

CS 491 - Jeremiah Webb

Engineering Notebook

1/17/2023 - Initial Project Vision and Issue Tracking

- **Component:** Project Vision and Issue Tracking System
- **Problem Solved:** Established the product vision in a group meeting and decided on adopting Github for improved issue tracking.
- **Documentation:** Created initial documentation on Github to outline the project vision and issue tracking methodology.

1/20/2023 - API Authorization Method Implementation

- **Component:** API Cognito Authorization
- **Problem Solved:** Successfully implemented the Cognito Authorization method for the API and figured out its application in JavaScript.
- **Documentation:** Documented the authorization method and its implementation process in JavaScript.
- See Github Issues: [AddCognitoIssue](#)

1/23/2023 - Meeting Schedule Establishment

- **Component:** Project Management
- **Problem Solved:** Set a weekly meeting schedule to ensure regular team collaboration and progress checks.
- **Documentation:** Updated the project management documentation to reflect the meeting schedule.

1/29/2023 - Filtering Code Development

- **Component:** Data Filtering System
- **Problem Solved:** After struggling with Rust, switched to Python and efficiently completed the new filtering code within 4 hours.
- **Documentation:** Documented the filtering system's development process and its implementation using the Boto3 library in fetchDashboard.py.

1/30/2023 - Recommendation System Discussion

- **Component:** Recommendation System
- **Problem Solved:** Engaged in a productive discussion on designing the recommendation system, focusing on the weighting of user inputs.
- **Documentation:** Drafted notes on potential approaches to the recommendation logic and user input weighting.

2/1-2/2023 - Pagination Implementation for Frontend

- **Component:** Frontend Pagination
- **Problem Solved:** Addressed challenges with React's useEffect methods and JS error handling to implement data pagination on the frontend.
- **Documentation:** Documented the pagination implementation process, highlighting solutions to encountered challenges.

- See Github Issues: [Dashboard Cleanup for loading Rows for dashboard](#), [Dashboard Pagination](#), [API Update for Pagination](#),

2/3-4/2023 - CI/CD Setup

- **Component:** Continuous Integration/Continuous Deployment (CI/CD)
- **Problem Solved:** Overcame issues with AWS Amplify and Github repository organization to establish a CI/CD pipeline.
- **Documentation:** Detailed the CI/CD setup process, including troubleshooting and solutions.

2/5/2023 - CORS Issue Resolution and Productive Coding Session

- **Component:** API Gateway and Lambda Functions
- **Problem Solved:** Resolved CORS request issues for the production website by specifying return headers for API gateway interactions with lambda functions.
- **Documentation:** Documented the solution to CORS issues and the progress made during the coding session.

2/8/2023 - Github Issue Management

- **Component:** Project Management
- **Problem Solved:** Created new Github issues for ongoing project tasks and noted the need for notebook organization.
- **Documentation:** Updated project management documentation with new issues and organizational tasks.

2/14/2023 - Tool Upload Page Development

- **Component:** Tool Upload Web Page
- **Problem Solved:** Began developing the tool upload page and sought assistance with pagination on the tool dashboard. Finished!
- **Documentation:** Documented the development process of the tool upload page and collaboration efforts.
- See github issues: [UploadToolMechanism](#), [UploadToolWebPage](#), [EditAPIUpload](#),

2/16/2023 - Recommendation Algorithm and Tool Data Management

- **Component:** Recommendation Algorithm and Tool Data
- **Problem Solved:** Discussed future tasks with Troy, focusing on improving the tool data management and enhancing the recommendation algorithm.
- **Documentation:** Outlined plans and approaches for enhancing the recommendation algorithm and managing tool data.

2/17/2023 - Add Navbar

- **Component:** Frontend & Webpages
- **Problem:** Needed ability to easily call a way to go to new pages. Since all pages need a way to get to the other web pages, this was just easier to containerize into a single react component.
- See github issue: [link](#)

2/19/2023 - Handle Approval & Denial of a tool from Frontend

- **Component:** Frontend, Upload tool Webpage
- **Problem:** Admins needed a way to enable approval or denial of user submitted tools, this needed the api methods on API gateway and needed python methods for it.
- See github issue: [link](#)

2/20/2023 - Admins can Upload a Tool via website.

- **Component:** Frontend, physical upload tool webpage form section
- **Problem:** With the ability to approval a tool now programmatically we need to be able to upload a tool's data. These uploaded tools will be shown on the pending tool dashboard. Worked on creating python method to validate submissions then store into dynamodb.
- See github issue: [link](#)

2/23/2023 - View Single tool data

- **Component:** Frontend, Tool Dashboard specifically
- **Problem:** Users can view tools in a dashboard format; however, they cannot currently view a singular tool's data itself. To solve this, a new react component must be made to work with the table's data. This React component takes in the json object of the tool and creates a popup window that the user can use to view and scroll through all the fields of a tool.
- See github issue: [link](#)

2/29/2023 - Updated Questionnaire Output to sync with Troy

- **Component:** Frontend to Backend connection
- **Problem:** To properly serialize with Troy's Rust code for the API to begin report generation, I needed to update the output of the questionnaire to distinct data types and variable headers for the json in the http request.
- See github issue: [link](#)

3/6/2023 - Py script to Upload tools programmatically

- Component: Tool Upload Utilities
- Problem: The team needed a way to get our research uploaded to DynamoDB via a csv. This wasn't too difficult to achieve, essentially load the csv via Pandas. See `lambda/InputToolsIntoDynamoDb/input.py` in repo.
- See github issue: [link](#)

3/24-26/2023 - Merge Filter with Sarah

- Component: Filter for Dashboard, dashboard webpage on frontend
- Problem: Sarah finished her filter and needed help to properly merge the filter with the current dashboard. She began the filter at the end of the last sprint. Overall, the filter works! The pagination needs to be fixed but that's it.
- See github issue: [link](#)