*Florida International University*

*School of Computing and Information Sciences*

Software Engineering Focus

Final Deliverable

Project Title: Nihon-GO 1.0

**Team Members:** Eduardo Porto, Miguel Rischmaui, Luis Granados, Gino Tuero

**Product Owner(s)**: Asuka Mashav

**Mentor(s)**: N/A

**Instructor**: Masoud Sadjadi

The MIT License (MIT)

Copyright (c) *2016 Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

***Abstract***

*This document presents the information necessary to gain a good understanding of Nihon-GO 1.0. It is a cross-platform mobile application that allows users to share travel experiences in the form of itineraries. It will be mainly used by foreigner travellers who would like easy access to travel itineraries created by foreigners. The application becomes a social platform where foreigners guide foreigners and allows them to create itineraries that they can also publish. The application also has an editable user profile that allows user to have a personalized application. The user profile contains basic information as well a list of the itineraries created by the user and the itineraries marked favorite. Another feature is a guide section which contains useful information and tips on travelling Japan.*

[**Introduction**](#_30j0zll) **6**

[Current System](#_p2u8i3fc1akz) 6

[Purpose of New System](#_ya4g7dblq7sb) 6

[**User Stories**](#_2et92p0) **7**

[Implemented User Stories](#_c7hih0xo5acb) 7

[Pending User Stories](#_sd9mg6dm4e5l) 21

[**Project Plan**](#_1t3h5sf) **22**

[Hardware and Software Resources](#_4d34og8) 22

[Sprints Plan](#_2s8eyo1) 24

[Sprint 1](#_17dp8vu) 24

[Sprint 2](#_gisnnli1yy7) 24

[Sprint 3](#_7ithpevh3l7u) 24

[Sprint 4](#_kn472y31dels) 25

[Sprint 5](#_gouq25ti2vtd) 26

[Sprint 6](#_r13686w47xek) 27

[**System Design**](#_3rdcrjn) **28**

[Architectural Patterns](#_h9iwtez4c3bx) 28

[System and Subsystem Decomposition](#_lnxbz9) 29

[Deployment Diagram](#_35nkun2) 30

[Design Patterns](#_b3cfp9gjvk3x) 30

[**System Validation**](#_44sinio) **31**

[**Glossary**](#_2jxsxqh) **32**

[**Appendix**](#_z337ya) **33**

[Appendix A - UML Diagrams](#_j4zozibrbpb8) 33

[Appendix B - User Interface Design](#_2xcytpi) 43

[Appendix C - Sprint Review Reports](#_x74uo56n9auu) 43

[Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents](#_6w97znsp1i6t) 48

[**References**](#_2bn6wsx) **48**

# 

# 

# 

# 

# 

# Introduction

Nihon-GO is a Social travel application for foreigners travelling to Japan. Users can find, share and use experiences in the form of itineraries. It features an interactive map of Japan that allows users to explore regions in Japan, an itinerary system that allows users to create and use itineraries, and a profile system that allows users to track itineraries they create, start and favorite.

## Current System

This is the first iteration of the application, so there is no current system.

## Purpose of New System

The project was a newly created project with no previous code. For this reason, our main goal was to create an application with core features and functionalities that also has the capability to scale as it continues to grow. For scaling purposes, we used Redux which creates global states that can be modified and allows future code to be easily merged and expanded.

The core features that were implemented were: creating and publishing itineraries, viewing itineraries, editable user profile, option to upload pictures, manage user created itineraries and list of favorites, use a drawer for quick navigation, and create a guide section. Material Design was used to help design the application and provide simple and common design features that are used in many projects. This allows the app to be intuitive and require minimal effort. React Native Elements was used to help create a cross-platform application by providing the tools and components to create and modify design components. Our project was based on Expo which provided the necessary tools to implement our features as well as facilitate the creation of new projects.

# User Stories

The following section provides the detailed user stories that were implemented in this iteration of the …. project. These user stories served as the basis for the implementation of the project’s features. This section also shows the user stories that are to be considered for future development.

## Implemented User Stories

* 696 Create Home page
* 709 Home page Navigation
* 715 Subcategory guide list
* 697 Homepage interactive map
* 717 Regions Map
* 723 Start Itinerary functionality
* 694 Consolidate Create Itinerary
* 701 Guide hub that lists relevant guides
* 708 Modify Blog/Guide feature
* 727 Connect Guides to firebase
* 695 Consolidate Create Itinerary
* 731 Implement Create Itinerary
* 702 A guide template for each guide
* 726 Refine and reformat guides
* 718 Complete Itinerary Core Functionality
* 722 Link Itinerary Functionality to Firebase
* 729 Input Validation for Create Itinerary
* 730 Image Picker for Itineraries
* 671 Create User Profile Page
* 672 Create Option to Edit User Profile
* 674 Create Option to Add User Profile Picture
* 705 Modify Login Page
* 706 Modify User Profile Page
* 713 Redesign Login Page 1.0
* 714 Redesign User Profile 1.0
* 719 Redesign Login Page 2.0
* 720 Redesign User Profile 2.0

**Link Itinerary Functionality to Firebase:**

* Description: As a developer I would like the itineraries to be saved on the database and linked to users who made them.

Acceptance Criteria

* Itinerary create should save an itinerary to the database
* Users should be able to modify itineraries they created.
* Itineraries should be linked to the users who created them.

**Use Case**

* Name: User wants to create an Itinerary.
* Actor: User
* Preconditions: App needs to be installed, user must have an account, and user must be logged in.
* Description: A logged in user navigates to the Create Itinerary form through the floating action button. Once at the Create Itinerary form, the user fills out the required fields, and for every event the user wishes to add, the user clicks Add Event and fills out the add event form. Once the user is satisfied with their itinerary, the user may submit the itinerary to the application, saving it to the database.

**User Story Name: Regions Map**

* Description: As a user I would like see information on regions through the interactive map, so that I may explore Japan. I should then be able to view the itineraries of that region

Acceptance Criteria

* Map must be divided in 9 regions that when interacted with, bloom and shortly navigate the user to the region modal.
* Region modal must have a link to itineraries of that region.

**Use Case**

* Name: View region itineraries
* Actor: User
* Preconditions: Be on homepage
* Description: User selects a region on homepage map, Information on the region is displayed with a link to itineraries, user clicks link and is taken to a list of itineraries of that region ordered by favorites (desc).
* Name: Select region
* Actor: User
* Preconditions: Be on Homepage
* Description: User opens homepage and select a region on the interactive map. Information on the region is displayed.

**User Story Name: Homepage interactive map**

* Description: As a user I would like to select a region on the map of japan in the home page that takes me to a region profile page, as I would like to explore itineraries and info on that region.

Acceptance Criteria

* Make home page map interactive
* Divide map by regions
* Regions take you to a respective regions page

**Use Case**

* Name: Select region
* Actor: User
* Preconditions: Be on Homepage
* Description: User opens homepage and select a region on the interactive map. Information on the region is displayed.

**User Story Name: Home page Navigation**

* Description: As a user I’d like to have a Menu to navigate the app in an intuitive way, since I may not be familiar with the ideal flow of the app.

Acceptance Criteria

* Add a drawer for menu functionality.
* drawer displays links to homepage, guides (current page highlighted) and login button when not logged in
* drawer displays links to homepage, guides, profile (current page highlighted) and profile card when logged in

**Use Cases**

* Name: Login from menu
* Actor: Any User
* Preconditions: App opened to homepage, user not logged in
* Description:

1. User clicks on menu icon on the top right corner, drawer with scene options (no profile option, homepage highlighted) and login button is displayed.
2. User clicks login, drawer closes and login page is displayed (back icon in corner instead of menu icon).
3. User logs in and is redirected to homepage
4. User clicks on menu icon on the top right corner, drawer with scene options (homepage highlighted) and profile card is displayed.

* Name: Navigate to homepage
* Actor: Any User
* Preconditions: App opened on profile or guides page
* Description:

1. User clicks on menu icon on the top right corner, drawer with scene options (current page highlighted) and profile card or login button is displayed.
2. User clicks home , drawer closes and home page is displayed.
3. User clicks on menu icon on the top right corner, drawer with scene options (profile highlighted) and profile card is displayed.

* Name: Navigate to profile
* Actor: Any User
* Preconditions: App opened to homepage, user logged in
* Description:

1. User clicks on menu icon on the top right corner, drawer with scene options (homepage highlighted) and profile card is displayed.
2. User clicks home, drawer closes and home page is displayed.
3. User clicks on menu icon on the top right corner, drawer with scene options (profile highlighted) and profile card is displayed.

* Name: Navigate to Guides
* Actor: Any User
* Preconditions: App opened to homepage
* Description:

1. User clicks on menu icon on the top right corner, drawer with scene options (homepage highlighted) and profile card or login button is displayed.
2. User clicks Guides, drawer closes and Guides page is displayed.
3. User clicks on menu icon on the top right corner, drawer with scene options (Guides highlighted) and profile card is displayed.

**User Story Name: Create Home Page**

* Description: As a user I would like the app to take me to the home page when I open it, because I might not want to register to use the app. I want the home page to have links to all the features of the app

Acceptance Criteria

* Link to user services (login, register, edit profile)
* Link to guides
* Display map of japan.
* List itineraries that cover all of Japan.

**Use Case**

* Name: Open app for first time
* Actor: New User
* Preconditions: App installed
* Description:

User clicks on Nihon-GO icon on their phone, the app opens the home page which displays a list of itineraries (Global), an interactive map of Japan, a guide button and a link to sign in.

**User Story Name: Start Itinerary functionality**

* Description: As a user I would like to select an itinerary and use it. I want to track my progress with it and be able to return to it from the menu (drawer) at any time, so that I don’t have to search for it to use it.

Acceptance Criteria

* Users start an itinerary which has a state tied to progress.
* User can cancel a started itinerary
* User can continue a started Itinerary
* If user has started an itinerary other itineraries are in view only mode
* Only logged in user can start
* Itineraries, once started should be accessible from the menu until they are completed or abandoned.

**Use Case**

* Name: Start Itinerary
* Actor: User
* Preconditions:In an itinerary view, user logged in
* Description:User is viewing an itinerary, user clicks on start itinerary, start button replaced by a cancel button.

**User Story Name: Consolidate Itinerary View**

* Description: As a developer, I would like to have all parts of viewing an itinerary brought together to functional completion.

Acceptance Criteria

* Should have Itinerary page showing all itineraries
* Once Itinerary is selected, should expand into an itinerary view
* Once an event within an itinerary is selected the event should expand into an event view.

**Use Case**

* Name: The User wants to see available itineraries
* Actor: Any User
* Preconditions: App Installed and at Itinerary Page
* Description: After having loaded the application, the user is on the home page and navigates to the list of existing itineraries. Upon selecting an itinerary, the itinerary expands to show the individual events in the itinerary. Upon selecting an event, the event expands to show more details.

**User Story Name: Guide hub that lists relevant guides**

* Description: **As a** User **I would like** to be able to access guides **so that** I can be more informed.

Acceptance Criteria

* Shows guides
* Has buttons for interaction

**Use Case**

* Name: The user wants to view the available guides
* Actor: Any user
* Preconditions: App installed and is at the home page
* Description: After having opened the app the user is on the homepage and selects guides from the page. Upon selecting the user is taken to a page listing a series of guides on a scrollable page. The guide page presents the user with interactive buttons for each guide.

**User Story Name: Modify blog guide feature**

* Description: **As a** Developer **I would like** to be able to have a form I can fill out with criteria to create a new Guide and I want the guides themselves to open a more detailed page when selected **so that** I can set up guides for users to use and open them to make sure they are displaying correctly.

Acceptance Criteria

* Can press a button to add a new guide which will take the developer to a form to fill out the guide
* Can view the guide with a button on each guide that opens it.

**Use Case**

* Name: Developer wants to add a guide
* Actor: Developer
* Preconditions: App opened and at guide page
* Description:

1. Developer clicks on the Add Blog button on the top right corner and they are taken to a create a page form.
2. Developer fills out the provided form by typing into each of the fields.
3. Developer presses submit and the text in the fields is stored.

* Name: User wants to view a guide
* Actor: User
* Preconditions: App opened on the guide page
* Description:

1. User presses the view button on a guide, they are taken to a new page
2. The page takes the stored information, and shows an expanded view of the guide to the developer or user.

**User Story Name: Subcategory guide list**

* Description: **As a** User **I would like** to be able to open a blog category and view the subcategories **so that** I can can view only the things in my category of interest

Acceptance Criteria

* Can enter subcategory list
* Subcategory can expand with animation

**Use Case**

* Name: Select Category
* Actor: User
* Preconditions: Be on the Guides page
* Description: User opens the guides page and selects the category object they wish to view. Then a set of subcategories is presented to the user.

**User Story Name: refine and reformat guides**

* Description: **As a** User **I would like** to select a category for guides and open a list of subcategories which when selected a subcategory expands **so that** I can read up on relevant information to the region.

Acceptance Criteria

* Correct existing errors in feature
* Increase efficiency by implementing list view to limit the amount of loaded objects
* Fix Redux

**User Story Name: Connect guides to database**

* Description: **As a** User **I would like** to the guides to write and receive data from our database **so that** I have a place to store and read that from.

Acceptance Criteria

* Data in database is formatted with unique keys and unique id’s for organization
* Interacts with the database reading/writing
* Can make use of the data

**Use Case**

* Name: User wants to select a subcategory
* Actor: User
* Preconditions: User needs to have the app and be on the guide page
* Description Upon being on the guide page the user selects a category of choice which loads all subcategories under that and then the subcategories expand when pressed

**User Story Name: Create user profile page**

* Description: As a user, I would like to create a profile, so that I can personalize the app.

Acceptance Criteria

* Allow option to add user profile information.
* Allow option to edit user profile.
* Display user profile and get data from database.

**Use Case**

* Name: User wants to see their profile.
* Actor: User with an account.
* Preconditions: App needs to be installed. Only signed in users can see and edit profile.
* Description: A signed in user opens the app and sees the User Profile button. They click it and the app navigates to the User Profile page and loads the user’s data.

**User Story Name: Modify login page**

* Description: As a user, I would like to login, in order to see user-only data.

Acceptance Criteria

* Allow option to login using credentials.
* Allow option to register using credentials.
* Use Firebase as the authentication service.

**Use Case**

* Name: User wants to login.
* Actor: User with an account.
* Preconditions: App needs to be installed.
* Description: A user with an account opens the app and signs in using user credentials.

**User Story Name: Modify user profile**

* Description: As a user, I would like to edit and see my user profile, in order to personalize my profile.

Acceptance Criteria

* Allow option to add user profile information.
* Allow option to edit user profile.
* Display user profile and get data from database.

**Use Case**

* Name: User wants to see their profile.
* Actor: User with an account.
* Preconditions: App needs to be installed. Only signed in users can see and edit profile.
* Description: A signed in user opens the app and sees the User Profile button. They click it and the app navigates to the User Profile page and loads the user’s data.

**User Story Name: Redesign Login Page 1.0:**

* Description: As a user, I would like to login in order to see my profile

Acceptance Criteria

* Login functionality with error responses.
* Register functionality with error responses.

**Use Case**

* Name: User wants to login.
* Actor: User with an account.
* Preconditions: App needs to be installed.
* Description: A user with an account opens the app and signs in using user credentials or creates an account.

**User Story Name: Redesign User Profile 1.0:**

* Description: As a user, I would like to see my user profile, in order to see my current profile information.

Acceptance Criteria

* Display user profile information.
* Display user profile picture and background picture.
* Display list of itineraries.

**Use Case**

* Name: User wants to see their profile.
* Actor: User with an account.
* Preconditions: App needs to be installed. Only signed in users can see profile.
* Description: A signed in user opens the app and sees the User Profile button. They click it and the app navigates to the User Profile page and loads the user’s data. They can click on the About tab and the Itinerary Tab.

**User Story Name: Redesign Login Page 2.0**

* Description: As a user, I would like to login in order to see my profile

Acceptance Criteria

* Login functionality with error responses.
* Register functionality with error responses.

**Use Case**

* Name: User wants to login.
* Actor: User
* Preconditions: App needs to be installed.
* Description: A user with an account opens the app and signs in using user credentials or creates an account.

## Pending User Stories

* 690 Set Cloud 9 for peer programming

# Project Plan

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

## Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

***Hardware:***

* A phone running and optimized for Android, but also runs on iOS.
* A computer running windows.

***Software:***

**Front end:**

* React-Native v16.3.0-alpha.1
* Has-flag v3.0.0
* React-native-action-button v2.8.4
* React-native drawer v2.5.0
* React-native-elements v0.19.0
* React-native-fetch-blob v0.10.8
* React-native-overlay v0.5.0
* React-native-image-resizer v1.0.0
* React-native-modal v5.2.0
* React-native-router-flux v4.0.0-beta.28
* React-native-svg v6.2.2
* React-native-vector-icons v4.5.0
* React-redux v5.0.6
* React-router-flux v1.0.0
* Redux thunk v2.2.0

**Middle-ware:**

* Redux v3.7.2

**Back end:**

* Firebase v4.9.0

**Other:**

* Git v2.16.1

## 

## 

## Sprints Plan

### Sprint 1

N/A

### Sprint 2

Attendees: Eduardo, Miguel, Luis, Noel, Gino

Start time: 01/30/18 10:25am

End time: 01/30/18 10:25am

After discussion, the velocity of the team were estimated to be 112.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story 696 - Create Home Page
* User Story 671 - Create User Profile Page
* User Story 694 - Consolidate Viewing Itinerary features
* User Story 695 - Consolidate Create Itinerary features
* User Story 699 - Create Event Location Hierarchy
* User Story 701 - Guide hub that lists relevant guides
* User Story 702 - A guide template for each guide
* User Story 697 - Homepage interactive map

The team members indicated their willingness to work on the following user stories.

* Eduardo
* User Story 696 - Create Home Page
* User Story 697 - Homepage interactive map
* User Story 701 - Guide hub that lists relevant guides
* User Story 702 - A guide template each guide

### Sprint 3

Attendees: Miguel, Eduardo, Noel

Date: 02/27/18

Start time: 10:20am

End time: 11:00am

After discussion, the velocity of the team were estimated to be 112.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story 713 Redesign Login Page 1.0
* User Story 714 Redesign User Profile 1.0
* User Story 674 Create Option to Add User Profile Picture
* User Story 697 Homepage interactive map

The team members indicated their willingness to work on the following user stories.

* Miguel
* User Story 713 Redesign Login Page 1.0
* User Story 714 Redesign User Profile 1.0
* User Story 674 Create Option to Add User Profile Picture
* Eduardo
* User Story 697 Homepage interactive map
* Noel
* User Story 711 Regions/Prefects/Cities/Itineraries Overhaul
* Gino
* User Story 715 Create and edit blogs (for developers)

### Sprint 4

Attendees: Miguel, Eduardo, Gino, Luis, Noel

Start time: 03/20/18 10:00am

End time: 03/20/18 10:00am

After discussion, the velocity of the team were estimated to be <Enter the estimated team velocity>.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story 674 Create Option to Add User Profile Picture
* User Story 719 Redesign Login Page 2.0
* User Story 720 Redesign User Profile 2.0

The team members indicated their willingness to work on the following user stories.

* Miguel
* User Story 719 Redesign Login Page 1.0
* User Story 720 Redesign User Profile 1.0
* User Story 674 Create Option to Add User Profile Picture
* Eduardo
* User Story 721 Sample itineraries on homepage
* …
* Gino
* User Story 726 Refine and Reformat guides
* User Story 727 Connect guides to database
* …
* Luis
* User Story 731 Implement Create Itinerary
* …

### Sprint 5

Attendees: Eduardo, Gino, Luis, Miguel

Start time: 10:00am

End time: 10:50am

After discussion, the velocity of the team were estimated to be 104.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

* User Story 724 Manage itineraries.
* User Story 723 Start Itinerary functionality
* User Story 721 Sample itineraries on homepage
* User Story 725 Favorite itineraries
* User Story 720 Redesign User Profile 2.0
* User Story 718 Complete Itinerary Core feature
* User Story 727 Connect guides to database
* User Story 726 Refine and Reformat guides

The team members indicated their willingness to work on the following user stories.

* Eduardo
* User Story 723 Start Itinerary functionality
* User Story 721 Sample itineraries on homepage
* Gino
* User Story 727 Connect guides to database
* User Story 726 Refine and Reformat guides
* Luis
* User Story 724 Manage itineraries.
* User Story 725 Favorite itineraries
* Miguel
* User Story 720 Redesign User Profile 2.0

### Sprint 6

### 

N/A

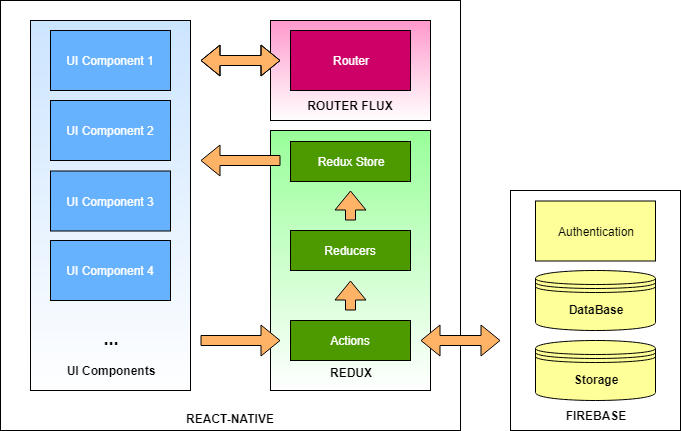
# System Design

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

## Architectural Patterns

The application uses a combination of React Native, React-Redux, React Native Router Flux, and firebase as a database, data store, and authentication service. This architecture separates UI, application logic in actions, application state (data) in the redux store, view switching in the router, and firebase services.

The flow of the application is as follows: UI components call Redux Actions. Redux actions perform queries on the firebase database, store and retrieve files in the firebase storage, and authenticates users through firebase. These Redux Actions also call Redux Reducers. Reducers modify the application state and store the state in the Redux Store. The Redux Store listens for changes in the application state. When the application state changes, the Redux Store rerenders UI components reliant on the application state.



## System and Subsystem Decomposition

The system is made up of two main subsystems, Firebase back-end and the React-Native client application. The subsystems are decomposed into its functional components

Firebase back-end:

* Authentication service that validates users.
* Database that stores user, itinerary and blog data.
* Storage that hold image files for Users and itineraries.

React-Native client application:

* UI components that dispatch redux actions.
* Redux Actions that either change the reduces or access firebase.
* Redux Reducers that update the data store when changed.
* Redux data store which updates the UI components when modified.
* Router flux navigation router that handles which UI components are displayed.

## Deployment Diagram

N/A

## Design Patterns

The following design patterns were used in the application.

**Redux Data Store Handling:** This is a core design pattern in our application. Redux allows data to be globalized to avoid having to pass data to components individually, and rerenders components when this global state is changed.

**React Native Router Flux:** Scenes can be navigated to from any part of the app. The scenes navigated to are pushed to a stack and Actions.Pop( ) returns you to the previous scene.

**Upload Picture Custom Component:** All images uploaded to firebase storage were uploaded using the Upload Picture component. Javascript is a single-threaded language which causes issues when functions are run asynchronously. The Upload Picture component ensures that images are stored before they are fetched through a series of function callbacks.

# System Validation

N/A

# 

# 

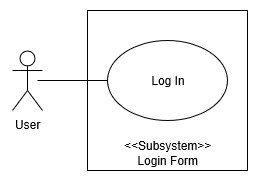
# 

# 

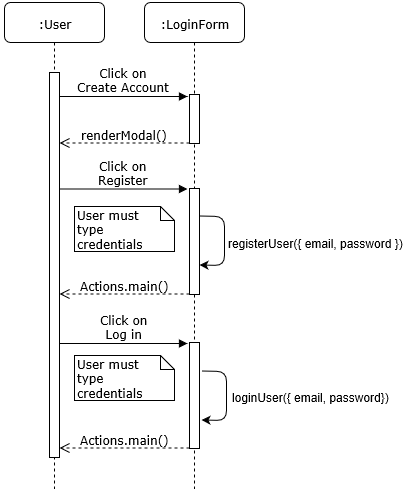
# Glossary

# Appendix

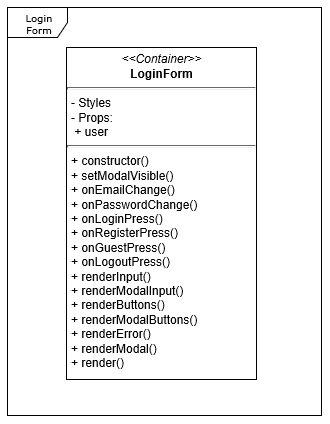
## Appendix A - UML Diagrams



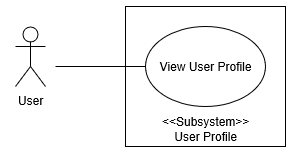
Login Form Use Case



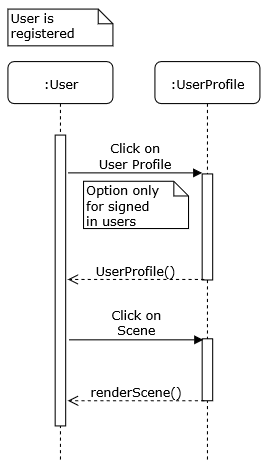
Login Form Sequence Diagram



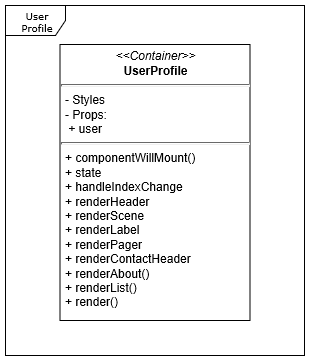
Login Form Class Diagram



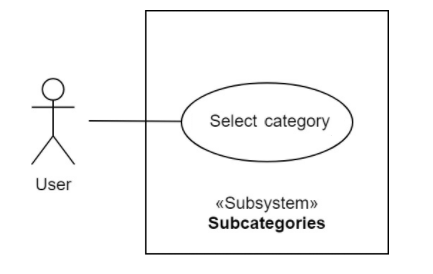
User Profile Use Case



User Profile Sequence Diagram



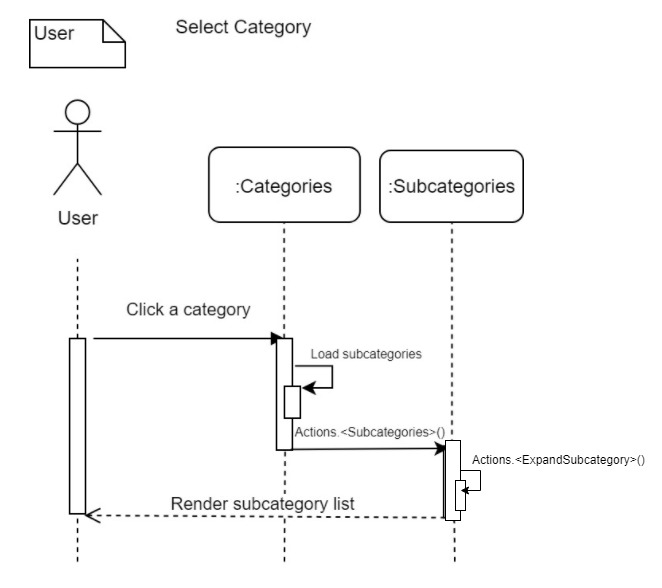
User Profile Class Diagram



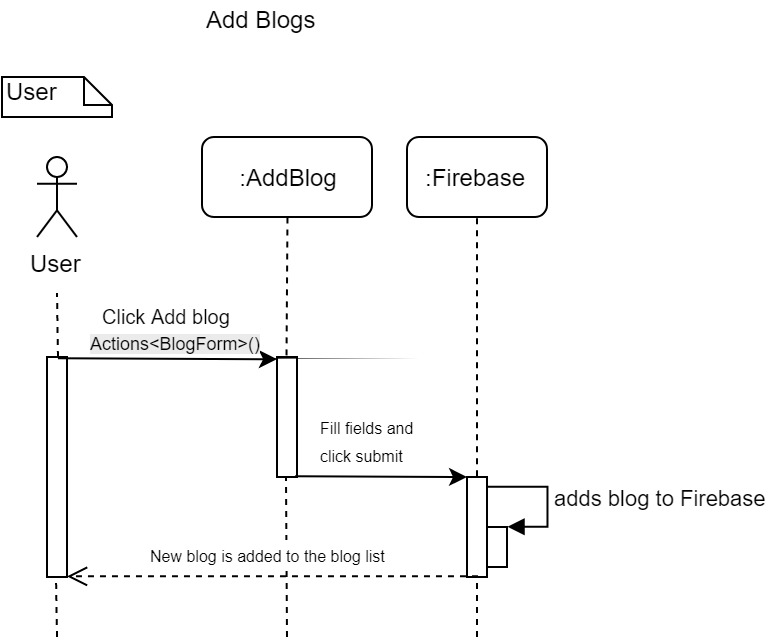
Select Category Use Case

## 

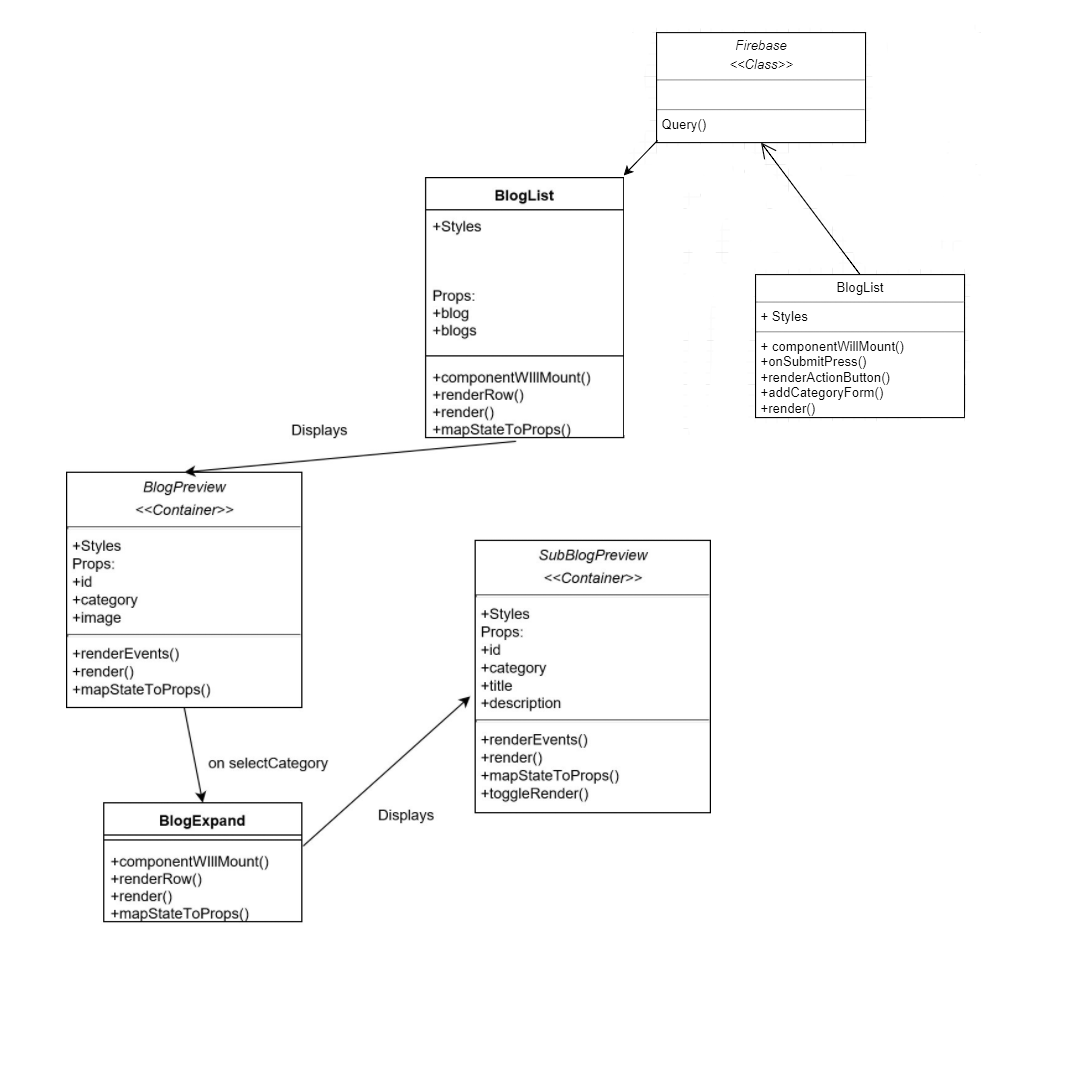
Can write a blog to firebase Use Case

****

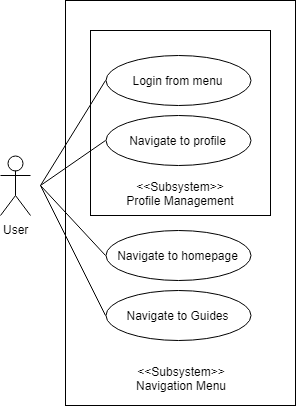
Select Category Sequence diagram



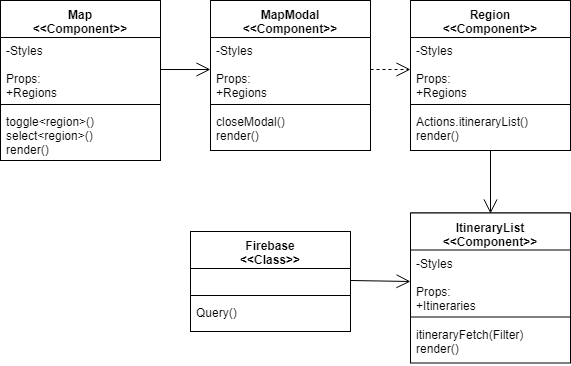
Can write a blog to firebase Sequence diagram

****

Blogs Class Diagram

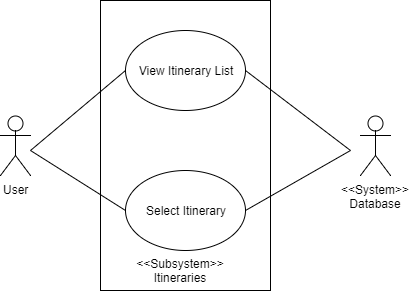


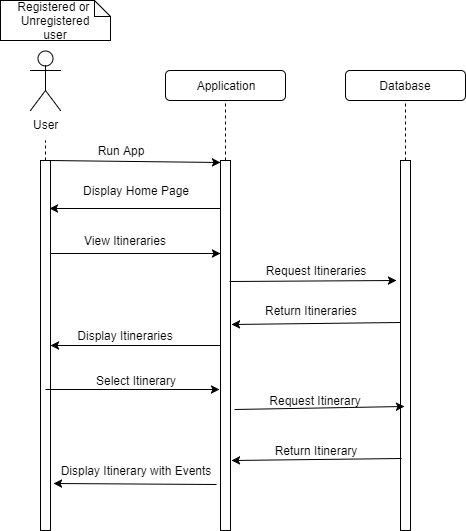
Use Case Diagram Navigation

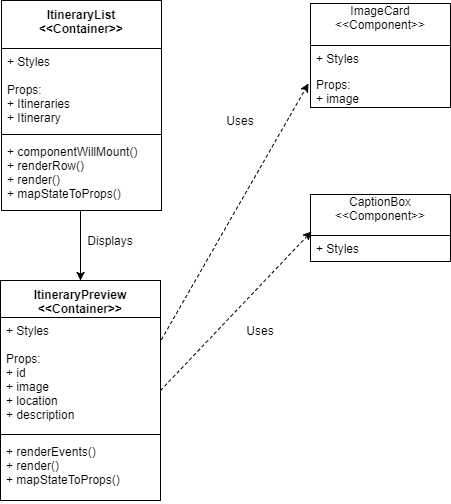


Class diagram Map

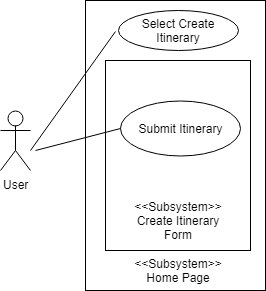
**Use Case 694:**

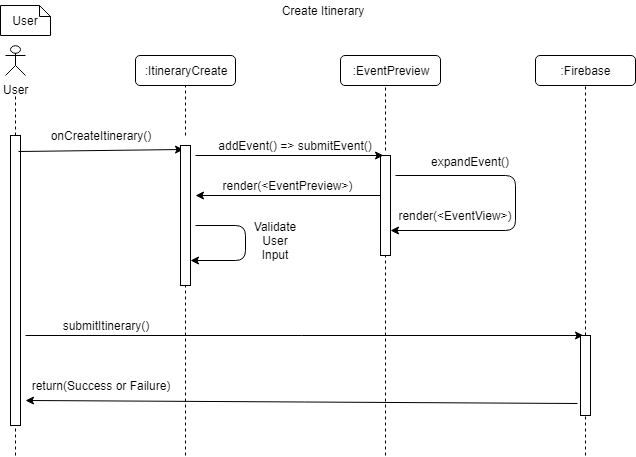


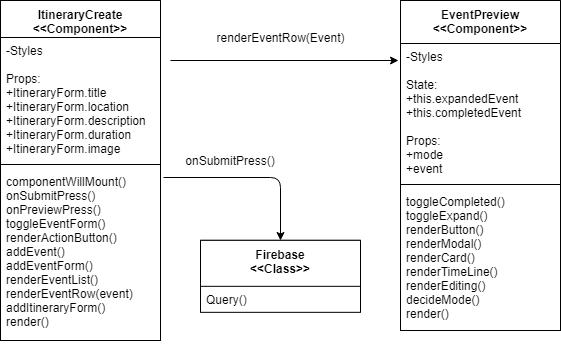


****

**Use Case 722**

****





## 

## Appendix B - User Interface Design

For the UI of the application, we settled on Material Design. Material design (MD) is a set of design guidelines developed by Google in 2014 mainly for the android platform. MD defines how to design UI components and other style principals through specific rules. By using MD, an application is guaranteed to have a consistent UX. MD is a standard used by all of Google's mobile applications, such as gmail, google home or google drive. MD features a variety of layouts and styles to follow, so that a developer may still have creative freedom with the UI, the guidelines are more for maintaining a consistent UX. MD is also constantly being updated as stated by the introduction: "This spec is a living document that will be updated as we continue to develop the tenets and specifics of material design." \cite{Google} so the guidelines are still subject to change.

## 

## Appendix C - Sprint Review Reports

**Sprint 1**

Attendees: Eduardo, Miguel, Luis, Noel

Start time: 01/30/18 10:25am

End time: 01/30/18 10:45am

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

* User Story All

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Spring Planning meeting.

* None

**Sprint 2**

Attendees: Eduardo, Luis, Miguel, Noel

Start time: 02/13/18 9:12am

End time: 02/13/18 10:00am

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* User Story 696 - Create Home Page
* User Story 671 - Create User Profile Page
* User Story 694 - Consolidate Viewing Itinerary features
* User Story 695 - Consolidate Create Itinerary features
* User Story 699 - Create Event Location Hierarchy
* User Story 701 - Guide hub that lists relevant guides
* User Story 702 - A guide template for each guide
* User Story 697 - Homepage interactive map

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* None

**Sprint 3**

Attendees: Eduardo, Miguel, Noel

Start time: 9:24AM

End time: 10:00am

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* User Story 709 Home page Navigation
* User Story 672 Create Option to Edit User Profile
* User Story 706 Modify User Profile
* User Story 705 Modify Login Page

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* User Story 674 Create Option to Add User Profile Picture
* How this should be reflected on the user story definition in Mingle:
  + Moved to sprint 4
* User Story 708 Modify Blog/Guide feature
* How this should be reflected on the user story definition in Mingle:
  + Refined to a new user story that was moved to sprint 4
* User Story 710 Modify Itinerary Feature Functionality
* How this should be reflected on the user story definition in Mingle:
  + Move to a new sprint
* User Story 711 Regions/Prefects/Cities/Itineraries Overhaul
* How this should be reflected on the user story definition in Mingle:
  + Refined and moved to sprint 4

**Sprint 4**

Attendees: Eduardo, Luis, Gino, Noel, Miguel

Start time: 03/19/18 10:10pm

End time: 03/19/18 11:??pm

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* User Story 697 Homepage Interactive Map
* User Story 718 Complete Itinerary Core feature
* User Story 713 Redesign Login Page 1.0
* User Story 714 Redesign User Profile 1.0

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* User Story 674 Create Option to Add User Profile Picture
* How this should be reflected on the user story definition in Mingle:
  + Moved to sprint 5
* User Story 711 Regions overhaul
* How this should be reflected on the user story definition in Mingle:
  + Moved to backlog
* User Story 715 Subcategory list
* How this should be reflected on the user story definition in Mingle:
  + Moved to Sprint 5 along with reformatting

**Sprint 5**

Attendees: Eduardo, Luis, Miguel

Start time: 9:20am

End time: 10:00am

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* User Story 719 Redesign Login Page 2.0
* User Story 728 Create splash art
* User Story 717 Regions Map
* User Story 722 Link itinerary functionality to firebase

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* N/A

**Sprint 6**

Attendees: Gino, Luis, Miguel, Eduardo

Start time: 10:00AM

End time: 11:00AM

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners:

* User Story 720 Redesign User Profile
* User Story 725 Favorite itineraries
* User Story 723 Start Itinerary functionalities
* User Story 721 Sample itineraries homepage
* User Story 718 Complete Itinerary Core feature
* User Story 727 Connect Guides to Database
* User Story 729 Input Validation for Create Itinerary
* User Story 730 Image Picker for Itineraries

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* N/A

## 

## Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

### Installation Guide

1. Install Node.js
   1. <https://node.js.org/en/download>/
2. Install Expo for PC and mobile device
   1. <https://expo.io/>
3. Install React Native
   1. <https://github.com/react-community/create-react-native-app>
4. Open terminal and navigate to appropriate directory
5. After installation, add the Nihon\_GO project files from Github on the computer.
6. Navigate to project directory in console and run the following:
   1. rm -rf node\_modules
   2. npm install
7. Open Expo XDE on the computer
8. Click Open Project and navigate to the project directory
9. Once project opens, the app will start
10. Click Share to see the QR code
11. Open the Expo app on phone
12. Scan QR code
13. App will build and load on phone
14. If an issue with Node Modules occurs, restart from step 6.

# References

Node JS

<https://node.js.org/en/download>/

Expo

<https://expo.io/>

React Native

<https://github.com/react-community/create-react-native-app>