VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JNANA SANGAMA”, BELAGAVI - 590 018

A MINI PROJECT REPORT

on

“GROCERY DELIVERY SYSTEM”

Submitted by

Akhilesh M	4SF18IS008
Amaan Mohammad	4SF18IS010

In partial fulfillment of the requirements for the V semester

DBMS LABORATORY WITH MINI PROJECT

of

BACHELOR OF ENGINEERING

in

INFORMATION SCIENCE & ENGINEERING

Under the Guidance of

Mr. Rithesh Pakkala P.

Assistant Professor

Department of ISE

at

SAHYADRI

College of Engineering & Management

Adyar, Mangaluru - 575 007

2020 - 21

SAHYADRI

College of Engineering & Management

Adyar, Mangaluru - 575 007

Department of Information Science & Engineering

CERTIFICATE

This is to certify that the **Mini Project** entitled **“Grocery Delivery System”** has been carried out by **Akhilesh M (4SF18IS008)** and , **Amaan Mohammad (4SF18IS010)** the bonafide students of Sahyadri College of Engineering & Management in partial fulfillment of the requirements for the V semester **DBMS Laboratory with Mini Project (18CSL58)** of **Bachelor of Engineering in Information Science & Engineering** of Visvesvaraya Technological University, Belagavi during the year 2020 - 21. It is certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The mini project report has been approved as it satisfies the academic requirements in respect of mini project work.

Mr. Rithesh Pakkala P.	**Dr. Shamanth Rai**
Assistant Professor	HOD & Associate Professor
Dept. of ISE, SCEM	Dept. of ISE, SCEM

External Practical Examination:

Examiner’s Name Signature with Date

1. . . . .

2. . . . .

SAHYADRI

College of Engineering & Management

Adyar, Mangaluru - 575 007

Department of Information Science & Engineering

DECLARATION

We hereby declare that the entire work embodied in this Mini Project Report titled **"Grocery Delivery System"** has been carried out by us at Sahyadri College of Engineering and Management, Mangaluru under the supervision of **Mr. Rithesh Pakkala P.** as the part of the V semester **DBMS Laboratory with Mini Project (18CSL58)** of **Bachelor of Engineering** in **Information Science & Engineering**. This report has not been submitted to this or any other University.

Akhilesh M (4SF18IS008)
Amaan Mohammad (4SF18IS010)
SCEM, Mangaluru

Abstract

Helpy Hands is a startup where we specialize in groceries and essentials delivery to the customers doorstep. In this project we have highlighted the behind the scenes process. To improve the experience of both the customer, seller and the delivery agent our comprehensive database specializes in quick real-time updation of Customer details , Shop stocks and delivery agent availability and tracking. All these elements work seamlessly from when the order is placed from the customer side right through the Shop's inventory and the order tracking of the customer.

Acknowledgement

It is with great satisfaction and euphoria that we are submitting the Mini Project Report on **“Grocery Delivery System”**. We have completed it as a part of the V semester **DBMS Laboratory with Mini Project (18CSL58)** of **Bachelor of Engineering** in **Information Science & Engineering** of Visvesvaraya Technological University, Belagavi.

We are profoundly indebted to our guide, **Mr. Rithesh Pakkala P.**, Assistant Professor, Department of Information Science & Engineering for innumerable acts of timely advice, encouragement and We sincerely express our gratitude.

We express our sincere gratitude to **Dr. Shamanth Rai**, Head & Associate Professor, Department of Information Science & Engineering for his invaluable support and guidance.

We sincerely thank **Dr. Rajesha S**, Principal, Sahyadri College of Engineering & Management and **Dr. D. L. Prabhakara**, Director, Sahyadri Educational Institutions,who have always been a great source of inspiration.

Finally, yet importantly, We express our heartfelt thanks to our family & friends for their wishes and encouragement throughout the work.

Akhilesh M
4SF18IS008
V Sem, B.E., ISE
SCEM, Mangaluru

Amaan Mohammad
4SF18IS010
V Sem, B.E., ISE
SCEM, Mangaluru

Table of Contents

List of Figures

Chapter 1

Introduction

When the entire world was reeling from the effects of Covid-19 and lockdown, it was tough time for all of us to be locked inside our homes. There arised a problem of people having difficulties in shopping for food , essentials and medicines. Therefore we founded a start-up where it enabled anyone to order essentials or medicines from the shop of their choice and have it delivered to their home. This discouraged people from stepping outside which helped curb the spread of virus considerably.

1.1 Database Management System

With the widespread use of computer technology and network technology, the development of database technology has become an important part of advanced information technology. The core of the supermarket management system is how to use and operate database, so the database design is critical. This system uses the Oracle10g database which is a relational database management system of Oracle. It is a product that always has been a leading position in the field of database. The Oracle database system is the world popular relational database management system which easy to use, strong function and suitable for all kinds of large, medium and small, microcomputer environment. It can realize data sharing and the facilities don't need to have the powerful data storage and processing capabilities so that to reduce the hardware cost of supermarket. Many enterprises have their own database, and store a large number of key data in it, which shows the importance of the database.

Every table in the database is broken up into smaller entities called fields. The fields in the Customers table consist of CstID, CstName ,CstPhoneNo , CstEmail A field is a column in a table that is designed to maintain specific information about every record in the table. A record, also called a row, is each individual entry that exists in a table. A record is a horizontal entity in a table. A column is a vertical entity in a table that contains all information associated with a specific field in a table. In addition to tables, a database can also contain other objects including views, stored procedures, indexes and constraints, along with a transaction log.

1.2 Structured Query Language

SQL is a domain specific language used in programming and designed for managing data held in a database management system. SQL consists of a data definition language, data manipulation language and data control language. The scope of SQL includes data insert, query update and delete, schema creation and modification and data access control. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications which may run either on the same computer or on another computer across a network. The main mode of retrieving data from a SQL Server database is querying for it. The query declaratively specifies what is to be retrieved. It is processed by the query processor, which figures out the sequence of steps that will be necessary to retrieve the requested data. The sequence of actions necessary to execute a query is called a query plan. There might be multiple ways to process the same query.Stored procedures can accept values sent by the client as input parameters, and send back results as output parameters. They can call defined functions, and other stored procedures, including the same stored procedure.

1.3 Stored Procedure

A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again. So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it.We can also pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.A stored procedure is nothing more than prepared SQL code that you save so you can reuse the code over and over again. So if you think about a query that you write over and over again, instead of having to write that query each time you would save it as a stored procedure and then just call the stored procedure to execute the SQL code that you saved as part of the stored procedure.In addition to running the same SQL code over and over again you also have the ability to pass parameters to the stored procedure, so depending on what the need is the stored procedure can act accordingly based on the parameter values that were passed.

1.4 Normalisation

Database Normalization is a technique of organizing the data in the database. Normalization is a systematic approach of decomposing tables to eliminate data redundancy(repetition) and undesirable characteristics like Insertion, Update and Deletion Anamolies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.
Normalization is used for mainly two purposes:
1.Eliminating redundant(useless) data.
2.Ensuring data dependencies make sense i.e data is logically stored.

1.5 Application

This project enables customers to buy anything and everything from the comfort of their homes. Groceries , essentials and pretty much everything can be home delivered from nearby shops. Helps save fuel and money whilst also helping customers save time. This is especially usefull during the time of pandemics. It encourages people to stay home and limits the spread of virus.

Chapter 2

Hardware and Software Details

2.1 Hardware Details

- Processor : Any Processor more than 500 MHz
- RAM : 2GB
- Hard Disk : 5GB
- Input Device : Standard Keyboard and Mouse
- Output Device : Monitor

2.2 Software Details

- Database : MySql
- Programming Language : Python
- IDE : Visual Studio Code
- Operating System : Windows 10

Chapter 3

System Design

3.1 Entity Relation Diagram(ERD)

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. 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.

ERD Entity Symbols:

Entities are objects or concepts that represent important data. Entities are typically nouns such as product, customer, location, or promotion.An entity is represented as rectangle in an ER diagram. There are three types of entities commonly used in entity relationship diagrams.

Cardinality Ratios:

Cardinality refers to the maximum number of times an instance in one entity can relate to instances of another entity.

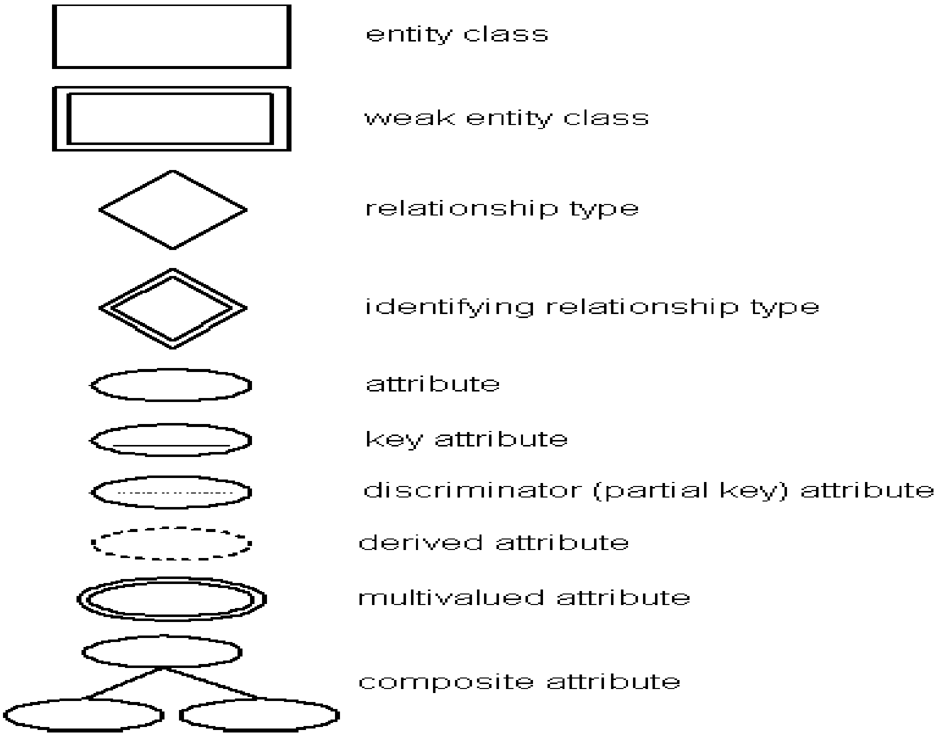


Figure 3.1: ER Diagram Symbols

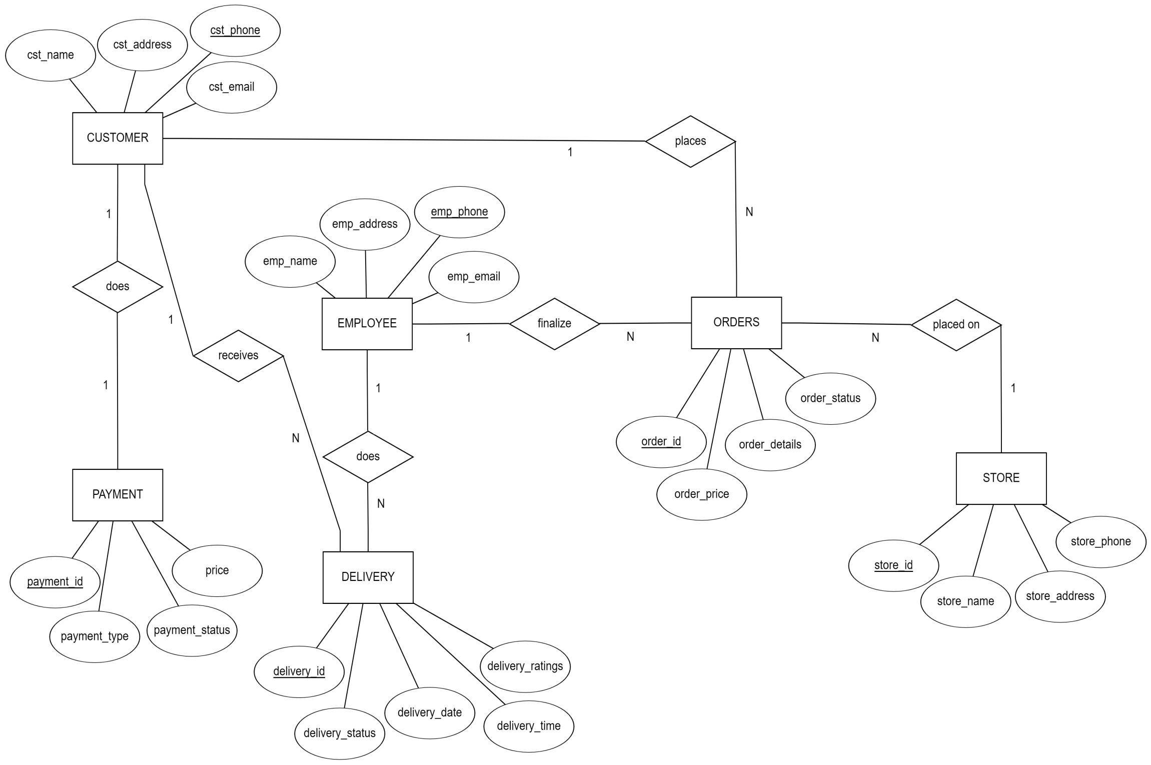


Figure 3.2: ER Diagram Cardinality Ratios

3.2 Mapping From ER Diagram to Schema Diagram

1.Mapping of Regular Entities:This step involves mapping all the regular entity types to tabular format by identifying their primary keys.

2.Mapping of 1:1 Relation: In this step foreign keys are assigned using foreign key approach.The primary key of the participating relation R or S is added as primary key to second entity types by looking at the participating constraints.

3.Mapping of 1:N Relation: Foreign key approach is used to add one sided primary key to the n sided entity at foreign key.

4.Mapping of M:N Relation: Here we use the cross reference approach where the relationship is converted to a new relation within attributes on primary keys of both participating relation.

5.Mapping of Weak Entity: When mapping weak entity types along with other attributes the partial key and primary key of parent entity together will form their primary key of the new relation.

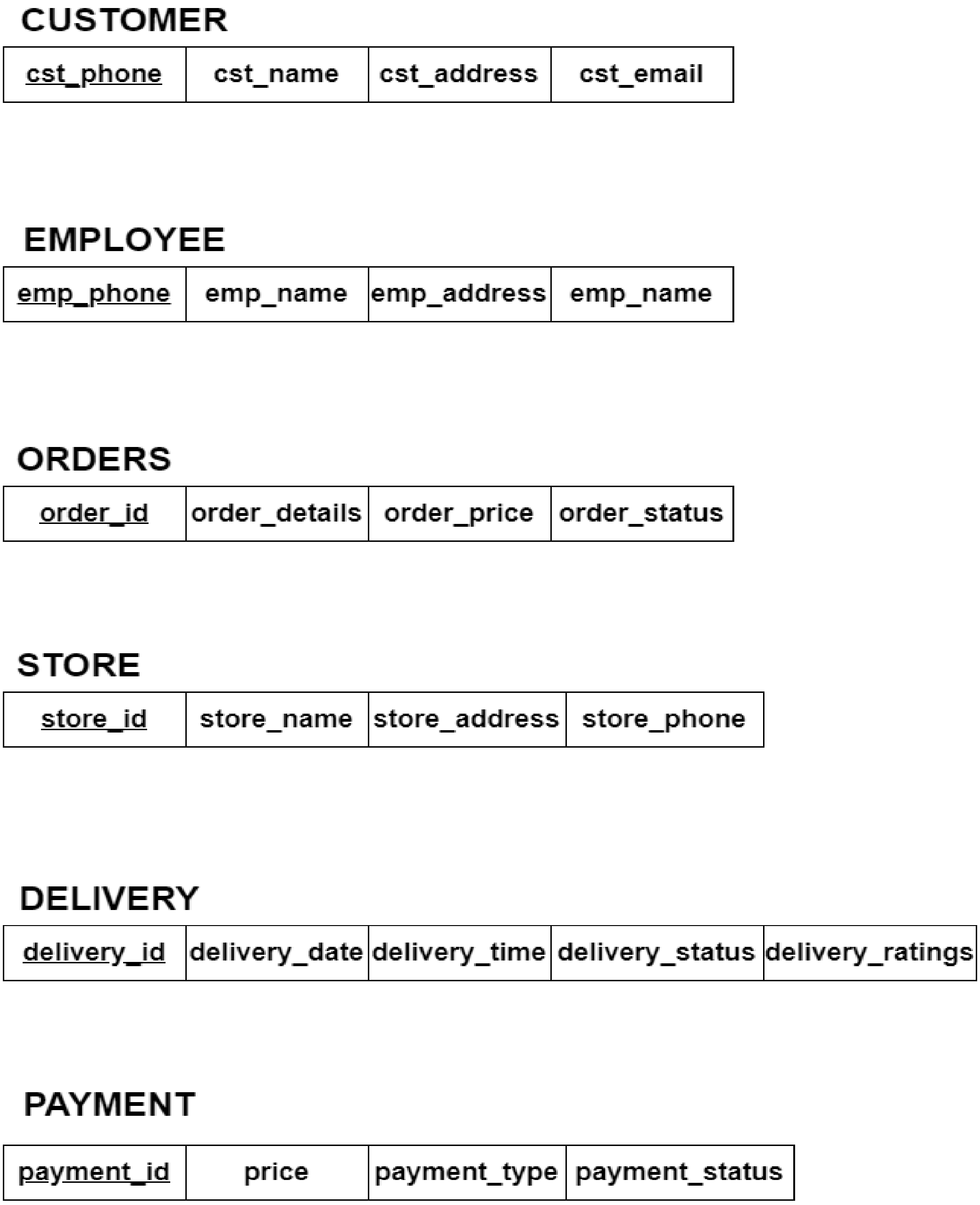
6.Mapping of N-ary Relation: For mapping N array relationship we create a new relation with a relationship name in its attribute and primary keys of all participating entity types.

7.Mapping of Multivalued Relation: For multivalued attributes a separate relation has to be created along with primary key of parent relation.

In our database we have the following mappings:

Step – 1 : Mapping of Regular Entities.

For each regular (strong) entity type E in the ER schema, create a relation R that includes all the simple attributes of E. Include only the simple component attributes of a composite attribute. Choose one of the key attributes of E as primary key for R. If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.



CUSTOMER

cst_phone	cst_name	cst_address	cst_email

EMPLOYEE

emp_phone	emp_name	emp_address	emp_name

ORDERS

order_id	order_details	order_price	order_status

STORE

store_id	store_name	store_address	store_phone

DELIVERY

delivery_id	delivery_date	delivery_time	delivery_status	delivery_ratings

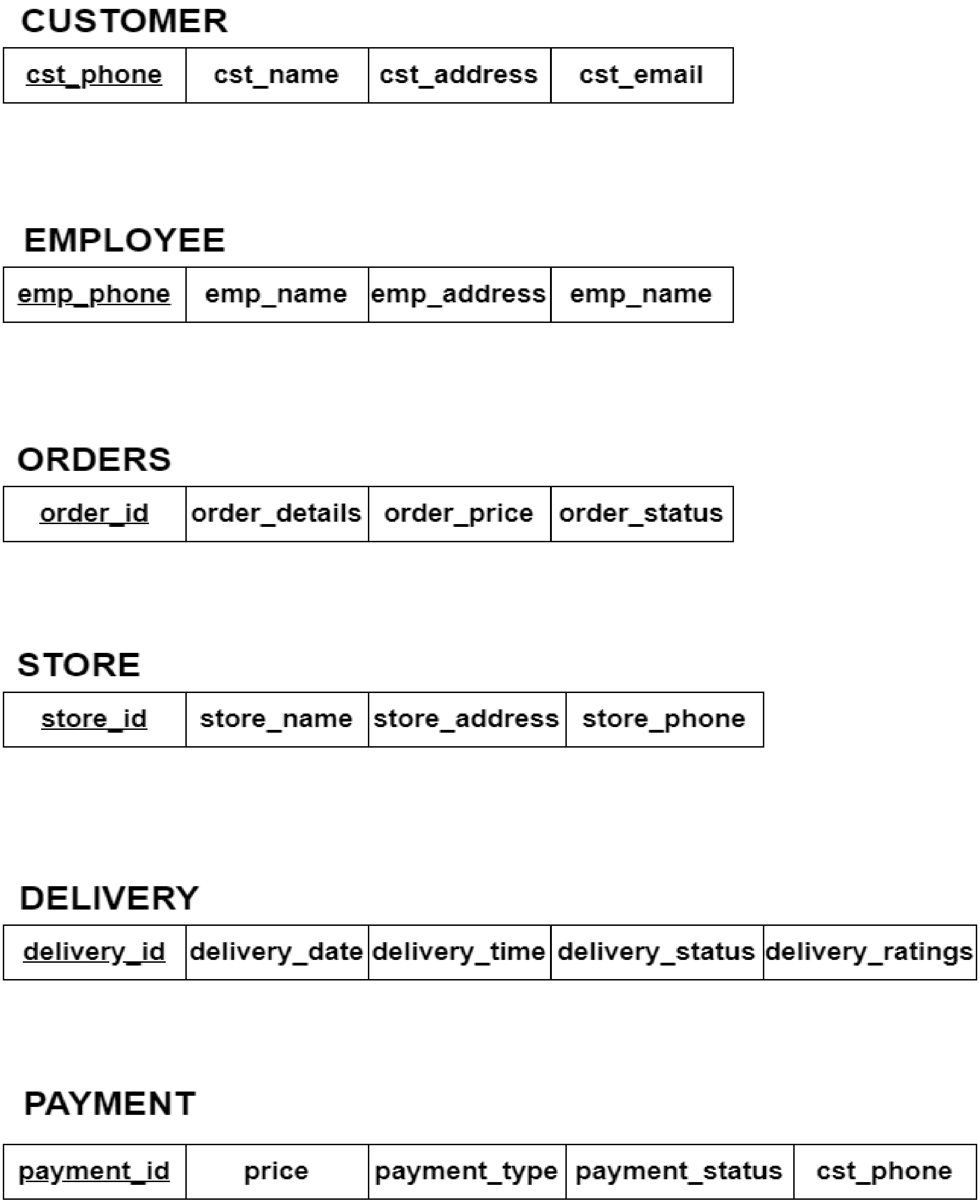
PAYMENT

payment_id	price	payment_type	payment_status

Figure 3.3: Regular Entities

Step – 2 : Mapping of binary 1:1 Relation Types.

The Payment and the Customer entities are participating in the 1:1 relation type. Since Payment has a total participation in the relation we include the primary key of Customer entity as the foreign key in Payment entity.



CUSTOMER

cst_phone	cst_name	cst_address	cst_email

EMPLOYEE

emp_phone	emp_name	emp_address	emp_name

ORDERS

order_id	order_details	order_price	order_status

STORE

store_id	store_name	store_address	store_phone

DELIVERY

delivery_id	delivery_date	delivery_time	delivery_status	delivery_ratings

PAYMENT

payment_id	price	payment_type	payment_status	cst_phone

Figure 3.4: 1:1 Relations

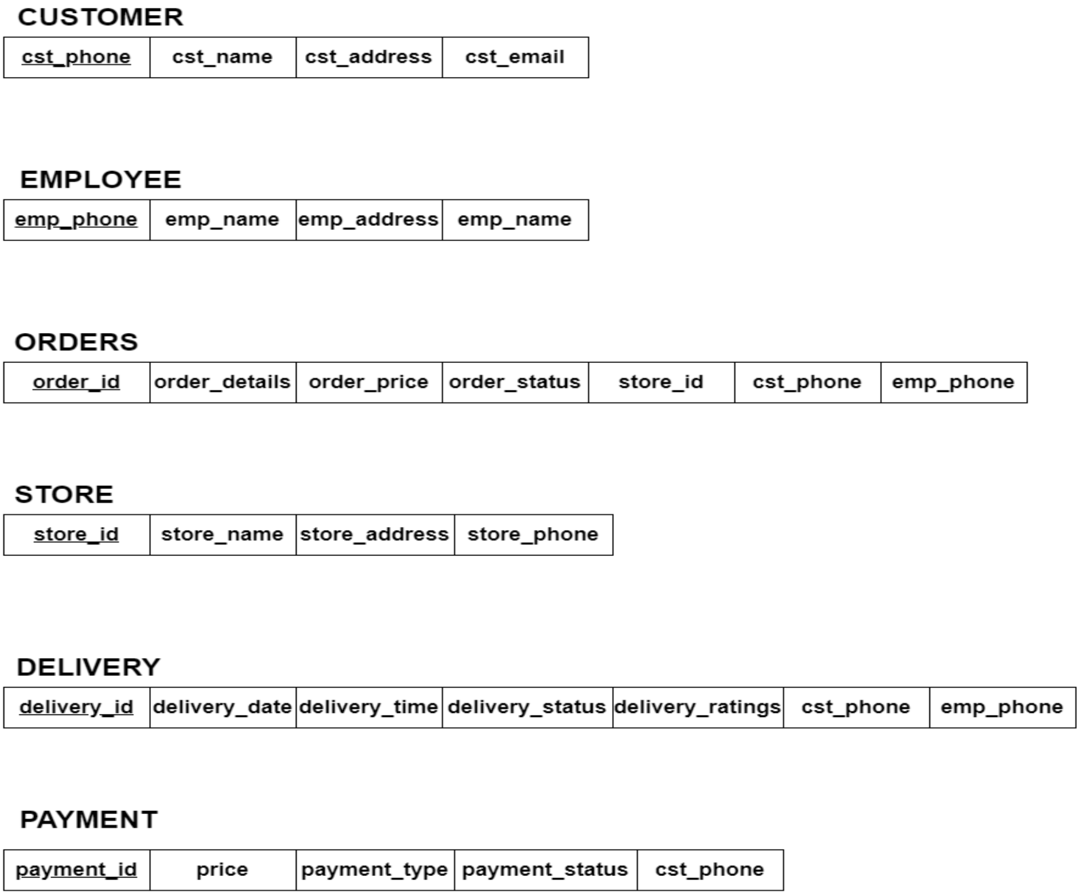
Step – 3 : Mapping of binary 1:N Relation Types.

The Customer and the Order entities are participating in the 1:N relation type. Since Order in one the nth side of the relation we include the primary key of Customer entity as the Foreign key in Order entity.

The Employee and the Order entities are participating in the 1:N relation type. Since Order in one the nth side of the relation we include the primary key of Employee entity as the Foreign key in Order entity.

The Store and the Order entities are participating in the 1:N relation type. Since Order in one the nth side of the relation we include the primary key of Store entity as the Foreign key in Order entity.

The Employee and the Delivery entities are participating in the 1:N relation type. Since Delivery in one the nth side of the relation we include the primary key of Employee entity as the Foreign key in Delivery entity.



CUSTOMER

cst_phone	cst_name	cst_address	cst_email

EMPLOYEE

emp_phone	emp_name	emp_address	emp_name

ORDERS

order_id	order_details	order_price	order_status	store_id	cst_phone	emp_phone

STORE

store_id	store_name	store_address	store_phone

DELIVERY

delivery_id	delivery_date	delivery_time	delivery_status	delivery_ratings	cst_phone	emp_phone

PAYMENT

payment_id	price	payment_type	payment_status	cst_phone

Figure 3.5: 1:N Relations

3.3 Schema Diagram

A Schema is a pictorial representation of the relationship between the database tables in the database that is created. The database schema of a database system is its structure described in a formal language sup ported by the database management system (DBMS). The term ”schema” refers to the organization of data as a blueprint of how the database is constructed (divided into database tables in the case of relational databases). The formal definition of a database schema is a set of formulas (sentences) called integrity constraints imposed on a database. These integrity constraints ensure compatibility between parts of the schema. All constraints are expressible in the same language. A database can be considered a structure in realization of the database language.The states of a created conceptual schema are transformed into an explicit mapping, the database schema. This describes how real-world entities are modelled in the database.

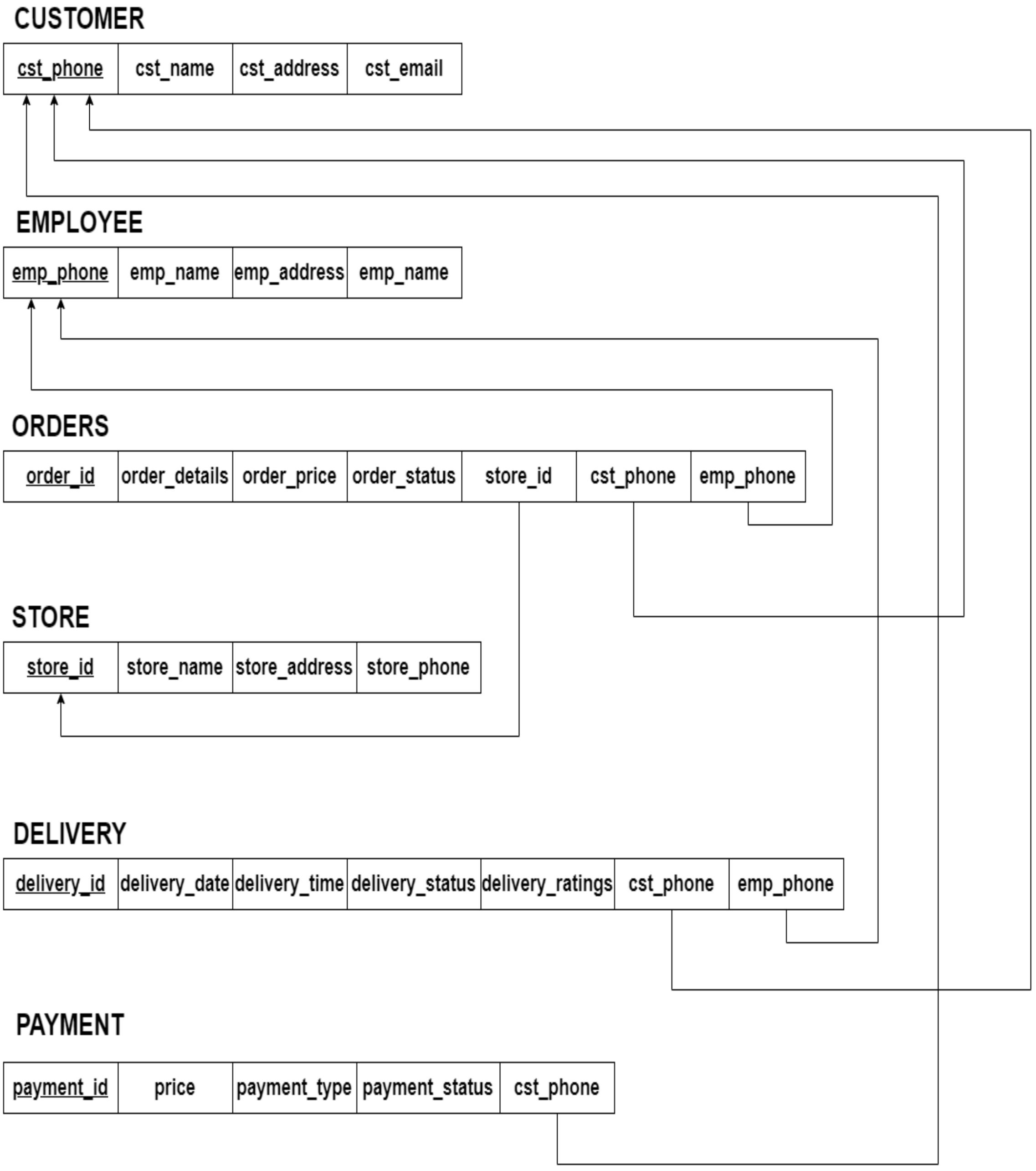


Figure 3.6: Schema Diagram Of Grocery Delivery System

Chapter 4

Implementation

4.1 Programming Languages Used

Python

Python is an interpreted, high-level and general-purpose programming language usedworldwide. Python's design philosophy emphasizes code readability with its notable useof significant white space.FlaskFlask is a micro web framework written in Python. It is classified as a micro-frameworkbecause it does not require particular tools or libraries. It has no database abstractionlayer, form validation, or any other components where pre-existing third-party librariesprovide common functions.Sqlite3SQLite is a relational database management system contained in a C library. In contrastto many other database management systems, SQLite is not a client–server databaseengine. Rather, it is embedded into the end program.

Flask

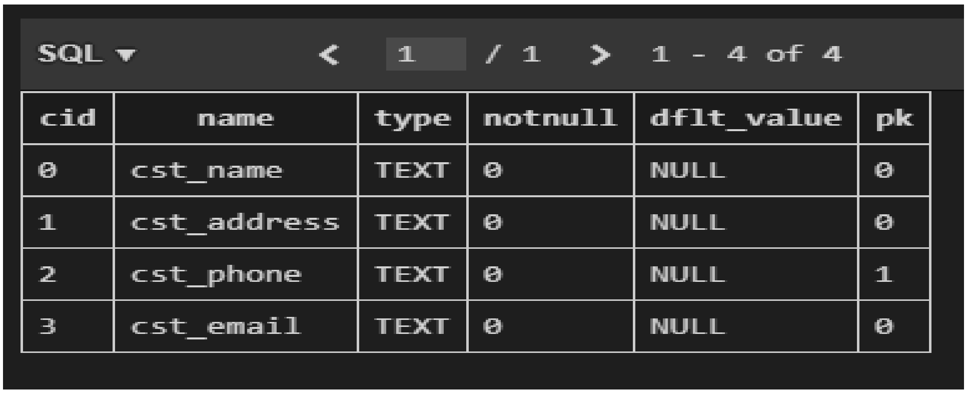
Flask is a micro web framework written in Python. It is classified as a micro-frameworkbecause it does not require particular tools or libraries. It has no database abstractionlayer, form validation, or any other components where pre-existing third-party librariesprovide common functions.

SQLite3

SQLite is a relational database management system contained in a C library. In contrastto many other database management systems, SQLite is not a client–server databaseengine. Rather, it is embedded into the end program.

4.2 Table Structure

CUSTOMER



cid	name	type	notnull	dflt_value	pk
0	cst_name	TEXT	0	NULL	0
1	cst_address	TEXT	0	NULL	0
2	cst_phone	TEXT	0	NULL	1
3	cst_email	TEXT	0	NULL	0

CREATE TABLE IF NOT EXISTS CUSTOMER(

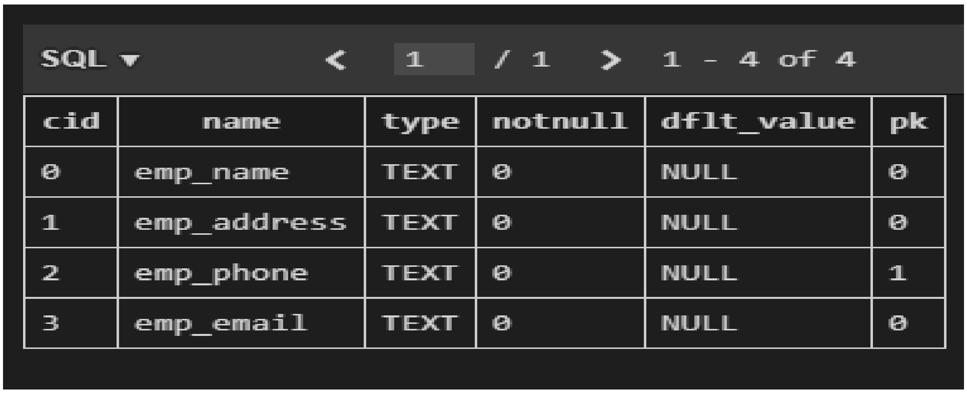
cst_name TEXT,

cst_address TEXT,

cst_ phone TEXT PRIMARY KEY,

cst_email TEXT);

EMPLOYEE



cid	name	type	notnull	dflt_value	pk
0	emp_name	TEXT	0	NULL	0
1	emp_address	TEXT	0	NULL	0
2	emp_phone	TEXT	0	NULL	1
3	emp_email	TEXT	0	NULL	0

CREATE TABLE IF NOT EXISTS EMPLOYEE(

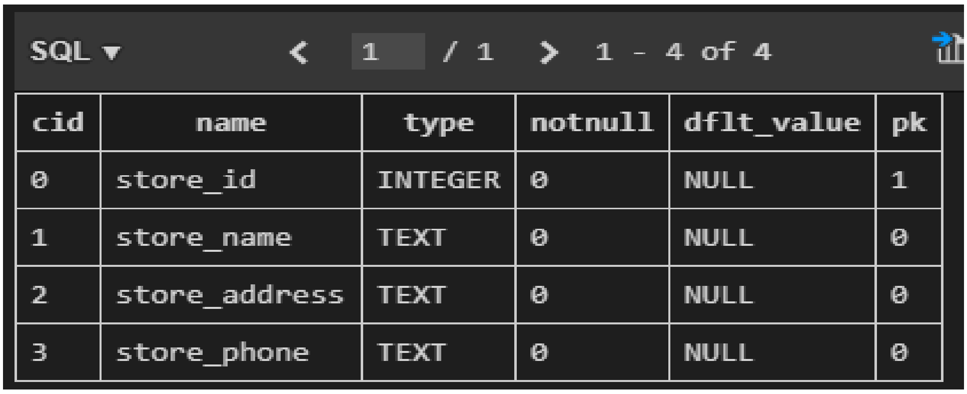
emp_name TEXT,

emp_address TEXT,

emp_phone TEXT PRIMARY KEY,

emp_email TEXT);

STORE



cid	name	type	notnull	dflt_value	pk
0	store_id	INTEGER	0	NULL	1
1	store_name	TEXT	0	NULL	0
2	store_address	TEXT	0	NULL	0
3	store_phone	TEXT	0	NULL	0

CREATE TABLE IF NOT EXISTS STORE(

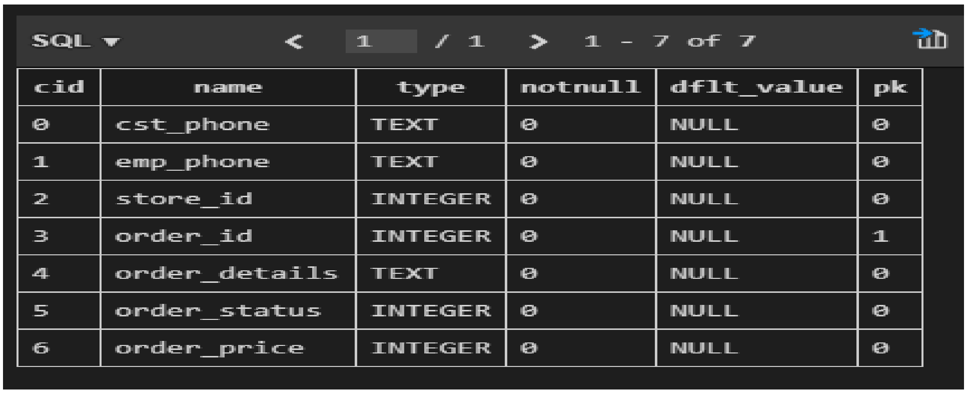
store_id INTEGER PRIMARY KEY,

store_name TEXT,

store_address TEXT,

store_phone TEXT);

ORDER



cid	name	type	notnull	dflt_value	pk
0	cst_phone	TEXT	0	NULL	0
1	emp_phone	TEXT	0	NULL	0
2	store_id	INTEGER	0	NULL	0
3	order_id	INTEGER	0	NULL	1
4	order_details	TEXT	0	NULL	0
5	order_status	INTEGER	0	NULL	0
6	order_price	INTEGER	0	NULL	0

CREATE TABLE IF NOT EXISTS ORDERS(

cst_phone TEXT,

emp_phone TEXT,

store_id INTEGER,

order_ id INTEGER PRIMARY KEY,

order_details TEXT,

order_status INTEGER,

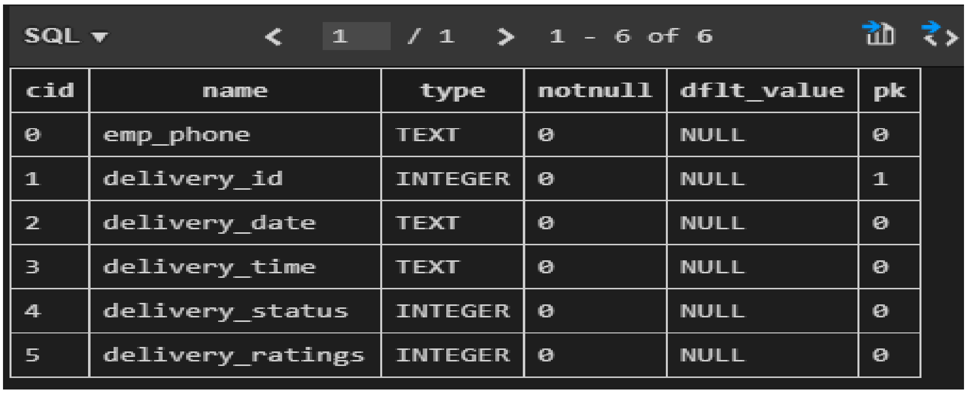
order_price INTEGER,

FOREIGN KEY(cst_phone) REFERENCES CUSTOMER(cst_phone) ON DELETE CASCADE,

FOREIGN KEY(emp_phone) REFERENCES EMPLOYEE(emp_phone) ON DELETE CASCADE,

FOREIGN KEY(store_id) REFERENCES STORE(store_id) ON DELETE CASCADE);

DELIVERY



cid	name	type	notnull	dflt_value	pk
0	emp_phone	TEXT	0	NULL	0
1	delivery_id	INTEGER	0	NULL	1
2	delivery_date	TEXT	0	NULL	0
3	delivery_time	TEXT	0	NULL	0
4	delivery_status	INTEGER	0	NULL	0
5	delivery_ratings	INTEGER	0	NULL	0

CREATE TABLE IF NOT EXISTS DELIVERY(

emp_phone TEXT,

cst_phone TEXT,

delivery_id INTEGER PRIMARY KEY,

delivery_date TEXT,

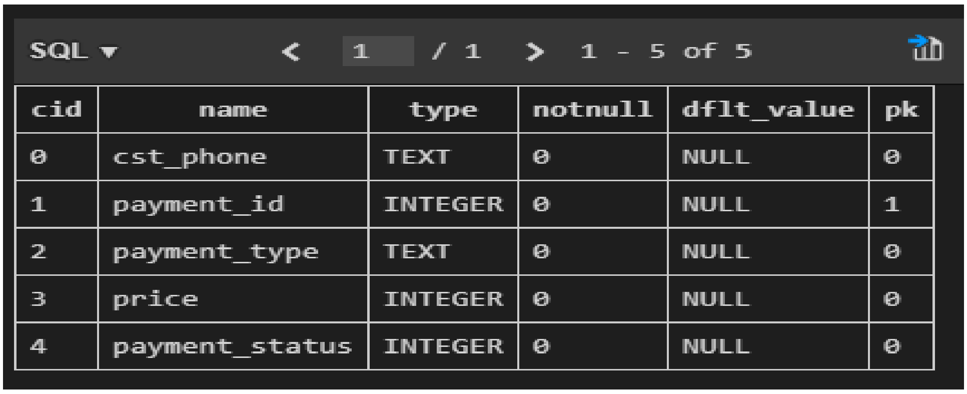
delivery_time TEXT,

delivery_status INTEGER,

delivery_ratings INTEGER,

FOREIGN KEY(cst_phone) REFERENCES CUSTOMER(cst_phone) ON DELETE CASCADE,

FOREIGN KEY(emp_phone) REFERENCES EMPLOYEE(emp_phone) ON DELETE CASCADE);



cid	name	type	notnull	dflt_value	pk
0	cst_phone	TEXT	0	NULL	0
1	payment_id	INTEGER	0	NULL	1
2	payment_type	TEXT	0	NULL	0
3	price	INTEGER	0	NULL	0
4	payment_status	INTEGER	0	NULL	0

PAYMENT

CREATE TABLE IF NOT EXISTS PAYMENT(

cst_phone TEXT,

payment_id INTEGER PRIMARY KEY,

payment_type TEXT,

price INTEGER,

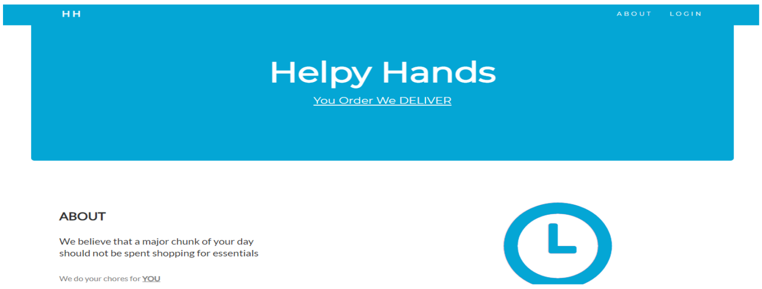
payment_status INTEGER,

FOREIGN KEY (cst_phone) REFERENCES CUSTOMER (cst_phone) ON DELETE CASCADE):

Chapter 5

Results and Discussion

Home Page:



Helpy Hands

You Order We DELIVER

ABOUT

We believe that a major chunk of your day should not be spent shopping for essentials

We do your chores for YOU

Figure 5.1: Home Page

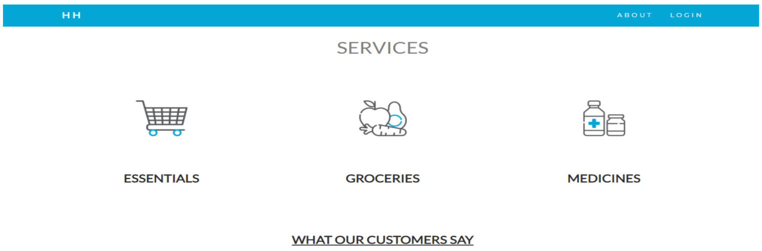


Figure 5.2: Home Page

This is the Helpy Hands home page. It includes the company goals and mission along with a nav link to the login page. The bottom has the customer testimonials reagarding our service.

Login Page:

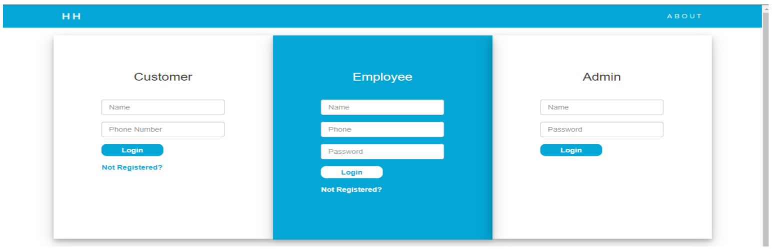


Figure 5.3: Login Page

Login form for Employee , Customer and Admin

Customer Menu

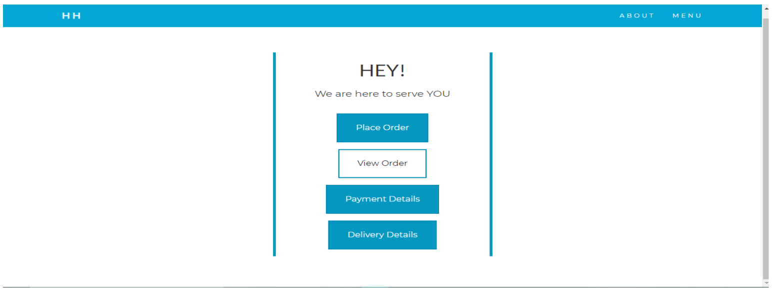
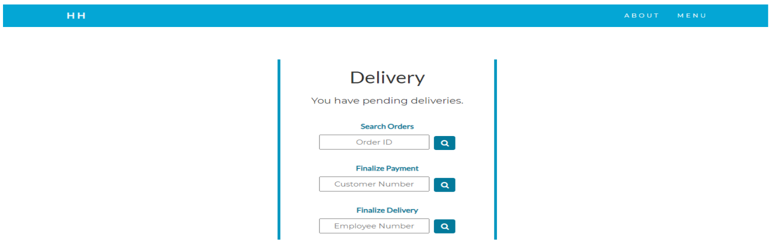


Figure 5.4: Customer Menu

This is the customer menu where a customer can place an order or view details of thier orders.

Employee Menu

The delivery employee is alerted of his deliveries. The delivery agent can track all details of his deliveries and customers



Delivery

You have pending deliveries.

Search Orders

Order ID

Finalize Payment

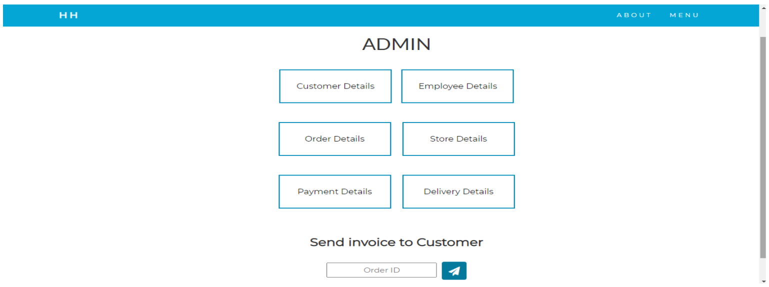
Customer Number

Finalize Delivery

Employee Number

Figure 5.5: Employee Menu

Admin Menu



ADMIN

Send invoice to Customer

Order ID

Figure 5.6: Admin Menu

Admin handles the order processing. Admin has master access to everything in the database

Chapter 6

Conclusion and Future work

Helpy Hands Grocery Delivery System will greatly help people from all walks of life. Since we are sometimes too busy or too lazy to go shopping, Having someone do it for us is a life saver. All supermarkets, vegetable shops, butchers and so on can register themselves as sellers on Helpy Hands and start earning soon after your first sale. Having practically put this idea into action, We have obtained positive results and looks promising. Our project can be improved by branching the service into other sectors like , local couriers, medicines and other essentials. Automating the task of the admin willl also improve customer shopping experience and speed. Giving a warning when the product quantity goes low. Keeping a track of the sales that have happened on a particular day, Issuing gift voucher or discount coupons for the customers.

References

[1] Database systems Models, Languages, Design and Application Programming, Ramez Elmasri and Shamkant B. Navathe, 6th Edition, Pearson.

[2] Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill.

[3] Silberschatz Korth and Sudharshan: Database System Concepts, 6th Edition, McGraw Hill, 2013.

[4] Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012.