Application FluentForge

# 1. Application concepts:

Application to simulate process of learning languages

## Description of goals and functions:

* *Giving opportunity to learn foreign languages with effective and no-standard way*
* *By usage of interactive exercises, quizzes, games, tests user won’t get bored and come back to application with more enthusiasm*
* *By giving opportunity of free learning more users will try it out*
* *Offering possibility to choose custom way of learning*

## Target group and its needs:

*Target group :*

* *People of any age only ones which desire to learn new things*
* *Mainly young people aged around 15 – 30*

*Needs:*

* *Application easy to use with many functionalities, variety, without monotony*
* *Possibility of a version for a computer and smartphone*

## Used technologies:

* *Java, Spring Boot, SQLite: backend*
* *JavaScript, React: frontend*
* *Python: Generating exercises and progress analisis*
* *C++: Optimization*

## Project functionalities:

* *Backend methods:*
* *SQL models:*
  1. *Aprrentice:*
  2. *Each user*

Main project need:

1. **Zdefiniuj Cele Projektu**
   * Określ główne cele aplikacji i funkcje, które ma oferować.
   * Zidentyfikuj grupę docelową i ich potrzeby.
2. **Stwórz Dokumentację Projektu**
   * **Dokumentacja Wymagań**: Szczegółowy opis funkcji aplikacji, oczekiwań i wymagań. Możesz użyć narzędzi takich jak [Notion](https://www.notion.so/) lub [Google Docs](https://docs.google.com/).
   * **Specyfikacja Techniczna**: Opis technologii, frameworków i narzędzi, które będą używane.
3. **Zaprojektuj Architektura Systemu**
   * **Diagram Architektury**: Stwórz diagram pokazujący, jak poszczególne komponenty będą ze sobą współpracować (np. frontend, backend, analiza danych).
   * **Diagramy Przepływu Danych**: Pokaż, jak dane będą przepływać przez system.
4. **Opracuj Plan Projektu**
   * **Harmonogram**: Opracuj harmonogram z kamieniami milowymi i terminami.
   * **Podział Zadań**: Określ, jakie zadania są do wykonania i kto będzie za nie odpowiedzialny (jeśli pracujesz z zespołem).

### Etap 2: Projektowanie

1. **Projekt Interfejsu Użytkownika (UI/UX)**
   * **Makiety i Prototypy**: Stwórz makiety (mockups) i prototypy UI, aby wizualizować, jak aplikacja będzie wyglądać i działać. Możesz używać narzędzi takich jak [Figma](https://www.figma.com/) lub [Adobe XD](https://www.adobe.com/products/xd.html).
   * **Analiza UX**: Przeanalizuj, jak użytkownicy będą korzystać z aplikacji i jak zapewnić intuicyjność interfejsu.
2. **Projekt Bazy Danych**
   * **Schemat Bazy Danych**: Stwórz diagram ERD (Entity-Relationship Diagram) pokazujący tabele bazy danych, ich kolumny oraz relacje między nimi.
   * **Planowanie Zapytania SQL**: Określ, jakie zapytania SQL będą potrzebne do realizacji funkcji aplikacji.
3. **Projekt API**
   * **Dokumentacja API**: Stwórz dokumentację opisującą endpointy API, formaty danych wejściowych i wyjściowych, oraz przykłady użycia. Możesz użyć narzędzi takich jak [Swagger](https://swagger.io/) lub [Postman](https://www.postman.com/).

### Etap 3: Implementacj