



iKARD

**YUELING CEN, JUSTIN RUSSO,
MARIO TAGLIC, ZHEN XIN TAN RUAN**

**OBJECTED ORIENTED DESIGN (CSE - 687)
SYRACUSE UNIVERSITY**

FINAL STRUCTURE OF A REQUIREMENT

SYRACUSE, NEW YORK

SPRING 2021

TABLE CONTENTS

1. PREFACE

1.1 PURPOSE.....	04
------------------	----

2. INTRODUCTION

2.1 PRODUCT SCOPE.....	04
2.2 WHY IS IT A KILLER APP.....	04
2.3 TARGET AUDIENCE.....	05

3. GLOSSARY

3.1 DEFINITIONS, ACRONYMS AND ABBREVIATIONS.....	06
--------------------------------------------------	----

4. USER REQUIREMENT DEFINITION

4.1 PRODUCT PERSPECTIVE AND FUNCTIONS.....	06
4.2 DESIGN AND IMPLEMENTATION CONSTRAINTS.....	07

5. SYSTEM EVOLUTION

5.1 ASSUMPTIONS AND DEPENDENCIES.....	08
---------------------------------------	----

6. APPENDICES

6.1 HARDWARE AND DATABASE CONFIGURATION.....	09
6.2 SOFTWARE AND SYSTEM CONFIGURATION.....	10

7. INTERFACE REQUIREMENTS

7.1 SOFTWARE INTERFACES.....	10
------------------------------	----

8. REQUIREMENTS

8.1	FUNCTIONAL USER REQUIREMENTS.....	21
8.2	FUNCTIONAL SYSTEM REQUIREMENTS.....	21
8.3	NON-FUNCTIONAL USER REQUIREMENTS.....	22
8.4	NON-FUNCTIONAL SYSTEM REQUIREMENTS.....	23
8.5	BUSINESS CARD COMPONENTS.....	23
9.	SYSTEM ARCHITECTURE	
9.1	FRONT END ARCHITECTURE.....	25
9.2	BACK-END ARCHITECTURE.....	26
10.	SYSTEM MODELS	
10.1	USE CASE DIAGRAMS.....	27
10.2	ACTIVITY DIAGRAMS.....	32
10.3	STATE DIAGRAM.....	35
10.4.	CLASS DIAGRAM.....	36
10.5	SEQUENCE DIAGRAMS.....	38
10.6.	SYSTEM DIAGRAM.....	48
11.	BIBLIOGRAPHY.....	50

1. PREFACE

1.1 PURPOSE

The documents below are for the app iKard v.1.0, a digital wallet for business card. The document provides the system's purpose and services as well as the constraints under which the system must operate. Additionally, the document describes the characteristic, system overview and architecture, technical specification, and expectations for the application as a whole for the first version of iKard mobile app. This SRS is written for system stakeholders including users, developers, UI designers, testers, project managers and software architects.

2. INTRODUCTION

2.1 PRODUCT SCOPE

The world is turning to a digital landscape, and it is time for old school business tactics to adapt. Most professionals carry business cards with them everywhere to go. Business cards are a networking tool for professionals to promote a business, company division, or service. It is also known that with the correct design, color, and images provide advantages when sharing your card. There are many disadvantages to the old school business card model. The first disadvantage, current business cards can be lost and end up somewhere in the garbage. Next, professionals will continuously have to spend money on ordering business cards from services like Vistaprint either to restock or to change information on the current card. Third, there is a limited potential reach in networking on your business card.

2.2 WHY IS IT A KILLER APP?

iKard mobile app provides solutions for all the problems described above. Based on recent data 88% of business cards handed out get thrown away in less than one week, and 39% of people won't do business with a company if their card is cheap looking. iKard application can revolutionize the way professionals can network by providing a modern, digitally safe, and new way to connect other professionals with geolocation. Business professionals are having a hard time interacting with other new business professionals without knowing their prior information. A combination of using the app's map and cards geolocation feature makes this application different from other similar apps currently in the Appstore or the google play in the category. Moreover, providing digital business cards for individuals to use for contactless interaction and exchange information with just a single scan. Finally, the simplicity and fluidity of how iKard mobile app handles user interaction is a big step to making a more modern world for old school ways. The target user for iKard is for business professionals who want to adapt and network in the modern world.

2.3 TARGET AUDIENCE

The app provides solutions for all of the problems mentioned in the product scope section for business cards. Additionally, the app targets users who want the opportunity to share and promote their own business on the application's map for other users who are looking to connect or network with other professionals. Generally, our target users can be divided into three group:

- 1. Executive and CEO**
- 2. Students, Recruiters and entrepreneurs (Networking)**
- 3. Customers looking for services**

3. GLOSARRY

3.1 DEFINITIONS, ACRONYMS AND ABBREVIATIONS

1. **Mapbox** - mapbox is a mapping framework that provides precise location data and powerful developer tools to facilitate enhanced mapping features. In iKard mapbox helps deliver a seamless way to find local contacts by using the map feature. Mapbox studio allows for complete and easy customization of the map system which helps to tailor the map around our needs.
2. **Firebase** - Firebase is a cloud database system by Google which provides numerous advanced features such as Realtime Database systems, Authentication, Machine Learning, and other various functions. Cloud Firestore, allows iKard to remain serverless while maintaining the opportunity to scale. Additionally, Firebase provides an end-to-end identity solution, keeping user authentication simple and secure.
3. **User** - This role is defined for users who will use the application to create, store and share business cards with other users.
4. **QR Code** - Two-dimensional version of the barcode, typically made up of black and white pixel patterns. It allows the user to store information and share it with other users with just a single scan.

4. USER REQUIREMENTS DEFINITION

4.1 PRODUCT PERSPECTIVE AND FUNCTIONS

1. User will login using the credentials and if validated correctly, it will redirect to the home screen in the app. User can see their wallet with the added business cards in their account. User can create, add or scan a new business card to their wallet. In case the user wants to scan someone code to add their business card, the app will automatically

lead the user to the camera to scan the code and add the business card into the user' wallet.

2. The application will use firebase system with email and password to login and sign up. In case the user forgets the password, they can reset the password using their email account.
3. The application will close when the user exits the app automatically, but the user default login information will be saved inside the phone so next time the user opens the app, the user is not required to input the information again. The app will automatically login and direct the user to the home screen with the business card wallet.

4.2 DESIGN AND IMPLEMENTATION CONSTRAINTS

1. One of the critical aspects of our application is its inherent dependency on a stable internet connection because the app requires the use of API for the map and geolocation feature. Also, the app requires the user to run the latest IOS version on their phones.
2. In addition to the latest IOS version installed on the phones, it is also necessary for the user to allow the app to access the camera on their devices to use the scan feature.
3. We have used a third-party library inside swift environment to generate a unique QR code using the business card information.
4. We have implemented an API library mapbox to generate and plot the user' business location into the app map view. Thus, the system is dependent of the API being available always. In addition, the loading time of the map depends on the user internet.

5. SYSTEM EVOLUTION

This system is fundamentally predicated on the assumption that serverless architecture will satisfy all current requirements. While such an environment provides increased flexibility and expedited development time, it also introduces limitations such as higher latency in certain scenarios and diminished control over the software stack. As features are added the serverless design could become a limiting factor in both the implementation and design of such features. At this point a centralized server architecture may be considered, or perhaps various other cloud platform solutions in order to fulfill additional requirements.

5.1 ASSUMPTIONS AND DEPENDENCIES

1. ASSUMPTIONS

- The app will be deployed on Firebase and Google Cloud; thus, we make an assumption that it is available without interruptions.
- The system also heavily relies on swift libraries. An assumption that the future version of the libraries does not drastically change/affect the application.
- The system will not be able to detect if the information entered about a business is real or not. We assumed that if the user is creating a business card for his/her company, the user will input the correct information and the business that the user is registering is legal and complies with state and federal laws.

2. DEPENDENCIES

- The app will rely on firebase and google database tools to manage and maintain the app running.

- The system will depend on mapbox for the geolocation and map functionality.
- The app relies heavily on swift libraries UIKit, Firebase/Analytics, and JGPProgressHUD for the app UI integration and functionality.
- The loading time of the app will heavily depend on Firebase and the user internet connection.

6. APPENDICES

For an optimal experience using iKard, we recommend running using the latest version of iOS that is supported on your device. Additionally, a stable internet connection is needed to establish connection with Firebase to, search, share, or access saved and created cards. To ensure logical organization of the data, two collections are implemented in Firestore. One collection manages the users of the system, while the other collection focuses solely on the cards created and managed by the users. In this scenario, the users in the user collection have the ability to manage the data associated with their cards in the card collection. However, users are not able to manage the data of cards which they did not create. While they can't change the data of cards they did not create, they can read data from other users' cards if they have the associated id received from a QR code or featured locally on the map.

6.1 HARDWARE AND DATABASE CONFIGURATION

- Firebase iOS version 7.10.0 – April 2021
 - Real time Database (Write and read enabled)
 - Cloud Firestore (Write and read enabled)
 - Authentication (Email and password)

- Analytics
 - Remote Config
- iPhone device with access to Appstore in united states

6.2 SOFTWARE AND SYSTEM CONFIGURATION

- IOS software version 13.0 or higher system version
- Permission active for camera access
- Xcode version 12.4 (simulator)

7. INTERFACE REQUIREMENTS

7.1 SOFTWARE INTERFACES

Our app will use swift libraries, mapbox and IOS tools to implement all of the functionalities. The mobile app has one interface for all users. The interface for the entire application is going to be developed using swift and Xcode. At the time of login, firebase will authenticate the user using the user input for email and password text field. The first view for the mobile app is the loading screen followed by the onboarding screen. The onboarding screen will only appear once after the user installs the app. When the user sees the onboarding screen the app will no longer show it to the user.

After the onboarding the user will have the start screen where they can choose if they want to login or create a new account. If the user selects login, it will redirect the user to the login screen in which the user will input the email and password to access their account. If the user select create an account, it will redirect the user to the sign-up screen in which the user will register a new account using email and password. Once the user login or sign up to a new account the app will redirect the user to the home view where they will be able to see all the business cards they

have on the wallet, add new cards or create a new card. The home view will have a tab bar menu with the different view the user can access. The tab bar menu consists of (Home, Maps, Scan, Profile and Settings). Each option in the tab bar menu is a different view that the user can go to.



Figure 1. Launch Screen

2:39



Welcome to iKard

Explore and engage with the people and places around you instantly. iKard is bridge between digital space and you please take a moment to review our guideline

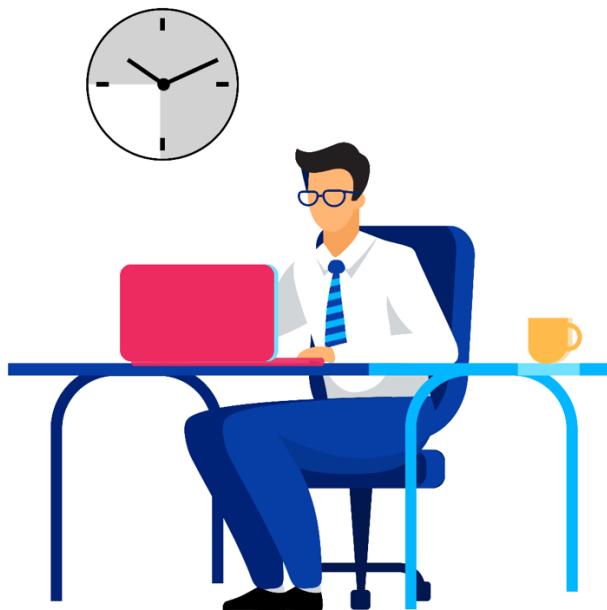
• • •

Next

Skip

Figure 2. Onboarding screen

2:37



Fast And Easy To Use

Very time you need to update your card, instead of reprinting the entire batch, you can simply edit your car on the app



Next

Skip



Figure 3. Onboarding screen

2:37 ⚡



Multiple Customization

An electronic business card is sure to distinguish
your brand from the rest at the touch of your
fingers



Next

Skip

Figure 4. Onboarding screen

2:37



Simple Scan Technology

Add any business card to your e-wallet account at any time by simply scanning the QR code inside their card



Continue



Figure 5. Onboarding screen

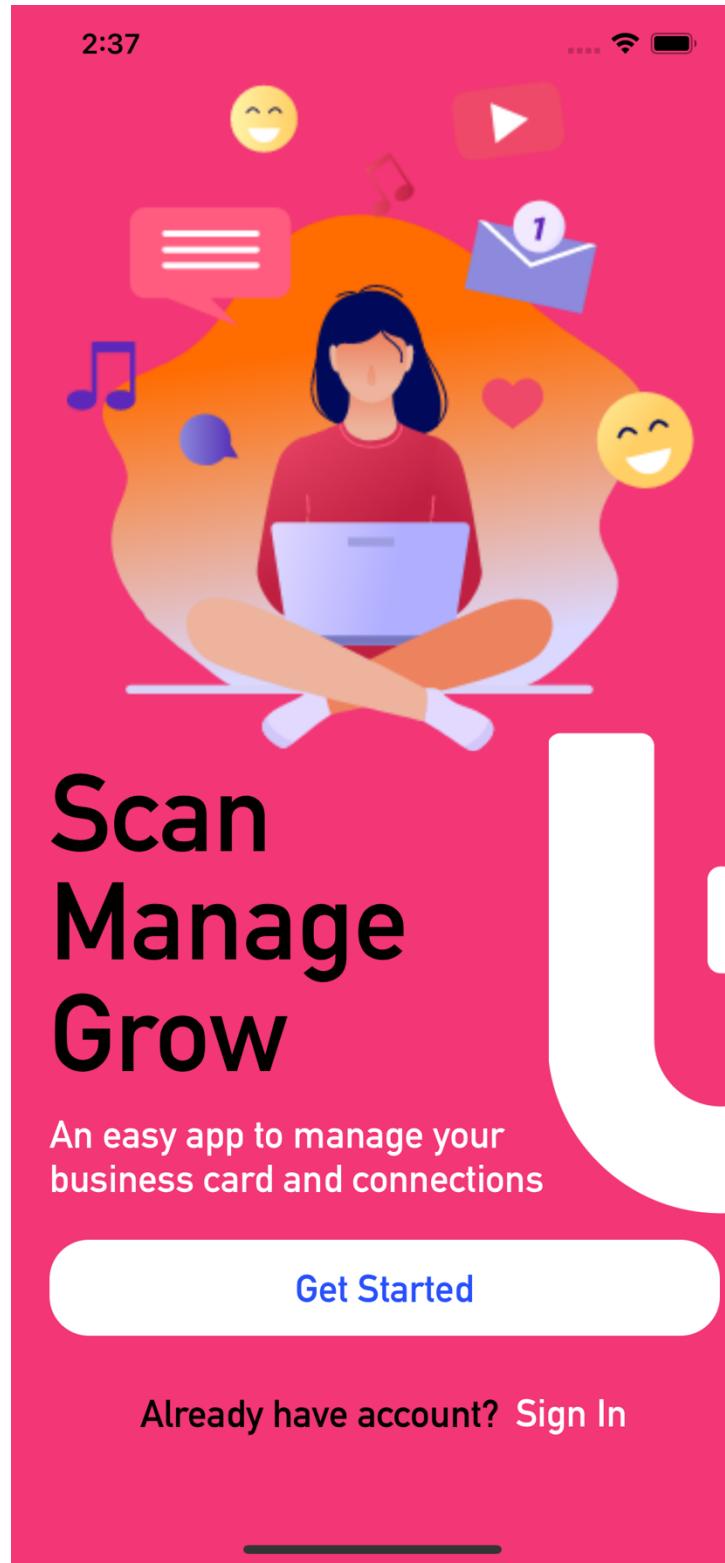


Figure 6. Start screen

2:37



WELCOME

Email:

Password:

Sign In

[Forgot password?](#)

Don't have an account? [Join now](#)

Figure 7. Login screen

2:37



REGISTER

Email:

Password:

Repeat password:

By signing up, you agree our terms & conditions and privacy policy

Sign Up

Already have an account? [Sign In](#)



Figure 8. Registration screen

2:37



PASSWORD

Enter your email and we will send you
instructions on how to reset your password

Email:

Submit

Cancel



Figure 9. Forgot screen

2:36



Wallet

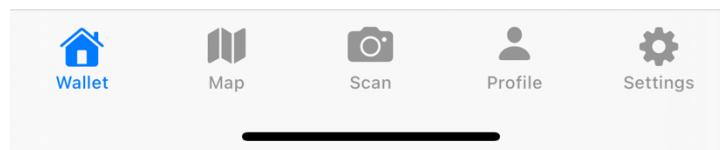


Figure 10. Home screen

8. REQUIREMENTS

8.1 FUNCTIONAL USER REQUIREMENTS

UR.1 ACCOUNT CREATION: Each user will be able to create an account with a unique username and minimum 6-digit password.

UR.2 CREATE BUSINESS CARDS: Each user will be able to create an infinite amount of business cards.

UR.3 ADD CARDS: Each user will be able to add another business card by the other card's unique code or by the add button to their card collection.

UR.4 EDIT CARDS: Each user will be able to edit any of their own business cards

UR.5 REMOVE CARDS: Each user will be able to remove a shared business card from their card collection.

UR.6 SEARCH WALLET: Each user will be able to search/look through their business card wallet.

UR.7 FIND CARDS ON GEOLOCATION: Each user will be able go on the map within the app to find other business cards in the location to look for other professionals and promote their own business or service.

8.2 FUNCTIONAL SYSTEM REQUIREMENTS

APP FUNCTIONAL COMPONENTS: User verification, geolocation, maps system, custom business card design, electronic cards, and QR code scanner.

SR.1 USER VERIFICATION: The system will authenticate the user information and verify that the information entered is correct.

SR.2 DATABASE STATUS: The database holding the user's information should be available whenever the user wants to login.

SR.3 DATABASE UPDATE: Firebase should update the user new password as soon as the user decides to change it.

SR.4 BUSINESS CARD: The system should allow the user to create new business cards using the business card components

SR.5 DATABASE STORE: Firebase should store and update the card and user information in real time.

SR.6 GEOLOCATION: The system should allow the user to access the map and see other user's business on the map.

SR.7 QR CODE: The app should allow the user to scan someone else QR code to add their business card into their wallet app.

8.3 NON-FUNCTIONAL USER REQUIREMENTS

UR.1 STORE DATA: The user will be able to store the account information in a database.

UR.2 STORE BUSINESS CARD: The user will be able to store the information about the card inside the database.

UR.3 USER INTERFACE: The user app interface should be user friendly, and user should be able to navigate through the app with ease and with minimal training

UR.4 ACCESSIBLE: The app should be accessible only to registered users with the correct credentials.

8.4 NON-FUNCTIONAL SYSTEM REQUIREMENTS

APP NON-FUNCTIONAL COMPONENTS: Firebase database, google cloud, user authentication, and geolocation.

SR.1 USER AUTHENTICATION: The application should allow users, with the correct email and password to access the app.

SR.2 RESPONSIVENESS: The application should run fast with minimal lag in the processing.

SR.3 SHOW CARDS: system should be able to retrieve the business card information and show it on the wallet view.

8.5 BUSINESS CARD COMPONENTS

1. **LOGO:** A user can add an image of their choice (picture, logo, etc.)
Optional
2. **NAME:** A user can add a full name to display on their card. Required
3. **ADDRESS:** A user can add their business address on their card. Optional
4. **EMAIL:** A user can add their email on their card. Required
5. **POSITION:** A user can add their position in the business. Required
6. **PHONE NUMBER:** A user can add their business number, fax, telephone number etc. Optional
7. **SECONDARY PHONE NUMBER:** A user can add their business number, fax, telephone number etc. Optional
8. **UNIQUE SHAREABLE CODE:** A unique code is generated for each business card to share Generated

9. SHOW ON MAP: A user will be able to select if they want the business card to display on the app's map. Optional

9. SYSTEM ARCHITECTURE

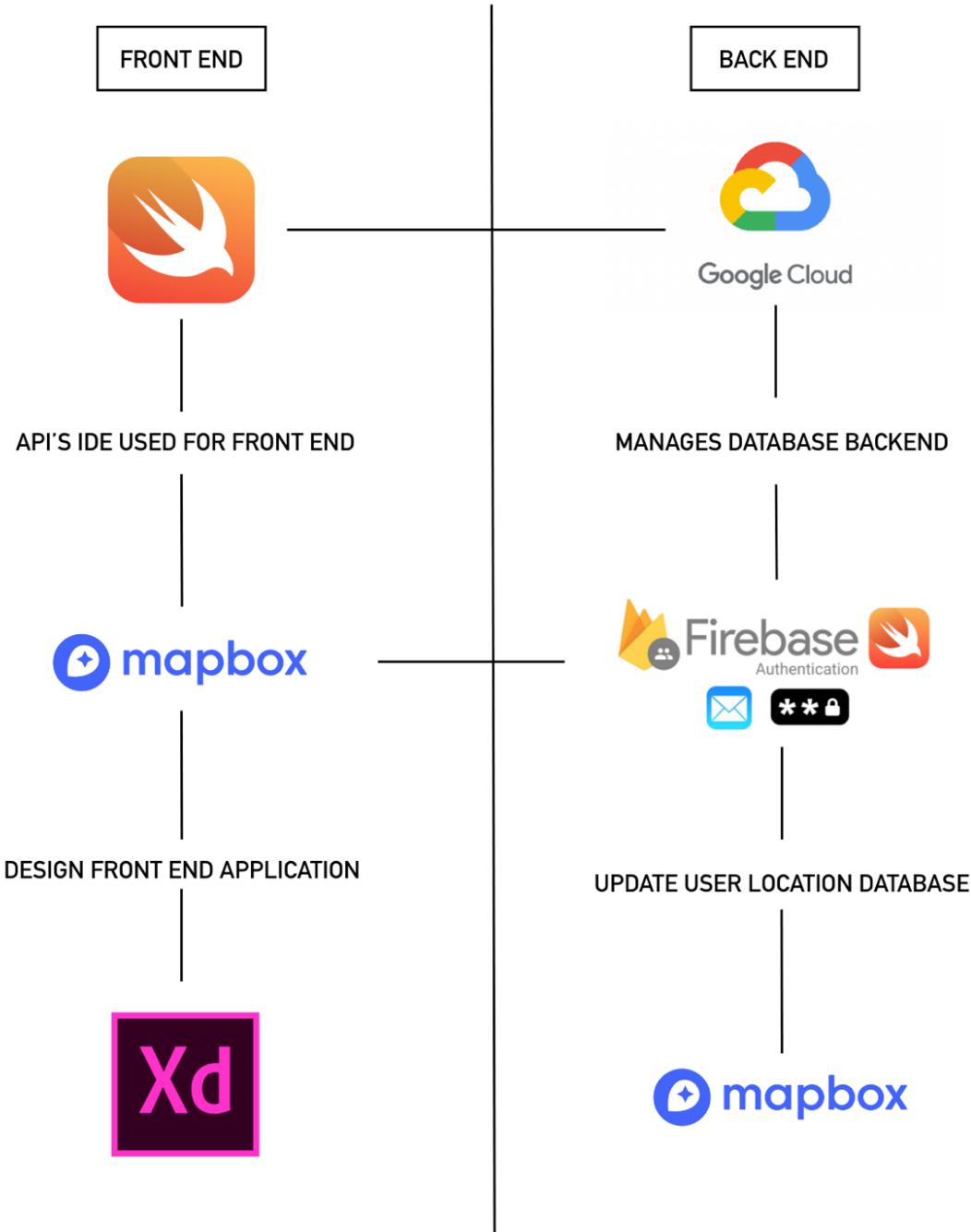


Figure 11. iKard system architecture

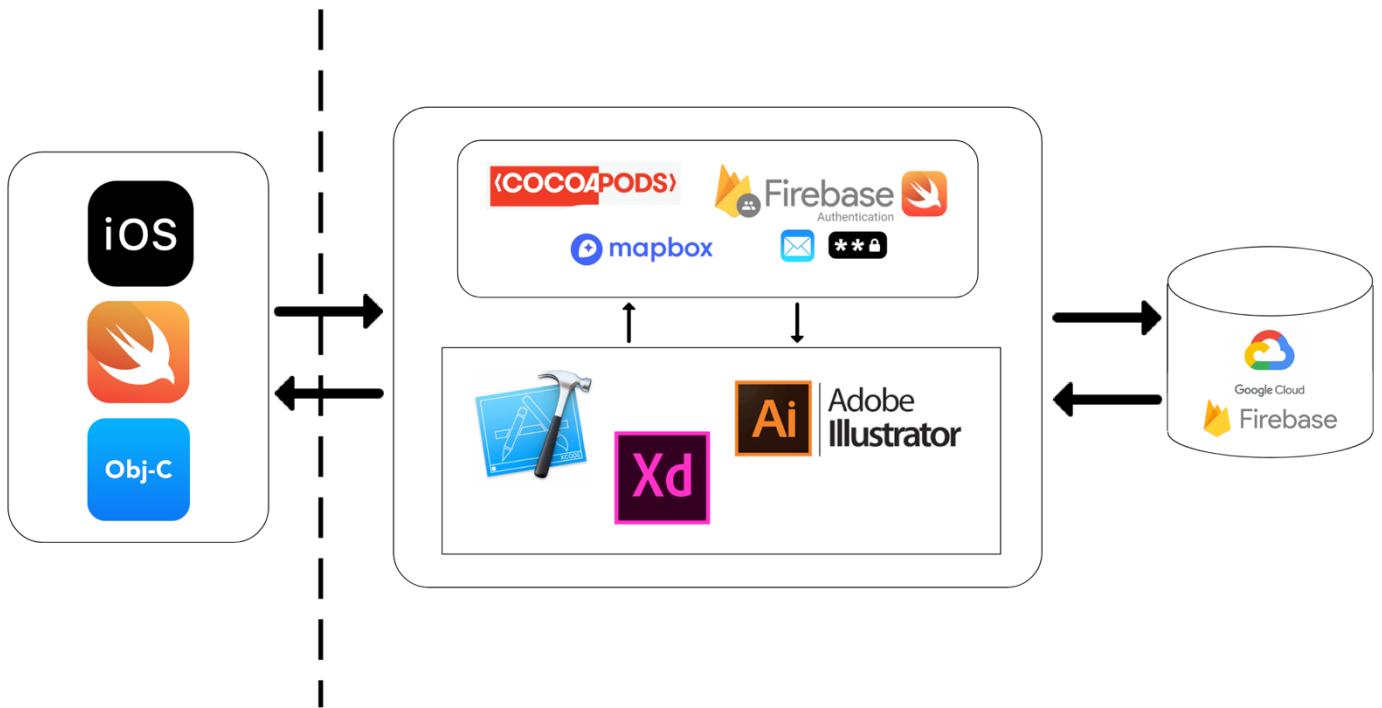


Figure 12. System architecture

9.1 FRONT END ARCHITECTURE

Front end of the mobile app is developed using swift and objective-C library and storyboard. User will use login, sign up, start view, onboarding and home view to interact with the application. The user will have a mobile and user-friendly interface easy to interact and understand. We use adobe XD and adobe illustrator to create a visual image of the UI interface for the entire application with interactive buttons and functions to simulate the flow of the real app before coding in Xcode.

9.2 BACK-END ARCHITECTURE

The back end of the application is powered by Firebase and google cloud services. We are also using external libraries and API services; mapbox,

cocoa pods, firebase authentication for the different features of the application.

1. MAPBOX

We are using mapbox API to design and integrate the geolocation and maps for the application. With this library we can assign the user business card inside a map that we can customize to display and show the business details.

2. COCOAPODS

Cocoapods is an application-level dependency manager for objective C, that provides a standard format for managing external libraries for the project. For this application we are using cocoapods to install and manage all the external libraries that we are using to develop the mobile application.

3. FIREBASE AUTHENTICATION

We use Firebase authentication on the back end to verify and store the user login information in the database. Firebase authentication library allows us to create an account for the user using email and password and then allow that user to login to the account using those same credentials.

10. SYSTEM MODELS

10.1 USE CASE DIAGRAMS

REGISTRATION AND LOGIN (UR.1 -> SRC.2)

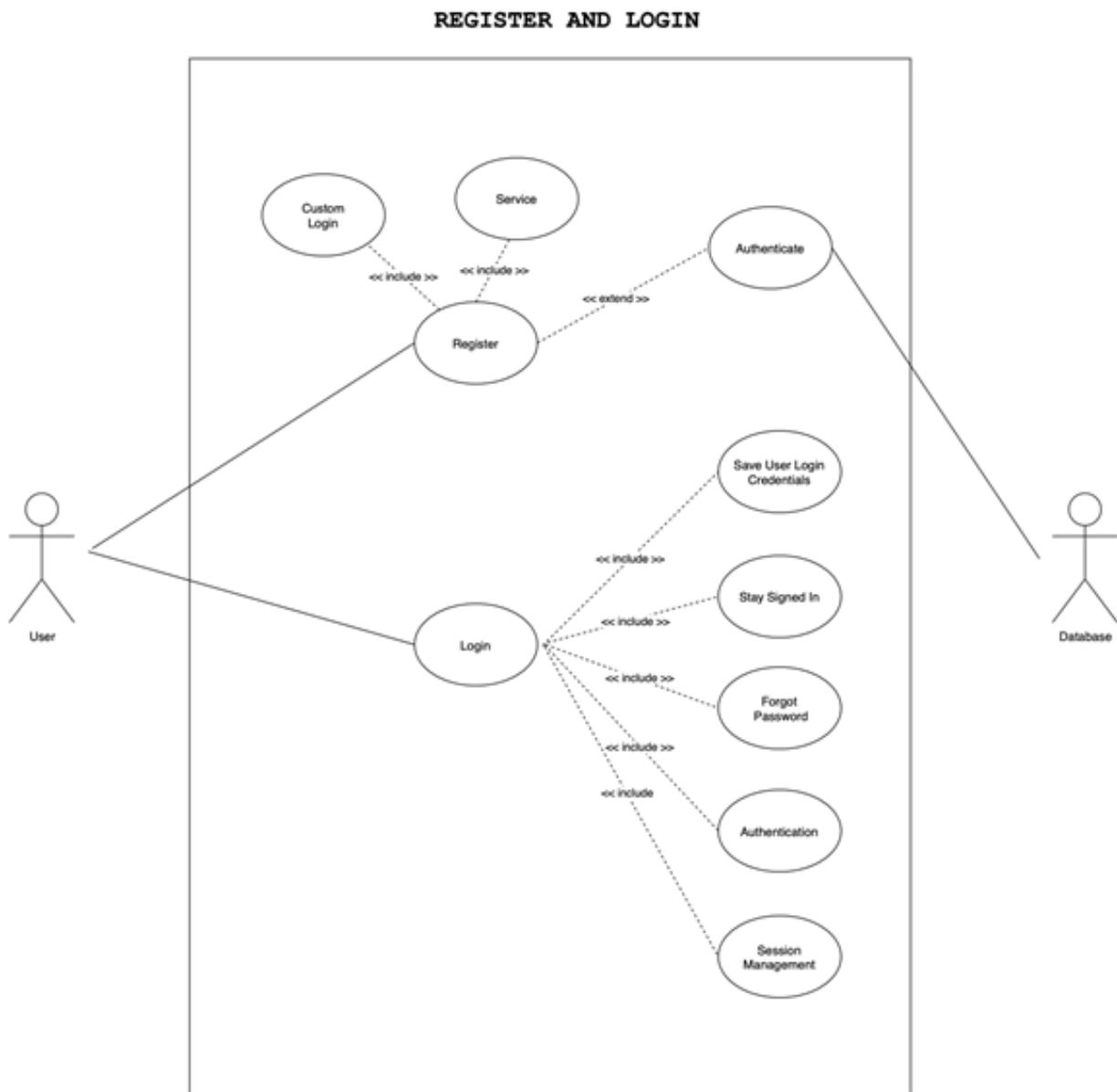


Figure 13. Registration and login use case diagram

- The use case diagram for login and registration shows the functionalities for login and register for a new account on the iKard mobile application.
- When the user opens the app, the user has two options either to login to an existing account or create a new account.
- When the user register for a new account the firebase will validate the email address and store the data in the database for the new user. If the user wants to login to an existing account, firebase will check and authenticate the user credentials and make sure it matched with a record in the database. If the user forgets the password the user can select to reset password and firebase will send an email for the user to set up the new password.
- **Login** – The user can use the login functionality to login to their account.
- **Save login credentials** – The user login information will be saved on database and user device for fast login next time.
- **Stay signed in** - Once the user logins the app will be running on the background so the user can access to the business card wallet instantly.
- **Forgot password** – The user can use this functionality to reset the account password
- **Authentication** – The system will verify the user credentials and authenticate the user to access their account.
- Session Management – This functionality will manage the status of the account.

- **Register** – The user can use this functionality to register a new account with email and password.

RESET PASSWORD (UR.1 -> SRC.3)

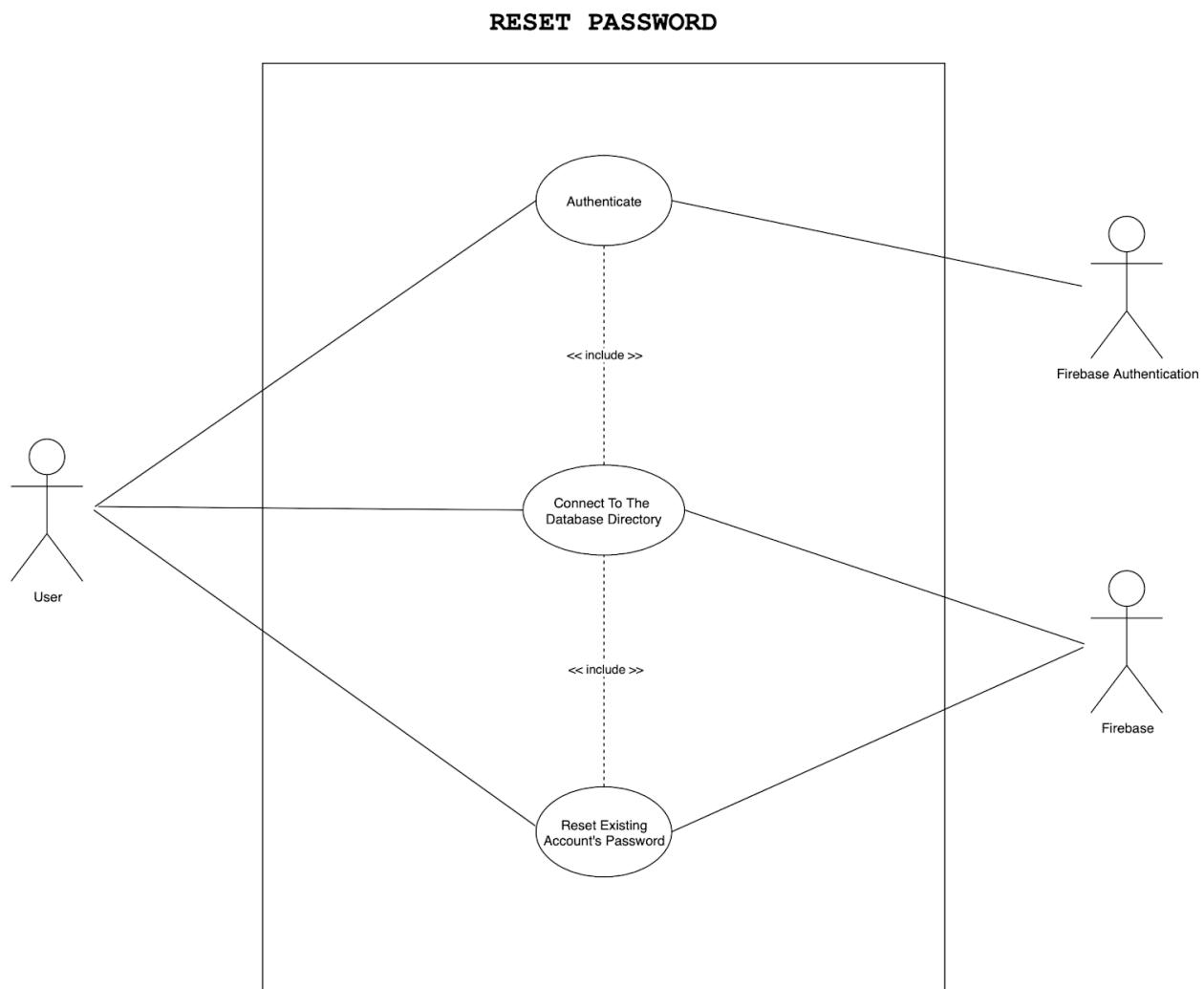


Figure 14. Reset password use case diagram

- The use case diagram for reset password represents the functionalities that the user can use to reset the password.

- **Authenticate** – Firebase will authenticate that the email that the user is trying to reset the password is valid.
- **Connect to the database directory** – This functionality will connect the app to firebase and find the locate the email that the user wants to reset the password.
- **Reset existing account's password** – The user can only reset the password of an existing account on the app. And firebase will notify the user via email with a link to set up the new password for the account.

MANAGE CARD (UR.3, UR.4, U.R.5 -> SRC.1)

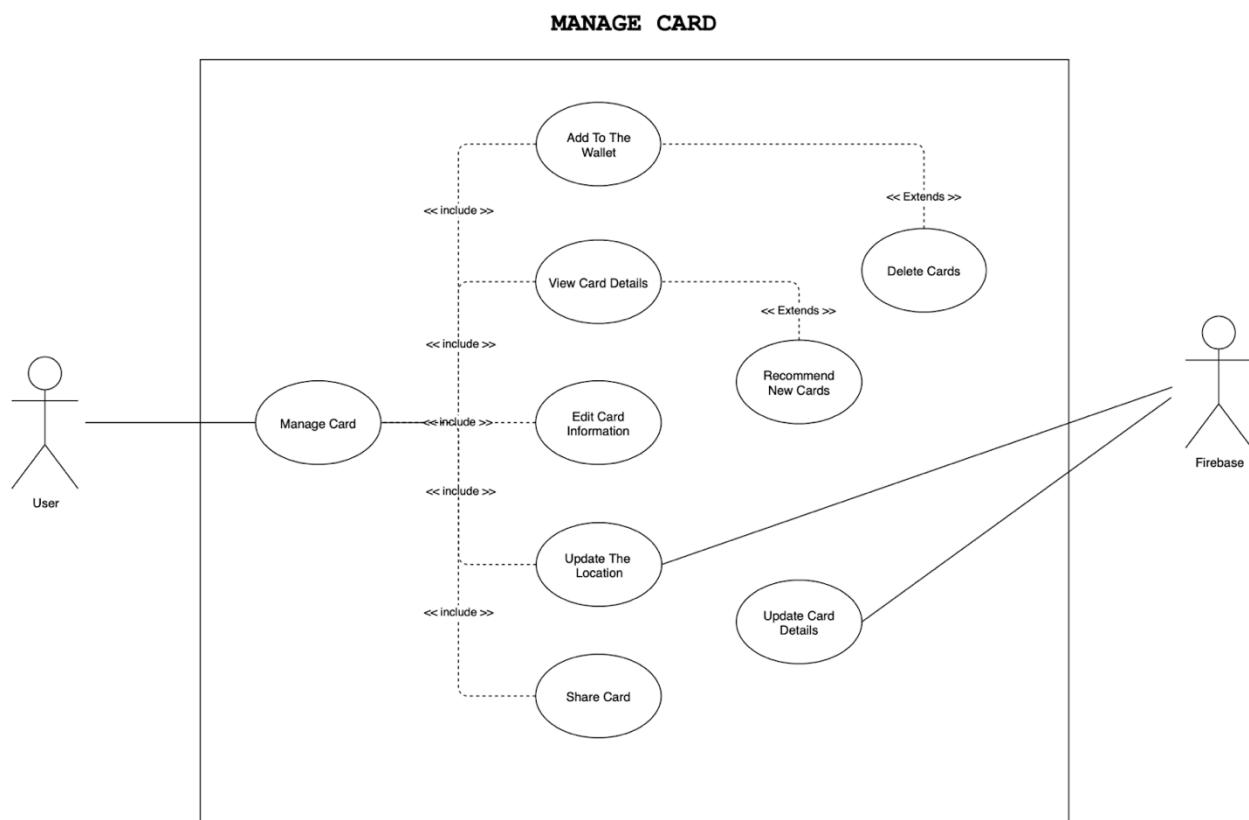


Figure 15. Manage card use case diagram

- This use case diagram for manage card shows the functionalities that the user can use to manage the cards on their wallet.
- The user has different functionalities about the business cards. Users can add, create, edit or remove a business card in their account.
- **Add to the wallet** – The user can add a new card into their wallet by searching for the card or scanning the code.
- **View card details** – Users can view the details on the business card.
- **Edit card information** – Users can update or change the information on their business card.
- **Update the location** – Users can use this functionality to change the location that they want their business card to appear on the map (address).
- **Share card** – Users can use this functionality to share the QR code of their business card.
- **Recommend new cards** – This functionality will show business cards close to the user location (if the user location is active).
- **Delete cards** – Users can use this functionality to remove a card from their wallet.
- **Update card details** – Users can use this functionality to change information on their business card such as address, phone number, email, etc.

10.2 ACTIVITY DIAGRAMS

OPEN APP (UR.1 -> SRC.2)

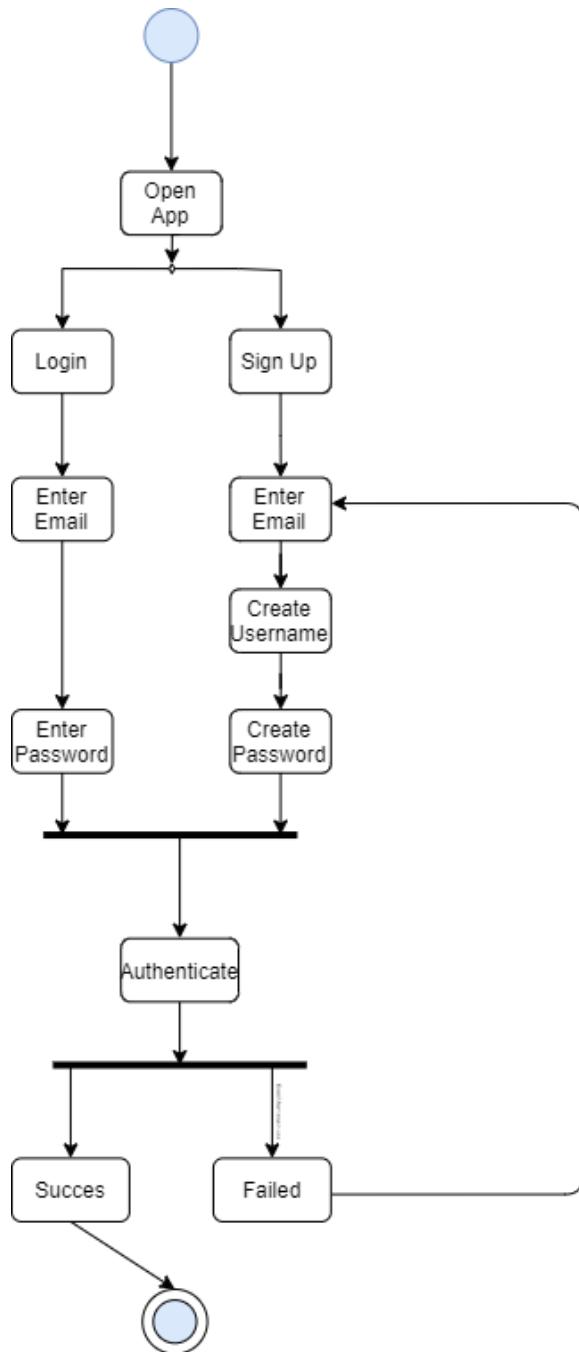


Figure 16. Open app activity diagram

- The activity diagram for open app represents how the users can access to their account or create a new account inside the iKard mobile app.
- Users can either login to an existing account with the correct email and password. Or users can create a new account with an email that it's not registered on the iKard database.
- Once the credentials are entered on the login view; firebase will verify the information and authenticate the user to access the account on the app. If the login information is incorrect the app will show an error message asking the user to verify the information they entered.

LOGIN (UR.3, UR.4, UR.5 -> SRC.1, SRC.2)

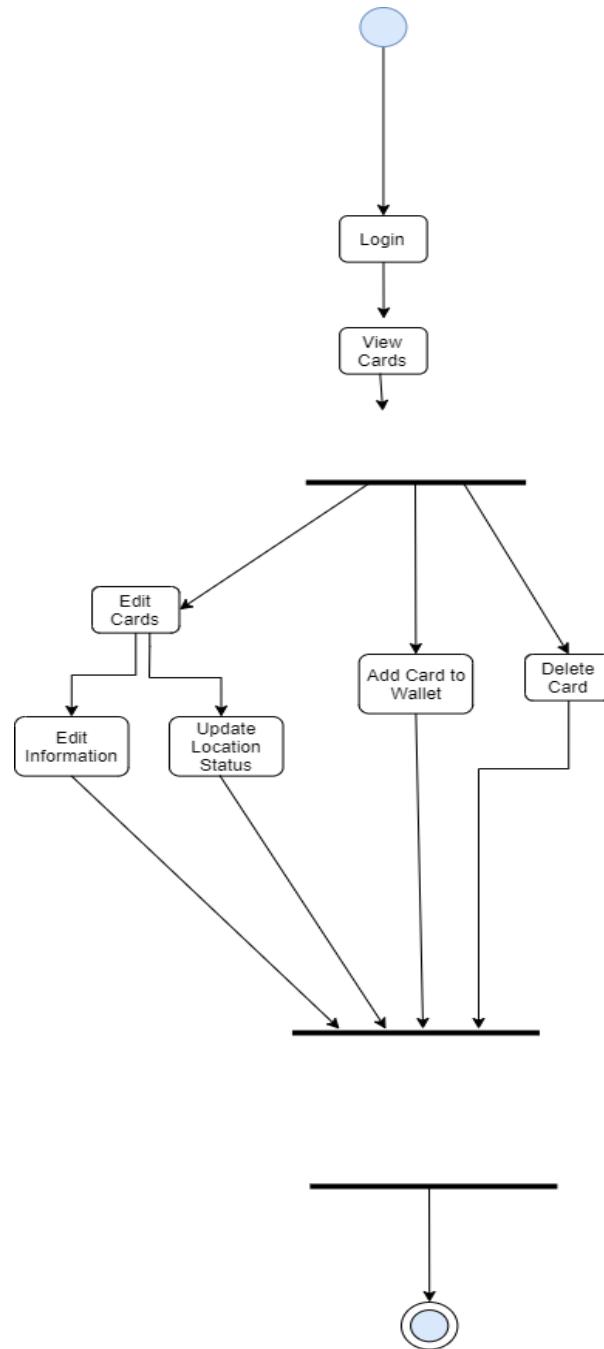


Figure 17. Login activity diagram

- The activity diagram for login represents how the user can see and manage the business cards.
- The user can edit their own the existing cards on their wallet. And when the user saves the card the card information will be updated on the database.
- The user can add a new card to the wallet by scanning the QR code or by searching the card information on the search bar. Also, users can delete an existing card on their wallet.

10.3 STATE DIAGRAM

RESET PASSWORD (UR.1 -> SRC.3)

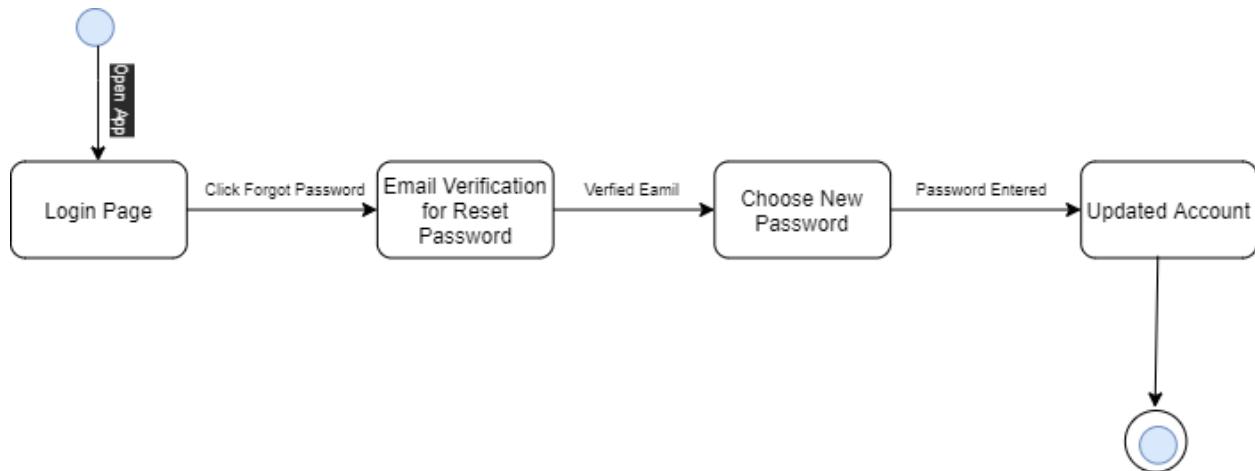


Figure 18. Reset password state diagram

- The reset password state diagram shows the state of the reset password process.
- The initial state of users is opened the app, from there the user need to press on the forgot password button. The user will be redirected to the forgot password view.

- On the forgot password view the user need to enter the email of the account they want to reset the password. Firebase will verify if the email address belongs to an existing account. An email address will be sent to that email for the user to create the new password for that account.
- Once the new password is created the user can go back to the app and login with the email and the new password they just created.

10.4 CLASS DIAGRAM

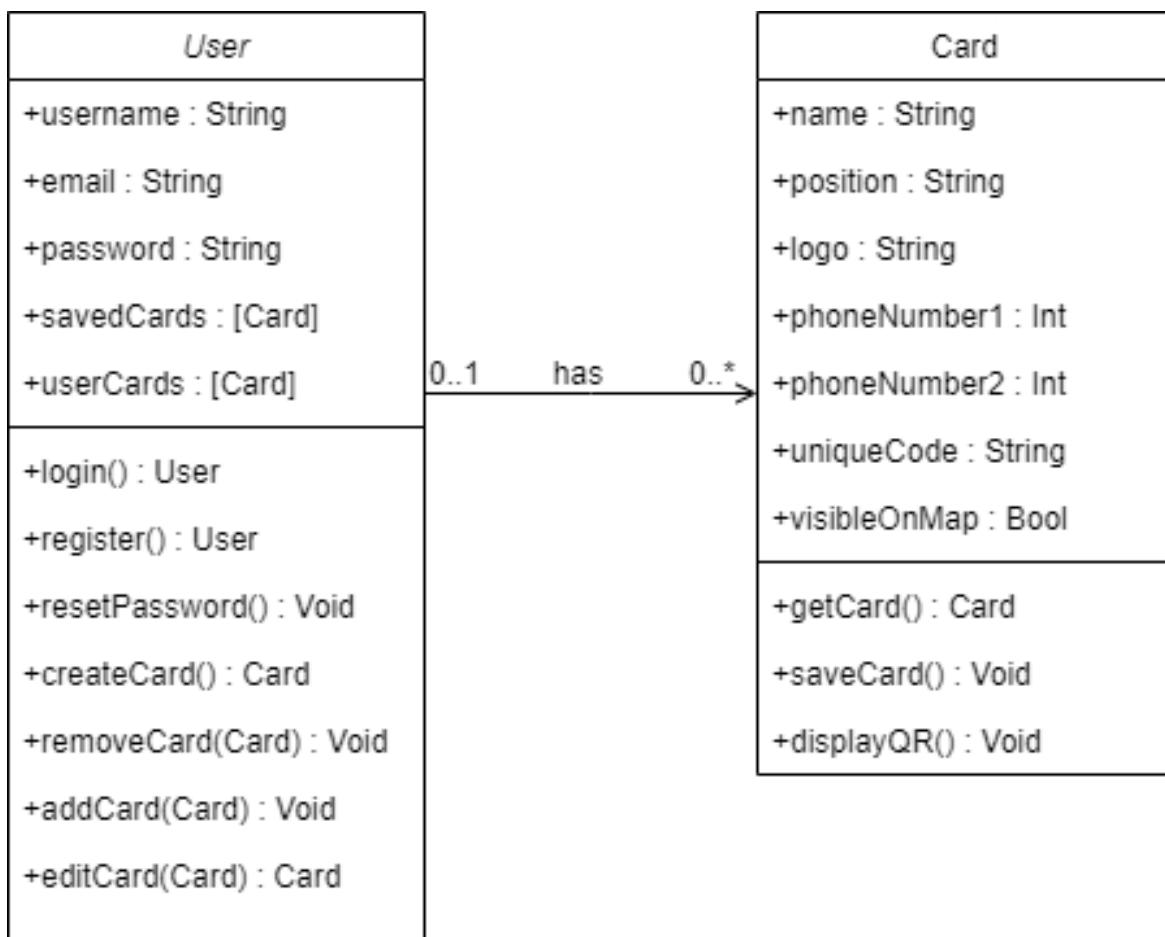


Figure 19. User and card class diagram

- Users can create a new account and multiple business card. Each user has a unique username and email address.
- Each business card created by the user will have a unique code that the user can share with other users.
- All business cards will contain the name of the user, position in the company, a logo image, main phone number and an optional secondary phone number, and the user will have the option to enable or disable the geolocation for that specific card.

10.5 SEQUENCE DIAGRAMS

REGISTRATION (UR.1 -> SRC.3)

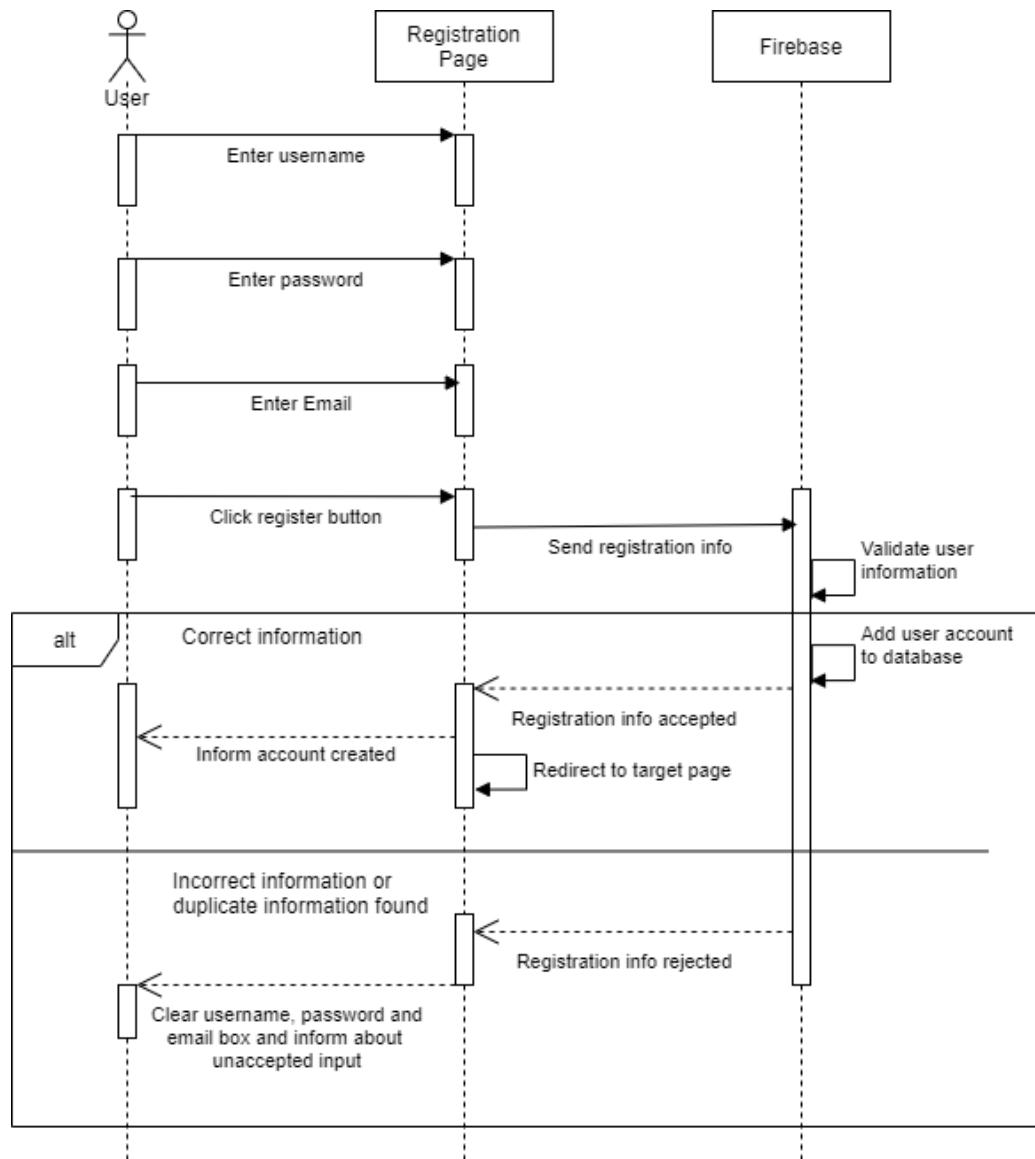


Figure 20. Registration sequence diagram

- To create an account in the application, the user needs to enter the credentials for account creation such as username, password, and email.
- Once the user has entered the required information, the user then proceeds by pressing the register button and the registration request will be passed to the Firebase to validate the information.
- If the registration information is correct and not duplicate of an existing account, the Firebase will add the user's account to the database and inform the user account created and redirect to the account page.
- If the registration information is incorrect or duplicate information is found in the database, then the Firebase will reject the process and inform the user of appropriate error.

LOGIN (UR.1 -> SRC.2)

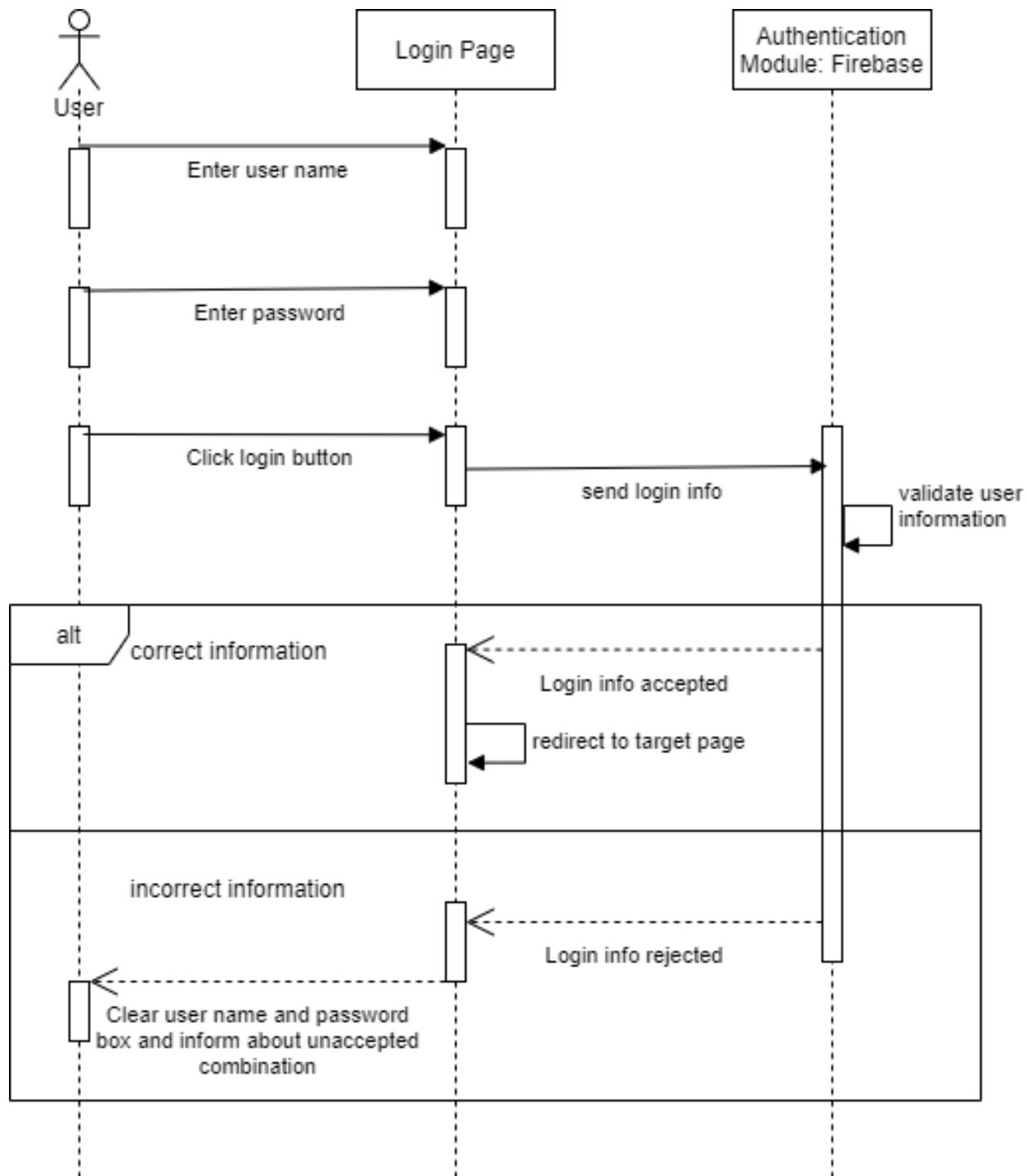


Figure 21. Login sequence diagram

- To login in the user's account, the user needs to enter username and password then proceeds by pressing the login button, which will send the login information to the Firebase, where the Firebase will validate the information of the existing account.
- If the information is entered correctly, Firebase will accept the information and redirect the user to its account page.
- On the other hand, if the information is not entered correctly, the Firebase will reject the login process and will clear the login boxes and inform about the unacceptable combination.

RESET PASSWORD (UR.1 -> SRC.1)

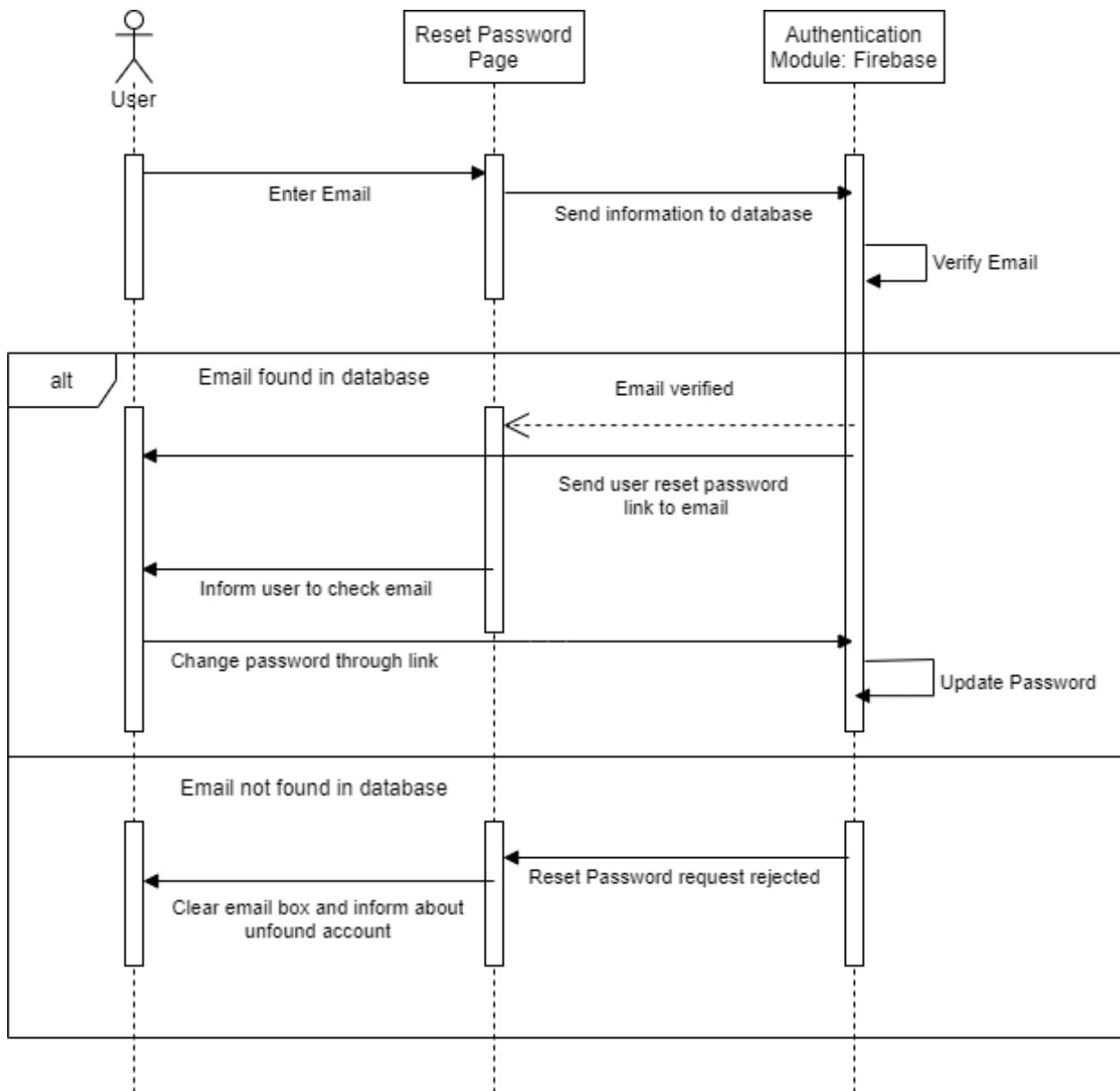


Figure 22. Reset password sequence diagram

- To reset the password of an account, the user needs to enter its email in the reset password page, then the information will be sent to the database to verify if there is an existing account with the email submitted.
- If an account is found with the email, the Firebase will send a reset password link to the user's email and inform the user to change through the link. After the user has entered a new password via the link, the Firebase will update the password accordingly.
- If an account is not found associated with the email, the Firebase will reject the request and inform the user that no account has been found with the email.

CREATE CARD (UR.2 -> SRC.1)

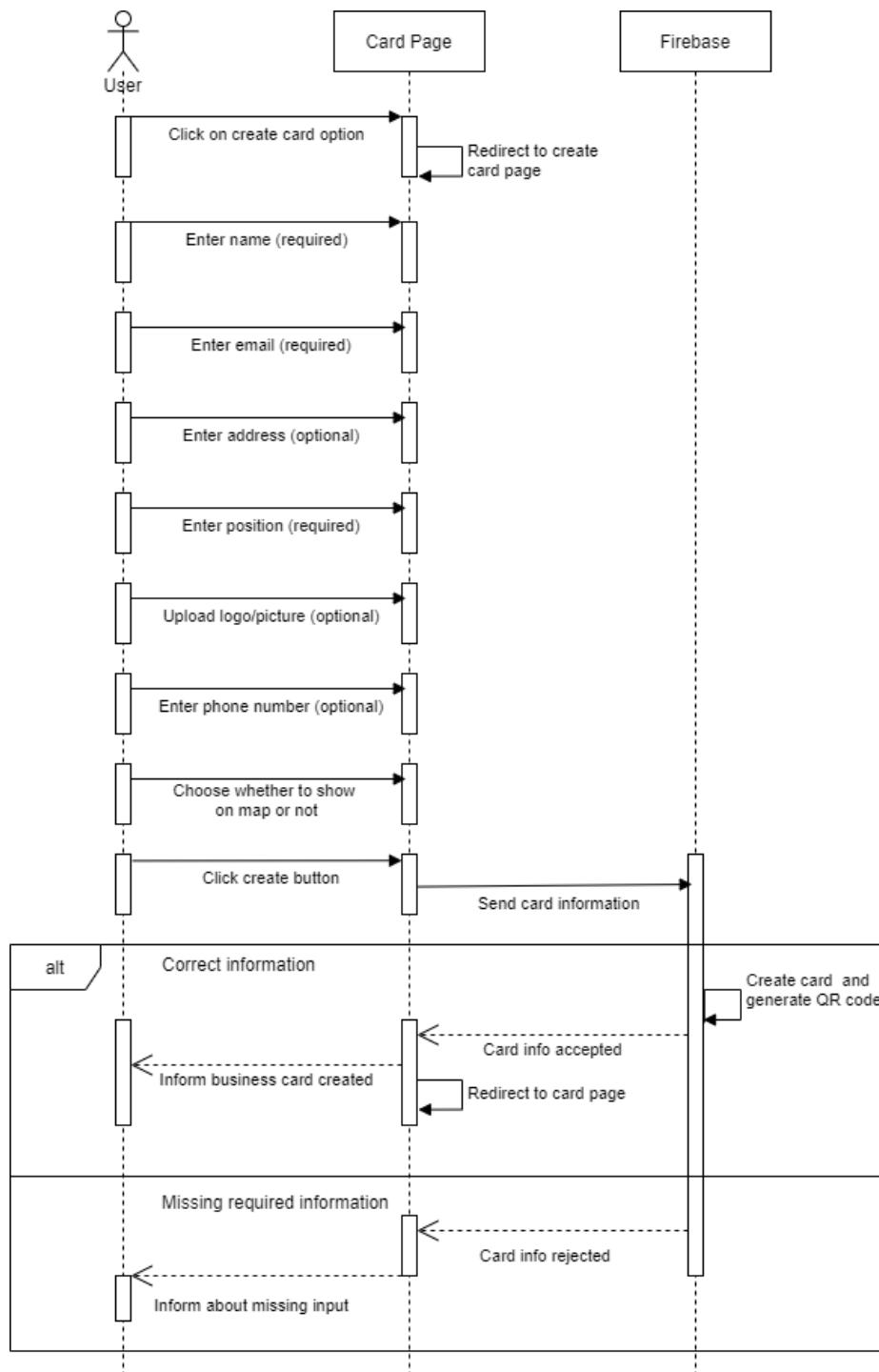


Figure 23. Create card sequence diagram

- To create a business card in the application, the user needs to click on the create card option in the card page, then the user needs to enter all the required information to create a virtual card such as name, email, position, etc. By clicking the create card button, this information will be sent to the database and Firebase will verify the information.
- If all the required information is entered, then Firebase will create the card in the database and generate a QR code associated with the card. If some of the required information is missing, then Firebase will reject card creation and inform about missing information.

ADD CARD (UR.3 -> SRC.1)

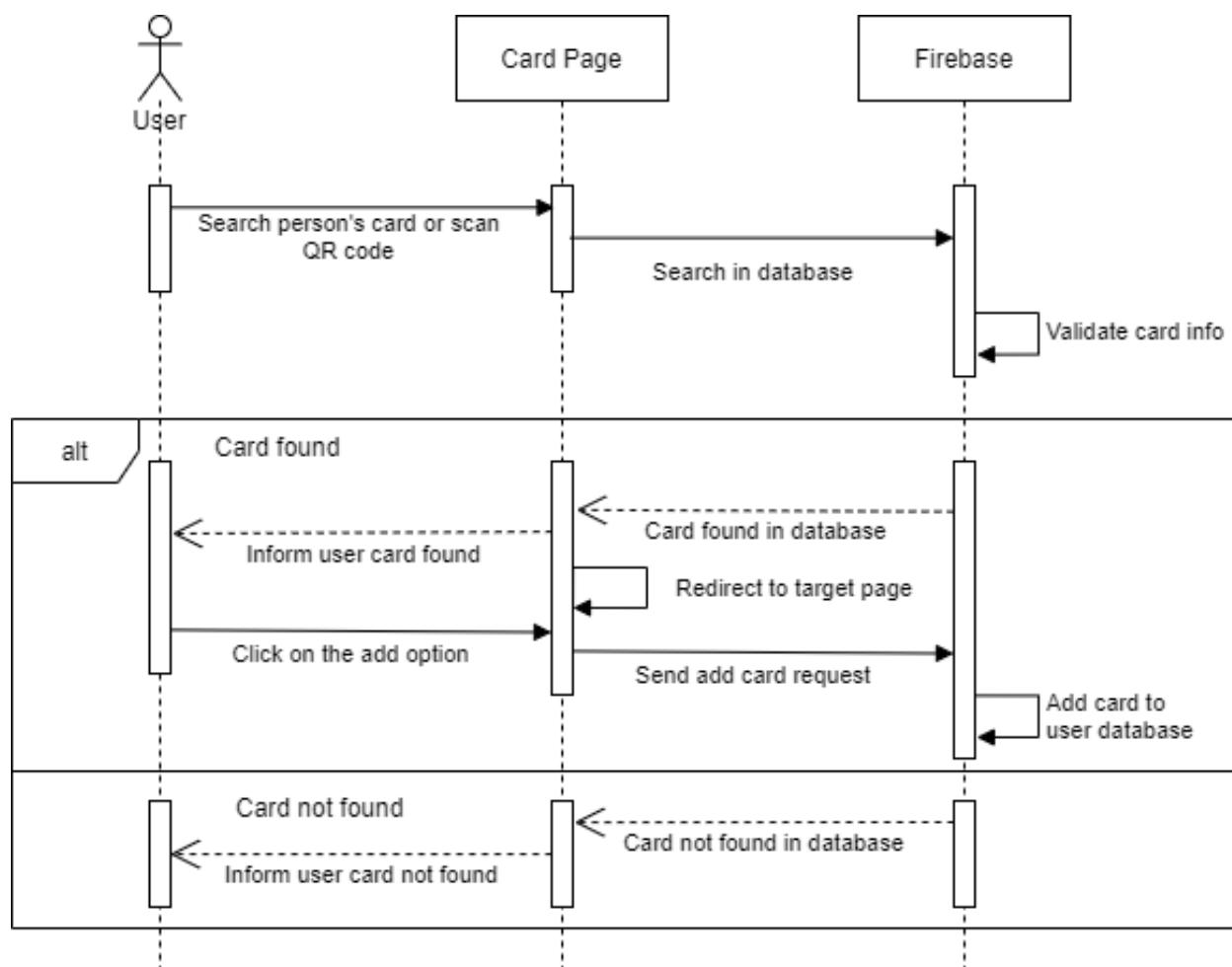


Figure 24. Add card sequence diagram

- To add another user's card to a user's account, the user needs to search the person's name or scan the person's QR code in the card page, then Firebase will perform search and validation operations.
- If the card is found in the database, then Firebase will inform the user card found and display detailed information about the card by redirecting to the card page where the user can click on the add button, which will send a add card request to Firebase and Firebase will add the card to the user's account.
- If the card is not found in the database, then Firebase will inform the user card not found.

REMOVE CARD (UR.4 -> SRC.1)

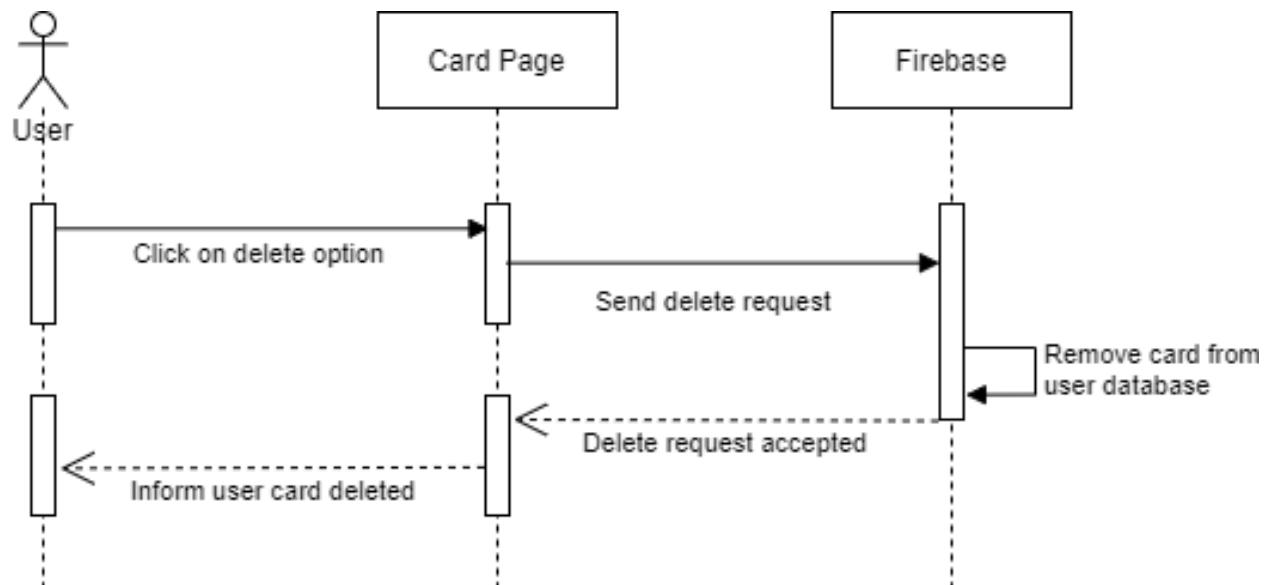


Figure 25. Remove card sequence diagram

- To remove a card from an user's account, the user simply needs to click on the delete option in the card page and choose which card to remove, then it will send a delete request to Firebase.
- After successfully removing the card from the user's account, the user will be informed that the card has been removed from the account.

EDIT CARD (UR.5 -> SRC.1)

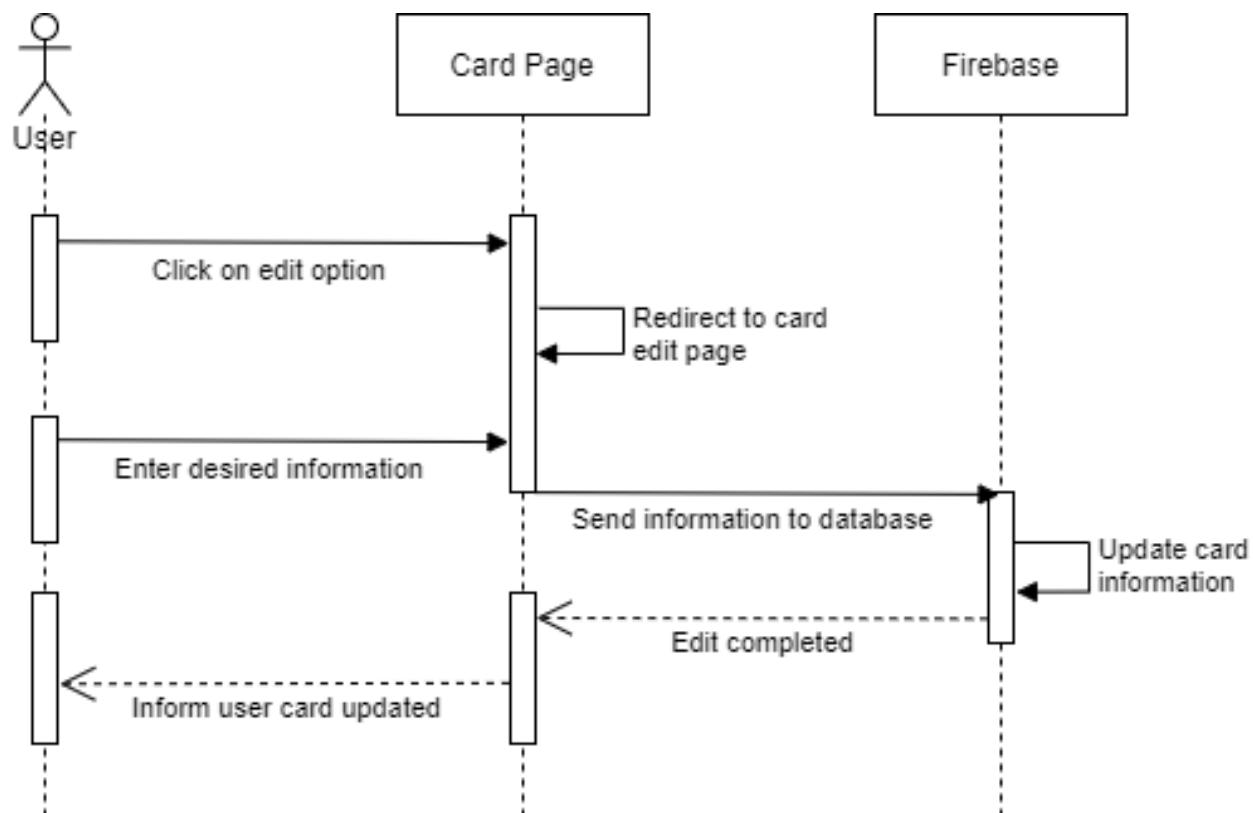


Figure 26. Edit card sequence diagram

- To edit a card's information, the user needs to choose which card he/she desires to change in the card page, and it will redirect the user to the card editing page, where the user can edit desired information. Then the new information will be sent to the database, where Firebase will update the card information in the database.
- After finishing updating the card information, the user will get a notification that the card information has been updated.

10.6 SYSTEM DIAGRAM

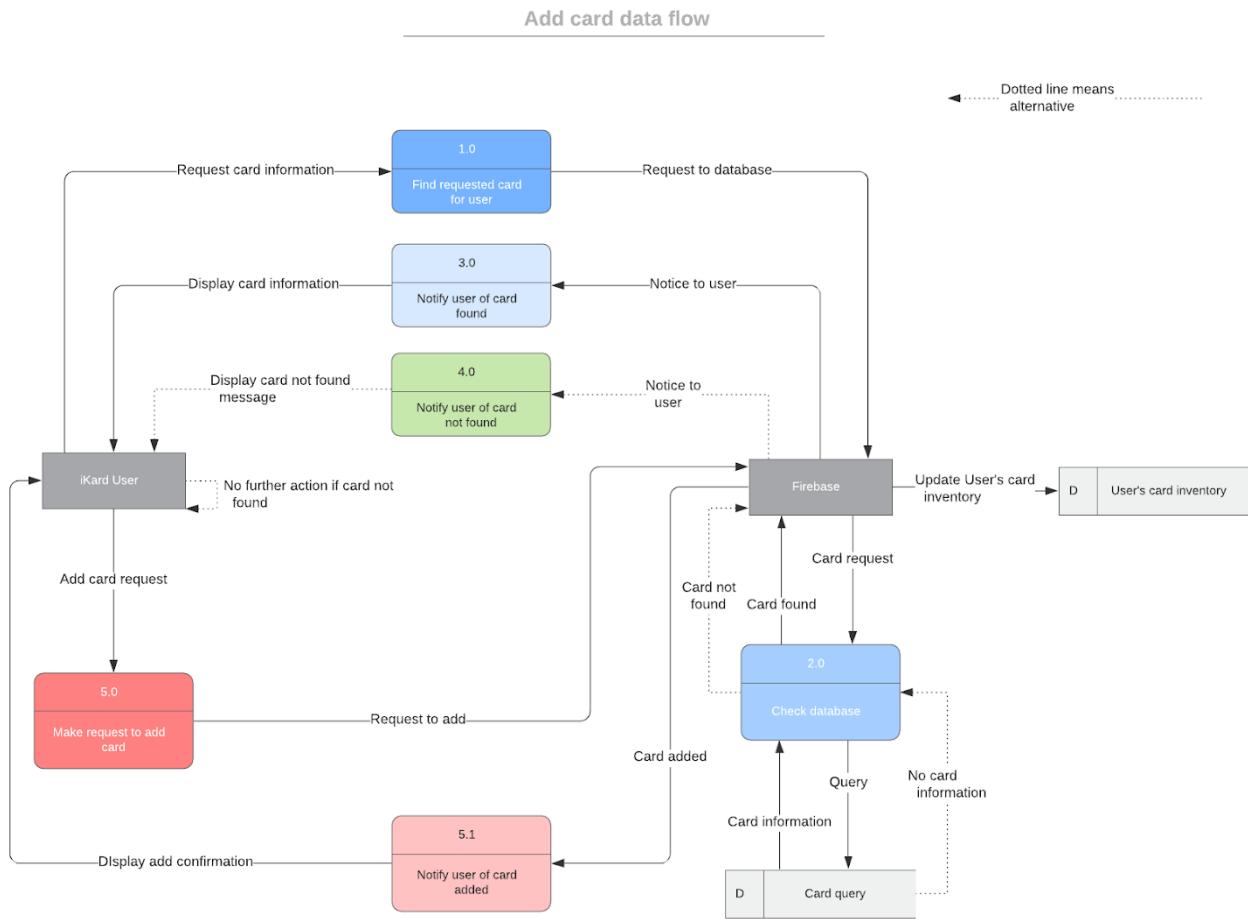


Figure 27. Data flow system diagram

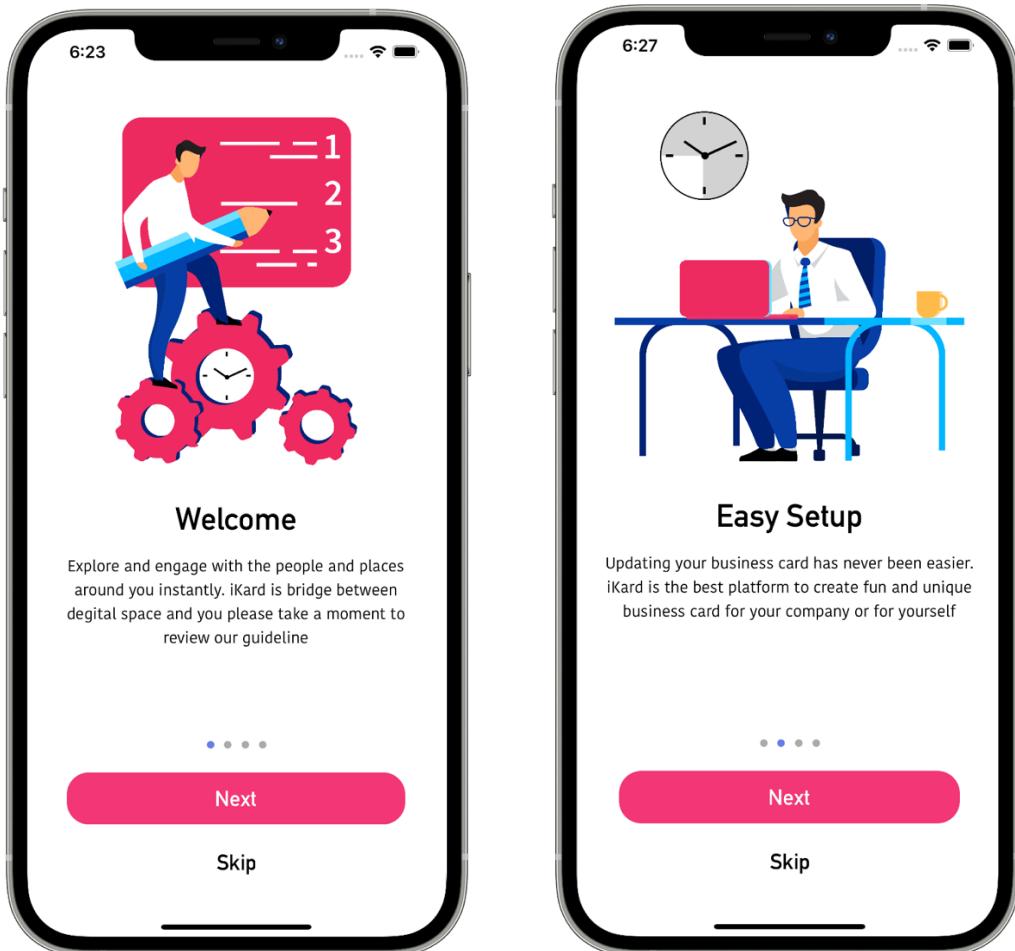
- This system diagram represents the add card data flow.
- Users can add, edit, remove or create a new business card inside the app main interface. If a card is not found when the user is trying to add it will display an error message to the user.
- Users can update the information of any of their cards and it will automatically update the card information for all the users that have

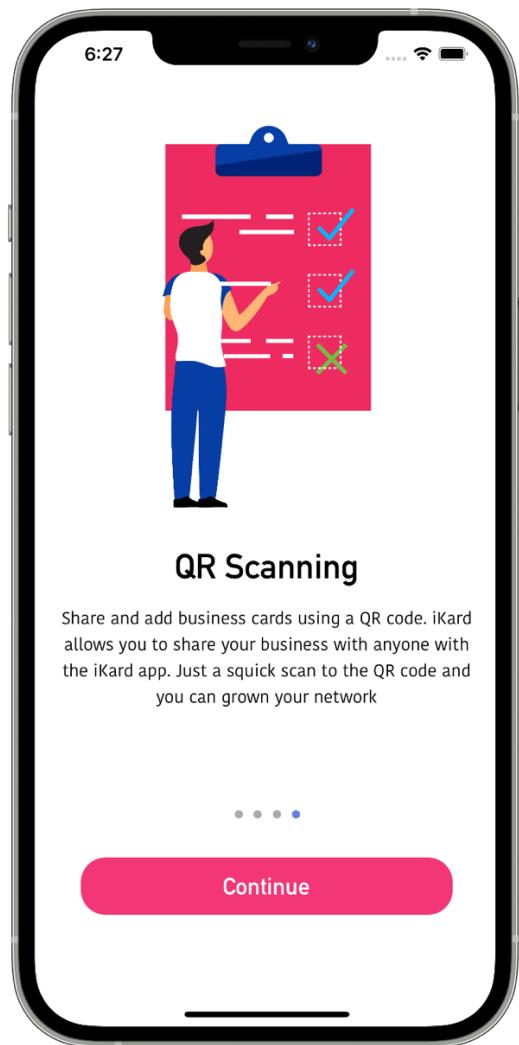
that business card on their wallet. All the users will be able to see in real time the most recent information of a user business card.

FINAL APP

1. ONBOARDING IMPLEMENTATION

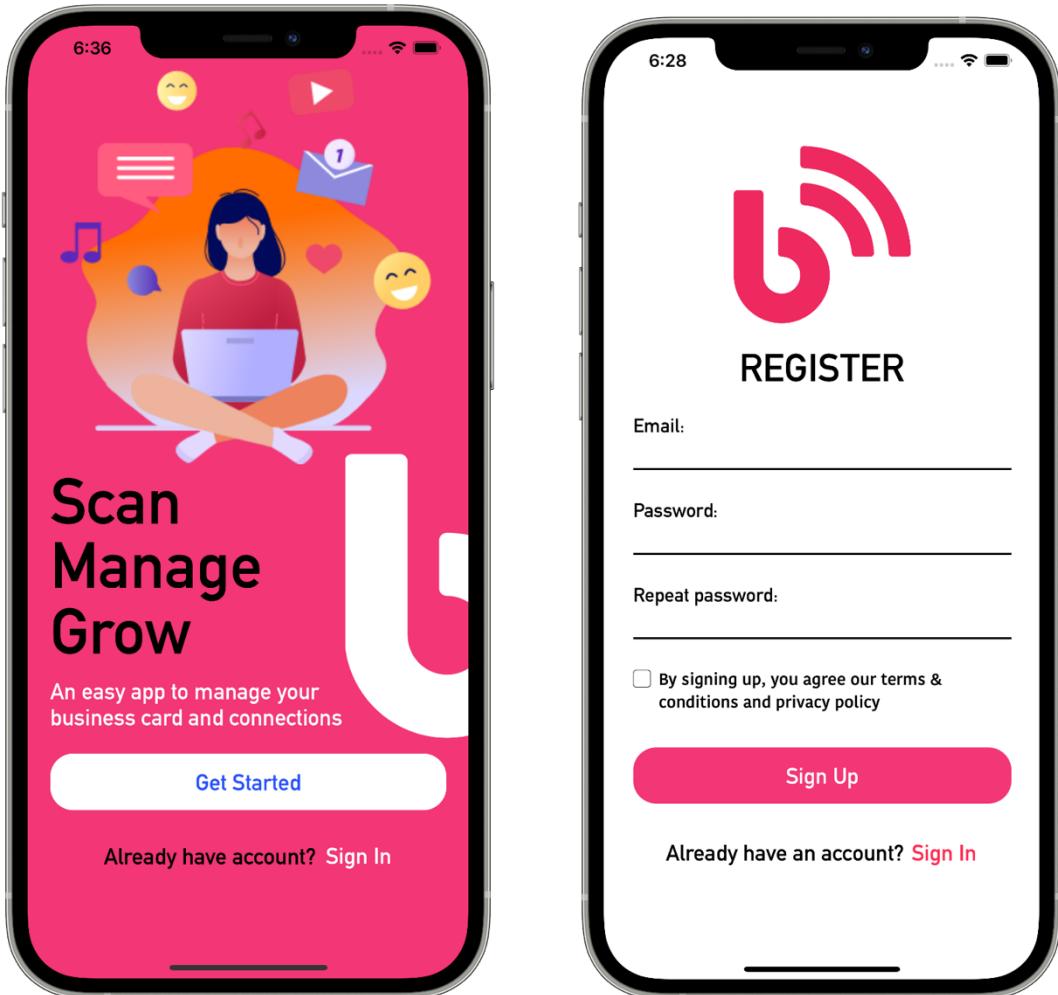
The onboarding screen of the application allows the user to learn about the app and the features of the application. Users can go through the entire onboarding or skip the onboarding section to go directly to the sign-up screen of the application.

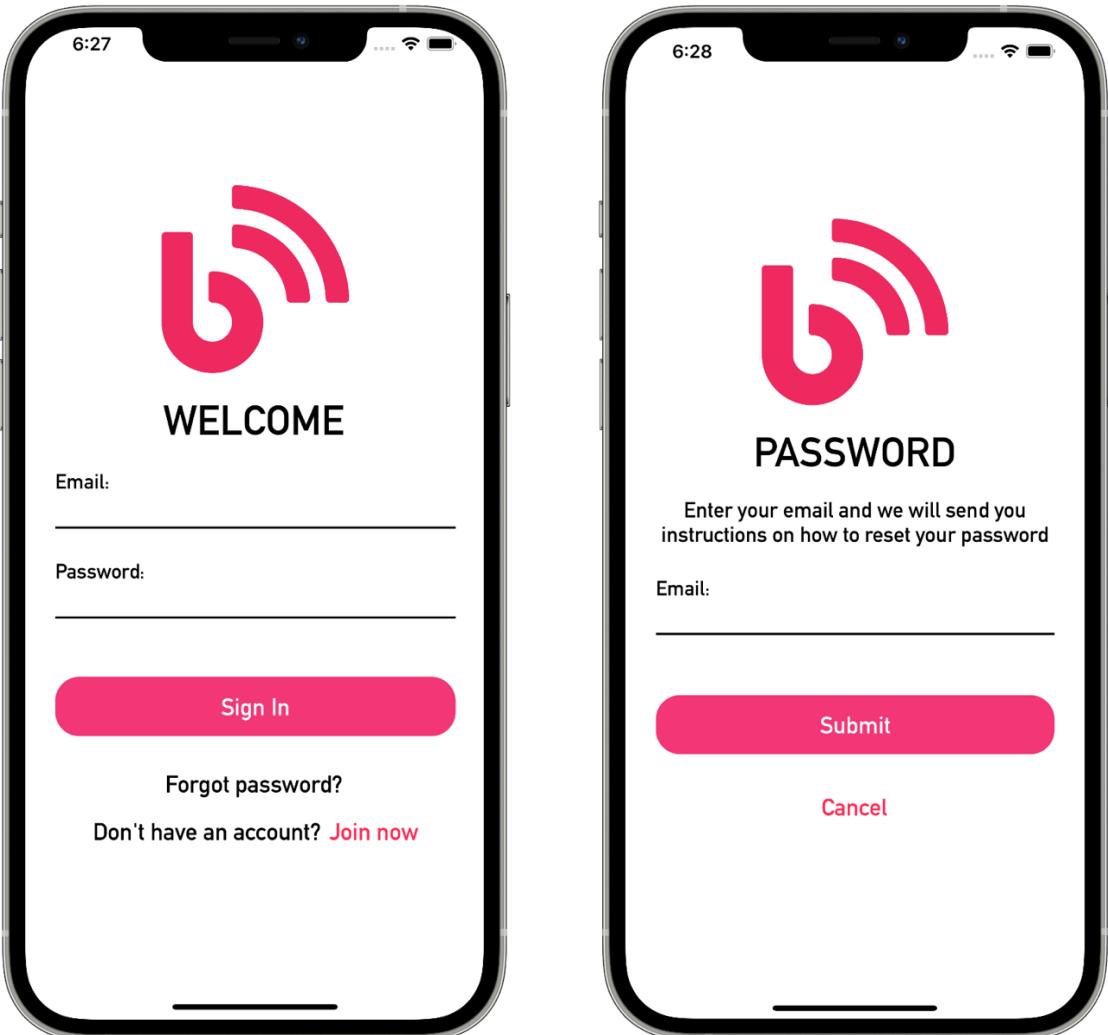




2. REGISTRATION/LOGIN IMPLEMENTATION

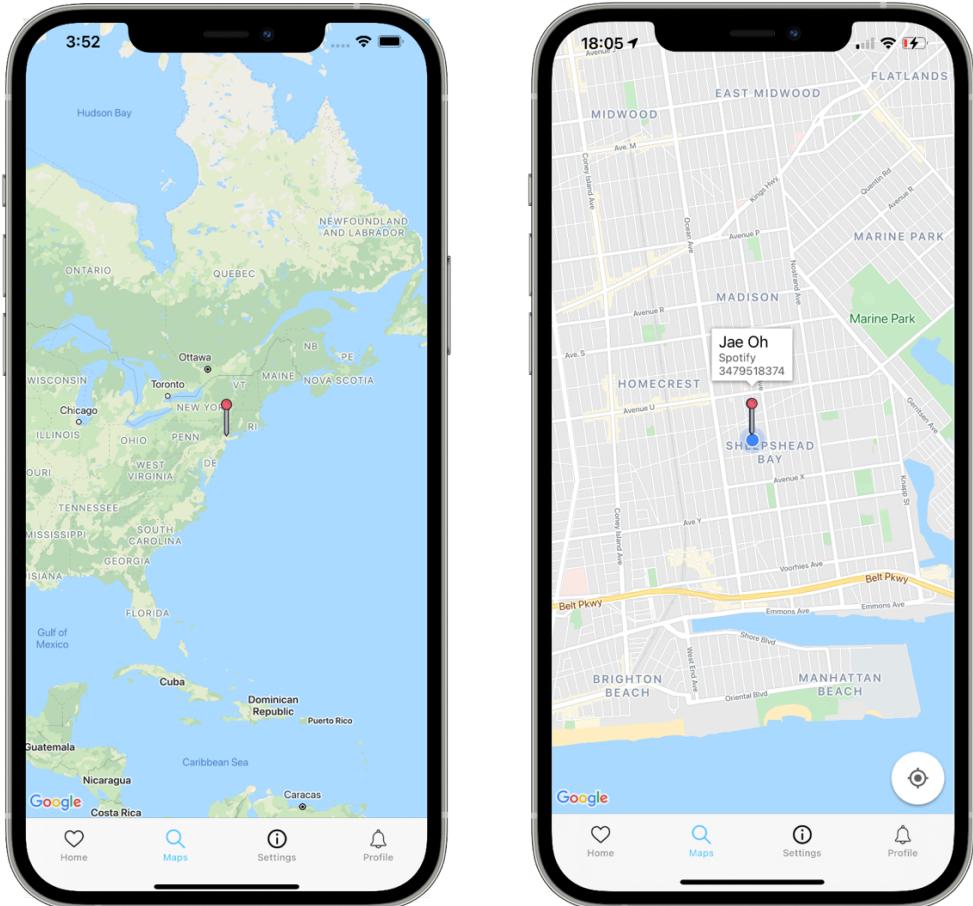
On the registration screen, users can register for a new account using email and password. Users cannot sign for two accounts with the same email. If the email is already in used, the app will let the user know the email is already registered. Returned users can log in to their account using email and password. If the user doesn't remember the password, the user can reset the password on the reset view. The user will receive an email with the steps to reset their account password.





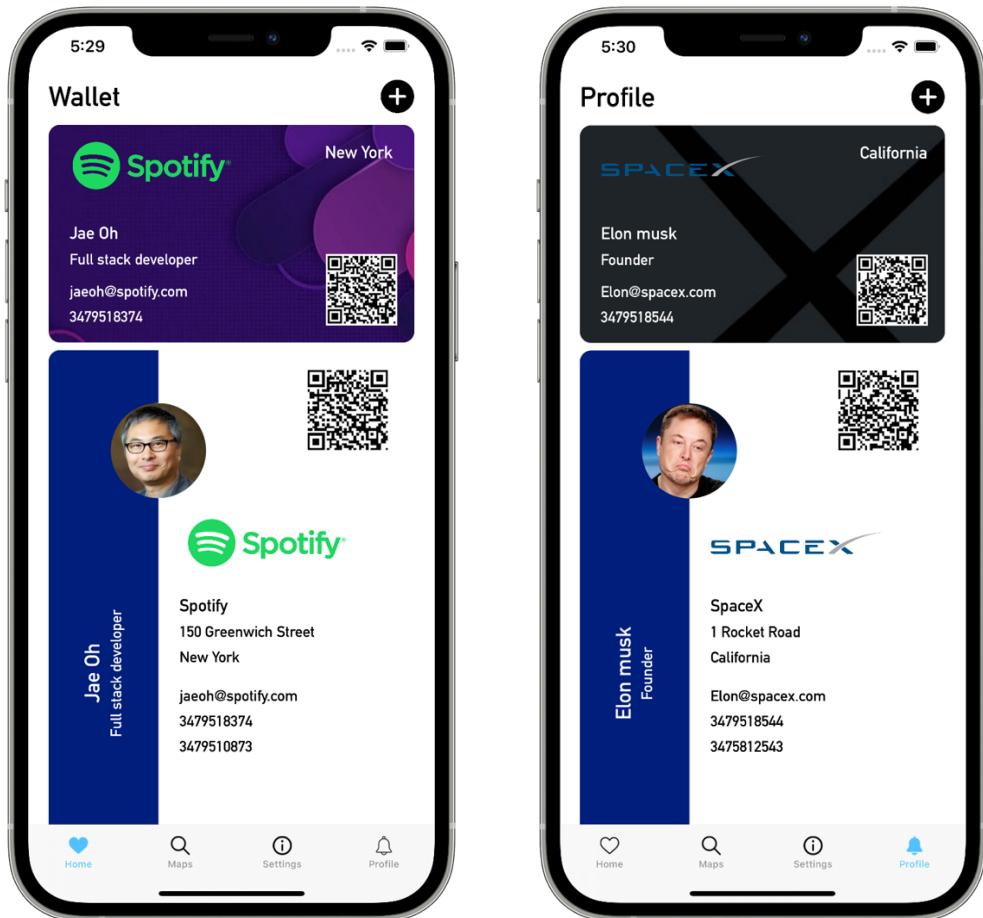
3. MAP/LOCATION IMPLEMENTATION

Users will be able go on the map within the app to find other business cards in the location to look for other professionals and promote their own business or service. User can see their business card contact from the wallet inside the map view on the app. Currently, users can only see one card in the exact location. But users can tap on the location to see the following business card contact on that location on the map.



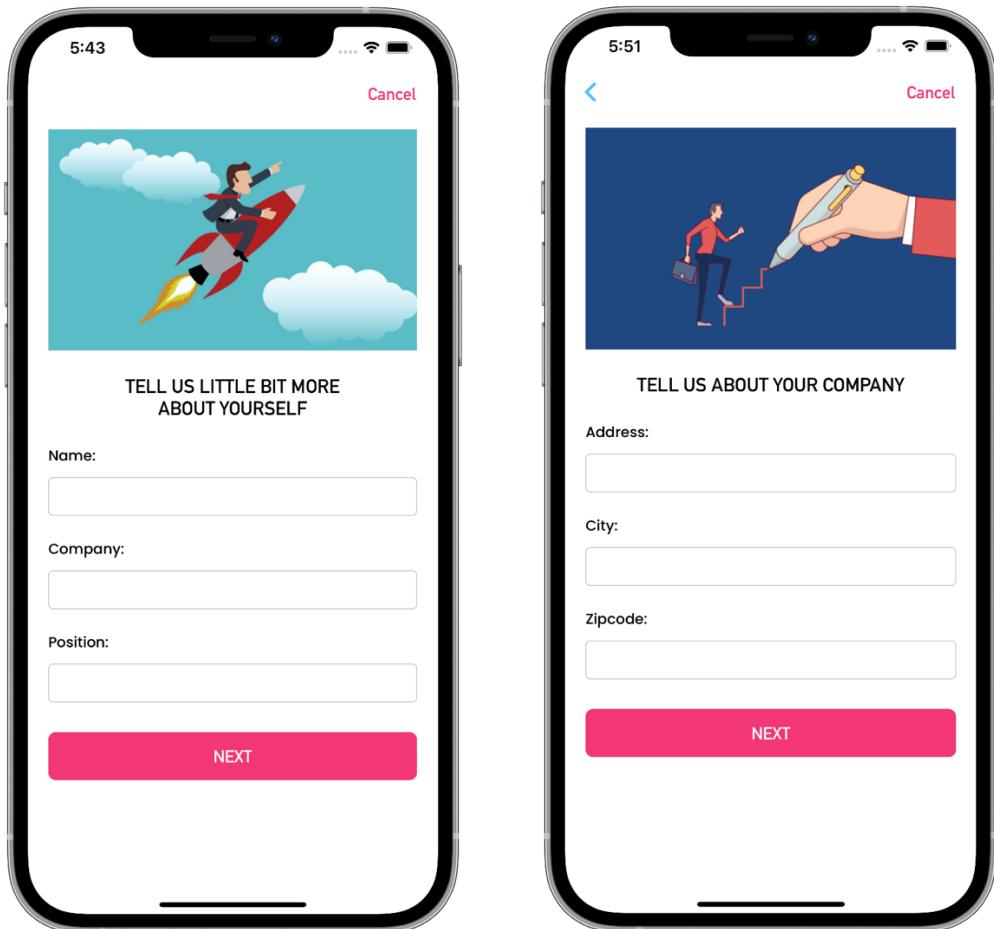
4. QR CODE IMPLEMENTATION

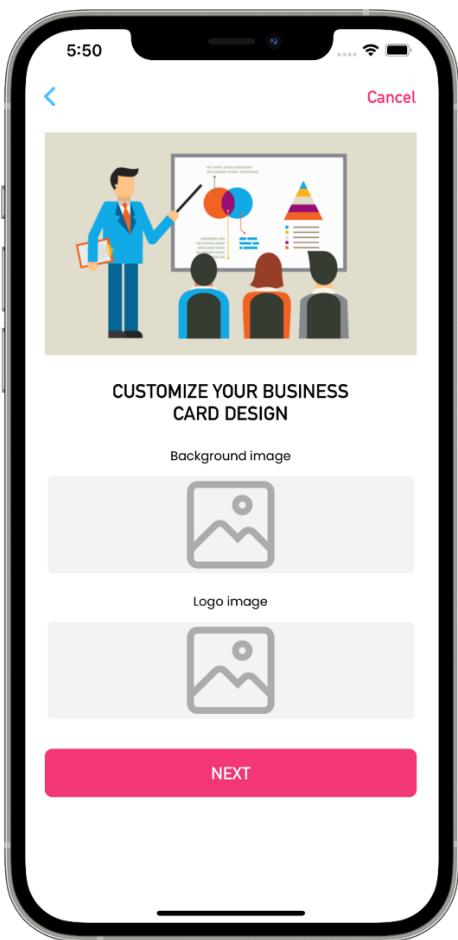
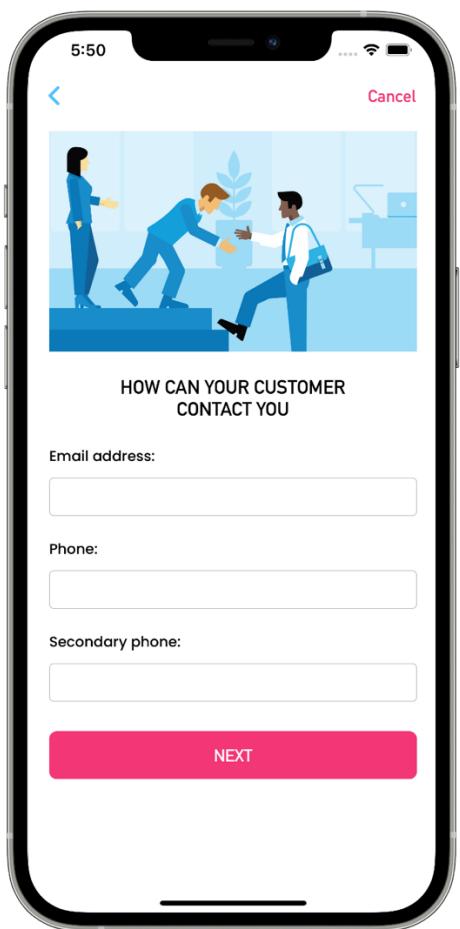
Each business card the user creates will generate a QR code associated with content of the business card. User can share their QR code to other users to scan and will automatically add the business card to wallet. It saves times for users to exchange business card and can be accessed online, where users can directly scan QR code shared on the internet and add to their wallet.

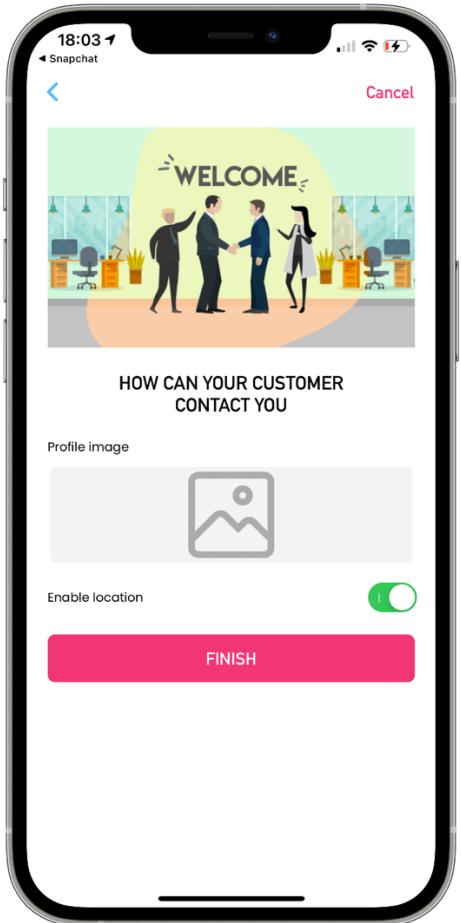


5. CREATE NEW BUSINESS CARD

A user can create as many cards as possible they want in their profile wallet view. A user can share the QR code with any user to scan the code and add their card to the wallet. A user can edit or remove the card, and it will automatically update the information so it will display the latest information on the user's wallet.





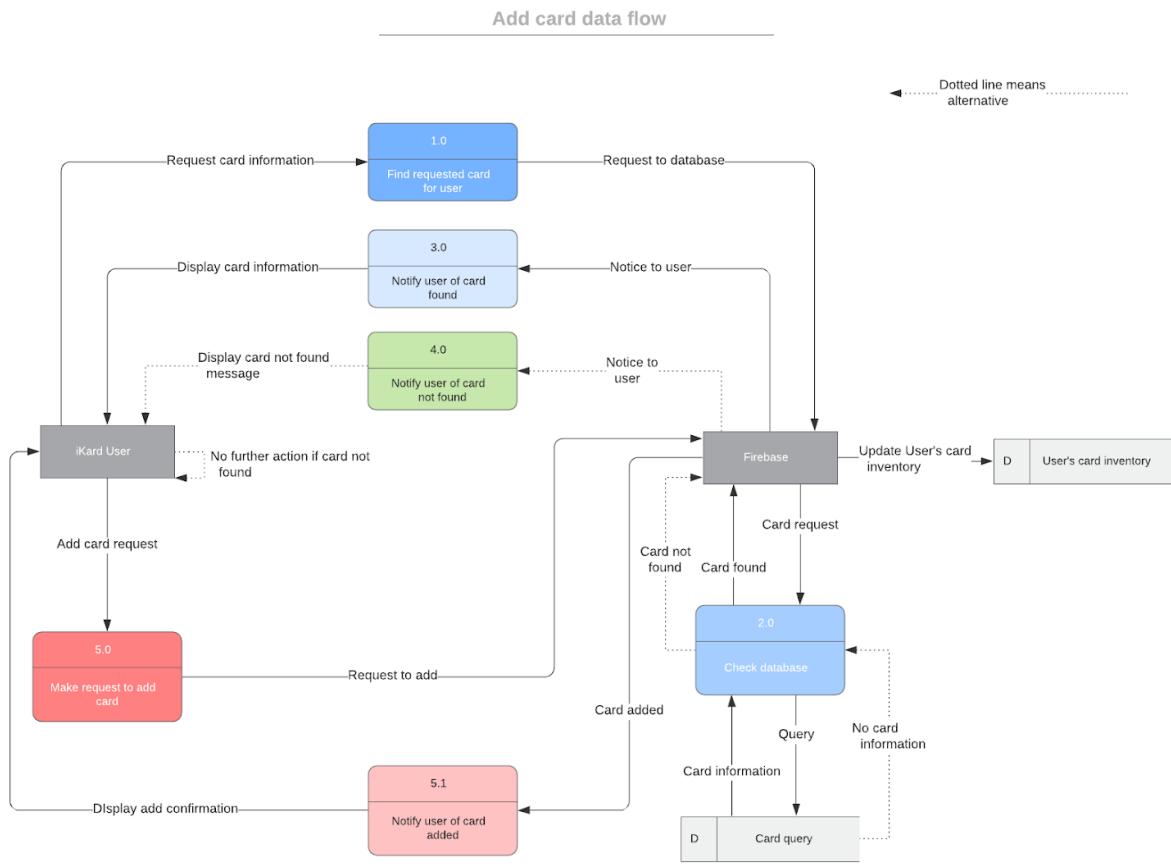


6. EDIT/REMOVE CARD IMPLEMENTATION

Currently, users can edit or remove their business cards from the account. If they want to edit it, the information will be updated on firebase, and every user who have that person business card will see the most recent information on the card



7. SYSTEM DIAGRAM



8. SYSTEM ARCHITECTURE

