# MyParking

Database used: Firebase

ER diagram link: https://drive.google.com/file/d/18EXLVrPvINUunm0UO69njjVWhpicZyWX/view?usp=sharing

Work division:

Front End Development: Rajdeep

=> Design of all the UI, screens, layout.

=> Validations.

=> use of DBHelper method to send & retrieve data to & from the Firebase.

Back End Development : Inzamam

=> Setting up Firebase, GoogleService-Info.plist file.

=> DBHelper file and setting up all the relevant functions as per the project requirement.

Update (18/05/2021):

1. Sign In Screen is completed with all the validations.

2. Sign Up Screen is completed with all the validations.

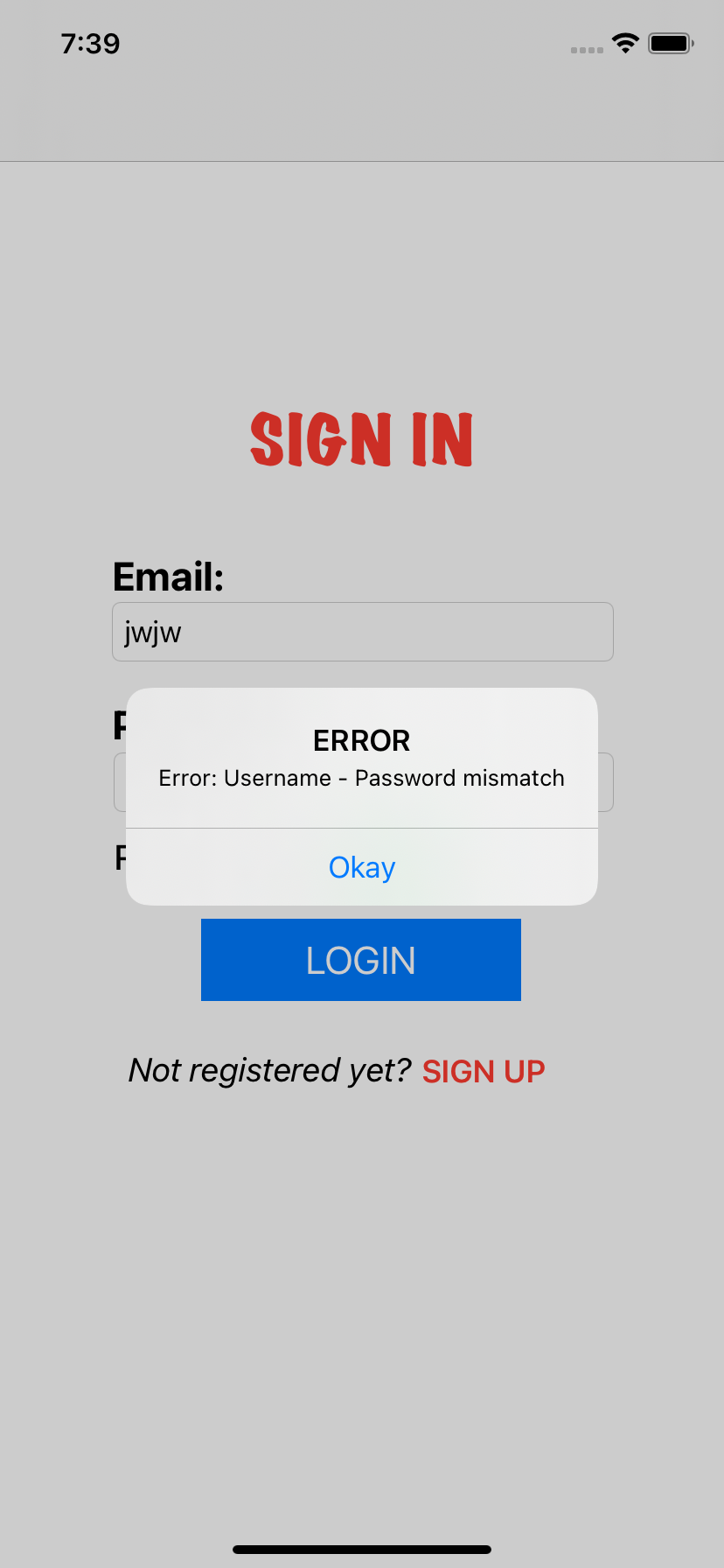
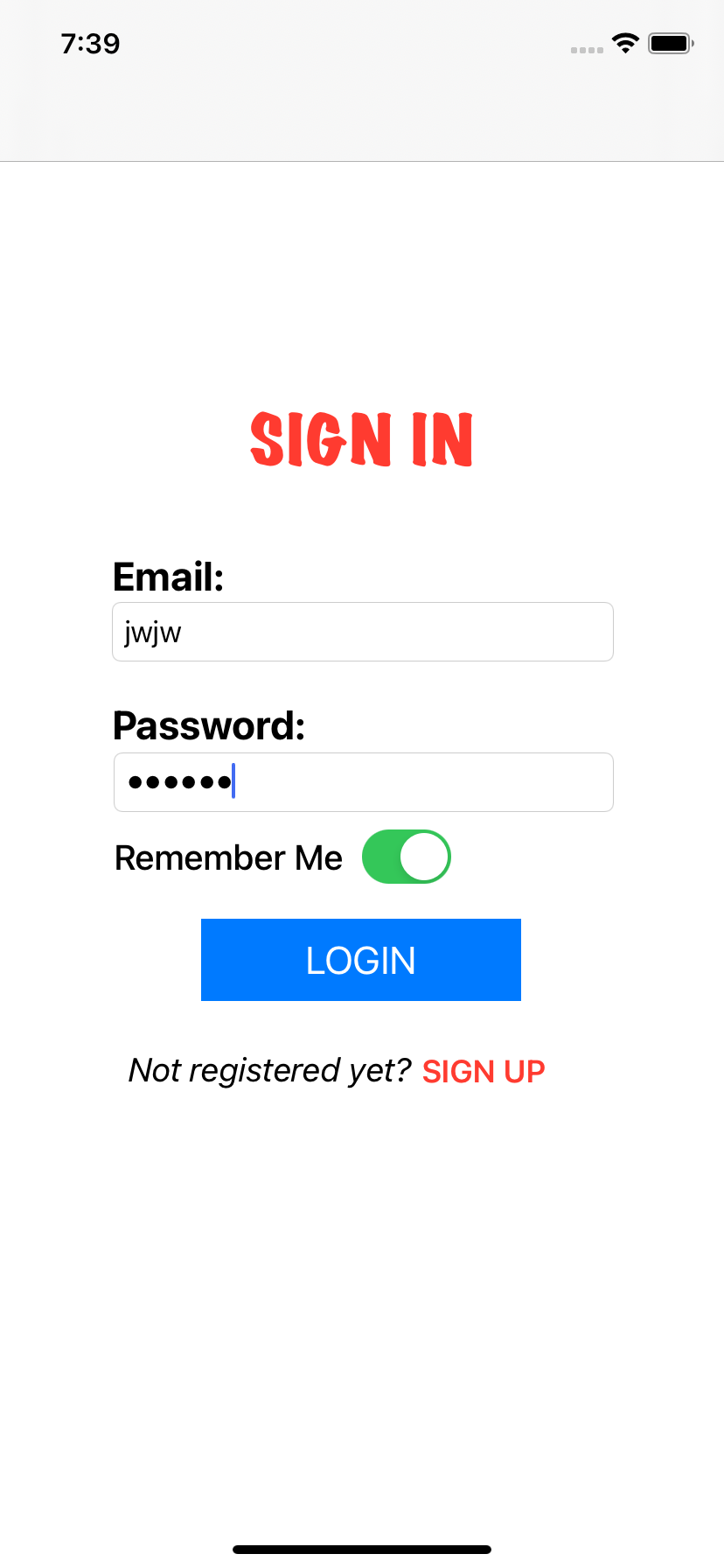
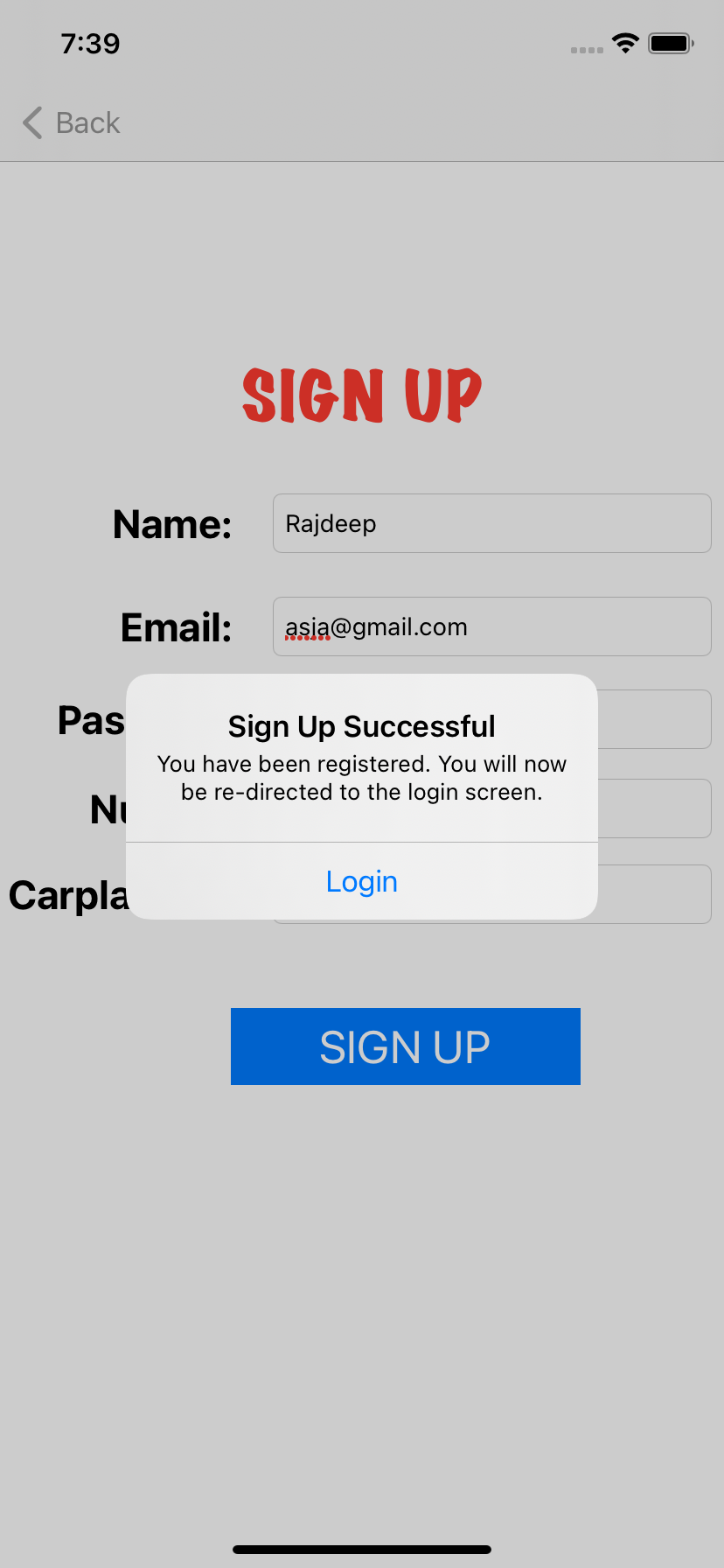
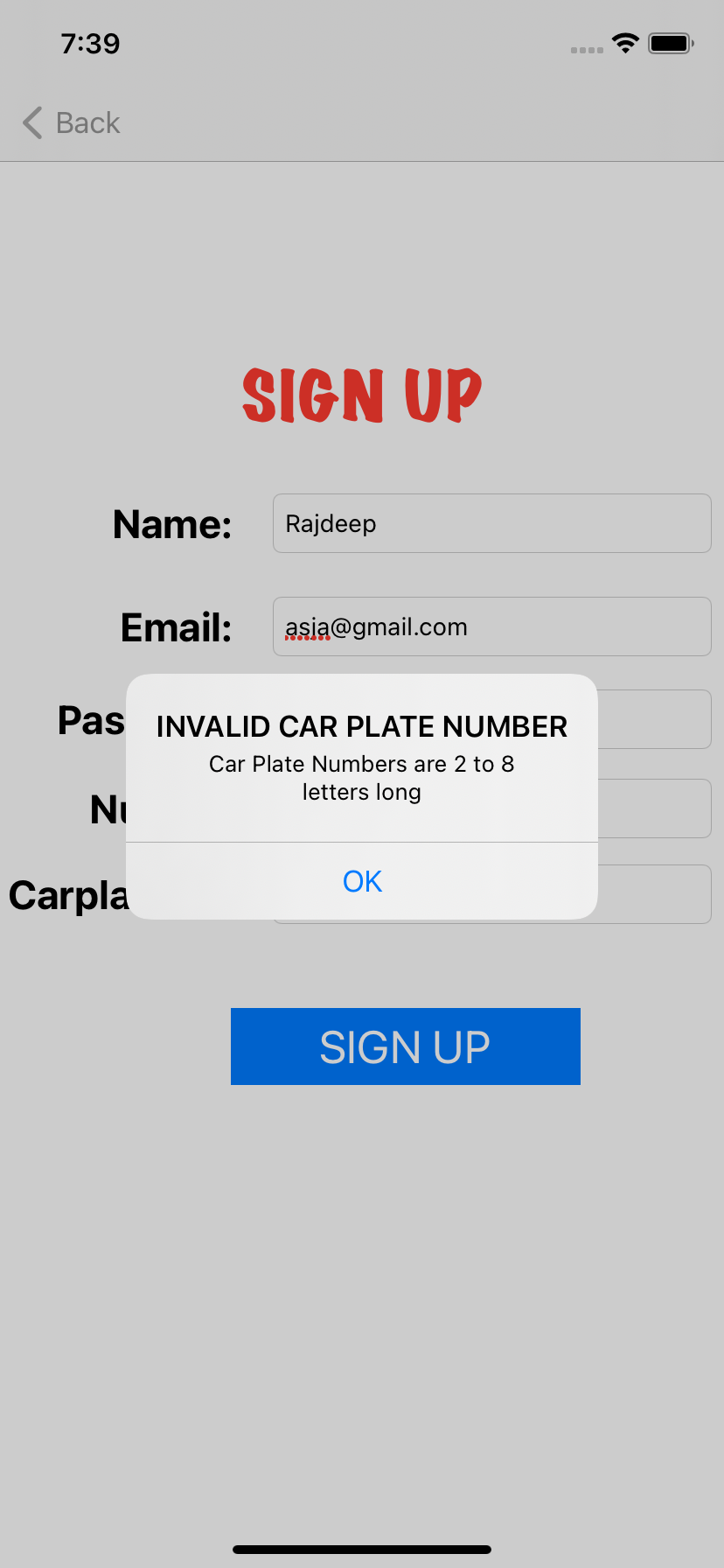
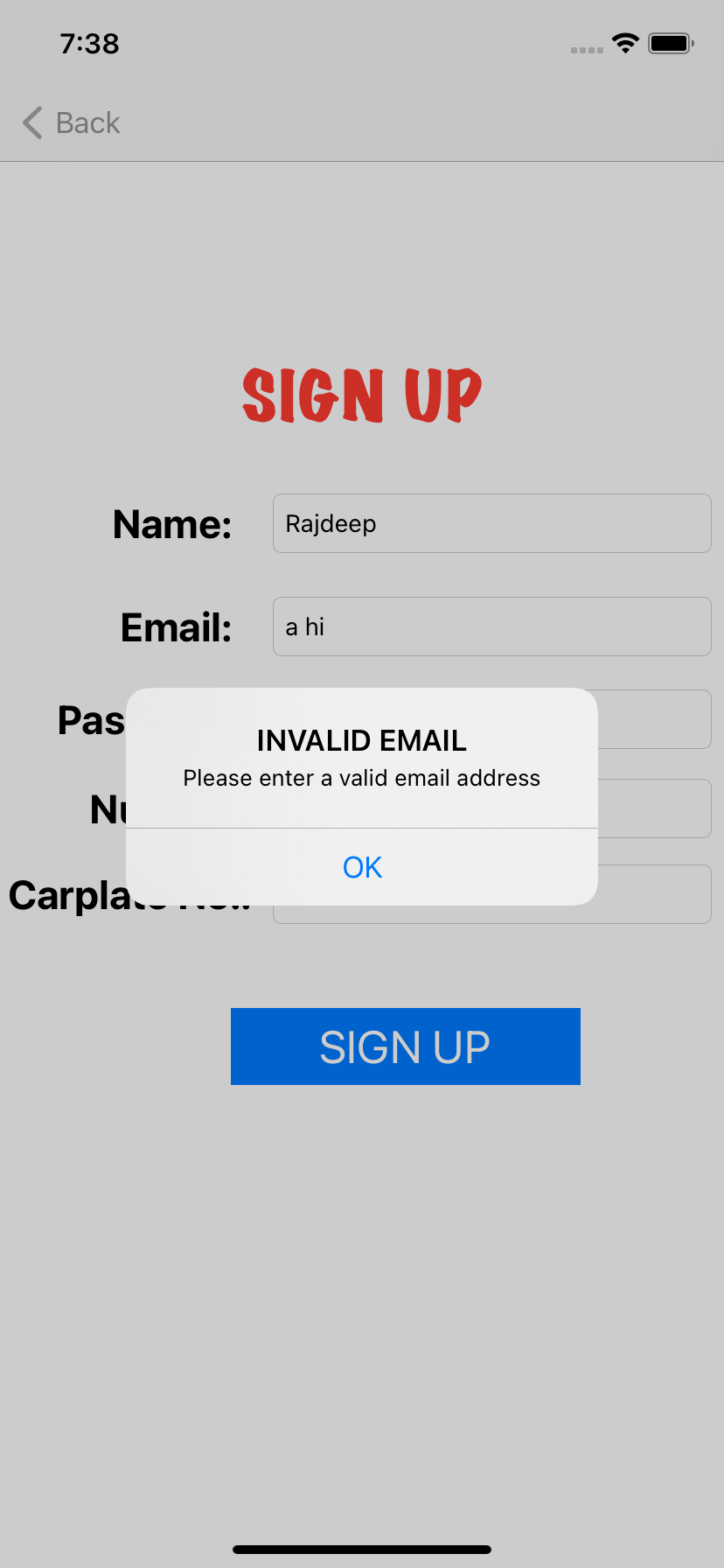
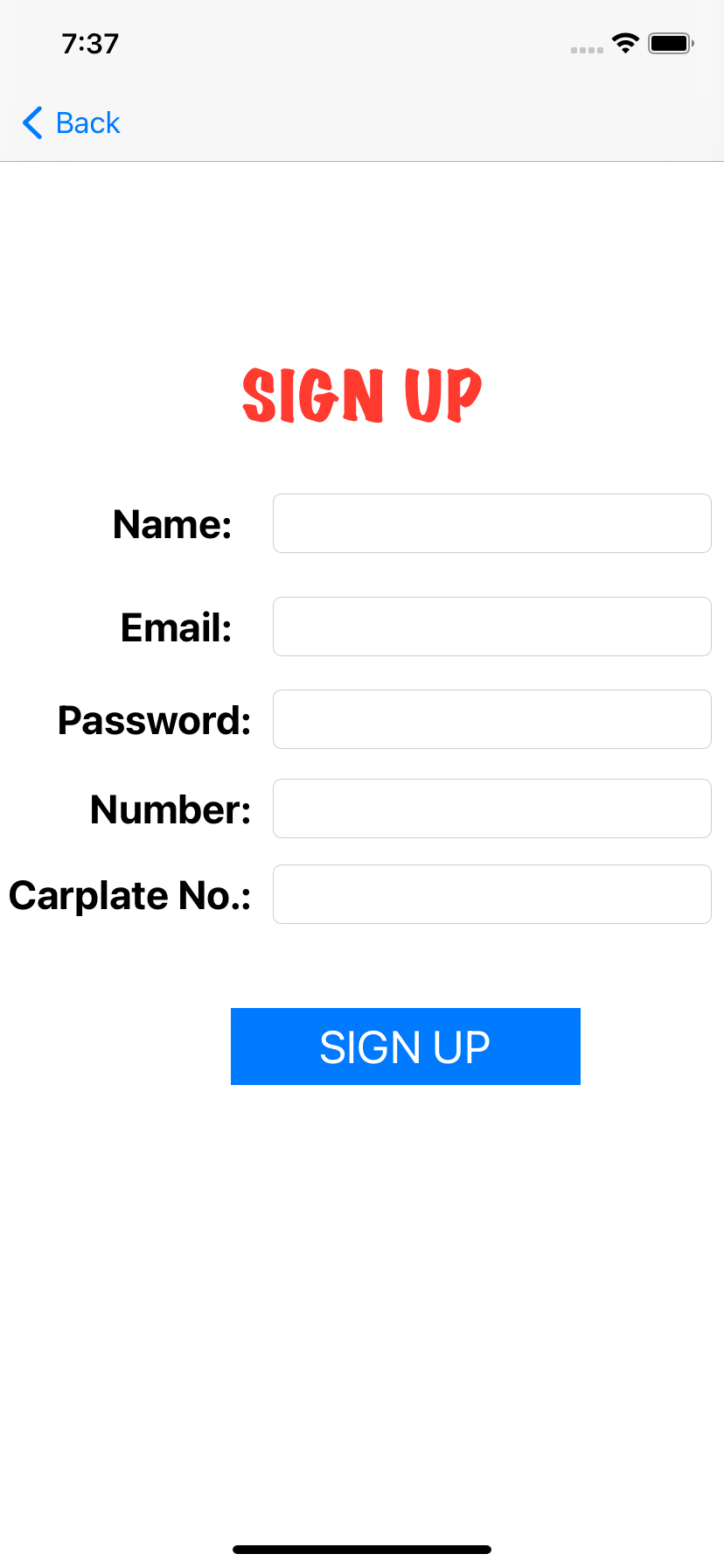
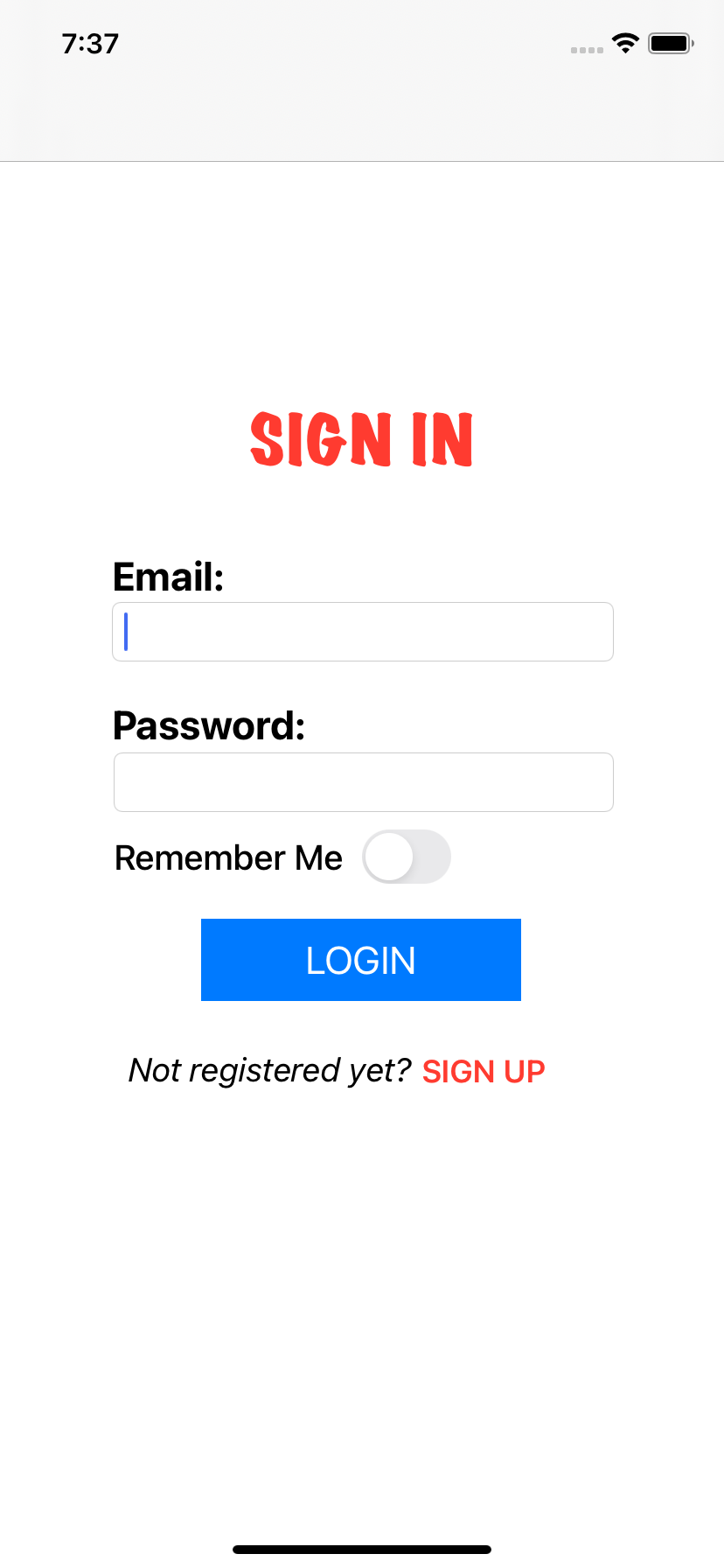
3. Valid users can log in to the app and will be directly re-directed to the Parking List Screen directly if they have selected "Remember Me" at the time of logging in.

4. User can logout from the system.

5. ER diagram updated. Link: https://drive.google.com/file/d/18EXLVrPvINUunm0UO69njjVWhpicZyWX/view?usp=sharing

6. Pods and Google Plist file is created and updated in the project.

Screenshots:



----------------------------------------------------------------------------------------------------------------------------------------

This is an project that is being developed during the assessment of "Advance iOS Development" in the 2nd semester of my "Mobile Application Development and Strategy" program at George Brown College, Toronto.

The project requiremnts are as stated below.

Instructions :

MADS4005 – Advanced iOS Development Project

● This assignment is to be done in a group of 2 members.

● The final grades will be dependent on individual work as well as successful completion of planned

project work as a team.

● Your work will only be graded if you are present in the project demonstration and are able to successfully demonstrate your project.

● Your app will be tested on iPhone 11 Pro for the evaluation. Make sure your app looks and run appropriately on the same.

=> Submission Checklist:

● Write the comments containing your Group number and name at top of each file along with student ID and full name of all group members.

● Once you are done with the project, upload the following to the drop box:

a .zip file (no .rar or .7zip please) containing an entire project having functionalities of all the team members,

b. a link to the master branch of the GitHub repository of your project,

c. a screen recording or video demonstrating app execution with all the functionalities of

the app.

\*\*\*\*\*\* PROJECT REQUIREMENTS AND IMPLEMENTATION \*\*\*\*\*\*\*

Create an iOS application for the Parking App with the following functionalities:

=> User Profile:

Your app must provide appropriate interface to sign-up, sign-in, sign-out, update profile and delete account. Consider requesting name, email, password, contact number and car plate number from user when they create their account.

If you are inclined and interested in adding profile picture functionality, you may add that in your app. However, it is not the official requirement of the project.

No additional grades will be awarded or deducted for including (or not including) this functionality.

=> Add Parking:

The add parking facility should allow the user to create a new parking record with the following information.

- Building code (exactly 5 alphanumeric characters)

- No. of hours intended to park (1-hour or less, 4-hour, 12-hour, 24-hour)

- Car License Plate Number (min 2, max 8 alphanumeric characters)

- Suit no. of host (min 2, max 5 alphanumeric characters)

- Parking location (street address, lat and lng)

- date and time of parking (use system date/time)

You should allow the user to input the parking location in two ways:

- enter street name [In this case the app should obtain location coordinates using geocoding]

- use current location of the device [In this case the app should use reverse geocoding to obtain street address]

After accepting and verifying all information, all parking information must be saved to database.

You must use either CoreData or Cloud Firestore to save the records.

When adding the parking information in the database, make sure that you associate the record with the currently logged in user.

=> View Parking:

This facility will allow the user to view the list of all the parking they have made.

You should display the list with most recent parking first.

You should also display the detail view of each parking when user taps on any item of the list.

When displaying detail view, display all the information about the parking in appropriate format.

In the detail view of parking, allow the user to open the parking location on map and display the route to the parking location from the current location of the device.

=> Evaluation Criteria:

Fundamental iOS Functionalities: (10%)

You should use appropriate iOS view components such as TableViews, Pickers, Navigation Controller, etc. for suitable

features in your app. There should be appropriate navigation between the views.

Use appropriate mechanisms to exchange or share data between views. User inputs must be validated and verified

with appropriate rules.

Data Persistence: (15%)

You must use CoreData or Firebase for data persistence in your app.

You are responsible to design and use appropriate architecture to organize your data for the app.

Location Services: (15%)

You must use location services such as fetching device location, geocoding and/or reverse geocoding for appropriate functionality in your app.

User Interface: (10%)

● Your app must have a UI which should look crisp and beautiful.

● The text in all the Views must be legible.

● Adornments must be subtle and appropriate.

● Consider designing your app using the Human Interface Guidelines provided by Apple.

=> Code Organization and Version Controlling: (10%)

● Your app must be structured using appropriate architecture such as MVVM or MVC.

● The code must be modular, and use appropriate naming conventions.

● Your project must be version controlled using GitHub. The GitHub repository should be private having all the team members and prof as the collaborators.

Each of the team member must be working on their individual branch of the project, perform regular commits and merge their work with master branch of the repository.

Your repository must be private. A public repository means that anyone (including other learners) can see your code.

This puts you at risk of Academic Integrity issues.

● Add the instructors as collaborators to your Github repository. This allows instructors to assess your work. See course webpage for instructor information.

● Contribution in the project by each of the team member will be determining factor for individual grading.

● In the ReadMe file of your GitHub repository, provide the work distribution for each of the team member. There must be exactly one owner for a functionality and/or use-case. Specify the name and student id for each of the members along with functionalities and use-cases they will be responsible for. Your individual grade will depend on the functionality and/or use-case for which you are the owner.

=> Project Demonstration: (40%)

Before demonstration of the project, make sure that you

● merge all the branches of individual team members to the master branch of your repository on GitHub,

● update the ReadMe file of your repository to reflect the functionalities accomplished and add the screenshots for various screens in your app, During the demonstration of your project

● code walkthroughs are to be provided by each team member for the functionalities and/or use-cases they are listed as owner and

● question-answer to be conducted with entire team and individual team members by prof.

● each team member must be able to demonstrate proficiency for the topics included in the project work such as fundamental iOS functionalities, user-interface, data persistence, location services, version controlling, etc.

=> Note:

● Each member of the team must be available at the time of project demonstration. Final grade for the project will only be assigned after the demonstration given by all the team members.

● More details about the mechanism and scheduling of the demonstration will be provided to you.

● Your work will only be graded if you are present in the project demonstration and are able to successfully demonstrate your project.