**PROJECT REPORT**

# **Restaurant Management System**

Name of Group Members: Sarmad Jamal (19k-1116)

Mansoor Butt (19k-1114)

Khizer Jilani (19k-1057)

# 

DETAILED DESCRIPTION:

The Restaurant Management System that we try to deliver as our project is mainly based on the management problems that are most commonly faced by management systems. For those common problems we have designed a web application with a several features in it. The basic functionalities that our system is providing are:

* Order Monitoring
* Finance Monitoring
* Complain Monitoring
* Table Reservation

These four basic functionalities have been implemented on our system interface. Let’s discuss these functionalities in detail :

Order Monitoring:

Well order is the most basic feature in the restaurant management system. We have implemented order monitor interface that would help our administrator or management team to monitor the order and only they can have the access to monitor the order. Since order could be take away or dine-in, our interface have been designed in that way that it would support both kinds of order. The administrator can view both kinds of orders and can update them.

Finance Monitoring:

Finance Monitoring is important for running a restaurant management system. It would help administrator to go through the expense and revenue of the past months and on the basis of those administrator could change his business policy. After every order added to the system there would be change of revenue and expense in the that month analysis. On the basis of the result of expenses and revenue the system analysis will also show a pictorial description of loss or profit.

Complain Monitoring:

To know about your user feedback and reviews is importance for an organization to run. We have provided an interface that would help administrator or management team to monitor the complaint. It would help our organization to grow and develop. Once the customer launches a complain it would be seen on the administrator interface in immediate time

Table Reservation:

In our Restaurant Management System, we have provided a book table functionality in order to provide administrator the knowledge about the tables that are currently reserved or non-reserved. If customer requests for a table, then administrator will look for the unreserved tables and if there is none it would apologize the customer otherwise, they will assign the customer table.

**RMS Core Features:**

Customer Panel :

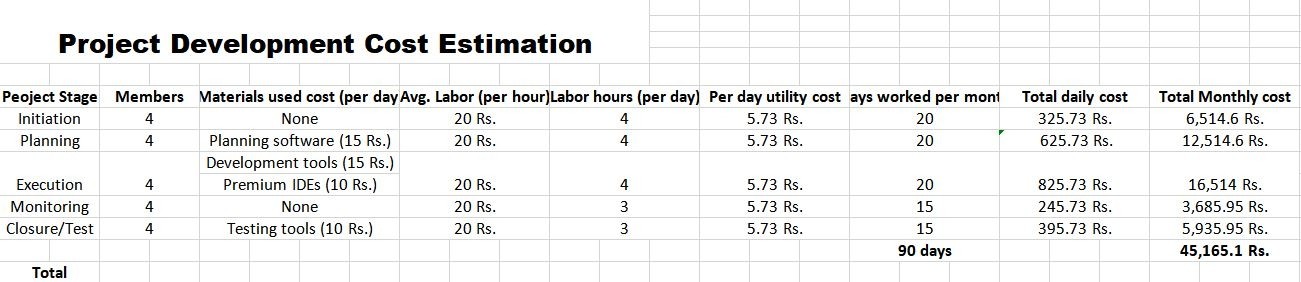
1. Add a Order
2. View Order Status
3. Order History
4. Table View
5. Table Reservation
6. Booked Table
7. Add a complain to a particular Order
8. View Complain Status
9. Add rating to a order
10. Pay the order Bill
11. Editing the info about the customer himself

Admin Panel :

1. View Order
2. Accept Order
3. Status of Order
4. View Complain
5. Resolve Complain
6. Status of Table
7. Adding a Table
8. Editing chairs in a Table
9. Viewing rating
10. Notify user by message using Twilio about the table booking
11. Viewing registered customers
12. Editing price of a food item

# 

Cost Estimation



Gantt Chart

**--**



### For high resolution view: [Click here](https://miro.com/app/board/o9J_lQqLKVw%3D/)

# Project Feasibility Report

## Feasibility Study

* Motivations

In the restaurant management system, all features required for managing a restaurant are available but for management side. It can help management to easily manage restaurant and carry out various tasks like take orders, assign orders to riders etc. This app is distinct from previous existing restaurant apps because it is completely designed for management to make their tasks easy. It is a web app and can be accessed across any platform.

## Input and Assumptions

The system takes the following inputs:

* + Experienced worker chefs
  + Best quality material
  + Project management system
  + Budgeting
  + Costing
  + Technology
  + Production

## Problem Definition

Following are commonly faced problems:

* Many a times, we have to wait in long queues for the orders, which in return waste a lot of time and requires greater level of patience.
* Sometimes an order can be missed.
* Many human errors occur when we deal with information manually.
* There are many benefits for online restaurant management including customer satisfaction, better product quality and fast service.

## Objective

The software we are proposing have many enhance features then previous software, as going through pervious software lack of many features are there which cause lacking behind for providing good services.

1. Order monitoring
2. Complains monitoring
3. Table Reservation
4. Customer Booking

## 1. Project Economic Feasibility:

The purpose of economic feasibility of project is to determine the economic benefits of proposed system. The proposed system is designed to maximize saving resources I.e., both money and time. Following table shows expected profitability.

|  |  |  |  |
| --- | --- | --- | --- |
| Cost | Year 0 | Year 1 | Year 2 |
| Development Cost |  |  |  |
| Hardware | 30,000 |  |  |
| Software | 15,000 |  |  |
| Training | 165 |  |  |
| Total | Rs. 45,165 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Production Cost |  |  |  |
| Supplies |  | 450 | 450 |
| Personnel |  | 5,600 | 6,048 |
| Maintenance |  | 1,500 | 1,500 |
| Annual Production Cost |  | 7,550 | 7,998 |
| Present Value |  | 6,864 | 6,610 |
| Accumulate Cost |  | 50,364 | 56,974 |

|  |  |  |  |
| --- | --- | --- | --- |
| Benefit | Year 0 | Year 1 | Year 2 |
| Labor savings |  | 27,000 | 28,350 |
| Increase sales |  | 48,000 | 52,800 |
| Total |  | 75,000 | 81,150 |
| Present Value |  | 68,182 | 67,066 |
| Accumulate  Benefit |  | 68,182 | 135,248 |
| Gain/Loss |  | 17,818 | 78,274 |
| Profitability Index | 4.52 | Feasible |  |

## Technical Feasibility

The system uses many materials and supplies. It will be developed using latest and well-developed IDEs and tools that are used by many tech giant organizations. The system is a web based and consists of many features. It has a front end and back end which is in HTML, CSS and JavaScript along with SQL Database. The ideas for the system are well defined and tested. All the features of the system perform a distinct function and the system itself will be simple to use and a pleasing experience alongside the profitability it will provide. All the members of a group are tasked with a specific component of a system according to their capability.

The system will be technically modern and up-to date and will require a very simple knowledge and basic training to be an expert. All the difficult tasks in previous management systems are hoped to be made simple enough to be used easily by a newbie using this system.

However, the system maybe prone to online cyber-attacks or data theft/data loss which may result in customer dissatisfaction or loss of money, for this problem, a cybersecurity expert and data analyst should be hired or any other person with technical background. The restaurant must hire maintenance engineers by its own too because the system is online and will require a maintenance to work well.

The system will require a decent internet connection 24/7 to work and will host a separate admin dashboard which can be used by only one person. Hardware and software requirements for the system are not too much, it will run fine on a decent machine with 8GB of Ram and other specs that can run a browser on which system will be used.

## Operational Feasibility

When talking about restaurants, there are many problems faced by people in restaurants but typical restaurant management website helps only customer to carry out their tasks and ignore restaurant staff. This system will provide every restaurant staff with a help and an easy way to carry out their tasks without wasting time and efforts.

The system provides stock control by which chefs can check how much stock of any supply is currently available and in need, they can order new stock.

When compared to manual, this feature is quite helpful as checking manually can cause human errors and waste resources, making it online will save time and money.

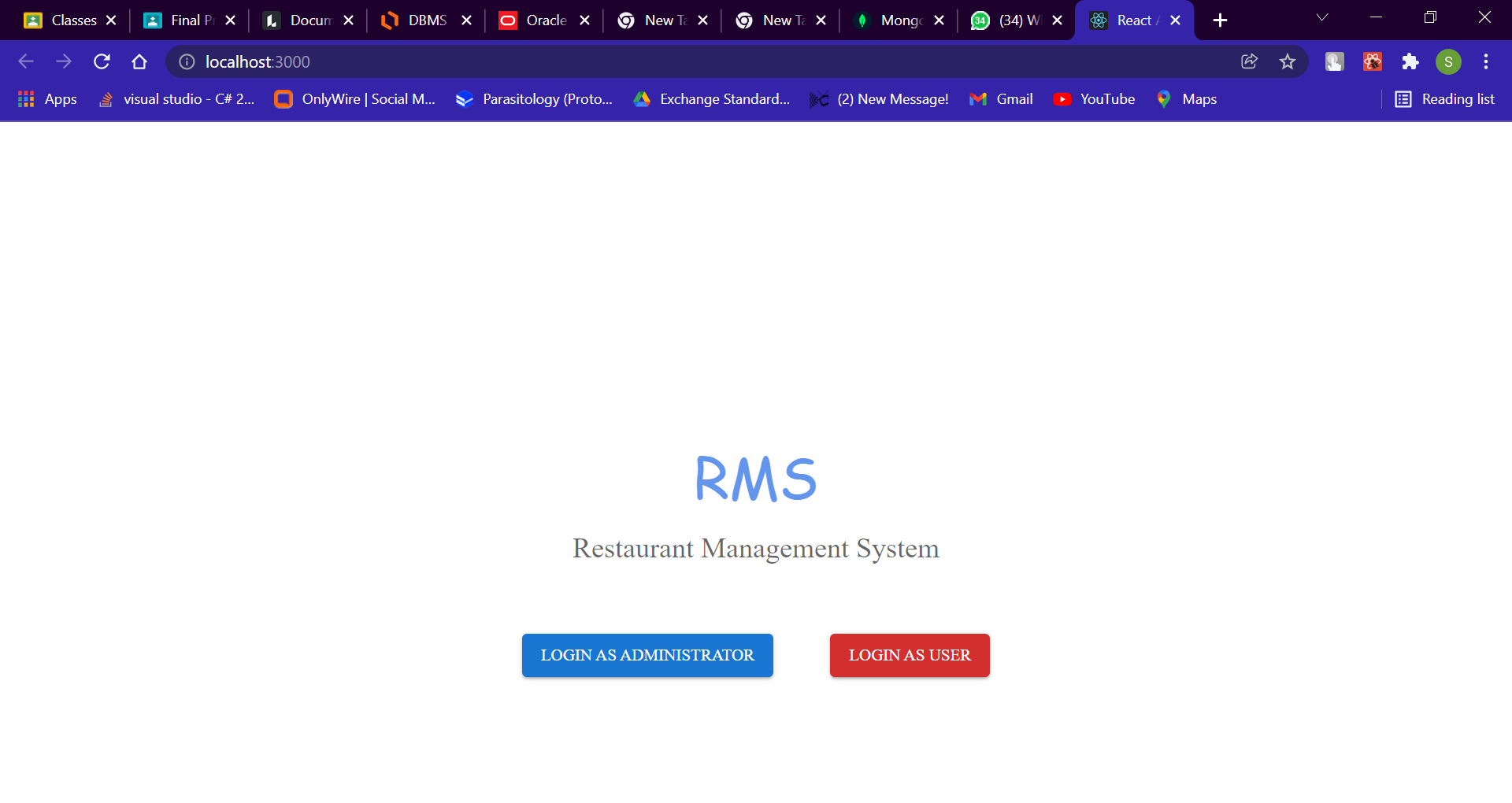
Admin can control restaurant staff using this system. It provides attendance system for waiters, janitors, manager and rider which will help in many way for example it can inform of crew shortage beforehand so that new staff could be

arranged. Furthermore, expenses and maintenance duties can be managed too

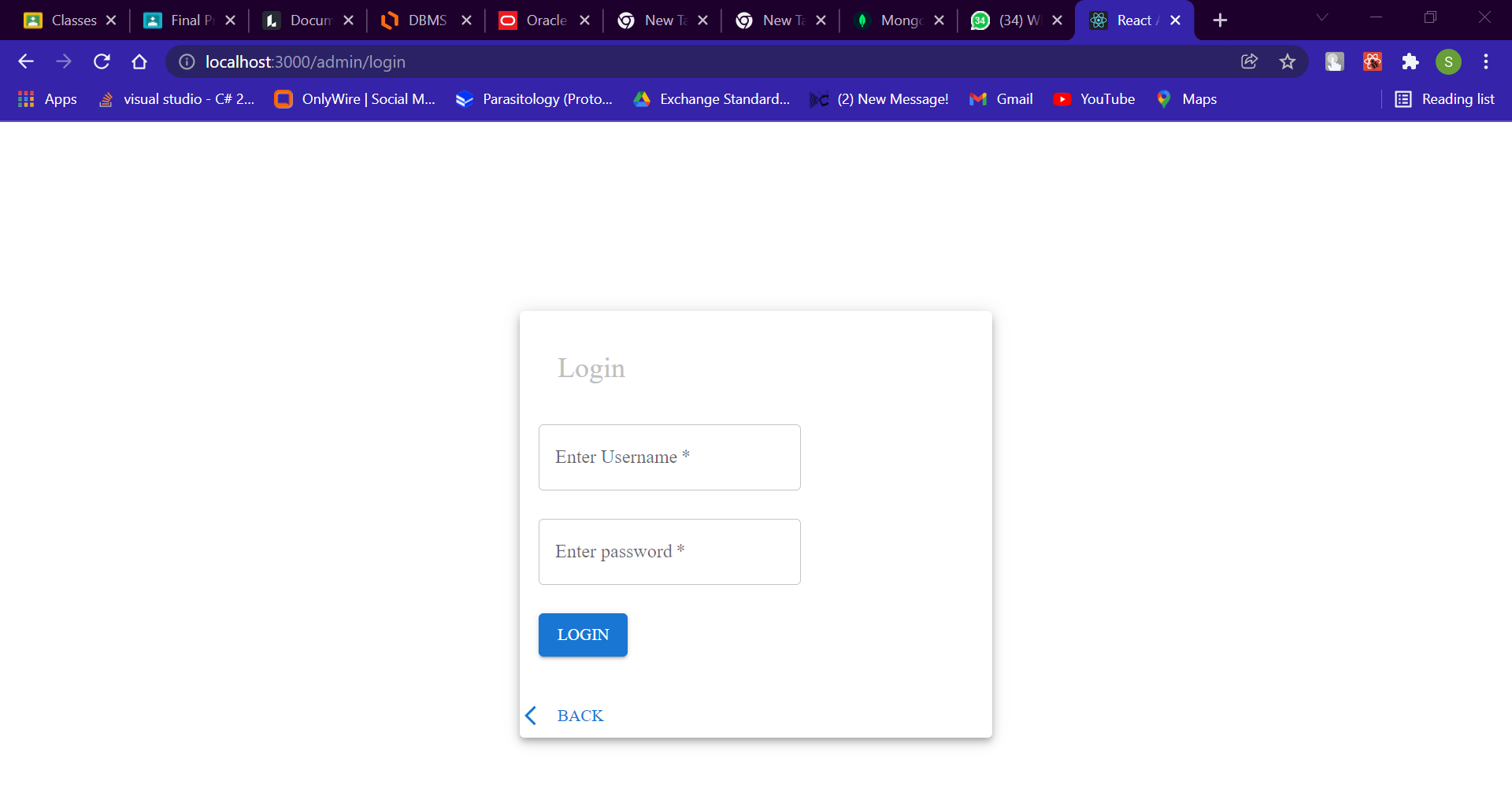
There is also a monitoring system for complains and order. Order can be accepted, viewed and its status can be tracked. People had to wait in a long queue to order which was a big problem and effected customer satisfaction, through this system, orders can be taken online saving both customer and chef’s efforts. Issues can be reported online too which can be viewed by respective staff and resolved. It can help to get information about user’s side.

# Snap shots

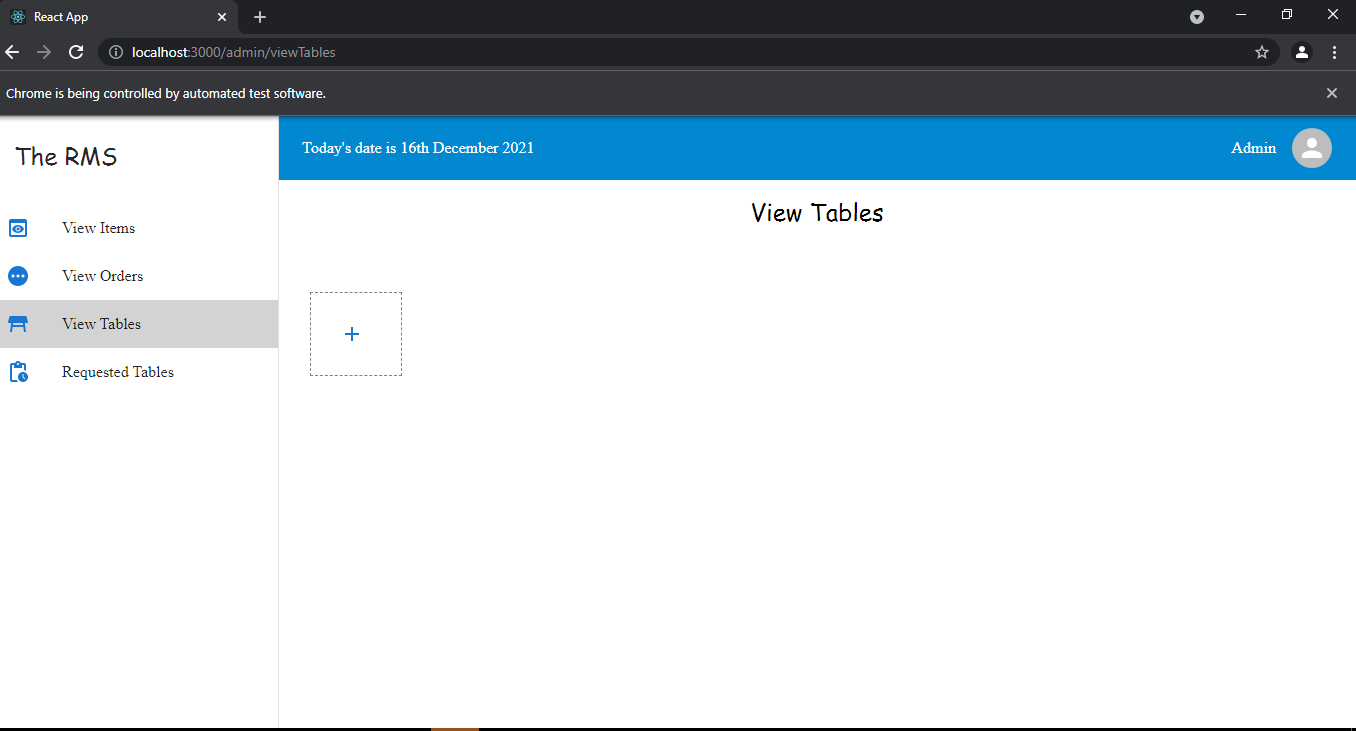
->HOMEPAGE

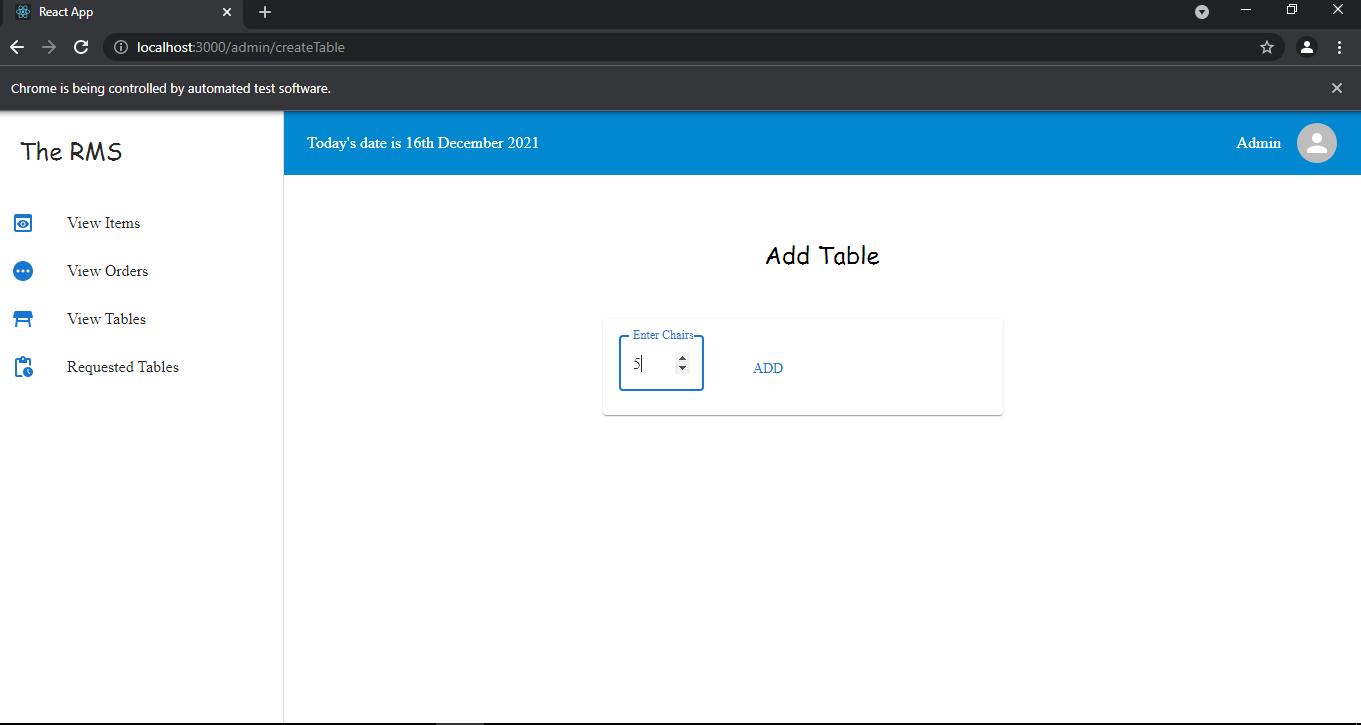


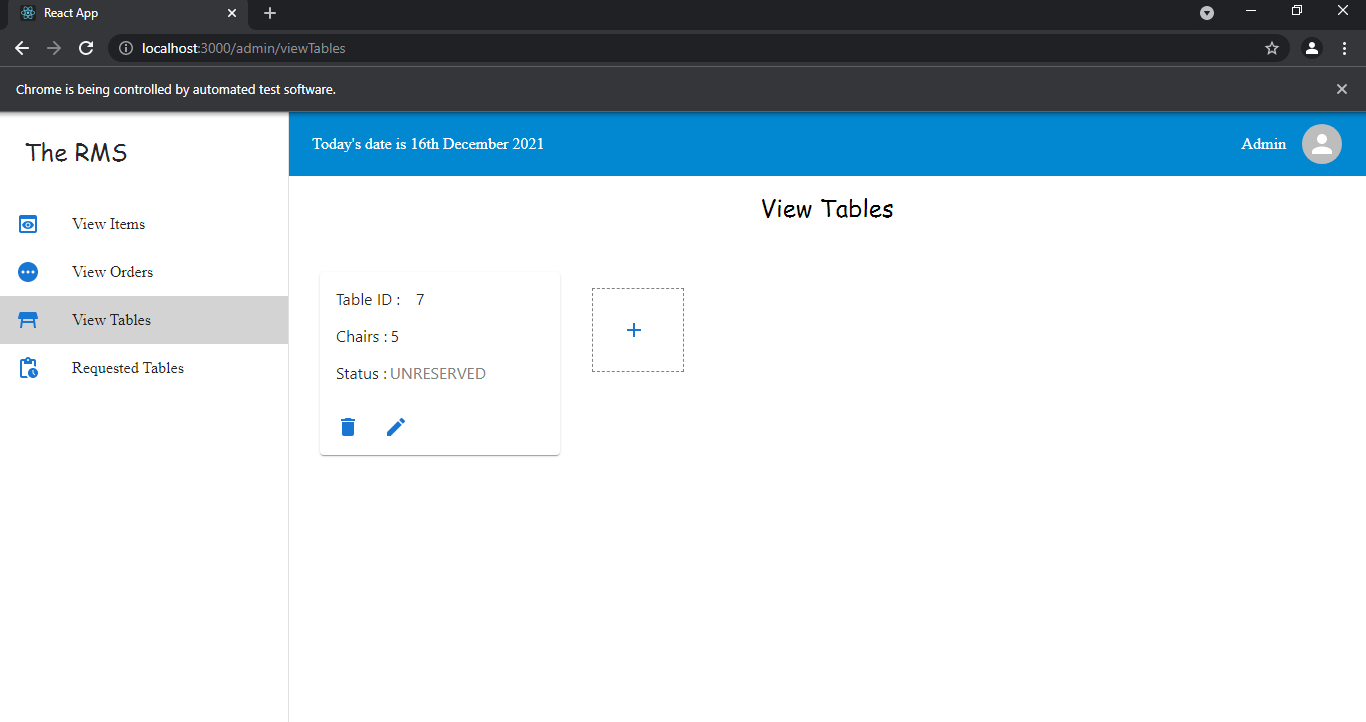
->Admin login Page



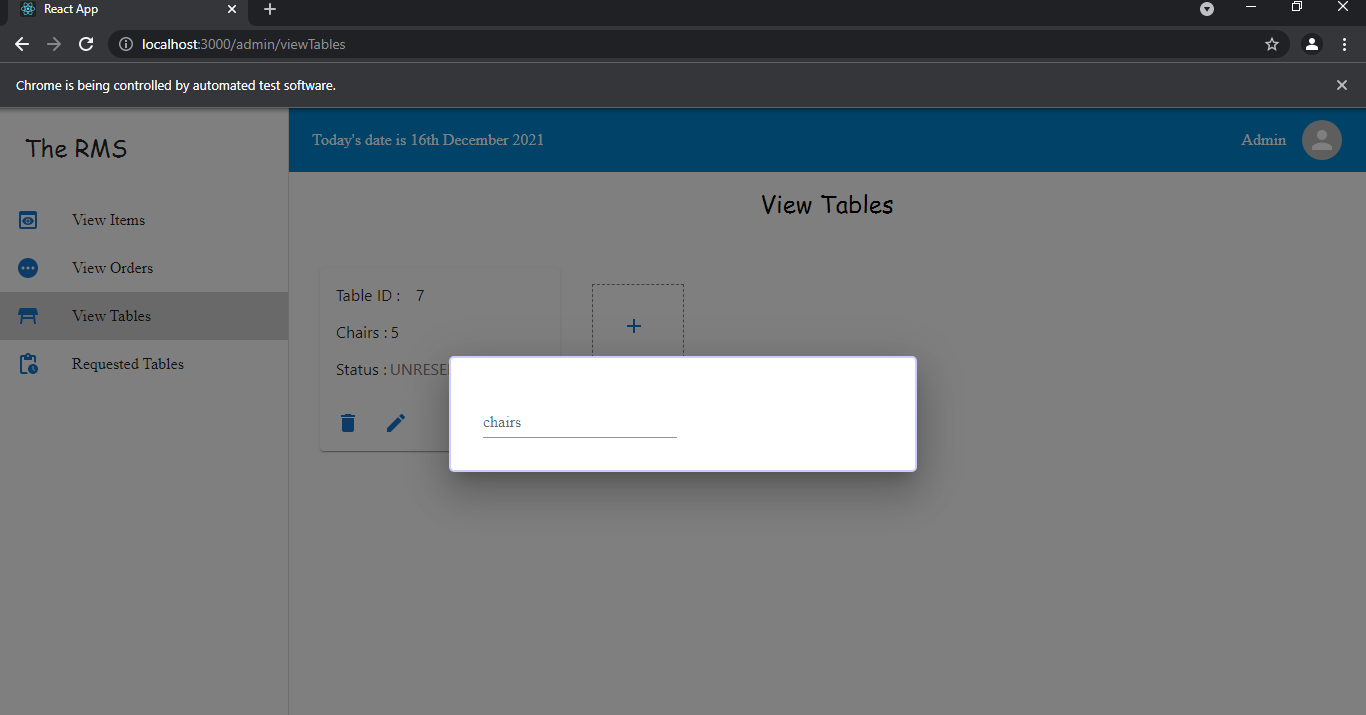
->Admin Adding tables

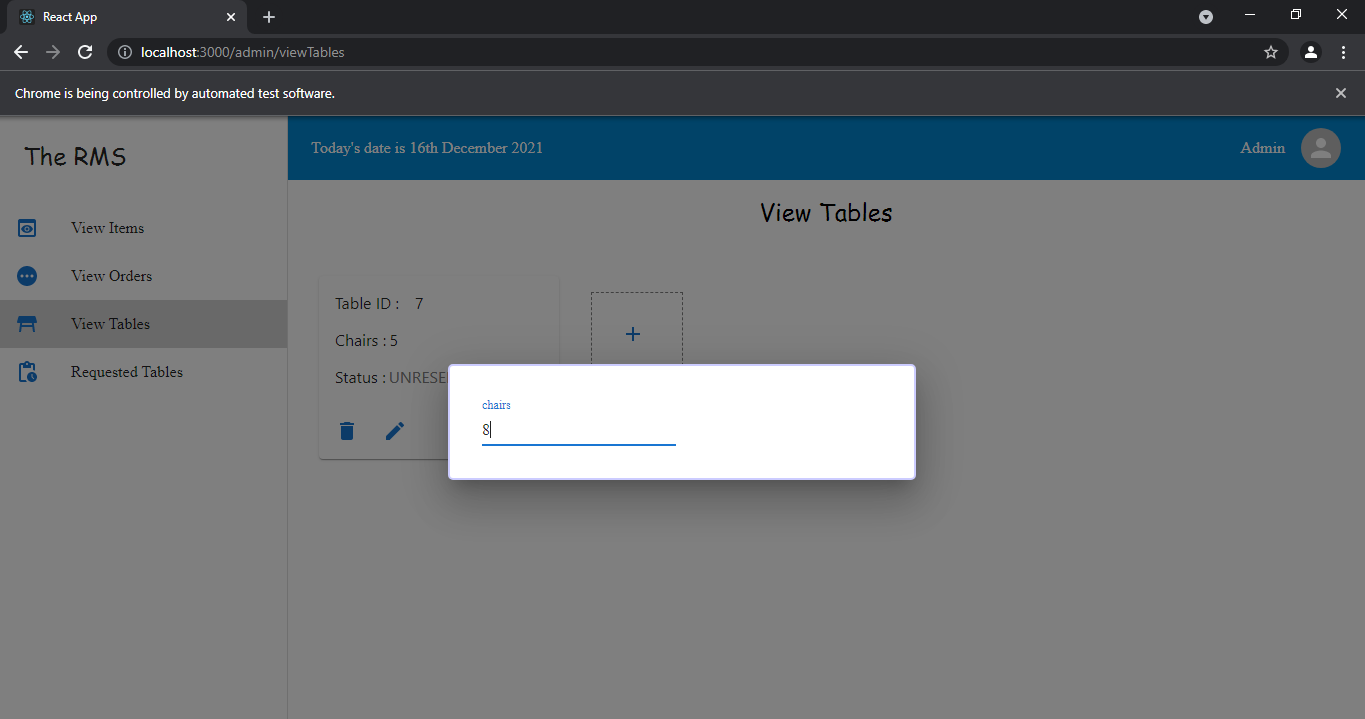


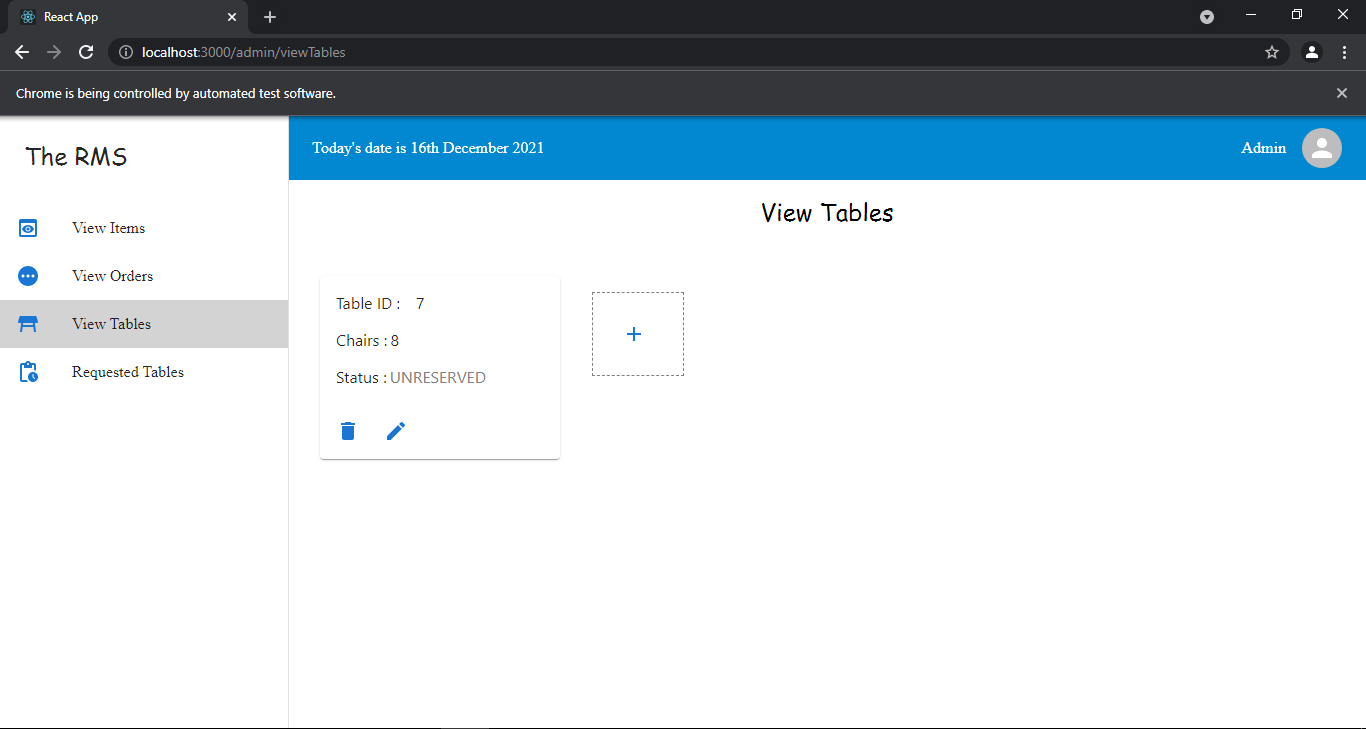




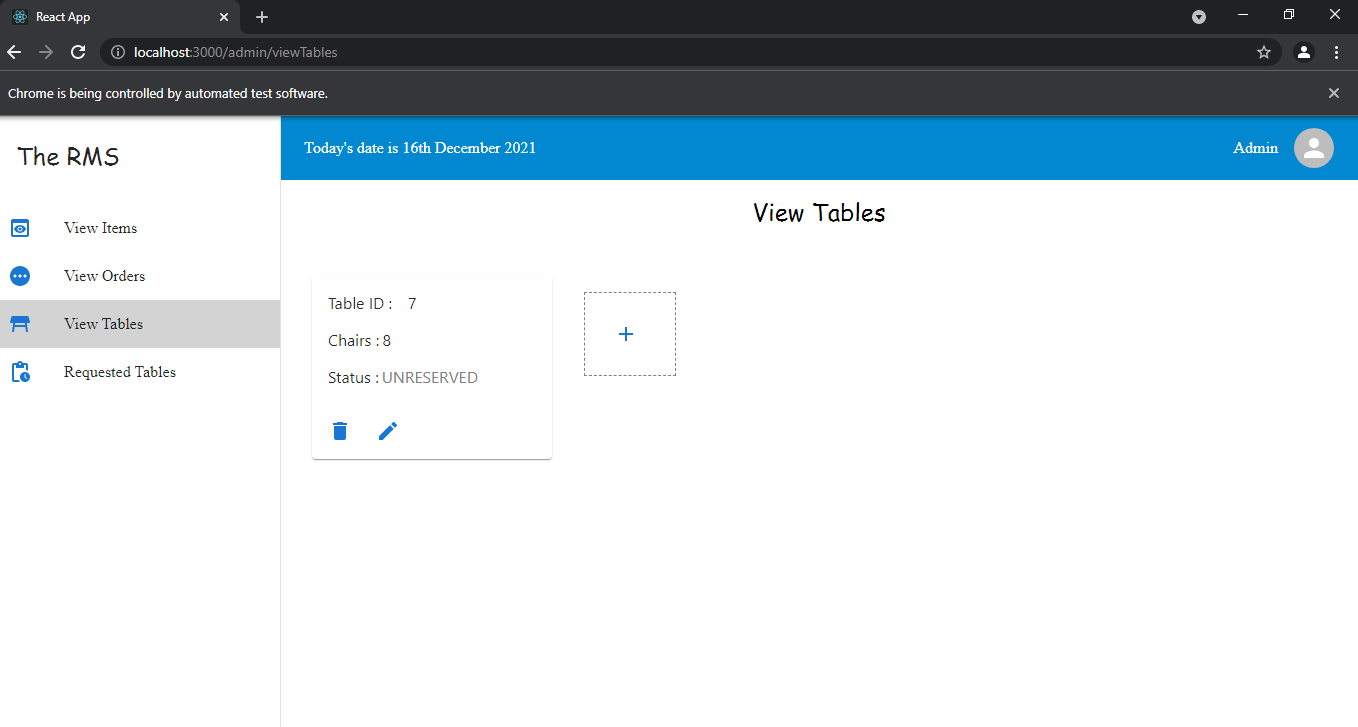
->Admin Updating Chairs

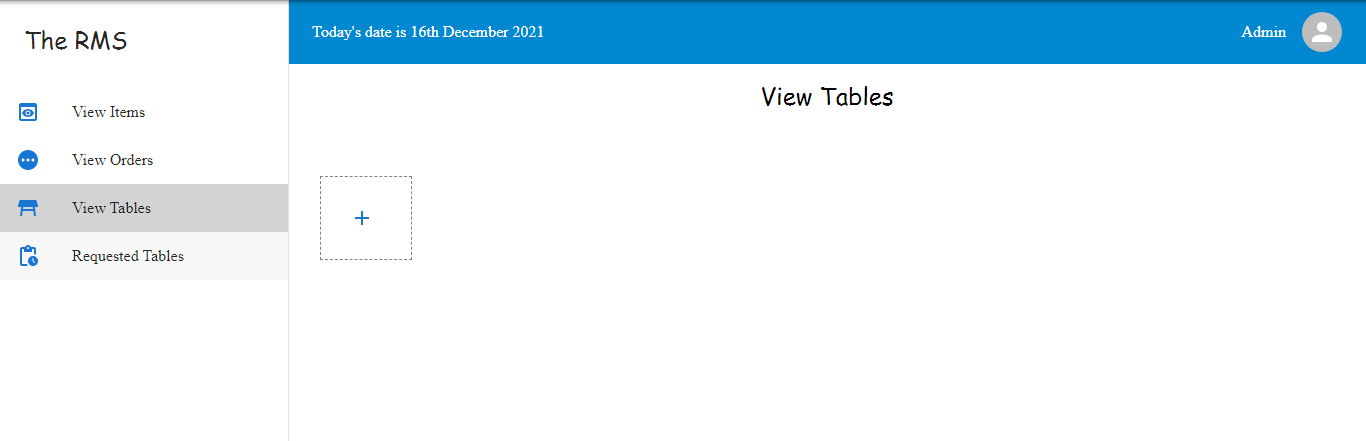




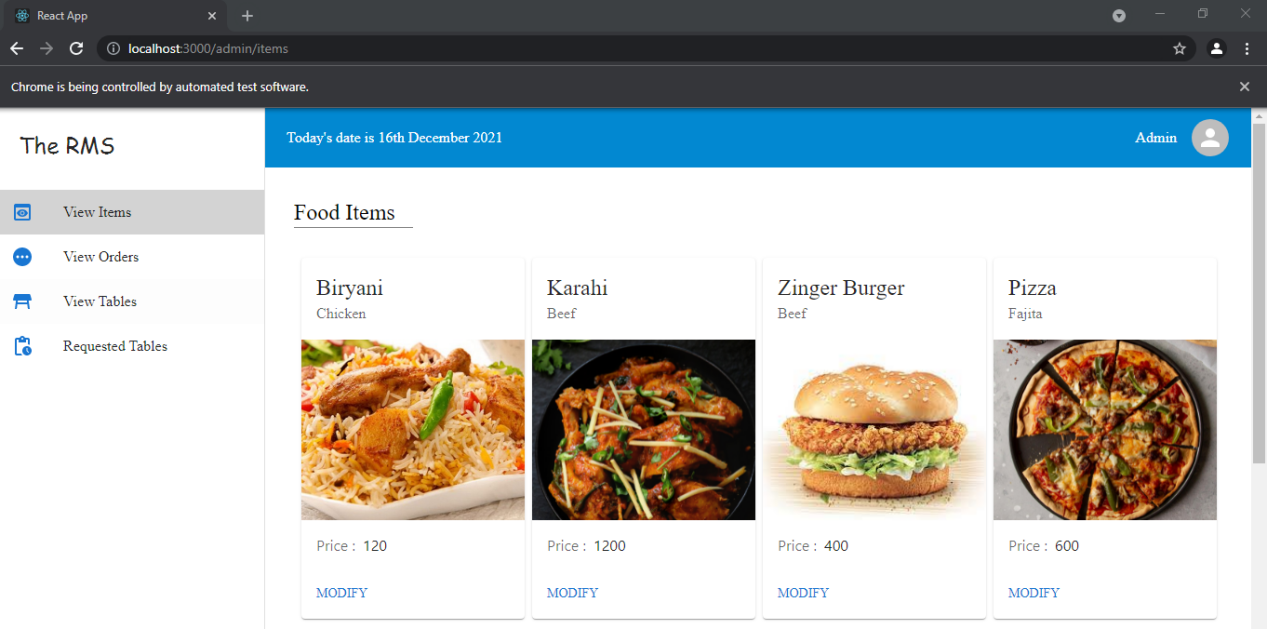


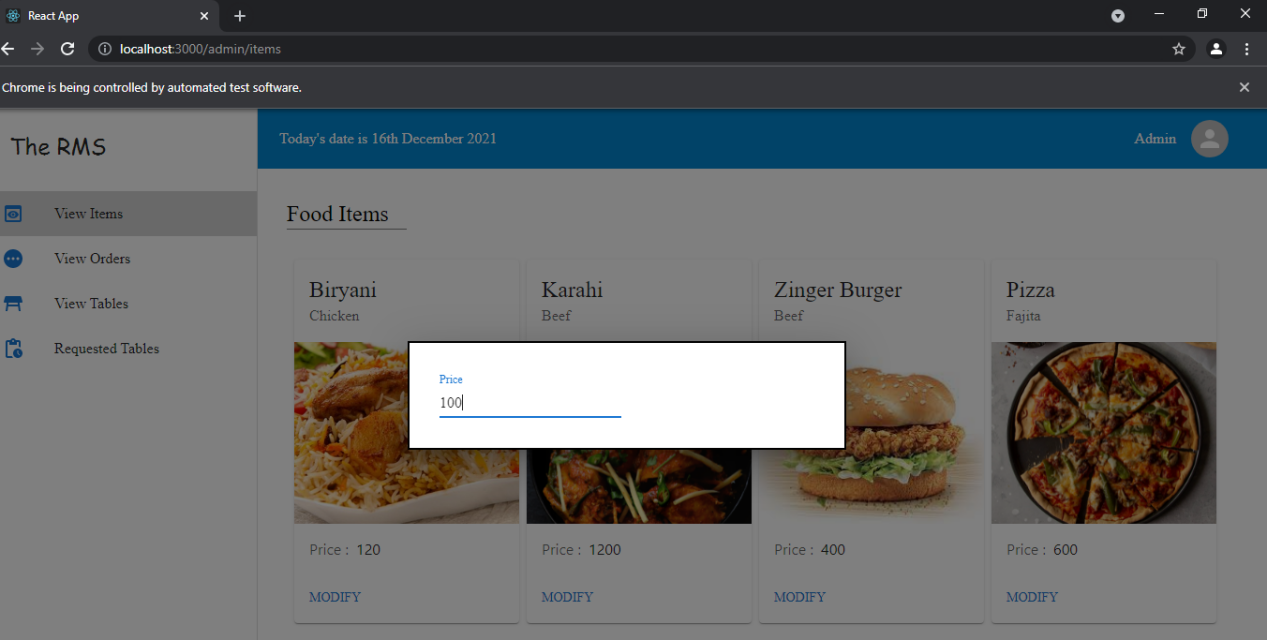
->Admin Deleting tables

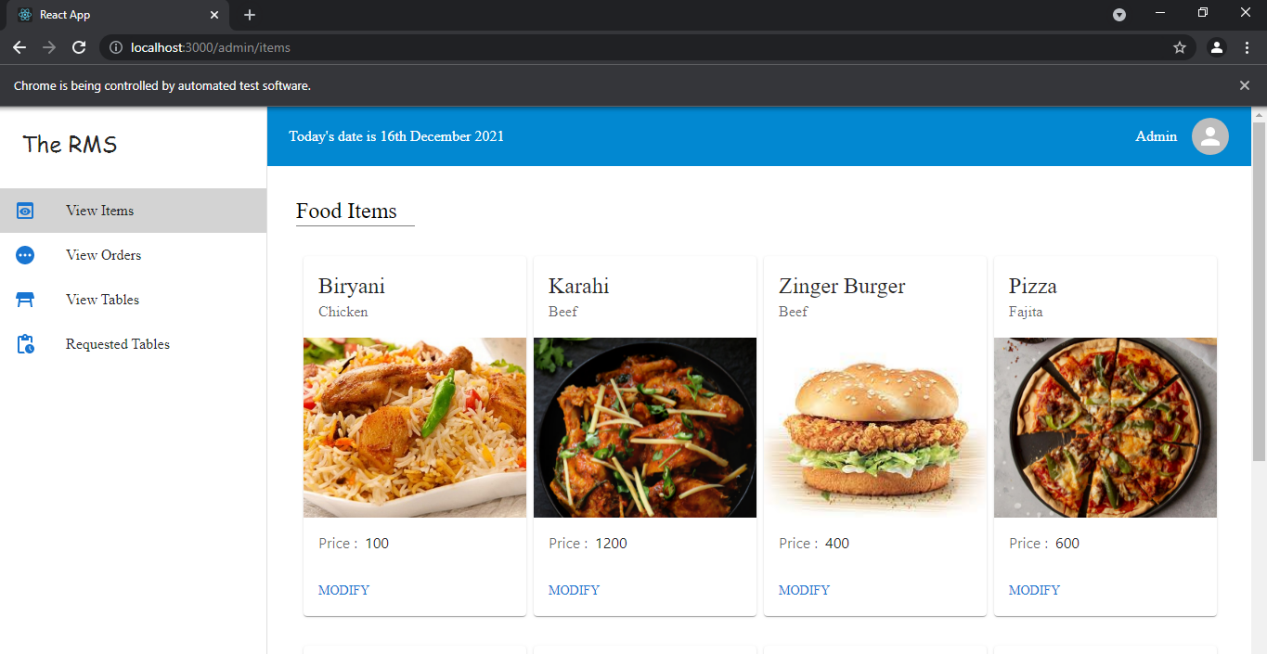




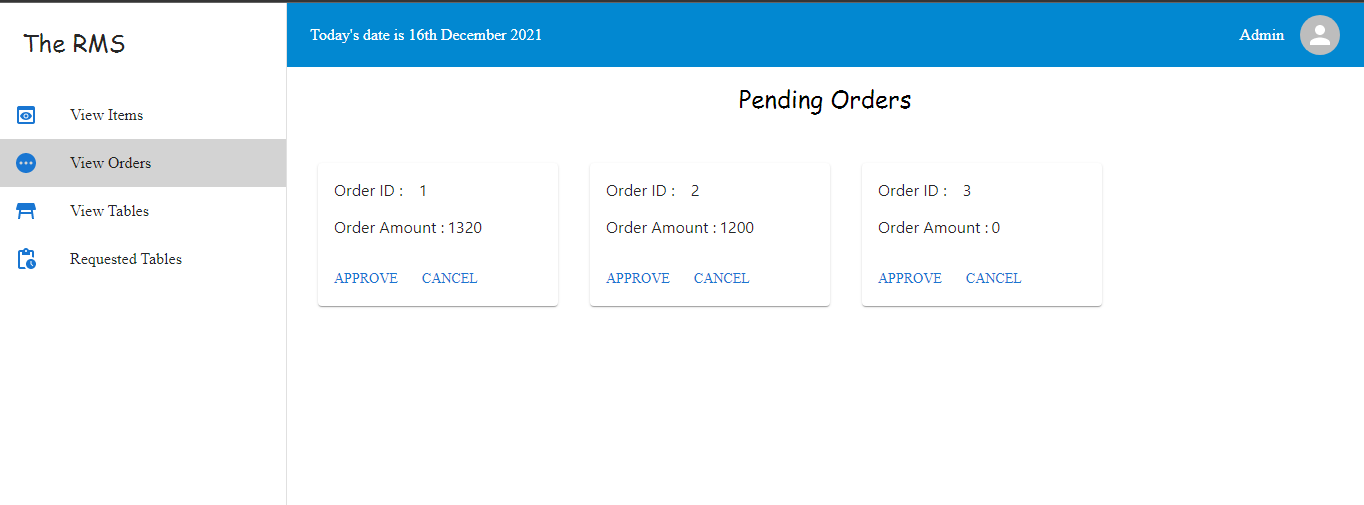
->Admin modifying Items price

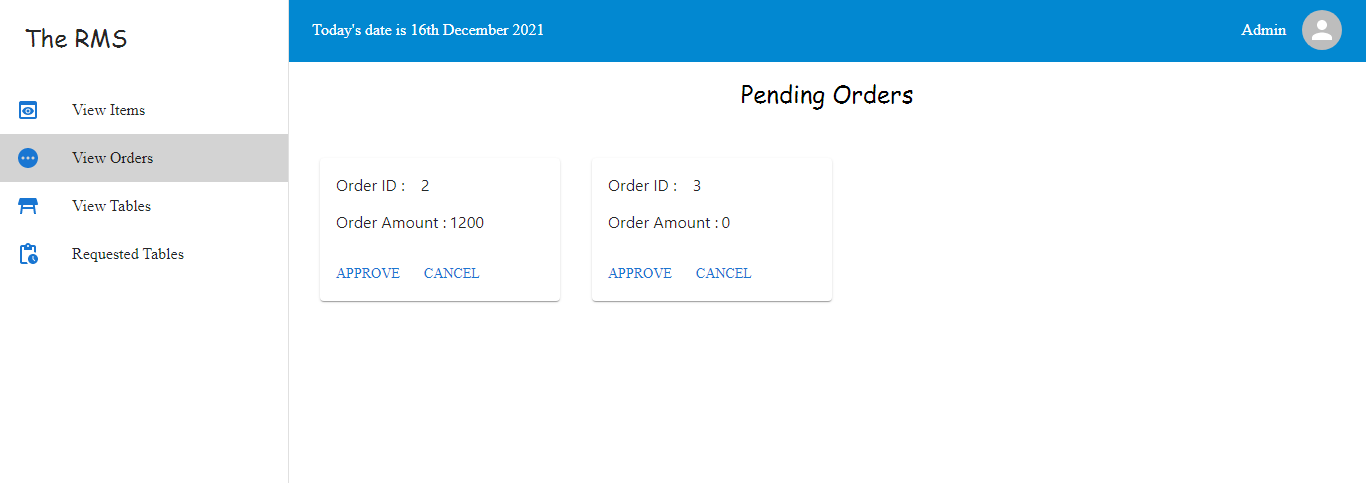




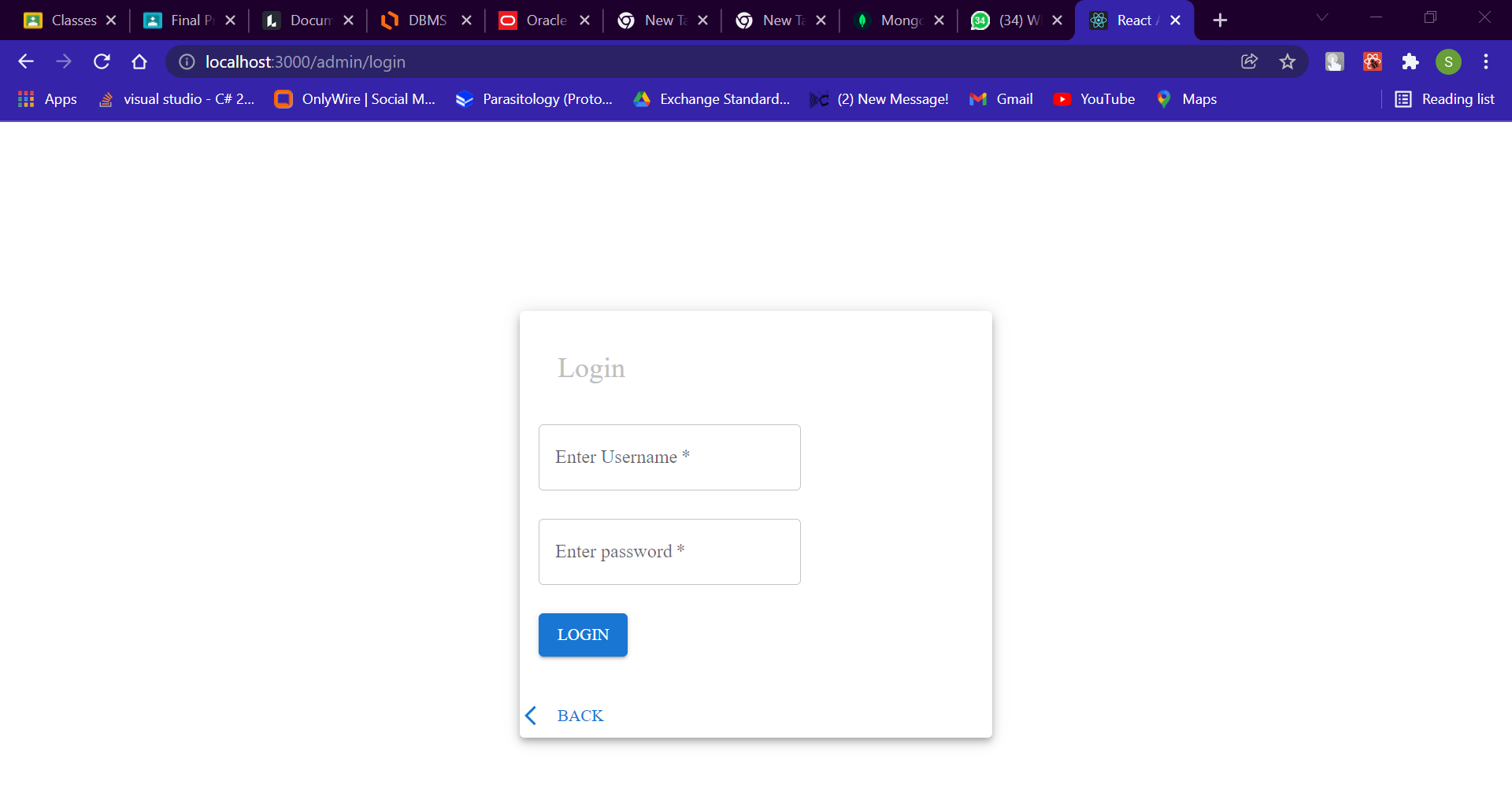


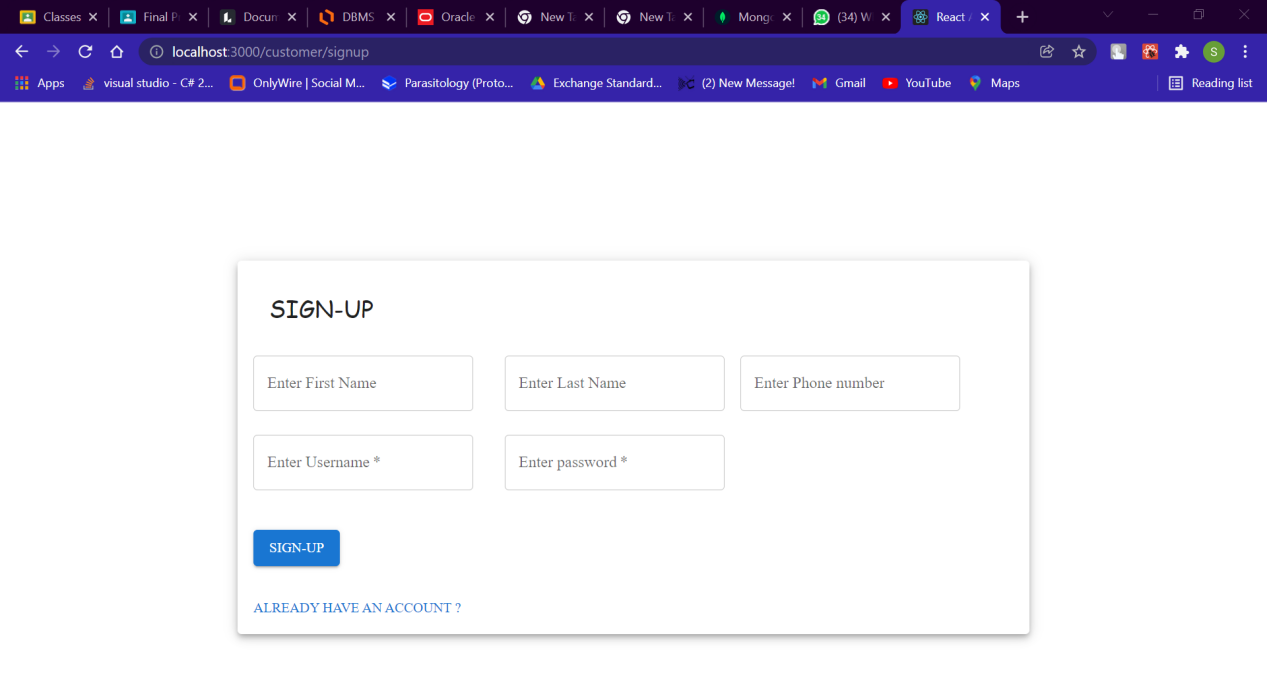
->admin approving Orders

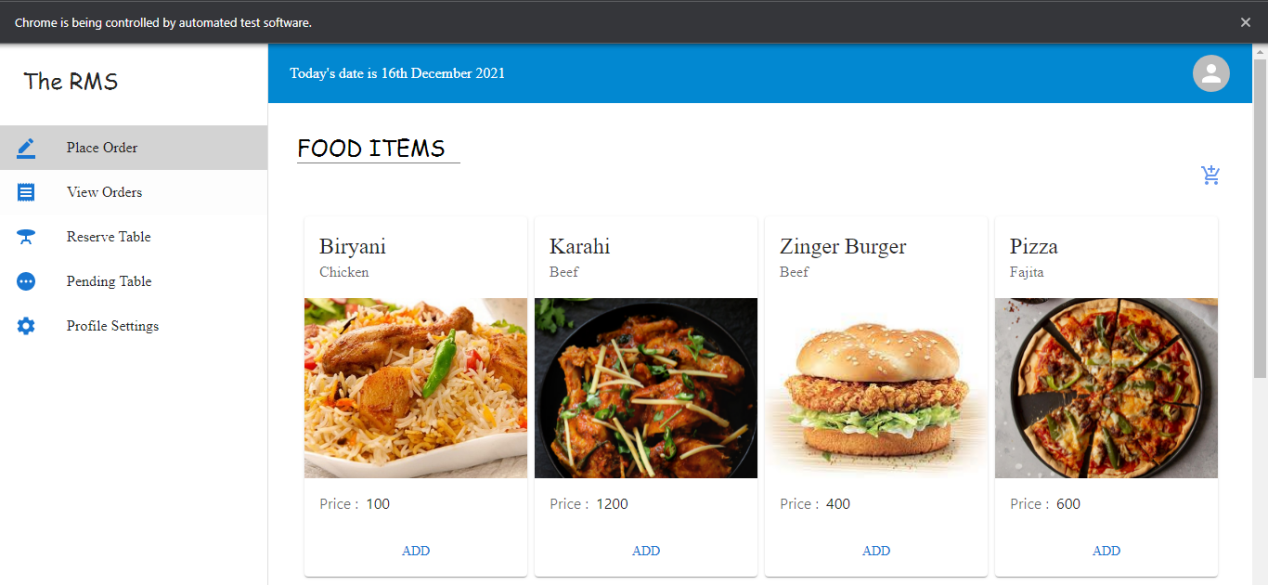


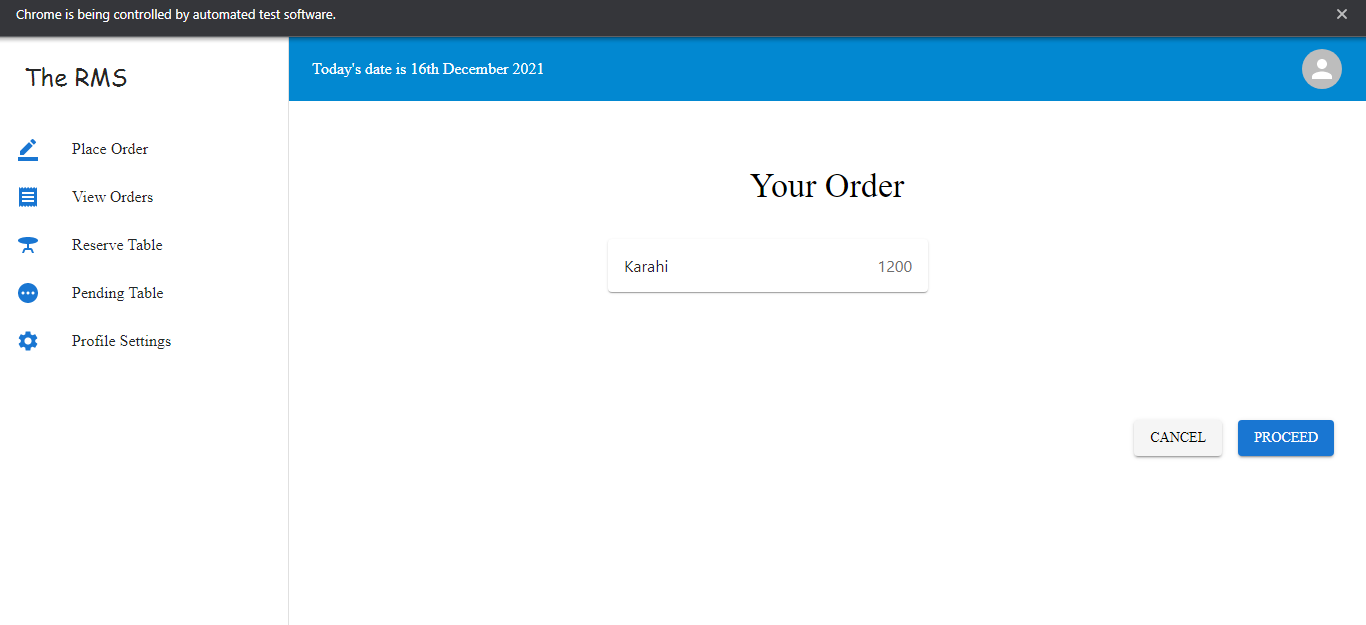


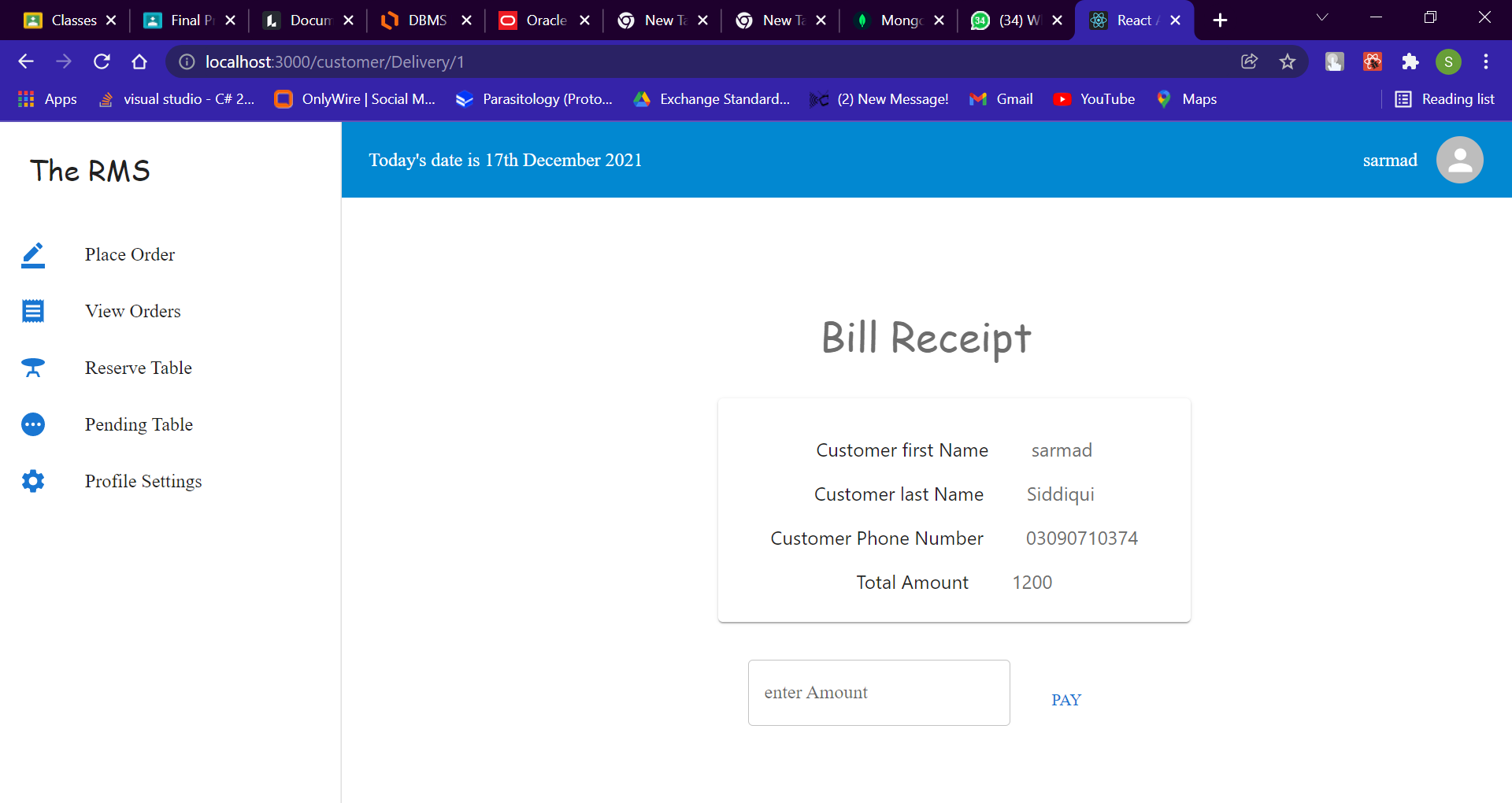
->Customer Flow of adding order

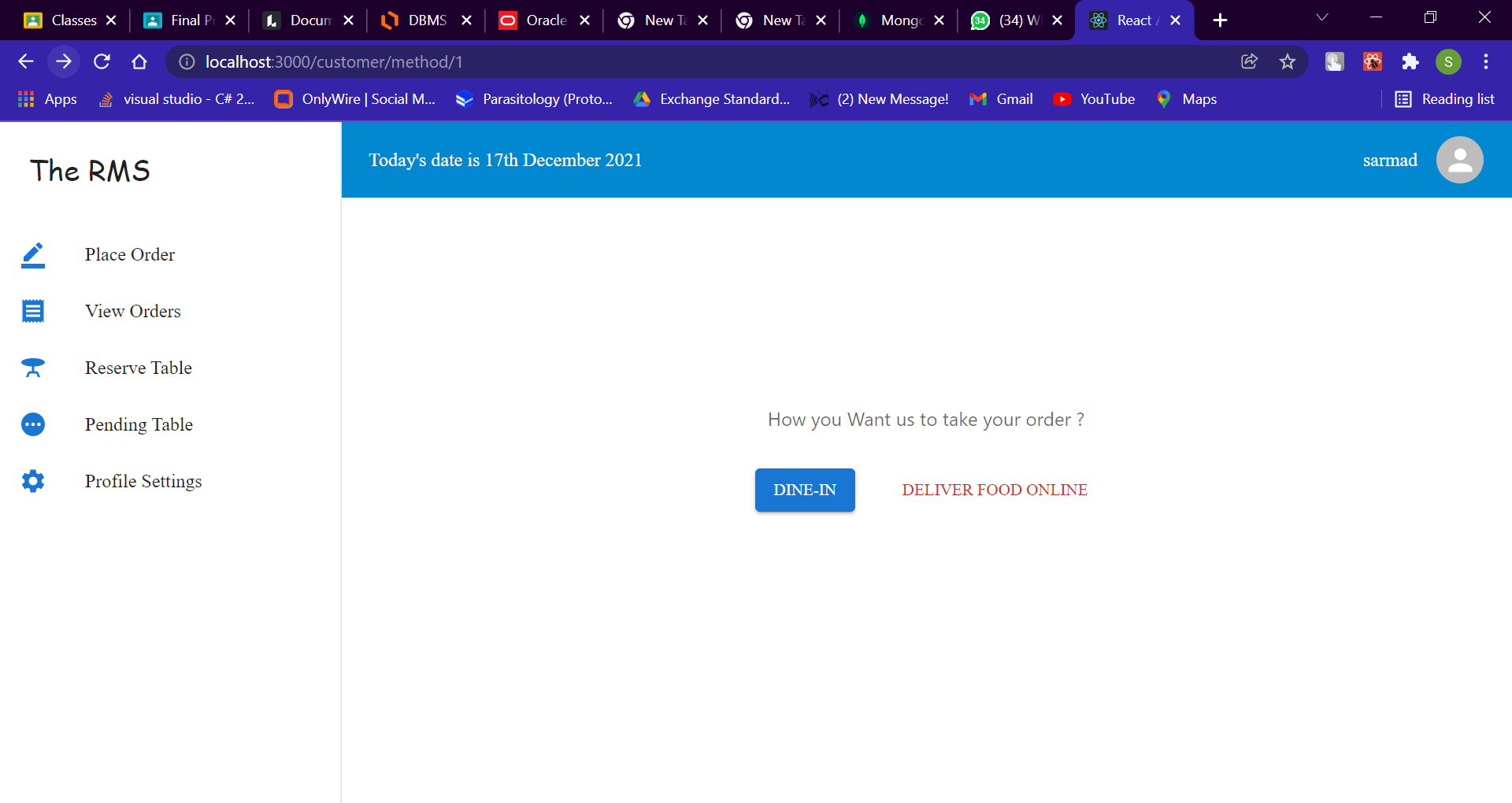




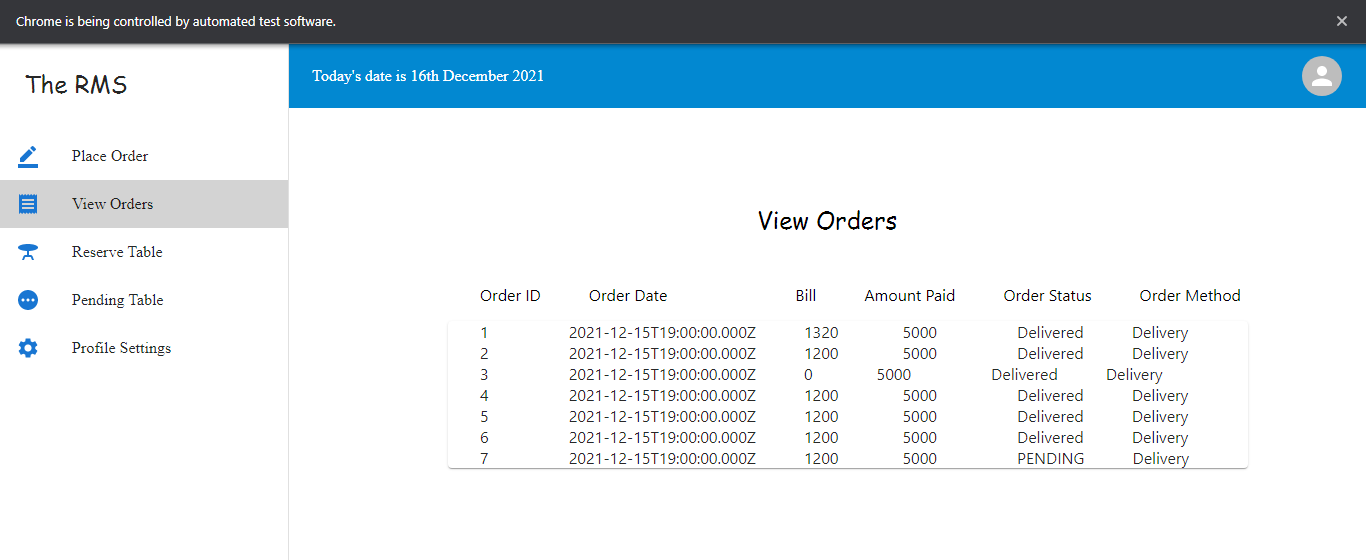


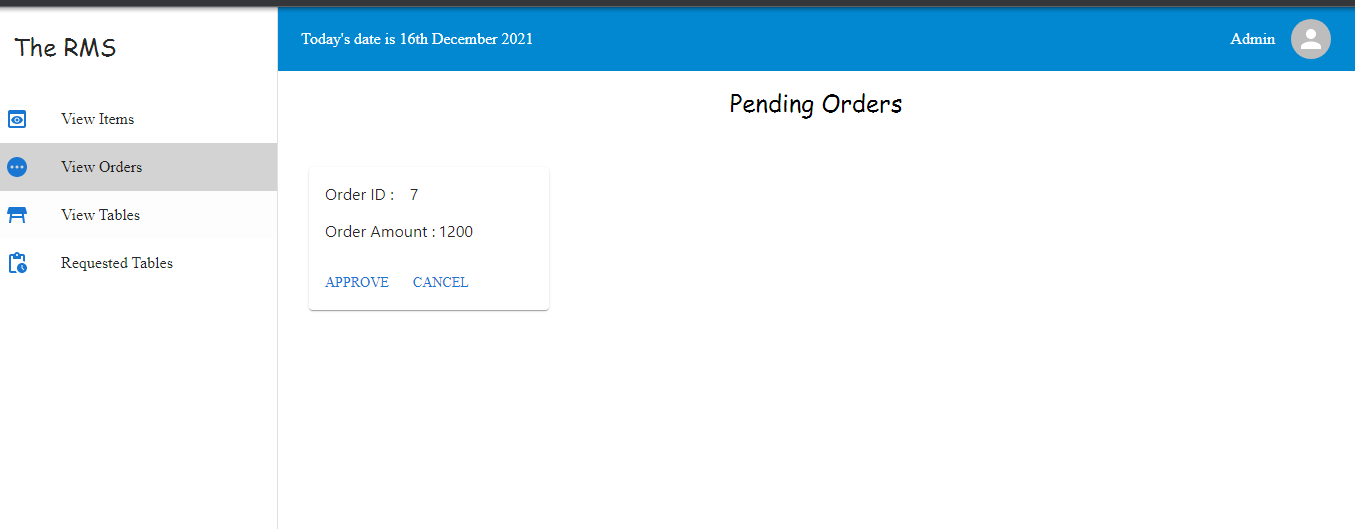




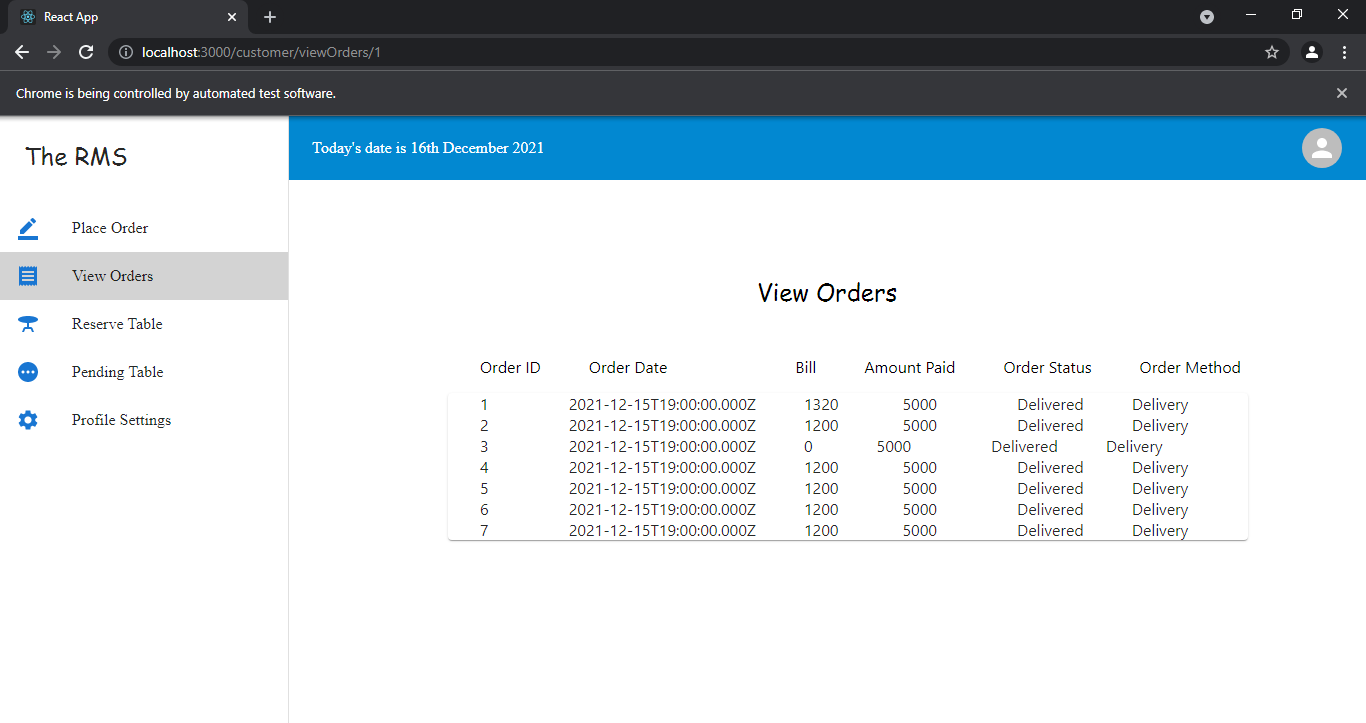


->Order is pending since admin has not approved it

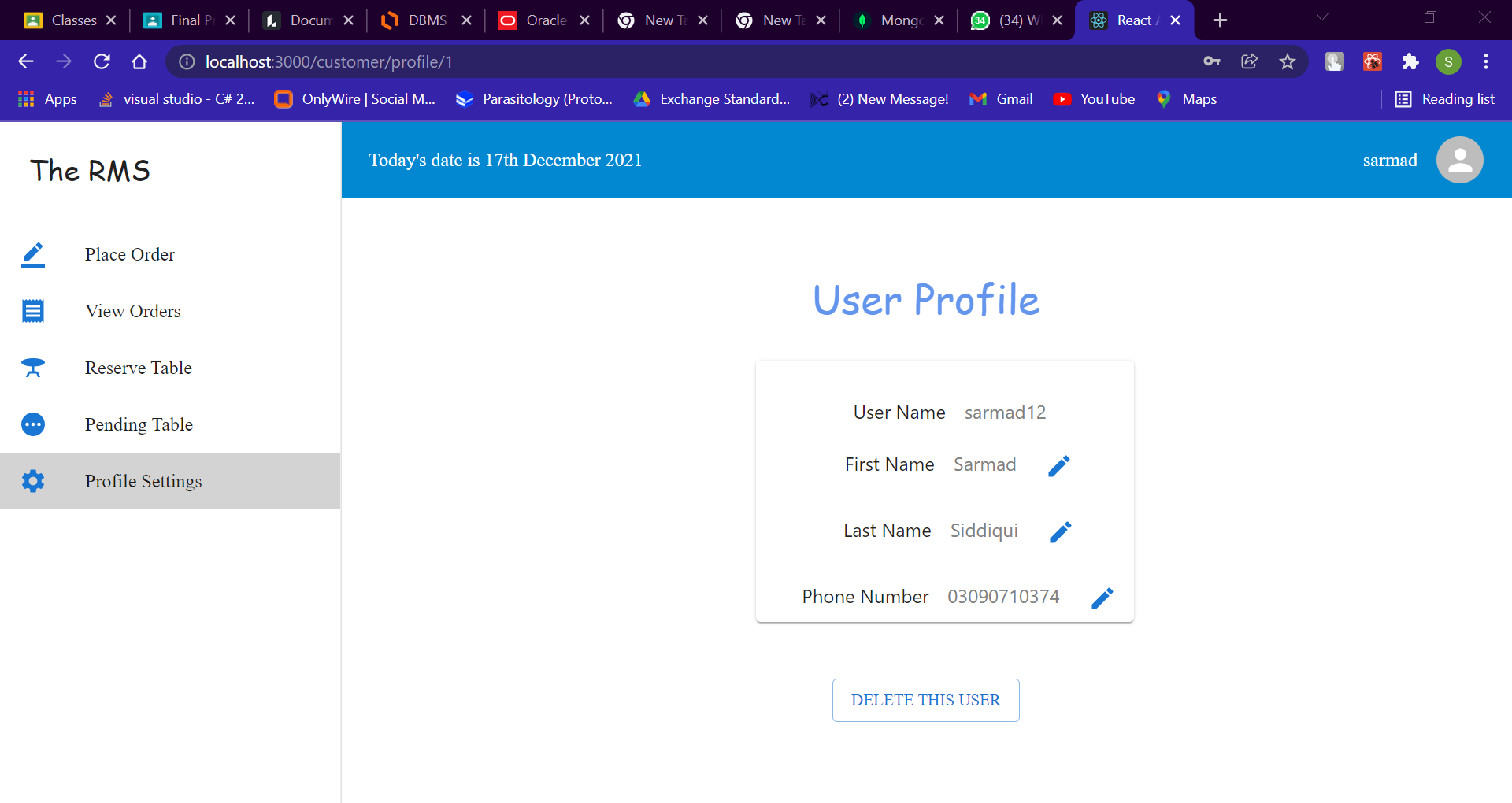




->Order is delivered since admin has approved it



-> Customer Profile

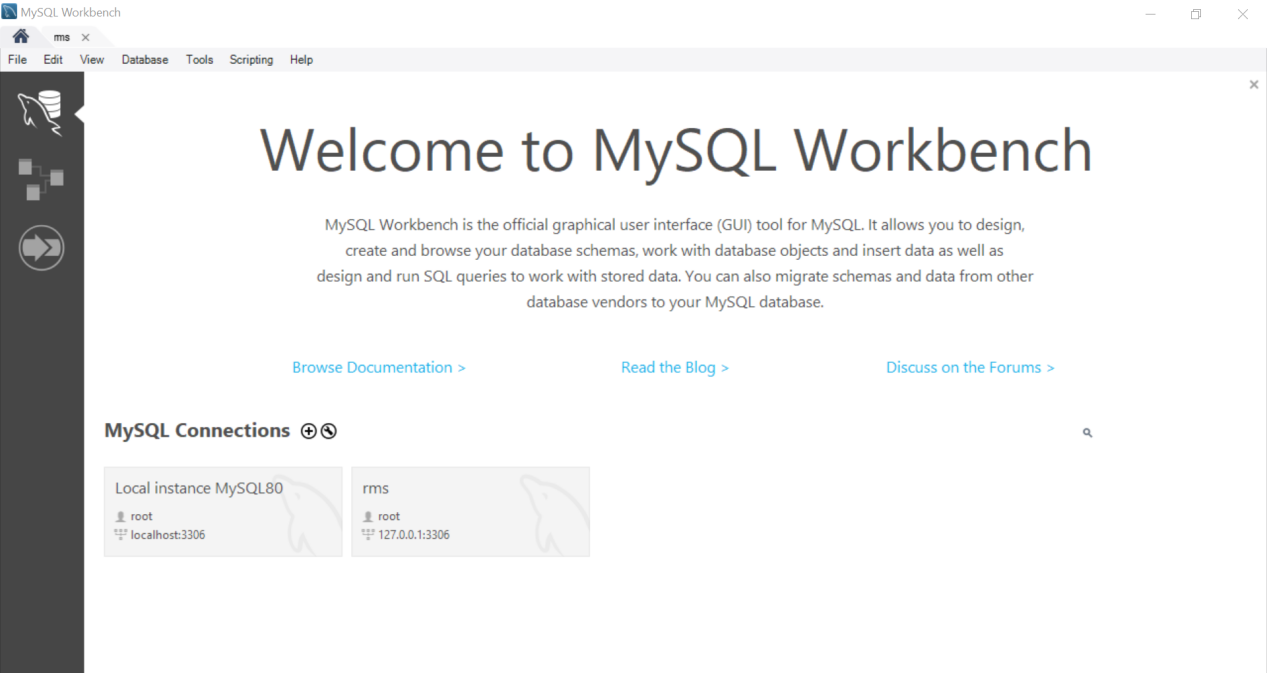
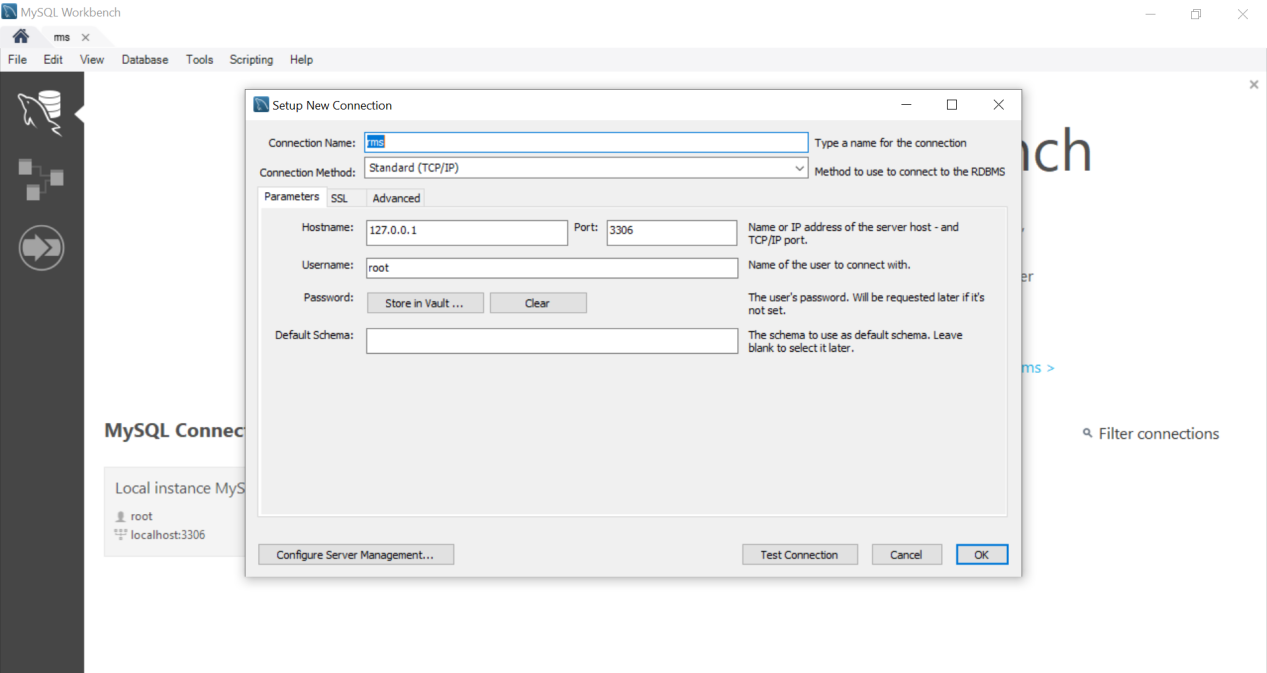


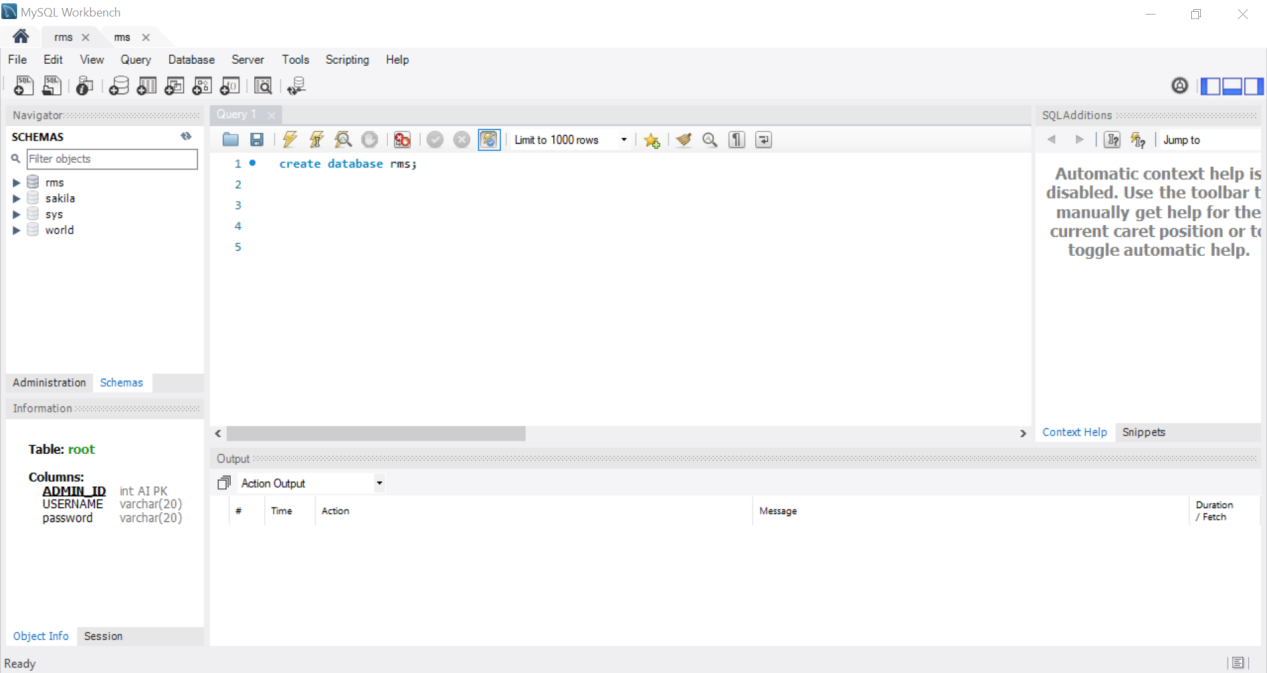
SETTING UP PROJECT

**Step 1: Download workbench**

**Step 2:Create a connection as we have created**

**Step 3: create database xyz(we have named it rms)**





**Step 4: In your Project download npm pkg mysql**

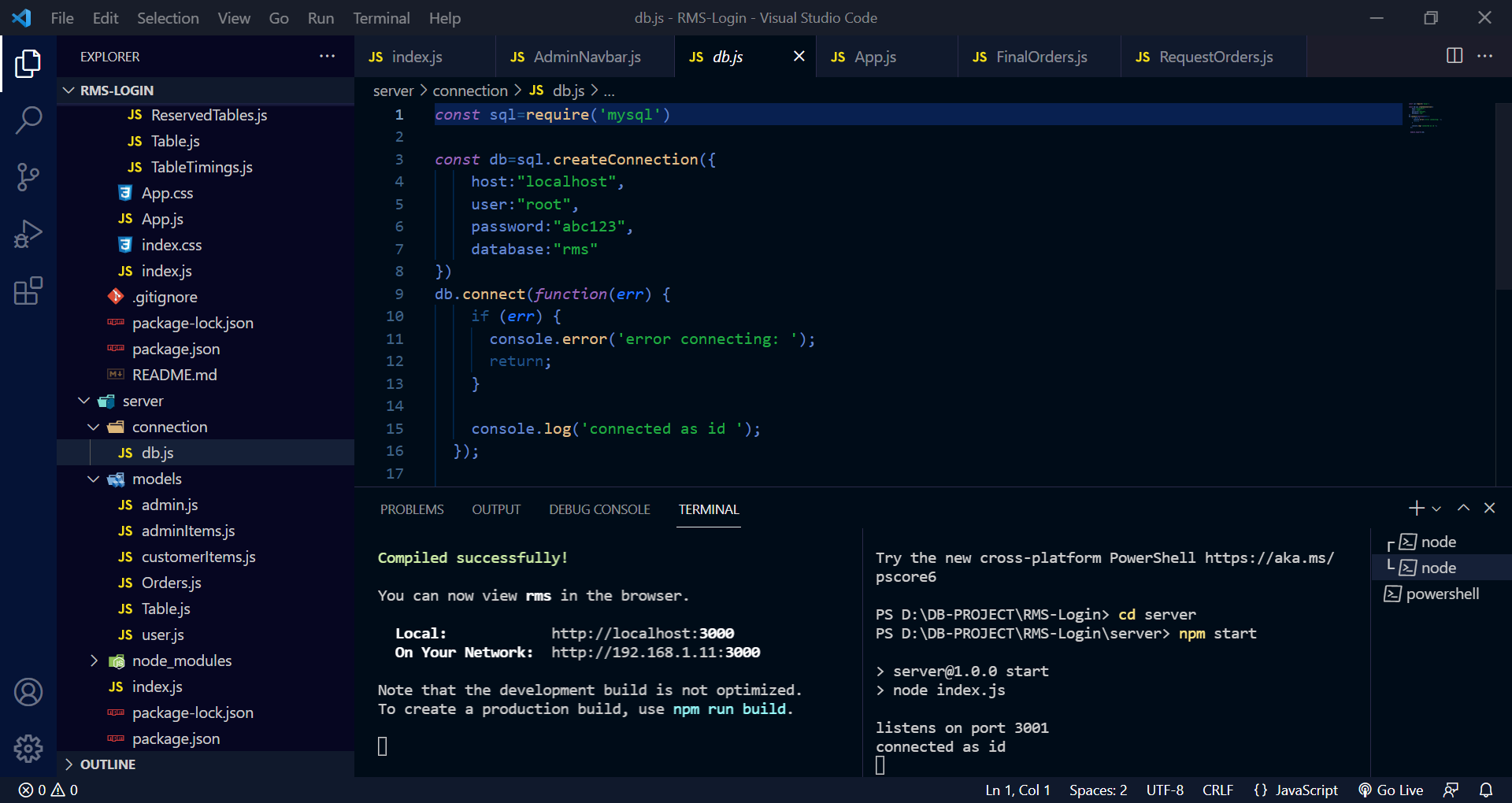
**Step 5: Create connection as we have created(username,password can vary )**

**Step 6: change directory to server**

**Step 7: run npm start**

**Step 8: change directory to rms/client**

**Step 9:run npm start**

**11**