**Temat projektu: Platforma do nauki za pomocą fiszek**

*Inspiracje: Anki, Quizlet*

Technologie

1. Spring – backend

2. React – frontend

3. Node.js – obsługa niektórych procesów, API

4. Docker – konteneryzacja aplikacji

5. Git – kontrola wersji i współpraca zespołowa

Podział Prac

* Frontend:
  + Kacper
  + Julka
* Backend:
  + Marcin
  + Maciek

Funkcjonalności Aplikacji

1. Logowanie i Rejestracja

* Logowanie za pomocą konta i hasła
* Rejestracja przez email
* Odzyskiwanie hasła
* Logowanie za pomocą OAuth

2. Tworzenie Fiszek

* Tworzenie folderów na fiszki
* Obsługa różnych trybów powtarzania materiału
* Tryby powtarzania fiszek:
  + Klasyczne fiszki
  + Wpisywanie odpowiedzi i sprawdzanie poprawności
* Import i eksport fiszek

3. Algorytm Powtarzania – Spaced Repetition

* Algorytm zaplanowany jako serwis, planujący pojawianie się fiszek zgodnie z metodą „spaced repetition”

4. Statystyki

* Śledzenie postępów użytkownika
  + Statystyki użytkownika:
  + Ilość przejrzanych fiszek w ciągu dnia
  + Ilość fiszek do przejrzenia
  + Procent zdanych fiszek

5. Ustawienia Użytkownika

* Opcje personalizacji:
  + Wybór języka aplikacji
  + Wybór czcionki
  + Tryb dark mode/light mode

6. Zarządzanie Fiszkami

* Zamiana przodu fiszki z tyłem
* Możliwość dostosowania tyłu fiszki (np. dodawanie zdjęcia)
* Przechowywanie danych w bazie danych

Komunikacja Frontend-Backend

1. Wymiana informacji

* Klasa Fiszek
  + Tworzenie, usuwanie i aktualizacja fiszek
* Klasa Folderu Fiszek
  + Przechowywanie i zarządzanie folderami bez konieczności ładowania całej zawartości na raz
* Klasa Statystyki
  + Forma przechowywania i prezentacji postępów użytkownika
* Klasa Ustawień Użytkownika
  + Dane do logowania (wysyłanie loginu i hasła, odbieranie tokenu sesji)
  + Zmiana statusu fiszki (informacje o dacie kolejnego powtórzenia)

2. Synchronizacja danych

* Strategia przechowywania danych zależy od zaimplementowanego algorytmu powtarzania:
  + SchedulerInterfaceData – uniwersalna klasa przechowująca dane do planowania powtórek
  + Aktualizacja statusu fiszki – metoda API wywoływana na osobnym wątku:
  + SchedulerInterfaceData.update(Enum.VERY\_GOOD)