Tech Exercise Write Up

Zach Randolph

April 5th, 2022

Summary

For my tech-exercise I chose to create a to-do list. I wanted something practical that I could actually use in my everyday life rather than something that I just made for a class. It is a relatively simple application that allows a user to add chores by assigning a task and a priority to a chore. It also allows the user to view the chores that they have added to their to-do list by simply clicking a button to view them. Finally, it allows users to remove chores from their list by supplying the ID of the chore, which is auto generated by the database, the application then searches the database for a chore with the corresponding ID, and removes the entire row from the database. It's similar in function to what we have done in class before with each function belonging on its own page. Each page has its own HTML file that displays all the text on a page and hosts the buttons, text boxes, and other forms of input such as links to navigate to other pages. Each button has a certain function if its requirements are met, as in, if you are trying to insert a chore, you have to have things in the input boxes to add something.

The search button, when pressed, runs a MySQL query that searches the database and prints all rows that are in the table. When adding a chore on the add chores page, once the text input boxes are filled, pressing the insert button will run a MySQL query that inserts your chore to the database, aka your to-do list. When deleting a chore, you simply need to provide the ID of the chore to the text input box, once the button is pressed, a MySQL query is executed that searches the database for the row with the provided ID and removes that row.

Install and Configuration

I used an EC2 server hosted on AWS to host my website. On that EC2 server, I have installed a JDK to allow my application to run java, the JDBC connector that allows my application to use MySQL, and configured Apache tomcat to run on my server. All of the setup and configuration was done in class which really saved me a lot of time when starting my project.

For the webserver, I used Apache Tomcat 7 to run and deploy my project. It was also configured in class which made things very easy to deploy.

I used MySQL as my DB system for this exercise. We configured it in class so I used it to save time on the project.

There is no need to download or install anything as the application is entirely web based and you simply need to use the link to access the application.

Tech Exercise Write Up

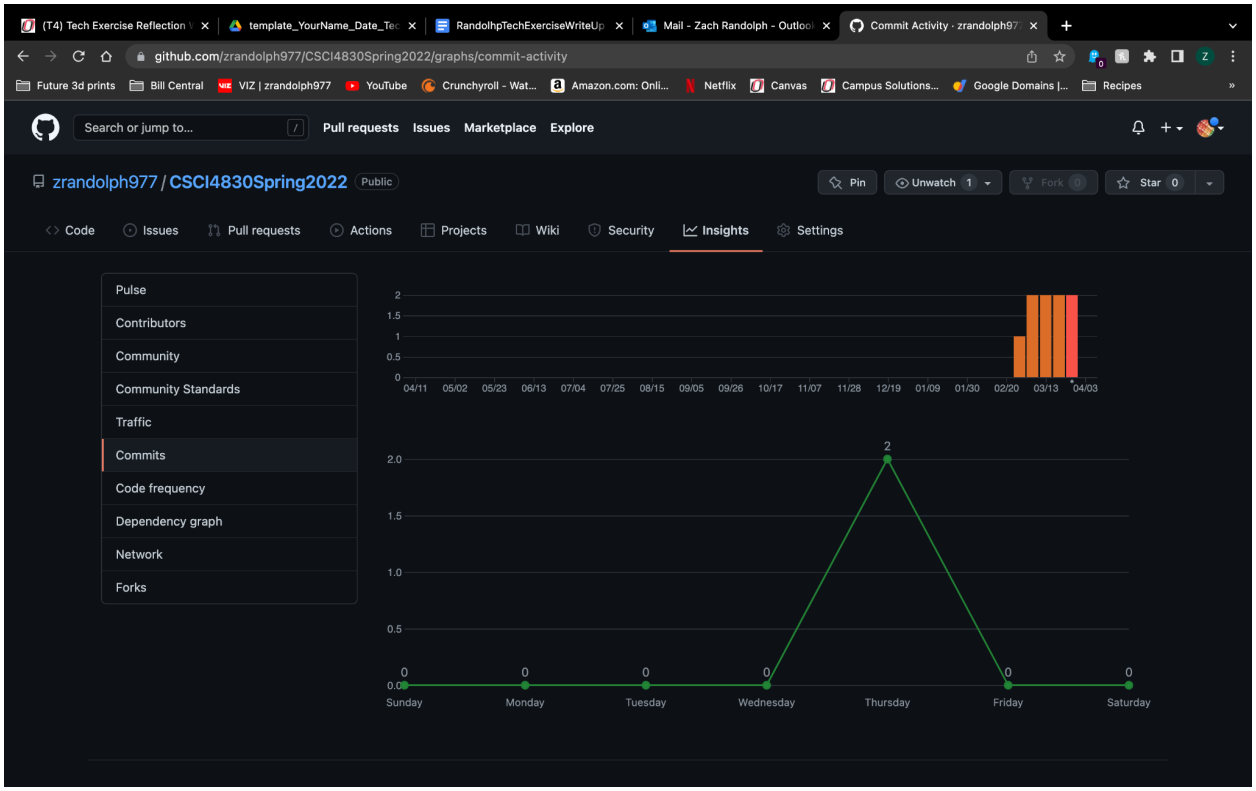
Zach Randolph

April 5th, 2022

I deployed the application following my manual testing by downloading a .war file from eclipse and having Tomcat 7 point to where that file is to run the application. To use the application, simply follow the on screen instructions on each page.

App Management and Evolution

I did have prior experience with Github both from class work as well as personal projects and my job. However, I did not do a very good job at using it correctly. I had very few commits which is bad practice. Because I forgot to commit my changes each day as I worked on it, my commits all happened on one day, being the day before it was due.



As we can see in the graph, I had two commits on Thursday, the day before the project was due. I committed the entirety of my project then committed a bug fix as I caught an issue while testing. I did deviate from my original plan as well. I decided that it didn't really make sense to track a name or an email on this application because it's meant to be a to-do list used by a single person. Chore categories also didn't make a lot of sense considering the same thing. As well as because categorizing chores doesn't really have a use other than changing how the list is sorted. On sorting, I wanted to try and sort the list using the priority provided by the user but I couldn't quite figure out how to do that.