



SUPER HEROES IN TRAINING

My Academic Pyramid

Business Requirements Document

CECS 491A Sec 05

November 1, 2018

Team Leader: Krystal Leon, 013986607

Arturo Peña Contreras, 010914811

Luis Julian, 007472593

Hyunwoo Kim, 014392909

Victor Kim, 012016990

Trong Nguyen, 016208983

Table of Contents

Abstract	2
Scope	2
Project Scope	2
In Scope Functionality	3
Out of Scope Functionality	4
Constraints	4
Project Function and Non-functional Requirements	4
Business Requirements (Core Components)	5
Security	5
Login	6
Password reset	9
Log out	11
User Access Control	13
User Management	15
Registration	18
Error handling	20
Logging / Archiving	22
Usage Analysis Dashboard	23
Data Store Access	25
Network Communication	27
Documentation	28
Business Requirements (Application Features)	29
Discussion Forum	29
Search	32
Tutoring	34
Ranking System	40
Text Messenger	40
User Profile	42
Notification System	44
Payment	45
Glossary	46

Abstract

My Academic Pyramid is a social media web application where students can communicate with their peers and ask questions on a discussion board. The application will also provide an online tutoring service, allowing students to converse with tutors and seek assistance online without the need to travel to campus. The goal for our project is to help students build relationships among each other and include features which can assist students with their assignments.

Scope

1. Project Scope

- a. Initial System
 - i. Will be a Single Page Application web application.
- b. Initial Browser Support
 - i. Will support the latest two version of Chrome.
- c. Initial Data
 - i. Location of the user
 - The application will support storing location data of cities and universities within the state of California.
 - ii. Personal information of the user
 - Age
 - The application will support birthdays ranging from January 1, 1970 up to the present day. However, users will only be allowed to register with birthdays resulting in ages over or equal to 18 years.
 - Name
 - The user will no provide initial data for a user's name. The user will be able to freely type text in the name fields. Each first name, middle name, and last name must be at least 1 character and at most 35 characters.
 - Major
 - The application will provide majors compiled from three universities in California. This will allow for a diverse range of majors for users to choose from.
 - School

- The application will provide university names from all universities in the state of California.
- Role
 - The application will support 4 user roles. A user can either be a Student, a Non-Student, an Administrator, or a System Administrator.
- iii. Permission for collecting user data
 - The user must initially consent to have the following data collected from them:
 - a. IP address
 - b. Login Attempts
 - c. Successful Logins
 - d. Time and Location of Feature Use
- iv. Security Questions
 - The application will support 15 different security questions for the user to choose from.
- v. Country IP data of the user
 - The application will support IP addresses of users in North America within the IP address range 1.22.56.0-223.197.44.239.
- d. Initial Audience
 - i. College students in North America.

2. In Scope Functionality

- a. User Registration.
 - i. Student Account
 - ii. Non-Student Account
 - iii. Administrator Account
- b. Login & Logout
- c. Logging/ Archive
- d. Error Handling
- e. Discussion Forum
 - i. Ask questions
 - ii. Answer questions
- f. Searching
 - i. Users can search discussion topics, tutors, and other students.
- g. Ranking System
- h. Text Messenger
 - i. Chat communication between users.
- i. User Profile
- j. Notification System
 - i. Subscribe to discussion posts.

3. Out of Scope Functionality

- a. Tutoring
 - i. Tutors can schedule a tutoring session.
 - ii. Students can join a tutoring session.
 - iii. Payment
- b. Image message type in Text Messenger.
- c. Image post type in Discussion Forum.
- d. Out-of-App notifications using Push API.
- e. Video chat support for tutoring sessions.
- f. User can buy items in the point shop.
- g. User can communicate in a general live chat.

4. Constraints

- a. Time
 - i. Our final deadline is May 2019.
- b. Resources
 - i. Our team is compiled of six members.
 - ii. As students, our budget is limited.

Project Function and Non-functional Requirements

Functional Requirements	Non Functional Requirements
One web page is used to house the application.	
The page changes to display new section based on user interactions.	Page loading of new section of web app should take less than 3 seconds.
Web application is deployed on a cloud server to allow anyone to access the page online.	Cloud service shouldn't take down our application for daily maintenance any longer than thirty minutes.
Cloud service should be resistant to most common attacks.	
Application should have a modular design to allow any one part to be modified without it needlessly affecting any other part.	
Parts should be built in a way to allow a unit test to be created for each.	

Business Requirements (Core Components)

Security

Background: Any data and information that is logged into our website will be encrypted or password protected. This will ensure that sensitive information is not stolen by malicious users.

All data and information that identifies a person will be protected. This data can only be accessible in a secure manner. Additionally, passwords will be encrypted.

Actor: Students, Non-Students, System Administrator

Precondition: The program is released and users start to input information in their profiles and discussion forums

Postcondition: The information and logs are protected in application servers and only accessed to provide useful features to users.

User Stories: “As a user, I want my private information protected to not worry about identity theft.” “As a user, I’m providing my information to be able to use the features that will help me.” “As a user, I want control over what information can be collected about me.” “As a system administrator, I want to make sure that any information within our website is protected from anyone besides the students, tutors, and administrators that use or provided them.” “As a system administrator, I want to make sure that the user profiles can only be accessed by the user themselves, where no one else can view them without their permission.”

Business Rules:

- Encrypt passwords in database.
- Provide an End-User License Agreement (EULA) that users must agree to before accessing the system.
 - System administrators are the only ones able to:
 - Add a EULA
 - Update a EULA
 - Delete a EULA
 - Multiple EULA’s cannot be active at once.
 - A user cannot access the system without accepting the EULA.
- Personal information about the user, such as their first name, last name, and location can be collected.
- Telemetry data, such as when a user logs on, their IP address, what content they view, and what features they view is collected to gain a better understanding of how the application is used.
- Users must provide their consent before any data can be collected from them.
- Users can stop the application from having their telemetry data collected at any time.

- The application must be able to show the user the telemetry data that is being collected from them.
- A user who chooses to delete data collected from them can do so by deleting their account.

Pass Criteria:

- When private information about a user is stored in database, only modules that require it to perform their function can access it.
- When a user deletes their account all information about them is deleted from the database, logs, and archives.
- A user performing an action will be recorded.
- After a user stops data collection by the application, no more telemetry data is collected.

Fail Criteria:

- A user is able to modify information from another user's profile.
- Logs, statistics, and any personal information was accessed and modified by anyone who isn't considered an administrator in the application.

Login

Background: A screen that welcomes the user and prompts them to enter credentials to advance further into the program.

Actor: Students, Non-Students

Precondition: The web application has loaded successfully, and the user should know the credentials which can be used.'

Postcondition: Their attempts of login will be validated.

User Stories: "As a student, I want to access features on the web application."

Business Rules:

- Users are required to have an username and password as their credentials to login into the system.
- The usernames that users choose must be any valid email addresses. If they are not, then the account isn't created.
- The password length must be between 12-2000 characters. Any length that is over or under that value is invalid.
- Passwords must be encrypted and and not saved in a plaintext format in order to protect them from being used illegally.

- If someone were to try to brute force their way into the passwords using a non-quantum computing system, the encryption must last as long as 50 years before they successfully crack it.
- If the user is unable to access their account after attempting to 3 times, the account is disabled.
- Only admin accounts are able to reinstate locked accounts if they were disabled from invalid access. If the account is an admin account, then a system admin would be needed to enable it.

Functional Requirements	Non Functional Requirements
Both username and password fields have to be filled to proceed into the application	Password letter should be hidden under *, but it should show the amount of characters which are being typed in.
Check if users enter an email address (check if it has a “@” and a domain name (e.g : .com, .net) in the username field.	Continuously checking and give warning if user enter valid email address format.
Check if users enter password at least 12 characters to 2000 characters as maximum	Continuously checking and give warning if the password is not in valid range
Use hashing algorithm to store password (e.g MD5, SHA-1, SHA-256). Brute force will take over 50 years to crack passwords	Encrypted password should be saved in a secured database.
System will automatically lock user account if 3 login attempts is detected	Have a notification to let user know when their account get locked. Ask user to contact admin for account unlocked
There is an option to allow administrators remove a lock on account	Tell users to reset password before attempting to login again

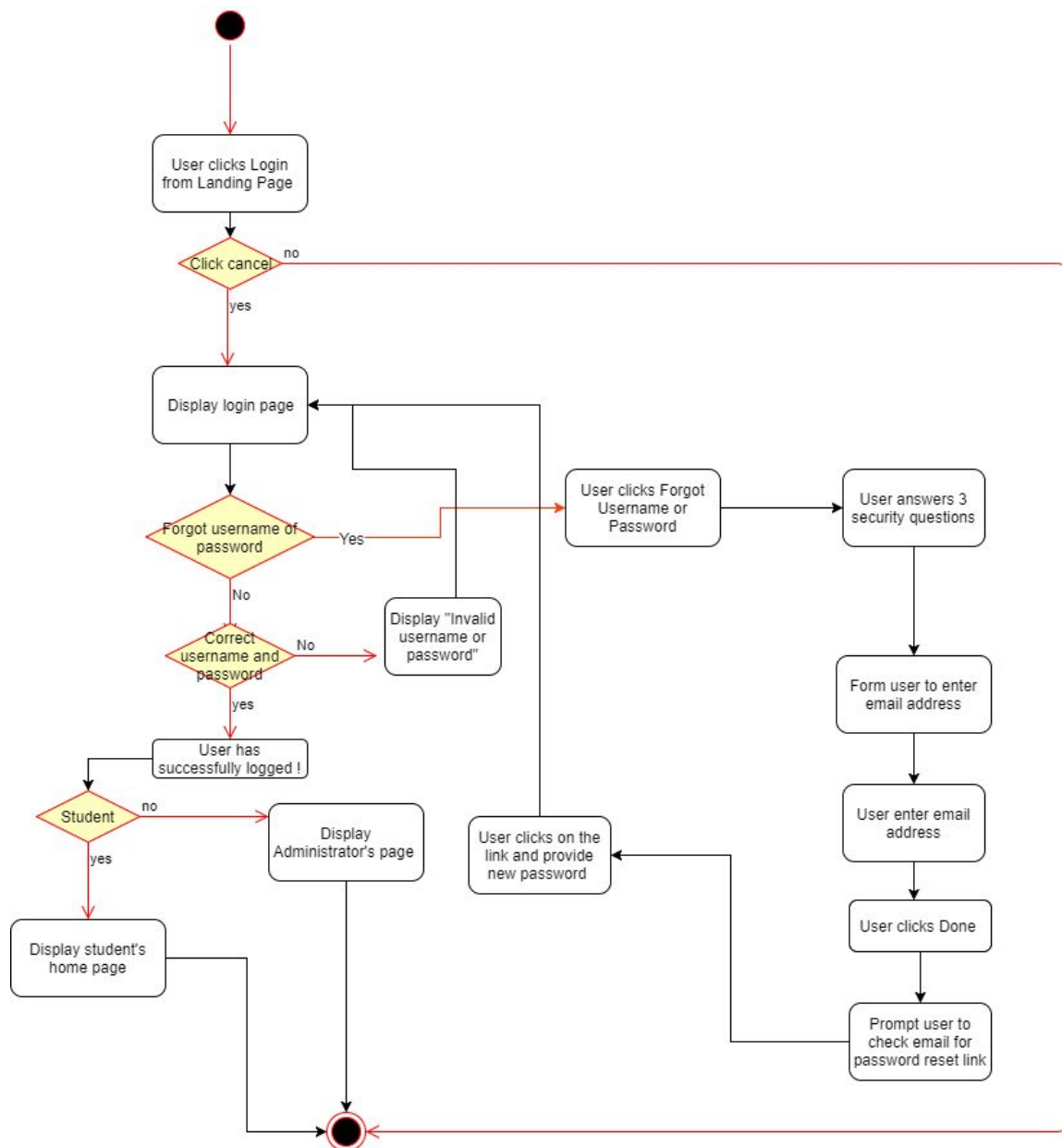
Pass Criteria:

- User enters a correct username and password and the system lets them in.
- User does not enter both a correct username and password and the system gives them an error message telling them it was entered incorrectly.
- User enters a valid email address format and the system accepts it.
- User enters an invalid email address format and the system gives an error message.
- User enters a password between 12 and 2000 characters and the system accepts it.
- User enters a password less than 12 characters or longer than 2000 characters and the system gives an error message.
- A brute force attack does not crack a password within 50 years.

- User's account is locked after 3 failed login attempts and they can no longer attempt to login.
- Account lock is removed by an administrator and the user can attempt to login again.
- User can use the web applications features right away after submitting valid credentials.
- System checks user's email address and displays an error message if it is entered incorrectly.
- System checks if user entered password within the range of 12 and 2000 characters and checks password validity if it is within range.
- System checks if user entered password within the range of 12 and 2000 characters and displays an error message if not within range.
- A brute force attack does not crack password within 50 years.

Fail Criteria:

- User enters a correct username and password and the system does not let them in.
- User does not enter both a correct username and passwords and the system lets them in.
- User enters a valid email address format and the system gives an error message..
- User enters an invalid email address format and the system gives accepts it.
- User enters a password between 12 and 2000 characters and the system gives an error message.
- User enters a password less than 12 characters or longer than 2000 characters and the system accepts it.
- A brute force attack is able to crack a password in less than 50 years.
- User can continue to attempt to log in, even after 3 failed attempts.
- Account is unsuccessfully unlocked by administrator and user still cannot attempt to login to their account.



Password reset

Background: An option allow users have their password reset in case if they forget

Actor: Student, Non-Student.

Precondition: The users forget password and know the answer for three security questions

Postcondition: The users have their password reset successfully by assigning the account a new password

User Stories: “As a student, I forgot my password, but I still want to login to the application.”

Business Rules:

- During an user's session, they have the right to update their password as long as their session has not been timed out.
- If the users wish to reset their password, they must not be logged into the system.
- In addition to answering 3 security questions, users must provide a secondary authorization in order to confirm that the user is actually changing the password and not anyone else.

Functional Requirement	Non Functional Requirement
In user account setting ,there is an option to update password. User have to enter the current password and enter the new password twice.	Old password and new password have to be hidden under “*”.
Password reset option is only available in login page . The option is not available after user login successfully.	May offer password reset after user give wrong credentials 3 times.
To reset the password, user have to answer 3 security questions. In addition, user have to submit their registered email and click on the password reset link.	The password reset link will be expired in 24 hours.

Pass Criteria:

- User enters correct current password along with new password twice and their password is updated.
- User does not enter current password along with new password twice and system gives them an error message.
- User is able to select the reset password option before login into their account and they begin the process to reset their password.
- User enters answers 3 security questions correctly and completes secondary authorization to reset their password and their password is reset.
- User does not answer 3 security questions correctly and completes secondary authorization to reset their password and is given an error message.

Fail Criteria:

- User enters correct current password along with new password twice and their password is not updated.
- User does not enter correct current password and is able to update their password.
- User does not enter the same new password twice and is able to update their password.

- User selects the reset password option before login into their account and process to reset their password does not initiate. .
- User enters answers 3 security questions correctly and completes secondary authorization to reset their password and their password is not reset.
- User enters answers 3 security questions correctly and completes secondary authorization to reset their password and their password is not reset.
- User does not answer 3 security questions correctly and completes secondary authorization to reset their password and their password is reset.

Log out

Background: A button to allow users to log out their account.

Actor: Students, Non-Students

Precondition: The user is already logged in successfully.

Postcondition: The user successfully logs out the application.

User Stories: “As a student, I want to log out the web application and end my session.”

Business Rules:

- As long as users are in active session, they can logout anytime.
- System will terminate the session if it does not detect any user's activity after 30 minutes.
- A session inactivity is when a user has not opened any browsers or done any action to the page for at least 30 minutes.
- If a user decides to communicate with the server before 30 minutes are up, the user session will be extended.

Functional Requirement	Non Functional Requirement
There is a logout button in the user panel.	The logout button will be labeled “Logout” and always be visible to the user.
There is a feature that counts down for 30 minutes to detect inactivity .	User will be notified at the last 5 minutes.
Web server will automatically terminate user’s session if the 30 minute countdown timer runs out.	The user will be notified of their inactivity with a message.

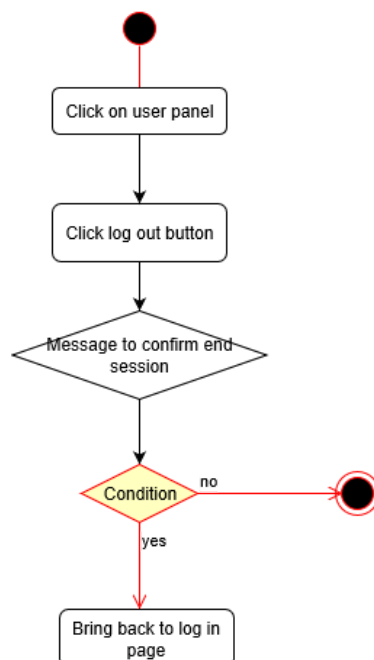
Web server will extend user session immediately when it detects a new activity and reset the countdown timer.	There is an option to allow user to continue the session when they receive a notification in the last 5 minutes.
---	--

Pass Criteria:

- User can log out successfully after clicking logout button.
- System successfully counts down user's inactivity session and detect inactivity after 30 minutes.
- User's session is terminated after 30 minutes of inactivity and will be directed to the login page.
- Inactivity timer resets after web server detects user activity.

Fail Criteria:

- User is unable to log out after clicking logout button.
- System does not detect user inactivity after 30 minutes.
- User's session is not terminated after 30 minutes of inactivity.
- System does not time user's inactivity session
- System times user inactivity even when they are currently active within the web app.
- User's session is not terminated after 30 minutes of inactivity.
- Timer does not reset after web server detects user activity.



User Access Control

Background: There are four types of accounts. Users can manually register for an account but only college students in North America when they register can access system features. Admin accounts can't be created without the developer's help.

Actor: Student, Non-Student, Administrator, System Administrator

Precondition: User must be logged in to the web application.

Postcondition: User can access limited features based on their role.

User Stories: "As a student, I want to access features specific to my school to gain help with class work." "As an admin, I want to manage user activity to prevent any abuse." "As a system administrator, I want to create an admin account to delegate menial activities related to managing users."

Business Rules:

- Students can only see discussion topics specific to their school.
- Non-Students can register, but cannot access any other features that are specifically meant for students..
- Students that aren't tutors won't have a calendar nor will they be able to create sessions.
- Students will not have the privileges of an administrator or systems administrator.
- Administrator will not be able to create or delete administrator accounts and won't have any of the privileges that are unique only to the system administrator.
- Users who are within North America when they register are not authorized to use the website functionality.
- Unauthorized users will be redirected to an error page when they try to log in.
- Users who are unauthorized to access a page will be given an error message before being redirected to an appropriate page.
- User session must be authorized before user access control (UAC) validates their permissions.
- Users should have an interface to specify how much of their telemetry data and user content can be recorded by application.
- Users need to 18 or over to gain access to system.

Functional Requirements	Non-Functional Requirements
Students can use <u>Discussion Forum</u> , <u>User Profile</u> , <u>Tutoring</u> , <u>Searching</u> , and <u>Text Messenger</u> .	Students should have access to these features at all times, unless the user is disabled.

Students can only edit information from their own <u>User Profile</u> .	Students should have access to this feature at all times.
Tutors can use all of the student features and the <u>Tutoring</u> feature to set up sessions. Unofficial transcript and 30 EXP allows users to become a tutor.	The process of screening tutor has to be less than a day.
Administrators can use <u>User Management</u> feature. Administrators cannot use <u>Discussion Forum</u> , <u>User Profile</u> , <u>Tutoring</u> , <u>Searching</u> , and <u>Text Messenger</u> .	Administrators should have access to these features at all times.
Admin can remove or add a user's ability to ask or answer questions.	Admin can apply actions to a collection of users.
Admin can downgrade or upgrade a user between student and tutor.	Only Students can become tutors. Tutors can only be downgraded back to Students.

Pass Criteria:

- Students, tutors, and administrators have access to the appropriate features and has access after selecting a certain feature they have access to.
- Students, tutors, and administrators have access to the appropriate features and will be given an error after selecting a certain feature they do not have access to.
- Student can edit information from their profile and it will update successfully.
- Student will not be able to see an "edit" option when viewing another student's profile.
- Tutor still has access to all student features after becoming a tutor..
- Student can become tutor after completing the criteria to become a tutor.
- Administrator can use User Management feature.
- Administrator does not have access to Discussion Forum, User Profile, Tutoring, Searching, and Text Messenger features.
- Administrator can remove a user's ability to ask questions.
- Administrator can remove a user's ability to answer questions.
- Administrator can add a user's ability to ask questions.
- Administrator can add a user's ability to answer questions.
- Administrator can upgrade a student to a tutor.
- Administrator can downgrade a tutor to a student.

Fail Criteria:

- Students, tutors, or administrators do not have access to all their appropriate features.
- Students, tutors, or administrators are able to access a feature they do not have access to.
- Student can edit information from their profile and it will not update successfully.
- Student can edit information another student's profile.
- Tutor does not have access to all student features.
- Student can become a tutor without meeting all the criteria to become a tutor.
- Student cannot become a tutor even after having all the criteria to become a tutor.
- Administrator has access to Discussion Forum, User Profile, Tutoring, Searching, or Text Messenger features.
- Administrator cannot remove a user's ability to ask questions.
- Administrator cannot remove a user's ability to answer questions.
- Administrator cannot add a user's ability to ask questions.
- Administrator cannot add a user's ability to answer questions.
- Administrator cannot upgrade a student to a tutor.
- Administrator cannot downgrade a tutor to a student

User Management

Background: Specialized accounts will be made for administrators to manage the web application and user activity. They will have the privilege to enable a user, disable a user, delete a user, and delete an inappropriate post from a user.

Actor: Administrator, Student, Non-Student

Precondition: An administrator must have an administrator account.

Postcondition: An administrator is able to implement administrator-only features.

User Stories: "As an administrator, I want to manage the capabilities of all users."

Business Rules:

- Administrator privileges
 - Administrators have the function to enable a disabled user account whether they are a student or non-student.
 - If an account is already active, it cannot be enabled again.
 - Administrators do not have the privilege to enable other administrator accounts that are disabled.
 - Administrators have the function to disable an user account whether they are a student or non-student.
 - If a user is disabled, they will not be given access to login. They will be notified that they are disabled at the Login page.
 - If an account is already disabled, it cannot be disabled again.

- Administrators do not have the privilege to disable other administrator accounts that are active.
 - Administrators have the function to delete a Student or Non-Student user account.
 - All PII data within the database and the logs will be deleted along with the account.
 - Administrators do not have the privilege to delete other administrator accounts.
 - Administrators can alter the UAC for Student or Non-Student user accounts.
 - Administrators do not have the privilege to modify the UAC for other administrator accounts.
- System administrator privileges
 - System administrators have the function to enable a disabled administrator account.
 - System administrators have the function to disable an active administrator account.
 - System administrators have the function to delete an administrator account.
 - System administrators have the function to modify the UAC of administrator accounts.
- Student and Non-Student user privileges
 - Delete their own account.
 - All PII data within the database and the logs will be deleted along with the account.

Functional Requirements	Non-Functional Requirements
An administrator can disable a user.	A user is able to access the web application, but is not able to ask questions, answer questions, schedule tutoring sessions, or send messages.
An administrator can enable, disable, and delete a user by clicking a button next to their name.	The enable button will be labeled “Enable”, the disable button will be labeled “Disable”, and the delete button will be labeled “Delete”.
After clicking to disable a user, a text box will appear for the administrator to input a reason.	The text box will be large enough to clearly read 500 characters. The font will be black and legible.
An administrator must enter a reason for disabling a user.	The reason must coincide with our banning policy.
An administrator can enable a user.	The user is able to access all the features privileged to them based on their role.

A secondary message will appear after clicking the “Enable”, “Disable”, or “Delete” user buttons for the admin to confirm their action. The message will ask the admin to confirm by providing two buttons to either cancel or submit their request.	The cancel button will be labeled “Cancel” and the submit button will be labeled “Submit.”
An administrator can delete a user’s posts from the database.	The post must go against our discussion forum posting policy.

Pass Criteria:

- Administrator can successfully disable a user and user can no longer log in.
- Administrator can successfully enable a user and user can attempt to log in again.
- Administrator can successfully delete a user and user will be removed from the system.
- When disabling a user, a message must appear asking the administrator to input a reason after clicking to disable a user.
- Administrator successfully enters reason for disabling user and user account is disabled.
- Error message is displayed if administrator does not enter a reason for disabling a user.
- After disabling a user, the user must not be able to post a question or answer, schedule tutoring sessions, or send messages.
- After enabling a user, the user must have full access to the features of a user of that role.
- All of the buttons carry out their intended actions.
- Confirmation message is displayed to administrator after clicking the enable button.
- Confirmation message is displayed to administrator after clicking the disable button.
- Confirmation message is displayed to administrator after clicking the delete button.
- Administrator can successfully delete a user’s post.

Fail Criteria:

- Administrator cannot successfully disable a user and user can still attempt to log in.
- Administrator cannot successfully enable a user and user cannot log in.
- Administrator cannot successfully delete a user and user still appears in the system.
- An administrator is allowed to disable a user without inputting a reason.
- When disabling a user, a message does not appear asking the administrator to input a reason after selecting to disable a user.
- After disabling a user, the user is still able to post questions and answers, schedule tutoring sessions, or send messages.
- After enabling a user, the user’s access to features is still limited.

- There is no confirmation message displayed to administrator after clicking the enable button.
- There is no confirmation message displayed to administrator after clicking the disable button.
- There is no confirmation message displayed to administrator after clicking the delete button.
- Administrator is unable to delete a user's post.

Registration

Background: Users who decide to access and use our web application must create an account.

Actors: Students, Non-Students, Administrator

Precondition: Users must have a valid email address not already attached to an account.

Postcondition: User is registered to the web application and can now login.

User Stories: “As a student, I want to access the features of the web application.” “As an administrator, I want to manage user activity on this web application.”

Business Rules:

- User Registration
 - In order for users to register, they must provide an username and password for their account as well as their personal information such as date of birth and location of where they live.
 - Users must also provide answers to 3 security questions we provide in order to secure their access from unauthorized users.
 - If the user chooses an username that already exists in the system, even if the account is disabled, then the user cannot register unless they choose a different unique username for their account.
- Administrator Registration
 - Administrators will only need to input an username and their date of birth and where they are located in order to create their account.
 - Administrators do not have the ability to create admin accounts for new users.
 - The passwords that the administrators will use are randomly created and unique among each administrator.
 - The administrators are not able to select their security questions and answers during the registration process.
 - Like users, the administrator cannot choose an username that already exists in the system even if the account is disabled.

- It is only during the first time the administrator logs in that they can create their own password and choose new security questions to answer to.

Functional Requirements	Non-Functional Requirements
A College Student must enter a school issued email address.	Email address must end in '.edu' for student account.
Users will be sent a confirmation email to validate their submitted email.	A unique code will be sent to each user.
The user's information is added to the database.	All information submitted through registration is stored.
Name, Username, Password, and Security Question Answers will be given text fields in order for the user to fill.	All fields will be an appropriate size, shape, and color for the user to see. All fields will be labeled.
Role, Birth Date, Location, University, Major, and Security Questions will be given drop down menus for the user to select.	
There is a button under all the credential fields to complete registration.	The button will be labeled "Submit"

Pass Criteria:

- A user submits valid information and they are added to the database as a user.
- A user submits invalid information and they are not added to the database as a user. They are given an error message.
- Student successfully enters a school issued email address and they are added.
- The user receives a confirmation email to the email address they inputted after submitting valid information.
- User is given an error message when submitting invalid email information.
- User's information is added to the database after confirmation.
- User can enter username, password, and security answers and it will be added.
- User will be given an error message if not all fields are completed.
- User can select role, birthdate, location, and security questions from a drop down menu.
- The confirmation email provides a link informing them they have been successfully registered.

Fail Criteria:

- A user submits valid information and they are not added to the database as a user.
- A user submits invalid information and they are added to the database as a user.

- Student enters a non '.edu' email address and they are added.
- Student enters a valid school issued email address and they are given an error message.
- User is not given an error message after entering invalid email address
- The user does not receive a confirmation email to the email address they inputted after submitting valid information.
- User's information is not added to the database after confirmation.
- User cannot enter username, password, or security answers.
- User cannot select role, birthdate, location, or security questions from a drop down menu.
- User can proceed without completing all necessary fields.
- The confirmation email provides a broken link.

Error handling

Background: When an error occurs, we will keep the application functional so that users can continue using the application. When an error in the system occurs, the user will be shown a message with a simple and understandable explanation. When the user makes an error, they will be shown a simple message of how to overcome it.

Actor: Students, Non-Students, Admin

Precondition: A system error or user error occurs.

Postcondition: The user is shown a message explaining the error and the web application is still functional.

User Stories: "As a user, I want to know why a feature does not work the way I think it should."

Business Rules:

- A user friendly message will appear whenever there happens to be errors on the client side. This includes server errors from timeouts and requests as well as user invalidation to inputs and access. Any error handling will result in the system contacting an administrator.
- User friendly messages will also appear to errors on the server side. This includes invalid requests and errors from the server and unauthorized access to areas the server should not have access to. An administrator will also be contacted if this ever occurs.
- The client side should present appropriate details about the issue to the user.
- The administrator must also have the details regarding any client side or server side errors.
- Error exceptions should not crash the system unless they are critical errors such as server/network shutdown.

- Any error that is found must have an actionable message that will give instructions to overcome it.

Functional Requirements	Non-Functional Requirements
A user is not able to register using a blank or an invalid email address or password. A message will prompt the student to enter new credentials.	Error messages must be in a red font, located in a position noticeable to the user. Error messages must be clear, simple, and no more than 2 sentences.
A user is not able to log in using a blank or an invalid email address or password. A message will prompt the user to try again with new credentials.	
A student is not able to post a blank question or answer. A message will prompt the student to input text or cancel the post.	
When searching for students, tutors, or discussion topics that do not exist, a message will inform the student to change their search input.	
An administrator cannot ban a user without a reason. A message will prompt the administrator to input a reason before banning.	

Pass Criteria:

- The web application is still running and functional after an error occurs.
- An error message appears after each error occurs.
- Error message will appear when entering invalid email or password.
- Error message will appear when a field for email or password are left empty.
- Error message will appear when user attempts to enter an empty question to post.
- Message will appear when a search result comes back empty.
- Error message will appear when administrator attempts to ban a user without a reason.

Fail Criteria:

- The web application is no longer functional after an error occurs.

- No error message appears after entering an invalid email or password.
- No error message appears after a field for email or password are left empty.
- No error message appears after a question field is left empty.
- No message will appear to tell user to enter something when they attempt to post an empty question.
- No error message will appear when administrator attempts to ban a user without a reason.

Logging / Archiving

Background: Every action that a user takes, whether it is the student, administrator, or tutor, will be recorded and saved into a database. This will allow any administrator to look at the data in case they need to troubleshoot errors, manage the user analysis dashboard, review a ban or review a deleted message.

Actor: Administrator, Students, Non-students

Precondition: The web application is active.

Postcondition: Logs are continuously added and archived.

User Stories: “As an administrator, I want to log information about the client, server, and user activity to help manage the web application.”

Business Rules:

- Error Logging:
 - The system administrator should be notified if there is 100 error logs within the system
 - The system administrator is the only user that is able to delete the error logs
 - Information regarding the error is recorded on the server for all users. This includes the day it occurred, what error message it resulted, where in the code the error came from, who the user was, and what the user did that resulted in the error.
- Telemetry
 - The system administrator should be notified if there happens to 100 failed telemetry logs within the system.
 - User data will be collected for every user unless the user opt out certain specific data. Data that will be collected would be the data and time the user logged in, logged out, what page they visited, what they executed within those pages, and the IP address and location of the user during their session.
- Malicious Attacks
 - Any requests that is being sent to the server will be collected, logged, and checked in order to monitor for any suspicious denial of service attacks to our website.

- Log Archiving
 - Logs that are older than 30 days are stored into the database in order to free space for any newer logs.
 - Any logs that reach over 2 years of age should be archived and deleted from the system in order to hold newer logs.
 - The archive will retry every 2 hours when it fails to do so.
 - The administrator will only be notified if the archiving fails more than 3 times.

Functional Requirements	Non-Functional Requirements
Metrics to log <ul style="list-style-type: none"> ● Input validation failures (invalid parameter) ● Output validation failure (unable to match data in database) ● Authentication success and failures ● Authorization failures ● Session management failures ● Application error (include runtime errors, system error , connectivity problem) ● Configuration change ● System startup and shutdown ● High risk functionality (e.g addition or deletion of user, changes to privileges, data import and export) 	All logs need to have date and time All logs need to have application identifier (e.g service name, module name code location ,) All logs need to have user identity or source address Need to specify type of events and description

Pass Criteria:

- A log is made during appropriate situations.
- Every events mentioned above that alter the application is logged when it needs to.

Fail Criteria:

- The logging system failed to record an event occurring.
- The log does not have sufficient information.

Usage Analysis Dashboard

Background: Various metrics will be tracked and organized into visual graphs in order to analyze user activity and improve user engagement.

Actor: Administrator, Student, Non-Student

Precondition: The web application is fully functional and released.

Postcondition: Data is gathered and stored as users interact with the application.

User Stories: “As an administrator, I want to know the flow of activity of the users while using the web application.”

Business Rules:

- In order to track usage analysis, user login and logout as well as the duration of a user session will be recorded and collected from the logs.
- After data is collected from the logs, the results should be visually displayed as either a bar and a line graph.
- For bar graphs:
 - The bar charts would have a chart that records the average successful logins from every month compared to the total amount number of registered users. The chart would also show maximum and minimum bars.
 - The bar chart would also show the average session duration that users use the website every month which will also include the maximum and minimum bars.
 - The bar chart will record the number of failed login attempts as well as the number of successful ones.
 - The bar chart will also record the top 5 features that were used the most as well as the top 5 average time spent per page within the system.
- For line graphs:
 - There should be a chart that records the average session duration per month over the course of 6 months.
 - There should also be a chart that will record the number of users that logged in per month over 6 months.
- The data that is collected should also be available to the administrator.

Functional Requirements	Non-Functional Requirements
<p>Additional Tracked Metrics:</p> <ul style="list-style-type: none">● Number of total users● Number of active users● Number of users who are tutors● Number of new users● Number of disabled users● Number of posted questions● Number of posted answers● Number of scheduled tutoring sessions	<p>Metrics should be displayed in either a bar or line graph.</p> <p>The following metrics should have a count by days, weeks, months, and years.</p>

<ul style="list-style-type: none"> • Number of students scheduled for tutoring sessions 	
A user is considered “inactive” when they have not posted to a discussion forum or scheduled a tutoring session.	Inactivity must last for at least a week.

Pass Criteria:

- The usage analysis dashboard will return bar and line graphs that specify the tracked metrics for the website such as the number of users, questions posted, and questions answered.
- The users that are considered “inactive” will be labeled out from the rest in order for other users to know whether that person has been an active user within the web application.

Fail Criteria:

- The data from the bar and line graphs do not return or return inaccurate data from the tracked metrics.
- Users that are considered “inactive” have no labels that differentiates between them and active users within the web application.

Data Store Access

Background: Users will be able to access the information on our website based on the privileges we give them. If they use the website to access information, their choices in accessing that data will be limited. If they happen to be moderating the website instead, they would need to access information in our website for any justifiable reason.

Actor: Students, Administrators, and Non-Students

Precondition: The users logs in successfully.

Postcondition: The students would be able to access information from the forums and user profiles. The administrators will have full control to look at any information within our website, moderating the site for any potential abusers.

User Stories: “As a student, I want to access the data in the forums and in user profiles in order to look for help with my assignments.” “As a non-student, I want to look at the application and the information it has.” “As an administrator, I want to access logs, files, and any information students put up in order to see if a student was using the site improperly”

Business Rules:

- The program has the ability to create new records and add them inside the storage.
 - The program must be able to detect duplicate entries and, if there are, not add new records to the data store if another exact record exists.
 - The program cannot add records to the data storage if there is not enough physical space within the system.
 - The adding operation must be quick unless the application feature does not require it.
- The program must be able to read every kind of data from the data store.
 - The program should not take any action if the record isn't found inside the data store.
 - The amount of data that a user can read depends on the data store and the limitations of the machine.
 - The read operation must be quick unless the application feature does not require it.
- The program must be able to update records that are inside the data storage.
 - If the records that the user is trying to update isn't found, then the program doesn't need to access the data store.
 - The update operation must be quick unless the application feature does not require it.
- The program must be able to delete records that are inside the data storage.
 - The program cannot access or delete from the data store if the records the user is looking for is not found.
 - The delete operation must be quick unless the application feature does not require it.
- The program must be able to control what data the Data Access Layer request can retrieve based on certain scenarios.

Functional Requirements	Non-Functional Requirements
Students can access data in forums, adding data into the forums.	The students would be able to only see information and user interface that was meant for student users, based on their school.
If a tutor signs in, they should also have the same access as students, as the only difference between them would be the scheduling system.	Tutors would see the same features as students along with their scheduling system.
If an administrator signs in, they should be given the option to access logs, change	The Administrators will have access to view all data, but they cannot add to the

forums if necessary, and access information for justifiable reasons.	discussion forum or schedule tutoring sessions.
Non-students are able to observe the data in the forums, but cannot add into it. They are also unable to look at user profiles.	Non-students are only able to add, edit, and delete data from their own User Profile.

Pass Criteria:

- The students, non-students and tutors can only access data in forums.
- A tutor cannot access more data than a student can.
- The administrator should have all the power to access any files that they need to look at, in order to make sure the website is safe to students to use.
- A non-student will not be able to add any data and can only see the data.

Fail Criteria:

- The students, non-students and tutors can access data outside forums.
- A tutor has more access than a student.
- Students, non-students, and tutors being able to access personal or restrictive files when they do not have the permission to do so.
- Administrators unable to access any sort of document or file from the users, tutors, or the website itself.
- A non-student is able to add data and can see user profiles.

Network Communication

Background: As our web application is an online service, the server will need to use the Internet in order to allow the users use and communicate with the service.

Actor: Users

Precondition: The application is released and is connected to the internet.

Postcondition: The users are able to access the application through the internet.

User Stories: “As a user, I want to access to the web application from anywhere through the internet.”

Business Rules:

- The system must be deployed into the internet so users have access to it.

Functional Requirements	Non-Functional Requirements
-------------------------	-----------------------------

Use cloud service Azure to deploy application so it would be accessible over the Internet.	The website must be secured in order to protect information from being read in plain text.
Users should be able to log into our website at any time unless due to emergencies.	Loading speed should be under 5 seconds.

Pass Criteria:

- The web application is accessible from the internet when typing in the URL link.
- Users are able to log into our website under normal circumstances

Fail Criteria:

- The web application is limited or has no accessibility from the internet when typing in the URL link.
- Users are unable to log into our website under normal circumstances.

Documentation

Background: The web application needs to provide Developer Documents, User Manual, and FAQ to anyone who has an access to the internet.

Precondition: Person who has an access to the internet

Postcondition: User is able to access to the documents

User Stories: User is able to access to the documents for the information.

Business Rules:

- Anyone who have access to the web application can access to Developer Documents, User Manual, and FAQ.
- User Manual and FAQ are designed for users who don't have any type of engineering knowledge.
- Developer Documents are designed for software developers or those who understand the program..
- System administrator is the only one who can modify the contents of User manual and FAQ.
- User Manual must include everything about the web application.
- The FAQ section doesn't need to include everything about the web application.
- Error exception always must provide a Remedy(actionable message).

Functional Requirements	Non-Functional Requirements
-------------------------	-----------------------------

Users are able to download the documents as pdf file.	Font size of the documents: 12 pts, Arial
---	---

Pass Criteria:

- User downloads and has access to pdf file.

Fail Criteria:

- Users is unable to download pdf file.
- User is unable to view pdf file after downloading it. .

Business Requirements (Application Features)

Discussion Forum

Background: Users will be able to post questions on the discussion forum. In addition, users will be able to answer questions from other users. The ability to post questions will rely on a point system which will also reward users with points for answering other questions. Giving correct answers to questions will also grant the user EXP in addition to points. The discussion forum will be regulated by the point system as well as by giving users the ability to mark answers as “helpful” and “unhelpful” as well as giving them the ability to mark posts as “spam”.

Actor: Student, Non-Student, Administrator

Precondition: User is logged in to their account.

Postcondition: A user views existing questions and answers on the discussion forum and is able to post new questions and answers based on the points they earn.

User Stories: “As a student, I want to use the discussion forum to post and answer questions.”

Business Rules:

- Users should have the options to navigate forum such as, department, class, section
- Highest rated answers should appear on top
- A question should have 500 to 2000 characters
- User should be able to save a question as a draft to complete later
- A reply to a question should be allowed until it is closed
 - The user can close it or it will automatically close after some amount of time passes

- Students cannot post questions if they do not have enough points
- Students can't edit their questions after someone answers
- Students can't edit their answers after posting
- Students can't mark their own answer as "helpful" or "unhelpful"
- Students can only get a maximum of 10 points per day for answering a question
- Questions posted by users with low EXP will be filtered out of the discussion forum
 - Less than 20 EXP

Functional Requirements	Non-Functional Requirements
Questions are displayed in chronological order on the forum.	Start at most recent post.
Answers are ordered by rank of users which give them	Start at answer with highest rank.
There is an option to post a question. Students can type in a text box then choose an option to submit that question to web application.	Posts cannot be deleted once posted.
Each question has a reply button under it. Students click the button to open a reply text box to submit an answer.	Users cannot answer their own question.
System will deduct points from student's account in order to approve a question.	10 points will be deducted for each question.
If a user does not have a sufficient amount of points to ask a question, the system will not post the question.	No points will be deducted.
User will gain points and EXP after answering a question.	User will gain 2 points for answering a question and 10 points for posting the correct answer. User will gain 5 EXP for posting the correct answer.
Option to mark another user's answer as "helpful" or "unhelpful".	User will gain 5 points and 1 EXP for every answer marked "helpful"
Option to mark a post in the forum as "spam".	The spam report will be used to identify spammer.
User can upload a picture as a question or an answer.	The size limit is 5MB
Display a poster's name next to their post.	The font and size used for poster's name should be different from the question.

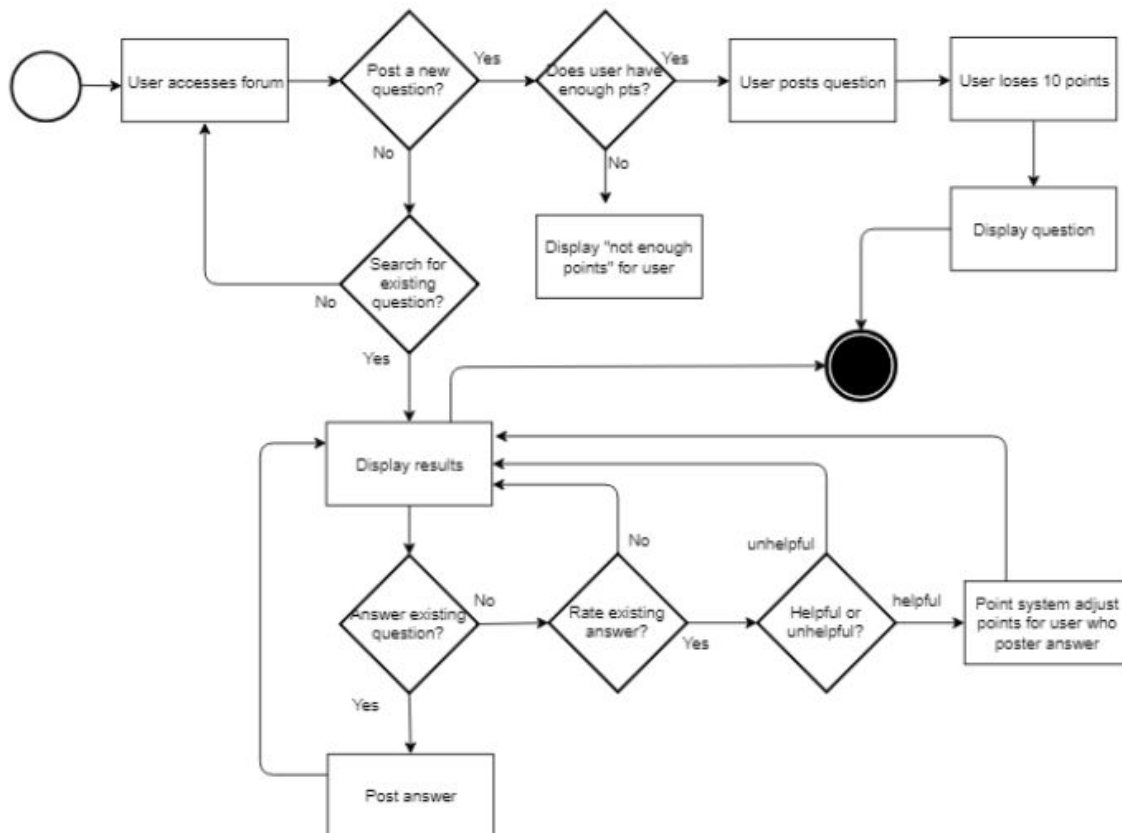
Option to click on a user's name to view profile in a new tab.	A user's name should be clearly clickable.
--	--

Pass Criteria:

- Questions are displayed in chronological order on the discussion forum.
- Answers are displayed by rank of users when accessing a question.
- User clicks to post a question and types their question and it posts to discussion forum.
- User clicks to reply to a question and a reply is added under that question in the discussion forum.
- User loses 10 points from their profile after posting a question.
- User attempts to post a question with insufficient points and error message is displayed to user.
- User answers a question and their EXP is increased.
- User successfully searches for existing question and finds an answer without the need to create a new post.
- User marks an answer as “helpful” or “unhelpful” after they receive an answer.
- User marks an answer as spam and the questions spam count is increased.
- User cannot post a picture as a question or answer.
- User can view a user's profile after clicking on their name next to the question.

Fail Criteria:

- Questions are not displayed in chronological order on the discussion forum.
- Answers are not displayed by rank of users when accessing a question.
- User clicks to post a question and types their question and it does not get posted to the discussion forum.
- User clicks to reply to a question and a reply is added under that question not in the discussion forum.
- User does not lose 10 points from their profile after posting a question.
- User posts a question even if they do not have sufficient points.
- User answers a question and their EXP is not increased.
- User is able to mark their own answer as “helpful” or “unhelpful”.
- User cannot mark another student's answer as “helpful” or “unhelpful”.
- User marks an answer as spam and the questions spam count is not increased.
- Point system does not work correctly and users can post without having enough points or do not receive or lose points correctly.
- User is unable to post when they should be allowed to do so.
- User cannot post a picture as a question or answer.
- User is not displayed a user's profile after clicking on their name next to the question.



Search

Background: Users will be able to search the application based on the specificity of the information they seek. Discussion forums, students, and tutors can be found based on subject, course, or professor.

Actor: Student, Administrator

Precondition: The student logged in successfully.

Postcondition: The student is able to find discussion forums, other students, and tutors while filtering by subject, course, or professor.

User Stories: “As a student, I want to find other students in my same major.” “As a student, I want to find tutors who specialize in this course.” “As a student, I want to ask a question about this professor’s homework.”

Business Rules:

- A user can filter search categories.
 - A student can search for discussion topics.
 - A student can search for other students.
 - A student can search for tutors.
 - A user can further filter each search category by subject, course, or professor.

- A maximum of 25 search results will display per page.

Functional Requirements	Non-Functional Requirements
A search bar will be displayed for users to input text.	A maximum of 100 characters can be searched.
A drop down menu will be next to the search bar for the user to select how to filter the search.	The user will have the options to select discussion posts, students, or tutors.
A search button will be next to the search bar to submit the search.	The button will be a search icon.
If a student selects a subject, the search results will be filtered based on that subject.	A student can search for topics, students, and tutors within a subject.
If a student selects a course from a subject, the search results will be filtered based on that subject.	A student can search for topics, students, and tutors within a course.
If a student selects a professor from a course, the search results will be filtered based on that subject.	A student can search for topics and students within a professor's course.

Pass Criteria:

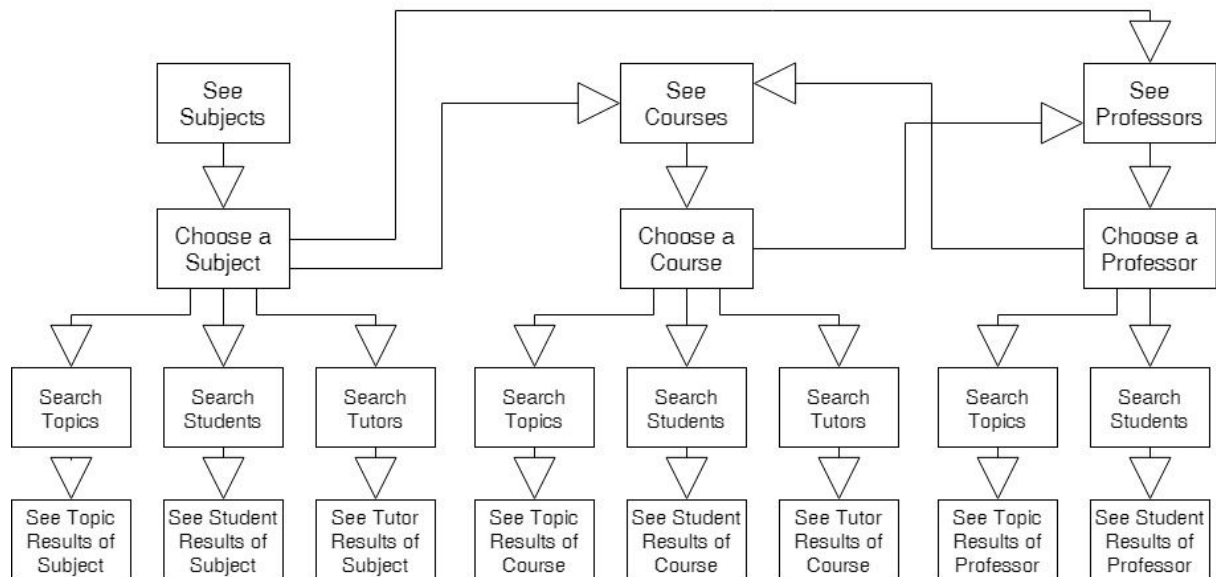
- Users are able to see and use the search bar which will allow users to input text.
- Users can use the drop down menu next to the search bar and the filters accurately choose the correct links or questions based on what the user chooses.
- The search button next to the search bar works when clicked with the mouse key.
- If a certain subject is chosen within the filter, then the only search results that a user should see is ones based on the subject.
- If a certain course is chosen from the subject within the filter, then the only search results that a user should see is ones based on the course.
- If a certain professor is chosen from the course within the filter, then the only search results that a user should see is ones based on the professor.

Fail Criteria:

- Users cannot see nor use the search bar and does not allow users to input text.

- Users cannot use the drop down menu or the filters do not accurately choose the correct links or questions.
- The search button does not work or send the user to suggestions for their question.
- Users get inaccurate results when choosing the subject within the filter.
- Users get inaccurate results when choosing the course within the filter.
- Users get inaccurate results when choosing the professor within the filter.

Search Flow:



Tutoring

Background: Students can choose a tutor that may help them with whatever subject is causing them problems. Students can sign up to an available session of the tutor and write a review afterwards. A student can become a tutor and advertise a weekly schedule of when they are available to teach.

Actor: Student

Precondition: A student has registered and obtained an account.

Postcondition: A student has set a time to meet with the tutor and obtains help.

User Stories: “As a student, I want to get help from a tutor and learn more about this subject.” “As a student, I want to find the most effective tutor that meets my needs best.” “As a student, I want to share my experience with my tutor after our session.” “As a tutor, I want to set my own tutoring availability.”

Business Requirements:

- Student Focused Requirements:
 - Display courses the tutor teaches.

- Calendar will display the sessions the tutor has set up.
- A student can sign up to an open session of their choice.
- A student may write a tutor evaluation after they have attended one of their sessions.
- Students can't schedule tutoring sessions for the rest of the month if they have two no-shows for tutoring sessions.
- Students can only book one session at a time. After their session is over, they can book another.
- A student can't sign up to a session that has already passed.
- A student who signs up to a full session is added to a queue.
- If a student registered to a session cancels, the first student in queue is added to session automatically, if they aren't currently registered to a session.
- Tutor Focused Requirements:
 - Tutors can indicate when a student doesn't show up to a tutoring session for which they signed up.
 - If a large number of students want an open session, tutors are required to schedule one group session a week.
 - Allow integration of a third party chat service to allow students and tutor to collaborate.
 - Record the amount of time the tutor and user are using the chat system.
 - Student prompted to pay the day before session starts if they haven't paid or their session will be canceled.
 - Payment made to website, then percentage of it goes to tutor.
 - Tag system is used to indicate whether the student is present for a session on time and/or they complete the session without disappearing.
 - Tutors can set sessions to not allow students that have certain tags.
 - Tutors have to be students
 - Tutors can be rated to indicate how effective they are in their work helping students.

Functional Requirements	Non-Functional Requirements
The reviews of a tutor and the courses they can teach are displayed when clicking on their name.	25 reviews are displayed per page. 25 courses are displayed per page. 25 results per page.
The available sessions of a tutor can be displayed on a calendar and a session can be clicked to sign up to it.	Calendar refreshes every 5 minutes.

A message is displayed when a student clicks on a session, asking the user to confirm the appointment.	
The message will restate the date, time, and tutor, to clarify the session.	The text color will be black and have a font size between 10 to 12
A tutor can indicate if a student is a no show.	
A student can unsign up after scheduling a session.	Can unsign up until 24 hours before the session begins.
Once session is filled up, attempts to add it will send the user to queue.	The message will be in red font above the session.
Registration to a session closes an hour before it starts.	
Previous sessions will stay displayed on the tutor's calendar.	They become unclickable 10 milliseconds after the threshold has passed.
Tutor can edit the session until one student has signed up to it.	<p>A tutor can't select a date or time for a session before the current date and time.</p> <p>A tutor can't register a session a year after the current date or later.</p>
<p>Tag System:</p> <ul style="list-style-type: none"> • Serious student • Good student • Standard • Flaky student • Multiple misses 	<p>Missing three sessions will move a student to "Flaky". Missing another three will move the student to "Multiple misses". After two weeks with no misses the student with "Multiple misses" will move to Flaky and the student with "Flaky" will move to "Standard".</p> <p>"Serious" and "Good" indicate students who, not only haven't missed any sessions of their last ten sessions, but also are highly rated by tutors.</p>

A tutor with at least three sessions with queues will have to create a group session a week or be banned from the web application.	Number of sessions with queues reset each month.
A student that has been registered to a session will be removed from all queues they are in.	
A tutor can delete session until the day before the session begins.	
When tutor deletes a session, student gets a full refund.	

Pass Criteria:

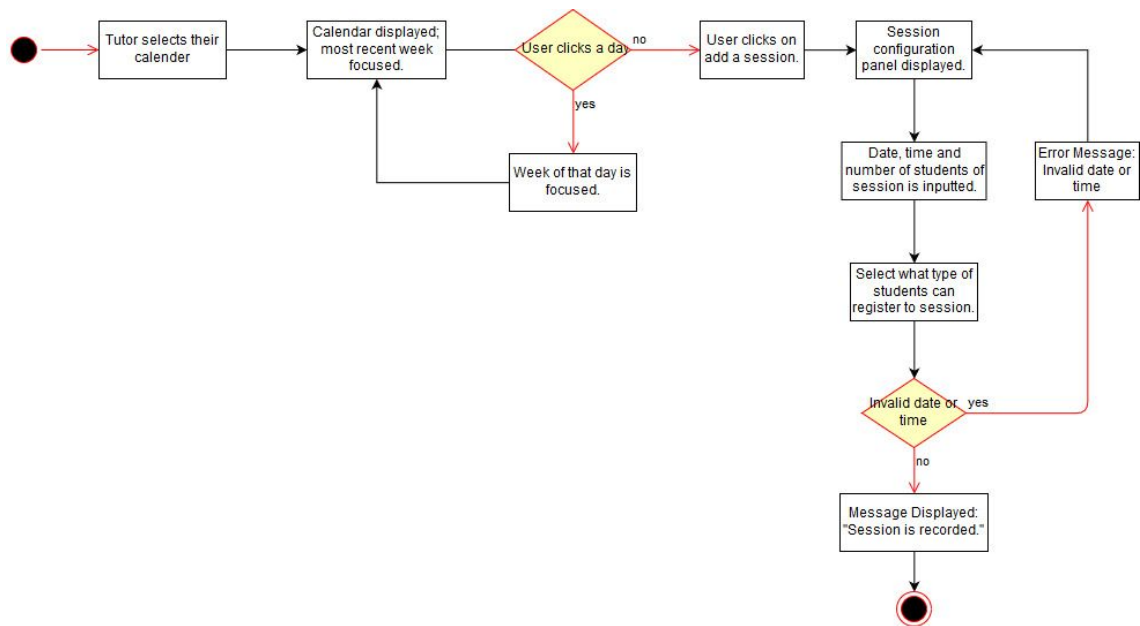
- Clicking on a tutor's name will display the reviews and courses of a tutor
- A session that still hasn't occurred is clickable.
- A student selecting an open session will cause a prompt to be displayed asking if they would like to be a part of that session.
- Message displays the date and time of the session.
- When student signs up to an unfilled session, the system will record their name to that session.
- A student that becomes a tutor will have calendar to record sessions on their profile page.
- Clicking on the calendar will open a new page displaying the calendar.
- Clicking on no show for a student's name will label that student as a no show for the session.
- A session that has passed will become unclickable.

Fail Criteria:

- A student clicking on a session with no open spots will cause the application to display a message indicating the no more students can sign up for this session.
- A tutor selecting a date or time before the current date and time for a tutoring session will cause an error message when they try to add it.
- A tutor selecting a date a year after the current day will cause an error message when adding it.

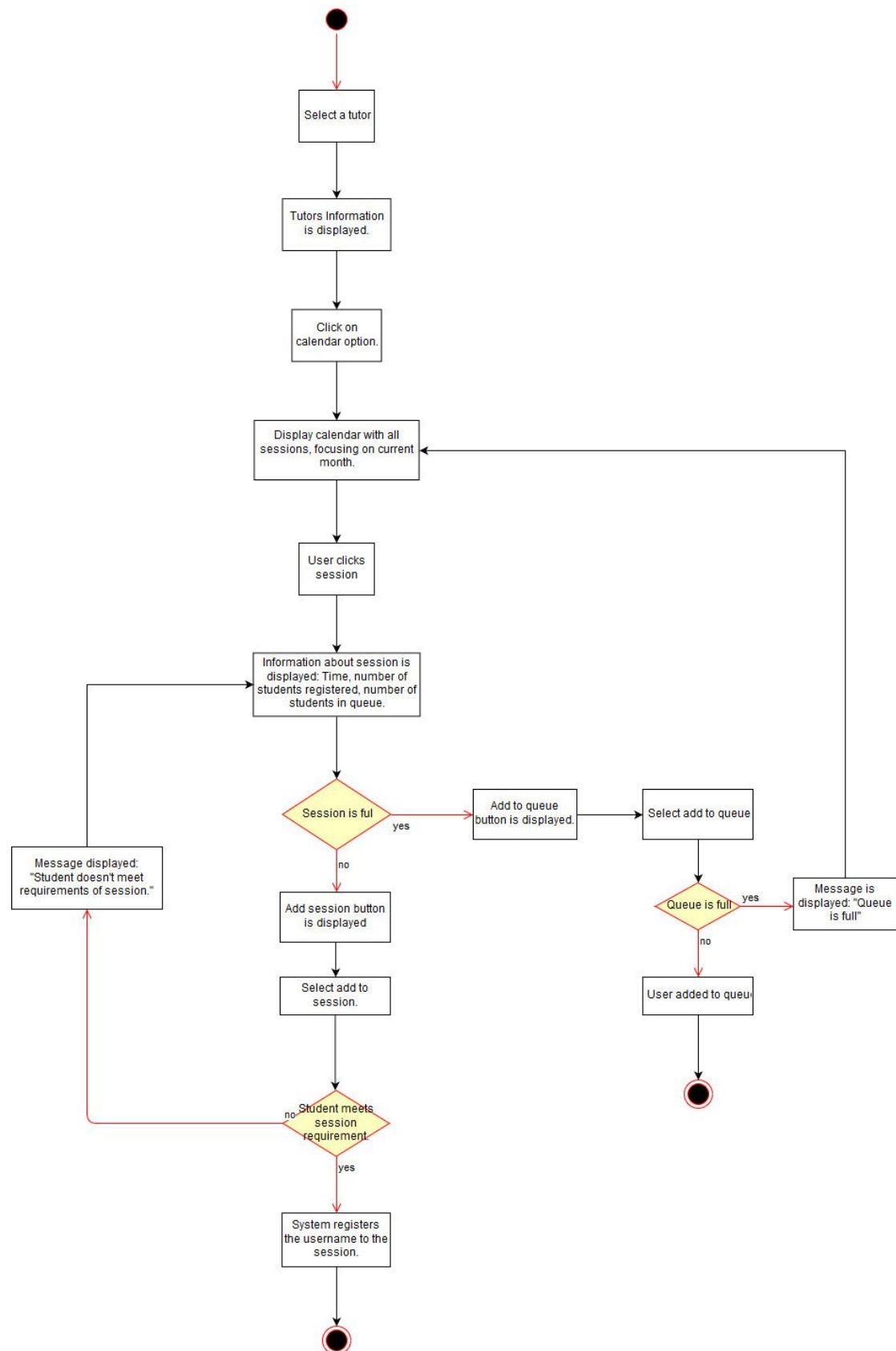
Precondition: A tutor wants to add a session.

Postcondition: Session has been registered to system and is displayed on calendar.



Precondition: A student has registered and obtained an account.

Postcondition: A student has set a time to meet with the tutor and obtains help.



Ranking System

Background: Users will be ranked based on their activity and accuracy of their answers. It will encourage users to answer other user's questions, increase the reputation of tutors, and eventually increase the number of active users.

EXP: A numerical representation of how accurately a user answers questions.

Actor: Students

Precondition: A student has a registered account.

Postcondition: A user's EXP is affected by answering questions correctly.

User Stories: "As a tutor, I want to gain recognition for answering questions correctly." "As a student, I want to see the rankings of potential tutors."

Business Rules:

- Users don't lose their EXP points

Functional Requirements	Non-Functional Requirements
A user gains EXP for posting the correct answer to a question.	5 EXP
A user gains EXP for getting marked "helpful" on an answer.	1 EXP

Pass Criteria:

- The user gains EXP based on whether they post the right answer to the question.
- The user gains EXP if someone marks their answer "helpful" in the discussion forum.

Fail Criteria:

- The user does not gain or loses EXP to answers that are correct.
- The user does not gain or loses EXP for posting their answers.
- The user does not gain or loses EXP when their answers are marked "Helpful" by other users.

Text Messenger

Background: Users will be able to use the real-time text messaging to communicate with other users.

Actor: Students

Precondition: User owns an account of My Academic Pyramid.

Postcondition: User is able to send and receive the messages with other users.

User Stories: “As a student, I want to communicate with other students privately, off the discussion forum.”

Business Rules:

- Each message should contain less than a thousand characters.
- User doesn't receive the message from the blocked user.

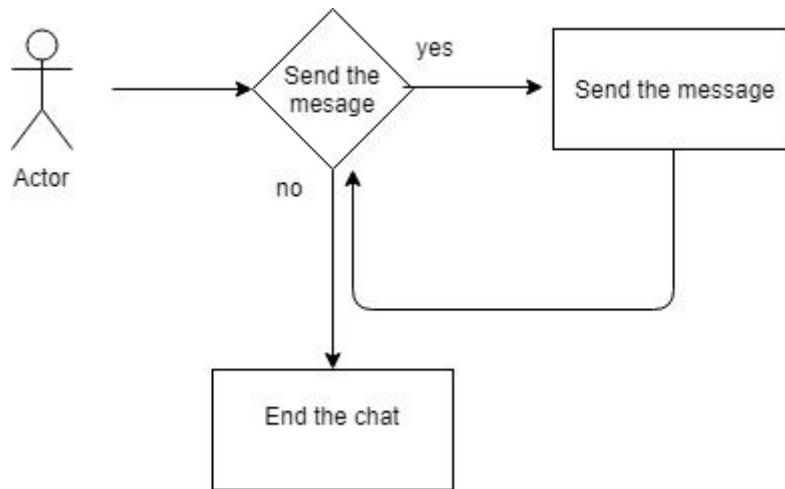
Functional Requirements	Non-Functional Requirements
A chat window will open that displays the conversation between two users.	The chat window can be full screen or partial screen.
A text box will be within the window to input messages to send.	Each message should contain less than 1000 characters.
A button will be next to the text box to send messages	The button will be labeled “send”

Pass Criteria:

- Users will see a chat window that will show conversations from other users.
- Users will also be able to input text from the text box in order to send messages.
- Users will be able to send a message by pressing the button.

Fail Criteria:

- Users won't be able to see a chat window or open one.
- Users won't be able to input any text from the text box when they press the button next to the text box..
- Users won't be able to send messages when they press the button next to the text box.



User Profile

Background:

User profile is a feature that stores comprehensive information of a student. The information includes student's nickname, social media rank, activity history including questions and answers, friends, interests online status, date of registration, and an option to send a private message.

Users can share their personal information with other users. It helps users to find the other users who have same interests.

Actor: Students

Precondition: User owns an account for My Academic Pyramid.

Postcondition: Users are able to add their personal information into their account and share it with other users.

User Stories: "As a student, I want to share my personal information to connect with other students."

Business Rules:

- Users can display their picture in their profile
- Users can connect their account into their social media account
- Users can see their activity history in their user profile tab
- Users can hide their profile at any time.
- Users can add or modify their information.

Functional Requirements	Non-Functional Requirements
There are field for user to fill in name and social media address	Mouse click and touch screen will be supported for all user interaction.

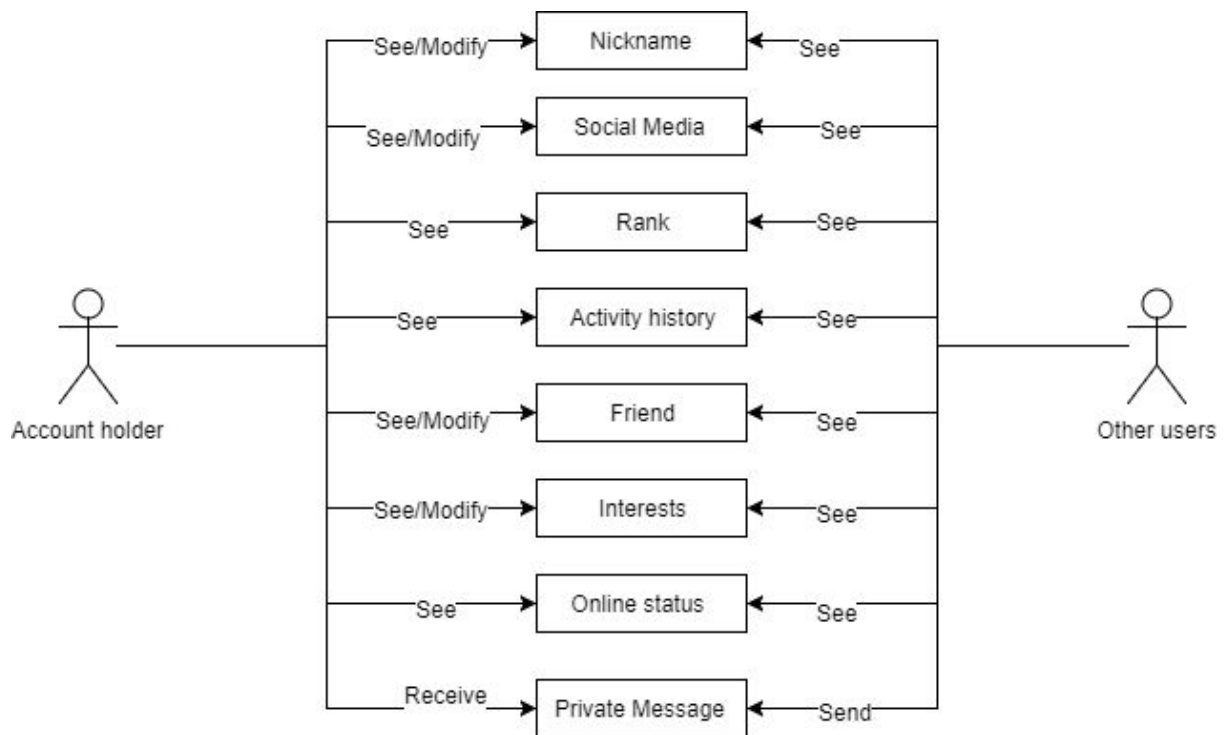
	Text Fields will be labeled.
There is an upload button for users to upload their profile picture and a picture frame to contain the picture.	The size of a profile picture must be less than 3 MB.
There is a list that contain a user's friend's name.	The maximum number of friends is 500.
There is a place to show user rank	Current EXP will be shown.
There is a tab to show user history of activities.	
There is a option for uses to hide their profile	

Pass Criteria:

- Users will be able to see a field that will allow users to input their name and social media address..
- Users can upload photos by pressing the upload button and drag their image to fit inside the picture frame.
- Users will have their own list that contains the user's friend list.
- Users will be able to see their rank on the user profile.
- Users can open a tab that will accurately display the history of activities from the user.
- Users can check an option that will hide profiles from other users.

Fail Criteria:

- Users won't be able to see the field or allow users to input their information.
- USers are not able to upload photos or see the upload button to upload their profile picture. The system will also not be able to read files properly to use the image for certain or all picture files.
- The user's friend list will incurrately show friends from the user.
- There is no interface that shows user rank.
- The user history of activities shows inaccurate data.
- Users are not able to hide their profiles from any user.



Notification System

Background: Users will receive a notification within the app based on a given event that occurs within their account. Users can receive notifications for certain activities depending on what they select on their notification settings. Examples of notifications include: receiving a friend request, a direct message, an answer to a question they asked on the discussion forum, or confirmation for a tutoring session.

Actor: Student, Administrator

Precondition: User is logged in to their account.

Postcondition: User receives a notification with details of an event that occurred on their account.

User Stories: “As a student, I want to be notified when my questions get answered.” “As a tutor, I want to be notified when students submit a request for a tutoring session.” “As an administrator, I want to be notified when a student submits an appeal to reactivate his/her account.”

Business Rules:

- Users should have the option to subscribe to other students’ questions and topic to receive notifications from them.
- Users should receive notifications within web application.

Functional Requirements	Non-Functional Requirements
-------------------------	-----------------------------

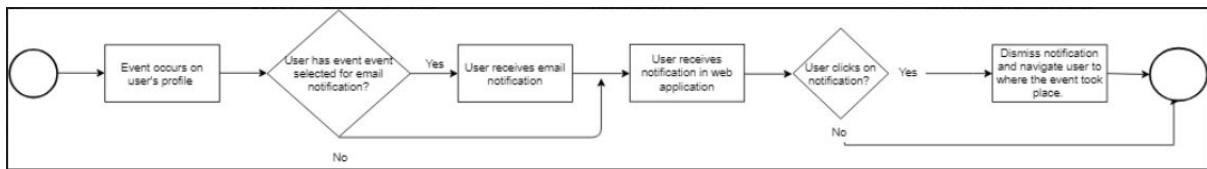
A notification will appear on a user's account to give them notice of an event on their account.	Notification message will be written in bold and font-size 24.
Notifications will remain until user clicks on it.	A red mark will be visible.
Will have a "settings" option so the user can select which notifications they want to receive.	Notifications would be sorted in chronological order and listed in notification tab.

Pass Criteria:

- A user receives a notification on their account and is dismissed once it is resolved.

Fail Criteria:

- User doesn't receive a notification, when other user answers his or her question on the discussion forum.
- Tutor doesn't receive a notification, when a student submits a request for a tutoring session.
- Administrator doesn't receive a notification, when a student submits an appeal to reactivate his or her account.
- User receives a notification, when the user turned off the notification system in the settings.



Payment

Background: Users will be able to submit any payments through the payment feature within the application. A user can pay for a tutoring session and the tutor will be paid as well for the session.

Actor: Student, Non-Student

Precondition: User is logged in to their account and has entered their payment information.

Postcondition: A user successfully makes a payment for any given amount.

User Stories: "As a student, I want to submit a payment in order to receive a tutoring session."

Business Rules:

- User cannot advance without canceling or submitting a payment.
- Payment must be through Paypal.

Functional Requirements	Non-Functional Requirements
Display amount due to user.	Amount displayed should be in accurate US Dollars.
Input for user to enter their payment information.	User payment information should be secure.
Tutor is displayed the amount they will be paid.	Amount displayed should be in accurate US Dollars.

Pass Criteria:

- Tutor receives a payment from the user.

Fail Criteria:

- User enters correct payment information but payment does not go through.
- Tutor does not receive payment submitted by student.

Glossary

- **EULA:** End-User License Agreement
- **PII:** Personally Identifiable Information
- **SPA:** Single Page Application
- **UAC:** User Access Control