**Web Application Development 1 - Final Project**

**Team Name**: - **JS Warriors**

**Team members:**

* Daniel Nigussie
* Temesgen Assefa
* Amelework Cheklie

**Completed Requirements**

* We Developed reusable React Components
* We have used class and functional based component

We used

* Immutable state
* use React Hooks.
* use Lifecycle Hooks.
* routes.
* use forms.

Implemented UI

* Users can sign up and sign into the system.
* Admin can perform CRUD operations
* Customers can write a comment (review) to food item.
* Users can list all of the reviews of food.

Not used

* Redux

**Technologies that we have been used**

MongoDB as a persistent data storage

Node JS

Mongoose

Express

React

3 party dependency

* + Axios
  + React dom
  + React-router-dom
  + Bcrypt
  + Jsonwebtoken

**Challenges**

Below are the lists we want to share: -

* We had a time constraint and barely find small breaks to finish the project.
* It was challenging to connect front and backend
* We had a difficulty on showing the reviews and rating when the product displayed.
* The database was not able to include the image url and reviews.
* The menu form would not show up when we clicked the menu button.
* it was hard to differentiate role of the user and admin.
* Due to Covid-19 we were not able to meet with group physically and made it hard to communicate in Microsoft Teams.

**How we solved it**

* we watched class videos and lecture notes and it was helpful.
* also, vlogs and gave a little push.
* Asked help from classmates.

**MEMBER ONE ADMIN**

**Explain your tasks clearly**

* create the git repository
* crate and setup Backend
* Built forms (log in component)
* follow up the team members
* built admin component
* manage the project, submit, deployed

**Explain your components clearly**

* Admin has the Authorization to access & control the **server (**main data storage).
* When Admin logged in and once, he/she get Authorization the page redirected him to the admin component
* In Admin component admin could update and delete menu from the list.

**SECOND MEMBER USER**

**Explain your tasks clearly**

* Contribute on the server-side building process
* User able to register and go to login

**Explain your components clearly**

Register form,

* Anyone can fill out register form then after he/she can log in (POST METHOD).
* And also, the user will see the menu automatically if they have already done the above one action.
* For the user component, we have login form, the form use event handler (onClick)
* Once user can click to the login button (POST METHOD) and get the approval it can take him to the menu list display.
* User can be select the menu from the list to see the details.
* User also write the comment, give rating, and submit it

**THIRD MEMBER MENU**

**Explain your tasks clearly**

* Built Menu components
* Contribute on the server-side building process

Explain your components clearly

When the user selects “MENU” from navbar and click on it, user can see the menu lists

Menu component can fetch (POST METHOD) all the menu and display

Problems that we face