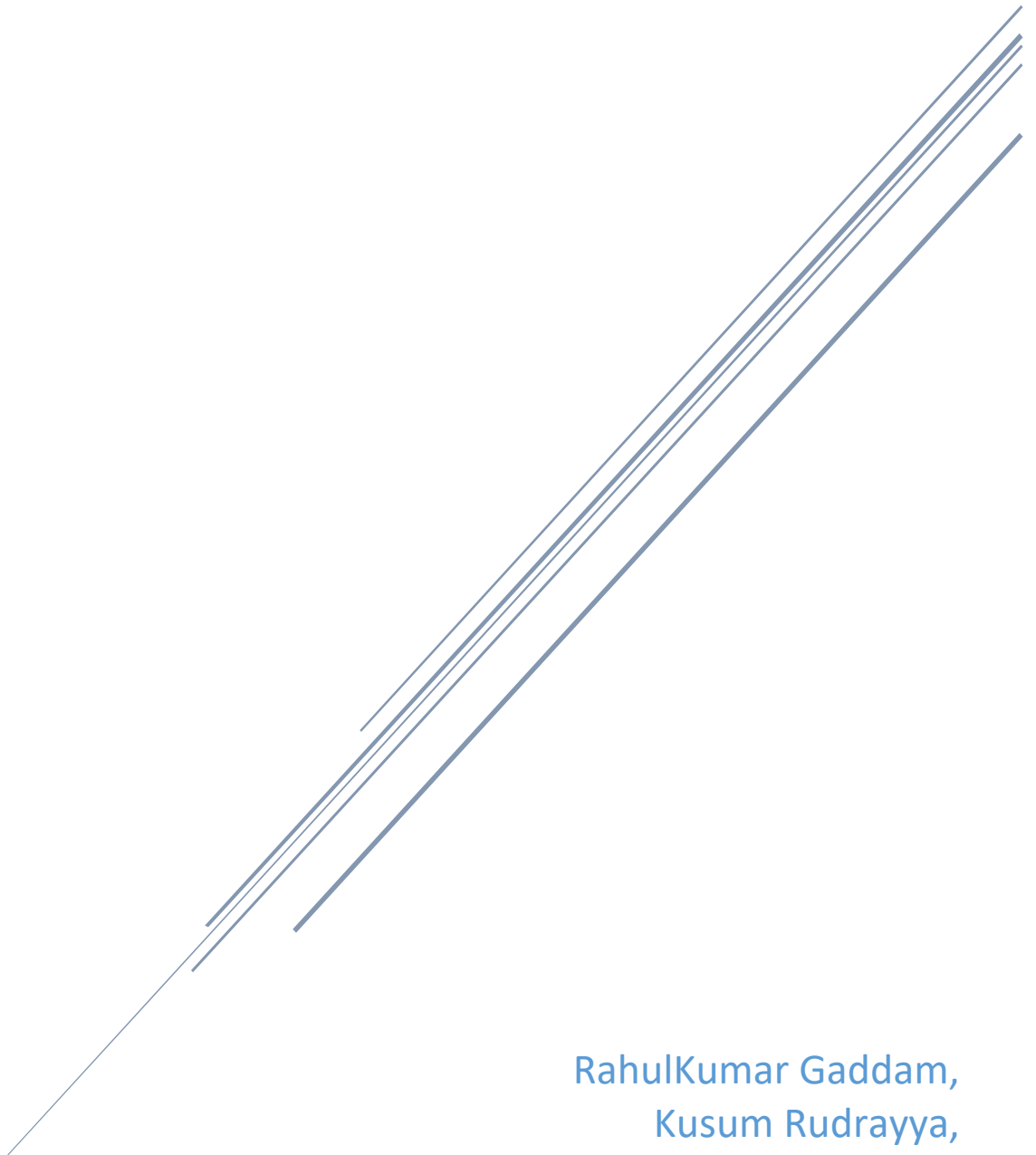


# MEDASSIST

Personalised Medication Management App



RahulKumar Gaddam,  
Kusum Rudrayya,  
PremVemoshKumar Kadavathu Surendran Menon  
CIS/CSE 675 – Mobile Application Programming –Final Project

## About the App:

MedAssist is an iOS application that servers a medication management application for its users. Below you can find the functionalities and brief working of the app.

The user will be able maintain all the details his medication and dosage in the app through his profile. A user profile is needed by anyone who wants to use the app. The user must register with his E-mail ID and password to create a new profile. User will receive E-mail and text notifications on successful registration. He/she must then use the credentials to login into the app. A forgot password option is provided as part of the password management.

This app will require the user to provide their medicine details along with the dosage of the day. The user will be apply to set a time to notify him/her, when it is due for an intake. The application will provide an in-app notification to the user to remind the user about his intake. On successful intake, the user should visit the app and set the time for his next intake. On the home page, the user will show a list of all his medication and the dosage remaining for the day. On selecting an option, he/she will be taken to the medication control page where he/she can edit the dosage and add new medications if needed.

Users are allowed to capture the pictures of prescriptions and bills to associate the image to their medications. Users can also capture or upload their picture to associate it with the profile and use it as a profile picture.

Apart from helping the user with daily medication, the app also provides some convenient features. The user will be able to determine the near-by pharmacies based on his current location. On clicking on the annotation on the map, complete information of the selected app will pop-up on the screen.

Another such feature provided by the app, is the Status page. The status of the user's medication for the day will be determined with the inputs provided by him on the dosage and timely consumption. This status will be shown through an animation in percentage, will 100% being the complete intake of his dosage.

Apart from the other two, the user will also have an emergency information and settings page. In the settings page, the user can turn on and off the notification (Only Email/Text) settings. In the emergency information page, all the necessary information during a medical emergency like Name, Address, Age, Blood Group, Insurance ID, Emergency contact person and their contact number, etc. There will be button near the emergency contact, on clicking it, a call will placed to the contact number.

We have used ParseAPI to act as a database for the application. GoogleMaps API was used to find and show the near-by pharmacies and its details. External application API's were used as servers to send E-mail and text notification.

For more details about working and functionalities of the app, kindly refer the app manual below. We still have future plans for this app to add more exciting and deeply helpful functionalities.

### **App Manual:**

This manual will show how to work on the app, from right after installation. Along with the working, we have also explained in detail about the implementation of the functionalities and resources used to achieve it.

### **Service Used:**

- Use of Animations [Login Button, Loading Screen and Status Page]
- Use of Database [Parse – Cloud based database]
- Notification Services [In-app Notifications]
- Email Notification [Achieved through third party services named 'Mailgun']
- Text Notification [Achieved through third party services named 'Twilio']
- Web Services [GooglePlaces API, GoogleMaps API, Twilio API, Mailgun API and Parse API]
- Phone Services
- Camera Services
- Location Services
- Multithreading [Usage of background threads to handle UI requests]

### **Components Used:**

- UI Scroll View
- UI Table View
- Container View
- Map View
- UI Image View
- Navigation through Slide Bar

### **Note:**

A GPX file is provided along with the project for your reference. If you would like to simulate to a location of your choice, kindly do so.

### Third Party API's and Services Used:

We have used different third party services and APIs to achieve the different functionalities of the app. You can find below the all the details of it. Some credentials of these services are given along with the details, so that verification can be done.

1. Parse: We have used Parse services for database and profile management purposes. Find below the credentials and link to check and verify;  
User Name – rgaddam@syr.edu , Password – Mobile#143, <https://parse.com/login>  
Once logged in, click on the MedAssist option that is shown on the screen.  
Select Core tab from the top menu in the page. Side bar will display the details of the tables.  
Clicking on one of the tables, will list all the values belonging to it.
2. Twilio: Twilio services were used to successfully send text messages to users. Find below the credentials and link to check and verify;  
User Name – rgaddam@syr.edu , Password – Mobile#143, <https://www.twilio.com/login>  
Once logged in, click on the Dashboard tab at the top menu to see the activity from the app.  
Click on PhoneNumbers tab at the top menu to add the phone number to the filtered list.  
Once added, the phone number can receive texts from the app.
3. Mailgun: Mailgun services were used to successfully send Email notifications to users. Find below the credentials and link to check and verify;  
User Name – rgaddam@syr.edu , Password – Mobile#143, <https://mailgun.com/cp>  
Once logged in, the screen will show a dashboard representing the activity of Email with regards to the app. No of mails delivered and failed will be shown along with date of the email trigger.

### Limitations:

Listed below are some of the limitations that the app has because of using free third party services. Kindly make a note of these before testing or probing the application.

1. Camera Feature - Camera doesn't work on simulator and has to be test on a device. We showed you the working of camera feature in our app, during our presentation by taking a picture of you.
2. Nearby Pharmacy Page - If you are not able to see the red pins on arriving at the page, kindly zoom and move the map. There aren't a lot of listing for pharmacies and you have a search a little to find the red pins.
3. Mail Notification – Because of the use of the free third party mail sending server 'Mailgun', the app has a limitation of sending mails only to Gmail account.

4. Text Notification - Because of the use of the free third party mail sending server 'Twilio', a mobile number has to be registered with the server at Twilio's site. Credentials and link to register are provided in the before section.

### **User Registration, User Login and Profile Management:**

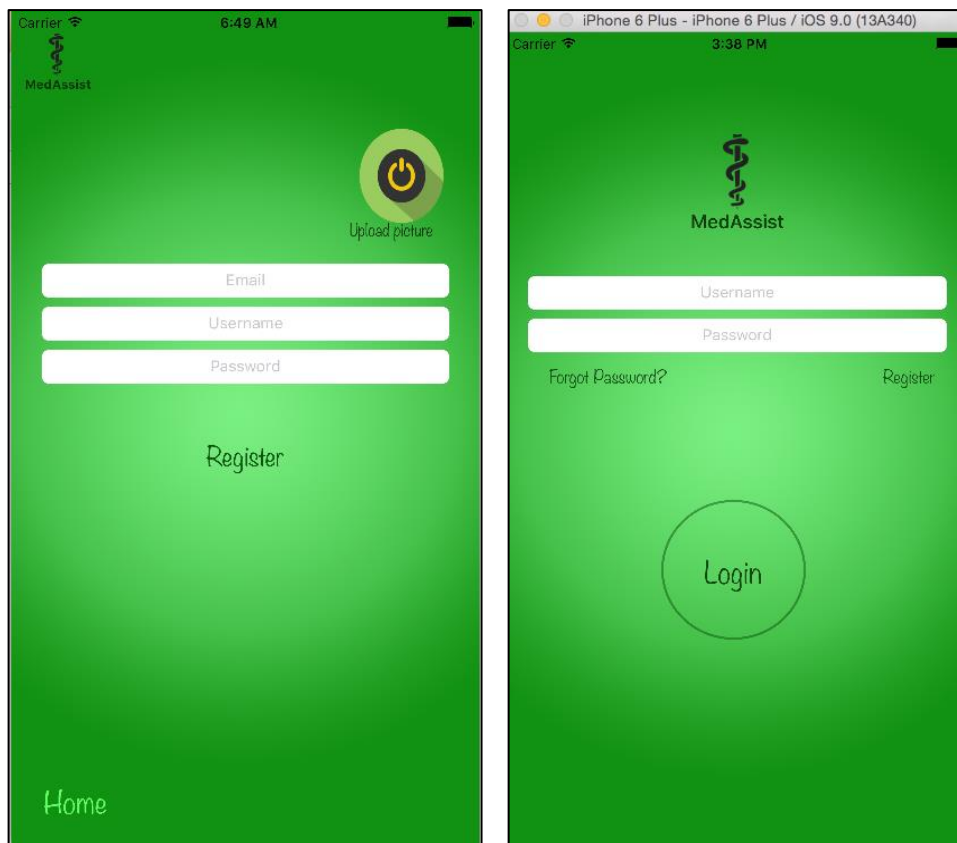
After installation, the user needs to register to use the app. The login and registration page are as shown below. User would need to key in Email ID, username, phone number and password to register for the app. On successful registration, notification will be sent in form of both Email and Text. Now the user will be able to login into the application with the credentials registered.

We allow the user to capture or upload his picture, to use it as a profile picture in the app. The upload button will access the internal iOS photo library to fetch existing pictures and camera for capturing new pictures.

On every login, a notification (both Email and Text) will be sent to the user. The setting can be switched off from the settings page, which will be discussed later in the manual.

We have used Parse, a powerful cloud database as our back-end database. All the information provided by the user will be saved in this database.

We also provide a Forgot Password option in the app. When selected, the app will request for the registered Email. A mail will be sent to his Email ID with a link to reset the password. This functionality is developed with the help of ParseAPI.



### Home Page and Slide Bar:

On successful login, the app will take the user to the home page. The home page for new user will be blank. He has to add his daily medication and its dosage for the home page to populate. Once the necessary additions are done, the page will display the progress each medication with respect to its consumption by the user. On clicking on any of the medicine from the list in home page, the app will show the medication control page of the selected medicine.

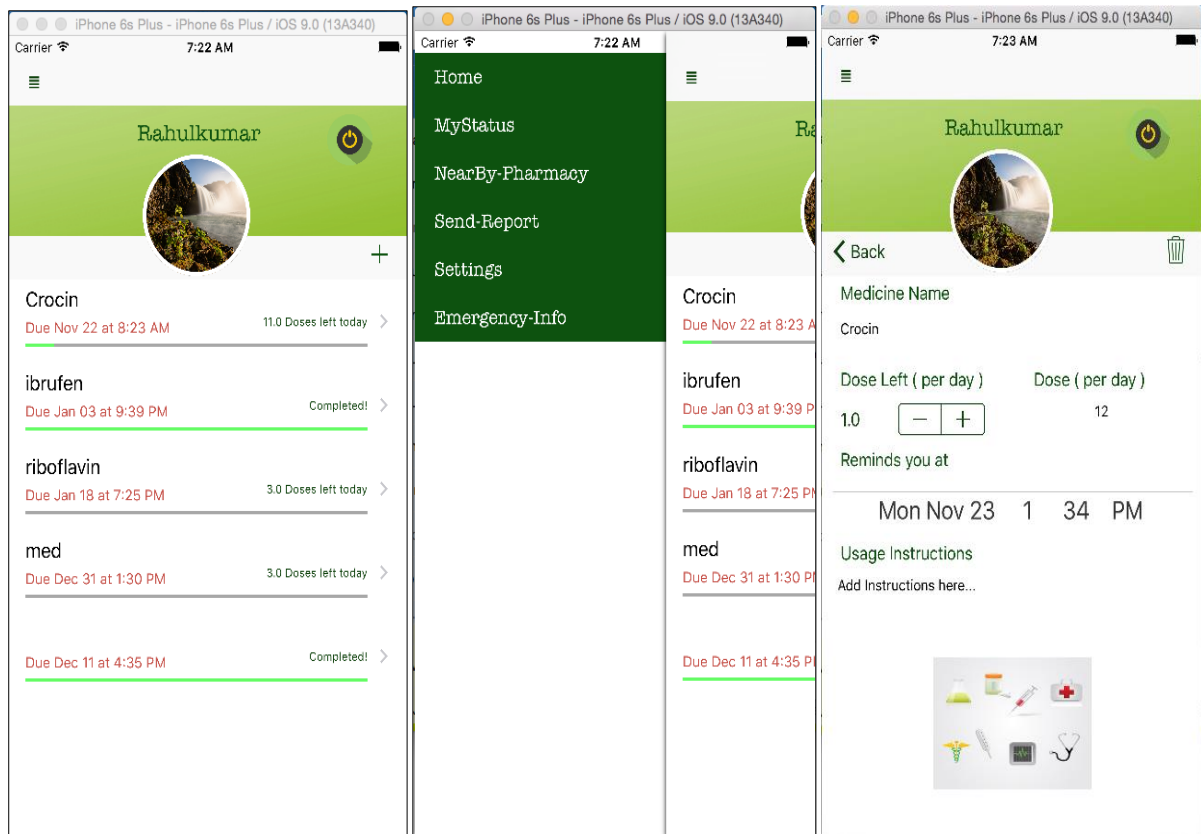
A logout button is shown in the top right corner of the screen to log out of the profile. A plus signed button is located below the log out button. On clicking this, it will be redirected to medication control page, where the user can add new medicine for the day to manage. Details about this page is discussed in the corresponding section, later in the document.

On returning to the home page from anywhere, always refresh it by sliding down. The refresh details are shown while the action is performed.

Apart from this, the home page also houses a slide bar. The slide bar contains options to navigate to different pages of the app. The options provided in the slide bar are Home, Status, Send Report, Near-by Pharmacy, Settings, and Emergency Information.

Details of the screenshots below from left to right;

1. Home page of a registered user, listing all his medicines for the day.
2. Home page with the slide page, showing all the options to navigate to.
3. Medication control page by clicking on the first option. You can find the resemblance in details on the 1 and 3 picture.



### Medication Control Page:

The page will appear at two instances;

1. When the user clicks on the plus signed button on the top right side of the home page.
2. On clicking on any of the list of medicines and their progress that appear on the home page.

This page allows the user to add daily medication to manage, control the dosage of the medicines, and upload prescription/bills to associate to the medicine, set up a remainder notification for medicine intake and delete the medicine from the list.

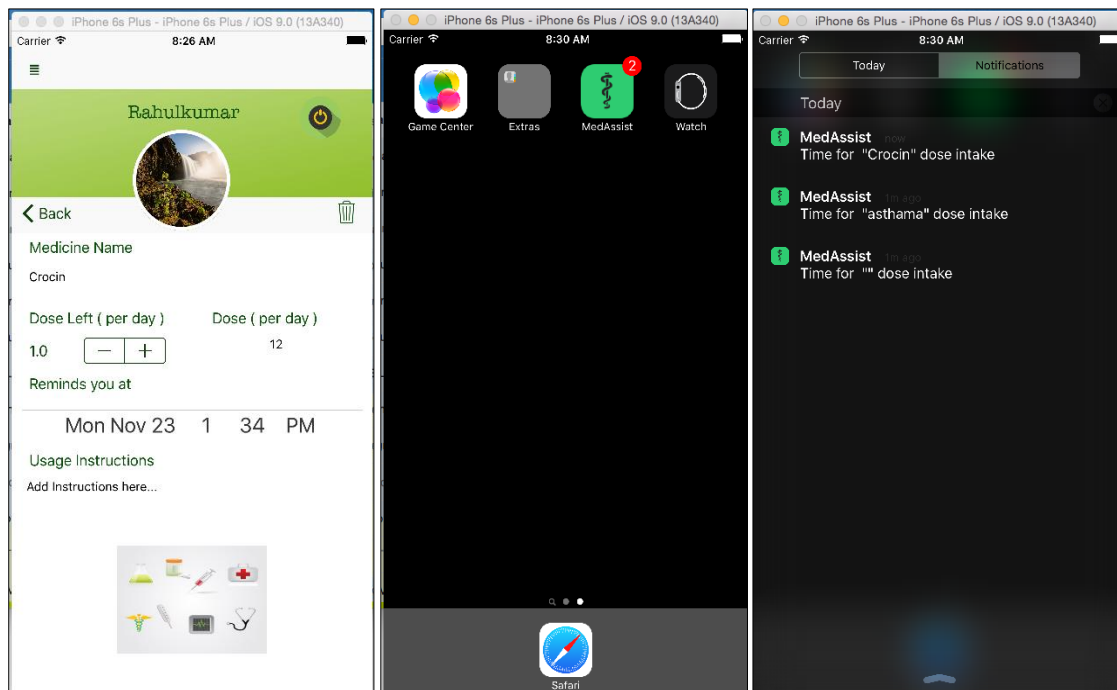
When adding a new medicine, the user is prompted to enter its name and dosage. Dosage can be added using the '+' and '-' signed buttons. The remainder for the intake can be set with the slider representing a timer. Usage of the respective medicine can also be added. A save button is given at top right corner, below the logout button. Clicking on this button, the camera will be accessed and opened to get the picture of bill/prescription.

The user can go back to the home page either through the back button or the slide bar.

Details of the screenshots below from left to right;

Designed and Developed by,  
RahulKumar Gaddam, Kusum Rudrayya and PremVemoshKumar Kadavathu Surendran Menon

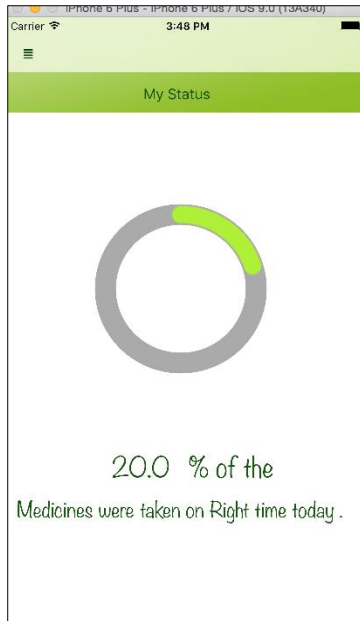
1. Medication control with information of the option clicked on the home page. If the plus sign in the homepage was clicked, then the medication control page will be unpopulated.
2. App icon showing the notifications count.
3. The notifications from the app are listed, if the top notification bar is slide down.



### Status Page:

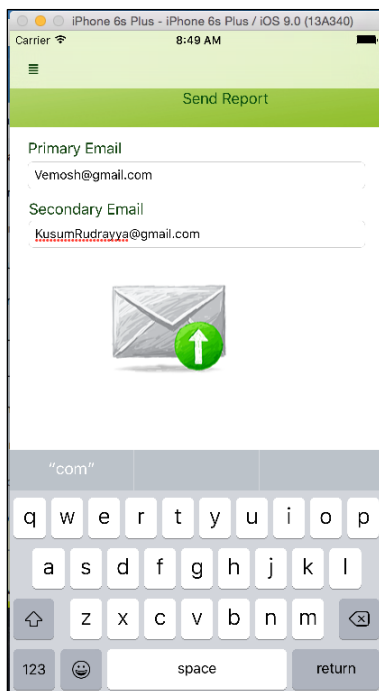
Status page can be accessed from the slide bar. The status page shows the consolidated progress of the user's medication for the day. The progress is shown in percentage, with 100% being the total consumption of the dosage keyed in by the user. The progress is shown through custom created animation. The screen shot of the page can be found below;





### Send Report Page:

Status page can be accessed from the slide bar. This page sends a report of the day's medicine consumption of the user against the dosage prescribed. The functionality can be used to send reports at the end of the day to the user's doctor or concerned person, to keep track of the user's proper medication consumption.



### Near-by Pharmacy Page:

Near-by Pharmacy page can be accessed from the slide bar. On selecting this page from the slide bar, a map is shown with red pins indicating the location of the pharmacies near-by. User's current location is obtained before the search for pharmacy location. With the user's location as reference point, near-by pharmacies are located and shown on the map. On

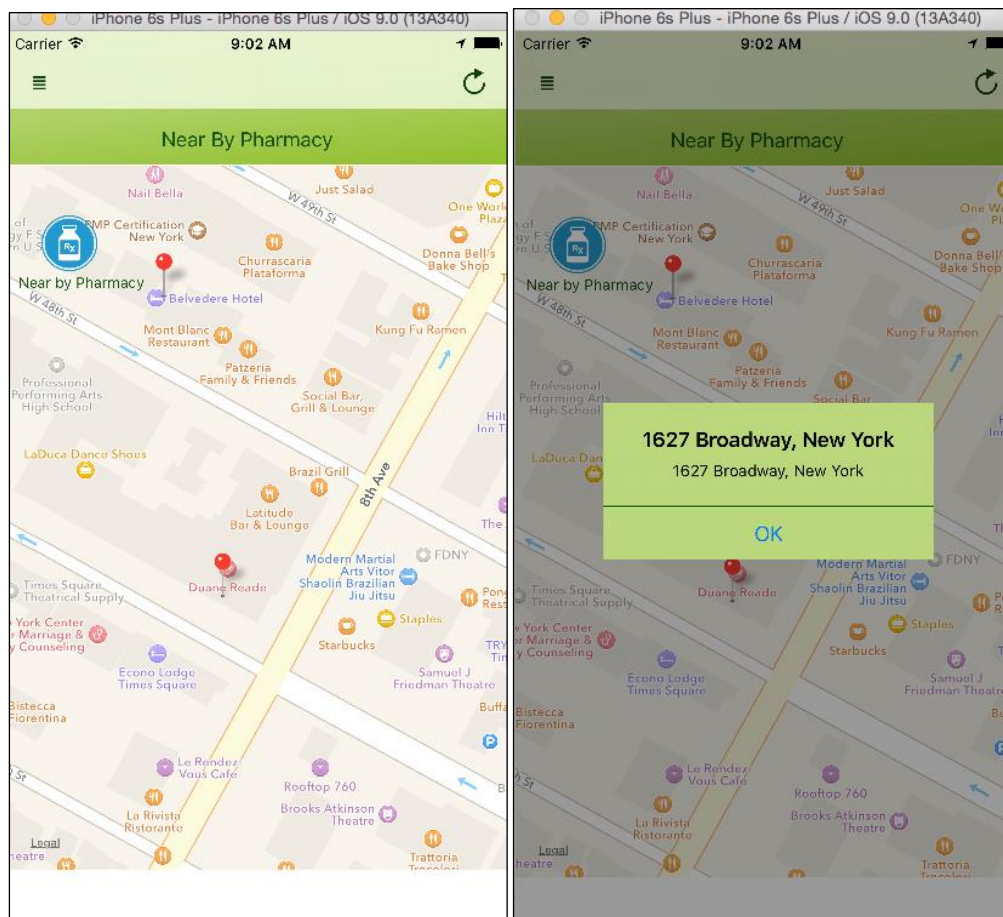
clicking on the red pins, the complete address of the selected pharmacy is shown as annotations.

GoogleMaps and GooglePlaces API is used to show the map and locate the pharmacy locations.

The user can go back to the home page either through the back button or the slide bar.

Details of the screenshots below from left to right;

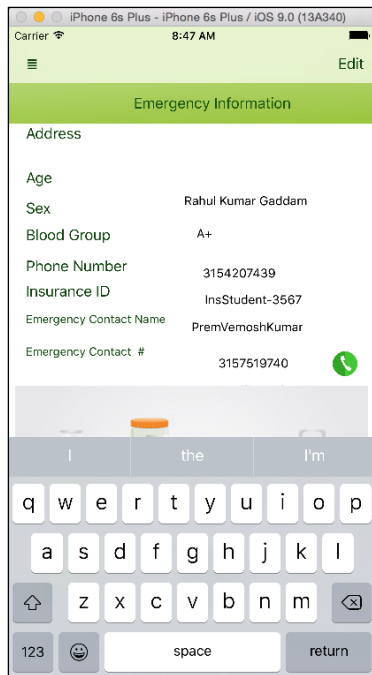
1. In the image, red pins show the location of near-by pharmacies.
2. Image of an annotation providing the address of the pharmacy selected, by clicking on the red pin.



### Emergency Information Page:

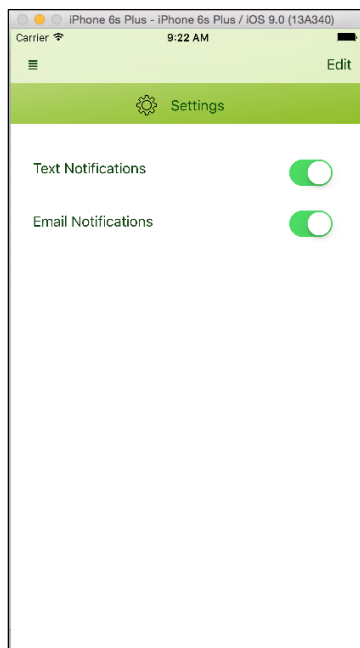
Emergency Information page can be accessed from the slide bar. This page will hold all the information needed at a time of medical emergency. The details shown can be seen in the screen shot of the page shown below. On clicking the phone button next to emergency contact, a call will be placed immediately to the mentioned phone number.

Designed and Developed by,  
RahulKumar Gaddam, Kusum Rudrayya and PremVemoshKumar Kadavathu Surendran Menon



### Settings Page:

Settings page can be accessed from the slide bar. This page will allow the user to stop the notifications which will be send, when the user logs in into the app. Separate controls are given to stop Email and text notification. This way, the user can keep the type of the notification that he desires. The screen shot of the page can be found below;

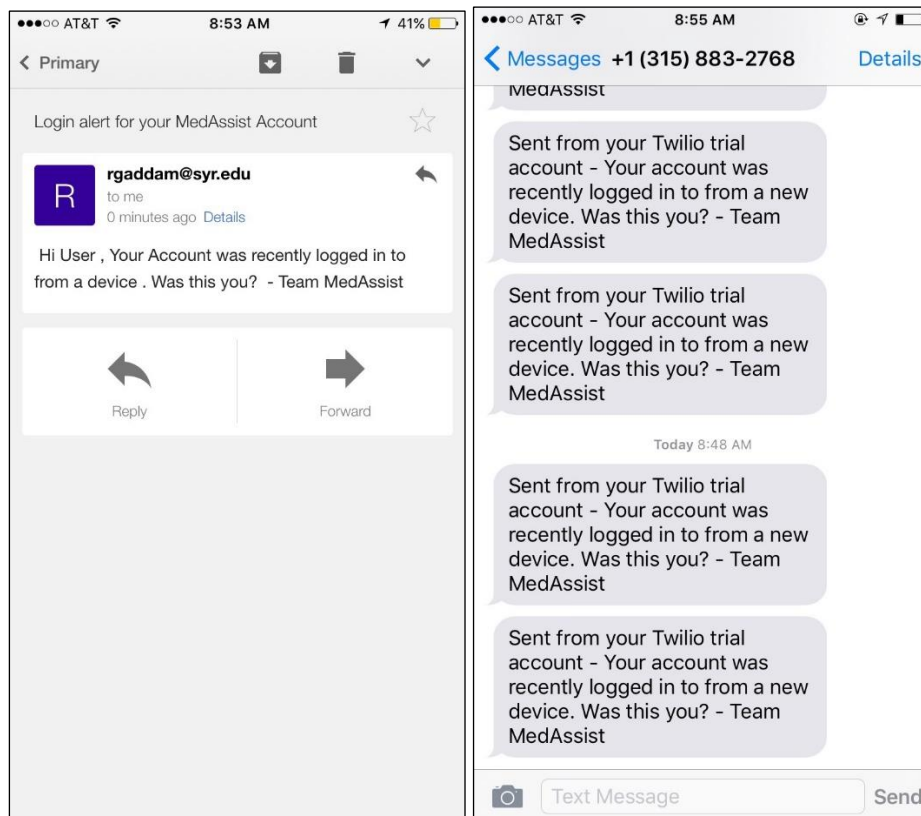


### Notifications:

This section shows the screen shots of the email and text notification that are obtained through this app. The notification are sent only when the user logs in to the app. The user can stop the notifications by using the settings page as explained above.

Details of the screenshots below from left to right;

1. This is a sample screen shot of the Email that was received because of the app. As explained in limitations, mail can only be sent Gmail accounts.
2. This screenshot shows the text message obtained from the app. As explained in limitations, the mobile number must have been registered at Twilio server through their site to receive text.



All the details about the app's working and functionalities are explained as possibly can. We do have future plans for this app, to add more meaningful and user interactive functionalities.