|  |  |
| --- | --- |
| Track my mrt  Mobile Wireless Solution Design (SG4207) | Abstract  “Track My MRT” is a native ios application, that helps you track the MRT’s arrival time and fare. It also allows you to share the details of the MRT with a friend.  Team 03  Moushumi Seal (A0163243M)  Mridul Agarwal (A0163351M)  Thirumal (A0163271L)  Varun Sampath (A0163297W) |

Track My MRT

# Salient Features

* Tracks the MRT’s arrival time and fare.
* Locate your boarding and destination stations and your current location.
* Share your current location via SMS.
* Share the timings, fare and other details by sending screen shot via Airdrop, WhatsApp and so on.
* MRT guide map for quick reference.

# Design Considerations

The application is a native ios application developed using the swift 3.0. The app uses the REST API, [MRTAPI.com](http://mrtapi.com/) which provides the MRT Train Arrival Timings in Singapore. This service provides live train arrival timings of different stations. In addition to the arrival time, our application also shows the fare for travelling from the boarding station to the destination station.

We have integrated the Google Maps API to track the current location of the user and also point out the location of the boarding and destination stations. The user may choose to share his/her current location via sms to a friend.

The app also allows the user to share the timings, fare and intermediate station details as a screen shot via Airdrop, WhatsApp, Message (MMS), Twitter, Facebook and other social media.

## Reasons for using Native over hybrid

* Users expect the app to launch faster than a website, are very concerned with mobile app performance and if they have a poor user experience, they will most likely stop using the app and switch to a competitor’s app
* Native Apps have easier access to device sensors (GPS, accelerometer, etc.)
* Betteruser experience.

# Test cases

|  |  |  |
| --- | --- | --- |
| **Scenario** | **Expected Result** | **Pass/Fail** |
| User selects From and To stations from the dropdowns and clicks the “Track My MRT” button. | The screen will display the arrival time and next train’s arrival time of the From station and total fare of the journey. The screen also shows a list of intermediate stations. |  |
| User selects only From station and clicks the “Track My MRT” button. | The user is prompted with an alert box with an appropriate error message to enter the To station. |  |
| User selects only To station and clicks the “Track My MRT” button. | The user is prompted with an alert box with an appropriate error message to enter the From station. |  |
| User does not select any station and clicks the “Track My MRT” button. | The user is prompted with an alert box with an appropriate error message to enter both the stations. |  |
| User selects both From and To station on the home page and clicks on the “View Map” tab. | The map shows the user’s current location and the two selected stations in the form of markers. |  |
| User selects From station on the home page and clicks on the “View Map” tab. | The map shows the user’s current location and the From station in the form of markers. |  |
| User selects To station on the home page and clicks on the “View Map” tab. | The map shows the user’s current location and the To station in the form of markers. |  |
| User does not select any station on the home page and clicks on the “View Map” tab. | The map shows only the user’s current location. |  |
| User clicks on the “View Map” tab and clicks the “Send My Location” button | The messaging app opens with google map link of the user’s current location. Now the user can select a contact and send the sms. |  |
| User selects from and to station from the Home tab and clicks on the share button in the result page. | The user is given a variety of options to share the screenshot of the screen with any contact or person. |  |
| User clicks on the “MRT” Guide” tab | The MRT Guide map is displayed on the screen. |  |

# Lessons Learnt

* Consuming APIs and fetch data from them.
* Reading JSON objects in ios swift
* Sending and receiving data in wireless networking
* Using Async task for preprocessing of data
* Integrating the Google Map API to fetch user’s current Location
* Different components of ios swift (ViewControllers, Storyboard and so on)
* Deploying application in iphone
* Wireless messaging with UIActivityControntroller

# Work distribution

The work had been equally divided among all the team members.

The following are the mobile technologies that we have used in our app:

a) Network Programming

b) Wireless Messaging (in the form of sms)

c) Location and Map

d) Sharing of data with other devices using Airdrop, WhatsApp, Twitter, Facebook and other social media.

The work distribution among team members were as follows:

* **Moushumi**: UI Design and wireless messaging.
* **Mridul:** Integration of Google maps and Location based service
* **Thirumal:** Fetching data from MRTAPI.com
* **Varun:** Integration ofthe sharing feature

Further both Thirumal and Varun have contributed in preparation of JSON data, loading static data into the database and fetching data from the database. Mridul and Moushumi have worked on annotating the map with user’s location and from and to stations.

# References

* <https://developer.apple.com/swift/>
* <http://mrtapi.com/>
* <https://grokswift.com/>
* <https://developers.google.com/maps/documentation/ios-sdk/>