John Gay

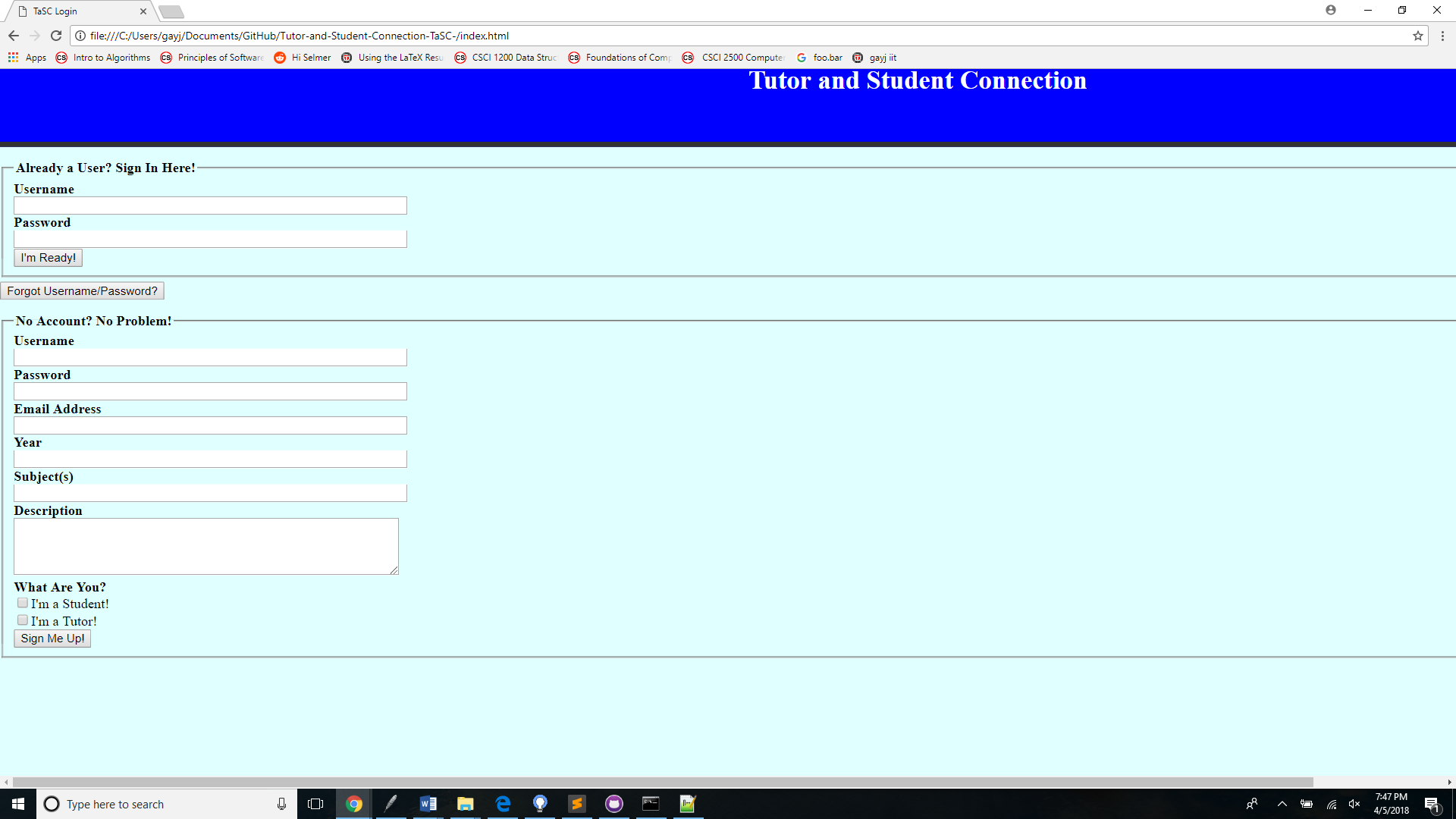
Jonny Koppelman

Wilson Wong

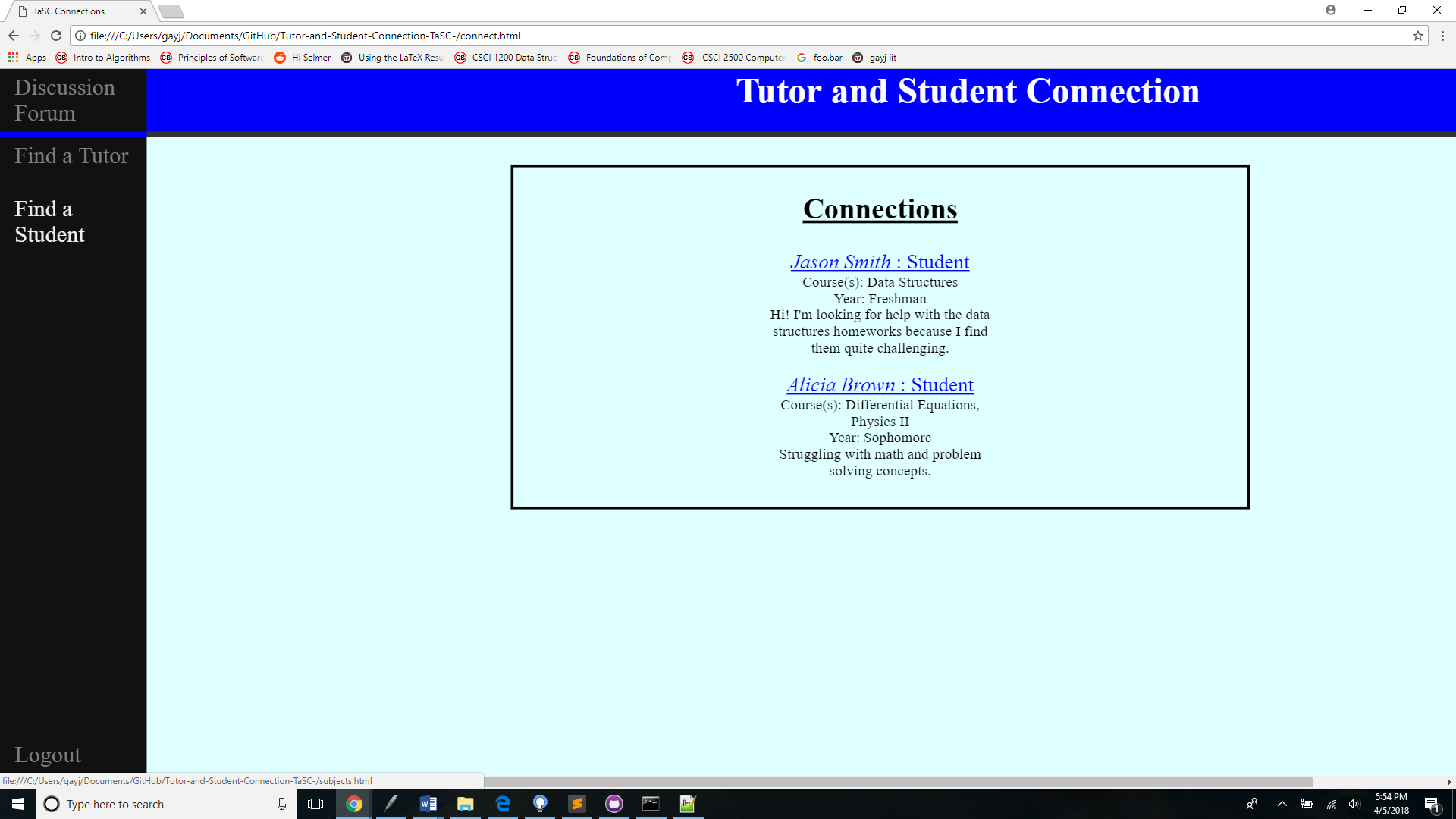
Lab 7: TaSC Web App Mock-Ups

For this lab we have made a mock-up of our web application: Tutor and Student Connection (TaSC). We have done this using html and css because we decided this would be the most efficient way to think about the design of our project while also making progress on the actual website itself. Below is a series of screenshots of each page we have designed for our website. Some of the pages have been zoomed in to improve readability, which can make the heading appear shifted over too far or the logout button not be visible.

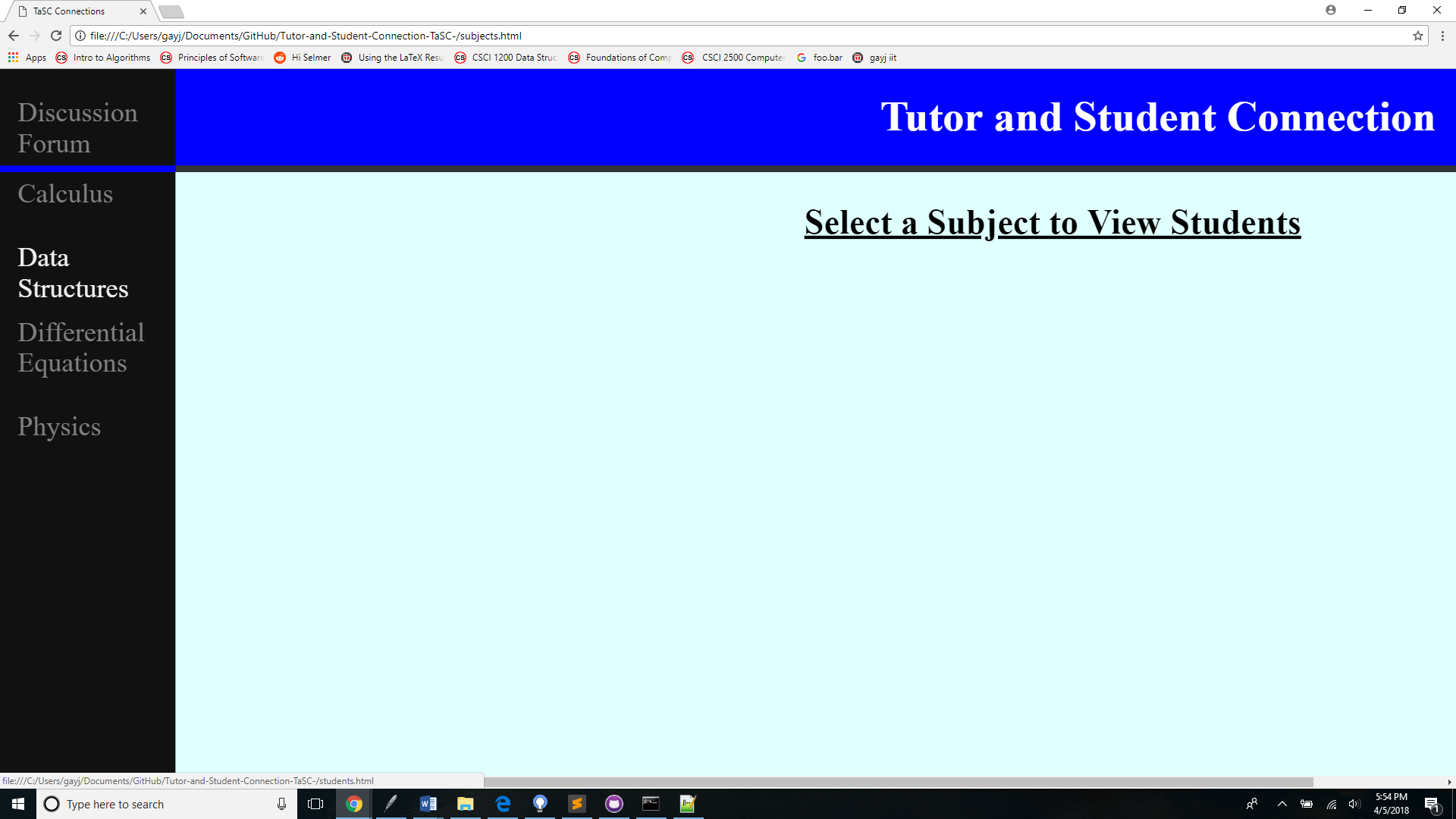
The index of our site is the login page. Upon visiting our website, you will be prompted to login or signup if you do not have an account already. The top block is for a user to sign in, where they must input a username and password in order to login. If they have forgotten their username or password, they can click on the “Forgot Username/Password?” button to change their password. The second block is for new users who do not yet have an account. They must input a username, password, and email to create an account, along with designating whether they are a student or a tutor. Then they must enter information about themselves, including their year in college, their subjects of interest for tutoring, and a quick description about their tutoring abilities/needs.



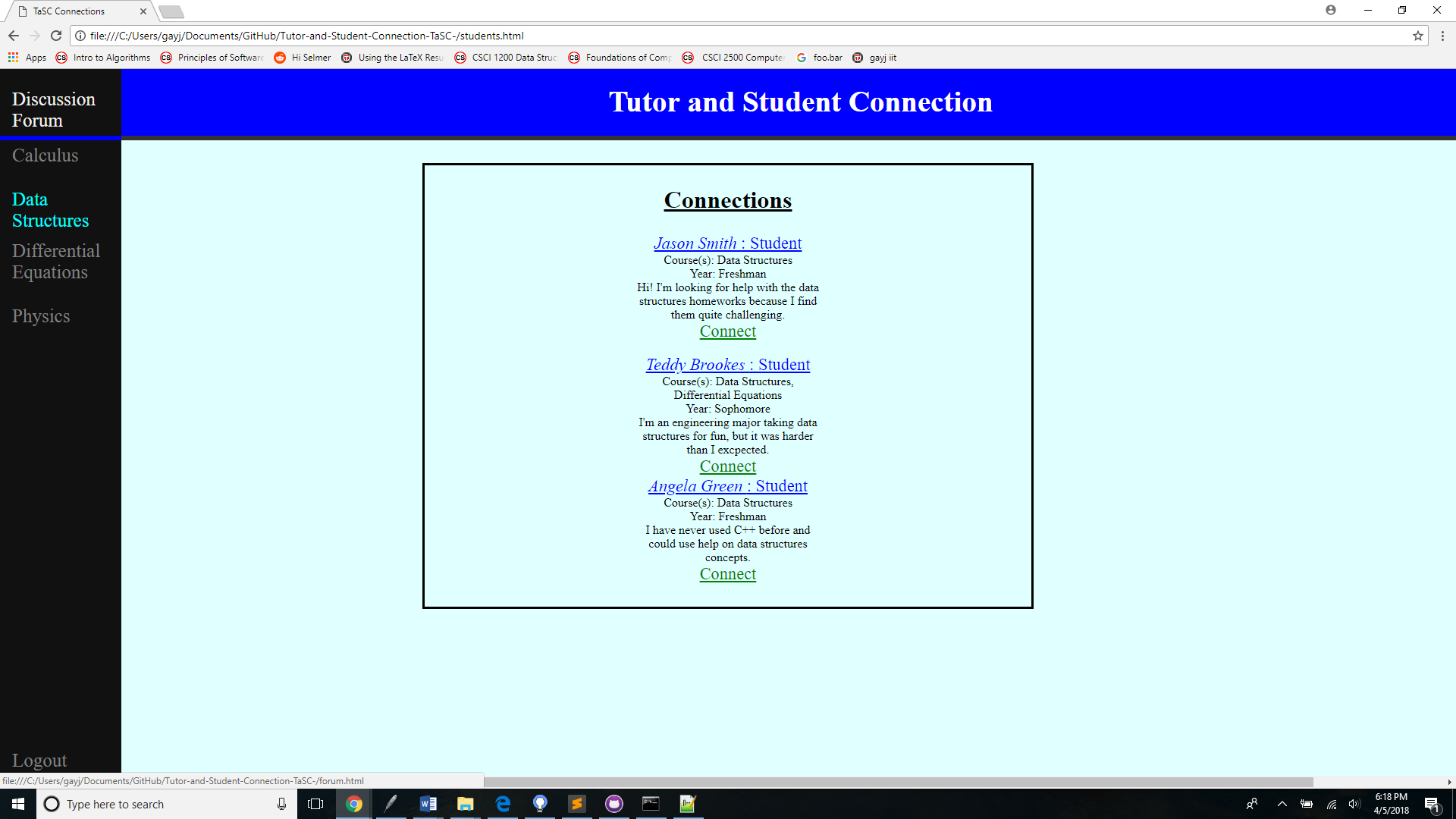
Once signed in, the user will be brought to the connect page. This is the main page for making a connection between a student and tutor. On the navigation bar on the left, there is a link to the discussion forum at the top, which is the other feature of this application. Below that a user can select whether they are looking for a tutor or a student to connect with. On the bottom of the bar, there is a logout button. In the center of the screen is a list of all the people that the user has already connected with. In this example, this is a tutor that has connected with two students already. In this mock-up, we click on the Find a Student link.



After selecting the “Find a Student” button, the user is brought to another page that requires the student to select a subject in order to view the students that are requesting tutoring in that subject. In this example we are selecting the “Data Structures” subject.



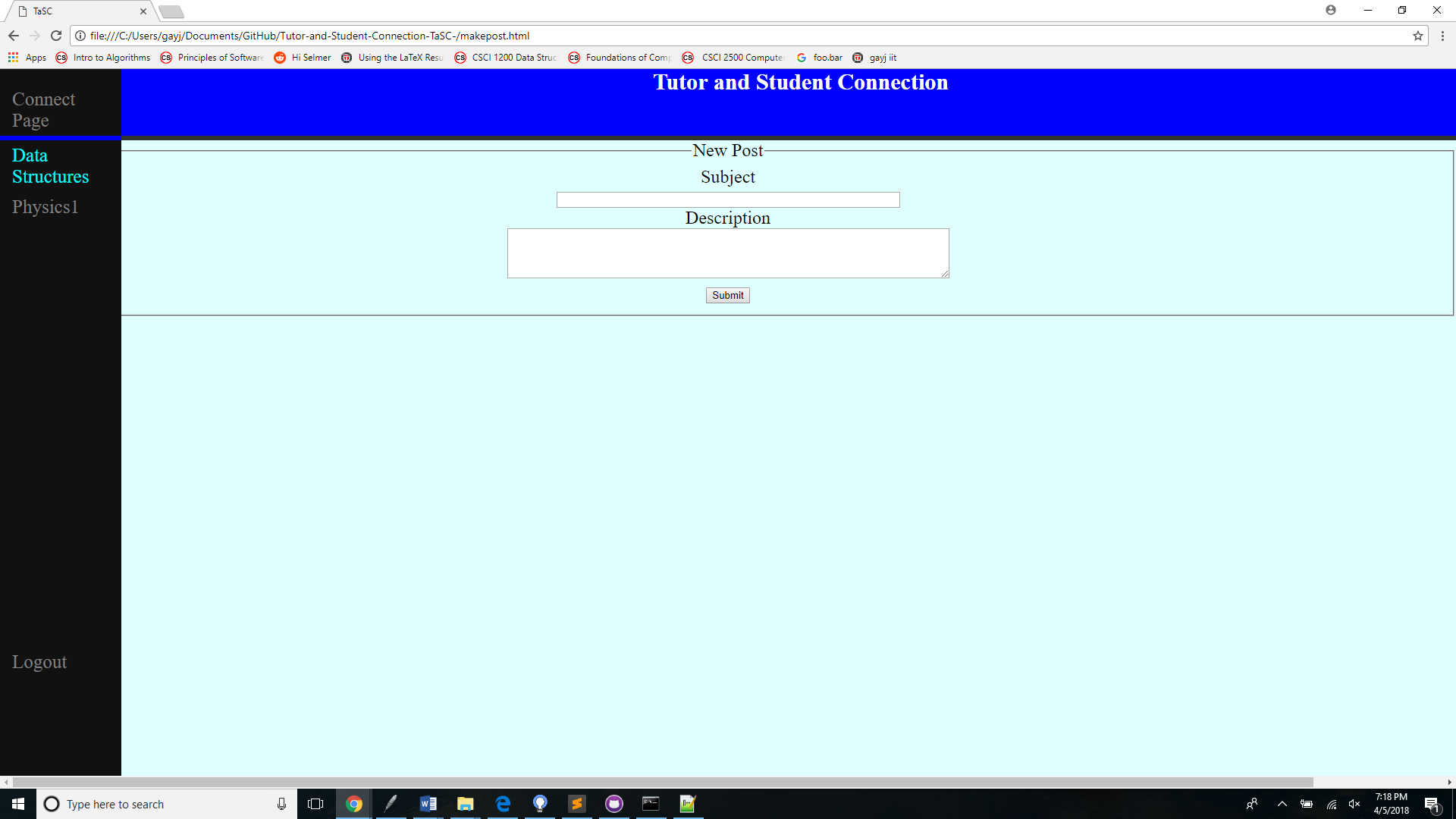
After selecting the subject, the user is then given a list of students in that subject that are in need of tutoring. If the tutor decides they want to help a student, they can click on the connect button, adding them to their connections. Each student has a description of their tutoring needs, along with their subjects and their year in college.



Finally, we are brought to the discussion forum page. This page operates similarly to the connect page, except it remains on the same page when you click the links, but changes the subject material. By clicking on the “Data Structures” subject, we have opened up the discussion forum for the respective class. This gives us a list of all of the posts made in the class so that students can traverse through old posts. If a user wants to add a new post, they can click on the link “Make a Post” in order to write in a new post.



When the user clicks on the “Make a Post” link, they then are given two input fields: a subject and a description. There they can place the subject of their post and then ask a question or write a statement describing the subject they are writing about.



These are the 6 pages that our web application will have. The users can navigate the site using the descriptive links provided in order to gain full functionality that the app is supposed to provide. After logging in, the user can navigate back and forth between the connect page and the forum page. These are the two main pages of the site, and within them, the users can use the sidebar to find the correct subject or student that they are looking for.