

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**  
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК, ФІЗИКИ ТА МАТЕМАТИКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ПРОГРАМНОЇ ІНЖЕНЕРІЇ

**РОЗРОБКА FRONTEND ЧАСТИНИ МОБІЛЬНОЇ ВЕРСІЇ САЙТУ  
PUBLICATION**

Кваліфікаційна робота  
на здобуття ступеня вищої освіти «бакалавр»

Виконала: студентка 4 курсу 441 групи

Спеціальності: 121 Інженерія програмного  
забезпечення

Освітньо-професійної програми:

Інженерія програмного забезпечення

Волобуєва В.А.

Керівник: доктор пед.н., професор

Вінник М.О.

Рецензент: Senior Software

Engineer (EPAM)

Кожевніков Д.І.

Херсон – 2022

## ЗМІСТ

ЗМІСТ .....	2
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....	3
ВСТУП.....	4
РОЗДІЛ 1 АНАЛІЗ ТЕХНОЛОГІЇ СТВОРЕННЯ ДИЗАЙНУ САЙТУ .....	5
1.1 Особливості розробки мобільної версії сайту .....	5
1.2 Аналіз способів представлення аналітичних даних. ....	6
1.3 Особливості використання Bootstrap. ....	7
РОЗДІЛ 2 ПРОЕКТУВАННЯ ДИЗАЙНУ САЙТУ PUBLICATION .....	10
2.1 Проектування user flow сайту .....	10
На основі переліченого вище була розроблена наступна діаграма: .....	11
2.2 Розробка UX прототипу майбутнього сайту.....	11
2.3 Розробка UI дизайну сайту .....	16
РОЗДІЛ 3 СТВОРЕННЯ FRONT-END ЧАСТИНИ САЙТУ PUBLICATION. 22	
3.1 Основна розмітка.....	22
3.2 Створення діаграм .....	24
3.3 Інтерактивність таблиці .....	26
ВИСНОВКИ .....	30
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....	32

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

iOS	iPhone OS
UI	User Interface
UI-дизайн	дизайн інтерфейсу користувача
UI kit	набір готових елементів
UX	User Experience
UX-дизайн	проектування інтерфейсу на основі вивчення користувацького досвіду
Footer	структурний елемент, розташований внизу сторінки.

## ВСТУП

Використання інформаційних технологій помітно зростає в усіх сферах людського життя, зокрема це стосується сфери освіти. Накопичення різноманітних даних є логічним наслідком наукової діяльності та її необхідною умовою, проте, це також може стати причиною ускладнення пошуку, аналізу та презентації необхідної інформації. Саме це обумовлює актуальність створення зручних інструментів для оптимізації структури аналітичних даних стосовно наукових публікацій в межах університету.

Ідея редизайну сайту Publication розвинулась із необхідності в зручному, інтуїтивно зрозумілому й презентабельному сайті, якого не вистачало студентам й працівниками нашого університету, зокрема тим, хто зацікавлений у його науковій діяльності.

Актуальність даної роботи обумовила мету, об'єкт та предмет дослідження.

**Мета дослідження** – розробка front-end частини мобільної версії сайту Publication.

В процесі досягнення мети перед нами постали наступні **завдання**:

- дослідити особливості розробки дизайну сайтів та виконати аналіз способів представлення аналітичних даних.
- спроектувати user-flow для кожного екрану відповідно до вимог технічного завдання.
- розробка UX сайту.
- розробка UI сайту.
- верстка мобільної версії сайту.

**Об'єкт дослідження** – сайт Publication.

**Предмет дослідження** – розробка UI/UX дизайну та front-end частини мобільної версії сайту Publication за допомогою онлайн-сервісу для створення інтерфейсів та прототипів Figma та фреймворку Bootstrap.

# РОЗДІЛ 1

## АНАЛІЗ ТЕХНОЛОГІЇ СТВОРЕННЯ ДИЗАЙНУ САЙТУ

### 1.1 Особливості розробки мобільної версії сайту

Інтуїтивно зрозумілий та зручний інтерфейс є основним критерієм, за яким користувач обирає сайт, тоді як складність у використанні найчастіше стає причиною його уникнення. Продукт має допомагати у вирішенні якоїсь конкретної проблеми або задачі, що постає перед користувачем, тож забезпечення швидкості та безболісності цього процесу є першочерговим завданням дизайнера.

Аналіз літературних джерел дозволив виявити наступні особливості розробки дизайну мобільних версій сайтів:

#### 1. Маленький екран.

Мобільні пристрої мають невеликий розмір екрану, маленьку клавіатуру та обмежені пропускну здатністю для мультимедійних файлів, тож доцільним є фільтрація й відображення лише найнеобхіднішого контенту в межах однієї сторінки. Також може бути необхідним уникнення появи смуги прокрутки, що передбачає компактність та мінімалістичність дизайну, спрощення елементів інтерфейсу [7].

#### 2. Адаптивність.

Адаптивний дизайн – не новий напрям у web-розробці, тож йому приділяють достатньо уваги. Користувач повинен мати змогу коректно взаємодіяти з функціоналом сайту з будь-якого пристрою, будь то планшет, комп'ютер, чи смартфон. Функціонал Figma містить у собі всі необхідні інструменти для створення адаптивного дизайну, що покращує та прискорює роботу.

#### 3. Спрощення навігації.

Інтерфейс сайту має бути продуманим та лаконічним. Слід пам'ятати, що гаджети у сучасному світі використовуються майже на ходу. З цього випливає необхідність розташування основних елементів інтерфейсу у нижній частині екрану – так, щоб користувач мав змогу виконувати усі необхідні дії однією рукою – та скорочення шляхів досягнення будь-якої мети до трьох дотиків [3, 5, 7].

#### **4. Повернення на попередній екран.**

У процесі взаємодії з сайтом у користувача може з'явитися потреба повернутися на головний екран. У зв'язку з цим кнопка «Головна», або відповідна іконка має бути присутня на кожній сторінці, запобігаючи необхідності повторювати весь шлях спочатку [3]. Існує тренд заміни кнопки або іконки на клікабельний логотип компанії або закладу.

#### **5. Простий фон.**

Дизайн iOS додатків став основою сучасного тренду мінімалізму. Дизайнери використовують прості фони: однотонний, градієнт або розмите зображення - що дозволяє акцентувати увагу на елементах інтерфейсу та текстовому контенті як на основній інформації [5].

#### **6. Меню «бургер».**

Маленький екран смартфонів обумовлює згортання меню при великій кількості сторінок сайту. Зазвичай для цього використовують іконку «бургер», що після кліку замінюється на іконку «X». Зазвичай меню не зміщується при прокрутці, дозволяючи користувачеві скористатися його функціоналом в будь який момент.

### **1.2 Аналіз способів представлення аналітичних даних.**

Основним завданням сайту Publication є презентація аналітичних даних, доцільним було знайти та проаналізувати способи їх представлення.

#### **1. Таблиці.**

Таблиці є найраціональнішим способом представлення великої об'ємів інформації щодо явища.

Завдяки структурованості за рядками та стовпцями є можливим сортування даних, що спрощує їх аналіз. Перевагами також є наглядність, чіткість та лаконічність.

З недоліків можна виділити обмежені варіації проектування UI в порівнянні з діаграмами/зображеннями.

## **2. Графіки та діаграми.**

Графіки є загальним зображенням інформації, що, з одного боку, дозволяє простежити закономірності та зв'язки представленої інформації, а з іншого – не може вмістити великої кількості типів інформації.

З переваг можна виділити простір для проектування креативного UI.

### **Види діаграм за формою:**

- стовпчасті
- кругові
- лінійні
- фігурні

## **3. Зображення.**

Зображення відрізняються великою кількістю варіації дизайну, проте є складними в редагуванні й проектуванні. Даний спосіб більш доцільно використовувати для представлення схем, макетів або як додаткові елементи всередині таблиць та діаграм, ніж для безпосередньо аналітики та статистики.

## **4. Текст.**

Найпростіший у редагуванні й найскладніший для розуміння. Текст доцільно використовувати для пояснень діаграм/таблиць/графіків, або як їх складові елементи.

### **1.3 Особливості використання Bootstrap.**

Bootstrap як HTML/CSS фреймворк містить у собі ряд інструментів, зокрема систему класів, що полегшують верстку адаптивних сайтів, забезпечуючи швидкість та зручність роботи. Серед переваг можна виділити:

- масштабованість

- легке налаштування (достатньо завдати нові правила для класів, що дозволяє уникнути використання атрибуту !important.

- система сіток

- велика кількість доступних шаблонів дизайну

- велика кількість навчальних матеріалів (зокрема документації, курсів та туторіалів).

Серед недоліків можна виділити відсутність гнучкості в розробці (через шаблонність, яка може заважати в масштабних й креативних проектах) та некоректне відображення у старих браузерях.



## **Висновки:**

Були проаналізовані особливості розробки дизайну мобільних версій web-сайтів. Основними є простий фон, адаптивність, лаконічна та зручна навігація, проектування UX з урахуванням маленьких розмірів екранів та згортання меню за іконкою «бюргер». Інтерфейс насамперед має бути функціональним та зрозумілим, що робить елементи UI другорядними. Також важливо надати користувачу можливість повторити шлях сценарію в зворотному порядку, що реалізується кнопкою "головна", або відповідною іконкою.

Були проаналізовані способи представлення аналітичних даних. Основними є текст, зображення, діаграми, графіки та таблиці. Найдоцільнішим для великих кількостей даних є використання таблиць, тоді як для малих є можливим використання діаграм та графіків, які дають змогу урізноманітнити UI продукту.

Були проаналізовані недоліки та переваги Bootstrap й зроблено висновок, що фреймворк ідеально підходить для невеликих сайтів, дозволяючи швидко реалізувати якісний продукт, проте може створити проблему втрати оригінальності в більш масштабних проектах.

## РОЗДІЛ 2

### ПРОЕКТУВАННЯ ДИЗАЙНУ САЙТУ PUBLICATION

#### 2.1 Проектування user flow сайту

Першим етапом проектування було створення user flow мапи, що освітлювала типові сценарії користувача.

User flow – візуалізація послідовностей дій, які виконує користувач для досягнення конкретної мети. Діаграма дозволяє продумати функціонал продукту й відповідні до дій елементи інтерфейсу, що згодом використовується на етапі розробки UX прототипу. На етапі створення UI user flow дає розуміння акцентів на певних елементах дизайну. Наприклад, користувач обов'язково має помітити кнопку вибору бази даних, тоді як номер відображуваної сторінки є другорядною інформацією, тож не повинен привертати зайвої уваги [2, 6, 9].

В процесі створення user flow були дані відповіді на наступні питання:

**1. Хто основний користувач:**

студенти та працівники Херсонського державного університету.

**2. Яка ціль сайту:**

покращити роботу університету та спростити пошук аналітичних даних щодо публікації наукових робіт.

**3. Які кроки користувач має здійснити для досягнення цих цілей:**

- досягнення будь-якої сторінки повинне бути здійснене не більш ніж трьома кліками: клік на іконку «бюргер» -> пункт меню -> на сторінці кластерного аналізу – вибір дати.
- сортування для кожного з рядків таблиці – клік по заголовку рядка.
- отримати точну кількість записів у діаграмі: клік на відповідний сектор або рядок.

Основне меню має бути доступним на будь-якому етапі сценарію. Також користувач повинен мати змогу повторити пройдений шлях у зворотному напрямку (до головної сторінки).

**На основі переліченого вище була розроблена наступна діаграма:**

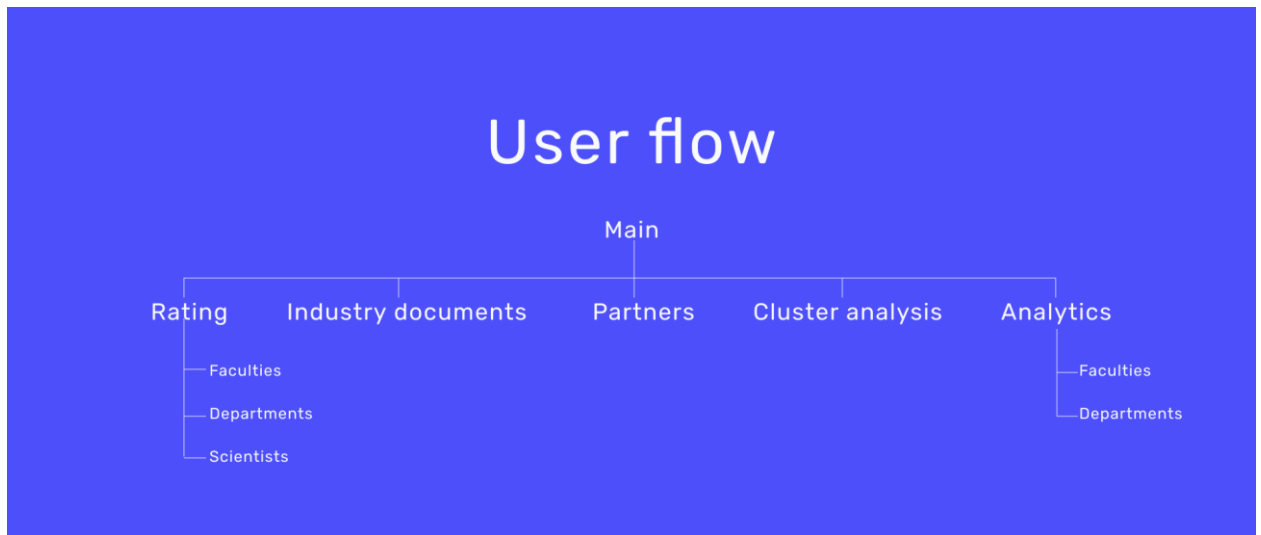


Рис. 2.1 user flow діаграм–а.

Для її проектування використовувався онлайн-сервіс для розробки дизайну Figma, що обґрунтовується зручністю використання як для самостійного дизайнера, так і для командної роботи.

Цей сервіс використовується й на наступних етапах проектування.

## 2.2 Розробка UX прототипу майбутнього сайту

На етапі розробки UX прототипу на основі результатів минулого етапу було визначено розташування елементів інтерфейсу, блоків контенту та інших елементів.

Відкривши сайт, користувач потрапляє на головну сторінку. Перший екран містить меню сайту з наступними сторінками: Рейтинг, Документи за галузями знань, Заклади-партнери, Кластерний аналіз та Аналітика. Також наявна кнопка «Головна», що має бути реалізована іконкою, або логотипом.

Іконка меню завжди знаходиться у верхній частині екрану, де наявні кнопки переходу на інші сторінки. Порядок розташування пунктів обумовлюється частотою використання відповідних розділів (найчастіше використовуються – розташовуються зверху).



Рис. 2.2 Меню.

Також перший екран має містити елементи UI, наприклад ілюстрації, іконки, що зробить перше враження користувача більш приємним.

Нижче розташовані корисні посилання, сформовані у розділи. Для полегшення прокрутки було вирішено сховати їх у інтерактивних вкладках.

Далі відображена статистика реєстрації факультетів у основних базах Scopus, Google Scholar, Semantic Scholar, Orchid та Web of Science.

Завершує будь-яку сторінку footer, в якому зазначені автори сайту та рік розробки.



Рис. 2.2 Footer.

Інші сторінки містять стандартну структуру: меню, заголовки, рядок пошуку вченого, контент, footer. У залежності від типу даних, на сторінці можуть бути розміщені таблиці, стовпчасті або кругові діаграми.

Таблиці містять вкладки вибору бази, випадне меню вибору кількості відображуваних записів, інтерфейс переходу на наступну або попередню сторінку та безпосередньо розміщувані дані. Маленький екран обумовлює появу горизонтальної смуги прокрутки для стовпців. Для зручності було вирішено зафіксувати стовпець назв факультетів. Також була реалізована можливість сортування за кожним з стовпців таблиці.


Факультет	Індекс Хірша 	Докуме
Факультет біології, географії і екології	12	12
Факультет комп'ютерних наук, фізики та математики	12	12
Соціально-психологічний факультет	12	12
Факультет економіки і менеджменту	12	12
Факультет фізичного виховання та спорту	12	12
Факультет іноземної філології	12	12
Медичний факультет	12	12
Педагогічний факультет		

Рис. 2.4 Варіант відображення інтерфейсу таблиці.

Кругові діаграми реалізовані у вигляді інтерактивного кола. В неактивному стані внутрішня частина діаграми може бути порожньою, або відображати заголовок діаграми та загальну кількість записів. В активному стані, під час наведення на один з секторів, у внутрішній частині з'являється заголовок сектору й кількість відповідних записів. Інші сектори змінюють стан (згодом було реалізовано завдяки зміні їх кольору). Збоку або унизу (залежно від структури сторінки) від діаграми розташований список усіх секторів з відповідними їм кольорами.

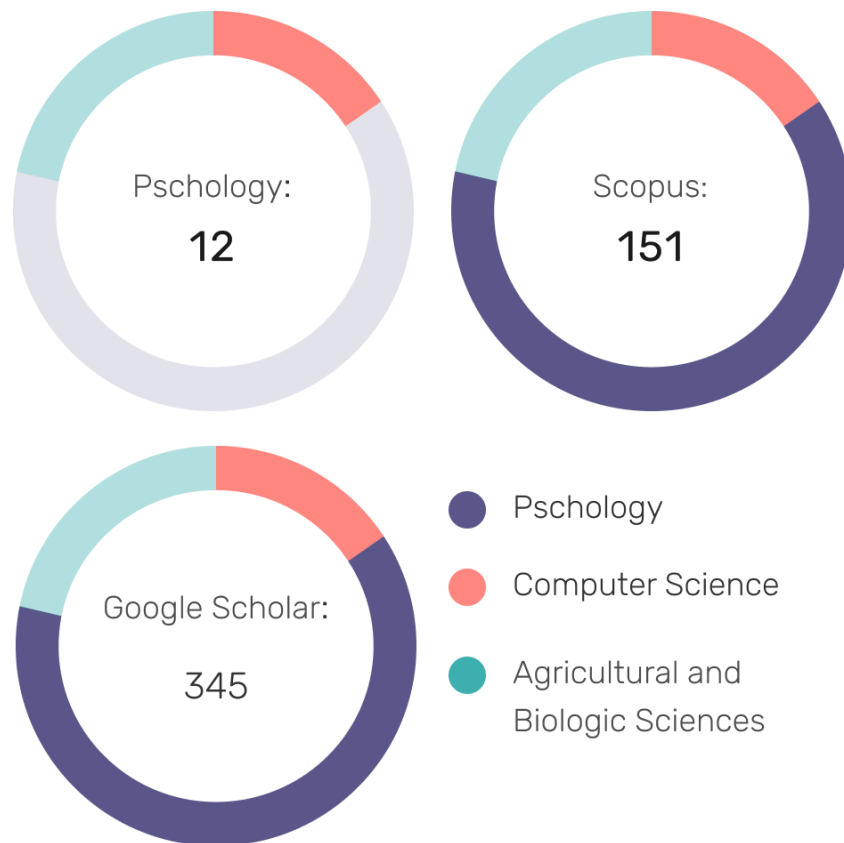


Рис. 2.4 Варіанти відображення кругових діаграм: активний та неактивні.

Інші діаграми реалізовані у вигляді рядків. Статистика за факультетами складається з безпосередньо діаграми, логотипу факультету та заголовком, у якому вказана назва факультету. В активному стані відповідні рядки відображають кількість записів (стан є активним для всіх факультетів), тоді як інші рядки змінюють стан на неактивний. Для різних кількостей записів було розроблено свій спосіб розміщення числа.

Кафедра інформатики, програмної інженерії та економічної кібернетики



Рис. 2.4 Діаграма рейтингу кафедр.

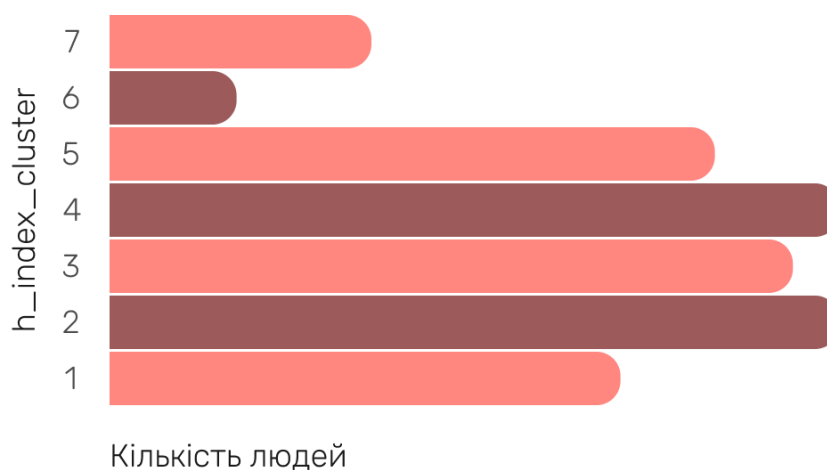


Рис. 2.4 Результати кластерного аналізу.

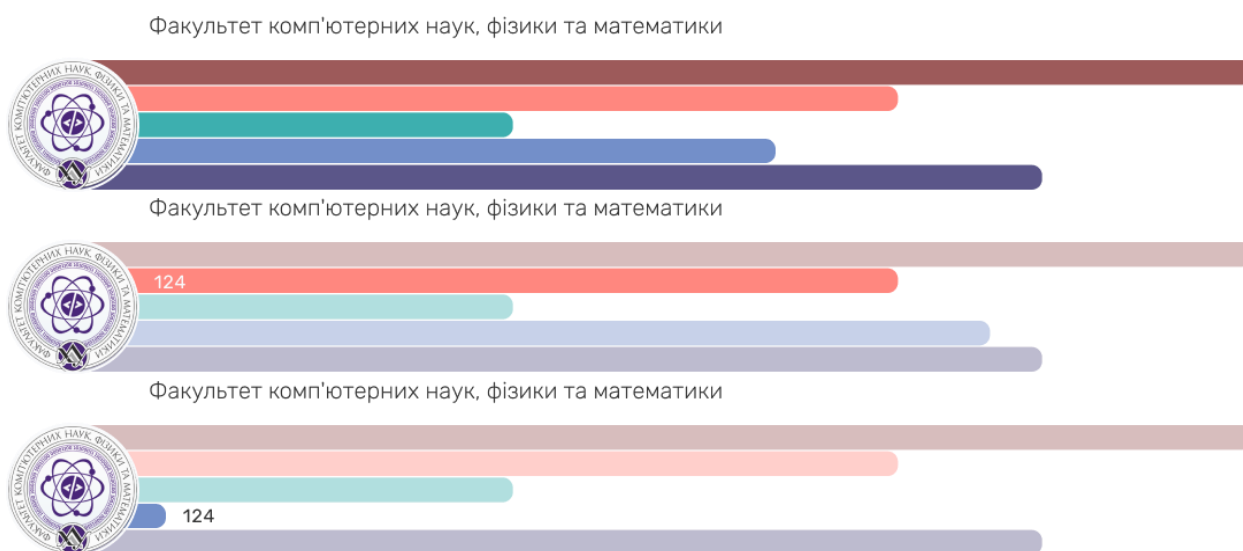


Рис. 2.4 Варіанти відображення стовпчастих діаграм: неактивний та активні.

### 2.3 Розробка UI дизайну сайту

Цей етап розробки полягає в створенні картинки, яку бачитиме користувач в процесі взаємодії з продуктом. Для цього дизайнер обирає шрифти, працює за палітрою, займається розробкою шаблонів компонентів додатку й збирає воєдино остаточний макет. Для кожного з типів елементів було розроблено UI kit [8].



## 1. Шрифти.

Майже половину контенту складає текст. З цієї причини шрифт є центральним елементом дизайну додатку й має відповідати ряду умов. По-перше, лаконічність - текст через свою кількість не потребує додаткового акценту окрім кольорового, тому було обрано шрифт без зарубок й інших додаткових елементів [1].

По-друге, читабельність - для досягнення цього було збільшено міжбуквений інтервал.

Ще одна особливість - акценти були зроблені не за рахунок виділення основної інформації. Замість цього було зменшено помітність другорядної інформації завдяки використанню типу накреслення «Light», малого розміру тексту та прозорості .

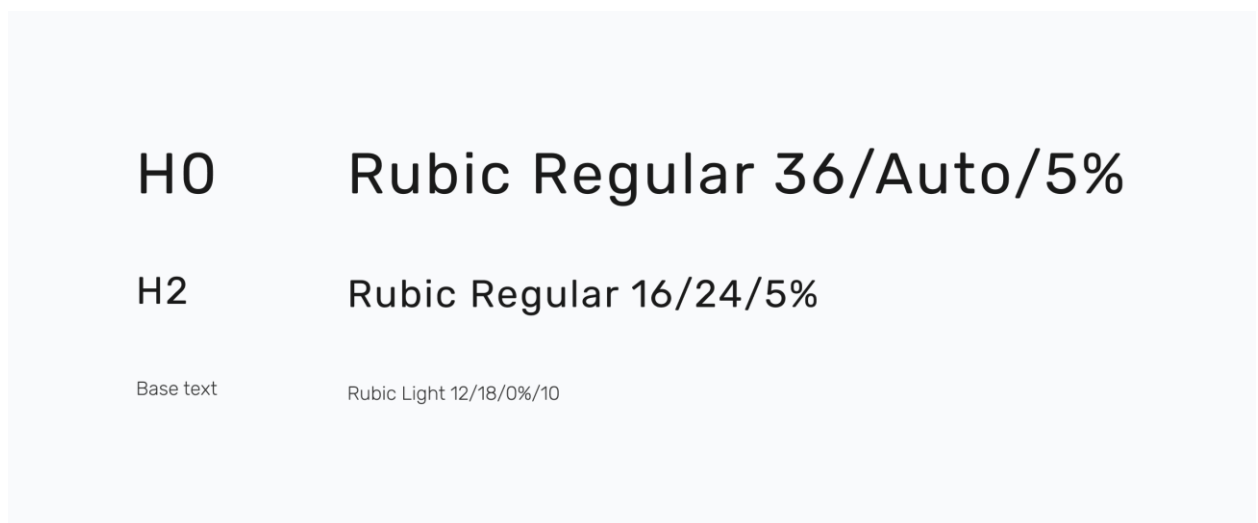


Рис. 2.5 Шрифти в UI kit.

## 2. Палітра.

Дизайн додатку було розроблено в поліхромній стилістиці на основі закону контрасту. Основним кольором обрано різні відтінки блакитного, що обумовлено активним використанням цієї гами в межах університету, зокрема на його офіційному сайті. В діаграмах та ілюстраціях також

відображаються відтінки контрастного йому кольору – помаранчевого. Згодом кількість інформації обумовила розширення палітри, яка використовується для побудови діаграм.



Рис. 2.6 Палітра в UI kit.

### 3. Компоненти.

За шаблон більшості використаних компонентів було обрано блок без фону, контуру та ефектів. Елементи діаграм мають заокруглення в правій частині та залиті одним з кольорів палітри.

Як компоненти були реалізовані меню, кругові діаграми, стовпчасті діаграми, таблиці, елементи списків, рядок пошуку, випадні меню, іконки

меню, сортування, прокрутки, вкладки вибору бази, вкладки посилань, footer, меню вибору сторінки записів.

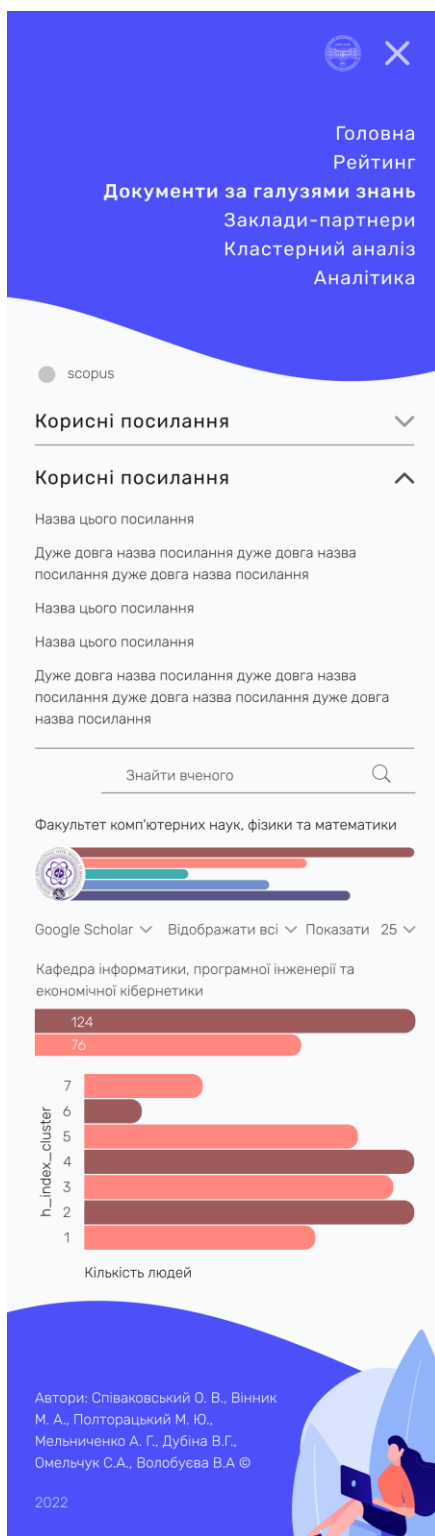


Рис. 2.7 Компоненти в UI kit.

Нижче представлений остаточний мобільної версії сайту Publication.

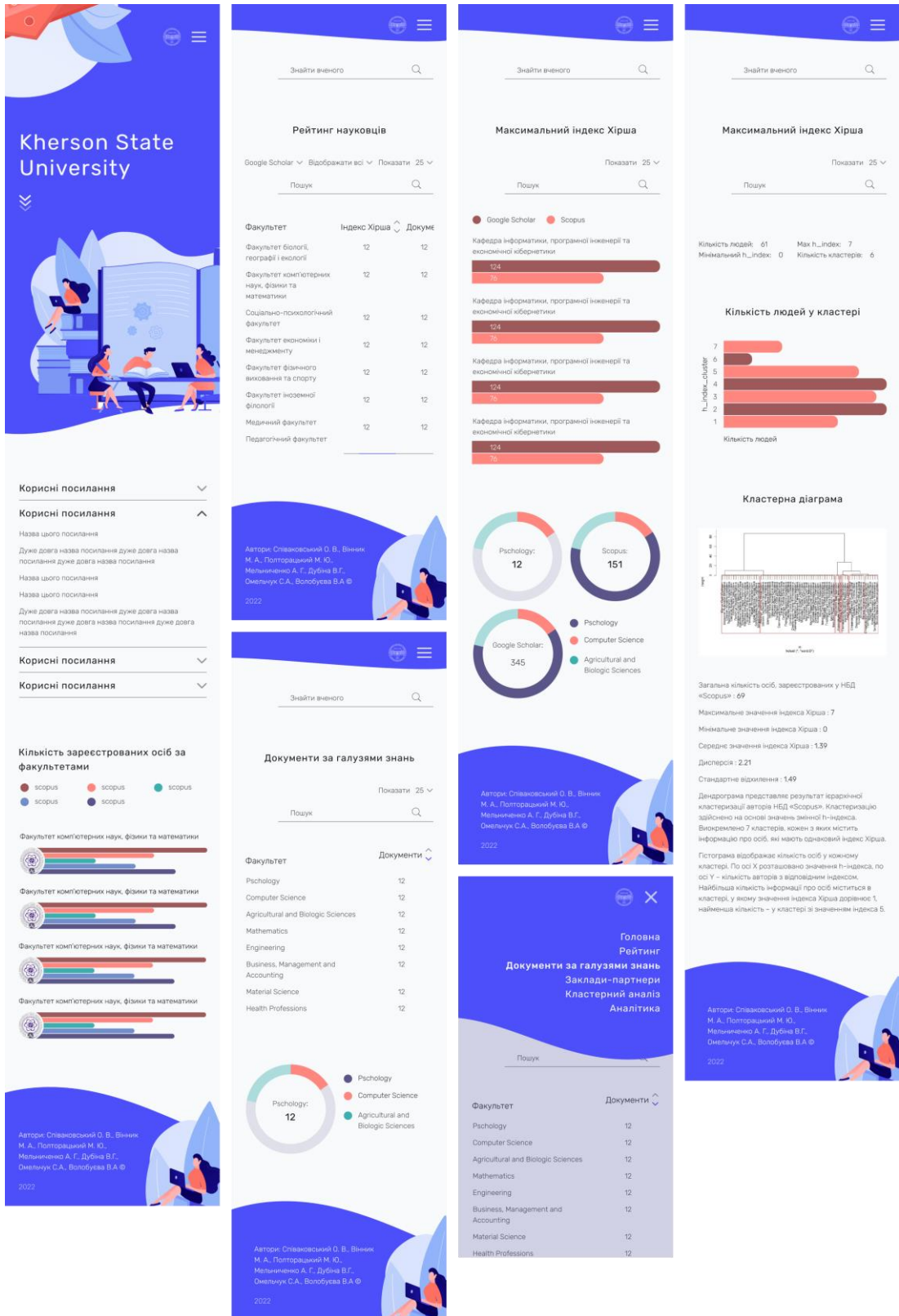


Рис. 2.8 UI дизайн десктопної версії сайту Publication.

## **Висновки:**

Визначено цільову аудиторію: студенти та працівники університету - та розроблено основні можливі сценарії: перегляд сторінок, сортування, вибір дати та кількості відображуваних записів, взаємодії з діаграмами, здійснення пошуку. На основі цього було спроектовано User flow.

На етапі створення UX прототипу було визначено розташування елементів інтерфейсу та контенту, спроектована навігація. Додаток містить дев'ять основних сторінок: головна, рейтинг за факультетами, рейтинг кафедр, рейтинг науково-педагогічних працівників, документи за галузями знань, заклади-партнери, кластерний аналіз, аналітика за факультетами, аналітика за кафедрами. Також були передбачені кругові діаграми, стовпчасті діаграми, таблиці та спадне меню.

На етапі UI дизайну були розроблені UI kits для кольорів, шрифтів та основних компонентів, на основі яких згодом спроектовано фінальний макет додатку.

## РОЗДІЛ 3

### СТВОРЕННЯ FRONT-END ЧАСТИНИ САЙТУ PUBLICATION

#### 3.1 Основна розмітка

Першим етапом проектування було створення HTML розмітки та подальше редагування їх стилів, для чого були використані влаштовані шаблони Bootstrap.

В якості header частини були використані:

- `container-fluid`, що дозволяє розтягнути контент по всій ширині екрану
- шаблон `navbar fixed-top`, що реалізує адаптивне меню, що не зникає при прокрутці
- атрибут `data-bs-toggle` зі значенням `"collapse"`, що автоматично створює спадне меню-бургер на мобільних версіях.

```

<body>
  <div class="container-fluid header">
    <div class="head"></div>
    <nav class="navbar fixed-top header" id="menu" data-spy="affix" data-offset-top="171">
      <!--  -->
      <a class="navbar-brand" href="#"></a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-expanded="false" />
      
      <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav">
          <li class="nav-item">
            <a class="nav-link active currentpage" aria-current="page" href="index.html">Головна</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="rating.html">Рейтинг факультетів</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="rating.html">Рейтинг кафедр</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="rating.html">Рейтинг працівників</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="docs.html">Документи за галузями знань</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="partners.html">Заклади-партнери</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="cluster.html">Кластерний аналіз</a>
          </li>
          <li class="nav-item">
            <a class="nav-link active" aria-current="page" href="analytys.html">Аналитика</a>
          </li>
        </ul>
      </div>
    </nav>
    
  </div>

```

Рис. 3.1 Розмітка header-у.

Також атрибут `data-bs-toggle` використано для реалізації прихованих посилань, зокрема на головній сторінці.

```

</div>
<div class="content">
  <button class="btn btn-link collapsed" data-bs-toggle="collapse" href="#hide1">Виступи на конференціях</button>
  <div class="collapse link_content" id = "hide1">
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
    <p>Назва цього посилання</p>
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
  </div>
  <button class="btn btn-link collapsed" data-bs-toggle="collapse" href="#hide2">Бібліотека ХДУ</button>
  <div class="collapse link_content" id = "hide2">
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
    <p>Назва цього посилання</p>
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
  </div>
  <button class="btn btn-link collapsed" data-bs-toggle="collapse" href="#hide3">Наукометрія</button>
  <div class="collapse link_content" id = "hide3">
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
    <p>Назва цього посилання</p>
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
  </div>
  <button class="btn btn-link collapsed" data-bs-toggle="collapse" href="#hide4">Корисні посилання</button>
  <div class="collapse link_content" id = "hide4">
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
    <p>Назва цього посилання</p>
    <p>Назва цього посилання</p>
    <p>Дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання дуже довга назва посилання</p>
  </div>
</div>

```

Рис. 3.2 Розмітка прихованих посилань.

Меню реєстрів ідентифікації вчених було реалізоване не зразок табів завдяки системі класів `nav` і `tabs`.

```

<nav>
  <div class="nav nav-tabs id="nav-tab" role="tablist">
    <button class="nav-link active" id="orcid_tab" data-bs-toggle="tab" data-bs-target="#orcid" type="button" role="tab" aria-controls="orcid" aria-selected="true">Orcid</button>
    <button class="nav-link" id="scopus_tab" data-bs-toggle="tab" data-bs-target="#scopus" type="button" role="tab" aria-controls="scopus" aria-selected="false">Scopus</button>
    <button class="nav-link" id="web_tab" data-bs-toggle="tab" data-bs-target="#web" type="button" role="tab" aria-controls="web" aria-selected="false">Web</button>
    <button class="nav-link" id="google_tab" data-bs-toggle="tab" data-bs-target="#google" type="button" role="tab" aria-controls="google" aria-selected="false">Google</button>
    <button class="nav-link" id="semantic_tab" data-bs-toggle="tab" data-bs-target="#semantic" type="button" role="tab" aria-controls="semantic" aria-selected="false">Semantic</button>
  </div>
</nav>

<div class="tab-content" id="nav-tabContent">
  <!-- Первый блок (он отображается по умолчанию, т.к. имеет классы show и active) -->
  <div class="tab-pane fade show active" id="orcid" role="tabpanel" aria-labelledby="orcid_tab">
    <div class="table_panel">
      <div class="form-numrows"><label for="numrows" style="margin-right: 5px; margin-bottom: 0px;">Показати:
      </div>
      <input class="form-control" type="text" placeholder="Новик" id="search-text" onkeyup="tableSearch()">
    </div>
    <!-- Table start -->
    <div class="table-responsive">
      <div id="pagesControllers" class="pagesControllers"><button value="prev" class="pagecontroller pagecontroller-p">&lt;
      </button><button value="1" class="pagecontroller pagecontroller-num" style=""></button><button value="2" class="
      "pagecontroller pagecontroller-num" style="">2</button><button value="3" class="pagecontroller pagecontroller-num" style=
      "">3</button><button value="4" class="pagecontroller pagecontroller-num" style="">4</button><button value="5" class=
      "pagecontroller pagecontroller-num currentPage" style="">5</button><button value="next" class="pagecontroller
      pagecontroller-n">&gt;</button></div>
    </div>
    <div class="tab-pane fade show" id="scopus" role="tabpanel" aria-labelledby="scopus_tab"> <p>Тут скоро буде таблиця для scopus ;>
    <br> <br> <br> </p></div>
    <div class="tab-pane fade show" id="web" role="tabpanel" aria-labelledby="web_tab"> <p>Тут скоро буде таблиця для web ;> </p> <br>
    <br> </div>
    <div class="tab-pane fade show" id="google" role="tabpanel" aria-labelledby="google_tab"> <p>Тут скоро буде таблиця для google
    scholar ;> </p> <br> <br> </div>
    <div class="tab-pane fade show" id="semantic" role="tabpanel" aria-labelledby="semantic_tab"> <p>Тут скоро буде таблиця для
    semantic scholar ;> </p> <br> <br> </div>
  </div>
</div>
<!-- <div class="selecttabs" style="display: inline-block"><label for="numrows" style="margin-right: 5px; margin-bottom: 0px;">Показати:

```

Рис. 3.3 Розмітка табів.

Шаблон блоку таблиць містить у собі поле пошуку, спадне меню кількості видимих записів, власне таблицю й блок-перемикач сторінок. Для їх реалізації були використані шаблони form-select, form-control, table та pagesControllers відповідно. Клас table-responsive описує дизайн та механіку горизонтальної прокрутки таблиць на малих екранах, зокрема в мобільній версії.

```

<div class="tab-content" id="nav-tabContent">
  <!-- Перший блок (он отображается по умолчанию, т.к. имеет классы show и active) -->
  <div class="tab-pane fade show active" id="orcid" role="tabpanel" aria-labelledby="orcid_tab">
    <div class="table_panel">
      <div class="for_numrows " ><label for="numrows" style="margin-right: 5px; margin-bottom: 0px;">Показати:
      </label><select class="form-select" id="numrows"></select></div>
      <input class="form-control" type="text" placeholder="Поиск" id="search-text" onkeyup="tableSearch()">
    </div>
    <!-- Table start -->
    <div class="table-responsive">
      <div id="pagesControllers" class="pagesControllers"><button value="prev" class="pagecontroller pagecontroller-p">&lt;
      </button><button value="1" class="pagecontroller pagecontroller-num" style="">1</button><button value="2" class=
      "pagecontroller pagecontroller-num" style="">2</button><button value="3" class="pagecontroller pagecontroller-num" style=
      "">3</button><button value="4" class="pagecontroller pagecontroller-num" style="">4</button><button value="5" class=
      "pagecontroller pagecontroller-num currentPage" style="">5</button><button value="next" class="pagecontroller
      pagecontroller-n">&gt;</button></div>
    </div>
  </div>

```

Рис. 3.4 Розмітка блоку таблиці.

## 3.2 Створення діаграм

Була використана бібліотека chart.js, що дозволяє створити 9 основних типів діаграм, з яких нас цікавили дві: donut chart (пончик) та bar chart (стовпчикова діаграма, що, завдяки зміні значення атрибуту axes, стала рядковою). Суттєвим недоліком бібліотеки є неможливість редагувати стилі чартів, що обумовило використання плагіну datalabels для налаштування дизайну діаграм відповідно до раніше розробленого макету.



```

var ctxD = document.getElementById("doughnutChart").getContext('2d');
var myLineChart = new Chart(ctxD, {
  type: 'doughnut',
  data: {
    labels: ["Psychology", "Computer Science", "Agricultural and Biologic Sciences", "Mathematics", "Engineering",],
    datasets: [{
      data: [12, 97, 182, 82, 75],
      backgroundColor: ["#9D5A5A", "#FF877F", "#5B5689", "#738FC9", "#3DAFAF"],
      hoverBackgroundColor: ["rgba(157, 90, 90, 0.4)", "rgba(255, 135, 127, 0.4)", "rgba(91, 86, 137, 0.4)", "rgba(115, 143, 201, 0.4)", "rgba(61, 175, 175, 0.4)"],
      borderWidth: 0,
      weight: 1
    }]
  },
  plugins: [ChartDataLabels, options: {plugins: {
    title: {
    },
    legend: {
      display: false,
    },
    labels: {
      font: {
        color: 'rgba(0,0,0,0.8)',
        family: 'Rubik Light',
        size: 12,
        weight: 'light'
      },
      usePointStyle: true,
      boxWidth: 14,
      boxHeight: 14,
      padding: 15
    },
  }},
  datalabels: {
    formatter: (value) => {
      return Math.round(value*100/550) + '%';
    },
    color: "#F9FAFC",
    align: 'center',
  }
}

```

Рис. 3.5 Створення donut chart.

В процесі розробки було зроблено висновок, що плагін не вирішує усіх проблем, зокрема не дозволяє коректно відображати кругові чарти за великої кількості даних, а також налаштовувати відступи та розміри внутрішніх блоків. Частково це було виправлено створенням блоку легенди вручну за допомогою стандартної розмітки HTML та розміщенням діаграми в адаптивний контейнер.

```

var ctx1 = document.getElementById("barChart1").getContext('2d');
var myBarChart = new Chart(ctx1, {
  type: 'bar',
  data: {
    labels: ["orcid", "scopus", "web", "semantic scholar", "google scholar"],
    datasets: [{
      label: '# of Votes',
      data: [30, 70, 35, 40, 20, 40],
      backgroundColor: ["#305A5A", "#FF877F", "#585689", "#738FC9", "#30AFAF"],
      hoverBackgroundColor: ["rgba(157, 90, 90, 0.4)", "rgba(255, 135, 127, 0.4)", "rgba(91, 86, 137, 0.4)", "rgba(115, 143, 201, 0.4)", "rgba(61, 175, 175, 0.4)"],
      borderColor: [
        'rgba(255,99,132,1)',
        'rgba(54, 162, 235, 1)',
        'rgba(255, 206, 86, 1)',
        'rgba(75, 192, 192, 1)',
        'rgba(153, 102, 255, 1)',
        'rgba(255, 159, 64, 1)'
      ],
      borderWidth: 0,
      borderRadius: 9,
      // barThickness: 8,
      height: 44,
    ]
  }
},
  plugins: [ChartDataLabels], options: {
    indexAxis: 'y',
    categoryPercentage: 1,
    scales: {
      y: {
        display: false,
        beginAtZero: true,
        grid: {
          display: false,
          ticks: false
        }
      },
      x: {
        display: false,
        grid: {
          display: false,
          ticks: false
        }
      }
    },
    plugins: {
      legend: {
        display: false,
      },
      datalabels: {
        formatter: (value) => {
          return Math.round(value*100/235) + '%';
        },
        color: "#F9FAFC",
        font: {
          family: 'Rubik',
          size: 10,
          weight: 'lighter'
        },
        align: 'right',
      }
    }
  }
});

```

Рис. 3.6 Створення bar chart.

### 3.3 Інтерактивність таблиці

Налаштування інтерактивності таблиці було реалізовано за допомогою мови JavaScript. Наразі забезпечені:

- функція пошуку записів таблиці

```
function tableSearch() {
    var phrase = document.getElementById('search-text');
    var table = document.getElementById('myTable');
    var regPhrase = new RegExp(phrase.value, 'i');
    var flag = false;
    for (var i = 1; i < table.rows.length; i++) {
        flag = false;
        for (var j = table.rows[i].cells.length - 1; j >= 0; j--) {
            flag = regPhrase.test(table.rows[i].cells[j].innerHTML);
            if (flag) break;
        }
        if (flag) {
            table.rows[i].style.display = "";
        } else {
            table.rows[i].style.display = "none";
        }
    }
}
```

Рис. 3.7 Функція пошуку.

- функція сортування за зменшенням і збільшенням для будь-якого зі  
СТОВПЦІВ

```
/**
function multipleSortCol(array, data) {
    console.log(array);
    var cols = [];
    var orders = [];
    var types = [];
    var formats = [];
    for (i = 0; i < data.length; i++) {
        cols.push(data[i][0]);
        orders.push(data[i][1]);
        types.push(data[i][2]);
        formats.push(data[i][3]);
    }
    array.sort(function (a, b) {
        for (i = 0; i < cols.length; i++) {
            // initialize variables
            var first = "";
            second = "";
            var order = orders[i];
            // get inner text from stringified elements
            let firstCol = new DOMParser().parseFromString(
                a[cols[i]],
                "text/html"
            ).documentElement.textContent;
            let secondCol = new DOMParser().parseFromString(
                b[cols[i]],
                "text/html"
            ).documentElement.textContent;
            // If it's last col selected and a = b return 0
            if (i == cols.length - 1 && firstCol == secondCol) {
                return 0;
            }
            // check col type if is alphanumeric or date
            if (types[i] == "alphanumeric") {
                if (firstCol > secondCol) {
                    return order;
                } else if (firstCol < secondCol) {
                    return -1 * order;
                }
            } else if (types[i] == "date") {
                if (
                    formatDate(formats[i], firstCol) >
                    formatDate(formats[i], secondCol)
                ) {
                    return order;
                } else if (
                    formatDate(formats[i], firstCol) <
                    formatDate(formats[i], secondCol)
                ) {
                    return -1 * order;
                }
            }
        }
    });
}
```

Рис. 3.9 Функція сортування.

## - налаштування кількості відображуваних записів та зміна меню вибору сторінки

```

/**
function generatePaginationValues() {
  // get first select numRows value
  numRows = selectNumRowsVal;
  numPerPage = Math.ceil(rows.length / numRows);
  // append first controllers for pages
  appendPageControllers(numPages);
  // give currentPage class to first page number
  $(".pagecontroller-num").eq(0).addClass("currentPage");
  paginate(currentPage, numPerPage);
  pageControllersClick();
  filterPages();
}
}

/**
Pagination function: reorganize entire table by pages
curPage = (can be null) current page if it's set
perPage = (can be null) number of rows per page
**/
function paginate(curPage = null, perPage = null) {
  var curPage = curPage === null ? currentPage : curPage;
  var perPage = perPage === null ? numRows : perPage;
  Table.on("paginating", function () {
    $(this)
      .find("tbody tr")
      .hide()
      .slice(curPage * perPage, (curPage + 1) * perPage)
      .show();
  });
  Table.trigger("paginating");
}

/**
Page controllers append: generate page controllers and append them on bottom of table
**/
function appendPageControllers(nPages) {
  // reset div
  $("#pagesControllers").html("");
  // first
  $("#pagesControllers").append(
    $("<button>").append(
      {
        value: "first",
        text: "<<",
        class: "pagecontroller pagecontroller-f",
      }
    )
  );
  // Previous
  $("#pagesControllers").append(
    $("<button>").append(
      {
        value: "prev",
        text: "<",
        class: "pagecontroller pagecontroller-p",
      }
    )
  );
  // Numbers
  for (i = 1; i <= nPages; i++) {
    $("#pagesControllers").append(
      $("<button>").append(
        {
          value: i,
          text: i,
          class: "pagecontroller pagecontroller-num",
        }
      )
    );
  }
  // Next
  $("#pagesControllers").append(
    $("<button>").append(

```

Рис. 3.10 Функція пагінації.

## **Висновки:**

На етапі верстки було написано HTML розмітку сайту, надано базових стилів за допомогою фреймворку Bootstrap, згодом відредагованих засобами CSS.

Для реалізації діаграм було обрано бібліотеку chart.js, перевагами якої є швидкість, мала кількість коду, достатня кількість шаблонів та інтерактивність, тоді як недоліками – недостатня гнучкість стилізації і некоректне відображення на деяких пристроях. Ці проблеми було вирішено використанням плагіну datalabels та заміни влаштованої легенди діаграми на власну.

Для таблиць було реалізовано функції сортування, пошуку та пагінації мовою JavaScript.

## ВИСНОВКИ

Сучасний розвиток інформаційно-комунікаційних технологій розкриває широкі можливості для використання технологій у системі освіти, зокрема в повсякденному житті студентів та викладачів. Редизайн сайту університету дозволить оптимізувати процеси пошуку й аналізу необхідної інформації, а також спростити та зробити зручнішою комунікацію в рамках наукової діяльності.

Під час роботи були отримані наступні результати. По-перше, було проаналізовано особливості розробки мобільних версій сайту та приклади дизайну діаграм та таблиць. На основі результатів цих досліджень були сформульовані основні вимоги щодо функціоналу та дизайну сайту:

- мінімалістичний, лаконічний та інтуїтивно зрозумілий дизайн.
- доступне в будь-який момент взаємодії користувача з продуктом меню.
- наявність пункту “головна” або відповідної іконки.
- адаптивність.
- надання аналітичних даних здебільшого у вигляді таблиць, графіків й діаграм.

Також на цьому етапі проаналізовано функціонал плагіну Bootstrap й виділено переваги (швидкість та зручність роботи, масштабованість, легке налаштування, влаштована система сіток, легкий доступ до навчальних матеріалів і документації) та недоліки (відсутність гнучкості, шаблонність дизайнів, некоректне відображення в старих браузерях).

По-друге, було освітлено цільову аудиторію: студенти та працівники університету, - та задачі, які має виконувати продукт: полегшення збору, аналізу та презентації аналітичних даних щодо публікацій наукових робіт. В результаті було розроблено основні сценарії використання й згодом створено діаграму user flow, яка дозволила їх візуалізувати.

По-третє, було спроектовано UX макет, у ході розробки якого було визначено елементи інтерфейсу, блоків контенту та оптимізована навігація.

По-четверте, на останньому етапі розробки макету було визначено кольорову палітру, стилі шрифтів та основних компонентів. Для кожного з відповідних елементів створено UI kits, на основі яких згодом було спроектовано UI дизайн продукту.

По-п'яте, була реалізована верстка усіх сторінок для мобільної версії сайту. В процесі використані мови HTML, CSS та JavaScript, фреймворк Bootstrap, бібліотеку для створення діаграм Chart.js і плагін DataLabels.

Сайт Publication містить корисний функціонал: інтерактивні діаграми, таблиці та інтерфейс взаємодії з ними. Інтерфейс перш за все націлений на зручність й відрізняється лаконічністю, зрозумілістю для користувача та ненав'язливістю.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Как выбрать шрифт — руководство для начинающих: [он-лайн ресурс]. URL: <https://vc.ru/flood/23884-typography-basics> (дата останнього звернення: 10.05.2021).
2. Как составлять User Experience Flow: [он-лайн ресурс]. URL: <http://progamedev.net/kak-sostavlyat-user-experience-flow/> (дата останнього звернення: 22.05.2020).
3. Особенности дизайна мобильных приложений: [он-лайн ресурс]. URL: <https://webevolution.ru/blog/ajdentika/osobennosti-dizajna-mobilnih-prilozhenij/> (дата останнього звернення: 10.05.2021).
4. Процесс разработки дизайна мобильного приложения с нуля: [он-лайн ресурс]. URL: <https://medium.com/product-design/%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81-%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8-%D0%B4%D0%B8%D0%B7%D0%B0%D0%B9%D0%BD%D0%B0-%D0%BC%D0%BE%D0%B1%D0%B8%D0%BB%D1%8C%D0%BD%D0%BE%D0%B3%D0%BE-%D0%BF%D1%80%D0%B8%D0%BB%D0%BE%D0%B6%D0%B5%D0%BD%D0%B8%D1%8F-%D1%81-%D0%BD%D1%83%D0%BB%D1%8F-909c0443ef92> (дата останнього звернення: 10.05.2021).
5. Разработка дизайна мобильных приложений. 7 тонкостей, о которых вы должны знать: [он-лайн ресурс]. URL: <https://artjoker.ua/ru/blog/razrabotka-dizayna-mobilnykh-prilozheniy-7-tonkostey-o-kotorykhy-dolzheny-znat/> (дата останнього звернення: 10.05.2021).
6. Что строить в первую очередь: User Journey Map или User Flow?: [он-лайн ресурс]. URL: <https://medium.com/%D0%BD%D0%B0%D1%87%D0%B8%D0%BD%D0%B0%D1%8E%D1%89%D0%B5%D0%BC%D1%83-ux-%D0%B4%D0%B8%D0%B7%D0%B0%D0%B9%D0%BD%D0%B5%D1%80%D1%83/%D1>



[%87%D1%82%D0%BE-%D1%81%D1%82%D1%80%D0%BE%D0%B8%D1%82%D1%8C-%D0%B2-%D0%BF%D0%B5%D1%80%D0%B2%D1%83%D1%8E-%D0%BE%D1%87%D0%B5%D1%80%D0%B5%D0%B4%D1%8C-user-journey-map-%D0%B8%D0%BB%D0%B8-user-flow-4ece3ed60fb5](https://habr.com/ru/post/496760/) (дата останнього звернення: 10.05.2021).

7. 10 универсальных правил для дизайна мобильных приложений: [он-лайн ресурс]. URL: [https://geekbrains.ru/posts/mobile\\_design\\_principles](https://geekbrains.ru/posts/mobile_design_principles) (дата останнього звернення: 10.05.2021).

8. UI kit — что это и зачем дизайнеру: [он-лайн ресурс]. URL: [https://skillbox.ru/media/design/ui\\_kit\\_chno\\_eto\\_i\\_zachem/](https://skillbox.ru/media/design/ui_kit_chno_eto_i_zachem/) (дата останнього звернення: 10.05.2021).

9. User flow: как создаются популярные приложения и сайты: [он-лайн ресурс]. URL: <https://habr.com/ru/post/496760/> (дата останнього звернення: 10.05.2021).