Lab 2 : A Team

Deliverables:

* Refined Use Case Description (Pages 1 to 6)
* Refined Use case Diagram (Page 7)
* Classes (Pages 8 to 9)
* Class Diagram (Page 9)
* Sequence Diagram (Page 10)
* Dialog Map (Page 13)

1. Refined Use Case Description

Use Case Template

| Use Case ID: | 1 | | |
| --- | --- | --- | --- |
| Use Case Name: | Login/Sign up | | |
| Created By: | Yi Shen | Last Updated By: | Irfan, Nithyasri |
| Date Created: | 26/1/2023 | Date Last Updated: | 16/2/2023 |

| Actor: | User, Registered User Database |
| --- | --- |
| Description: | Allows users to login/signup to the application. |
| Preconditions: | User is not logged in to the system |
| Postconditions: | User will logged into the system |
| Priority: | 1 |
| Frequency of Use: | Always |
| Flow of Events: | 1. Users will key in their username and password. 2. Registered User Database will verify details 3. User will be logged in |
| Alternative Flows: | 1. Users will click the sign-up button. 2. Users will key in an email, username, password and reconfirmed password. 3. An email verification process is done 4. Users will be redirected to the login page. |
| Exceptions: | EX1: If the user has keyed in the wrong username and password   1. Registered User Database will reject details 2. User will be prompted to try again   EX2: If the user has keyed in the wrong username and password three times   1. Registered User Database will reject details 2. User will be shown and error message |
| Includes: | NIL |
| Special Requirements: | NIL |
| Assumptions: | User must have an email account |
| Notes and Issues: | NIL |

Use Case 2

| Use Case ID: | 2 | | |
| --- | --- | --- | --- |
| Use Case Name: | Manage Favourite Locations | | |
| Created By: | Yi Shen | Last Updated By: | Yi Shen |
| Date Created: | 26/1/2023 | Date Last Updated: | 26/1/2023 |

| Actor: | User, Registered User Database |
| --- | --- |
| Description: | Allows users to manage favourite locations |
| Preconditions: | Users must be logged in |
| Postconditions: | Update list of favourited locations will be saved into Registered User Database |
| Priority: | 1 |
| Frequency of Use: | Seldom |
| Flow of Events: | 1. User choose a location 2. User will press ‘Favourite Location’ 3. RouteManager will call edit(Location) to Registered User Modeller 4. Location will be favourited |
| Alternative Flows: | 1. User choose a location which has been favourited 2. User will press ‘Unfavourite Location’ 3. RouteManager will call edit(Location) to Registered User Modeller 4. Location will be unfavourited |
| Exceptions: | NIL |
| Includes: | NIL |
| Special Requirements: | NIL |
| Assumptions: | The location is a valid location in Singapore |
| Notes and Issues: | NIL |

Use Case 3

| Use Case ID: | 3 | | |
| --- | --- | --- | --- |
| Use Case Name: | Display Fares | | |
| Created By: | Yi Shen | Last Updated By: | Yi Shen |
| Date Created: | 26/1/2023 | Date Last Updated: | 26/1/2023 |

| Actor: | User, Fare API |
| --- | --- |
| Description: | Allows users to see the current fares for the various modes of transport |
| Preconditions: | Users must be logged in |
| Postconditions: | A table with various fares will be displayed to the User |
| Priority: | 1 |
| Frequency of Use: | Seldom |
| Flow of Events: | 1. User will press the “Fares” option 2. System will use Fare API to get the updated fares for bus/train and taxi 3. System will display a User-Friendly table of current fare prices |
| Alternative Flows: | NIL |
| Exceptions: | NIL |
| Includes: | NIL |
| Special Requirements: | NIL |
| Assumptions: | NIL |
| Notes and Issues: | NIL |

Use Case 4

| Use Case ID: | 4 | | |
| --- | --- | --- | --- |
| Use Case Name: | Determine Route | | |
| Created By: | Yi Shen | Last Updated By: | Yi Shen |
| Date Created: | 26/1/2023 | Date Last Updated: | 26/1/2023 |

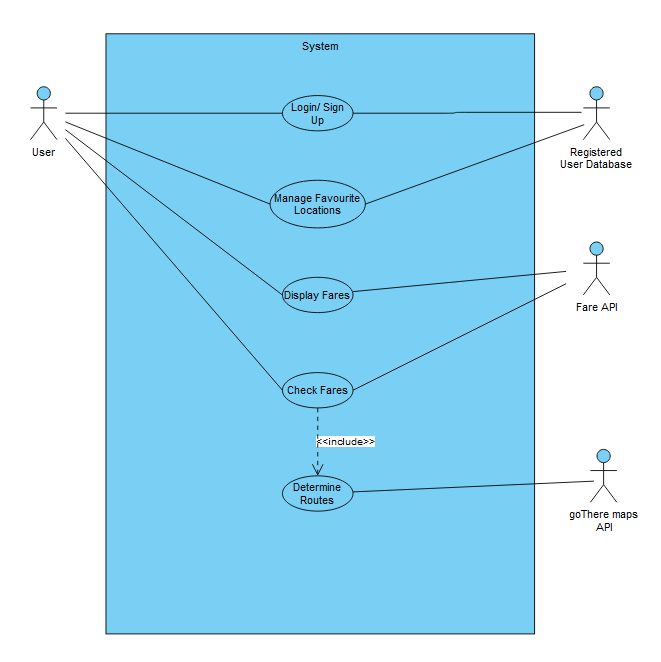
| Actor: | User, goThere maps API |
| --- | --- |
| Description: | Determine routes from origin to destination either by public transport or taxi |
| Preconditions: | 1. Users must be logged in. |
| Postconditions: | Routes for public transport and taxi will be determined |
| Priority: | 1 |
| Frequency of Use: | Always |
| Flow of Events: | 1. User will choose origin and destination 2. System uses goThere maps API to determine the routes for public transport and taxi |
| Alternative Flows: | NIL |
| Exceptions: | NIL |
| Includes: | NIL |
| Special Requirements: | NIL |
| Assumptions: | 1. The locations are based in Singapore |
| Notes and Issues: | NIL |

Use Case 5

| Use Case ID: | 5 | | |
| --- | --- | --- | --- |
| Use Case Name: | Check Fares | | |
| Created By: | Yi Shen | Last Updated By: | Yi Shen |
| Date Created: | 26/1/2023 | Date Last Updated: | 26/1/2023 |

| Actor: | Fare API, goThere maps API |
| --- | --- |
| Description: | Determine routes from origin to destination either by public transport or taxi |
| Preconditions: | 1. Users must be logged in.  2. Routes for public transport and taxi will be determined |
| Postconditions: | Routes and fares of public transport and taxi will be displayed |
| …Priority: | 1 |
| Frequency of Use: | Always |
| Flow of Events: | 1. System will call Fare API to get the current fares for taxi and public transport 2. System will calculate the fares for each route using current fares 3. System will display routes and fares for each route |
| Alternative Flows: | NIL |
| Exceptions: | NIL |
| Includes: | Determine Routes |
| Special Requirements: | NIL |
| Assumptions: | NIL |
| Notes and Issues: | NIL |

2. Use case diagram:



3. Classes:

* **Viewer Classes (Boundary)**

MainMenuUI

GoThereMapViewer

* **Model Classes (Entity)**

TaxiModeller

PublicTransportModeller

RegisteredUsersModeller

goThereMapModeller

* **Controller Classes**

AccountManager

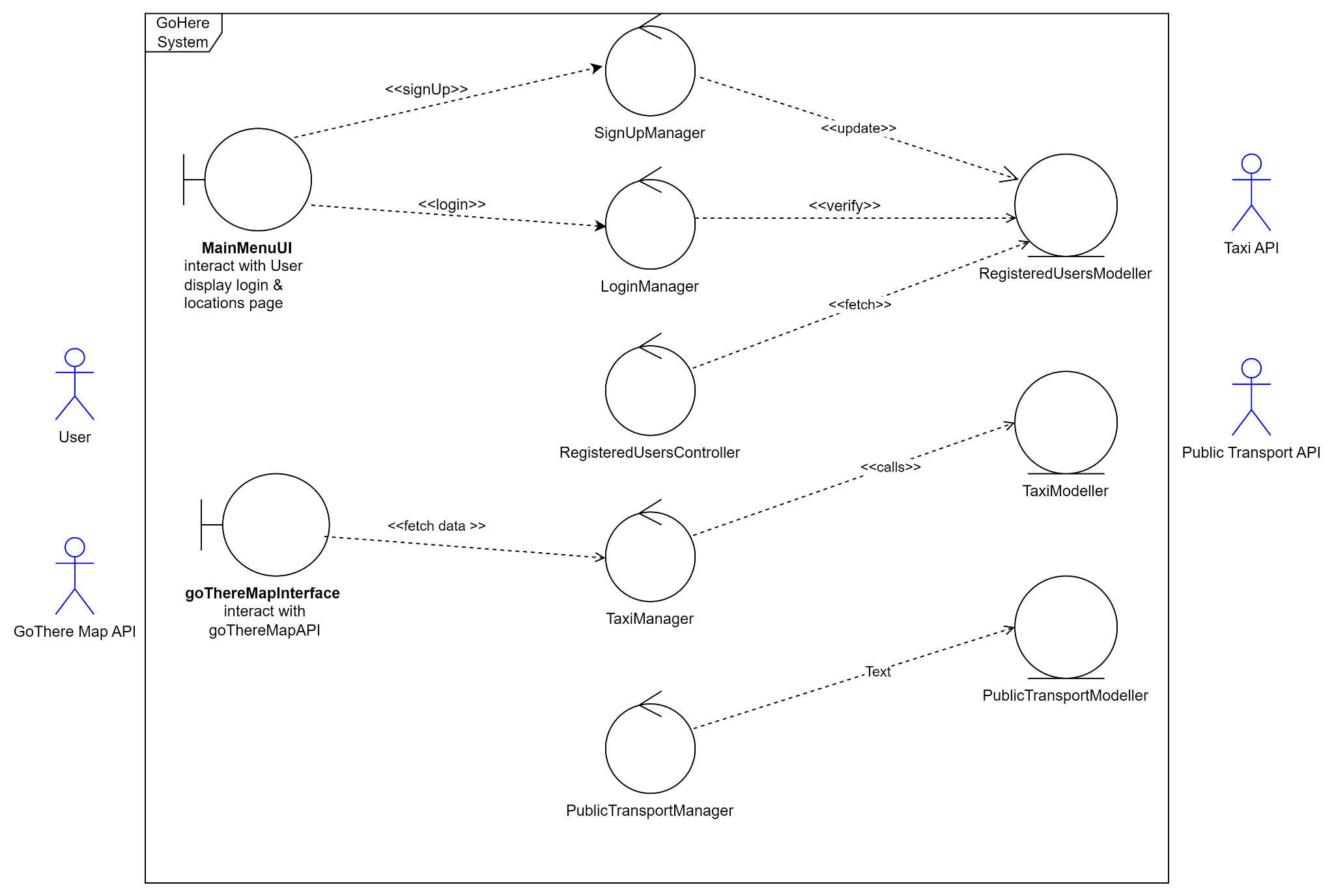
TaxiManager

PublicTransportManager

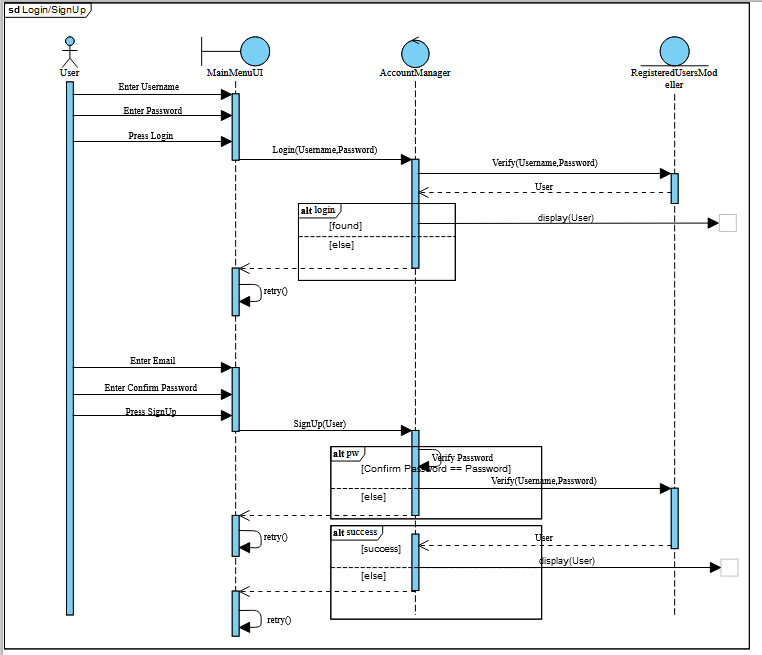
RouteManager

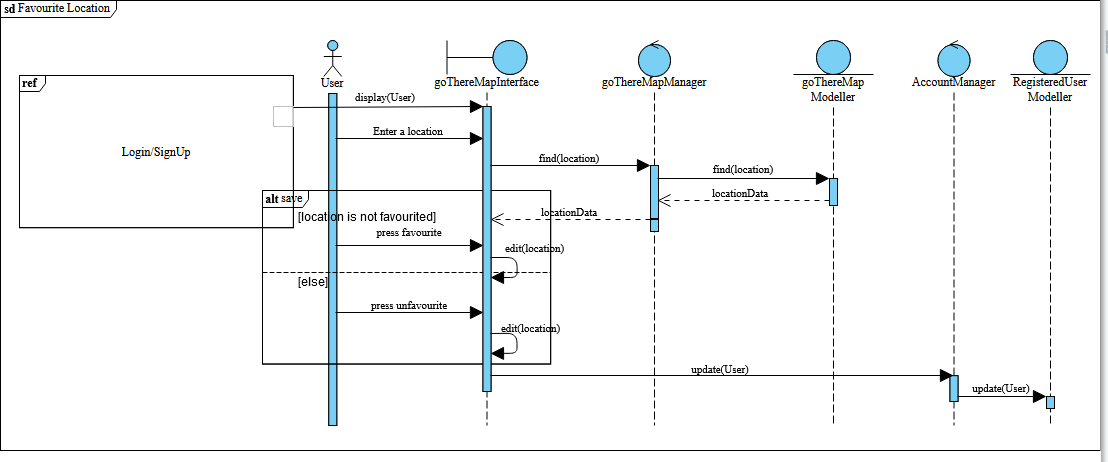
goThere MapManager

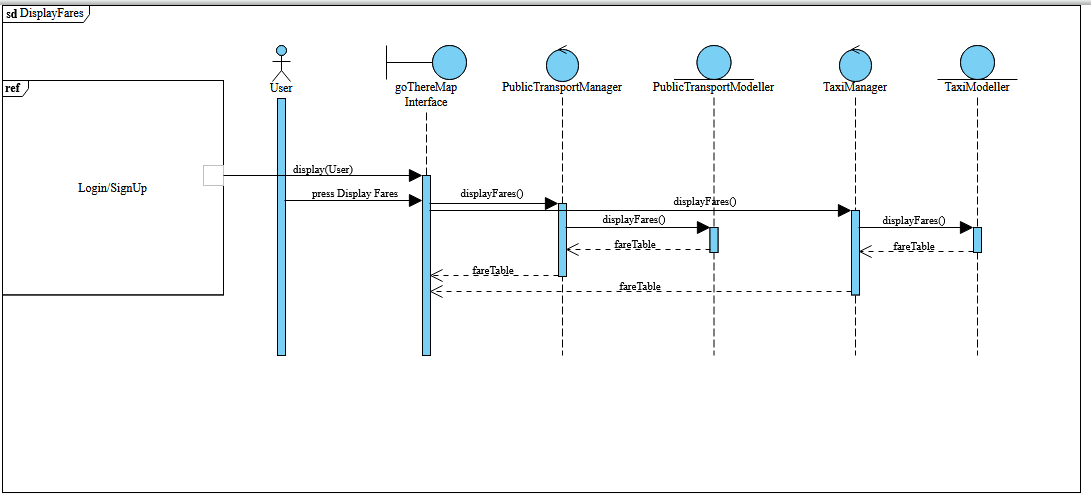
4. Class Diagram

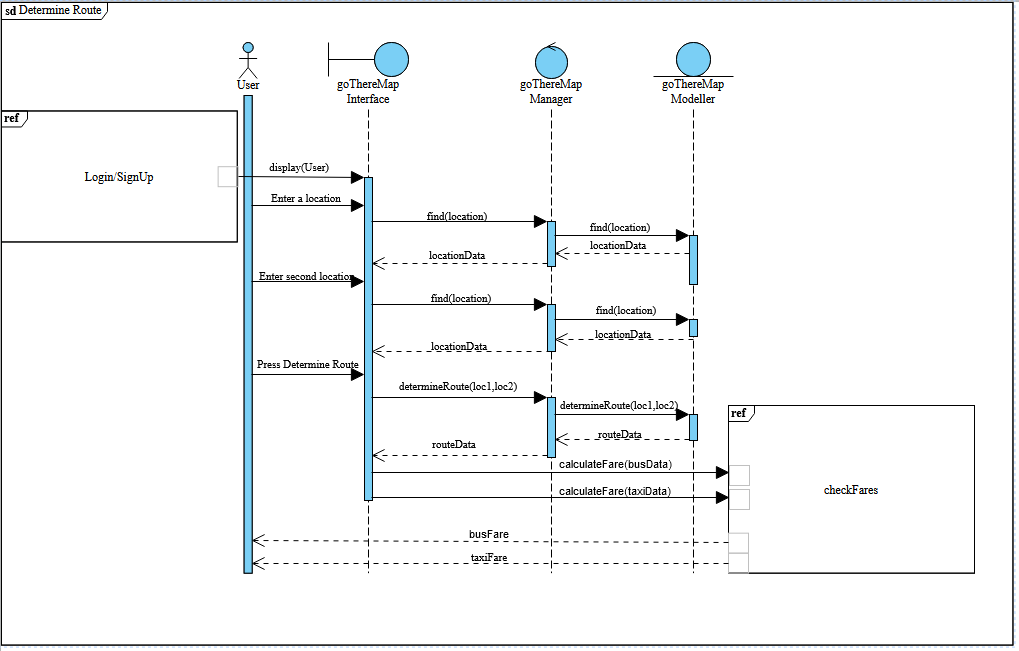


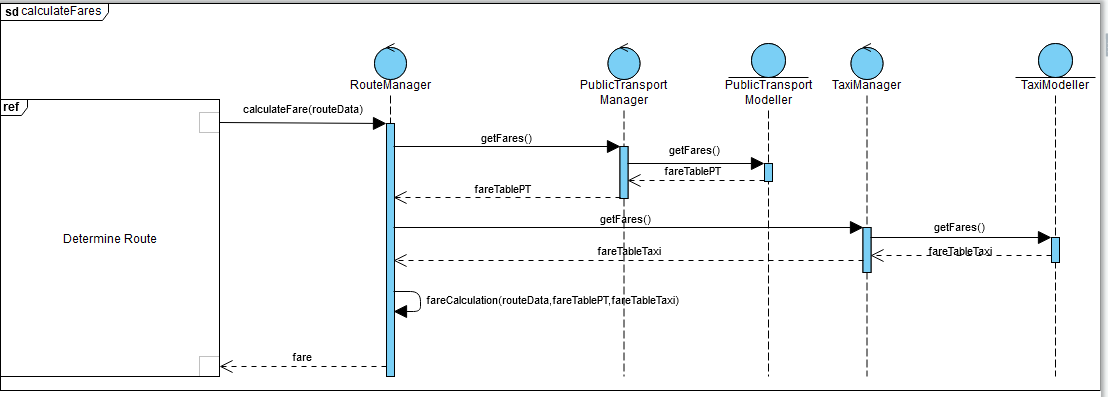
5. Sequence Diagram:





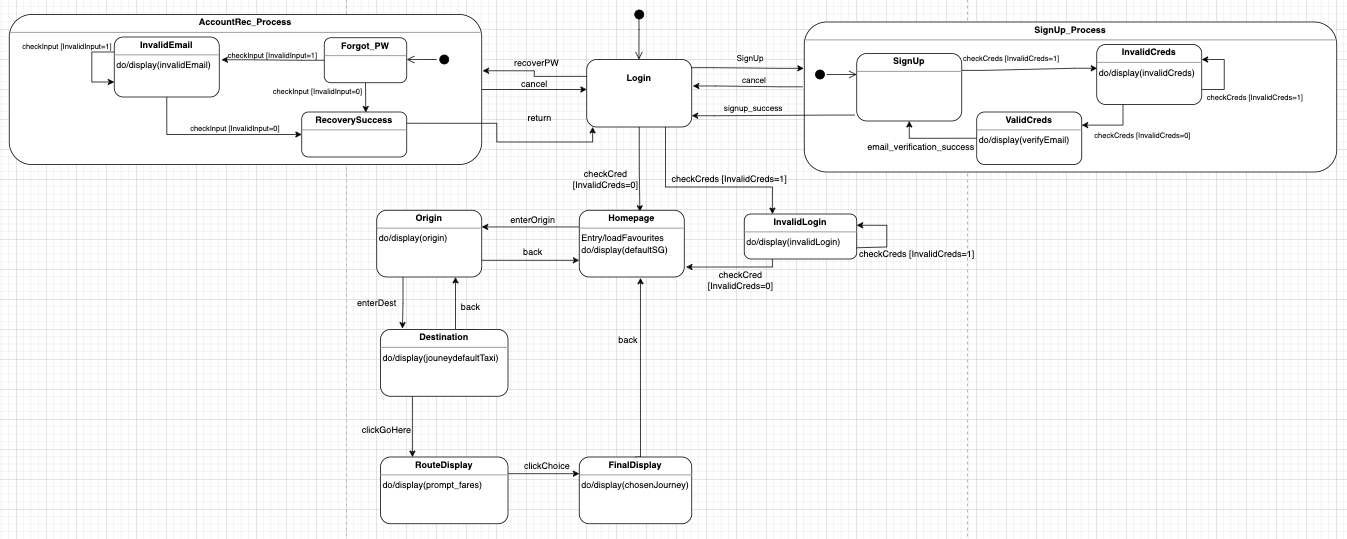






<https://online.visual-paradigm.com/share.jsp?id=323335323633342d37>

6. Dialog Map:



<https://app.diagrams.net/#G1FdfFKxyP2Mcjk6QXLlxh-KbmGHxPLuh6>