavaScript на веб-сайтах. Це не окрема мова програмування і працює в поєднанні з JavaScript. З jQuery ви будете робити набагато більше з меншими витратами - дозвольте нам пояснити що таке jQuery більш детально.

Написання коду може стати виснажливим, особливо якщо в нього включено багато рядків. jQuery стискає кілька рядків коду в одну функцію, тому вам не потрібно переписувати цілі блоки коду для виконання одного завдання.

Причина того, чому jQuery стала настільки успішною і популярною, це, ймовірно, кросплатформені можливості. Вона автоматично виправляє помилки і працює таким же чином в найбільш часто використовуваних браузерах, таких як Chrome, Firefox, Safari, MS Edge, IE, Android і iOS. JQuery також робить Ajax набагато простіше [7].

Ajax працює асинхронно з іншою частиною коду. Це означає, що код, написаний на Ajax, може взаємодіяти з сервером і оновлювати його вміст без необхідності перезавантаження сторінки.

Однак це пов'язано з проблемами. Різні браузери виконують Ajax API по-різному. Таким чином, код повинен відповідати всім браузерам. Вручну, це важка і трудомістка робота. На щастя, jQuery виконує всю важку роботу і адаптує код для всіх веб-браузерів.

Потім є маніпулювання DOM у якому є кілька методів, як це зробити. Простіше кажучи, він дозволяє вставляти або видаляти елементи DOM на HTML-сторінці, а також спрощує перенесення рядків.

---

<sup>7)</sup> [7] Javascript і jQuery. Інтерактивна веб-розробка. Дакетт Джон, Ескмо, 2017 -136 с.

Створення анімації також спрощено за допомогою jQuery, обхід документів HTML, а також виконання ефектів і обробка подій також поліпшені за допомогою jQuery.

Оскільки це одна з найпопулярніших доступних бібліотек JavaScript, в інтернеті є безліч ресурсів про те, як підключити jQuery. Незважаючи на базове розуміння JavaScript, HTML і CSS можуть бути необхідні.

### **3.3 Мова гіпертекстової розмітки документа HTML**

Мова гіпертекстової розмітки HTML повинна містити не тільки теги розмітки і теги логічного форматування документів, а й забезпечувати будь-яке оформлення документів, яке виноситься за межі коду в стилях. Такий підхід дозволяє незалежно управляти видом елементів html-сторінки і її вмістом. Дизайнер, верстальник і програміст працюють над своїми завданнями автономно, зменшуючи час на розробку системи.

Для здійснення розробки були використані каскадні таблиці стилів CSS для створення єдиного стилю html-сторінок. Основними вимогами до дизайну інтерфейсу web-додатка є простота, єдиний стиль і уніфіковане відображення інформації різними браузерами<sup>[8]</sup>.

Для задоволення цих вимог була розроблена власна система стилів на основі специфікації CSS. Використання CSS зумовлено для опису та оформлення зовнішнього вигляду сторінок системи, написаних за допомогою мови розмітки HTML. Такий підхід до проектування web-додатку дозволяє, у випадку необхідності, змінити дизайн системи шляхом зміни стилів без модифікації програмного коду. Основною метою розробки CSS було розділення опису логічної структури web-сторінки (яке проводиться за допомогою HTML або інших мов розмітки) від опису зовнішнього вигляду цієї web-сторінки (яке тепер проводиться за допомогою формальної мови CSS). Такий поділ може збільшити доступність документа, надати велику гнучкість і

---

<sup>8)</sup> [8] Петюшкін Олексій HTML в Web-дизайні: - Москва, БХВ-Петербург, 2004 р 86 с.

можливість управління його поданням, а також зменшити складність і повторюваність в структурному вмісті [9].

Основне навантаження при розробці web-додатку лягло на стилі, з їх допомогою можна управляти видом будь-якого елемента, в тому числі змінювати значення динамічно через JavaScript.

### **3.4 Мова каскадних стилів CSS**

Каскадні таблиці стилів CSS – мова, яку використовують для опису вигляду (кольори, шрифти, розташування блоків та інші аспекти) веб-сторінок. Це спрощує процес створення сторінок і поліпшує їх дизайн.

Стиль – це набір правил оформлення та форматування, який можна застосовувати до різних елементів веб-документа. У разі використання стандартної мови HTML для надання кільком елементам певних властивостей (наприклад, призначення стилю шрифту) доводиться здавати ці властивості для кожного елемента. Використання таблиць стилів дає змогу уникнути цього, оптимізувавши розробку веб-сайтів. Стилї дають змогу позиціювати елементи сторінки (наприклад, тексти і графіку), задаючи координати. Крім того, таблиці стилів часто використовують під час створення так званих динамічних сторінок<sup>[10]</sup>.

Таблиці каскадних стилів містять параметри форматування частини або всього тексту веб-сторінки. Якщо таку таблицю підключено, то у тегах можна просто вказувати посилання на неї, а не задавати велику кількість атрибутів. У цьому випадку стилі названі каскадними тому, що в одному документі їх можна описати кілька, і браузер використовуватиме їх каскадом відповідно до їхнього пріоритету.

Використання стилів у веб-технологіях схоже на їх використання в сучасних текстових редакторах — текст спочатку вводять, а потім форматують,

---

<sup>9)</sup> [9] Мержевич В.А. HTML і CSS на прикладах. - СПб.: БХВ-Петербург, 2005. - 40 с.: іл. ISBN 5-94157-360-X

<sup>10)</sup> [10] Мержевич В.А. HTML и CSS на примерах. – СПб.: БХВ-Петербург, 2005. – 40 с.: ил. ISBN 5-94157-360-X

користуючись стилями. Застосування стилів дає змогу виводити на веб-сторінку потрібні тексти та інші елементи, не задумуючись над їхнім зовнішнім виглядом і розташуванням. При описі стилів вказується селектор - назва тега (чи список назв тегів) і у фігурних дужках опис стилю (пари властивість:значення записують через крапку з комою) (Рис 2.2).

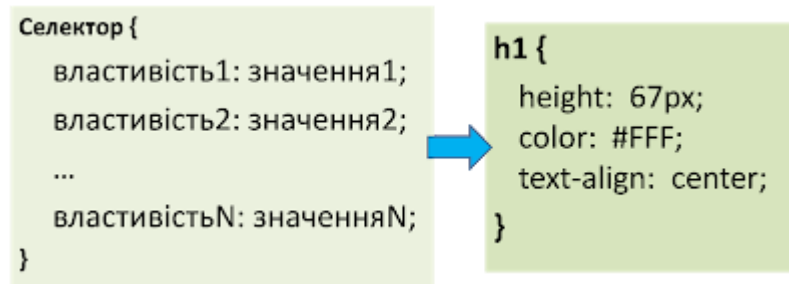


Рисунок 3.1 – Селектори CSS

### 3.5 Вибір середовища розробки

Для написання веб порталу підходить і звичайний текстовий редактор блокнот, але краще використовувати середовище розробки з великим функціоналом, наприклад інтегроване середовище розробки, це збільшить швидкість роботи в рази, швидка орієнтація по проекту, і безліч інструментів. Інтегроване середовище розробки - це система програмних засобів, яка використовується програмістам для розробки програмного забезпечення. В англійській мові таке середовище називається Integrated development environment або скорочено IDE.

ІСР зазвичай включає в себе текстовий редактор, компілятор, інтерпретатор, засоби автоматизації розробки і збірки програмного забезпечення і відладчик. Іноді також містить засоби для інтеграції з системами управління версіями і різноманітні інструменти для спрощення конструювання графічного інтерфейсу користувача. Багато сучасних середовищ розробки також включають вікно перегляду програмних класів, інспектор об'єктів і діаграму ієрархії класів - для використання при об'єктно-орієнтованій розробки

ПЗ. Більшість сучасних ІСР призначені для розробки програм на кількох мовах програмування одночасно.

Один з окремих випадків ІСР - середовища візуальної розробки, які включають в себе можливість візуального редагування інтерфейсу програми.

Основним вікном, є текстовий редактор, який використовується для введення вихідного коду в ІСР і орієнтований на роботу з послідовністю символів в текстових файлах. Такі редактори забезпечують розширену функціональність - підсвічування синтаксису, сортування рядків, шаблони, конвертацію кодувань, показ кодів символів. Іноді їх називають редакторами коду, так як основне їхнє призначення - написання вихідних кодів комп'ютерних програм.

Підсвічування синтаксису - виділення синтаксичних конструкцій тексту з використанням різних кольорів, шрифтів і накреслень. Зазвичай застосовується в текстових редакторах для полегшення читання вихідного тексту, поліпшення візуального сприйняття. Часто застосовується при публікації вихідних кодів в Інтернет.

### **3.5.1 Visual Studio Code**

Visual Studio Code - це редактор вихідного коду. Він підтримує ряд мов програмування, підсвічування синтаксису, IntelliSense, рефакторинг, налагодження, навігацію по коду, підтримку Git та інші можливості. Багато можливості Visual Studio Code недоступні через графічний інтерфейс, найчастіше вони використовуються через палітру команд або JSON файли (наприклад, призначені для користувача настройки). Палітра команд представляє собою подобу командного рядка, яка викликається поєднанням клавіш.

Visual Studio також дозволяє замінювати кодову сторінку при збереженні документа, символи перекладу рядка і мову програмування поточного документа. З 2018 року з'явилися розширення Python для Visual Studio Code з відкритим вихідним кодом. Воно надає розробникам широкі можливості для

редагування, налагодження і тестування коду. На березень 2019 року за допомогою вбудованого в продукт призначеного для користувача інтерфейсу можна завантажити і встановити кілька тисяч розширень тільки в категорії «programming languages» (мови програмування).

### 3.6 Структура веб-порталу

Структура сайту - це логічна побудова всіх сторінок ресурсу. Схема, за якою розподіляється шлях до папок, категоріям, підкатегоріями, картками товарів (якщо вони передбачені). З технічної точки зору, навігація ресурсу являє собою набір URL, логічно вибудованих в певній послідовності. Структура взаємопов'язана з семантичним ядром. Саме воно говорить про те, які папки і документи повинні бути присутніми на сайті. Тому, зібравши семантику, вже можна зробити начерки схеми побудови кожного майбутнього урла [11].

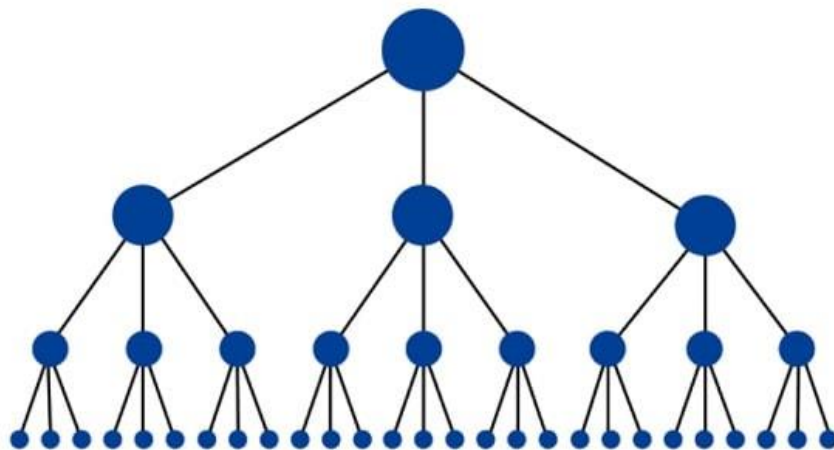


Рисунок 3.2 – Приклад структури

Структуру можна виділити:

- За рівнем вкладеності сторінок - глибина, кількість кліків від головної сторінки.

<sup>11)</sup> [11] Правильна структура веб сайту під SEO. URL: <https://impulse-design.com.ua/pravilnaya-struktura-veb-sajta-pod-seo.html>

- Інформаційна або семантична структура - дає поняття про те, які сторінки є на сайті, і як вони охоплюють попит по семантиці.
- За сегменту в URL - кількість слешів в URL. Іноді структура сайту відображається в адресах для зручності сприйняття користувачами.
- Побудова структури є важливим етапом внутрішньої технічної оптимізації сайту. Структура сайту впливає на:
  - Поведінковий фактор SEO-оптимізації. Від того, наскільки зручна і зрозуміла структура сайту для користувачів, залежить їх поведінку всередині сайту і показник відмов.
  - Конверсію. Вона безпосередньо пов'язана з юзабіліті сайту, яке грає в процесі покупки чи не вирішальну роль.
  - Передачу посилального ваги. Правильна структура сайту допоможе правильно розподілити контрольний вагу за пріоритетом.
  - Швидкість індексації. Проста і прозора структура сайту спрощує маршрут пошуковим роботам і сприяє швидшій і великій індексації сайту.
  - Сайтлінкі. Для відображення в пошуковій видачі додаткових внутрішніх посилань вашого сайту необхідно створити правильну структуру.
  - Основні вимоги до структури сайту полягають в наступному:
    - На сторінках присутні навігаційні ланцюжки (хлібні крихти).
    - Рівень вкладеності сторінок не перевищує 4 (тобто для переходу до шуканої сторінці потрібно не більше трьох кліків від головної)(рисунок 3.3).
    - З усіх сторінок повинна бути посилання на головну сторінку сайту.
    - При збільшенні або зменшенні кількості категорій і підкатегорій структура сайту повинна залишатися незмінною.

### 3-х уровневая структура сайта



Рисунок 3.3 Приклад трирівневого сайту

Сайт розробляється як веб-портал КНУА. Структура пред'явлена на рисунку 3.2:

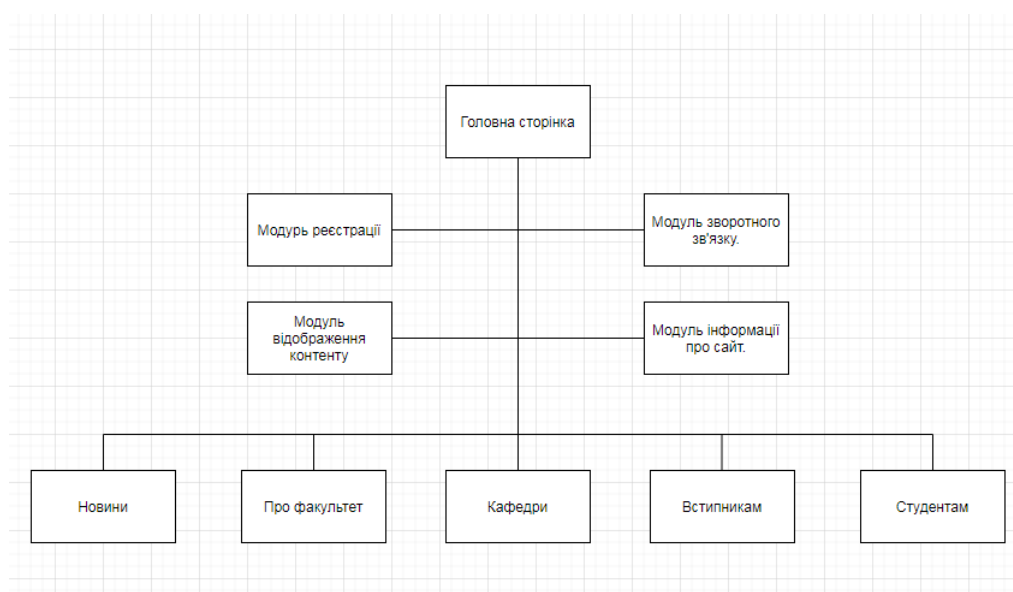


Рисунок 3.4 – Структура Сайту



### 3.7 Вибір системи для управління версіями проекту

Система управління версіями (Version Control System, VCS) - це програмне забезпечення для полегшення роботи зі змінною інформацією. Система управління версіями дозволяє зберігати кілька версій одного і того ж документа, при необхідності повертатися до попередніх версій, визначати, хто і коли зробив ту чи іншу зміну, і багато іншого. Такі системи найбільш широко використовуються при розробці програмного забезпечення для зберігання вихідних кодів програми, що розробляється.

#### 3.7.1 Версіонування

Розглянемо приклад програміста, який закінчив працювати над проектом і відправив фінальну версію замовнику. У програміста є папка, в якій зберігається фінальна версія проекту:

```
source /  
project_index_final.php
```

Програміст закінчив роботу, але замовник надіслав у відповідь правки. Щоб була можливість повернутися до старої версії проекту, програміст створив новий файл `project_index_final_2.php`, вніс зміни і відправив замовнику:

```
source /  
project_index_final.php  
project_index_final_2.php
```

Цим все не обмежалося, в результаті структура проекту розрослася і стала виглядати так:

```
source /  
project_index_final.php  
project_index_final_2.php  
...  
project_index_final_19.php  
...
```

project\_index\_latest\_final.php

project\_index\_latest\_final\_Final.php

У професійній розробці використання нових файлів для версіонування є поганою практикою. Зазвичай у розробників в папці проекту зберігається безліч файлів. Також над одним проектом може працювати кілька людей. Якщо кожен розробник для версіонування створюватиме новий файл, трохи змінюючи назву попередньої версії, то незабаром в проекті почнеться хаос і ніхто не буде розуміти які файли потрібно відкривати.

### 3.7.2 Git

Для вирішення проблеми зі збереженням нової версії файлів зручно використовувати систему Git так як в середовищі розробки яка була обрана для проекту є плагіни і підтримка роботи з цією системою. Роботу Git можна порівняти з процесом збереження та завантаження в комп'ютерних іграх:

- якщо попереду чекає важкий бій, то перед цим краще заздалегідь зберегтися;
- щоб це зробити, потрібно виконати спеціальну команду;
- після чого збереження потрапляє в спеціальну папку і містить стан гри.

Тепер при необхідності завжди є можливість повернутися до попередньої версії гри.

Файли, необхідні для роботи програми, зберігаються в робочій області. В папці saves зберігається історія всіх збережень гри. Git зберігає код вашого проекту за таким же принципом: збереження потрапляють в спеціальну приховану папку, а робочою областю є вміст кореневої папки. Основні поняття:

Репозиторій. Проект, в якому була ініціалізована система Git, називається репозиторієм. При ініціалізації в проект додається прихована папка .git. Репозиторій зберігає всі робочі файли та історію їх змін.

Комміт.

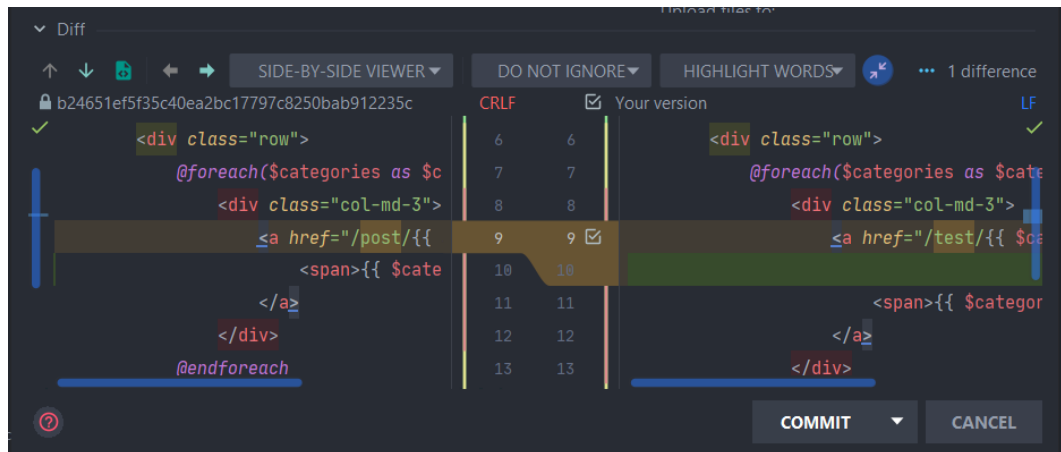


Рисунок 3.5 – Приклад комміту

Точно так же, як і в грі, в системі контролю версій Git можна зберегти поточний стан проекту. Для цього є спеціальна команда - commit. Вона робить так, що нова версія проекту зберігається і додається в сховище. У файлі зі збереженням відображаються: всі зміни, які відбувалися в робочій області, автор змін і короткий коментар, що описує суть змін. Кожен Комміт зберігає повний стан робочої області, її папок і файлів проекту. У підсумку проект працює так:

Репозиторій зберігає всі версії проекту. У разі передачі цього проекту іншій людині, він побачить все, що з ним відбувалося до цього.

Нічого не втрачається і не видаляється безслідно. При видаленні файлу в новій версії додається запис про те, що файл був видалений.

Завжди можна повернутися до будь-якої з версій проекту, завантаживши її зі сховища в робочу область.

### 3.7.3 GitHub

GitHub - сервіс онлайн-хостингу репозиторіїв, що володіє всіма функціями розподіленого контролю версій і функціональністю управління вихідним кодом - все, що підтримує Git і навіть більше. Зазвичай він використовується разом з

Git і дає розробникам можливість зберігати їх код онлайн, а потім взаємодіяти з іншими розробниками в різних проектах.

Також GitHub може похвалитися контролем доступу, багтрекінгом, управлінням завданнями для кожного проекту. Мета GitHub - сприяти взаємодії розробників. До проекту, завантаженому на GitHub, можна отримати доступ за допомогою інтерфейсу командного рядка Git і Git-команд.

Також є й інші функції, такі як документація, запити на прийняття змін (pull requests), історія коммітів, інтеграція з безліччю популярних сервісів, email-повідомлення, емодзі, графіки, вкладені списки завдань.

Git - це інструмент, що дозволяє реалізувати розподілену систему контролю версій, а GitHub - це сервіс для проектів, що використовують Git.

Якщо робота над проектом ведеться в команді, то перед тим як почати писати код потрібно отримати останню версію проекту. Для цього потрібно виконати команду pull. Так ми забираємо всі зміни, які були здійснені з часу останньої синхронізації з віддаленим репозиторієм. Тепер вони у нас в репозиторії на локальному комп'ютері.

Щоб відправити колегам останню версію проекту виконуємо команду push. Якщо у віддаленому репозиторії з моменту останньої синхронізації не було ніяких змін, то всі збережені зміни успішно завантажаться в хмару і колеги отримають останню версію проекту, виконавши команду pull. Якщо ж були зміни, то Git попросить вас перед відправкою підтягнути останні версії, зробивши pull.

## 4 ПРОЄКТНІ РІШЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ

### 4.1 Розробка дизайну веб-порталу

Інформаційний портал складається з різних модулів, кожен з яких реалізує певний функціонал. Весь контент, розміщений на сайті, зберігається в базі даних і завантажується при необхідності за допомогою запитів.

Згідно з розробленою структурою була спроектована головна сторінка Web-сайту. До її складу входять всі основні структурні елементи, перехід за якими здійснюється за допомогою гіперпосилань.

У розробку сайту входить процес створення макета Web-сторінок, на який згодом будуть нанизуватися всі інші елементи. При цьому відбувається формування так званих структурних блоків сайту - відокремлених модулів, кожен з яких відіграє певну роль і відповідає за певний функціонал ресурсу.

#### 4.1.1 Основна область сайту

Самий верхній блок часто називають шапкою сайту, або заголовком сайту, або хедер від англ. header. Місце, в якому зазвичай розташовується хедер - верхня частина сторінки. Орієнтація хедера - альбомна (рисунок 4.1).

В даному блоці розміщується:

- назва сайту;
- головне меню;
- модуль авторизації і реєстрації;

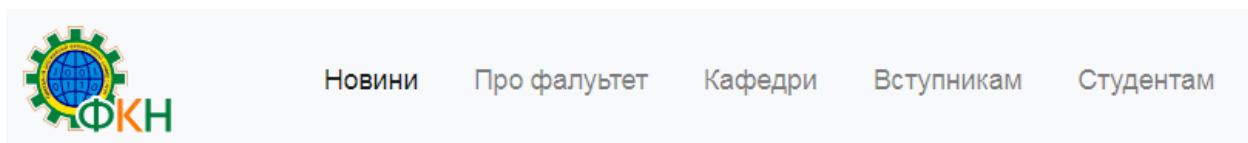


Рисунок 4.1 - Шапка розроблюваного сайту «КНУА»

## 4.1.2 Основна область сайту

Основна область сайту (область основного контента). Слово «Контент» походить від англійського «content» - зміст. Це найбільша за площею і найбільш важлива для відвідувачів сайту частина сторінки. Osnovnaya oblast 'sayta (oblast' osnovnogo kontenta). Slovo «Kontent» proiskhodit ot angliyskogo «content» - sodержaniye. Eto samaya bol'shaya po ploshchadi i naiboleye vazhnaya dlya posetiteley sayta chast 'stranitsy. Основна область сайту (область основного контенту). Слово «Контент» походить від англійського «content» - Зміст. це найбільша за площею и найбільш важлива для відвідувачів сайту частина Сторінки.

В даному блоці розміщується:

- текстовий контент;
- графічний контент.

Ширина області основного контенту може варіюватися в досить широких межах - тут все залежить від типу макета сайту (жорсткий або гумовий). Гумовий макет дозволяє змінювати ширину блоку в залежності від дозволу монітора, з якого користувач переглядає сайт. жорсткий макет цього зробити не дозволяє (рисунок 4.2):



### О факультете

>Lorem ipsum dolor sit amet consectetur adipisicing elit. Reiciendis inventore, rem libero soluta voluptatibus nihil id deleniti quo officia optio asperiores molestiae earum doloremque sit perspiciatis quis illum a consequatur?

## Рисунок 4.2 - Основная область сайта (область основного контента)

У верхній частині була додана карусель, у якій можливо переглядати голову новину, або оголошення.

Carousel (карусель) - це елемент інтерфейсу для демонстрації на сайті серії зображень (слайдів). Кожне зображення (слайд) демонструється зазвичай кілька секунд, поки не зміниться наступним. Зміна зображення (слайд) може здійснюватися як автоматично через рівні проміжки часу, так і вручну.

Слайд в Bootstrap каруселі може бути представлений не тільки зображенням, а й текстовим контентом. Крім цього при створенні слайда можна використовувати розмітку.

```
<body>
  <nav class="navbar navbar-expand-md navbar-light bg-light sticky-top">...
  </nav>
  <div class="carousel slide" data-ride="carousel" id="slides">
    <ul class="carousel-indicators">
      <li data-target="#slides" data-slide-to="0" class="active"></li>
      <li data-target="#slides" data-slide-to="1"></li>
      <li data-target="#slides" data-slide-to="2"></li>
    </ul>
    <div class="carousel-inner">
      <div class="carousel-item active">
        
        <div class="carousel-caption">
          <h1 class="display-2">Новина</h1>
          <h3>ОПИС НОВИНИ</h3>
          <button type="button" class="btn btn-outline-light btn-lg">Подробнее</button>
        </div>
      </div>
      <div class="carousel-item">
        
      </div>
      <div class="carousel-item">
        
      </div>
    </div>
  </div>
```

Рисунок 4.4 – Код каруселі

#### 4.1.4 Футер

Футер сайту (або підвал). Слово «Футер» походить від англ. «Footer» - підвал. зазвичай футером називають область сайту, яка розташована під усіма іншими блоками. За аналогією з шапкою сайту, футер також має альбомну орієнтацію. Ширина футера також може змінюватися в залежності від дозволу монітора користувача (рисунок 4.4). У підвалі сайту розміщені:

- соціальні посилання;
- дублікат інформації про сайт;
- дублікат форми зворотного зв'язку;

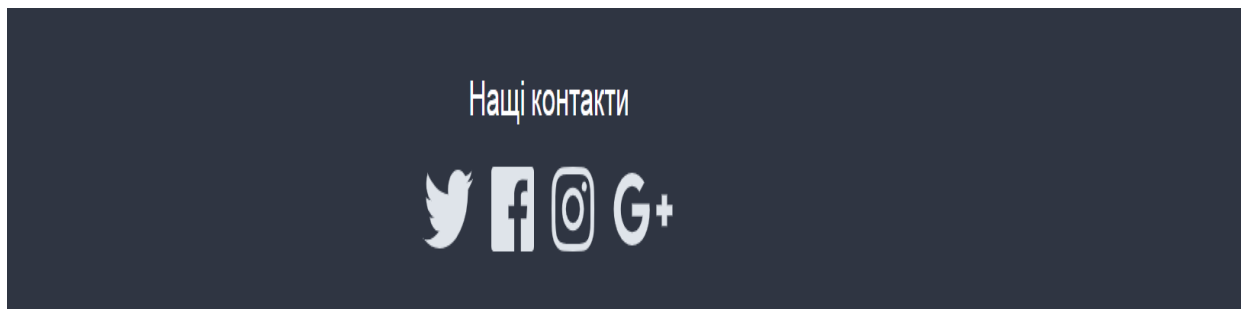


Рисунок 4.3 – Футер

#### 4.1.5 Фон сайту

Фон сайту (фонова область). Фонова область сайту не зайнята ніякими елементами (є повністю вільною). Розмір фонові області залежить від типу макета сайту. При використанні гумового макета фону може і не бути, якщо макет жорсткий, то розміри фонові області будуть прямо залежати від вирішення монітора, з якого користувач переглядає сайт.

Чим більше буде дозвіл, тим більше місця буде займати фон для сайту. Фонова область може заповнюватися певним кольором (за допомогою CSS-атрибута background-color), а також одним великим або декількома маленькими симетричними зображеннями (які будуть рівномірно заповнювати собою весь вільний простір).



Після аналізу цільової аудиторії, було вирішено обрати нейтральну, мінімалістичну палітру.

## 4.2 Опис інтерфейсу користувача.

Інтерфейс розробленого сайту «КНУА» складається з головного меню яке трансформується у меню бургер на пристроях з малім розширенням, за допомогою якого можна переглянути ту чи іншу необхідну інформацію. (рисунок 4.5):

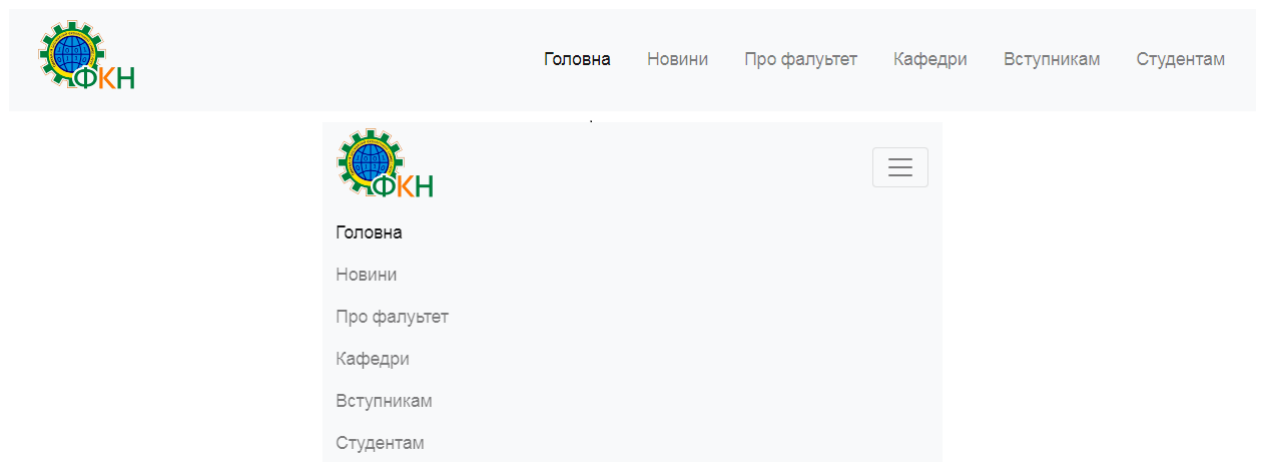
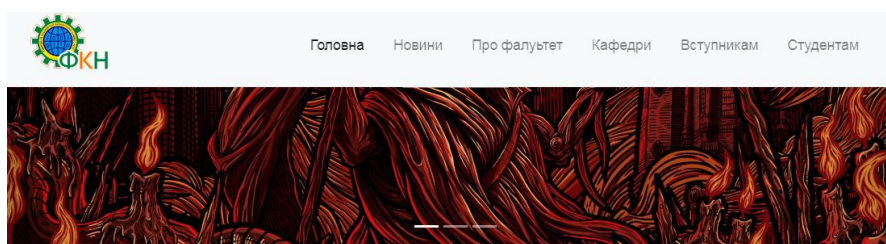


Рисунок 4.5 – Головне меню

Дане меню реалізовано за допомогою фреймворку bootstrap, що забезпечує гнучкість та адаптивність, така реалізація буде однаково зручною на усіх пристроях.

На головній сторінці сайту відображаються головні новини, або оголошення, нижче знаходиться коротка інформація про факультет (рисунок 4.6), над футером можливо зайти коротку інформацію про кафедри (рисунок 4.7).



## О факультете

Lorem ipsum dolor sit amet consectetur adipiscing elit. Reiciendis inventore, rem libero soluta voluptatibus nihil id deleniti quo officia optio asperiores molestiae earum doloremque sit perspiciatis quis illum a consequatur? Lorem ipsum dolor sit amet consectetur adipiscing elit. Reiciendis inventore, rem libero soluta voluptatibus nihil id deleniti quo officia optio asperiores molestiae earum doloremque Lorem ipsum dolor sit amet consectetur adipiscing elit. Reiciendis inventore, rem libero soluta voluptatibus nihil id deleniti quo officia optio asperiores molestiae earum doloremque sit perspiciatis quis illum a consequatur? Lorem ipsum dolor sit amet consectetur adipiscing elit. Reiciendis inventore, rem libero soluta voluptatibus nihil id deleniti quo officia optio asperiores molestiae earum doloremque sit perspiciatis quis illum a consequatur? sit perspiciatis quis illum a consequatur?

Рисунок 4.6 – Інформація про факультет

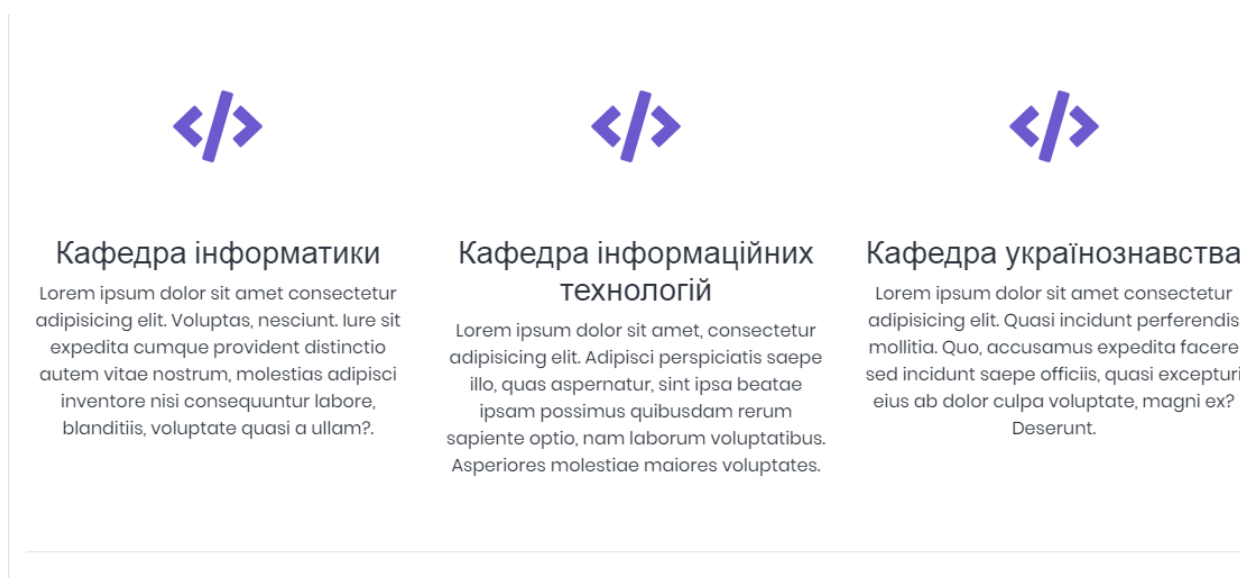


Рисунок 4.7 – Інформація про кафедри

Сторінка «Новини» відображаються новини факультету (рисунок 4.8). На сторінці «Новини», розташовані короткі витяги з новин, що включають заголовок новини, зміст новини. Коли користувач натискає кнопку «Детальніше» перенесе на сторінку з повним змістом новини.

При натисканні на заголовок новини або посилання «Читати далі» користувач потрапляє на сторінку з новиною. Вибірка новин виробляється з бази даних.

На сторінці з повним змістом новини викладачі та студенті можуть ознайомитися з новинами та коментувати. Згідно з рішенням, цей елемент сайту також був розроблений за допомогою фреймворку bootstrap, з використанням класу jumbotron рисунок (4.9).

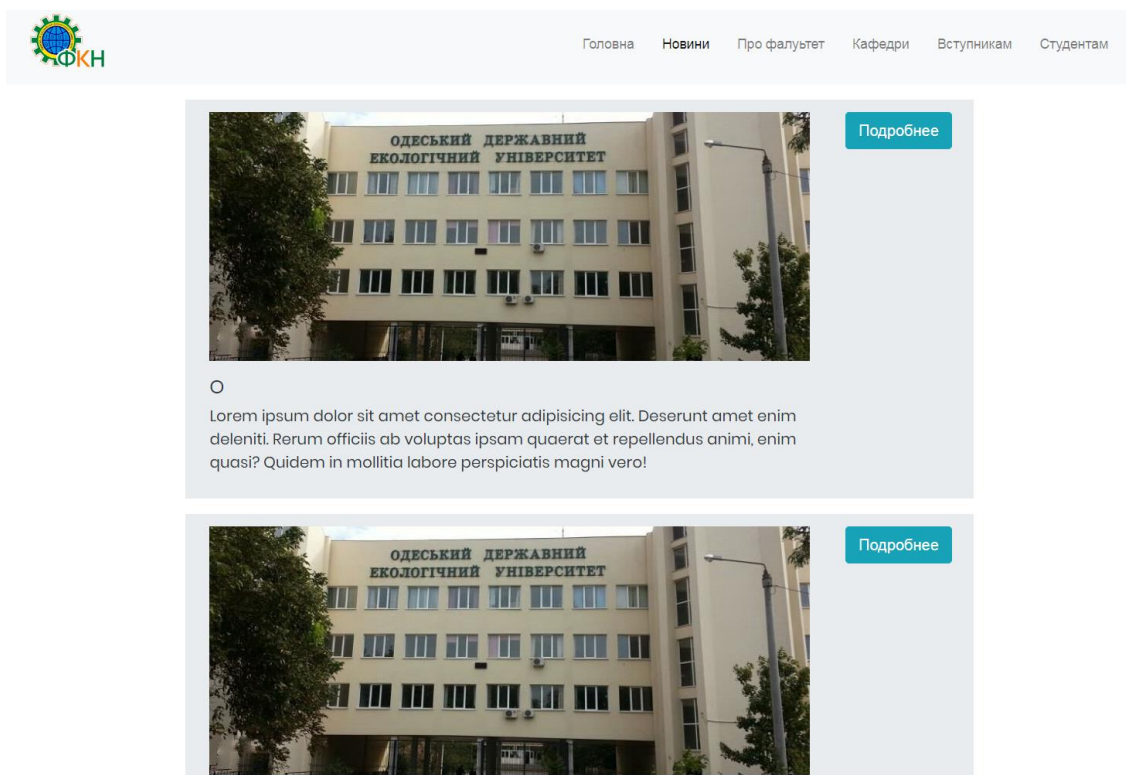


Рисунок 4.8 – Новини

```

class= "container-fluid" >
<div class="row jumbotron">
  <div class="col-xs-12 col-sm-12 col-md-9 col-lg-9 col-xl-10">
    <div class="news-img">
      
    </div>
    <h4>0</h4>
    <p class="lead">Lorem ipsum dolor sit amet consectetur adipisicing el
  </div>
  <div class="col-xs-12 col-sm-12 col-md-3 col-lg-3 col-xl-2">
    <a href="#"><button class="btn btn-info btn-lg" type="button">Подробн
  </div>
</div>

```

## Рисунок 4.9 – Код блоку новин

На сторінці «Про факультет» головного меню сайту знаходиться повна інформація про даний. На сторінці «Кафедри» знаходиться повна інформація про кафедри. На сторінці «Вступникам інформація необхідна вступникам. На сторінці «Студентам» інформація необхідна студентам, розклад, або оголошення від викладачів.

Для здійснення реєстрації на сайті створений модуль реєстрації, який знаходиться у верхній правій частині сайту. При натисканні на посилання «Реєстрація» з'явиться форма для заповнення, в якій має бути вказано необхідні дані (рисунок 4.10), а саме: логін, пароль, e-mail, мобільний телефон, після чого натиснути кнопку «Реєстрація»

Реєстрація

Ім'я

E-Mail адреса

Пароль

Повторіть пароль

## Рисунок 4.10 – Модуль реєстрації

У футері знаходяться посилання на соціальні мережі (рисунок 4.11). Це допоможе поширити популярність соціальних мереж, або допомогти студенту знайти зв'язок з потрібним йому викладачем. Також є посилання на пошту Gmail.

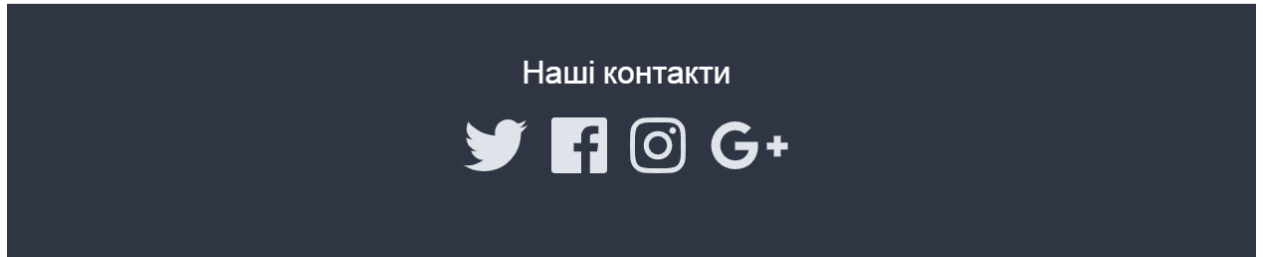


Рисунок 4.11 – Футер

Футер зроблений за допомогою фреймворку bootstrap, та бібліотеки іконок Font Awesome (рисунок 4.12). Іконки мають анімацію, так при звертанні переносять користувача на обрану соц-мережу

```
<footer class="container-fluid">
  <div class="container-fluid">
    <div class="row padding text-center">
      <div class="col-12">
        <h2>Наші контакти</h2>
      </div>
      <div class="col-12 social padding">
        <a href="#"><i class="fab fa-twitter"></i></a>
        <a href="#"><i class="fab fa-facebook"></i></a>
        <a href="#"><i class="fab fa-instagram"></i></a>
        <a href="#"><i class="fab fa-google-plus-g"></i></a>
      </div>
    </div>
  </div>
</footer>
```

Рисунок 4.12 – Код футеру

## ВИСНОВКИ

В ході виконання даної дипломної роботи були розглянуті та проаналізовані найважливіші вимоги для розробки інформаційного порталу, а також була вивчена предметна область розробляється інформаційного порталу і були виявлені основні компоненти, які будуть складати основну архітектуру веб-ресурсу. Проаналізовано та обрані засоби розробки інформаційного порталу, а також вимоги до нього

Розглянуті питання в дипломній роботі є актуальними в області розробки і створення сучасного веб-сайту. В процесі виконання роботи були вирішені наступні приватні задачі:

- розглянуті мови програмування які що використовуються для створення веб-ресурсів;
- вивчено сучасне програмне забезпечення, що застосовується для створення веб-сайтів;
- проведений аналіз сучасних фреймворків для створення веб-сайтів;
- розглянуті і застосовані основні правила і рекомендації по розробці та створенню веб-сайтів;
- визначена структура і зміст веб-сторінок для сайту;
- проведено аналіз працездатності веб-сайту;
- В результаті вивчення даного напрямку була вибрана мова програмування php, розроблені таблиці стилів для дизайну сайту, створений сучасний веб-портал для факультета КНУА, сайт наповнений контентом.
- Розроблений сайт має ряд переваг:
- сучасний дизайн;
- гнучка система настройки відображення інформації;
- кроссбраузерність розробленого сайту;
- розроблено модуль реєстрації та пошуку.
- розроблено модуль зворотного зв'язку.

Виходячи з матеріалів дипломної роботи можна впевнено сказати, що поставлені завдання, сформульовані у вступі, повністю вирішені. Мета дипломної роботи досягнута.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Браун Бредлі, Річард Дж. Ніемік, Джозеф С. Трезза Oracle Database. Створення Web-додатків: - Санкт-Петербург, Лорі, 2007 р.- 722 с.
2. Браун Д. Розробка веб-сайту. Взаємодія з замовником, дизайнером і програмістом: - Санкт-Петербург, Пітер 2009 р.- 336 с.
3. Офіційний сайт фреймворку bootstrap.URL:  
<https://getbootstrap.com/docs/4.5/getting-started/introduction/>
4. Офіційний сайт фреймворку foundation. URL:  
<https://get.foundation/showcase/about.html>
5. Офіційний сайт фреймворку bulma URL:  
<https://bulma.io/>
6. Дмитрієва Марина JavaScript: - Москва, БХВ-Петербург, 2004 р 336 с
7. Javascript і jQuery. Інтерактивна веб-розробка. Дакетт Джон, Ескмо, 2017 -136 с.
8. Петюшкін Олексій HTML в Web-дизайні: - Москва, БХВ-Петербург, 2004 р 400 с.
9. Мержевіч В.А. HTML і CSS на прикладах. - СПб .: БХВ-Петербург, 2005. - 40 с .: іл. ISBN 5-94157-360-X
10. Правильна структура веб сайту під SEO. URL:  
<https://impulse-design.com.ua/pravilnaya-struktura-veb-sajta-pod-seo.html>



## ДОДАТОК А

### Код клієнтської частини

```

<body>
  <nav class="navbar navbar-expand-md navbar-light bg-light
sticky-top">
    <div class="container-fluid">
      <a href="#" class="navbar-brand"></a>
      <button class="navbar-toggler" type="button" data-
toggle="collapse" data-target="#navbarResponsive">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div class="collapse navbar-collapse"
id="navbarResponsive">
        <ul class="navbar-nav ml-auto">
          <li class="nav-item active">
            <a href="#" class="nav-
link">Головна</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-
link">Новини</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-link">про
фалуьтет</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-
link">Кафедри</a>
          </li>
          <li class="nav-item">
            <a href="#" class="nav-
link">Вступникам</a>
          </li>

```

```

        <li class="nav-item">
            <a href="#" class="nav-
link">Студентам</a>
        </li>
    </ul>
</div>
</div>
</nav>
<div class="carousel slide" data-ride="carousel"
id="slides">
    <ul class="carousel-indicators">
        <li data-target="#slides" data-slide-to="0"
class="active"></li>
        <li data-target="#slides" data-slide-to="1"></li>
        <li data-target="#slides" data-slide-to="2"></li>
    </ul>
    <div class="carousel-inner">
        <div class="carousel-item active">
            
            <div class="carousel-caption">
                <h1 class="display-2">Новина</h1>
                <h3>ОПИС НОВИНИ</h3>
                <button type="button" class="btn btn-
outline-light btn-lg">Подробнее</button>
            </div>
        </div>
        <div class="carousel-item">
            
        </div>
        <div class="carousel-item">
            
        </div>
    </div>
</div>

```

```

<div class="container-fluid">
  <div class="row text-center alert">
    <div class="col-12">
      <h1 class="display-4">0 факультете</h1>
    </div>
    <hr>
    <div class="col-12">
      <p class="lead">Lorem ipsum dolor sit amet
consectetur adipisicing elit. Reiciendis inventore, rem libero
soluta voluptatibus nihil id deleniti quo officia optio
asperiores molestiae earum doloremque sit perspiciatis quis
illum a consequatur?Lorem ipsum dolor sit amet consectetur
adipisicing elit. Reiciendis inventore, rem libero soluta
voluptatibus nihil id deleniti quo officia optio asperiores
molestiae earum doloremque Lorem ipsum dolor sit amet
consectetur adipisicing elit. Reiciendis inventore, rem libero
soluta voluptatibus nihil id deleniti quo officia optio
asperiores molestiae earum doloremque sit perspiciatis quis
illum a consequatur?Lorem ipsum dolor sit amet consectetur
adipisicing elit. Reiciendis inventore, rem libero soluta
voluptatibus nihil id deleniti quo officia optio asperiores
molestiae earum doloremque sit perspiciatis quis illum a
consequatur? sit perspiciatis quis illum a consequatur? </p>
    </div>
  </div>
</div>
<div class="container-fluid padding">
  <div class="row text-center padding">
    <div class="col-xs-12 col-sm-6 col-md-4">
      <i class="fas fa-code"></i>
      <h3>Кафедра інформатики</h3>
      <p>Lorem ipsum dolor sit amet consectetur
adipisicing elit. Voluptas, nesciunt. Iure sit expedita cumque
provident distinctio autem vitae nostrum, molestias adipisci

```

inventore nisi consequuntur labore, blanditiis, voluptate quasi a ullam?.

</div>

<div class="col-xs-12 col-sm-6 col-md-4">

<i class="fas fa-code"></i>

<h3>Кафедра інформаційних технологій</h3>

<p>Lorem ipsum dolor sit amet, consectetur

adipisicing elit. Adipisci perspiciatis saepe illo, quas aspernatur, sint ipsa beatae ipsam possimus quibusdam rerum sapiente optio, nam laborum voluptatibus. Asperiores molestiae maiores voluptates.

</div>

<div class="col-xs-12 col-sm-6 col-md-4">

<i class="fas fa-code"></i>

<h3>Кафедра українознавства</h3>

<p>Lorem ipsum dolor sit amet consectetur

adipisicing elit. Quasi incidunt perferendis mollitia. Quo, accusamus expedita facere sed incidunt saepe officiis, quasi excepturi eius ab dolor culpa voluptate, magni ex?

Deserunt.

</div>

</div>

<hr class="mv-4">

</div>

<footer class="container-fluid">

<div class="container-fluid">

<div class="row padding text-center">

<div class="col-12">

<h2>Наші контакти</h2>

</div>

<div class="col-12 social padding">

<a href="#"><i class="fab fa-twitter"></i></a>

<a href="#"><i class="fab fa-facebook"></i></a>

```

        <a href="#"><i class="fab fa-
instagram"></i></a>
        <a href="#"><i class="fab fa-google-plus-
g"></i></a>
            </div>
        </div>
    </div>
</footer>
</body>

</html>
```