Beheashta Atchekzai, Aidan Chu, Ronny Yehia, Khambrel Austin, Ryan Kim

SRS Planning

Functionality

- The functionality of the full application (beyond the scope of the implementation phase) would enable users to track all relevant information regarding the campus, and the events taking place on campus. The application would be developed in a way such that use of the app even without wifi would be almost as useful as its use cases with internet connection. An interactive map of campus would be stored locally, as well as a list of events and their description, location, and time. In periods of available internet connection the map and event list would be checked for necessary changes to represent the most accurate events and image of campus as possible. In addition to being used to track active campus events and navigating campus using the map, the app would also serve to replace a physical ID card that identifies students. Being an opt in functionality, identification integrity would be ensured by the server, not allowing students to falsely represent their identity.
- The design prototype will be able to:
  - Simulate network connection with a serve side
  - Simulate map updates through database interaction
  - Allow the user to view their copy of the map.
  - Allow the user to view their copy of the event

External Interfaces

How does the software interact with people? (Final product, purely in theory, not being developed for implementation phase of project)
- Users may interact with the application via a **mobile app** or using a **web browser**.
- When using the mobile version users can interact with the following features **offline**:
  - Map of campus.
  - Important dates that were stored at the time of last being online.
- When using the mobile version **with internet access**:
  - Up to date information regarding campus events.
  - Up to date map of campus (including area closures, and navigation?)
  - When the application is launched with an internet connection all locally stored information is compared against the server-side information to check if an update is necessary.
- When using the application in a web browser
  - The user can use the full functionality of the mobile application.

How does the software interact with people? (Prototype, application being developed for implementation phase)
- Prototype will be developed as a desktop application.
- Within the application testing parameters may be entered.
    - Toggle internet connectivity
    - Enter upcoming alterations to map.
    - Enter alterations to the event calendar.
- MVP user functionality
    - View map (offline/online)
    - View a simple calendar or list of upcoming events.
    - Display virtual student ID card
- MVP backend functionality
    - Check for updates required in:
        - Map
        - Events
    - Locally store map data
    - Locally store event list
    - Locally store student ID card

Performance

- Application performance (Client)
    - Considering the power of modern cell phones little limitation exists in the performance of an application meant primarily for viewing data stored locally.
    - Trackable metrics:
        - Startup time of the application (Seconds)
        - Time to display map (from device storage to display) (Seconds)
        - Average time to update data on user device (Seconds)
        - Size of storage on device consumed by application (bytes)
- Application performance (Server)
    - The server application layer between the network requests of a user and the database would also be relatively lightweight.
    - The chief purpose would be to compare a user's current data, to the data that is present in the database and provide the user with up to date information, or inform it that the data is accurate.
        - This can be done with a similar system to network caching.
        - Each package of data the user has stored would be dated with when the server provided this information.
        - If the server finds that there was a new piece of data that the user has yet to receive, or that the information that user has is out of date, the appropriate changes can be made.
    - Trackable metrics:
        - Average time to service a client (Seconds)
        - Average time to develop response to client, without data (Seconds)

- Average time to develop response to client, with data (Seconds)
- Network performance
  - While using the application physical location will likely be close to the host server (Waterloo campus), this reduces network round trip time.
  - Bandwidth requirements would under normal circumstances be very small. Except for irregular instances where large portions of the map are being updated, the network would only need to communicate a relatively small amount of data, kilobytes at most. Events would be represented in string format, and metadata would be a few integers at most.
  - Trackable metrics:
    - Average round trip time per user (Seconds)
    - Average bandwidth consumption per user (bytes)
    - Average bandwidth consumption per user, with an update (bytes)
    - Average bandwidth consumption per user, without update (bytes)
    - Average size of packet
- Database performance
  - Database management would also be simple. The database would only see moments of increased flow when a new piece of information becomes available and thus must be pushed to many users. In addition to many small updates burdening the database (new event for calendar), depending on how updates for the map are handled retrieving large sections of stored data could also cause issues where a large number of users are left unserviced for large periods of time.
  - Trackable metrics:
    - Average size of data per user, with map update, without event update. (bytes)
    - Average size of data per user, without map update, with event update. (bytes)
    - Average number of requests to database per hour (units/hour)
    - Average size of data package requested (bytes)
    - Maximum and minimum data output of database (bytes)

Attributes

Prototype Implementation Language
- GUI → python
- Database → simulated in python
- Network layer → simulated in python

Operating environments:
- In situ (offline):
  - User looking for a way to class without internet can find the location on the map stored by the application
- In situ (online):

- Users looking for a way to class with the internet can be navigated by the application to their desired room.
    - At home (online):
        - Users can see upcoming dates.
        - Users can see that their usual route to class has been blocked by renovations.
        - Users can see that class is canceled due to inclement weather.
    - At home (offline):
        - User can look at the last updated (since last period of connected internet access) information for events happening on campus
    - In the web browser (online)
        - Users can utilize the full functionality of the application.

Database policy
- Data that will be stored on the server-side:
    - Users are managed without identification, therefore users will be identified by a unique ID related to the device accessing the app.
- Data that will be stored on client device:
    - Maps of campus
    - Last known information for events on campus
    - Important information regarding the user (not synced with the server, i.e: not stored externally)
        - A virtual student ID card.

The SRS writer(s) should avoid placing either design or project requirements in the SRS

1. The student is unable to find their classroom (UC-01) - Kham

Use Case ID	UC - 1
Use Case Name	Room Locator
Actors	Customer (Faculty, Students, Future Students)
Use Case Overview	Allow users to find and room Within the Laurier campus without having to go around and ask people for directions

Trigger	User is unable to find a certain space or classroom on Laurier Campus
Preconditions	There is an active and functioning connection to the Internet. A smartphone is easily accessible. Application or website is downloaded
Postconditions	User has successfully located a route to the space or classroom they are looking for?
Use Case Associations	UC – 2,3,4,5,6
Inputs	Here are the basic requirements for the program to run efficiently Users name Users Current Location User Requested Destination
Outputs	A route to the current destination based on the options they have selected

Normal Flow	1) This begins by having the user input their details related to their school account ex. name Student number
	2) Once logged in User enters their current location
	3) Then followed by their destination for example a specific classroom or the nearest gym
	4) The application will create the shortest and safest route for the user to take based on their selected options
	5) User follows the guided route
	6) Destination has been reached
	  .
	  .

Alternative Flow	In the Event Step 2 fails, Unable to find current location ● User will be given an option to USe GPS services ● User can also enter nearby buildings that they see to allow the program to grasp their general location if they are unable to pinpoint it exactly · Continue following remaining steps
Priority	High.

2. Student needs to locate schooling resources (wellness center, financial aid) (UC-02)

Use case ID	UC-02
Use case name	Locating services
Actors	Student (primary), Service provider (secondary)
Triggers	The student is looking to use a service provided by the school like the wellness center or financial aid
Preconditions	- The user has the app downloaded and requires assistance from a resource - the person booking an appointment must be a current student - Access to the internet and a cellular device - In order to book an appointment, there must be time slots available

Postconditions	- the student has contacted the resources they are looking for and booked an appointment at an appropriate time - the student receives whatever assistance from the resource they required
Use case associations	UC-06
Inputs	- Student ID - Input the service you are looking for in the search bar on the application - Book your available time slot
Outputs	- An appointment is booked and the service will be completed on the associated date

Normal flow	1. This begins when the student requires assistance from a school resource
	2. The student then opens the application on their device, using the search bar on the top of the home screen the student is able to search for the service they require
	3. When the service is found on the device, they are able to see where the room is located on campus and what the quickest route to it is
	4. The student then calls or emails the person providing the service and is able to discuss with them
	5. If the student finds that the service is adequate for them, they provide their student number and book a time slot that is available
	6. On the day of their appointment, the student would go to the location and provide their student ID to ensure they are an actual student at the school
	7. The service is then provided to the student, who can then book another appointment if deemed necessary
	8. The use case ends successfully

Alternative flow	Alt 1: Not a student - The user is able to use the application and view the services but due to them not being a registered student at the university they are unable to book an appointment Alt 2: Student misses their appointment - if the student manages to complete all the previous steps but does not show up to their appointment, they will face a penalty that does not let them book for another 2 weeks - After 2 missed appointments, they must contact the school directly to appeal and book an appointment - An appointment must be cancelled 6 hours ahead of time to not be considered a no show Alt 3: no available time slots - if the student has confirmed that they require assistance from said service, but there are no time slots available that work with their schedule, they must inform the person providing the service about this and will be contacted when a time slot is an available
Priority	Medium-high

3. Students in need of jobs are able to apply for available positions (UC-03)

Use case ID	UC-03
Use case name	Job openings
Actors	Student (primary), hiring manager (secondary)

Use case overview	A student is looking for a job during the school year
Trigger	The student requires a job for a source of income to cover bills, tuition or just wants a source of income
Preconditions	- The student is of legal age to work and has correct documentation to work - Meets the health requirements of the job (ex. Fully vaccinated) - Positions at the workplace are available
Postconditions	- the job position has been filled - the student now has a source of income - if the student required money to pay bills, they are now able to do so
Use case associations	N/A
Inputs	- resume/cover letter - student ID
Outputs	- if the student meets the job requirements, they are given an interview - if the interview is successful a job is offered

Normal flow	1. The student opens up the application
	2. The student then searches on the home page for "Job listings"
	3. The student then searches for a listing that suits them
	4. They are then able to apply for the position and provide the proper documentation (resume, cover letter)
	5. They must also be a registered student at the university
	6. After the application is filled out, the hiring manager goes through application
	7. If the student is suitable for the position, they will offer an interview
	8. The manager and student must agree on a time for the interview
	9. If the interview is a success for the student the manager may offer a position
	10. If the student is willing, they can accept the offer from the manager
	11. The student is now hired, the use case ends successfully

Alternative flow	Alt 1: The interview is not offered
	- the student may apply for the position with all the required documents but may not meet the manager's needs so they are not offered an interview, they must now apply for other jobs
	Alt 2: The interview was poorly done
	- the student successfully got the interview but did not have a good interaction with the manager, this also sets the student back to the beginning stages
	- they must now apply for a different job
	Alt 3: No proper legal documentation
	- if the student does not have the correct legal documentation, they are unable to apply to any positions as they are not permitted to work in Canada
Priority	Medium

4. Planning your route to get home safely at night - gets you to local transit shelters safely (UC-04)

Use Case ID	UC-04
Use Case Name	SafeRoute Planner
Actors	Student (Primary)
Use Case Overview	Often times classes can run late and navigating a safe route home from campus can be hard to manage. This use case allows students leaving school buildings to plan a safe route to a local transit shelter by calculating a route that keeps them on a public walkway or in close proximity to emergency poles. Outputted route will highlight all emergency poles in close proximity for the user as well as highlighting the ideal route.

Trigger	User selects button to request help planning route
Preconditions	There is an active internet connection on the device, the device has the app installed and has location services enabled.
Postconditions	User has successfully planned route home
Use Case Associations	UC – 8
Inputs	User's current location on campus (GPS), Users starting destination (building they are leaving), User's final destination / which transit they prefer (GRT, GO Bus, GO Train)
Outputs	Description of a safe route to destination
Normal Flow	Users will select a button that begins the process of route planning. User will be prompted to enter the location / building that they are currently in from a drop down menu of registered school buildings. User will be prompted to enter the transit option they would like (three possible options are GRT, GO Bus and GO Train) System will calculate the best possible route from location to selected destination using campus pathways so that the user is always on a visible pathway or in close proximity to the emergency poles on campus (which will be highlighted on the route). System will allow user to select start route; user will be able to watch their own progression on the route in Realtime on the screen. While the route is in progress, the system will provide quick access to emergency services in the form of bold, easily clickable buttons in case of an emergency. Once user has reached the destination the system shows a prompt saying thank you for using the service and follows with a prompt to close the process.
Alternative Flow	N/A

Exceptional Flow	Users will select a button that begins the process of route planning. User will be prompted to enter the location / building that they are currently in. If the User inputs an incorrect location/building option. System will prompt them to select the building they are closest to from the drop down menu and use that location to map out the route. User will be prompted to enter the transit option they would like (three possible options are GRT, GO Bus and GO Train) System will calculate the best possible route from location to selected destination using campus pathways so that the user is always on a visible pathway or in close proximity to the emergency poles on campus (which will be highlighted on the route). System will allow user to select start route; user will be able to watch their own progression on the route in Realtime on the screen. While the route is in progress, the system will provide quick access to emergency services in the form of bold, easily clickable buttons in case of an emergency. Once user has reached the destination the system shows a prompt saying thank you for using the service and follows with a prompt to close the process.
Priority	Medium.

5. Booking appointments with academic advising (UC-05)

Use Case ID	UC-05
Use Case Name	Academic Standing Hub
Actors	Student (primary), Academic Advisor (secondary)
Use Case Overview	Understanding program requirements and navigating booking a session with an academic advisor is hard due the school's disorganization. This use case allows a student to navigate which faculty they need to contact, helps answer basic questions

	and provides links to booking and contacting individual academic advisors
Trigger	Student requires help navigating academic advising, they select the academic standing hub button that begins the process.
Preconditions	User has a stable internet connection; user has a general idea of what services they may potentially require; user has their current GPA or can use the calculator to enter.
Postconditions	User has successfully contacted the service/academic advisor they need or have received enough information/clarification on their query that they do not need an appointment.
Use Case Associations	
Inputs	User's respective faculty and program, User's current GPA OR User's classes, weighting and grade in each class to calculate the GPA
Outputs	User's current academic standing and GPA, respective program/faculty's basic academic requirements AND OR contact information of respective academic advisor
Normal Flow	User is prompted to select their respective program and faculty from a dropdown menu. User is prompted to either enter their current GPA System displays the respective programs GPA requirements, and based on the users entered GPA, the system displays whether they are in good or bad standing (if they meet their respective GPA requirements) If a user is in bad standing, they are prompted to contact the respective academic advisor System displays the name and contact information of respective academic advisor (regardless of result) User is prompted to close process

Alternative Flow	User is prompted to select their respective program and faculty from a dropdown menu.
	User does not know their GPA, they select the "Calculate My GPA" option.
	System prompts user to enter their courses, course weighting and final grade in class.
	System calculates and displays current GPA based on entry by user.
	System displays the respective programs GPA requirements, and based on the user's GPA
	The system calculates displays whether they are in good or bad standing (if they meet their respective GPA requirements)
	If a user is in bad standing, they are prompted to contact the respective academic advisor
	System displays the name and contact information of respective academic advisor (regardless of result)
	User is prompted to close process
Exceptional Flow	User is prompted to select their respective program and faculty from a dropdown menu. User is prompted to enter their GPA or use the calculator.
	If User is unable to select or enter any of the above information, they are prompted with instructions on how to check the status of such information from LORIS.
Priority	Medium

6. Discussion board with other students for FAQ and discussing with higher year students about their program (UC-06)

Use Case ID	UC-06
Use Case Name	FAQ/ Discussion board about current Laurier topics or question
Actors	Customer (Faculty, Students, Future Students)

Use Case Overview	Allow the user to have questions answered by their fellow golden hawks. Can post the question to a board or search to see if their questions had already been answered
Trigger	User has questions about anything related to the Laurier lifestyle
Preconditions	Need an account created with Laurier Need an active internet connection Access to the website/ application
Postconditions	Customer found the answer to the question Customer Posted their question on the board and are waiting for a response
Use Case Associations	UC – 2,3,4,5,6
Inputs	Here is a list of required information that the application will need to run ● Need input from user ● Name ● Student ID ● Program ● Year of study
Outputs	Answers to their questions
Normal Flow	1) The user will first be prompted to enter their information into the application and or website. This information should be the same as the one on your LORIS account

	2) USer will then be prompted to either enter a question that can be posted on the discussion board or that can be searched with an already created answer. 3) Based on their selection the user will have to wait for a response to the question or exit the app after their question has been answered. .
Alternative Flow	Search Step 2 ● If the user selects to search for the answer for the question in the FAQ they will have to enter it in the search bar ● A list of specific and related answers will appear in the FAQ section where they can click on each and find their answer or return to the main menu and post a question to the discussion board Question Step 2 ● · When entering a question the to post the user will have an option to enter the question followed by a description to provide the readers with background knowledge in relation to their question ● The user can choose what section to post their question in and if they would like it to be viewed by all Laurier students or just those in a specific area/ program or prestige ● Once posted they must wait for a response. ● A link will be provided to quickly re-access the question and view all posted responses ● Then have the option to return to the main menu and search for a new question or enter a new question
Priority	LOW

7. Personality forum to help you interact and meet with fellow Laurier students (UC-07)

Use Case ID	UC-07
Use Case Name	Discussion board for extra-curricular topics.
Actors	Students or future students (primary)
Use Case Overview	User can post questions about the city, conversation starters, introductions, share images and links. Students can then reply to these topics.
Trigger	User is looking to create or view discussion.
Preconditions	User has an account created with Laurier User has an active internet connection User has access to the website/ application
Postconditions	User views or posts a topic of discussion.
Use Case Associations	UC – 6
Inputs	Here is a list of required information that the application will need to run ● Need input from user ● Name ● Student ID ● Program ● Year of study
Outputs	● Answers to their questions

Normal Flow	1. The user will first be prompted to enter their information into the application and or website. This information should be the same as the one on your LORIS account 2. User will then be prompted to create a post on the discussion board or continue viewing posts. 3. Based on their selection the user will have to wait for a response to the question or exit the app after their question has been answered.
Alternative Flow	If the user chooses to view and reply to other posts instead of creating their own, they can scroll down and view recently created posts.
Priority	Low

8. Interactive map that displays hours of operation of each building and some services found there (UC-07)

Use Case ID	UC-08
Use Case Name	Hawk Map
Actors	Student (primary), Building/Service (secondary)
Use Case Overview	User can view a map of campus and surrounding buildings with respect to current location. Also displays information of each building such as services within and hours of operation.
Trigger	User is looking for location of a certain service on campus.
Preconditions	User has a stable internet connection; user has a general idea of what services they may potentially require; user has their current GPA or can use the calculator to enter.

Postconditions	User has found the building, service, or information needed.
Use Case Associations	UC – 1, 2, 3, 4
Inputs	User's current location on campus (GPS)
Outputs	- Location of building or service - Descriptions of buildings or service - Hours of operation
Normal Flow	1. User opens the map 2. User searches building or service on campus 3. System displays location of requested service and current user location on map. 4. User can view what services are offered in a building and its location.
Alternative Flow	In the Event Step 3 fails and current location is unavailable, request access for device gps. Display the map of campus and the full information with user location marker.
Exceptional Flow	User can choose to use the map without gps services. The system will still display the map of campus and the full information.
Priority	High