Course and Section: PRJ666 NZA, Team 3

Student Name: Di Liu, Omkar Bharat Patel, Shivkuma Hiteshkumar Patel, Vanshika Sharma

Professor Name: Yasser Elmankabady

Submission Date: Dec 09, 2024

DevFusion-iOS-Android Technical Report

Table of Contents

[1. Introduction 1](#_Toc184655819)

[1.1 Project Overview 1](#_Toc184655820)

[1.2 Purpose 1](#_Toc184655821)

[2. System Architecture 1](#_Toc184655822)

[2.1 Overview of Architecture 1](#_Toc184655823)

[2.2 Key Components 1](#_Toc184655824)

[2.3 Technologies Used 2](#_Toc184655825)

[3. Functional Specifications 2](#_Toc184655826)

[3.1 User Registration and Login 2](#_Toc184655827)

[3.2 Learning Modules 2](#_Toc184655828)

[3.3 Test and Certification 2](#_Toc184655829)

[3.4 User Reviews 3](#_Toc184655830)

[4. Non-Functional Requirements 3](#_Toc184655831)

[4.1 Performance 3](#_Toc184655832)

[4.2 Security 3](#_Toc184655833)

[4.3 Scalability 3](#_Toc184655834)

[4.4 Usability 3](#_Toc184655835)

[5. Database Design 4](#_Toc184655836)

[5.1 Schema Overview 4](#_Toc184655837)

[5.2 Relationships 4](#_Toc184655838)

[6. Testing and Quality Assurance 4](#_Toc184655839)

[6.1 User Acceptance Testing (UAT) 4](#_Toc184655840)

[7. Deployment Strategy 4](#_Toc184655841)

[7.1 Deployment Phases 4](#_Toc184655842)

[8. Conclusion 4](#_Toc184655843)

# 1. Introduction

### 1.1 Project Overview

This project aims to develop a user-friendly learning application that provides educational content, tests, final test, and certifications for its users. The application is designed to create a seamless learning environment where users can engage with the material, assess their understanding, and receive certifications upon successfully completing tests. Additionally, the app allows users to post reviews, offering valuable feedback that contributes to continuous improvement and ensures the app meets user expectations.

### 1.2 Purpose

The purpose of the learning application is to offer a structured and engaging platform for users, providing:

* A comprehensive learning experience with various educational modules.
* Interactive tests and final test to evaluate understanding and knowledge.
* Instant feedback and certification upon passing the tests.
* A review system that lets users share their experiences and opinions on the modules and the application.

# 2. System Architecture

### 2.1 Overview of Architecture

The application adopts a client-server architecture, where the client interacts with the server to request and retrieve data. The frontend (client) is responsible for the user interface and user experience, while the backend (server) processes business logic, user requests, and database operations.

### 2.2 Key Components

* **Frontend**: The user-facing interface built with JavaScript and EJS templates, designed to ensure a clean, responsive, and intuitive experience for users. It handles the display of content, tests, certificates, and reviews, ensuring easy navigation and accessibility.
* **Backend**: Built with Node.js and Express.js, the backend processes requests, manages user authentication, and handles data storage/retrieval operations. It acts as the intermediary between the frontend and database.
* **Database**: MongoDB serves as the NoSQL database to store various types of data such as user profiles, educational modules, tests, final test, test results, certifications, and reviews. MongoDB's flexible schema is well-suited for the evolving structure of the application.
* **Authentication**: Secure authentication is implemented using OAuth 2.0 and JWT (JSON Web Tokens), which ensures secure and reliable user login and session management.

### 2.3 Technologies Used

* **Frontend**: JavaScript, EJS framework for dynamic HTML rendering.
* **Backend**: Node.js for the server runtime, Express.js for handling routes and middleware.
* **Database**: MongoDB for data storage, designed to handle both structured and unstructured data.
* **Authentication**: OAuth 2.0 for user authentication, JWT for managing user sessions.
* **Deployment**: Render, a platform used for the application’s deployment and hosting, ensuring scalability and high availability.

# 3. Functional Specifications

### 3.1 User Registration and Login

* Users can register with their email address and create a password for authentication.
* The login system uses JWT for secure, token-based authentication, allowing users to stay logged in across sessions.
* After logging in, users can access their profile and the learning modules available within the app.

### 3.2 Learning Modules

* The application features various educational modules, including text, images to cater to different learning styles.
* Modules are logically structured to allow users to progress through content in a meaningful order.
* Upon successfully completing the tests and final test, users take to assess their knowledge and understanding of the material.

### 3.3 Test and Certification

* After completing the learning content, users can take a test to evaluate their understanding. These tests consist single selection questions.
* The test is automatically scored, and users receive immediate feedback on their performance.
* If users pass the test, they are awarded a certificate with their name.

### 3.4 User Reviews

* Users can post reviews and feedback about the learning modules or the overall application experience.
* Reviews are rated by other users, which helps future users make informed decisions about which modules to take.
* The review system also helps administrators gather insights for improving the app and modules.

# 4. Non-Functional Requirements

### 4.1 Performance

* The application must load content within 3 seconds to ensure a smooth user experience.
* Data processing tasks such as test scoring and certification generation must be completed within 5 seconds to ensure real-time interaction.

### 4.2 Security

* User data, including personal information, test scores, and reviews, must be encrypted both in transit and at rest.
* User passwords are securely hashed before being stored in the database, and OAuth 2.0 along with JWT are used to authenticate and manage sessions.

### 4.3 Scalability

* The system should handle at least 1,000 concurrent users without a noticeable drop in performance.
* Auto-scaling features should be implemented to accommodate increased demand during peak periods.

### 4.4 Usability

* The application must offer a responsive and intuitive user interface, ensuring a seamless experience across different devices such as desktop and mobile.
* Accessibility features should be incorporated, making the app usable by people with disabilities (e.g., screen reader compatibility, keyboard navigation).

# 5. Database Design

### 5.1 Schema Overview

The MongoDB database is structured into several collections, each with specific data relevant to the app's functionality:

* **users**: Stores user profile data, including authentication details.
* **contents**: Contains content for each educational module, including text, images.
* **testsets**: Stores weekly test and final test.

### 5.2 Relationships

* A **User** can have multiple **testsets**.
* A **testset** is linked to a specific **Test**.
* A **User** can post multiple **Reviews**, each associated with a particular **Module**.

# 6. Testing and Quality Assurance

### 6.1 User Acceptance Testing (UAT)

* A group of real users participates in UAT to assess the usability and functionality of the app in real-world conditions.
* Feedback from UAT is gathered and used to improve the user experience and address any identified issues before production deployment.

# 7. Deployment Strategy

### 7.1 Deployment Phases

1. **Development Environment**: The application is initially developed and tested in a local environment to ensure the features function as expected.
2. **Staging Environment**: A staging server is set up for further testing with a broader user group, allowing for feedback and fine-tuning before the live release.
3. **Production Deployment**: After successful testing and validation, the app is deployed to a production server. Ongoing updates, bug fixes, and new feature rollouts are performed periodically.

# 8. Conclusion

This learning application successfully meets its objectives by offering a comprehensive and engaging platform for users to learn, assess their knowledge, receive certifications, and share feedback. The continuous incorporation of user reviews and feedback ensures the platform evolves to meet users' needs, improving the learning experience and expanding the app's capabilities over time.