



INSTITUTE FOR ADVANCED
COMPUTING AND
SOFTWARE DEVELOPMENT
AKURDI, PUNE

Documentation On

**“Slot Booking for Charging an EV at
Charging Station”**

PG-DAC SEPT 2021

Submitted By:

GROUP NO 54

Joshi Sushant Prakashrao-210941220080

More Sangram Rajkumar-210941220108

Mrs. Shilpa Pawale
Project Guide

Mr. Kashinath Patil
Project Guide

Index

Sr. No.	Content	Page No.
1	Introduction	3
2	Architectural Design	3
3	Purpose of Document	4
4	Project Overview	4
5	Assumptions and Dependencies	5
6	User Classes and Characteristics	6
7	User Documentation	7
8	External Interface Requirements	7
9	Non-Functional Requirements	8
10	High level Design	9
	10.1 ER Diagram	9
	10.2 Page Navigation Diagram	10
	10.3 Data Flow Diagram	12
11	Low level Design	15
	11.1 Database Design	16
	11.2 Stored Procedure	18
12	Details of page Navigation	18

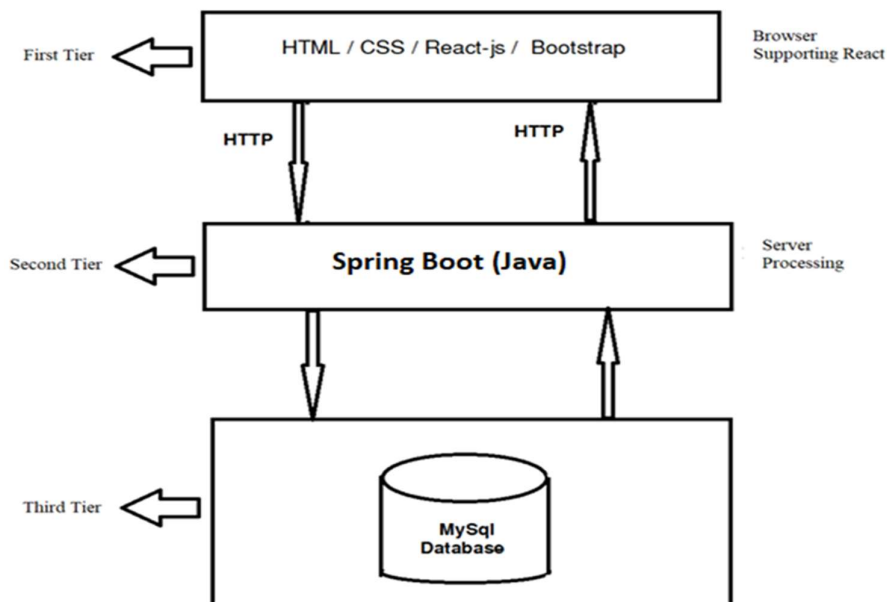
Introduction

This document is meant for the description of the structure and the database which we are using in this project. This document gives brief description about Architecture of the system, E-R diagram of the system and the table descriptions, the page navigation diagrams and the detail description for the page navigation.

Electric vehicles are more efficient, and that combined with the electricity cost means that charging an electric vehicle is cheaper than filling petrol or diesel for your travel requirements. Using renewable energy sources can make the use of electric vehicles more eco-friendly.

Architecture Design

Following diagram shows the details of the e-faming system architecture



This System consist of three tiers as listed below:-

First Tier:

This tier is used for user interface and it is called as client tier. In this tier we are using React-JS because of it provides better interactivity, easier navigation, compact and also helps us to create single page application . The use of JSX script facilities us for the client side validation. That's why in first tier we are using the JSX script. We are using HTML, CSS , BOOTSTRAP for the presentation purpose.

Second Tier:

Server Process:

This application requires database access to fetch various information. It will also need to update the database. To sync my application front-end, We are using Spring boot as a middle tier. The choice of Spring boot was made because we want it to scale properly and it is the fastest way we can achieve our desired result. This middle tier will serve the front-end with all the required data that it needs. It will also receive data from the front-end and update the back-end database.

Third Tier:

MySQL Database :

We are using MySQL as our database management system. The reason we choose this as our data base because it is a relational data base system. It is the open source easy to use we can build and interact with MySQL by using only a few simple SQL statements. MySQL follows the working of a client/server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they can query data, save changes, etc. MySQL supports multi-threading that makes it easily scalable. It can handle almost any amount of data, up to as much as 50 million rows or more. The default file size limit is about 4 GB. However, we can increase this number to a theoretical limit of 8 TB of data MySQL uses Triggers, Stored procedures, and views that allow the developer to give higher productivity.

Purpose of Document

The purpose of this document is to enlist the software requirements and specifications for the '**Slot Booking for Charging an EV at Charging Station**' which will be a new added feature to current charging stations India.

Intended Audience

This document is intended for developers, users, testers and project managers for the purpose of understanding the design of systems in terms of different perspectives. Further, this document contains functionalities and characteristics of the system along with the working environment. It also includes other information related to systems such as external interface requirements, features and other non - functional requirements.

Project Overview

The Slot Booking for an EV is a facility is used by Indian citizens who owns Electrical Vehicle, to find out the nearest charging station and book charging slots in advance. Currently if people want to charge their vehicles at charging stations, they have to wait till charging slots to get empty and the time required may vary from type of e-vehicles. It is estimated that the total potential demand for the full range of electric vehicles in India (mild hybrids to full electric

vehicles) will be in the range of 5–7 million units in new vehicle sales by 2020.

As number of e-vehicles are increasing day by day. Crowd at charging stations is required to manage, because e-vehicle takes time to charge and if crowd is present at charging stations, then it will take more time to get a slot for charging. So, with the help of our project, we trying to minimize the waiting time required at charging stations to get slot by booking slot in advance and also helping user to find out the nearest charging station and we are also providing facility for premium customer not to wait for charging the battery so customer can directly exchange their uncharged battery with already charged battery at charging stations.

Assumptions and Dependencies

Assumptions

1. There is an active internet connection with the system.
2. The system has an internet browser installed.
3. Users know the English language.

Dependencies

Users should remember patterns for authentication while confirming his or her authenticity in the whole authentication process.

Purpose and Scope

Swift Charge is a platform where we are giving facilities to users like finding nearest charging stations, in advance charging slot booking and for premium customers directly battery exchange to exclude the charging time means premium customer don't need to wait for charging they can simply exchange their battery with charging stations and can continue their journey. The purpose of this project is to minimize the waiting time of customers, manage the crowd at charging stations, increase the productivity of charging stations, adding the new charging stations in the current network of charging stations.

There is a wide scope for Ev's in four-wheeler market, auto rikshaws, goods vehicle, bus as well as two wheelers. All of this market is waiting for a change. We can integrate this web app with restaurants and malls which have charging points in their parking, we can also make the mobile application for this Swift Charge. In future demand for e-vehicles will increase more rapidly so need to manage this crowd of e-vehicles at charging stations this can be achieved by increasing number of charging stations.

Product Functions

- The user will be able to find the nearest charging stations.
- User can able to book charging slot in advance.
- Users can be able to exchange existing battery at their end with the already charged battery at charging stations this will also helps to minimize crowd and customer also don't need to wait till battery get charged.
- We are also providing facility to add the valid charging station in current network of charging stations so this will also minimize crowd at charging stations in future.
- Analysis of charging stations such as for how much time charging station is productive and for how much time charging station is idle.

User Classes and Characteristics

1. Admin Class:

- o The user class contains attributes pertaining to users such as Role, Name, email, password, Phone number etc.

2. Station Class:

- o The Station class contains the details about the stations like Name, Station registration number, Station location, Number of slots station contains.

3. Customer Class:

- o The Customer class contains the details about the customer such as Customer Id, Vehicle type, vehicle number, etc.

MVC design patterns will be followed with DAO and Service layers for the user module.

Operating Environment

●Hardware Platform:

- o The hardware infrastructure requirements for Swift Charge will be addressed in the Infrastructure and Deployment Architecture document.

●Software Platform:

- o Front-end: - React Bootstrap.
- o Back-end: - Spring boot, MySQL.

●Supported Tools:

- o Spring Tool Suite, Vs Code.

Design and Implementation Constraints

- User interface is only in English. No other language option is available.
- Users can log-in only with his assigned user-name and password based on Roll.
- Limited to HTTP.

User Documentation

User documentation mainly comprises the *resource menu* on the website which will contain manuals particularly for this project. It will give all the minute details about the authentication process, if any user has any query about any module or functionality, one can refer to it and see how to proceed for the authentication process. This report is the complete documentation of the authentication process. It gives complete details about the authentication process, its functionality, users, software used, hardware requirement, environment and so on.

External Interface Requirements

User Interfaces

The main element is webpages using Reactjs. The pages will use JavaScript for the UI design and validation. The website will be responsive. bootstrap will be used for the CSS and styling of the webpages. The webpages will be rendered on the server-side using Servlets.

Hardware Interfaces

The user-end system can be either a smartphone or a computer device. The application supports all major web browsers. The web browser should be JavaScript enabled.

Software Interfaces

In software interfaces, Spring and Hibernate are the back-end technology used along with MySQL Database. The front-end technologies include Reactjs, CSS, Bootstrap, JavaScript and. Data will be communicated between these interfaces accordingly.

Communications Interfaces

The main communication interface for interacting with the System will be the web Browser using HTTP request.

System Features

- User Registration.
- User Login.
- Forgot pattern.
- Role base login.
- Slot booking.
- Register Charging Station.

Other Non-functional Requirements**Performance Requirements**

The system should store all the database records of each user properly and the application should be available for use 24*7 through the server. The system should authenticate and register users properly in a secured manner. Also, the application should be user friendly with a proper user interface which makes it easy for the user to understand. All the options should be present in properly accessible places for user convenience.

Safety Requirements

All login Ids and passwords of the users should be protected for privacy using whatever constraints required in the database or the application. User records are to be backed up securely across database servers. In case database is hacked by someone and data is deleted a backup server should be present for such purpose.

Security Requirements

All passwords of the users should be protected for privacy using whatever constraints required in the database or the application. The database should be protected from attacks and unauthorized access. The interface should be protected from attacks. All passwords should be encrypted and stored.

Software Quality Attributes**Availability**

- The system should run on a variety of operating systems that support the JavaScript language. The system should run on a variety of hardware.

Accessibility

- The software will be accessible to admin, Customers and Station owner.

Compatibility

- The software will be compatible with multiple platforms.

Durability

- The software will be tested for working with multiple users.

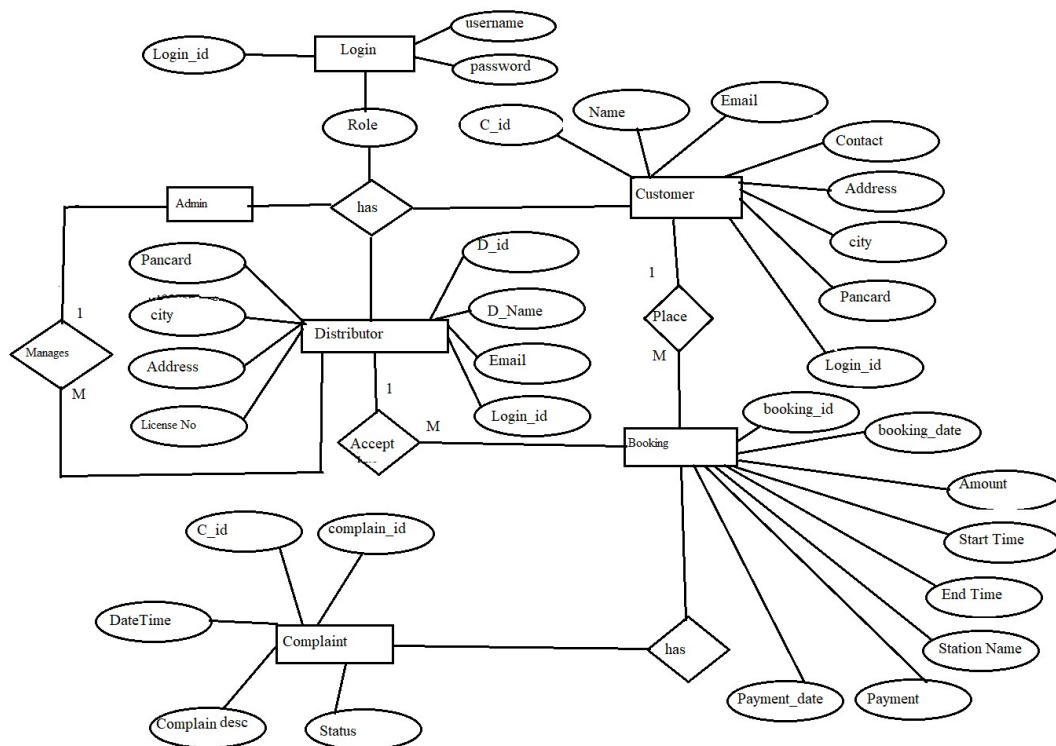
Effectiveness

- The software will be made to handle operations effectively.

Maintainability

The system should be easy to maintain. There should be a clear separation between the interface and the business logic code. There should be a clear separation between the data access objects that map the database and the business logic code.

High Level Design: E-R Diagram:



Above E-R Diagram shows that database of Slot booking for an Ev at Charging station consist of following entities:

- **Customer :**

This entity contains the Customer id , Name , Email-id , Password Contact Number ,Consumer Number , Address , Pan Card .

- **Distributer :**

This entity contains the Email id , Distributer name ,Distributer id , License -number , Location .

- **Message**

This entity contains the Message Id, Date-Time, Receiver-Status, Sender-Status, Receiver Id, Sender Id, Message, Subject attributes.

- **Admin**

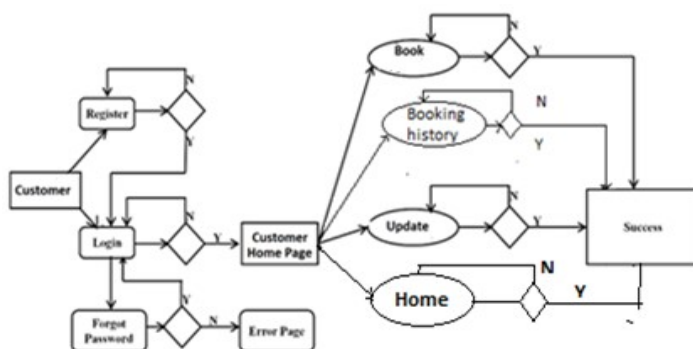
This entity contains the Admin id, Password .

- **Payment**

This entity contains the Transection id, Withdraw, Fid, Sender Account Number , Receiver Account Number , Amount, Date Time, Seller Status, Buyer Status attributes.

Page Navigation Diagram:

- **Customer :**Following diagram explains the page navigation for the Customer module:



- Distributer**

Following diagram explains the page navigation for the farmer module:

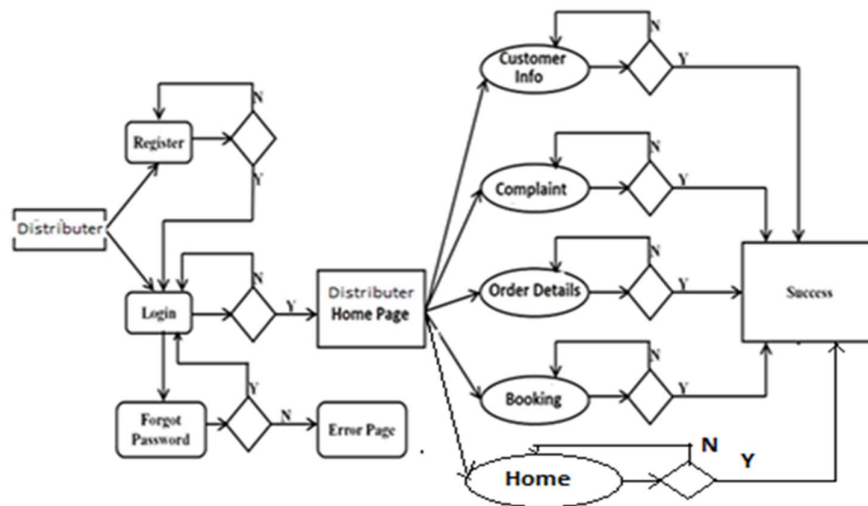


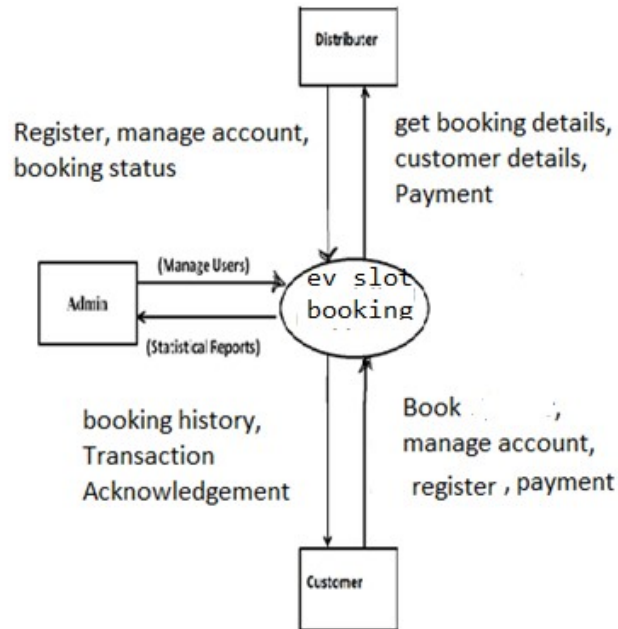
Fig. Page Navigation for Distributer

- Admin**

Following diagram explains the page navigation for the Admin module:



Fig. Page Navigation For Admin

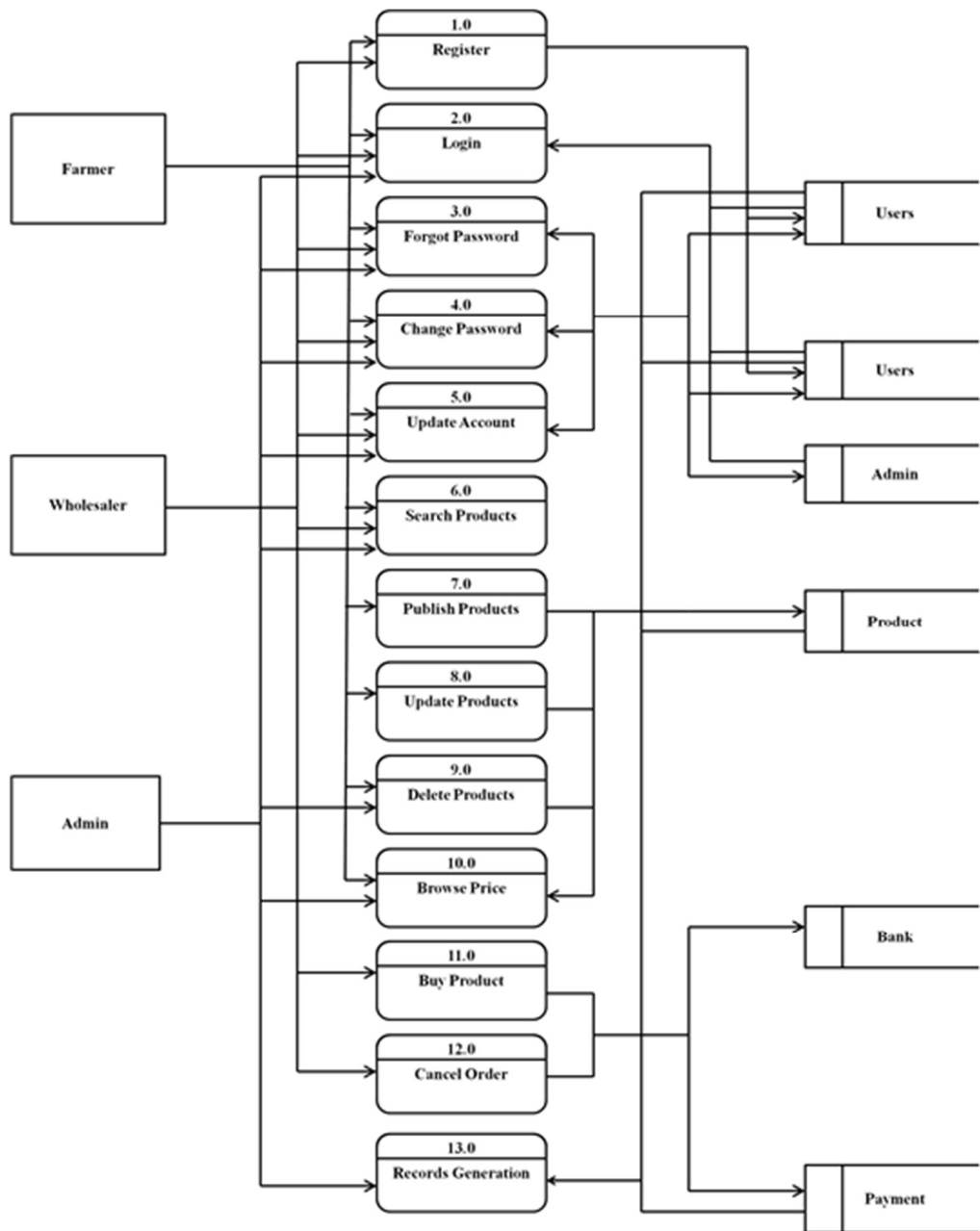
Data Flow Diagram:*Fig 0-Level DFD***•In 0-Level DFD, there are three Entities:**

•Customer

•Distributer

•Admin

- 1-Level DFD:



In 1-Level DFD,**Costumer Entity having following processes:**

- Registration/new connection
- Select distributor
- Sign in
- Profile management
- Update profile
- Booking management
- View all booking
- Add a booking
- Cancel booking
- Logout

Distributor Entity having following processes:

- Registration
- Sign in
- Distributor Profile Management
- update profile
- Consumer Management
- approve consumer (with sending e-mail notification)
- view all the consumers
- Logout

Admin Entity having following processes:

- Sign In
- Profile Management
- Update profile
- Distributor Management
- Approve distributor (with sending e-mail notification)

- View distributor list
 - Suspend/block distributor
 - Unblock distributor
 - Consumer Management
 - View all consumers
 - Suspend consumers
 - Orders Management
 - View all orders of distributor
 - View pending orders of distributor
 - Feedback Management
 - feedbacks from consumers
 - Orders Management
 - Payment management
 - Payment status of consumers
 - Logout
 - **Low Level Design:**
 - **Database Design:**
- 1] **Login Table**

Field	Type	Null	Key	Default	Description
login_id	Integer(5)	No	Primary key	-	Login_id
user_name	Varchar(15)	No		-	User name
password	Varchar(8)	No		-	Password
role	Varchar(15)	No		-	As a Admin/Customer/ Distributor

2] Table_Customer_detail

Field	Type	Null	Key	Default	Description
cid	Integer	No	Primary key	Null	User ID
name	Varchar(45)	No		-	Name
email	Varchar(45)	No		-	Email ID Of Consumer
contact	Integer(10)	No		-	Contact No. Of Consumer
address	Varchar(100)	No		-	Permanent Address
city	Varchar(15)	No		-	City
login_id	Integer(5)	No	Foreign key	-	Login id of customer
pancard	Varchar(8)	No	-	-	Pancard for proof customer
consumer_number	Integer(17)	No		-	Consumer Number Given By Agencies
d_id	Integer	No	Foreign key	-	Distributor id to link with Order id and customer id

3] Table_Distributor_detail

Field	Type	Null	Key	Default	Description
d_id	Integer	No	Primary key	-	Distributor ID
d_name	Varchar(45)	No		-	Distributor Name
email	Varchar(45)	No		-	Email ID Of Distributer
login_id	Integer(5)	No	Foreign key	-	Login id of distributor
license_Number	Integer	No		-	License Number of Distributer
pancard	Varchar(8)	No	-	-	Pancard for proof Distributer
address	Varchar(100)	No		-	Permanent Address
city	Varchar(15)	No		-	City

4] Tbl_booking_table

Field	Type	Null	Key	Default	Description
booking_id	Integer	No	Primary key	-	Booking_ID
booking_date	Date	No		-	Booking_Date

amount	Double	No		-	Amount To Be Transfer
Start time	Time	No		-	Charging start time
End time	Time	No		Null	Charging end time
Vehicle no	Varchar (15)	No		-	Vehicle No
Charging station Name	Varchar (15)	No	Foreign key		Distributor Name
payment_mode	Varchar (15)	No			COD/UPI/NetBanking
transection_id	Integer	No	-	-	Transection_id
payment_date	Date	No	-	-	Payment date
status	Varchar (15)	No	-	-	Completed/Cancel/pending

5] Tbl_Complain

Field	Type	Null	Key	Default	Description
complain_id	Integer	No	Primary key	-	Comaplain
cid	Integer	No	Foreign key	-	Customer ID
date_time	Date_time	No	-	-	Date And Time Of Message
status	Varchar(200)	No	-		Pending/received /cancel
complain_description	Varchar(200)	No	-	-	Complain Description

- **Stored Procedure:**
- **CreateCustomer:**

This stored procedure is used for creating new user

Input parameters:

Cid, Name, Email, Password, Contact, Address, , Consumer No , Pan card .

- **SendMessage:**

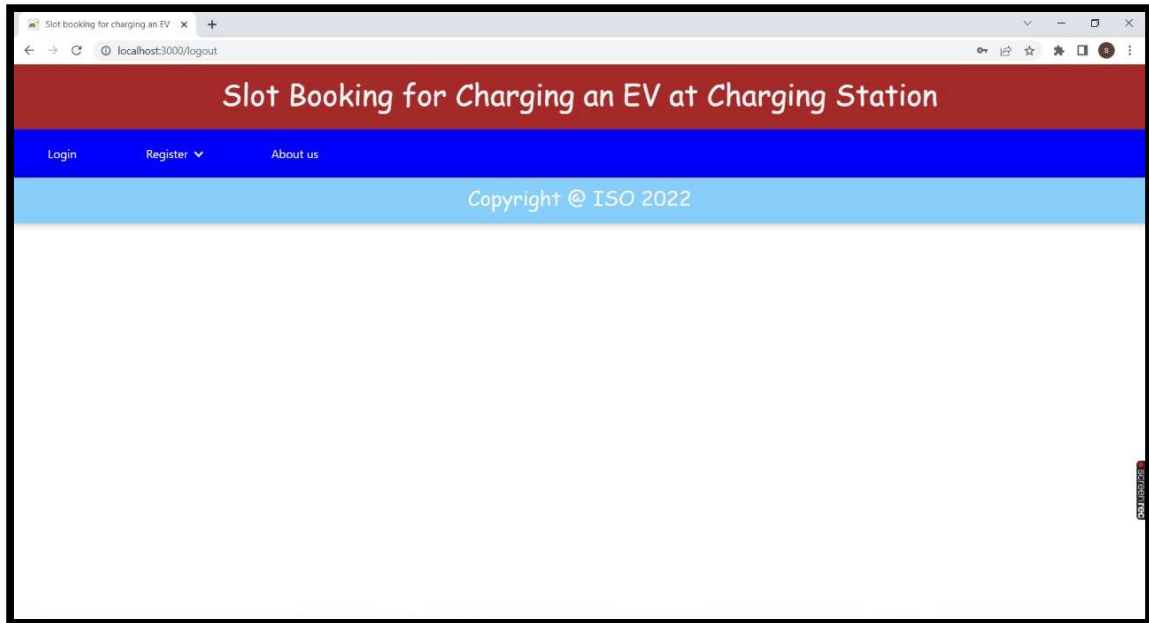
This stored procedure is used for inserting message details

Input parameters:

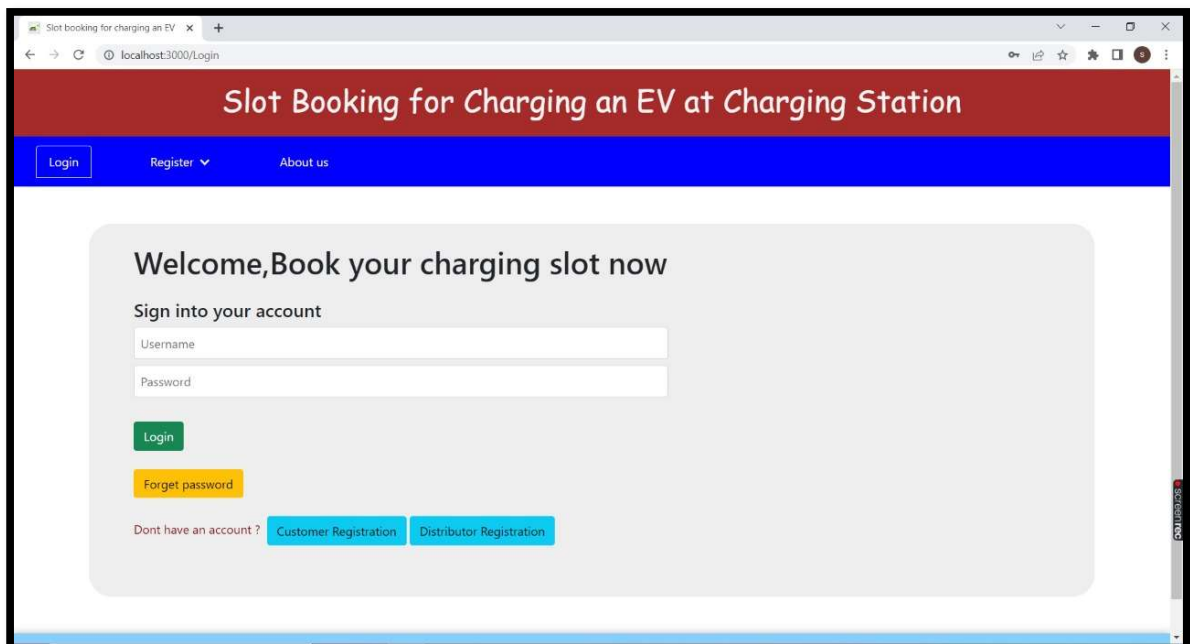
ReceiverId, SenderId, Message, Subject

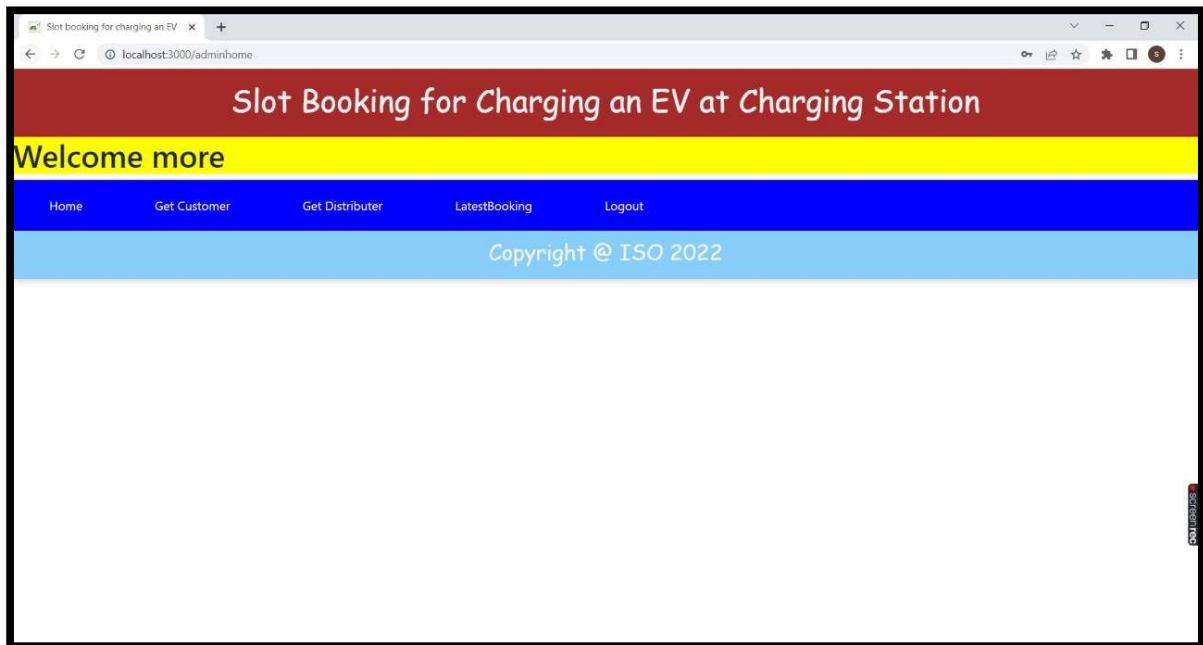
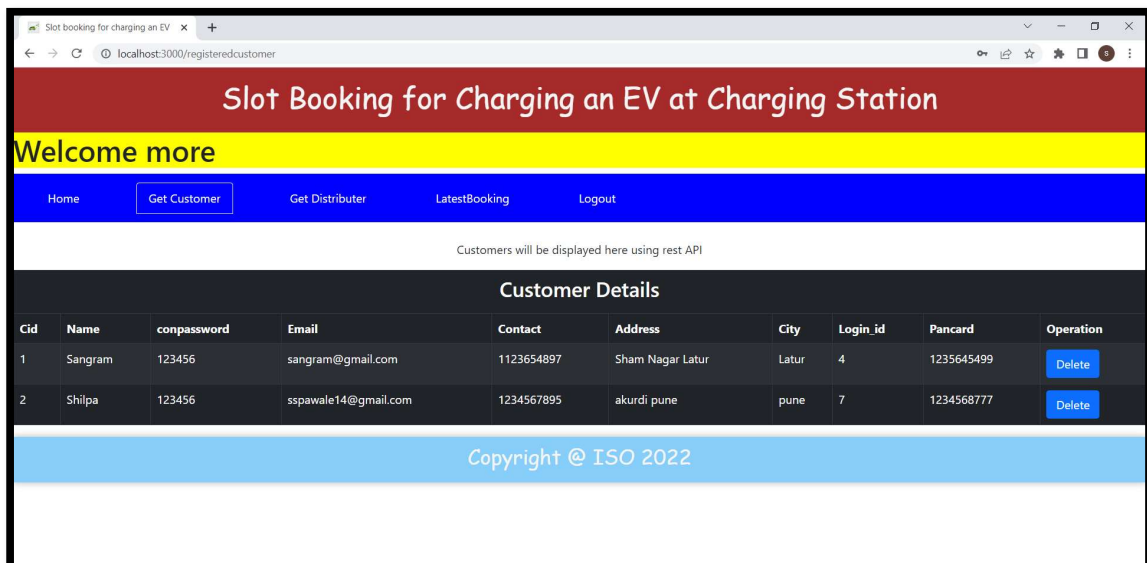
- **Details Page Navigation:**
- Home Page:
- Following snapshot shows the home page of the Slot Booking for an EV at charging station.

Home page:



Login Page:



Admin page/Admin Dashboard**Admin -> get customer page:**

Admin -> get distributor page

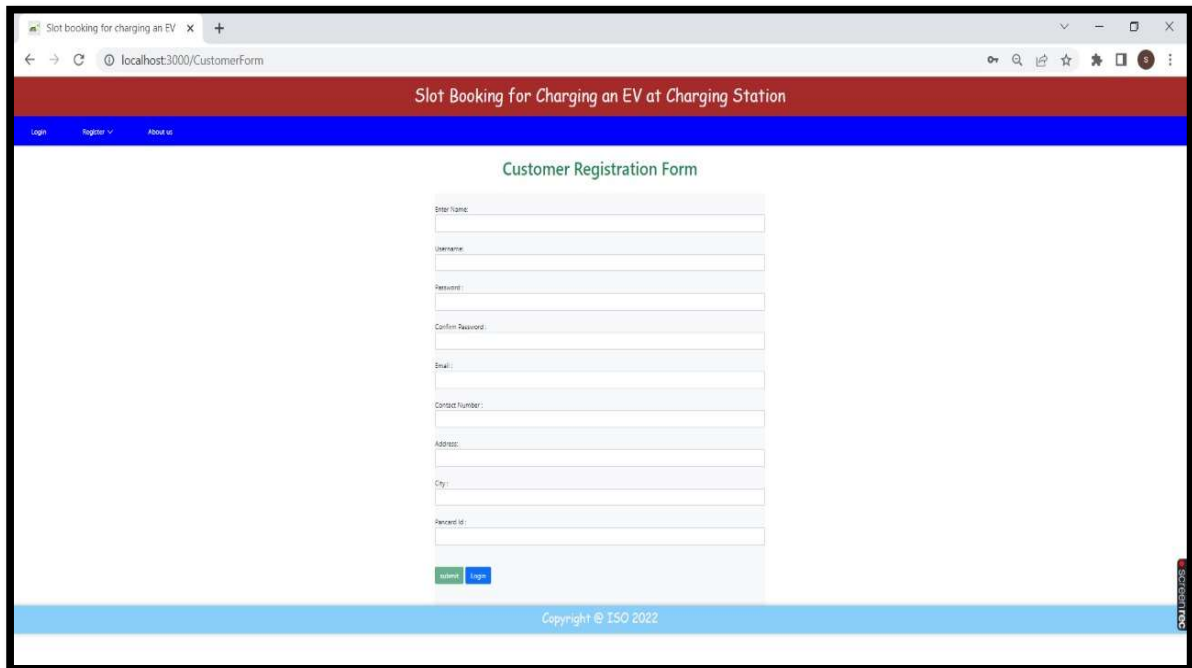
The screenshot shows a web browser window with the URL `localhost:3000/registereddistributor`. The page has a red header with the title "Slot Booking for Charging an EV at Charging Station" and a yellow banner with "Welcome more". A blue navigation bar contains links: Home, Get Customer, Get Distributer (active), LatestBooking, and Logout. Below the navigation bar, a message states "Customers will be displayed here using rest API". A table titled "Distributor Details" lists three distributors with columns for ID, Name, compassword, Email, Contact, Login_id, Pancard, Address, City, and Operation. Each row has a "Delete" button. The footer shows "Copyright @ ISO 2022".

Distributor ID	Name	compassword	Email	Contact	Login_id	Pancard	Address	City	Operation
1	Spark	123456	spark@gmail.com	1234567891	2	1236549875	Ganjgolai Latur	Latur	Delete
2	Electric	123456	electric@gmail.com	1234569875	3	1236548975	Gandhi Chowk Latur	Latur	Delete
3	Mobility	123456	mobility@gmail.com	1236549874	8	9875641235	akurdi pune	pune	Delete

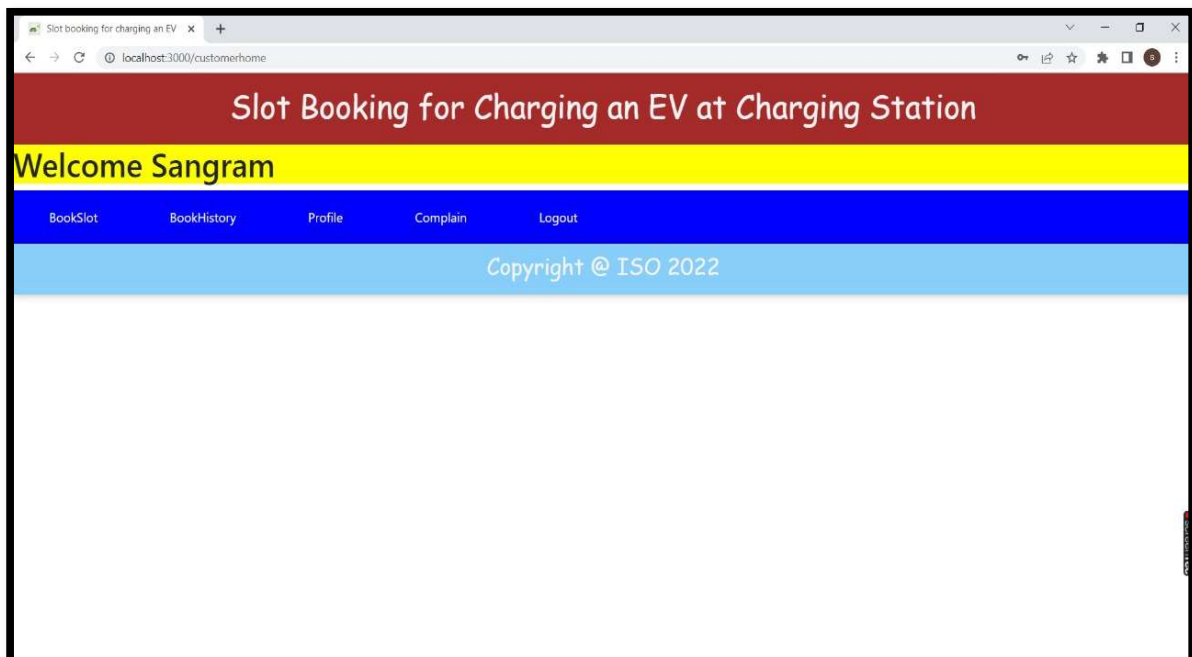
Admin -> latest booking page

The screenshot shows a web browser window with the URL `localhost:3000/LatestBooking`. The page has a red header with the title "Slot Booking for Charging an EV at Charging Station" and a yellow banner with "Welcome more". A blue navigation bar contains links: Home, Get Customer, Get Distributer, LatestBooking (active), and Logout. Below the navigation bar, a message states "Customers will be displayed here using rest API". A table titled "Booking Details" lists three bookings with columns for bid, book_date, amount, payment_mode, and Operation. Each row has an "Edit" button. The footer shows "Copyright @ ISO 2022".

bid	book_date	amount	payment_mode	Operation
1	2022-04-10T00:00:00.000+00:00	100	Cash	Edit
2	2022-04-10T00:00:00.000+00:00	100	Cash	Edit
3	2022-04-13T00:00:00.000+00:00	100	Cash	Edit

New Customer registration form:

The screenshot shows a web browser window with the address bar displaying "localhost:3000/CustomerForm". The page title is "Slot Booking for Charging an EV at Charging Station". The page has a red header bar with the title, a blue navigation bar with links for "Login", "Register", and "About us", and a light blue footer bar with "Copyright © ISO 2022". The main content area is titled "Customer Registration Form" and contains a registration form with the following fields: "Enter Name", "Username", "Password", "Confirm Password", "Email", "Contact Number", "Address", "City", and "Pincode". There are "Submit" and "Login" buttons at the bottom of the form.

New Distributer registration form

The screenshot shows a web browser window with the address bar displaying "localhost:3000/customerhome". The page title is "Slot Booking for Charging an EV at Charging Station". The page has a red header bar with the title, a yellow banner with "Welcome Sangram", a blue navigation bar with links for "BookSlot", "BookHistory", "Profile", "Complain", and "Logout", and a light blue footer bar with "Copyright © ISO 2022". The main content area is empty.

Customer -> book slot page

Slot booking for charging an EV x +

localhost:3000/BookSlot

Slot Booking for Charging an EV at Charging Station

Welcome Sangram

BookSlot BookHistory Profile Complain Logout

Available Charging Stations will be displayed here

Available Charging Station Details

Distributor ID	Name	Address	City
1	Spark	Sangam Nagar	Latur
2	Batrick	Sangam Chowk Latur	Latur

Book Slot

Booking Date: 2022-04-10T10:00:00.000+00:00

Enter Charging Station Name:

Amount: 100

Charging start time: 10:06

Charging end time: 11:06

Vehicle No: MH24M3264

Payment Mode: Cash

book

Copyright © ISO 2022

Customer -> book history page

Slot booking for charging an EV x +

localhost:3000/BookHistory

Slot Booking for Charging an EV at Charging Station

Welcome Sangram

BookSlot BookHistory Profile Complain Logout

Booking Details

bid	book_date	Charging Station Name	amount	Charging start time	Charging end time	Vehicle No.	payment_mode	Operation
1	2022-04-10T00:00:00.000+00:00	Spark	100	10:06	11:06	MH24M3264	Cash	
2	2022-04-10T00:00:00.000+00:00	spark	100	13:15	14:15	MH24X1975	Cash	
3	2022-04-13T00:00:00.000+00:00	Spark	100	12:49	13:49	Mh24X1234	Cash	

Copyright © ISO 2022

Customer -> profile page

Slot Booking for Charging an EV at Charging Station

Welcome Sangram

BookSlot BookHistory **Profile** Complain Logout

Customers will be displayed here

Customer Details

Cid	Name	conpassword	Email	Contact	Address	City	Login_id	Pancard	Operation
1	Sangram	123456	sangram@gmail.com	1123654897	Sham Nagar Latur	Latur	4	1235645499	<input type="checkbox"/>
2	Shilpa	123456	sspawale14@gmail.com	1234567895	akurdi pune	pune	7	1234568777	<input type="checkbox"/>

Copyright @ ISO 2022

Customer -> complaint page

Slot Booking for Charging an EV at Charging Station

Welcome Sangram

BookSlot BookHistory Profile **Complain** Logout

Register Complain

Customer ID:

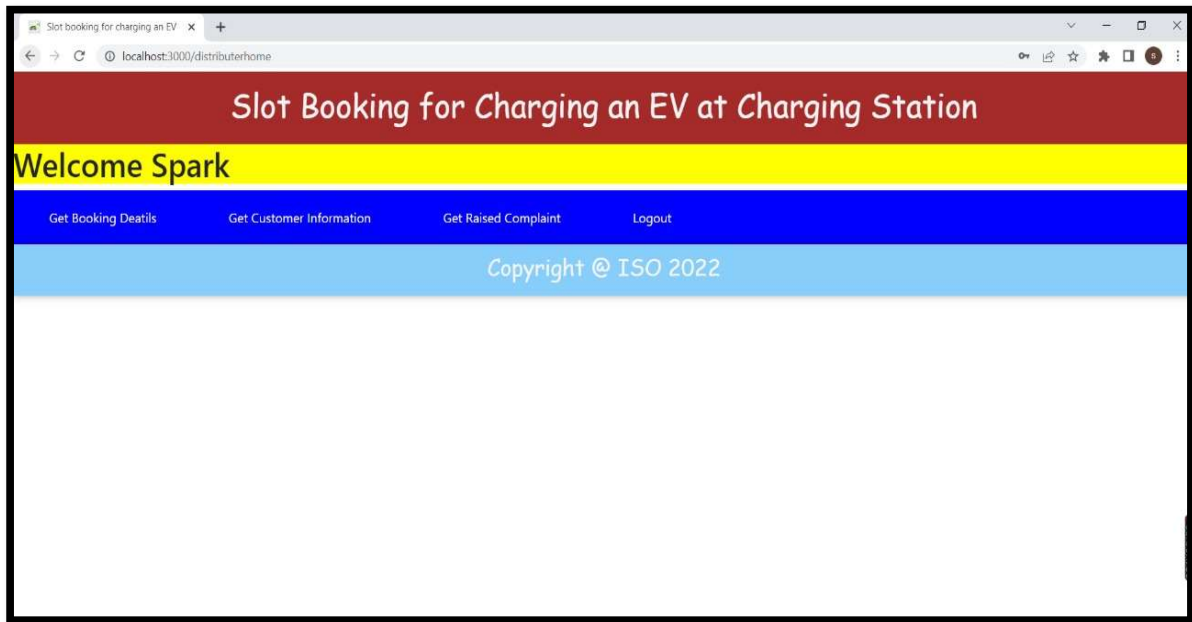
Complaint Date:
dd-mm-yyyy

Complain Description :

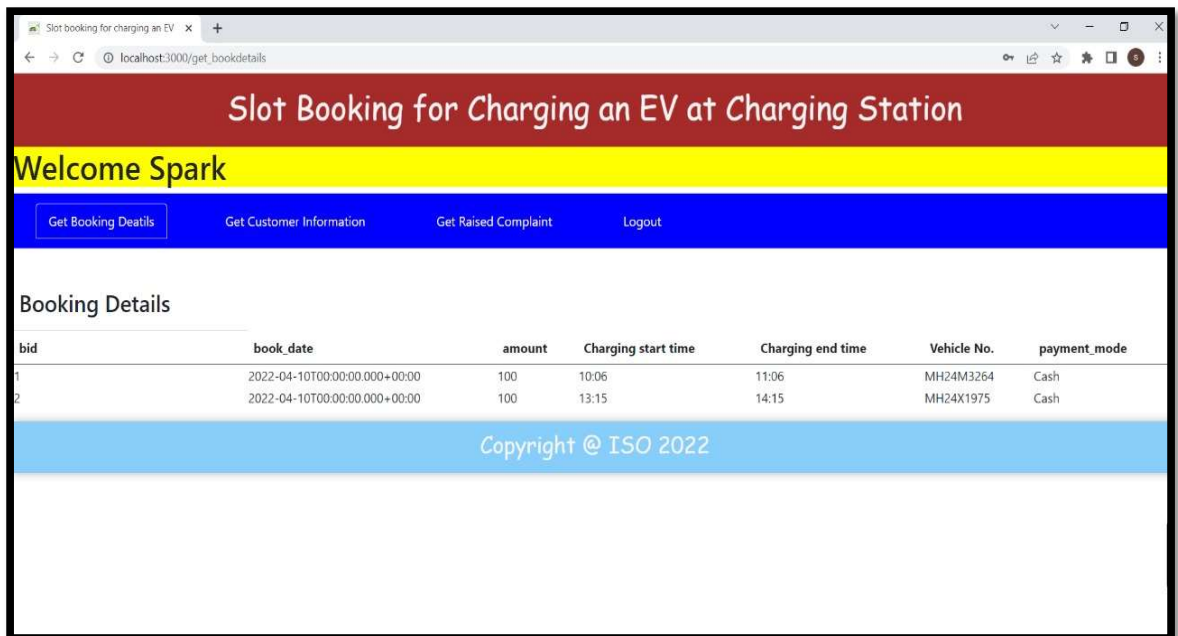
Complaint

Copyright @ ISO 2022

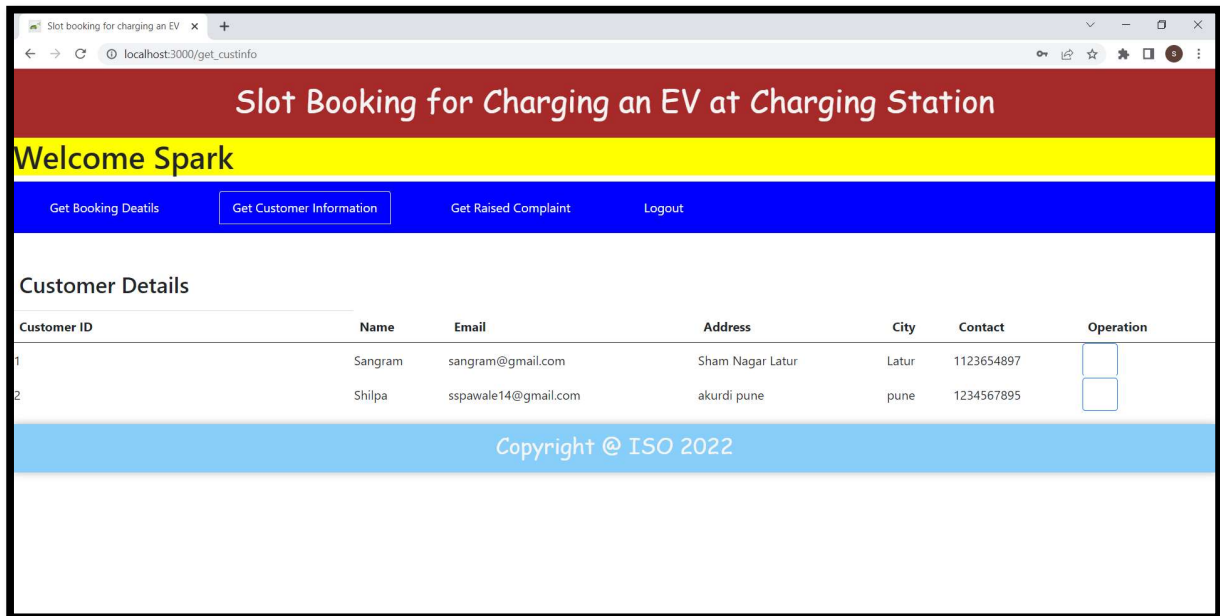
Distributer -> home/dashboard page



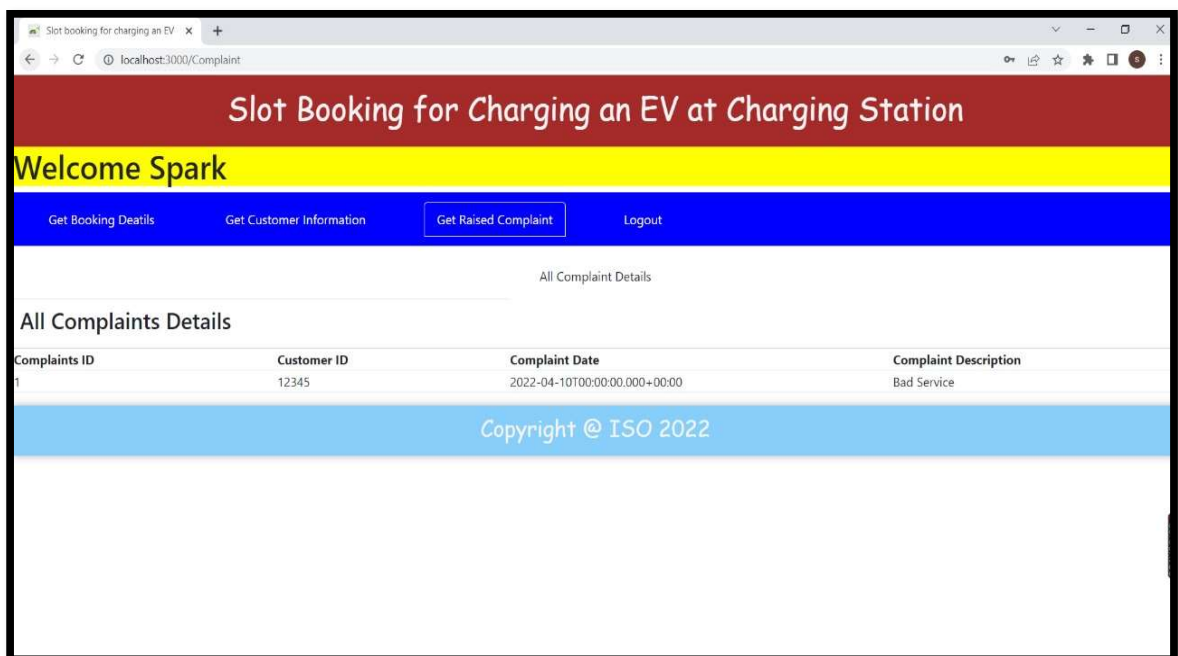
Distributer -> get booking details



Distributer -> get customer information page



Distributer -> get raised complaint page



REFERENCES

<http://www.google.com>

<http://www.webdevelopersjournal.com/>

<http://www.w3.org>

<http://www.wikipedia.org>

<http://reactjs.org>

<http://getbootstrap.com>