

# **Food Truck Management System**



# Introduction

In our busy lives we sometimes may not spend too much time making food at home. And sometimes like we will just chill while having outside food. Some people like a variety of food items. There can be times when we want to dine early in the morning and there are no eateries open late at night. But we can always get food from trucks. To make life easier, we considered ordering food from trucks online.

# Objective of the project

In our project there are three modules : Food truck, delivery boy , customer. The food truck can register into the website with their details like name , email, phone and password into the website. The food truck needs to provide a verification code to verify the account. The truck can login to the website. The food truck will update the timings like opening time and closing time. The food truck will add the food categories to provide the menu. And also can view the food categories which are added. The food truck will add the food items to provide menu. The food truck can view the added food items .The truck can enable or disable the food items. The customer can register to the website with their details like name, email, phone number, password and address. The customer can login to the website with credentials. The customer can view the available trucks. The customer can view the food truck based on location. The customer can view the food items in the website which are added by the food truck. The customer can search the food items and can add the food items to the cart. The customer can view the cart and can place the order with payment.

## Continue...

The customer can view the orders. The customer can also cancel their orders before cooking. The food truck can view the order placed by the customer. The food truck should mark the order status as “cooking” while preparing the food. The food truck should mark the order status as “cooked” when food is prepared. The Delivery boy get registered and can login to the website. The food truck will assign the food orders to the delivery boys. The delivery boy will receive the delivery request. The delivery boy will accept the delivery request and then reach the food trucks. The food truck will mark the delivery status as “picked up”. The delivery boy will deliver the order to the customer. The delivery boy will make the status as “delivered to customer”. The customer will confirm the received order by making status as “order received”. The customer can provide the review and rating for the received orders. The customer can view the order history. The food truck can view the order history.

# Team Members

## **Member 1: DIVYA TEJASWI EJJUROTHU 700745894**

Working with the database development to design tables and database ER diagrams. The project developer uses the MySQL workbench database server.

## **Member 2-3: VAMSI DOSAPATI 700740054**

Developers collaborate with a back-end developer to create code and logical components. Using the Mysql database server with the flask framework.

## **Member 4-5:**

Working with on frontend developers to produce HTML, CSS and Bootstrap

# Database Description:

## Objective:

In our Project there are food\_truck, truck\_timings, Categories, food\_items, customers, Customer\_orders, customer\_order\_items, review and delivery\_boy tables. The food\_truck will register on the website by giving details like food\_truck\_title, name, email, password, phone and verification\_status these are all stored in the food\_truck table. The food\_truck login to the website through their credentials like email and password. The food\_truck will add the truck\_timings by giving from\_time, to\_time, date, location and food\_truck\_id is the foreign key comes from the food\_truck table these are all stored in the truck\_timings table. The food\_truck will add the categories by giving category\_name and food\_truck\_id is the foreign key comes from the food\_truck table these are all stored in the categories table. The food\_truck will add the food\_items by giving food\_item\_name, price, quantity, units, picture, status, description and category\_id if the foreign key comes from the categories table; these are all stored in the food\_items table.

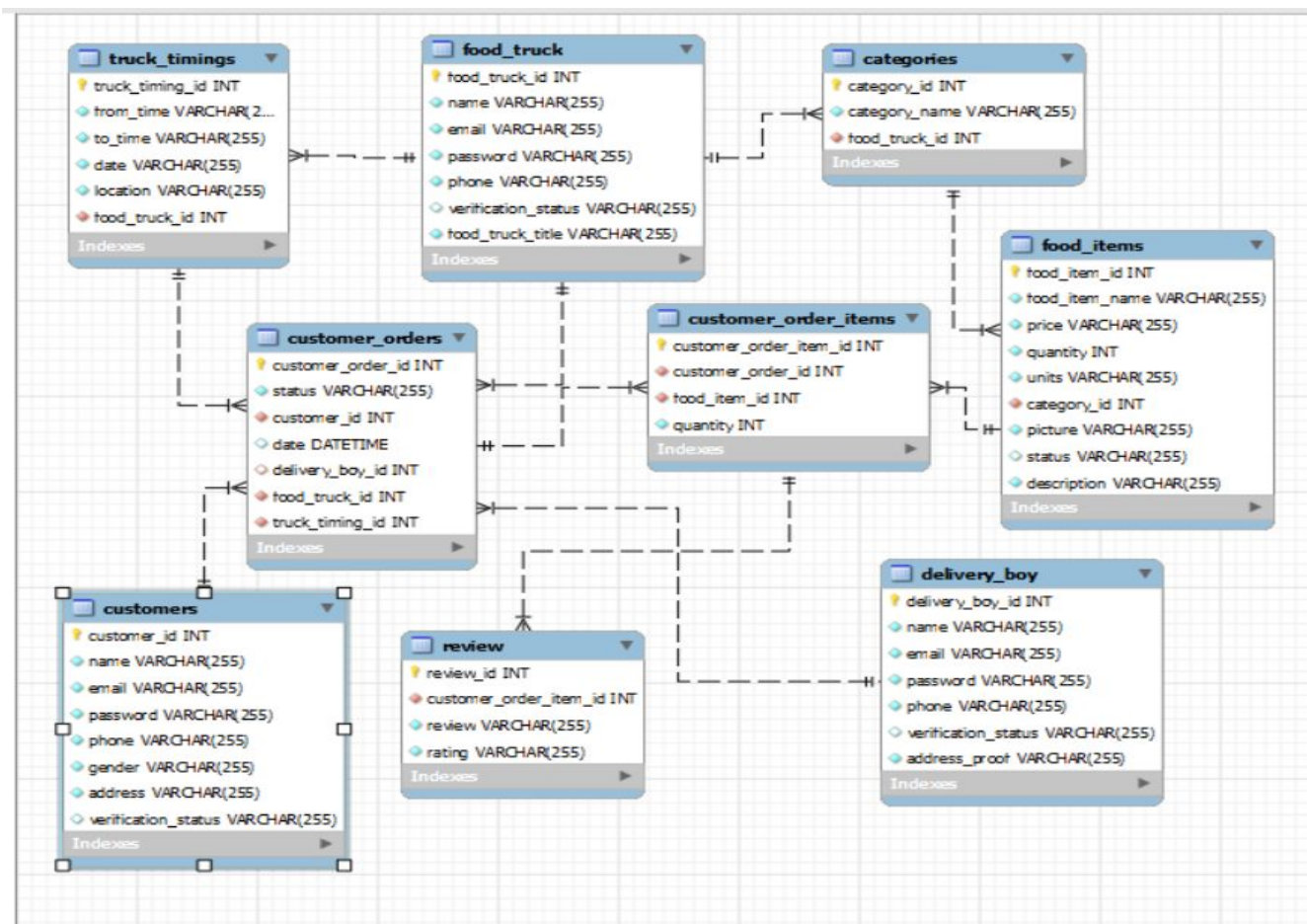
The delivery\_boy will register into the website by giving name, email, password, phone, verification\_status and address\_proof in the delivery\_boy table. The customer will register into the website by giving their details like name, email, password, phone, gender, address and verification\_status these are all stored in the customers table. The customer login to the website through their credentials like email and password. The customer will create the cart in that cart containing the status, date and customer\_id, delivery\_boy\_id, food\_truck\_id, truck\_timing\_id are the foreign keys; these are all stored in the customer\_orders table. The customer will select the order items by giving their quantity and customer\_order\_id, food\_item\_id are the foreign keys; these are all stored in the customer\_order\_items table.

# Complexity

In our project ,the transaction management technique is used to collect payment.The food items are ordered by the customer. After you have a paid, you can place your order.



# ER Diagram



# Schema Design

Food Truck:

	Field	Type	Null	Key	Default	Extra
►	food_truck_id	int	NO	PRI	NULL	auto_increment
	name	varchar(255)	NO		NULL	
	email	varchar(255)	NO	UNI	NULL	
	password	varchar(255)	NO		NULL	
	phone	varchar(255)	NO		NULL	
	verification_status	varchar(255)	YES		not verified	
	food_truck_title	varchar(255)	NO		NULL	

Truck Timings:

	Field	Type	Null	Key	Default	Extra
►	truck_timing_id	int	NO	PRI	NULL	auto_increment
	from_time	varchar(255)	NO		NULL	
	to_time	varchar(255)	NO		NULL	
	date	varchar(255)	NO		NULL	
	location	varchar(255)	NO		NULL	
	food_truck_id	int	NO	MUL	NULL	

## Food Categories:

	Field	Type	Null	Key	Default	Extra
►	category_id	int	NO	PRI	NULL	auto_increment
	category_name	varchar(255)	NO		NULL	
	food_truck_id	int	NO	MUL	NULL	

## Food Items:

	Field	Type	Null	Key	Default	Extra
►	food_item_id	int	NO	PRI	NULL	auto_increment
	food_item_name	varchar(255)	NO		NULL	
	price	varchar(255)	NO		NULL	
	quantity	int	NO		NULL	
	units	varchar(255)	NO		NULL	
	category_id	int	NO	MUL	NULL	
	picture	varchar(255)	NO		NULL	
	status	varchar(255)	YES		Available	
	description	varchar(255)	NO		NULL	

## Customer orders:

	Field	Type	Null	Key	Default	Extra
►	customer_order_id	int	NO	PRI	NULL	auto_increment
	status	varchar(255)	NO		NULL	
	customer_id	int	NO	MUL	NULL	
	date	datetime	YES		CURRENT_TIMESTAMP	DEFAULT_GENERATED
	delivery_boy_id	int	YES	MUL	NULL	
	food_truck_id	int	NO	MUL	NULL	
	truck_timing_id	int	NO	MUL	NULL	

## Customer order\_items:

	Field	Type	Null	Key	Default	Extra
►	customer_order_item_id	int	NO	PRI	NULL	auto_increment
	customer_order_id	int	NO	MUL	NULL	
	food_item_id	int	NO	MUL	NULL	
	quantity	int	NO		NULL	

## Customer:

	Field	Type	Null	Key	Default	Extra
▶	customer_id	int	NO	PRI	<b>NULL</b>	auto_increment
	name	varchar(255)	NO		<b>NULL</b>	
	email	varchar(255)	NO	UNI	<b>NULL</b>	
	password	varchar(255)	NO		<b>NULL</b>	
	phone	varchar(255)	NO		<b>NULL</b>	
	gender	varchar(255)	NO		<b>NULL</b>	
	address	varchar(255)	NO		<b>NULL</b>	
	verification_status	varchar(255)	YES		not verified	

## Delivery boy:

	Field	Type	Null	Key	Default	Extra
▶	delivery_boy_id	int	NO	PRI	<b>NULL</b>	auto_increment
	name	varchar(255)	NO		<b>NULL</b>	
	email	varchar(255)	NO	UNI	<b>NULL</b>	
	password	varchar(255)	NO		<b>NULL</b>	
	phone	varchar(255)	NO		<b>NULL</b>	
	verification_status	varchar(255)	YES		not verified	
	address_proof	varchar(255)	NO		<b>NULL</b>	

## Review:

	Field	Type	Null	Key	Default	Extra
▶	review_id	int	NO	PRI	<b>NULL</b>	auto_increment
	customer_order_item_id	int	NO	MUL	<b>NULL</b>	
	review	varchar(255)	NO		<b>NULL</b>	
	rating	varchar(255)	NO		<b>NULL</b>	

# Back-end Requirements

## Database Engine:

The basic functionality enabling mysql to work with and process data is provided by database engines. MySQL storage engines are components that conduct SQL operations on various table types. The MySQL workbench server is used in the database in this project.

## Crud Operations:

The four basic processes that a software program should be able to do are referred to as crud. Users in such apps must be able to create data, access the data in the UI, read the data, update or edit the data, and remove the data.

**Create:** The CREATE operation in crud performs exactly what is name suggests.To create is to INSERT in a SQL database.The insert() method is used to generate objects in a NoSQL database like MongoDB.

**Read:**The READ operations grants access to the UI's inputs.

**Update:** The UPDATE methods allows you to change existing data.UPDATE is used to update an entry in a SQL database.

**Delete:** To delete means to remove an entry from the UI and the database.

# Front-end Requirements

## User Interface:

The technical implementation of the software's user interface(UI) is referred to as front-end development. In our project, we used front-end technologies such as HTML,CSS,Bootstrap and javascript. HTML is the industry standard markup language for developing web pages.CSS is the programming language used to style a web pages.CSS is used to styles colors, fonts,sizes, and background images.Bootstrap is used to design navbars and menus for web pages.



# Deployment Requirements

**Cloud Deployment:** AWS EC2 is a cloud computing service that allows Amazon web services clients to launch, execute, and terminate applications. It serves as a framework for developing, delivering, and scaling web applications. AWS is designed to be the most flexible and secure cloud computing platform available today. This is known as cloud computing. The AWS cloud comprises a broad range of global cloud-based goods such as computing, storage, databases, analytics, networking, mobile, developer tools, management tools, IoT, security and business applications that are offered on-demand and with pay-as-you-go pricing.

# Security Measures

A web application is a computer program that runs on a server. Users gain access to these resources via a network or the internet. The administrator is logged in to the website using default credentials such as a valid email address and password. The user is logged in with the default credentials, such as email and password.

**Accessibility:** Amazon web services (AWS) is a cloud computing service that allows users to rent virtual computers on which to run their own computer programs. Amazon Elastic compute cloud (Amazon EC2) is a web service that provides safe, resizable processing capability in the cloud.

To construct an EC2, the public DNS web link will be opened in the website.

# Timeline

**OCT WEEK-1:** Our team has five members.

1-tech leader, 2-Database development 3-4 Back-end Development  
5-Front-end development.

**OCT WEEK-2:** No classes

**OCT WEEK-3:** Discuss the project plan.

**Nov WEEK-1:** Proposal Presentation (10-15 min)

**Nov WEEK 2-3:** Project Implementation

**Nov WEEK-4:**Final presentation(15-20 min)

# Example Ideas

Furthermore, After creating the web link DNS weblink ,the EC2 web link opens and runs the website.