

Critika

Team 8 - Sprint 2 Planning Document

Brandon Sung, Cyrus Santiago, Connor Todd

Keith Tan, Ken Sodetz, Julien San Diego

Sprint Overview:

The main goal of the second sprint is to integrate the frontend and backend components and make a functional build that can fulfill the expected user stories. As of now, users can only sign in, edit their profile, and create a submission. However one of the most central features of our product is the ability to critique other user submissions (hence the name). Sprint 1 was building the foundation, and Sprint 2 will focus on integration and user interaction within the application. We plan on fixing the mistakes of Sprint 1 and further the features available in the application beyond the base features. Being able to give feedback, receive and view feedback, and explore communities will propel our application, and those will be the focus of Sprint 2. This will involve adding more routes, adding to the schema and database, and integrating the frontend and backend components.

Scrum Master: Keith Tan

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

Risks and Challenges:

The first and most important task for us to complete will be the integration of the frontend and backend. We unsuccessfully tried to do this in Sprint 1, as we encountered unexpected problems. All of the backend routes interact with the database in the way we want them, and return the correct JSON objects in the request, however parsing these in the frontend for use and making a build has been a challenge. We plan to finish this as soon as possible, so we can expand on our work from Sprint 1, fix errors and add more crucial features to our product.

Current Sprint Details

User Story #1 - 22 hrs

As a user, I would like to be able to login and use my credentials for actions

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Implement frontend authentication	8	Frontend	Keith
2	Implement a listener so we can determine if a user is logged in	4	Frontend	Keith
3	Save authentication and username to cookies	2	Frontend	Keith
4	Automatically assign authentication token to all routes that require it	5	Frontend	Keith
5	Integrate backend route to login using frontend form input	3	Frontend	Keith

Acceptance Criteria:

- ❖ Given that the authentication is implemented correctly, on refresh the user should still be able to use their token for actions
- ❖ Given that route guarding is implemented, when a user tries to access a route that needs authentication
- ❖ Given that the frontend and backend are integrated, a user should be able to log in using the data they input to the login fields

User Story #2 - 20 hrs

As a developer, I would like the frontend and backend to be integrated.

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Integrate Add Submissions Page --	4	Front end	Brandon
2	Integrate Edit Submission Page	4	Front end	Brandon
3	Integrate Login Page ----	3	Front End	Julien
4	Integrate Reset Password	3	Front End	Julien

	Page			
5	Integrate Edit Profile Page	3	Front End	Julien
6	Integrate Register Page ----	3	Front End	Julien

Acceptance Criteria:

- ❖ Given that the submission page is integrated, the user will be able to see the users submissions on his or her submissions page.
- ❖ Given that the edit submission page is integrated, the user will be able to submit his or her submission and have it displayed on the user's submission page.
- ❖ Given that the login page is integrated, the user will be able to see site content after pressing the login button.
- ❖ Given that the reset password page is integrated, the user will be directed back to the login page with their new password.
- ❖ Given that the edit profile page is integrated, the user will be able to save changes to any of the field they modified and be directed back to their profile page.
- ❖ Given that the register page is integrated, the user will be sent to a page where they need to verify their account.

User Story #3 - 14.5hrs

As a user, I would like to be able to critique and give feedback to other users' submissions

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Add a critique button to user submissions and Generate a "Critique Submission" screen	0.5	Frontend	Julien
2	Include UI that displays the submission name, thumbnail, and type of feedback requested, along with a form to input text to critique the submission	1	Frontend	Julien
3	Update notifications screen of user whose submission was critiqued	1	Frontend	Julien
4	Give 1 Critika Coin to user who provided feedback	1	Backend	Keith
5	Create a backend route to	3	Backend	Ken

	handle the given critiques and add to the database			
6	Create Unit tests for Backend	2	Backend	Connor
7	Create tests for frontend	2	Frontend	Julien
8	Integrate frontend and backend	4	Frontend/Backend	Keith

Acceptance Criteria:

- ❖ Given that a user has selected to critique a submission, a page with a form to critique the submission should be displayed
- ❖ Given that a user has completed the form and clicked the submit button, a notification should be sent to the user whose submission was critiqued. Additionally, a single Critika Coin should be awarded to the user who critiqued the submission
- ❖ Given that a user critiques a submission, the backend should appropriately handle that action and save the critique to the database

User Story #4 - 13.5 hrs

As a user, I would like to be able to rate the feedback I've been given.

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Generate a "Rate Feedback" screen	1	Frontend	Julien
2	Include form to rate feedback on scale of 1-5 and to include text input fields for comments/rationale	1	Frontend	Julien
3	Update notifications screen of user whose feedback was rated	1	Frontend	Julien
4	Adjust critika score on profile page of user whose feedback was rated	1	Frontend	Julien
5	Create a route to handle the rating for the particular piece of feedback and update the rating in the database	2.5	Backend	Keith
6	Create Unit tests for	2	Backend	Connor

	Backend			
7	Create tests for frontend	2	Frontend	Julien
8	Integrate frontend and backend	3	Frontend/Backend	Keith

Acceptance Criteria:

- ❖ Given that a user has selected to rate a piece of feedback, a page with a form to rate the feedback should be displayed
- ❖ Given that a user has completed the form and clicked the submit button, a notification should be sent to the giver of feedback containing their rating and any comments/rationale. Additionally, their Critika score should be updated on their profile page.
- ❖ Given that a user has rated a particular piece of feedback, the backend should appropriately handle that action and save the rating to the database under the appropriate critique.

User Story #5 - 15.5 hrs

As a user, I would like to be able to view and utilize my Critika Coins

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Add Critika Coin counter with icon that persists on Navbar	1	Frontend	Brandon
2	Add Critika Coin counter to profile screen	1	Frontend	Brandon
3	Add option to spend critika coin on user's current submissions	1	Frontend	Brandon
4	Add option to spend critika coin on newly uploaded submissions on the user's "create submissions" page	1	Frontend	Brandon
5	Initialize 5 coins upon account creation	.5--	Backend	Keith
6	Create route that handles the accumulation of coins and reflect the change in the database	2--	Backend	Keith
7	Create route that handles the spending of coins and	2--	Backend	Keith

	reflect the change in the database			
8	Create Unit tests for Backend	2	Backend	Keith
9	Create tests for frontend	2	Frontend	Brandon
10	Integrate frontend and backend	3	Frontend/Backend	Brandon

Acceptance Criteria:

- ❖ Given that a user is anywhere on the Critika site, a count of their critika coins should be displayed on the navbar
- ❖ Given that a user has used a critika coin on their submission, that submission should populate the queues to be critiqued by other users.
- ❖ Given that a user has successfully created an account, 5 coins should be made available for the user to spend
- ❖ Given that a user has spent a coin, the backend should be equipped to handle that change in the database
- ❖ Given that a user has earned a coin, the backend should be equipped to handle that change in the database

User Story #6 - 7 hrs

As a user, I would like to be able to easily view all communities

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create page for viewing all communities in more detail	2	Front End	Cyrus
2	Generate thumbnails for each category	1	Front End	Cyrus
3	Include short description of each category	1	Front End	Cyrus
4	Make clickable link for each category	1	Front End	Cyrus
6	Create tests for frontend	2	Frontend	Cyrus

Acceptance Criteria:

- ❖ Given that a user has selected categories in explore, they should be linked to a page to view all communities in more detail.

- ❖ Given that a user is on the communities page, they should be able to see a thumbnail and short description of each communities
- ❖ Given that a user has clicked a certain category, they should be directed to that specified page

User Story #7 - 13.5hrs

As a user, I would like to be able to easily see submissions from other users I can give feedback to.

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create UI carousels on dashboard page to display queue of user submissions	1	Frontend	Brandon
2	Create UI carousels on categories and explore screens to display trending or popular user submissions	1	Frontend	Brandon
3	Design algorithm for deciding who's queues a user's submission should be displayed on if they spent a Critika Coin	1.5	Backend	Connor
4	Create a backend route to pull submissions for display	2	Backend	Keith
5	Create unit tests for backend	2	Backend	Connor
6	Create tests for frontend	3	Frontend	Ken
7	Integrate Frontend and Backend	3	Frontend/Backend	Keith

Acceptance Criteria:

- ❖ Given that a user is on their dashboard, a queue of other user submissions to critique should be displayed
- ❖ Given that a user is on the categories or explore pages, a queue of trending or popular submissions should be displayed
- ❖ Given that a user has navigated to the page, the page should populate with submissions available for critiquing.

User Story #8 - 14.5 hrs

As a user, I would like to be able to report bad feedback.

#	Task Description	Estimated Time	Team	Owner
---	------------------	----------------	------	-------

		(Hours)		
1	Generate a "Report feedback" screen	0.5	Frontend	Julien
2	Include form to specify reason for report via generic dropdown options along with text input field to further specify rationale for report	1	Frontend	Julien
3	Update notifications screen of user whose feedback was reported	1	Frontend	Julien
4	Update notifications screen of admins to alert them of the report	1	Frontend	Julien
5	Update reports screen for admins to display the reported feedback	1	Frontend	Julien
6	Create a route to report poor feedback and mark the feedback for review in the backend	2	Backend	Ken
7	Create Unit tests for Backend	2	Backend	Connor
8	Create tests for frontend	2	Frontend	Julien
9	Integrate frontend and backend	4	Frontend/Backend	Ken

Acceptance Criteria:

- ❖ Given that a user has selected to report a piece of feedback, a page with a form to report the feedback should be displayed
- ❖ Given that a user has completed the form and clicked the submit button, a notification should be sent to the giver of feedback as well as the admins, alerting them of the report. Additionally, the report should appear on the reports page, viewable by admins
- ❖ Given that the piece of feedback has been selected for being reported, the backend should mark the feedback in question for review in the backend.

User Story #9 - 13.5 hrs

As a user, I would like to be able to report bad comments.

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Generate a "Report	0.5	Frontend	Julien

	comment" screen			
2	Include form to specify reason for report via generic dropdown options along with text input field to further specify rationale for report	1	Frontend	Julien
3	Update notifications screen of user whose comment was reported	1	Frontend	Julien
4	Update notifications screen of admins to alert them of the report	1	Frontend	Julien
5	Update reports screen for admins to display the reported comment	1	Frontend	Julien
6	Create a route to report poor comments and mark the comment for review in the backend	2	Backend	Ken
7	Create Unit tests for Backend	2	Backend	Connor
8	Create tests for frontend	2	Frontend	Ken
9	Integrate frontend and backend	3	Frontend/Backend	Julien

Acceptance Criteria:

- ❖ Given that a user has selected to report a comment, a page with a form to report the comment should be displayed
- ❖ Given that a user has completed the form and clicked the submit button, a notification should be sent to the commenter as well as the admins, alerting them of the report. Additionally, the report should appear on the reports page, viewable by admins
- ❖ Given that the comment has been selected for being reported, the backend should mark the comment in question for review in the backend.

User Story #10 - 15 hrs

As an admin, I would like to be able to unban users, and make another user an admin.

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create a UI that displays a form for which user to un-ban	3	Frontend	Cyrus

2	Create a UI that displays the users to make an admin	3	Frontend	Cyrus
3	Create route to unban a user	2	Backend	Connor
4	Create route to make another user an admin	3	Backen	Connor
5	Create Unit tests for Backend	2	Backend	Connor
6	Create tests for frontend	2	Frontend	Cyrus
7	Integrate frontend and backend	3	Frontend/Backend	Julien

Acceptance Criteria:

- ❖ Given that a user is banned, an admin should be able to restore their status to good standing.
- ❖ Given that a user is an admin, the user should be able to promote other users to admins.
- ❖ Given that a user is an admin, they should be able to access admin only routes

User Story #11 - 13 hrs

As a user, I would like to be able to create a new community

#	Task Description	Estimated Time (Hours)	Team	Owner
1	On the categories screen a button should be accessible for a user to propose a new category	0.5	Frontend	Cyrus
2	Create a "create community screen"	0.5	Frontend	Cyrus
3	Include a form on propose category screen to indicate the community name, rationale for why category should be included, and a button to submit proposal	1	Frontend	Cyrus
4	Update notifications screen of admin whenever a user submits a proposal	1	Frontend	Cyrus
5	Create a route to submit the	2	Backend	Ken

	new category for review			
6	Create Unit tests for Backend	3	Backend	Ken
7	Create tests for frontend	2	Frontend	Cyrus
8	Integrate frontend and backend	3	Frontend/backend	Ken

Acceptance Criteria:

- ❖ Given that a user is on the categories screen, a button should be accessible to propose a new category
- ❖ Given that a user has clicked the propose category button, a screen should be displayed where a user can write and submit their proposal
- ❖ Given that a user has submitted their proposal, the notifications screen should update for admins

User Story #12 - 11 hrs

As a user, when I'm critiquing, I would like the option to have my information anonymous from the person I'm critiquing

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Include radio button to indicate desire for anonymity on Critique Submission screen	0.5	Frontend	Cyrus
2	Hide user info from notifications screen and feedback received screen	1	Frontend	Cyrus
3	Create a route to switch anonymous boolean for a particular instance of feedback	1.5	Backend	Ken
4	Create unit tests for backend	3	Backend	Ken
5	Create tests for frontend	2	Frontend	Cyrus
6	Integrate frontend and backend	3	Frontend/Backend	Ken

Acceptance Criteria:

- ❖ Given that a user is on the Critique submission screen, an option to remain anonymous should be accessible
- ❖ Given that a user has chosen to be anonymous, their user info should be absent from notifications and the feedback itself
- ❖ Given that a user is an admin, they should still be able to see the user info.
- ❖ Given that the user has selected anonymous button from their client, the database should always reflect that the instance of feedback shall remain anonymous from the user receiving the critique.

User Story #13 - 12.5 hrs

As a user, I would like to be able to comment on users' submissions

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Generate a input field to add a comment on another user's submission	0.5	Frontend	Cyrus
2	Update user's submission to display the comment	1	Frontend	Cyrus
3	Update notifications screen of user whose submission was commented on	1	Frontend	Cyrus
4	Give 0.25 Critika Coins to user who provided comment	1--	Backend	Keith
5	Create a route to add the comment to the particular submission and reflect the change in the database	2--	Backend	Keith
6	Create Unit tests for Backend	2	Backend	Connor
7	Create tests for frontend	2	Frontend	Cyrus
8	Integrate frontend and backend	3	Frontend/Backend	Julien

Acceptance Criteria:

- ❖ Given that a user is viewing another user's submission, a field should be available to comment on it
- ❖ Given that a user has commented on a submission, the comment should be viewable on the submission and a notification should be sent to the user who owns the submission. Additionally, 1/4th Critika Coin should be awarded to the user who commented on the submission

- ❖ Given that a user has submitted a comment for a submission, the comment should be added to the submission object in the database.

User Story #14 - 14.5 hrs

As an admin, I want to be able to ban users

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create "Ban User" screen	0.5	Front End	Brandon
2	Include form that displays user info along with text input fields to specify rationale for banning user	1	Front End	Brandon
3	Remove user and all child objects of user from all screens	2	Front End	Brandon
4	Send notification to user that they have been banned	1	Front End	Brandon
5	Lock site content from user if login is attempted with banned username	1	Front End	Brandon
6	Create an admin specific route to handle the banning of a particular user and reflect that change in the database.	2	Back End	Keith
7	Create unit tests for back end	2	Back End	Connor
8	Create Unit tests for front end	2	Front end	Brandon
9	Integrate front and back end	3	Front/Back	Brandon

Acceptance Criteria:

- ❖ Given that an admin has selected the ban option on the users menu, a "Ban user" screen should be displayed
- ❖ Given that an admin has banned a user, that user and all child objects of that user should no longer be viewable
- ❖ Given that a user has been banned, they should not be able to access site content outside of the login screen

User Story #15 - 14 hrs

As a user, I'd like to be able to see submissions for a given community.

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Create UI carousels on the Explore page to display the user submissions of all catecommunitygories.	3	Frontend	Cyrus
2	Create UI carousels on the Explore page to display the user submissions of a specified community.	2	Frontend	Cyrus
3	Create a route to pull all submissions from a given category	2	Frontend	Keith
4	Create Unit tests for front end	2	Frontend	Cyrus
5	Create unit tests for back end	2	Backend	Connor
6	Integrate front and back end	3	Frontend/Backen d	Julien

Acceptance Criteria:

- ❖ Given that a user navigates to the Explore page, the user will be able to view all the submissions of all categories on a carousel.
- ❖ Given that the a user specifies which category he or she want to view, the user will be able to view all the submission on a carousel for the specified category.
- ❖ Given that the frontend and backend are integrated successfully, the Explore page will be populated with the submissions for each category.

User Story #16 - 15 hrs

As a developer and as a user, I would like the application bug free as possible

#	Task Description	Estimated Time (Hours)	Team	Owner
1	Fix bugs from Sprint 1	2		Connor
2	Implement more verbose edge casing for passwords / fix password hashing bug	6		Ken
3	Implement more verbose edge casing for emails	4		Ken

4	Expand on existing unit tests from sprint 1	2		Connor
5	Refactor unit tests from sprint 1	1		Connor

Acceptance Criteria:

- ❖ Given that the tests have been implemented correctly, we should have coverage of nearly all possible cases
- ❖ Given that the user changes their password, the new password is updated, hashed, and stored correctly.
- ❖ Given that the user has a valid email, an email should be sent correctly to the user