

Github Link: https://github.com/HaydenDuong/SIT305-Mobile_Application_Development/tree/main/Task%206.1D

Video Demo: <https://deakin.au.panopto.com/Panopto/Pages/Viewer.aspx?id=408cedef-2cce-463e-8d30-b2e601003625>

Personalized Learning Experiences App

Technical Report

Introduction

The Personalized Learning Experiences App is designed to provide users with a tailored educational journey, leveraging modern Android development practices and advanced payment integration. This report outlines the app's features, architectural decisions, and how it aligns with current best practices. It also explores how Large Language Models (LLMs) can further enhance the app's capabilities.

1. App Features and Enhancements

1.1 Core Functionality

The app offers a comprehensive suite of features to support personalized learning:

- **Quiz and Assessment Engine:** Users can take quizzes tailored to their learning needs, with real-time feedback and progress tracking.
- **Account Management:** Secure user authentication and profile management.
- **Payment Integration:** Users can upgrade their accounts to access premium content using Google Pay, implemented with the latest APIs for security and user experience.

1.2 New Features Added

In response to the latest requirements, the following screens and features were implemented:

a) Profile Screen (see Picture 1)

- Serves as the main hub after login.
- Displays user stats (e.g., quizzes taken, score, streak).
- Includes a QR code for sharing the app/profile.

- Provides an "Upgrade" button for account tier purchasing.
- Navigation to History and other features starts here.

b) History Screen (see Picture 2)

- Displays a chronological record of user quiz attempts and learning milestones.
- Utilizes Room database for persistent local storage.
- Employs RecyclerView for efficient, dynamic list rendering.

c) Sharing Screen (see Picture 3)

- Enables users to share achievements and progress via social platforms.
- Integrates Android's native sharing intents for seamless cross-app communication.
- Ensures user privacy by allowing granular control over shared data.

d) Purchasing Screen (see Picture 4)

- Provides a user-friendly interface for purchasing premium content.
- Integrates Google Pay using ActivityResultLauncher, replacing deprecated onActivityResult.
- Implements robust error handling and user feedback mechanisms.

2. Modern Android Development Practices

2.1 Architecture

- **MVVM Pattern:** The app is structured using the Model-View-ViewModel architecture, promoting separation of concerns and testability.
- **Jetpack Components:** Utilizes LiveData, ViewModel, and Navigation components for lifecycle-aware, modular development.
- **Room Database:** Ensures efficient, type-safe local data storage and retrieval.

2.2 UI/UX

- **Material Design:** Adheres to Google's Material Design guidelines for a consistent and intuitive user experience.
- **Responsive Layouts:** Supports multiple screen sizes and orientations.
- **Locale-Aware Formatting:** All user-facing strings and currency values are formatted according to the device's locale.

2.3 Code Quality and Security

- **Null Safety:** Comprehensive null checks to prevent runtime exceptions.

- **Exception Handling:** All critical operations are wrapped in try-catch blocks with user-friendly error messages.
- **Secure Payment Flow:** Sensitive operations, such as payments, are handled using secure, up-to-date APIs.

3. Leveraging LLMs for Further Improvement

3.1 Current Opportunities

a) Personalized Content Generation

- LLMs can generate quiz questions and explanations tailored to individual learning gaps.
- Dynamic feedback and hints can be provided based on user responses.

b) Adaptive Learning Paths

- Analyze user performance data to recommend optimal learning sequences.
- Adjust content difficulty in real-time using LLM-driven insights.

c) Natural Language Interaction

- Implement chat-based support for user queries.
- Enable conversational interfaces for navigating content and receiving feedback.

3.2 Implementation Considerations

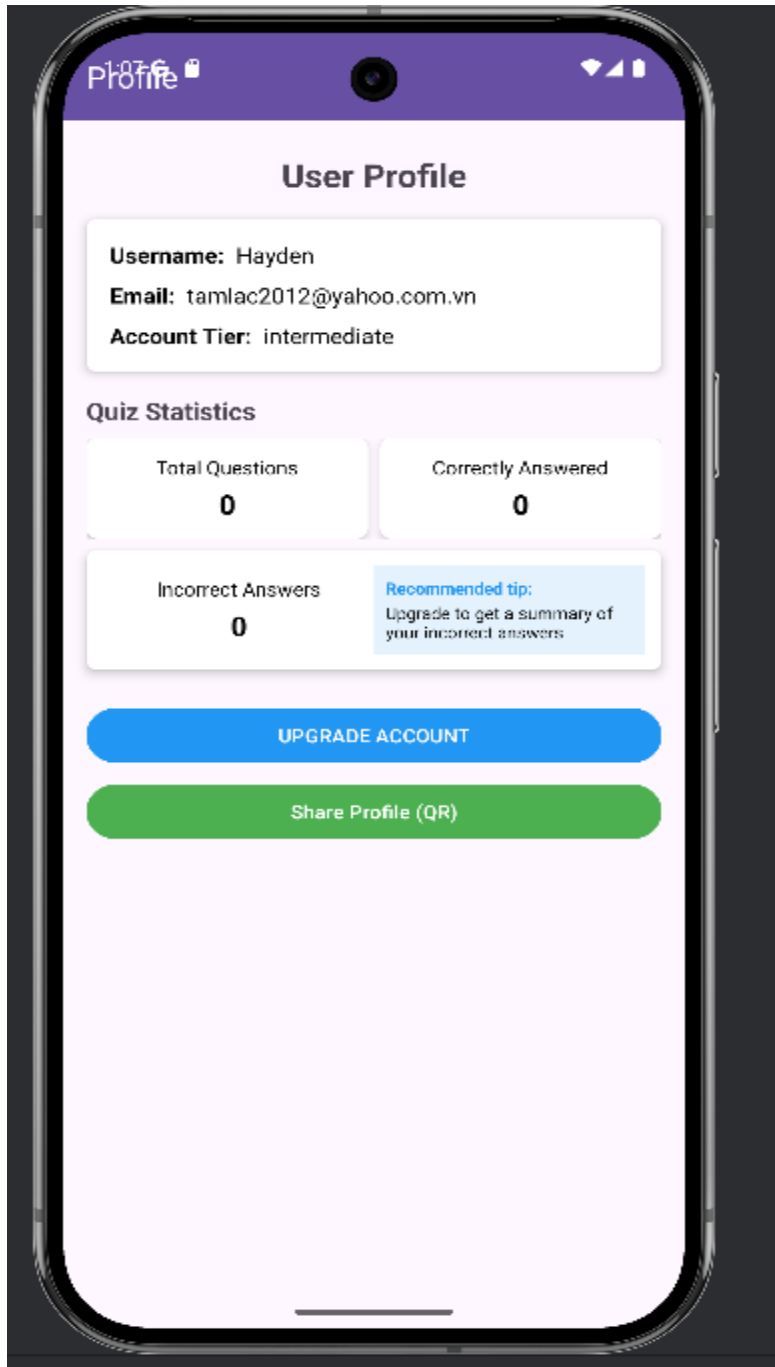
- **API Integration:** LLMs can be accessed via secure cloud APIs, with prompt engineering to ensure relevant, context-aware responses.
- **Privacy:** User data should be anonymized before being sent to external LLM services.
- **Caching:** Frequently used LLM responses can be cached locally to improve performance and support offline use.

4. Conclusion

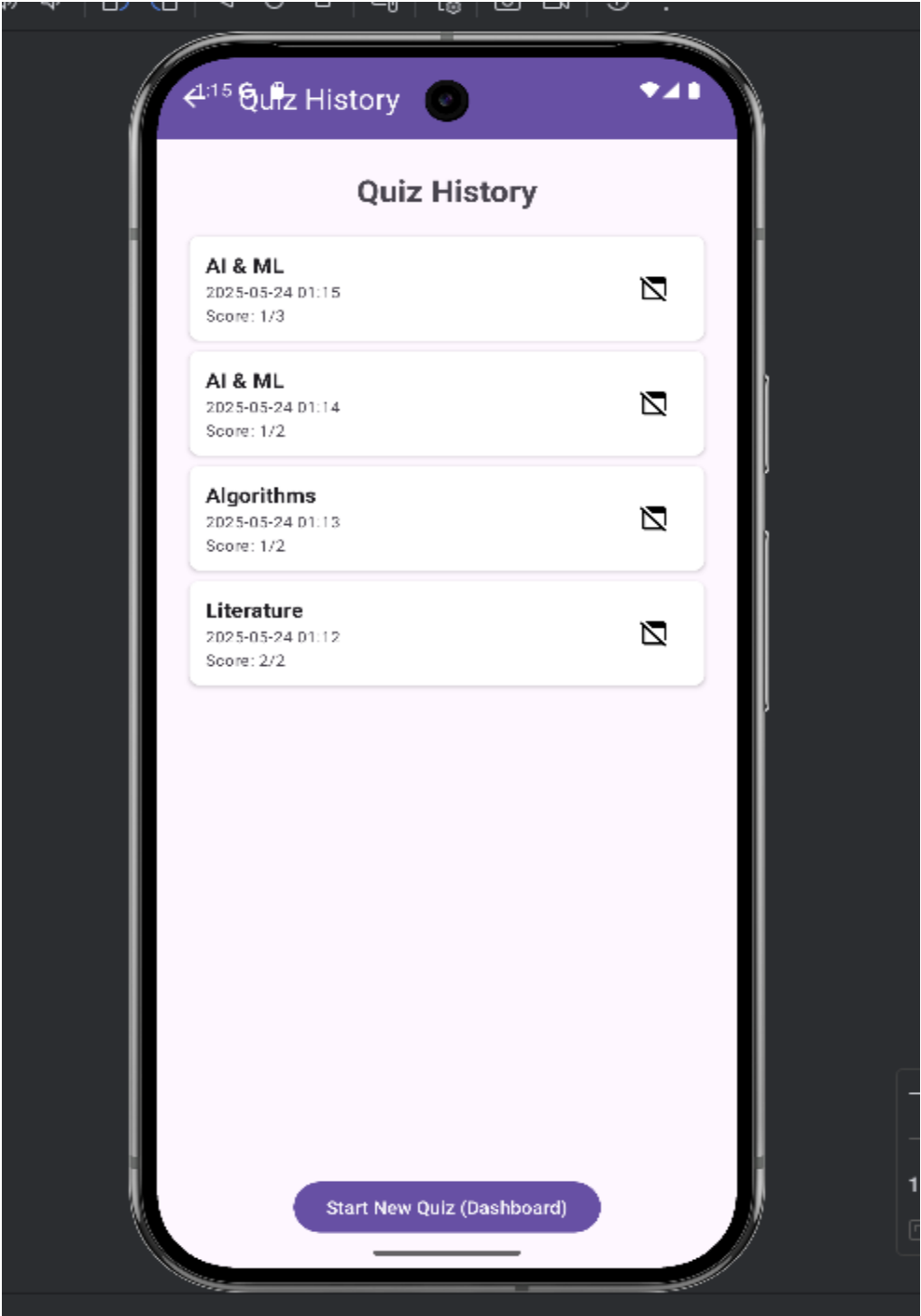
The Personalized Learning Experiences App exemplifies modern Android development, with a robust architecture, seamless payment integration, and a user-centric design. The addition of History, Sharing, and Purchasing features enhances the app's value proposition. Looking forward, integrating LLMs offers significant potential for deeper personalization, adaptive learning, and intelligent user support, positioning the app at the forefront of educational technology.

Appendices

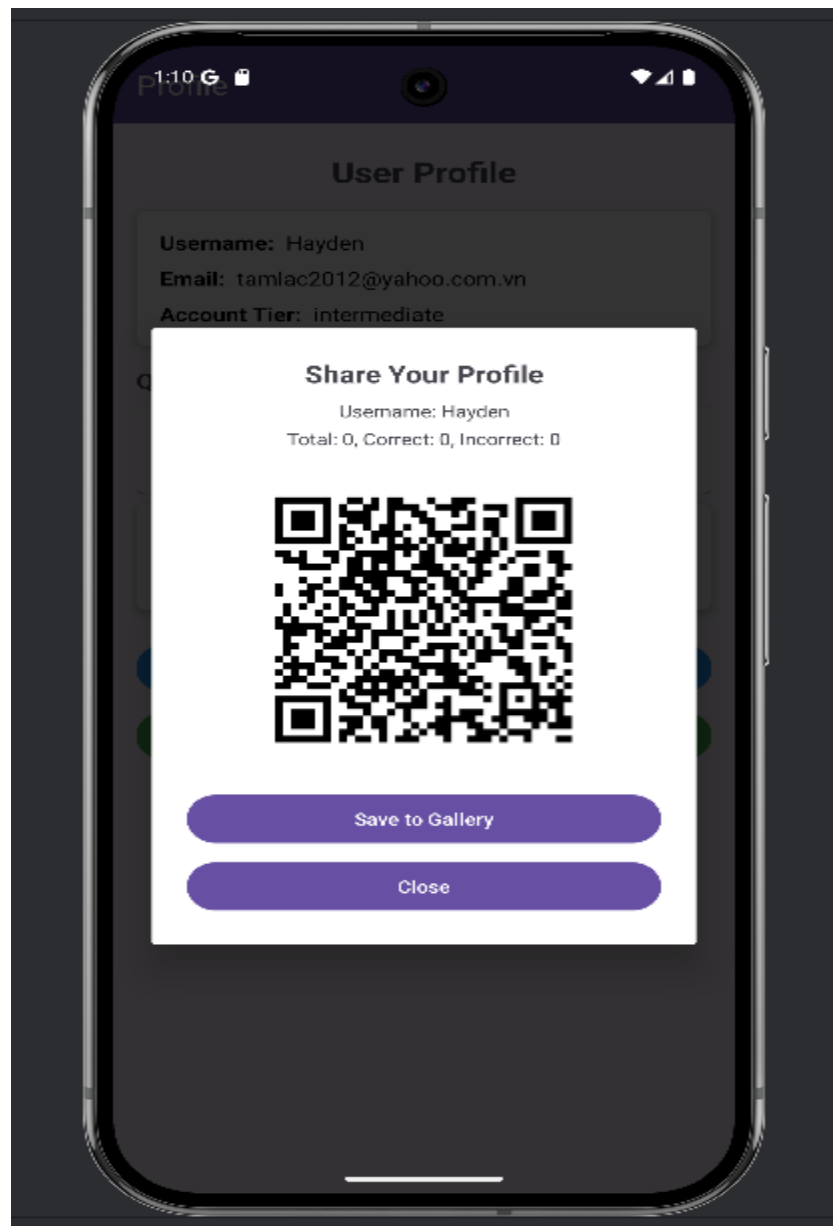
Picture 1



Picture 2



Picture 3



Picture 4

