Progress Report

- Increment 2 -

Group #3

1) **Team Members**

David Markel, dm20gg, david-markel

Victor Marques, vfm20e, VFM20E

Dylan Gire, dog18, dog2943

Tommy Chong, tc18cj, toch24

Abraham Beltran, ab19bb, abrahambeltran

2) **Project Title and Description**

DataTweet, a web app for sharing visualized data similar to Twitter where users can post any data they may want to share like research results with visualization options. It is using React for the front end, Django for the backend, and sqlite for our database.

3) **Accomplishments and overall project status during this increment**

For posting, the user is now able to upload a csv file and select its x and y axis and preview how the graph will look like before posting. The file is being uploaded to our database which we are storing in an AWS S3 bucket.

For displaying the graphs for all posts, we might not need to upload the file but just the data in the x and y axis. Implemented error checking such as a user won't be able to post if it's not logged in or there is no file being uploaded or if the file is not csv format.

For the search it is functional with JSON data and can be changed to search based on content, author, title, or whichever way we need to search.

Links to share posts across social media have been added but need work to further the information passed on from posts on DataTweet to any other network.

Users can now edit their credentials, login, and view their own posts.

4) **Challenges, changes in the plan and scope of the project and things that went wrong during this increment**

It was a real challenge to figure out how to display the graph when a file was being uploaded. At first we planned to use the CanvasJS React component to display the graphs but it was getting too complicated so we switched to another library to display the graphs which is HighCharts. We didn't realize we had to choose an x and y axis to display the graph, so implementing a dynamic way to show the labels from the csv was complicated.

Instancing and distributing data for posts might prove to be complicated based on our current structure but more testing is needed to check. The follow button will most likely not be implemented due to time constraints but the like/dislike and comments should be.

The search functionality took a lot of iterations to get it to something that was okay for the time being. Moving it to its separate page helped but still having troubles making it fully functional to look up specific posts and profiles to pull up. Right now it can only search through a sample json file of mock data but will be looking to make it fully functional and integrated.

The data encrypt files that were in increment 1 have been removed due to information regarding Django's built in encryption. The built-in encryption provided by the Django_cryptography module created issues when pulling users and posts up. Work on migrating the database and continued encryption in increment 3.

Linking our login user table with the useAuth0 login system got weird with the timing of hooks in react.

5) Team Member Contribution for this increment

- **Tommy Chong:**

A. Added to part 3 about my contributions for this increment. Added to part 4 about the challenges I encountered for this project and my solutions.
B. Changed the description for #8 in part 2 from increment 1.
C. Added a library to the frontend in part 2 and 3 other libraries in the backend for part 2 as well.
D. Created an AWS S3 bucket to host our files being uploaded. Implemented the file uploading functionality by connecting the AWS S3 bucket to our database in the backend and getting the data in the frontend by creating a file input. Implemented HighCharts to the home page when a file is being uploaded. Users can choose the x and y axis and preview how the graph will look once posted.
E. Added my contributions (parsing the csv file and previewing the graph before posting) to the video demo.

- **Dylan Gire:**

A. Added to part 3 about the integration of the search and where it is at this moment. Added to part 4 about the challenges of finishing the search tab that I started in increment 1 and where I foresee it landing in the future.
B. Updated the description for #7 in part 2 with more specific goals and vision on how it will integrate.
C. No changes from increment 1.
D. Revamped the search.jsx file completely to make it functional and configurable with data sets. Added the search.css file to change the aesthetic and make it more user friendly. Added these changes to the navbar.jsx and app.js files.
E. Making/Editing video handled by Victor.

- **Victor Marques:**

A. Added to part 3 on my addon of social links for the posts on the feed of DataTweet. Added to part 4 on the challenges with the encryption provided by django.
B. Added to part 2 in the libraries for the backend used.
C. Added to part 7 for the requirements
D. Backend: No encryption added as Django takes care of it. Frontend: added social media sharing links within the post to be shared
E. Making/Editing Video

- **Abraham Beltran:**

A. Fixed some formatting in 3-6, added to part 4, 5, and 6 about my commits during this increment
B. Fixed some formatting and typos, the document is basically set in stone.

C. No changes from first increment
D. Fixed css styling to be more comprehensible, fixed graph preview for posts, fixed file naming/implemented templating for future use. Eliminated unused files like the about page. Added rudimentary like and dislike buttons.
E. Added my contributions to the project for Victor

- **David Markel**

A. Added create_user api
B. added get_users api
C. added log in functionality
D. added update_user api
E. enhanced basic formatting/css

6) Plans for the next increment

- Tommy: Display the data as graphs for each post. Graphs will be dynamic, meaning it won't be just an image. Will have to store the x and y axis data in the database for each post.
- Dylan Gire: Add functionality to bring up other profile pages and to be able to search by post or profile. Make a page for a specific post and make the connection between the components complete.
- Abraham Beltran: Work on graph displays, fix functionality, refactor code, likes and dislikes persistent storage and full functionality. Add the ability to follow other users if time allows.
- Victor Marques: Add back encryption and decryption for the users class. Will find a better way to migrate data to another social media site to allow others to share more easily. I will continue searching for bugs in the site that can be fixed within time constraints.
- David Markel: Finish CRUD for both users and posts, right now we can only create, read the users and posts, as well as update the users. Need to add delete and update. After that just overall clean up the site to make it as official looking as possible. If time, add visualization tool

7) Link to video

(AUDIO WARNING UNTIL 0:16): here