Answer these questions:

1. **Discuss your initial thoughts in details on how you will design this application? (2 points)**

For this project, group 30 will be using MongoDB, Express React, Node (MERN). With these tools our group will create a web application that the user can interact with to register, login, create fuel quotes, view fuel quote history.

The front-end will be a single page application built with React that will allow the users to login, register, manage their profile, input data used for the Fuel Quote Form, as well as view their fuel quote history.

We will create the backend using Node and Express. The backend will handle all business logic for the Fuel Quote Form using data entered in by the user from the front end. The backend will communicate with the MongoDB database and handle login/authentication.

MongoDB will be our database that will store user login information, client registration details, and Fuel Quote History.

2. **Discuss what development methodology you will use and why? (2 points)**

Group 30 will be using the Waterfall model for our project implementation because the way the waterfall model is structured is similar to the way that the course is structured. The first step is requirements and that has already been given to us. The second step is software design and that is what we set out to accomplish in assignment #1. The next steps are implementation and testing and we will be following those steps as the assignments are given. The final steps are deployment and maintenance which will correspond to the demo at the end of the semester.

After careful consideration of choosing the Waterfall methodology, we have decided that the issues associated with the model are unlikely to become a problem during implementation. Per example "requirements cannot be changed in the middle." Should not be a problem as we do not believe Dr. Singh will change the project during the semester.

3. **Provide high level design / architecture of your solution that you are proposing? (6 points)**

We will create a front-end user interface with React allowing users to register an account, edit existing account details, submit fuel quotes, get fuel quote history by exchanging information through the node/Express backend. Long term data (client information, fuel quote history) will be stored/retrieved from the MongoDB database through the backend API. The React front-end will not access the database directly, it must be done through the node/Express backend server for security reasons.

Note: A1_Architecture.pdf will contain the architecture UML diagram.

4. IMPORTANT: list who did what within the group. TAs should be able to validate in GitHub, otherwise team members who didn't contribute will receive a ZERO.

Fill in this table, provide as much details possible:

Group Member Name	What is your contribution?	Discussion Notes
1. Kyle James UHID: 2001473	UML Diagram collaboration and wrote assignment #1.	-Trying various UML template / Software. -Group discussion on implementation methodology. -Framework/stack discussion
2. Taylor Barnett UHID: 1831737	UML Diagram collaboration and wrote assignment #1.	-Trying various UML template / Software. -Group discussion on implementation methodology. -Framework/stack discussion
3. N/A	N/A	N/A

Github link:

https://github.com/taylorb123/Software_Design_COSC4353