# DB PROJECT REPORT

MEMBERS:

SARMAD JAMAL (K19-1116)

KHIZER JILANI (K19-1057)

MANSOOR BUTT(K19-1114)

**Problem Statement**

## Problem Definition

Following are commonly faced problems:

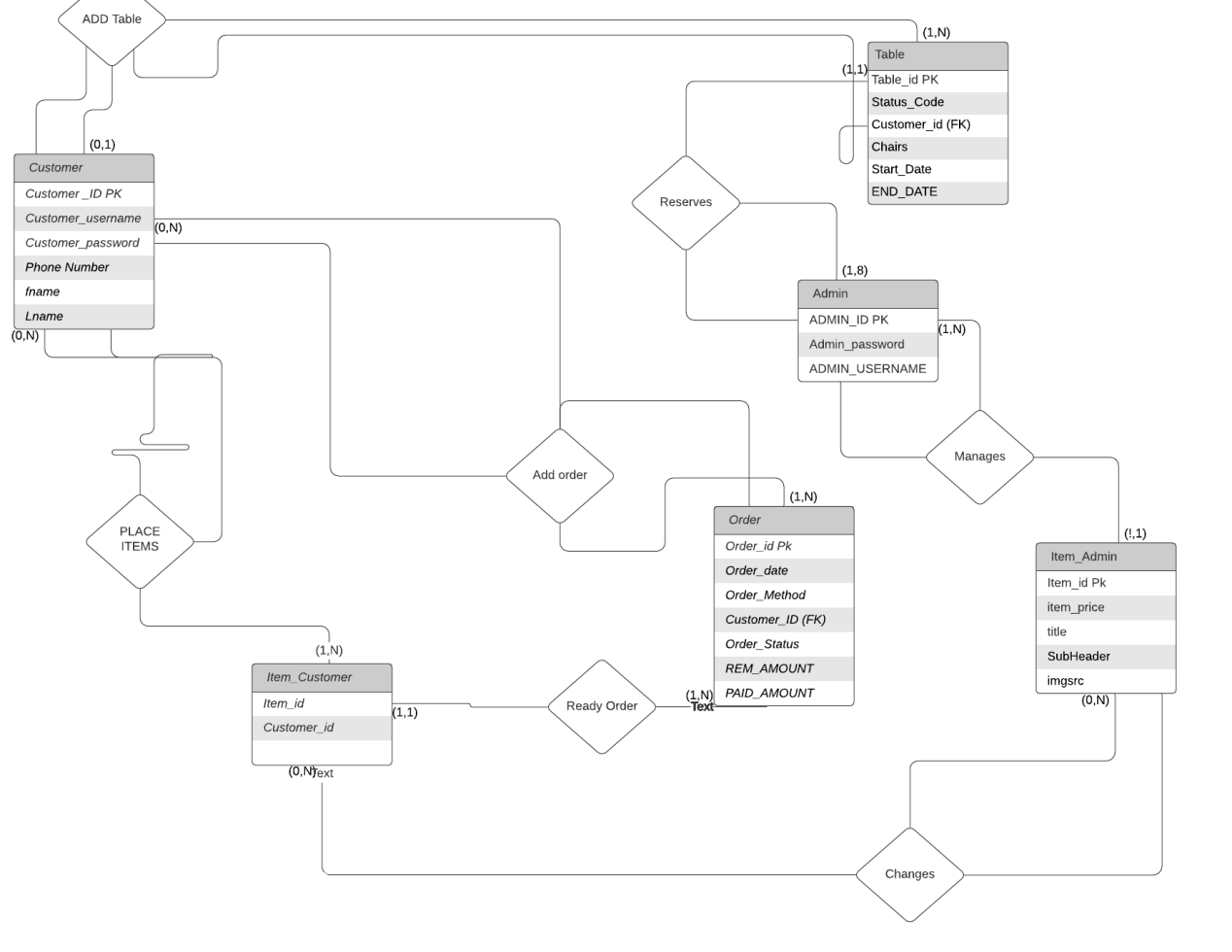
* Many times, we have to wait in long queues for the orders, which in return waste a lot of time and requires greater level of patience.
* Sometimes an order can be missed.
* Many human errors occur when we deal with information manually.
* There are many benefits for online restaurant management including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

## Objective

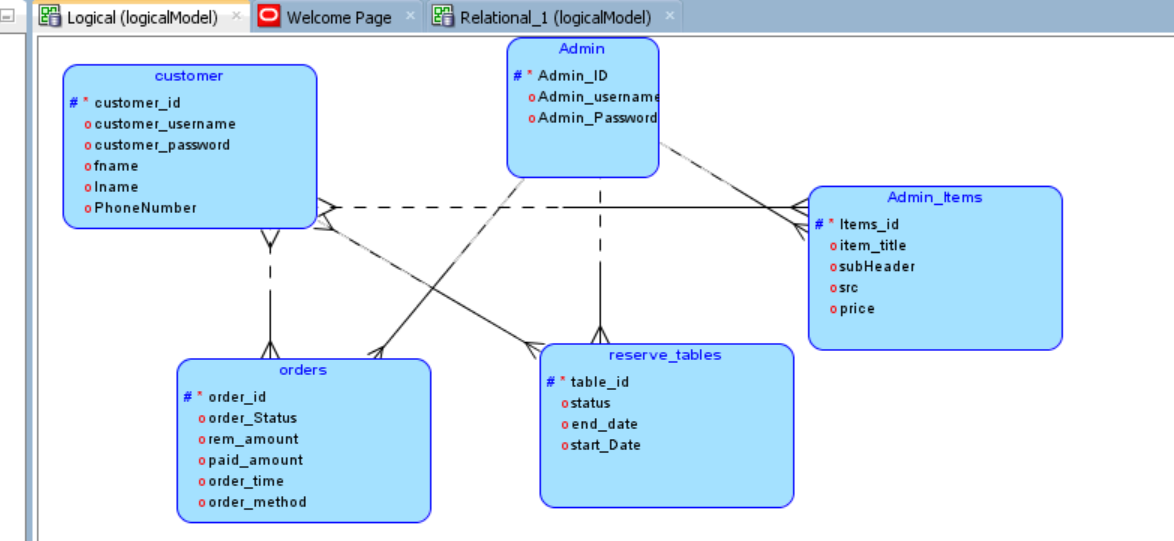
*The software we are proposing have many enhance features than other RMS(Restaurant Management Software), as we propose a system which would help every member of the restaurant to do his task with less effort and time,When talking about restaurants, there are many problems faced by people in restaurants but typical restaurant management website helps customers 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.*

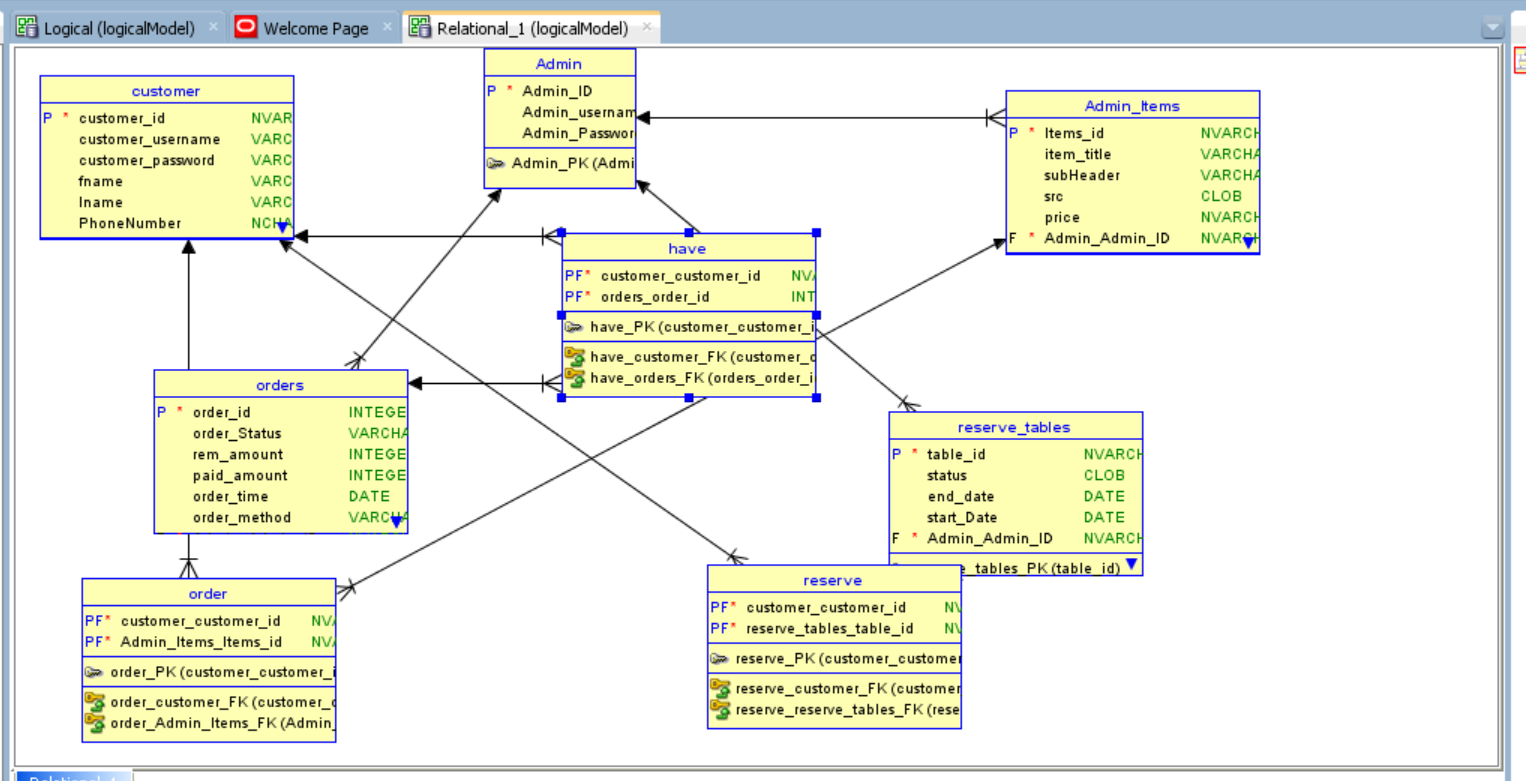
There is also a monitoring system for reserving tables and adding orders. Orders can be accepted, viewed and its status can be tracked. People had to wait in a long queue to order which was a big problem and effected customer satisfaction, through this system, orders can be taken online saving both customer and admin's efforts and also increase usability of the system.

ERD



RELATION AND LOGICAL MODEL USING DATA MODELER





GENERATED DDL

CREATE TABLE admin (

admin\_id NVARCHAR2(1) NOT NULL,

admin\_username VARCHAR2(10),

admin\_password VARCHAR2(10)

);

ALTER TABLE admin ADD CONSTRAINT admin\_pk PRIMARY KEY ( admin\_id );

CREATE TABLE admin\_items (

items\_id NVARCHAR2(1) NOT NULL,

item\_title VARCHAR2(10),

subheader VARCHAR2(10),

src CLOB,

price NVARCHAR2(1),

admin\_admin\_id NVARCHAR2(1) NOT NULL

);

ALTER TABLE admin\_items ADD CONSTRAINT admin\_items\_pk PRIMARY KEY ( items\_id );

CREATE TABLE customer (

customer\_id NVARCHAR2(1) NOT NULL,

customer\_username VARCHAR2(10),

customer\_password VARCHAR2(500),

fname VARCHAR2(10),

lname VARCHAR2(10),

phonenumber NCHAR(11)

);

ALTER TABLE customer ADD CONSTRAINT customer\_pk PRIMARY KEY ( customer\_id );

CREATE TABLE have (

customer\_customer\_id NVARCHAR2(1) NOT NULL,

orders\_order\_id INTEGER NOT NULL

);

ALTER TABLE have ADD CONSTRAINT have\_pk PRIMARY KEY ( customer\_customer\_id,

orders\_order\_id );

CREATE TABLE "order" (

customer\_customer\_id NVARCHAR2(1) NOT NULL,

admin\_items\_items\_id NVARCHAR2(1) NOT NULL

);

ALTER TABLE "order" ADD CONSTRAINT order\_pk PRIMARY KEY ( customer\_customer\_id,

admin\_items\_items\_id );

CREATE TABLE orders (

order\_id INTEGER NOT NULL,

order\_status VARCHAR2

,

rem\_amount INTEGER,

paid\_amount INTEGER,

order\_time DATE,

order\_method VARCHAR2(10),

admin\_admin\_id NVARCHAR2(1) NOT NULL

);

ALTER TABLE orders ADD CONSTRAINT orders\_pk PRIMARY KEY ( order\_id );

CREATE TABLE reserve (

customer\_customer\_id NVARCHAR2(1) NOT NULL,

reserve\_tables\_table\_id NVARCHAR2(1) NOT NULL

);

ALTER TABLE reserve ADD CONSTRAINT reserve\_pk PRIMARY KEY ( customer\_customer\_id,

reserve\_tables\_table\_id );

CREATE TABLE reserve\_tables (

table\_id NVARCHAR2(1) NOT NULL,

status CLOB,

end\_date DATE,

start\_date DATE,

admin\_admin\_id NVARCHAR2(1) NOT NULL

);

ALTER TABLE reserve\_tables ADD CONSTRAINT reserve\_tables\_pk PRIMARY KEY ( table\_id );

ALTER TABLE admin\_items

ADD CONSTRAINT admin\_items\_admin\_fk FOREIGN KEY ( admin\_admin\_id )

REFERENCES admin ( admin\_id );

ALTER TABLE have

ADD CONSTRAINT have\_customer\_fk FOREIGN KEY ( customer\_customer\_id )

REFERENCES customer ( customer\_id );

ALTER TABLE have

ADD CONSTRAINT have\_orders\_fk FOREIGN KEY ( orders\_order\_id )

REFERENCES orders ( order\_id );

ALTER TABLE "order"

ADD CONSTRAINT order\_admin\_items\_fk FOREIGN KEY ( admin\_items\_items\_id )

REFERENCES admin\_items ( items\_id );

ALTER TABLE "order"

ADD CONSTRAINT order\_customer\_fk FOREIGN KEY ( customer\_customer\_id )

REFERENCES customer ( customer\_id );

ALTER TABLE orders

ADD CONSTRAINT orders\_admin\_fk FOREIGN KEY ( admin\_admin\_id )

REFERENCES admin ( admin\_id );

ALTER TABLE reserve

ADD CONSTRAINT reserve\_customer\_fk FOREIGN KEY ( customer\_customer\_id )

REFERENCES customer ( customer\_id );

ALTER TABLE reserve

ADD CONSTRAINT reserve\_reserve\_tables\_fk FOREIGN KEY ( reserve\_tables\_table\_id )

REFERENCES reserve\_tables ( table\_id );

ALTER TABLE reserve\_tables

ADD CONSTRAINT reserve\_tables\_admin\_fk FOREIGN KEY ( admin\_admin\_id )

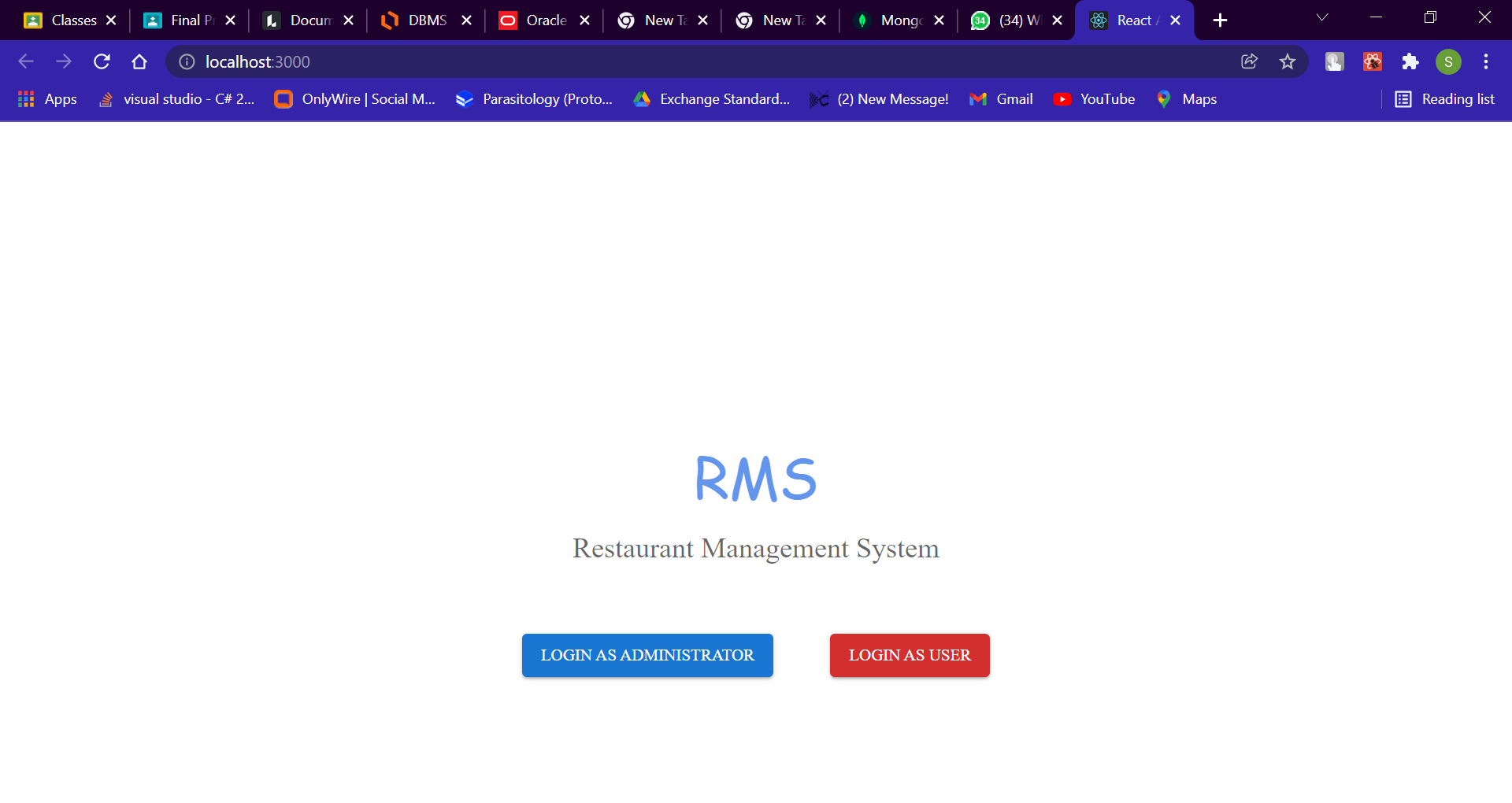
REFERENCES admin ( admin\_id );

NORMALIZED TABLES UPTO 3NF

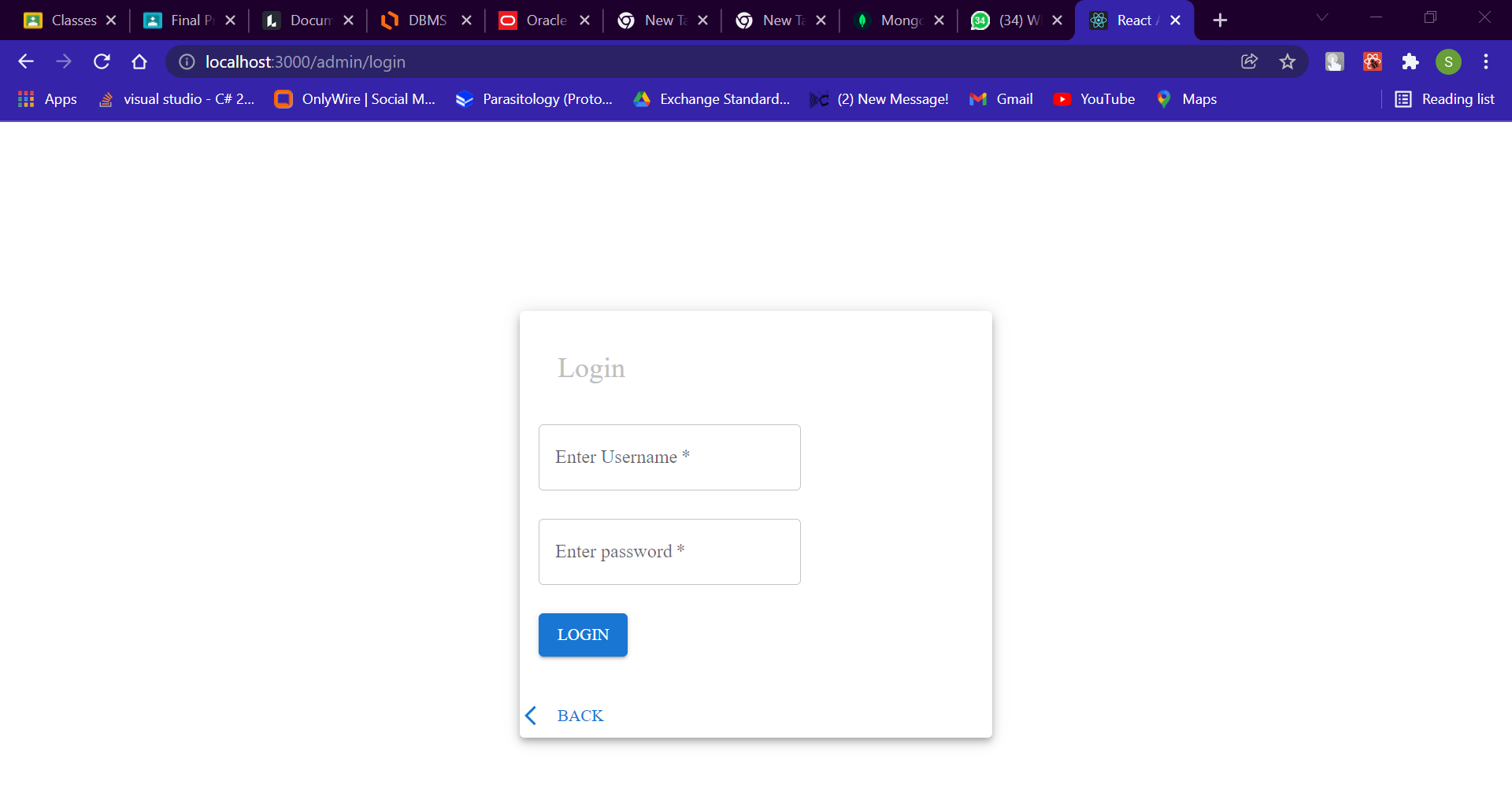
* admin (admin\_id ,admin\_username ,admin\_password)
* Customer(*customer\_id* ,customer\_username ,customer\_password ,fname ,lname ,phonenumber)
* reserve\_tables (table\_id ,status ,end\_date ,start\_date,customer\_id)
* item\_admin (*item\_id* ,item\_price , item\_title ,item\_subHeader,item \_src);
* item\_customer (*item\_id* ,customer\_id );
* Orders (order\_id, order\_status, rem\_amount,paid\_amount, order\_time, order\_method );

# Snap shots

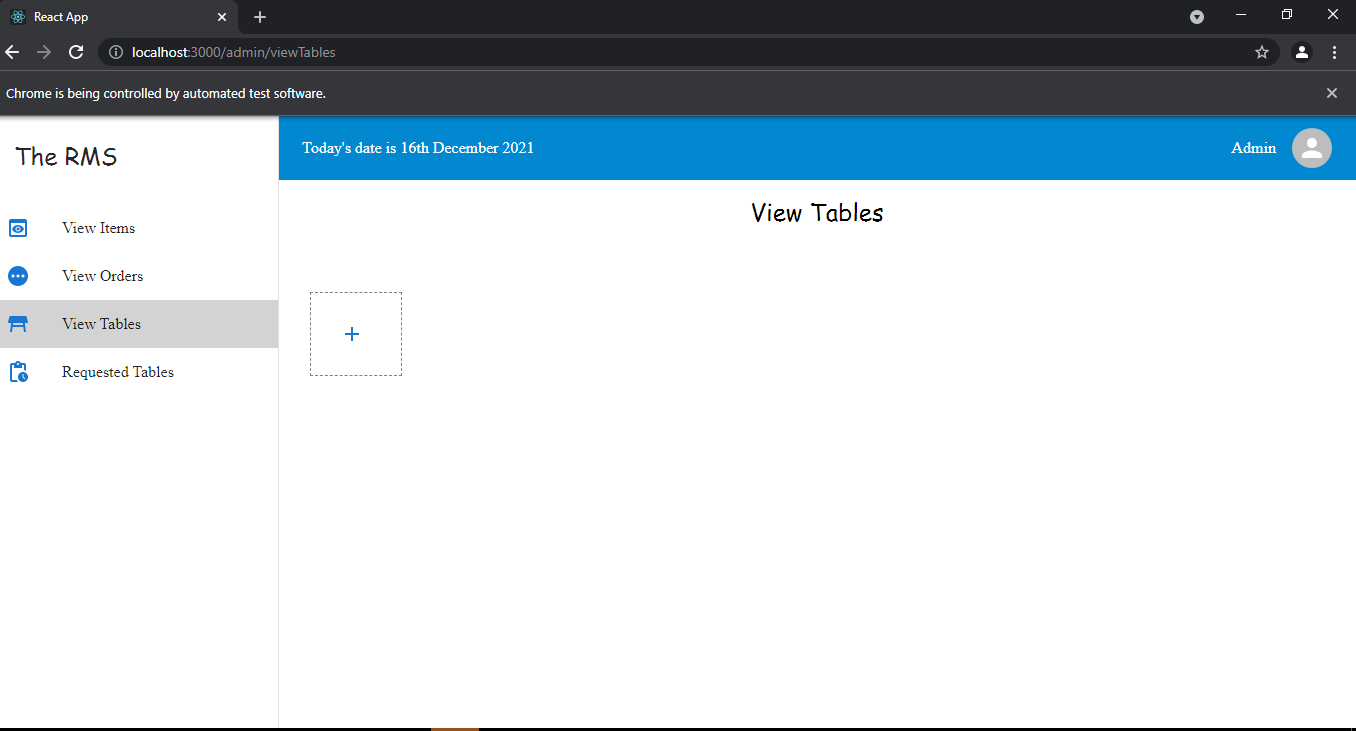
->HOMEPAGE

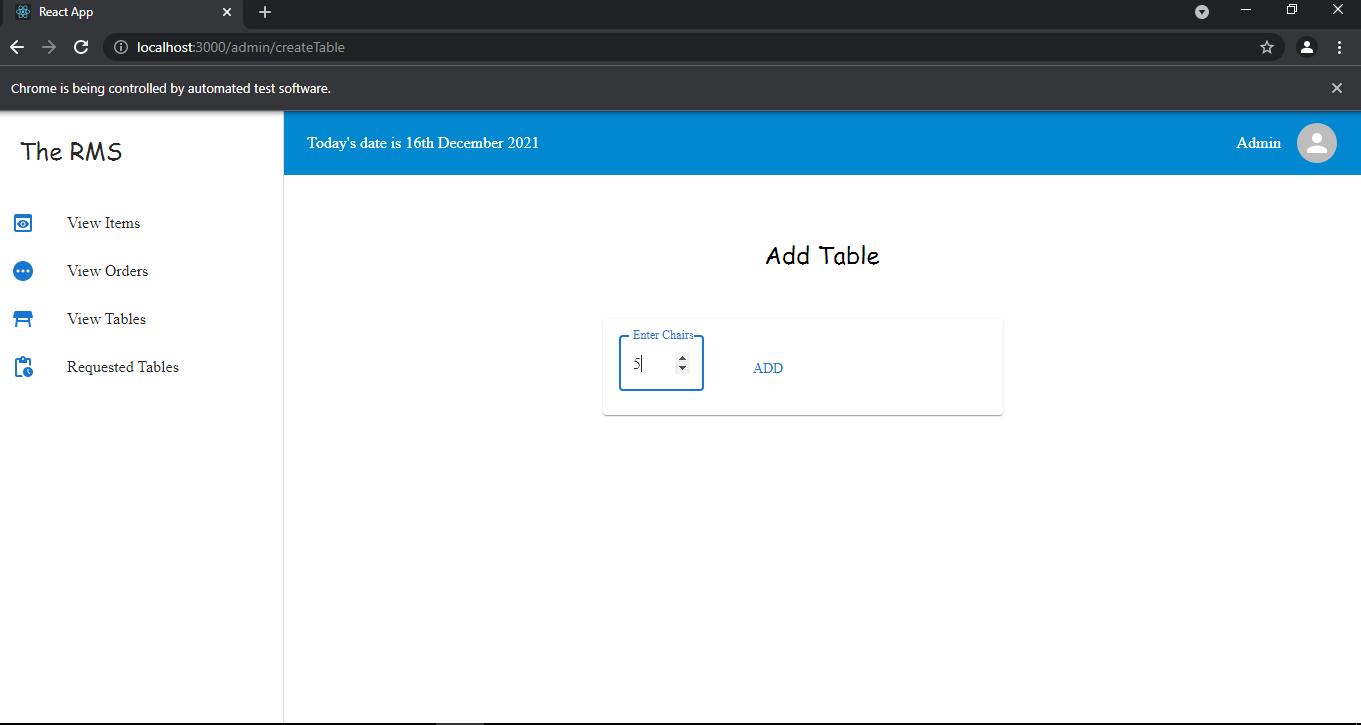


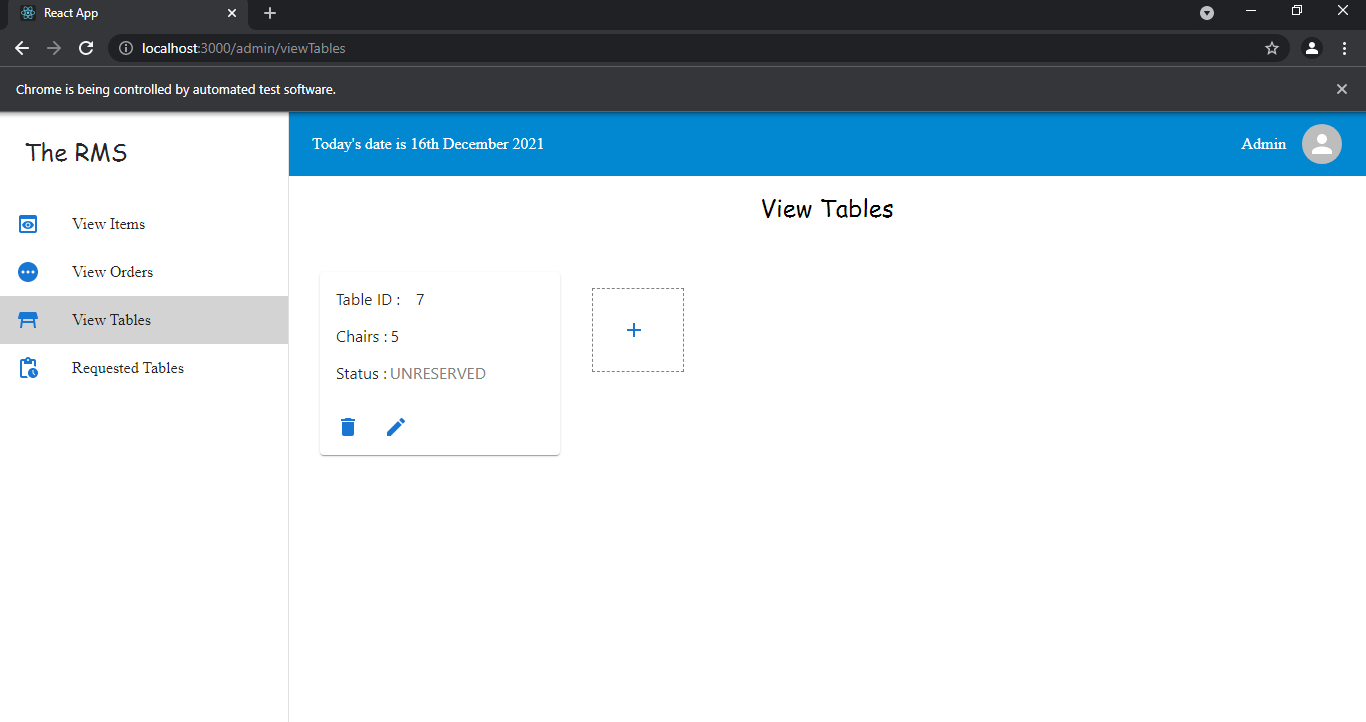
->Admin login Page



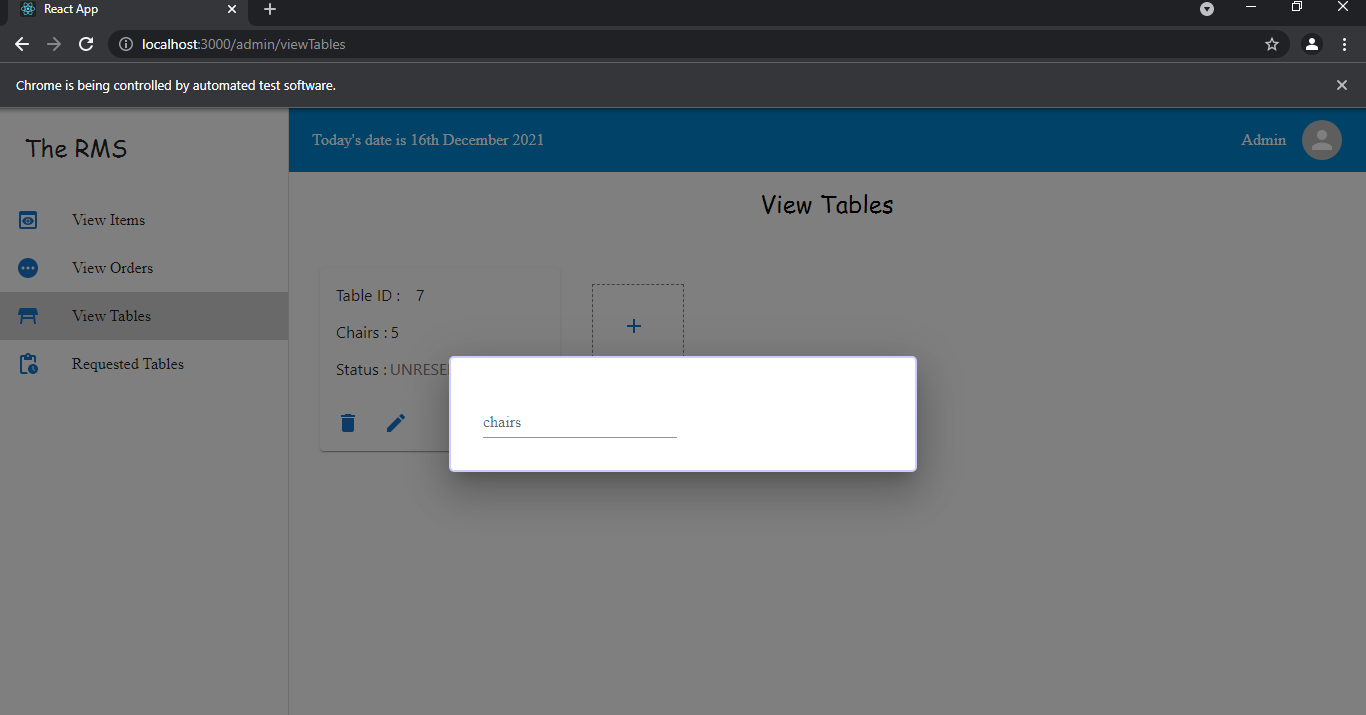
->Admin Adding tables

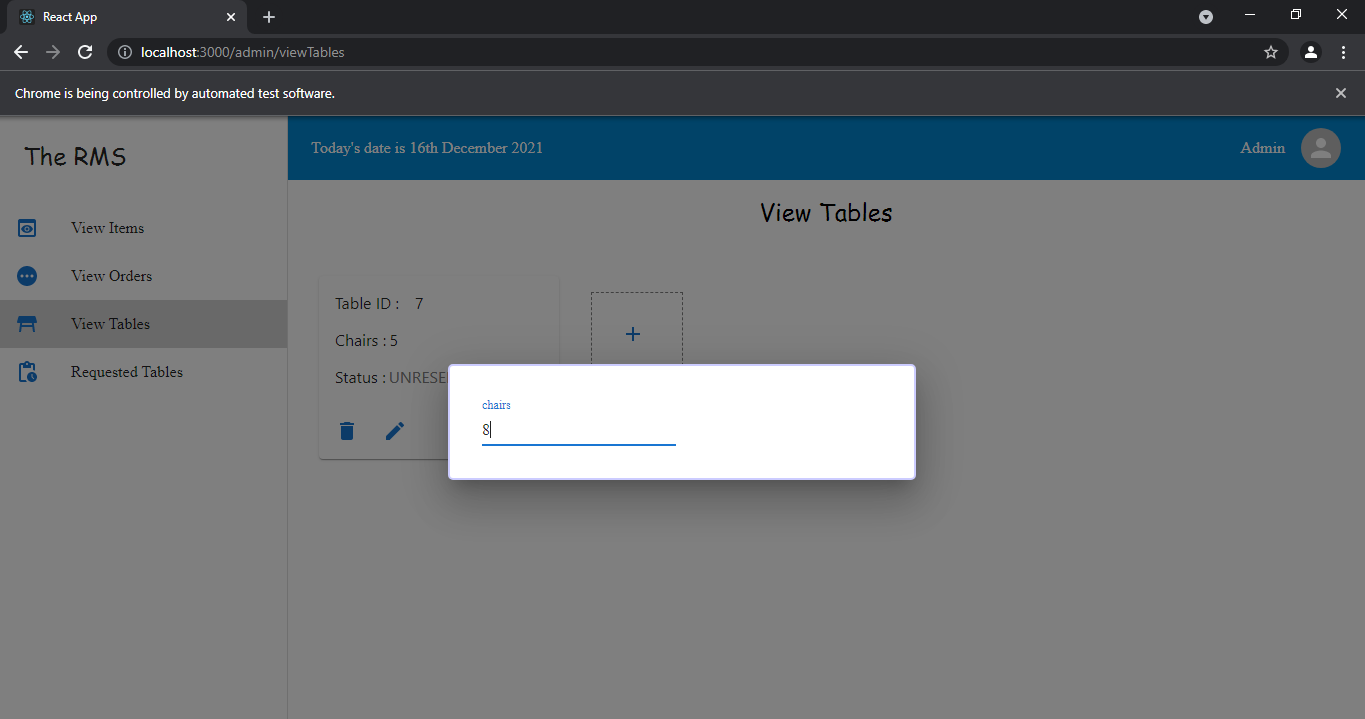


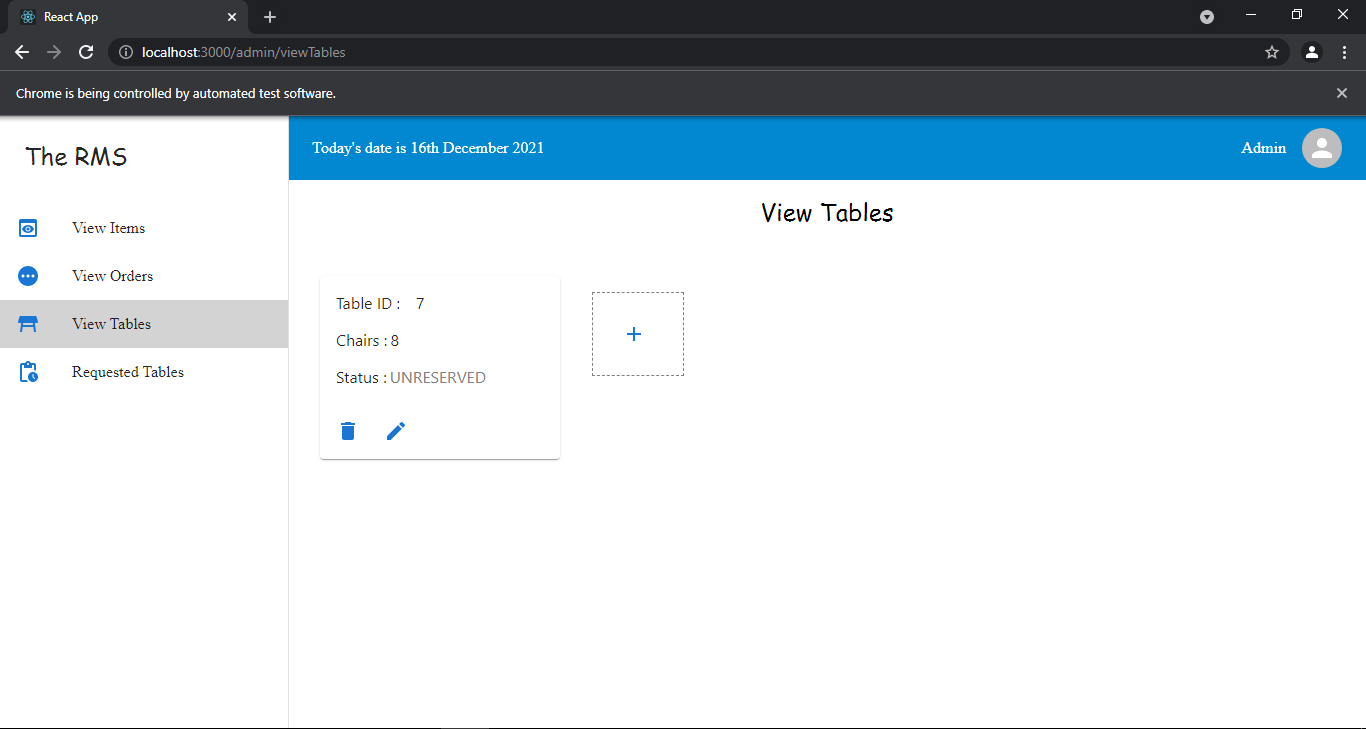




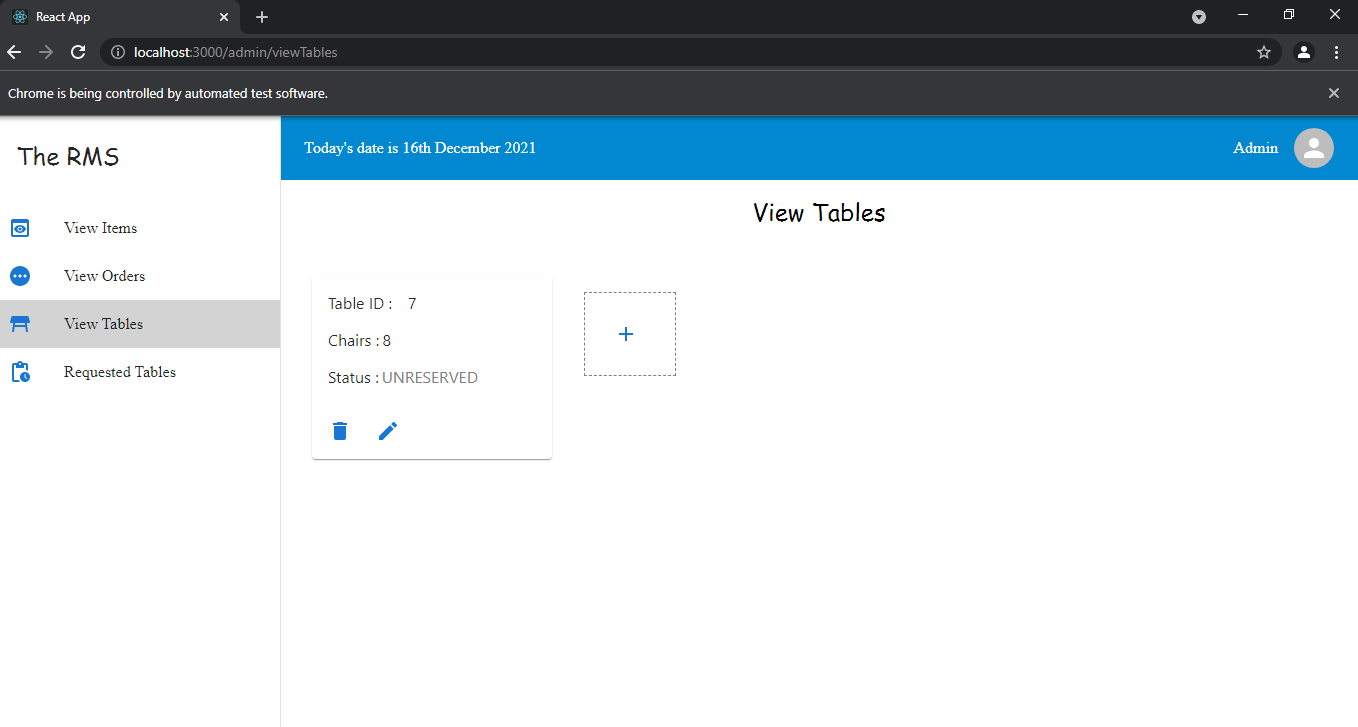
->Admin Updating Chairs

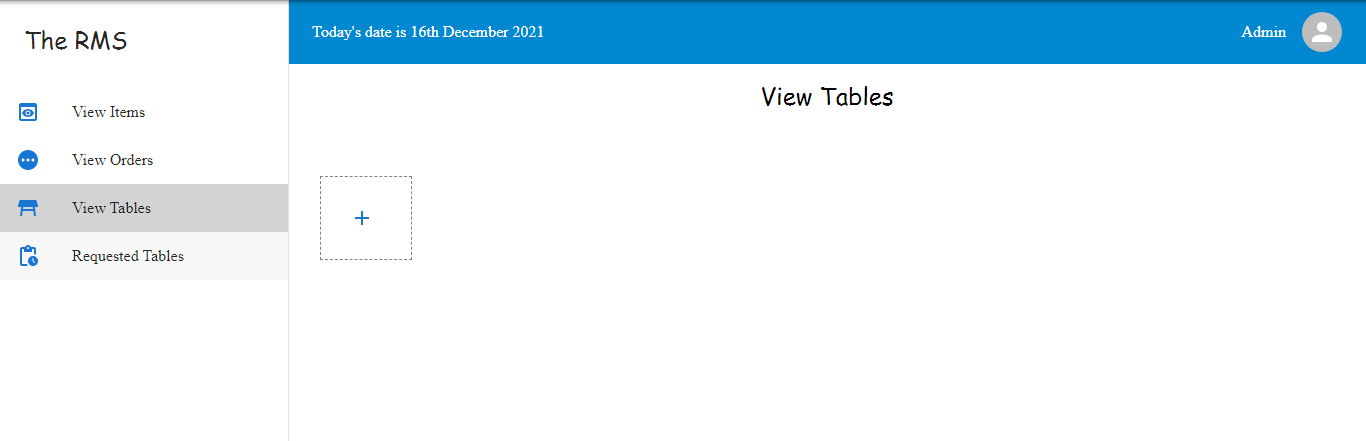




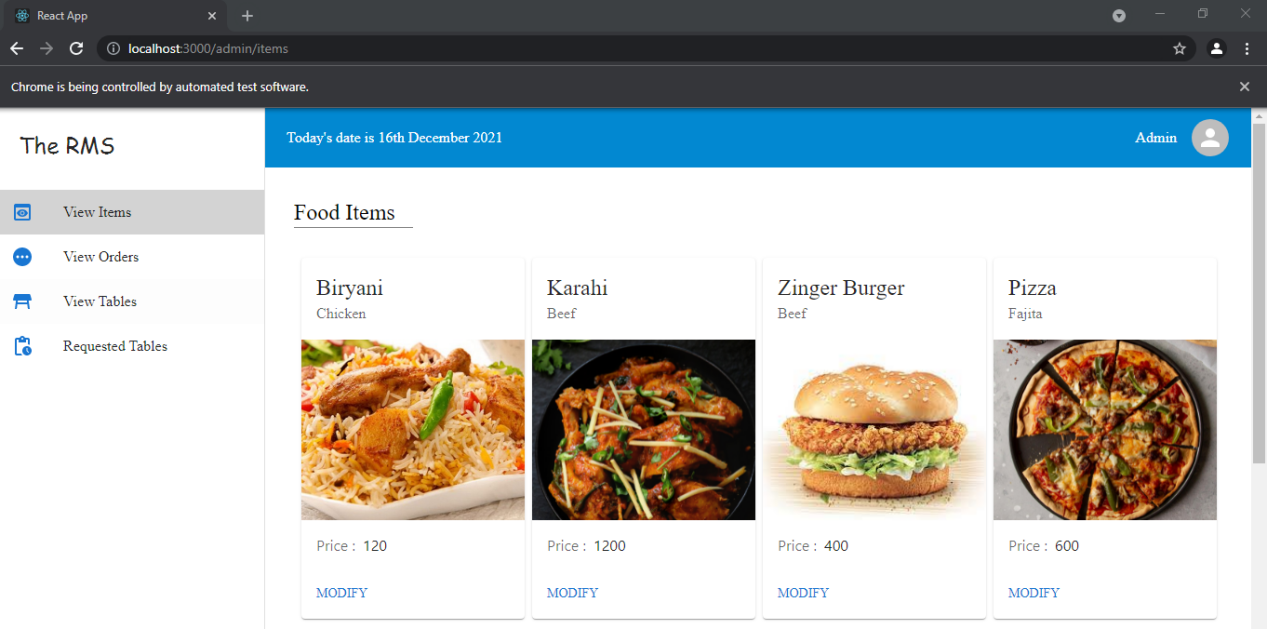


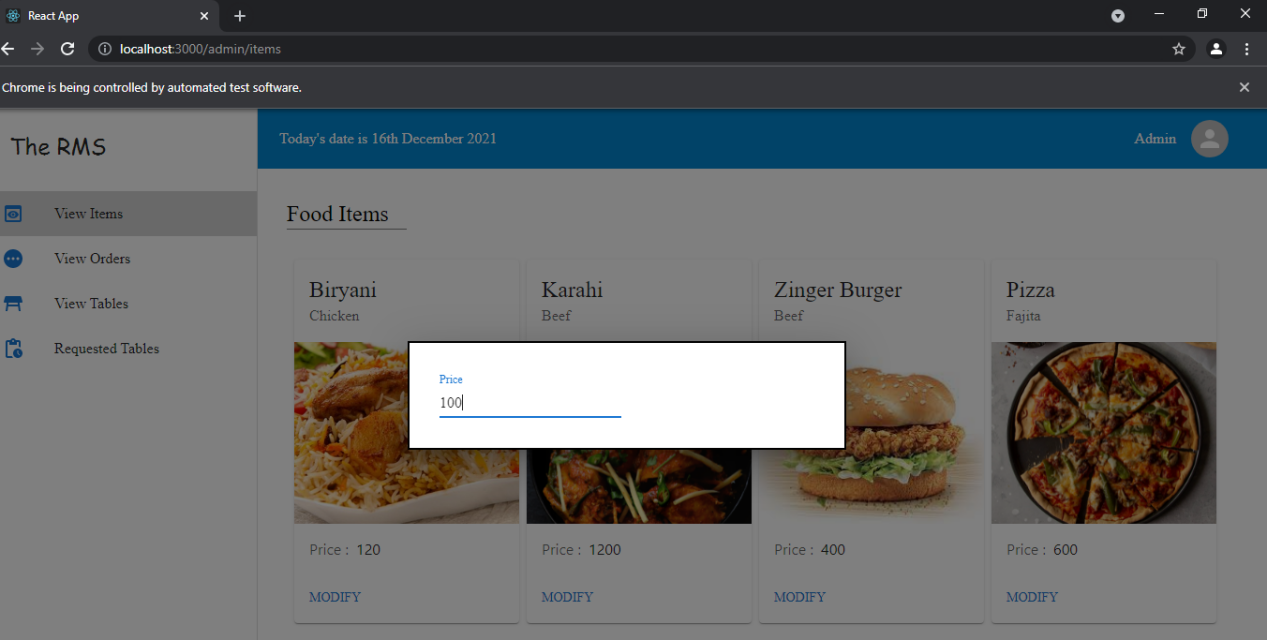
->Admin Deleting tables

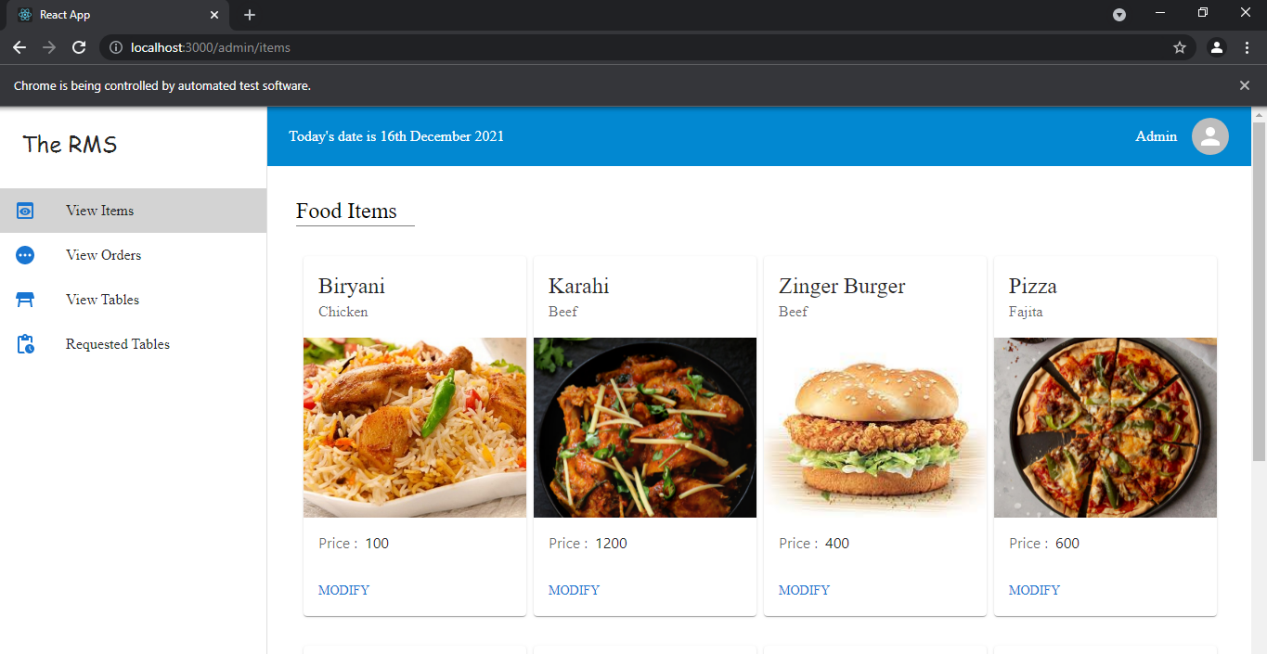




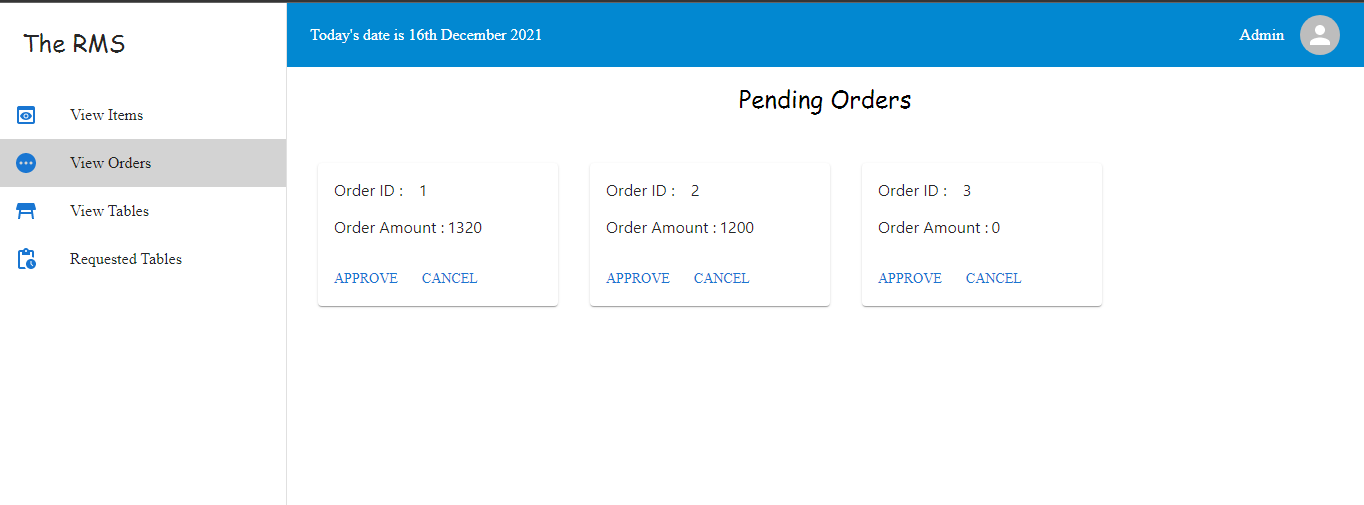
->Admin modifying Items price

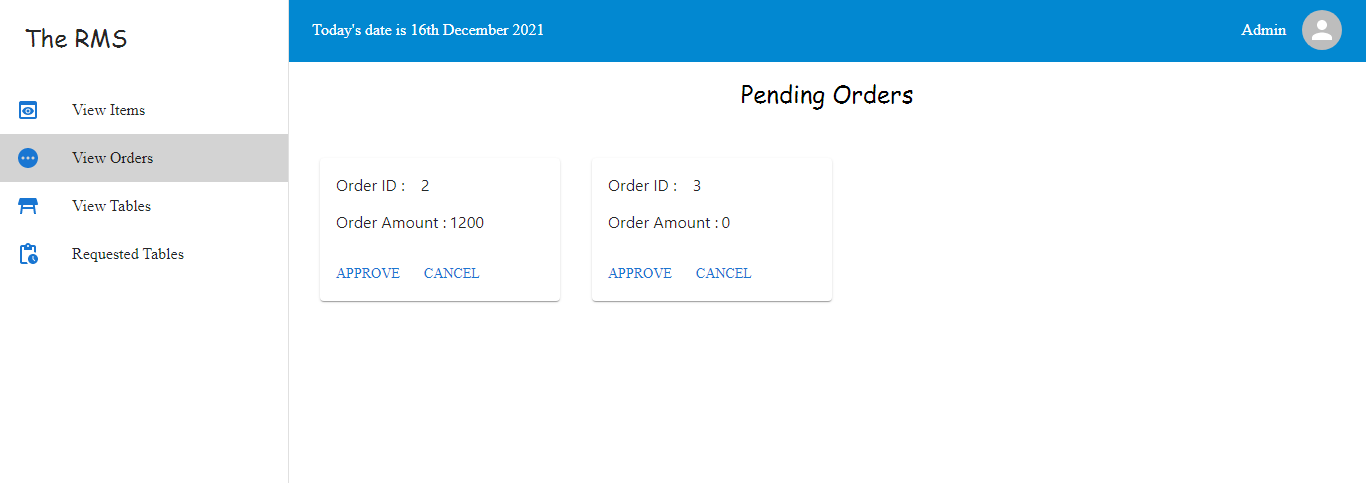




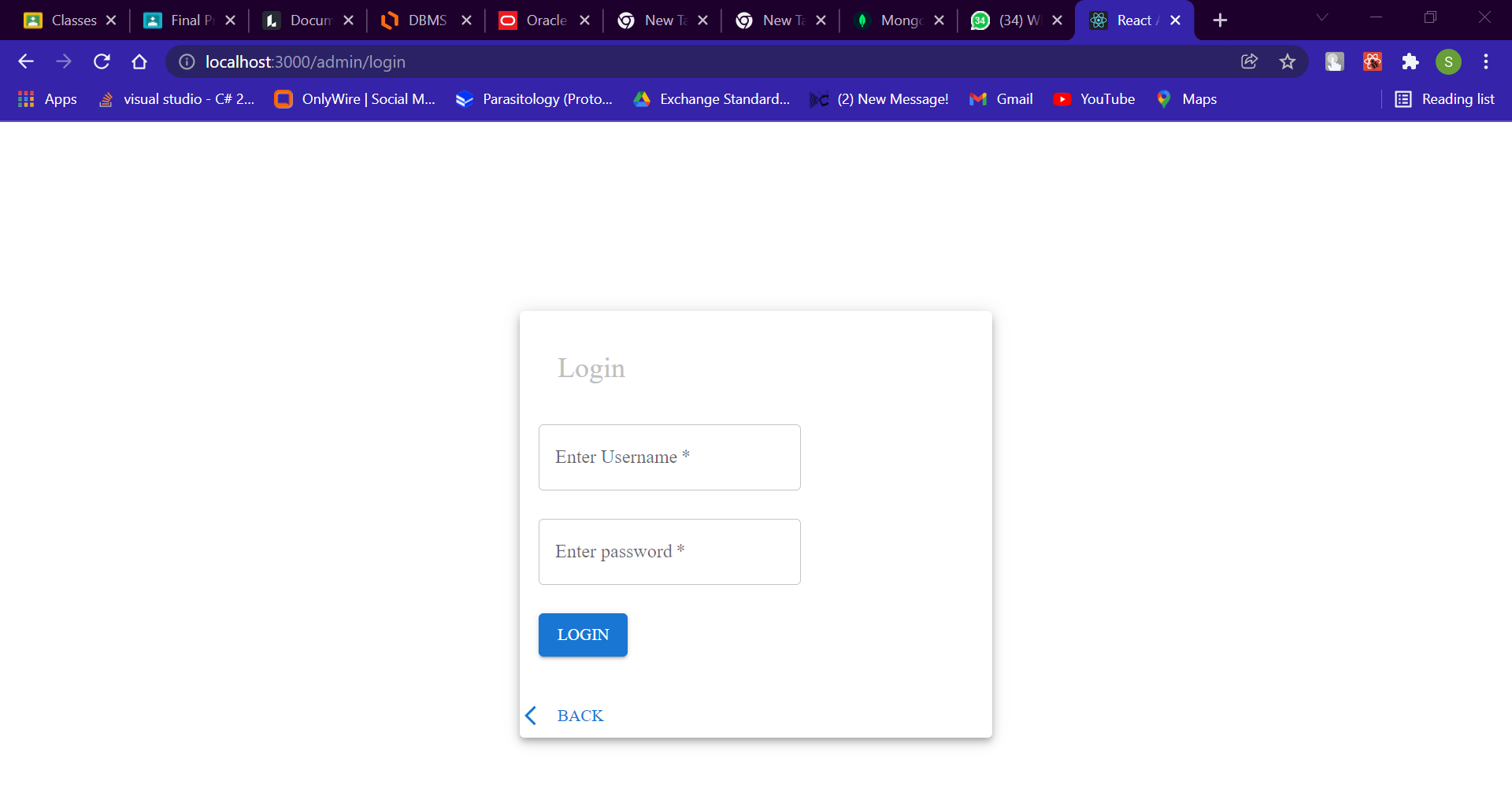


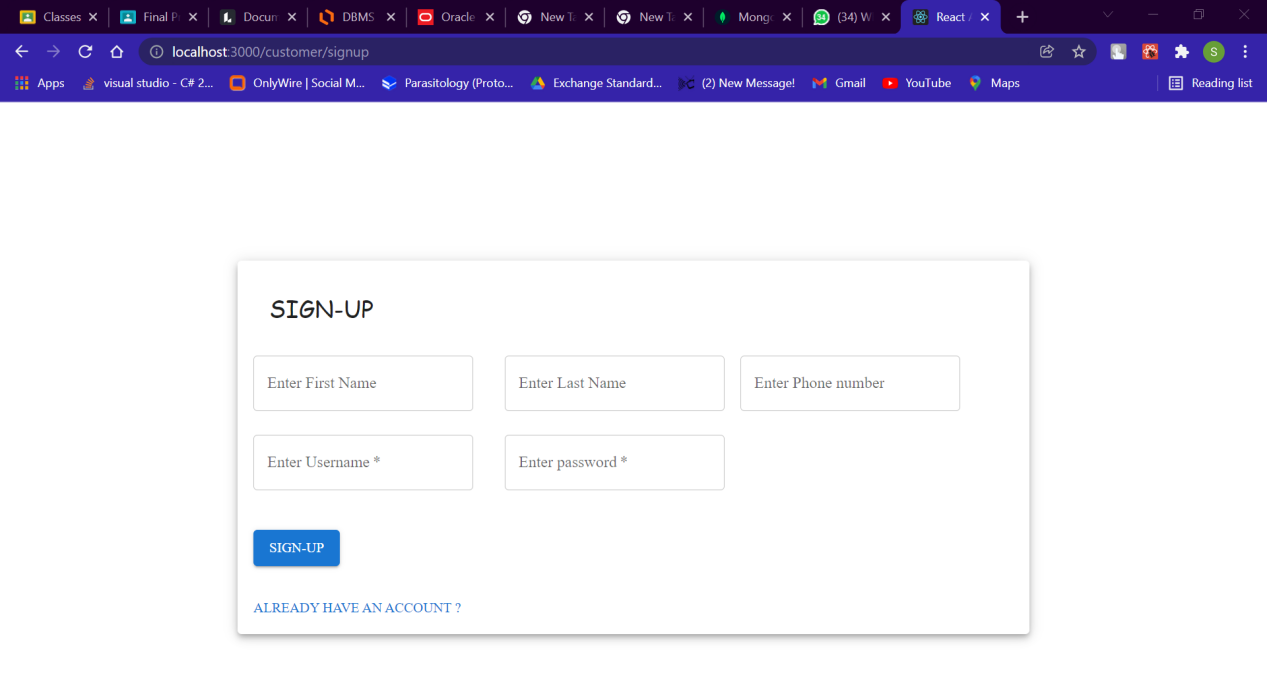
->admin approving Orders

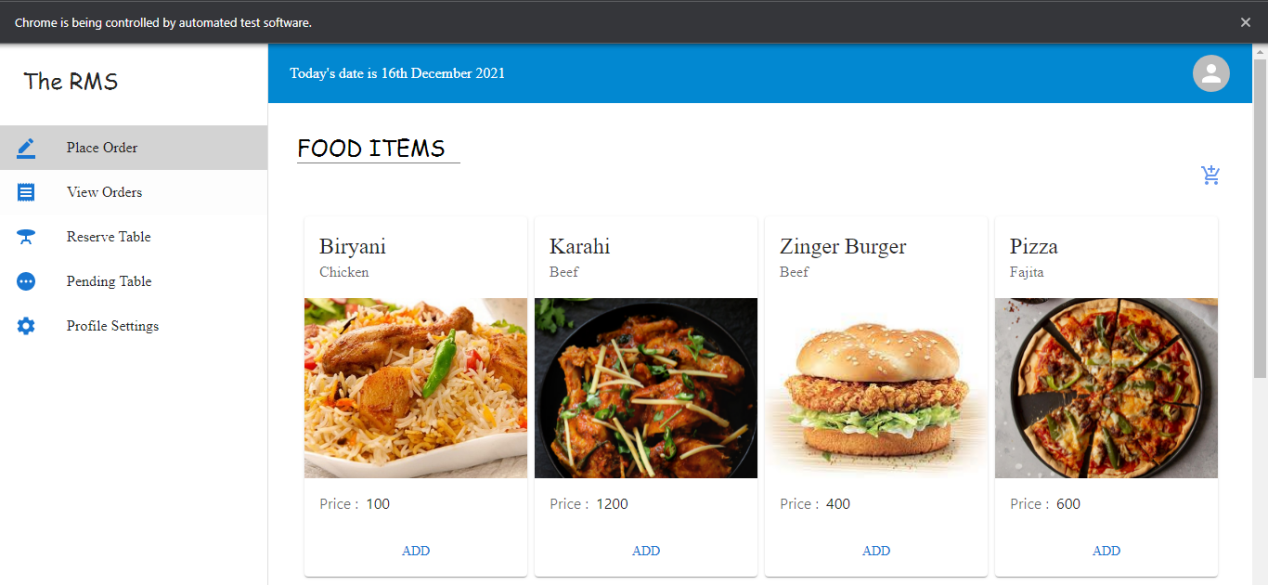


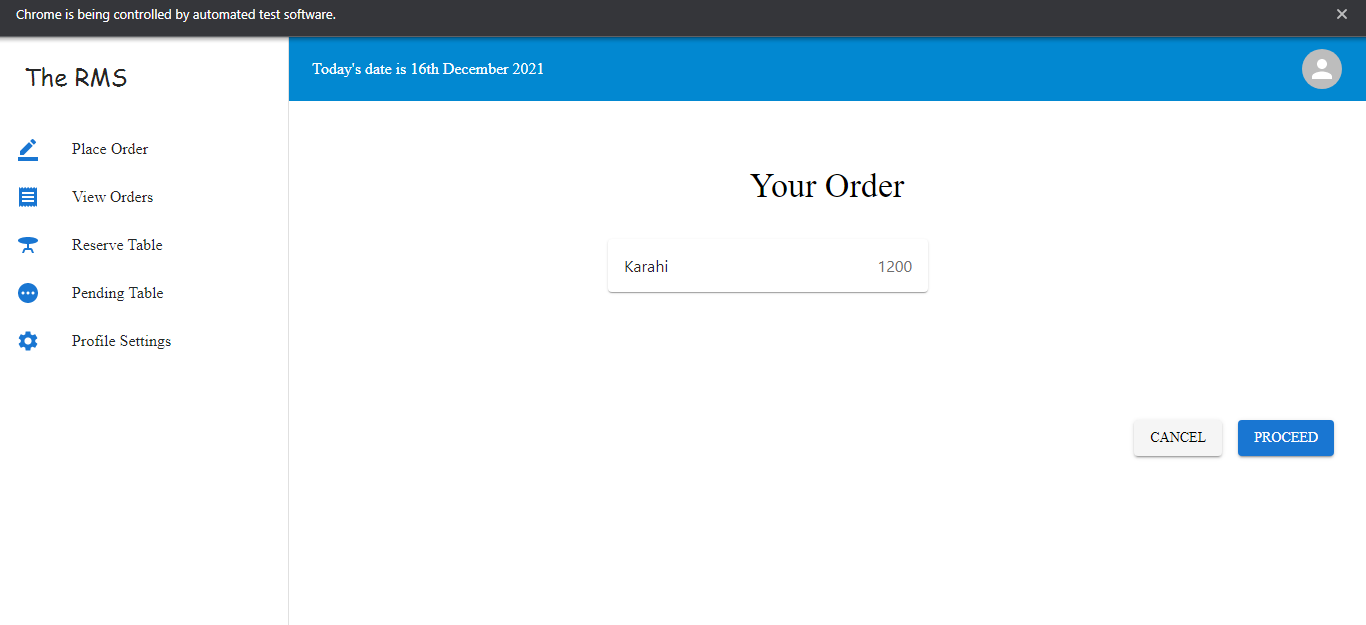


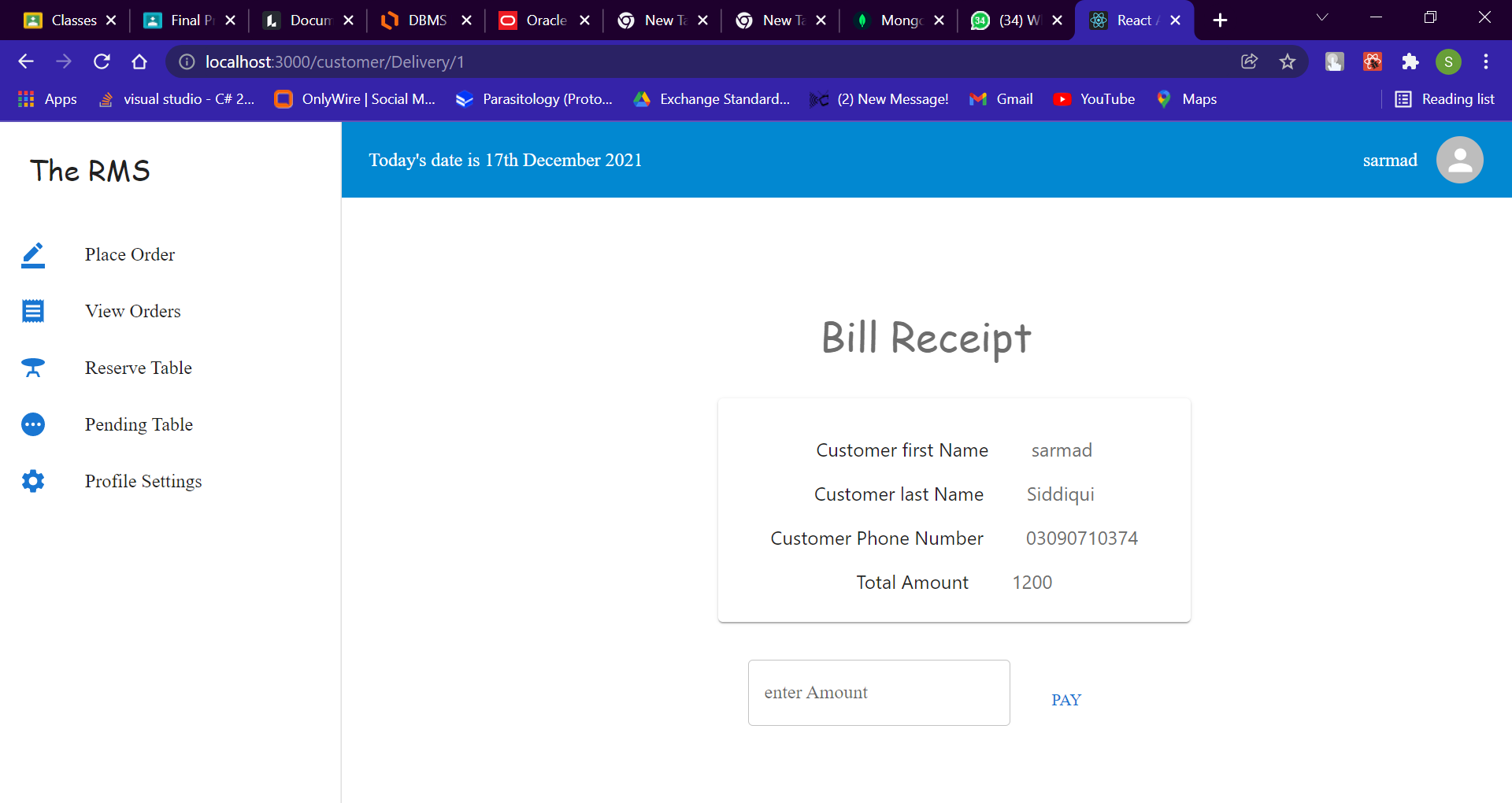
->Customer Flow of adding order

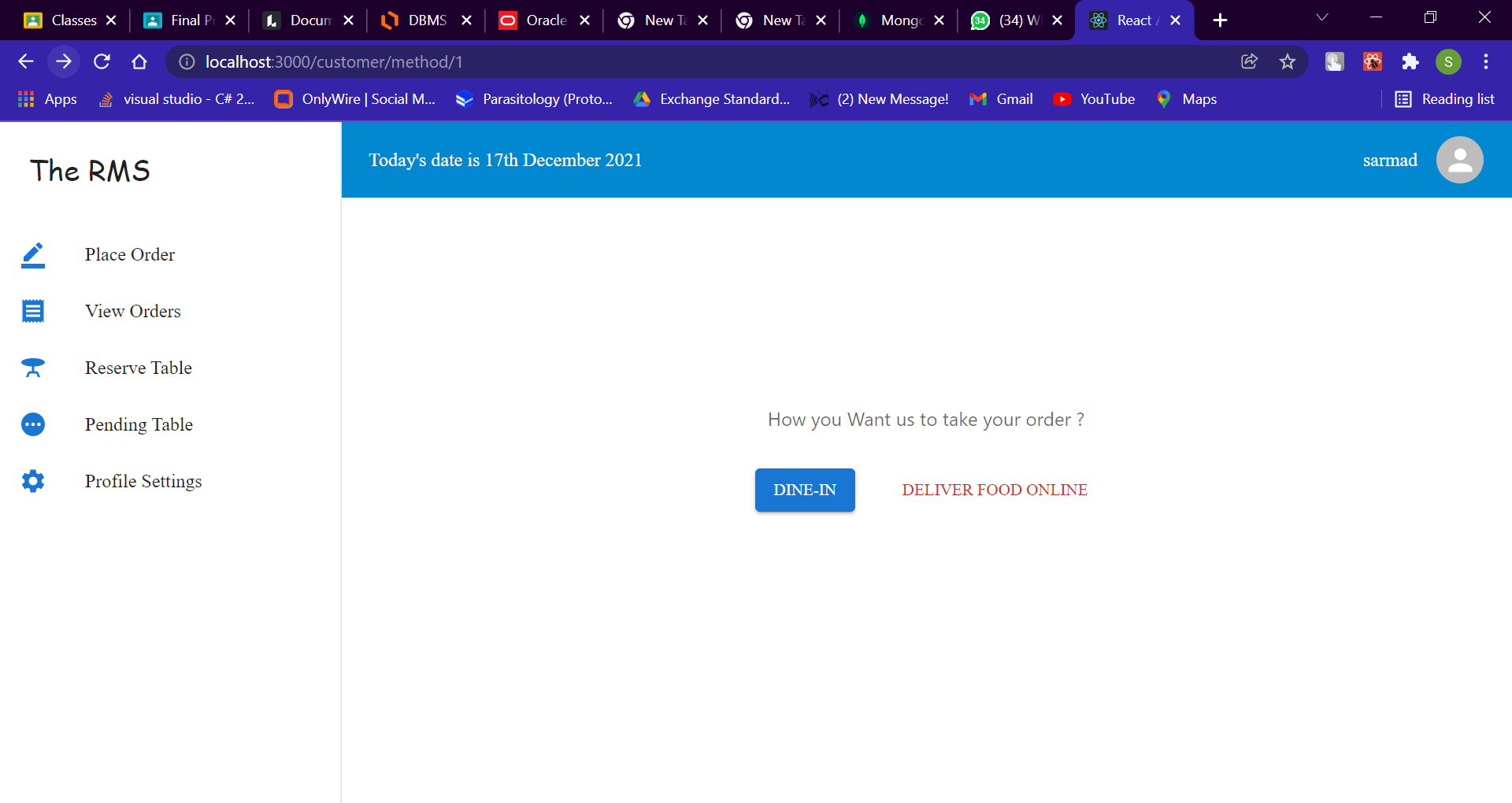




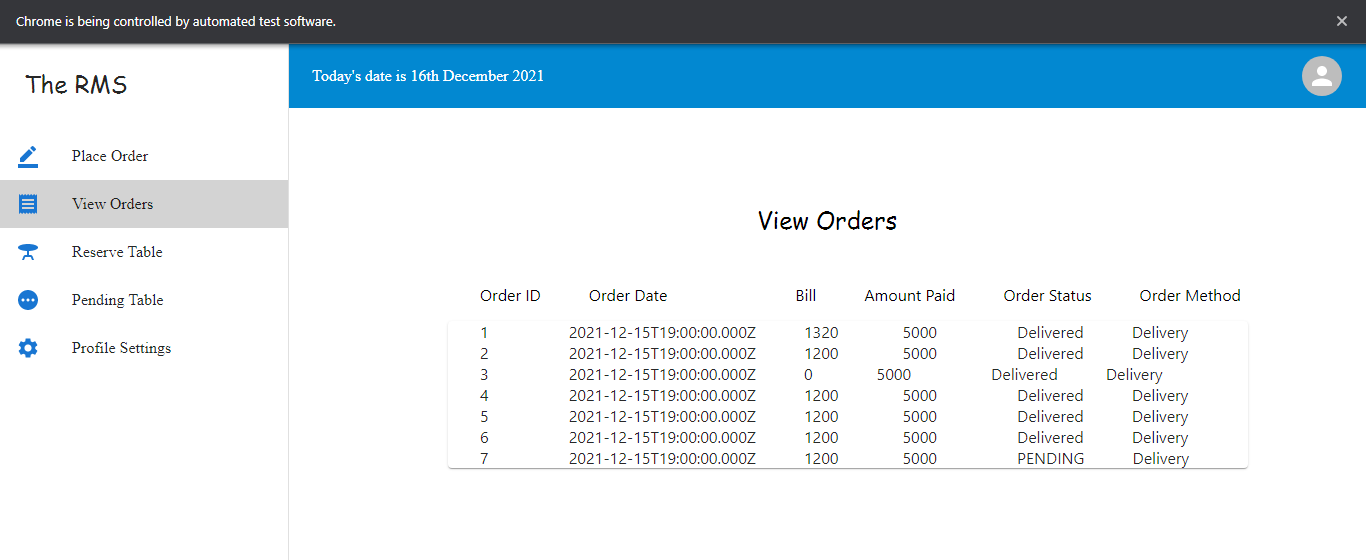


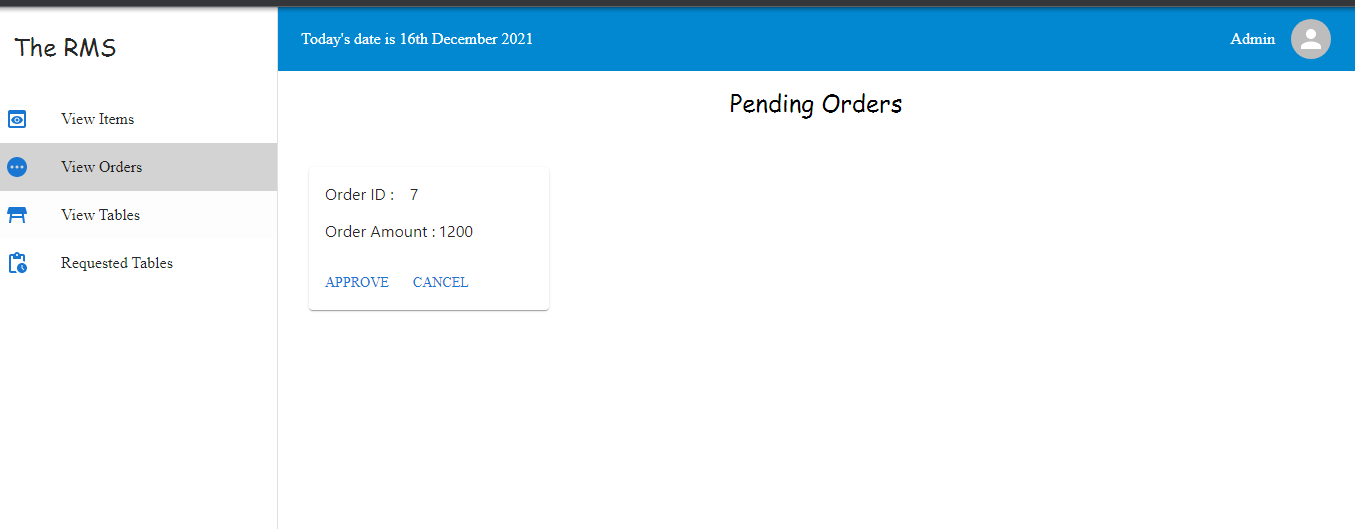




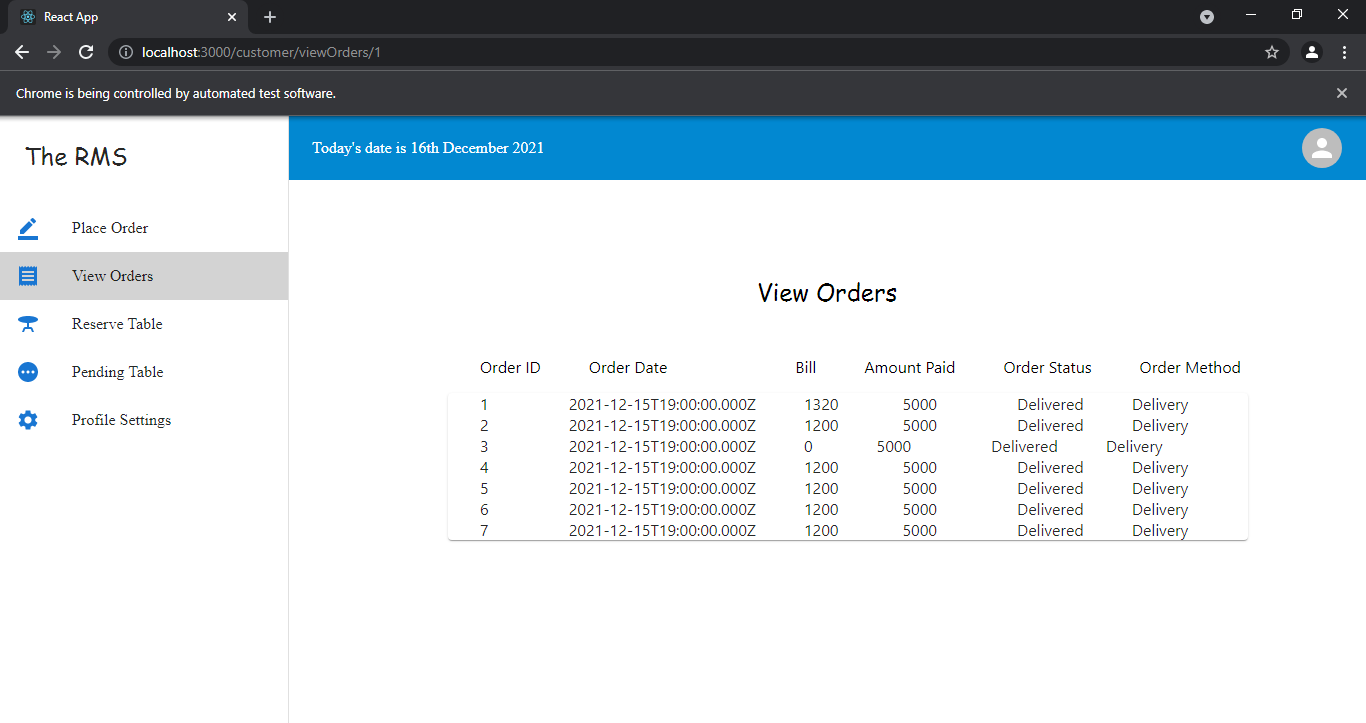


->Order is pending since admin has not approved it

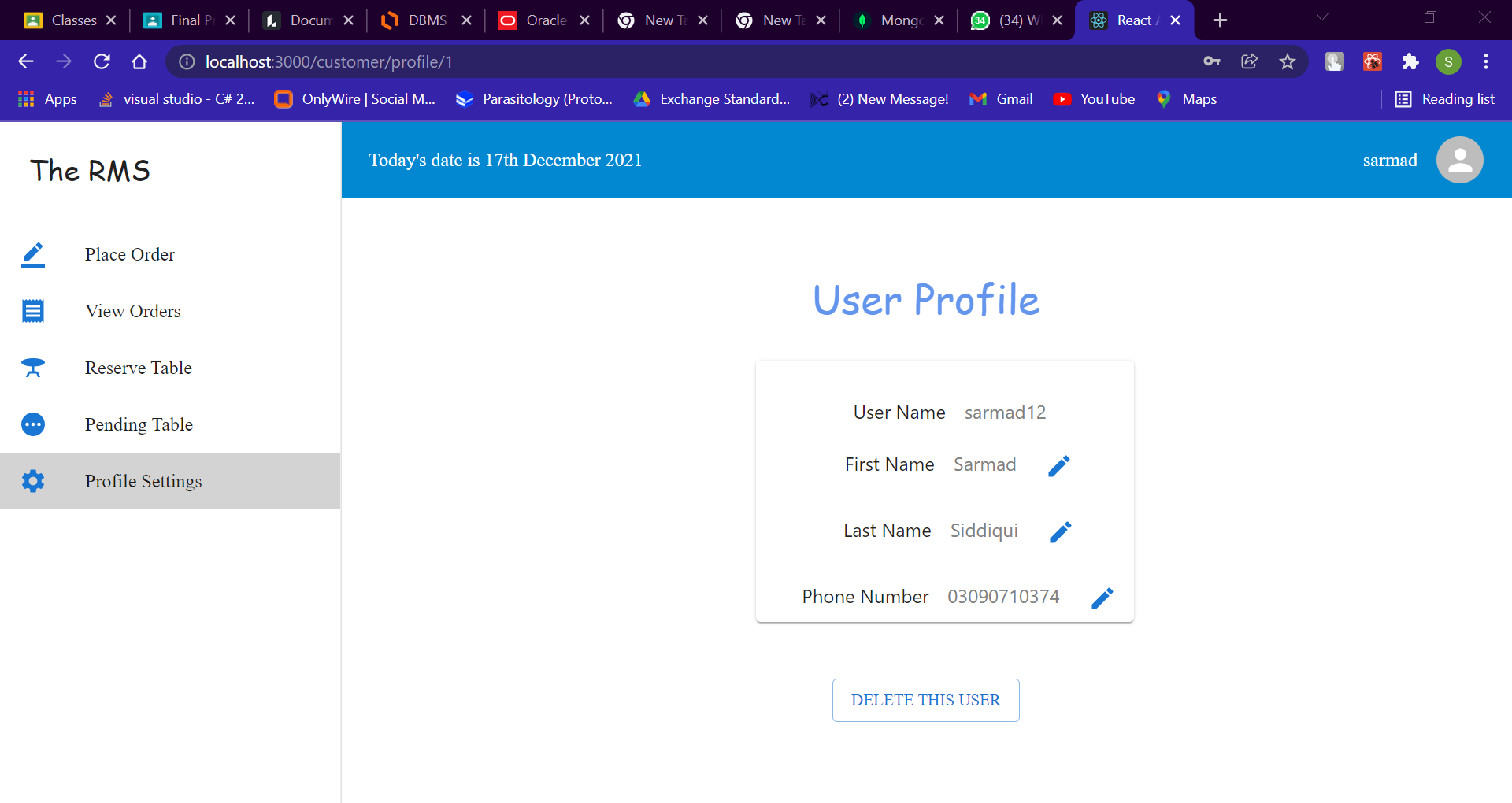




->Order is delivered since admin has approved it



-> Customer Profile



**RMS Core Features:**

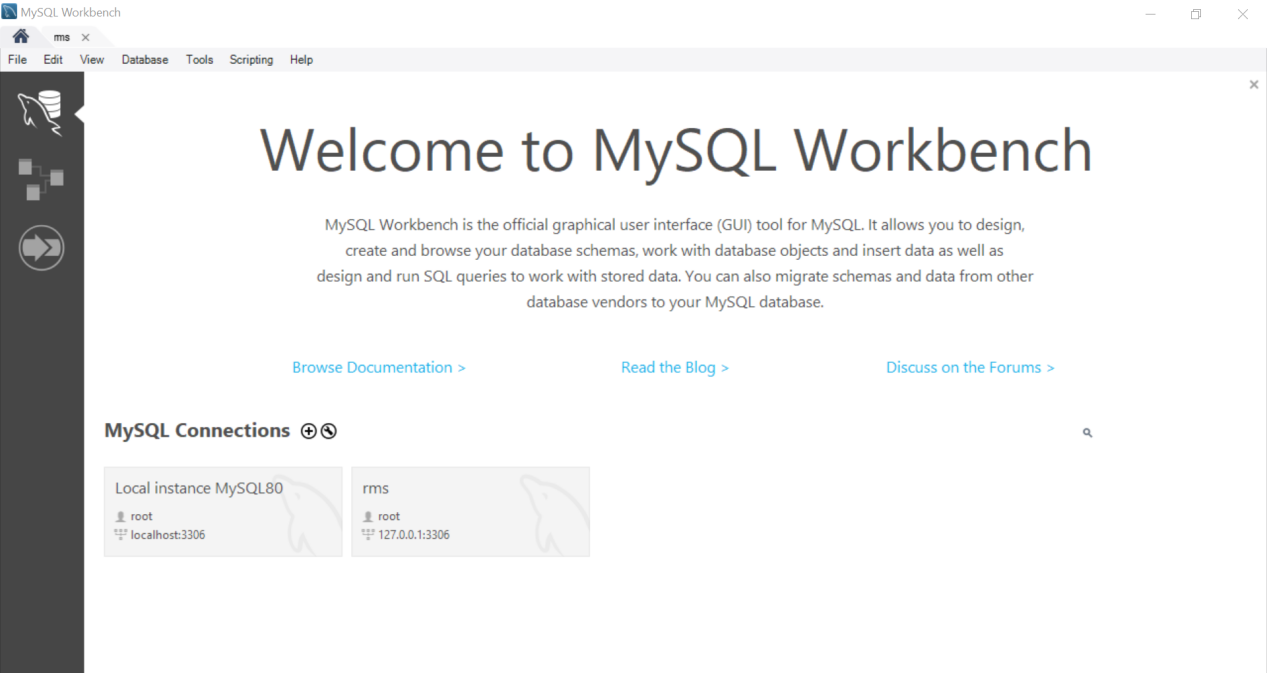
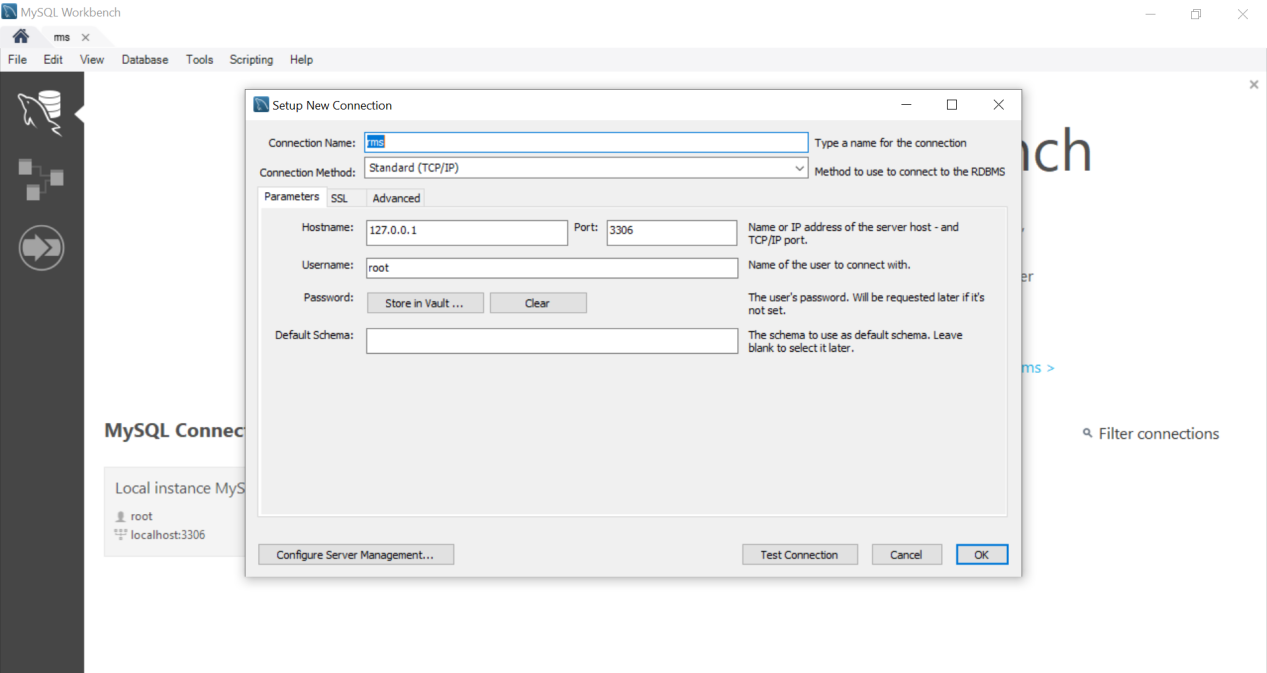
1. View Order (ADMIN)
2. Accept Order (ADMIN)
3. Status of Order (ADMIN/CUSTOMER)
4. Home Delivery (CUSTOMER)
5. Reservation (CUSTOMER)
6. Changing profile data (CUSTOMER)
7. Adding tables for customers (ADMIN)
8. Reserving/Canceling Table Reservations (Admin)
9. Updating Chairs in a Table(Admin)
10. Dine-IN option for customers
11. Viewing previous orders
12. Generating Bill according to the items
13. Changing food Items price for every item(Admin)

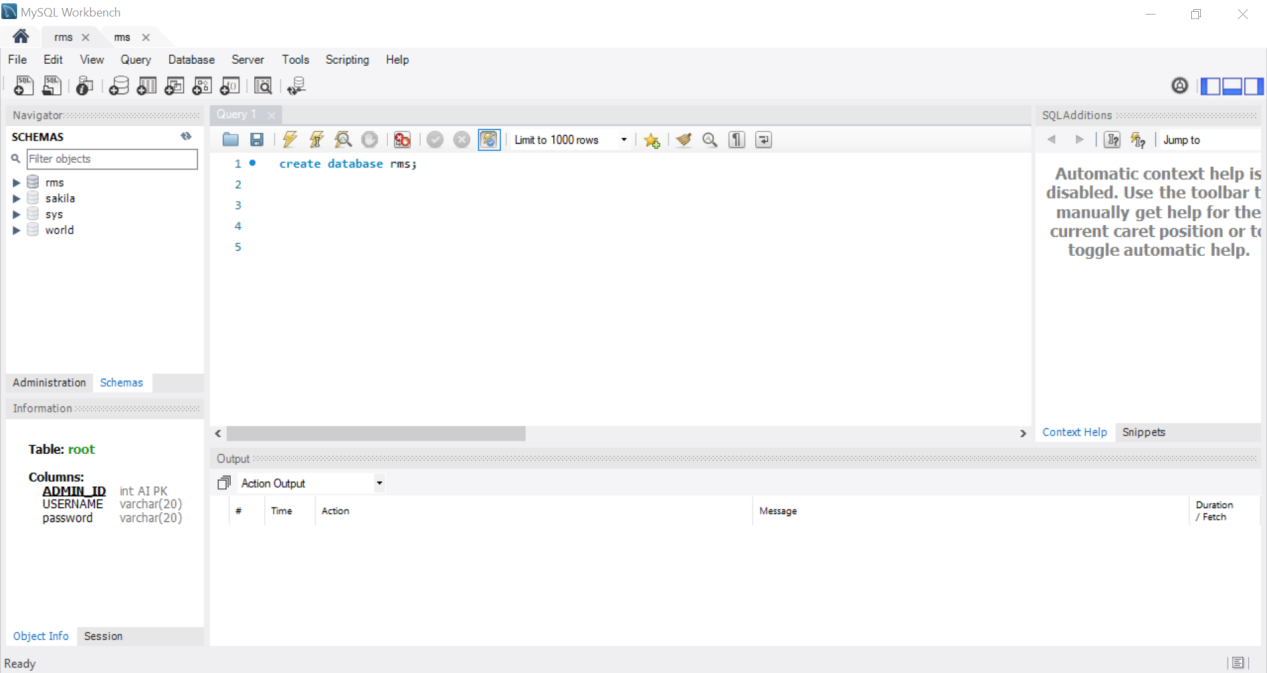
SETTING UP PROJECT

**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**

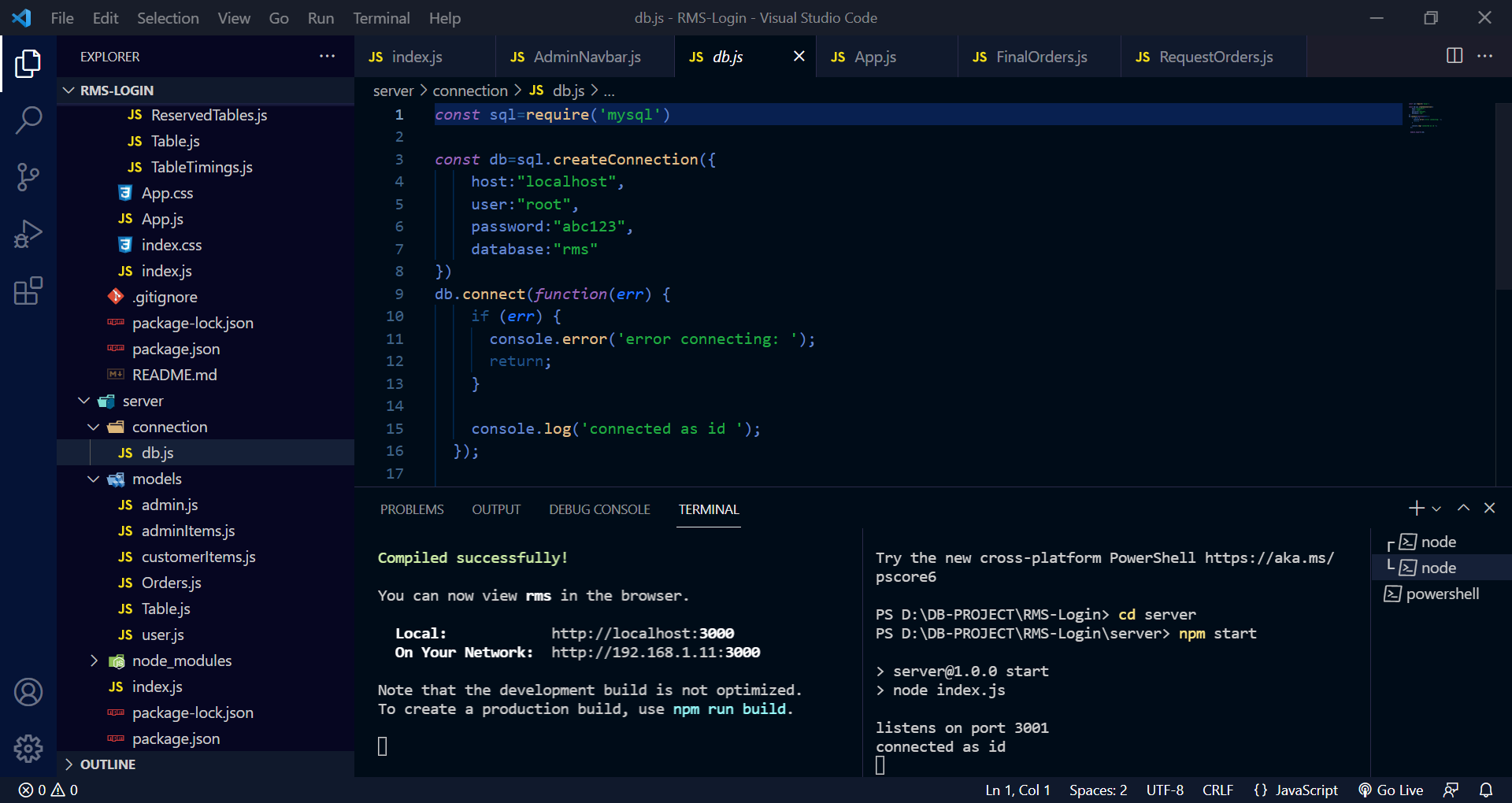
**Step 5: Create connection as we have created(username,password can vary )**

**Step 6: change directory to server**

**Step 7: run npm start**

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

**Step 9:run npm start**

**11**