FULL STACK ENGINEERING

COUSE 2- WEB DEVELOPMENT

ASSIGNMENT2- FRONTEND DEVELOPMENT WITH REACT

Submission Date: 22 Jan 2023

Assignment Submission Mode: Canvas LMS- File Upload

Weightage: 20%

I OBJECTIVE:

To create frontend UI for the given user stories using ReactJS

Note: This a group assignment. Each group has to do the problem statement based on the assignment competed in Course 1

II PROBLEM STATEMENT 1:

A web application which offers food ordering services to the Indian rail passengers. Passengers can order food by specifying the station and from which nearby restaurant they want to order food.

As a part of this assignment, you are required to implement the User Interface and its related functionalities for the features listed below as user stories.

EPIC 1: LOGIN AND REGISTRATION

1.1) Registration:

User Story: As a new user, I should be able to register as a customer /restaurant.

Acceptance Criteria:

- Input data fields to enter - 1) Username [email id] 2) Password 3) Basic Details depending on customer or restaurant required.
- Password criteria should be displayed to the user when user clicks inside the password data field.
- If 'Password' entry does not match criteria specified and user hits Submit, show error alert "Password entry does not meet criteria"
- After successful validation of all entered fields and on clicking Submit, show message indicating successful account creation

1.2) Login:

User Story: As a user, I should be able to login with the username and password to the portal.

Acceptance Criteria:

- Input data fields to enter - 1) Username[email id] 2) Password 3) Role

- Indicate invalid usernames and passwords as alerts to the user.
- After successful validation of all entered fields and on clicking Submit, show message "Login Successful" and redirect to the Dashboard page of user.

Assumptions for EPIC 1[User Stories 1.1 and 1.2]

Only the user interfaces and client side validation need to be implemented.

It can be assumed that the users are successfully created/logged in when the entered values are valid or Mock APIs can be used.

Any other assumptions made can be listed in the documentation

EPIC 2: CUSTOMER FOOD ORDER

User Story 2.1: As a customer, I should be presented with a dashboard page on successful login.

Description: The dashboard page should contain navigation menu options to search for available restaurants and order food, view previous order details. The Dashboard to contain the status of active order details

Acceptance Criteria:

If there are no active orders, it should display a message "no orders".

The status of active orders with details to be displayed.

User Story 2.2: As a customer, I should be able to view the available restaurants at a given location.

Description: On the customer entering the required details like PNR number, location (station name) and other details, the list of available restaurants should be displayed. Filtering options to be present

Acceptance Criteria:

The restaurants details to be displayed

Pagination to be applied.

User Story 2.3: As a customer, I should be able to select the items and order food

Description: On clicking the available restaurants, the user should be able to select items from the restaurant menu and place the order. The user should be made to enter the appropriate payment mode.

Acceptance Criteria:

Once the user selects the items and places the order, the details of the order with status to be displayed on dashboard.

Assumptions for EPIC 2

For this assignment, assume that details are fetched from the backend and available in JSON format. You can use Mock APIs also

Any other assumptions made can be listed in the documentation.

EPIC 3: RESTAURANT DASHBOARD

User Story 3.1: As a restaurant administrator, I should be presented with a dashboard page on successful login.

Description: The dashboard page should contain navigation menu options to add/update menu items, time, availability status etc., The Dashboard to contain the status of active order details (Orders pending for confirmation, orders awaiting delivery

Acceptance Criteria:

If there are no active orders, it should display a message "no orders".

The status of active orders with details to be displayed.

User story 3.2: As a restaurant administrator, I should be able to accept or reject the orders

Description: The restaurant administrator should be able to view the details of the order and reject /accept it

Acceptance Criteria:

The status of upcoming orders with details to be displayed in dashboard

Assumptions for EPIC 3:

For this assignment, assume that details are fetched from the backend and available in JSON format. You can use Mock APIs also

Any other assumptions made can be listed in the documentation.

Menu Edit/update pages can be kept under construction. Bulk Data can be also be uploaded

IV: SUBMISSION INSTRUCTIONS

1) Upload the complete code to the github repository.
2) Create a word document, with the brief description of user stories implemented, snapshots of UI, Github link and assumptions made.
3) Upload the word document to the Canvas LMS
4) While naming the files use your Group ID and group name for identification purposes
5) **Academic Honesty**: You are welcome to discuss with peers and refer the internet in order to better understand the concept, but you may not share code or do not do a verbatim copy of the source code from the internet.
6) The participants are allowed to apply their own UI styles, add more UI elements as per the requirement. The participants can leverage on use cases defined as part of course 1 assignment and course 2 assignment 1 for improving the user experience.

Files to Be Submitted

1) Complete code in Github- Link to be shared. Repository access to be given to akshaya.ganesan@pilani.bits-pilani.ac.in
2) Description document to be uploaded to Canvas.
   It contains
   - Description of features implemented
   - Snapshots of UI
   - Assumptions made
   - Github Link
   - Execution instructions
   - A video recording of the Demo (If case, you face size limit issue while uploading the video to canvas, add it to the google drive and share the link)

V WEIGHTAGE:

Total: 20 Points

EPIC 1: Implementation and Demo: 5%

EPIC 2: Implementation and Demo: 9%

EPIC 3: Implementation and Demo: 6%

VI EVALUATION

The relevant documents to be submitted in Canvas. Evaluation will be done based on the documents and code submitted.

VII NOTES:

- This is a take-home assignment to be carried out by each group independently
- In case of any further queries, use discussion forums, or reach out to me at akshaya.ganesan@pilani.bits-pilani.ac.in.