**CSSE4103-FULL STACK WEB DEVELOPMENT**

**Mock Exam**

**Instructions:**

1. The exam must be fully practical.
2. Exam duration: **1 hour and 15 minutes** (including downloading and uploading).
3. Download the exam files from E-learning, unzip the folder in your Desktop.
4. Open the folder in Visual Studio Code.
5. Open a terminal, change directory to the client folder and execute the command: **npm install.**
6. Start the server.

**Section 1-Front-End Development**

1. In the **Order** component:
   1. Use the Reactstrap component to achieve the given layout.

|  |
| --- |
| Image |
| Order Form |

* 1. Write the code to add the image.
  2. Create the state variables required for all the data for the input fields in the form.
  3. Use the register function to register all form elements in the react-hook-form.
  4. Integrate validation schema that you have created with React Hook Form using yupResolver.
  5. Display an error message display for each validation error encountered.

1. In **src/Validation/ orderSchemaValidation.js**, define the schema to validate the order data.
   1. Product name must be a string and required.
   2. Price must be a number and required.
   3. Quantity must be a number with a minimum of 1 and maximum of 100. It must also be required.
   4. Import this schema in the Order component to validate the order data.
2. Update the **App.js** component and create the following routes.
   1. /order-render the Order components
   2. /list-render the ListOrder component
3. Update the **Header.js** component and create the links to the routes.

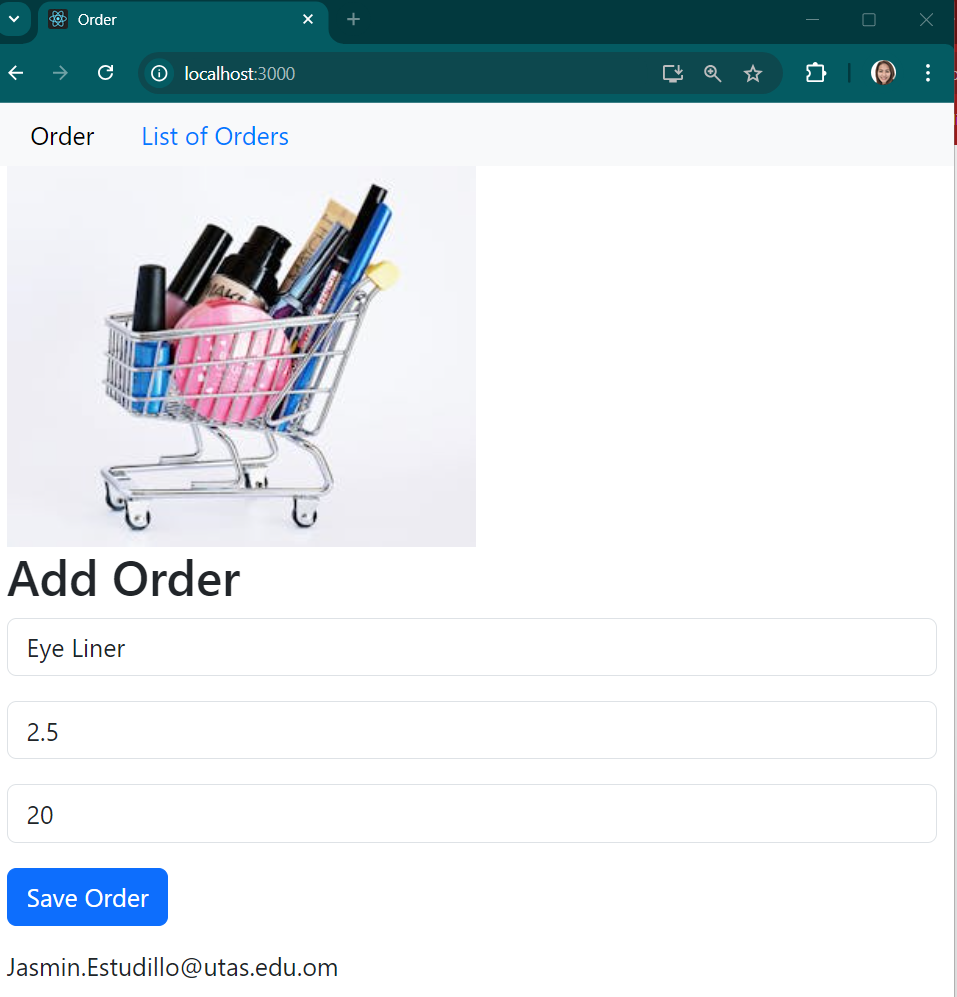
**Section 2-Redux**

1. In **src/Features/OrderSlice.js:**
   1. Create a slice with the following details:
      1. The name is “orders”.
      2. Initial value set to an empty array.
      3. One reducer “addOrder”.
   2. In the **addOrder** reducer, write the code to add the new order to the state with the data submitted from the form.
   3. Export the addOrder reducer function.
2. In **src/Store/store.js**:
   1. Configure the Redux store to manage the orders state slice by adding the ordersReducer..
3. In **index.js**:
   1. Import the Redux store.
   2. Import the Provider component from react-redux.
   3. Wrap the app with Provider to make the store available to all the components.
4. In the **Order** component enables users to place an order. Edit the component to:
   1. Add the internal state variables and corresponding event handlers for the input fields.
   2. Update the function **onSubmit()** to dispatch the addOrder action and pass the order data as argument to the Redux store. The order data should have the following values:

* Name
* Price
* quantity,
* payable(price\*quantity)

1. In **ListOrder** component, use the useSelector from react-redux to retrieve the order data from the store and display it in a table.

Example Application Output:



A screenshot of a computer

Description automatically generated