Use Cases

for

Fast and Furious

Version 1.0 approved

Prepared by Brendon Tan, Hendy, Cheng, Clement, Ee Chern

NTU

9/3/2022

Revision History

Name	Date	Reason For Changes	Version

Guidance for Use Case Template

Document each use case using the template shown in the Appendix. This section provides a description of each section in the use case template.

1. Use Case Identification

1.1. Use Case ID

Give each use case a unique numeric identifier, in hierarchical form: X.Y. Related use cases can be grouped in the hierarchy. Functional requirements can be traced back to a labeled use case.

1.2. Use Case Name

State a concise, results-oriented name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. Include an action verb and a noun. Some examples:

- View part number information.
- Manually mark hypertext source and establish link to target.
- Place an order for a CD with the updated software version.

1.3. Use Case History

1.3.1 Created By

Supply the name of the person who initially documented this use case.

1.3.2 Date Created

Enter the date on which the use case was initially documented.

1.3.3 Last Updated By

Supply the name of the person who performed the most recent update to the use case description.

1.3.4 Date Last Updated

Enter the date on which the use case was most recently updated.

2. Use Case Definition

2.1. Actor

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor(s) that will be performing this use case.

2.2. Description

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

2.3. Preconditions

List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples:

1. User's identity has been authenticated.
2. User's computer has sufficient free memory available to launch task.

2.4. Postconditions

Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples:

1. Document contains only valid SGML tags.
2. Price of item in database has been updated with new value.

2.5. Priority

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. The priority scheme used must be the same as that used in the software requirements specification.

2.6. Frequency of Use

Estimate the number of times this use case will be performed by the actors per some appropriate unit of time.

2.7. Flow of Events

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, "How do I <accomplish the task stated in the use case name>?" This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system.

2.8. Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative course, and describe any differences in the sequence of steps that take place. Number each alternative course using the Use Case ID as a prefix, followed by "AC" to indicate "Alternative Course". Example: X.Y.AC.1.

2.9. Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use

case execution fails for some unanticipated reason. Number each exception using the Use Case ID as a prefix, followed by “EX” to indicate “Exception”. Example: X.Y.EX.1.

2.10. Includes

List any other use cases that are included (“called”) by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.

2.11. Special Requirements

Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.

2.12. Assumptions

List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.

2.13. Notes and Issues

List any additional comments about this use case or any remaining open issues or TBDs (To Be Determineds) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.

Use Case Template

Use Case ID:			
Use Case Name:	Role selection		
Created By:	Hendy	Last Updated By:	Brendon
Date Created:	3/9/2022	Date Last Updated:	6/9/2022

Actor:	User, Databases
Description:	At the start of our program, the user will be asked to choose a role. This choice will affect the subsequent prompts and functions the user will be able to access. As a driver, the user will be able to use the navigation and carpark function. As a commuter, the user will be able to check the arrival times of public transport which includes bus, MRT and LRT.
Preconditions:	1. The user's phone must be connected to the internet. 2. The user's phone must be connected to the GPS. 3. The user's phone must be an android.
Postconditions:	1. Database reads in JSON files only. 2. User is able to successfully choose a role
Priority:	
Frequency of Use:	5 times per day
Flow of Events:	1. User opens the application through an android phone 2. The app prompts the user to choose the role (driver/commuter) and the system uses included a use case to assign role 3. The system displays the appropriate user interface (UI) based on role chosen
Alternative Flows:	
Exceptions:	EX1: If the system crashes due to a bug 1. The application displays the message "App is temporarily unavailable, please try again later." 2. The application closes and returns the phone to the home screen. 3. The system will send an error message to and alert the admin about the situation
Includes:	Allocate role
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case Template

Use Case ID:			
Use Case Name:	Carpark selection		
Created By:	Hendy	Last Updated By:	Brendon
Date Created:	3/9/2022	Date Last Updated:	6/9/2022

Actor:	User, Databases, Admin
Description:	Once the user identifies as a driver, the user can use the find car-park function to look for carparks near a chosen location which the user can manually input into the system. The user will also be able to look for carparks near their current location.
Preconditions:	1. The user's phone must be connected to the internet. 2. The user's phone must be connected to the GPS. 3. The user's phone must be an android.
Postconditions:	1. Database reads in JSON files only. 2. User is able to see the information of the selected car parks
Priority:	
Frequency of Use:	5 times per day
Flow of Events:	1. User opens the application through an android phone 2. The app prompts the user to choose the role (driver/commuter) and the system uses included a use case to assign role 3. The system displays the appropriate UI based on role chosen 4. If the user selects "driver", then the system will prompt the user to select either (navigation/find car-park) options 5. If the user selects "find car-park", then the system will prompt the user to tap on (select carpark/ display nearby carpark), and then display the information of the car park

	6. If the user chooses the "select carpark" option. the system will prompt the user to input a location and then display nearby carparks around that location.
Alternative Flows:	AF-S5: If the user selects the star icon 1. The system will display the carparks saved by the user previously 2. The user can click on a saved carpark 3. The system will display the information of the selected carpark AF-S6: If the user inputs an invalid location 1. The system will display "Invalid location! Please enter another location." 2. The application returns to step 6 AF-S6: If the user selects the "display nearby carpark" option 1. The system will use the GPS to get the user's current location 2. The system will collect the information of nearby carparks and display it on the screen.
Exceptions:	
Includes:	Add as favorite
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case Template

Use Case ID:			
Use Case Name:	Navigation Selection		
Created By:	Hendy	Last Updated By:	Brendon
Date Created:	3/9/2022	Date Last Updated:	6/9/2022

Actor:	User, Databases, Admin
Description:	Once the user identifies as a driver, the user can use the navigation function to check the shortest ways to get to a destination of their choice. After which, the system will be able to guide the user to his destination through the use of the GPS. The user can also add his chosen destination as favorite to have faster access to it in the future.
Preconditions:	1. The user's phone must be connected to the internet. 2. The user's phone must be connected to the GPS. 3. The user's phone must be an android.
Postconditions:	1. Database reads in JSON files only. 2. User is able to use the GPS to get to his selected destination
Priority:	
Frequency of Use:	5 times per day
Flow of Events:	1. User opens the application through an android phone 2. The app prompts the user to choose the role (driver/commuter) and the system uses included a use case to assign role 3. The system displays the appropriate UI based on role chosen 4. If the user selects "driver", then the system will prompt the user to select either (navigation/find car-park) options 5. If the user selects "navigation", then the system will prompt the user to input starting and ending location and then display the shortest possible route to destination 6. The user clicks on "Go" 7. The system will guide the user to the selected destination through the GPS.
Alternative Flows:	AF-S5: If the user inputs invalid starting or ending location

	1. The application will display “Invalid location! Please enter another location.” 2. The application returns to step 5 AF-S5: If the user selects the star icon 1. The system will display the locations saved by the user previously 2. The user can click on a saved location 3. The system will display the shortest route to the selected location 4. The user can click on “Go” 5. The system will guide the user to the selected destination through the GPS.
Exceptions:	
Includes:	Add as favorite
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case Template

Use Case ID:			
Use Case Name:	Bus timing		
Created By:	Hendy	Last Updated By:	Brendon
Date Created:	3/9/2022	Date Last Updated:	6/9/2022

Actor:	User, Databases, Admin
Description:	Once the user identifies as a commuter, the bus function will be available to the user. The user can check the bus arrival timings to a chosen bus stop which can be manually inputted into the system. The user can add his selected bus stop(s) number as favorite to have faster access to them in the future.
Preconditions:	1. The user's phone must be connected to the internet. 2. The user's phone must be connected to the GPS. 3. The user's phone must be an android.
Postconditions:	1. Database reads in JSON files only. 2. User is able to look at the arrival time for the bus at the selected bus stops.
Priority:	
Frequency of Use:	5 times per day
Flow of Events:	1. User opens the application through an android phone 2. The app prompts the user to choose the role (driver/commuter) and the system uses included a use case to assign role 3. The system displays the appropriate UI based on role chosen 4. If the user selects "commuter", then the system will prompt the user to select either (find-bus-timing/find MRT/LRT) options 5. If the user selects "find-bus-timing", then the system will prompt the user to input bus-number and bus-stop 6. The system will display the arrival time of buses at the selected bus stop.
Alternative Flows:	AF-S4: If the user chooses to be a commuter when all public transport is terminated

	1. The application will display "There is no public transport available now, please use other forms of transport." 2. The application returns to step 2 AF-S5: If the user inputs invalid bus-stop number 1. The application will display "Invalid bus-stop number! Please enter another bus-stop number." 2. The application returns to step 5 AF-S5: If the user selects the star icon 1. The system will display the bus stops saved by the user previously 2. The user can click on a saved bus stop 3. The system will display the arrival time of buses at the selected bus stop.
Exceptions:	
Includes:	Add as favorite
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case Template

Use Case ID:			
Use Case Name:	MRT/LRT selection		
Created By:	Hendy	Last Updated By:	Brendon
Date Created:	3/9/2022	Date Last Updated:	6/9/2022

Actor:	User, Database, Admin
Description:	Once the system identifies as a commuter, MRT/LRT function will be available to the user, the user can check the MRT/LRT next arriving timing to certain MRT station
Preconditions:	1. The user's phone must be connected to the internet. 2. The user's phone must be connected to the GPS. 3. The user's phone must be an android.
Postconditions:	1. Database reads in JSON files only. 2. User is able to look at the arrival time for the train at the selected MRT/LRT stations.
Priority:	
Frequency of Use:	5 times per day
Flow of Events:	1. User opens the application through an android phone 2. The app prompts the user to choose the role (driver/commuter) and the system uses included a use case to assign role 3. The system displays the appropriate UI based on role chosen 4. If the user selects "commuter", then the system will prompt the user to select either (find-bus-timing/find MRT/LRT) options 5. If the user selects "find MRT/LRT", then the system will prompt the user to select MRT/LRT line 6. The system will display the arrival time of the MRT/LRT at the selected station
Alternative Flows:	AF-S5: If the user selects the star icon 1. The system will display the MRT/LRT stations saved by the user previously 2. The user can click on a saved station

	3. The system will display the arrival time of the MRT/LRT at the selected station
Exceptions:	EX-S5: If the user inputs invalid MRT/LRT station 1. The application will display "Invalid station! Please enter another station." 2. The application returns to step 5 EX-S4: If the user chooses to be a commuter when all public transport is terminated 1. The application will display "There is no public transport available now, please use other forms of transport." 2. The application returns to step 2
Includes:	Add as favorite
Special Requirements:	
Assumptions:	
Notes and Issues:	

Use Case Template

Use Case ID:			
Use Case Name:	Retrieving/updating data		
Created By:	Hendy	Last Updated By:	Brendon
Date Created:	3/9/2022	Date Last Updated:	6/9/2022

Actor:	User ,Database, Admin
Description:	Our application requires live transports or traffic related data to inform the latest information to end users such as when is the next bus arriving, the number of car park lots available, and etc. The information can be retrieved using API services or extracting data from different sources such as the Land Transport Authority(LTA) website.
Preconditions:	1. The user's phone must be connected to the internet. 2. The user's phone must be connected to the GPS. 3. The user's phone must be an android.
Postconditions:	1. Database reads in JSON files only. 2. The latest data and information have been updated successfully
Priority:	
Frequency of Use:	5 times per day
Flow of Events:	1. System retrieves the latest data from the databases 2. System will then proceed to update the information to the user
Alternative Flows:	AF-S1: If System fails to retrieve and update data: 1. Admin will have choice to manually update the data
Exceptions:	EX-S1: If System fails to retrieve and update data: 1. The User and Admin will be alerted through an error message
Includes:	
Special Requirements:	
Assumptions:	
Notes and Issues:	