The Project

As a team we chose option 1 - Messaging Application for Universities. We came to this decision after a general discussion on what everyone in the team is comfortable working with. We decided to stay away from Android based projects as not many of us have experience with the platform. We wanted something more useful and polished than a console application, so we decided to work towards a web messaging application as we see the opportunity to implement many great ideas that we discussed together. We decided to stay away from time management application as it seemed that there are less moving pieces and, therefore, it will be less challenging to implement such application. Our application is a messaging platform that allows users to create, edit and view posted topics. One of the unique features of this application is the advanced notification system that allows users to interact with the topics as quickly as possible.

The Personas

Ivan the instructor 33 from Toronto. Ivan is a lecturer at the University of Toronto. He has been teaching for five years and begun teaching shortly after receiving his degree. He does not own his own car and takes public transportation. Outside of lecturing Ivan enjoys biking and hiking. Ivan single and lives in a condo near campus.

Freda the freshman 19 from Barrie. Freda is a first year at University of Toronto, who is studying electrical engineering. She is living away from home for the first time and is living on residence. She spends most of the week studying in the library on campus and works weekends at the local grocery store. She eats most of her meals on campus and spends her time off playing a level 85 human paladin in World of Warcraft.

User Stories

Priority Scale:
1 - We must have this feature, without it our product has no value to our users.
2 - Valuable feature, but we can still release a product without it.
3 - Might be valuable, but we should look at it later.

Difficulty Scale: 1 - Easy, 2 - Medium, 3 - Hard

User Story 1

Priority: 1
Difficulty: 3
A student Eric, living on campus, wants to sell her/his MAT102 textbook. Eric creates an account posting with a topic called "Selling Book for MAT102". Eric waits for other students to subscribe to his posting. Once the item gets sold Eric is able to mark the topic as closed. The posting gets archived.

User Story 2
Priority: 3
Difficulty: 3
A CS student Bob subscribes to all topics that instructor Gerhard posts on the discussion board, because Gerhard likes to host assignment help sessions occasionally. When the user subscribes to another user they get to choose the types of notifications they will get. When Gerhard posts a new topic called "CSC201 - Assignment 1 help", Bob gets a notification in his email and on his phone.

User Story 3
Priority: 3
Difficulty: 3
A user who has not been logged into the system for more than one day gets automatic SMS notification on new posts in his topic. The SMS will contain the number of new posts added along with its headings.

User Story 4
Priority: 1
Difficulty: 1
An anonymous user can view posts and search through topics. Such user will not be able to make own postings or start new topics

User Story 5
Priority: 2
Difficulty: 2
As Fred, I would like to connect to other students or TA's within my courses. I would like to create group topics (meeting sessions) in which posts are updated in real time for projects or assignments.

User Story 6
Priority: 3
Difficulty: 3
Student Bob completed his course in CSC201, but he still gets the text message when Gerhard posts CSC201 help session. He replies to this number saying "Unsubscribe", he is now unsubscribed from Gerhard and no longer receives notifications on "CSC201 - Assignment 1 help" topic.

MVP - Minimum Viable Product

List of Features:
- Each user may login and post with their own email account.
- User will get their password emailed to them when they cannot remember.
- Can subscribe to many topics, and will get notification when a new reply is added, or when a post is finished or changed.
- Logged in users can post new topics; topics are displayed in descending chronological order so new topics are always at the top.
- Owner of topic may edit at any time.
- Owner can close a topic once it is done. After topic closed it is archived, and still able to be read.
- Anyone logged in may reply to open posts.
- Searching through all post and list all related topics.
- Registration system requires email, phone number and password.
- User will receive a short text message when a reply is posted.
- User can text "unsubscribe when no longer want to be subbed to topic.
- Subscription only view and all view.

Technical Challenge:
A major highlight of our message board is it sends SMS notifications, and user may reply to the SMS and perform actions such as unsubscribe. We use Twilio (using the provided Java SDK) to implement this.

Unique User Experiences:
Student Bob likes to do his assignment at library, but also wants to get immediate notifications when the instructor he subscribe to posts assignment help. Then Gerhard (Bob's instructor) adds a new post, Bob gets a SMS and knows a help session is available without being near a computer at all!

Student Bob completed the course, but did not unsubscribe from Gerhard. Gerhard posted a new help session for new students in his class, Bob is not near a computer but is able to unsubscribe from Gerhard by replying "Unsubscribe".

Release and Iteration Planning

We are planning to have primary functionality for user creation and posting topics using a basic user interface during the first iteration. This involves setting up a web based solution that allows users to interact with the interface. The system should be able to authenticate new and old users, and accept phone numbers.

The user stories that we plan to implement in Phase 2 are User stories 1 and 4. User stories 1 and 4 describes the average registered user and anonymous user, and their interactions. The minimum features should be login, post, and read posts, and that's User story 1 and 4. Other User stories are an extension of 1 and 4, and we cannot complete the rest without completing 1 and 4.

We are excluding user story 2, 3, 5, and 6, because they all rely on basic features that will be implemented once user story 1 and 4 are completed.

During the first week of Phase 2, user story 4 will be finished. Since it uses the least amount of features; Topic, Topic Manager and Reponse. User story 4 will also be the easiest implementation to test.

CRC cards

User	
- know it's name - know it's email - know it's phone number - has password - know topics posted	Topic Subscribers

Login	
- able to authenticates - can register users - can recovers passwords by sending an email - perform phone number verification?	User

Topic Manager	
- can get a list of subscribed topics - can query the topic list - can perform search - can opens/closes topic - archives topics	Topic Response

Topic	
- know who posted it - know date time it's created - know title message - know list of replies attached	User Response

Response	
- has body (message) - metadata (date, time etc) - know who posted the response	Topic User

Notification	
- has type (email or SMS) - has body (message) - has destination - has source	Response Topic User

Notification Manager	
- can send emails - can send SMS to phone numbers - can interact with the Twilio SDK - can read text reply - notify subscribed users to a new/ updated post	Notification Subscribers User

Subscribers	
- Know which user is subbed to which topic - Can unsubscribe users	User Topic Response

User story 1
As a user, I want to create a topic and be able to close it.
Create a Topic using Topic Manager.
From Topic Manager, we can close the Topic.

User story 3
As a user, I want to receive SMS messages for threads I am subscribed to.
Notification Manager sends SMS's.

User story 4
As an anonymous user, I want to be able to read a post.
From Topic, we can get Response.
From Topic Manager, we can search through Topics.