

StudyPal - A Group Study Application

Developed by: Alexander Asiamah Boakye

Index Number: 9401219

CHAPTER 1: INTRODUCTION

In an era where everything is getting virtual and moving straight to our screens, meeting in person for group projects can be hectic and stressful. Also, being able to collaborate with students from other campuses and sharing resources is a struggle considering they aren't in close proximity to you.

Proposed Solution

StudyPal is a web-based mobile application which seeks to help students collaborate and get information in three ways. They are:

- A public forum where students can ask questions on topics and get answers from other students.
- Group calls and group chats to help with collaborating on group projects.
- Share resources and download from different schools and courses.

CHAPTER 2: LITERATURE REVIEW

Core Technology

Firebase:

Firebase is a backend service provided by Google which takes away the stress of creating and implementing a backend service for authentication, storage of media and creating of a database. It also provides functionality to send notifications to users of your application or services.

Review of some closely related works

- **Zoom:**

Zoom is a call conferencing application which provides a platform for a large audience to join a call either by video or audio. Zoom has a 40 minute time limit for free users, which can sometimes be woefully inadequate when using it for a longer period of time. Also, you cannot share resources over Zoom. You'd have to share a required resource either through cloud or a group chat in another application.

- **Microsoft Teams:**

Microsoft Teams is a collaboration applications where small to large organisations can have meetings and share information. Microsoft Teams is part of the Microsoft 365 suite. Hence, you'll have to pay a subscription to access it.

CHAPTER 3: METHODOLOGY

This application was developed using the waterfall development method. It was adopted to ensure your project's success even if there are unanticipated changes in the course of making the application. The linear nature of the system makes the project efficient to manage because it is easy to know where the project is at any given time.

The project was done in 4 parts. They are:

- **Requirement Gathering**

I gathered information about the project and how to implement the objectives of the project. I asked myself questions about the functions the application would perform and what technologies would be suitable to implement it.

- **System Design**

Using the information I gathered in the previous stage, I established the properties of the app such as the programming language to be used and the hardware requirements.

- **Implementation**

This is where all the coding was done. Taking the information gathered from the previous stages, the functionality of the application was coded in bits, implementing one functionality at a time.

- **Design Tools:**

The following are the design tools, programming languages, and frameworks used for developing this application:

- **SwiftUI:**

SwiftUI is Apple's new framework for building user interfaces across all Apple platforms using the same code. It lets us design apps in a declarative way. That's a fancy way of saying that we tell SwiftUI how we want our user interface to look and work, and it figures out how to make that happen as the user interacts with it.

- **Agora SDK:**

Agora is an SDK that allows you to easily implement video calls and live distribution on your own iOS/Android applications and websites.

- **Firebase Authentication:**

The Firebase Authentication SDK provides methods to create and manage users that use their email addresses and passwords to sign in. Firebase Authentication also handles sending password reset emails.

The screenshot shows the Firebase Authentication console for the 'StudyPal SwiftUI' project. The left sidebar includes links for Project Overview, Firestore Database, Authentication (which is selected), Storage, and Extensions. The main 'Authentication' page has tabs for Users, Sign-in method, Templates, Usage, and Settings. A search bar at the top allows searching by email address, phone number, or user UID. Below the search bar is a table listing nine users. Each user entry includes columns for Identifier, Providers (showing email), Created (date), Signed In (date), and User UID. The table also features standard data grid controls like sorting, filtering, and row selection.

Identifier	Providers	Created	Signed In	User UID
a@9.co	✉️	Aug 17, 2022	Aug 17, 2022	2C1qeNvGjBNlw14sE7p4LHAuane2
a@8.co	✉️	Aug 17, 2022	Aug 17, 2022	QyfzwkpYSvbhGqRGNrctekt4Z43
a@7.co	✉️	Aug 17, 2022	Aug 17, 2022	VB8k15L65xUme1TshKVdjRhksUX2
a@6.co	✉️	Aug 17, 2022	Aug 17, 2022	lysCzprFeSWG16j2hjcM5IKEG4Q2
a@5.co	✉️	Aug 17, 2022	Sep 24, 2022	EPKCNf15rCYQnYM7yiK0wKdloNw...
a@4.co	✉️	Aug 16, 2022	Sep 29, 2022	KCJhyrzSimeNfyTo1DuWeL73p6u1
a@3.co	✉️	Jul 29, 2022	Aug 16, 2022	btl1qsJxhXZYalecGyP0p2zZII2
a@2.co	✉️	Jul 29, 2022	Sep 29, 2022	KsSaQG15Kg5UMRcwW8qbF6L12...
aalexanderkwaku@yahoo...	✉️	Jul 21, 2022	Jul 26, 2022	Ll8ShEWWA7RC5KenxMTb7SyM...

Firebase Users and Authentication for StudyPal

• Firebase Cloud Firestore:

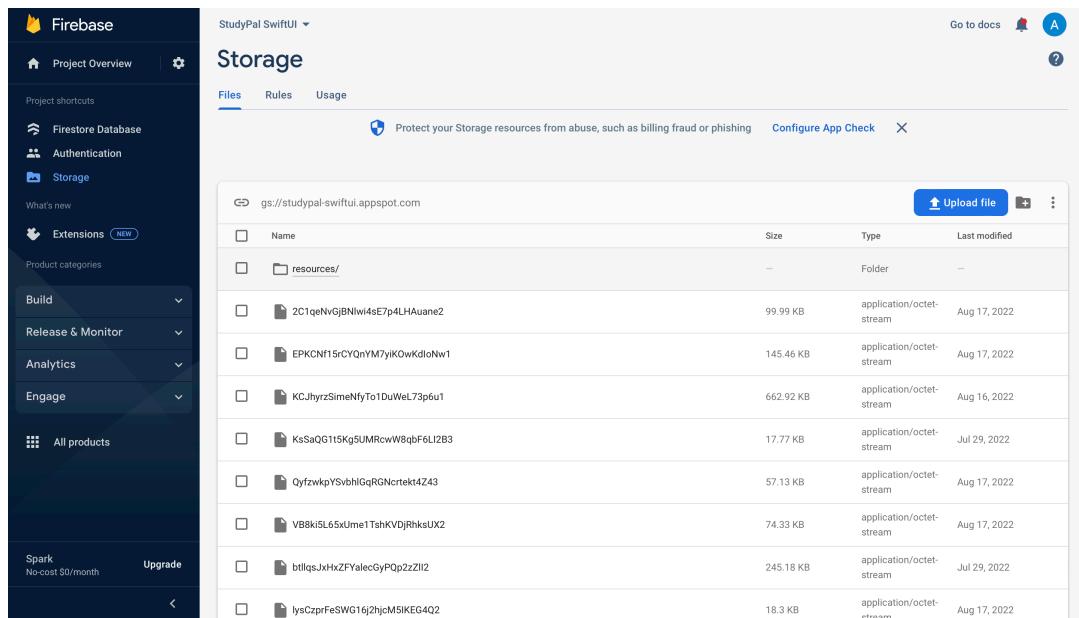
Cloud Firestore is a NoSQL document database that lets you easily store, sync, and query data for your mobile and web apps - at global scale.

The screenshot shows the Cloud Firestore console for the 'StudyPal SwiftUI' project. The left sidebar includes links for Project Overview, Firestore Database (selected), Authentication, and Storage. The main 'Cloud Firestore' page has a message about the Local Emulator Suite and a 'Get started' button. A banner at the top indicates 'NEW: More administrative and troubleshooting features in Google Cloud' with a 'Dismiss' button. The main area shows a 'users' collection under the 'studypal-swiftui' database. The 'users' collection contains documents for each of the nine users listed in the previous screenshot. The interface includes standard Firestore navigation and editing tools.

Firestore Database for StudyPal

- **Firebase Storage:**

Cloud Storage for Firebase lets you upload and share user generated content, such as images and video, which allows you to build rich media content into your apps. Your data is stored in a Google Cloud Storage bucket — an exabyte scale object storage solution with high availability and global redundancy.



The screenshot shows the Firebase Storage interface for the 'StudyPal SwiftUI' project. On the left, there's a sidebar with links to Project Overview, Firestore Database, Authentication, Storage, Extensions, Analytics, Engage, and All products. The Storage section is selected. The main area is titled 'Storage' and has tabs for Files, Rules, and Usage. A banner at the top right says 'Protect your Storage resources from abuse, such as billing fraud or phishing' and includes a 'Configure App Check' button. Below is a table of uploaded files:

Name	Type	Last modified
resources/	Folder	—
2C1qeNvGjBNlw4sE7p4LHuanne2	application/octet-stream	Aug 17, 2022
EPKCNf15rCYQnYM7ylK0wKdloNw1	application/octet-stream	Aug 17, 2022
KCJhyrzSimeNfyTo1DuWeL73p6u1	application/octet-stream	Aug 16, 2022
KsSaQG1t5KgSUMRcwW8qbF6Ll2B3	application/octet-stream	Jul 29, 2022
QyfzwpYsvbhGqRGNrctek4Z43	application/octet-stream	Aug 17, 2022
VB8kiSL65xUme1TshKVDRhksUX2	application/octet-stream	Aug 17, 2022
btlqsJxHxFYalecGyPQp2zZl2	application/octet-stream	Jul 29, 2022
lysCzprfeSWG16j2hcM5IKEG4Q2	application/octet-stream	Aug 17, 2022

Firebase Storage for StudyPal

- **Xcode:**

Xcode is Apple's integrated development environment for macOS, used to develop software for macOS, iOS, iPadOS, watchOS, and tvOS.

- **Cocoapods:**

CocoaPods is a dependency manager for Swift and Objective-C Cocoa projects. It has over 92 thousand libraries and is used in over 3 million apps. CocoaPods can help you scale your projects elegantly.

- **Swift Package Manager (SPM):**

The Swift Package Manager is a tool for managing the distribution of Swift code. It's integrated with the Swift build system to automate the process of downloading, compiling, and linking dependencies.

- **Testing:**

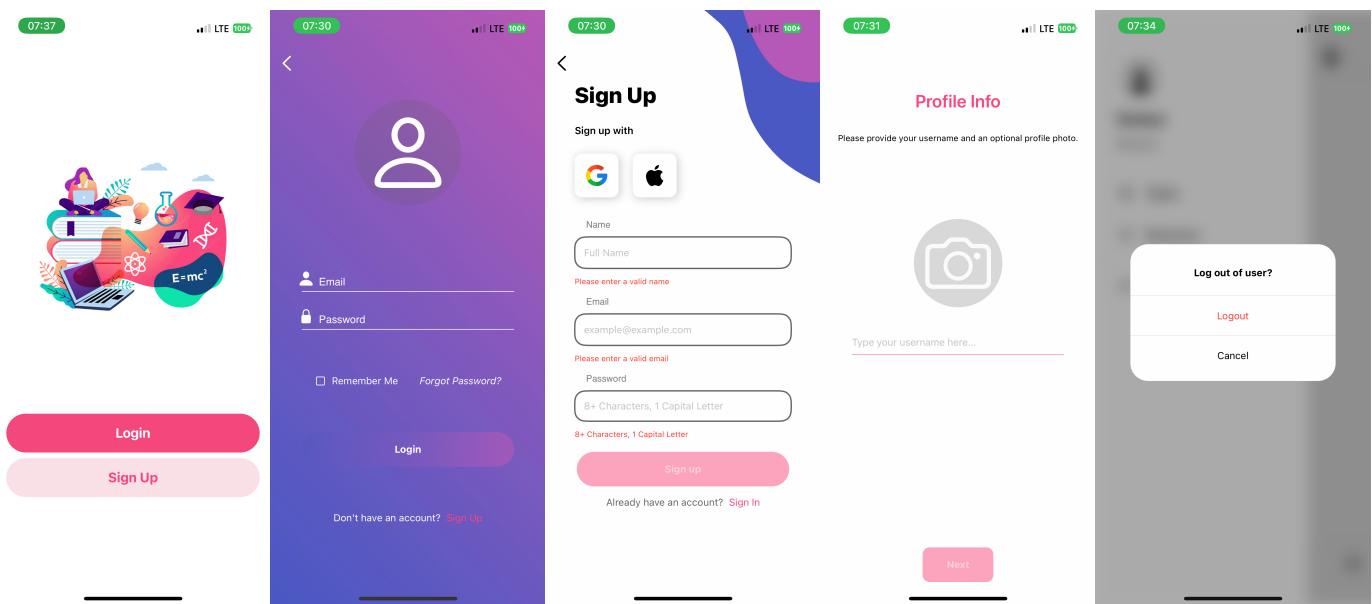
After a function was coded and implemented, it was tested and debugged when a bug was found.

CHAPTER 3: IMPLEMENTATION AND RESULTS

Usage

- **Sign Up / Login / Logout:**

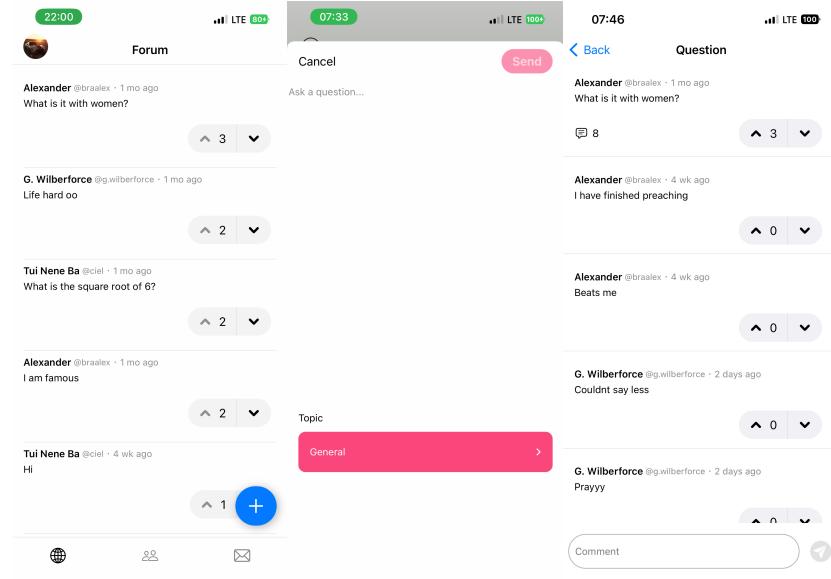
Already existing users can sign in to access the features of the app. New users will have to sign up using their name, a valid email, strong password, username and an optional display picture. When a user is done using the platform, they can choose to logout or stay logged in.



- **Public Forum:**

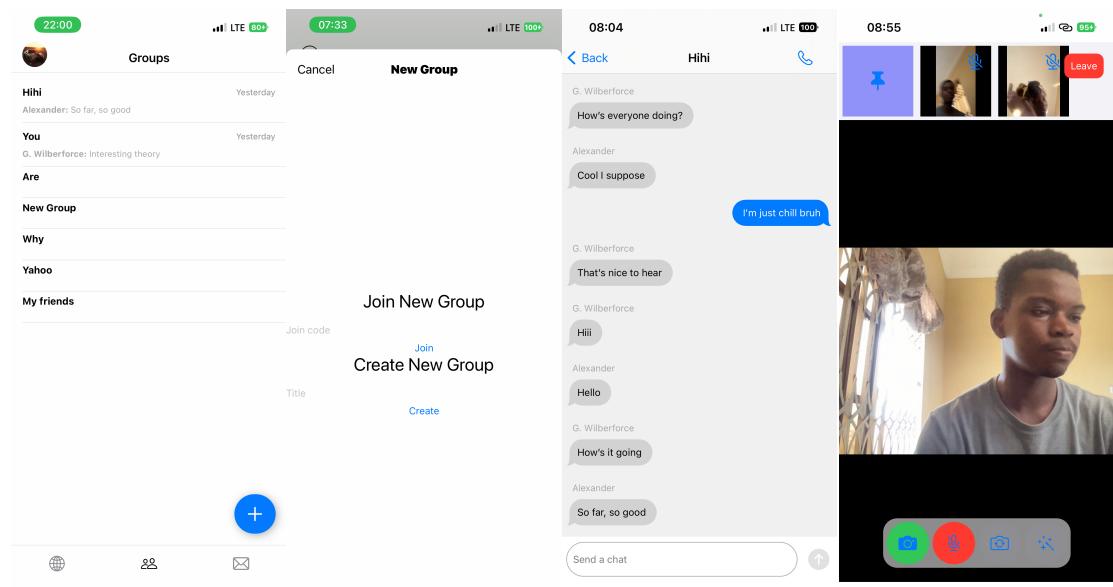
Users can post questions they need help with or discussions they want to start.

Other users can comment and either upvote or downvote the question or discussion and people's comments.



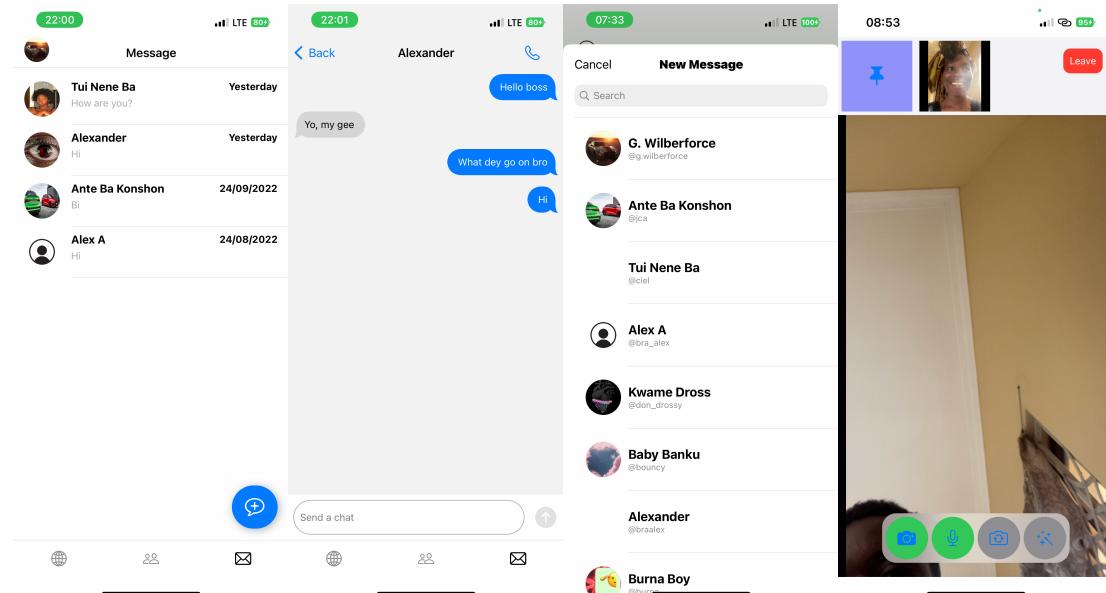
- **Groups:**

Users can create or join groups, send group messages and make group video/voice calls.



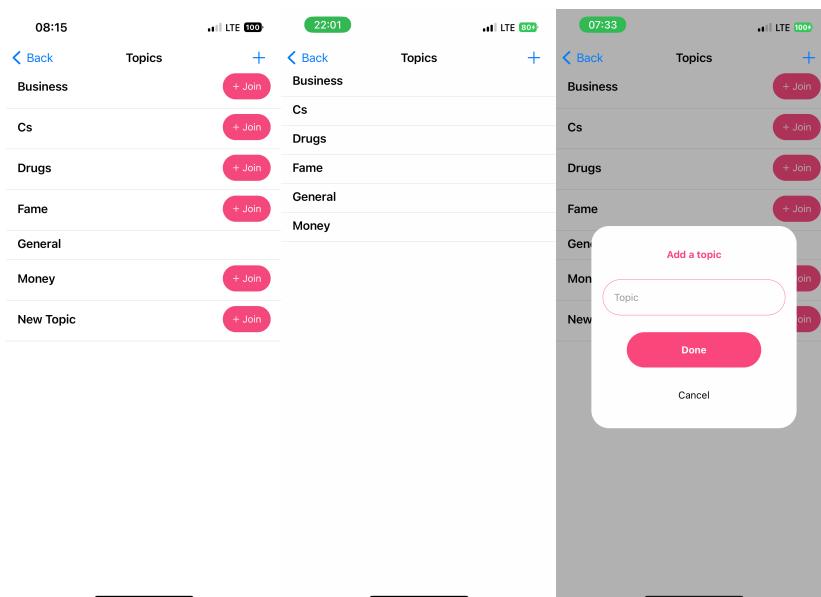
- **Messages:**

Users can start new messages with peers and make video/voice calls.



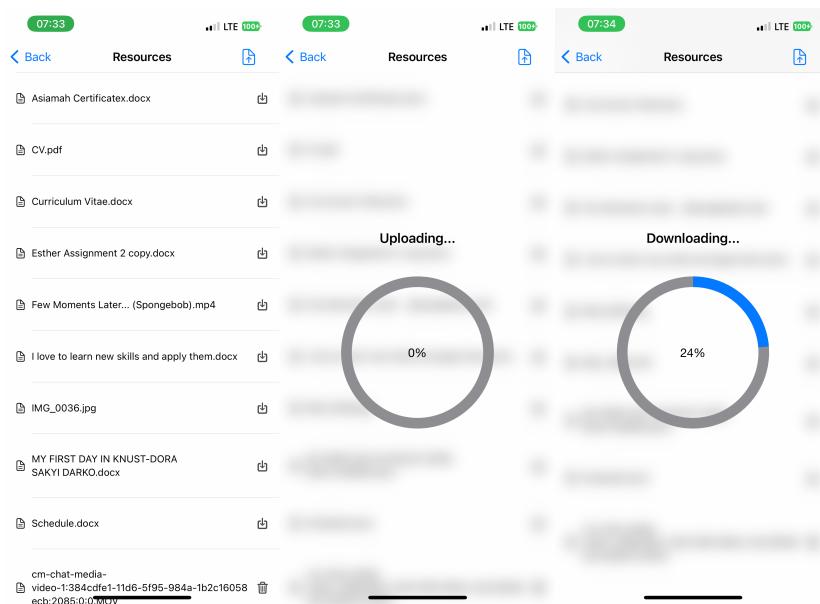
- **Topics:**

Users can join topics of interest or add new topics.



- **Resources:**

Users can upload resources or documents they want to share in any file format for other users to download or access.



CHAPTER 5: CONCLUSION

The aim of StudyPal is to provide a free avenue for students to share, collaborate and discuss ideas but it's not perfect. The application can be improved by adding sticky notes to help users take notes while learning, improving the user experience of the app and allowing users to send media files in groups and private messages.