

Database Management System

Mini Project Report

On

FOOD ORDER MANAGEMENT SYSTEM

Submitted by

LIKITH G

KARTHIK BALLAL

4SN19CS052

4SN19CS041

COMPUTER SCIENCE & ENGINEERING
(Visvesvaraya Technological University)

Under the Guidance of

Mr. SUDARSHAN K

Assistant/Associate Professor



Department of Computer Science & Engineering

SRINIVAS INSTITUTE OF TECHNOLOGY

(NAAC ACCREDITED)

MANGALURU-574143, KARNATAKA

2021 – 2022

**SRINIVAS INSTITUTE OF TECHNOLOGY
(NAAC ACCREDITED)**

MANGALURU -574143 – KARNATAKA

2021 – 2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRINIVAS GROUP

CERTIFICATE

*This is to certify that the project entitled “**FOOD ORDER MANAGEMENT SYSTEM**”,
is an authentic record of the work carried out by*

**LIKITH G
4SN19CS052**

**KARTHIK BALLAL
4SN19CS041**

*as prescribed by Visvesvaraya Technological University, Belagavi, for V Semester B.E. in
Computer Science & Engineering during the year 2019-2020.*

SAMAGRA GNANA

Mr. Sudarshan K

Project Guide

Prof. Ravishankar

Head of the Department

Dr. Shrinivasa Mayya D

Principal

Name of the Examiners

Signature with Date

1.

2.

ABSTRACT

The “Food Order Management System” has been developed to override the problems prevailing in the practicing manual system. This is used to reduce the hardships faced by the existing system. Moreover, this system is designed for the particular need of the company to carry out the operations in a smooth and effective manner.

The application reduced as much as possible to avoid errors while entering the data. No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user friendly. Food Order Management System, as described can lead to error free, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources.

Every organization whether big or small, has challenges to overcome and managing the information of Category, Food Item, Order, Payment, Confirm Order. The project is built only on administrative end and thus only the administrator is granted the access. The purpose of this project is to build an application program to reduce man work for managing the Item Category, Food, Customer, Bill.

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any work would be incomplete without thanking the persons who made it perfect with their constant guidance and encouragement. We take this opportunity to express my sincere thanks and indebtedness to our Project guide and mentor, Mr. Sudarshan K, Assistant/Associate Professor, Department of CS&E, for his/her support and guidance. His/her vision and suggestions throughout the project period has been fundamental in the completion of the project. We would like to extend my deep sense of acknowledgement to our Project Co-ordinator Mr. Sudarshan K Assistant/Associate Professor, Department of CS&E, for his/her encouragement and timely advice provided to us during the project work. We extend my warm gratitude to Prof. Ravishankar, Head of the Department, Department of CS&E, for his constant support and advice that helped us to complete this project successfully. We are extremely grateful to our beloved Principal, Dr. Shrinivasa Mayya D for his encouragement to come up with new ideas and advice to express them in a systematic manner. We also like to thank all Teaching & non-teaching staff of Srinivas Institute of Technology, Mangalore for their kind co-operation during the course of the work. Finally, we are extremely thankful to our family and friends who helped us in our work and made the project a successful one.

LIKITH G

KARTHIK BALLAL

TABLE OF CONTENTS

CHAPTER No			TITLE	PAGE No
1			INTRODUCTION	
	1.1		Problem Statement	
	1.2		Scope Of The Project	
2			REQUIREMENT SPECIFICATION	
	2.1		Functional Requirements	
	2.2		Non-Functional Requirements	
	2.3		Hardware Requirements	
	2.4		Software Requirements	
	2.5		Software Tools Used	
		2.5.1	Front End Tool	
		2.5.2	Backend Database Used	
3			SYSTEM DESIGN	
	3.1		Flowchart	
	3.2		Database Table Design	
	3.3		Schema Diagram	
	3.4		E-R Diagram	
	3.5		Use-Case Diagram	
	3.6		Normalization	
4			IMPLEMENTATION	
	4.1		Creating Database Using MySQL	
	4.2		Stored Procedure	
	4.3		Triggers	

5			SCREEN SHOTS	
6			CONCLUSION AND SCOPE FOR FUTURE WORK	
			BIBLIOGRAPHY	

LIST OF FIGURES

FIGURE No	TITLE	PAGE No
3.1	Flowchart	
3.2	Database Table Design	
3.3	Schema Diagram	
3.4	E-R Diagram	
6.1	Welcome Page	
6.2	Login Page	

LIST OF TABLES

Table No	TITLE	PAGE No
3.1	CUSTOMER	
3.2	FOOD	
3.3	FOOD CATEGORY	
3.4	ORDERS	
3.5	BILL	

CHAPTER 1

INTRODUCTION

The “Food Order Management System” has been developed to override the problems prevailing in the practicing manual system. This software supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Also, this system offers many flexible and convenient features, allowing customers and users to maximize time and efficiency.

1.1 Problem Statement

Many restaurants are storing all of their data in manual way. They have huge number of customers daily. So, because large number of customers, they need the help of some features so they can maintain and store the records accurately.

For this they need full-fledged software to maintain the day-to-day transactions, orders and also regular update on records, cash transactions, orders and also regular update on records, etc. In the existing system all the details are done manually, it is taking lots of time and also there are chances for mistakes.

1.2 Scope of the Project

The scope of Food Order Management System includes:

- Create distinct product users based on their roles and permissions.
- Authenticate user at their login.
- Provide the list of food items where the users can order.
- Facility for classified food items.
- A status page for all users to view food ordered by them.
- Facility to cancel the food item ordered by user earlier.
- A status page for all users to view food ordered by them, their individual payment details.
- An interface to view and edit the own profile and easy add to cart with one click item.
- Providing interface to add or delete food items of the restaurant.

CHAPTER 2

REQUIREMENT SPECIFICATION

2.1 Functional Requirements

As each of the three system components essentially provides a layer of isolation as between the end user and the database. The motivation behind the isolation is twofold firstly, allowing the end user to interact with the system through rich interface provide a much more enjoyable user experience particularly for the non-technical users which will account for the majority of the systems users. In addition, this isolation layer also protects the integrity of the database by preventing users from taking any action outside those which the system as designed to handle because of this design pattern, it is essential to enumerate exactly which functions a user will be presented and these functions are outlined.

2.2 Non- Functional Requirements

A non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than behaviours.

Some of the non-functional requirements include:

- Backup- provision for data backup.
- Maintainability- easy to maintain
- Performance/ response time- fast response.
- Usability by target user community- easy to use.
- Expandability- needs to be future proof or upgradable.
- Safety- should be safe to use.

2.3 Hardware Requirements

The minimum/recommended hardware configuration required for developing the proposed software is given below:

- 8GB RAM
- 1.2GHz Processor
- Windows 7/8/10

2.4 Software Requirements

- **Front end**
 - Apache NetBeans IDE 12.0
 - MySQL-Connector-Java-8.0.27
- **Back end**
 - MySQL Workbench 8.0 CE

2.5 Software Tools Used

Food Order Management System is designed using Apache NetBeans IDE 12.0 as a front-end user interface design tool and MySQL Workbench 8.0 CE at back-end for creating tables and storing related information.

2.5.1 Front End Tool

Apache NetBeans is top level Apache Project dedicated to providing rock solid software development products (the Apache NetBeans IDE and the Apache NetBeans Platform) that address the needs of developers, users and the businesses who rely on NetBeans as a basis for their products; particularly, to enable them to develop these products quickly, efficiently and easily by leveraging the strengths of the Java platform and other relevant industry standards.

2.5.2 Back End Database Used

MySQL pronounced as either "My S-Q-L" or "My Sequel is an open-source relational database management system is based on the structure query language (SQL) which is used for adding, removing, and modifying information in the database, such as ADD, DROP, INSERT, and UPDATE can be along with M MySQL can be used for variety of applications, but is most command in d servers. A website that uses MySQL may include Web pages that access information from the database. These pages are often referred to as "dynamic," meaning to content of each page is generated from a database as the page leads Websites that use dynamic Web pages are often referred to as database-driven websites and accessed by users indirectly through an external.

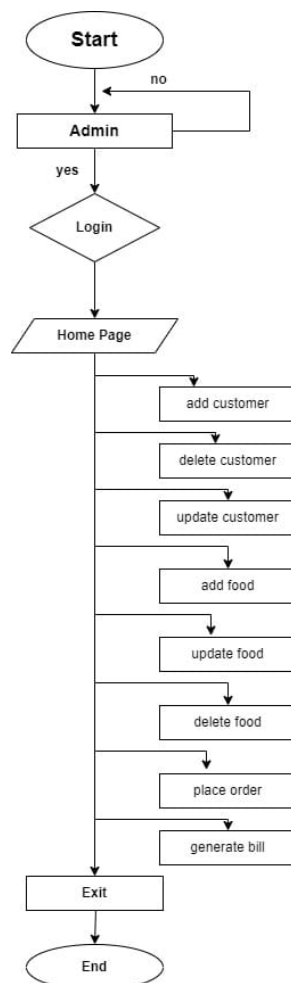
CHAPTER 3

SYSTEM DESIGN

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. It is meant to satisfy specific needs and requirements of a business or organization through the engineering of a coherent and well-running system.

3.1 Flow Chart

A flowchart is a type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. With proper design and construction, it communicates the steps in a process very effectively and efficiently. Figure 3.1 depicts the flow chart of LMS which shows various processing steps based on different events, actions and conditions.



3.2 Database Table Design

The basic database unit is the table. A table is a unit consisting of rows of related information. Each row consists of fields of information where data is stored. Field attributes include information and rules that govern the data stored in the field. The field attributes and rules may limit the type of data stored in the field.

A field may be defined as a key or may be limited by rules requiring specific masks, such as a field may be limited to dates, formatted numbers like telephone numbers, or be limited to a specific number of characters. The database schema contains these rules. Database tables used in LMS are shown below.

1.Customer:

Name	Datatype	Constraint
cus_id	Int	Primary key
cus_name	Varchar(20)	Not null
phone_num	Bigint	Not null
cus_address	Varchar(20)	Not null
cus_age	int	Not null

2. Food Category

Name	Datatype	Constraint
food_cate_no	Int	Primary key
food_cate_name	Varchar(20)	Not null

3.Food

Name	Datatype	Constraint
food_Id	Int	Primary Key
food_name	Varchar(20)	Foreign key
food_cost	Decimal(5,2)	Not null
food_cate_no	Int	Not null

4.Orders

Name	Datatype	Constraint
ord_id	Int	Primary Key
cus_id	Int	Not null
food_id	Int	Not null
quantity	Int	Not null
order_amount	Int	Not null

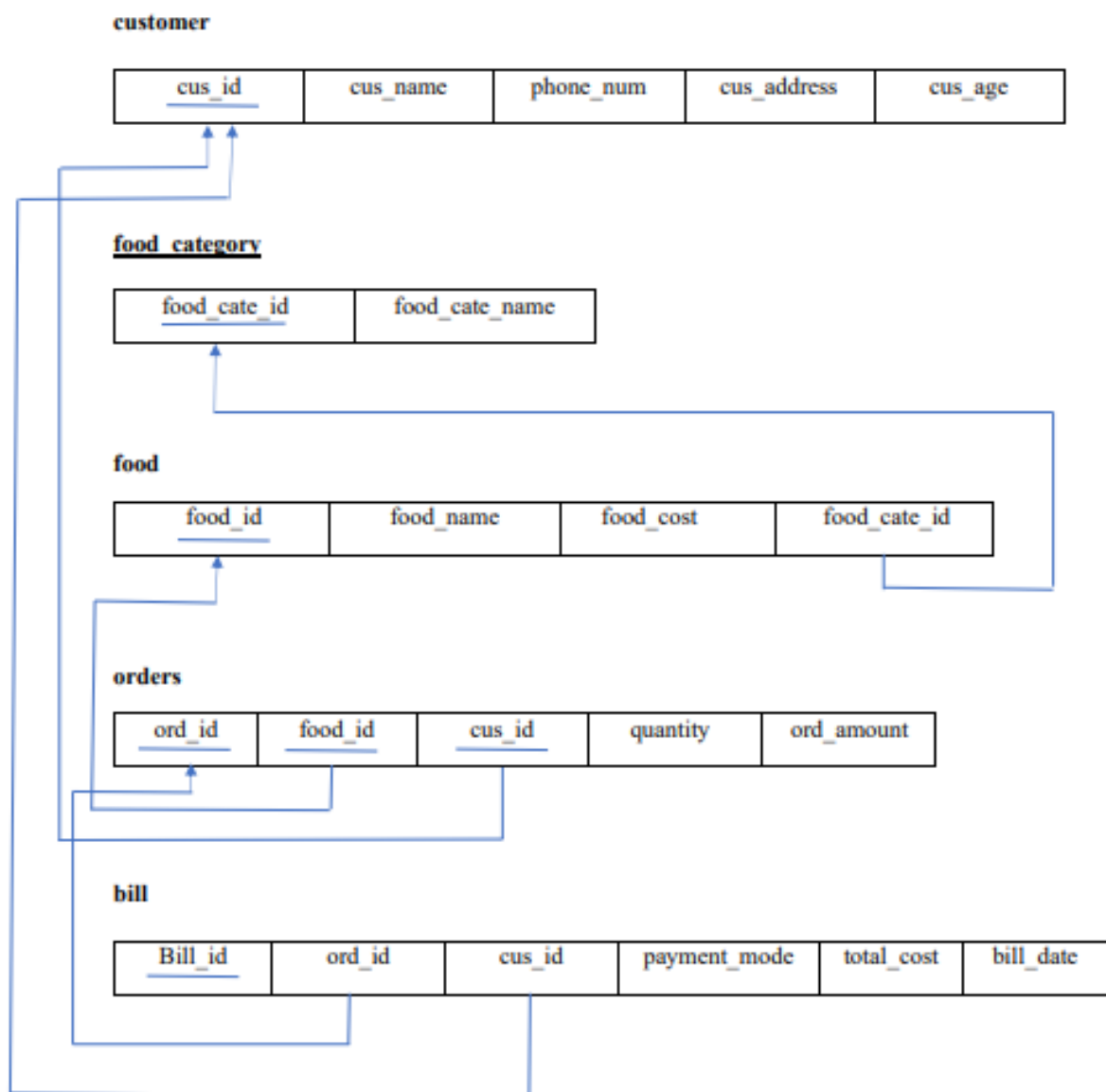
5.Bill

Name	Datatype	Constraint
bill_id	Int	Primary Key
ord_id	Int	Foreign key
cus_Id	Int	Foreign key
payment_mode	Varchar(10)	Not null
total_cost	Int	Not null
bill_date	Date	Not null

3.3 Schema Diagram

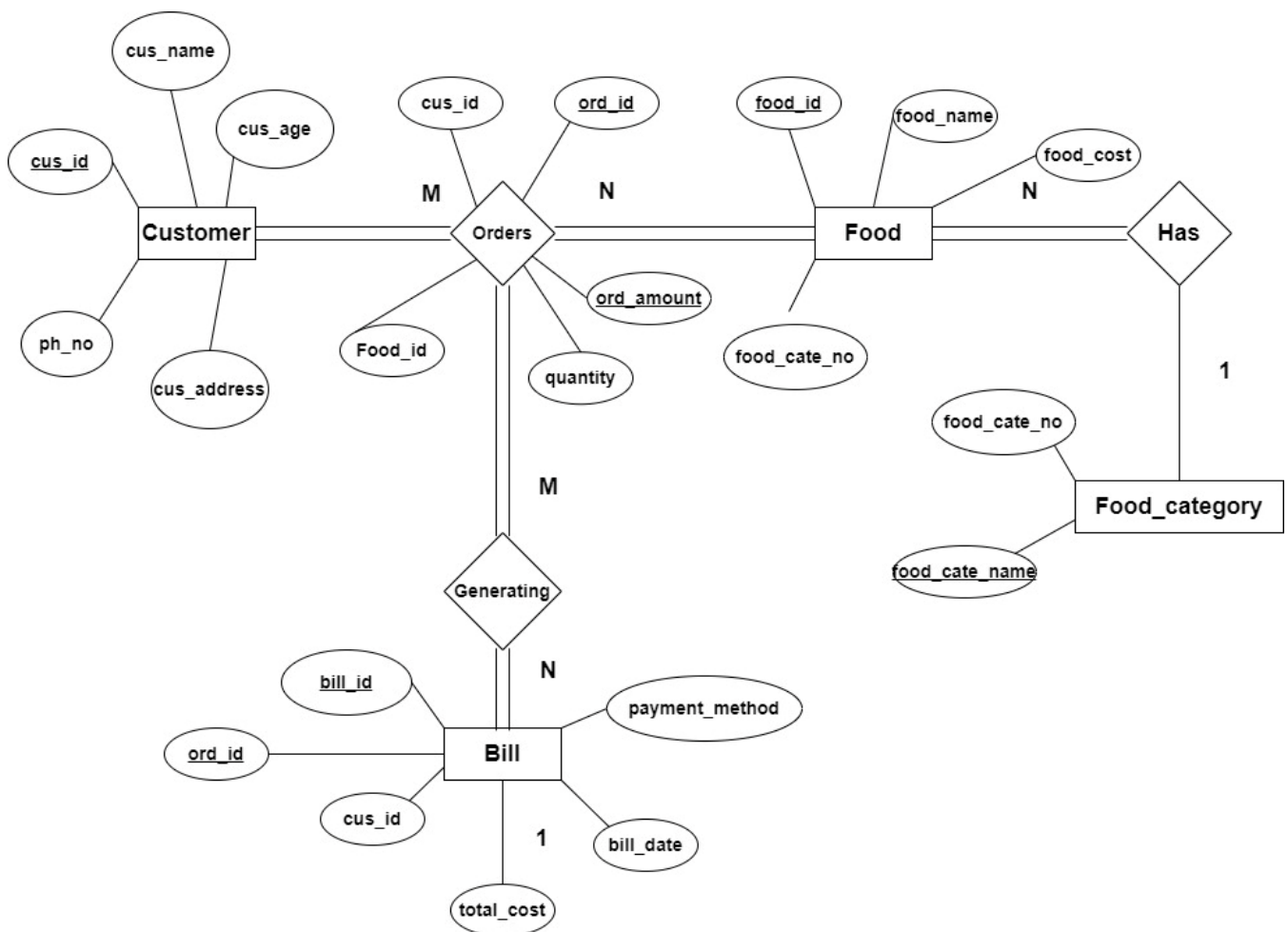
A schema is the structure behind data organization. It is a visual representation of how different table relationships enable the schema's underlying mission business rules for which the database is created. In a schema diagram, all database tables are designated with unique columns and special features, e.g., primary/foreign keys or not null, etc. Formats and symbols for expression are universally understood, eliminating the possibility of confusion. The table relationships also are expressed via a parent table's primary key lines when joined with the child table's corresponding foreign keys.

Schema diagrams have an important function because they force database developers to transpose ideas to paper. This provides an overview of the entire database, while facilitating future database administrator work. Figure 3.2 shows the schema diagram of FOMS.



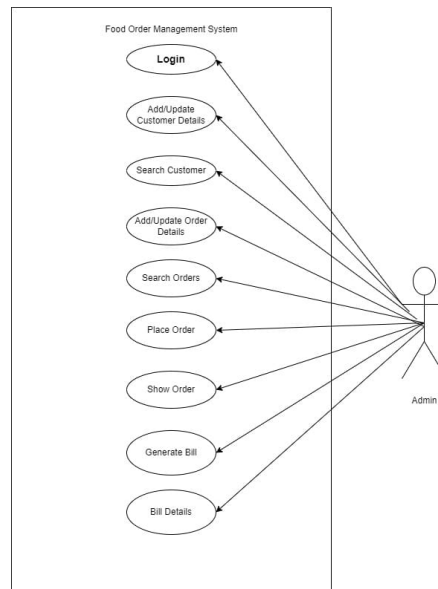
3.4 ER Diagram

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases. ER diagrams are used to sketch out the design of a database. Figure 3.3 depicts the ER diagram of FOMS. It shows various entities, their attributes, relationships with other entities, cardinality ratios between entities and participation constraints used in designing FOMS database.



3.5 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. Users interacting with application are shown outside with stickman symbol.



3.6 Normalization

Database normalization is the process of structuring a relational database [clarification needed] in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity. It was first proposed by Edgar F. Codd as part of his relational model.

Normalization entails organizing the columns (attributes) and tables (relations) of a database to ensure that their dependencies are properly enforced by database integrity constraints. It is accomplished by applying some formal rules either by a process of synthesis (creating a new database design) or decomposition (improving an existing database design)

First normal form (1NF):

First normal form (1NF) is a property of a relation in a relational database. A relation is in first normal form if and only if the domain of each attribute contains only atomic (indivisible) values, and the value of each attribute contains only a single value from that domain.

All relations in Food Order management system have only atomic values, so they are in 1NF.

Second normal form (2NF):

A relation is said to be in second normal form if it is already in first normal form and it has no partial dependency

Customer:

<u>cus_id</u>	cus_name	phone_num	cus_address	cus_age
---------------	----------	-----------	-------------	---------

Third normal form (3NF):

It is a normal form that is used in normalizing a database design to reduce the duplication of data and ensure referential integrity by ensuring that:

- ☐ The entity is in second normal form.
- ☐ No non-prime (non-key) attribute is transitively dependent on any key i.e. no non-prime attribute depends on other non-prime attributes. All the non-prime attributes must depend only on the candidate keys.

All entities of this table are in 3NF form

CHAPTER 4

IMPLEMENTATION

System implementation is the important stage of project when the theoretical design is tuned into practical system

4.1 Creating Database Using MySQL

Queries used for creating database and different tables used in FOMS are given below.

To create new database

```
mysql> create database FOMS;
```

To use newly created database

```
mysql> use FOMS;  
Database changed
```

To create table CUSTOMER

```
mysql> create table customer(cus_id int,cus_name varchar(20),phone_num  
int(10),cus_address varchar(20),cus_age int,constraint cus_id_prim primary key(cus_id));
```

To create table FOOD CATEGORY

```
mysql>create table food_category(food_cate_no int,food_cate_name varchar(20),constraint  
food_cate_prim primary key(food_cate_no));
```

To create table FOOD

```
mysql> create table food(food_id int,food_name varchar(20),food_cost  
decimal(5,2),food_cate_no int,constraint food_id_prim primary key(food_id),constraint  
food_cate_no_fore1 foreign key(food_cate_no) references food_category(food_cate_no)on  
delete cascade);
```

To create table ORDERS

```
mysql> create table orders(ord_id int,cus_id int,food_id int,quantity int,order_amount  
int,constraint ord_id_prim primary key(ord_id,cus_id,food_id),constraint cus_id_fore foreign  
key(cus_id) references customer(cus_id)on delete cascade,constraint food_id_fore foreign  
key(food_id) references food(food_id)on delete cascade);
```

To create table BILL

```
mysql> create table bill(bill_id int,ord_id int,cus_id int,payment_mode varchar(20),total_cost  
decimal(7,2),bill_date date,constraint bill_id_prim primary key(bill_id),constraint
```

```
ord_id_fore1 foreign key(ord_id) references orders(ord_id)on delete cascade,constraint  
cus_id_fore3 foreign key(cus_id) references customer(cus_id)on delete cascade);
```

4.2 Stored Procedure

A stored procedure is a prepared SQL code that can be reused over and over again. So, if an SQL query needs to be written over and over again, save it as a stored procedure, and then just call it to execute it. It is also possible to pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.

4.3 Triggers

A trigger is a special type of stored procedure that automatically executes when an event occurs in the database server. DML triggers execute when a user tries to modify data through a data manipulation language (DML) event. DML events are INSERT, UPDATE, or DELETE statements on a table or view. These triggers fire when any valid event is fired, regardless of whether or not any table rows are affected.

Trigger used in FOMS is as follows Specify trigger used here along with quantity Trigger to automatically reject number of food quantity when return or issue is made

CHAPTER 5

SCREEN SHOTS

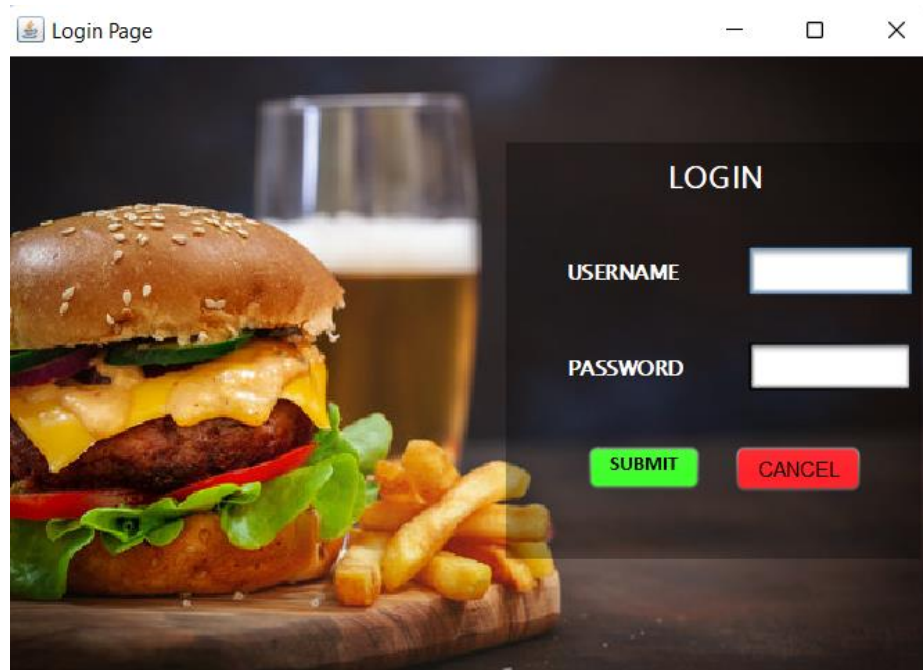


Fig 6.1: Front Screen

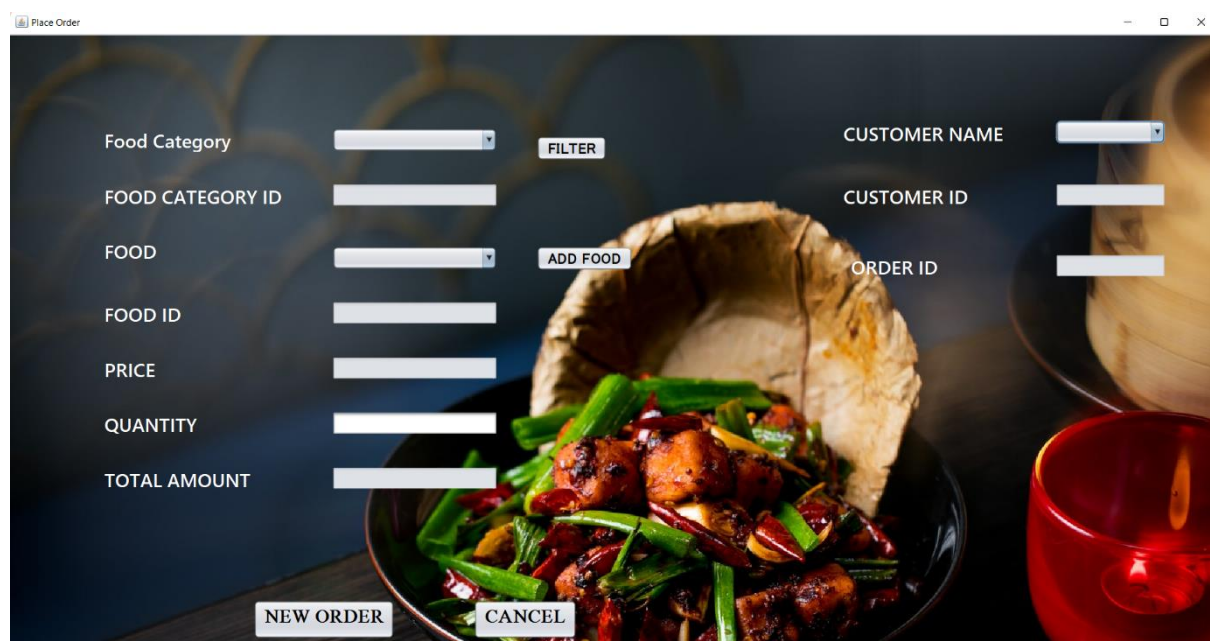


Fig 6.2: Crazy Cat Screen

CONCLUSION AND SCOPE FOR FUTURE WORK

The conclusion of the proposed system is user's need and is user centred. The system is developed in considering all issues related to all user which are included in this system. Wide range of people can use this if they know how to operate android smart phone and Various issues related to Mess Service will be solved by providing them a full-fledged system. Thus, implementation of Food Ordering system is done to help and solve one of the important problems of people. Based on the result of this research, it can be concluded: It helps customer in making order easily; It gives information needed in making order to Customer The Food application made for restaurant and mess can help restaurant and mess in receiving orders and modifying its data and it is also made for admin so that it helps admin in controlling all the Food system.

BIBLIOGRAPHY

- Learn Java GUI Applications - 11th Edition: A JFC Swing Tutorial.
- Learning Java, 4th Edition by Patrick Niemeyer, Daniel Leuck.
- The JFC Swing Tutorial: A Guide to Constructing GUIs (K. Walrath).
- <https://www.javatpoint.com/java-swing>
- <https://docs.oracle.com/javase/tutorial/uiswing/>
- <https://stackoverflow.com/>