SMART PUBLIC RESTROOM

Phase-4: Development Part 2

Introduction:

The "Smart Public Restroom Management System" is an innovative solution designed to enhance the efficiency and user experience of public restrooms. Leveraging the power of IOT (Internet of Things) technology and a user-friendly web application built with React, this project offers a comprehensive and intelligent restroom management system.

In today's fast-paced world, the demand for clean and accessible public restrooms is ever-increasing. However, managing these facilities can be a challenging task. This project addresses these challenges by providing real-time monitoring, control, and user feedback through a web application.

Key Features:

1. **Real-time Monitoring:** Sensors within each restroom continuously monitor occupancy, cleanliness, and supply levels. This data is instantly accessible through the web application.
2. **Smart Occupancy Management:** Users can check the availability of restrooms through the web application before visiting a location, minimizingwaiting times.
3. **Automated Maintenance Alerts:** The system sends maintenance alerts when restrooms require cleaning or supply replenishment, ensuring a consistent level of service.
4. **User Feedback:** Visitors can rate and review the cleanliness and overall experience of each restroom, allowing for immediate improvements.
5. **Sustainability:** The system promotes eco-friendliness by reducing water andenergy wastage through smart controls.
6. **Web Application:** The React-based web application offers an intuitive interface for users to find and review restrooms, while administrators can manage and monitor multiple locations seamlessly.

Smart Public Restroom Information Platform:

1. Dashboard:

- Provides an overview of multiple public restrooms.
- Lists the available restrooms, their status, cleanliness, and supply levels.

2. Individual Restroom Details:

- Users can click on a specific restroom for more information.
- Real-time status (e.g., open or closed) and occupancy information.

3. Cleanliness Monitoring:

- Visualizes the cleanliness of each restroom, with real-time updates.
- May include ratings from previous users.

4. Supply Management:

- Shows supply levels (e.g., toilet paper, soap, hand sanitizer).
- Alerts facility managers when supplies are low.

5. User Feedback:

- Allows users to rate and comment on the cleanliness and service.
- Visualizes the cleanliness of each restroom, with real-time updates.
- May include ratings from previous users.

6. Supply Management:

- Shows supply levels (e.g., toilet paper, soap, hand sanitizer).
- Alerts facility managers when supplies are low.

7. User Feedback:

- Allows users to rate and comment on the cleanliness and service.
- Provides a feedback loop for continuous improvement.

MOBILE APPS:

1. Real-time Restroom Status:

- Displays the current status of nearby public restrooms (open/closed).
- Utilizes IOT occupancy sensors for real-time updates.

2. Cleanliness Information:

- Shows cleanliness ratings and information for each restroom.
- Allows users to provide ratings and comments.

3. Supply Levels:

- Indicates the availability of essential supplies (toilet paper, soap, hand sanitizer).
- Notifies users if supplies are running low.

4. Navigation:

- Offers directions to the nearest smart public restroom**.**

5. IOT Integration:

- Integrates with IOT sensors for occupancy and cleanliness data.
- Uses sensors for real-time monitoring.

6. Push Notifications:

- Notifies users about restroom status changes, cleanliness updates, or low supply alerts.

7. Security and Privacy:

- Implements user authentication and data encryption to protect user information.

8. Encryption:

-Use strong encryption protocols (e.g., HTTPS) to protect data in transit between

IOT devices, the app, and the back-end server.

9. Data Analytics:

- Collects and analyses historical data for insights and trends.

- Helps facility managers make informed decisions.

10. Accessibility:

- Ensures that the app is accessible to people with disabilities.

11. Social Sharing:

- Allows users to share their restroom experience on social media.

12. Multiple Languages:

- Supports multiple languages for a diverse user base**.**

USING A WEB DEVELOPMENT:

Creating a complete smart public restroom using IOT webpage requires a substantial amount of code, and it can be quite complex. However, I can provide you with a simplified example of an HTML, CSS, and JavaScript template for a webpage that displays restroom information. You would need to integrate this with IOT devices and server-side logic for real-world functionality.

Sample Code:

HTML:

We have a simple interface for viewing real-time traffic information.

<!DOCTYPE html>

<html>

<head>

<title>Smart Restroom Dashboard</title>

<style>

/* Add your CSS styling here */

```
body {
font-family: Arial, sans-serif;
}
.container {
max-width: 800px;
margin: 0 auto;
padding: 20px;
}
/* Add more CSS as needed */
</style>
</head>
<body>
<div class="container">
<h1>Smart Restroom Dashboard</h1><h2>Restroom Information</h2>
<p>Status: <span id="restroom Status">Loading...</span></p>
<p>Cleanliness: <span id="cleanliness">Loading...</span></p>
<p>Supply Levels: <span id="supply Levels">Loading...</span></p>
</div>

<script>
// Add your JavaScript code to fetch IoT data and update the webpage
function updateRestroomData() {
// Simulated IOT data (replace with actual data retrieval)
const restroomData = {
status: 'Open',
cleanliness: 'Good',
supplyLevels: 'Adequate'
```

```
};

// Update the webpage elements with IoT data
document.getElementById('restroomStatus').textContent = restroomData.status;
document.getElementById('cleanliness').textContent = restroomData.cleanliness;
document.getElementById('supplyLevels').textContent = restroomData.supplyLevels;
}

// Periodically update the data (e.g., every 30
secondssetInterval(updateRestroomData, 30000);

// Initial data update
updateRestroomData();
</script>
</body>
</html>
```

CSS:

Create a stylesheet (styles.css) to define the layout and styling of your web page.

```
/* Reset some default browser styles */
body, h1, h2, p {
margin: 0;
padding: 0;
}

/* Set a background color and text color */
```

```
body {
background-color: #f0f0f0;
color: #333;
}

/* Center the content */
.container {
max-width:
800pxmargin: 0 auto;
padding: 20px;
background-color: #fff;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
border-radius: 5px;
}

/* Style headings */
h1 {
font-size: 24px;
margin-bottom: 10px;
}
h2 {
font-size: 20px;
margin-bottom: 10px;
}

/* Style data elements */
p {
```

font-size: 16px;

margin-bottom:

15px;}

/* Update link styles if needed */

a {

color: #0077cc;

text-decoration: none;

}

a:hover {

text-decoration: underline;

}

JavaScript:

JavaScript code snippet that updates the restroom information on your smart public restroom webpage.

Javascript function updateRestroomData() {

// Simulated IoT data (replace with actual data retrieval)

const restroomData = {

status: 'Open',

cleanliness: 'Good',

supplyLevels: 'Adequate'

};

// Update the webpage elements with IOT

```
document.getElementById('restroomStatus').textContent = restroomData.status;

document.getElementById('cleanliness').textContent = restroomData.cleanliness;

document.getElementById('supplyLevels').textContent = restroomData.supplyLevels;

}


// Periodically update the data (e.g., every 30 seconds)

setInterval(updateRestroomData, 30000);


// Initial data update

updateRestroomData();
```

DESIGNING A MOBILE APP:

Designing a mobile app for a smart public restroom using IoT involves several key considerations. Here's a high-level guide to get you started:

1. Define the Purpose and Features:

- Clearly define the objectives of the mobile app, such as remote restroom monitoring, user experience enhancement, or resource management

2. User Research:

- Understand the needs and preferences of the restroom users, facility managers, and other stakeholders. This will help you design a user-centered app.

3. IOT Integration:

- Identify the IOT devices and sensors you'll use in the restroom (e.g., occupancy sensors, water quality monitors, or smart dispensers). Ensure they are compatible with your app.

4. Wireframing and Prototyping:

- Create wireframes and prototypes of the app's user interface, considering ease of use and accessibility. Test the user flow and make improvements based on feedback.

5. Real-time Monitoring:

- Design features that allow real-time monitoring of restroom conditions. Users can check restroom availability, cleanliness, and supply levels through the app.

6. Resource Management:

- Implement features for managing restroom resources efficiently. For instance, automatic alerts for refilling supplies or scheduling cleaning based on usage patterns.

7. User Feedback and Ratings:

- Include a feature for users to provide feedback or rate the restroom's cleanliness and service, helping facility managers improve.

8. Security and Privacy:

- Prioritize security to protect IOT data and user information. Implement encryption and authentication measures

9. Cross-Platform Considerations:

- Decide whether you'll develop native apps for iOS and Android or use cross-platform development tools.

10. IOT Data Visualization:

- Create clear and visually appealing ways to present data from IoT devices in the app. Charts, graphs, and maps can be useful.

11. App Testing:

- Rigorously test the app, ensuring that it functions smoothly with IoT devices. Check for any connectivity issues.

12. Compliance and Regulations:

- Ensure that your app complies with relevant regulations and standards, especially concerning data privacy and IOT device safety.

13. App Launch and Promotion:

- Launch the app on app stores and promote it to users. Consider partnerships with facility owners or local authorities to gain visibility.

14. Feedback and Updates:

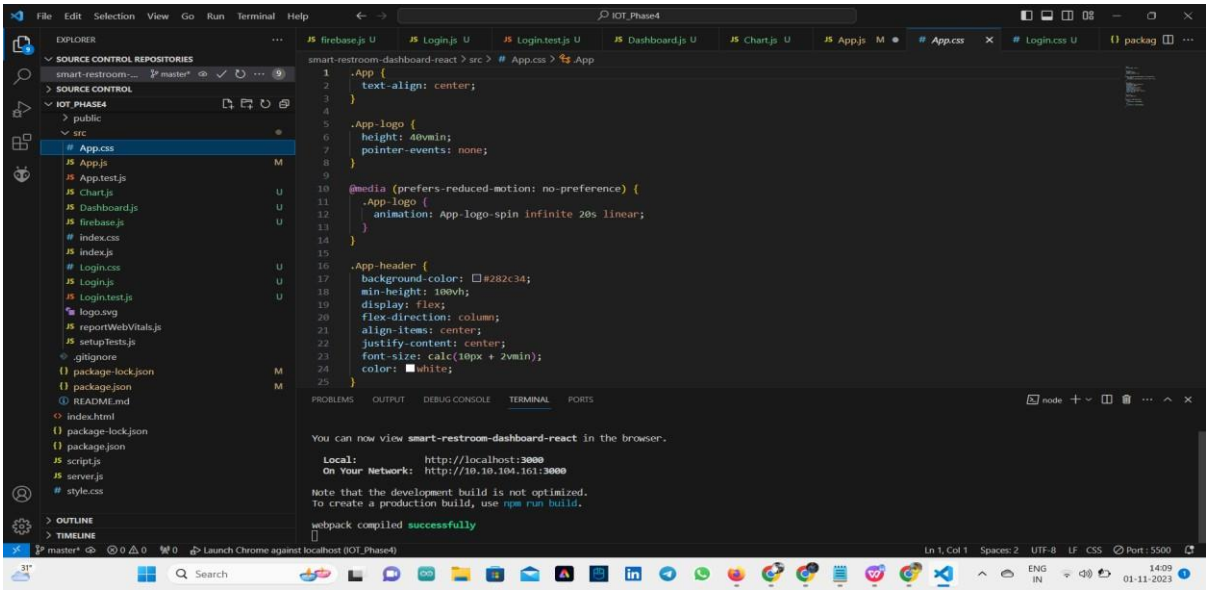
- Continuously gather user feedback and make updates to enhance the app's performance and usability.

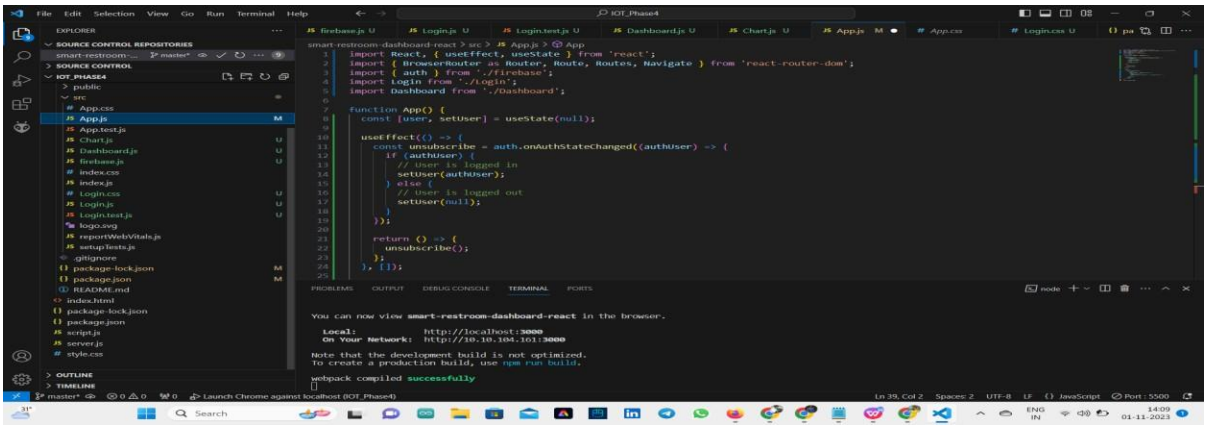
15. Maintenance and Support:

- Provide ongoing support and maintenance to address issues and keep the app up to date.

Screenshots and Output of My

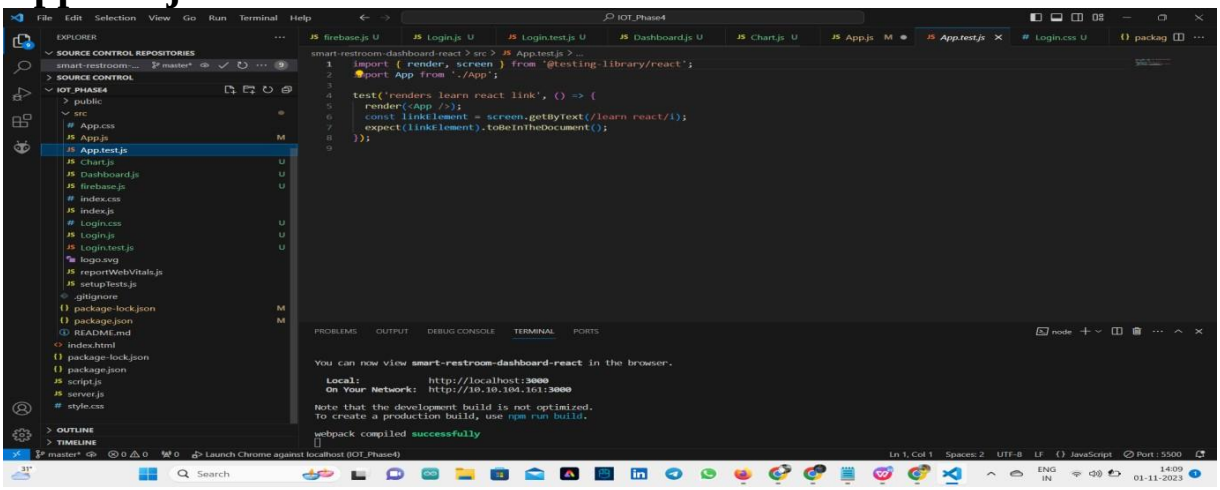
Project: App.css





App.js

App.test.js



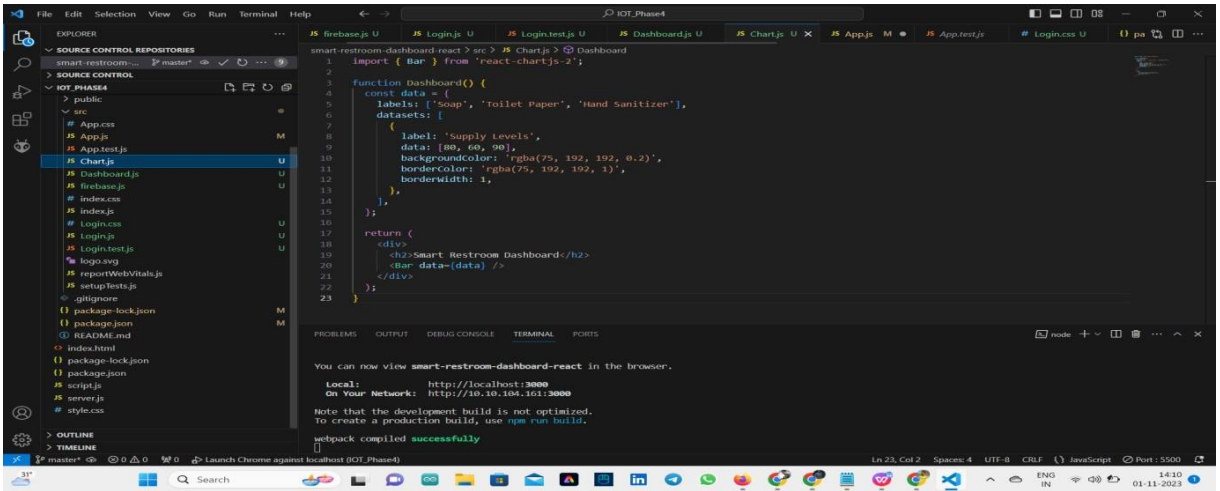
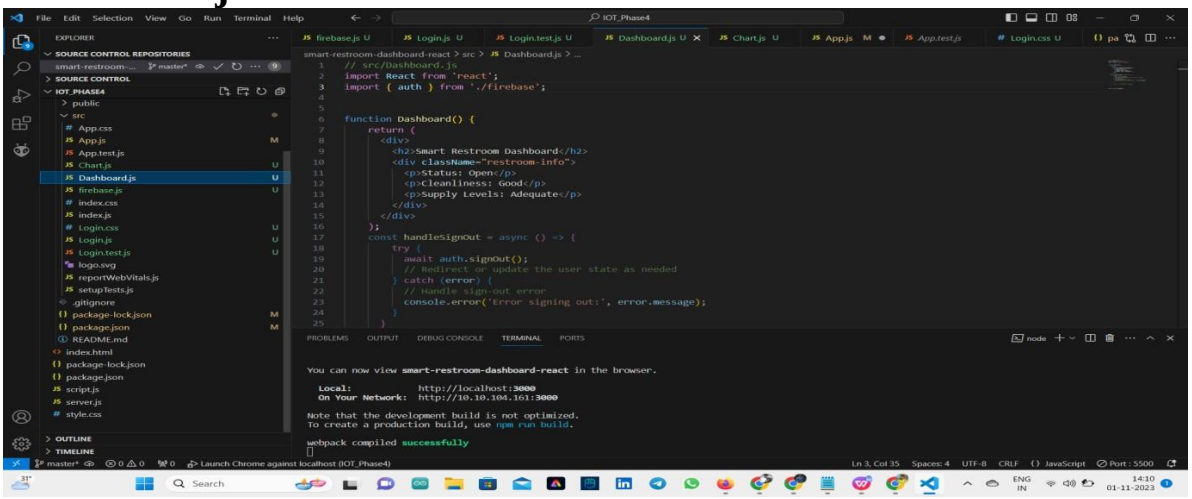
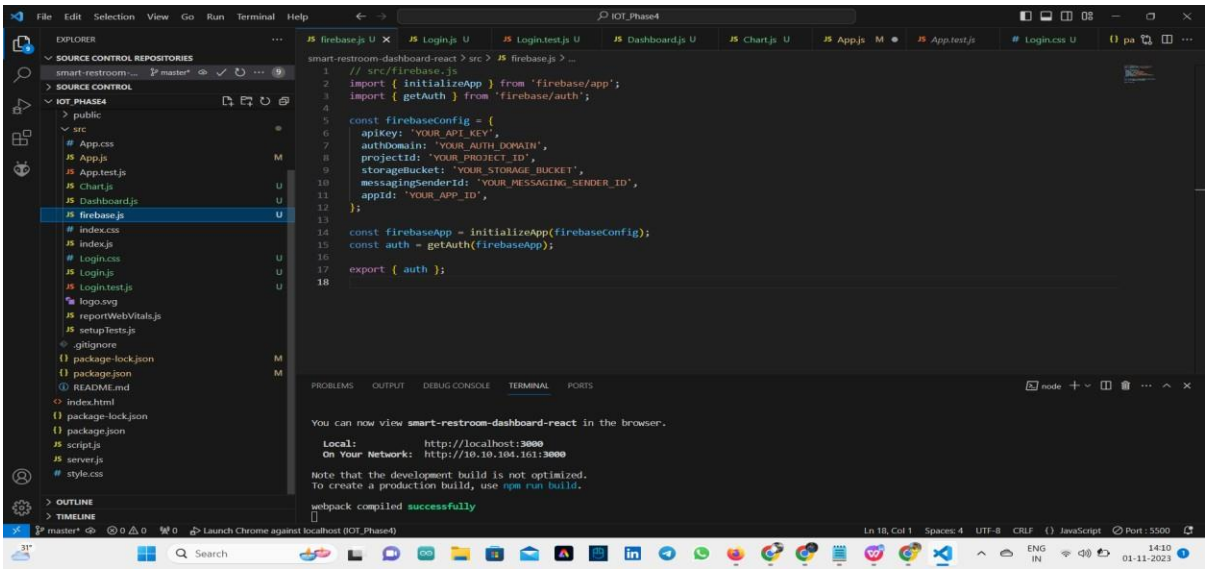


Chart.js

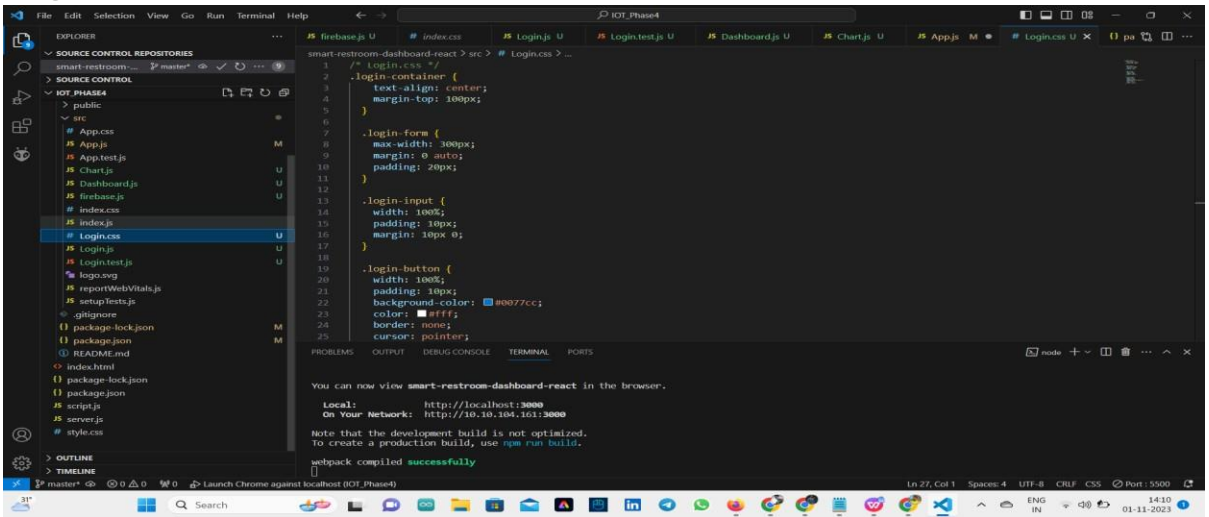
Dashboard.js

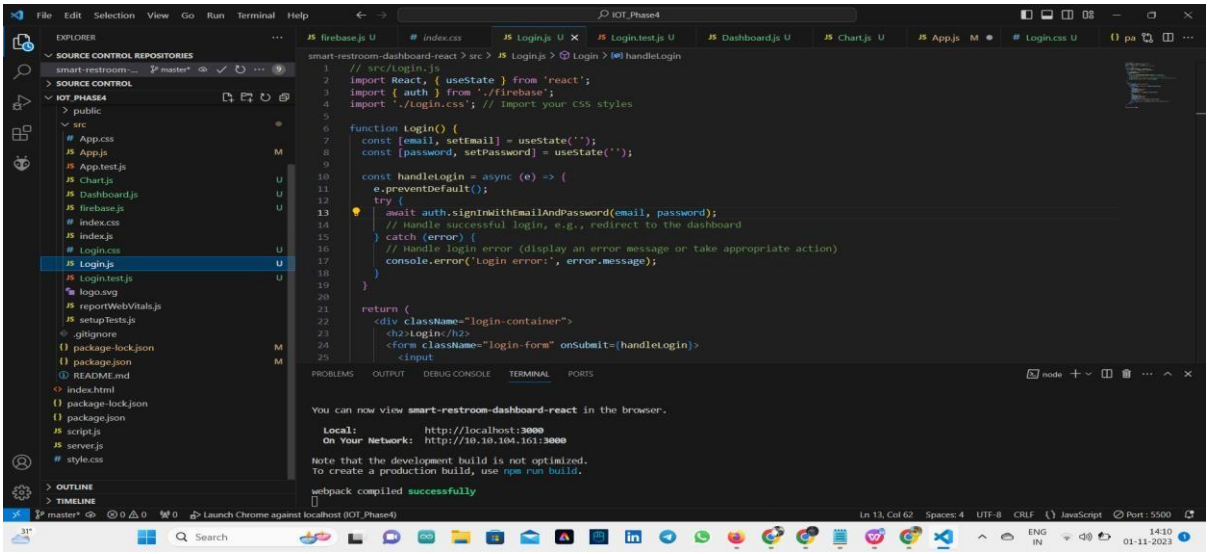




Firebase.js

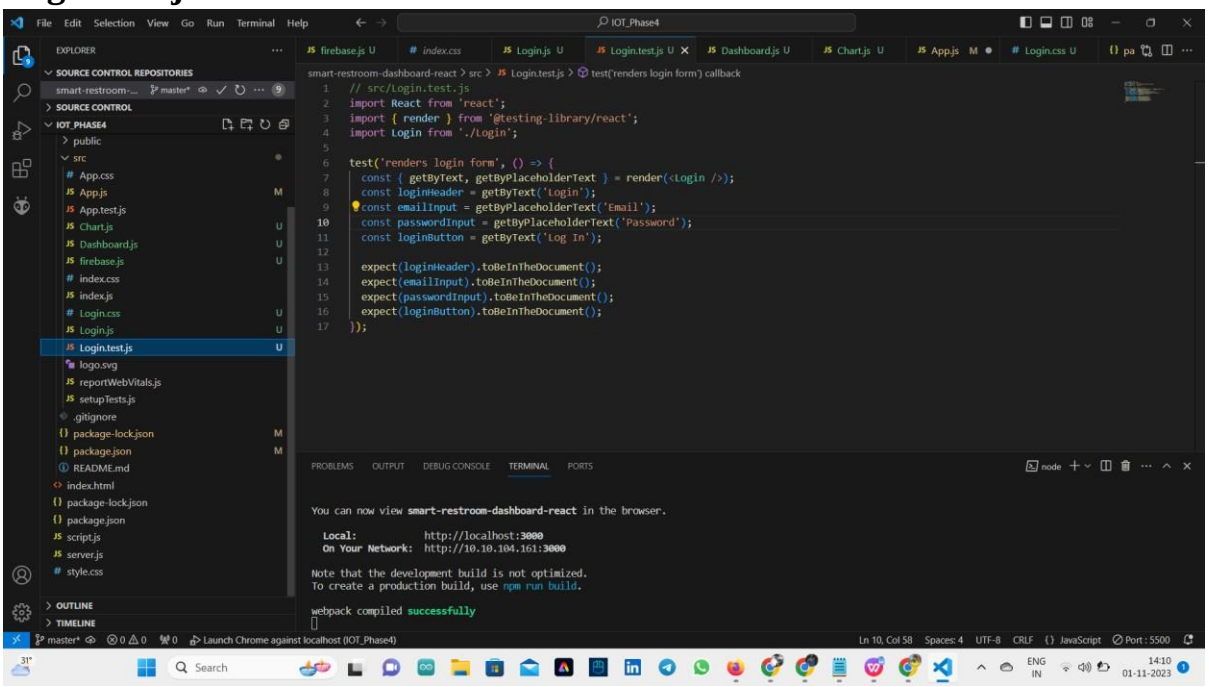
Login.css

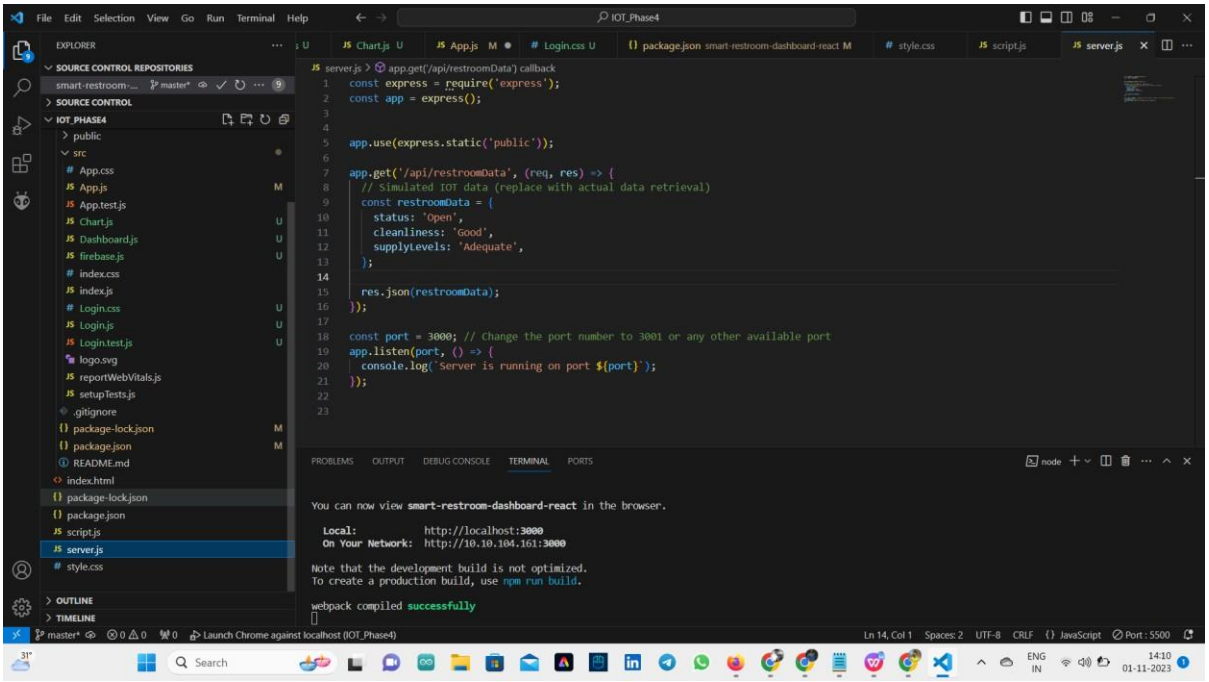




Login.js

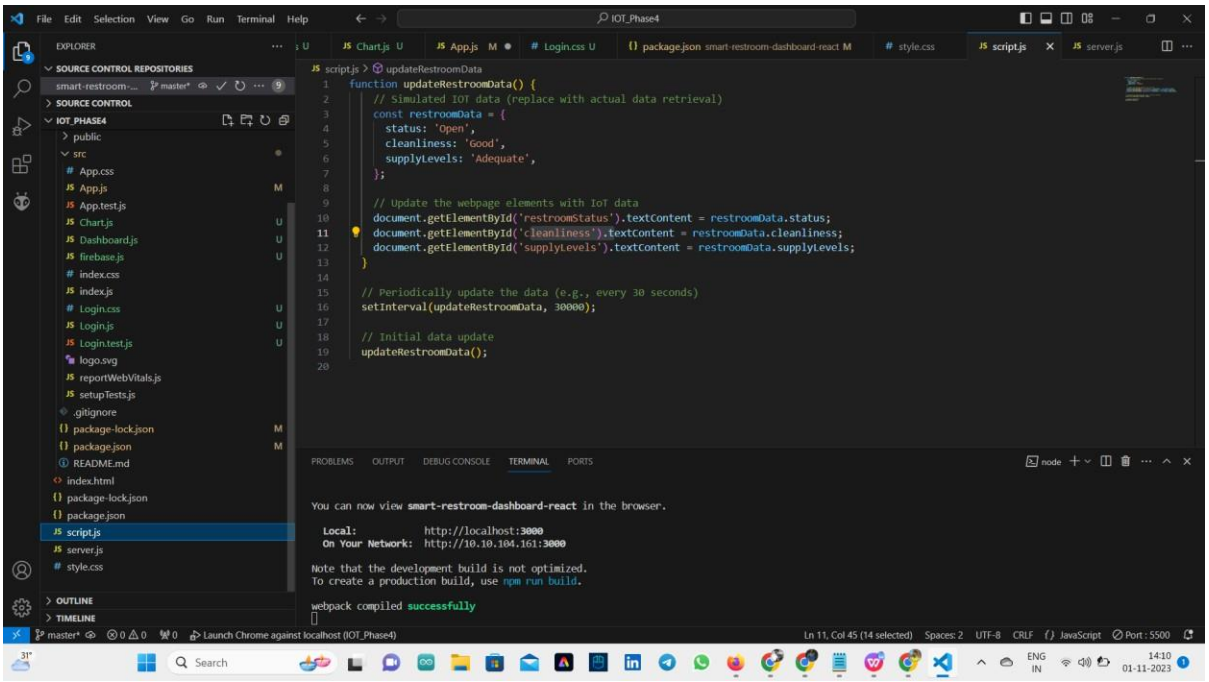
Login.test.js

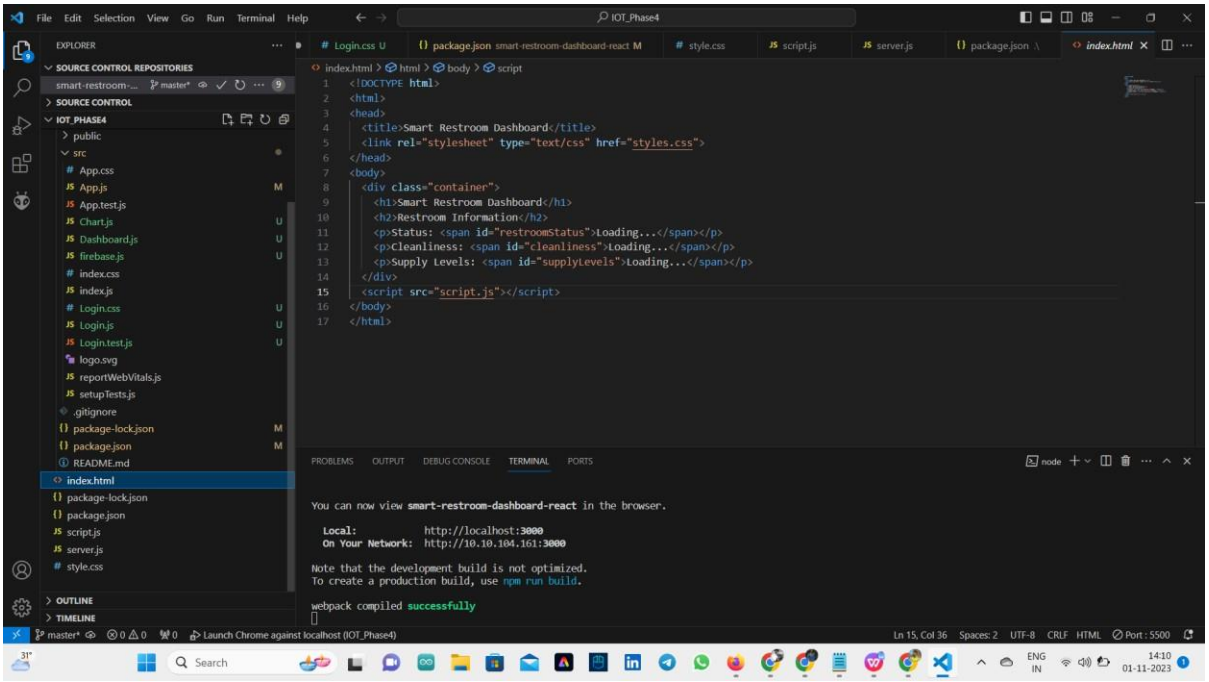




Server.js

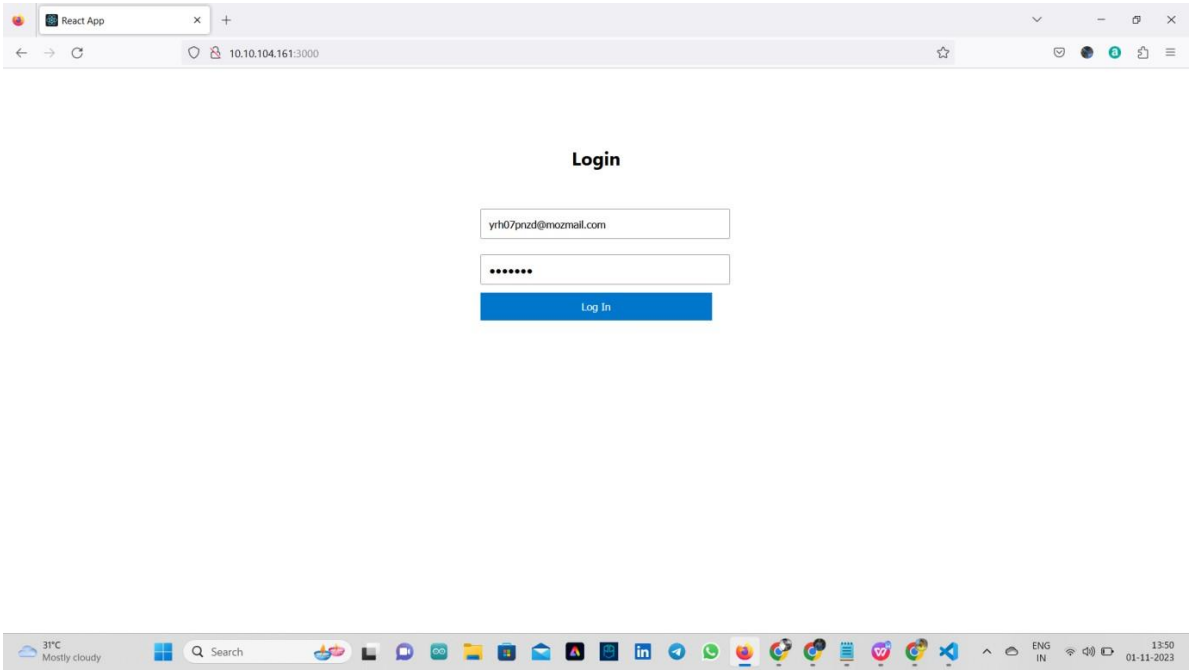
Script.js



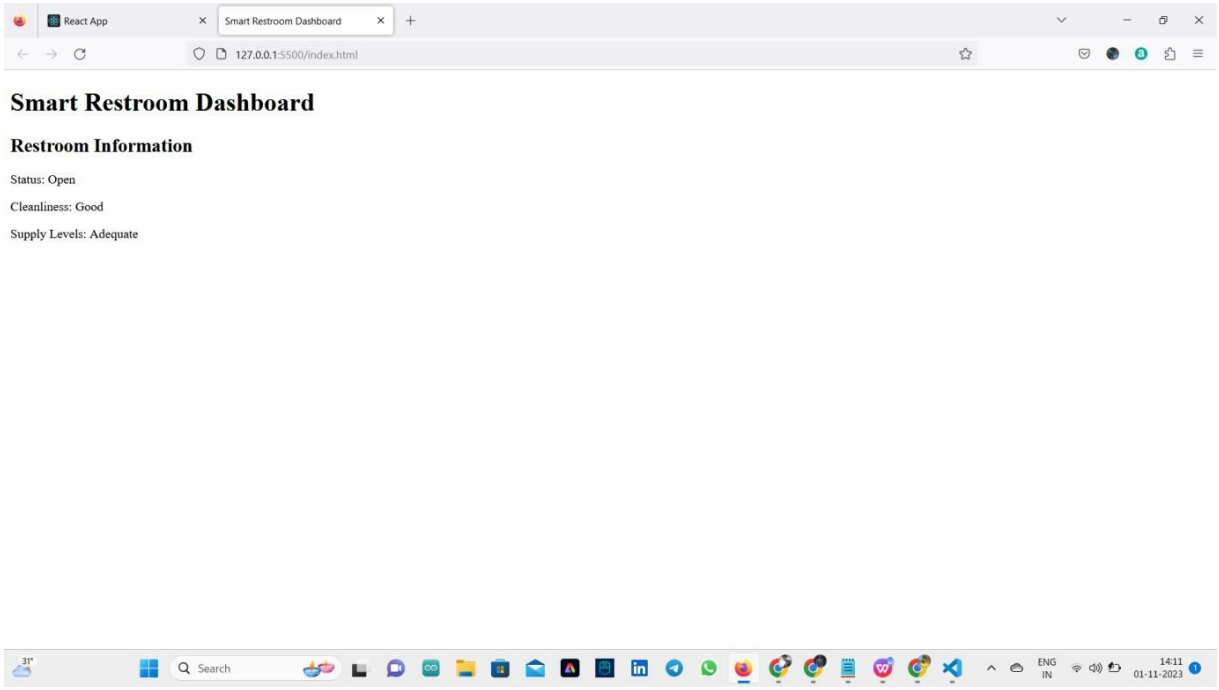


Index.html

Login page



Data Visualization



Note: Designing an app for a smart public restroom using IOT can improve user experience and efficiency, benefiting both restroom users and facility managers. Remember to stay flexible and adaptable, as IOT technology and user needs mayevolve over time.