

Software Requirements Specification
NavUP
COS 301
Team: Hotpink

Gregory Austin 14039712
Stephanie Groutsch 14293324
Timothy Kirker 11152402
Drew Langley 11039753
Xoliswa Ntshingila 13410378
Melvin Zitha 12138747

Friday, February 24, 2017

Contents

1	Introduction	3
1.1	Purpose	3
1.2	Scope	3
1.3	Overview	3
2	Overall Description	4
2.1	Product Perspective	4
2.2	Product Functions	4
2.3	User Characteristics	4
2.4	Constraints	5
2.5	Assumptions and Dependencies	6
3	Requirements Specification	7
3.1	Functional Requirements Specification	7
3.1.1	Requirements	7
3.1.2	REQ1: Navigation	7
3.1.3	REQ2: Provide Information	7
3.1.4	REQ3: Provide Different Routes	7
3.1.5	REQ4: User Profiles	7
3.1.6	REQ5: Weekly goals	7
3.1.7	REQ6: Personalization	7
3.1.8	REQ7: Maintenance	7
3.2	Use Cases	8
3.2.1	Use Case for REQ1: Navigate user to their desired location	8
3.2.2	Use Case for REQ1: Obtain Current Location	9
3.2.3	Use Case for REQ1: Search for Location	10
3.2.4	Use Case for REQ1: Calculate the estimated time of arrival	11
3.2.5	Use Case for REQ2: Display metadata with information regarding venues and points of interest	12
3.2.6	Use Case for REQ2: Provide information regarding sports and cultural activities	12
3.2.7	Use Case for REQ3: Provide special routes for users with disabilities	12
3.2.8	Use Case for REQ3: Provide the fastest route to a destination	13
3.2.9	Use Case for REQ4: The user logs into the application	14
3.2.10	Use Case for REQ5: Collecting and viewing rewards	15
3.2.11	Use Case for REQ6: Allow users the ability to save a location(drop a pin)	16
3.2.12	Use Case for REQ7: Add or remove a venue	17
3.2.13	Use Case for REQ7: Add or remove event	18
3.2.14	Use Case for REQ7: Update Information	19
3.2.15	Use Case for REQ7: Add or remove goals	20
3.3	Quality Requirements	21
3.4	Access Channel Requirements	21
3.5	Integration Requirements	21
3.6	Architecture Constraints	22
3.7	User Interface	22

1 Introduction

1.1 Purpose

The purpose of this document is to provide a detailed description of the requirements for the NavUP application, a navigation application which is intended to assist students, staff and the public with navigating the University of Pretoria. Additionally it will outline and explain the complete plan for building the NavUP system. The explanations include the constraints for the system and how the system will interact with external applications. Ultimately this document is to be used for client approval such that the system may be implemented.

1.2 Scope

The NavUP mobile application makes use of the Wi-Fi on campus in order to assist the user in finding the route to their destination from their current location or from a location which is on the University of Pretoria facilities this includes but is not limited to lecture halls, food courts, ablution facilities or possibly even sport or faculty houses.

The NavUP mobile application will use the users GPS location, on their smart devices, in conjunction with all the Wi-Fi routers on campus in order to determine the best route from their starting location. As a result the users require Wi-Fi enabled devices and login credentials to the Universities Wi-Fi network. The system will not be able to navigate a user from a location which is not within the range of the campus Wi-Fi.

Further more the NavUP system also includes best route capabilities, optimal routes for users with disabilities, status of the number of people currently on campus based on the number of devices currently connected to the Wi-Fi network, points of interest, activities and events that are happening on campus. It will also track the activity of the user e.g. The number of steps taken and reward the user based on the goals they have achieved.

1.3 Overview

These are the chapters that follow; The Overall description chapter which describes the overall system, the modules and the interfaces that exist within the system, the system functions, the characteristics of the users of the system and the system constraints. The Specific Requirements chapter is intended for developers and contains the external interface requirements, functional requirements, performance capabilities of the system as well as the design constraints and quality related requirements.

2 Overall Description

2.1 Product Perspective

Similar to existing navigation applications such as: Google Maps, Apple Maps and Waze, NavUP is a standalone mobile application that acts as a campus navigation system which will be open for all UP students, staff and guests to use. It encompasses all technologies, which will be required for the necessary functionality as described in the product functions section.

The application will be implemented using the relevant, already existing hardware, software interfaces, OS interfaces and technologies such as, Wi-Fi, Cellular Data and GPS. Implementation of the hardware interfaces will require the existing software interfaces/OS interfaces to access data from the hardware interfaces relating to the main functionality of the NavUP application.

The application will use communication interfaces to determine traffic and the best routes to follow, these communication interfaces will work through the required hardware interfaces. The main operations of the application will be accessed through the user interface, which will utilize underlying technologies. These operations include: navigating to a class, providing direction (showing where to go), displaying destination information(describing where and what it is you're going to). The DBMS will have to be manually maintained with location and event data or interfaced with an existing DBMS / DSS that the university already has.

2.2 Product Functions

The NavUP mobile application will support a variety of functions namely; navigation, providing information, allowing for users to decide on different routes, achieving goals and personalisation capabilities.

The navigation functionality will consist of locating the user and then navigating them to their required location. They will be provided with searching capabilities in order to obtain their required location. Users will then be able to save their destination for future use.

Information will be provided to users when they are near points of interest. This information will consist of historical and general information regarding venues and/or landmarks. Information regarding cultural and sporting events that relate to the user will also be shown when appropriate.

Different routes will be provided based on the users needs. Such routes will include the fastest path from one location to the next. This path will be determined based on the amount of pedestrian traffic. Another route will be provided for users with special needs in order to allow them to reach their destination using the most user friendly path for their disability.

Weekly goals will be provided to motivate users to attend all their lectures during the week, explore the campus and be more informed about it as well as to increase the amount of steps they take this being beneficial to their overall wellbeing.

2.3 User Characteristics

There are three main categories of individuals that will make use of the application namely administrators, general users and third party rewards managers.

The administrators will ensure that the software works as intended and will be in charge of maintaining and upgrading the application. These users will need to understand how the system works as well as have programming knowledge in order to work on the software.

The general users of the system will consist of students, lecturers, guests and university employees. They will only make use of the mobile application. This means that they will not require any knowledge of how the software is implemented however, they will need to have knowledge regarding how to use a smart device. These users will mainly use the application for navigational purposes.

Third party rewards managers will consist of business parties who will provide rewards to individuals who have obtained their weekly goal. Since they will only make use of the client side of the mobile application, they will not require any knowledge of how the system is implemented. They will however, require knowledge regarding whether the user has reached their goal or not.

2.4 Constraints

The following are restrictions related to the application:

1. Security - The users personal information and current location should not be accessible to the public.
2. Accuracy - The users location should be found whether the user is indoors or not. The location of the user should also be found in terms of which floor they are on in the building.
3. Performance - The system should be able to handle a large amount of users making use of the software at the same time.
4. Reliability - The application should still operate when one or more data access points are no longer available.
5. Accessibility - The application should be easily accessible to everyone that requires it.
6. Usability - The applications interface should be easy to use and understand.
7. Size - The application should not require too much memory in order to operate.

2.5 Assumptions and Dependencies

Since the NavUP system is being created for mobile devices, it is assumed that all those who require the application, will have access to a mobile device. It is then assumed that these devices will have the required capabilities necessary to run the application. Such dependencies include built in WiFi, GPS and cellular connectivity as well as the required data/airtime(in the event that they are not connected to the University's Wi-Fi).

Assumed users will be registered students of the University of Pretoria thereby having access to their modules and class data for navigating when required with minimal input from the user, this depends on access to students data for the DBMS, staff navigating to their work stations , guests wanting to navigate to a location which they only know by name.

In terms of the usability of the application, it is assumed that users will have knowledge regarding how to use mobile devices and applications on such devices such as the downloading and installation of applications to a smart device.

The assumption relating to accuracy includes the belief that the application will be up to date with the latest information in terms of venues, landmarks and events. This assumes that there will always be someone maintaining and updating the applications data, or a DBMS that is interfaced with the university DBMS.

An assumption in terms of the systems performance includes the idea that it will still be able to function when a large amount of users are making use of the application at the same time. This depends on server capacity as well as solid code structures that are efficient as well as secure.

3 Requirements Specification

The NavUP system will be available for use on many platforms and make use of multiple means of network links and other external interfaces such as GPS and GIS services. Other requirements such as quality, integration and architectural constraints will also be covered in this section.

3.1 Functional Requirements Specification

3.1.1 Requirements

3.1.2 REQ1: Navigation

Provide navigation functionality.

3.1.3 REQ2: Provide Information

Provide information regarding points of interest and sporting and cultural activities.

3.1.4 REQ3: Provide Different Routes

Provide different routes based on the user's needs and preferences.

3.1.5 REQ4: User Profiles

The system should allow users and administrators login capability.

3.1.6 REQ5: Weekly goals

Provide weekly goals based on the number of steps or distance travelled by the user.

3.1.7 REQ6: Personalization

Users should be allowed to save personalized information as well as retrieve it.

3.1.8 REQ7: Maintenance

The system should allow administrators the ability to C.R.U.D it.

3.2 Use Cases

3.2.1 Use Case for REQ1: Navigate user to their desired location

1. Use Case ID: UC1
2. Precondition: User is running the NavUP application and enters the navigation module
3. Postcondition: Client device receives and displays navigational information to destination.
4. Actor-System interaction model:

Figure 1: Use Case Diagram - UC1 Navigate user to their desired location

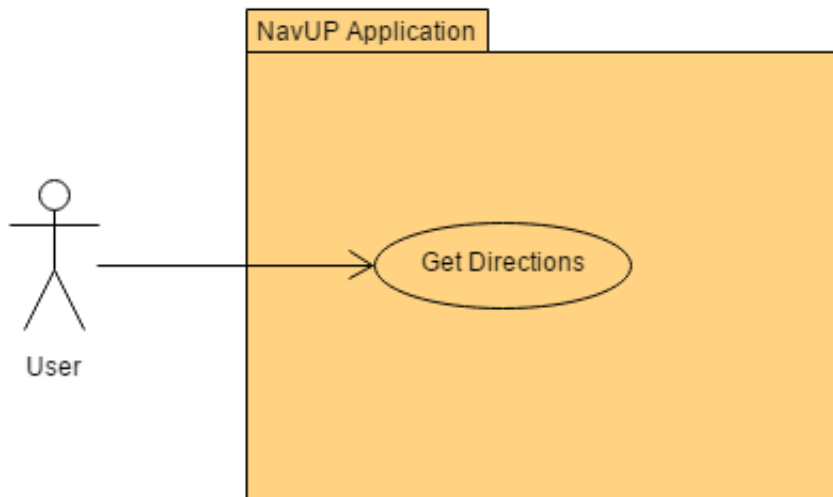


Table 1: UC1 - Navigate user to their desired location

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks on the navigation button in the main menu	2. The system displays the navigation page to the user
3. The user clicks on the Get Directions button on the Navigation page	4. The system prompts the user for their required destination
5. User enters desired destination	6. The system calculates and displays the route and directions to the destination

3.2.2 Use Case for REQ1: Obtain Current Location

1. Use Case ID: UC2
2. Precondition: User is running the NavUP application and requires their location
3. Postcondition: Client device receives and displays the user's current location data.
4. Actor-System interaction model:

Figure 2: Use Case Diagram - UC2 Obtain Current Location

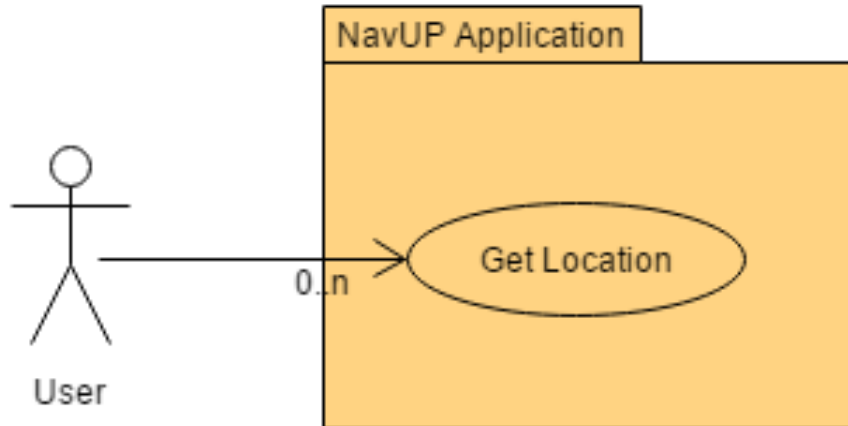


Table 2: UC2 - Obtain Current Location

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks on the navigation button in the main menu	2. The system displays the navigation page to the user
3. The user clicks on the Get Location button on the Navigation page	4. The system displays the user's location

3.2.3 Use Case for REQ1: Search for Location

1. Use Case ID: UC3
2. Precondition: User is running the NavUP application and searches for a given location.
3. Postcondition: Information regarding the searched location os provided.
4. Actor-System interaction model:

Figure 3: Use Case Diagram - UC3 Search for location

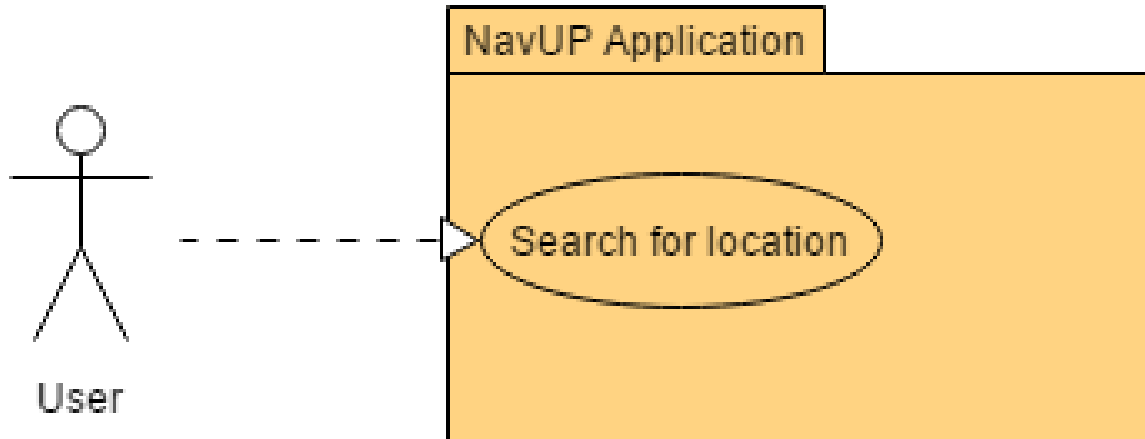


Table 3: UC3 - Search for location

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks on the navigation button in the main menu	2. The system displays the navigation page to the user
3. The user clicks on the Search for Location button on the Navigation page	4. The system prompts the user for the location they wish to search
5. User enters desired location	6. The system provides information regarding the searched location

3.2.4 Use Case for REQ1: Calculate the estimated time of arrival

1. Use Case ID: UC4
2. Precondition: User is running the NavUP application and currently travelling to a destination.
3. Postcondition: Client device receives and displays the estimated time of arrival.
4. Actor-System interaction model:

Figure 4: Use Case Diagram - UC4 Calculate the estimated time of arrival

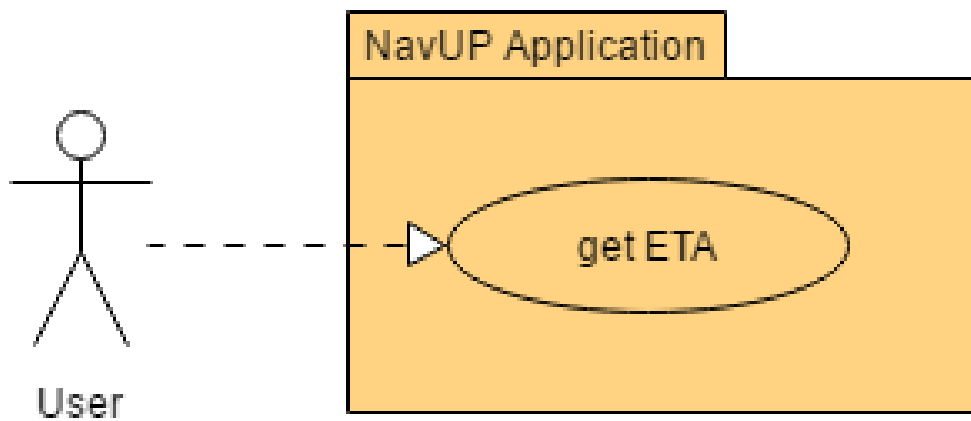


Table 4: UC4 - Calculate the estimated time of arrival

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks on the navigation button in the main menu	2. The system displays the navigation page to the user
3. The user clicks on the Estimated Time of Arrival button on the Navigation page	4. The system displays the estimated time of arrival to the user

3.2.5 Use Case for REQ2: Display metadata with information regarding venues and points of interest

1. Use Case ID: UC5
2. Precondition: User is running the NavUP application and is currently travelling to a destination.
3. Postcondition: Client device displays a popup of information a point(s)of interest.

Table 5: UC5 - Display metadata with information regarding venues and points of interest

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user travels past a point of interest or venue	2. The system displays information related to the point of interest or venue to the user

3.2.6 Use Case for REQ2: Provide information regarding sports and cultural activities

1. Use Case ID: UC6
2. Precondition: The user is running the application and requests to view activities.
3. Postcondition: Client device recieves and displays information regarding the requested activities.
4. Actor-System interaction model:

Table 6: UC6 - Provide information regarding sports and cultural activities

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks the Activities button on the main menu	2. The system displays information regarding sporting and cultural activities that relate to the user

3.2.7 Use Case for REQ3: Provide special routes for users with disabilities

1. Use Case ID: UC7
2. Precondition: The user is running the application and requires a special route as they have a disability.
3. Postcondition: Client device recieves and displays information regarding the optimal navigational route for the special needs user.
4. Actor-System interaction model:

Figure 5: Use Case Diagram - UC7 Provide special routes for users with disabilities

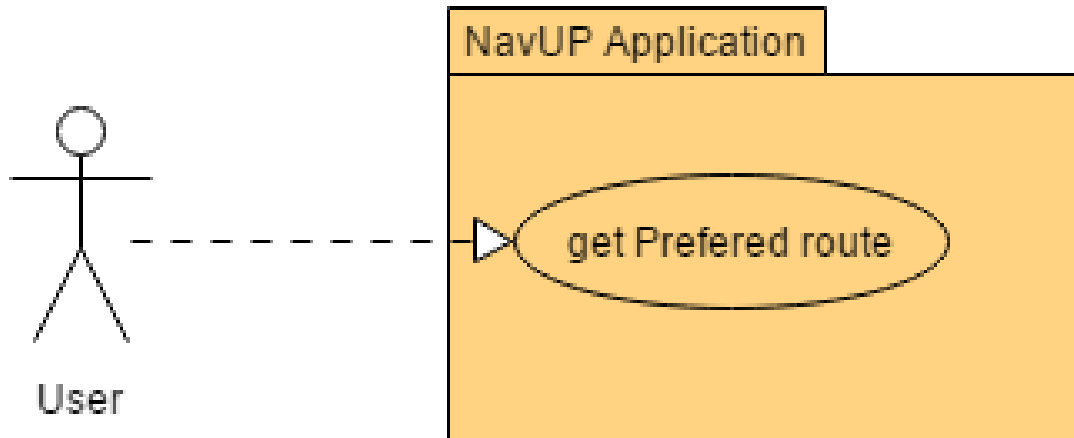


Table 7: UC7 - Provide special routes for users with disabilities

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks the Routes button on the main menu	2. The system prompts the user for a destination
3. The user enters a destination	4. The system prompts the user for the type of route they require
5. The user selects the "special needs" option	6. The system displays the route and directions to the destination.

3.2.8 Use Case for REQ3: Provide the fastest route to a destination

1. Use Case ID: UC8
2. Precondition: The user running the application and requires the fastest route from one location to the next.
3. Postcondition: Client device receives and displays information regarding the optimal navigational route.
4. Actor-System interaction model:

Figure 6: Use Case Diagram - UC8 Provide the fastest route to a destination

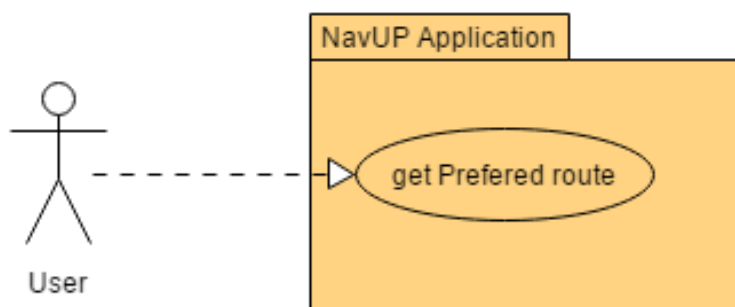


Table 8: UC8 - Provide the fastest route to a destination

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks the Routes button on the main menu	2. The system prompts the user for a destination
3. The user enters a destination	4. The system prompts the user for the type of route they require
5. The user selects the "fastest path" option	6. The system displays the route and directions to the destination.

3.2.9 Use Case for REQ4: The user logs into the application

1. Use Case ID: UC9
2. Precondition: The user wishes to log into the application.
3. Postcondition: Client device logs the user into the system and displays the user's login page.
4. Actor-System interaction model:

Figure 7: Use Case Diagram - UC9 The user logs into the application

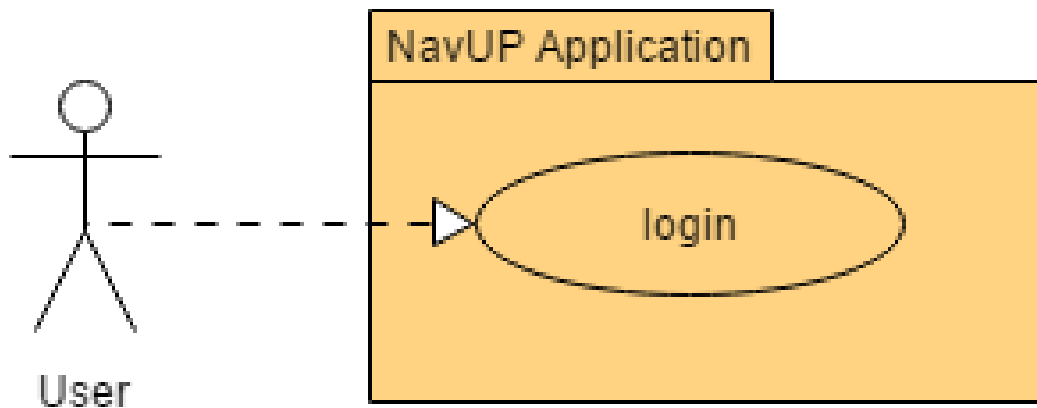


Table 9: UC9 - The user logs into the application

Actor: User	System: NavUP
	0. The NavUP system displays the main window
1. The user clicks the Login button on the main menu	2. The system prompts the user to enter the required login details
3. The user enters their login information	4. The system logs the user into the application and displays the user's user page

3.2.10 Use Case for REQ5: Collecting and viewing rewards

1. Use Case ID: UC10
2. Precondition: The user is logged in and completes an activity.
3. Postcondition: Client device receives a notification concerning the completion of an activity and is directed to the third party rewards system.
4. Actor-System interaction model:

Figure 8: Use Case Diagram - UC10 Collecting and viewing rewards

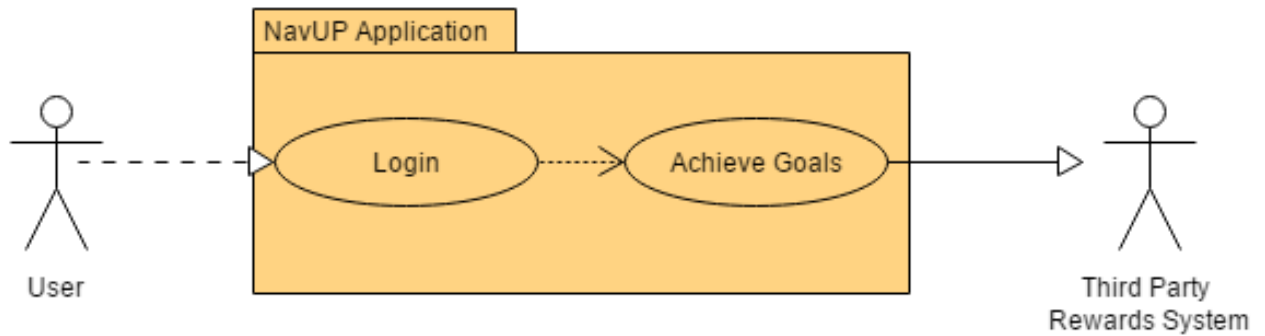


Table 10: UC10 - Collecting and viewing rewards

Actor: User	System: NavUP
	0. The NavUP system displays the user's page
1. The user completes an activity	2. The system notifies the user concerning the completion of an activity and directs the user to the third party rewards system

3.2.11 Use Case for REQ6: Allow users the ability to save a location(drop a pin)

1. Use Case ID: UC11
2. Precondition: The user is logged in and wishes to save their location.
3. Postcondition: Client device displays storage of a location by showing an icon over the specified position.
4. Actor-System interaction model:

Figure 9: Use Case Diagram - UC11 Allow users the ability to save a location(drop a pin)

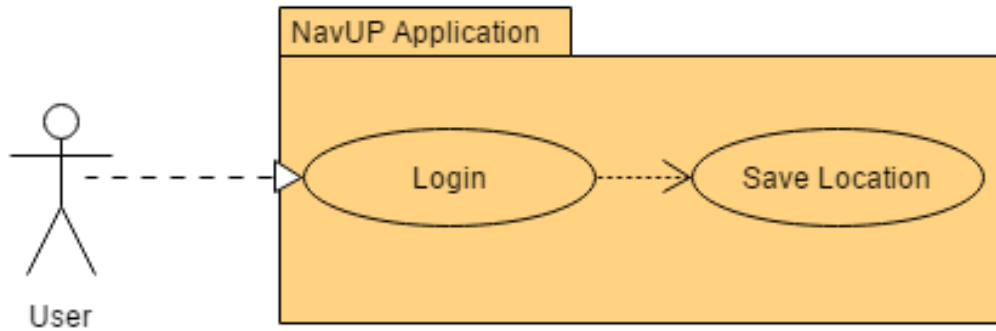


Table 11: UC11 - The user logs into the application

Actor: User	System: NavUP
	0. The NavUP system displays the user's page.
1. The user clicks the Save Location button	2. The system prompts the user to enter a description for the saved location
3. The user enters a description for the saved location and confirms the action	4. The system saves the location and displays an icon over the location

3.2.12 Use Case for REQ7: Add or remove a venue

1. Use Case ID: UC12
2. Precondition: The administrator is logged in and wishes to add or remove a venue.
3. Postcondition: The system is updated with the added venue.
4. Actor-System interaction model:

Figure 10: Use Case Diagram - UC12 Add Venue

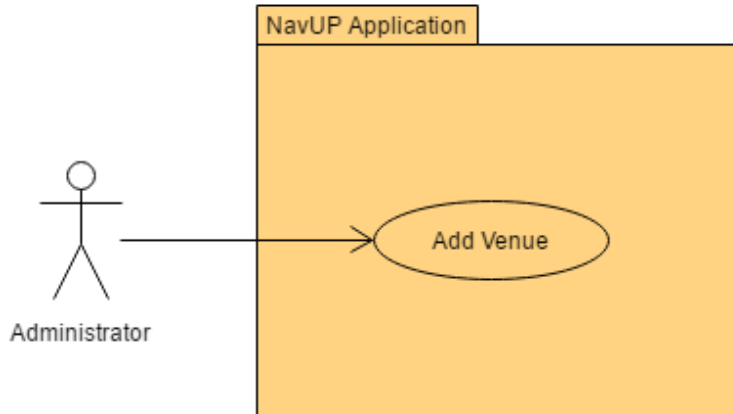


Figure 11: Use Case Diagram - UC12 Remove Venue

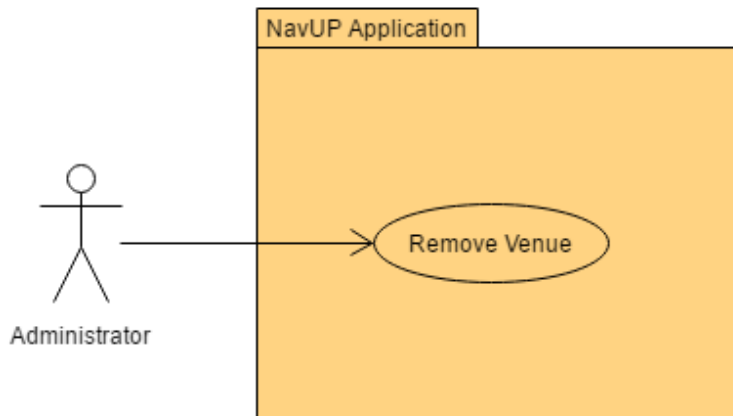


Table 12: UC12 - Add or remove venue

Actor: User	System: NavUP
	0. The NavUP system displays the administrator's page.
1. The administrator clicks on the add/remove venue button	2. The system prompts the for the location and name of the new venue.
3. The administrator enters the required information and confirms the action	4. The system saves and adds the given venue

3.2.13 Use Case for REQ7: Add or remove event

1. Use Case ID: UC13
2. Precondition: The administrator is logged in and wishes to add or remove an event.
3. Postcondition: The system is updated with the added event.
4. Actor-System interaction model:

Figure 12: Use Case Diagram - UC13 Add Event

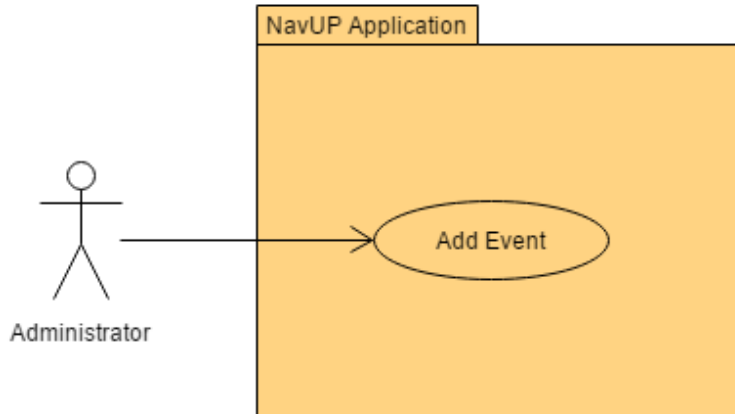


Figure 13: Use Case Diagram - UC13 Remove Event

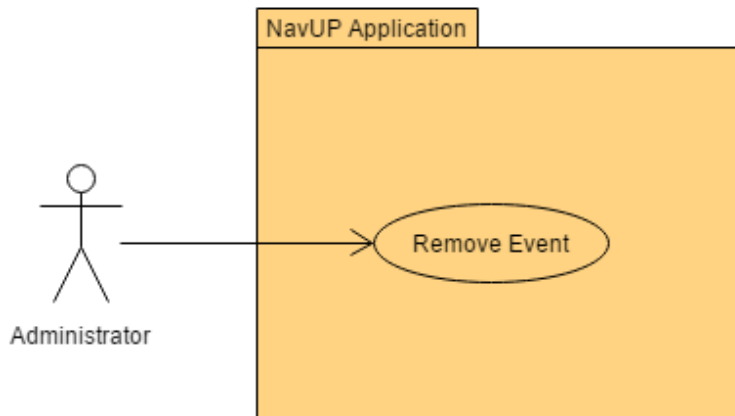


Table 13: UC13 - Add or remove event

Actor: User	System: NavUP
	0. The NavUP system displays the administrator's page.
1. The administrator clicks on the add/remove event button	2. The system prompts the for the location and name of the new event.
3. The administrator enters the required information and confirms the action	4. The system saves and adds the given event

3.2.14 Use Case for REQ7: Update Information

1. Use Case ID: UC14
2. Precondition: The administrator is logged in and wishes to update the information of a venue or landmark.
3. Postcondition: The system is updated with the added new information regarding the venue or landmark.
4. Actor-Sytem interaction model:

Figure 14: Use Case Diagram - UC14 Update Information

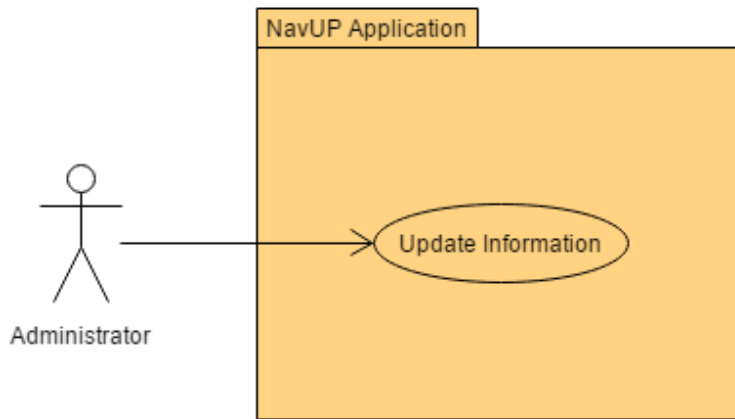


Table 14: UC14 - Update Information

Actor: User	System: NavUP
	0. The NavUP system displays the administrator's page.
1. The administrator clicks on the update information button	2.The system prompts the administrator for the name or the venue or landmark for which the administrator wants to update the information.
3. The administrator enters the required information and confirms the action	4.The system prompts the administrator for the new updated information for the venue.
5. The administrator enters the required information and confirms the action	7.The system updates the application with the new information

3.2.15 Use Case for REQ7: Add or remove goals

1. Use Case ID: UC15
2. Precondition: The administrator is logged in and wishes to add or remove a goal.
3. Postcondition: The system is updated with the new goals information.
4. Actor-System interaction model:

Figure 15: Use Case Diagram - UC15 Add Goals

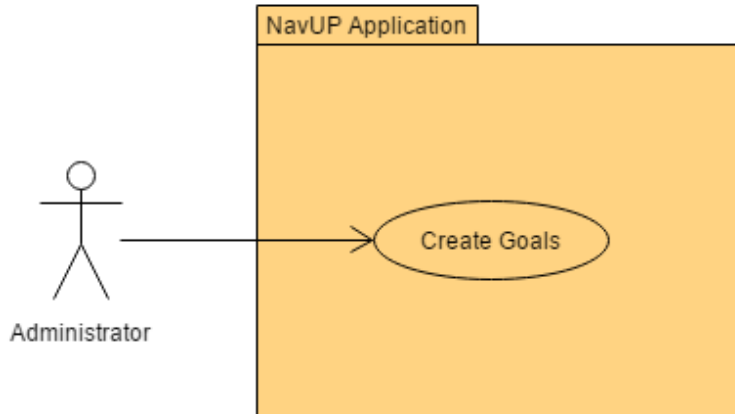


Figure 16: Use Case Diagram - UC15 Remove Goals

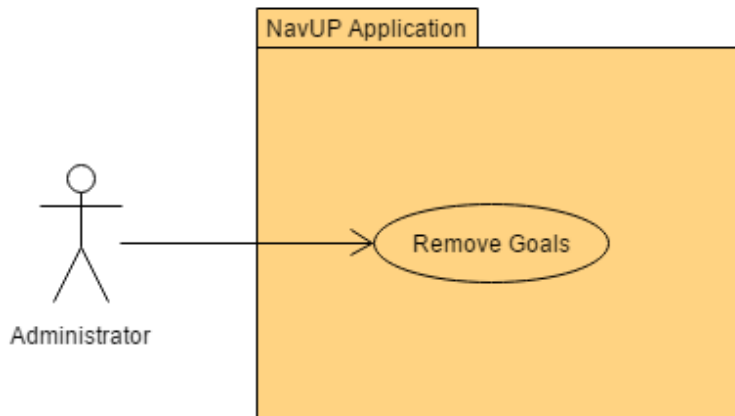


Table 15: UC15 - Add or remove goals

Actor: User	System: NavUP
	0. The NavUP system displays the administrator's page.
1. The administrator clicks on the add/remove goals button	2. The system prompts the for the requirements and name of the new goals etc.
3. The administrator enters the required information and confirms the action	4. The system updates the application with the new ionformation

Requirement	Priority	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8
REQ1	1	x	x	x	x				
REQ2	3					x	x		
REQ3	2							x	x
REQ4	4								
REQ5	5								
REQ6	5								
REQ7	2								
UC Priority		1	1	1	3	3	5	2	2

Figure 17: Traceability Matrix 1

Requirement	Priority	UC9	UC10	UC11	UC12	UC13	UC14	UC15	UC16
REQ1	1								
REQ2	3								
REQ3	2								
REQ4	4	x							
REQ5	5		x						
REQ6	5			x	x				
REQ7	2					x	x	x	x
UC Priority		4	5	3	3	2	2	4	3

Figure 18: Traceability Matrix 2

3.3 Quality Requirements

A system of the complexity and scale of NavUP must meet specific quality requirements such as:

1. Performance - the system must handle a possibility of over 50000 users simultaneously and still perform the core functionality without failure.
2. Reliability - the system should perform its functionality reliably, most importantly the navigational requirements.
3. Usability - the system should be easy to use while sporting an aesthetic and ergonomic interface.
4. Integrability - the system should allow for additional features and modules to be added easily.

3.4 Access Channel Requirements

From the typical user's perspective, approximately 60000 students and staff, the NavUP system will largely be focused on mobile devices such as Android or iOS smartphones. These devices, will make use of location services and internet capabilities to send and receive data to and from access channels and servers using mobile data, Wi-Fi signals, crowdsourcing and geolocation systems such as GPS and GIS. The campus Wi-Fi network will be the main form of system access as the signal strength covers most of the large area the University spans, and will help location accuracy indoors where the mobile devices may struggle to find decent mobile and location based signals.

3.5 Integration Requirements

The NavUP system will be implemented simultaneously by all the students enrolled in COS 301, it is therefore imperative that the system complies to a low coupling and high cohesion standard.

3.6 Architecture Constraints

The clients have specified the use of technologies such as GIS, UI/UX, Mobile Development (Android/iOS), Real-time Data Analysis, Data Streaming and Persistence, and a large focus on the use of Wi-Fi Networking to boost the system's accuracy. A module that supplies targeted delivery of information was also requested hence an intelligent program is also required.

3.7 User Interface

The user interface is likely to be used by students and staff alike, and many visitors to the campus, the interface therefore should be friendly and easy to use for all ages. The interface should include functionality for users to login and create personal profiles so that they can use the rewards system and access other functionality. There should also be different interface for administrative users who must be able to create point of interest and update activities and update and capture location data.