

**ADAMA SCIENCE AND TECHNOLOGY UNVERSITY**

**SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING**

**Department of Software Engineering Course Title: Mobile app design and development**

**Project Title: My Tutor (Mobile application)**

**Members Name Id No Group**

|  |  |  |
| --- | --- | --- |
| 1. Bekan Deresa | UGR/25307/14 | 6 |
| 2. Muluken Ageri | UGR/25993/14 | 6 |
| 3. Meriat Akalu | UGR/25909/14 | 6 |
| 4. Habtamu Baye | UGR/25825/14 | 5 |
| 5. Yeabsira Tesfaye | UGR/25748/14 | 6 |
| 6. Yosef Tesfaye | UGR/25366/14 | 5 |

Submitted To: Mr. Arefat.

Submission Date: jun 25, 2024

**Table of content**

[1. Project Overview 1](#_Toc28120)

[A. Purpose and Scope: 1](#_Toc14201)

[B. Problem Domain: 1](#_Toc32041)

[C. Target Audience: 1](#_Toc20562)

[2.Requirements and Functionalities 1](#_Toc1628)

[3. Design Concepts 2](#_Toc16672)

[4. Development Approach 3](#_Toc30682)

[5. Technological Stack 3](#_Toc25997)

[6. Implementation Details 4](#_Toc20993)

[Key Features: 4](#_Toc27246)

[7. Testing and Quality Assurance 4](#_Toc19579)

[Testing Strategies: 4](#_Toc5705)

[Quality Metrics: 5](#_Toc21297)

[Beta Testing and User Feedback: 5](#_Toc20212)

[QA Team Collaboration: 5](#_Toc32746)

[Development,Security and Operations Practices: 5](#_Toc3387)

[Effectiveness: 5](#_Toc7643)

[8. Future Enhancements 5](#_Toc14005)

[Proposed Enhancements: 5](#_Toc28332)

[Alignment with Objectives and User Needs: 6](#_Toc22311)

# 1. Project Overview

## A. Purpose and Scope:

My Tutor is a mobile application designed to facilitate the process of finding home tutors easily and cost-effectively. It provides a platform where students or parents can connect with qualified tutors without unnecessary expenses or time-wasting activities. Tutors can also use the app to find clients and manage their tutoring services efficiently.

## B. Problem Domain:

The main issue addressed by My Tutor is the difficulty in finding reliable home tutors and the inefficiencies of traditional tutor-finding methods, such as through agencies or personal networks.

## C. Target Audience:

- Parents and students seeking home tutoring services.

- Tutors looking to offer their services and expand their client base.

# 2.Requirements and Functionalities

* **User Registration and Authentication**
* Users can create account by providing their username, email and password
* After registering, Users can login to the system by providing username/email and password
* **Profile Update and customization**
* Users can update their profile by changing their profile picture, bio, and location
* **Creating a Tutorial and Posting**
* Users can create a tutorial by providing information like grades range, subjects covered, monthly price per student and working location.
* After creating Tutorials, users can make a post of their tutorials my providing additional information like Title, description and Banner Picture (optional).
* **Seeing Tutorial Posts**
* Users first see a list of highly rated available tutors on the front page of the app.
* Users can see the detail of Tutorial posts by clicking on them
* **Searching and Filtering of Tutors and Tutorial Posts**
* Users can search for Tutors and Tutorial posts by the Tutor’s name, and Tutorial post’s title and description
* Users can filter tutorial posts by grade range, working location, and subjects covered
* **Rating and profile view**
* Users can see other users (tutors’) profile page by clicking on their username or profile picture
* Users can rate other users (tutors) after going to the tutors’ profile page

# 3. Design Concepts

* + - **UI Design:**
* Clean, intuitive interface with a consistent color scheme and branding.
* Clear visual hierarchy to guide users through tasks.
  + - **UX Considerations:**
* Simple on boarding process.
* Smooth navigation with minimal steps for key actions.
* Feedback mechanisms to inform users of successful actions or errors.
  + - **Navigation Flow:**
* Home screen with top-rated tutors and search functionality.
* Easy access to profile settings, tutorial creation, and tutorial posts from the main menu.
* Seamless transition between list views and detailed views.
  + - **Visual Elements:**
* High-quality images for profiles and banners.
* Consistent use of icons and buttons.
* Responsive design for different devices.

# 4. Development Approach

* **Methodology:**
* Agile Methodology: Chosen for its flexibility, iterative progress, and ability to incorporate user feedback.
* **Challenges and Solutions:**
* Ensuring data security: Implemented robust encryption and secure authentication protocols.
* Designing an intuitive UI: Conducted user testing and iterative design refinements.
* Balancing performance with features: Optimized code and leveraged efficient technologies.
* Being new for the programming language also affected on time consumption.

# 5. Technological Stack

* **Front end:**
* Framework: React Native for cross-platform development.
* **Back end:**
* Framework: Node.js with Express.
* Database: MongoDB.
* **Authentication:**
* Tool: Fire-base Authentication.
* **Cloud Services:**
* Provider: AWS for hosting and cloud services.
* **Additional Tools:**
* Version Control: Git.
* CI/CD: Jenkins.

# 6. Implementation Details

## Key Features:

1. **User Registration and Login:**

- Registration page with fields for username, email, and password.

- Login page allowing login via username/email and password.

1. **Profile Management**:

- Profile page for updating profile picture, bio, and location.

1. **Creating and Posting Tutorials:**

- Tutorial creation form with fields for grades, subjects, pricing, and location.

- Post creation with title, description, and optional banner picture.

1. **Tutorial Listings and Details:**

- Homepage displaying top-rated tutors.

- Detailed view of tutorial posts with all relevant information.

1. **Search and Filter:**

- Search bar for finding tutors and posts.

- Filters for grade range, location, and subjects.

1. **Rating System:**

- Rating mechanism on tutor profile pages.

# 7. Testing and Quality Assurance

## Testing Strategies:

* **Unit Testing**: Testing individual components or modules of the app to verify their correct functionality.
* **Integration Testing:** Verifying how different components work together and that the apps overall functionality is as expected.
* **UI/UX Testing:** Evaluating the apps user interface, user experience, and accessibility.
* **Functional Testing:** Validating that the app meets the specified requirements and business goals.
* **Performance Testing:** Assessing the apps behavior under different load conditions, network conditions, and device configurations.

## Quality Metrics:

* + Defining and tracking key quality metrics, such as code coverage, bug density, user retention, and crash rates.
  + Continuously monitoring these metrics to identify and address any quality issues.

## Beta Testing and User Feedback:

* + Conducting beta testing with a selected group of users (group members) to gather feedback and identify real-world issues.
  + Implementing a system to collect, analyze, and respond to user feedback and bug reports.

## QA Team Collaboration:

* + Involving the QA team throughout the development life-cycle, from requirements gathering to post-launch support.
  + Encouraging close collaboration between the QA team, developers, and product managers to address quality concerns.

## Development,Security and Operations Practices:

* + Integrating security testing into the QA process to identify and mitigate potential vulnerabilities.
  + Adopting Development,Security and Operations practices to embed security throughout the development and deployment pipeline.
  + Regularly reviewing and optimizing the QA processes and practices to adapt to changing requirements and technologies.
  + Incorporating lessons learned from previous releases to improve the overall quality assurance approach.

## Effectiveness:

- Early identification of bugs and issues through continuous testing.

- Improved user satisfaction by incorporating feedback from UAT.

- Ensured the apps reliability and performance with thorough performance testing.

# 8. Future Enhancements

## Proposed Enhancements:

1. In-app Messaging:

* Allow tutors and clients to communicate directly within the app.
* Facilitates easy scheduling and clarifications.

1. Payment Integration:

* Enable secure payment transactions within the app.
* Simplifies the payment process for clients and tutors.

1. Advanced Matching Algorithm:

* Implement an AI-based matching algorithm to suggest the best tutors based on user preferences and past ratings.

1. Expanded Rating System:

* Allow ratings on multiple criteria such as punctuality, teaching effectiveness, and communication skills.

1. Multilingual Support:

* Offer the app in multiple languages to cater to a broader audience.

## Alignment with Objectives and User Needs:

* Enhancements like in-app messaging and payment integration improve convenience and user experience.
* Advanced matching and expanded rating systems enhance the apps effectiveness in connecting users with suitable tutors.
* Multilingual support broadens the apps reach and accessibility.