Project\_1 Report

By: Atoosa Fasihi

Submitted to: Motasem Aldiab

**The App:**

A picture containing diagram

Description automatically generatedThe application I have created is designed to store student grades in a database as well as display the average grade for the class as a whole. People using the app are able to submit the name of the student and the grade they received into the web app. The name and grade of the students will be displayed in a list at the bottom of the page.

A picture containing timeline

Description automatically generated

To show the class average grade the user must simply press the **Display class average** grade button. After doing so the class top student will be replaced by the class average. To clear the screen of the class average the user must press the **Delete class average grade** button, which removes the class average from the database.

Graphical user interface, text, application

Description automatically generatedGraphical user interface, text

Description automatically generated

**The Structure:**

In creating this app, I used HTML and JavaScript for the front end of the web app. The data that the user inserts into the web app will be stored in MongoDB Atlas which is a cloud-based service.

For the backend of the app, I used Node JS, and Express JS to create a server that the browser can connect to.

Diagram

Description automatically generated

**The Building Process:**

The very first step in the process was to use Node JS in creating a server.js file and a package.json file which helps in managing dependencies. I then used Express to create a server which browsers can connect to. This allows the communication between the Express server and the browser. Following this, I created an index.html file which can be viewed by the browser by adding the **sendfile method** that is provided by **res** object in the server.js file.

After completing the basic app structure, I then focused my attention to creating the database. In creating the database, I used MongoDB Atlas. MongoDB Atlas allows us to store information in a cloud server which can be retrieved and displayed to the user who is viewing our web app.

After successfully connecting to MongoDB Atlas, I implemented a **find method** for MongoDB which allows access to the grades that have been stored in MongoDB Atlas cloud service.

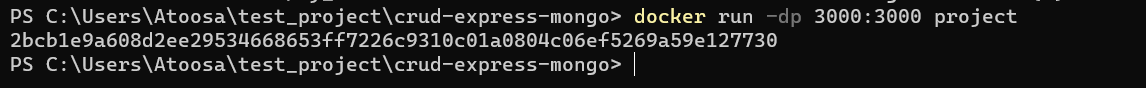
The next step is to render the HTML. I did this by using Embedded JavaScript for my template engine. With this we are able to render the grades to the users on the web app. One of the final steps in creating this web app was to use JavaScript to add two buttons. One of these buttons are to display the class average and the other is to delete the displayed class average from the grades list.

Finally, to deploy the student grades app, I created a Dockerfile in order to create a container image. In the image below we can see the content of the Dockerfile.

Text

Description automatically generated

After creating the container image using **build** command, I was able to run the application by using the **docker run** command.



After running the application using Docker, we are able to view the app that is running successfully in the web browser. I then created a docker-compose.yml file for the MongoDB database. This ensures that the database and the web app are successfully communicating.