CPEN321: Software Engineering

Software Requirements Document

Last updated:
20th November 2016

Group: Cyann

Chen Chen (3281 6143)
Howard Zhou (4898 6137)
Justin Toh (2954 9136)
Luvian Wang (1693 9134)
Xi Chu (1735 3137)
Yufei Qiao (4885 7130)

Table of Contents

1. Product Vision Statement

Cyann is a Q&A board for students and instructors to have better communication.

2. Problem Statement

The problem of	Lack of incentive to communicate
affects	All students and instructors
The impact of which is	Less and less interaction between them
a successful solution would be	A platform consisting of a mobile & web application with a motivation system that enable better discussion between students

3. Product Position Statement

For	All Students and professors
Who	Encounter education communication on a daily basis
Our System	Is a Q&A platform
That	Provides basic posting and commenting functionality
Unlike	Other online forum-based platforms that doesn't provide any incentives for students to actively participate in
Our Product	Provides such motivation

4. User Demographics

Users of Cyann fall into 4 categories: students with questions, students with solution, and instructors and TAs, meanwhile there will be moderators to ensure the application is properly used.

Students with questions: Students with an academic question can post the question to the forum. After other people respond to the question, this student and other students in the class may evaluate the helpfulness of this response. Students with questions may also use email other users in the system, to solve more specific problems.

Students with solutions: Students with solutions can post their answers and gather upvote that will later be recorded in the honour system. After they gather enough honour points, student with questions can select them to offer tutoring before exams.

Instructors and TAs: In addition to actions that can be done by students, instructors can also endorse the best student answer and upload files to the system. However, while instructors can also answer question, they will not get any upvotes and receive any honor points. Instructor also have the option of sending out an email notification to everyone in the class whenever they upload a document.

Moderators: moderators are the developers of this application. They have full access to all data and functionalities and are responsible for ensuring this application is properly used and functioned, and solving any issue raised and any bugs detected by users.

5. Detailed List of Use Cases

UC-1. User Authentication (WEB)

- **Primary Actor**
  - Students and instructor
- **Stakeholders and interests**
  - Students and instructors
- **Preconditions**
  - The user has to have a Facebook account
- **Postconditions**
  - User is authenticated
- **Main Success Scenario**
  1. The Instructor click/touches “Log in via Facebook as Instructor” button.
  2. The student click/touches “Log in via Facebook as User” button.
  3. The TA click/touches “Log in via Facebook as TA” button.
  4. The system popups a window that ask for user’s permission to access his/her Facebook account.
  5. The user click/touches the “login” button to login with their Facebook account
  6. The user gives the application permission to use his/her Facebook account information.
  7. The user enter email and password for his/her facebook account.
  8. The system will get access to his/her Facebook name, profile image and email address.
  9. The user is authenticated
- **Extensions and Alternative Flows**
  1. Server connection lost.

  2a. Facebook authentication failed due to wrong password.
  2b. User gets an error message.
  2c. Authentication failed.

  3a. Users does not give permission to use their Facebook account information.
  3b. The user cannot be authenticated.
- **Open Issues**
  - None

UC-2. User Authentication (MOBILE)

- **Primary Actor**
  - Students
- **Stakeholders and interests**
  - Students
- **Preconditions**
  - The user has to have a Facebook account
- **Postconditions**
  - User is authenticated
- **Main Success Scenario**
  1. User touches “Log in via Facebook as Instructor” button.
  2. The system transitions to Facebook mobile app or Facebook website
  3. The user touches the “login” button to login with their Facebook account
  4. The user gives the application permission to use his/her Facebook account information.
  5. The user enter email and password for his/her facebook account.
  6. The system will get access to his/her Facebook name, profile image and email address.
  7. The user is authenticated
- **Extensions and Alternative Flows**

  2. Server connection lost.

  2a. Facebook authentication failed due to wrong password.
  2b. User gets an error message.
  2c. Authentication failed.

  3a. Users does not give permission to use their Facebook account information.
  3b. The user cannot be authenticated.
- **Open Issues**
  - None

UC-3. User creates a course

- **Primary Actor**
  - Instructors who wants to create a course
- **Stakeholders and interests**
  - Instructor who wants to create a course
  - The instructor is signed in via website
- **Preconditions**
  - User needs to be an instructor
  - User is authenticated
- **Postconditions**
  - After aa instructor successfully creates a course, all users will be able to search this course and add this course to their course list.
- **Main Success Scenario**
  1. User notifies system that he/she wants to create a course.
  2. System displays a course creation form.
  3. User enters the course name and a list of instructors and TAs.
  4. User notifies system that input is completed.
  5. System display the new course on the user's course list.
- **Extensions and Alternative Flows**
  1. User did not supply enough information for course creation.

  1.a Course creation failed.
- **Open Issues**
  - None

UC-4. User updates a course

- **Primary Actor**
  - Instructors who wants to create a course
- **Stakeholders and interests**
  - Instructor who wants to create a course
  - The instructor is signed in via website
- **Preconditions**
  - User needs to be an instructor
  - User is authenticated
- **Postconditions**
  - After an instructor successfully updates a course, all users will be able to search this course and add this updated course to their course list.
- **Main Success Scenario**
  1. User notifies system that he/she wants to updates a course.
  2. System displays a course updating form.
  3. User enters the updated course information (e.g. course's name, instructors and TAs).
  4. User notifies system that input is completed.
  5. System display the updated course on the user's course list.
- **Extensions and Alternative Flows**
  2. User did not supply enough information for course creation.

  1.a Course creation failed.
- **Open Issues**
  - None

UC-5. User registers a course

- **Primary Actor**
  - User who want to add a new course
- **Stakeholders and interests**
  - User who is taking this course currently
  - User who is teaching this course currently
- **Preconditions**
  - User is authenticated
- **Postconditions**
  - A new course will be displayed on the user's course list
  - The user will be able to access all of the course's resource. e.g: posts, assignments and lecture notes
- **Main Success Scenario**
  1. User notifies system that he/she wants to add a course to his/her course list
  2. System displays a course adding form
  3. User enters the course name
  4. System search for this course name
  5. User taps the "add" icon next to the course's name
  6. System add this course to the user's course list
  7. User navigates to the main page of this course
- **Extensions and Alternative Flows**
  2. System cannot find the course in the database

  2.a System prompts a error message
- **Open Issues**
  - None

UC-6. User creates a post

- **Primary Actor**
  - Students with questions
- **Stakeholders and interests**
  - All registered users who are able to see the post.
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
- **Postconditions**
  - Question is posted.
  - Question can be seen and answered by authenticated users.
- **Main Success Scenario**
  1. The user clicks/touches an existing course to enter it.
  2. The user clicks/touches the button for posting a question.
  3. The system display an embedded text editor for the user to enter his/her question.
  4. The user types his/her post.
  5. The user touches the "post" button to push the question into the forum corresponding to the course.
  6. The system will display all posts in this course, with the one posted just now on top.
- **Extensions and Alternative Flows**

  1a. Server connection lost.

  1. User gets an error message.

  2.a User post a course-irrelative question.

  1. Post gets deleted by instructor/moderator

  3.a The user who made the post is an instructor

  1. System sends an email notification to every student registered in the class
- **Open Issues**
  - None

UC-7. User reads a post

- **Primary Actor**
  - All users registered in a certain course
- **Stakeholders and interests**
  - Students and Instructors in a certain course communicate with each other through reading posts.
  - Posts solves some users' questions.
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
- **Postconditions**
  - None
- **Main Success Scenario**
  1. User go to a course page and click to open a certain question
  2. System display the content of this post & a list of all the post's comment.
  3. User goes back to the previous view
  4. System returns to the previous view that user was in.
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-8. User edits a post

- **Primary Actor**
  - Students with questions or instructors
- **Stakeholders and interests**
  - All registered users who are able to see the post.
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
  - User has to be the post's author.
- **Postconditions**
  - Question was updated.
- **Main Success Scenario**
  1. The user click on the "edit" button to edit his/her own question
  2. The user then edits his/her own question in a textbox
  3. The user touch the "post" button to push the edited question to a forum
  4. Other users see the edited question on the forum
- **Extensions and Alternative Flows**

  1.a Server connection lost

      1. User gets an error message
- **Open Issues**
  - None

UC-9. User deletes a post

- **Primary Actor**
  - Instructors or the post's author.
- **Stakeholders and interests**
  - All registered users who are able to see the answer.
- **Preconditions**
  - The post has been created.
  - User has to be an instructor or be the author of the post being deleted
- **Postconditions**
  - The post is deleted.
  - The deleted post will not be displayed when a user views a list of posts.
  - Any honor points gained through the posts will be retained.
- **Main Success Scenario**
  1. User navigates to the posts view
  2. User clicks the cross button at the upright corner of his/her post
  3. System prompt for user confirmation using a modal
  4. User click confirm button to proceed this deletion
  5. The post is removed from database
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-10. User creates a comment on post

- **Primary Actor**
  - Students who want to ask a questions or provide a solution for a post
  - Instructor who wants to make a comment or answer a question for a post
- **Stakeholders and interests**
  - Students with questions, students with solution
  - Students who answer the question are able to gain honor points
- **Preconditions**
  - User is authenticated.
  - User is registered in the course.
- **Postconditions**
  - After successful comment creation, all users in the class will be able to see the created comment
- **Main Success Scenario**
  1. User navigates to the post's view
  2. User enters a post by touching on it
  3. User touches the "add comment" button to reveal comment textarea
  4. User enters the text of the comment in the comment textarea
  5. User touches the "add comment" button notifies system that input is completed
  6. System displays the created comment on the post's view with the information of it's author and created date
- **Extensions and Alternative Flows**

  5.a Text edit window is blank

  1. System prompt a warning that an empty response cannot be accepted.
  2. System continues prompting this warning until the user type something in the text edit window.
- **Open Issues**
  - None

UC-11. User edits a comment

- **Primary Actor**
  - Students with questions, students with solution & Instructors wants to edit their comments being made
- **Stakeholders and interests**
  - User who want to edit their posted comments/questions
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
  - User has to be the comment's author.
- **Postconditions**
  - After successfully editing the comment, all users within the class will be able to see the edited comment
- **Main Success Scenario**
  1. User navigates to the post's view
  2. User notifies system that he/she wants to edit a comment
  3. System displays a comment editing textarea
  4. User enters the new text in a text-area of the comment
  5. User notifies system that input is completed
  6. System displays the edited comment on the post's view and update the comment created date
- **Extensions and Alternative Flows**

  1.a User's text for edited post is blank

  1. System shows error message
- **Open Issues**
  - None

UC-12. User deletes a comment

- **Primary Actor**
  - Students, instructors or TAs who want to delete their own comments
- **Stakeholders and interests**
  - Students, instructors and TAs who posted a comment before
  - All registered users who are able to see the comment
- **Preconditions**
  - User is authenticated
  - User is registered in this course
  - User is the author of the comment to be deleted
- **Postconditions**
  - the comment is deleted
- **Main Success Scenario**
  1. User navigates to comments view
  2. User clicks the cross button at the upright corner of his/her comment
  3. System prompt for user confirmation using a modal
  4. User clicks the confirm button to proceed this deletion
  5. The comment is removed from database
- **Extensions and Alternative Flows**
  1. The user is not the author of the comment
  2. There is no cross button on the upright corner of the comment
- **Open Issues**
  - None

UC-13. User sets a comment as answer

- **Primary Actor**
  - Instructors
- **Stakeholders and interests**
  - All registered users who are able to see the comment.
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
  - Only instructor and TAs can set a comment as an answer
  - A question has been posted.
  - A comment has been made
- **Postconditions**
  - The comment that was marked as an answer will have a visual indicator to show that it's being marked as an answer.
  - The user who created the comment will gain honor points.
- **Main Success Scenario**
  1. User navigates to the posts view
  2. User enters a post by touching on it
  3. User sees a list of comments
  4. User touches the upvote icon to set a comment as answer
  5. System displays that the comment is marked as an answer.
  6. The honor point of the user who posted this answer has increased by 2.
  7. System will add honor points to the author of the comment.
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-14. User unsets a comment as answer

- **Primary Actor**
  - Instructors
- **Stakeholders and interests**
  - All registered users who are able to see the comment.
- **Preconditions**
  - Only instructor and TAs can unset a comment as an answer.
  - A question has been posted.
  - A comment has been made & already being set as an answer.
- **Postconditions**
  - The comment that was unmarked as an answer will no longer have a visual indicator to show that it's being marked as an answer.
  - The user who created the comment will have their previously awarded honor points removed.
- **Main Success Scenario**
  1. User navigates to the posts view
  2. User enters a post by touching on it
  3. User sees a list of comments
  4. User touches the upvote icon again to unset the comment
  5. System displays that the comment is no longer marked as an answer.
  6. The honor point of the user who posted this answer has decreased by 2.
  7. System will remove the previously awarded honor points from the author of the comment.
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-15. User upvotes a comment

- **Primary Actor**
  - Students and instructor
- **Stakeholders and interests**
  - All registered users who are able to see the comment.
- **Preconditions**
  - A question has been posted.
  - A comment has been made
  - User cannot be the author of the comment being voted
  - User has never voted for the comment before
- **Postconditions**
  - The comment that was upvoted will have a visual indicator to show that the number of votes have increased by 1.
  - The user who created the comment will gain honor points.
- **Main Success Scenario**
  1. User navigates to the posts view
  2. User enters a post by touching on it
  3. User see a list of comments
  4. User touches the upvote icon
  5. System displays that the comment upvote count of upvote have increased by 1.
  6. The honor point of the user who posted this answer has increased by 1
  7. System will add honor points to the author of the comment.
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-16. User removes vote on a comment

- **Primary Actor**
  - Instructors
- **Stakeholders and interests**
  - All registered users who are able to see the answer.
- **Preconditions**
  - A question has been posted.
  - A comment has been made
  - User cannot be the author of the comment being voted
  - User has already voted for the comment.
- **Postconditions**
  - The comment that was upvoted will have a visual indicator to show the number of votes have decreased by 1.
  - The user who created the comment will have their previously gained honor points removed.
- **Main Success Scenario**
  1. User navigates to the posts view
  2. User enters a post by touching on it
  3. User see a list of comments
  4. User touches the highlighted upvote icon again
  5. System displays that the comment upvote count has decreased by 1.
  6. The honor point of the user who posted this answer has decreased by 1.
  7. System will remove the previously awarded honor points from the author of the comment.
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-17. User searches for posts

- **Primary Actor**
  - Students/Instructor wants to view posts
- **Stakeholders and interests**
  - Students/Instructor wants to view posts
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
- **Postconditions**
  - A search result is presented on screen
- **Main Success Scenario**
  1. The click on the “search” button to apply filter to a list of posts
  2. The user then apply filters such as keyword, author, and number of weeks ago
  3. The system then search for posts based on the filters applied
  4. The user gets a list filtered posts as result
- **Extensions and Alternative Flows**

  1.a Server connection lost

  1. User gets an error message

  2.a No post in the database that matches the criteria

  1. No posts will show on screen as a result
- **Open Issues**
  - None

UC-18. User uploads a file

- **Primary Actor**
  - Instructor with readings and assignments to upload
- **Stakeholders and interests**
  - Students with a specific question would directly benefit from a relative note
  - The instructor is signed in via website
  - Some files are posted for specific audiences
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
  - User is an instructor.
- **Postconditions**
  - After the notes is uploaded, it should be ready for everybody in the course to read
- **Main Success Scenario**
  1. The Instructor posts a reading, or assignment.
  2. The Instructor selects if he/she wants to send an email notification to the class about the new uploaded file.
  3. The system check if the file is oversized, in unreadable format or uploaded unsuccessfully.
  4. File is NOT oversized, the file will be successfully uploaded to the file section.
  5. All the users are able to access the file.
  6. If the Instructor selected to send email, all the students in this course will receive an email notification of this newly uploaded file.
- **Extensions and Alternative Flows**

  4.a If file is oversized
    1. Systems will show a warning message.
    2. User acknowledges by tapping the “confirm” button on error modal.
    3. System bring user back to the file upload view.
- **Open Issues**
  - None

UC-19. User reads a file

- **Primary Actor**
  - All users registered in a certain course
- **Stakeholders and interests**
  - Students read the files uploaded by instructors to solve questions or work on assignments.
- **Preconditions**
  - User is authenticated.
  - User is registered in this course.
- **Postconditions**
  - None
- **Main Success Scenario**
  1. User go to the clicks on “uploaded files” button.
  2. System displays a list of upload files associated to the course.
  3. User selects the file to view.
  4. System display the content of this file.
  5. User goes back to the previous view and system closes the view of this file.
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-20. User deletes an uploaded file

- **Primary Actor**
  - Instructor who uploaded the file
- **Stakeholders and interests**
  - Instructor who uploaded the file no longer wants to share the file
  - The instructor is signed in via website
  - Some files are posted for specific audiences
- **Preconditions**
  - File was successfully uploaded
  - File was uploaded by the user.
  - User is authenticated.
  - User is registered in this course.
  - User is an instructor.
- **Postconditions**
  - All traces of this file is deleted from the file system
- **Main Success Scenario**
  1. Instructor who uploaded the file click the button to delete the file.
  2. System displays a dialog window to confirm deleting the file.
  3. User clicks on YES.
  4. Systems deletes the uploaded file and inform the user using a modal.
- **Extensions and Alternative Flows**

  3.a User clicks NO in the dialog window

  1. System go back to the previous window
- **Open Issues**
  - None

UC-21. User views honor points leaderboard

- **Primary Actor**
  - Students with questions
- **Stakeholders and interests**
  - User who is taking this course currently
- **Preconditions**
  - User is authenticated
  - User is registered in the course
- **Postconditions**
  - The user will be able to contact other users on the list via email
- **Main Success Scenario**
  1. User navigates to his/her list of course
  2. User taps the “users” icon
  3. A list of users registered within the course will be displayed on user’s screen
- **Extensions and Alternative Flows**
  2. System cannot find the course in the database

  2.a System prompts a error message
- **Open Issues**
  - None

UC-22. User views another user’s profile

- **Primary Actor**
  - Students with questions
- **Stakeholders and interests**
  - User who is taking this course currently
- **Preconditions**
  - User is authenticated
  - User is registered in the course
- **Postconditions**
  - The user will be able to view another user’s past records
- **Main Success Scenario**
  1. User navigates to the honor system list
  2. User taps the “profile” icon next to the a user’s profile picture and name
  3. System displays the selected user’s past posts and comments
- **Extensions and Alternative Flows**
  - None
- **Open Issues**
  - None

UC-23. User contacts other user for help

- **Primary Actor**
  - Students with questions and students with solution
- **Stakeholders and interests**
  - Students with questions and students with solution
- **Preconditions**
  - User is authenticated
  - User's email is provided
- **Postconditions**
  - User either respond to email or ignore email
- **Main Success Scenario**
  1. User navigates to the honor system list to find another user
  2. User taps another user's profile picture in the user row
  3. System redirects user to email section
  4. User enter texts and touch the "send button"
  5. The receiver reads the email
- **Extensions and Alternative Flows**

  1.a User with answer turned down the request
- **Open Issues**
  - None

6. Non-functional Requirements

6.1 Performance Requirements

- API response time to be within 0.5s.

6.2 Safety Requirements

- To ensure a real-name system, changing names will not be allowed.

6.3 Security Requirements

- API calls/requests between the frontend clients & the application's backend server will only be done over a secured HTTPS connection.
- Application will use JSON Web Tokens to implement Token-based authentication, removing the need to store sensitive user information (such as passwords) on the server's database.
- The secret key used for signing our application's JSON Web Token (JWT) will be kept within the server so that the token cannot be easily decrypted by third parties. (Chen)
- Only after gaining permission from the user, the application will have access to the user's id, name, profile image, and email address associated to the facebook account that was being used to sign up.

6.4 Legal Requirements

- Respect the intellectual properties of content creator.
- Respect UBC's code of conduct.

6.5 Software Quality Attributes

- Pass App Store's validation

7. Design Changes and Rationale

- To make to make Cyann easier for instructors to access, we have decided to make a separate website since it's hard to implement file upload/deleting functionalities on mobile platform.
- The website would provide the same functionalities the mobile app provides with added use case for uploading/deleting files and course creation.
- The corresponding use cases are UC-3, UC-18, UC-19, UC-20. Due to time and security concern, the tutoring platform we planned is replaced by use case UC-23.
- Users can now directly contact other user in the same class via email and ask specific questions, enabling better communication between classmates.