**Короткий опис проєкту**

Сервіс для зберігання текстових нотаток з можливістю надсилання цих нотаток іншим користувачам.

Кожна нотатка прив'язана до одного користувача. Один користувач може мати багато нотаток (0 або більше).

**Авторизация**

Адреса - /login

Якщо користувач відкриває кореневу адресу (/), і він авторизований у цьому браузері – його перекидає на список нотаток (/note/list).

Якщо користувач відкриває кореневу адресу, і він не авторизований - його перекидає на сторінку входу (/login).

<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362679673943&cot=14>

Імена користувачів зберігаються у базі даних. Допустиме ім'я користувача - будь-які символи латиниці та цифри. Довжина імені – від 5 до 50 символів включно.

Пароль користувача включає будь-які символи від 8 до 100 символів включно. У БД не зберігаємо паролі користувачів, лише хеши паролів.

**Реєстрація**

Адреса - /register

Якщо користувач не зареєстрований, то на сторінці логіна він може натиснути кнопку Зареєструвати, і потрапити на сторінку реєстрації.

<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362684191724&cot=14>

Після успішної реєстрації користувача перекидає на сторінку входу, де він може ввести свої дані та увійти до системи.

Якщо реєстрація невдала (дані користувача невалідні), він потрапляє на екран реєстрації з усіма порожніми полями та текстовою помилкою, що саме невалідно

<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362685696636&cot=14>

**Користувач за замовчуванням**

За замовчуванням при запуску програми повинен створюватись користувач з ім'ям admin та паролем super\_secret\_password.

**Список нотаток**

Адреса - /note/list

На цьому екрані бачимо всі нотатки користувача, оформлені таблицею.

<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362681258202&cot=14>

Текст "Мої нотатки - 3 шт" - це скільки у користувача є нотаток.

**Створення нотатки**

Адреса - /note/create

На цьому екрані ми створюємо нову нотатку. <https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362683621729&cot=14>

Поле "Ім'я нотатки" - <input type="text">

Поле "Вміст нотатки" - <textarea rows="20">

Вибір типу нотатки - це радіобаттони, за замовчуванням вибрано варіант "Приватне посилання"

За натисканням кнопки зберегти надсилається запит POST з усіма даними за адресою /note/create, що ввів користувач. Якщо нотатка пройшла валідацію, ми перекидаємо користувача на екран всіх нотаток (/note/list), де він бачить список своїх нотаток, включаючи щойно створену.

Якщо ж зберегти нотатку не вийшло (які поля невалідні), то нотатку не зберігаємо і перекидаємо користувача на екран помилки (<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362683874933&cot=14> ), де вказуємо що саме пішло не так.

З екрана помилки натискання кнопки “Повернутися до списку нотаток” користувача перекидає на список нотаток (/note/list)

**Редагування нотатки**

Адреса - /note/edit

Дуже схоже на екран створення нотатки, ключові відмінності:

* Замітка вже має ідентифікатор. Тому на сторінку редагування потрібно прокидати цей ідентифікатор, зберігати його в невидимому полі форми (<input type=”hidden” value=”{id}”>), і передавати той самий ідентифікатор разом з іншими полями, що редагує користувач
* Дані, введені користувачем, зберігаємо POST запитом за адресою /note/edit
* Усі поля вже заповнені даними редагованої нотатки - ім'я, вміст та тип доступу
* У заголовку сторінки текст не "Створення нотатки", а "Редагування нотатки"

Кнопка "Зберегти" поводиться так само, як і при створенні нотатки.

**Обмеження для сутності Примітка**

* **Ідентифікатор** - рядок, що генерується на підставі UUID.
* **Ім'я** – рядок, від 5 до 100 символів включно
* **Вміст** - рядок від 5 до 10 000 символів включно.
* **Тип доступу** - enum, два варіанти:

1. **private** - доступ за публічним посиланням вимкнено
2. **public** - доступ за публічним посиланням включений

У БД зберігаємо enum рядком, а не числом, як дефолтом це робить Hibernate.

**Публічні посилання на замітки**

Адреса - /note/share/{id}

Посилання має бути доступне за адресою типу host/note/share/{id}. {id} – це ідентифікатор нотатки.

<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362683008530&cot=14>

Наприклад, якщо ваша програма розгорнута за адресою <https://mynotes.com> , то публічне посилання на замітку з id=xxx буде виглядати так:

<https://mynotes.com/note/share/xxx>

Якщо користувач відкриває сторінку з неіснуючою або приватною нотаткою, він бачить відповідний екран.

<https://miro.com/app/board/o9J_l1t0r8Y=/?moveToWidget=3074457362683207447&cot=14>

**Запуск програми**

Програма запускається на порту 9999.

Має бути два профілю запуску (почитати можна - <https://www.baeldung.com/spring-profiles> )

Профіль за замовчуванням запускається з in-memory БД H2.

Профіль prod запускається з СУБД PostgreSQL. Параметри БД (шлях до БД, логін-пароль) беруться зі змінних оточення. Наприклад:

spring.datasource.username=${DB\_USERNAME}

Змінні оточення повинні бути описані в файлі readme.md.

Така структура, коли є профіль за замовчуванням з h2 БД in-memory, дозволяє запускати програму локально та тестувати її в браузері за адресою <http://localhost:9999>

У той же час профіль prod дозволить пізніше розгорнути програму на бойовому сервері.

**Верстка сторінки**

Використовуємо bootstrap, підключаємо його із CDN. Наприклад <https://www.bootstrapcdn.com/>

У Spring Boot використовуємо шаблонизатор Thymeleaf <https://www.thymeleaf.org/>

**Работа с БД**

Використовуємо СУБД PostgreSQL.

Використовуємо фреймворк для міграцій Flyway.

Генерацію сутностей Hibernate відключаємо, тобто в application.properties має бути рядок:

spring.jpa.hibernate.ddl-auto=validate

**Приклад**

<https://github.com/1nt3g3r/goit-e-dobavki>

У репозиторії є приклади роботи з більшістю функцій, що є вище.