**1. Introduction**

**1.1 Purpose**

This document outlines the architecture of StudyBuddy, our social accountability app that allows users to share study session photos and track when their friends are studying. The architecture is designed for flexibility, scalability, and ease of maintenance

**1.2 Scope**

StudyBuddy provides users with a platform where they can:

* Post study session updates with images.
* View friends' study activities.
* Stay motivated through social accountability and streak tracking.

The app follows a client-server architecture to ensure modularity, uptime, and ease of development.

**1.3 Stakeholders**

* **Ethan Morin** - Project Manager
* **Noah Rushing** - Designer
* **Alex Ayer** - Developer
* **Sany Dagher** - Developer
* **Denis Simost** - Developer
* **Xander Drolet** - Developer

**2. Architecture Design**

StudyBuddy is built using a layered architecture that separates concerns wherever possible, minimizing requests handled by the server to reduce overhead and ensure maximal functionality during times of high load.

**3. Design Description**

The StudyBuddy app consists of three primary layers:

**3.1 Top Layer (User Interface Layer)**

* Built using Expo (React Native) for cross-platform support.
* Provides the UI for posting study session updates, viewing friends' activities, and managing user profiles.
* Captures user input, including images and text, and sends requests to the backend for processing.

**3.2 Middle Layer (Application Logic Layer)**

* Manages authentication, session management, and business logic.
* Uses Firebase Authentication for secure login and user identity management.
* Processes study session posts, streak tracking, and notifications
* Interfaces with the backend Go service for image compression before storing them on Firebase

**3.3 Bottom Layer (Data & Storage Layer)**

* Firetore will store user profiles, study sessions, and friend connections.
* Firebase Storage with Google Cloud will store user-uploaded study session images.
* A Go backend service will handle image compression before storage to optimize load times and bandwidth usage.

**4. Conclusion**

This architecture ensures that StudyBuddy remains a maintainable, scalable, and reliable application for all of its users. It also ensures that it is simple to be iterated upon as development furthers, and allows for the flexible environment demanded by a project of this type.