

LANGUAGE LEARNING APP

Overview

Language Learning Website is designed to make the learning easier and interactive for every user. It provides a simple and best user interface for user to learn the language effectively. Here every user starts from basic level and increase their knowledge level by level, so it makes user to learn faster and in better way.

The Concept is split into three levels Basic, Intermediate and Advanced where each session contains n number of sessions so user can learn the language in proper structural format. Quiz is provided at each of the session to make the learning interactive and also to assess the user's progress with the respective language.

After completing each level, test is provided based on the level which is used to analyse the user's progress in that level. You should pass certain percentage in each level to access the next level, this ensure that each user possess enough knowledge in the corresponding level so that it will help the user to understand the next level in a better way. This website ensures that each user should learn the respective language at the end of completing all levels.

After completing all levels there is a final test after completing it the certificate is provided to the respective user for completing the corresponding language.

Components used in the project

Frontend

- React.js
- Material UI

Backend

- Express.js
- Node.js

Database MySQL

- MongoDB

Hosting Platforms

- Frontend - Netlify, Vercel
- Backend - Heroku, Render

Frontend Components

The frontend is the user-facing part of the application. Here are the essential components:

Login/Signup

- **Signup Page:** Users provide basic details for profile creation.
- **Login Page:** Allows users to sign in and track their learning progress.

Home Page

- Welcome page introducing the platform
- Simple **navigation** to different sections like Levels, Progress, and Settings.

Displaying Selection Page

- Displaying a list of languages to learn
- User can select any language and start to learn

Level Selection Page

- Shows Different Levels : Basic, Intermediate and Advanced.
- Next level is unlocked based on the users progress.
- Progress bar displaying the users progress.

Session Page

- Lists all **sessions** within a selected level.
- Each session includes:
 - A **title** (e.g., "Common Words," "Simple Sentences").
 - A **short description** of the lesson.
 - A **completion indicator** showing session progress.
- The final test at end of each level after completing all sessions in that level

Test Page

- Each session have a test at the end
- A **final test** at the end of each level to advance to the next stage.
- A **final test** after completing all levels to verify proficiency.

Progress Tracking System

- **Progress bar** on the session and level pages.
- Displays **percentage of completion** within a level.

Certificate & Congratulations Page

- After passing the final test, users receive a **certificate of completion**.
- A **congratulatory message** to celebrate their achievement.

Backend Components

User Model

- **id** (unique identifier)
- **name**
- **email**
- **password** (for authentication)
- **selected_language** (language the user is learning)
- **current_level** (tracks the user's progress)
- **experience_points** (XP gained through sessions and tests)

Language Model

- **id** (unique identifier)
- **name** (e.g., English, Spanish, French)
- **description** (brief details about the language)

Level Model

- **id** (unique identifier)
- **language_id** (links to the selected language)
- **level_name** (e.g., Basic, Intermediate, Advanced)
- **required_xp** (XP needed to unlock this level)

Session Model

- **id** (unique identifier)
- **level_id** (links to the level it belongs to)
- **title** (e.g., "Common Words," "Basic Phrases")

- **content** (text, audio, or video materials for learning)
- **status** (e.g., "Locked," "Completed")

Test Model

- **id** (unique identifier)
- **session_id** (links to the session it belongs to)
- **questions** (list of quiz questions)
- **correct_answers** (stored for result comparison)
- **pass_status** (e.g., "Passed," "Failed")

User Progress Model

- **id** (unique identifier)
- **user_id** (links to the user)
- **completed_sessions** (list of finished sessions)
- **current_xp** (experience points accumulated)
- **certification_status** (e.g., "Pending," "Earned")

Certificate Model

- **id** (unique identifier)
- **user_id** (links to the user)
- **language_id** (links to the learned language)
- **completion_date**
- **certificate_url** (link to the downloadable certificate)

These backend components handle **user data, learning content, progress tracking, and certification**, ensuring a smooth and structured language learning experience.

Database Components

Use **MySQL** or **MongoDB** for simplicity.

- **Users:** id (unique identifier), name, email, password, selected_language, current_level, experience_points.
- **Languages:** id (unique identifier), name, description.
- **Levels:** id (unique identifier), language_id (links to Languages), level_name, required_xp.
- **Sessions:** id (unique identifier), level_id (links to Levels), title, content, status.
- **Tests:** id (unique identifier), session_id (links to Sessions), questions, correct_answers, pass_status.
- **User Progress:** id (unique identifier), user_id (links to Users), completed_sessions, current_xp, certification_status.
- **Certificates:** id (unique identifier), user_id (links to Users), language_id (links to Languages), completion_date, certificate_url.
- **Relationships:** A language has many levels, a level has many sessions, a session has one test, a user can complete many sessions and earn experience points, and a user can receive multiple certificates after completing different languages.

Frontend Hosting

Platform: Netlify

- Free tier with essential features.
- Simple deployment for static sites like HTML, CSS, and JavaScript projects.
- Seamless integration with GitHub for automatic updates.
- Handles builds and hosting efficiently.

Backend Hosting

Platform: Render or Heroku

- Free tier available for small projects.
- Easy deployment with minimal setup.
- No need to manage servers or infrastructure.
- Beginner-friendly with simple integration.

Process Workflow

