Project Report

Installation, initialization/running database, and server

- Open a terminal in the directory of the server.js file
- How to install:
    - To begin, install all the necessary dependencies by using the following command:
        - > npm install
    - Then you need to start the database. To do this, run the following command:
        - > mongod --dbpath='data/db'
    - Once the database is launched, you will need to initialize it. To do this, open a new terminal in the same directory and run the following command:
        - > node .\database-initializer.js
        - You might see some warnings, but this is not because of the code, It is simply because of the importing of modules.
    - Once that is finished you will see that the database connection is closed.
    - Now we can run the server. Use the following command:
        - > node .\server.js
        - Same thing applies for the warnings
    - Now head to http://localhost:3000

Discussion and Critique

- For the back-end code, I believe it is well organized because I divided the routes into different routers and imported them into the main server.js file. This improves clarity and organization when analyzing the code.
- I effectively used the RESTful design principles because I have self-descriptive messages and follow the formal meaning of the HTTP verbs. It also follows a uniform interface which make the API easier to use and interact with.
- I have productively made the use of asynchronous operations when retrieving and updating the database.
- For the bonus points, I implemented a workshop page which displays the title, description, and users enrolled. From this page you can also enroll.
    - FYI: I made it so an artist can enroll in their own workshop.
- Some possible improvements are for the front-end code. I currently only have one file that handles all the client-side JavaScript. To improve readability and organization, I should have divided the client-side JavaScript into their respective files.

Design Decisions/Extra Functionality

- When following, liking, and enrolling. You cannot do the same thing twice. What I mean by this is if you follow an artist, you can only follow them once. If you like an artwork, you can only like it once. If you enroll into a workshop, then you cannot enroll again.
- When creating artwork/workshop the name needs to be unique and all the fields need to be filled or you will be prompted with an alert.
- You can delete your own reviews, unlike artworks, and unfollow artists.
- For the notifications, you need to click on the notifications tab to view all your notifications. (Sorted by oldest to newest, oldest on top)
- If you want to head to the search artwork page, you can click on RexArt (the Logo) to go to the search page.
- If you want to see all artwork, click on the artworks tab.
- When looking at a someone else's review on a post, you can click on their username to go to their profile.
- For the artist profile, I decided not to show the reviews that the artist created. The reason for this is because I like the clean look of the artist profile that I designed. You can still view other artist reviews under the post of the artwork. (Shown in video)
- As mentioned before, I have completed one of the bonus implementations:
    - "For the bonus points, I implemented a workshop page which displays the title, description, and users enrolled. From this page you can also enroll."

YouTube Link:

https://www.youtube.com/watch?v=g2snvb-dg7U