**Checklist**

* Are all functional requirements covered?
* Are all non-functional requirements covered?
* Do we mention which routes “extends” or “include” other routes?
* Q: When do the customers agree to share their current location?
* Note: Marked green when done, marked yellow if not completed
* Q: Shouldn’t Route-4 Add Favorite extend only the Route-1??
* Note: We should redraw the Use Case Diagram so that Recalculate Route extends Start a Route (not the other way around).
* Note: Have no use case for view Favorites
* Note: Have no use case for view History
* Check that “Actors” are correct, shouldn’t several have both User and System, and declaration of which is initiating?

# Route-1: Enter route information

Alternative to Route 1:

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-1 | | |
| Use Case Name: | Enter route information | | |
| Created By: | Isabel | Last Updated By: | Isabel |
| Date Created: | 2019-02-25 | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | The User enters the information for the journey, including mode of transportation, origin, destination, and time of departure/arrival. |
| Preconditions: | The user has downloaded the application. |
| Postconditions: | By default, the fastest route will be calculated. |
| Priority: | High |
| Frequency of Use: | Once a week |
| Flow of Events: | 1. The user selects the option to “Plan My Trip”  2. The system prompts the user with options of “Bus” and “MRT”  3. The user checks the wanted options and selects the Arrow icon to move to the next step  4. The system asks the user for the starting point and the ending point  5. User selects the starting point fill-in box and enters in the starting address  6. System verifies that it is a valid address with Google Maps API  7. If address is verified by the system, the system prompts the user for the address of the destination.  8. User selects the ending point fill-in box and enters the ending destination  9. System verifies that it is a valid address with Google Maps API  10. If address is verified by the system, the user may proceed  11. User selects the Arrow Icon to proceed  12. System asks the user for time and date and option of “Time of Arrival” or “Time of Departure”  13. User fills in time and date and selects the wanted option.  14. User selects the Arrow Icon to proceed. |
| Alternative Flows: | AFS-7 & 10:If address entered cannot be verified   1. The system displays the message “Location is not found. Please try again.” 2. System returns to previous step ( ie step 6 if AFS7, step 9 if AFS 10)   AFS5: If the user uses GPS to set starting location   1. User selects the “Current Location” option in the drop down menu 2. System converts live location to a Geolocation with the Google Maps API 3. System returns to step 8   AFS5 & 8: If the user uses a previous Favourite Location   1. User selects a previous Favourite Location option in the drop down menu 2. System retrieves the geolocation of the Favourite Location from database 3. System returns to the next step   AFS13: If the user adds to Favorites   1. User clicks on “Add to Favorites” 2. System prompts with “Route added to Favorites” 3. System returns to step 14 |
| Exceptions: |  |
| Includes: | ROUTE-2 Calculate Fastest Routes |
| Special Requirements: | 1. 99 % of users must be able to complete ROUTE-1, ROUTE-2, and possibly ROUTE-3 within 2 minutes of starting the system |
| Assumptions: | System would always be able to find a route from users origin to their destination, given that the locations can be verified. |
| Notes and Issues: | If the user is planning a journey when MRT/bus services are not available (e.g outside of operating hours), the system will show the next possible route (e.g when services start operating again). |

# Route-2: Calculate Fastest Routes

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-2 | | |
| Use Case Name: | Calculate Fastest Routes | | |
| Created By: |  | Last Updated By: | Isabel |
| Date Created: |  | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | System |
| Description: | Displaying for the user the fastest way to get from origin to destination by bus and/or MRT |
| Preconditions: | The user must have completed Use Case ROUTE-1 |
| Postconditions: | Display the various routes and their timing needed, sorted by time. |
| Priority: | High |
| Frequency of Use: | Once a week |
| Flow of Events: | 1. The system requests multiple alternative routes from   Google Maps API   1. The system reads the estimated duration time required for each possible route 2. The system lists possible routes sorted by time, ascending |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: | This use case includes ROUTE-6 |
| Special Requirements: | 1. 99 % of users must be able to complete ROUTE-1, ROUTE-2, and possibly ROUTE-3 within 2 minutes of starting the system 2. System shall be able to identify the fastest route within 1 minute |
| Assumptions: | System would always be able to find a route from users origin to their destination |
| Notes and Issues: | If the user is planning a journey when MRT/bus services are not available (e.g outside of operating hours), the system will show the next possible route (e.g when services start operating again). |

# Route-3: Calculate Cheapest Route

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-3 | | |
| Use Case Name: | Calculate Cheapest Route | | |
| Created By: |  | Last Updated By: | Isabel |
| Date Created: |  | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | Displaying for the user the cheapest way to get from origin to destination by bus and/or MRT. |
| Preconditions: | This use case extends ROUTE-2 and can be performed after it has finished. |
| Postconditions: | Display the various routes and their associated travel time, sorted by price. |
| Priority: | High |
| Frequency of Use: | Once a week |
| Flow of Events: | 1. When ROUTE-2 is finished, the system displays the possible routes sorted by time  2. The user selects the option to sort route of lists by price  3. The system calls to database to retrieve stored fares  4. The system calculates the fare based on mode of transportation(s) and distance  5. The system returns the list of possible routes sorted by price, ascending |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: | 1. 99 % of users must be able to complete ROUTE-1, ROUTE-2, and possibly ROUTE-3 within 2 minutes of starting the system 2. System shall be able to identify the cheapest route within 1 minute |
| Assumptions: | The system should always find fares for the suggested route. |
| Notes and Issues: |  |

# Route-4: Add favorite route

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-4 | | |
| Use Case Name: | Add favorite route | | |
| Created By: |  | Last Updated By: | Isabel |
| Date Created: |  | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User (initiating) & System |
| Description: | Allows the user to save routes as favorites for future use |
| Preconditions: | This use case extends ROUTE-2 and ROUTE-3. It is initiated when the user clicks on symbol next to route.  The user must have entered route information that has been verified. |
| Postconditions: | Route is added to Favourite list |
| Priority: | Low |
| Frequency of Use: | Once a month |
| Flow of Events: | 1. The system saves the origin, destination, mode of transport and specific calculated route to database 2. System displays a message or indication to the user to inform that the route has been added to favorites 3. The favorited route is now viewable in “Favorites” |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: | System response time must not exceed 1 minute |
| Assumptions: |  |
| Notes and Issues: |  |

# Route-5: Delete Route

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-5 | | |
| Use Case Name: | Delete Route | | |
| Created By: | Surabhi | Last Updated By: | Isabel |
| Date Created: | 10 March | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | Allowing the user to delete routes from Favorites |
| Preconditions: | This use case extends ROUTE-4.  The user must have at least one favorite route saved. |
| Postconditions: | Route is deleted from Favorites |
| Priority: | Low |
| Frequency of Use: | Once a month |
| Flow of Events: | 1. The user views favorited routes 2. The system shows a list of routes the user has favorited sorted by time 3. The user selects a route to be deleted 4. System displays a message “Route deleted from favourites” |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: | Users cannot delete a route from History. |

# Route-6: Start a Route

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-6 | | |
| Use Case Name: | Start a Route | | |
| Created By: |  | Last Updated By: | Isabel |
| Date Created: |  | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | Display information for selected route |
| Preconditions: | The user must have completed use case ROUTE-1 |
| Postconditions: |  |
| Priority: | High |
| Frequency of Use: | Daily |
| Flow of Events: | 1. The user selects their preferred route from list of routes. 2. The system uses live location of user and chosen route to call to Google Maps API. 3. System constantly updates the map shown based on changes in user location. 4. a) If the live location of user is near a bus stop and the chosen route includes taking a bus from that bus stop, the system would prompt for the next available bus in the LTA datamall API.   b) system displays the waiting time needed for the next bus  5. When destination is near, the system prompts the user using the included use case ROUTE-7 Alert alight. |
| Alternative Flows: |  |
| Exceptions: | EX1: User decides to cancel route   1. System displays “ You have chose to end the route” |
| Includes: | This use case includes ROUTE-7 |
| Special Requirements: | System response time must not exceed 1 minute |
| Assumptions: | GPS tracking is fairly accurate |
| Notes and Issues: |  |

# Route-7: Alert alight

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-7 | | |
| Use Case Name: | Alert alight | | |
| Created By: |  | Last Updated By: | Isabel |
| Date Created: |  | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | Alerts user when the next stop is their intended destination |
| Preconditions: | Invocation as an included case by ROUTE-6 Start a Route |
| Postconditions: | User has alighted at the correct stop |
| Priority: | High |
| Frequency of Use: | Daily |
| Flow of Events: | 1. The application uses the user’s location to determine if the user is approaching the intended stop  2. When application determines the user is approaching the intended stop, the application issues a notification to the user  3. User acknowledges notification by selecting the “Yes” option  4. System provides instructions on the next part of the user’s route |
| Alternative Flows: | AF-S3: User does not acknowledge and misses the stop   1. Application notifies user “You have missed stop” 2. User selects option to reroute 3. User is brought back to ROUTE-1   AF-S4: User has completed the route   1. Application notifies user “You have arrived at Your Destination!” |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: | On missing the transport, the app should recalculate a new route within 2 minutes. |
| Assumptions: | GPS tracking is fairly accurate |
| Notes and Issues: |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | | |
|  |  | | |
|  |  |  |  |
|  |  |  |  |

# Route-8: End a Route

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-8 | | |
| Use Case Name: | End a Route | | |
| Created By: | Surabhi | Last Updated By: | Isabel |
| Date Created: | 2019-03-12 | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | Allows the user to end a route. |
| Preconditions: | This use case extends ROUTE-6. |
| Postconditions: | Route has ended |
| Priority: | High |
| Frequency of Use: | Daily |
| Flow of Events: | 1. User has completed the route. 2. The system shows feedback “Route has been completed. Thank you! Have a nice day” 3. The system adds the route to history of journeys in database. |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: |  |
| Special Requirements: |  |
| Assumptions: |  |
| Notes and Issues: | From here, the user can start a new route |

# Route-9: Recalculate Route

|  |  |  |  |
| --- | --- | --- | --- |
| Use Case ID: | ROUTE-9 | | |
| Use Case Name: | Recalculate Route | | |
| Created By: | Surabhi | Last Updated By: | Isabel |
| Date Created: | 2019-03-10 | Date Last Updated: | 2019-03-12 |

|  |  |
| --- | --- |
| Actor: | User |
| Description: | User has missed a stop. |
| Preconditions: | User must be on a route and has not followed directions. This use case extends ROUTE-6 |
| Postconditions: | User is assigned a new route |
| Priority: | High |
| Frequency of Use: | Daily |
| Flow of Events: | 1. If a user has missed a route, system calculates a new route to destination. 2. The system displays new instructions for the user to follow. |
| Alternative Flows: |  |
| Exceptions: |  |
| Includes: | This use case includes ROUTE-2. |
| Special Requirements: | On missing the transport, the app should recalculate a new route within 2 minutes. |
| Assumptions: |  |
| Notes and Issues: |  |