



Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук  
Кафедра Медіасистеми та технології  
Рівень вищої освіти другий (магістерський)  
Спеціальність 186 Видавництво та поліграфія  
Тип програми Освітньо-професійна  
Освітня програма Технології електронних мультимедійних видань  
(шифр і назва)

ЗАТВЕРДЖУЮ:  
Зав. кафедри МСТ \_\_\_\_\_  
(підпис)  
« 1 » жовтня 2021 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студентові Страховій Юлії Олександрівні  
(прізвище, ім'я, по батькові)

Тема роботи Дослідження можливостей фреймворків для створення інформаційних сайтів

затверджена наказом по університету від 29 жовтня 2021р. № 1612 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 13 грудня 2021 р.

3. Вихідні дані до роботи

Мета – дослідження можливостей фреймворків для створення інформаційних сайтів;

Об'єкт – процес вибору фреймворка при створенні інформаційних сайтів;

Предмет – фреймворки для створення інформаційних сайтів

4. Перелік питань, що потрібно опрацювати в роботі

Вступ; 1. Аналіз стану проблеми і постановка задачі дослідження; 2 Теоритичні дослідження;  
3 Експериментальні дослідження; Економічна частина; Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

Титульна сторінка; Завдання на магістерську атестаційну роботу; Актуальність  
дослідження; Мета і задачі роботи; Об'єкт і предмет дослідження; Аналіз відомих  
фреймворків; Вибір і обґрунтування фреймворків; Опис проведення теоритичного  
дослідження; Опис проведення експериментального дослідження; Економічна частина;  
Висновки

6. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	доц. Колесникова Т.А.		
Економічна частина	проф. Полозова Т.В.		

**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Визначення мети та завдань дослідження	21.01.2021	
2	Аналітичний огляд сучасних фреймворків	12.04.2021	
3	Теоритичне дослідження фреймворків	27.05.2021	
4	Експериментальне дослідження	11.09.2021	
5	Економічна частина	19.11.2021	
6	Оформлення пояснювальної записки	07.12.2021	
7	Оформлення графічної частини	09.12.2021	

Дата видачі завдання 1 жовтня 2021 р.

Студент \_\_\_\_\_ Страхова Ю.О.  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Колесникова Т.А.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 76 с., 12 табл., 12 рис., 13 джерел.

ФРЕЙМВОРКИ, ІНФОРМАЦІЙНИЙ САЙТ, ДОСЛІДЖЕННЯ, СТВОРЕННЯ САЙТІВ, СУЧАСНІ ТЕХНОЛОГІЇ, РОЗРОБКА САЙТІВ.

Метою кваліфікаційної роботи є – дослідження можливостей фреймворків для створення інформаційних сайтів.

Об'єктом кваліфікаційної роботи є – процес вибору фреймворка при створенні інформаційних сайтів.

Предметом кваліфікаційної роботи є – фреймворки для створення інформаційних сайтів.

В процесі виконання роботи було проведено аналіз літератури з теми дослідження, проведено аналіз існуючих фреймворків, розглянуто відомі фреймворки. В якості експериментальної частини атестаційної роботи було розроблено сторінку інформаційного сайту. Сторінку сайту було протестовано на різних пристроях.

Було проведено економічне обґрунтування доцільності проведення даної науково-дослідної роботи (НДР).

## РЕФЕРАТ

Объяснительная записка квалификационной работы: 76 с., 12 табл., 12 рис., 13 источников.

ФРЕЙМВОРКИ, ИНФОРМАЦИОННЫЙ САЙТ, ИССЛЕДОВАНИЕ, СОЗДАНИЕ САЙТОВ, СОВРЕМЕННЫЕ ТЕХНОЛОГИИ, РАЗРАБОТКА САЙТОВ.

Целью квалификационной работы является исследование возможностей фреймворков для создания информационных сайтов.

Объектом квалификационной работы является процесс выбора фреймворка при создании информационных сайтов.

Предметом квалификационной работы являются фреймворки для создания информационных сайтов.

В процессе выполнения работы был проведен анализ литературы по теме исследования, проведен анализ существующих фреймворков, рассмотрены известные фреймворки. В качестве экспериментальной части аттестационной работы была разработана страница информационного сайта. Страница сайта была протестирована на разных устройствах.

Было проведено экономическое обоснование целесообразности проведения данной научно-исследовательской работы (НИР).

## ABSTRACT

The explanatory note of qualification work: 76 p., 12 tabl., 12 pic., 13 sources.

FRAMEWORKS, INFORMATION SITE, RESEARCH, WEBSITE CREATION, MODERN TECHNOLOGIES, WEBSITE DEVELOPMENT.

The purpose of the qualification work is to study the capabilities of frameworks for creating information sites.

The object of the qualification work is the process of choosing a framework when creating information sites.

The subject of the qualification work is the frameworks for creating information sites.

In the process of performing the work, an analysis of the literature on the research topic was carried out, an analysis of existing frameworks was carried out, and well-known frameworks were considered. As an experimental part of the certification work, an information sat page was developed. The site page has been tested on different devices.

An economic justification for the feasibility of this research work was carried out.

## ЗМІСТ

	С.
ВСТУП .....	9
1 АНАЛІЗ СТАНУ ПРОБЛЕМИ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ .....	11
1.1 Аналіз проблеми вибору фрейморка для створення інформаційних сайтів .....	11
1.2 Постановка цілей і завдань дослідження, предмет і об'єкт дослідження .....	13
2 ТЕОРИТИЧНІ ДОСЛІДЖЕННЯ .....	15
2.1 Огляд сучасних фреймворків .....	15
2.1.1 Bootstrap .....	15
2.1.2 Tailwind CSS .....	17
2.1.3 Tailwind CSS .....	18
2.1.4 Material Design Lite .....	19
2.1.5 Bulma .....	20
2.1.6 Foundation.....	22
2.1.7 Skeleton.....	23
2.1.8 Semantic UI.....	24
2.1.9 Pure CSS .....	25
2.1.10 UI Kit .....	26
2.2 Метод експертних оцінок .....	27
2.2.1 Зміст методів експертних оцінок.....	27
2.2.2 Класифікація видів експертних оцінок та їх коротка характеристика .....	32
2.2.3 Організація проведення експертного опитування .....	39
2.3 Проведення аналізу .....	44
3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ .....	50
3.1 Вибір та обґрунтування програмного продукту для створення інформаційного сайту .....	50

3.2 Створення сайту за допомогою Bootstrap.....	52
3.3 Створення сайту за допомогою Tailwind CSS .....	60
4 ЕКОНОМІЧНА ЧАСТИНА .....	66
4.1 Характеристика науково-дослідницьких рішень .....	66
4.2 Розрахунок кошторисної вартості НДР .....	67
4.3 Оцінка результатів НДР .....	71
4.4 Визначення економічної ефективності результатів НДР.....	73
ВИСНОВКИ.....	75
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	76

## ВСТУП

За оцінками, існує понад 1,7 мільярда веб-сайтів, але ця кількість коливається щодня, оскільки нові веб-сайти з'являються щодня. Незважаючи на втрату інформаційних ресурсів, Інтернет є величезним, і 4,5 мільярда людей у всьому світі беруть участь у онлайн-взаємодіях.

Трохи більше, ніж за 30 років з моменту свого народження, всесвітня мережа повністю змінила життя людей. За допомогою кількох клацань миші або простої голосової команди можна отримати необхідні речі прямо до дверей користувача або зв'язатися з другом по всьому світу.

На сьогоднішній день веб-розробка активно розвивається, з'являється величезна кількість тегів та CSS-селекторів, які буває важко запам'ятати, тож на допомогу прийшли фреймворки.

Отже, що таке фреймворки і навіщо вони потрібні?

"Фреймворк" - це слово, яке часто використовується у зв'язку з кодом. Коли ми говоримо про фреймворки, можна поділити їх на три різні категорії: інтерфейс, бекенд і фреймворки CSS або інтерфейсу користувача. Але навіть коли ви знаєте, що таке фреймворки, які з них слід вивчити або використовувати і чому? У представленій роботі буде обговорено, що таке фреймворк, які фреймворки існують і як приймати рішення про те, який фреймворк потрібно використовувати для своїх проєктів! Що таке фреймворки? Якщо подивитися на кілька різних веб-додатків, то можна побачити, що існує багато повторюваних функцій. Багато з них мають аутентифікацію користувачів, сторінки, які відображаються, вони підключаються до баз даних, у них є профілі, які можна переглядати, усі вони мають стилізовані канали інформації тощо. Можна самостійно писати всю цю логіку щоразу, при бажанні створити нові заявки; однак, є також багато доступних інструментів, які можуть зробити це замість нас. Крім того, багато часу, при створенні програми. Вебфреймворки допомагають досягти

структури в програмах, і вони дають додаткові функції, які можна додати до них без зайвої роботи. Framework дає можливість почати, щоб існувала можливість зосередитися на функціях, а не на деталях конфігурації. На сьогоднішній день існує багато фреймворків, тож іноді буває складно обрати, з якого саме почати. Для певної потреби існує свій фреймворк. Перш ніж обирати потрібний фреймворк спочатку потрібно дізнатися про важливі елементи, які слід враховувати.

# 1 АНАЛІЗ СТАНУ ПРОБЛЕМИ І ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

## 1.1 Аналіз проблеми вибору фрейморка для створення інформаційних сайтів

«Розробка веб-сайтів» стала важливою для компаній усіх типів і розмірів, незалежно від їх ціннісної позиції. Однак існує багато способів створення веб-сайту, найпопулярнішим є «використовування фреймворків».

Популярність цих веб-сайтів зростає останніми роками, оскільки вони пропонують просте використання фреймворків для розробки веб-сайтів. Крім того, розробники матимуть підвищену швидкість реагування та розуміння структурованого програмного забезпечення, щоб спростити його програмування на підтримуваних мовах.

Але перш за все, фреймворки забезпечують спрощення мов, які вони підтримують, дозволяючи розробникам прискорити редагування та прискорити роботу.

Фреймворк – це абстрактна конкретна/концептуальна платформа, де розробники можуть вибірково змінювати заплановану заміну за допомогою стандартних кодів для стандартних функцій. Однак фреймворк постачається з попередньо визначеною базою коду, яку можна використовувати для зміни простору кампанії. Фреймворки веб-сайтів зазвичай підтримують одну або кілька програм кодування, які прискорюють розробку за допомогою мінімальних наборів кодування.

На відміну від багатьох інших інструментів розробки, фреймворк надає попередньо налаштовану базу коду та рекомендації для полегшення веб-розробки. У той же час він прискорює розробку програмного забезпечення та плагінів, таких як менеджер акцій WooCommerce, що допомагає збільшити конверсію електронної комерції та скорочує час перебування на ринку, скорочуючи життєвий цикл системи розробки.

Нижче наведено основні переваги використання фреймворків для розробки онлайн-програм.

### 1. Полегшує процес розробки.

Фреймворки надають інструменти та пакети, які допомагають людям розпочати розробку своїх веб-сайтів.

Завдяки фреймворкам програмному забезпеченню не потрібно писати всі сценарії з нуля, оскільки фреймворк обробляє багато стандартних шаблонів, розроблених з нуля. Вони також пропонують новачкам можливість досліджувати важливі сфери та пропонують людям широкий спектр інструментів та експертів.

Фреймворки значно скорочують час кодування, оскільки запускають базову структуру і зберігають мережеві програми.

### 2. Спрощує налагодження та обслуговування програм.

Багато мов програмування не приділяють великого значення читабельності та надійності коду.

Фреймворки роблять це, хоча розробка адаптованої до платформи онлайн-програми наполегливо рекомендується, вона має багато спільного з простотою налагодження та обслуговування.

Оскільки з кожним фреймворком пов'язана спільність розробок, завжди є негайна реакція на будь-які проблеми, які виникають із фундаментом. Крім того, багато моделей дозволяють розробникам запускати свої перші тести.

### 3. Зменшує довжину коду.

Немає необхідності писати довгі рядки правил, але слід додати загальні шаблони на веб-сайт.

Це скорочення додаткового часу та зусиль часто пов'язане із загальною продуктивністю.

Крім того, фреймворки надають функції та інструменти, які дозволяють розробникам автоматизувати повсякденні завдання, такі як розробка в Інтернеті, зберігання тексту, аутентифікація керування сеансами та відображення URL-адрес.

#### 4. Покращує знання бази даних.

Багато фреймворків починаються з набору правил, які дозволяють програмам багаторазово працювати з пов'язаними даними.

Хоча деякі стандарти полегшують інтеграцію даних, вони включають механізм реляційного відображення об'єктів (ORM). Інші приклади використовують цей об'єкт ORM для запису безпосередньо в мови програмування.

Більше того, розробники можуть використовувати цю модель ORM для виконання функцій бази даних без написання довгих кодів SQL.

Дійсно, фреймворки дуже корисні з точки зору економії часу та ефективності. Скажімо, навіщо створювати цілу систему, якщо це вже зробив хтось інший і хоче нею поділитися? У них також є величезна спільнота, яку ви можете запитати в будь-який час, і на неї легко знайти відповідь.

### 1.2 Постановка цілей і завдань дослідження, предмет і об'єкт дослідження

Метою магістерської кваліфікаційної роботи є дослідження можливостей фреймворків для розробки інформаційних сайтів.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- аналіз стану проблеми вибору фреймворка при створенні інформаційних сайтів;
- огляд і аналіз існуючих фреймворків;
- сформулювати список критеріїв, які важливі для вибору фреймворка;
- розробити методіку оцінки якості можливостей фреймворка;
- на підставі отриманих даних розробити інформаційний сайт;
- провести оцінку якості результату.

Об'єкт – певна частина наукових знань, що піддається дослідженню.

Об'єкт є якоюсь областю наукового знання, яка може існувати в незалежності від людського знання.

Як предмет виступає та галузь знань, яка укладена в межах об'єкта. Предмет – конкретний аспект проблеми, займаючись розглядом якого авторами пізнається цілісний об'єкт, позначаються і виділяються його характерні властивості. Предмет дослідження часто дуже близький до обраної теми роботи або повністю з нею збігається. Як категорії науки об'єкт і предмет співвідносяться як загальне і часткове.

Об'єктом дослідження даної роботи є процес вибору фреймворка при створенні інформаційних сайтів.

Предмет дослідження – фреймворки для створення інформаційних сайтів.

Таким чином, в ході дослідження пропонується проаналізувати існуючі сучасні фреймворки, визначити параметри та можливості, які є найбільш важливими для розробки інформаційного сайту, розробити інформаційний сайт.

## 2 ТЕОРИТИЧНІ ДОСЛІДЖЕННЯ

### 2.1 Огляд сучасних фреймворків

Існує багато доступних фреймворків CSS, і, звісно, тут і там важко знайти потрібний фреймворк CSS. Отже, нижче представлений список найкращих фреймворків CSS 2021, який допоможе отримати огляд конкретних фреймворків CSS.

#### 2.1.1 Bootstrap

Bootstrap – це найкращий у світі фреймворк CSS з великою підтримкою спільноти. Ця структура побудована на HTML, SASS та javascript. Наразі Bootstrap 4.5.0 є останньою версією з більшою чутливістю до класів утиліт і нових компонентів. Він спрямований на адаптивну розробку інтерфейсу для мобільних пристроїв, що робить його зручним для будь-якого пристрою та зручним для розробників. Bootstrap підтримує всі сучасні браузерери. Найкраща перевага bootstrap полягає в тому, що цей фреймворк має чудові компоненти JavaScript із користувацькими файлами або CDN.

Причини використання Bootstrap:

- Bootstrap пропонує безліч прикладів і попередньо встановлений макет, щоб ви могли почати;
- за допомогою Bootstrap розробники можуть легко з'єднувати різні компоненти та макети, щоб створити новий і вражаючий дизайн сторінки;
- з цими макетами надається багато детальної документації, щоб користувачі могли легко їх зрозуміти;
- Bootstrap заснований на ліцензії MIT, тому він безкоштовний у використанні, безкоштовний для розповсюдження, тож ви можете розвивати та робити внесок у спільноту;

– сторінка Bootstrap Github GitHub складається з понад 19 000 комітів і 2000 учасників;

– Bootstrap CSS фреймворк.

Додаткова інформація:

– дата виходу: 19 серпня 2011 року;

– Git star, Forks, Учасники: 144k, 70,2k, 2261;

– Hacker News, Reddit, Stack overflow: 3,3k, 9k, 98k;

– ліцензія: MIT;

– кількість сайтів: 20 737 671;

– документація: відмінно;

– налаштування: базовий інструмент налаштування графічного інтерфейсу (потрібно ввести значення кольору вручну) ;

– основна концепція: RWD і мобільний в першу чергу;

– крива навчання: помірна;

– сітка: на основі Flexbox до 12 стовпців;

– підтримка браузера: новітні версії Chrome, Safari, Firefox, Opera, Safari, Edge та IE 10+, Android версії 5.0+;

– компанії, які використовують: Spotify, Coursera, Vine, Twitter, Walmart та багато інших.

Ідеально підходить для:

– початківець, який новачок у CSS, оскільки він чи вона може запустити Bootstrap без будь-яких перешкод;

– розробник з невеликим знанням JavaScript, який все ще може використовувати компоненти Bootstrap без написання рядка на JS;

– внутрішній розробник, який хоче внести деякі зміни в інтерфейс користувача, навіть якщо він або вона новачок у HTML і CSS.

## 2.1.2 Tailwind CSS

Tailwind CSS – це високонастроюваний низькорівневий утилітний каркас CSS, який надає вам усі будівельні блоки, необхідні для створення індивідуальних проектів без будь-яких дратівливих упевнених стилів, з якими ви повинні боротися, щоб перевизначити. На відміну від інших фреймворків CSS (Bootstrap або Materialize CSS), він не має попередньо визначених компонентів. Натомість він працює на нижчому рівні і пропонує вам набір допоміжних класів CSS. Використовуючи ці класи, ви можете швидко створювати власний дизайн легко. Tailwind CSS дозволяє створити власний унікальний дизайн.

Причини використання Tailwind:

- немає теми за замовчуванням;
- не нав'язує дизайнерські рішення, за які вам доведеться боротися;
- пропонує фору в реалізації індивідуального дизайну з власною ідентичністю;
- постачається з меню попередньо розроблених віджетів для створення вашого сайту.

Додаткова інформація:

- дата виходу: 2 листопада 2017 року;
- Git star, Forks, Учасники: 27,1k, 1,3k, 255;
- Hacker News, Reddit, Stack overflow: h, 245,525;
- ліцензія: MIT;
- кількість сайтів: 11 671;
- документація: добре;
- основна концепція: утиліта в першу чергу, чуйний;
- сітка: Flexbox;
- підтримка браузера: всі сучасні браузери;
- компанії, які використовують: Setel, Livestorm, Mogenic, Declik.

### 2.1.3 Tailwind CSS

Materialize CSS – це адаптивний інтерфейсний фреймворк, заснований на матеріальному дизайні з колекціями UI-компонентів з мінімальними ефектами, на які можна легко залучити користувачів. Materialize повністю реагує на планшети та мобільні пристрої. Його легко вивчити, а також надається чудова документація. Ця структура має велику підтримку спільноти та позитивні відгуки. Materialize CSS дозволяє налаштовувати параметри за допомогою вражаючого набору колірних колекцій.

Шаблони адміністратора Materialize, засновані на фреймворку Materialize CSS, широко використовуються в усьому світі завдяки його швидкості реагування.

Причини використання Materialize CSS:

- сторінка документації Materialize є дуже вичерпною і досить легкою для початку;
- на GitHub від Materialize перелічено понад 3800 комітів і 500 учасників;
- сторінка компонентів Materialize містить картки, кнопки, навігацію та багато інших додаткових функцій;
- матеріалізуйте CSS Framework.

Додаткова інформація:

- дата випуску: вересень 2011 року;
- Git star, Forks, Учасники: 38k, 4,9k, 515;
- Reddit, переповнення стека: 374, 3,2 к;
- ліцензія: MIT;
- кількість сайтів: 111 481;
- документація: добре;
- налаштування: Базовий GUI Customizer;
- основна концепція: RWD, Mobile-first, semantic;
- сітка: XY 12- Колонна сітка, Floted (flexbox в останній версії) ;

- підтримка браузера: Chrome 35+, Firefox 31+, Safari 9+, Opera, Edge, IE 11+;
  - компанія використовує: Avhana Health, Mid Day, Architonic та багато інших.
- Він доступний кожному і його легко забрати швидко.

#### 2.1.4 Material Design Lite

Material Design Lite – це бібліотека компонентів інтерфейсу користувача, створена за допомогою CSS, HTML і JavaScript. Це дає змогу додавати до ваших веб-сайтів зовнішній вигляд і відчуття Material Design . Крім того, він не покладається на жодних фреймворків JavaScript і має на меті оптимізацію для використання на різних пристроях, витончену деградацію в старіших браузерах і запропонувати відразу доступний досвід. Ви можете використовувати компоненти для створення привабливих, послідовних і функціональних веб-сторінок і веб-програм. Сторінки, розроблені за допомогою MDL, зможуть підтримувати всі сучасні принципи веб-дизайну, такі як портативність браузера, витончена деградація та незалежність від пристрою.

Бібліотека компонентів MDL пропонує нові версії звичайних елементів керування інтерфейсом користувача, таких як кнопки, текстові поля та прапорці, що відповідає концепціям Material Design. Бібліотека також містить розширені та спеціалізовані функції, такі як картки, макети стовпців, повзунки, типографіка, вкладки тощо. MDL можна безкоштовно завантажувати та використовувати, і може використовуватися з будь-якою бібліотекою чи середовищем розробки або без неї (наприклад, Web Starter Kit). Це кросбраузерний набір інструментів веб-розробника між ОС.

Ви можете перевірити Material Design Lite на Github.

Причини використання Material Design Lite:

- створений Google, MDL є сучасним, простим у використанні, має широкий охоплення функцій і не має зовнішніх залежностей;
- важливою перевагою є те, що MDL можна використовувати з Elm , мовою графічних інтерфейсів користувача;

- MDL забезпечує чудовий готовий вигляд, який, можливо, не потребує налаштування;
- завдяки шаблонам для ведення блогів Material Design Lite дає змогу розпочати блог за лічені хвилини;
- MDL надає багатий набір компонентів , у тому числі кнопки дизайну матеріалів, текстові поля, підказки, спінери та багато іншого.

Додаткова інформація:

- дата виходу: 19 червня 2014 року;
- Git star, Forks, Учасники : 31,7k, 5,3k, 345;
- Hacker News, Reddit, Stack overflow: h, 197, 648;
- ліцензія: Apache-2;
- кількість сайтів: 74 521;
- документація: добре;
- основна концепція: використання різних пристроїв;
- крива навчання: помірна;
- сітка: 12: Робочий стіл: 12 настільний, 8- планшет, 4- телефон;
- підтримка браузера: IE9, IE10, IE11, Chrome, Opera, Firefox, Safari, Chrome (Android), Mobile Safari;
- компанії, які використовують : Google Wallet , Google Project Sunproof, Talks at Google.

Ідеально підходить: для всіх, хто хоче писати більш продуктивні, портативні та, головне, зручні для використання веб-сторінки, оскільки MDL, по суті, робить концепцію доступною для плавного та легкого ознайомлення.

### 2.1.5 Bulma

Bulma – це адаптивний сучасний фреймворк CSS. Ця структура є вбудованим HTML, SASS CSS Prospector і CSS flexbox. Bulma надає безліч варіантів налаштування відповідно до ваших вимог за допомогою файлів sass,

веб-пакетів і змінних. Vulma створена у чистому CSS., що означає, що під час використання фреймворка вам потрібен лише один файл .css і жодного .js.

Цей фреймворк має деякі надзвичайно розширені функції, які допоможуть вам зробити ваш веб-сайт більш привабливим і менше коду. Ви можете використовувати функції утиліти для створення темних і світлих кольорових візерунків. Він має ті ж сітки, що й bootstrap. Vulma дозволяє додати модульність SASS. Він сумісний як з Font Awesome 4, так і з Font Awesome 5 завдяки елементу .icon.

Причини використання Vulma:

- Vulma пропонує чисті та прості пресети, які полегшують вибір відповідно до тем, які хоче досліджувати розробник;

- Vulma надає пристойну кількість веб-компонентів, з яких можна просто вибрати та використовувати їх для проектування відповідно до вимог і модифікацій;

- сторінка Vulma на GitHub має понад 1000 комітів і 600 учасників.

Додаткова інформація:

- дата виходу: 24 січня 2016 року;

- Git star, Forks , Учасники : 41k, 3,5k, 655;

- Reddit, переповнення стеку: 1,2k, 581;

- ліцензія: MIT;

- кількість сайтів: 30 987;

- документація: добре;

- налаштування: Базовий GUI Customizer;

- основна концепція: RWD, мобільний першим, сучасний безкоштовно;

- сітка: Проста побудова макета колон;

- підтримка браузера : новітні версії Chrome, Edge, Firefox, Opera, Safari, IE 10+ (частково підтримується);

- компанії, які використовують: Dev Tube , Economist , Rubrik та багато інших.

Ідеально підходить для: від новачка до професіонала, будь-який розробник може використовувати його завдяки його простоті.

### 2.1.6 Foundation

Foundation – це розширений інтерфейсний фреймворк CSS, вбудований HTML, CSS, SASS і javascript. Цей фреймворк має компілятор sass із швидким способом створення веб-сайту. Foundation має власний інтерфейс командної команди, який можна встановити на вашому комп'ютері/ноутбуці за допомогою певних команд, і ви можете легко дивитися. Подібна така сама структура файлів, як bootstrap, Bulma і materialize CSS. Це CSS-фреймворк для мобільних пристроїв, повністю адаптований до компонентів.

Доступно багато форумів для підтримки та швидкого вирішення будь-яких проблем. Здебільшого цей фреймворк використовується для великих веб-додатків, щоб створити дивовижний веб-сайт із початковим шаблоном та створити чудовий веб-сайт із привабливим інтерфейсом користувача. Він семантичний, читабельний, гнучкий і повністю налаштований. Фонд має вичерпну документацію та відеоуроки на офіційному сайті ZURB Foundation.

Причини використання Foundation:

- Foundation – це найсучасніший фреймворк CSS, який дозволяє користувачам створювати великі веб-додатки та багато іншого;
- на сторінці GitHub Foundation показано майже 2000 учасників і 17 000 комітів;
- він модульний і складається в основному з таблиць стилів Sass;
- він постійно оновлюється для підтримки новітніх функцій, таких як сітки з підтримкою flexbox.

Додаткова інформація:

- дата виходу: 4 листопада 2014 року;
- Git star, Forks, Учасники: 28,6k, 5,8k, 2045;
- Reddit, переповнення стека: 1,2k, 803;
- ліцензія: MIT;
- кількість сайтів: 441 292;
- документація: добре;

- налаштування: Advanced GUI Customizer (для попередньої версії);
- основна концепція: RWD і мобільний в першу чергу;
- сітка: стандартна сітка з 12 колонками, що реагує на рідину;
- підтримка браузера: останні дві версії Chrome, Firefox, Safari, Opera, Mobile Safari, IE Mobile, Edge та IE 9+, браузер Android 4.4+;
- компанії, які використовують: Amazon, Hp, Adobe, Mozilla, EA, Disney та багато інших.

Ідеально підходить для: професійних, висококваліфікованих розробників і дизайнерів, які прагнуть створити унікальний веб-сайт і хочуть налаштувати фреймворк.

### 2.1.7 Skeleton

Skeleton – це невелика колекція файлів CSS, яка може допомогти вам швидко розробляти сайти, які виглядають красиво будь-якого розміру, будь то 17-дюймовий екран ноутбука чи iPhone. Це інструмент для швидкого розвитку. Ви можете швидко розпочати роботу з найкращими методами CSS, добре структурованою сіткою, яка полегшує розгляд мобільних пристроїв, з організованою структурою файлів та базовими елементами інтерфейсу, як-от легко оформлені форми, кнопки, вкладки тощо.

Причини використання Skelton:

- легкий;
- відповідна сітка;
- Vanilla CSS;
- медіа-запити.

Додаткова інформація:

- дата виходу: 15 травня 2020 року;
- Git star, Forks, Учасники: 17,8k, 3k, 41;
- Hacker News, Reddit, Stack overflow: 1.1k, 799, s;
- ліцензія: MIT;

- основна концепція: RWD і мобільний в першу чергу;
- сітка: 12-стовпкова рідка сітка;
- підтримка браузера: остання версія Chrome, остання версія Firefox, остання версія Opera, остання версія Safari, пізній IE.

Ідеально підходить для: початківців, оскільки це найпростіший каркас.

### 2.1.8 Semantic UI

Semantic UI побудований навколо унікальної мети створення спільного словника навколо інтерфейсу. Заснований на принципах природної мови, Semantic надає розробникам і розробникам можливості, роблячи код більш читабельним і легшим для розуміння. Користувачі кажуть, що з семантичним інтерфейсом легко працювати і він має сенс.

Семантичний інтерфейс користувача виділяється функціональністю, яка виходить за рамки CSS і включає спрощене налагодження, а також можливість визначати елементи, колекції, уявлення, модулі та поведінку елементів інтерфейсу користувача.

Причини використання семантичного інтерфейсу користувача:

- Semantic UI пропонує дуже добре організовану документацію. Крім того, фреймворк має окремий веб-сайт із посібниками для початку роботи, налаштування та створення тем;
- усі класи семантичного інтерфейсу є людськими словами, а кодування нагадує написання звичайного тексту. Цей зручний підхід полегшує сприйняття та розуміння фреймворку навіть новачкам.

Додаткова інформація:

- дата виходу: 26 вересня 2013 року;
- Git star, Forks, Учасники: 48,4k, 5,1k, 391;
- Hacker News, Reddit, Stack overflow: 1,5k, 897, 2,6k;
- ліцензія: MIT;
- кількість сайтів: 124 579;

- документація: добре;
- основна концепція: семантичний тег, амбівалентність, адаптивний;
- крива навчання: помірна;
- сітка: Тема за замовчуванням: 16 стовпців;
- підтримка браузера: останні 2 версії FF, Chrome, Safari Mac, IE 11+, Android 4.4+, Chrome для Android 44+, iOS Safari 7+, Microsoft Edge 12+;
- компанії, які використовують: Snapchat, ESPN, Avira Lingo.

Ідеально підходить для: просунутих розробників, які добре знають JavaScript.

### 2.1.9 Pure CSS

Pure – це набір невеликих модулів CSS, які відповідають всім вашим потребам. Розмір Pure неймовірно малий – лише 3,8 КБ. Крім того, якщо ви вирішите використовувати лише підмножину доступних модулів, ви заощадите ще більше пропускну здатності. Він побудований на Normalize.css, Pure надає макет і стилі для власних елементів HTML, а також найпоширеніші компоненти інтерфейсу користувача. Його мінімальні стилі заохочують вас писати свої стилі додатків поверх нього.

Причини використання Pure CSS:

- дизайн Pure полегшує перевизначення стилів. Його мінімалістичний вигляд дає дизайнерам основу, на якій вони можуть будувати свій дизайн. Тим не менш, Pure надзвичайно легко налаштувати;
- каркас дуже простий. Назви класів легко запам'ятовувати, розширювати та підтримувати.

Додаткова інформація:

- дата виходу: 15 травня 2013 року;
- Git star, Forks, Учасники: 21,1k, 2,2k, 105;
- хакерські новини, Reddit, переповнення стека: 825, 698;
- ліцензія: Yahoo;

- кількість сайтів: 11 900;
- документація: добре;
- основна концепція: SMACSS, мінімалізм;
- крива навчання: помірна;
- сітка: сітка на основі одиниць 5ths 24ths;
- підтримка браузера: IE 10+, остання стабільна версія: Firefox, Chrome, Safari, iOS 12+, Android 6+;
- компанії, які використовують: Yahoo, LastPass, Flickr, CanYouSeeMe.org.

Ідеально підходить для: для тих, кому не потрібен повнофункціональний фреймворк, а лише конкретні компоненти для включення в свою роботу.

### 2.1.10 UI Kit

UI Kit – це легкий модульний інтерфейсний фреймворк CSS і веб-інтерфейсу, який пропонує майже всі основні функції інших фреймворків. Набори інтерфейсу користувача містять багато попередньо вбудованих компонентів, таких як Accordion, Alert, Drop, Iconnav, анімації, Padding тощо, які показують шаблони використання, параметри компонентів і методи.

UI Kit допомагає веб-розробникам створювати чисті та сучасні інтерфейси. Він пропонує вражаючі функції, особливо коли справа доходить до дизайну, немає конкуренції з UI kit.

По суті, UIKit – це майбутнє розробки додатків на платформах Apple.

Причини використання UI Kit:

- легкий і модульний інтерфейсний фреймворк для розробки швидких і потужних веб-інтерфейсів. UI Kit визначає основні компоненти, такі як кнопки, мітки, контролери навігації та подання таблиць;

- набір інтерфейсу користувача містить попередньо вбудовані компоненти, такі як Drop, Alert, Accordion, Padding, Iconnav, анімації тощо;

- це допомагає розробляти чутливі, потужні та швидкі веб-інтерфейси. Він складається з повної колекції компонентів CSS, HTML і JS;

- GitHub набору інтерфейсу містить більше 4000 комітів;
- він розширюється, простий у налаштуванні та простий у використанні.

Додаткова інформація:

- дата виходу: 19 липня 2013 року;
- зірка Git, Forks, Учасники: 15,7k, 2,2k, 51;
- Hacker News, Reddit, Stack overflow: 139, 21, 8.2k;
- ліцензія: MIT;
- кількість сайтів: 311 897;
- документація: добре;
- налаштування: Базовий GUI Customizer;
- основна концепція: адаптивний веб-дизайн, орієнтований на UX;
- сітка: сітка, гнучкість і ширина, межа між стовпцями сітки;
- підтримка браузера: останні версії Chrome, Firefox, Opera, Edge і Safari 9.1,+ IE 11+;
- компанія використовує: Crunchyroll, LiteTube, Rover.com та багато інших.

Ідеально підходить для: професійних і досвідчених розробників через відсутність доступних навчальних ресурсів.

## 2.2 Метод експертних оцінок

### 2.2.1 Зміст методів експертних оцінок

Вибір найбільш прийняттого методу прогнозування визначається наявною інформацією, особливостями об'єкта прогнозування та іншими чинниками.

Формалізовані раціональні методи прогнозування (екстраполяція тенденції, моделювання та ін.) базуються на достатньо розгорнутій ретроспективній інформаційній базі.

Однак, в умовах високого ступеня гостроти конкуренції на внутрішньому і особливо зовнішньому ринках, необхідне підвищення

гнучкості та адаптивності підприємств до зовнішнього середовища, стрижневим напрямком проведення ефективної економічної політики є орієнтація на інноваційні процеси.

Основоположник теорії інновацій Й. Шумпетед визначив такі основні групи інновацій (нововведень): продуктивна, технологічна, управлінська або організаційна.

Сьогодні додатково виділяються також економічні, соціальні, ринкові та інші групи інновацій.

Таким чином інновації охоплюють практично всі основні сфери суспільного життя.

При створенні нововведень слід врахувати, що зміни відбуваються особливо часто, тому рівень новизни надзвичайно високий.

З огляду на вищевикладене в процесі розробки нововведень виникає ціла низка проблемних завдань, зокрема:

- визначення перспектив створення нововведень;
- відбір основних параметрів нововведень;
- визначення сфери використання нововведень;
- оцінка оптимальних характеристик нововведень;
- оцінка порівняльної важливості характеристик нововведень.

Відтак, надзвичайний рівень новизни, пов'язаний з інноваційними процесами, з одного боку, обмежує можливість одержання достатньої ретроспективної інформації, необхідної при застосуванні формальних методів прогнозування, а, з іншого боку, нагальна потреба в аналізі і прогнозуванні якісних змін, робить названі методи цілком непридатними.

Особливо складні проблеми виникають, коли необхідно дати перспективні оцінки якісно новим процесам і явищам, які раніше не зустрічалися в суспільному житті і про які, природно, відсутня будь-яка інформація.

До сказаного вище слід додати, що кількісні методи не дозволяють передбачити можливі випадкові зміни у поведінці явищ (процесів, об'єктів), які у нинішніх умовах відбуваються частіше ніж у минулому.

Однак і в умовах обмеженості чи взагалі відсутності початкової інформації розробка прогнозів не тільки не виключається, а, навпаки, стає особливо актуальною і практично важливою, тому що лише таким чином можна дещо знизити рівень невизначеності та ризику і відповідно підвищити достовірність управлінських рішень.

В таких умовах повсюдно стала гостріше усвідомлюватися практична необхідність інших методів аналізу і прогнозування, які б не базувались винятково на статистичних даних. Можливість вирішення названих проблем, навіть в умовах відсутності теоретичних обґрунтувань, досягається за рахунок умілого використання досвіду, інтуїції та знань спеціалістів, вчених, що працюють над розв'язанням відповідної проблематики: науково-дослідні роботи, впровадження розробок і т.д.

Використання інтуїтивно-логічного аналізу обумовило розвиток і вдосконалення ряду методик, які дозволили значно поглибити дослідження проблем без їх кількісної формалізації.

Методи, які ґрунтуються на припущенні про те, що на базі думок спеціалістів у певній галузі знань можна побудувати адекватну картину майбутнього розвитку з урахуванням всіх можливих зсувів і стрибків отримали назву методів експертиз або методів експертних оцінок.

«Експерт» в дослівному перекладі з латинської мови означає «досвідчений». Тому метод експертних оцінок полягає в обробці інформації, отриманої шляхом опитування експертів. Використання експертів як джерел інформації про майбутній розвиток досліджуваного процесу (явища, об'єкта), ґрунтується на гіпотезі наявності бодай у частини провідних спеціалістів конкретної області глибоких і достатніх знань про шляхи розв'язання досліджуваних проблем.

Доречно зазначити, що широке використання методів експертних оцінок в сучасній методології прогнозування «Форсайт» обумовлено тим, що управлінці мають справу з «живим життям», а не з стабільною системою. Таке твердження пояснюється тим, що статистичний апарат прогнозування, який

використовується в економічних дослідженнях, виник із методології експериментальної фізики – зокрема, теорії експерименту.

У сучасному світі внаслідок науково-технічного прогресу особливу увагу у прогнозуванні і стратегічному аналізі звертають на якісні аспекти змін. Тенденції зростання цих змін обумовили, як свідчить огляд літературних джерел, збільшення питомої ваги експертних методів прогнозування приблизно до 40–50 %.

Отже, зростання якісних змін сприяє розвитку і вдосконаленню експертних методів прогнозування і водночас звужує можливість використання точних розрахунків, які ґрунтуються на формалізованих моделях, що відображають різноманітність причинно-наслідкових зв'язків еволюції явищ (процесів, об'єктів).

Підсумовуючи сказане, можна зі всією визначеністю стверджувати, що методи експертних оцінок в прогнозуванні використовуються в таких випадках:

- в умовах відсутності достатньої за обсягом та достовірної ін-формації про прогнозовані явища (процеси, об'єкти);
- в умовах значної невизначеності середовища, де функціонує об'єкт;
- в умовах дефіциту часу або екстремальних ситуаціях;
- при розробці середньо- та довгострокових прогнозів об'єктів, які підпадають під вплив корінних змін, наприклад, наукові відкриття.

Досвід засвідчує, що чим складніша досліджувана проблема, тим більше необхідності опиратися не оцінки експертів, використовуючи інтуїтивно-логічний аналіз. Все це обумовило подальший розвиток і удосконалення зазначеного аналізу, підвищення рівня його надійності. З цією метою не протязі останніх десятиріч були розроблені ряд прийомів і методик, що сприяли значному поглибленню дослідження проблем, які не піддавались кількісній формалізації. Діапазон таких методик досить широкий – від відбору експертів, до розробки і вдосконалення методів оброблення даних анкетного опитування.

В основі використання експертних методів лежать глибокі знання спеціалістів, вміння узагальнити свій та світовий досвід досліджень і розробок

по певній проблемі, гіпотеза про наявність у експерта так званої «практичної мудрості», далекоглядності, що стосується певної області знань і практичної діяльності.

При організації і проведенні процедури анкетного опитування слід дотримуватися певних принципів, зокрема:

- до експертного опитування слід долучити знаних, висококваліфікованих спеціалістів у відповідних областях знань;
- думки спеціалістів повинні бути подані у такій формі, що дозволяє їх систематизувати;
- велика кількість висловлених ідей неодмінно включає певну кількість плідних ідей;
- дотримуючись закону великих чисел, думки достатньо великого числа експертів слід вважати точними характеристиками досліджуваної проблеми;
- отримання відповідей експертів у максимально систематизованій формі можливе лише за умови чітко визначеного завдання і конкретно заданих питань;
- при наданні експерту нової інформації він повинен уміти творчо використати її і удосконалити відповіді у напрямку підвищення вірогідності і точності оцінок.

Кожен експерт створює свою інтуїтивну модель досліджуваного явища, що дозволяє йому за певних умов формувати наближені кількісні оцінки.

Чим складніші і суперечливіші явища, що досліджуються, тим більша відповідальність за наслідки прийнятих рішень і тим, відповідно, нагальніша необхідність при їх підготовці використовувати не тільки якісний аналіз, але і залучити кількісні показники з метою зіставлення альтернативних варіантів, що врешті-решт дозволить вибрати оптимальне управлінське рішення.

Прогнозні експертні оцінки відображають індивідуальне судження спеціалістів щодо перспектив розвитку об'єкта і ґрунтуються на мобілізації професійного досвіду доповненого знаннями та інтуїцією.

Отримані в результаті анкетного опитування якісні і кількісні показники, після відповідної обробки можуть бути використані для обґрунтування управлінських рішень. Перевага експертних методів полягає у швидкості отримання інформації про досліджуваний об'єкт для обґрунтування управлінських рішень за неможливості виміряти параметри і характеристики об'єкта кількісними методами.

Крім того, слід звернути увагу на той факт, що саме експерт є фахівцем, який глибоко розуміє зміст і особливості «вузьких місць» і стан досліджуваної проблеми. В результаті лише висококваліфікований спеціаліст може дати найбільш точний прогноз розвитку досліджуваного процесу.

І нарешті, безсумнівно, слід віддати належне експертним оцінкам у тому, що вони універсальні за своїм змістом і можуть бути застосовані для різних об'єктів прогнозування, відносно прості з методичної точки зору, не пред'являють підвищених вимог до якості початкової продукції. В деяких випадках, як це зазначалось, прогнозування може здійснюватися в умовах обмеженої інформації і навіть повної відсутності її.

### 2.2.2 Класифікація видів експертних оцінок та їх коротка характеристика

Методи, які використовуються для одержання експертних оцінок, достатньо численні і різноманітні. Тому для їх систематизації доцільно провести класифікацію окремих груп і видів експертного опитування.

Методи, що ґрунтуються на використанні експертних оцінок, діляться на дві групи: індивідуальні (персональні) експертні оцінки та групові (колективні) експертні оцінки.

Поділ на методи індивідуальних та колективних експертних оцінок проводиться в залежності від того, розробляється прогноз на основі висновків окремих ізольованих один від одного експертів, або групи експертів, які певним чином зв'язані між собою.

Отже, якщо методи індивідуальних експертних оцінок ґрунтуються на виявленні індивідуальних думок про розвиток досліджуваного об'єкта (процесу, явища), то метод колективних експертних оцінок базується на виявленні колективних думок про перспективи розвитку об'єкта прогнозування.

Індивідуальність опитування полягає в тому, що експерти не збираються разом; незнайомі з оцінками інших експертів; різних експертів можуть опитувати відносно різних аспектів однієї проблеми; нарешті, опитування різних експертів може проводитись за різними процедурами.

Від прогнозування за методом експертних оцінок слід відрізнити, так зване, прогнозування, яке широко застосовується в соціології, політології, маркетингу та інших сферах. Таке прогнозування базується на репрезентативних даних, отриманих в результаті опитування респондентів у випадковому порядку.

Виділяється декілька методів індивідуальних експертних оцінок: інтерв'ю, метод парного зривняння і аналітичні записки.

Метод інтерв'ю передбачає бесіду організатора експертизи (прогнозиста) з спеціалістом-експертом у певній галузі знань, що проводиться у відповідності за заздалегідь розробленою програмою. Прогнозист ставить перед експертом питання відносно перспектив розвитку об'єкта прогнозування. В процесі проведення індивідуального опитування програма дослідження може неодноразово корегуватися в наслідок отримання нової інформації на проміжних етапах дослідження.

Теоретично, при індивідуальному опитуванні може бути задіяний лише один експерт за умови, що рівень його знань дозволить інформаційно забезпечити потреби організаторів експертизи по досліджуваній проблематиці. Однак, зазвичай до опитування з метою підвищення надійності експертизи залучають групу експертів.

Ступінь формалізації інтерв'ю по певній проблематиці може бути різною. Низький рівень формалізації опитування – неформальна бесіда, в результаті якої визначається тільки тема проблеми. Експерт у такому випадку

самостійно вирішує як її висвітлювати. При необхідності, організатор експертизи може задати експерту уточнюючі або навідні питання.

Високий рівень формалізації передбачає чітко листа з питаннями відкритого типу. Цей метод у порівнянні з попереднім складніший як на етапі проведення опитування, тому що вимагає високої кваліфікації інтерв'ю, так і на етапі інтерпретації отриманої інформації – вимагає високої кваліфікації дослідника.

Успіх експертизи за методом інтерв'ю в значній мірі визначається здібністю експерта експромтом давати відповіді на найрізноманітніші складні, фундаментальні питання про перспективи розвитку досліджуваного об'єкта (процесу, явища). Суттєвий недолік зазначеного методу – недостатність часу для експерта на підготовку відповідей.

Метод аналітичних записок проводиться у письмовій формі (анкета) шляхом надсилання експерту питань з зацікавленої проблематики, на які повинні бути отримані однозначні відповіді. Питання можуть бути як відкритого, так і закритого типу. В останньому випадку повинні бути запропоновані варіанти відповідей. Анкета може бути відправлена по звичайній або електронній пошті, однак заздалегідь повинна бути попередня домовленість з експертом. Цей метод передбачає високий рівень кваліфікації організаторів експертизи на етапі постановки питань і організації проведення опитування, а також в частині обробки одержаної інформації.

На відміну від методу інтерв'ю метод аналітичних записок надає можливість експерту на проведення тривалої і ретельної роботи над аналізом тенденцій, оцінкою стану і шляхів розвитку прогнозованого об'єкта. Цей метод дозволяє експерту використати всю необхідну йому інформацію про об'єкт прогнозування. Свої міркування і висновки експерт оформляє у вигляді аналітичної записки.

Основними перевагами розглянутих методів є можливість максимального використання потенційних можливостей експертів та незначний психологічний тиск, який чиниться на спеціалістів.

Суттєвим недоліком методу індивідуальних експертних оцінок є те, що не кожний експерт бере на себе відповідальність самостійно дати оцінку складним явищам (процесам, об'єктам) без урахування думок інших експертів. Власне відсутність наукових зв'язків між експертами, обмеженість знань окремих спеціалістів робить розглянуті методи мало придатними для прогнозування найбільш складних загальних стратегій.

Вироблення ефективних управлінських рішень в умовах невизначеності, або складання науково-технічного прогнозу потребує участі групи ерудованих спеціалістів, добре обізнаних у багатьох галузях знань. Один спеціаліст не в змозі врахувати всі фактори і взаємозв'язки складних проблем, або оцінити ймовірність великого числа можливих альтернатив.

Методи колективних експертних оцінок ґрунтуються на принципах виявлення колективної думки експертів про перспективи розвитку об'єкта прогнозування.

Гіпотеза про наявність у експертів знань і умінь дозволяє їм з достатньою ступенню достовірності оцінити важливість і значення досліджуваної проблеми; перспективність розвитку певного напрямку дослідження; відносну важливість певних показників, параметрів, характеристик; терміну здійснення тієї чи іншої події; доцільність вибору одного із альтернативних шляхів розвитку об'єкта прогнозування і т.д.

У порівнянні з індивідуальними, групові методи, окрім іншого, мають переваги з точки зору надійності експертизи. Водночас вони досить складні в процесі їх підготовки і проведення.

Так, для розробки процедури групового оцінювання і гармонізації взаємодії між окремими ланками процесу опитування необхідні висококваліфіковані спеціалісти-експерти, яких далеко не завжди щастить зібрати в один і той же час, в одному і тому ж місці, в оптимальній кількості і які до того ж відповідали б встановленим критеріям.

Метод передбачає використання експерта, який проводить оцінку цілей:  
 $Z_1, Z_2, \dots, Z_n$ .

Відповідно до методу здійснюються парні порівняння цілей у всіх можливих поєднаннях. У кожній парі виділяється найкраща мета. І ця перевага виражається за допомогою оцінки за якоюсь шкалою. Обробка матриці оцінок дозволяє знайти всі цілі, що характеризують їх відносну важливість. Одна з можливих модифікацій методу полягає в наступному:

- складається матриця бінарних переваг, в якій перевага цілей виражається за допомогою булевих змінних;
- визначається ціна кожної мети шляхом підсумовування булевих змінних за відповідним рядком матриці.

До колективних методів експертних оцінок відносяться, насамперед, метод комісій і метод Дельфі.

У нинішній час широке розповсюдження отримали експертні методи, які ґрунтуються на роботі створених спеціальних комісій.

Суть методу комісії полягає в тому, що спеціалісти, які входять до однієї групи, погоджують свою думку про стан будь-якого об'єкта в майбутньому або шляхах і методах досягнення цілей у відкритій дискусії, найчастіше за круглим столом.

В ході проведення дискусії узгоджуються думки всіх експертів і розробляється загальний експертний висновок на основі консенсусу.

Колективна думка експертів в результаті дискусії визначається шляхом відкритого або таємного голосування. В деяких випадках до голосування не удаються, коли загальна думка виявляється в процесі дискусії.

Метод комісії має як свої переваги, так і недоліки. До переваг методу комісії можна віднести наступне.

По-перше, досвідом доведено, що сукупність інформації, якою володіють всі члени групи принаймні не менша обсягу інформації, якою володіє найбільш досвідчений експерт.

І якщо навіть і є спеціаліст, який в більшій мірі знайомий з об'єктом дослідження ніж решта частина групи, то все ж останні здатні зробити корисний внесок у прогнозу оцінку досліджуваної проблеми.

Як правило, до складу групи входять спеціалісти, які добре обізнані на певній проблематиці, тому їх сукупний обсяг знань перевищує кількість інформації, якою володіє будь-який із спеціалістів групи, інакше кажучи, група спеціалістів виробляє більше «розумової» енергії ніж один спеціаліст.

До того ж, в процесі дискусії, коли експерти обговорюють проблеми і аргументують свою точку зору, зазвичай, можливе зростання інформованості спеціалістів по досліджуваній проблематиці.

По-друге, загальновідомо, що кількість факторів (напрямків), що визначає розвиток об'єкта (процесу, явища), і які розглядаються всіма членами групи, по меншій мірі не менше тієї, яку може визначити будь-який член групи.

І нарешті по-третє, група експертів, принаймні, з більшою готовністю бере на себе відповідальність за прийняття важливих, ризикованих рішень, ніж окремих спеціаліст.

Незважаючи на простоту процедури опитування і широке розповсюдження, метод комісії має і принципові недоліки.

По-перше, частина групи може мати певний вплив на решту членів групи. Нерідко «криклива меншість», керуючись власними інтересами, може подавити більшість, і ті вимушені будуть погодитись під тиском наполегливості меншості, розуміючи при цьому, що аргументи останніх помилкові.

По-друге, суттєвим фактором, який впливає на узагальнюючі висновки, стає різна активність експертів групи. Дискусія в групі зводиться, зазвичай, до полеміки найбільш авторитетних експертів (знані спеціалісти, вченкорифеї, крупні керівники). І якщо названі експерти мають талант переконання, то вони в змозі рішуче впроваджувати свої ідеї шляхом наполегливої постійної аргументації, навіть при наявності у інших членів групи своїх протилежних переконань.

Крім того, публічність висловлювання може призвести до небажання окремих експертів відмовитися від раніше висловлених думок, навіть в умовах, коли вони зазнали змін в процесі дискусії.

По-третє, група експертів, як і будь-яка група, є самостійним організмом і функціонує завдяки цьому за певними законами. В групах існує думка, що досягти згоди більш важливо, а ніж розробка найобґрунтованішого і практично корисного прогнозу, або іншими словами, група у своїх судженнях керується в основному логікою компромісу, а не однією лише внутрішньою логікою досліджуваної проблеми. Все це пояснюється психологією групи експертів, а саме: схильністю окремих експертів періодично міняти свою точку зору; небажанням відкрито дебатовати; схильністю окремих експертів відстоювати один раз висловленим судженням, якщо вони навіть виявились помилковими, що стало очевидним самому експерту і т.д. При цьому не фіксуються думки і аргументи тих експертів, які не співпадають з думками більшості.

Метод комісії можливо поліпшити, якщо забезпечити безперешкодний обмін інформацією усередині групи і створити умови для вільного, незалежного висловлення суджень кожним експертом.

По-четверте, інерційність мислення, «честь мундира» переконує певну частину групи схилити решту членів до прийняття рішень, особливо якщо вони були орієнтовані на ці рішення з самого початку.

Прискорення науково-технічного прогресу, характерне для повоєнного періоду, обумовило потребу в нових, більш досконалих експертних методах прогнозування без притаманних методу комісії недоліків. І як це нерідко буває в сучасних умовах, на таку нагальну потребу природно, була дана відповідь: у другій половині ХХ ст. співробітниками науково-дослідної корпорації США «Rand Corporation» Т. Гор-доном і О. Хелмером був розроблений метод колективних експертних оцінок Дельфі, або Дельфійська процедура (термін «Дельфі» походить від назви грецького міста Дельфі і мудреців, які славились у давнину передбаченням майбутнього). З огляду на це, метод Дельфі спочатку розглядався виключно як метод прогнозування, однак згодом виявилось, що він має і достатньо значні аналітичні можливості.

### 2.2.3 Організація проведення експертного опитування

Ініціатором проведення експертизи виступає особа, що приймає рішення (керівник організації). Ця ж особа визначає цілі дослідження і орієнтовний ефект, який повинен бути досягнутий в результаті впро- вадження рекомендацій експертизи.

Особа, що приймає рішення, своїм розпорядженням підбирає і формує групу організаторів експертизи, далі – «Робочу групу», яка розробляє технічне завдання (програму) проведення експертного опитування.

Досвід свідчить, що для успішного проведення експертизи ініціатива повинна неодмінно виходити від керівництва організації, тому що тільки таким чином можна забезпечити вирішення загально організаційних і фінансових питань.

У технічному завданні на проведення досліджень повинна міститись розгорнута програма організації експертизи, яка включає:

- кадровезабезпеченнядослідження;
- відбір експертів та формування експертної групи в залежності від напрямку дослідження;
- методичне забезпечення дослідження;
- інформаційне забезпечення дослідження;
- програмне та технічне забезпечення дослідження;
- стимулювання членів експертної групи.

Кадрове забезпечення експертизи полягає у формуванні високоп- професійної робочої групи.

Керівництво організації підбирає наукового керівника і секретаря робочої групи; решту складу робочої групи формує науковий керівник за згодою особи, що приймає рішення.

На наукового керівника покладена відповідальність за організацію і проведення експертного опитування на всіх етапах експертизи, в тому числі за аналіз зібраних результатів опитування та формулювання висновків

експертної групи. Секретар робочої групи несе відповідальність за ведення документації і вирішення організаційних питань.

Для своєчасного і кваліфікованого проведення експертного опитування у складі робочої групи виділяються окремі ланки: аналітична, комп'ютерна, економетрична, організаційна.

Аналітична ланка розробляє детальний сценарій (тобто регламент) проведення збору і аналізу експертних оцінок. Сценарій повинен передбачити конкретні методи обробки і аналізу інформації, які виконують економетрична і комп'ютерна ланки робочої сили.

Аналітична ланка розробляє також методику опитування, підбір факторів, які характеризують досліджувану проблему.

Без ґрунтовної і ретельної підготовки експертизи має місце традиційна помилка поведінки персоналу робочої групи: спочатку зібрати інформацію, а потім думати як її використати. Результатом такого нехтування підготовчої роботи до проведення експертизи є те, що інформація, як свідчить досвід, використовується лише на 1–2 %.

Для того, щоб сприяти ефективному і своєчасному проведенню експертизи, робоча група заздалегідь знайомить всіх залучених до експертного опитування спеціалістів з цілями дослідження. Це дозволяє спеціалістам експертної групи умовно уявити собі правдоподібну модель досліджуваної проблеми, створеної на першому етапі дослідження, і створити умови для її подальшого удосконалення.

На стадії формулювання постановки задачі необхідно чітко визначити, в яких напрямках здійснюється експертиза: визначення терміну настання певної події; оцінка оптимальної величини певного параметра, характеристики об'єкта; виявлення відносної важливості окремих факторів, параметрів, напрямків розвитку тощо.

Суб'єктивний чинник багато в чому визначає точність, достовірність, надійність і, що вкрай важливо, практичну цінність експертизи. Це

пояснюється тим, що основним джерелом інформації є експерт, його судження, якісні і кількісні оцінки.

В процесі еволюції експертних оцінок вплив людського фактора не тільки знижується, а, як показав досвід, неперервно зростає. Звідси велика відповідальність лягає на робочу групу при формуванні складу експертної групи.

На стадії формування експертної групи організатори експертизи складають список експертів. З цією метою проводяться переговори з потенційними експертами, виявляються ті, хто по тим чи іншим причинам відмовляються від участі в експертизі.

Особа, що приймає рішення, затверджує склад експертної групи, поданий організаторами експертизи; при цьому можливе виключення із списку або навпаки додаткове включення частини експертної групи; укладаються договори з експертами про умови їх роботи та оплату праці.

В процесі проведення опитування можуть виникнути різні ситуації, на які повинні звертати увагу організатори експертизи.

Так, у деяких випадках спостерігається значна поляризація точок зору спеціалістів, що може пояснюватися, крім іншого, наявністю представників різних наукових шкіл у експертній групі.

Для перевірки готовності експертів до проведення експертизи робоча група нерідко проводить свого роду тренування.

Для проведення тренування використовується дві групи експертів, із яких одна – тільки що підібрана, без всякого досвіду; друга – складається із спеціалістів, які є найінформованими в досліджуваній проблемі, і члени якої неодноразово приймали участь у експертному опитуванні. Порівняння результатів опитування кожного експерта і кожної з груп дозволяє оцінити відносну ступінь надійності наново відібраних експертів. Як показник, для порівняння результатів можна використати середню групової думки: чим менше відхилення середньо групових показників, тим компетентніша

новостворена група експертів, і навпаки, якщо відхилення значні, то це свідчить про неготовність групи до участі в експертизі.

Тренування такого роду виконують ще одну важливу функцію: організатори експертизи на основі виявлених недоліків формування експертної групи можуть на останньому етапі відбору експертів внести при необхідності корективи щодо складу і чисельності групи.

Методичне забезпечення включає насамперед розробку опитувальних анкет, складання інструкцій по їх заповненню, сукупність моделей і методик, необхідних для розрахунку системи показників, які використовуються в анкетному опитуванні.

При організації опитувань слід взяти до уваги, що далеко не всі експерти в достатній мірі обізнані зі змістом та методикою використання експертних методів. Тому в момент, коли анкета вперше направляється експертам, до неї слід додати коротке описання суті методу та інструкцію по заповненню анкет.

Обробка анкет пов'язана з обчисленням системи показників, зміст яких дозволяє зробити остаточні, узагальнюючі висновки щодо реалізації поставленого завдання. Для цього необхідно, щоб серед організаторів експертизи були спеціалісти, які досконало володіють методикою обробки даних опитувальних анкет та змістом розрахованих показників.

Організація будь-якого дослідження передбачає формування добре розвинутої інформаційної бази, що складається з нормативно-довідкової та змінної інформації.

Нормативно-довідкова інформація – це словники, коди, класифікатори, різні інструктивні матеріали по формуванню, оформленню, заповненню та обробці анкет.

Змінна інформація – це, насамперед, дані, що містяться у заповнених анкетах.

На стадії передпрогнозної орієнтації, що передуює вибору та формулюванню задачі колективної експертної оцінки, здійснюється глибокий

всебічний аналіз стану та тенденції розвитку прогнозованого процесу (явища, об'єкта) в країні та за кордоном.

Джерелом інформації для проведення такого аналізу є :

- періодична література (статті у вітчизняних та закордонних журналах, збірниках та наукових бюлетенях, газетах);
- наукові звіти та огляди науково-дослідних та проектно-конструктивних організацій;
- наукові монографії, праці та дисертації;
- статистичні дані;
- патенти та авторські свідоцтва, раціоналізаторські пропозиції;
- нормативна-технічна документація: стандарти, технічні умови, норми і правила розрахунків та проектування;
- звіти про закордонні відрядження спеціалістів;
- дані про діяльність закордонних фірм;
- матеріали симпозіумів, семінарів, конференцій;
- інформація INTERNET.

Перелічені джерела інформації використовуються не тільки організаторами експертизи, але й експертами.

Реалізація методу експертних оцінок, як і іншого методу прогнозування, за окремими винятками, неможлива в широких масштабах і особливо в реальному режимі часу без відповідного програмного забезпечення та технічних засобів – обчислювальної техніки, засобів зв'язку, оргтехніки.

Програмні засоби повинні забезпечити формування та зберігання масивів інформації, видачу їх на друкувальні пристрої та монітори за графіком чи на запит.

Вдале проведення експертизи практично неможливе без стимулювання членів експертної групи – морального та матеріального. Моральне стимулювання включає, наприклад, гарантію для експертів на основі правових норм оформлення пріоритету та авторства на нові ідеї, які висунуті в процесі експертизи, включення в план роботи, співавторства в наукових звітах та ін.

Науково обґрунтований прогноз – це цінний товар, на розробку якого витрачається праця високопрофесійних спеціалістів. І як будь-який товар, він не може бути безкоштовним. Ось чому матеріальна зацікавленість експертів – важлива умова добросовісного проведення експертизи. Тим більше, як свідчить досвід, відсутність прогнозу чи поганий прогноз обійдеться споживачам прогнозної інформації дорожче.

Складність і різноманітність завдань, які покладені на робочу групу, пред'являють винятково високі вимоги до організаторів експертизи. Проблеми, які доводиться їм розв'язувати, вимагають, щоб рівень знань членів робочої групи принаймні не поступався знанням експертів, які залучені до експертного опитування. До того ж, організатори експертизи повинні заздалегідь всебічно вивчити зміст досліджуваної проблеми, уміти виявити взаємозв'язки між різними її аспектами, активно впливати на хід експертного опитування.

Діяльність робочої групи офіційно закінчується після затвердження особою, що приймає рішення, підсумкового документа, наукових і фінансових звітів про проведення дослідження, оплати праці експертів і співробітників робочої групи.

На завершення варто наголосити, що успішне проведення експертизи багато в чому залежить від рівня організації робіт з опитування експертів.

### 2.3 Проведення аналізу

Аналізуючи статистику, можна здійснити кількісну оцінку ефективності. Знайти найкраще рішення, спираючись на цифри, неможливо – вони дають лише загальний зріз. На відміну від стандартних метрик та критеріїв, парні порівняння допомагають виявити якісну, а не кількісну різницю у характеристиках. Це дозволяє швидше покращувати ресурс, не витрачаючи час збору достатнього обсягу даних. Саме тому, було вирішено проводити дослідження методом парного зрівняння.

Цей метод дослідження використовують у багатьох сферах, що з маркетингом, соціологією, психологією, дизайном, розробкою бізнес-моделей, політикою. Цілі теж можуть відрізнятись: знайти найкращий варіант, зібрати зворотний зв'язок чи вимоги клієнтів, провести ранжування елементів за пріоритетами тощо. У юзабіліті найчастіше метод парних зрівнянь застосовується для порівняння двох варіантів платформи.

Методика також дозволяє зібрати якісніші відгуки про дизайн. Якщо респондентові показати лише одну сторінку, він дасть загальну оцінку, не вдаючись до деталей. Коли перед очима два варіанти, людина точніше може передати свої емоції порівняно. Візуально вибрати те, що подобається більше, простіше, ніж шукати переваги та недоліки в одному місці.

Емоційний відгук – один із головних критеріїв юзабіліті, тому такий підхід допоможе проводити більш ефективні дослідження та швидше покращувати результати. Ця психологічна методика має й інші переваги:

- точне виявлення проблем, які можна легко перекласти на завдання для технічних фахівців;
- концентрація уваги на одному показнику, а не на всій сторінці відразу, що полегшує опрацювання;
- можна отримати як середню оцінку, а й розібратися у важливості чинника.

Для аналізу та зрівня було обрано наступні фреймворки:

- Bootstrap;
- Tailwind CSS;
- Materialize CSS;
- Material Design Lite;
- Bulma;
- Foundation;
- Skeleton;
- Semantic UI;
- Pure CSS;
- UI Kit.

Основними критеріями для зрівняння було визначено:

- кросбраузерність та кросплатформність(1);
- універсальність усіх випадків адаптивності(2);
- швидкість завантаження сайту з фреймворком(3);
- підтримка скриптових технологій (JavaScript, JQuery та ін.)(4);
- актуальна підтримувана версія фреймворку та повний комплект специфікацій(5);
- реалізація анімацій(6);
- чистота коду(7).

Оцінка методів за наведеними критеріями представлена у таблиці 2.1.

Таблиця 2.1 – Оцінка фреймворків

Критерій/ Фреймворк	1	2	3	4	5	6	7
Bootstrap	відмінно	відмінно	середнє	добре	відмінно	відмінно	відмінно
Tailwind CSS	відмінно	відмінно	відмінно	добре	відмінно	добре	добре
Materialize CSS	середнє	добре	середнє	відмінно	середнє	середнє	добре
Material Desidn Lite	добре	добре	середнє	погано	добре	середнє	відмінно
Bulma	добре	добре	середнє	відсутня	добре	середнє	середнє
Foundation	відмінно	відмінно	добре	середнє	добре	відмінно	середнє
Skeleton	середнє	відмінно	відмінно	відсутня	добре	добре	середнє
Semantic UI	добре	середнє	погано	відсутня	добре	добре	відмінно
Pure CSS	добре	добре	добре	відсутня	добре	середнє	добре
UI Kit	відмінно	відмінно	добре	відсутня	добре	середнє	середнє

Наступним етапом дослідження є побудова матриць зрівнянь фреймворків за кожним критерієм окремо. Матріці наведено у табл. 2.2-2.8.

Було побудовано матриці парних зрівнянь та визначено коефіцієнтів. Натомість, за допомогою коефіцієнтів за окремими критеріями було визначено загальний коефіцієнт для кожного фреймворка (табл. 2.9).

Таким чином дослідження показали, що найкращими фреймворками є Tailwind CSS та Bootstrap. Фреймворки, які не набрали більш ніж 13%, не розглядаються оскільки недостатньо відповідають поставленим критеріям. Результати дослідження представлені у вигляді діаграми на рисунку 2.1.

Таблиця 2.2 – Матриця зрівнянь за кросбраузерністю та кросплатформістю

Критерій 1	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	1	2	2	2	1	2	2	2	1	1,6
2 Tailwind CSS	1	1	2	2	2	1	2	2	2	1	1,6
3 Materialize CSS	0	0	1	0	0	0	1	0	0	0	0,2
4 Material Desidn Lite	0	0	2	1	1	0	2	1	1	0	0,8
5 Bulma	0	0	2	1	1	0	2	1	1	0	0,8
6 Foundation	1	1	2	2	2	1	2	2	2	1	1,6
7 Skeleton	0	0	1	0	0	0	1	0	0	0	0,2
8 Semantic UI	0	0	2	1	1	0	2	1	1	0	0,8
9 Pure CSS	0	0	2	1	1	0	2	1	1	0	0,8
10 UI Kit	1	1	2	2	2	1	2	2	2	1	1,6

Таблиця 2.3 – Матриця зрівнянь за універсальністю адаптивності

Критерій 2	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	1	2	2	2	1	1	2	2	1	1,5
2 Tailwind CSS	1	1	2	2	2	1	1	2	2	1	1,5
3 Materialize CSS	0	0	1	1	1	0	0	2	1	0	0,6
4 Material Desidn Lite	0	0	1	1	1	0	0	2	1	0	0,6
5 Bulma	0	0	1	1	1	0	0	2	1	0	0,6
6 Foundation	1	1	2	2	2	1	1	2	2	1	1,5
7 Skeleton	1	1	2	2	2	1	1	2	2	1	1,5
8 Semantic UI	0	0	0	0	0	0	0	1	0	0	0,1
9 Pure CSS	0	0	1	1	1	0	0	2	1	0	0,6
10 UI Kit	1	1	2	2	2	1	1	2	2	1	1,5

Таблиця 2.4 – Матриця зрівнянь за швидкістю завантаження сайту

Критерій 3	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	0	1	1	1	0	0	2	0	0	0,6
2 Tailwind CSS	2	1	2	2	2	2	1	2	2	2	1,8
3 Materialize CSS	1	0	1	1	1	0	0	2	0	0	0,6
4 Material Desidn Lite	1	0	1	1	1	0	0	2	0	0	0,6
5 Bulma	1	0	1	1	1	0	0	2	0	0	0,6
6 Foundation	2	0	2	2	2	1	0	2	1	1	1,3
7 Skeleton	2	1	2	2	2	2	1	2	2	2	1,8
8 Semantic UI	0	0	0	0	0	0	0	1	0	0	0,1
9 Pure CSS	2	0	2	2	2	1	0	2	1	1	1,3
10 UI Kit	2	0	2	2	2	1	0	2	1	1	1,3

Таблиця 2.5 – Матриця зрівнянь за підтримкою скриптових технологій

Критерій 4	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	1	0	2	2	2	2	2	2	2	1,6
2 Tailwind CSS	1	1	0	2	2	2	2	2	2	2	1,6
3 Materialize CSS	2	2	1	2	2	2	2	2	2	2	1,9
4 Material Desidn Lite	0	0	0	1	2	0	2	2	2	2	1,1
5 Bulma	0	0	0	0	1	0	1	1	1	1	0,5
6 Foundation	0	0	0	2	2	1	2	2	2	2	1,3
7 Skeleton	0	0	0	0	1	0	1	1	1	1	0,5
8 Semantic UI	0	0	0	0	1	0	1	1	1	1	0,5
9 Pure CSS	0	0	0	0	1	0	1	1	1	1	0,5
10 UI Kit	0	0	0	0	1	0	1	1	1	1	0,5

Таблиця 2.6 – Матриця зрівнянь за актуальною підтримуваною версією фреймворку та повним комплектом специфікацій

Критерій 5	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	1	2	2	2	2	2	2	2	2	1,8
2 Tailwind CSS	1	1	2	2	2	2	2	2	2	2	1,8
3 Materialize CSS	0	0	1	0	0	0	0	0	0	0	0,1
4 Material Desidn Lite	0	0	2	1	1	1	1	1	1	1	0,9
5 Bulma	0	0	2	1	1	1	1	1	1	1	0,9
6 Foundation	0	0	2	1	1	1	1	1	1	1	0,9
7 Skeleton	0	0	2	1	1	1	1	1	1	1	0,9
8 Semantic UI	0	0	2	1	1	1	1	1	1	1	0,9
9 Pure CSS	0	0	2	1	1	1	1	1	1	1	0,9
10 UI Kit	0	0	2	1	1	1	1	1	1	1	0,9

Таблиця 2.7 – Матриця зрівнянь за реалізацією анімацій

Критерій 6	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	2	2	2	2	1	2	2	2	2	1,8
2 Tailwind CSS	0	1	2	2	2	0	1	1	2	2	1,3
3 Materialize CSS	0	0	1	1	1	0	0	0	1	1	0,5
4 Material Desidn Lite	0	0	1	1	1	0	0	0	1	1	0,5
5 Bulma	0	0	1	1	1	0	0	0	1	1	0,5
6 Foundation	1	2	2	2	2	1	2	2	2	2	1,8
7 Skeleton	0	1	2	2	2	0	1	1	2	2	1,3
8 Semantic UI	0	1	2	2	2	0	1	1	2	2	1,3
9 Pure CSS	0	0	1	1	1	0	0	0	1	1	0,5
10 UI Kit	0	0	1	1	1	0	0	0	1	1	0,5

Таблиця 2.8 – Матриця зрівнянь за чистотою коду

Критерій 7	1	2	3	4	5	6	7	8	9	10	Коеф.
1 Bootstrap	1	2	2	1	2	2	2	1	2	2	1,7
2 Tailwind CSS	0	1	1	0	2	2	2	0	1	2	1,1
3 Materialize CSS	0	1	1	0	2	2	2	0	1	2	1,1
4 Material Desidn Lite	1	2	2	1	2	2	2	1	2	2	1,7
5 Bulma	0	0	0	0	1	1	1	0	0	1	0,4
6 Foundation	0	0	0	0	1	1	1	0	0	1	0,4
7 Skeleton	0	0	0	0	1	1	1	0	0	1	0,4
8 Semantic UI	1	2	2	1	2	2	2	1	2	2	1,7
9 Pure CSS	0	1	1	0	2	2	2	0	1	2	1,1
10 UI Kit	0	0	0	0	1	1	1	0	0	1	0,4

Таблиця 2.9 – Загальні коефіцієнти фреймворків

Критерій/Фреймворк	1	2	3	4	5	6	7	Загальний Коеф.
Bootstrap	1,6	1,5	0,6	1,6	1,8	1,8	1,7	1,514
Tailwind CSS	1,6	1,5	1,8	1,6	1,8	1,3	1,1	1,529
Materialize CSS	0,2	0,6	0,6	1,9	0,1	0,5	1,1	0,714
Material Desidn Lite	0,8	0,6	0,6	1,1	0,9	0,5	1,7	0,885
Bulma	0,8	0,6	0,6	0,5	0,9	0,5	0,4	0,614
Foundation	1,6	1,5	1,3	1,3	0,9	1,8	0,4	1,257
Skeleton	0,2	1,5	1,8	0,5	0,9	1,3	0,4	0,942
Semantic UI	0,8	0,1	0,1	0,5	0,9	1,3	1,7	0,771
Pure CSS	0,8	0,6	1,3	0,5	0,9	0,5	1,1	0,814
UI Kit	1,6	1,5	1,3	0,5	0,9	0,5	0,4	1,186

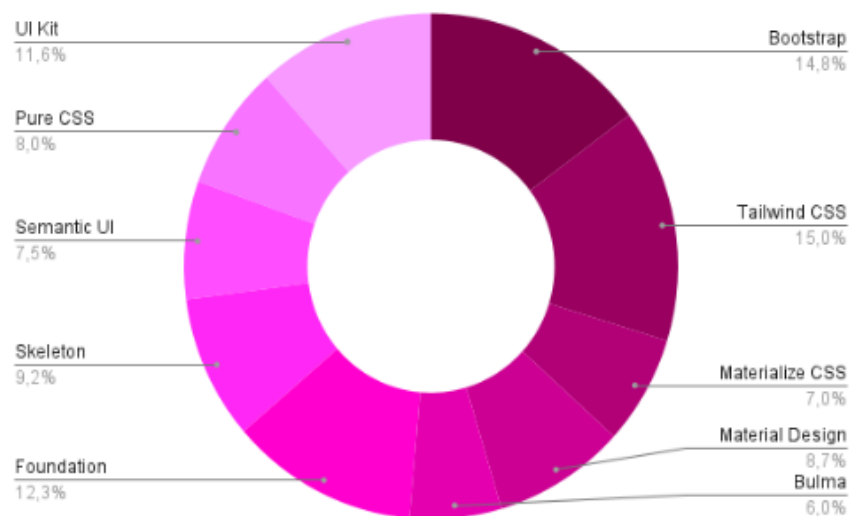


Рисунок 2.1 – Результати дослідження

## 3 ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

3.1 Вибір та обґрунтування програмного продукту для створення інформаційного сайту

Sublime Text – ідеально підходить професійним програмістам. Професійні користувачі цього інструменту знають, які плагіни підходять. Але для нового користувача, який тільки-но почав використовувати цей інструмент, перераховані плагіни допоможуть підвищити рівень продуктивності за рахунок більш швидкого кодування.

Sublime Text – це кросплатформовий текстовий редактор, розроблений для користувачів, які шукають ефективний, але мінімалістський інструмент для редагування коду. Редактор, звичайно, простий, в якому відсутні панелі інструментів або діалогові вікна.

Пильний погляд зсередини показує потужний текстовий редактор, в який завантажені плагіни, розумні функції автозаповнення та багато іншого.

Sublime Text пропонує багато функцій, які спрощують компіляцію коду.

Goto Anything – це зручна функція, яка дозволяє легко отримувати доступ до файлів. Для переходу до аспектів скомпільованого коду, таких як символи, рядки або слова, потрібно лише кілька дій.

Зіставлення дужок – це функція, що дозволяє швидко визначити неправильне зіставлення. Редактор безпосередньо виділяє відповідні набори дужок.

Множинне виділення – це зручний інструмент, який дозволяє швидко змінювати рядки коду на ходу. Змінюйте імена змінних або навіть файлів.

Наявність потужного Python API у Sublime виділяє текстовий редактор серед конкурентів. Потужний вбудований API дозволяє Sublime досягати більшої функціональності, дозволяючи плагінам розширювати вбудовану функціональність.

Кросплатформенність – Sublime Text доступний на декількох клієнтських комп'ютерах, включаючи Windows, Mac та Linux.

Переваги Sublime Text.

Sublime Text – це легкий текстовий редактор, який підійде будь-якому програмісту. Програма зроблена зі швидкістю, яка знаходиться в її основі. Особливість програми в її швидкості та чуйності користувальницького інтерфейсу.

У редакторі є безліч плагінів, які інтегруються в одному місці.

Повністю налаштований текстовий редактор створений, щоб дозволити кінцевому користувачеві легко «пограти» з ПЗ на свій лад. Sublime дозволяє налаштовувати безліч функцій, включаючи: прив'язки клавіш, меню, фрагменти, макроси та багато інших. Крім того, змінюйте зовнішній вигляд, настроївши свої теми для програмного забезпечення.

Кросплатформова підтримка - в редакторі доступна на більшості поширених настільних клієнтів, включаючи Windows, MacOS і Linux.

Sublime з відкритим вихідним кодом відповідно безкоштовний. Але в той же час, ПЗ можна купити - за бажанням. Важливо, що безкоштовна версія працює просто чудово.

З редактором ви можете комфортно перемикатися між різними файлами. До того ж завдяки функції Goto Anything, доступ до якої отримуєте безпосередньо з клавіатури за допомогою клавіш Ctrl або Command + P.

Простота у використанні. Редактор підходить для будь-якого користувача незалежно від рівня його досвіду.

Наявність плагінів.

Emmet – один з найпопулярніших плагінів Sublime Text 3, який завантажений понад 4 мільйони разів по всьому світу. Замість використання JavaScript цей плагін працює з CSS та HTML, що спрощує роботу користувачів.

В результаті це дозволяє додавати коди через сніпети, що значно прискорює весь процес для програмістів. Плагін вимагає базових знань HTML та CSS.

Sublime Text – гідно є одним з найпопулярніших текстових редакторів. Гідну конкуренцію редактору складає програмне забезпечення з відкритим вихідним кодом Atom.

### 3.2 Створення сайту за допомогою Bootstrap

Для реалізації експериментального дослідження було обрано дизайн банеру сайту-біографії про відомого художника Рубенса (рис. 3.1).



Рисунок 3.1 – Макет банеру сайту

На початку роботи необхідно підготувати усі необхідні файли, а саме: шрифти, зображення, графічні елементи. Було створено загальну папку де будуть знаходитися необхідні файли, у папці створено документ index.html. Саме у цьому документі було описано код сайту.

У програмі Sublime Text 3 було відкрито файл з сайтом. Для початку роботи необхідно прописати базовий код, та додати посилання безпосередньо на фреймворк.

За допомогою `@font-face` додаємо необхідні шрифти на сайт.

```
@font-face {
    font-family: "Eckhart-DisplayThin";
    src: url("Eckhart-DisplayThin.ttf");
}
@font-face {
    font-family: "Eckhart-DisplayExtraBold";
    src: url("Eckhart-DisplayExtraBold.otf");
}
```

Далі у `<body>` створюємо `<div>` з класом “`container`”, за допомогою цього класу на сторінці буде визначено контейнер розташований по центру та з відступами за замовчуванням. До елемента також додаємо клас “`col-lg-12`”, це означає що контейнер з контентом буде займати 12 колонок.

Наступним на сторінку необхідно додати фонове зображення. Додаємо посилання на зображення через стилі, а також вказуємо, що зображення повинно бути на всю сторінку, та не повторюватися.

```
body {
    background: url("image 1.jpg") no-repeat;
    background-size: cover;
    height: 100vh;
}
```

Наступним шагом є створення навігаційного меню. Робимо це за допомогою тега `<header>`. Вказуємо верхній відступ, шрифт та його розмір.

```
header {
    margin-top: 40px;
    font-family: Eckhart-DisplayThin;
    font-size: 14px;
}
```

Там розміщуємо логотип та навігацію сайту, які є посиланнями. За допомогою тегів `<a>` та `<img>` додаємо логотип на сайт.

```
<a href="#">
  
</a>
```

Для створення меню створюємо нумерований список `<ul>`, який містить у собі 3 теги `<li>`, які у свою чергу містять `<a>`: Home, About, Gallery.

```
<ul class="menu">
  <li><a href="#">Home</a></li>
  <li><a href="#">About</a></li>
  <li><a href="#">Gallery</a></li>
</ul>
```

До списку додаємо клас “d-flex”, що дозволить елементам меню розміщуватися в одну строку. До першого елемента `<a>` додаємо клас “active”, що буде означати активну сторінку. Надалі стилізуємо меню: прибираємо пункти, декорацію тексту, зменшуємо прозорість неактивних сторінок та додаємо відступи між елементами. Додаємо `hover`, щоб при наведенні на елементи прозорість збільшувалась до 100%.

```
li {
  list-style-type: none;
  padding-left: 25px;
}
ul a {
  text-decoration: none;
  color: white;
  opacity: 0.5;
}
a:hover {
  opacity: 1;
  color: white;
}
a.active {
```

```

        opacity: 1;
    }

```

До хедеру додаємо клас “d-flex”, що дозволить елементам хедеру розміщатися в одній строці. Далі додаємо клас “justify-content-between”, за допомогою якого внутрішні елементи сторінки будуть розміщуватися на різних кінцях сторінки. Таим чином було створено header сторінки, результат представлено на рисунку 3.2.



Рисунок 3.2 – header сторінки

Наступним шагом є додання заголовку та підзаголовку. Створюємо `<div>` з класом “main” та у ньому розміщуємо теги `<h1>` для заголовку та `<h2>` для підзаголовку. Для загального елемента додаємо класи “offset-lg-2”, “offset-md-1” та “offset-sm-1”. “offset-lg-2” означає, що на великих екранах відступ вмісту від початку буде дві колонки, “offset-md-1” та “offset-sm-1” означають, що на планшетах та телефонах відступ буде одна колонка. До елемента `<h1>` додаємо клас “col-lg-4” за допомогою якого ширина елемента займатиме чотири колонки. Теж саме з елементом підзаголовку, для різних пристроїв вказуємо різну ширину колонок.

```

<div class="main offset-lg-2 offset-md-1 offset-sm-1">
    <h1 class="col-lg-4 ">Rubens</h1>
    <h2 class="col-lg-5 col-md-8 col-sm-2">“I’m just a simple man
    standing alone with my old brushes, asking God for inspiration.”
    </h2>
</div>

```

За допомогою стилів вказуємо шрифт та його розміри для заголовку та підзаголовку, а також відступ від меню.

```

.main {

```

```
padding-top: 37vh;
}
h1 {
font-family: Eckhart-DisplayExtraBold;
font-size: 105px;
}
h2 {
font-family: Eckhart-DisplayDemiBoldItalic;
font-size: 22px;
}
```

Таким чином було створено основну частину сатйу. Результат представлено на рисунку 3.3.



Рисунок 3.3 – Основна частина сторінки

Останім етапом створення сторінки є footer. Для його створення додаємо тег <footer> у я кому буде розміщено дати народження та смерті художника, а також кнопку переходу. Дату додаємо за допомогою тега <p>, конпку – за допомгою тегів <a> та <img>. До тегу <a> було додано клас “text-end”, що дозволяє елементу розміщуватися наприкінці строки. До футера було додано

класи “d-flex”, “justify-content-between” та “align-items-end”, за допомогою якого елементи футера розміщуються унизу строки.

```
<footer class="d-flex align-items-end justify-content-between">
  <p class="col-lg-2">28 June 1577 - 30 May 1640</p>
  <a class="col-lg-1 text-end" href="#"></a>
</footer>
```

Використовуючи стилі, встановлюємо шрифт, розмір шрифту та відступи.

```
p {
  font-family: Eckhart-DisplayThin;
  font-size: 14px;
  line-height: 0px;
}
footer {
  margin-top: 22vh;
}
```

Для невеликої динаміки сатйу було додано анімацію на кнопку за допомогою @keyframes.

```
@keyframes move {
  from {
    transform: translateY(0%);
  }
  50% {
    transform: translateY(50%);
  }
  to {
    transform: translateY(0%);
  }
}
```

Анімацію названо “move”, оскільки вона описує рух та це єдина анімація на сторінці. На початковому етапі анімації рух кнопки за оссю Y буде дорівнювати 0%, на середині кнопка рухатиметься на 50%, на кінцевому етапі знову буде нерухомою. Для кнопки вказуємо створену анімацію, вказуємо її час, та постійність.

```
.footer a img{
    animation: move 3s infinite;
}
```

Таким чином було створено footer сайту. Результат представлено на рисунку 3.4.



Рисунок 3.4 – footer сайту

Нарикінці необхідно додати декілька медіазапитів, для коректного відображення на різних пристроях.

```
@media screen and (max-width: 768px){
    h1{
        font-size: 80px;
    }
    h2 {
        font-size: 18px;
    }
    .main {
        padding-top: 40vh;
    }
}

@media screen and (max-width: 576px){
    h1{
```

```

        font-size: 60px;
    }
    h2 {
        font-size: 14px;
    }
    .main {
        padding-top: 40vh;
    }
}

```

Результат готової сторінки представлено на рисунку 3.5. Адаптивність сайту під різні девайси представлена на рисунку 3.6.

Підводячи підсумки, можна сказати, що bootstrap відповідає практично усім необхідним критеріям, але більшість стилів все ж таки приходиться писати самостійно. Імпонує те, що фреймворк має сітку за замовчуванням і сам підстраює сайт під адаптивність. Але все ж таки має великий час завантаження сайту з фреймворком.



Рисунок 3.5 – готова сторінка сайту

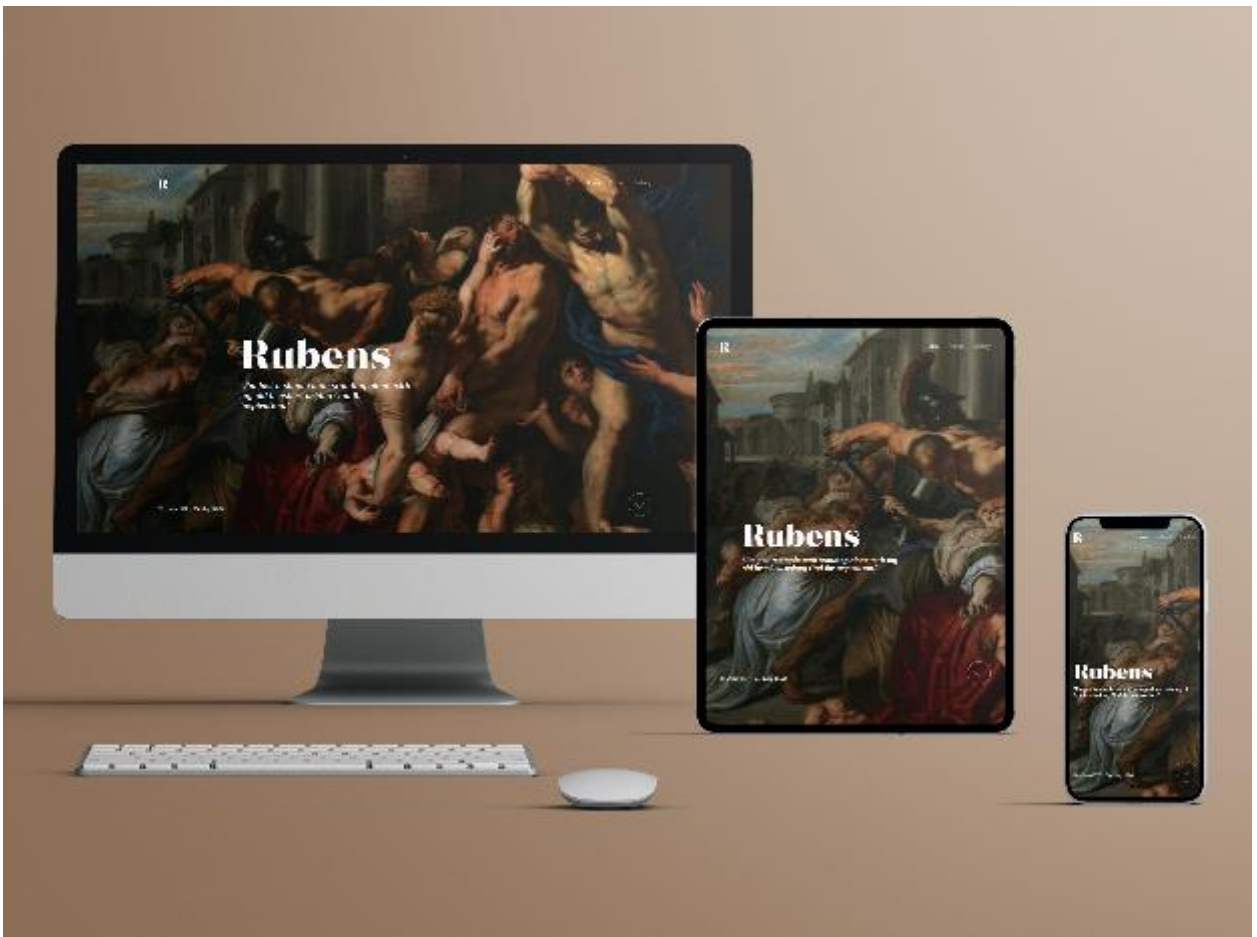


Рисунок 3.6 – Адаптивність сайтів на різних пристроях

### 3.3 Створення сайту за допомогою Tailwind CSS

На початку роботи необхідно підготувати усі необхідні файли, а саме: шрифти, зображення, графічні елементи. Було створено загальну папку де будуть знаходитися необхідні файли, у папці створено документ `index.html`. Саме у цьому документі було описано код сайту.

У програмі Sublime Text 3 було відкрито файл з сайтом. Для початку роботи необхідно прописати базовий код, та додати посилання безпосередньо на фреймворк.

```
<link href=https://unpkg.com/tailwindcss@^2/dist/tailwind.min.css
rel="stylesheet">
```

За допомогою `@font-face` додаємо необхідні шрифти на сайт.

```
@font-face {
```

```

font-family: "Eckhart-DisplayThin";
src: url("Eckhart-DisplayThin.ttf");
}
@font-face {
font-family: "Eckhart-DisplayExtraBold";
src: url("Eckhart-DisplayExtraBold.otf");
}

```

Далі у <body> додаємо фонове зображення за допомогою атрибута style. Додаємо настурні класи: “bg-no-repeat” щоб зображення не повторювалося, “bg-cover” щоб зображення займало увесь екран, “container” щоб було створено контейнер з контентом, “mx-auto” щоб контейнер розміщався у центрі, “xs:justify-center”, “text-white” щоб текст був білого кольору, “md:px-10 – відступи на планшетах”, “lg:px-10 – відступи на великих екранах” та “px-6 – відступи на мобільних пристроях”.

```

<body style="background-image: url('image 1.jpg');" class="bg-no-repeat
bg-cover md:h-screen h-screen container mx-auto xs:justify-center text-white
md:px-10 lg:px-10 px-6">
</body>

```

Наступним шагом є створення навігаційного меню. Робимо це за допомогою тега <header>. Вказуємо класи “flex” – для розміщення внутрішніх елементів в одній строці, “justify-between” – для розміщення елементів з різних сторін сторінки та “pt-6” – для верхнього відступу.

Там розміщуємо логотип та навігацію сайту, які є посиланнями. За допомогою тегів <a> та <img> додаємо логотип на сайт. До нумерованого списку додаємо класи “flex” та “text-sm”, вказуючи розташування елементів та розмір тексту. Для другого та третього елементів списку вказуємо класи “pl-8” – ліві відступи елементів, “opacity-50” – прозорість елементів та “hover:opacity-100” – при наведенні кнопкою миші на елемент непрозорість буде повною.

```

<header class="flex justify-between pt-6">

```

```

<div class="logo">
  <a href="#"></a>
</div>
<ul class="menu flex text-sm">
  <li><a href="#">Home</a></li>
  <li class="pl-8 opacity-50 hover:opacity-100">
    <a href="#">About</a>
  </li>
  <li class="pl-8 opacity-50 hover:opacity-100">
    <a href="#">Gallery</a>
  </li>
</ul>
</header>

```

Таим чином було створено header сторінки (рис. 3.7).



Рисунок 3.7 – header сторінки

Наступним шагом є додання заголовку та підзаголовку. Створюємо <div> з класом “main” та у ньому розміщуємо теги <h1> для заголовку та <h2> для підзаголовку. Для загального елемента додаємо класи “grid” – вказуємо що на елементі буде застосована сітка, “ grid-cols-12” – розмір сітки дванадцять колонок та “ lg:pt-96 md:pt-96 pt-60” – верхні відступи на різних пристроях. На елементах <h1> та <h2> за допомогою класів вказуємо ширину колонок, кількість колонок на яку будуть відступати елементи від лівого краю, розмір тексту на різних пристроях та верхні відступи на різних пристроях.

```

<div class="main grid grid-cols-12 lg:pt-96 md:pt-96 pt-60">
  <h1 class="col-start-2 col-span-9 lg:col-start-3 lg:col-span-5
  md:col-start-2 lg:text-9xl md:text-9xl text-6xl">Rubens</h1>
  <h2 class="col-start-2 col-span-11 lg:col-start-3 lg:col-span-5

```

md:col-span-9 md:col-start-2 text-2xl pt-6">“I’m just a simple man standing alone with my old brushes, asking God for inspiration.”

</h2>

</div>

Таким чином було створено основну частину сатйу (рис. 3.)8.



Рисунок 3.8 – Основна частина сторінки

Останім етапом створення сторінки є footer. Для його створення додаємо тег <footer> у я кому буде розміщено дати народження та смерті художника, а також кнопку переходу. Дату додаємо за допомогою тега <p>, конпку – за допомогою тегів <a> та <img>. До футера було додано наступні класи “flex”, “justify-between”, “ items-end”, “align-bottom” – для розміщення елементів знизу та “pt-48 lg:pt-56 md:pt-60” – для верхніх відступів на різних пристроях.

Для невеликої динаміки сатйу було додано анімацію на кнопку за допомогою @keyframes.

```
@keyframes move {
  from {
    transform: translateY(0%);
  }
}
```

```

50% {
    transform: translateY(50%);
}
to {
    transform: translateY(0%);
}
}

```

Анімацію названо “move”, оскільки вона описує рух та це єдина анімація на сторінці. На початковому етапі анімації рух кнопки за оссю Y буде дорівнювати 0%, на середині кнопка рухатиметься на 50%, на кінцевому етапі знову буде нерухомою. Для кнопки вказуємо створену анімацію, вказуємо її час, та постійність.

```

.footer a img{
    animation: move 3s infinite;
}

```

Таким чином було створено footer сайту. Результат представлено на рисунку 3.9. Результат готової сторінки представлено на рисунку 3.10. Адаптивність сайту під різні девайси представлена на рисунку 3.11.



Рисунок 3.9 – footer сайту

Підводячи підсумки, можна сказати, що Tailwind CSS безперечно економить час, але даний фреймворк не підійде для великих проектів оскільки через велику кількість класів код дуже важко зрозуміти. Також хоча і Bootstrap має гарну документацію, Tailwind CSS зправляється з цим значно краще. Плюсом є, що навідміну від попереднього фреймворка тут самому можна вказувати кількість колонок та управляти сіткою під себе.

Провівши експеримент з точністю можна сказати, що Tailwind CSS безумовно опережує Bootstrap та це новий рівень фреймворків.



Рисунок 3.10 – Готова сторінка сайту

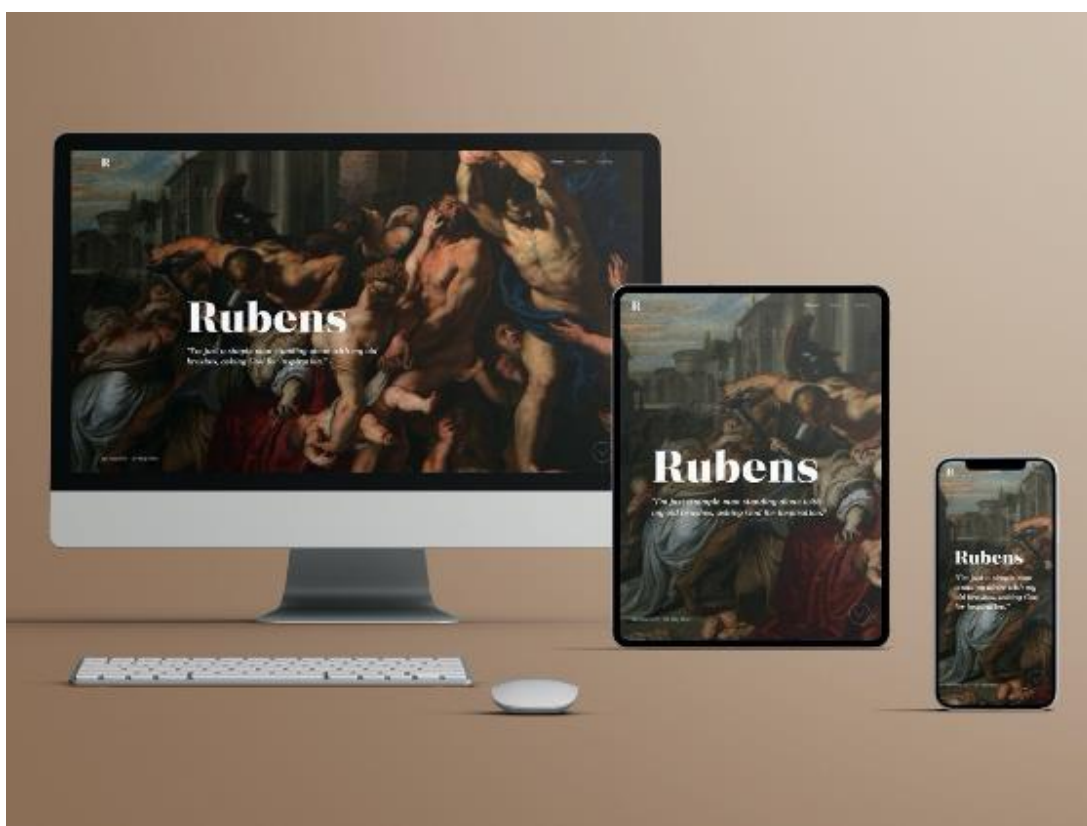


Рисунок 3.11 – Адаптивність сайтів на різних пристроях

## 4 ЕКОНОМІЧНА ЧАСТИНА

### 4.1 Характеристика науково-дослідницьких рішень

У роботі розглядається велика кількість фреймворків. У даному розділі розглянемо найбільш поширений, а саме Bootstrap. Даний фреймворк (до 4-ї версії, у 5-й версії – це вже бібліотека) є неймовірно популярним і затребуваним, його представили ще на початку 2011 року. Адаптивність – його головна перевага. Bootstrap дозволяє створювати проекти з неймовірно чуйним, стильним дизайном – проект автоматично підлаштовуватиметься, враховуючи розмір екрану комп'ютера або мобільного пристрою користувача, що переглядає сайт. До переваг належить: велика кількість стилів, шаблонів, посторінковий дизайн, що суттєво полегшує створення сайту.

Bootstrap став настільки популярним через величезну кількість переваг, у ньому практично відсутні недоліки. Це не тільки HTML/CSS-фреймворк, у Bootstrap також включені плагіни та готові стилі JS/Jquery. Знання Bootstrap часто є одним із обов'язкових вимог роботодавців.

Актуальність теми полягає в тому, що існує велика потреба у фреймворках, оскільки фреймворк – "золота середина" між написанням складного коду та застосуванням обмеженої за функціоналом CMS. Користувач отримує готовий каркас для свого проекту, при цьому не втрачаючи гнучкості в плані функціоналу.

Мета роботи полягає у дослідженні можливостей фреймворків під час створення інформаційних сайтів.

Об'єктом дослідження є можливості фреймворків. Насамперед Bootstrap для створення проектів.

Предметом дослідження є зручність у використанні фреймворків, і здатність швидко та ефективно зробити готовий проект.

## 4.2 Розрахунок кошторисної вартості НДР

Для розрахунку вартості дослідження, необхідно враховувати всі витрати, що мають місце в роботі. В цьому дослідженні:

- заробітна плата виконавців НДР;
- страхові нарахування на заробітну плату (єдиний соціальний внесок);
- вартість використаних матеріальних ресурсів;
- витрати на електроенергію;
- вартість використання основних засобів;
- оплата послуг зв'язку;
- адміністративні витрати.

Велика кількість витрат приходить на виплату заробітної плати працівникам. У роботі беруть участь наступні працівники: керівник проекту, програміст, дизайнер та тестувальник. Місячна заробітна плата керівника складає 21 000 грн, програміста – 16 000 грн, дизайнера – 10 000 грн, тестувальник – 7 000 грн.

Дослідження складається з трьох етапів: підготовчого, основного та заключного. На першому етапі проводиться підготовка та збір необхідних матеріалів, складання плану проведення робіт. На основному етапі виконується постановка задачі, дослідження принципів розробки веб сайтів за допомогою фреймворків, аналіз процесу розробки, аналіз можливості виникнення проблем при використанні фреймфорків.

На заключному етапі проводиться аналіз отриманих даних при використанні фреймворків та якщо потрібно внесення коректування.

Середньоденна заробітна плата за виконання окремого етапу роботи визначається за формулою:

$$ЗП_i = ЗП_\partial \times ТМ_i , \quad (4.1)$$

де  $ЗП_i$  – заробітна плата за виконання  $i$ -го етапу роботи, грн;

$ZП_{\partial}$  – середньоденний заробіток виконавця і-го етапу, грн./ос. на день;

$ТМ_i$  – трудомісткість і-го етапу, люд.-дні.

Середньоденна заробітна плата кожного фахівця розраховується:

$$ZП_{\partial} = ZП_{м} / n , \quad (4.2)$$

де  $ZП_{м}$  – місячна заробітна плата;

$n$  – кількість робочих днів у місяці ( $n=22$ ).

Використовуючи наведені вище формули було розраховано витрати на заробітну плату виконавцям. Результати було наведено в таблиці 4.1.

Таблиця 4.1 – Розрахунок заробітної плати виконавців

Вид роботи	Виконавець		Трудо- витрати, люд.-днів	Серед-ньоденна заробітна плата, грн.	Сума заробітної плати, грн. (гр.3хгр.4х гр.5)
	посада	кіль- кість			
1	2	3	4	5	6
1. Підготовчий етап					
1.1 Підготовка та збір необхідних матеріалів	Керівник	3	3	913,04	2739,12
	Програміст		1	695,65	695,65
	Дизайнер		2	434,78	869,56
1.2 Складання плану проведення робіт	Програміст	2	1	695,65	695,65
	Дизайнер		1	434,78	434,78
2. Основний етап					
2.1 Постановка задачі	Керівник	1	1	913,04	913,04
2.2 Дослідження прин- ципів розробки веб сайтів за допомогою фреймворків	Програміст	1	2	695,65	1 391,3
2.3 Аналіз процесу розробки	Програміст	2	2	695,65	1 391,3
	Дизайнер		2	434,78	869,56
2.4 Аналіз можливості виникнення проблем при використанні фреймфорків	Програміст	2	2	695,65	1 391,3
	Тестувальн ик		1	304,35	304,35
3. Заключний етап					
3.1 Аналіз отриманих даних	Програміст	2	1	695,65	695,65
	Керівник		1	913,04	913,04
3.2 Аналіз можли-вості виникнення проблем при використанні фреймфорків.	Тестувальн ик	2	1	304,35	304,35
	Програміст		1	695,65	695,65
Всього			21		14 304,2

Згідно з таблицею, сума витрат на заробітну плату керівника, програміста, геймдизайнера і тестувальника складе 14 304,20 гривень.

Необхідно врахувати єдиний соціальний внесок, який розраховується:

$$СВ = 0,22 \times ЗП, \quad (4.3)$$

$$СВ = 0,22 \times 14304,20 = 3146.92 \text{ (грн)},$$

де СВ – єдиний соціальний внесок, який відноситься на собівартість;

0,22 – ставка єдиного соціального внеску, частка одиниці;

ЗП – сума витрат на заробітну плату виконавців НДР.

Під час проведення дослідження, використовуються певні матеріали, вартість яких має бути включена в розрахунок вартості роботи.

Вартість використаних матеріалів визначається за формулою:

$$M = \sum_{j=1}^n Q_j \times C_j, \quad (4.4)$$

де М – сумарні витрати на матеріали, в тому числі малоцінні предмети, що швидко зношуються (носії, папір, канцелярські приналежності тощо);

$Q_j$  – кількість використаних одиниць j-го виду матеріалів;

$C_j$  – ціна одиниці j-го виду матеріалів.

У табл. 4.2 приведений список використаних матеріалів, їх кількість та ціна.

Таблиця 4.2 – Матеріальні витрати

Найменування	Од. вим.	Кількість, ( $Q_j$ )	Ціна ( $C_j$ ), грн	Сумарні витрати на матеріали (M), грн.
Ручка	шт.	4	4	16
Папір	уп.	1	90	90
Калькулятор	шт	1	84,60	84,64
Всього				190,64

Проведення дослідження потребує використання 2 комп'ютерів, та програми на якій буде використовуватися програмування за допомогою.

фреймворків. Вони є власністю організації. Вартість комп'ютера, монітора та миші складає 29 500 гривень, другий комп'ютер, монітор та мишка складає вартість 24 000 гривень, програма для програмування є цілком безкоштовною у використанні.

Необхідно розрахувати суму амортизаційних відрахувань:

$$AB = \sum_{k=1}^L \frac{BO_k}{TE_k} \times T, \quad (4.5)$$

де  $AB$  – сума амортизаційних відрахувань, нарахованих під час науково-дослідницької роботи;

$BO_k$  – вартість основних засобів  $k$ -го виду;

$TE_k$  – термін експлуатації основних засобів  $k$ -го виду, днів;

$T$  – термін науково-дослідницької роботи, днів;

$L$  – кількість видів обладнання.

Термін використання першого комп'ютера складає 1277,5 днів, другого комп'ютера – 730 днів. Підставивши відомі значення в (4.5) отримуємо:

$$AB = 395,30 \text{ (грн)}.$$

Таким чином, сума амортизаційних відрахувань склала 395,30 гривень.

Оскільки використовується обладнання, яке потребує електроенергії, то необхідно визначити відповідні витрати:

$$Z_e = M \times t \times T_{\text{кВт}}, \quad (4.6)$$

де  $M$  – потужність устаткування, тобто кількість енергії, споживаної за одиницю часу (кВт/година);

$t$  – кількість годин використання устаткування за період науково-дослідницької роботи;

ТкВт – тариф, тобто вартість використання 1 кВт електроенергії.

У дослідженні використовується два комп'ютери. Їх потужності 0,7 кВт та 0,5 кВт відповідно. Тариф на електроенергію для підприємств за першим класом напруги складає 1,75 грн./кВт на годину (без ПДВ). За допомогою (5.6) розраховуємо витрати на електроенергію:

$$Z_e = 0,7 \times 80 \times 1,75 + 0,5 \times 48 \times 1,75 = 98 + 42 = 140 \text{ (грн.)}$$

Таким чином, витрати на електроенергію складуть 140 гривень.

Під час виконання науково-дослідницької роботи необхідно використання інтернету. Його вартість складає 180 гривень на місяць.

Адміністративні витрати, такі як водопостачання, водовідведення, освітлення та опалення прийнято у розмірі 10% від витрат на оплату праці.

Виконані розрахунки та відомі дані були внесені до таблиці 4.3.

Загальна сума витрат на НДР складатиме 19 787,48грн.

Таблиця 4.3 – Розрахунок витрат на проведення НДР

Стаття витрат	Сума, грн.
1. Заробітна плата (ЗП)	14 304,2
2. Єдиний соціальний внесок (22% від п.1)	3 146,92
3. Матеріальні витрати	190,64
4. Амортизація основних засобів (вартість машинного часу)	395,30
5. Витрати на спожиту електроенергію	140
6. Інші витрати, у тому числі:	
- адміністративні витрати (10% від п.1);	1 430,42
- вартість послуг зв'язку;	180
7. Разом (Вр)	19 787,48

#### 4.3 Оцінка результатів НДР

Результат – це завершальний наслідок послідовності дій, виражений якісно або кількісно. В загальному випадку оцінка результатів НДР – це

визначення ефективності отриманих рішень порівняно з сучасним науково-технічним рівнем.

Покращення характеристик процесу, який є предметом дослідження, виникає завдяки тому, що досягаються цілі, заради яких було замовлене НДР.

Відповідно до теми даної роботи можна зробити висновок про те, що результатом впровадження НДР є розробка рекомендацій щодо правильного вибору використання фреймворків. У разі виконання рекомендацій процес програмування буде швидким, і кількість помилок які можуть виникнути у ході програмування буде значно менше, тобто буде часу витраченого на програмування буде зменшено. Потрібно враховувати що навіть дивлячись на всі попередження у програміста можуть виникнути деякі проблеми ході програмування і час на їх вирішення також буде витрачено. Приблизно 12% помилок можуть залишитися, та час на них також буде витрачено. Витрачений час приблизно буде дорівнювати 3 хвилини на годину, що за загальній час дорівнюватиме 9,2 години. В такому разі, результат від впровадження НДР розраховується за наступною формулою:

$$\Delta P_j = |X_{бj} - X_{нj}|, \quad (4.7)$$

де  $\Delta P_j$  – покращення  $j$ -тої характеристики системи (процесу) за рахунок впровадження результатів НДР ( $j=1,m$ ),

$X_{бj}$  – базове значення  $j$ -тої характеристики, тобто до впровадження результатів НДР;

$X_{нj}$  – нове значення  $j$ -тої характеристики після впровадження пропонуванних рішень

У якості досліджуваних характеристик було обрано час, який витрачається на програмування за допомогою фреймфорів, та кількість часу що витрачається на виправлення помилок. Процес програмування у середньому триває 10 днів, по 8 годин це 80 години, а при впровадженні

запропонованих рекомендацій цей час зменшується на 8 днів, тобто 64 години, що на 16 годин менше. Потрібно брати до уваги 10 % помилок, які можуть виникнути, та потребу у часі на їх виправлення, що дорівнюватиме 7 годин. Якщо враховувати заробітну плату, то за 10 днів програміст отримає 6 956,5 гривень, за 8 днів він отримає 5 565,44 гривень. При урахуванні затраченого часу на виправлення помилок, то програміст отримає 6,174,16 гривень.

$$\Delta P_1 = |80 - 64| = 16 \text{ (год.)},$$

$$|\Delta P_2 = |6\,956,5 - 6,174,16| = 782,34 \text{ (грн)}.$$

У результаті проведених досліджень можна зробити висновок, що використання на практиці запропонованих рекомендацій забезпечить скорочення часу на програмування на 16 годин, а сума зекономлена на роботі дорівнює 782,34 гривень.

#### 4.4 Визначення економічної ефективності результатів НДР

З метою визначення економічної ефективності від впровадження результатів НДР, необхідно порівняти витрати на розробку НДР з результатами.

Основним показником економічної ефективності науково-дослідницької роботи є коефіцієнт "ефект-витрати", який обраховується:

$$K_{ев} = \frac{\Delta P_j}{Bp}. \quad (4.8)$$

де  $K_{ев}$  – коефіцієнт "ефект-витрати", який відбиває, наскільки кожна гривня витрат НДР змінює  $j$ -ту характеристику досліджуваного процесу.

Підставивши відомі значення отримуємо:

$$K_{\text{ев}} = \frac{16}{19\,787,48} * 100\% = 0,081 (\%).$$

$$K_{\text{ев}} = \frac{782,34}{19\,787,48} * 100\% = 3,95 (\%).$$

У результаті проведених досліджень, можна зробити висновок про те, що кожна гривня витрат на розробку НДР забезпечує зниження витрат часу на програмування додатку на 0,081 %. Отриманий результат показує, що вкласти грошей в дослідження необхідно більше, ніж буде збережено від перепрограмованих помилок. Але дослідження виконуються лише 1 раз, а помилки можуть виникати постійно, і у кожному проекті можуть бути нові помилки. Тому, якщо проводити дослідження кожного разу, то це буде забезпечувати економію коштів підприємства. Дана науково-дослідницька робота має позитивний показник економічної ефективності. Роботу у цілому можна враховувати ефективною або такою, що має науковий та технічний рівень.

## ВИСНОВКИ

В результаті виконання атестаційної роботи було проведено дослідження можливостей фреймворків для створення інформаційних сайтів. А саме, було проведено аналіз літератури, аналіз проблеми вибору фреймворку, поставлено цілі завдань дослідження.

Для проведення досліджу було обрано найкращі фреймворки та виділено основні критерії оцінювання. Для вибору фреймворка було

В результаті виконання атестаційної роботи було проведено було проведено аналіз на основі методу експертних оцінок та парного порівняння кожного з фреймворків. Таким чином теоритичне дослідження показало, що найкращими фреймворками виявились Tailwind CSS та Bootstrap.

У ході виконання експериментального дослідження було створено сторінку інформаційного сайту на різних вреймворках. Дослідження показало, що кращим виявився Tailwind CSS, але карще не використовувати його для великих проєєтів.

Проведено економічне обґрунтування доцільності проведення даної науково-дослідної роботи (НДР).

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Best CSS Frameworks in 2021. URL: [https://dev.to/theme\\_selection/best-css-frameworks-in-2020-1jjh](https://dev.to/theme_selection/best-css-frameworks-in-2020-1jjh) (дата звернення: 20.10.2021).
2. Какой CSS-фреймворк выбрать?. URL: <https://itrack.ru/company/blog/article/какоу-freymvork-vybrat-/> (дата звернення: 20.10.2021).
3. Build fast, responsive sites with Bootstrap. URL: <https://getbootstrap.com> (дата звернення: 20.10.2021).
4. Метод парных сравнений в юзабилити. URL: <https://askusers.ru/blog/obuchenie/metod-parnykh-sravneniy-v-yuzabiliti/> (дата звернення: 20.10.2021).
5. Rapidly build modern websites without ever leaving your HTML. URL: <https://tailwindcss.com> (дата звернення: 20.10.2021).
6. Sublime Text – подробный обзор редактора с выбором плагинов. URL: <https://linuxvsem.ru/programs/sublime-text-obzor> (дата звернення: 20.10.2021).
7. The most popular css frameworks comparison. URL: <https://themeselection.com/wp-content/uploads/2020/05/popular-css-framework-comparison-2020.png> (дата звернення: 20.10.2021).
8. Meiert J.O. The Little Book of HTML/CSS Frameworks. 1005 Gravenstein Highway North, Sebastopol, CA: O`REILLY, 2015. 216 p.
9. Find the cost of adding a npm package to your bundle. URL: <https://bundlephobia.com> (дата звернення: 20.10.2021).
10. 14 Best CSS Frameworks for Front-End Developers. URL: <https://geekflare.com/best-css-frameworks/> (дата звернення: 20.10.2021).
11. 6 CSS micro frameworks compared. URL: <https://blog.logrocket.com/6-css-micro-frameworks-compared/> (дата звернення: 20.10.2021).
12. Foundation The most advanced responsive front-end framework in the world. URL: <https://get.foundation> (дата звернення: 20.10.2021).
13. Documentation Everything you need to create a website with Bulma. URL: <https://bulma.io/documentation/> (дата звернення: 20.10.2021).