CECS 491A - Sec 6 - Business Requirements Document

Project Name: ArrowNav

Team Longhorn:

Brayan Fuentes

Christian Lucatero

Curtis Nishihira

Miguel Zavala

Spencer Gravel (Team Leader)

December 15, 2021

Business Requirements Document (BRD)

Project: ArrowNav

December 2021

Version 1.1

1 Document Revisions

Date	Version Number	Document Changes
10/06/21	1.0	Initial Draft
12/10/21	1.1	Document Revision
02/01/22	1.2	Document Revision

2 Approvals

Role	Name	Title	Signature	Date
Client	Vatanak Vong	Professor/ Client		09/24/21
Client	Vatanak Vong			

3 Introduction

3.1 Project Summary

3.1.1 Objectives

· Help students navigate their campus with a location based navigation service that can generate routes to desired locations and calculate it's ETA.

· Help with student's productivity through features such as schedule integration, the student wellness hub, showing availability in capacity buildings and the quick find feature.

· Make the application enjoyable to use daily through the reward's system.

3.1.2 Background

The project came to be proposed through identifying a need for students to increase their productivity through the form of campus companion. This campus companion revolves around its navigation features which would help with the student's productivity while at the same time familiarizing them with their campus.

3.2 Project Scope

3.2.1 In Scope Functionality

- Display locations and traffic on campus for navigation
- Quick Find Feature for locations on map
- Display availability of capacity building
- Schedule integration
- Reward System
- Student Wellness Hub
- Authentication
- Authorization
- Logout
- Registration
- Account Deletion
- User Management
- Usage Analysis Dashboard
- Archiving

- User Privacy
- Error Handling
- UI/UX

3.2.2 Out of Scope Functionality

· Branch out the functionality of the application to campuses across the state
· Live updates of foot traffic and building capacities.
· Running advertisements for on-campus activities, shops, restaurants, and clubs.

4 Requirements

4.1

Functional:

Requirement: Display Locations and Routes on Interactive Map

Scope: A virtual map UI will be implemented showing the map of the campus with all modeled locations and routes on campus as well as the location of the user. The locations that are modeled are classroom buildings, administrative buildings, enrollment services, financial services, the Horn Center, tutoring centers for the various colleges of the university, health services building, physical health services, mental health services, Student Union building, gym, designated recreational areas, restaurants, cafeterias, shops, theaters, auditoriums, arenas, sporting fields, aquatic centers, parking structures, parking lots, water refill stations on campus and drinking fountains, gendered, unisex, and family bathrooms, family rooms, significant landmarks such as statues and water fountains found on campus and all routes and streets within and immediately around the campus. The map may be traversed by being able to scroll through it and zoom it in and out. Modeled Locations on map display info when selected such as hours of operation and location name and the routes always display information showing their foot traffic and denoting the type of transportation allowed on these routes.

Dependency: This feature is not dependent on any other feature in the application, every other feature in the application is dependent on this feature being implemented as

other features constantly reference or redirect to locations modeled on the interactive map.

Value: This feature helps the user know where everything is on campus allowing them to explore the campus through the map and shows them where a preferred location they would like to travel to is on campus.

Non-functional:

- Locations and routes displayed within 5 seconds on the application.
- Locations on the application must be readable by any user.
- Access to the locations and routes feature will always be available for the user to access (unless otherwise specified).
- The map has to be responsive to the actions of the users when navigating such as scrolling and zooming in on the map.

4.2

Functional:

Requirement: Route generation

Scope: Generate route from the user's current location to any preferred location through the routes that are modeled on the map when the location is selected anywhere in the app and the route calculates the ETA of routes generated by taking into account foot traffic (which comes from a default model of foot traffic data then later on baseline data gathered from a survey), distance and transportation method.

Dependency: Route generation is dependent on the Interactive map feature as it uses the user location, modeled routes and modeled locations from the interactive map to create a route to a desired location. The foot traffic used to calculate ETA will eventually be dependent on baseline data gathered through surveying users when the default foot traffic model is phased out.

Value: Route generation helps a user by navigating them to a preferred location on campus and its ETA information allows them to arrive at their destination on time aiding students with their time management.

Non-functional:

- Routes are generated within 5 seconds.
- Viable routes are generated on the application that the user can take.

- The routes have to be responsive to the actions of the users when navigating such as scrolling and zooming in on the map.
- User interaction with routes will yield results.

4.3

Functional:

Requirement: Show availability in capacity buildings

Scope: Shows availability in the three most important capacity buildings on campus that students can utilize: the USU, library, and gym. Availability will be determined through a prediction model based on data that's already available then based on baseline data gathered through user surveys.

Dependency: Availability of capacity building will be shown when they are selected on the virtual map therefore it is dependent on the virtual map feature as information regarding building capacity is linked to the modeled location of these capacity buildings on the map. Information provided is also dependent on the data gathered by users in surveys to determine the availability in capacity buildings.

Value: This feature is meant to save users the potential time wasted should they arrive at a location (such as the gym or library) and not be able to use the resources they offer. Time is a nonrenewable resource and therefore is crucial in optimizing its use.

Non-functional:

- Availability for capacity buildings will load within 5 seconds.
- Availability range for capacity buildings will be based on maximum capacity of the building itself (this value is determined by the International Building Code for each building).
- User interaction with the application will display the appropriate information.

4.4

Functional:

Requirement: Quick Find feature

Scope: The Quick Find feature has an alphabetical list of all locations on campus that are modeled on the interactive map for the user to scroll through to find a specific

modeled location and a text search function to find a specific modeled location and select it so that the app will highlight the location on the map. Once highlighted on the map the user may select it to generate a route using the route generation function on the interactive map.

Dependency: This function is dependent on the interactive map as the locations that can be found through the Quick Find Feature have to be modeled on the interactive map in order for it to be highlighted on the map when selected. Once highlighted on the map the user can select it to generate a route.

Value: If a user has a specific location in mind the Quick Find feature can help them find it in a more efficient manner than scrolling and looking for it in the virtual map. This feature makes it easier for users to find specific locations on campus without wasting time scrolling through the map trying to find said location.

Non-functional:

- The alphabetical list should load up within 5 seconds when the quick find feature is selected
- When typing up certain characters the quick find feature will pull up locations that match these characters within 5 seconds
- Interaction with Quick Find will allow the user to scroll down the list of options available from keywords input into the search bar.

4.5

Functional:

Requirement: User Account System

Scope: There will be a user account system in which a user can create an account to log into the app with a distinct username and a password of choice. The user is allowed to change their username if it is a distinct username and they may change their password as well. This user account will be used to track their reward system points and to associate with their schedule of classes.

Dependency: This function is not dependent on any other features in this app but the schedule integration feature as well as the reward system feature is dependent on this due to the points and schedule being associated with a user.

Value: The value of this app is that when a user logs in to their account the schedule that they created will be in the app and the reward points that they have earned will be loaded in their account.

Non-functional:

- User account system will load within 5 seconds.
- User account information, such as the username and password, is kept confidential.
- Users are able to change their username and password for their account as many times as they wish.
- Actions to either change username or password, and/or logout will happen according to the user.

4.6

Functional:

Requirement: Schedule Integration

Scope: In the schedule integration feature the user can customize their schedule by adding, editing or deleting classes. When creating a class the user must insert the name of the class, the date and time of the class and the building in which the class takes place so that it may be highlighted on the map on the day that the class takes place.

Dependency: The schedule integration is dependent on the interactive map feature as it needs for the buildings in which classes take place to be modeled on the map so that it may be highlighted on the map on the days the class takes place. Once highlighted on the map the user can select it to generate a route. Once a schedule is created it is associated with the specific user that created it therefore this feature is also dependent on the user account feature.

Value: This will show the user where on campus their classes will take place on the day the classes take place in order to help the student navigate through campus and gives them a reminder of which class they have for the day so that they do not forget and miss a class.

Non-functional:

- User schedules will load within 5 seconds.
- Users can add as many classes to their schedule as they want.
- Users are not required to have classes in their schedule.

- A user's schedule is kept confidential and is not made public to other users.
- Schedule integration is available for all user accounts.
- Interaction with classes on the schedule will be carried out according to the user's input.

4.7

Functional:

Requirement: Student Wellness Hub

Scope: The Student Wellness hub has three sections to it, the mental health section, the physical health section and the hydration reminder. The mental health section highlights recreational locations for stress relief, psychological services and services for mental disabilities on the map. The physical health section highlights medical service locations, recreational locations for fitness and services for physical disabilities on the map. Recreational time for the improvement of student wellness can be scheduled weekly by the user through setting up reminders. The hydration reminder section takes in the weight of the user to calculate their hydration baseline and allows them to edit the amount of water they will drink during their time on campus. They can customize their reminders by dividing the amount of reminders they want in between the time they are on campus. Once a reminder is sent the nearest water fountain or water refill station will be highlighted on the interactive map.

Dependency: The Student wellness hub is dependent on the interactive map feature as all locations from all three sections whether it be the mental health section, the physical health section and the hydration reminder section have to be modeled on that interactive map first for it to be highlighted when accessed from the Student Wellness hub. Once highlighted on the map the user can select it to generate a route.

Value: The student wellness hub will aid students in maintaining their health. Although this feature does not guarantee that the student's wellness will be improved, the information that this feature provides can provide assistance to the student should they choose to pursue it.

Non-functional:

- Student Wellness Hub and all of its features will load within 5 seconds after interaction.
- All users have access to the Student Wellness Hub.
- Interaction with features of the Student Wellness Hub will direct the user to the correct feature.

4.8

Functional:

Requirement: Reward System

Scope: Developers create new weekly locations objectives and random easter eggs which rewards user's with points when they complete the objectives or find the easter eggs. These points can be used to redeem prizes which are discounts for use on stores and restaurants on campus.

Dependency: The reward system will be most beneficial by using the interactive map UI in order to determine the location of the easter egg based on the clue(s) given. If the user has found the easter egg on campus, a code from the easter egg, when input into the rewards system, will award said points to the user. Since reward points are achieved and counted to the account of the user that achieved them, the reward system feature is dependent on the user account feature.

Value: This introduces a gamification strategy to the application in order for the user to come back and continue using the application on a daily basis with the incentive of prizes.

Non-functional:

- Rewards features will load within 5 seconds.
- Users will be able to interact and acquire deals on the reward system.
- All deals across all user devices will be the same. Not one device will be offered a different or specific deal compared to other users.
- Deals the user has acquired will be specific to the user’s account.

4.9

Functional:

Requirement: User Surveys to build baseline data

Scope: Users are anonymously surveyed by the application to build baseline data for foot traffic and building capacity. The survey asks the user for their location, whether it be a specific route or one of the three capacity buildings, to state the date and time they visit their location and to rate a route on an incremental level of "Peak Times","Off-Peak

Hours","Medium-Peak Hours and to rate a capacity building on a scale of 1-3 to represent completely empty (1), medium filled (2), or completely full (3) based on criteria given to them by the app.

Dependency: This function is based on both the route generating feature and the capacity buildings feature as the baseline data that it generates through it's surveys will be used to display the foot traffic and the availability in the capacity buildings.

Value: This feature will help the app display up to date and accurate information for the foot traffic and capacity building which will help students with their time management through the other features that this feature is dependent on.

Non-functional:

- Questions regarding availability in capacity buildings will load within 5 seconds of entering said building.
- User responses regarding building capacity are kept confidential and anonymous.

4.10

Functional:

Requirement: User Privacy

- EULA per GDPR or California Consumer Privacy Act (CCPA) / California Privacy Rights Act (CPRA)
- Opt-out of user data collection and selling of user data
- Explanation of use of data
- Deletion of user data / user account

Scope: All user related data within the system

Dependency: Security protocols established by the application and existing security between the application and the database.

Value: Establishing user privacy ensures that private data is confidential to the user and not made available to others.

Non-functional:

- User privacy will always remain active.
- Notification of privacy breach will be displayed.

4.11

Functional:

Requirement: Error Handling

Allowed System Failures:

i. Web Server loses internet access

ii. Cloud/Host Provider outage

- System failures from this feature must not result in the system going offline

Scope: The entire system

Dependency: Information/data entered into the application.

Value: In the event that information is not correctly read, or the code does not compile correctly in the application, error handling ensures that the application does not shutdown but instead handles the error and returns the application to a safe and stable state.

Non-functional:

- Error handling will always remain active.
- Error handling will account for any/all potential system errors that can occur while the application is running.

4.12

Functional:

Requirement: UI/UX

- All text must be in the selected language and culture setting
- All formats align with the selected unit of measurement
- All views must not require assistance from another human understand how to interact with the view
- All system messages to the user must be displayed in the default culture settings for non-Authenticated users and the selected culture settings for Authenticated users
- All system messages must appear within 5 seconds of the resolution of an operation
- System failures from this feature must not result in the system going offline

Scope: All features that requires user interaction within the system

Dependency:

Value: Being able to easily interact with the application can only be attained through the user interface and user experience design. Other forms of interaction on the application include logging into and out of a user's account or accessing settings (rather than the features of the application itself).

Non-functional:

- UI/UX loads within 5 seconds of prompted use.
- UI is displayed clearly.

4.13

Functional:

Requirement: **Authentication**

The user must provide a valid security credentials whenever attempting to authenticate with the system

- Valid security credentials consist of a valid username and valid time-based one time password (OTP)
  - Valid usernames will consist of the following:
    i. a-z
    ii. 0-9
    iii. .,@!
  - OTP is defined in NIST SP 800-63b section 5.1.4.1
    i. OTP is changed upon every successful use
    ii. OTP expires every 2 minutes
    iii. OTP must be at minimum 8 characters
    iv. Valid characters will consist of the following:
      a. a-z
      b. A-Z
      c. 0-9
- A maximum of 5 failed authentication attempts within 24 hours for the same account before account is disabled
  - 24 hour timer begins on the first failed authentication. If the account was not disabled after 24 hours, then the the fail count resets to 0.
  - Account is locked until a valid account recovery is performed by user or by system admin. Upon successful account recovery, the failed count resets to 0.
  - For each failed attempt, the account undergoing authentication and the IP address that initiated the authentication request will be recorded.
- System failures from this feature must not result in the system going offline

Scope: Any user attempting to use the system

Dependency: Valid credentials input by the user.

Value: Accessing privileges and determining if they correctly identify a user of the system so that the user may then access the application. Acts as a form of application security.

Non-functional:

- Authentication is always running when the user inputs credentials into the application.
- Authentication will successfully return either access or no access within 5 seconds.

4.14

Functional:

Requirement: Authorization

- By default, unauthenticated users will only be given access to resources or functionalities that does not require knowledge of user's identity
- The operation and timestamp of each unauthorized access will be recorded by the system
- The system must prevent unauthorized users from viewing, modifying or deleting any protected data (scalar or aggregate data)
- The system must prevent unauthorized users from executing any protected functionality
- The system must prevent unauthorized users from viewing or interacting with any protected views
- Any user access modifications should be active upon the next successful authentication by user
- System failures from this feature must not result in the system going offline

Scope: Any user attempting to use the system

Dependency: Valid credentials input by the user.

Value: Accessing privileges and determining if they correctly identify a user of the system so that the user may then access the application. Acts as a form of application security.

Non-functional:

- Authorization is always running when the user inputs credentials into the application.
- Authorization will successfully return either access or no access within 5 seconds.

4.15

Functional:

Requirement: Logout

- The current active session on the device will end within 5 seconds upon invocation
- The user will be navigated to the home view of the system upon successful completion
- System failures from this feature must not result in the system going offline

Scope: Any active authenticated user

Dependency: The user must be logged on to their account.

Value: Ensure that the user's account is not made available for use when not needed.

Non-functional:

- Account is logged off on the application within 5 seconds of requesting to log off.
- Logging off returns the user to the login page of the application.

Non-functional:

- Authentication is always running when the user inputs credentials into the application.
- Authentication will successfully return either access or no access within 5 seconds.

4.16

Functional:

Requirement: Account Creation

- System administrators cannot be created using Account Creation feature
- All user accounts must be stored in a persistent data store
- The user provides a valid email address that belongs to the user.
- The user provides a secret passphrase for requesting OTP
  - Secret passphrase must be a minimum of 8 characters
  - Valid characters will consist of the following:
    i. blank space
    ii. a-z
    iii. A-Z

iv. 0-9
v. .,@!

Scope: Any user attempting to use the system

Dependency: User having access to the application and not already having an existing account.

Value: Allows the user to use features offered by the application and save data specific to the user's needs.

Non-functional:

- Account generation takes no more than 5 seconds.
- Creating an account with the same username as an existing account will not be created.

4.17

Functional:

Requirement: Account Deletion

- Only a system administrator account can delete another system administrator account
- All personal identifiable information (PII) along with the user account data is permanently deleted from the system
- Account deletion is irreversible
- System failures from this feature must not result in the system going offline

Scope: Any registered user of the system

Dependency: Account must exist within the database in order to be deleted.

Value: When the user does not wish to keep their account, deleting the account ensures that all of the data is not made available for people to use and/or steal.

Non-functional:

- Deletion happens within 5 seconds of initiation.
- Deletion wipes all account data from the database.

4.18

Functional:

Requirement: User Management

- All operations are applied to a persistent data store

- Only system administrator have access to the User Management view - The system administrator will have access to view and modify all accounts and their associated user profile data within the system
- Single Operation
  - Create Account
  - Update Account
  - Delete Account
  - Disable Account
  - Enable Account
  - Operation should be completed within 5 seconds upon invocation - Bulk Operation
  - Multiple operations (e.g. all the same or mixed) within the same request • Maximum of 10K operations per request
  - Requests can be made through an uploaded file extract
  - File extract cannot be greater than 2GB in size
  - Operation should be completed within 60 seconds
- All single and bulk operations must be able to affect any user account/profile attribute within the system
- Only a system administrator account can create other system administrator accounts
- The system must have at least one system administrator account with total system access at all times
- System failures from this feature must not result in the system going offline

Scope: Any system administrator account

Dependency: User(s) must have the correct privileges to gain access to the features offered by the user management.

Value: Global access to the data store and being able to create, delete, enable, disable, and update user accounts.

Non-functional:

- User management always allows the user to create, delete, enable, disable, and update user accounts.

4.19

Functional:

Requirement: Usage Analysis Dashboard

- All data must be fetched from a persistent data store
- Key Performance Indicators (KPIs)
  - The top 5 most visited view of all time (bar chart)
  - The top 5 average duration per view of all time (bar chart)
  - The number of logins per day within the span of 3 months (trend chart) • The number of registrations per day within the span of 3 month (trend chart)
  - Two application specific feature metric
- All KPI data must be automatically refreshed in intervals of 60 seconds
- The view must load within 15 seconds upon completion of navigation.
- System failures from this feature must not result in the system going offline

Scope: Any system administrator account of the system

Dependency: Data gathered from user interactions on the application.

Value: Visualizes data gathered by the system to understand user behavior.

Non-functional:

- Visualization loads within 5 seconds of initiation.
- Displays data relevant to what is being analyzed. Does not display confidential data such as account username and password.

4.20

Functional:

Requirement: Logging

- All log entries must be immutable
- All log entries must be saved to a persistent data store
- All log entries must contain a UTC timestamp, log level, user performing operation, a category and a description/message
- Valid Log Levels
  i. Info - for tracking flow of system
  ii. Debug - for tracking key information crucial to maintainers of the system
  iii. Warning - for tracking events that may lead to system failures
  iv. Error - for tracking system errors
- Valid Categories
  i. View
  ii. Business
  iii. Server
  iv. Data

v. Data Store

- The logging process must not block any user from performing any interaction with the system
- The logging process must complete within 5 seconds upon invocation - System failures from this feature must not result in the system going offline

Scope: All system-initiated and user-initiated events within the system

Dependency: Data access layer and data store.

Value: Records and keeps track of all changes and interactions done on the application.

Non-functional:

- Logging is always active as long as there is a user actively using the application.

4.21

Functional:

Requirement: Archiving

- Archival process must execute every 00:00:00AM (local time) on 1st of the month - Archival process must only offload log entries that are older than 30 days - Archival process must consolidate and compress entries being archived - Archival process must offload entries to another location
- Archival process must remove offloaded entries from the system after successful archival
- Archival process must complete within 60 seconds upon invocation - System failures from this feature must not result in the system going offline Use Cases:

Scope: All log entries within the system

Dependency: Logging

Value: Offloads log entries to preserve system resources.

Non-functional:

- Archiving is always happening to maximize system resources.