Project Milestone 7

The Project: The NoteSquad

The Team: StackOverflowers

Grace Horton
Mark Davis
Sean Fitze
Ethan Bauer

Description

NoteSquad.com is a local student webpage for other students to exchange resources, notes, and find notetakers. The intention of this webpage is to make it easier for students, especially long-distance/remote learning students, to find notetakers and study material, all in one place.

The students can make a profile which contains their contact information, major, and a list of their current classes. They can upload their own notes to help other students as well. The search feature makes it easier for students to find what they're looking for quickly, whether it be a notetaker, note taking jobs, or study material relevant to their current courses and major. They can search by course ID, class name, etc. It is intended to be sort of a community for all students to work together and get through all of their difficult classes, whatever they may be.

Project Tracker

https://csci-3308-spring21-024-1.atlassian.net/jira/software/projects/S01/boards/1/roadmap
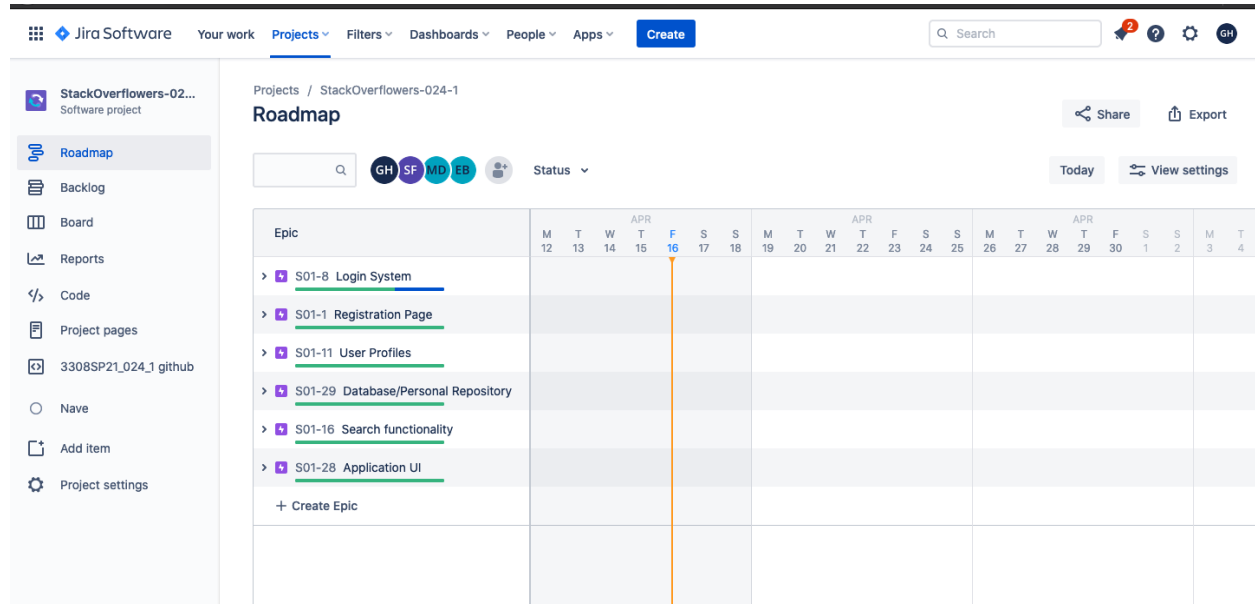
VCS

https://github.com/CSCI-3308-CU-Boulder/3308SP21_024_1

Deployment

To run the NoteSquad app you must have docker installed:
1) Clone the repo.
2) Currently using our project's amazon s3 bucket for storage but will probably be disabled after complete, check the .env file and input your own AWS credentials, create a bucket called "notetakers" and docker will read from this and use your bucket.
3) Navigate in a terminal to the Notesquad_FinalDraft folder in the All project code-components folder.
4) Type the command docker-compose up in your terminal
5) It's ready to use when the terminal input says ...
   [nodemon] starting `node server.js`
   web_1 | http://localhost:3000

web_1 | Seems all green!!

6) To access the login page type **localhost:3000/** into your preferred browser

7) All pages should be accessible from the login.

Testing

1) **Initial Login and Registration:** From the login page, <u>Don't have an account? Click here to sign up!</u> should take you to the registration page. The registration page has 7 fields you should have to fill out before you can **Submit** the form:

   Username, Name, Email, University, Password, Confirm Password, Account Type

   There are indicators at the bottom of the page that should be red when the page loads. They check if your password has the following:

   A lowercase letter

   A capital (uppercase) letter

   A number

   Minimum 8 characters

   Password and Confirm Password Match

   Special Character

   They should turn green when all conditions are met and then the submit button becomes enabled.

2) **User Profiles:** Once you have satisfied all the conditions hit submit you should be able to see your user profile. Your information from the registration page should be visible in the **Profile Info** card at the edge of the screen. The **Tour** button should grey out the page and produce a small box above all the highlighted features explaining them. You can end the tour at any time by pressing *End Tour*. You can edit your Name, Email, and

University sections of your Profile Info section by clicking **Edit Profile.** You can edit one or all of the fields and they will save and display appropriately.

3) **Uploading and Reporting Documents:** To upload a document press the **Upload File** button in the PDF upload block below Profile info. Enter your major and the Course ID. You can use only upper and lowercase letters and numbers. The Course ID cannot be more than 8 characters. If you try to upload a document other than a PDF, you should be redirected to a page that says **The file must be a PDF!** To return to your user profile simply hit the back button. You can view, download, and print your PDF with the **Preview/Download** button. To report it, click **Report**. After reporting a PDF, your page should refresh and the reported PDF should be removed. There is also a delete button that only allows the original uploader, or an admin, to delete the file from the database.

4) **Search:** You can search for notes via the search bar, by typing either the major (i.e. Computer Science, Math, Biology, etc.) or the course code (i.e. CSCI3308, MATH2033, etc.) or even partials. You can use spaces. The search bar is not case sensitive. Once you hit search you should be directed to another page with the search results. If there are no search results, you should still be redirected, the page will simply not contain any notes. To save the notes, click **Save**. That should automatically redirect you to your profile. If you don't want to save any notes, the **Home** button on the Navbar should bring you back to your profile if you're still signed in. If you are not signed in it takes you to the Login page.

5) **Admin Profile:** The login credentials for the admin profile are: admin@email.com password: admin1password Once logged in, you should be able to see all the reported notes in the **Reported Notes** section. You can still preview the notes if you'd like to look at them yourself, or you can delete them by clicking **Delete**. Once they are deleted, they should also be deleted from every user's saved notes.

6) **Logging in and out, and About:** To log out of any profile, click **Logout** on the Navbar. You can log back in with the correct email and password. If you provide incorrect login information, red text should appear just above the NoteSquad copyright saying '*Invalid Credentials*' . The **About** button on the navbar should redirect you to a page with the NoteSquad product description and Application Architecture Overview.

Users Included in the Database:

Admin: admin@email.com   password: admin1password
User1: user1@email.com   password: user1password
User2: user2@email.com   password: user2password
User3: user3@email.com   password: user3password
User4: user4@email.com   password: user4password

Contributions
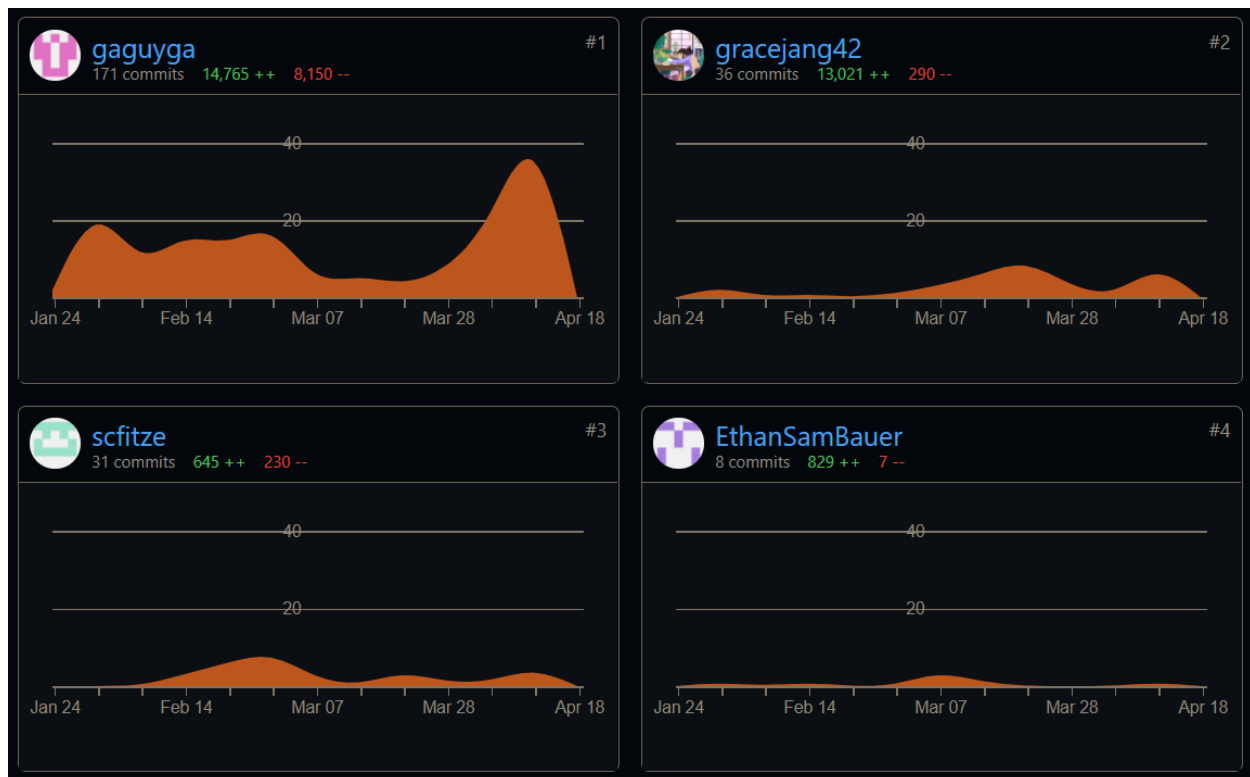
Grace Horton: Provided Wireframes, set-up and helped manage the Jira page, user-stories, Login front-end, Registration 'select account' and submit features, set up AWS account, assisted with upload function, architecture diagram design, added Report button on user profile, Testing, code/file clean-up, Presentation Slides, Demo main, Project Milestone document scribe, organized github for Final Submission, Created/organized Notesquad_FinalDraft folder and readMe

Mark Davis: Setup login sessions locally then saving to database, improved search and route, setup server-side constraints for user input, setup error display

for login and registration pages, made able to delete uploads, upload function and modal, got change profile modal fully working, implemented/setup a tour resource, created an iframe for viewing pdf, created about page and checked all navbar links worked correctly, restricted routes based on login, simplified database, setup s3 bucket to test with, setup docker and env variables, separated html to ejs files, assisted with admin container and report button, templated user ejs for data and errors, simplified profiles to be separate containers instead of tabs, setup basic routing for all, setup some of the routes like for user profile, added more to user route (to save uploads to user profile in database, retrieve information to display), added more entries for database to test and fixed some references, profile pages uniformity, code/file cleanup, recorded meeting logs, front-end screenshots, presentation contribution, password constraint updates, Admin Page, Wireframes, user stories, project plan structure and software, proposed architecture plan, use case diagram.

Sean Fitze: Added login and logout functionality by setting up post methods in the server file. Fixed bug so that when wrong credentials are put in the login fails. Added a way to display invalid credentials when login fails that was later modified. Added functionality to the search bar using post methods on the server file and a search page so users are able to search the database for notes they want. Also, added a way to view pdfs on the search page just like on the user profile page.

Ethan Bauer: Developed parts of user interface for admin/user profile views, server.js, some .ejs files. Worked on user report document feature and admin remove document feature. Helped with initial planning and project ideas stage, team meetings, Amazon s3 buckets, initial wireframes, team presentation, Jira user stories.