**CHAPTER 1**

**INTRODUCTION**

**1.1 OVERVIEW**

With the advent of technology, innovation and improvement of every economic activity has become necessary to optimize the economic gain. In this connection particular several management techniques have been professed. For efficient management we need vital statistics (data) for taking valuable decision, which we call database. A database system is an integrated collection of related files, along with details of the integration of the data contained therein.

A database management system (DBMS) is a software system that allows access to data contained in a database. The objective of the DBMS is to provide a convenient and effective method of defining, storing and retrieving the information contained in the database. The DBMS interfaces with application programs, so that multiple applications and users can use the data contained in the database.

Data and database technology are having a major impact on the management of resources and computer stored information in today's industries. A database can be, in general terms, referred to as a collection of known facts that have been recorded and have an implicit meaning. A database is designed, built and populated with data for a specific purpose. It has an intended group of users and some preconceived applications in which these users are interested.

A database management is a collection of programs that enables a user to create and maintain a database. The DBMS is a general-purpose software system that facilitates the process of defining, constructing, and manipulating databases for various applications.

Defining a database involves specifying the data types, structures and constraints for the data to be stored in the database.

Constructing the database is the process of storing the data itself on some storage medium that is controlled by the DBMS.

Manipulating the database includes such functions as querying the database to receive specific data, updating the database, and generating reports from the data.

The database along with the software to manage it is collectively called a database system.

**1.2 Characteristics**

* It represents complex relationships between data.
* Keeps a tight control over data redundancy.
* Enforces user-defined rules to ensure the integrity of table data.
* Has a centralized data dictionary for the storage of information pertaining to data and its manipulation
* Ensures that data can be shared across applications
* Enforces data authorization.
* Has an automatic, intelligent backup and recovery procedure for data.
* Have different interfaces via which users can manipulate the data.
* Provides different levels of security constraints for manipulation of database structures.

**CHAPTER 2**

**SYSTEM REQUIREMENT SPECIFICATION**

A System Requirements Specification (SRS) (also known as Software Requirement Specification is a document or set of documentation that describes the features and behavior of a system or software application. It includes a variety of elements (see below) that attempts to define the intended functionality required by the customer to satisfy their different users.

In addition to specifying how the system should behave, the specification also defines at a high-level, the main business processes that will be supported, what simplifying assumptions have been made and what key performance parameters will need to be met by the system.

The hardware and software requirements of a computer system are required to install and use the software efficiency in the SRS. The minimum requirements need to be met for the program to run efficiently all the times on the system are as follows:

**2.1 SOFTWARE SPECIFICATION**

The software requirements are description of features and functionalities of the target system. Requirements convey the expectations of users from the software product. The requirements can be obvious or hidden, known or unknown, expected or unexpected from client’s point of view. The goal of requirement engineering is to develop and maintain sophisticated and descriptive ‘System Requirements Specification’ document.

The software requirements specify the pre-installed software needed to run the code being implemented in this project.

* Operating System: Windows 10, MacOS 10.15
* Backend Software: PHP, MySQL, Xampp Server(Control Panel)
* Frontend Software: Java, HTML and CSS

**2.2 HARDWARE REQUIREMENT**

Usage of CPU, RAM and storage space can vary significantly based on user behavior. These hardware recommendations are based on traditional deployments and may grow or shrink depending on how active users are. The hardware requirement specifies the necessary hardware which provides us the platform to implement our programs.

* Processor: Any x86 or x64 Architecture Processor.
* Processor Speed: 1.4 GHZ or above.
* RAM: 4 GB RAM or above.
* HARD DISK: 20 GB hard disk or above.

The hardware and software components of s computer system that are required to install and use software efficiently are specified in SRS. The minimum system requirements need to be met for the program to run all times on the system.

**CHAPTER 3**

**DESCRIPTION**

A database is a structured collection of data. Data refers to the characteristics of people things and events. A database management system gives the user access to their data and helps them transform the data into information. Such database management system includes dBase paradox, IMS, SQL server and MySQL. These systems allow users to create, update and extract information from their database. Different Tables are created for the various groups of information. Related tables are grouped together to form a database.

This project has been designed using MySQL as back end and Java graphics programming API’s as front end.

**3.1 My STRUCTURED QUERY LANGUAGE (MySQL)**

**MySQL** is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS). Its name is a combination of "My", the name of co-founder [Michael Widenius](https://en.wikipedia.org/wiki/Michael_Widenius)'s daughter, and "[SQL](https://en.wikipedia.org/wiki/SQL)", the abbreviation for [Structured Query Language](https://en.wikipedia.org/wiki/Structured_Query_Language). A [relational database](https://en.wikipedia.org/wiki/Relational_database) organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data. SQL is a language programmers use to create, modify and extract data from the relational database, as well as control user access to the database. In addition to relational databases and SQL, an RDBMS like MySQL works with an [operating system](https://en.wikipedia.org/wiki/Operating_system) to implement a relational database in a computer's storage system, manages users, allows for network access and facilitates testing database integrity and creation of backups.

MySQL has stand-alone clients that allow users to interact directly with a MySQL database using SQL, but more often, MySQL is used with other programs to implement applications that need relational database capability. MySQL is a component of the [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) [web application](https://en.wikipedia.org/wiki/Web_application) [software stack](https://en.wikipedia.org/wiki/Software_stack) (and [others](https://en.wikipedia.org/wiki/List_of_AMP_packages)), which is an acronym for [*Linux*](https://en.wikipedia.org/wiki/Linux)*,*[*Apache*](https://en.wikipedia.org/wiki/Apache_HTTP_Server)*, MySQL,*[*Perl*](https://en.wikipedia.org/wiki/Perl)*/*[*PHP*](https://en.wikipedia.org/wiki/PHP)*/*[*Python*](https://en.wikipedia.org/wiki/Python_(programming_language)). MySQL is used by many database-driven web applications, including [Drupal](https://en.wikipedia.org/wiki/Drupal), [Joomla](https://en.wikipedia.org/wiki/Joomla), [phpBB](https://en.wikipedia.org/wiki/PhpBB" \o "PhpBB), and [WordPress](https://en.wikipedia.org/wiki/WordPress). MySQL is also used by many popular [websites](https://en.wikipedia.org/wiki/Website), including [Facebook](https://en.wikipedia.org/wiki/Facebook), [Flickr](https://en.wikipedia.org/wiki/Flickr), [MediaWiki](https://en.wikipedia.org/wiki/MediaWiki), [Twitter](https://en.wikipedia.org/wiki/Twitter), and [YouTube](https://en.wikipedia.org/wiki/YouTube).

**3.2 MySQL 8.0 command line client**

MySQL command line client is a free version of the world's most capable relational database. With MySQL command line client, you use an intuitive, command-prompt based interface, to:

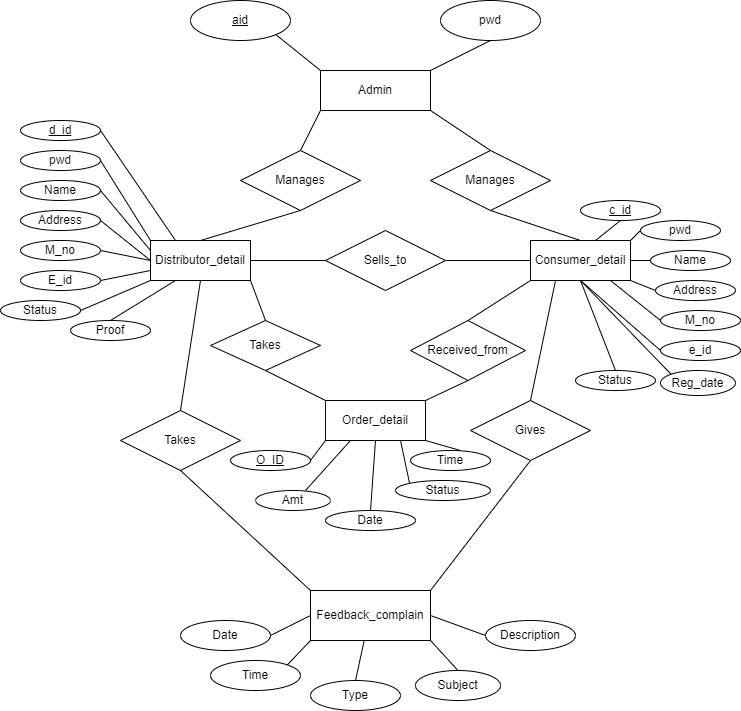
* Administer the database
* Create table, views, and other database object Import, export, and view table data
* Run queries and MySQL script.
* Generate reports

**3.3 ER DIAGRAM**

An entity-relationship model (ER model) is a data model for describing the data or information aspects of a business domain or its process requirements, in an abstract way that lends itself to ultimately being implemented in a database such as a relational database. The main components of ER models are entities (things) and the relationships that can exist among them. Entity-relationship modelling was developed by Peter Chen and published in a 1976 paper. The ER diagram is drawn to have a better understanding of the whole scenario, it is used to conceptualize the phenomena, actions and interactions between various entities and to arrive at the specific requirements in a comprehensive manner.

An entity-relationship model is the result of using a systematic process to describe and define a subject area of business data. The data is represented as components (entities) that are linked with each other by relationships that express the dependencies and requirements between them, such as: one building may be divided into zero or more apartments, but one apartment can only be located in one building. Entities may have various properties (attributes) that characterize them. Diagrams created to represent these entities, attributes, and relationships graphically are called entity relationship diagrams. An ER model is typically implemented as a database. In the case of a relational database, which stores data in tables, every row of each table represents one instance of an entity. Some data fields in these tables point to indexes in other tables, such pointers are the physical implementation of the relationship. The three-schema approach to software engineering uses three levels of ER models that may be developed.

A relationship captures how entities are related to one another. Relationships can be thought of as verbs, linking two or more nouns. Examples: An owns relationship between a company and a computer, a supervised relationship between an employee and a department, a performs relationship between an artist and a song relationship between a mathematician and a conjecture. Entities and relationships can both have attributes Examples: an employee entity might have a Social Security Number (SSN) attribute, the proved relationship may have a date attribute.

 Fig1: Entity-Relation Model of the Prescribed Management Model

**3.4 RELATIONAL SCHEMA**

A database schema is the skeleton structure that represents the logical view of the entire database. It defines how the data is organized and how the relations among them are associated. It formulates all the constraints that are to be applied on the data.

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams. It's the database designers who design the schema to help programmers understand the database and make it useful.

A database schema can be divided broadly into two categories -

* **Physical Database Schema** - This schema pertains to the actual storage of data sand its form of storage like files, indices, etc. It defines how the data will be stored in a secondary storage.
* **Logical Database Schema** - This schema defines all the logical constraints that need to be applied on the data stored. It defines tables, views, and integrity constraints.

Database schema is the skeleton of database. It is designed when the database doesn't exist at all. Once the database is operational, it is very difficult to make any changes to it. A database schema does not contain any data that will be inserted in the database later.

**Schema Diagram:**

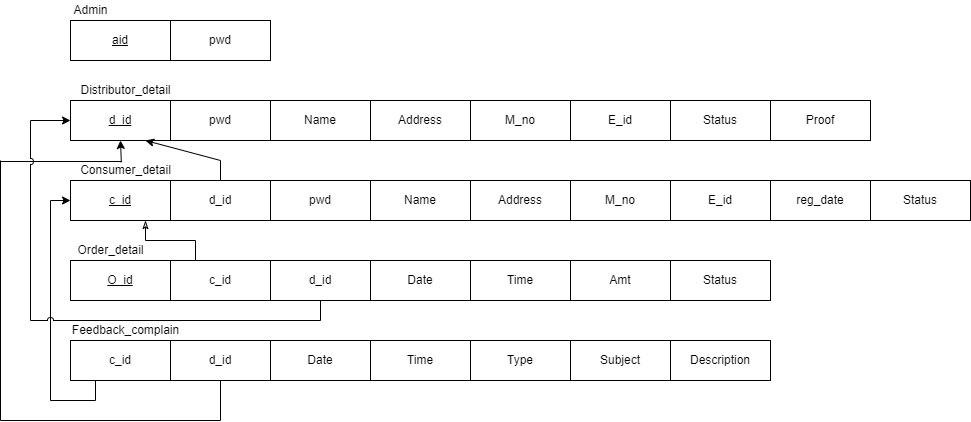


Fig 2: Relational Schema Model of Gas Agency Management System

**3.5 Entity :**

Basic object that the ER model represent in an entity, which is a thing in the real world with an independent existence. An entity may be an object with a physical existence (e.g. a company, a job or a universally course).

**3.6 Attribute :**

Each entity has attributes. It is a particular property that describes entity. For example, an employee entity maybe described by the employee’s name, age, address, salary and job. A particular entity will have a value for each of its attributes. The attribute value that describes each entity becomes a major part of the data stored in the database.

**CHAPTER 4**

**CODING**

**4.1 CREATION OF TABLE**

**ADMIN**

CREATE TABLE `admin` (

`aid` int(10) NOT NULL,

`pwd` varchar(16) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

INSERT INTO `admin` (`aid`, `pwd`) VALUES

(1, 'rootroot');

**CONSUMER DETAILS**

CREATE TABLE `consumer\_detail` (

`cid` int(10) NOT NULL,

`did` int(5) NOT NULL,

`pwd` varchar(16) DEFAULT NULL,

`name` varchar(30) NOT NULL,

`address` varchar(100) NOT NULL,

`city` varchar(10) NOT NULL,

`pin` int(6) NOT NULL,

`m\_no` bigint(10) NOT NULL,

`e\_id` varchar(30) NOT NULL,

`reg\_date` date DEFAULT NULL,

`status` varchar(15) NOT NULL DEFAULT 'Not Registered'

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

**DISTRIBUTOR DETAIL**

CREATE TABLE `distributor\_detail` (

`did` int(5) NOT NULL,

`pwd` varchar(16) NOT NULL,

`name` varchar(20) NOT NULL,

`address` varchar(100) NOT NULL,

`city` varchar(10) NOT NULL,

`pin` int(6) NOT NULL,

`m\_no` bigint(10) NOT NULL,

`e\_id` varchar(30) NOT NULL,

`status` varchar(10) NOT NULL DEFAULT 'Deactive',

`proof` text NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

**FEEDBACK COMPLAIN**

CREATE TABLE `feedback\_complain` (

`cid` int(10) NOT NULL,

`did` int(5) NOT NULL,

`date` date NOT NULL,

`time` time NOT NULL,

`type` varchar(10) NOT NULL,

`subject` varchar(30) NOT NULL,

`description` varchar(300) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

**ORDER DETAIL**

CREATE TABLE `order\_detail` (

`oid` int(10) NOT NULL,

`cid` int(10) NOT NULL,

`did` int(5) NOT NULL,

`date` date NOT NULL,

`time` time NOT NULL,

`amt` float NOT NULL DEFAULT '475',

`status` varchar(20) NOT NULL DEFAULT 'Pending'

) ENGINE=InnoDB DEFAULT CHARSET=latin1;

**4.2 INSERTION OF VALUES IN THE TABLE:**

**ADMIN**

INSERT INTO `admin` (`aid`, `pwd`) VALUES

(1, 'rootroot');

**CONSUMER DETAIL**

INSERT INTO `consumer\_detail` (`cid`, `did`, `pwd`, `name`, `address`, `city`, `pin`, `m\_no`, `e\_id`, `reg\_date`, `status`) VALUES

(2, 1, '111111', 'Raj Pravinbhai Zalavadiya', '201,Sai Darshan, Nana Varachha', 'Surat', 394525, 8787954652, 'raj@outlook.com', '2018-02-07', 'Deactive'),

(3, 2, '333333', 'Biren V. Gadhiya', 'B12, Khodiyar row house, Katargam', 'Ahmedabad', 366666, 7877777777, 'bg@outlook.com', '2018-02-08', 'Active'),

(5, 1, '555555', 'Jenish Kishorbhai Mangukiya', 'D2/304, Gangotri rec., Sudama chowk, Mota Varachha', 'Surat', 394101, 9904436106, 'mjenish8@gmail.com', '2018-02-09', 'Active'),

(6, 1, NULL, 'Krunal Dineshbhai Lukhi', '21, White House, Panvel point road', 'Surat', 394101, 9876543210, 'kd\_likhi@gmail.com', NULL, 'Not Registered'),

(225, 1, '225225', 'Ravi K. Kanpara', '22', 'Surat', 388554, 8565655521, 'ravi@gmail.com', '2018-02-21', 'Active');

**DISTRIBUTOR DETAIL**

INSERT INTO `distributor\_detail` (`did`, `pwd`, `name`, `address`, `city`, `pin`, `m\_no`, `e\_id`, `status`, `proof`) VALUES

(1, '111111', 'Jai Ganesh Agency', 'D12, GIDC-Kapodra', 'Surat', 384511, 8200703812, 'jganesh@gmail.com', 'Active', 'distributor/proof/1.jpg'),

(2, '222222', 'Gitanjali PVT LTD', '12, Amazon plaza, Ring road', 'Ahmedabad', 355002, 8547856321, 'gpvtltd@hotmail.com', 'Active', 'distributor/proof/2.jpg'),

(9, '111111', 'Nandan LTD', '12, Birla Mart, Behind Maruti Chowk', 'Bhavnagar', 555656, 7898765421, 'nan@ymail.com', 'Active', 'distributor/proof/9.jpg'),

(10, '333333', 'Tapovan Group', 'A55, JZ Shopping Mart, LH road', 'Vadodara', 366444, 9875452121, 'tapo@gmail.com', 'Deactive', 'distributor/proof/10.png');

**FEEDBACK COMPLAIN**

INSERT INTO `feedback\_complain` (`cid`, `did`, `date`, `time`, `type`, `subject`, `description`) VALUES

(5, 1, '2018-02-25', '10:34:46', 'Feedback', 'Product related', 'Last product delivery taken two weeks.'),

(5, 1, '2018-02-25', '10:34:53', 'Feedback', 'Website related', 'Your website\'s interface is awesome.'),

(225, 1, '2018-02-25', '10:35:48', 'Complaint', 'Defected Product', 'Delivered product is defected.'),

(225, 1, '2018-02-25', '10:35:54', 'Feedback', 'Website related', 'Good user-interface.'),

(3, 2, '2018-04-06', '18:14:48', 'Complaint', 'Notification related', 'Since two month, I didn\'t getting any SMS/Email regarding product delivery, I am doing manually by checking order status in website. Please fix this issue.');

**ORDER DETAIL**

INSERT INTO `order\_detail` (`oid`, `cid`, `did`, `date`, `time`, `amt`, `status`) VALUES

(10, 5, 1, '2018-02-14', '10:07:17', 475, 'Delivered'),

(15, 5, 1, '2018-02-14', '10:15:24', 475, 'Delivered'),

(16, 5, 1, '2018-02-14', '11:13:52', 475, 'Delivered'),

(17, 225, 1, '2018-02-21', '10:10:04', 475, 'Delivered'),

(18, 225, 1, '2018-02-26', '10:04:35', 475, 'Approved'),

(19, 5, 1, '2018-03-21', '18:36:45', 475, 'Delivered'),

(24, 5, 1, '2018-03-22', '13:32:31', 475, 'Pending'),

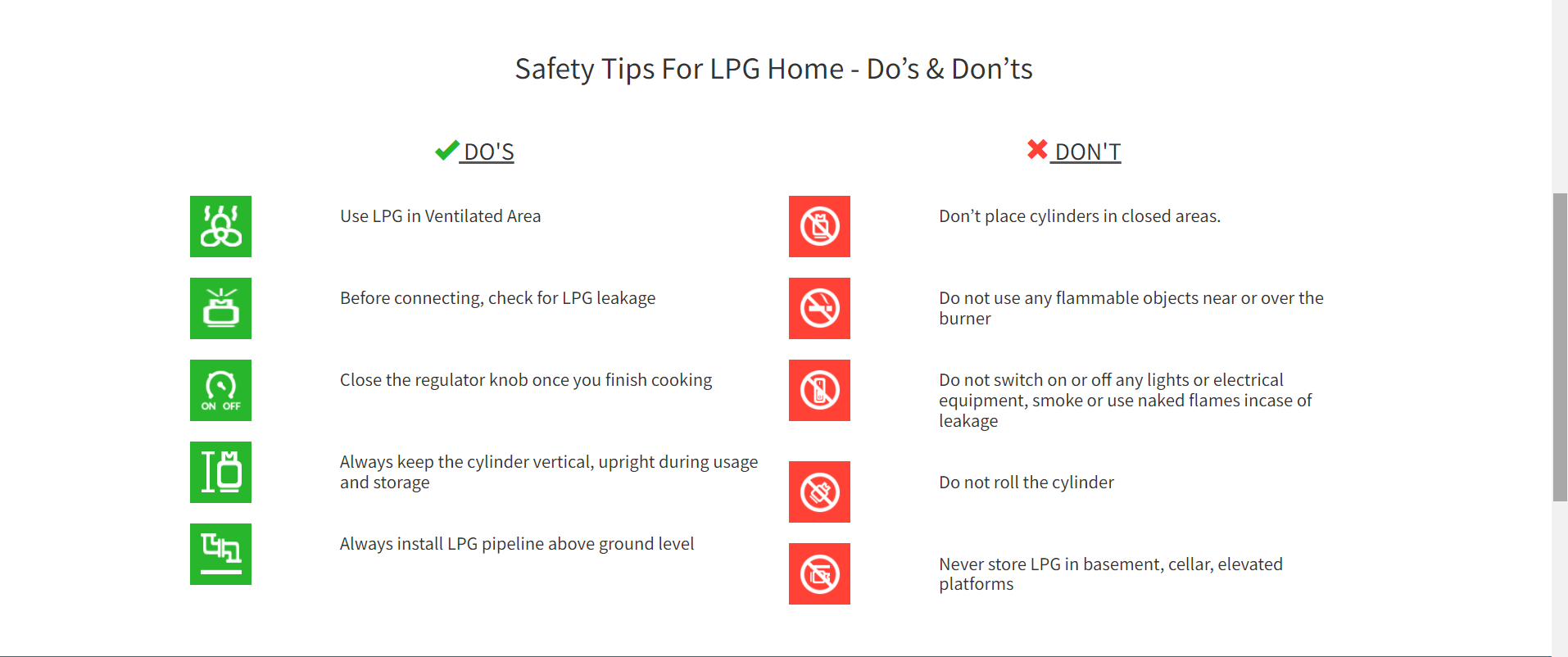
(25, 3, 2, '2018-04-06', '17:39:56', 475, 'Pending');

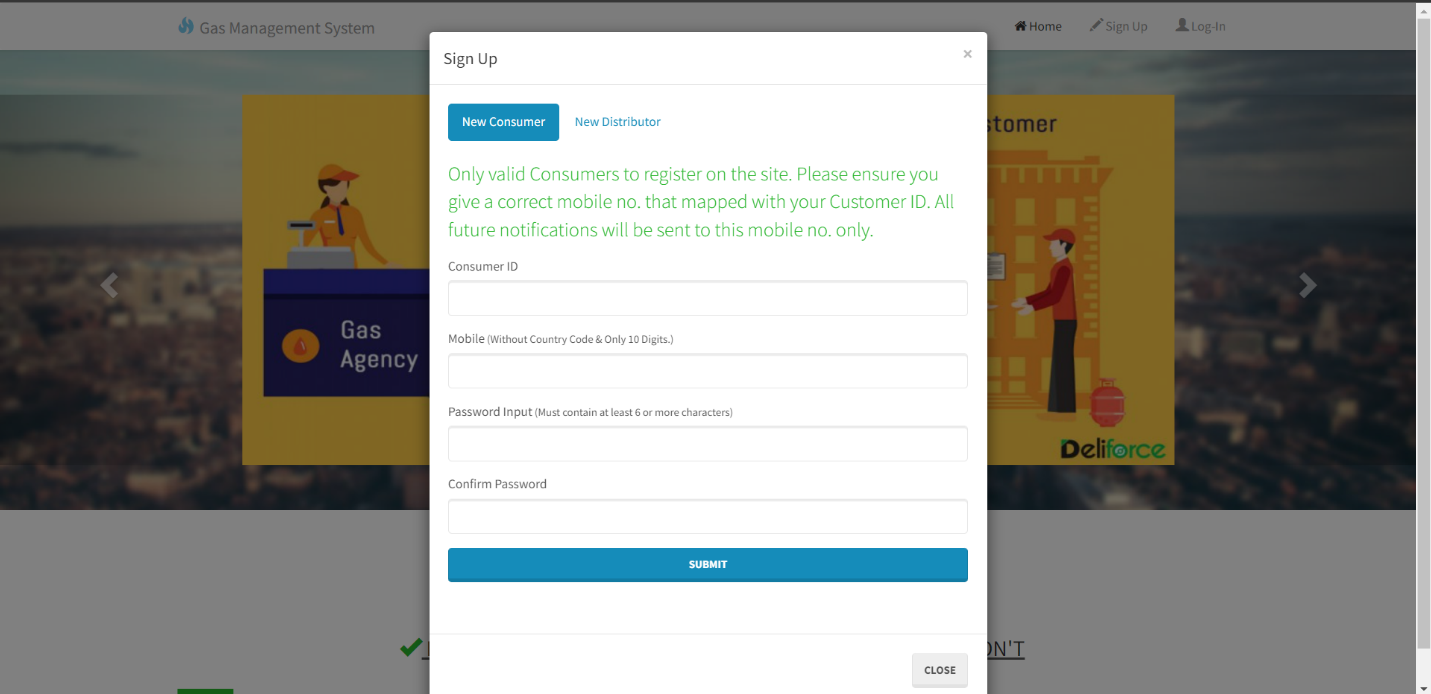
**CHAPTER 5**

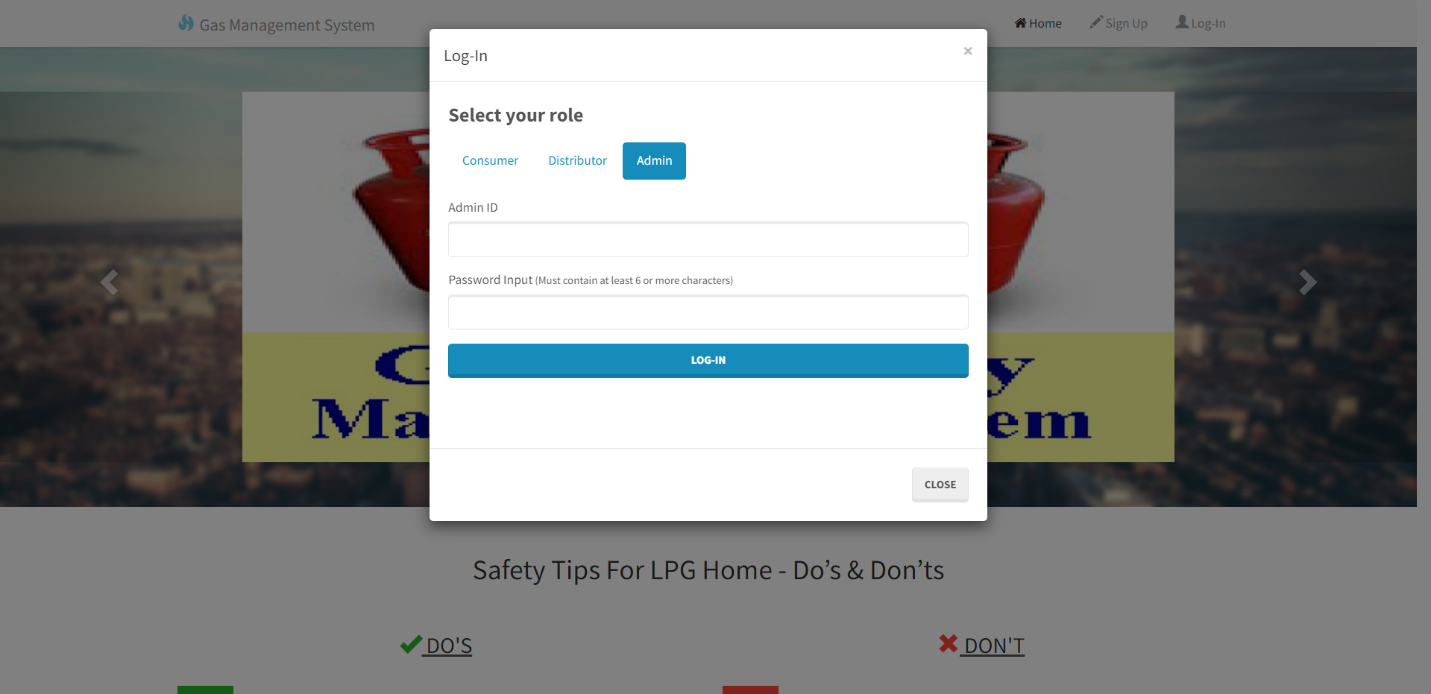
**SNAPSHOTS**

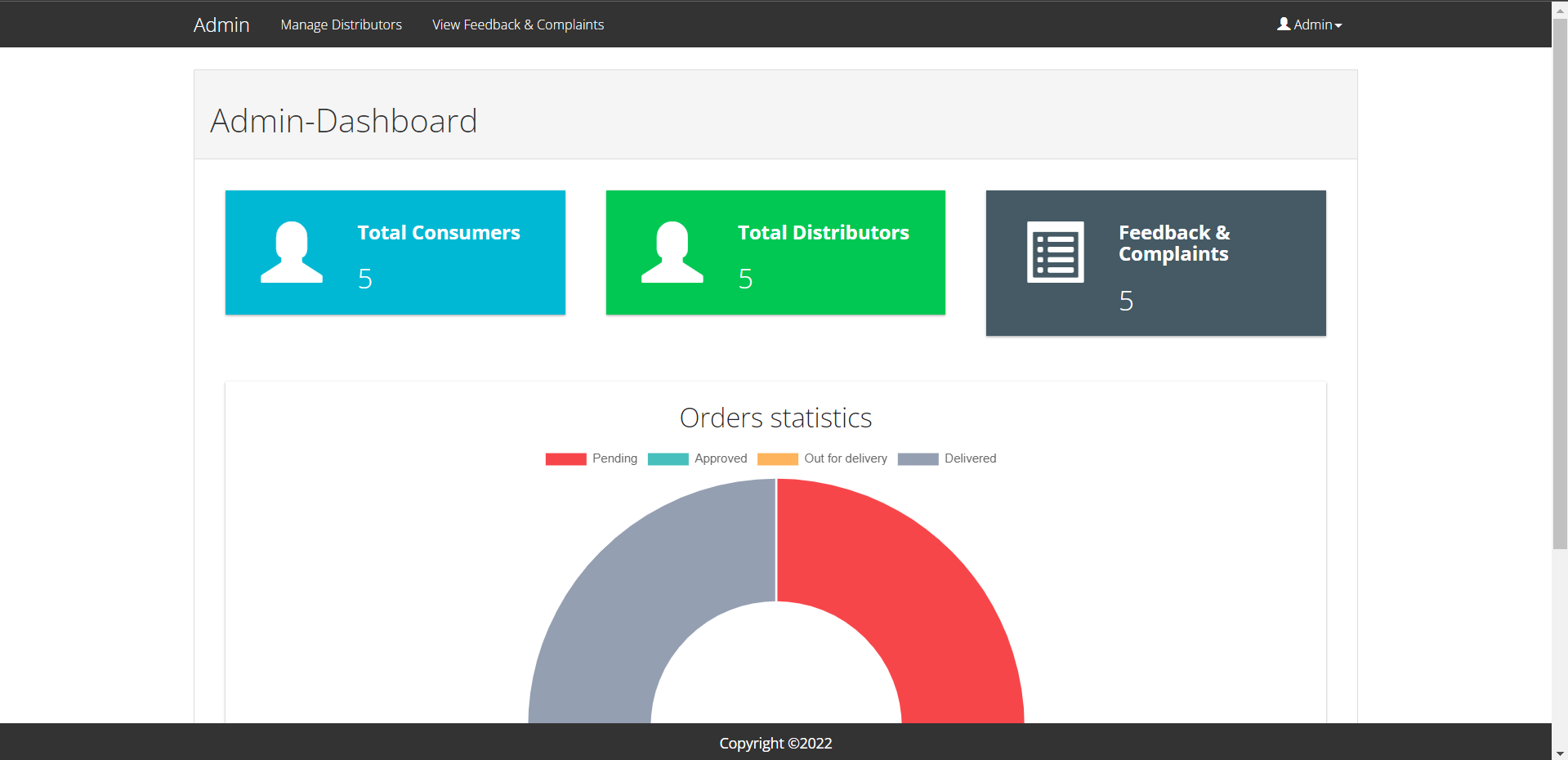
**5.1 Frontend Structure**

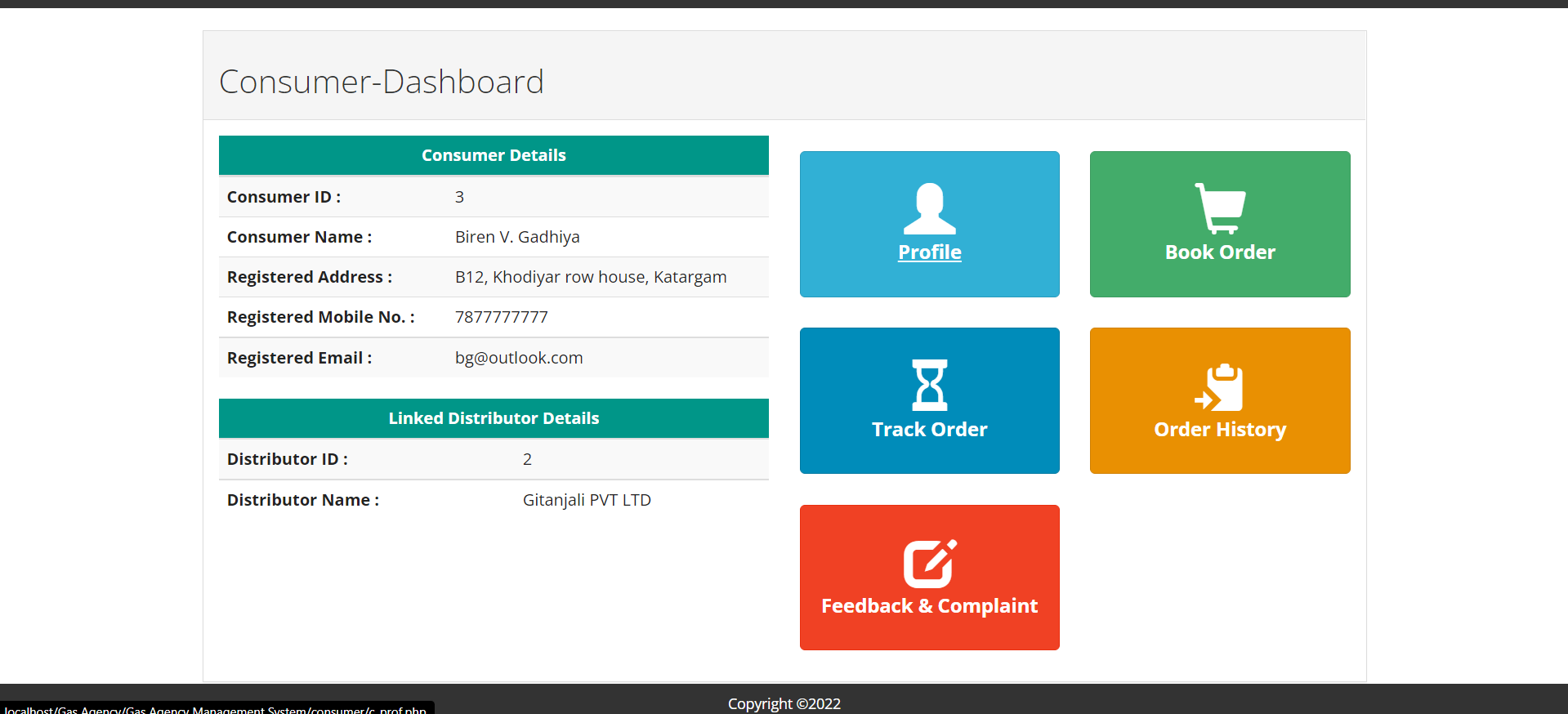


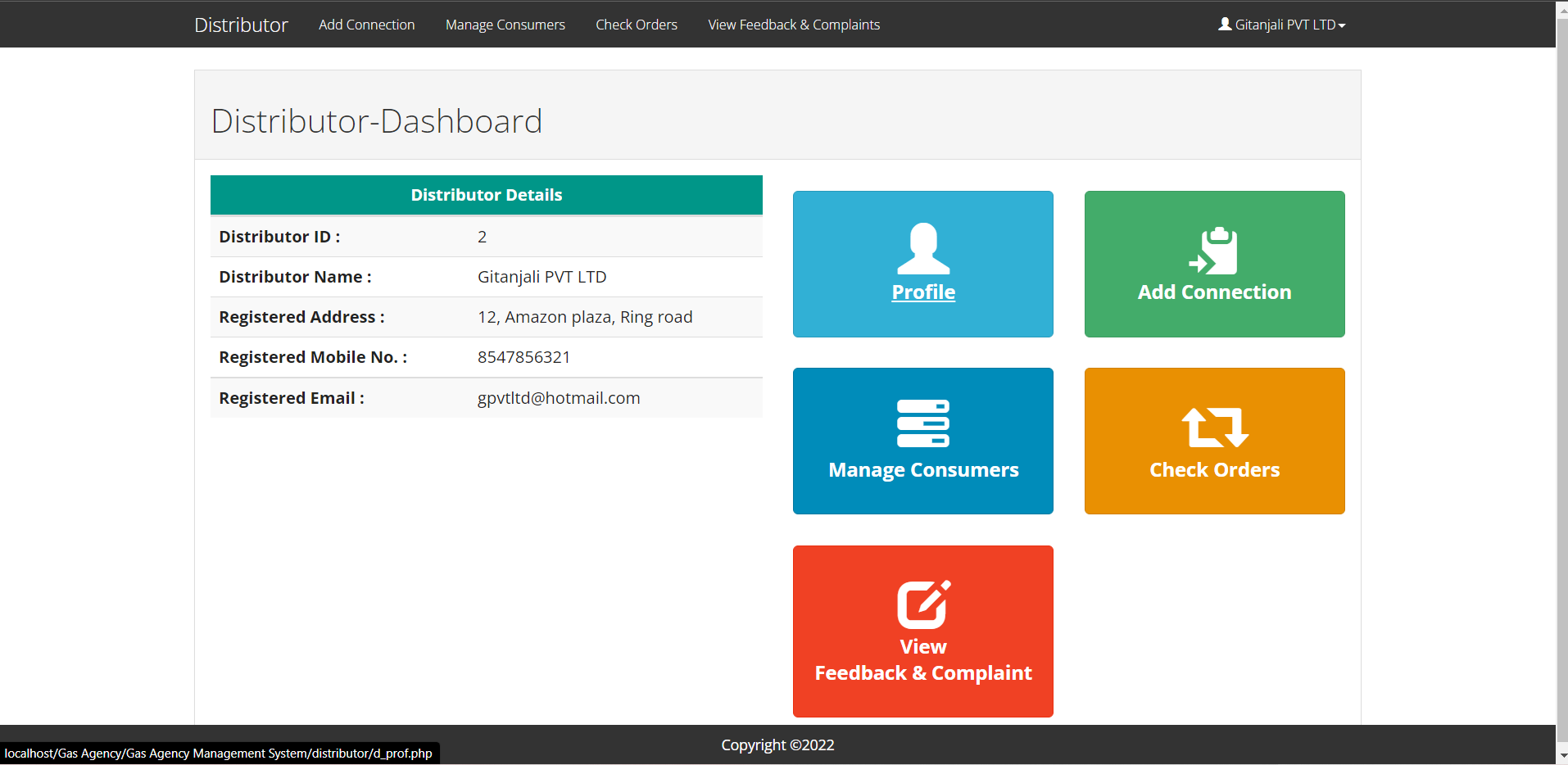


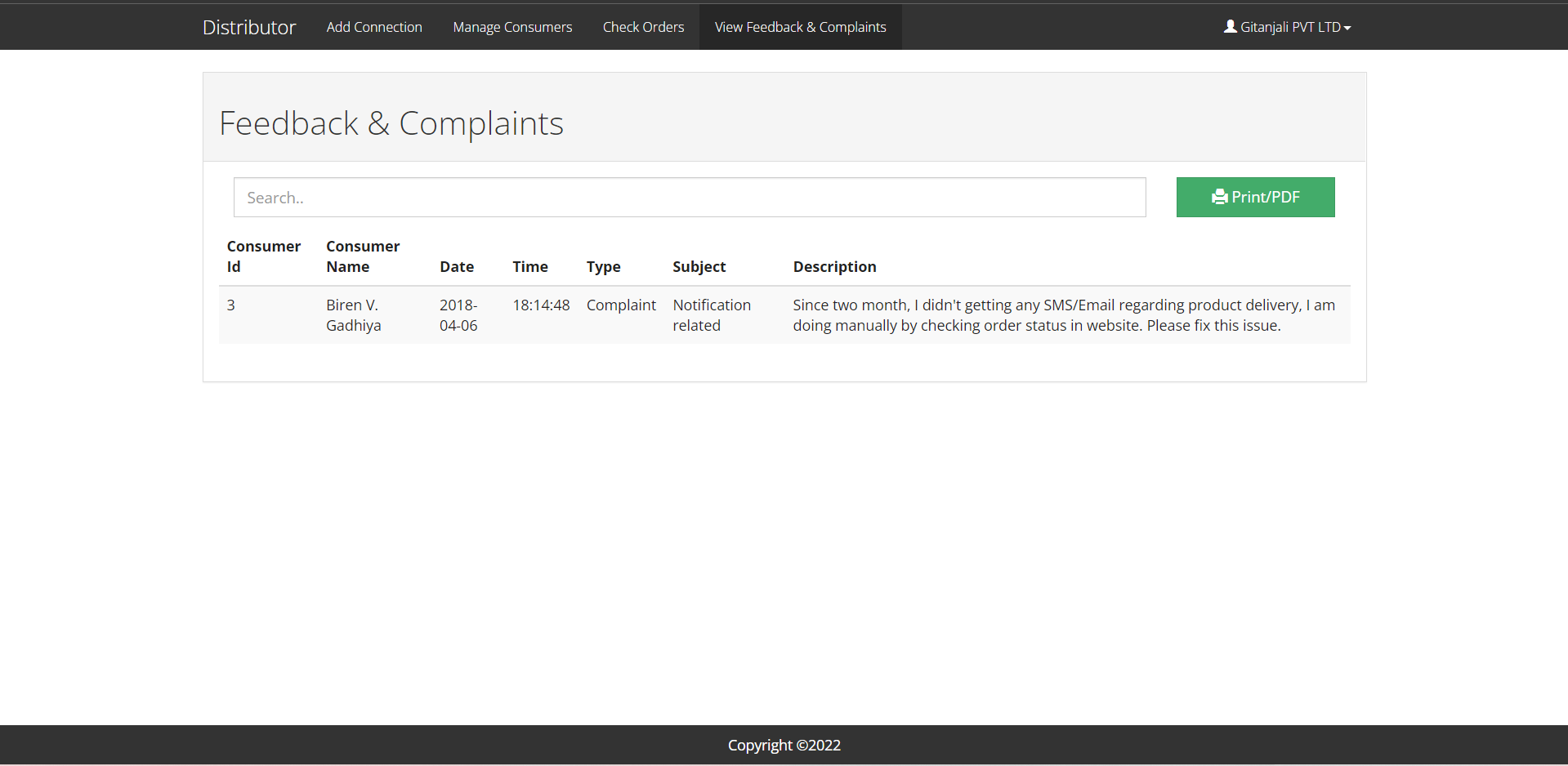




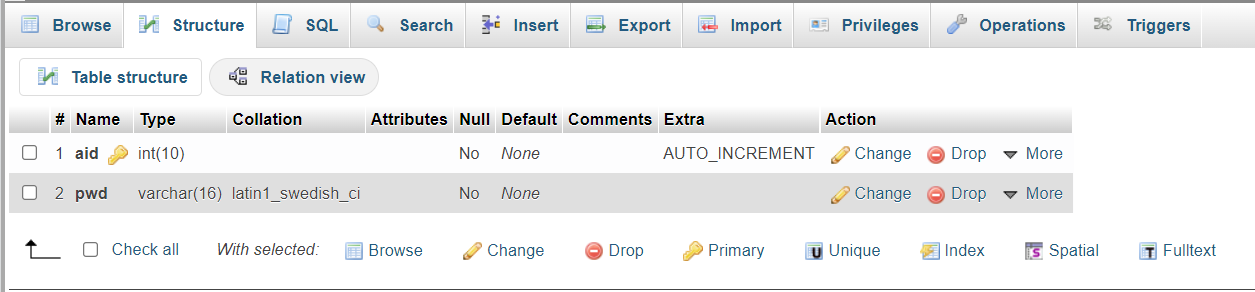


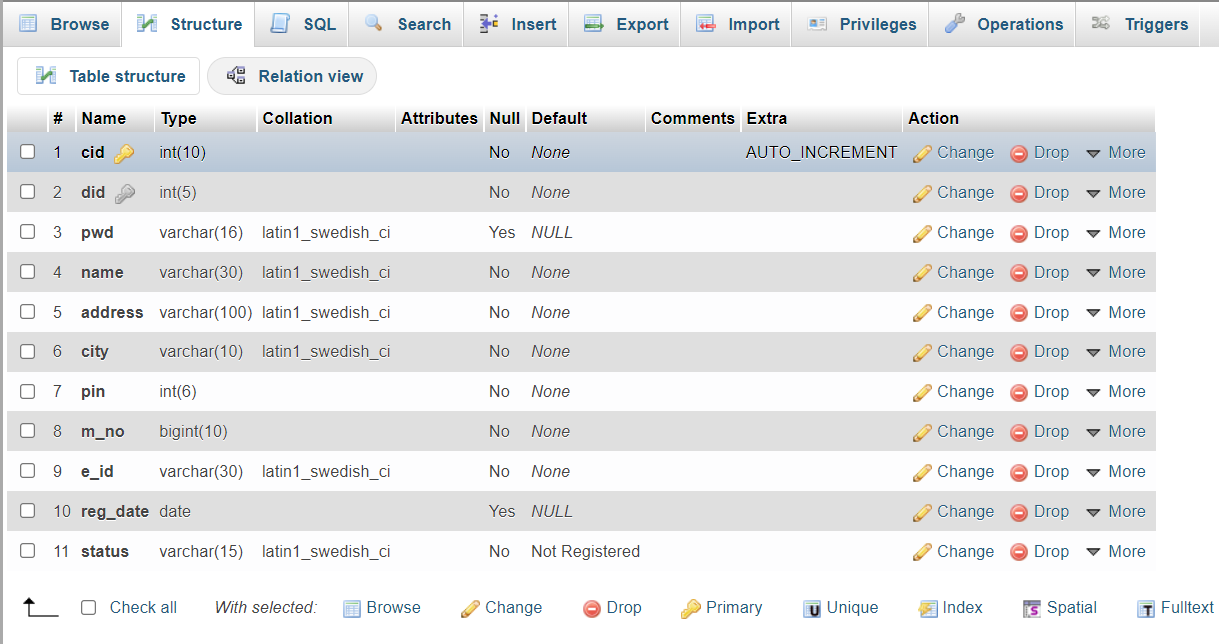




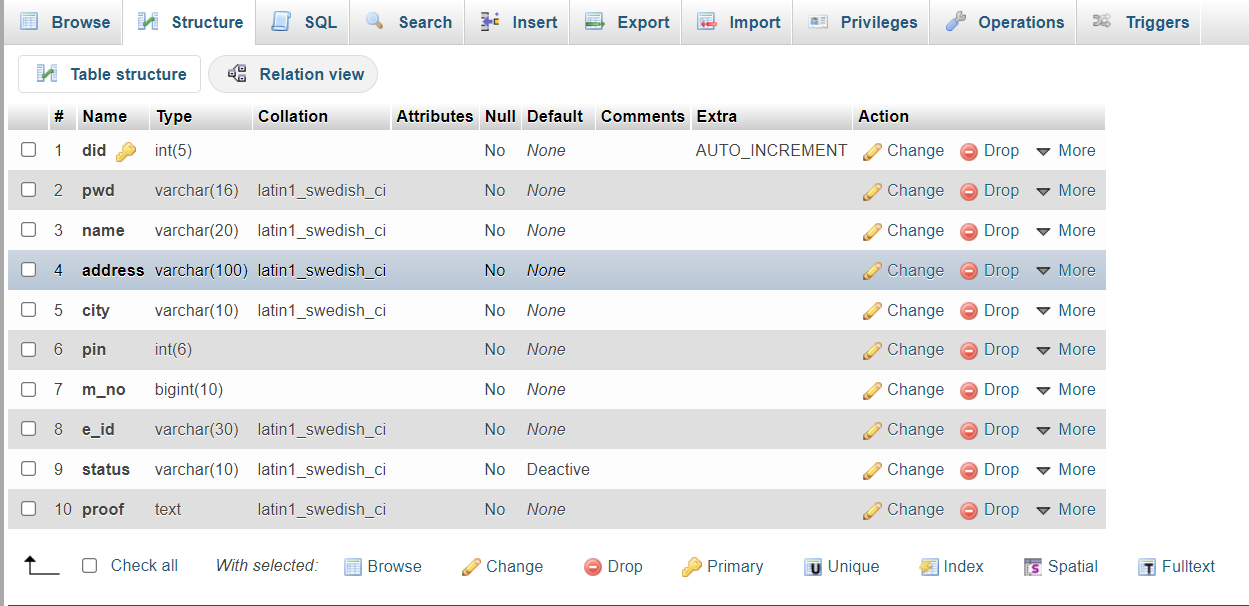


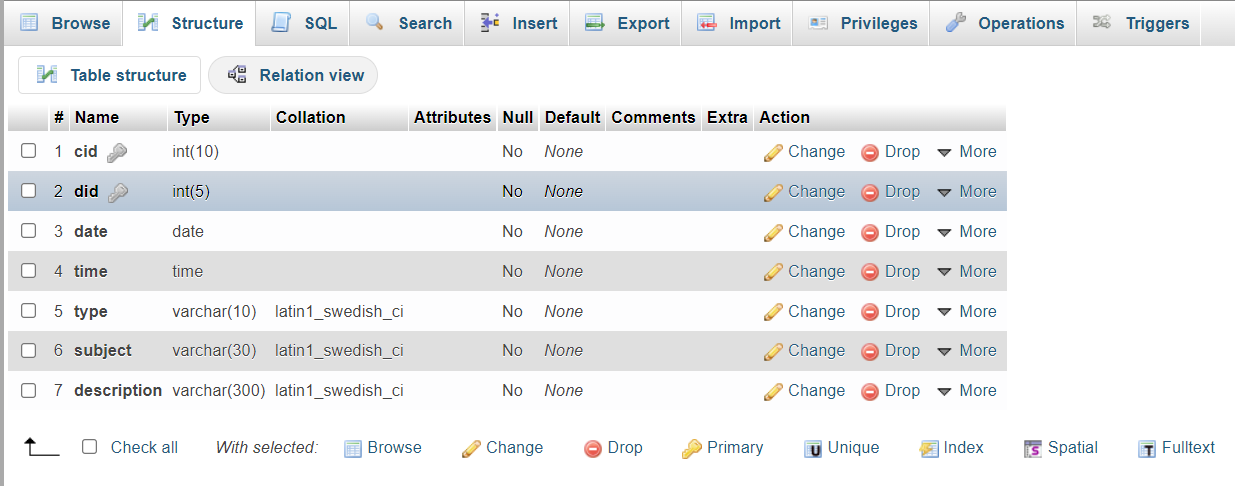
**5.2 Table Structure:**

**ADMIN**

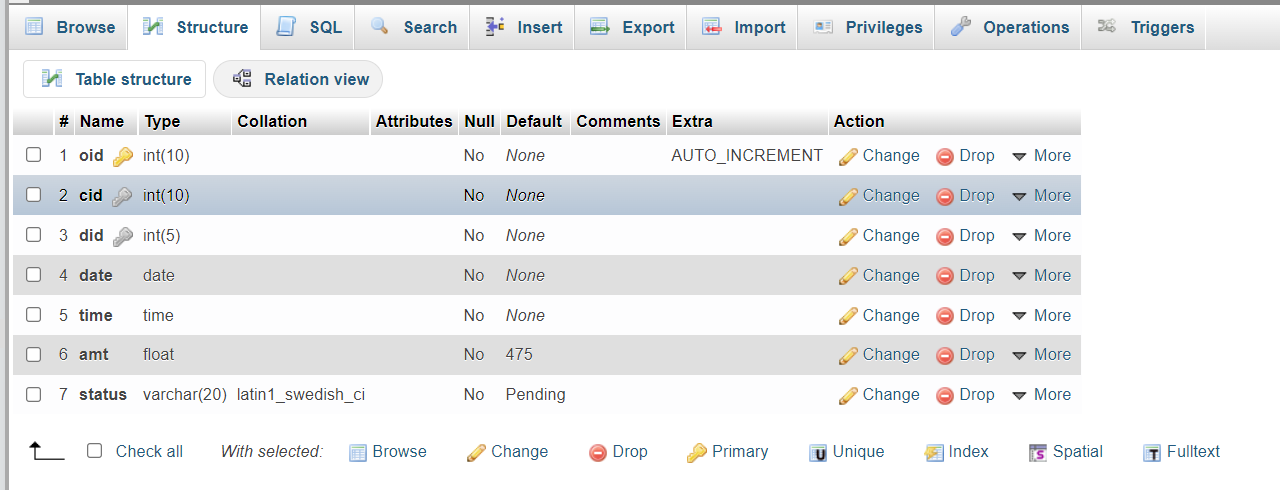
**CONSUMER DETAIL**

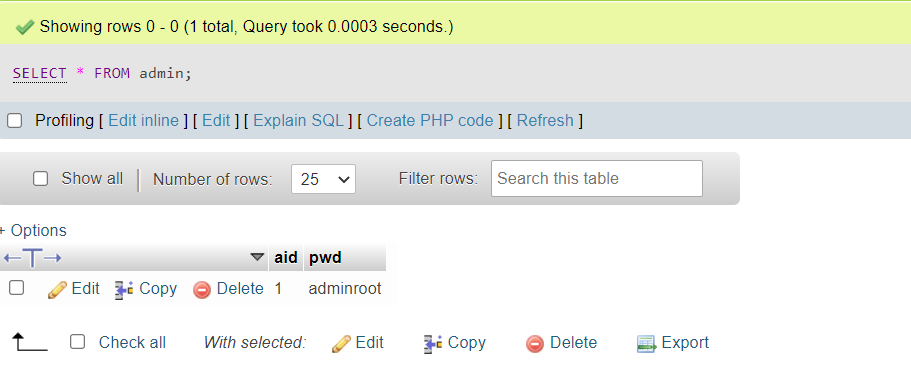
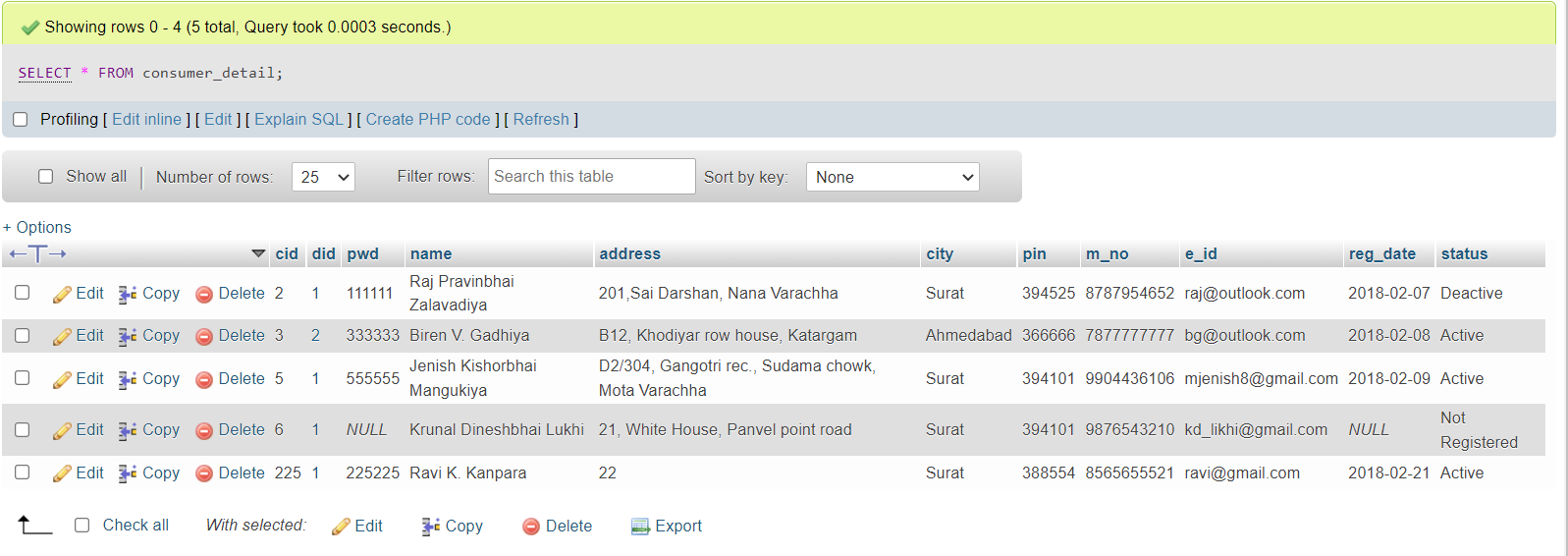
**DISTRIBUTOR DETAIL**

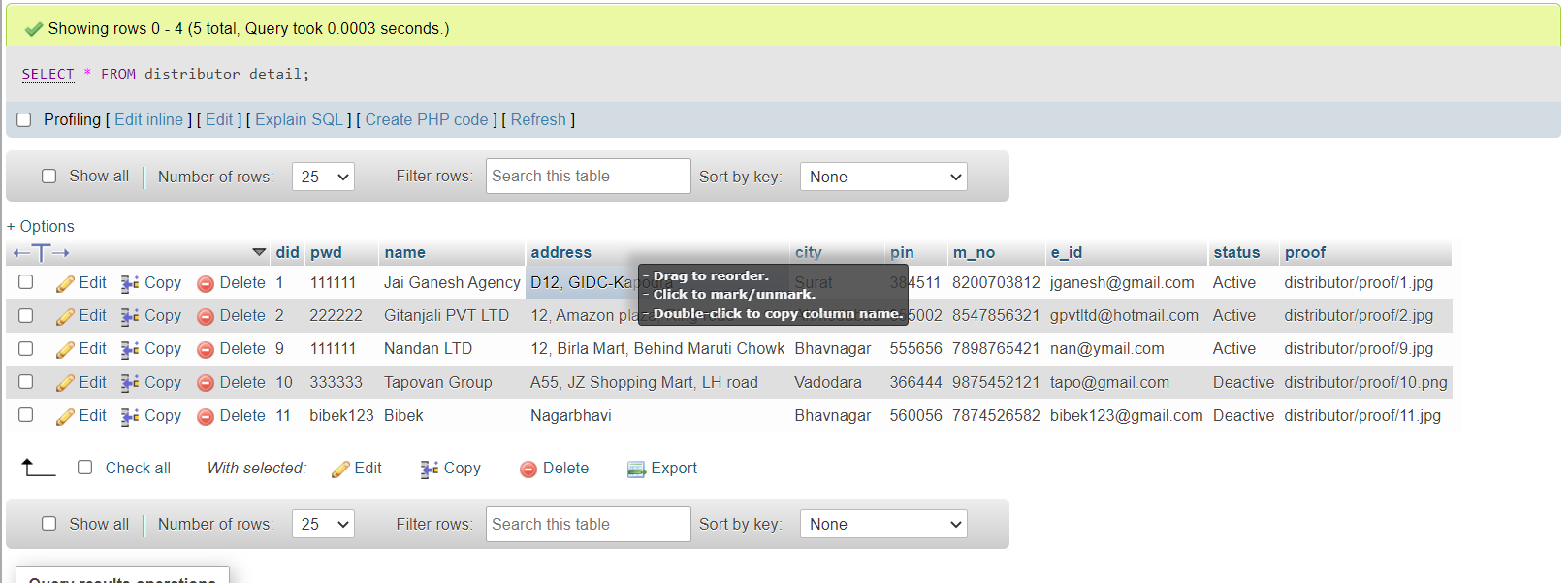
****

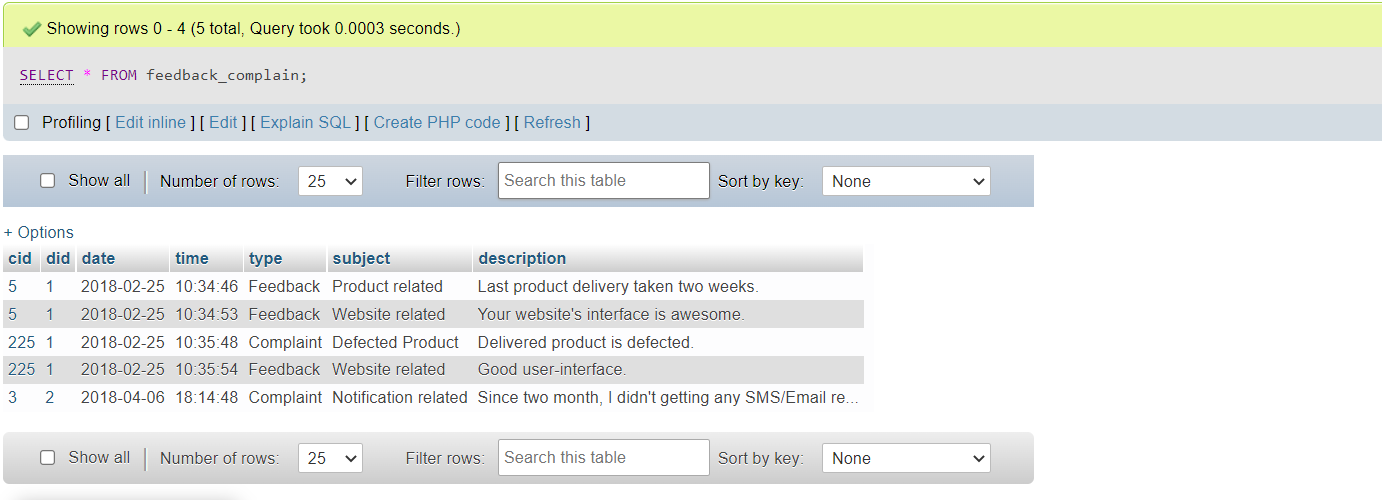
**FEEDBACK COMPLAIN**

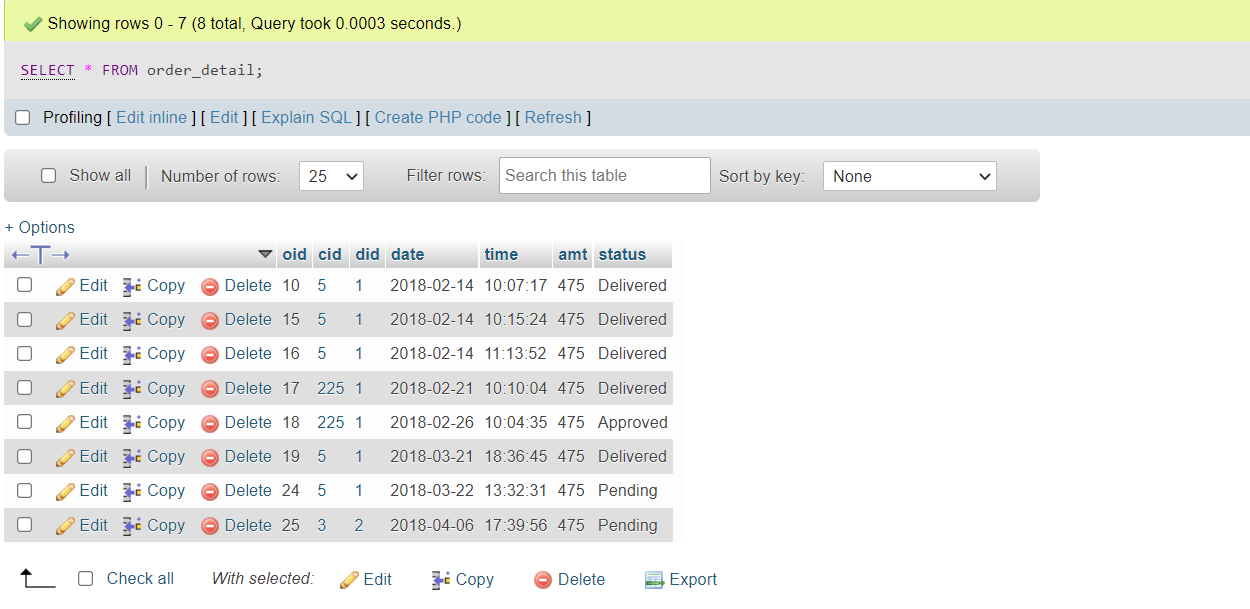
**ORDER DETAIL**



**5.3 Table Result:**







**QUERIES**

1. Retrieve name of the consumer who has registered under Jai Ganesh Agency.

SELECT name

FROM consumer\_detail

WHERE did = 1;



1. List the name, mobile no. and email of the customer who lives in surat.

SELECT name,m\_no,e\_id,city

FROM consumer\_detail

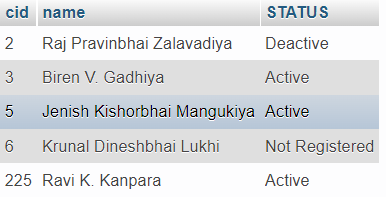
WHERE city = 'surat';



3.Select the name of consumer with their status.

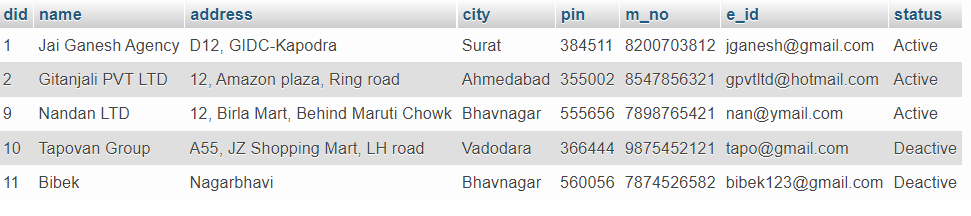
SELECT cid,name,status

FROM consumer\_detail;



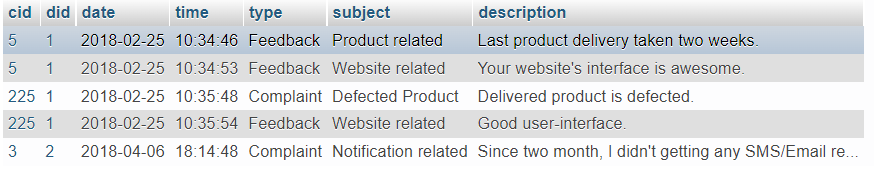
4.List the name of Distributor with their full address and status.

SELECT did,name,address,city,pin,m\_no,e\_id,status

FROM distributor\_detail;

5.List out all the feedback complain registered.

SELECT \* FROM `feedback\_complain`



6.Retrieve the name of customers whose order has been delivered.

SELECT c.name

FROM consumer\_detail c, order\_detail o

WHERE o.cid = c.cid and o.status = 'Delivered';



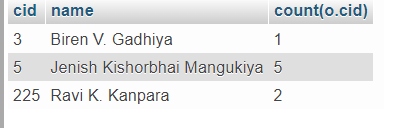
7. List out the name of customer with their no. of order.

SELECT o.cid,c.name,count(o.cid)

FROM order\_detail o, consumer\_detail c

WHERE o.cid = c.cid

GROUP BY o.cid



**CONCLUSION**

The project Gas Agency Management System is for computerizing the working in a gas agency. The software takes care of all the requirements of an average gas agency and is capable to provide easy and efficiency storage of information related to overall transaction of all the branches of a agency. It generates transaction report of a customer and cylinders provided to the customers.

There is always a scope of betterment and the candidate system is not against this perception. At present the system satisfy most of the functions of Gas Agency management system. This project is especially designed for Gas Agency to take appropriate steps to improve the working standard and documentation through computerization.

• Generating the important report on day wise or month wise.

• Storing large amount of data for future point of view.

• Reducing manual efforts for maintaining the system.

• Assures security and validity.

• Provision for enhancement without disturbing the developed modules.

**BIBLIOGRAPHY**

Few of the books and websites that were instrumental in helping us to complete this project are as mentioned below:

**BOOKS**

* Elmasri and Navathe, “Fundamental of Database System", Addison-Wesley,5th Edition 2007
* Database System Concepts, 6th Edition by Silberschatz, Korth and Sudarshan

**URL’s**

<https://www.wikipedia.com>

<https://w3schools.com>

<https://www.draw.io>

<https://www.php.net.com/manual>

<https://www.researchgate.net>

https://www.ecomputernotes.com