

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
Кафедра інженерії програмного забезпечення

ЛАБОРАТОРНА РОБОТА №3

з дисципліни «Вебтехнології»

на тему: «Вибір користувацького інтерфейсу»

ЛРІПЗ.2201101.01.08.ЛР  
(шифр, назва)

**Виконав:**

Студент 3 курсу, група	ІПЗ-22-1	_____	Владислав КВІТ
	(шифр)	(підпис)	(ініціали, прізвище)

**Перевірив:**

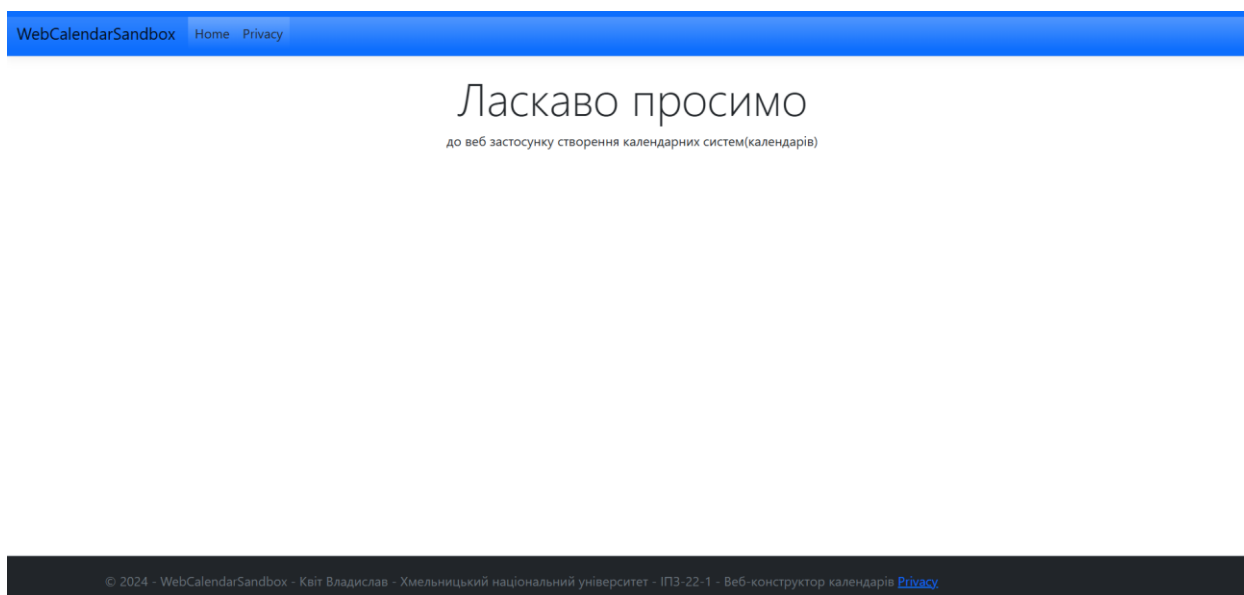
_____	Оксана ЯШИНА
(підпис)	(ініціали, прізвище)

**Мета роботи:** здійснити вибір тих функціональних можливостей, які слід об'єднати, щоб відповідати потребам користувацького вебінтерфейсу застосунку.

## ЗВІТ

Завдання: Здійснити реалізацію користувацького інтерфейсу інтернет-магазину, онлайн-ресурсу, маркетплейсу або іншого вебзастосунку відповідно до тематики курсового проекту

Шляхом редагування файлу `_Layout.cshtml` було обрано користувацький інтерфейс. На рисунку нижче показано, як він виглядатиме на прикладі головної сторінки.



## Контрольні запитання

1. Які технології дозволяють здійснювати розробку користувацького інтерфейсу?
2. Що таке ASP.NET Core Blazor?
3. Як функціонує Razor Pages в ASP.NET Core?
4. Вказати переваги використання Blazor?
5. Вказати недоліки використання Blazor?

6. Як можна здійснити розробку користувацького інтерфейсу за допомогою Angular?

7. Охарактеризуйте основні переваги ASP.NET Core MVC?

8. Які переваги Razor Pages в ASP.NET Core?

1. Для розробки користувацького інтерфейсу використовують HTML, CSS і JavaScript, а також фреймворки та бібліотеки, такі як Angular, React, Vue.js, Blazor (ASP.NET Core), а також технології, зокрема Razor Pages та ASP.NET Core MVC, що забезпечують зручне створення динамічних веб-інтерфейсів.

2. ASP.NET Core Blazor — це фреймворк для створення інтерфейсу користувача, що дозволяє писати інтерактивні веб-додатки з використанням C# замість JavaScript. Blazor підтримує два режими роботи: серверний (Blazor Server) та на стороні клієнта з використанням WebAssembly (Blazor WebAssembly).

3. Razor Pages — це функція в ASP.NET Core для створення сторінок з інтерфейсом користувача. Вона дозволяє поєднувати логіку контролера і представлення в одному файлі, що спрощує розробку для сторінок, зосереджених на виконанні конкретних завдань (наприклад, форми).

4. Переваги використання Blazor:

-Можливість використання C# для розробки як серверної, так і клієнтської частини;

-Висока продуктивність завдяки WebAssembly у Blazor WebAssembly;

- Розширені можливості інтеграції з .NET екосистемою та повторне використання коду;

- Підтримка реального часу у Blazor Server, що знижує час відгуку.

## 5. Недоліки використання Blazor:

- Обмеження продуктивності в Blazor WebAssembly на низькопотужних пристроях;

- Залежність від завантаження великого обсягу файлів для WebAssembly;

- Потреба в стабільному підключенні до сервера у Blazor Server;

- Відносно нова технологія, що може мати обмежену підтримку сторонніх бібліотек.

## 6. Angular — це фреймворк для створення динамічних веб-застосунків, що використовує TypeScript для побудови компонентів, сервісів та маршрутів. Angular дозволяє створювати SPA (односторінкові застосунки) з інтерфейсом, який оновлюється динамічно, без повного перезавантаження сторінки.

## 7. Основні переваги ASP.NET Core MVC:

- Чітке розділення між логікою та інтерфейсом завдяки архітектурі MVC;

- Зручний спосіб управління маршрутизацією за допомогою контролерів;

- Підтримка моделей та прив'язка даних, що знижує кількість ручного коду;

- Легкість валідації даних та інтеграції з іншими технологіями ASP.NET Core.

## 8. Переваги Razor Pages в ASP.NET Core:

- Простота для створення сторінок, зосереджених на певних завданнях;
- Об'єднання логіки і відображення в одному файлі (Page Model), що спрощує організацію коду;
- Оптимізовані для сценаріїв CRUD, оскільки кожна сторінка фокусується на конкретній функції;
- Легкість у налаштуванні маршрутизації без необхідності контролерів.