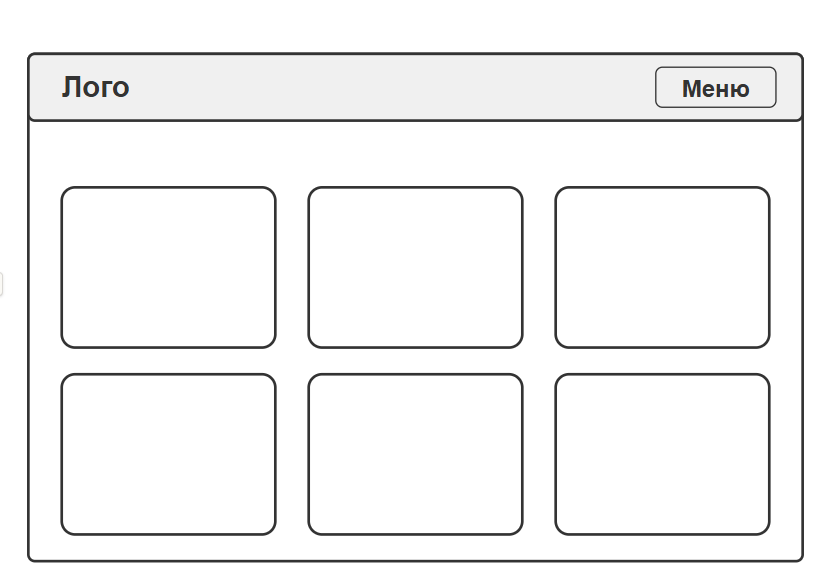
**РОЗДІЛ 5. СТВОРЕННЯ ВЕБ-ДОДАТКУ, ЯК ОСНОВИ РОЗРОБЛЮВАНОЇ АСУП**

**5.1.** **Розробка макетів для інтерфейсів користувачів та їх дизайну**

Макет інтерфейсу користувача - це візуальне представлення дизайну веб-сторінки або додатку, яке відображає його структуру, розташування елементів і вигляд. Це є прототип або карта, яка показує, як буде виглядати інтерфейс для користувача.

*Головний екран:*

* Верхнє меню з логотипом та навігаційними посиланнями.
* Кнопка "Пошуку запису" для клієнта.
* Вибір філіалу для подальшого бронювання.

Рис. 5.1 – Макет інтерфейсу користувача.

*Сторінка вибору послуги:*

* Перелік послуг філіалу з вказанням ціни та потрібного часу для послуги.
* Кнопки вибору послуги.



Рис. 5.2 – Макет сторінки «Вибір послуги»*.*

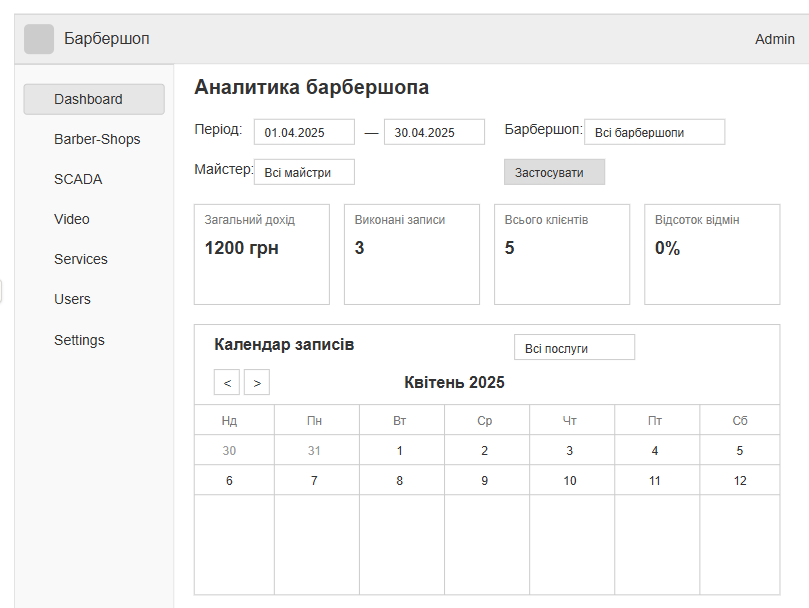


Рис. 5.3 – Макет інтерфейсу системного користувача (системна панель).

**5.2. Верстка дизайн-макетів**

При формуванні інтерфейсних архітектур фахівець має брати до уваги множинність критеріїв: візуальну гармонізацію, інтеракційні парадигми, ергономічні характеристики та корпоративну ідентичність. Хроматичні рішення, типографічні системи, графічні композиції та функціональні елементи інтегруються для досягнення оптимальної візуальної експресії.

Імплементація веб-інтерфейсів реалізується через диверсифікований технологічний інструментарій, що забезпечує конструювання, конфігурацію та презентацію контентних структур. Фундаментальними технологічними компонентами у цьому контексті виступають:

Гіпертекстова мова розмітки (HTML) постає як базовий семантичний інструментарій, що детермінує структурну ієрархію та таксономію елементів інтерфейсу. За її допомогою артикулюються інформаційні блоки, текстові конструкції, табуляційні представлення, інтеракційні форми та інші структурні одиниці.

Каскадні таблиці стилів (CSS) функціонують як механізм естетичної модифікації та стилістичної уніфікації контенту. Даний інструментарій уможливлює параметризацію метричних характеристик, хроматичних атрибутів, типографічних особливостей, просторової організації, демаркаційних елементів, фонових зображень та інших візуальних детермінант.

JavaScript виступає як програмна платформа, що імплементує інтерактивність та процедурну варіативність інтерфейсів. Ця технологія застосовується для реалізації функціональних алгоритмів: кінетичних трансформацій, валідаційних процедур, користувацьких інтеракцій, асинхронної комунікації з серверними системами тощо.

Архітектурні фреймворки (Bootstrap, Foundation) та функціональні бібліотеки (jQuery) оптимізують процеси розробки інтерфейсів. Вони пропонують прекомпільовані модулі, структурні шаблони, стилістичні специфікації та функціональні абстракції для експедитивної розробки веб-систем.

Синергічна інтеграція вищезазначених технологій уможливлює створення ергономічних, естетично довершених та функціонально насичених веб-ресурсів, що відповідають сучасним критеріям користувацького досвіду.

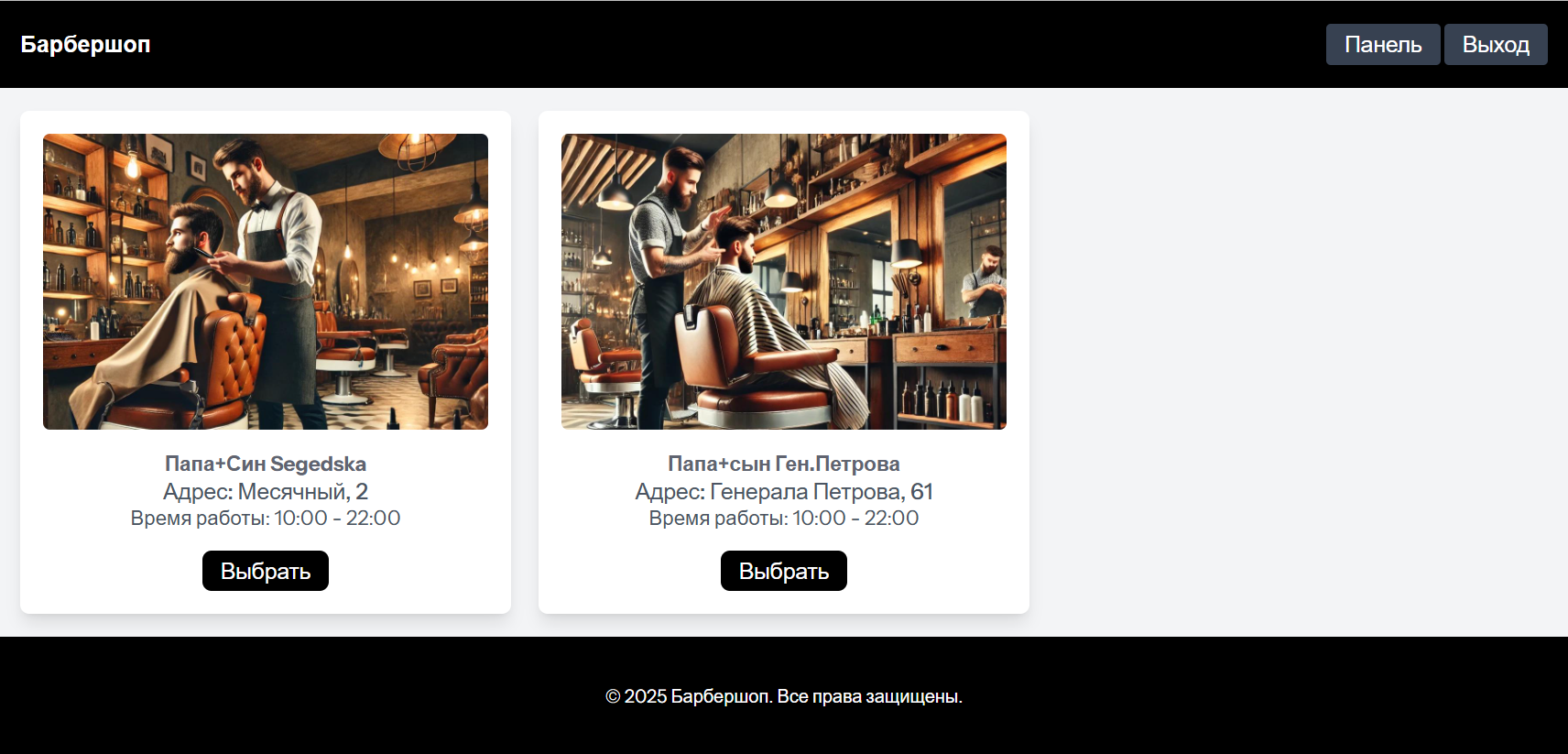


Рис. 5.4 – Головна сторінка.



Рис. 5.5 – Функція головної сторінки з перевіркою наявності активного бронювання часу.

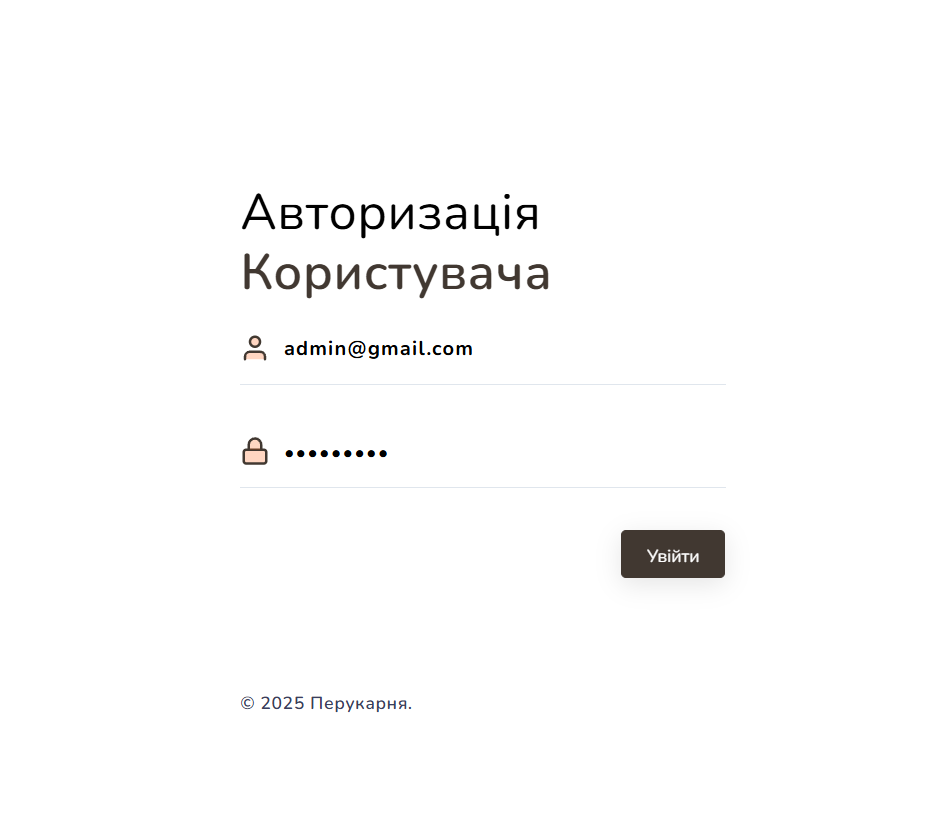


Рис. 5.6 – Форма входу.



Рис. 5.7 – Верстка форми входу.

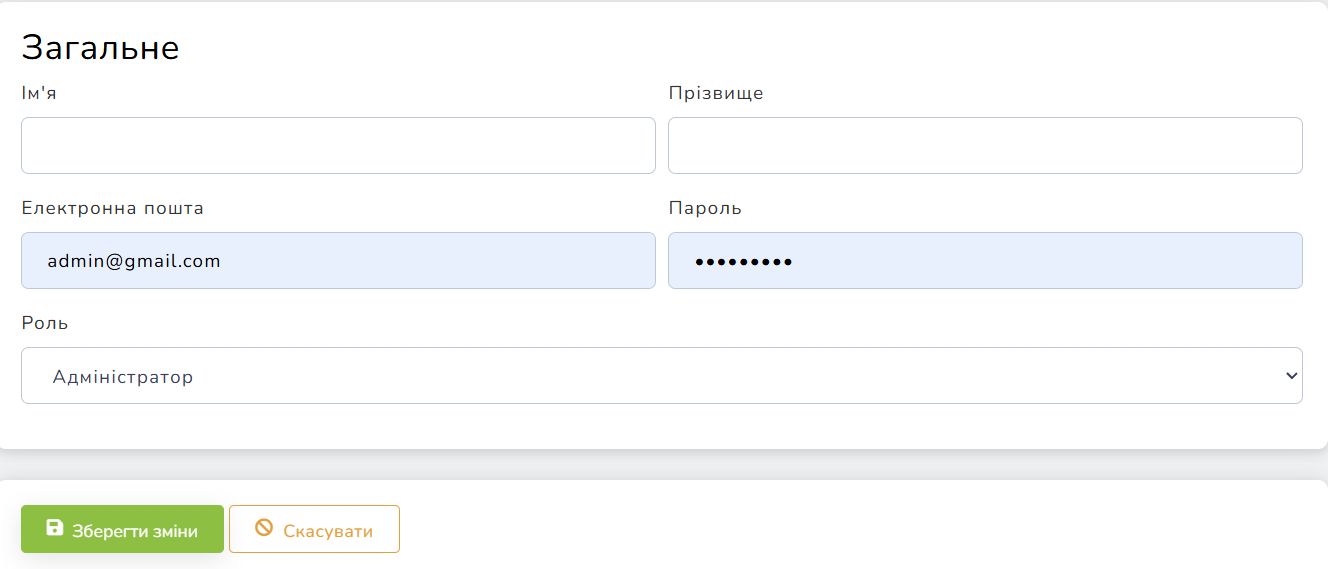


Рис. 5.8 – Форма створення нового користувача.

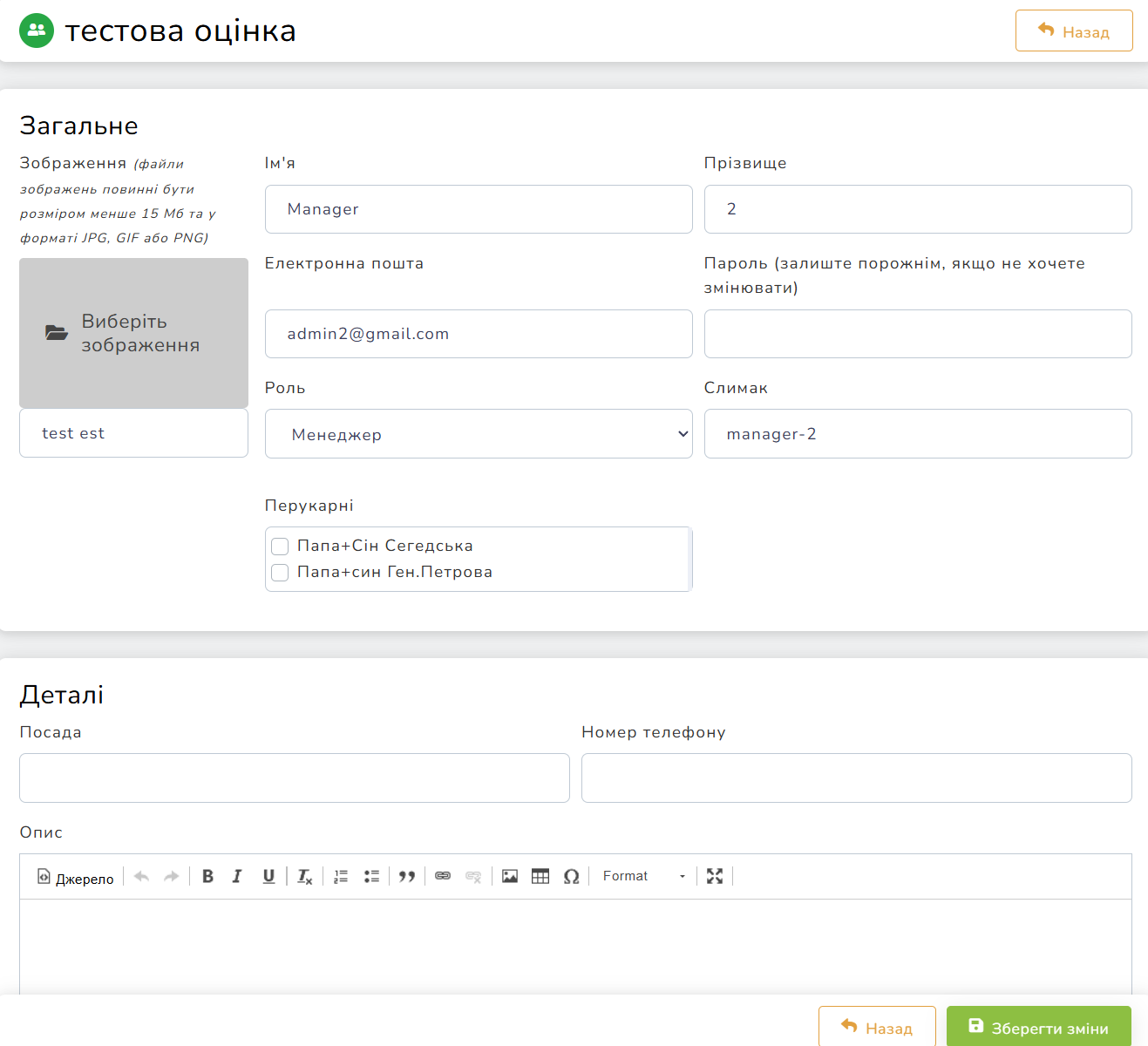


Рис. 5.9 – Сторінка системного користувача «Менеджера».



Рис. 5.10 – Верстка системної сторінки послуг

**5.3. Створення функціональних модулів програмного продукту**

Під час розробки функціональних модулів програмного продукту необхідно врахувати деякі критерії, щоб забезпечити їх якість та ефективність:

1. Модуль повинен виконувати свою основну функцію та досягати поставленої мети. Це означає, що він повинен обробляти вхідні дані або події, виконувати необхідні обчислення або операції, та повертати очікувані результати або зміни стану.

2. Модуль повинен працювати стабільно та надійно. Він має бути придатним для роботи при заданому навантаженні, стійким до помилок, здатним виявляти та обробляти виняткові ситуації, а також відновлюватися після збоїв.

3. Модуль повинен бути добре структурованим та розділеним на логічні компоненти. Це допомагає полегшити процес розробки, тестування та підтримки.

4. Модуль повинен бути легким у розумінні та управлінні. Це означає, що код модуля має бути зрозумілим для розробників із відповідними коментарями та документацією, а також повинна бути забезпечена можливість ефективного управління модулем.

5. Модуль повинен бути захищеним від потенційних загроз безпеці. Це означає, що необхідно проводити валідацію вхідних даних, обмежувати доступ до конфіденційної інформації, захищати його від зловмисних атак та забезпечувати цілісність даних.

Визначимо функціональні модулі кожної ролі.

*Менеджер:*

* Замовлення.

*Адміністратор:*

* Користувачі.

*Керівник:*

* Відеоспостереження*.*
* Сировина.
* Продукція.
* Замовлення сировини.
* Користувачі.

*Завідуючий складом:*

* Пошук деталей (штрих-код)*.*
* Замовлення сировини.
* Відеоспостереження

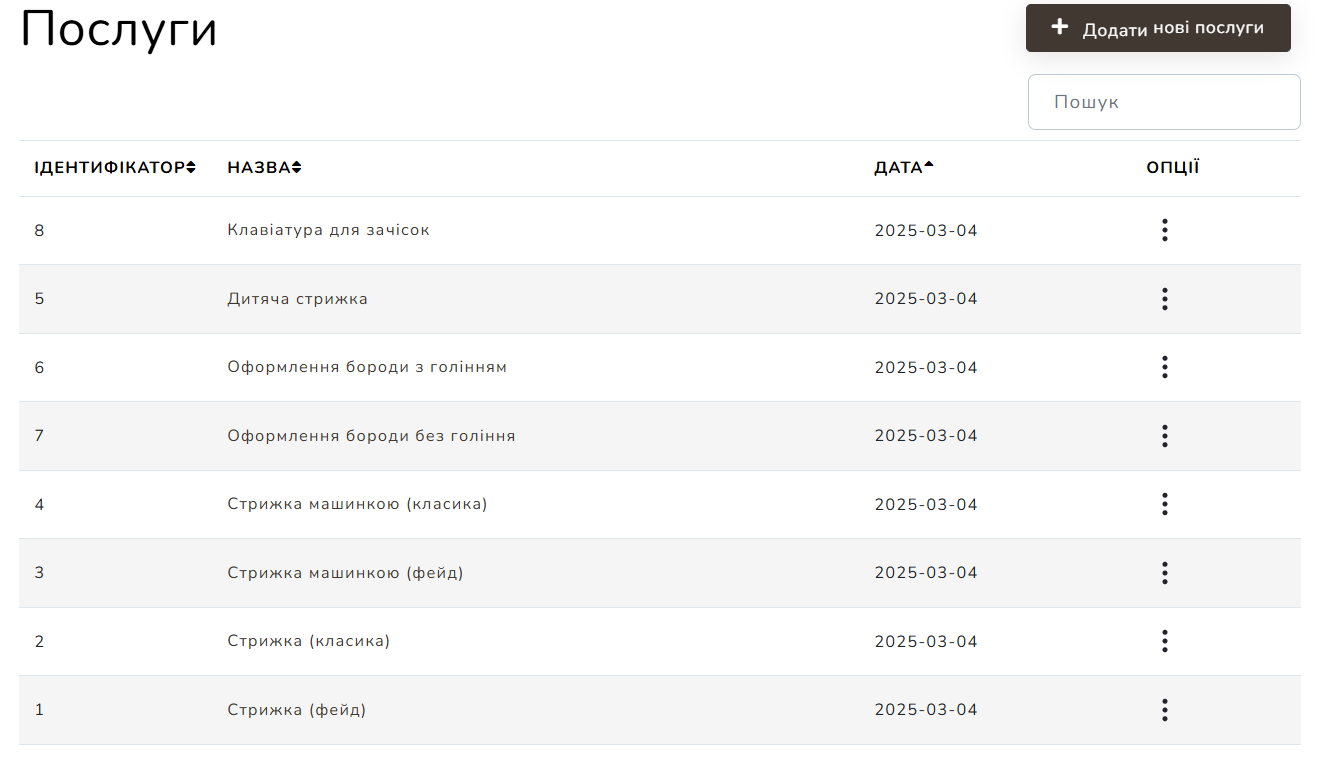


Рис. 5.11 – Модуль «Послуги» головна сторінка.

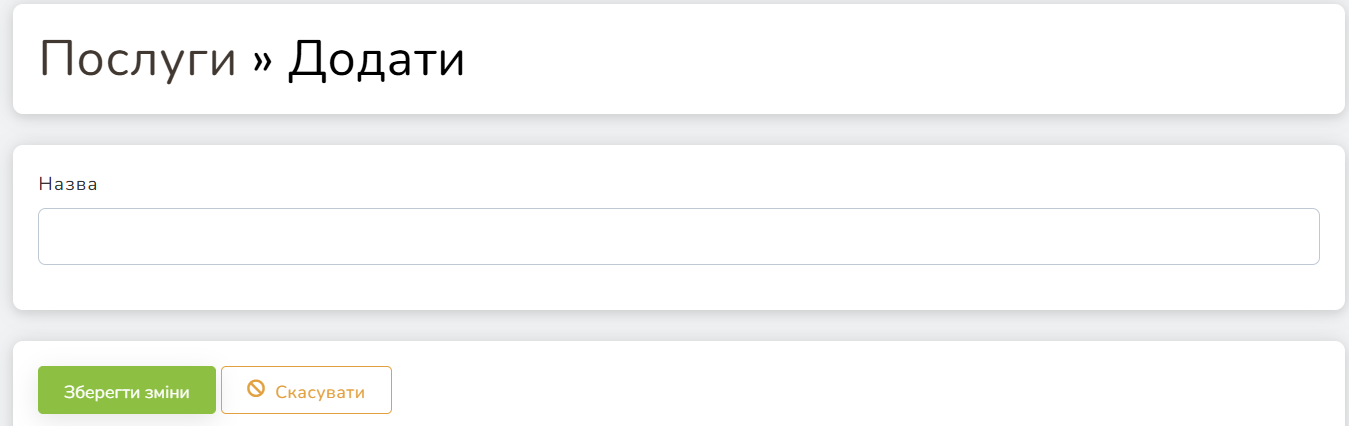


Рис. 5.12 – Модуль «Послуги» вікно додавання продукту.



Рис. 5.13 – Функція додавання продукту.

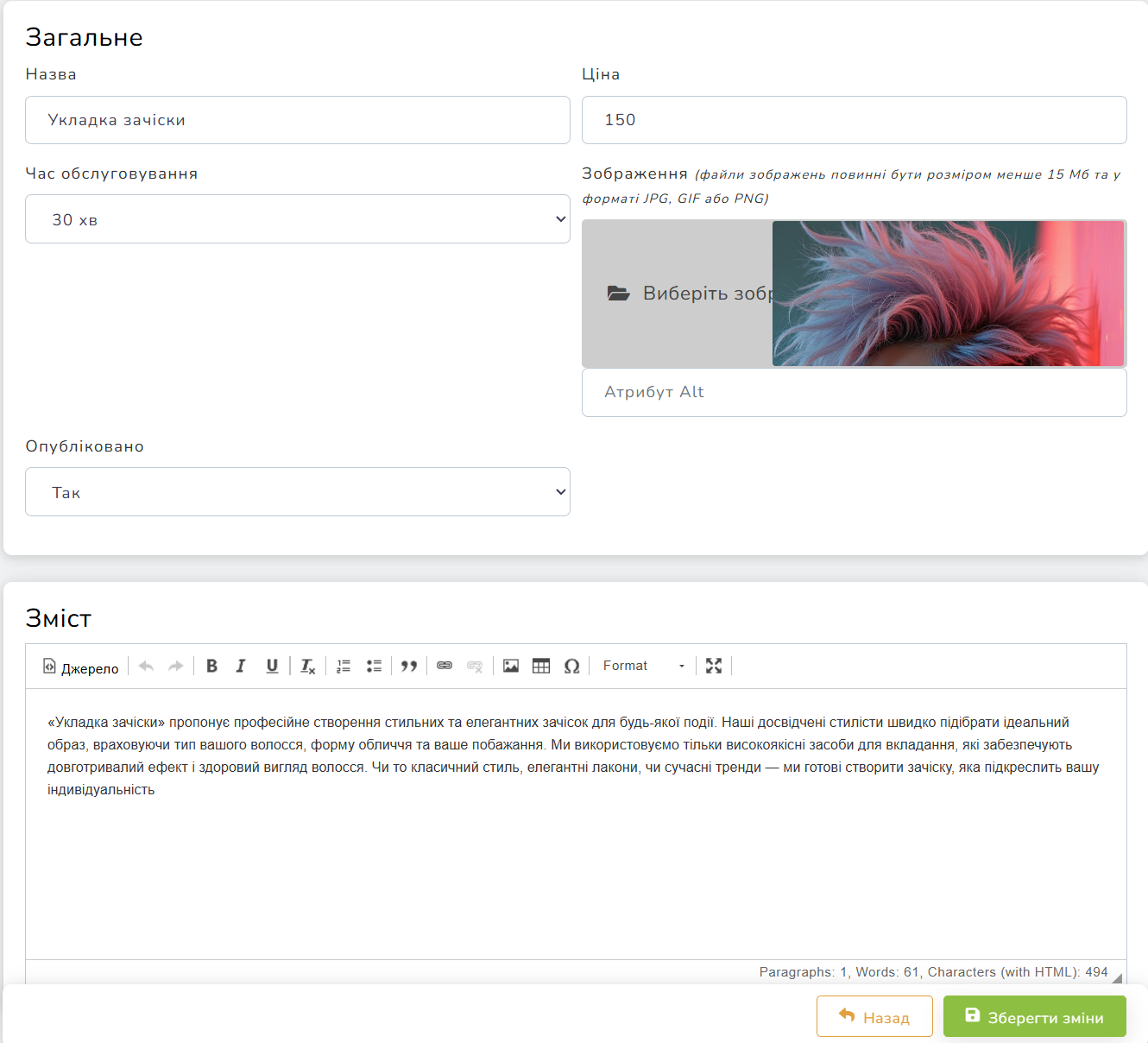


Рис. 5.14 – Модуль «Послуги» вікно редагування послуги.



Рис. 5.15 – Функція редагування послуг.

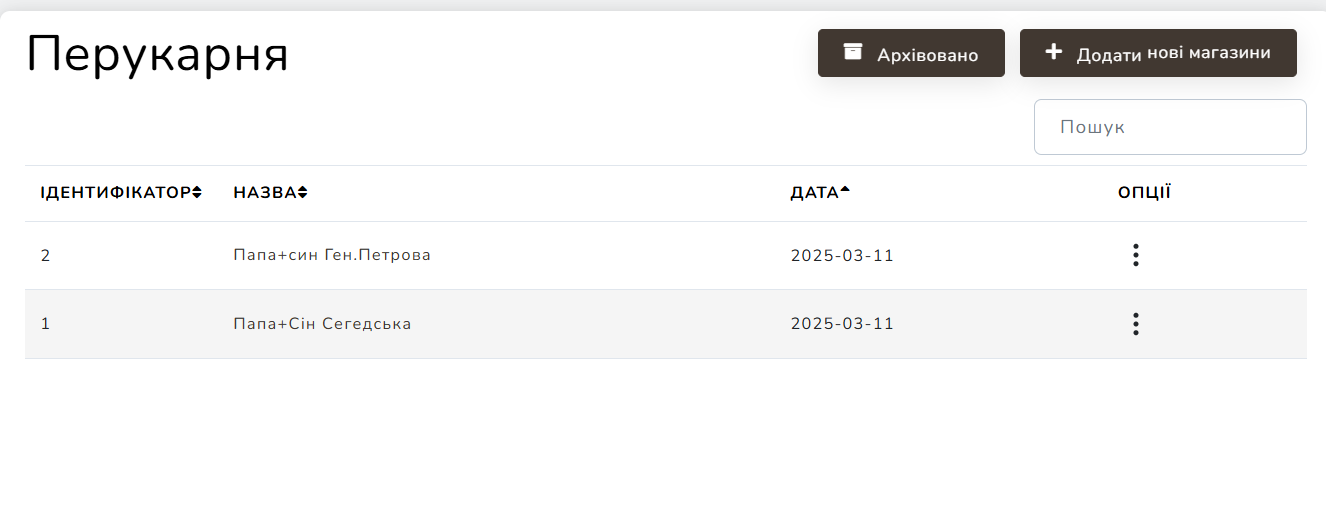


Рис. 5.16 – Модуль «Перукарні» головна сторінка.

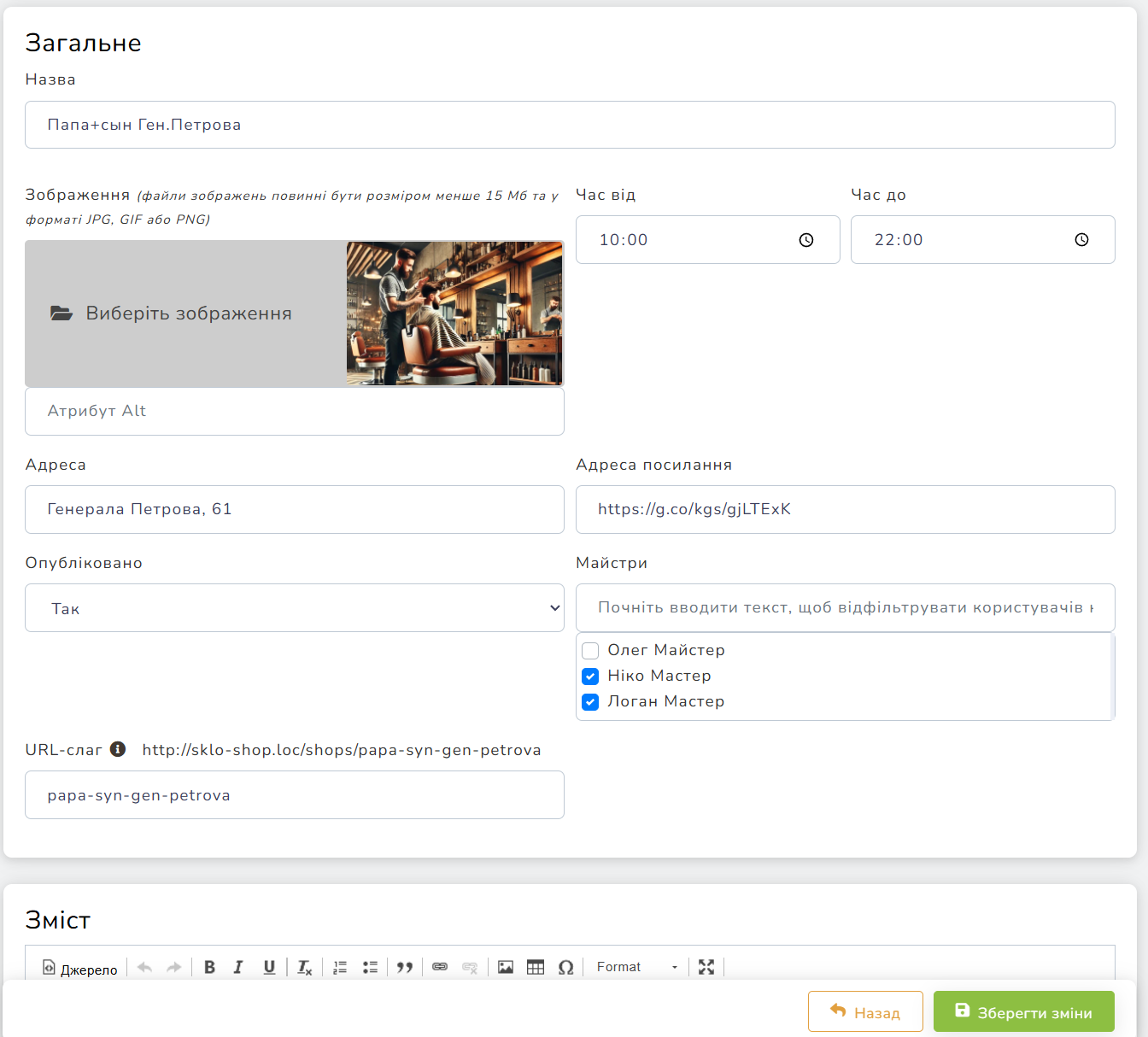


Рис. 5.17 – Модуль «Перукарні» вікно редагування перукарні.



Рис. 5.18 – Верстка сторінки редагування замовлення.

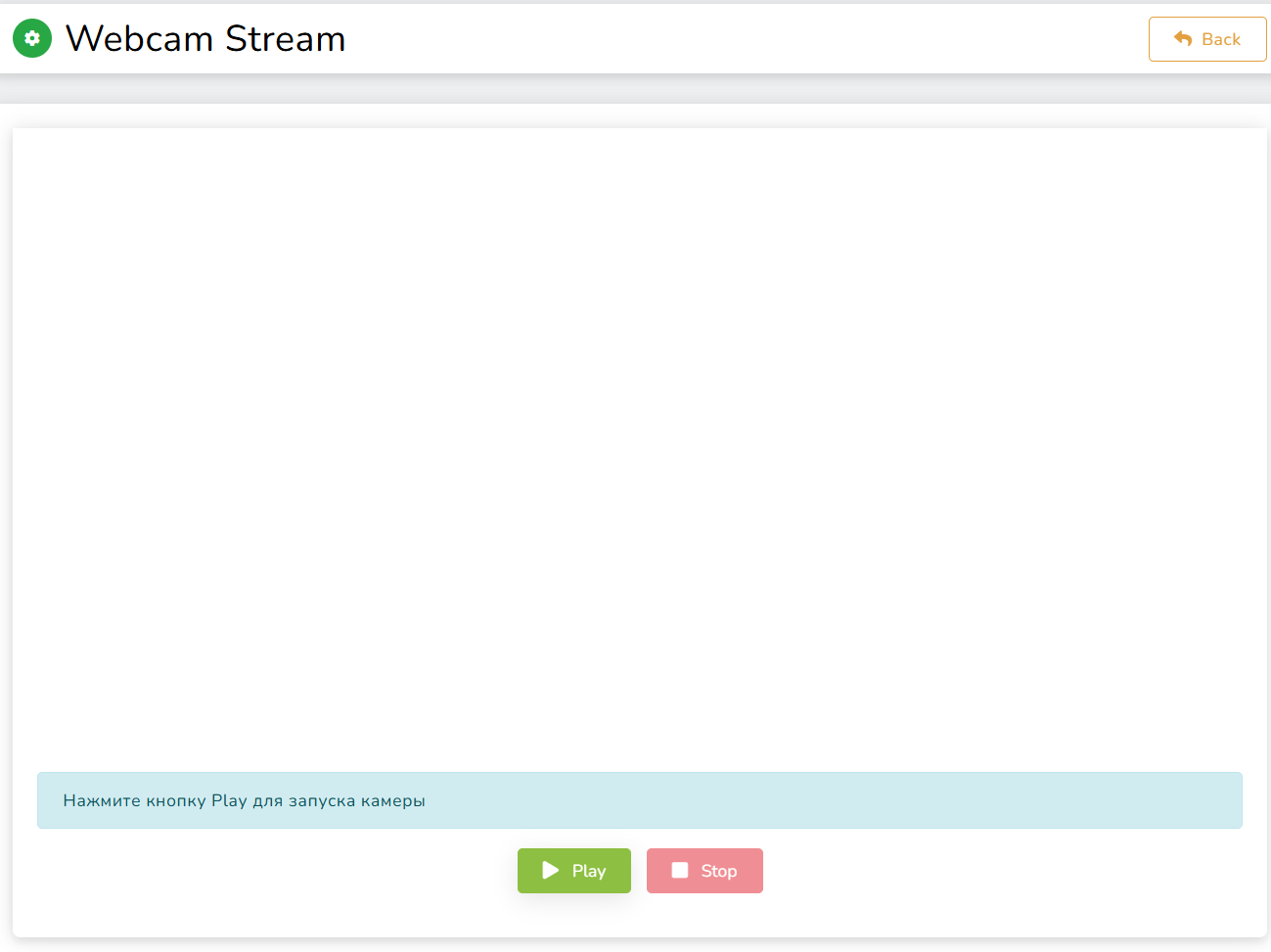


Рис. 5.19 – Модуль «Відео».



Рис. 5.20 – Скрипт реалізації відеоспостереження та його стилі.

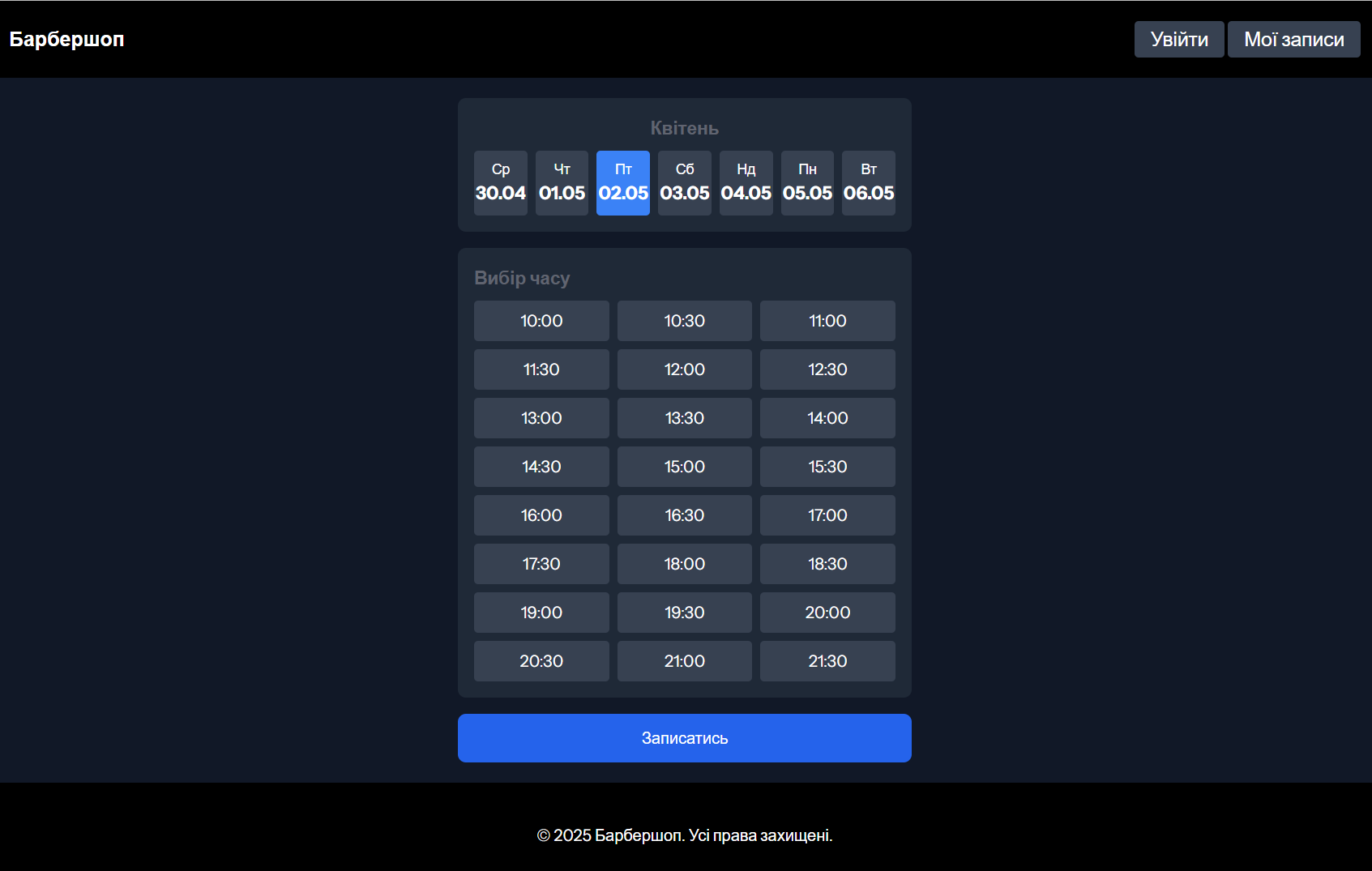


Рис. 5.21 – Модуль «Запис на послугу перукарні».



Рис. 5.22 – Функція додавання запису на послугу.



Рис. 5.23 – Функція створення pdf бронювань.

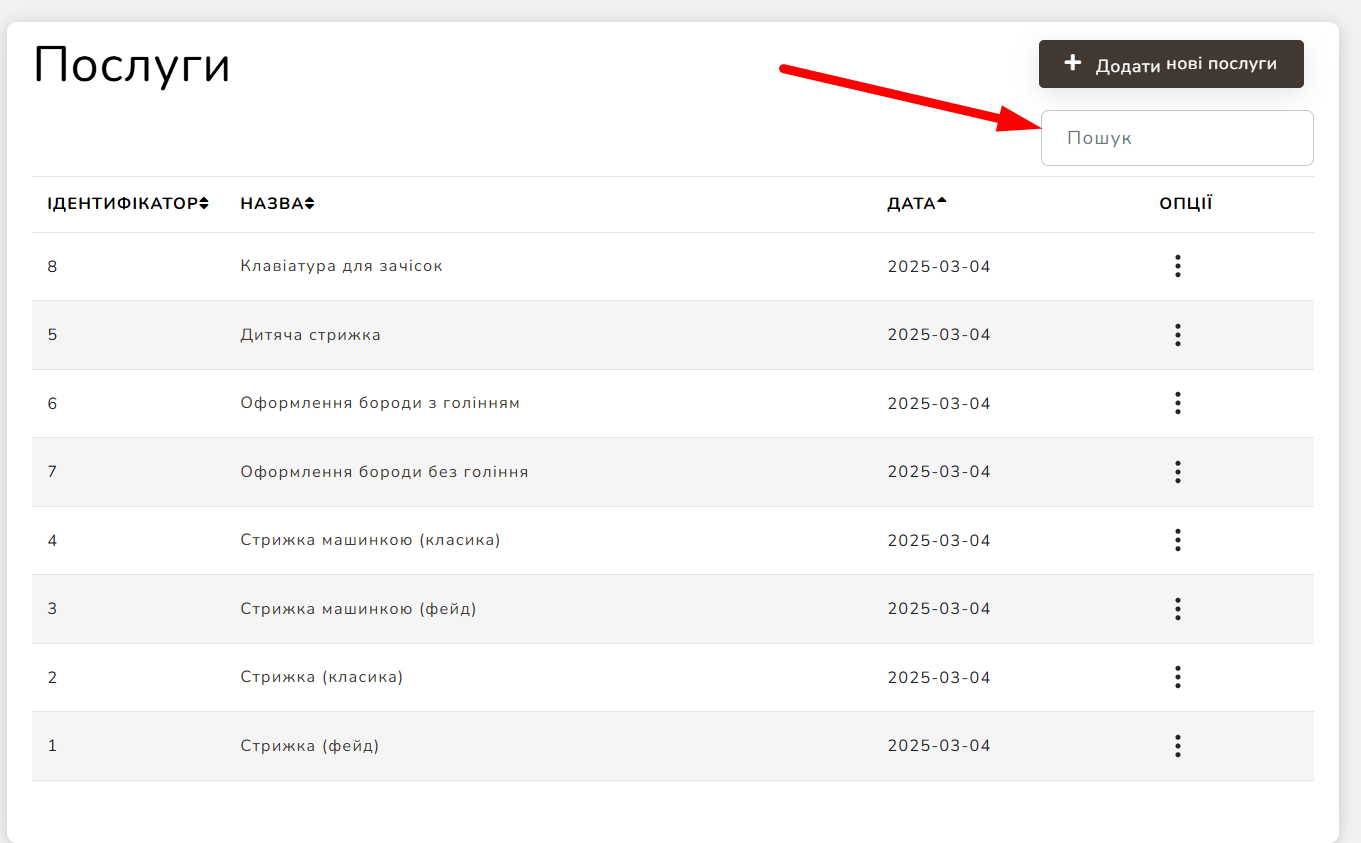


Рис. 5.24 – Поле пошуку послуги.

**5.4. Опис реалізованої програмної розробки (інструкція користувача)**

1. **Менеджер:**

a. Бронювання:

* + Переглядайте список поточних бронювань.
  + Створюйте нові бронювання в системі.
  + Оновлюйте статус виконання бронювань.
  + Спілкуйтеся з клієнтами.

1. **Адміністратор:**

a. Користувачі:

* + Додавайте нових користувачів до системи та надавайте їм необхідні ролі.
  + Забирайте права доступу у користувачів, які більше не мають їх використовувати.

1. **Власник:**

a. Користувачі:

* + Додавайте нових користувачів до системи та надавайте їм необхідні права.
  + Редагуйте вже створених користувачів.

b. Замовлення:

* + Переглядайте список поточних бронювань.
  + Створюйте нові бронювання в системі.
  + Оновлюйте статус виконання бронювань.

c. Моє замовлення:

* + Переглядайте свої бронювання по номеру клієнта.
  + Отримуйте повідомлення про зміни в статусі замовлень.

d. Послуги:

* + Переглядайте список доступних послуг.
  + Додавайте нові послуги до системи.
  + Видаляйте застарілу інформацію або послуги.

f. Відео:

* + Дивіться навчальні відео з процесу надання послуг перукарн.

1. **Користувач:**

a. Сервіси:

* + Бронюйте вільну дату та час на обрану послугу.
  + Оформлюйте замовлення для подальшої обробки працівниками закладу.
  + Перевіряйте свій запис

Кожна роль повинна авторизуватися в системі використовуючи свій індивідуальний обліковий запис. Будьте обережні зі збереженням своїх облікових даних та не розголошуйте їх іншим користувачам.