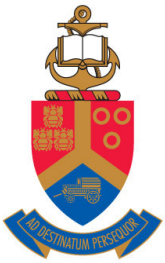


UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA

NavUp Requirements Specification

February 24, 2017

Group Lime

Lyle Nel - u29562695
Dail Nathan Jonker - u13325630
Siyabonga Magubane - u15289347
Ritesh Doolabh - u15075754
Matthew Perry - u13017030
Siavhe Rotondwa -

GitHub Repository:
`https://github.com/lyle-univ/cos301-team-lime`

Contents

1 Introduction

This section gives a description and an overview of all the information in the SRS document. The purpose of the system will be provided including additional information of the system.

1.1 Purpose

This document serves to describe the requirements of the NavUP system. This includes the overall features of the system, functionality, interfaces, constraints and integration. The purpose of this document is intended as a communication tool between developers and client to reach consensus on what the client's requirements are for the system. After requirements elicitation and final draft of the SRS is complete, the client should sign off on the document if the client agrees with the specification. After sign off, this document shall be used as a reference material for developers when creating the system.

1.2 Scope

NavUP is a project proposed by the University of Pretoria's department of Computer Science. The objective of the system is to provide the user with an interface from which they can select a location(building or classroom). The system will then calculate the fastest route for the user based on user specified restrictions. The system will also provide information on points of interest on campus. The system will be available to staff, students and visitors to the University of Pretoria.

The system will provide users with an easy way to traverse campus and get information on the various points of interest. This will include historical buildings, events and activities. Users will be able to set restrictions on the navigation part of the system, for example, avoiding pedestrian traffic and accessibility for disabled people. There will also be an option to save locations and routes for future use by the user.

The system will determine the users location both indoors and outdoors through various ways like wifi signal strength, gps location and crowd sourcing. This will allow users to connect to the system and provide them with the necessary information to navigate campus. This is beneficial to new students and visitors that do not know the layout of campus yet. This will also provide users with fast routes to their location when time is of the essence by avoiding pedestrian traffic and activities/events.

1.3 Definitions, Acronyms and Abbreviations

Term	Definition
NavUP	The navigation system that is proposed in this document.
User	The people that will be using the system.
Guest User	The people that will use the system, but will only have limited functionality.
Admin/Administrator	Person given permission for managing and controlling the system.
Location	A precise point on a map.
Venue	A building name or a room within the building itself
Points of interest	Various locations that may interest users to visit them.
Events and Activities	Various events/activities that may take place on the campus.
Restrictions	User selected preferences when using the navigation system NavUP.
Network	A system consisting of various interconnected computers and hardware.
Heatmaps	An indication (usually in color) of a congested or populated area.
Wifi	A facility allowing computers, smartphones, or other devices to connect to the Internet or communicate with one another wirelessly within a particular area.
Wifi hotspots	Areas where users will be able to connect to the network wirelessly.
GPS	GPS, which stands for Global Positioning System, is a radio navigation system that allows land, sea, and airborne users to determine their exact location, velocity, and time 24 hours a day, in all weather conditions, anywhere in the world.
UI/User Interface	The method of which users will interact with the system.
GIS	A geographic information system (or GIS) is a system designed to capture, store, manipulate, analyze, manage, and present spatial or geographic data.
Android	An open-source operating system used for smartphones and tablet computers.
iOS	An operating system used for mobile devices manufactured by Apple Inc.
CRUD	Shorthand for create, read, update and delete.

1.4 References

Bibliography: Kung, D.C. (2013) Object-oriented software engineering: An agile unified methodology. 2013th edn. New York: McGraw Hill Higher Education. In-line Citation: (Kung, 2013)

1.5 Overview

The remainder of the document consists of two more chapters, an appendixes and a index.

The second chapter provides the overview description of the system which gives a more explicit structure of the NavUP system's functionality in terms of the product's perspective, functions, user characteristics, constraints, assumptions and dependencies.

The third chapter provides the specific software requirements the system consists of which is the external interface, functional, performance requirements.This chapter also specifies the design constraints, software system attributes and other requirements.

The appendixes provides the subsidiary matter at the end of the document together with the index which is a list of names or words which were mentioned in the document with reference to their page number.

2 Overall Description

2.1 Product Functions

Products function

The products function enables the user to enter their personal information and interests to be used in the functionality of the system.The system enables user's to find a route based on their selected destination, preferences and interests using they personal profile. The mobile functionality is mainly for the user's while the web-based side is more for the administration purposes.

For the mobile functionality the system should allow privileged managers which may specify the location and movement activities therefore gaining access to individuals data records.The user's details will be used in tracking the user for various activities.Their personal interests are to be used in new information provided in the event of passing a point of interest, event taking place in the system or route provided to personal preferences.The result of the route should be viewed in the form of a map.The user's location , designated destination, pedestrian traffic and new information should be also visible whilst moving towards the designated area.

For the web-based functionality the system should be allow for overall data analysis.Data analysis performed should include user, performance and statistical analysis which aid in maintaining and developing the system at large.

2.2 User Characteristics

The bulk of users will consist of mobile users. Mobile users will consist of guests with the age group ranging from primary school children to students, alumni and lecturers. Basic competence with smartphones is assumed in addition to navigation skills with map based apps such as google maps.

The web based side of the system will consist of administrators, who should be able to amend, capture and delete information from the database, thus some knowledge of database management is required. The web based interface will also serve researchers who will be able to analyse data. It is assumed that the researchers have the appropriate skills to conduct their research, this includes domain specific knowledge and basic skills in querying a database.

Lastly the system aims to cater for the disabled. The disabilities which will be accommodated have not yet been fully clarified. But the same level of competence is expected from the disabled as from the general mobile user population to make use of the navigation features. This system is mostly geared toward the physically disabled in order to find the most accommodating routes to classes and buildings.

2.3 Constraints

No constraints have been voiced by the client

2.4 Assumptions and Dependencies

Factors that could affect the requirements include compliance with the Protection of Private Information Act of 2013. The POPI act is applicable since user location data gets captured and assigned to a persistent ID and then stored in a database. This means that compliance requires certain additions to functional requirements such as asking the data-subject for consent when the app is opened for the very first time and allowing withdrawal of consent and destruction of identifiable data within a reasonable time. It could also potentially restrict certain functional requirements such as surveillance and analysis.

For the system to function as expected, we must assume that there will be sufficient wifi and GPS coverage and the sensors, when integrated, will provide sufficient accuracy to serve its purpose. We also assume that map data will be available with a sufficient degree of accuracy and completeness to satisfy the requirements of the system.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interface

The user interface for NavUP application will be specifically designed with users in mind, allowing them to have the best and easiest user interface possible. The aim is to make the user interface easy to navigate so that users spend less time trying to figure out how to use it.

We will offer a menu service to allow users of different levels to access their respective sections with ease, these sections would be Student, Lecturer and Guest. If the user chooses Student or Lecturer then possible registration and login screens may be necessary. There will also be a menu service after each user has selected their particular section so as to allow the user to choose their current use of the application.

3.1.2 Hardware Interface

The hardware interface to support the NavUP application would require a smart-phone on which the application will be downloaded and run on. The smart-phone would have to meet the standards of the operating system that the application is going to be developed on,as well as a colour screen with a resolution to match that specified later.

3.1.3 Software Interface

The NavUP application would be designed to run on Android and IOS devices, this is to be confirmed at a later stage. Most devices with this platform feature touchscreens that would allow the application to run at its best. Information can be entered using the on-screen keyboard and the pre-existing touch and drag approach could be used to navigate the map interface.

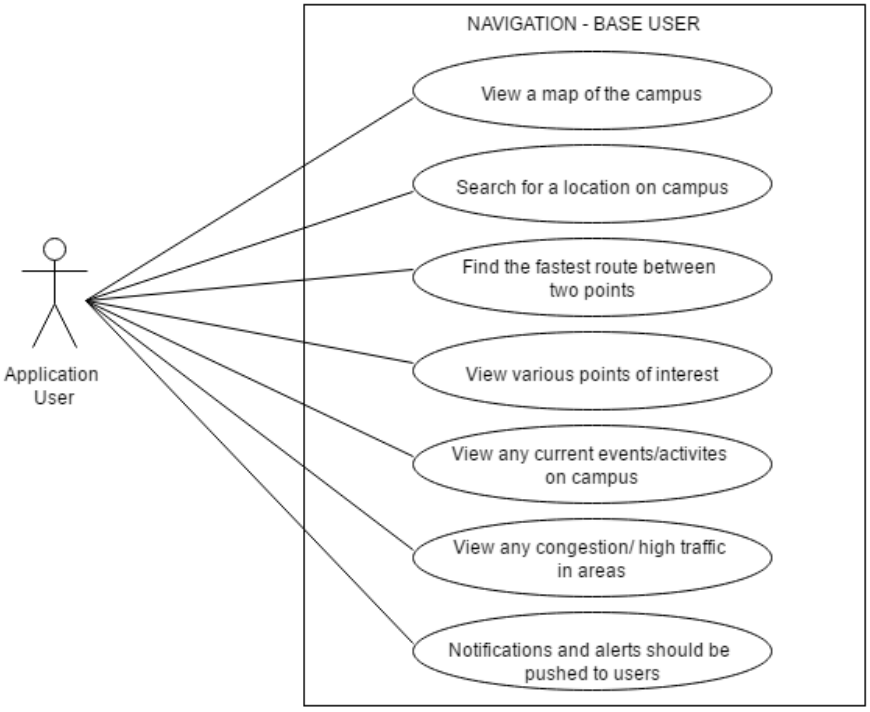
3.1.4 Communications Interface

The application would mainly use WiFi connections, GPS system and cellphone towers to triangulate and estimate the position/s of the user/s. Information would also have to be collected from the user/s, to allow this, any form of data communications would be allowed, although in essence this would be done mainly through a WiFi or cellphone data connection.

3.2 Functional Requirements

3.2.1 Navigation

Base User requirements
The base user requirements outlines the base/foundation functionality of the system. The inner workings of the system and it's subsystems will be expanded later in the document. Your typical base user will be able to do the following:



Serial:F1
Abstract: View the map of campus.
Description: The user will be able to view the map through the use of the UI. The map itself will be saved on the system enabling access to it when connected or disconnected. It will contain all locations and venues located on campus itself.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: View the map of campus.

Actor: Mobile User	System: NavUp
	0. The mobile app displays the main menu with the map and map info occupying most of the screen at the default position.
1. The user scrolls around the map	2. The app updates the map the map and map info that's in view, pulling information from the remote repository if needed and if possible.

Serial:F2
Abstract: Search for a location/venue on campus to be displayed on the map.
Description: The user needs to be able to search for a location/venue. This will involve a search bar (possible auto-complete functionality) where the user will enter the name of the location/venue. The mobile device will then search through it's database and indicate on the map when the location/venue has been found or display an appropriate prompt when the location/venue is not found.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: Search for a location/venue on campus to be displayed on the map.

Actor: Mobile User	System: NavUp
0. The user clicks on the magnification button	1. The mobile app displays a menu to type in the location and venue.
2. The user types in the location or venue	3. Using information from cache and the remote repository, the app tries to perform autocomplete based on what the user is typing.
4. The user chooses the autocomplete suggestion or types the search term out.	5. The app generate a list of candidate locations/venues that match its internal cache or the remote repository.

Serial:F3
Abstract: Find the fastest route between two points.
Description:

1. The user will be able to set the start point in the following ways:
   (a) By clicking on the map to indicate the start point or typing the name of the location into the search bar as described in another functional requirement. This should work for both online and offline.
   (b) The start point is determined by integrating information from various sensors such as wifi and cellphone tower triangulation, gps, ac-

celerometer and gyroscope. The functionality may work for both online and offline if possible.

2. The user will be able to set the end point by tapping on the map to indicate the end point or typing the name of the location into the search bar as described in another functional requirement. This should work for both online and offline.

3. After setting the start and end point the mobile app must calculate the fastest path between these two points and provide the directions to the user. If the device is offline it must make use of cached data to make an informed choice. If it is online, it must consult the remote repository for the most up to date congestion data.

Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model:Find the fastest route between two points.

Actor: Mobile User	System: NavUp
0. The user clicks on the navigation button	1. The mobile app displays a menu to type in the starting point and end point.
2. The user selects the starting point by making a mark on the map or setting it as current location of the mobile device or searching for a place on the map	3. If the mobile device is online, the remote server sends up to date congestion data to the mobile device. If the mobile device is offline, it uses cached data. In both cases the mobile phone calculate the fastest path using the congestion data, then displays the fastest path to the user.

Serial:F4
Abstract: View various points of interest.
Description: When selecting a certain building or location the mobile device should display information where applicable for the user to review. When offline the mobile device will use whatever it has in cache, if it is online it may consult the remote repository for the most up to date information.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: View Various Points of Interest.

Actor: Mobile User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects a destination	2. The mobile application will display various points of interest along the path to the destination.

Serial:F5
Abstract: View any current events or activities happening on campus.
Description: The mobile device should display various events and activities happening on campus on the current day as well as upcoming events and activities. When offline it should attempt to display activities and events that have not expired that resides in cache. When online, it should retrieve the most up to date list of events and activities from the remote repository.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: View any current events or activities happening on campus.

Actor: Mobile User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects a destination	2. The mobile application will display various activities at venues along the way to the users destination.

Serial:F6
Abstract: Users should be able to view any congestion/ high traffic in areas.
Description: When using the application in a map view of the campus, users should be able to view the amount of pedestrian traffic in surrounding areas. In addition to this any congestion in areas will also be indicated, this could be in the form of pedestrians or any other kind.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: Users should be able to view any congestion/ high traffic in areas.

Actor: Mobile User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects to view an indication of the pedestrian traffic and congestion on campus	2. The mobile application will display various points at which there is high congestion and traffic.

Serial:F7
Abstract: View the map of campus.
Description: The user will be able to view the map through the use of the UI. The map itself will be saved on the system enabling access to it when connected or disconnected. It will contain all locations and venues located on campus itself.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: View the map of campus.

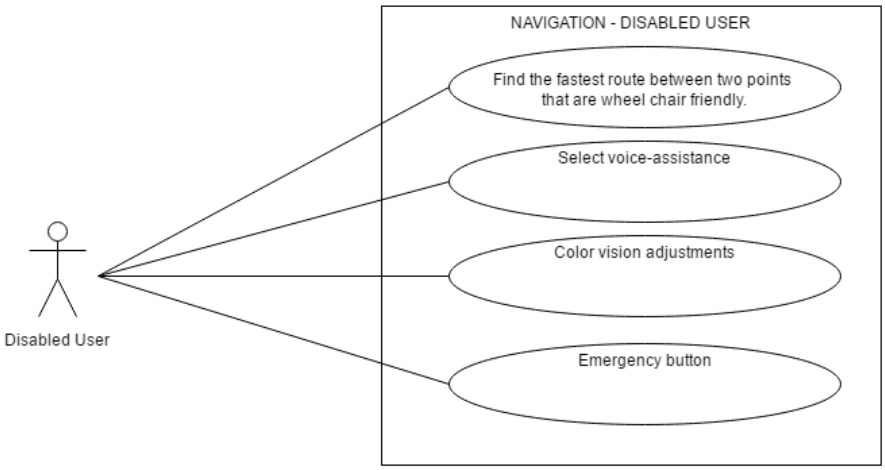
Actor: Mobile User	System: NavUp
	0. The mobile app displays the main menu with the map and map info occupying most of the screen at the default position.
1. The user scrolls around the map	2. The app updates the map the map and map info that's in view, pulling information from the remote repository if needed and if possible.

Serial:F8
Abstract: Emergency notifications and alerts should be pushed to users.
Description: The mobile app will receive broadcasts from the remote repository in the event of a emergency. In emergency situations the application could be used to give users directions to assembly places or other appropriate directions.
Pre-condition: A emergency scenario has taken place and device is connected.
Post-condition: Trivial

Actor system interaction model: Notifications and alerts should be pushed to users.

Actor: Mobile User	System: NavUp
	0. The remote repository broadcasts an emergency message to connected mobile devices.
1. The mobile application will display alerts and important messages in the form of notifications on the mobile device.	
2. The user can click on the notification to view an expanded explanation and description.	3. The repository may push emergency routes to users if needed.

Disabled User requirements
The Disabled User Requirements extends from the Logged in or Guest User Requirements. The aim of the disabled navigation is to accommodate disabled users be it students, employees or visitors to successfully use the system.



Serial:F9
Abstract: Select voice-assistance.
Description: The user can select the voice assistance function so that the mobile application may use voice-commands instead of manually selecting the mobile application's functions.
Pre-condition: Trivial
Post-condition: Voice assist is enabled.

Actor system interaction model: Select voice-assistance.

Actor: Disabled User	System: NavUp
	0. The mobile app displays an option to choose a voice command function to communicate with the user.
1. The user selects to employ the functionality.	2. The application communicates to the user by use of audible commands as well as a visual representation.
3. The user may respond in the form of a voice command or by using the touch interface on the mobile device.	

Serial:F10
Abstract: Color blind.

Description: The user is able to adjust the color displayed on the mobile device to accommodate various kinds of colour blindness.
Pre-condition: Trivial
Post-condition: Mobile app has changed its colour scheme to accommodate the selected type of colour blindness.

Actor system interaction model: Color blind

Actor: Disabled User	System: NavUp
	0. The mobile app displays a menu to choose a colour scheme that accommodates the user.
1. The user selects their type of colour blindness.	2. The mobile app adjusts the colour scheme to accommodate the type of colour blindness.

Serial:F11
Abstract: View the map of campus.
Description: This feature extends on the base user functionality for viewing a map but also accommodates the selected disability. This can be in the form of changing the colour scheme for the colour blind or ramp locations for the physically disabled. It should also indicate the location of the disability unit on campus.
Pre-condition: User cannot be blind.
Post-condition: Mobile app has adjusted its output to suite the selected disability.

Serial:F12
Abstract: Search for a location/venue on campus to be displayed on the map.
Description: This feature extends on the base user functionality for searching for a location. The mobile app should allow searching of locations by voice command.
Pre-condition: Disablility only applies to disabilities that require voice command.
Post-condition: Trivial

Serial:F13
Abstract: Find the fastest route between two points that are wheel chair friendly.
Description: This feature extends on the base user functionality for fastes route finding. The mobile application will navigate a path for the user that is disabled and wheelchair friendly, allowing them to navigate with minimal obstruction.
Pre-condition: The user has indicated his/her disability in the accessibility options.
Post-condition: Trivial

Serial:F14
Abstract: View various points of interest.
Description: This feature extends on the base user functionality of viewing various points of interest but includes the functionality to accommodate certain disabilities. In the case of the blind, the mobile app will ask the user if he/she would like to hear about the points of interest.
Pre-condition: Trivial
Post-condition: Trivial

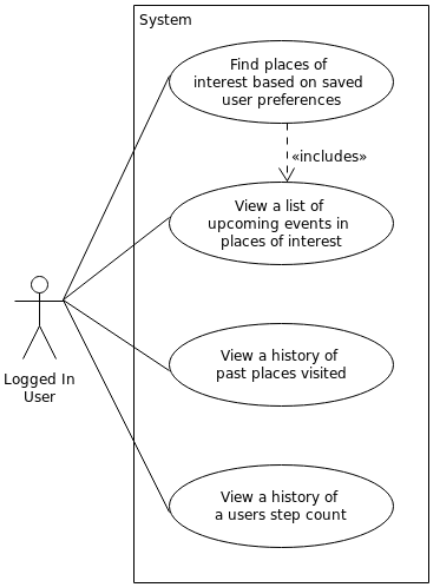
Serial:F15
Abstract: View any current events or activities happening on campus.
Description: This feature extends on the base user functionality for viewing current events and activities but includes the functionality to accommodate certain disabilities.In the case of the blind, the mobile app will ask the user if he/she would like to hear about any current events or activities happening on campus.
Pre-condition: Trivial
Post-condition: Trivial

Serial:F16
Abstract: Emergency button.
Description: The user can use this special feature to alert the system in the event of an emergency so that nearest wheel chair friendly route to exit point for the disabled user can be displayed.The feature uses the current location of the user to find the fastest wheel chair friendly route.
Pre-condition: Trivial
Post-condition: Emergency personnel are sent to assist the user.

Actor system interaction model: Emergency Button

Actor: Disabled User	System: NavUp
	0. The mobile application will display an emergency button or respond to a push event from the remote repository.
	1. The emergency personnel are notified and are provided with the location of the user.

Logged-in User requirements
The logged in user will have certain privileges over and above those of the base and guest users. There are two types of logged in users, namely students and lecturers who would log into the system using some form of user-name and password. The typical logged in user will be able to do the following:



Serial:F17
Abstract: Find places of interest based on saved user preferences.
Description: The users will be given the option to populate a list of preferred preferences and places of interest. The system will then generate and display a list, when needed, of the places of interest and various events that are happening currently. This list that is provided to the user is unique and tailored to them.
Pre-condition: User needs to be logged in.
Post-condition: Trivial

Actor system interaction model: Find places of interest based on saved user preferences.

Actor: Logged In User	System: NavUp
	0. The system gathers all information on the logged in user.
1. The user opens prefrences on points of interests.	2. The mobile application will display various points of interest.
3. The user selects various prefrences.	4. The system updates the user's data on prefrences.
	5. The mobile application generates and displays a list of places of interest.

Serial:F18
Abstract: View a list of upcoming events in places of interest.
Description: Using the saved list of preferences and places of interest that the user has provided, the system will automatically generate a list of the upcoming events that are taking place at these places and the user will be able to view them in a time-line manner and set reminders for these events.
Pre-condition: User needs to be logged in.
Post-condition: Trivial

Actor system interaction model: View a list of upcoming events in places of interest.

Actor: Logged In User	System: NavUp
0. The user opens upcoming events and activities.	1. The mobile application will display various upcoming events and activities.
3. The user sets reminders to preferred events and activities.	4. The system saves the reminders in the database.
	5. The mobile application notifies the user when an event is happening at appropriate times.

Serial:F19
Abstract: View a history of past places visited.
Description: Users will be able to view a history of the past events that they have attended. This will include the type of event that took place and the location that the event was at.
Pre-condition: User needs to be logged in.
Post-condition: Trivial

AView a history of past places visited.

Actor: Logged In User	System: NavUp
0. The user opens their history.	1. The mobile application will display various locations and events the user has attended/visited.

Serial:F20

Abstract: View a history of a users step count.
Description: The system will will track and record the step-count of each user, this history will be viewable by the user according to the time-line that the user will be able to specify.
Pre-condition: User needs to be logged in.
Post-condition: Trivial

Actor system interaction model: View a history of a users step count.

Actor: Logged In User	System: NavUp
0. The user opens their profile/user information.	1. The mobile application will display statistics of the user's step count.
3. The user adjusts the timeline to view the amount of steps taken.	4. The mobile application updates the information accordingly.

Guest User requirements
In regards to the Guest Users, they will have the same requirements of a Base User (See Base User Requirements). They won't have an associative ID linking them to the database seeing as they will not be logged in. As soon as they do so they will be seen as a Logged In User. The only possible added functionality will be catering to people who don't visit campus often. See below mentioned for expanded requirements:

Serial:F21
Abstract: A better view of points of interest on the map.
Description: A guest user will most likely be someone visiting campus. It will thus be beneficial to the user if the points of interest are more visible to them. This will allow them to easily find any usefull information they might need. This can be done by highlighting or setting points on the map itself for the user to select and read further information on that point.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: A better view of points of interest on the map.

Actor: Mobile Guest User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects a destination	2. The mobile application will display in a more visual way various points of interest along the path to the destination.

Actor system interaction model: A better view of points of interest on the map.

Actor: Mobile Guest User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects a location on the map	2. The mobile application will display in a more visual way various points of interest in the surrounding areas.

Serial:F22
Abstract: A better view of activities/events happening on campus.
Description: Campus plays host to various events and activities like Business days or even music orchestras. This means that campus will have a number of guests unfamiliar with the layout of campus or the location of these activities/events. The guest user needs to be able to view the activities/events in a more visible way. This will allow them to navigate to their destination and/or

read more information on it. This can also be displayed on the map or even on a sub-menu for ease of access.
Pre-condition: Trivial
Post-condition: Trivial

Actor system interaction model: A better view of activities/events happening on campus.

Actor: Mobile Guest User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects a destination	2. The mobile application will display in a more visual way various events and activities along the path to the destination.

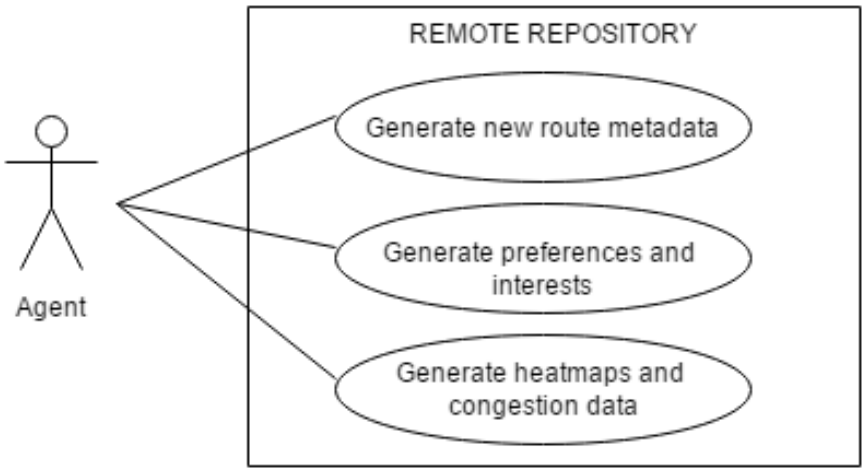
Actor system interaction model: A better view of activities/events happening on campus.

Actor: Mobile Guest User	System: NavUp
	0. The mobile app displays a map of the campus.
1. The user selects a location on the map	2. The mobile application will display in a more visual way various events and activities happening in the surrounding areas.

e

3.2.2 Agent Requirements

The agent will be an autonomous program acting as a special user that looks at the raw data stream and existing metadata from the remote repository and generates new metadata which is pushed back into the repository. The metadata is any processed data that the system needs to serve its purpose, such as congestion data, heat maps and new routes.



Serial:F23
Abstract: Generate new heat maps and congestion data
Description: Trivial
Pre-condition: The agent is authenticated and has sufficient privileges to perform its job.
Post-condition: Trivial

Serial:F24
Abstract: Generate new route metadata
Description: The agent can process raw location data from the remote repository to learn new routes through campus. Because of this, the workload of data capture is reduced since older students and lectureres on campus will map the routes out for the system by simply walking their usual routes through campus.
Pre-condition: The agent is authenticated and has sufficient privileges to perform its job.
Post-condition: Trivial

Serial:F25
Abstract: Generate preferences and interests

Description: There will be an agent that learns the preferences and interests of the data-subjects from existing metadata and the raw data stream from the remote repository.
Pre-condition: The agent is authenticated and has sufficient privileges to perform its job.
Post-condition: Trivial

3.2.3 WebUI

This pertains to all interaction with the remote repository via a web base user interface(UI).

Administration
Serial:F26
Abstract: Authentication
Description: Trivial
Pre-condition: Trivial
Post-condition: Trivial

Serial:F27
Abstract: Administrator CRUD data records
Description: User's information should be updated in database when administrator is done editing.
Pre-condition: Trivial
Post-condition: Trivial

Serial:F28
Abstract: Administrator block or unblock users.
Description: Trivial
Pre-condition: Trivial
Post-condition: Trivial

Serial:F29
Abstract: Administrator CURD user accounts.
Description: Trivial
Pre-condition: Trivial
Post-condition: Trivial

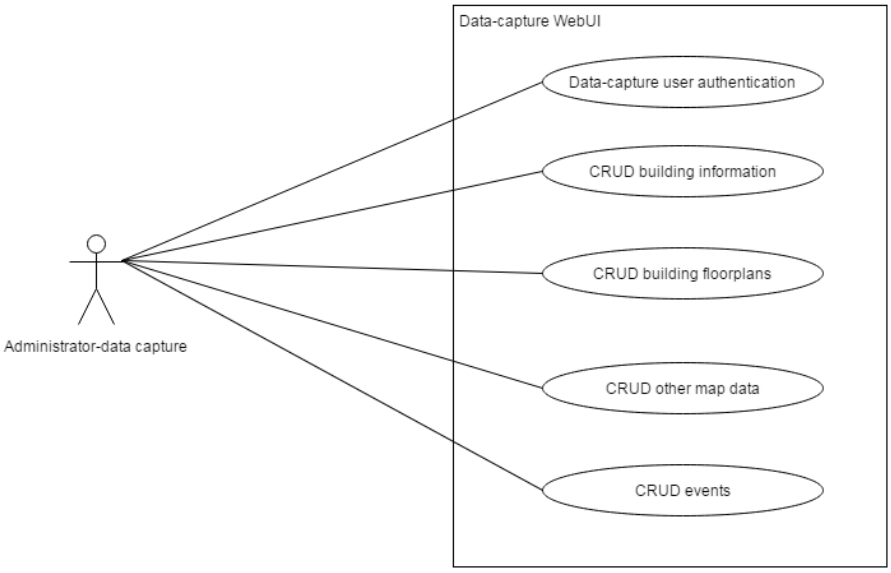
Serial:F30
Abstract: Administrator view system performance
Description: Trivial
Pre-condition: Trivial
Post-condition: Trivial

Serial:F31
Abstract: Display users information on request
Description: Administrator has to show the user all the information they have on a user when requested by the user according to the POPI Act.Administrator has to enter user's user-name to get the all the details of the user and return them in the form of a formal document via the users email address.
Pre-condition: Trivial

Post-condition: Trivial

Serial:F32
Abstract: Delete users information on request
Description: Administrator has to delete all of the user's information they have on a user when requested by the user according to the POPI Act.
Pre-condition: Trivial
Post-condition: Trivial

Data Capture
The data-capture adds the system's metadata such as building location , history ,campus events etc. Since the system will provide indoor navigation the metadata indoors will not be entirely visible on the external map hence upon entrance of a building the map needs to change to display a buildings internal map.



Serial:F33
Abstract: Data-capture user authentication.
Description: The usual things that are required for authentication.
Pre-condition: Trivial
Post-condition: Trivial

Serial:F34
Abstract: CRUD building information.
Description: CRUD of building information. This includes building names, positions and historical trivia.
Pre-condition: Trivial
Post-condition: Trivial

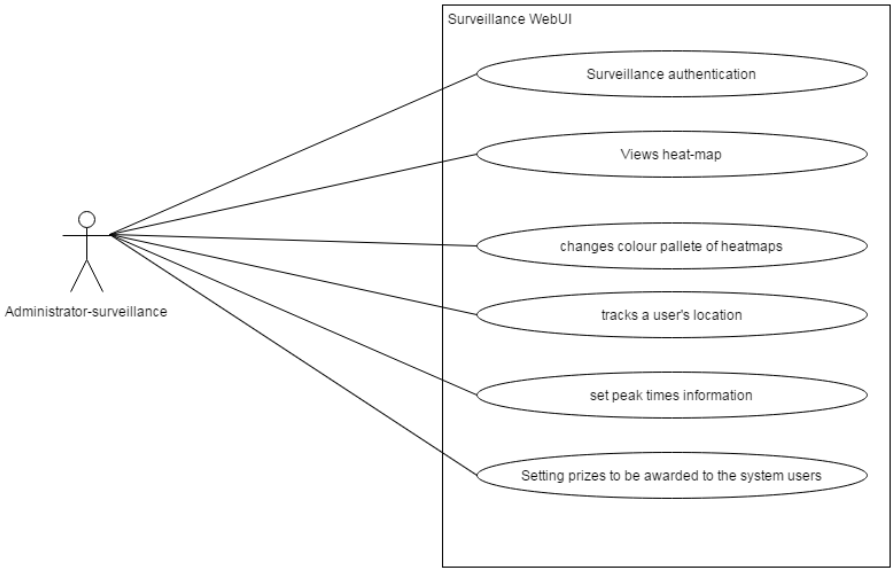
Serial:F35
Abstract: CRUD building floorplans.
Description: CRUD of building floorplans include location of rooms and disability ramps as well as conventional stairs.
Pre-condition: Trivial

Post-condition: Trivial

Serial:F36
Abstract: CRUD other map data.
Description: This includes routes, walkways, roads, entrances and other landmarks such as artwork.
Pre-condition: Trivial
Post-condition: Trivial

Serial:F37
Abstract: CRUD events.
Description: Events and Activities on campus should be added, removed and edited as required. This can be automated or done by administrators that have write access to the database.
Pre-condition: Trivial
Post-condition: Trivial

Surveillance
Surveillance allows the administrator to make changes to the heat map,track user and set up prizes for the user.



Serial:F38
Abstract: Surveillance authentication
Description: Trivial
Pre-condition: Trivial
Post-condition: Trivial

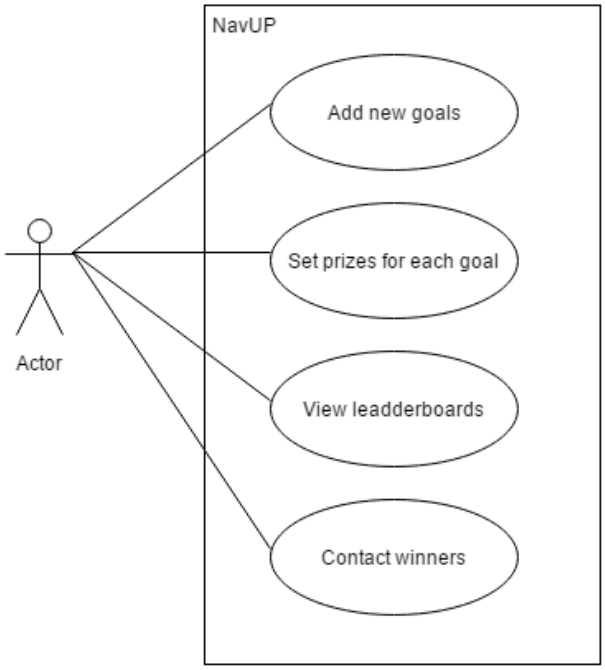
Serial:F39
Abstract: Views heat-map
Description: The remote repository returns metadata for generating heatmaps in browser
Pre-condition: Trivial
Post-condition: Trivial

Serial:F40
Abstract: Changes to the colour pallete of heatmaps
Description: Trivial
Pre-condition: Trivial
Post-condition: Trivial

Serial:F41

Abstract: Track a user's location
Description: Surveillance user has to select track user option. Surveillance user requires user-name of the user that should be tracked. Repository returns current location of user if the user is connected. Surveillance can view users current location and see distance traveled by user around the campus.
Pre-condition: Trivial
Post-condition: Trivial
Serial:F42
Abstract: Analyze certain location and set peak times information
Description: Surveillance user has to select the peak times option. Surveillance user submits a request to compare system user location times. Surveillance user sets peak times information. Surveillance user submits peak times to database.
Pre-condition: Trivial
Post-condition: Trivial
Serial:F43
Abstract: Setting prizes to be awarded to the system users
Description: Surveillance user has option to view statistical data of the users to select the winner for a specific prize. Surveillance can set a prize for the winner.
Pre-condition: Trivial
Post-condition: Trivial

Goals and Prizes
The Web User Interface will be used by system administrators to manage the system, as such the Web UI for goals and activities will be managed on this interface.



Serial:F44
Abstract: CRUD goals and activities.
Description: The system shall provide an interface for an admin user to add and create goals for the users to aim to achieve. The admin must be able to set what the requirements to reach the goals are, and the time frame allocated for achieving these goals, such as daily or weekly. The admin will also be able to see which users have met the goals that have been set. This data should be able to be sorted by time taken to reach the goal or by who achieved the highest score in order to decide on who can win a prize.
Pre-condition: The admin user must be authorised in order to make changes to the system.
Post-condition: Trivial

Serial:F45
Abstract: CRUD prizes.
Description: The system shall provide a way to add prizes that may be won

by users as an incentive for completing goals. Each goal will be able to have a prize associated with it so that the user may see what they are eligible to win for completing the goal. Harder goals to achieve can then have better prizes associated with them to add to the incentive. The system shall display users who have achieved a goal and are eligible to win a prize. The choosing of a winner for a goal is at the discretion of the admin. Contact options for user who have completed the goal will be available to the admin in order to inform them of their prize.
Pre-condition: User must be authorised.
Post-condition: Trivial

3.2.4 Goals and Prizes

One of the concerns raised by the developing team was the issue of the life time of the system/mobile application. Students might not see the benifit of the mobile apllication seeing that the majority of students know the layout of campus and where venues are located. The client mentioned incentive in the form of a Goal and Reward System. The client did not expand on this issue so this might be subject to change. Here are a few requirements if such a system was implemented in NavUP:

Serial:F46
Abstract: Track the user's steps or distance walked.
Description: The mobile application will have to keep track of the users steps or measure the distance they have walked. This data will be saved and used when tracking the goals and achievements of the user. It will also be visible to the user for them to see how much/far they have walked.
Pre-condition: The user needs to be logged in.
Post-condition: Trivial

Actor system interaction model: Track the user's steps or distance walked.

Actor: Mobile User	System: NavUp
	0. The mobile application will keep track of the users steps taken and/or distance traveled.
	1. The mobile application saves this data and the system stores it on the database.

Serial:F47
Abstract: Display the various fun run routes on campus.
Description: Campus has varius fun run routes marked by painted animal footprints. These footprints have in time faded away, therefore it will be benefitial to display these routes to the user when they want to reach a certain goal or even just have a jog for exercise. This can be displayed on the map as a navigational route.
Pre-condition: The user needs to be logged in.
Post-condition: Trivial

Actor system interaction model: Display the various fun run routes on campus.

Actor: Mobile User	System: NavUp
0. The user selects the fun run routes.	1. The mobile application displays the various fun run routes on the map.
2. The user chooses a fun run route to complete.	3. The mobile application starts the navigation of that route.
	4. The mobile application tracks the user's progress until the route is finished.

Serial:F48
Abstract: Set goals for the user to finish.
Description: The client has not expanded on this section, therefore there can only be assumed that the mobile app will have certain goals/achievements listed. This will most likely be in the form of a certain distance walked or a number of steps taken. These goals will have to be set by the administrators and will have various levels of completion, each level awarding a prize to the user.
Pre-condition: The user needs to be logged in.
Post-condition: Trivial

Actor system interaction model: Set goals for the user to finish.

Actor: Mobile User	System: NavUp
0. The user selects the goals on the mobile application.	1. The mobile application displays the various goals for the user to complete.
2. The user selects a goal to complete.	

Serial:F49
Abstract: Notify the user when they have reached/completed a goal.
Description: The mobile app will have to notify the user when they have reached a certain goal or even a user's personal goal. This will allow the user to move to the next goal. This can be done via push notifications. The goals can also be tracked by the user to see how far they are from completing it.
Pre-condition: The user needs to be logged in. The user needs to maintain some form of connection.
Post-condition: Trivial

Actor system interaction model: Notify the user when they have reached/completed a goal.

Actor: Mobile User	System: NavUp
	0. The mobile application tracks the user's progress on certain goals.
	1. The mobile application notifies the user when a goal has been completed.
2. The user selects the next goal to complete	3. The mobile application starts to track the progress of the next goal.

Serial:F50
Abstract: Display the user's statistics for review.
Description: The mobile app will need a statistics section where the overall progress and steps taken/distance traveled will be recorded and displayed. This may include maximum distance walked in a day to average step count depending on the clients final wishes on this section.
Pre-condition: The user needs to be logged in. The user needs to maintain some form of connection.
Post-condition: Trivial

Actor system interaction model: Display the user's statistics for review.

Actor: Logged In User	System: NavUp
0. The user opens their profile/user information.	1. The mobile application will display statistics of the user's step count.
2. The user adjusts the timeline to view the amount of steps taken.	3. The mobile application updates the information accordingly.

Serial:F51
Abstract: Notify the user when they have been awarded a prize.
Description: The mobile app will also need to notify the user when they have been awarded a prize. These prizes will be the responsibility of the client and it is up to them to set when such a prize is rewarded. The notification will then display a message of what the user has won.
Pre-condition: The user needs to be logged in. The user needs to maintain some form of connection.
Post-condition: Trivial

Actor system interaction model: Notify the user when they have been awarded a prize.

Actor: Logged In User	System: NavUp
	0. The mobile application will track the progress of a prize.
	1. The mobile application will notify the user when they have been rewarded a prize.
	2. The mobile application will track the progress of the next prize.

Serial:F52
Abstract: Display information/guide of how users can collect their prizes.
Description: The client will have to provide the information needed to collect a prize and where to do so. This information will then be displayed to the user along with contact details if anything is uncertain. Again this is up to the client and how they want to implement this section, for now only assumptions can be

made.
Pre-condition: The user needs to be logged in. The user needs to maintain some form of connection.
Post-condition: Trivial

Actor system interaction model: Display information/guide of how users can collect their prizes.

Actor: Logged In User	System: NavUp
	0. When a prize has been awarded the mobile application will display information on how to collect the prize
	1. The mobile application will also display contact details to the user for any queries.
2. The user follows these instructions to claim their prize.	

3.3 Traceability Matrix

Functional Requirements	Mobile	Web	Agent	Repository
F1	X			
F2	X			
F3	X			
F4	X			X
F5	X			X
F6	X		X	
F7	X		X	X
F8	X	X		
F9	X			
F10	X			
F11	X			
F12	X		X	X
F13	X		X	
F14	X		X	
F15	X			X
F16	X	X		
F17	X		X	X
F18	X	X		X
F19	X		X	
F20	X	X	X	X
F21	X			X
F22	X			X
F23			X	
F24			X	X
F25			X	X
F26	X	X		
F27		X		X
F28		X		X
F29		X		X
F30		X		
F31		X		X
F32		X		X
F33		X		X
F34				X

Functional Requirements	Mobile	Web	Agent	Repository
F35	X	X		X
F36				X
F37		X		X
F38		X	X	
F39	X	X		
F40	X			
F41		X	X	
F42		X	X	
F43		X		
F44		X		
F45		X		X
F46	X	X		
F47	X			X
F48		X	X	
F49	X		X	
F50	X			
F51	X			
F52	X			