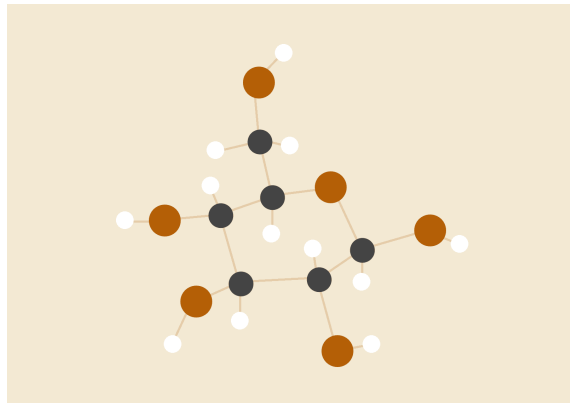


# Group 5 - Endurance

CEN 4010-Principles of Software Engineering:  
Project Proposal and High-Level Description



**Andy Cheng-** [wcheng3@fau.edu](mailto:wcheng3@fau.edu)

**Andrea Malvezzi-** [amalvezzi2019@fau.edu](mailto:amalvezzi2019@fau.edu)

**Georges Bourques-** [gbourque2015@fau.edu](mailto:gbourque2015@fau.edu)

**Noa Abiri-** [noaabiri@protonmail.com](mailto:noaabiri@protonmail.com)

**Ellie Moffett-** [amoffett2018@fau.edu](mailto:amoffett2018@fau.edu)

Revision:	Date:

## Executive Summary (GB)

For those who like to spend time outdoors and give back to the community, there aren't many options for combining charity, time with family and friends, and enjoying good weather. With beneCycle, users will have it all. Whenever a charity event is held, the application will bring people together and for some, give them the chance to push themselves with several routes that offer different levels of donations.

beneCycle is a web-based application that gives users the ability to participate in local cycling or pedestrian events. Cyclists also have the opportunity to help support charities that are within their community. They will have several options of set paths that they may choose from. Each path offers a challenge in itself and comes with a higher donation. The application offers an intuitive interface, and many features that make the outdoors more exciting.

The application's purpose is to give its users as many planning options as possible. With the tap of a finger, streets are highlighted and the route tailored for the individual's level of riding is selected. Share and comment on these routes with your family and friends. Compare distances, waypoints, times, and goals. Sync up routes for group runs or rides. Design your avatar so that you can exercise in teams or as a team. Everyone's avatars will be present on your map and you can even leave notes for others. Want to plan ahead of time? Create your route and set an event reminder. Events can be coordinated with friends through the web based app. We can achieve milestones together, and support each other.

The idea for this application came about when one of our team members described a charity event where people would cycle throughout the city on different paths depending on an entry fee. We sought to give people the ability to not only come together and make their own events, synchronizing their routes with others, but also to provide individuals with a way to push themselves to go the distance.

## Competitive Analysis (GB)

	beneCycle	Google Maps	Apple Maps	Waze
Time Estimations	4.2	4.5	3.8	4.5
Traffic Markers	4.5	4	3.5	4.5
Sharing	5	3	3	2
Integration	4	3	3	3
Customization	5	3	3	4
Saving Options	5	3	3	2
Mean:	4.6	3.4	3.2	3.3

beneCycle provides a progress bar that calculates the percentage of the trip that has been completed, the percentage of the trip that is left, the time elapsed, and the remaining time. The other applications may not always provide the most accurate time estimations as the rate at which a person is traveling is not taken into account.

Google Maps - <https://www.google.com/maps>

Google Maps has a limited amount of knowledge about the current traffic. They might not include traffic lights, road work, and are not suited for cyclists and pedestrians. The main audience are those who are not in cars, and therefore need to know how to navigate their area likewise.

Waze - <https://www.waze.com/>

Waze lacks a depth of options that enable users to fully personalize their experience. They need to be able to save various routes, add waypoints, track and compare their target destination to previous destinations, and more. Predetermined routes are limited at best, and the user should be able to edit their trip on a whim.

Apple maps - <https://www.apple.com/maps/>

There isn't any way to collaborate with other people to create events where groups can walk or ride together along the same path. The app needs features that allow groups to compare their progress. Also integrating social media into the application would keep them connected.

Advantages: beneCycle focuses on the social aspect as well as the personal aspect of travel. It is meant for the trailblazer who wishes to test their speed and push the boundaries of where they can go. beneCycle gives the user the flexibility to decide where to go and how to get there, while offering avatars, route sharing, commenting, and much more to keep people connected.

## Data Definition (AM)

Name	Meaning	Usage	Comment
Rider	actor	Use Case Scenarios	This person logs to become a member and select a path.
Merchant	actor	Use Case Scenarios	This person logs in to list location & product/services
Charity Worker	actor	Use Case Scenarios	This person logs in to view all users, contributions & communications
Municipal Worker	actor	Use Case Scenarios	This person logs in to communicate with charity workers
Member	actor	Use Case Scenarios	A user who is registered with the system
Non-Member	actor	Use Case Scenarios	A user not registered with the system
Path	data	Navigation	Delineates chosen path that the Rider/s must navigate. Perhaps in the future audio/visual can navigate the Rider through each of the paths.
Attraction	data	Activity Type	Charity-approved merchant product/service
Account-Rider	data	Use Case Scenarios	Store rider info
Account-Merchant	data	Use Case Scenarios	Store merchant info

Calculator	data	Use Case Scenarios	Keeps running total of contributions & total number of volunteers and riders.
------------	------	--------------------	---

Name	Meaning	Usage	Comment
Payment Page	User Interface	User Interface	Allows a user to buy a ticket or donate via link to an external site like Square.
System	Platform & Hardware Services	Use Case Scenarios	The mySQL database, code, front-end design and back-end supporting services
Benecycle	Domain Name	Use Case Scenarios	A unique and functional name that can represent the website and all pages.
Web address TBA	Production Server	Use Case Scenarios	Data Server
Donation Page	User Interface	User Interface	Allows a user to navigate directly to the payment page without becoming a member.
Filter	Service	Site User Service	Filters attractions by date

## Overview, Scenario and Use Cases (AM)

### Non-Member Expectation

#### 1.) Create account

1.1 The system shall allow the user to create an account by entering first name, last name, phone number, and email address. Page will specify that the transaction must be complete to allow registration. A button marked “Complete Transaction” will be provided. Clicking here will redirect to the Payment page, where an ID will be generated by an external site and returned to the system for storage with the other data. A welcome email will also be sent to the new user upon completed registration. The system will not allow a user to progress to the Payment page if preliminary data are not filled in. The system will not create a new user if a user id is not returned upon successful completion of payment by external site.

#### 1.2 Stimulus/Response Sequence

- i.) User enters first and last name
- ii.) User enters phone number
- iii.) User enters email address
- iv.) User clicks on the “Complete Transaction” button.
- v.) System redirects the user to the payment page.
- vi.) System stores user ID returned upon transaction completion with the other data fields, and the User is added to Members. Path selection is conveyed by user ID, and sent to the Path page.
- vii.) System will confirm this to the user, and provide a back button to navigate to the main page.

#### 1.3 Function Requirement Label

- i.) REQ 1.1 Creating Account

## 2.) View Information Page, Paths and Attractions Page and Contribution Page

2.1 The registered user may browse to these pages from the main page, and from these pages back to the main page or to the membership or donation page. Users may donate or view information, contributions, attractions and paths without becoming a member.

### 2.2 Stimulus/Response Sequence

i.) User scrolls to the desired pages.

ii.) User clicks on a button to donate, become a member, view paths, or information about the charity.

## 3.) Search Page

3.1 The user may search for participating merchants by clicking on a date. The system will return a filtered list.

### 3.2 Stimulus/Response

i.) User scrolls to the search page.

ii.) User clicks on a date.

iii.) System returns a filtered list of participating merchants.

### 3.3 Function Requirement Label

i.) REQ 2.1 Browse by search

## 4.) Donate

4.1 The user may browse to this page from the main page, and be directed to the Payment page without entering additional info. A 'Donate Now' button is available for this purpose.

### 4.2 Stimulus/Response

i.) User navigates to the Donation page.

ii.) User presses 'Donate Now' button

iii.) User is redirected to the Payment page.



iv.) System will confirm this to the user, and provide a back button to navigate to the main page.

## Member Expectations

### 5.) Post Feedback and Media

5.1 The user may post comments, videos or photos visible to other members. The system will not allow non-members to navigate to the Media/Feedback page, or to post feedback/media of any sort.

#### 5.2 Stimulus/Response

i.) The user may navigate to the Media/Feedback Page and enter a comment using a comment box or select 'upload other media.'

ii.) The system will store comments, photos or videos.

iii.) The system will display stored media at the bottom of the screen, sorted by newest to oldest.

iv.) The page will provide a back button to navigate back to the main page.

#### 5.3 Function Requirement Label

i.) REQ 5.1 Member Post

### 6.) Help

6.1 A member may post to Help screen to indicate a need for assistance. The message will transmit with their location on the path. Municipal workers as well as Charity workers may access the message.

#### 6.2 Stimulus/Response

i.) The member navigates to help screen and presses a button labeled "Help" to ask for assistance.

ii.) The system transmits the signal along with the member location to municipal and charity workers (perhaps by SMS, that is TBD.)

ii.) Once the signal is sent, the system confirms with the member.

iii.) The page will provide a back button to navigate back to the main page.

### 6.3 Function Requirement Label

i.) REQ 6.1 Help

## 7.) Map

7.1 A member may navigate to the Map page from the main page, and view their location on their path as well as starred nearby attractions.

### 7.2 Stimulus/Response

i.) The member navigates to the Map page.

ii.) The system presents a real-time view of their location on a map, with nearby attractions marked.

iii.) The page will provide a back button to navigate back to the main page.

### 7.3 Function Requirement Label

iv.) REQ 7.1 Map

## Initial List of High-Level Functional Requirements (NA)

- 1) Will generate route for user after user input of route type
  - a) Route shading
- 2) Utilize google map and traffic for cost function
- 3) The system will operate in a specific geographical location, and will not operate outside of it
- 4) Will create and store user profile and route history
- 5) Social Aspect:
  - a) Take user input to display social-media type posts with text and photo capabilities

- b) Route share to contacts
  - c) Group Runs/Events with reminders
    - i) Possible integration with calendar app
  - d) Rankings daily/weekly
  - e) Popular routes in user's area
- 6) Route user to an external site to process donations
- 7) Waypoints marked on map
- 8) Other functionalities:
  - a) Timer/Stopwatch
  - b) Distance logger
  - c) Calorie Estimator
  - d) Route history with statistics (mph) and record times/distances
  - e) Waypoint notifications

## List of Non-Functional Requirements (NA)

- 1) Product Requirements:
  - a) Speed - Refresh and processing times should be less than 3 seconds.
  - b) Size - Less than 10 MB
  - c) Ease of Use - Network pretrained, intuitive controls
  - d) Reliability - Mean time to failure twice the amount of time it takes to complete the longest route (probably 2 hours)
  - e) Robustness: Failure rate should be less than 1%
  - f) Portability: Functional for IOS and android OS
- 2) External Requirements/Organizational Requirements
  - a) Android compatible
  - b) IOS compatible, follows ios design themes and guidelines
  - c) Compatible with Square for payment processing
  - d) Access to system contacts and location, potentially calendar app

## High-Level System Architecture (EM)

1. **Lamp Server:** we will be using a Lamp Server to host our project after its completion.
2. **WhatsApp Messenger:** WhatsApp is what our group will be using to communicate with each other conveniently and share progress. We can use this to notify and update each other on plans and keep up with schedules.
3. **mySQL Database:** we will be implementing a mySQL database to keep track of the information required for this project. Organizations will be able to post events and users will be able to keep track of login information, donation and participation history, and social media interaction. All relevant information will be stored in the database for retrieval and use by the respective parties.
4. **GitHub:** this will be used to upload and share code with the other members of the group and keep track of progress made in a transparent format that each of us can view and edit without interfering with one another.
5. **Brackets:** for the development of the web pages associated with this web app, Brackets will be used as a convenient text editor allowing easy navigation of folders, useful autocomplete features, and, most importantly, a live preview of each stage of the web pages as they are being developed.
6. **Apache Cordova:** this is a mobile application framework that will allow us to keep our app compatible with the Android, iOS, and desktop platforms. It enables us to build hybrid web applications using CSS, HTML, and JavaScript
7. **Jira:** for the planning and organization of this project's development schedule, we will be using an agile development methodology, which will be kept track of using Jira. This platform features planning boards and objective lists and is a useful tool for coordinating the future steps of our team planning.
8. **Google Maps API:** we will be using this API to embed Maps capabilities into our application. Since location routing and geotagging are cornerstones of our requirements, this API will be necessary to implement to meet our goals. ( <https://developers.google.com/maps> )

## Team

1. **Andrea Malvezzi** - Product Owner
2. **Andy Cheng** - Scrum Master
3. **Georges Bourque**- Back end Lead
4. **Noa Abiri**- Front end Lead
5. **Ellie Moffatt**- Software Engineer

## Checklist (AC- To include Github push of document)

- a) Team decided on basic means of communications- **Done**
- b) Team found a time slot to meet outside of the class- **On Track**
- c) Front and back end team leads chosen- **Done**
- d) Github master chosen- **Done**
- e) Team ready and able to use the chosen back and front-end frameworks- **On Track**
- f) Skills of each team member defined and known to all- **Done**
- g) Team lead ensured that all team members read the final M1 and agree/understand it before submission- **Done**