**Software Requirements Specification**

War Notes

1. **Вступ**
   1. **Призначення**

Метою цього застосунку є:

* надання користувачам необхідної інформації у разі екстреної ситуації;
* надання теоретичних та практичних знань у випадку повітряних тривог
* навчити користувачів розпізнавати ознаки небезпеки
* надання експертних порад щодо першої медичної допомоги в надзвичайних ситуаціях
* доступ до контактів все-можливих допоміжних служб

Користувач матиме можливість зареєструватися та отримати доступ до всієї інформації, також лайкати статті, зберігати їх та переглядати їх в особистому кабінеті.

* 1. **Умовні позначення**

|  |  |
| --- | --- |
| DB | Database |
| API | Application Programming Interface |
| UML | Unified Modelling Language |

* 1. **Цільова категорія**

Цей застосунок надаватиме всю необхідну інформацію у разі екстреної ситуації. Він буде корисний для користувачів будь-якого віку, так як наданий матеріал буде легкодоступний та зрозумілий для кожного споживача. Усі статті будуть інформативними та допоможуть прийняти правильне рішення в ускладненій життєвій ситуації.

* 1. **Обсяг** **проекту**

Система являє собою застосунок заснований на базі даних з 4 таблицями:  
Перша відповідатиме за збереження користувачів  
Друга відповідатиме за збереження статей

Третя відповідатиме за вподобані користувачем статті

Четверта відповідатиме за збережені користувачем статті

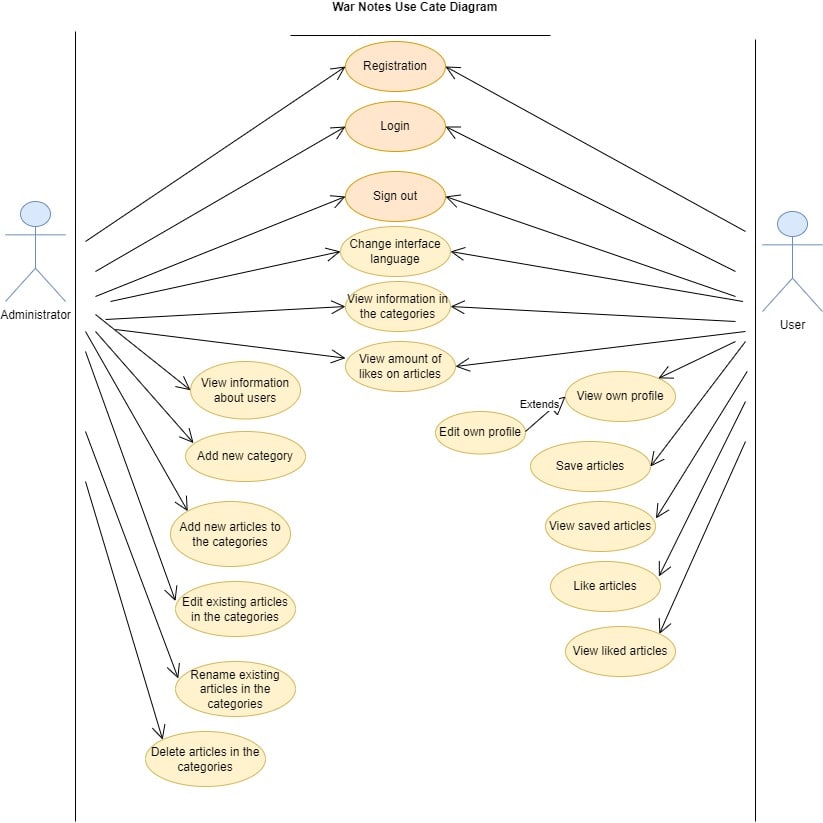
П’ята відповідатиме за категорії

1. **Загальний опис**
   1. **Перспективи продукту**

Застосунок матиме такі переваги(можливості):

* База даних статей, до якої матиме доступ адміністратор. Ця функція дозволить адміністратору додавати статті, редагувати їх та видаляти.
* Список вподобаних та збережених статей. Користувач зможе відслідковувати скільки статей він вподобав впродовж певного періоду часу та переглядати статті, які він зберіг.
  1. **Особливості продукту**
* Користувач, що не зареєструвався, зможе це зробити. Для цього він повинен ввести прізвище, ім’я, логін(email) та пароль .
* Якщо користувач вводить існуючий логін, він повинен отримати сповіщення, що такий логін вже використано іншим користувачем.
* Якщо користувач вводить новий логін та пароль, він повинен отримати сповіщення про успішну реєстрацію.
* Зареєстрований користувач може увійти в свій акаунт. Для цього йому потрібно ввести логін та пароль.
* Якщо користувач вводить невірний логін або пароль, він має отримати сповіщення про це. Також система повинна запропонувати йому ввести дані ще раз.
* Якщо користувач ввів всі дані правильно, то він входить в основну частину додатку.
* Зареєстрований користувач може редагувати та переглядати свій профіль.
* Зареєстрований користувач може змінити мову застосунку
* Зареєстрований користувач може лайкати статті
* Зареєстрований користувач може переглянути список вподобаних дописів
* Зареєстрований користувач може зберігати статті
* Зареєстрований користувач може переглянути список збережених дописів
* Зареєстрований користувач може переглянути кількість вподобань на статті
* Адміністратор може увійти в систему та переглядати всі статті, як і кожен користувач
* Адміністратор може перейменовувати статті, додавати, редагувати та видаляти їх
* Адміністратор може переглядати інформацію про користувачів
* Адміністратор може додавати нові категорії
* Адміністратор може переглянути кількість вподобань на статті
  1. **Користувачі та їх можливості**

Всю інформацію подано в usecase UML діаграмах:



* 1. **Середовище**

База даних : PostgreSQL

Клієнт/сервер : WPF

CI/CD Environment: GitHub

* 1. **Припущення та залежності**

Основне припущення перед початком роботи над цим проектом є те, що сучасне суспільство хоче бути обізнаним у таких поняттях:

* перша медична допомога
* дії під час повітряної небезпеки
* розпізнання небезпечних об’єктів
* куди звернутися, якщо потрібна допомога

Більша частина людей не мають базових знань чи навичок у цих сферах, тому застосунок гарантовано буде вирішувати проблему незнання. Так як застосунок не вимагатиме підключення до інтернету, користувач завжди буде мати можливість отримати необхідну йому інформацію.

1. **Характеристики системи**
   1. **Опис застосунку та пріорітет**

Основної ідеєю цього проекту є розробка зручного та простого в користуванні застосунку, щоб надати необхідну інформацію користувачам та покращити вже існуючі знання. Застосунок є актуальним для кожного тому немає якоїсь окремої цільової аудиторії.

Основними пріорітетами проекту є :

* актуальність
* зручність
* доступність
* якісність
  1. **Послідовність дій**

Користувач створює свій обліковий запис ввівши необхідні дані.

Користувач входить в систему через логін та пароль.

Користувач обирає категорію для вивчення тої чи іншої інформації.

Користувач може лайкати та зберігає статті.

Користувач може переглянути списки статей, які він зберіг та вподобав.

Адміністратор може змінювати, додавати та редагувати статті.

Адміністратор може спостерігати за списками користувача.

Проблема : реєстрація з новим логіном та паролем

Відповідь : користувача створено

Проблема : реєстрація з існуючим логіном

Відповідь : користувача не створено, повідомлення про помилку

Проблема : обрати категорію

Відповідь : категорія обрана та виділяється підсвічуванням

Проблема : вподобати статтю

Відповідь : статтю вподобано (позначено певним значком) та додано до папки Вподобані

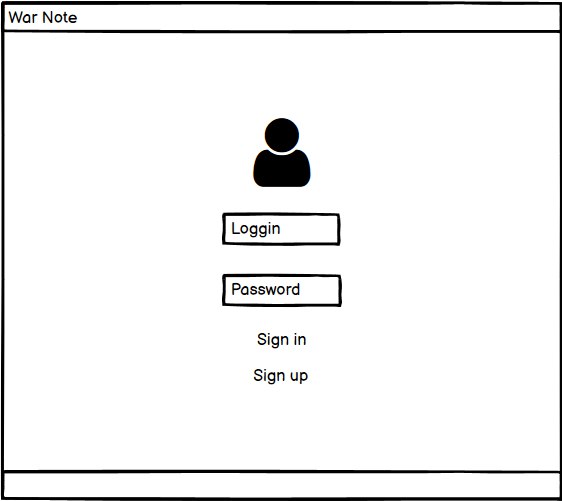
Проблема : зберегти статтю

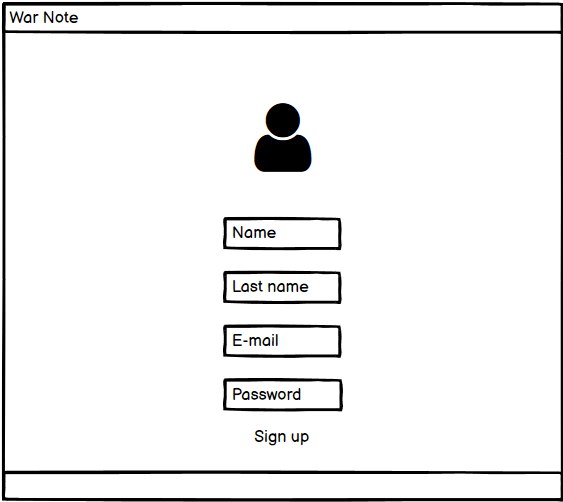
Відповідь : статтю збережено (позначено певним значком) та додано до папки Збережені

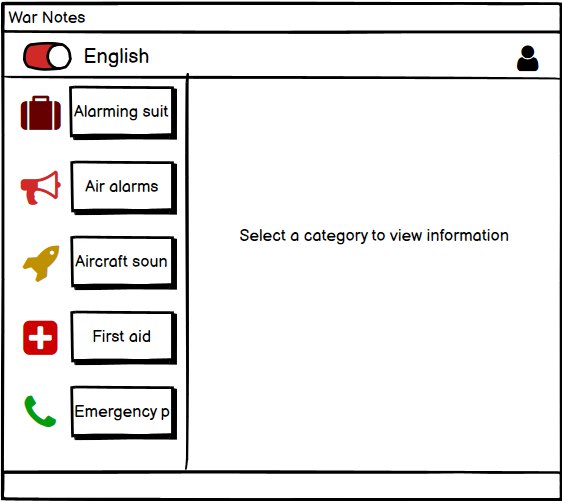
Проблема : вийти з акаунту

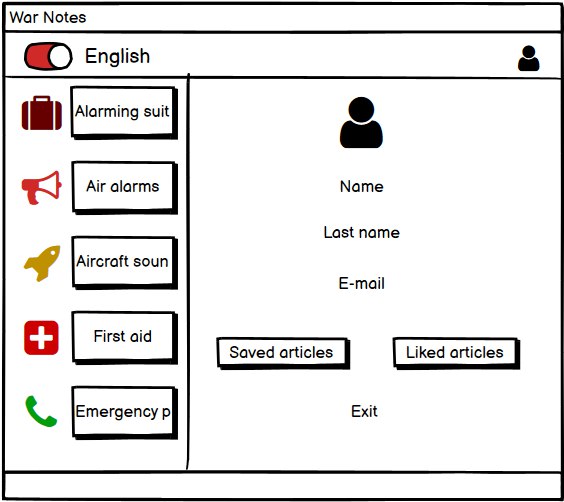
Відповідь : сесія для даного користувача закінчилась

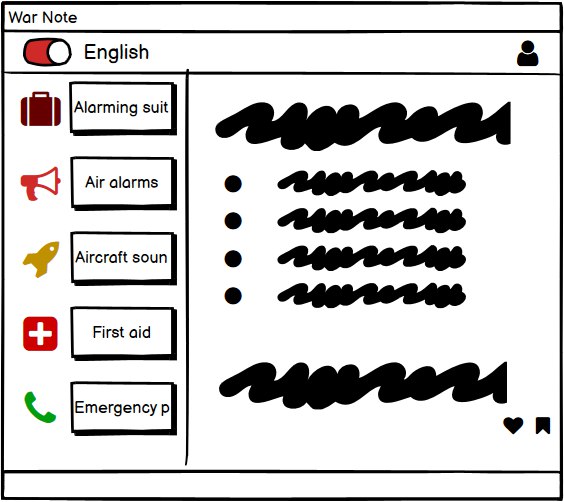
1. **Вимоги до інтерфейсу**
   1. **User Interface**

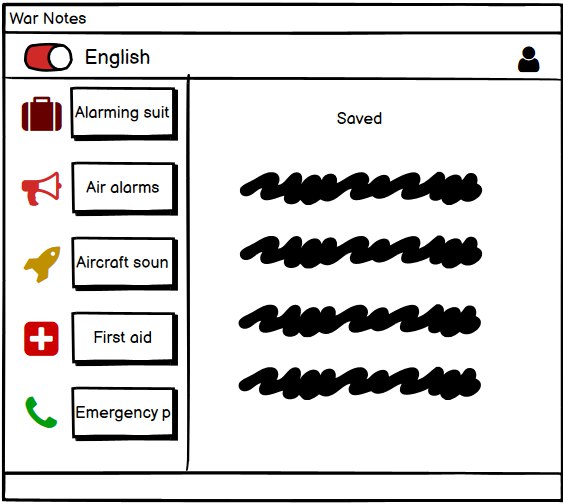
****

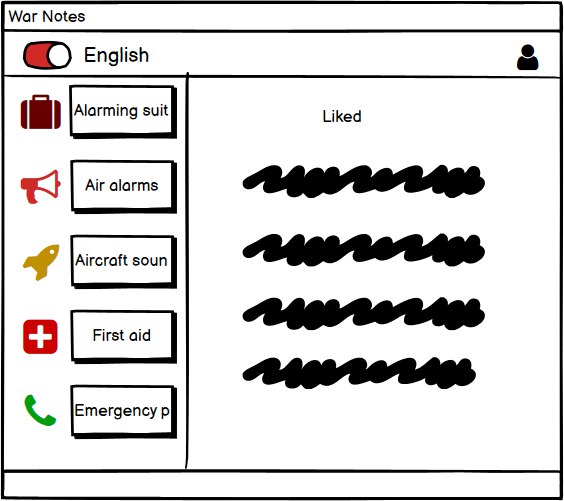
****

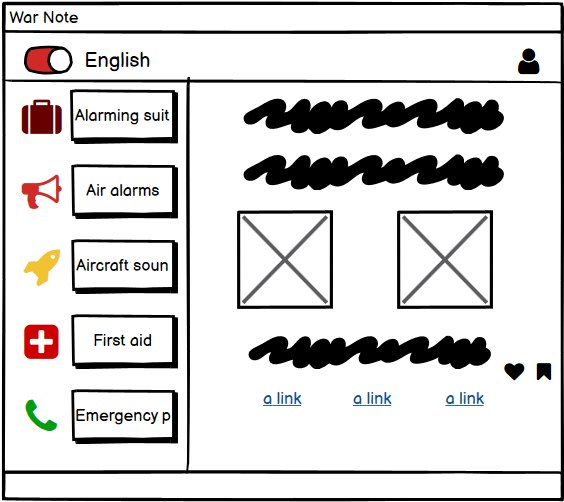
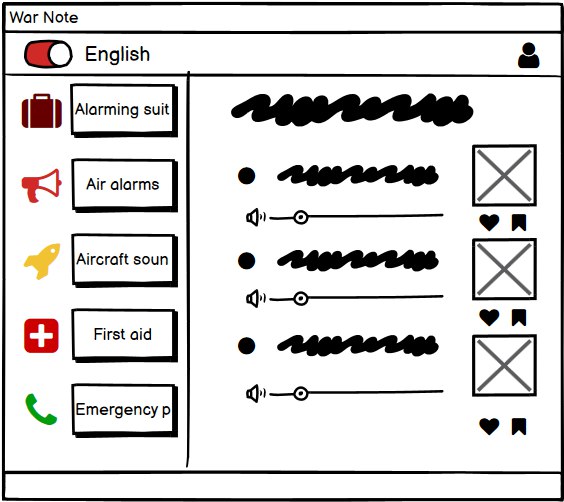
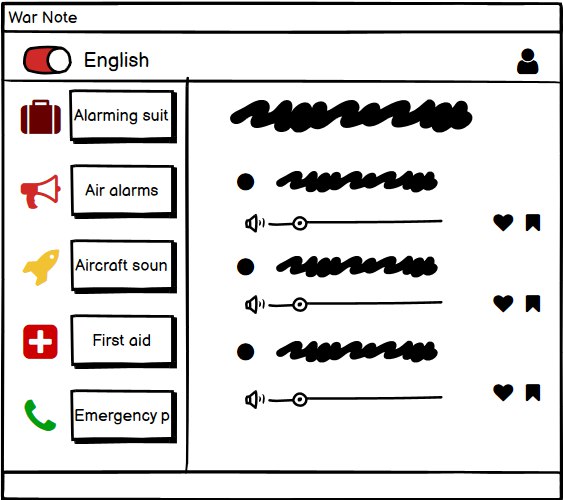
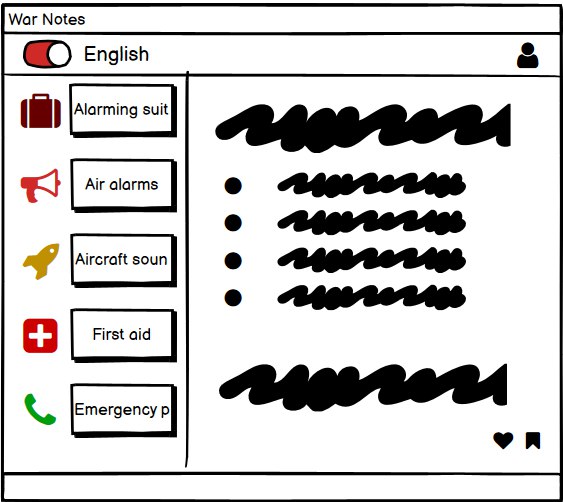
****

****

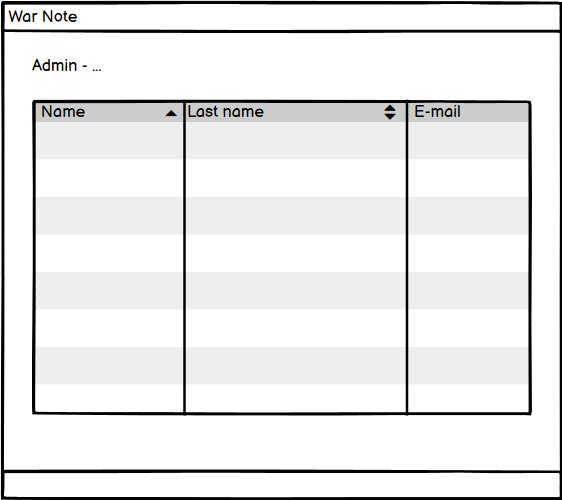
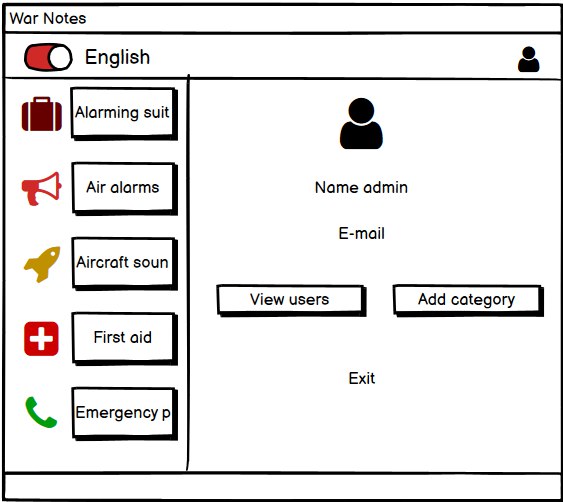


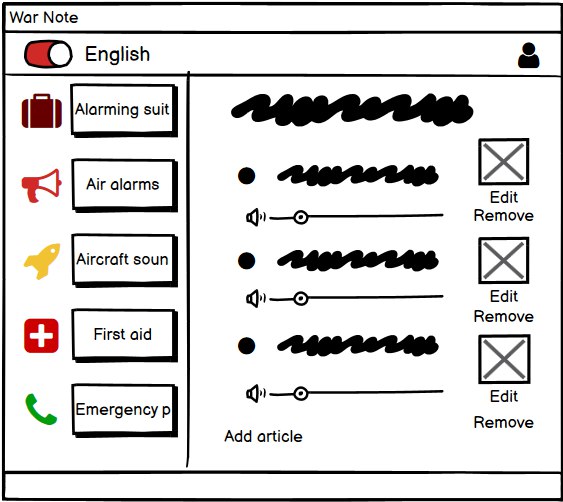
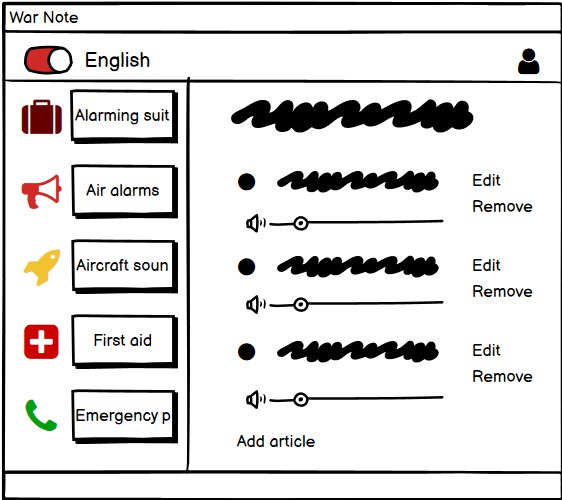
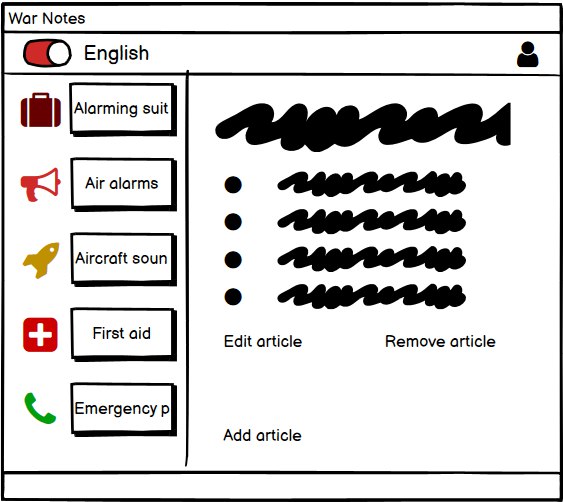
****

****

****

**Інтерфейс для адміністратора**

****

****

* 1. **Hardware Interfaces**

Комп’ютер з операційною системою Windows

* 1. **Software Interfaces**

|  |  |
| --- | --- |
| Операційна система | Для якісної роботи застосунку необхідна система Windows. |
| Framework | Обрано найбільш придатний для застосунку - .NET 6.0 |
| Database | PostgreSql |

1. **Нефункціональні вимоги**
   1. **Performance Requirements**

Уникати довгої затримки під час входу в систему.

Уникати затримки під час завантаження аудіо.

* 1. **Security Requirement**

Виключити можливість отримання доступу до даних користувача шляхом хешування його email та паролю.

* 1. **Software Quality Attributes**
* Актуальність

Система повинна містити лише актуальну інформацію

* Зручність

Система повинна задовільняти потреби користувачів

* Доступність

Система повинна надавати доступ до всіх ресурсів, що є в застосунку.

* Якісність

Система повинна надавати коректні дані. Адміністратори повинні підтримувати базу всіх статей, періодично її оновлювати.

* 1. **Критерії прийняття**

Подія: Вхід з правильним логіном та правильним паролем

Дано: Взято правильний логін та правильний пароль

Коли: Користувач не увійшов в систему

Результат: Користувача зареєстровано

Подія: Вхід з правильним логіном та неправильним паролем

Дано: Вхід з правильним логіном та неправильним паролем Коли: Користувач не увійшов в систему

Результат: Користувача не зареєстровано, повідомлено про неправильні дані

Подія: Вхід з неіснуючим логіном

Дано: Логін, якого немає в базі

Коли: Користувач не увійшов в систему

Результат: Користувача не зареєстровано, повідомлено про неправильні дані

Подія: Вихід з системи

Дано: Натиснуто кнопку вийти

Коли: Користувач в системі

Результат: Сесія для даного користувача закінчилася

Подія: Вибір категорії

Дано: Користувач в системі

Коли : Сторінку категорій відображено

Результат : Відображено сторінку з категоріями, обрану категорію підсвічено

Подія: Вподобати статтю

Дано: Статтю обрано

Коли: Користувач в системі

Результат: Кнопка <like> змінила колір на червоний

Подія: Зберегти статтю

Дано: Статтю обрано

Коли: Користувач в системі

Результат: Кнопка <save> змінила колір на синій

Подія: Перегляд вподобаних статей

Дано: Користувач в системі

Коли: Відкрито особистий кабінет

Результат: Список вподобаних статей відображено

Подія: Перегляд збережних статей

Дано: Користувач в системі

Коли: Відкрито особистий кабінет

Результат: Список збережених статей відображено

Подія: Вхід адміністратора в систему

Дано: Користувач увійшов з даними адміністратора

Коли: Користувач не в системі

Результат: Відображено головну сторінку адміністратора – список всіх користувачів

Подія: Адміністратор додає нову статтю

Дано: Адміністратор в системі

Коли: Натиснуто кнопку додати

Результат: Статтю додано

Подія: Адміністратор видаляє статтю

Дано: Адміністратор вибрав статтю

Коли: Натиснуто кнопку Видалити

Результат: Статтю видалено

Подія: Адміністратор редагує статтю

Дано: Адміністратор вибрав статтю

Коли: Натиснуто кнопку Редагувати

Результат: Статтю відредаговано

Подія: Адміністратор додає нову категорію

Дано: Адміністратор в системі

Коли: Натиснуто кнопку додати

Результат: Категорію додано

Подія: Адміністратор видаляє категорію

Дано: Адміністратор вибрав статтю

Коли: Натиснуто кнопку Видалити

Результат: Категорію видалено

Подія: Адміністратор редагує категорію

Дано: Адміністратор вибрав статтю

Коли: Натиснуто кнопку Редагувати

Результат: Категорію відредаговано