Contents

[BTS530 System Requirements 2](#_Toc453933213)

[(1) System Name and Team Number 2](#_Toc453933214)

[(2) Team Members 2](#_Toc453933215)

[(3) Client and/or Mentor 2](#_Toc453933216)

[(4) System Overview 2](#_Toc453933217)

[(5) Technology to be used 2](#_Toc453933218)

[(6) Stakeholders 3](#_Toc453933219)

[(7) A Diagrammatic Representation of the System 3](#_Toc453933220)

[(8) System Components 5](#_Toc453933221)

[Rating Component 5](#_Toc453933222)

[Create Account Component 6](#_Toc453933223)

[Update Component 7](#_Toc453933224)

[Comment Component 9](#_Toc453933225)

[Search Component 13](#_Toc453933226)

[Profile Component 13](#_Toc453933227)

[Recommendation Component 21](#_Toc453933228)

[(9) Who Did What Log 25](#_Toc453933229)

# BTS530 System Requirements

## System Name and Team Number

Study Space Finder Web System

Team 3

## Team Members

Kevin Pham

Faiq Malik

Vincent Wong

Iceal Claveria

## Client and/or Mentor

Kevin Pham - Client

## System Overview

The proposed system will be based on a web application accessible on browser, it should be able to locate potential study and meeting locations based on the user’s current location. The system will have filtration and selection options allowing to search the desired study environment locations. The system will also have the functionality to add new potential study locations based on user recommendations and votes. The system will have an update functionality to remove or add locations that are open or closed.

The proposed system’s database will be set up by us with some generalized locations (e.g. Starbucks, Second Cups, McDonalds, etc.). The monitoring and input of the locations will also be handled by us after validating its location via google and by the number of times the location has been recommended to us along with user inputs and flags. The rating system of this revolves around users rating the location based on its quality. The average of the votes will be shown to other users after reaching a certain point to allow consistency and to prevent locations from rating up their own establishment, the votes will also be unique to the user meaning the user will not be able to rate a location multiple times. The feedback will be monitored by us the overseers and will have a report spam function which will flag it allowing us to check the comment and determine if it will be deleted or not.

## Technology to be used

**Framework:**

ASP.NET Framework - Used to create user-interactive websites

Entity Framework - Used to create and manage databases for websites and other applications

Bootstrap Framework - Used to design websites with more features and modern looks.

**Software:**

Visual Studio - IDE used to develop wide range of applications.

Any Browser

## Stakeholders

|  |  |
| --- | --- |
| **Stakeholder** | **Role** |
| Kevin Pham - Founder | Authority/Domain Expert/Sponsor |
| Faiq Malik – Administrator | Domain Expert |
| Vincent Wong – Administrator | Domain Expert |
| Iceal Claveria Administrator | Domain Expert |
| Students | End User |
| Establishment Owners | End User/Onlooker |

## A Diagrammatic Representation of the System

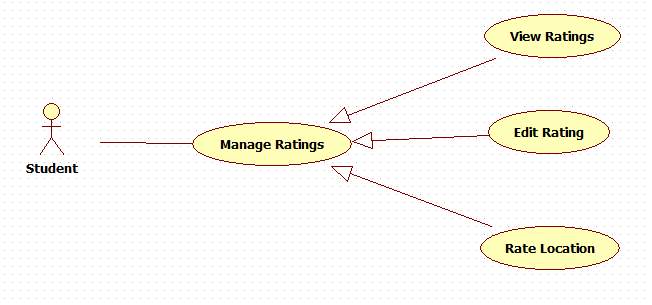
Please See Use Case Diagrams



## System Components

* Rating
* Create Account
* Update
* Comment
* Search
* Profile
* Recommendation

### Rating Component



**Main Flow Student:**

**Preconditions**: Student must be logged in and needs to be viewing the manage account page

|  |  |  |
| --- | --- | --- |
|  | Actor | System |
| 1 | Presses “Rating” | Displays all the ratings on the location |
| 2 | Enters rating and submit | Displays user’s rating on the location |

**Alternate Flows**: Student must be logged in and viewing a location that has already been rated

|  |  |  |
| --- | --- | --- |
|  | Actor(s) | System |
| 1 | Re-Rate the location | New rating is saved. System responds as in line 2 |

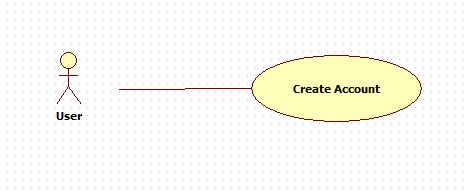
**Test Case Summary**:

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Successfully gave a location a rating | Successfully gives a rating to a location |
| Alt 1: Edit rating | Successfully changes the rating and gets displayed on the location |

Test Case Details:

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Successfully give rating | User enter valid rating between 1 and 5 for the location. | User’s rating gets saved for the location. Rating is then displayed |
| 2 | Alt: Re-rate the location | User enters valid rating between 1 and 5 | User’s rating gets replaced for the location. |

### Create Account Component



**Main Flow:**

|  |  |
| --- | --- |
| Actor(s): User | System |
| Click ‘Create Account’ | * Displays sign up form. |
| User enters all necessary data requested on the form. | * System validates fields as they are entered. If all validations pass, Submit button is enabled |
| User presses ‘Submit’ button | * Account is created and saved into the database |

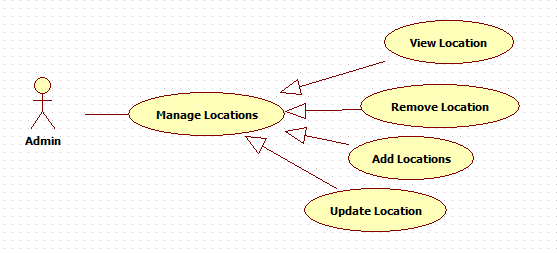
**Test Case Summary:**

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Account created | User successfully creates account |
| Alt 1: Verification Invalid | User failed to create an account |

**Test Case Details:**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: User creates account successfully | User fills in name, username, password, email, security questions and answers | Account gets created |
| 2 | Alt: Verification Invalid | User fills form and input marked not valid | Account not created |

### Update Component



**Main Flow Administrator:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Requests to view all locations | * Displays a list of locations |
| Selects a location to update | * Displays details of the location * Enables an ‘Edit’ button |
| Clicks ‘Edit’ | * Shows forms filled out with the location’s current details (may be subject to change later on) |
| Updates information and requests to save | * Location’s records in the database get updated and saved |

**Alternate Flow Administrator A:**

**Preconditions**: Administrator must be logged in and viewing a list of locations

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Selects a location to view | * Displays details of the location |

**Main Flow Administrator B:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Requests to view all locations | * Displays a list of locations |
| Selects a location to remove | * Displays details of the location * Enables an ‘Delete Location’ button |
| Clicks ‘Delete Location’ | * notification shows asking to confirm deletion of location |
| Confirms | * Location’s records in the database get deleted and database gets saved |

**Main Flow Administrator C:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Requests to view all locations | * Displays a list of locations |
| Click Location | * Displays two options. ‘View Recommendations’ and ‘Add Location’ |
| Clicks ‘Add Location’ | * Shows an empty form of that allows the Admin to add information for a new location. * Fields get validated as they are entered (may be subject to change later on) |
| Fills out forms and requests to save | * The new location gets added and the database updates |

**Test Case Summary:**

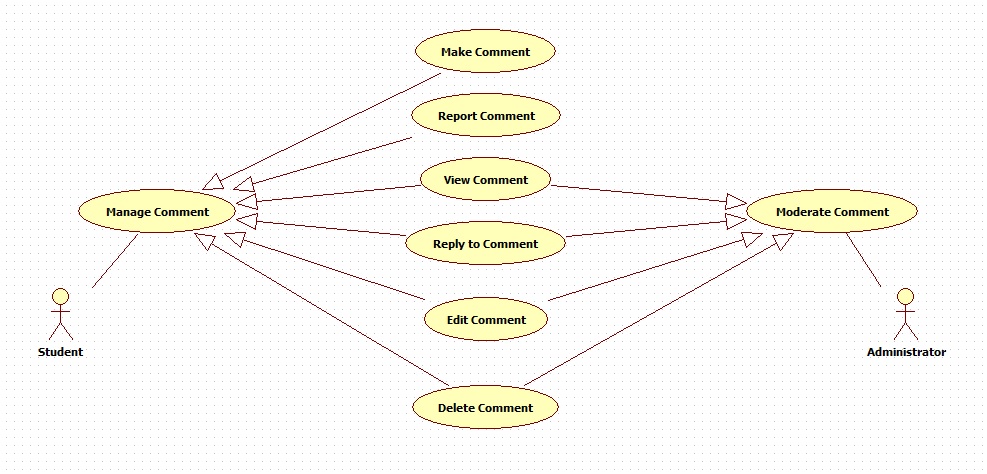
|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Edit location details | User successfully edits location information |
| Alt 1: Cancels editing | User cancels editing location information |
| Alt 2: Adds Location | User adds a location |
| Alt 3: Delete Location | User deletes a location |

**Test Case Details:**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Edit location details | User makes changes on an existing location | Changes made on existing location are saved and location information is updated |
| 2 | Alt: Cancels editing | User cancels editing location details | Changes made on a location are discarded and location is not updated |
| 3 | Alt: Add Location | User adds a new location | New location is saved and added to the location list |
| 4 | Alt: Delete Location | User deletes a location | Existing location gets deleted from |

For the ‘View Recommendation’ flow please check Recommendation component for more information.

### Comment Component



**Main Flow Student:**

**Preconditions**: Student must have an account and be logged in and needs to be viewing the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click ‘Leave a comment’ | * Displays a blank form |
| Enters comments in form | * Has ‘Submit Comment’ button enabled |
| Clicks ‘Submit Comment’ button | * Comment gets saved and displayed |

**Alternate Flow Student A:**

**Preconditions**: Student must have an account and be logged in and needs to be inside the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Clicks ‘View comments’ | * Already has a few comments displayed * Has ‘View more comments’ button enabled |
| Clicks ‘View More Comments’ button | * Displays all comments saved |

**Alternate Flow Student B:**

**Preconditions**: Student must have an account and be logged in and needs to be viewing the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click a comment to reply to | * Displays specific comment * Comment has a ‘Reply’ button enabled |
| Clicks ‘Reply’ button | * An empty form appears under the comment, allowing the user to fill it in (may be subject to change later on) |
| Enters reply and hits ‘Enter’ | * The reply is sent and is displayed underneath the comment (may be subject to change later on) |

**Alternate Flow Student C:**

**Preconditions**: Student must have an account and be logged in and needs to be viewing the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click on a specific comment | * Has specific comment displayed * Has ‘Delete’ button enabled |
| Clicks ‘Delete’ button | * Comment gets removed and is no longer available to be viewed |

**Alternate Flow Student D:**

**Preconditions**: Student must have an account and be logged in and needs to be viewing the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Clicks ‘Report comment’ | * Already ’Report Comment’ button enabled * A pop-up appears asking why the comment is being reported (ie. spam, offensive, other, etc.)   (may be subject to change later on) |
| User specifies reasoning | * Asks for confirmation |
| Confirms | * Comment gets flagged and will not be visible to users. The admins will decide whether or not to delete the comments |

**Alternate Flow Student E:**

**Preconditions**: Student must have an account and be logged in and needs to be viewing the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click comment to edit | * Comment gets replaced with form that is filled with the comment |
| Enters new comment and hits ‘Enter’ | * The old comment gets overwritten by the new comment |

**Main Flow Administrator:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click a comment to reply to | * Displays specific comment * Comment has a ‘Reply’ button enabled |
| Clicks ‘Reply’ button | * An empty form appears under the comment, allowing the user to fill it in (may be subject to change later on) |
| Enters reply and hits ‘Enter’ | * The reply is sent and is displayed underneath the comment (may be subject to change later on) |

**Alternate Flow Administrator A:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Wants to view comments | * Already has a few comments displayed * Has ‘View more comments’ button enabled |
| Clicks ‘View More Comments’ button | * Displays all comments saved |

**Alternate Flow Administrator B:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click on a specific comment | * Has specific comment displayed * Has ‘Delete’ button enabled |
| Clicks ‘Delete’ button | * Comment gets removed and is no longer available to be viewed |

**Alternate Flow Administrator C:**

**Preconditions**: Administrator must be logged in

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click comment to edit | * Comment gets replaced with form that is filled with the comment |
| Enters new comment and hits ‘Enter’ | * The old comment gets overwritten by the new comment |

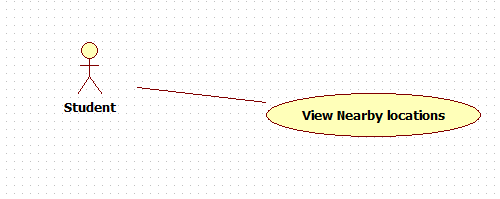
Test Case Summary:

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Successfully creates comment | User successfully adds a new comment for a location and comment shows up in the thread |
| Alt 1: Edit comment | User modifies a past comment and saves. Time stamp shows updated time and edit mark to specify that the comment was edited. |
| Alt 2: Delete comment | Removes comment posted by the user. |
| Alt 3: Reply to comment | User replies to other user’s or their own comment, which gets displayed underneath |
| Alt 4: Report a comment | User flags a comment they deem is inappropriate and that comment is flagged and not available to be viewed |

**Test Case Details:**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Successfully creates comment | User enters in intended comment in comment box | User’s comment is saved and displayed in the thread |
| 2 | Alt: Edit comment | User enters adjusted comment | A specific comment gets updated, saved and displayed with changes made |
| 3 | Alt: Delete comment | User deletes a comment | User’s comment gets removed from the thread and is no longer available to be viewed |
| 4 | Alt: Reply to comment | User types his/her opinion for/against a comment in words | User’s reply gets displayed underneath another user’s or their own comment |
| 5 | Alt: Report Comment | User reports a comment that they think is inappropriate | Reported comment gets flagged and hidden from view |

### Search Component



**Main Flow Student:**

Use Case Name: Search for location

Actor(s): User

Use Case Precondition:

* No required preconditions

Use Case Post Condition(s):

* User finds a location and gets directions on how to get to it

Applicable Business Rules:

* User must have an account and be signed in

Main Flow:

|  |  |  |
| --- | --- | --- |
|  | Actor | System |
| 1 | Presses ‘Locate Me’ | Asks to attain GPS location from users computer and upon approval gets the coordinates. Displays results of study locations |
| 2 | Uses default proximity | Uses the default proximity range |
| 3 | Presses ‘Search’ | Displays location based on the range set around user’s location. |
| 4 | Select a location | Shows location information and total user average ratings |
| 5 | Request for directions | Displays direction details |

Alternate Flows:

|  |  |  |
| --- | --- | --- |
|  | Actor | System |
| A1 | GPS cannot find user’s location | System asks to input custom location (e.g. postal code or address) |
| A2 | Uses custom proximity | Sets the proximity range from default to custom or pre-set ones. System responds as in line 3, proceeds to line 4 and then goes forward. |
| A3 | User selects a different location | System updates directions display |
| A4 | No locations nearby are found | System displays that no locations were found and requests to increase proximity range. |

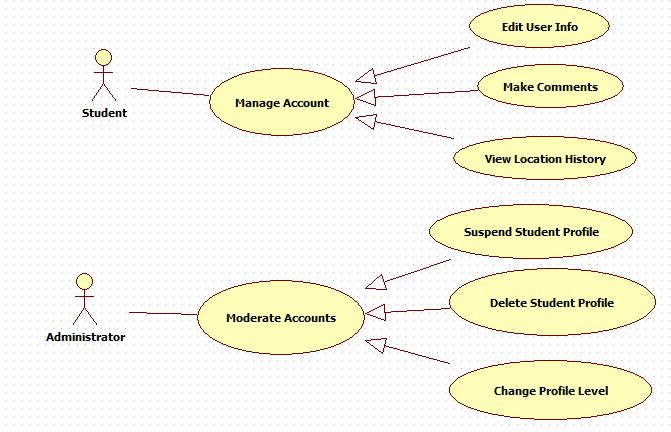
Test Case Summary:

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Successfully finds location | The system is able to find the location of the user and give results according to the inputs. |
| Alt 1: GPS cannot find user’s location | The system is not able to get the location data and a custom input is requested from the user. The rest of the procedure is the same as the main flow. |
| Alt 2: Custom proximity | The user sets a custom proximity and the system updates the results accordingly from the default settings. |
| Alt 3: Change location | The user selects another location rather than the closest location initially chosen by the system. |
| Alt 4: No location found | The system is unable to find a location based on the user’s current proximity, so the user enters a different proximity value until it yields a search |
| Alt 5: Overall Rating | The system displays the calculated overall rating of the location in its description page. |

Test Case Details:

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Successfully finds location | User searches for a location based on their current position on the GPS | Location is found based on user’s current proximity, and directions are ready to be viewed and followed |
| 2 | Alt: GPS cannot find user’s location | User searches for a location based on their current position on the GPS, but GPS is unable to track the user | Message saying “Cannot locate user” is displayed, custom input is requested |
| 3 | Alt: Custom proximity | User sets the range for how far they want to search a location within | Showing results based on set proximity by the user |
| 4 | Alt: No location found | Either the GPS is unable to pinpoint the location or the user entered an invalid address | Error message “No locations found” is displayed, system allows change of proximity |
| 5 | Alt: Overall | Location rating is averaged by the system based on total user rating submissions | Overall rating is displayed on Location description page |

### Profile Component



**Main Flow Student:**

**Preconditions:** Must have an existing account and must be signed in.

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click Manage Account | * Displays the manage account page with all the user information which has options View previously searched location history, Make comments on locations and edit user profile information |
| (Student has nothing to modify) | * No change |

**Alternate Flow Student A:**

**Precondition:** Must be logged in and be inside Manage Account Page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click ‘Edit User info’ | * Displays edit user information page with all the student profile data with modifiable fields |
| Make changes to some fields. | * System validates fields as they are entered. If all validations pass then the save button is enabled |
| Click ‘Save’ button | * Changes saved * Remain at the Edit User page |

**Alternate Flow Student B:**

**Precondition:** Must be logged in and be inside Manage Account Page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click ‘Make Comments’ | * Displays Make comments page with three fields which include Reason for contact (dropdown), Location (dropdown), Message (textbox) and a Submit button   (Subject to change as needed in the future) |
| Make selections on the field and type a message | * System validates fields as they are entered. If all validations pass then the Submit button is enabled |
| Click ‘Submit’ button | * Message sent to administrators. Remain at the Make comments page |

**Alternate Flow Student C:**

**Precondition:** Must be logged in and be inside Manage Account Page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click ‘View Location History’ | * Displays location history page which the user visited in the past as a list of locations.   (Subject to change as needed in the future) |
| Click one of the previous locations visited on the list. | * System shows a google map of the location next to the list with an arrow pointing to it * Shows button ‘Open in google maps’ |
| Click presses ‘Open in google maps’ | * System opens a new browser window and redirects the user to google maps with the destination location preselected with the origin location empty |

**Test Case Summary:**

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Successfully edits profile | Changes are saved and profile is updated |
| Alt 1: Cancels editing profile | User chooses not to make the changes |

**Test Case Details:**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Successfully edits profile | User enters new data in the fields and submits | Changes made to the profile are saved and the profile is updated |
| 2 | Alt: Cancels editing | User clicks “cancel” | Changes made to the profile are discarded and the profile is not updated |

**Main Flow Admin:**

**Precondition:** Must be logged in and be inside Manage Account Page

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click ‘Moderate Account’ | * Displays moderate accounts settings * This page has information like modifying other accounts status and rights level. Suspend and Delete Student profiles |
| (Administrator has nothing to modify) | No change |

**Alternate Flow Admin A:**

**Precondition:** Must be logged in and be inside Moderate Account Page

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click ‘Suspend Student Profile’ | * Displays the Suspend profile page. * Displays a search field with multiple fields.   (Name, Account ID, Email) |
| Admin enters user Email or Name or Account ID and presses Search | * System displays a list of accounts which match the search * The list shows the status of the account, name, email and account ID along with a button ‘Suspend Account’ for each list element   (If account ID is used then only one account matching is shown, if no such account exists then “No Account with ‘Account ID’ exists!” message is shown) |
| Admin presses the ‘Suspend Account’ button from the list. | * System displays the Suspend form page with selected account information already entered * Text box for reason is shown along with suspend duration (in days) box |
| Admin enters reason in the message box and enters duration in the Suspend Duration box. | * System validates the message and the duration * If all validations pass, then the ‘Suspend Account’ button is shown |
| Click ‘Suspend Account’ | * System updates database and is redirected to the search account page * Displays process completion or error message |

**Alternate Flow Admin B:**

**Precondition:** Must be logged in and be inside Moderate Account Page

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click ‘Delete Student Profile’ | * Displays the Delete profile page * Displays a search field with multiple fields   (Name, Account ID, Email) |
| Admin enters user Email or Name or Account ID and presses Search | * System displays a list of accounts which match the search * The list shows the status of the account, name, email and account ID along with a button Delete Account’ for each list element.   (If account ID is used then only one account matching is shown, if no such account exists then “No Account with ‘Account ID’ exists!” message is shown) |
| Admin presses the ‘Delete Account’ button from the list. | * System displays the Delete form page with selected account information already entered * Text box for reason is shown |
| Admin enters reason in the message box. | * System validates the message * If all validations pass, then the ‘Delete Account’ button is shown |
| Click ‘Delete Account’ | * System updates database and is redirected to the search account page * Displays process completion or error message |

**Alternate Flow Admin C:**

**Precondition:** Must be logged in and be inside Moderate Account Page

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click ‘Change Profile Level’ | * Displays the Change Profile page. * Displays a search field with multiple fields.   (Name, Account ID, Email) |
| Admin enters user Email or Name or Account ID and presses Search | * System displays a list of accounts which match the search * The list shows the status of the account, name, email and account ID along with a button ‘Change Account Access’ for each list element   (If account ID is used then only one account matching is shown, if no such account exists then “No Account with ‘Account ID’ exists!” message is shown) |
| Admin presses the ‘Change Account Access’ button from the list. | * System displays the Rights Change form page with selected account information already entered * ‘Grant Admin Access’ button is shown |
| Click ‘Grant Admin Access’ | * System updates account rights and redirects to the search page * Completion or error message is shown |

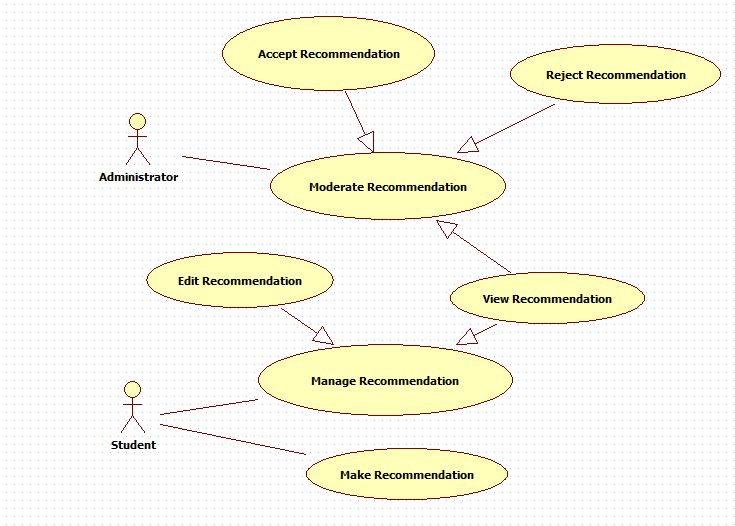
**Test Case Summary:**

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Unflags account | Account gets un-flagged |
| Alt 1: Suspends Account | Account gets suspended and cannot be accessed by the user. |
| Alt 2: Deletes Account | Account gets deleted |

Test Case Details:

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Unflags Account | Admin unchecks flag | Account status gets changed from “flagged” to “unflagged” |
| 2 | Alt: Suspends Account | Admin unchecks flag and checks suspend | Account status gets changed from “flagged” to “suspended” |
| 3 | Alt: Deletes Account | Admin deletes account | Account status gets changed from “flagged” to “deleted” |

### Recommendation Component



**Main Flow Student:**

**Preconditions:** Must have an existing account and be logged in and needs to be viewing the manage account page

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click ‘Make Recommendation’ | * Displays Recommendation page * Shows form for location recommendation which has fields for location name, address, and a few generic questions regarding the services and availability of the location |
| Enter data for the new location | * System validates the information as it is added * Also checks whether the location is already present in the database or not * If all validations pass then the submit button is enabled |
| Click ‘Submit’ | * Location data is entered into the system and an email is sent to the moderators to review the location * Message is shown whether the process has been completed or not |

**Alternate Flow Student A:**

**Preconditions:**

* Must have an existing account and be logged in and needs to be viewing the manage account page
* Must have made previous recommendations

|  |  |
| --- | --- |
| Actor(s): Student | System |
| Click ‘Manage Recommendation’ | * Displays Manage Recommendation page * Shows all the recommendations made by the user under that account * Shows the recommendations as a list * Has button ‘Edit Recommendation’ and ‘View Recommendation’ |
| Click ‘View Recommendations’ | * Displays a list of all recommendations made by the user of the account * Has flags on the recommendations as ‘Under Review’, ‘Passed’ or ‘Rejected’ * Each list object has an ‘Edit Recommendation’ button |
| Click ‘Edit Recommendation’ | * Displays the recommendation page with the prior data entered by the user filled in the fields |
| Modify data in the fields | * System validates the data as it is entered * If all validations pass, then the ‘Submit’ button is enabled |
| Click ‘Submit’ | * Data is updated in the database * The user is navigated to the ‘View Recommendations’ page * Message is displayed whether the modification was saved or cancelled |

**Main Flow Admin:**

**Preconditions:** Must have an existing account and be logged in.

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click ‘Moderate Recommendation’ | * Displays Recommendation page * Shows all the recommended locations as a list * Each list object has button ‘View Recommendation’, ‘Accept Recommendation’ and ‘Delete Recommendation’ |
| Click ‘View Recommendation’ on one of the elements in the list | * System displays all the data for the location * ‘Accept’ and ‘Reject’ buttons are shown at the bottom |
| Click ‘Accept’ | * Pop up appears to confirm Acceptance |
| Click ‘Yes’ or ‘No’ | * If ‘Yes’ then location recommendation is accepted and the location is added to the database for the available locations to search through * If ‘No’ then nothing happens and the pop up disappears * Admin is redirected to the ‘View Recommendations’ page |

**Alternate Flow Admin:**

**Preconditions:** Must have an existing account and be logged in.

|  |  |
| --- | --- |
| Actor(s): Administrator | System |
| Click ‘Reject Recommendation’ | * Pop up appears to confirm rejection |
| Click ‘Yes’ or ‘No’ | * If ‘Yes’ then location recommendation is Rejected and the location is not added to the database * If ‘No’ then nothing changes and the pop up disappears * Admin is redirected to the ‘View Recommendations page’ |
| Click ‘Cancel’ | * All unsaved data will be deleted and the list will remain as it’s unchanged state |

**Test Case Summary:**

|  |  |
| --- | --- |
| Scenario Name | Scenario Description |
| Main: Successfully creates recommendation | User successfully request to add a new location |
| Alt 1: Edit recommendation | User successfully edit recommendation details |
| Alt 2: Accept Recommendation | User accepts a recommendation |
| Alt 3: Reject Recommendation | User rejects a recommendation |

**Test Case Details:**

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Scenario | Data Values/Conditions being tested | Expected Result |
| 1 | Main: Create recommendation | User enters valid location information to be recommended | Successfully requested to add new location |
| 2 | Alt: Edit recommendation | User enters adjustments to the information in the recommendation | Updated recommendation location information |
| 3 | Alt: Accept recommendation | User chooses to accept a recommendation | Recommendation is accepted and location is added to location list |
| 4 | Alt: Reject recommendation | User chooses to reject a recommendation | Recommendation is rejected and location list is not updated |

## Who Did What Log

|  |  |
| --- | --- |
| Team Member | Work completed on this Project (be specific please) |
| Kevin Pham | Remade use cases and added the logic detail to each use case |
| Faiq Malik | Remade use cases and added the logic detail for Create Account, Profile and Recommendation components |
| Vincent Wong | Remade use cases and added the logic detail for Search and Rating components |
| Iceal Claveria | Remade use cases and added the logic detail for Update and Comment components |