JSS MAHAVIDYAPEETHA
JSS SCIENCE AND TECHNOLOGY UNIVERSITY
JSS Technical Institutions Campus, Mysuru – 570006

“Zomato clone”

Mini project report submitted in partial fulfillment of curriculum prescribed for the Database Management Systems (20CS510) course for the award of the degree of

BACHELOR OF ENGINEERING
IN
COMPUTER SCIENCE AND ENGINEERING

by

Divesh Kumar Chordia
(01JST20CS049)

Pavan Prakash
(01JST20CS109)

Shaik Mohd Mohsin
(01JST20CS145)

Darshan P S
(01JST20CS045)

Under the Guidance of

Dr. Manimala S
Associate Professor,
Dept.of CS & E,
SJCE, JSS STU Mysore

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
December 2022

JSS MAHAVIDYAPEETHA
JSS SCIENCE AND TECHNOLOGY UNIVERSITY
JSS Technical Institutions Campus, Mysuru – 570006

CERTIFICATE

This is to certify that the work entitled **“Zomato Clone”** is a bonafied work carried out **by Divesh Kumar Chordia, Pavan Prakash, Shaik Mohd Mohsin, and Darshan P S** in partial fulfillment of the award of the degree of **Bachelor of Engineering in Computer Science and Engineering of JSS Science and Technology, Mysuru during the year 2022**. It is certified that all corrections / suggestions indicated during CIE have been incorporated in the report. The mini project report has been approved as it satisfies the academic requirements in respect of mini project work prescribed for the Database Management Systems (20CS510) course.

Course in Charge and Guide

Dr. Manimala S
Associate Professor,
Dept.of CS & E,
JSS STU Mysore

Place: Mysore

Date :

CONTENTS

Introduction

Zomato is an online food ordering website, which is India’s biggest food delivery service. This application is used across India.

Zomato is an Indian multinational restaurant aggregator and food delivery company founded by Pankaj Chaddah and Deepinder Goyal in 2008. Zomato provides information, menus and user-reviews of restaurants as well as food delivery options from restaurants in almost all cities across the length and breadth of India.

Currently, there are over 100 million Zomato users worldwide and people use the application to get in touch with other foodies around the globe so that they can get the best of food in whatever place possible.

This being an added advantage, big restaurants from around the globe post their advertisements on the Zomato app and this is basically how Zomato makes its money. As of 2018, Zomato clocks 3 million orders on a monthly basis with 1.4 million restaurants listed on its platform. It is available in 10,000 cities and 23 countries worldwide.

Another form of income for them is the sale of tickets for food festivals that they partner with or sponsor.

1.1 Objective of the project:

In an Zomato clone, the main purpose of a database is to store information for retrieving the order details, customer information, delivery attributer, bill and payment methods, restaurant attributes. One of the biggest benefits of using a database for Zomato is structuring vast amounts of order and user data.

The Zomato clone System will be this system will provide the detailed description of the order to users sothat they can compare to the different dishes from a particular restaurant or from a different restaurant and will buy the one which is tastiest to them.

1.2 Features of the project:

1)Users have separate user IDs while creating account

Users will be provided with a separate user ID and this will be connected to their email id and their phone number.

2)Clean and simple user interface
The application is equipped with a clean and easy to use user interface that makes ordering food, payment and delivery tracking very easy.

3)Various options to order from
The application can allow us to show various categories of food like different restaurants and their menus. Users can select the restaurant in which they would be ordering their food in and can add the blog content later on accordingly to the chosen category.

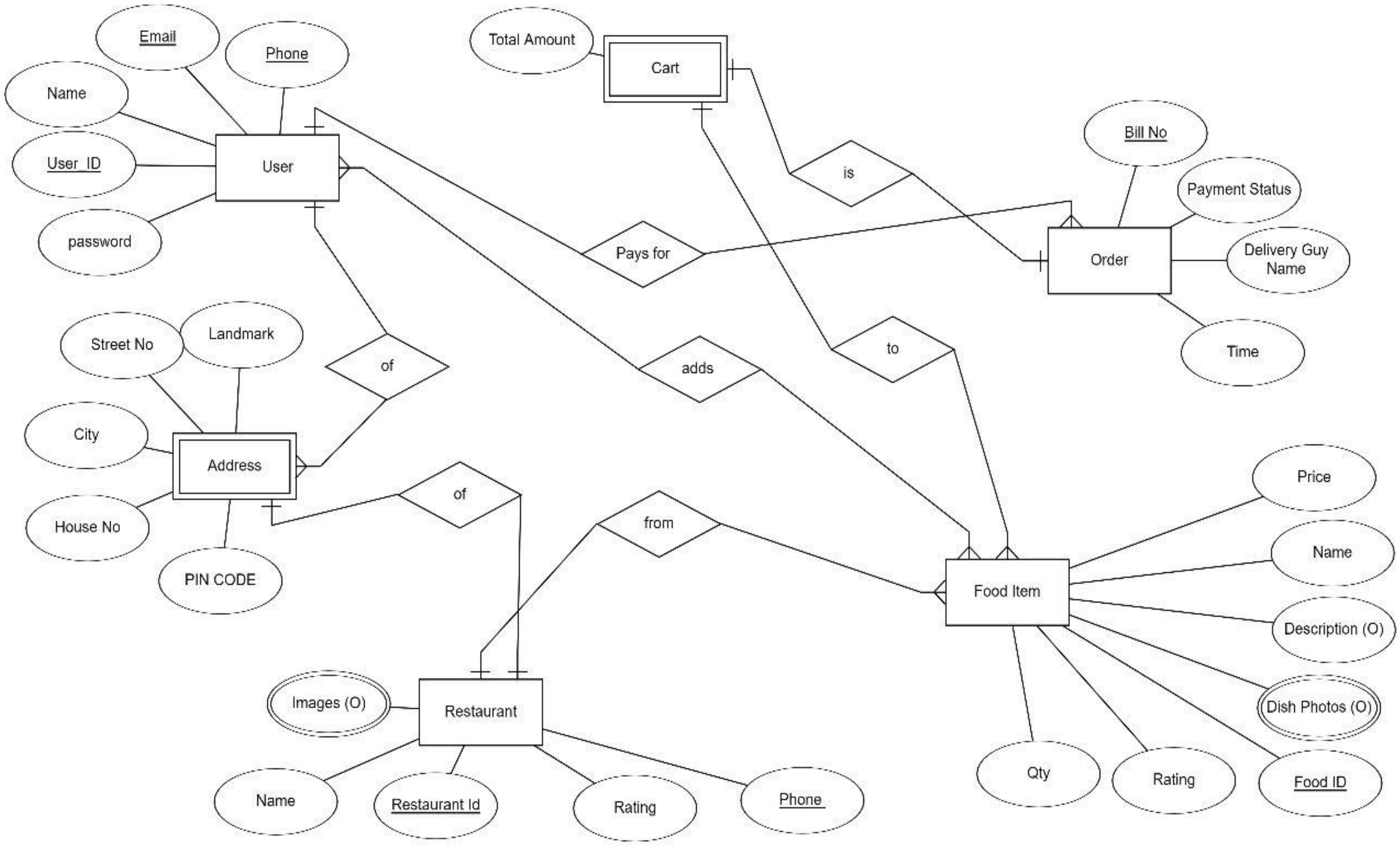
4)User-Friendly Design
Perhaps the most important feature customers want in a Zomato site is a good user experience. If your customers can't find their way around your website or they struggle to find what they're looking for, they will likely move swiftly on to one of the many other comparators.

5) Cart
A shopping cart is one of the key features of Zomato website that fills in the gap between menu, the order and the user purchasing the food from the restaurant.It is a software tool that enables the actual purchase. This tool acquires a customer's id andthe menu id along with the id of the restaurant from which the food is being ordered.

6) Paid Advertisements
Brands can endorse their content on the website in the form of clickable ad banners or even aspaid pictures for pushing their products / content to the audiences available on the website.

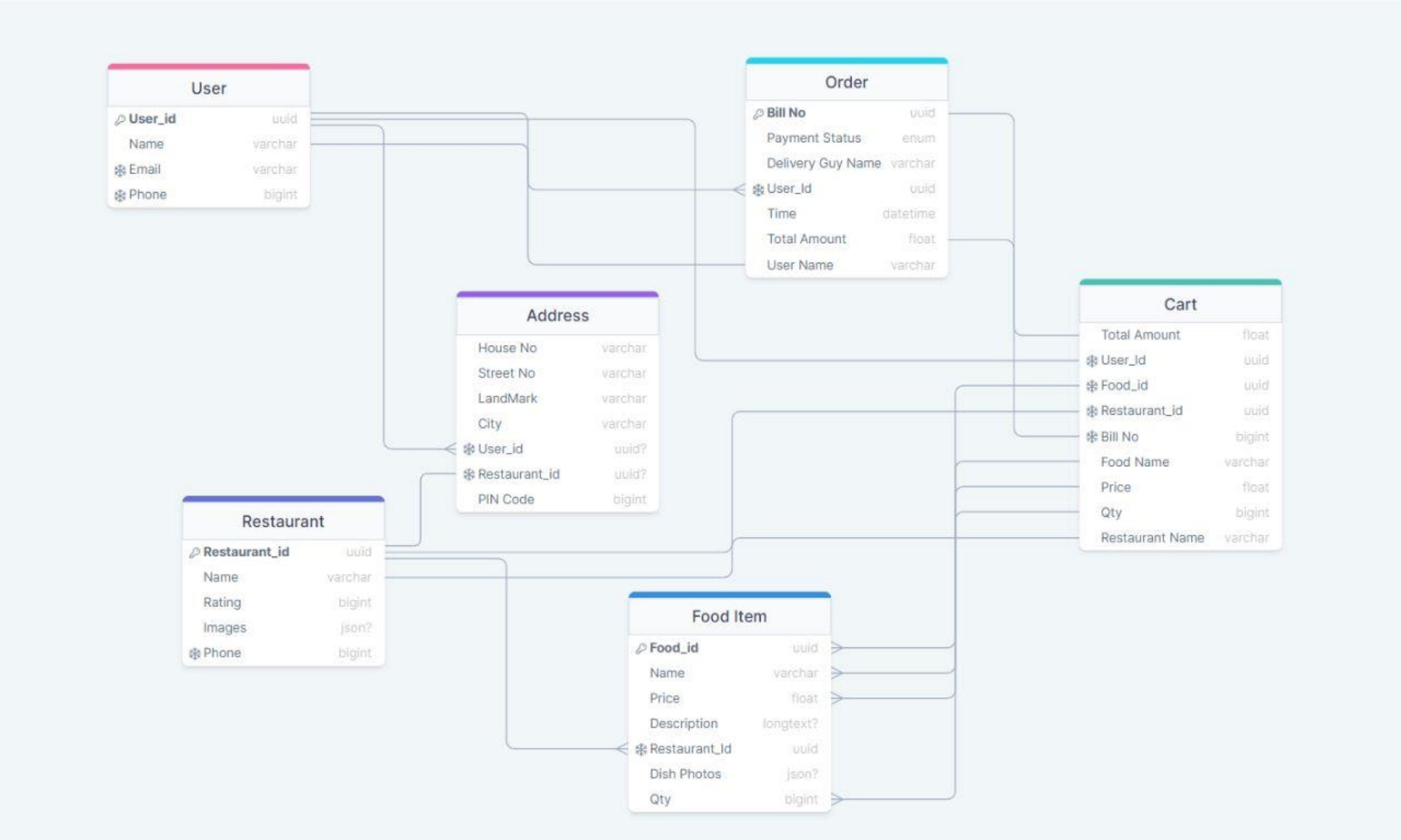
System design

2.1.ER Diagram-high level data modelling



ER DIAGRAM OF ZOMATO CLONE (FOOD ORDERING WEBSITE)

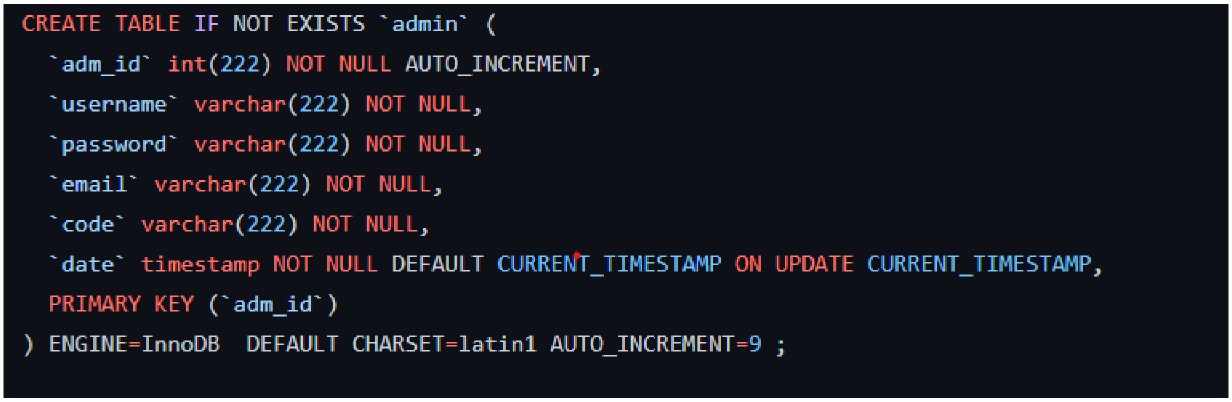
2.2. Schema Diagram-conceptual data modelling



Implementation

3.1 Backend Implementation

3.1.1 DATABASE TABLES CREATION



```
CREATE TABLE IF NOT EXISTS `admin` (
  `adm_id` int(222) NOT NULL AUTO_INCREMENT,
  `username` varchar(222) NOT NULL,
  `password` varchar(222) NOT NULL,
  `email` varchar(222) NOT NULL,
  `code` varchar(222) NOT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`adm_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=9 ;
```

This is a SQL query that creates a table called "admin". The CREATE TABLE IF NOT EXISTS statement ensures that the table is only created if it does not already exist. The table has six columns:

adm_id: an integer column with a maximum value of 222 and is set as the primary key of the table, which means that it is unique and identifies each row. The AUTO_INCREMENT attribute allows the value to automatically increase for each new row that is added to the table.

username: a string column (varchar) with a maximum length of 222 characters, which stores the username.

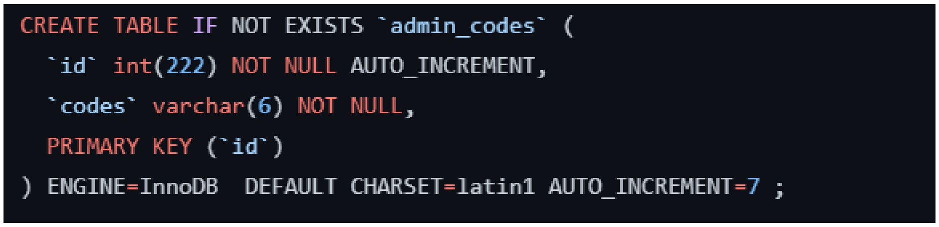
password: a string column (varchar) with a maximum length of 222 characters, which stores the password.

email: a string column (varchar) with a maximum length of 222 characters, which stores the email address.

code: a string column (varchar) with a maximum length of 222 characters, which stores a code.

date: a timestamp column that stores the date and time of the latest update to the row. The DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP attribute sets the default value to the current timestamp, which is automatically updated whenever the row is updated.

The table is stored using the InnoDB engine and the default character set is set to Latin1. The AUTO_INCREMENT attribute is set to 9.

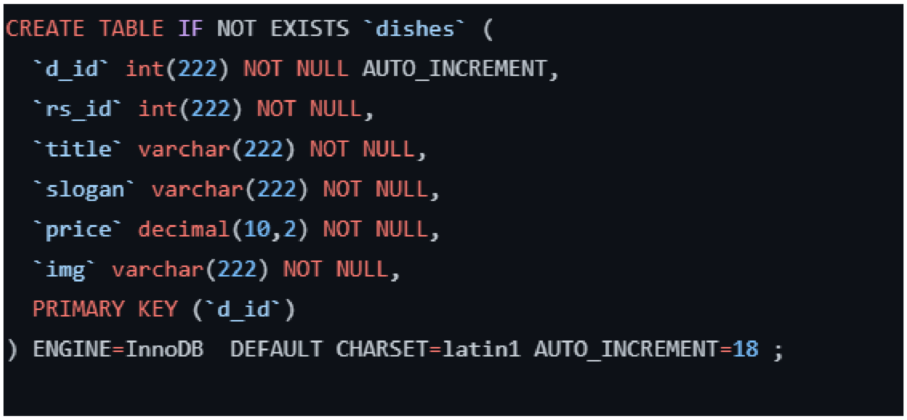


```
CREATE TABLE IF NOT EXISTS `admin_codes` (
  `id` int(222) NOT NULL AUTO_INCREMENT,
  `codes` varchar(6) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=7 ;
```

This is a SQL query that creates a table called "admin_codes". The **CREATE TABLE IF NOT EXISTS** statement ensures that the table is only created if it does not already exist. The table has two columns:

- **id**: an integer column with a maximum value of 222 and is set as the primary key of the table, which means that it is unique and identifies each row. The **AUTO_INCREMENT** attribute allows the value to automatically increase for each new row that is added to the table.
- **codes**: a string column (varchar) with a maximum length of 6 characters, which stores a code.

The table is stored using the InnoDB engine and the default character set is set to Latin1. The **AUTO_INCREMENT** attribute is set to 7.

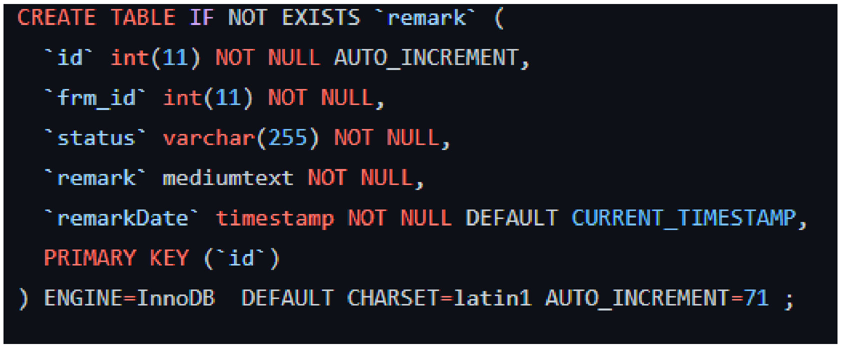


```
CREATE TABLE IF NOT EXISTS `dishes` (
  `d_id` int(222) NOT NULL AUTO_INCREMENT,
  `rs_id` int(222) NOT NULL,
  `title` varchar(222) NOT NULL,
  `slogan` varchar(222) NOT NULL,
  `price` decimal(10,2) NOT NULL,
  `img` varchar(222) NOT NULL,
  PRIMARY KEY (`d_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=18 ;
```

This is a SQL query that creates a table called "dishes". The **CREATE TABLE IF NOT EXISTS** statement ensures that the table is only created if it does not already exist. The table has six columns:

- **d_id**: an integer column with a maximum value of 222 and is set as the primary key of the table, which means that it is unique and identifies each row. The **AUTO_INCREMENT** attribute allows the value to automatically increase for each new row that is added to the table.
- **rs_id**: an integer column with a maximum value of 222, which stores the id of the restaurant.
- **title**: a string column (varchar) with a maximum length of 222 characters, which stores the title of the dish.
- **slogan**: a string column (varchar) with a maximum length of 222 characters, which stores a short description of the dish.
- **price**: a decimal column with a maximum value of 10 digits and 2 decimal places, which stores the price of the dish.
- **img**: a string column (varchar) with a maximum length of 222 characters, which stores the path or URL to the image of the dish.

The table is stored using the InnoDB engine and the default character set is set to Latin1. The **AUTO_INCREMENT** attribute is set to 18.



```
CREATE TABLE IF NOT EXISTS `remark` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `frm_id` int(11) NOT NULL,
  `status` varchar(255) NOT NULL,
  `remark` mediumtext NOT NULL,
  `remarkDate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=71 ;
```

The query creates a table in a database named "remark". The table has 5 columns: "id", "frm_id", "status", "remark", and "remarkDate".

"id" column is an integer with a max value of 11 and is set as the primary key, which means it will be unique for each row in the table.

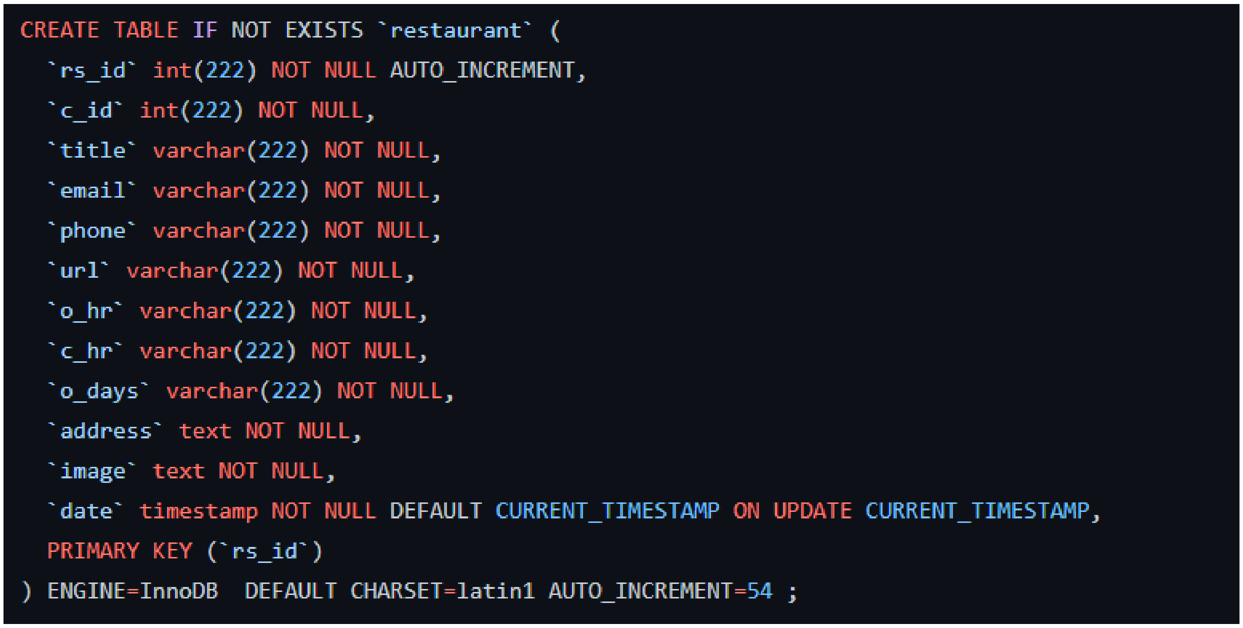
"frm_id" column is also an integer with a max value of 11 and it stores an identifier related to the remark.

"status" column is a string with a maximum length of 255 characters and stores the status of the remark.

"remark" column is a text field with medium length to store the actual remark.

"remarkDate" column is a timestamp that records the date and time of when the remark was made, and it is set to automatically insert the current timestamp.

The table is created using the "InnoDB" engine and has a default character set of "latin1". The AUTO_INCREMENT value for the "id" column is set to 71.



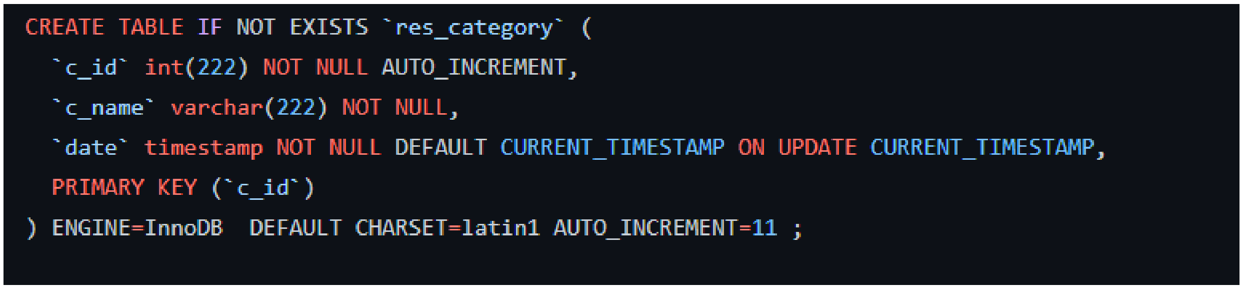
```
CREATE TABLE IF NOT EXISTS `restaurant` (
  `rs_id` int(222) NOT NULL AUTO_INCREMENT,
  `c_id` int(222) NOT NULL,
  `title` varchar(222) NOT NULL,
  `email` varchar(222) NOT NULL,
  `phone` varchar(222) NOT NULL,
  `url` varchar(222) NOT NULL,
  `o_hr` varchar(222) NOT NULL,
  `c_hr` varchar(222) NOT NULL,
  `o_days` varchar(222) NOT NULL,
  `address` text NOT NULL,
  `image` text NOT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`rs_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=54 ;
```

This query creates a table named **restaurant** in a database using the MySQL database management system. The table has 11 columns:

1. **rs_id**: a unique identifier for each record in the table, assigned automatically by the database.
2. **c_id**: an integer that specifies the category the restaurant belongs to.
3. **title**: the name of the restaurant.

4. **email**: the email address of the restaurant.
5. **phone**: the phone number of the restaurant.
6. **url**: the website URL of the restaurant.
7. **o_hr**: the opening hours of the restaurant.
8. **c_hr**: the closing hours of the restaurant.
9. **o_days**: the days the restaurant is open.
10. **address**: the physical address of the restaurant.
11. **image**: a text field storing the image URL of the restaurant.

The table is set up with the **InnoDB** engine and the **latin1** character set, and the default value for the auto-increment column **rs_id** is 54.

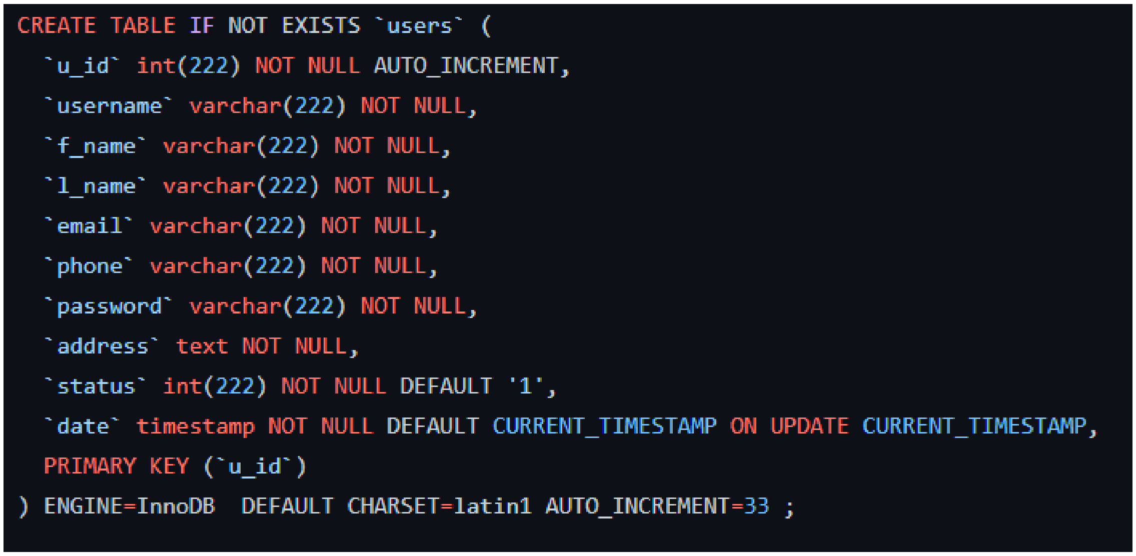


```
CREATE TABLE IF NOT EXISTS `res_category` (
  `c_id` int(222) NOT NULL AUTO_INCREMENT,
  `c_name` varchar(222) NOT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`c_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=11 ;
```

This SQL statement creates a table named "res_category" in a database if it does not already exist. The table has three columns:

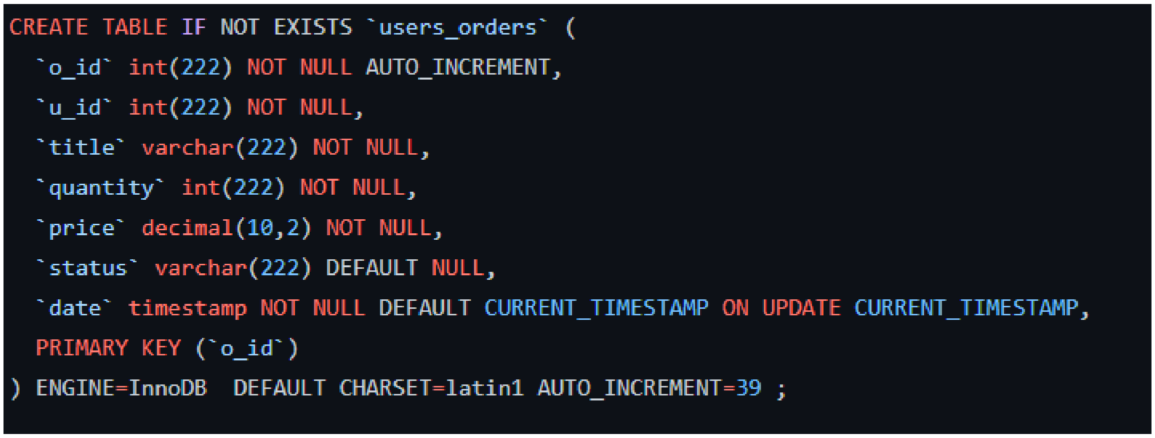
- "c_id" is an auto-incrementing integer that will be the primary key of the table
- "c_name" is a string with a maximum length of 222 characters that will store the name of the restaurant category
- "date" is a timestamp that will store the date and time the category was created or updated

The table will be created using the InnoDB engine and the default character set is latin1. The first auto-incremented value of the "c_id" column will be 11.



```
CREATE TABLE IF NOT EXISTS `users` (
  `u_id` int(222) NOT NULL AUTO_INCREMENT,
  `username` varchar(222) NOT NULL,
  `f_name` varchar(222) NOT NULL,
  `l_name` varchar(222) NOT NULL,
  `email` varchar(222) NOT NULL,
  `phone` varchar(222) NOT NULL,
  `password` varchar(222) NOT NULL,
  `address` text NOT NULL,
  `status` int(222) NOT NULL DEFAULT '1',
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`u_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=33 ;
```

This SQL statement creates a table called **users** with 9 columns: **u_id**, **username**, **f_name**, **l_name**, **email**, **phone**, **password**, **address**, and **date**. The **u_id** column is set as the primary key and has an auto-increment value starting from 33. The table has a default character set of **latin1** and uses the **InnoDB** storage engine. The **status** column is set to have a default value of 1 and the **date** column will automatically update with the current timestamp whenever there is a change in the table.



```
CREATE TABLE IF NOT EXISTS `users_orders` (
  `o_id` int(222) NOT NULL AUTO_INCREMENT,
  `u_id` int(222) NOT NULL,
  `title` varchar(222) NOT NULL,
  `quantity` int(222) NOT NULL,
  `price` decimal(10,2) NOT NULL,
  `status` varchar(222) DEFAULT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`o_id`)
) ENGINE=InnoDB  DEFAULT CHARSET=latin1 AUTO_INCREMENT=39 ;
```

This query creates a table named "users_orders" in a relational database management system. The table has 7 columns:

- o_id: an auto-incrementing primary key of type integer with a maximum length of 222.
- u_id: an integer column that holds the user ID.
- title: a varchar column with a maximum length of 222 that holds the title of the order.
- quantity: an integer column that holds the quantity of the order.
- price: a decimal column with 10 total digits and 2 decimal places that holds the price of the order.
- status: a varchar column with a maximum length of 222 that holds the status of the order, which can be set to NULL.
- date: a timestamp column that holds the date of the order and automatically updates to the current time whenever the record is updated.

The table uses the InnoDB storage engine and the default character set is latin1. The starting value for the auto-increment column **o_id** is 39.

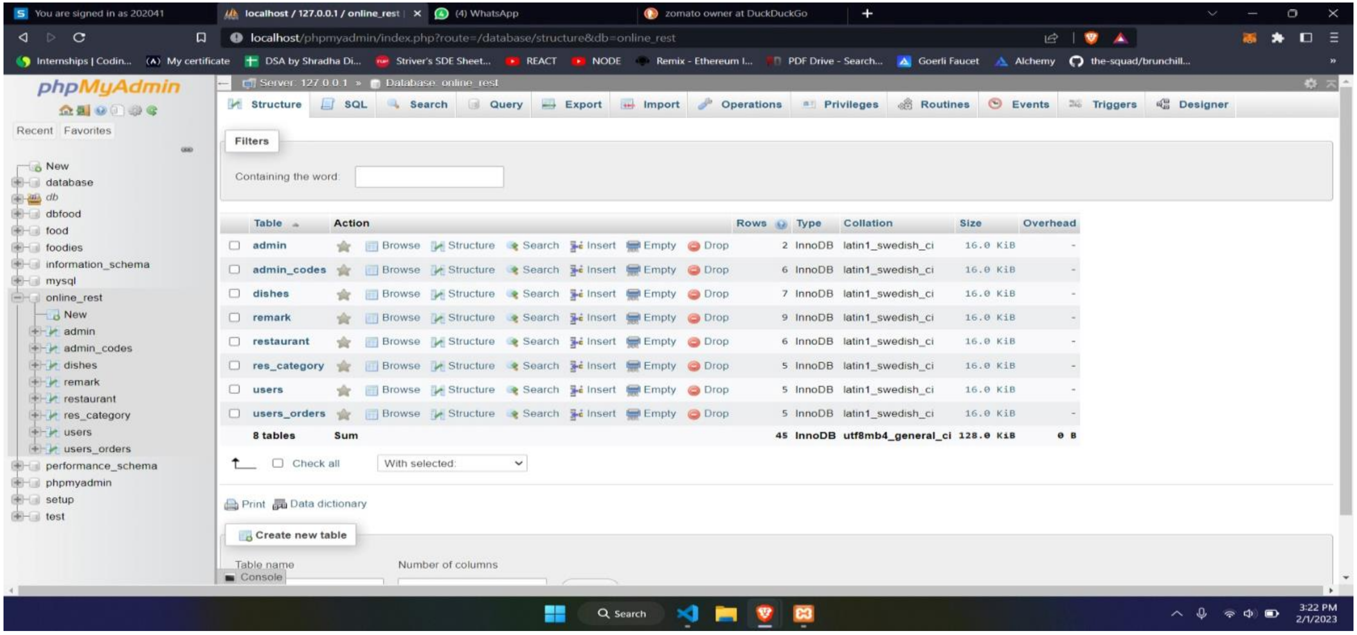
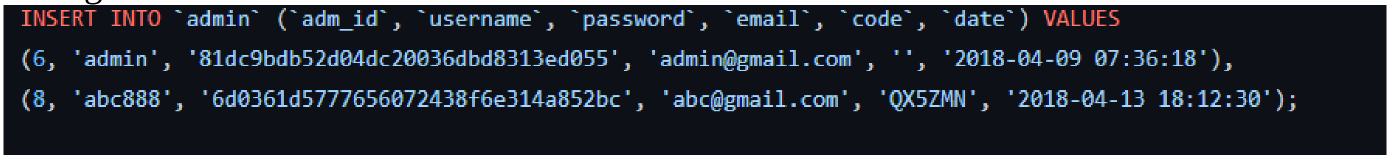
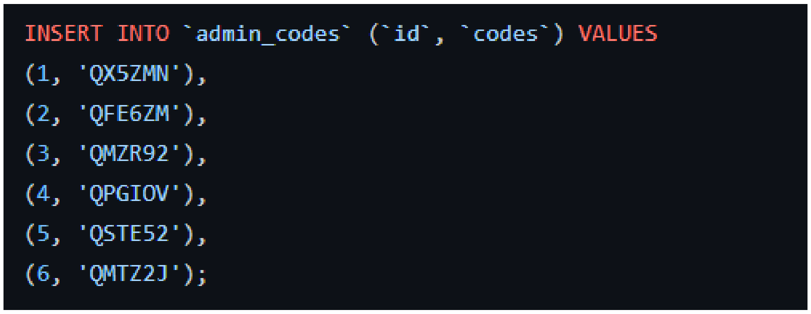


	Table	Action	Rows	Type	Collation	Size	Overhead
□	admin	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	admin_codes	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	dishes	Browse Structure Search Insert Empty Drop	7	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	remark	Browse Structure Search Insert Empty Drop	9	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	restaurant	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	res_category	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	users	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
□	users_orders	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16.0 KiB	-
	8 tables	Sum	45	InnoDB	utf8mb4_general_ci	128.0 KiB	0 B

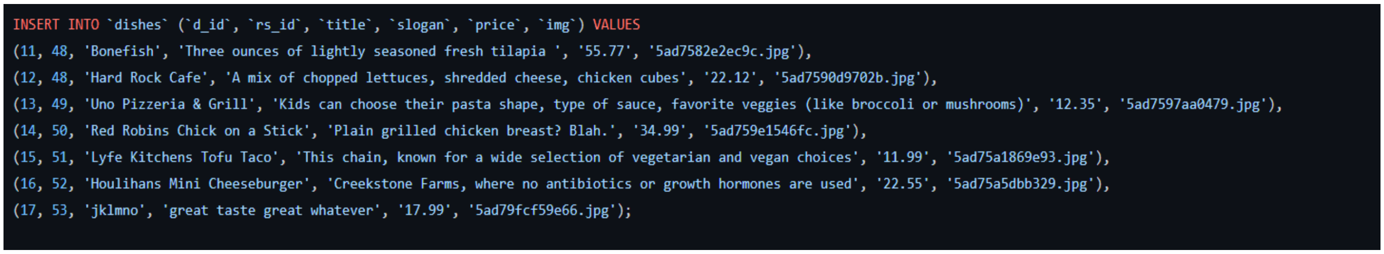
3.1.2 inserting into these tables



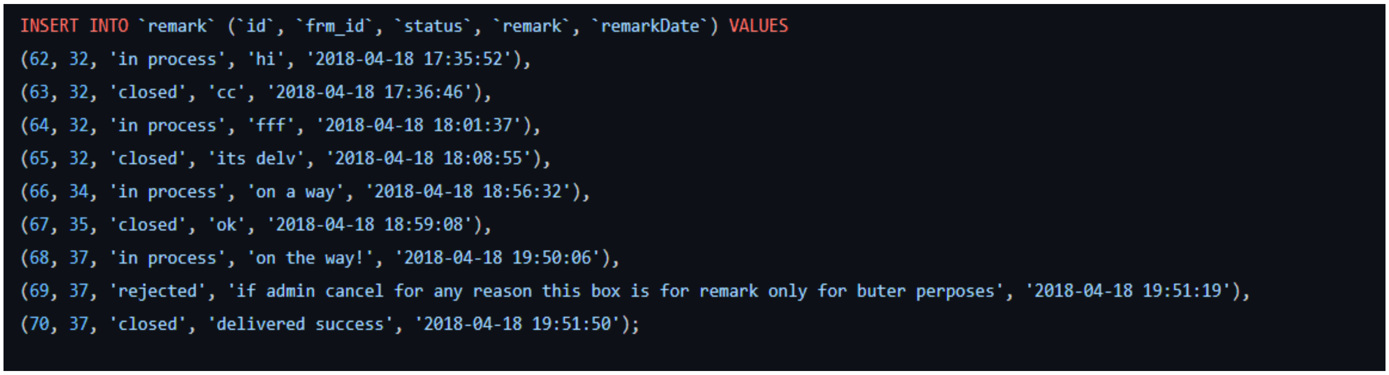
```
INSERT INTO `admin` (`adm_id`, `username`, `password`, `email`, `code`, `date`) VALUES
(6, 'admin', '81dc9bdb52d04dc20036dbd8313ed055', 'admin@gmail.com', '', '2018-04-09 07:36:18'),
(8, 'abc888', '6d0361d5777656072438f6e314a852bc', 'abc@gmail.com', 'QX5ZMN', '2018-04-13 18:12:30');
```



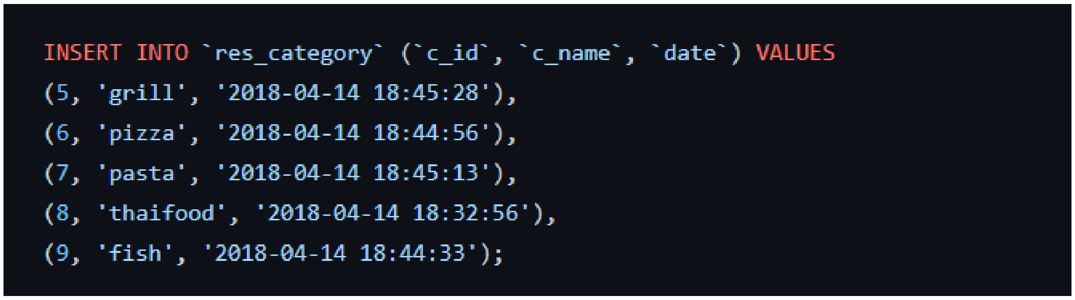
```
INSERT INTO `admin_codes` (`id`, `codes`) VALUES
(1, 'QX5ZMN'),
(2, 'QFE6ZM'),
(3, 'QMZR92'),
(4, 'QPGIOV'),
(5, 'QSTE52'),
(6, 'QMTZ2J');
```



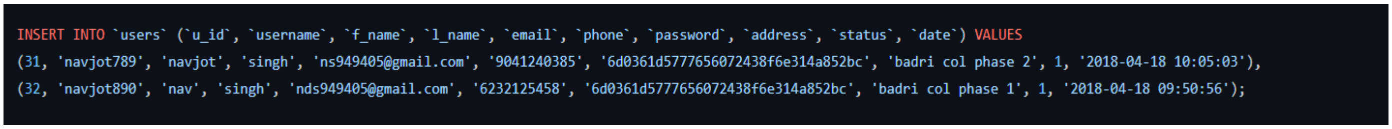
```
INSERT INTO `dishes` (`d_id`, `rs_id`, `title`, `slogan`, `price`, `img`) VALUES
(11, 48, 'Bonefish', 'Three ounces of lightly seasoned fresh tilapia ', '55.77', '5ad7582e2ec9c.jpg'),
(12, 48, 'Hard Rock Cafe', 'A mix of chopped lettuces, shredded cheese, chicken cubes', '22.12', '5ad7590d9702b.jpg'),
(13, 49, 'Uno Pizzeria & Grill', 'Kids can choose their pasta shape, type of sauce, favorite veggies (like broccoli or mushrooms)', '12.35', '5ad7597aa0479.jpg'),
(14, 50, 'Red Robins Chick on a Stick', 'Plain grilled chicken breast? Blah.', '34.99', '5ad759e1546fc.jpg'),
(15, 51, 'Lyfe Kitchens Tofu Taco', 'This chain, known for a wide selection of vegetarian and vegan choices', '11.99', '5ad75a1869e93.jpg'),
(16, 52, 'Houlihans Mini Cheeseburger', 'Creekstone Farms, where no antibiotics or growth hormones are used', '22.55', '5ad75a5dbb329.jpg'),
(17, 53, 'jklmno', 'great taste great whatever', '17.99', '5ad79fcf59e66.jpg');
```



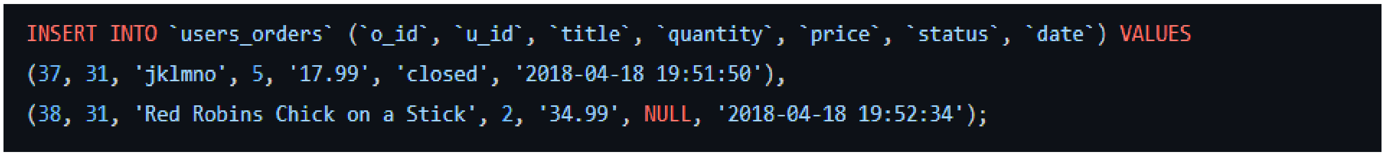
```
INSERT INTO `remark` (`id`, `frm_id`, `status`, `remark`, `remarkDate`) VALUES
(62, 32, 'in process', 'hi', '2018-04-18 17:35:52'),
(63, 32, 'closed', 'cc', '2018-04-18 17:36:46'),
(64, 32, 'in process', 'fff', '2018-04-18 18:01:37'),
(65, 32, 'closed', 'its delv', '2018-04-18 18:08:55'),
(66, 34, 'in process', 'on a way', '2018-04-18 18:56:32'),
(67, 35, 'closed', 'ok', '2018-04-18 18:59:08'),
(68, 37, 'in process', 'on the way!', '2018-04-18 19:50:06'),
(69, 37, 'rejected', 'if admin cancel for any reason this box is for remark only for buter perposes', '2018-04-18 19:51:19'),
(70, 37, 'closed', 'delivered success', '2018-04-18 19:51:50');
```



```
INSERT INTO `res_category` (`c_id`, `c_name`, `date`) VALUES
(5, 'grill', '2018-04-14 18:45:28'),
(6, 'pizza', '2018-04-14 18:44:56'),
(7, 'pasta', '2018-04-14 18:45:13'),
(8, 'thaifood', '2018-04-14 18:32:56'),
(9, 'fish', '2018-04-14 18:44:33');
```

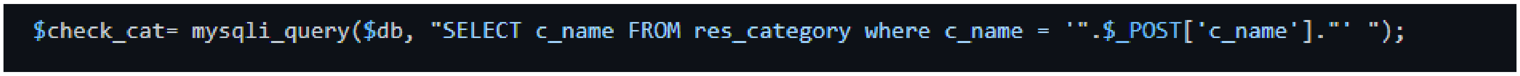


```
INSERT INTO `users` (`u_id`, `username`, `f_name`, `l_name`, `email`, `phone`, `password`, `address`, `status`, `date`) VALUES
(31, 'navjot789', 'navjot', 'singh', 'ns949405@gmail.com', '9041240385', '6d0361d5777656072438f6e314a852bc', 'badri col phase 2', 1, '2018-04-18 10:05:03'),
(32, 'navjot890', 'nav', 'singh', 'nds949405@gmail.com', '6232125458', '6d0361d5777656072438f6e314a852bc', 'badri col phase 1', 1, '2018-04-18 09:50:56');
```



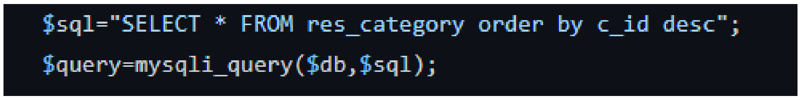
```
INSERT INTO `users_orders` (`o_id`, `u_id`, `title`, `quantity`, `price`, `status`, `date`) VALUES
(37, 31, 'jklmno', 5, '17.99', 'closed', '2018-04-18 19:51:50'),
(38, 31, 'Red Robins Chick on a Stick', 2, '34.99', NULL, '2018-04-18 19:52:34');
```

3.1.3 Queries used in PHP



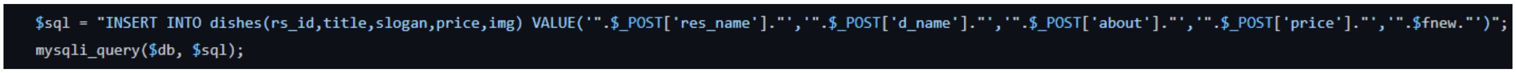
```
$check_cat= mysqli_query($db, "SELECT c_name FROM res_category where c_name = '".$_POST['c_name']."' ");
```

This is a query written in PHP that retrieves data from a database. The query is checking for the existence of a specific category name in a table called "res_category". The query is using the mysqli_query function, which is used to execute a query on a MySQL database. The first argument in the function is a database connection object named $db, and the second argument is the SQL query. The query is selecting the "c_name" column from the "res_category" table where the value of the "c_name" column is equal to the value of the "c_name" key in the $_POST array. The result of the query will be stored in the $check_cat variable.



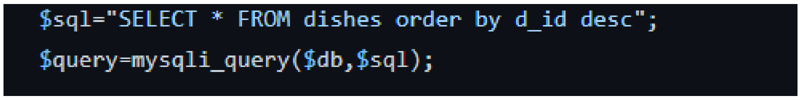
```
$sql="SELECT * FROM res_category order by c_id desc";
$query=mysqli_query($db,$sql);
```

This query is written in PHP, and it is used to retrieve all the data from the "res_category" table in the database. The "mysqli_query" function is used to execute the query. The "SELECT *" clause is used to retrieve all the columns from the table, while the "FROM res_category" clause specifies the table that the data should be retrieved from. The "order by c_id desc" clause sorts the data in descending order based on the "c_id" column. The result of this query will be a list of all the rows in the "res_category" table, sorted by the "c_id" column in descending order.



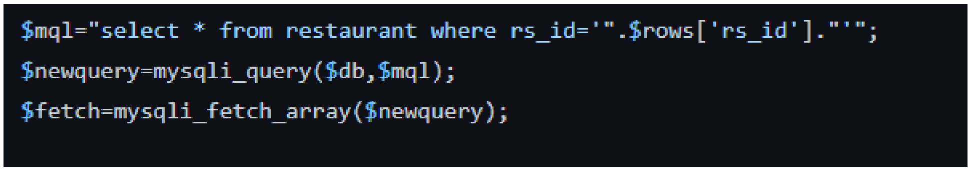
```
$sql = "INSERT INTO dishes(rs_id,title,slogan,price,img) VALUE('".$_POST['res_name']."','".$_POST['d_name']."','".$_POST['about']."','".$_POST['price']."','".$fnew."')";
mysqli_query($db, $sql);
```

This query is written in the PHP scripting language and is used to insert data into a database table named "dishes". The **$_POST** variables are used to get values from a form that the user has submitted. The values are then stored into the respective columns of the table. The **mysqli_query** function is used to execute the SQL query and insert the data into the database. The **$fnew** variable is used to store the name of an image file that the user has uploaded



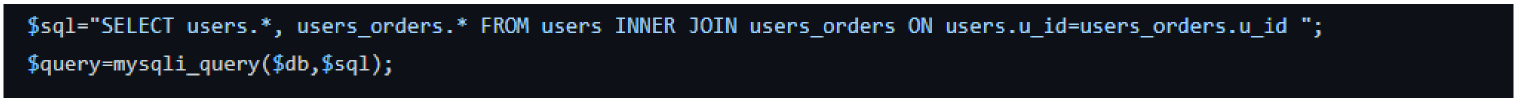
```
$sql="SELECT * FROM dishes order by d_id desc";
$query=mysqli_query($db,$sql);
```

This query retrieves all the records from the "dishes" table and orders them in descending order based on the "d_id" column. The result of this query is stored in the "$query" variable. This query is being executed using the mysqli_query function with the database connection "$db".



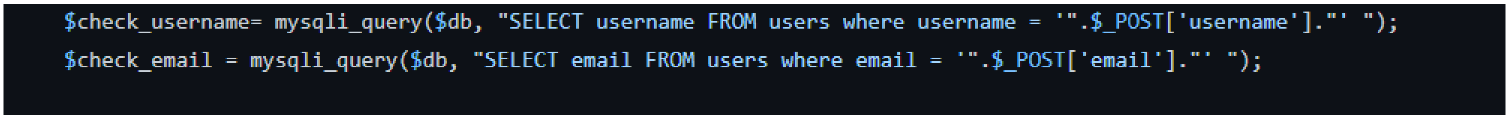
```
$mql="select * from restaurant where rs_id='".$rows['rs_id']."'";
$newquery=mysqli_query($db,$mql);
$fetch=mysqli_fetch_array($newquery);
```

This code is a PHP code that executes a SQL query. It selects data from the "restaurant" table and retrieves the information of the specific restaurant with the id that is saved in the "rows" variable. The query results are stored in the "$newquery" variable, and the data is then fetched using the "mysqli_fetch_array" function, which returns the result set as an associative array. The result set is stored in the "$fetch" variable.



```
$sql="SELECT users.*, users_orders.* FROM users INNER JOIN users_orders ON users.u_id=users_orders.u_id ";
$query=mysqli_query($db,$sql);
```

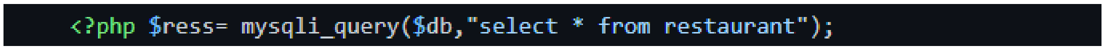
This is a SQL query in PHP that selects data from two tables named "users" and "users_orders" and performs an INNER JOIN between the two tables. The query combines rows from both tables where the values of the "u_id" column in the "users" table match the values of the "u_id" column in the "users_orders" table. The selected data is stored in the $query variable and can be used for further processing.



```
    $check_username= mysqli_query($db, "SELECT username FROM users where username = '".$_POST['username']."' ");
    $check_email = mysqli_query($db, "SELECT email FROM users where email = '".$_POST['email']."' ");
```

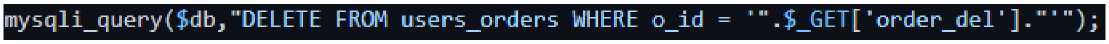
These are two SQL queries written in PHP using the MySQLi library. The first query is checking if a given username exists in the 'users' table in the database. It does this by selecting the 'username' column from the 'users' table where the value of the 'username' column matches the value of the 'username' key in the $_POST array.

The second query is checking if a given email exists in the 'users' table in the database. It does this by selecting the 'email' column from the 'users' table where the value of the 'email' column matches the value of the 'email' key in the $_POST array. Both of these queries are likely used to ensure that new users signing up with a specific username or email address have unique values.



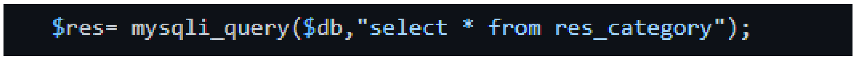
```
    <?php $ress= mysqli_query($db,"select * from restaurant");
```

The query **$ress= mysqli_query($db,"select * from restaurant");** retrieves all the data from the "restaurant" table in the database using the **mysqli_query()** function. This function is used to execute SQL queries and returns the result set as a **mysqli_result** object. The **$db** argument is the database connection that is created previously, and the SQL query "select * from restaurant" selects all the columns from the "restaurant" table. This query retrieves the entire data from the table, which includes the information about restaurants like the title, email, phone, URL, operating hours, closing hours, operating days, address, and image.



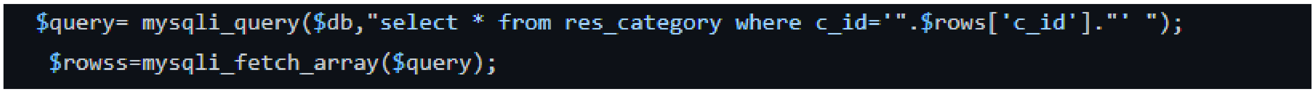
```
mysqli_query($db,"DELETE FROM users_orders WHERE o_id = '".$_GET['order_del']."'");
```

This is a MySQL DELETE statement using the mysqli_query function in PHP. It is used to delete a record from the "users_orders" table in the database connection represented by the variable "$db". The record to be deleted is specified by the o_id field and its value is taken from the $_GET array with the key "order_del". If the query is successful, it will delete the specified record from the "users_orders" table.



```
    $res= mysqli_query($db,"select * from res_category");
```

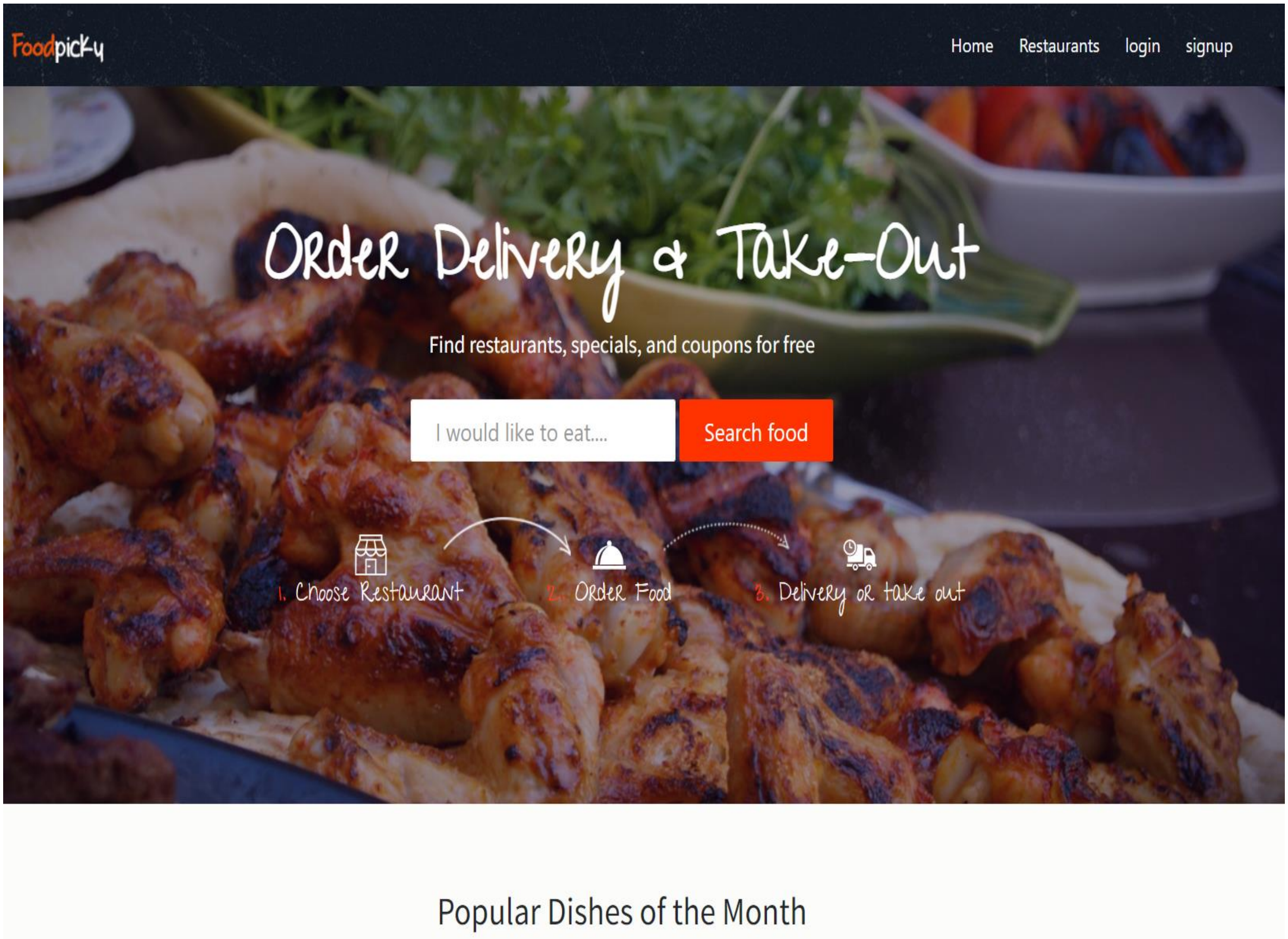
This is a MySQL query using the mysqli library in PHP. The **mysqli_query** function is used to execute a query on the database. The first argument is the database connection, represented by the **$db** variable. The second argument is a string containing a SELECT statement, which selects all columns (represented by the ***** symbol) from the **res_category** table. The resulting query data is stored in a result set represented by the **$res** variable.



```
  $query= mysqli_query($db,"select * from res_category where c_id='".$rows['c_id']."' ");
   $rowss=mysqli_fetch_array($query);
```

This is a PHP code that executes two MySQL queries. The first query selects all columns from the "res_category" table where the "c_id" value equals to the "c_id" value stored in the "$rows" array. The second query uses the mysqli_fetch_array() function to fetch a single row from the result set of the first query as an array, which is then stored in the "$rowss" variable.

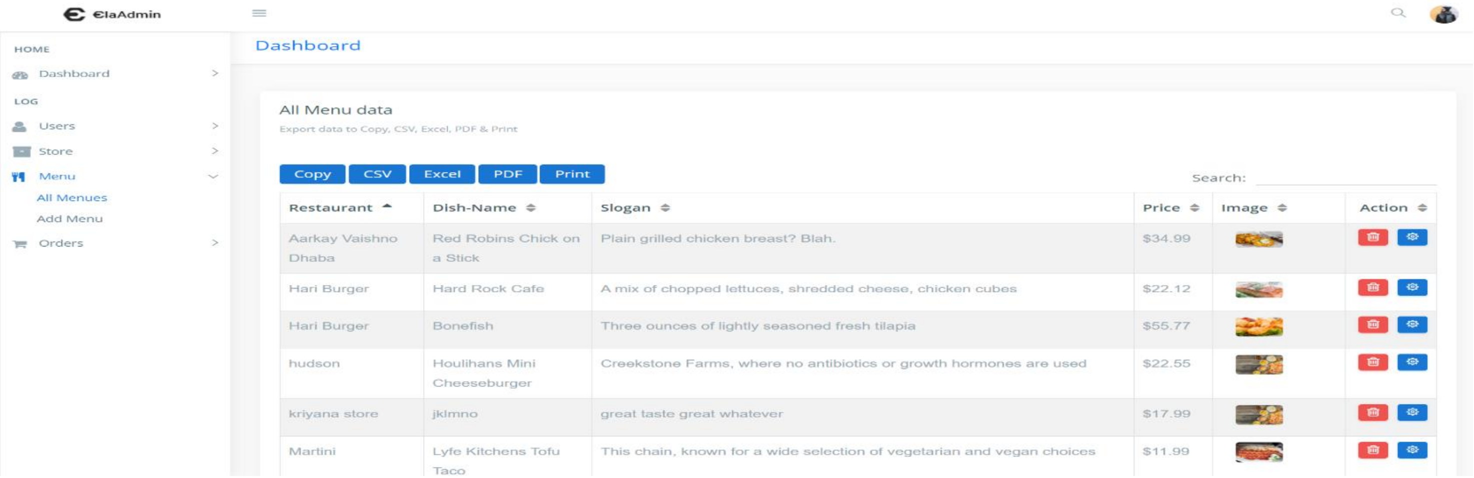
3.2 Frontend Implementation



Order Delivery & Take-Out

Find restaurants, specials, and coupons for free

Popular Dishes of the Month



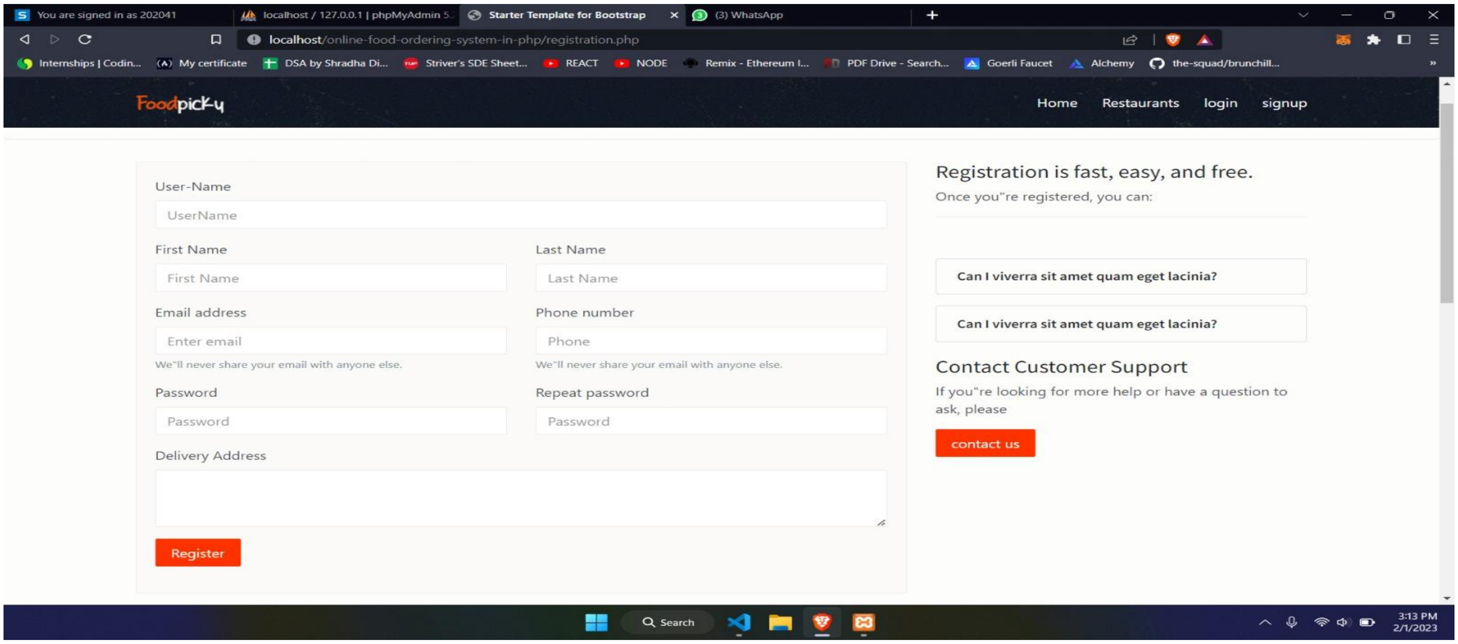
Dashboard

All Menu data

Export data to Copy, CSV, Excel, PDF & Print

Restaurant	Dish-Name	Slogan	Price	Image	Action
Aarkay Vaishno Dhaba	Red Robins Chick on a Stick	Plain grilled chicken breast? Blah.	$34.99		
Hari Burger	Hard Rock Cafe	A mix of chopped lettuces, shredded cheese, chicken cubes	$22.12		
Hari Burger	Bonefish	Three ounces of lightly seasoned fresh tilapia	$55.77		
hudson	Houlihans Mini Cheeseburger	Creekstone Farms, where no antibiotics or growth hormones are used	$22.55		
kriyana store	jklmno	great taste great whatever	$17.99		
Martini	Lyfe Kitchens Tofu Taco	This chain, known for a wide selection of vegetarian and vegan choices	$11.99		

3.3 Ordering food process



Foodpicky

Home Restaurants login signup

User-Name: UserName

First Name: First Name	Last Name: Last Name
Email address: Enter email We"ll never share your email with anyone else.	Phone number: Phone We"ll never share your email with anyone else.
Password: Password	Repeat password: Password

Delivery Address

Register

Registration is fast, easy, and free.

Once you"re registered, you can:

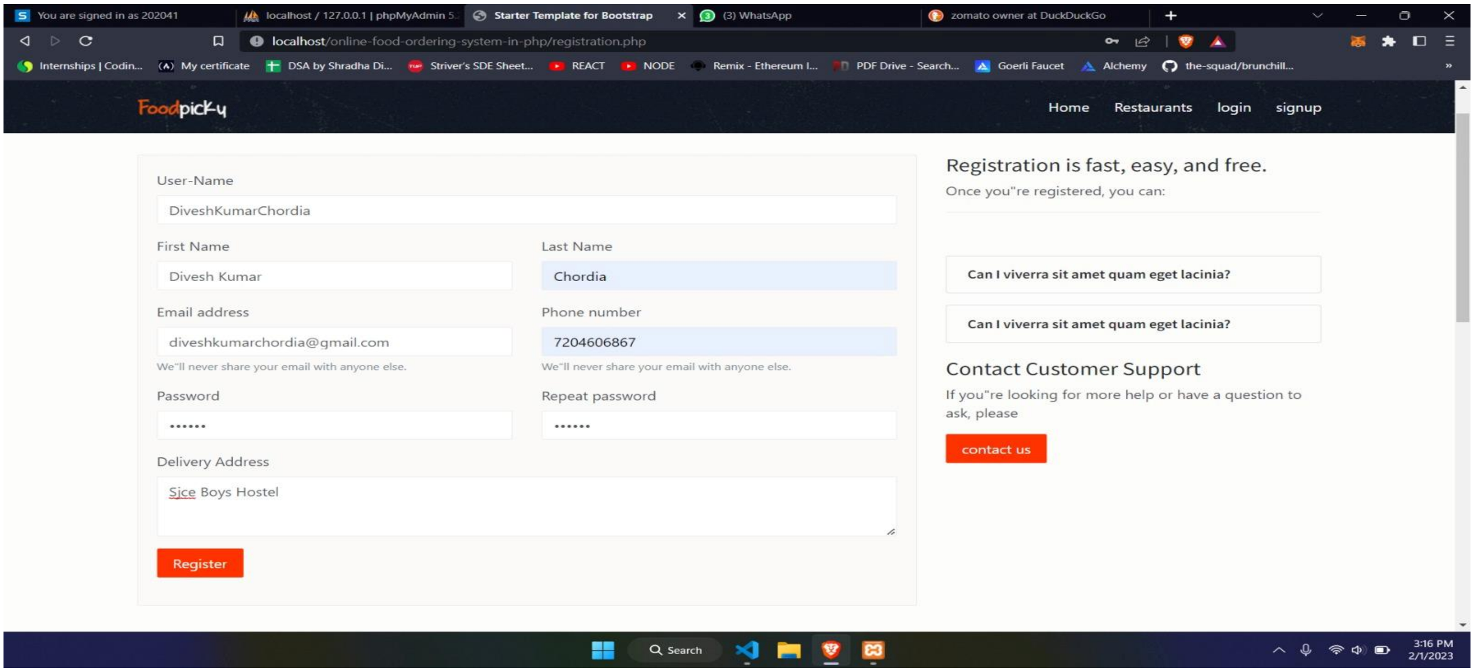
Can I viverra sit amet quam eget lacinia?

Can I viverra sit amet quam eget lacinia?

Contact Customer Support

If you"re looking for more help or have a question to ask, please

contact us



Foodpicky

Home Restaurants login signup

User-Name: DiveshKumarChordia

First Name: Divesh Kumar	Last Name: Chordia
Email address: diveshkumarchordia@gmail.com We"ll never share your email with anyone else.	Phone number: 7204606867 We"ll never share your email with anyone else.
Password: ••••••	Repeat password: ••••••

Delivery Address: Sjce Boys Hostel

Register

Registration is fast, easy, and free.

Once you"re registered, you can:

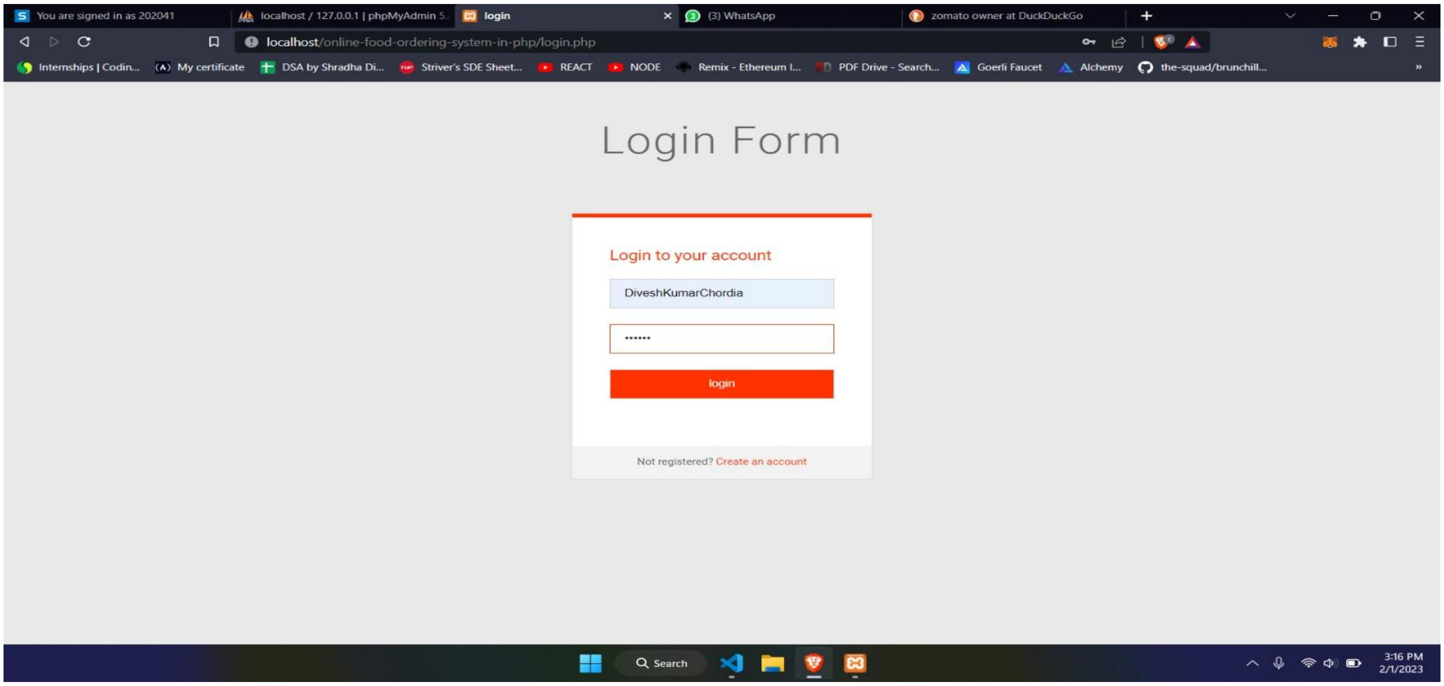
Can I viverra sit amet quam eget lacinia?

Can I viverra sit amet quam eget lacinia?

Contact Customer Support

If you"re looking for more help or have a question to ask, please

contact us



localhost/online-food-ordering-system-in-php/login.php

Login Form

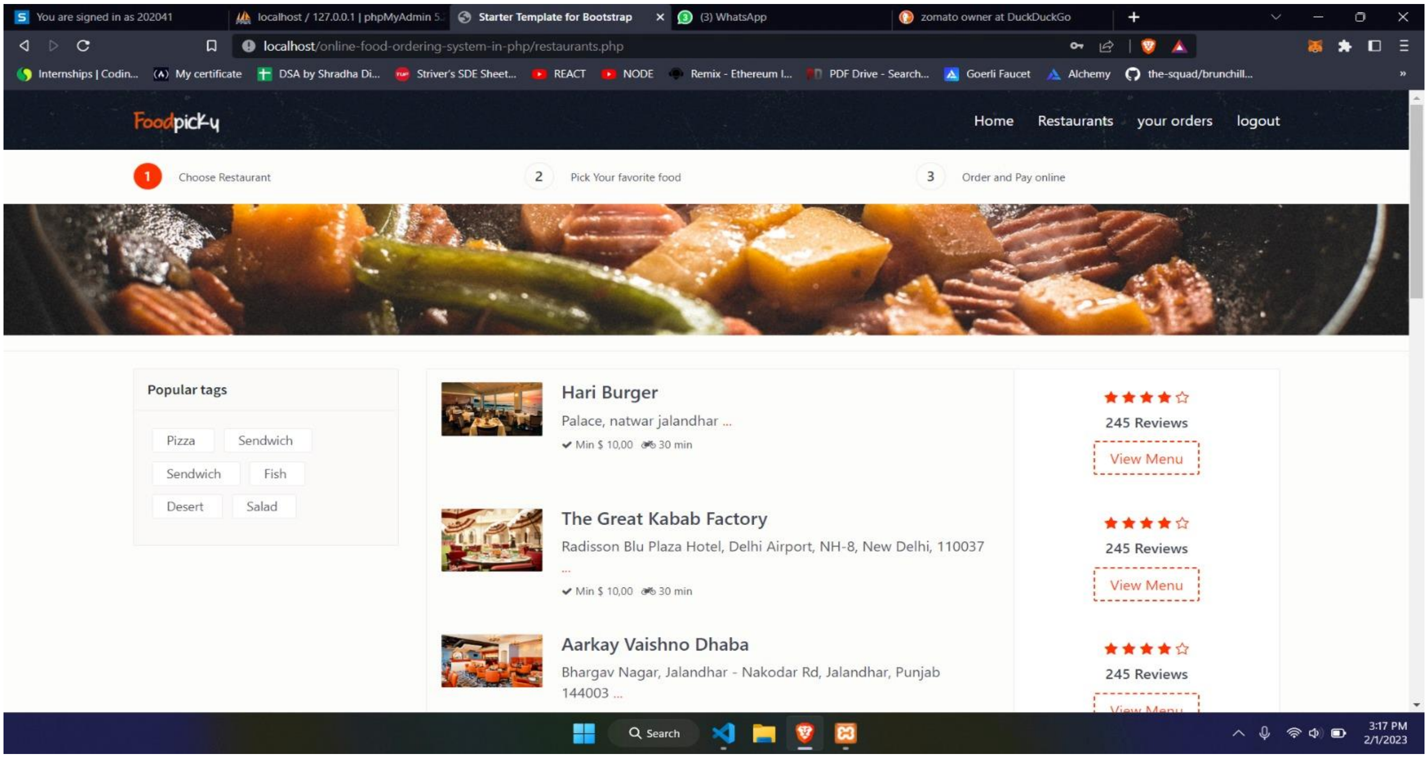
Login to your account

DiveshKumarChordia

••••••

login

Not registered? Create an account



localhost/online-food-ordering-system-in-php/restaurants.php

1 Choose Restaurant

2 Pick Your favorite food

3 Order and Pay online

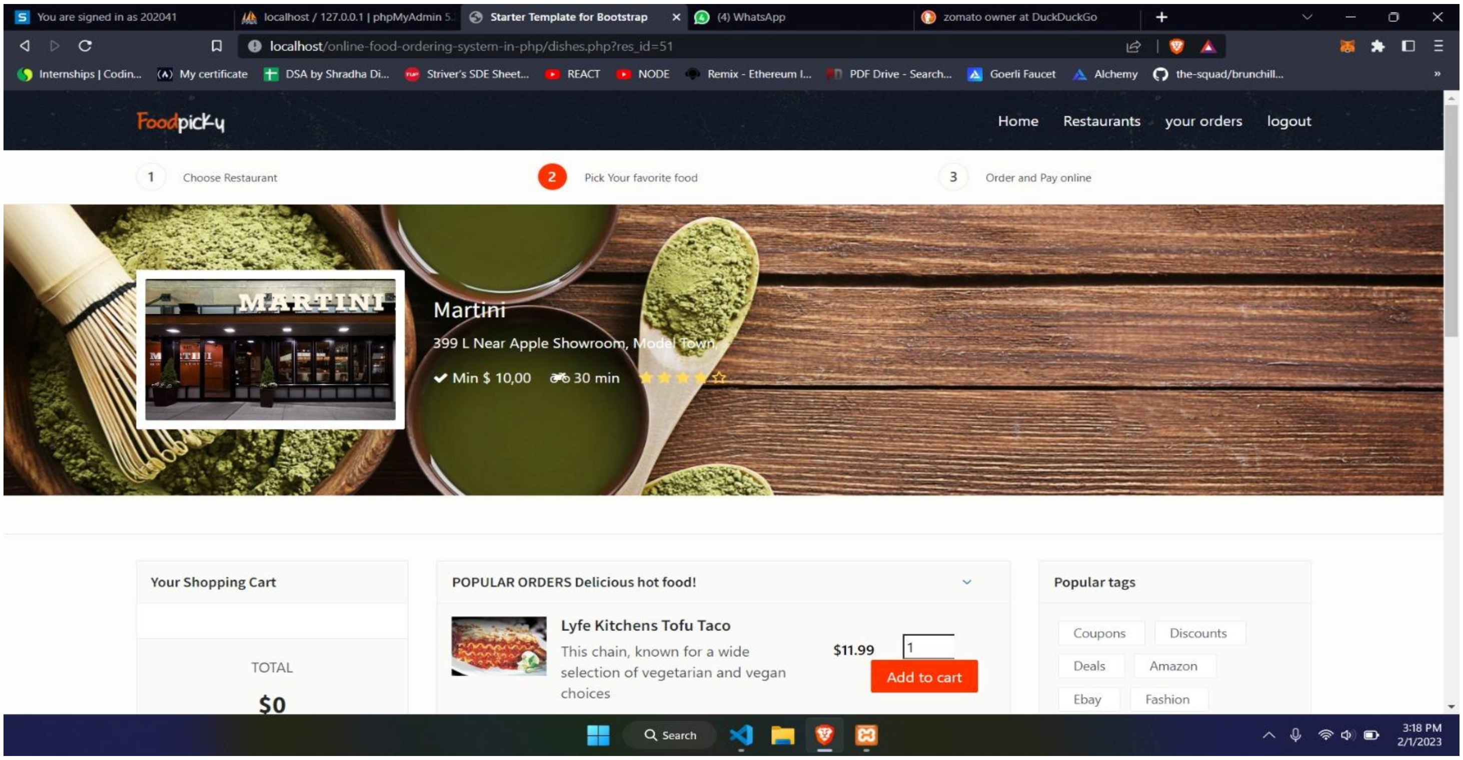
Popular tags

Pizza Sendwich Sendwich Fish Desert Salad

Hari Burger
Palace, natwar jalandhar ...
✓ Min $ 10,00 30 min
★★★★☆ 245 Reviews
View Menu

The Great Kabab Factory
Radisson Blu Plaza Hotel, Delhi Airport, NH-8, New Delhi, 110037 ...
✓ Min $ 10,00 30 min
★★★★☆ 245 Reviews
View Menu

Aarkay Vaishno Dhaba
Bhargav Nagar, Jalandhar - Nakodar Rd, Jalandhar, Punjab 144003 ...
★★★★☆ 245 Reviews
View Menu



localhost/online-food-ordering-system-in-php/dishes.php?res_id=51

1 Choose Restaurant 2 Pick Your favorite food 3 Order and Pay online

Martini

399 L Near Apple Showroom, Model Town

✓ Min $ 10,00 30 min ★★★★☆

Your Shopping Cart

TOTAL

$0

POPULAR ORDERS Delicious hot food!

Lyfe Kitchens Tofu Taco

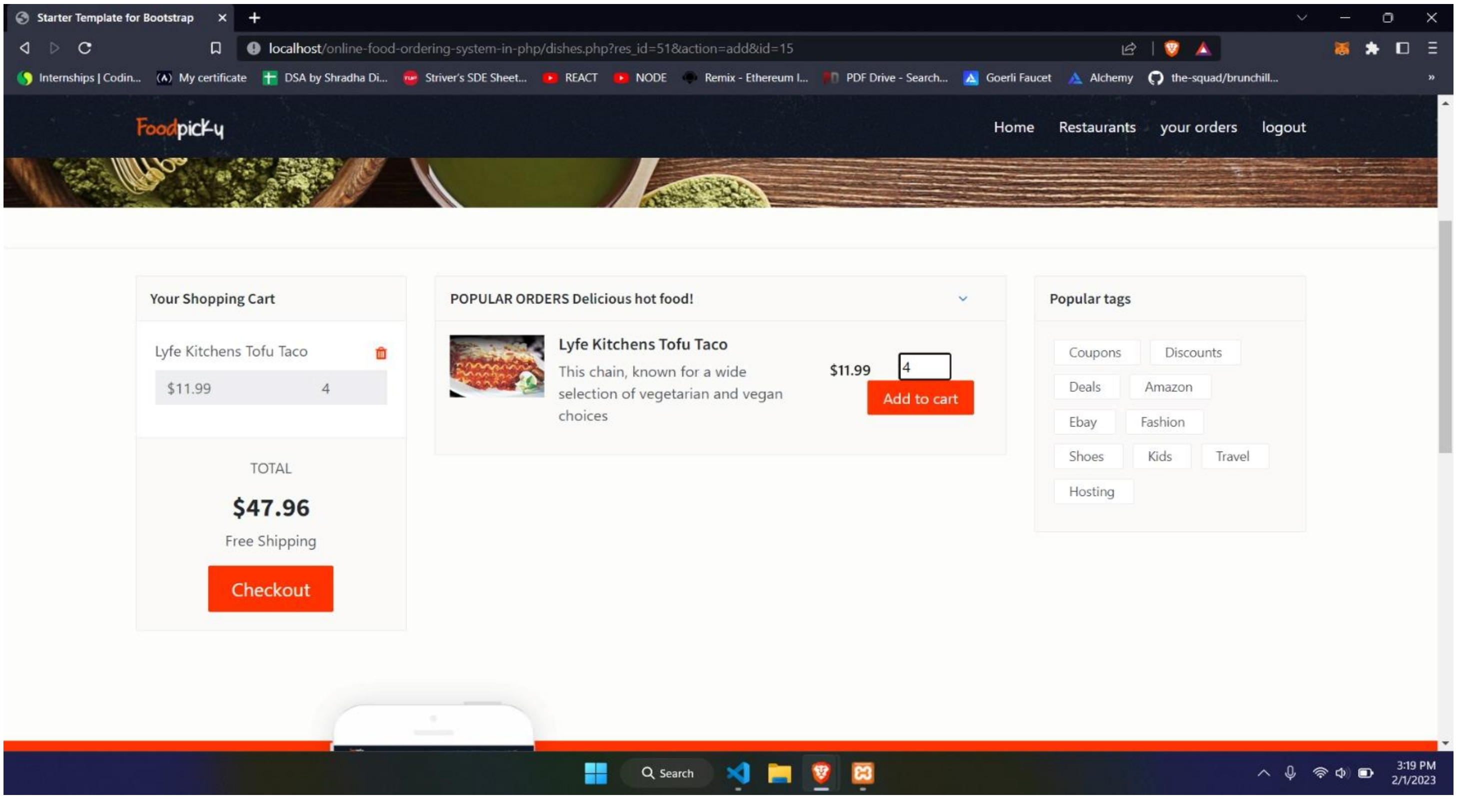
This chain, known for a wide selection of vegetarian and vegan choices

$11.99 1

Add to cart

Popular tags

Coupons Discounts Deals Amazon Ebay Fashion



localhost/online-food-ordering-system-in-php/dishes.php?res_id=51&action=add&id=15

Your Shopping Cart

Lyfe Kitchens Tofu Taco

$11.99 4

TOTAL

$47.96

Free Shipping

Checkout

POPULAR ORDERS Delicious hot food!

Lyfe Kitchens Tofu Taco

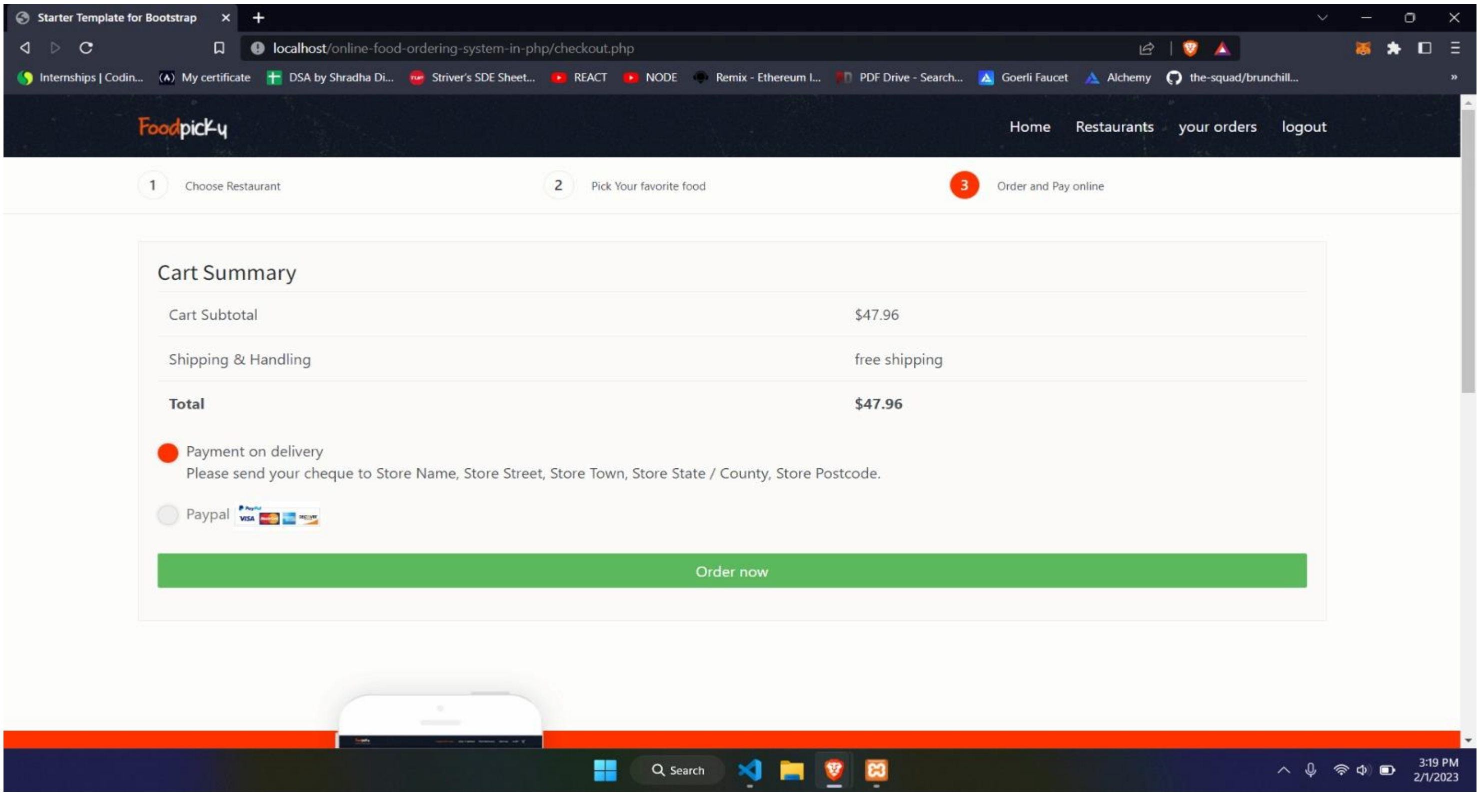
This chain, known for a wide selection of vegetarian and vegan choices

$11.99 4

Add to cart

Popular tags

Coupons Discounts Deals Amazon Ebay Fashion Shoes Kids Travel Hosting


Starter Template for Bootstrap
localhost/online-food-ordering-system-in-php/checkout.php
Foodpicky
Home
Restaurants
your orders
logout
1 Choose Restaurant
2 Pick Your favorite food
3 Order and Pay online

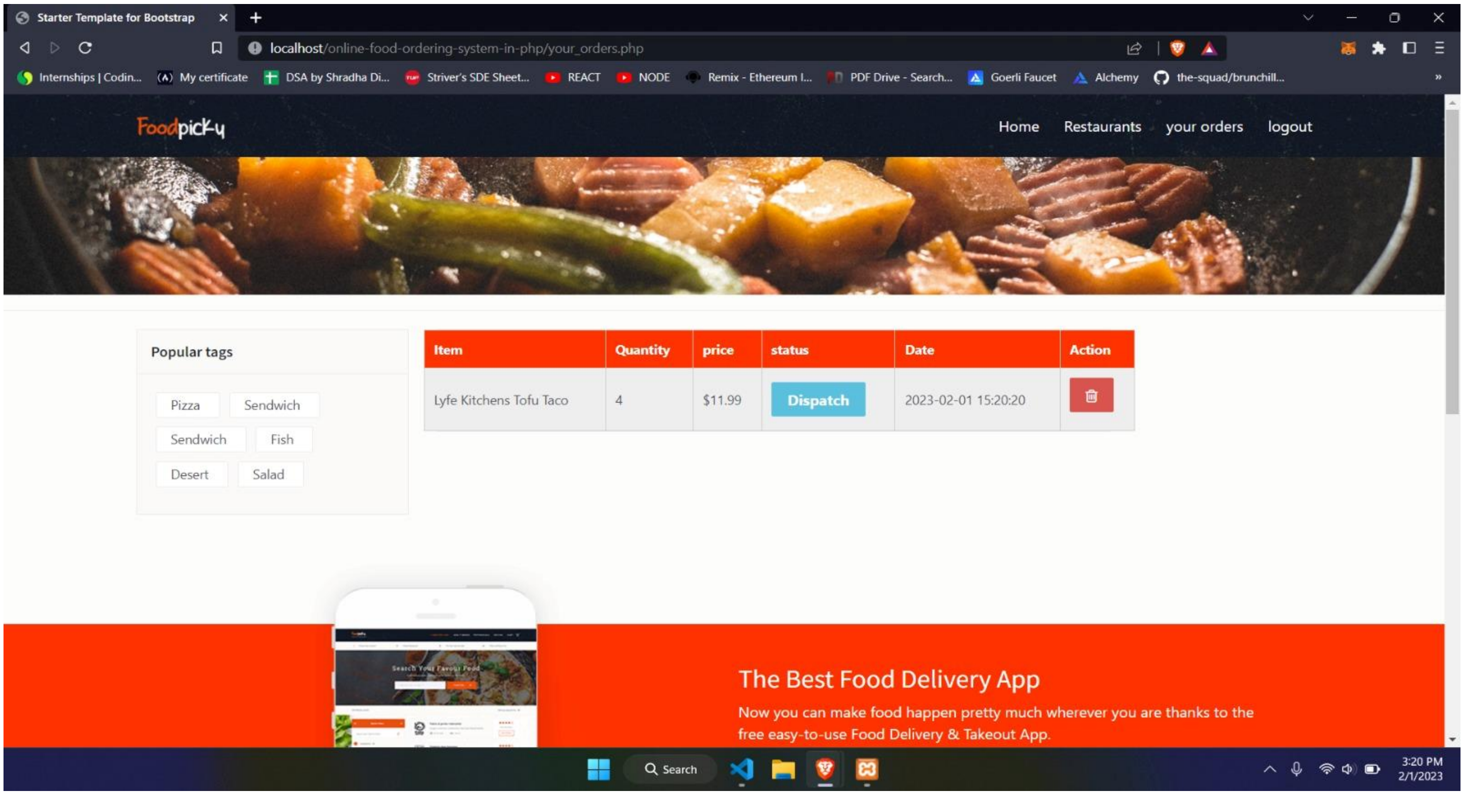
Cart Summary

Cart Subtotal	$47.96
Shipping & Handling	free shipping
Total	**$47.96**

● Payment on delivery
Please send your cheque to Store Name, Store Street, Store Town, Store State / County, Store Postcode.

○ Paypal

Order now
3:19 PM
2/1/2023


Starter Template for Bootstrap
localhost/online-food-ordering-system-in-php/your_orders.php
Foodpicky
Home
Restaurants
your orders
logout
Popular tags
Pizza
Sendwich
Sendwich
Fish
Desert
Salad

Item	Quantity	price	status	Date	Action
Lyfe Kitchens Tofu Taco	4	$11.99	Dispatch	2023-02-01 15:20:20	🗑

The Best Food Delivery App
Now you can make food happen pretty much wherever you are thanks to the free easy-to-use Food Delivery & Takeout App.

Conclusion

This project aimed at creating an online food ordering system from local restaurants. This project has been designed to provide users with a convenient platform for ordering food from local restaurants without leaving their homes. It covers the essential features of an online food ordering system and utilizes the following technologies:

PHP

PHP is a server-side scripting language that is used to build dynamic web applications. It is an open-source language and has a wide range of applications in web development. In this project, PHP has been used to interact with the database, process user requests, and display the results to the user.

SQL

SQL (Structured Query Language) is used to manage data stored in relational databases. It is used to insert, retrieve, update, and delete data in a database. In this project, SQL has been used to create tables and store data for the restaurant, menu, and user information.

Bootstrap 4

Bootstrap 4 is a popular front-end framework that provides pre-designed components for building responsive websites. It is widely used for building user interfaces for web applications and is known for its mobile-first approach. In this project, Bootstrap 4 has been used to design the front-end of the website and create an attractive user interface.

JQuery

JQuery is a JavaScript library that makes it easy to manipulate HTML documents and create dynamic user interfaces. It is a fast, small, and feature-rich library that makes it simple to add animations and interactivity to web pages. In this project, JQuery has been used to provide dynamic functionality to the website, such as fetching data from the database and displaying it to the user in real-time.

Technologies

This project showcases the power of PHP and SQL in building dynamic web applications, along with the front-end framework Bootstrap 4 and the JavaScript library JQuery. These technologies combined provide a powerful toolset for building web applications that can deliver an exceptional user experience.

In conclusion, this project serves as a great example of how to build an online food ordering system from scratch. It provides a solid foundation for anyone looking to build their own online food ordering system, and it highlights the importance of database management, user-friendly interface design, and dynamic functionality.

REFERENCES

(i) MySQL documentation - https://dev.mysql.com/doc/

(ii) SQL Quick Reference from W3Schools - https://www.w3schools.com/sql/sql_quickref.asp

(iii) FreeCodeCamp SQL - https://www.freecodecamp.org/news/sql-anddatabases-fullcourse/

(iv) Select Star SQL - https://selectstarsql.com