

Critika

CS 30700: Software Engineering 1
Design Document

Team 8:

Brandon Sung
Connor Todd
Cyrus Santiago
Keith Tan
Ken Sodetz
Julien San Diego

Sprint Overview:

In the last two sprints, we were able to have the backend and front-end functioning. The main goal of the third sprint is to integrate the frontend and backend components and make a functional build that can fulfill the expected user stories. As of right now, we completely understand how to integrate front-end and back-end functions, which mean that integration will be substantially easier from here on out.

Scrum Master:

Keith Tan

Meeting Plan:

Tuesdays/Thursdays (6:00 PM-7:30 PM)

Risks and Challenges:

The first and most important task for us to complete will be the integration of the frontend and backend. Currently, we have quite a bit from sprint 2 that has not been integrated yet, but given that we have integrated nearly all of sprint 1 items, integrating sprint 2 tasks should be relatively straightforward. While absolute functionality is the number one priority, a close second will be the final polish and feel of the application, which will come from lots of front-end and back-end testing, edge casing, and general front-end organization, as the current system is fairly unorganized on the UI side.

User Story 1: As an Admin I would like to be able to see all registered users (**14 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create UI table to show all users	2	Frontend	Brandon
2	Create search functionality for finding a specific user	3	Frontend	Brandon
3	Create service to route guard admin routes if the user isn't an admin	4	Frontend	Brandon
4	Implement pagination on user table	4	Frontend	Brandon
5	Unit Testing	1	Frontend	Brandon

- Given that an admin has logged into the secure admin page, an admin should be able to see a table of all current users.
- Given that an admin has logged into the secure admin page, an admin should be able to search through all the user for a specific user.
- Given that an admin has logged into the admin page, an admin should be able to click on any user and be directed to that user's profile.

User Story 2: As an Admin I would like to be able to ban users (**11 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create options for banning and unbanning users	2	Frontend	Brandon
2	Create service function to send ban user request to backend	2	Frontend	Brandon
3	Reflect ban user change in database and frontend	3	Frontend/Backend	Brandon
3	Unit Testing	4	Frontend	Brandon

- Given that an admin has logged into the admin page, an admin should be able to ban any user.
- Given that an admin has logged into the admin page, an admin should be able to unban any banned user.

- Given that an admin has logged into the admin page, and admin should be able to see all users' ban status

User Story 3: As an admin I would like to approve and deny certain category suggestions (17 hours)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create UI table to show all suggestions	3	Frontend	Brandon
2	Create options to approve and deny suggestions	1	Frontend	Brandon
3	Create service functions to retrieve, approve and deny suggestion	3	Frontend	Brandon
4	Create suggestion model	1	Frontend	Brandon
5	Create suggestion service	2	Frontend	Brandon
6	Implement suggestion routes to backend	3	Backend	Brandon
7	Front end tests	4	Frontend	Brandon

- Given that an Admin has logged into the admin page, an Admin should be able to see all suggestions for a new category
- Given that an Admin has logged into the admin page, an Admin should be able to approve or deny any suggestion
- Given that an Admin approved/denied a submission, it should be reflected in the database

User Story 4: As a user, I would like to be able to view and edit my profile. **(16 hours)**

#	Task Description	Estimated Time (Hours)	Team	Owner

1	Create a card to display user's Critika coins	1	Frontend	Julien
2	Create a card to display user's Critika rating	1	Frontend	Julien
3	Create a card to display when a user first joined	1	Frontend	Julien
4	Create a card to display user's About Me	1	Frontend	Julien
5	Create a card to display user's total number of submissions	1	Frontend	Julien
6	Allow a user to upload and crop a new profile picture	1	Frontend	Julien
7	Allow a user to edit his or her About Me	1	Frontend	Julien
8	Save edited information as an object for use to call backend API	1	Backend	Julien
9	Push edited information to backend to edit user profile in database	1	Backend	Julien
10	Create feedback service	3	Frontend	Julien
11	Create feedback model in frontend	1	Frontend	Julien
12	Integration	3	Frontend/Backend	Julien

- Given that a user chooses to edit his or her profile picture, the old profile picture should be updated to the new uploaded profile picture
- Given that a user provides feedbacks, a user is able to view his or her Critika ratings.
- Given that a user is provides feedback or files a new submission, the Critika coin value will change in response.
- Given that a user is on his or her profile,he or she is able to view his or her starting date.
- Given that a user is able to edit his or her About Me, the About Me section will be updated for the user and other users to see.

User Story 5: As a user, I would like to view/edit my account (**16 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Implement viewing account information on the frontend	2	Frontend	Ken
2	Save edited information as an object for use to call backend API	2	Frontend	Ken
3	Push edited information to backend to edit user account in database	2	Frontend	Ken
4	Implement account model	1	Frontend	Ken
5	Implement account service	2	Frontend	Ken
6	Implement email notification on account information change	2	Backend	Ken
7	Unit Testing	2	Frontend	Ken
8	Implement alerts for change submit/error	3	Frontend	Ken

- Given that a user has edited their account information, an email should be sent to them notifying them of the change
- Given that a user hits submit to change their account information, the changes should reflect on the backend
- Given that a user doesn't submit correct information for editing an account, an alert should show signifying that

User Story 6: As a user, I would like to be able to view all the feedbacks I have received **(14 hours)**

#	Task Description	Estimated Time (Hours)	Team	Owner
---	------------------	------------------------	------	-------

1	Create a UI that displays all the feedbacks given	2	Frontend	Julien
2	Retrieve all submissions with feedbacks from the backend	2	Backend	Julien
3	Create a UI that allows user to rate a feedback	1	Frontend	Julien
4	Create feedback service	3	Frontend	Julien
5	Create feedback model in frontend	1	Frontend	Julien
6	Integration (Displaying)	3	Frontend/ Backend	Julien
7	Unit tests	2	Frontend/ Backend	Julien

- Given that a user received feedbacks, a user should be able to view all the feedbacks he or she has received
- Given that a user received feedbacks, the frontend will make backend calls to populate the UI to display the feedbacks.
- Given that a user received feedbacks, a user is able to rate the feedback he or she is given.

User Story 7: As a user, I would to be able to view, edit, and add submissions **(20 hrs)**

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Integrate add submissions service to submissions tab	2	Frontend	Cyrus

2	Create search functionality in the add submission form to look for a category	4	Frontend	Cyrus
3	Create a dropdown to allow users to choose a specific tag under a specific chosen category	2	Frontend	Cyrus
4	Create service functions for finding categories	2	Frontend	Cyrus
5	Refactor component to retrieve information on render	2	Frontend	Cyrus
6	Integrate Critika coin submission for a specific submission	2	Frontend	Cyrus
7	Create service function for tagging a submission as available	1	Frontend	Cyrus
8	Edit the 'Edit Submission' form to include more robust information	1	Frontend	Cyrus
9	Refactor edit submission service for consistency on request	2	Frontend	Cyrus
10	Add alerts on successful/error adds/edits	2	Frontend	Cyrus

- Given that a user has submitted a submission, a user should be able to view all submissions.
- Given that a user has submitted a submission, a user should be able to edit any submissions he or she chooses.
- Given that a user chooses to make a submission, a user should be able to spend his or her Critika coins to file a new submission.

User Story 8: As a user, I would like to have a dashboard that shows submissions I can critique
(16 hours)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Integrate critique creation into the dashboard queue	3	Frontend	Keith
2	Add backend route functionality to feedback service	3	Frontend	Keith
3	Edit feedback model for passing information to backend	1	Frontend	Keith
4	Fix retrieval integration to make API call on component creation render	2	Frontend	Keith
5	Display important information on dashboard	2	Frontend	Keith
6	Change critique parameters to be more robust and reflect on frontend	2	Frontend	Keith
7	Add more returned information on frontend for a submission	1	Frontend	Keith
8	Unit Testing	2	Frontend	Keith

- Given that a user is on the dashboard, they should be able to see submissions for feedback
- Given that a user is on the dashboard, they should be able to give a critique for the submission displayed
- Given that a user visits the dashboard, the content should be retrieved on render

User Story 9: As a user, I would like to be able to login, logout, and register (**14 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Implement logout functionality	3	Frontend	Keith

2	Redirect register page to login page	1	Frontend	Keith
3	Implement alerts on register and login	2	Frontend	Keith
4	Refactor route guard to check for edge casing	1	Frontend	Keith
5	Implement authentication token expiration	3	Frontend	Keith
6	Allow bypass on authentication token check	2	Frontend	Keith
7	Unit Testing	2	Frontend	Keith

- Given that a user's authentication token is still valid, they should bypass the login screen
- Given that a user opts to logout, their token should be remove from their cache
- Given that a user is logged out, they should not be able to utilize any of the routes

User Story 10: As a user, I want to see a leaderboard ranking of users on the website (**16 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Implement filter service for leaderboard table	3	Frontend	Ken

2	Implement route to retrieve users by Critika rating	2	Backend	Ken
3	Create user model	1	Frontend	Ken
4	Create user service for use with leaderboard	3	Frontend	Ken
5	Implement column filtering in the table	3	Frontend	Ken
6	Integrate profile viewing for a specific user	2	Frontend	Ken
7	Unit Testing	2	Frontend	Ken

- Given that a user visits the leaderboard page, they should see the top 25 users ranked by their Critika score
- Given that filtering is implemented correctly, a user should be able to filter the ranking page on the table
- Given that the backend is implemented, a route should exist that returns the top 25 users by Critika score

User Story 11: As a user, I would like to be able to view another user's profile (**7 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create a UI component that displays another user's profile information	2	Frontend	Julien

2	Implement a backend route to get another user's profile information	2	Backend	Julien
3	Create a service function to retrieve other user's profile information	2	Frontend	Julien
4	Unit Testing	1	Frontend	Julien

- Given that a user clicks another user, the user is able to view another user's profile information
- Given that the other user's profile is rendered, the current user is able to view the other user's Critika rating, total number of submissions, registration date, About Me, and profile picture.
- Given that a user clicks another user, the user will be taken to another page to see their information.

User Story 12: As a user, when I spend Critika coins on a submission, it should be placed into an availability queue **(8 hrs)**

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Edit submissions model to include availability	1	Backend	Connor

2	Implement a system for keeping/removing a submission from critique consideration	5	Backend	Connor
3	Unit Testing	2	Backend	Connor

- Given that a user spends Critika coins on a submission, the users submission will be placed into a queue that will be randomly selected to be placed in the Dashboard page
- Given that the backend is implemented, the submission will be placed in a queue on the database
- Given that the backend is implemented, the queue will remove a submission once it has been critiqued.

User Story 13: As a user, I would like to be able to view other user's submission information (**11 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create a UI component that displays another user's submission	2	Frontend	Cyrus

2	Implement a backend route to get another user's submission information	2	Backend	Cyrus
3	Create a service function to retrieve other user's submission	3	Frontend	Cyrus
4	Implement user profile component into user submission component	3	Frontend	Cyrus
5	Unit Testing	1	Frontend	Cyrus

- Given that a user is able to view another user's submission information, the user should be able to see the submission's description.
- Given that a user is able to view another user's submission information, the user should be able to see the submission's user.
- Given that the backend is implemented, it should be possible to retrieve another user's submission information

User Story 14: As a user, I would like to be able to view Critika categories (**17 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create categories tab in frontend	1	Frontend	Connor
2	Add category service function to retrieve three	3	Frontend	Connor

	categories and submissions from the categories			
3	Add filter service function to search for category	3	Frontend	Connor
4	Integrate search into frontend	3	Frontend	Connor
5	Create category component to show category information	3	Frontend	Connor
6	Create category service to retrieve category information	3	Frontend	Connor
7	Unit Testing	1	Frontend	Connor

- Given that the frontend is implemented, a user should be able to view and search all categories in which submissions are submitted.
- Given that the frontend is implemented, a user should be able to view and search all tags under a specified category.
- Given that the backend is implemented, the specified categories page will be populated with all submissions tagged under that specified category and tag.

User Story 15: As a user, I would like to be able to comment on a submission (**8 hours**)

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Generate a edit textbox when Comment is pressed	1	Frontend	Connor
2	Add comment service function to submission service	2	Frontend	Connor

3	Add comment service to add Critika coin to user upon a successful comment	2	Frontend	Connor
4	On add comment, the comment should be shown on the submission page	2	Frontend	Connor
5	Unit Testing	1	Frontend	Connor

- Given that a user views another user's submission, the user should be able to comment on a submission.
- Given that a user comments on another user's submission, the user should be able to gain Critika coins.
- Given that a user gains Critika coins from commenting, the user's Critika coin value is updated on the backend and on the frontend.