PROJECT B

Project Name: Macquarie University Navigation Application (MUNA)

Engineer Name: Jannatul Munera Jalen
Student ID: 457 879 72
Client Representative: Seunghoon Kwak
Student ID: 443 955 15

Table of Contents

Vision Statement

For Macquarie University with a soon to return plan on-campus teaching, therefore welcoming a vast amount of people (students, staff, tourists etc.) who may be coming to campus for the first time, it may be hard for such people to work out where their scheduled classes, meetings or services such as food courts are. Thus, the main goal is to develop a well-planned system that can efficiently calculate people's travel routes and estimated waiting time based on how many people are willing to come back on campus, avoiding any unwanted routes or confusion within the campus.

1.0 System Design Document

1.1 System Architecture

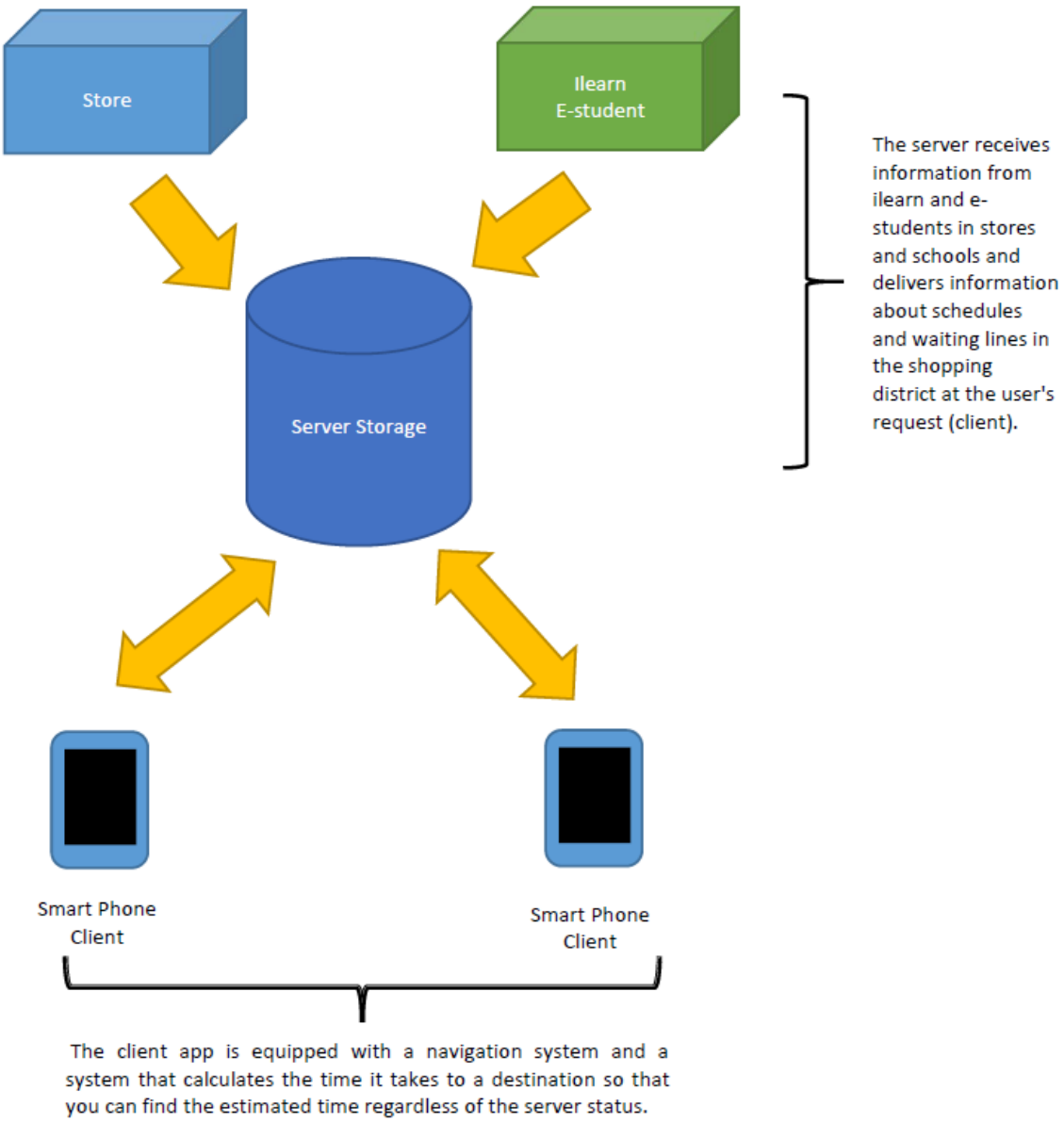

Fig 1: Role Sharing Strategy by Kwak_Seunghoon_Assignment_1, page 8

1.2 Storage Data Strategy

After reviewing through the system structure, it was decided to move the system to the client system to track users and calculate arrival time using GPS, hence getting the schedule from the server every time the application is turned on by the client. By moving the distance operation to the client, system change will reduce the system that the server has to pay. Moreover, the added schedule management method will be receiving data from eStudent and iLearn and send it to the client. The change will increase the client's weight, but it is expected that the server will be able to send information to more clients as the workload will be reduced. In addition, as the system for obtaining the estimated time to the destination changes independently from the server, the estimated time will be received by entering the destination regardless of the server's condition.

1.3 Noteworthy Trade-offs and Choices

Contrary to the expectation that the most time spent in the group was the architecture selection process, there were many areas to consider. To start off, the SC architecture is a structure that centralizes most functions. It must be an application that can be used by students and faculty or visitors, so it must purchase and maintain servers that can cover numbers including visitors or bear system errors and delays due to increased traffic. If any problems were to occur within the server, most functions can turn out to be problematic and if requests from more than capacity are repeated, traffic congestion can occur. Financial drawbacks must consider this disadvantage before the architecture decision. However, there are also clear advantages as mentioned above. For example, the SC architecture is a model that represents decentralized responsibility among independent computers and can be easily replaced, repaired or upgraded. At the same time, the client is not connected to the server, which reduces the need for patches and upgrades of users. In addition, the server client model supports better access and resource control than other architectures because it aims at centralized server and data management. As a result of comparing the advantages and disadvantages of an SC, it was finally decided to use the server- client as a system structure because it could be expected to five the system a clear advantage. However, it costs a lot of money and management.

1.4 Concurrent Process and How they will be coordinated

The SC architecture is an independent computer-executed system that has no reason to bear all functions on the server, so tasks that do not need to be calculated by receiving information from the server are allocated to the client, and the server allocates other functions.

Server

-The how-to busy calculation method expected from the client-side to the developer side was to count the waiting time using a camera and calculate the waiting time by the number. Accordingly, the method used in the development version is expected to be obtained by adding a count function to the camera in front of the counter and multiplying the number of hours in front of the count by the average time it takes to ship goods.

-The information to be transmitted at the client's request within the server is the store's waiting time calculated. Depending on the user, the schedule received in conjunction with the in school system should also be transmitted.

Client

-The smartphone app is given the role of a client in the system, which means that phones can calculate the simple degree of computation on their own without linking with the server. Marking maps and finding directions in connection with GPS will be handled within the client, and the calculation will be based on the adult male gait speed of 5km/h by setting the nearest road from the current user as the starting point and calculating the distance to the destination only by road.

In addition, if there are variables that visit the restaurant, it will be calculated by adding the time from the starting point to the restaurant, the time from the restaurant to the destination, and the waiting time of the restaurant.

Object recognition technology and navigation systems should be prioritized for the project, and the more comprehensive the error range of these two systems, the more problems will arise in the reliability of the application to be developed. Therefore, it is necessary to reduce the error range of the system as much as possible before starting development.

1.5 Package Diagram

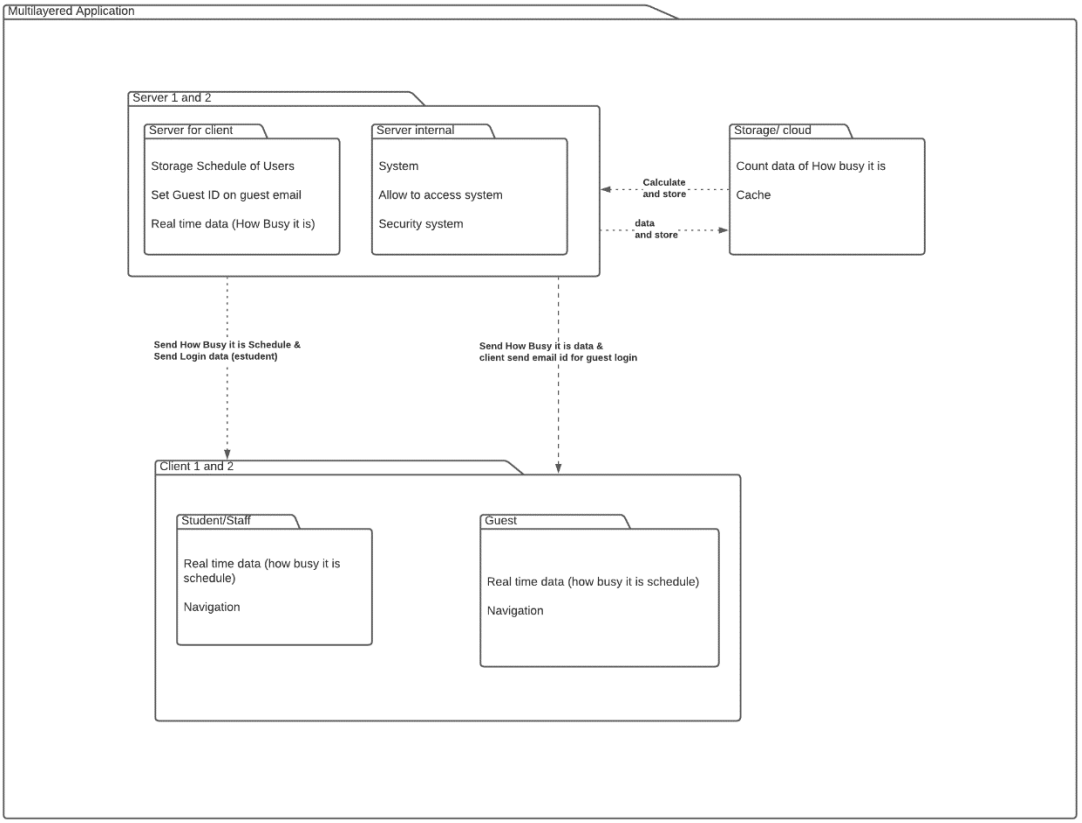

Fig 2: Package Diagram

2.0 Data Definitions

Field	Attributes	Values
User	User Name (Student/Staff)	45787972
User	Email (Student/Staff)	Pear.nice@mq.edu.au
User	Password	Pumpkin123
User	Account Status	ACTIVE
Navigation address Search	NavigationAddress	UBAR
Estimated Time	TotalEstimatedTime	10 mins
How busy	HowBusyItIs	Very Busy
Display Schedule	displayeStudentTimeTable	8 Ondaatje Ave, GR10

3.0 Analysis and Design Class Diagram

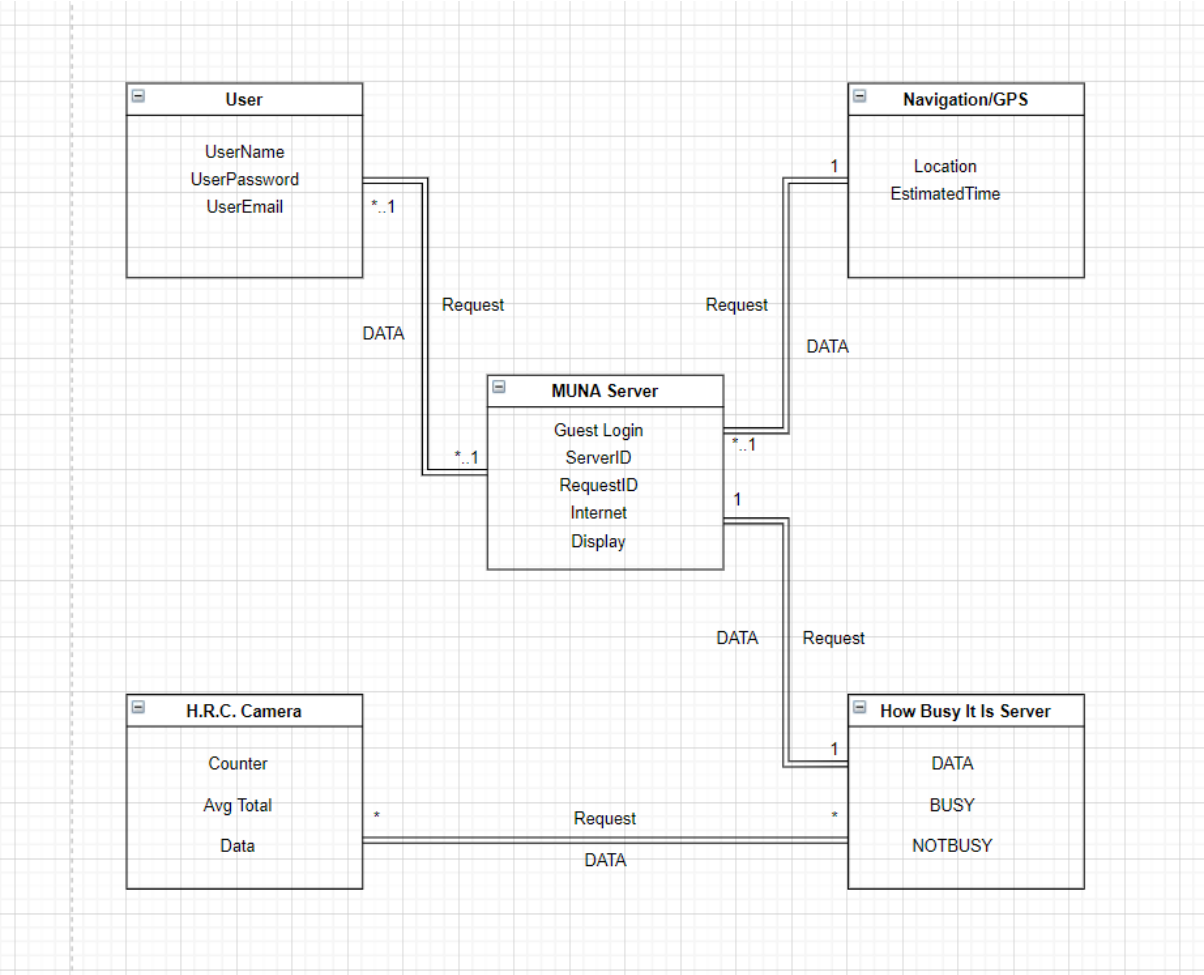

Fig 3: Analysis and Design class Diagram

4.0 State Diagram

4.1 How Busy It Is

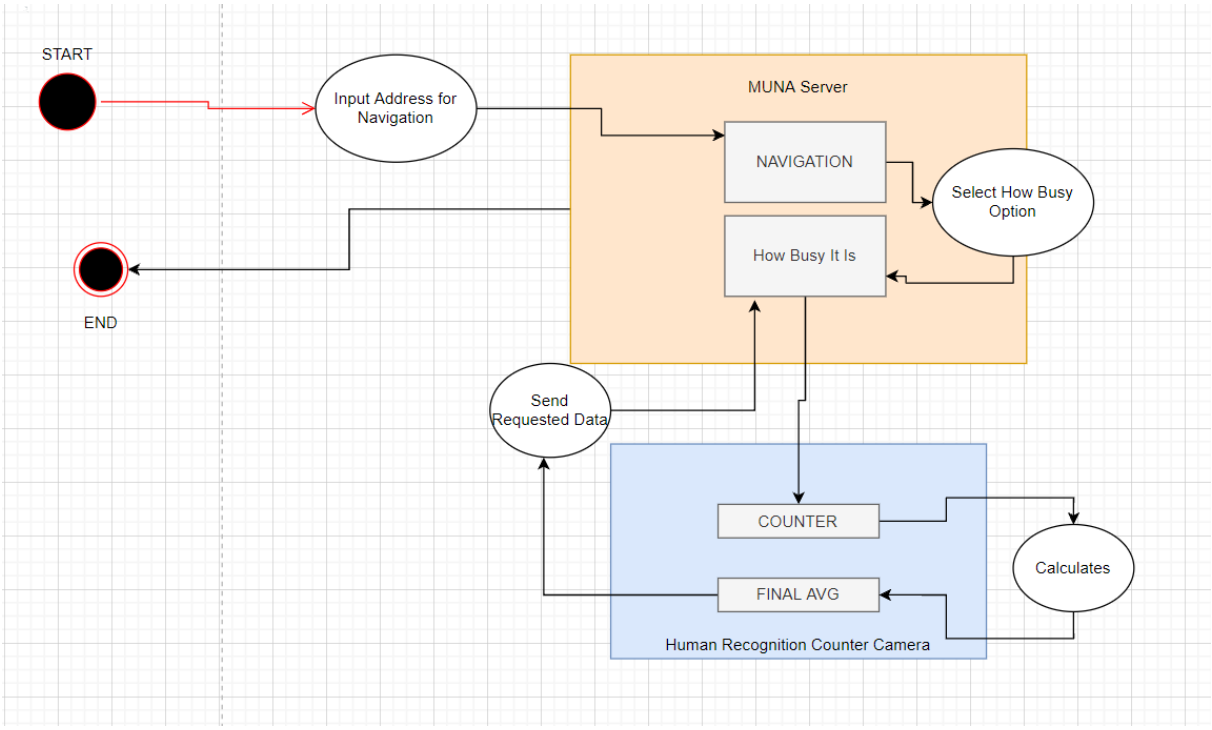

Fig 4: State Diagram

5.0 Requirement Traceability Matrix

Requirement ID	Use Case	Classes	Methods	Packages	Build No.
R1	- Log in	User MUNA Server	UserLogin(Email and Password) displayLogin	User- UserLogin(Email and Password) Server User- UserLogin(Email and Password) MUNA- displayLogin	
R2	- Log out	User Server	UserLogOut	User- UseLogOut	
R3	- Log in - Log in failed	User Server	UserLogin(Email and Password)	User- UserLogin(Email and Password)	
R4	- Enter Address - Navigation Display - Estimated time of arrival	User MUNA GPS	displayNavigation(address) inputNavigation(address) displayEstimatedTime	User- inputNavigation(address) MUNA- displayNavigation(address) displayEstimatedTime	
R5	- Enter Address - Display How busy it is in the said venue	User MUNA Server Internet How busy it is	inputNavigation(address) inputHowBusyItIs(count) displayHowBusyItIs	User- inputNavigation(address) inputHowBusyItIs How Busy It Is- displayHowBusyItIs	
R6	- Display scheduled Time table	User MUNA Server Internet eStudent	UserLogin(Email and Password) DisplaySchedule(class time and room)	User- UserLogin(Email and Password) eStudent- DisplaySchedule(class time and room)	
R7	- Access Denied for faulty internet - Display cache caught data	Server Internet GPS/Navigation Cache MUNA	offlineModeServer offlineModeGPS displayPrevNavigation	Server- offlineModeServer GPS/Navigation- offlineModeGPS MUNA-	

	- - Display previous navigation (for both guest and student/staffs)			displayPrevNavigation	
R8	- Update How busy it is on people entering or leaving the venue	Server How busy it is Human Recognition Counter Camera Cache	countRealTime UpdatethruRealTime	Human Recognition Counter Camera-countRealTime How Busy It is & Server-UpdatethruRealTime	
R9	- Log in - Log out	User	UserLogin(Email and Password) UserLogOut	User-UserLogin(Email and Password) UserLogOut	
R10	- Accurate Estimated time on journey to an inputted destination	GPS/Navigation Internet	calculateDistance calculateTime stableConnection	GPS/Navigation-calculateDistance calculateTime stableConnection	
R11	- Student accounts to be connected to eStudent for management of schedule and class navigation	MUNA MaqServer Internet Cache GPS	getEstdRecords calculateDistance calculateTime	GPS/Navigation-calculateDistance calculateTime MUNA-getEsrdRecords	
R12	- Accurate working of How busy it is Server	MUNA Server How Busy It Is Cache	updateHowBusyItIs	MUNA-updateHowBusyItIs Server-updateHowBusyItIs How Busy It is-updateHowBusyItIs Cache-updateHowBusyItIs	

6.0 Test Specifications

6.1 Test Case Specification

Identifier	T1
Description	User inputs their log in username or email and password to access all the exclusive features of the app (student/staff)
Input Specifications	- User - Macquarie University Navigation Application (App) - Server (Services)

Steps	Procedures	Expected Outcomes
1	User opens the app which is already installed in their phone.	User successfully opens the app and the app displays a log in screen
2	User inputs their username or email and password	Login successful and now the user has access to exclusive features such as schedule management, saved previous visited location etc. The account stays logged in unless the user logs out themselves, thus saving previous data.

Identifier	T2
Description	User inputs their log in username or email and password to access all the exclusive features of the app (guest)
Input Specifications	- User - Macquarie University Navigation Application (App) - Server (Services)

Steps	Procedures	Expected Outcomes
1	User opens the app which is already installed in their phone.	User successfully opens the app and the app displays a log in screen
2	User selects to login as guest	Login as guest is successful, but now the user has limited access to exclusive features such as schedule management, saved previous visited location etc. Guest login does not stay logged in and the app displays a log in screen every time the app is restarted, losing all previous data.

Identifier	T3
Description	User inputs their desired destination on the app for navigation of classroom, food court etc.
Input Specifications	- User - Macquarie University Navigation Application (App) - Server (Services and Campus Map) - GPS

Steps	Procedures	Expected Outcomes
1	User opens the app which is already installed in their phone.	User successfully opens the app (assuming the user is already logged in) and displays the main screen
2	User clicks on the search bar	App pulls out the keyboard for the user to input their desired destination
3	User enters their desired destination on the search bar	The app's server displays destination related to the user request in real time with options for the user to select from such as "Directions", "How busy it is", etc.
4	User selects "Directions" on one of the displayed options	App's navigation server displays the user's and the destination's distance and estimated time of arrival via GPS.

Identifier	T4
Description	User wants to check how busy a location is
Input Specifications	- User - Macquarie University Navigation Application (App) - Server (Services and Campus Map) - How busy it is

Steps	Procedures	Expected Outcomes
1	User opens the app which is already installed in their phone.	User successfully opens the app (assuming the user is already logged in) and displays the main screen
2	User clicks on the search bar	App pulls out the keyboard for the user to input their desired destination
3	User enters their desired destination on the search bar	The app's server displays destination related to the user request in real time with options for the user to select from such as "Directions", "How busy it is", etc.
4	User selects "How busy it is" on one of the displayed options	App's navigation server displays the real-time of how busy the requested destination is.

Identifier	T5
Description	Camera detection to assist in display of “How busy it is”
Input Specifications	- Human Recognition counter camera - Macquarie University Navigation Application (App) - Server - How busy it is

Steps	Procedures	Expected Outcomes
1	Cameras attached on the public venues and class rooms scans people entering and exiting the place.	Camera successfully calculates the total number of people inside a venue/classroom and sends the information over to the server. Keeps on updating in real time every time anyone enters or exits.
2	The server tallies the total count sent via the camera with the data.	Server successfully tallies the total avg count and sends over information to the app whether the venue/ class room is quiet or very busy depending on the data.

Identifier	T6
Description	The user enters an invalid address
Input Specifications	- Macquarie University Navigation Application (App) - Server (Services and campus map)

Steps	Procedures	Expected Outcomes
1	User opens the app which is already installed in their phone.	User successfully opens the app (assuming the user is already logged in) and displays the main screen
2	User clicks on the search bar	App pulls out the keyboard for the user to input their desired destination
3	User enters their desired destination on the search bar	The app’s server displays “0 matches” and displays destination related to the user request in real time with options for the user to select from such as “Directions”, “How busy it is”, etc.

Identifier	T7
Description	After downloading the app, operate only the client to check the navigation system.
Input Specifications	- Macquarie University Navigation Application (App) - Server (Services and campus map)

Steps	Procedures	Expected Outcomes
1	User opens the app which is already installed in their phone.	User successfully opens the app (assuming the user is already logged in) and displays the main screen
2	User clicks on the search bar	App pulls out the keyboard for the user to input their desired destination
3	User enters their desired destination on the search bar	The app's server displays destination related to the user request in real time with options for the user to select from such as "Directions", "How busy it is", etc.
4	User selects "Directions" on one of the displayed options	App's navigation server displays the user's and the destination's distance and estimated time of arrival via GPS.

6.2 Test Plan

6.2.1 Test Schedule

Test Case Identifier	Test Case Description	Start Date	End Date
T1	User inputs their log in username or email and password to access all the exclusive features of the app (student/staff)	28/10/2021	29/10/2021
T2	User inputs their log in username or email and password to access all the exclusive features of the app (guest)	30/10/2021	31/10/2021
T3	User inputs their desired destination on the app for navigation of classroom, food court etc.	1/11/2021	3/11/2021
T4	User wants to check how busy a location is	5/11/2021	6/11/2021
T5	Camera detection to assist in display of “How busy it is”	10/11/2021	15/11/2021
T6	The user enters an invalid address	17/11/2021	20/11/2021

T7	After downloading the app, operate only the client to check the navigation system.	22/11/2021	23/11/2021

6.2.2 Test Resourcing

- Hardware

Hardware	Configuration
Human Recognition counter camera	This Human Recognition counter camera as defined by the name is a test resource required to count the total number of people entering or exiting a venue or classroom and sends over the real time data over to the server. Essential subsystem for “How busy it is”.
Smart Phone	A test resource that is required for users to be able to gain information and navigation help. Must be compatible to various Operating system platform • IOS version 13 or newer • Android Version 9 or newer

- Software

Software	Configuration
MUNA/Server	Software that holds all information such as GPS of the campus map, how busy it is etc.
How busy it is	A subsystem of the Server software that sends information to the server on how busy a place. Connected to the human recognition counter camera.
GPS	Collaborated with google, a subsystem of the server software, a campus map to help users navigate via desired destination.

6.3 Test Milestones

Deliverables	Responsible Party	Date
T1	User MUNA (Client) Server	28/11/2021
T2	User MUNA (Client) Server	30/11/2021
T3	MUNA (Client) Server	1/12/2021
T4	MUNA (Client) Server GPS	5/12/2021
T5	MUNA (Client) Server GPS	10/12/2021
T6	MUNA (Client) Server GPS	17/12/2021
T7	MUNA (Client) Server GPS	19/12/2021

Report

While working on this particular project and further developing the requirements and specifications, I ran into some issues whist working with the SRS provided by my partner. These failures include prioritizing storage strategies for software systems, creating test cases that describe and enforcing functional and non-functional requirements for software systems, and designing system architectures to address the performance and maintenance of software systems. Processes to improve conservatism are included. Other factors that may affect the performance of the Software Product.

Prioritizing a storage strategy for a software system was one of the first issues I faced in this work. Because it was difficult to determine how the data was obtained and how to use it in the software system. The process of prioritizing an efficient storage strategy was also difficult because SRS did not define a specific approach to data storage. This made it difficult to respond to functional and non-functional requirements and to find a functional approach during the implementation phase. As a result, work on other important elements of this project, such as analysis and class diagrams, was delayed because I wasn't sure what strategy to use when creating this type of diagram. A limited form of interaction with other classes to store their own data was able to solve this problem. This allowed for better integration of other data processing technologies into the implementation process.

The creation of different test cases for this particular software product was one of the challenging tasks I had during the implementation phase of this project as the requirements within the SRS addressed different circumstances as well as different entities which were essential in reassuring a successful and thorough application of the software in different types of navigation access I this context. The design document provided by my partner only had limited number of requirements and was pretty hard to understand, so it was hard to make up particular test cases and I did my very best to include as many test cases as possible without trying to include extra test cases that would contradict the design specs. I have only included test cases I thought to be prioritized as the assignment has a page limit of 20 pages.