**ELECTRICOX**

## INTRODUCTION

### OBJECTIVE OF THE PROJECT

This is a web application for finding electric vehicle recharge nearest bunk. This paper examines various issues related to electric vehicle supply equipment (EVSE) or charging stations related to. The proposed system of EV charging web application to provide:

1. EV owner the convenience of locating charging station on a list.

2. Vacancy of charging slots.

3. Also provide booking facility for the charging slot.

Thus, it helps to increase the life of batteries and ensure smooth journeys long distance.

## SYSTEM ANALYSIS

### INTRODUCTION

Analysis is the process of breaking the problem into the successively manageable parts for individual study, system analysis is the study of various operations that has to be done to solve the problem .one aspect of the system analysis is defining the boundaries of the system and determining whether or not be proposed system should consider other related systems. One of the main meanings of the feasibility is possibility checking of the different criteria for success is included in feasibility study section .these criteria’s are cost ,time ,efficiency etc.…all these factors play an important role in achieving objective of the system .that means the system should be such it gives optimum performance at minimum cost ,time requirements .these system contributes to the overall objectives of the organization .the system be implemented using current technology and within given cost and schedule constrains .the system is integrated with systems which are already in place. It is a general term that refers to a structural process for identifying and solving problems .in a computer-based transformation system silk is the structured approach. Analysis implies the process of breaching something down in to its parts so that the whole may be understood. The definition of system analysis, but also that of synthesis, which is the process of putting parts together to form a new whole.

### IDENTIFICATION OF NEED

System analysis is the reduction of the entire system by studying various operations and their relationships with the system and the requirements of bit successor. A system can be defined as an orderly grouping of interdependent components linked together according to plan to achieve a specific objective.

The idea of the system has become most practical and necessary in conceptualizing the interrelationships and integrations of operations especially when using computers. Organizing consists of several interrelated and interacting components. Analysis is the detailed study of various operations performed by the system and their relations within and outside the system. During analysis, data are connected on the available files, decision points and is handled by the present system.

### EXISTING SYSTEM

The existing system is the manual system. Need to be converted into automated system.  In People should push the vehicles or get help to reach nearest electric recharge station.  In the above method time and manual work is done by owners of the vehicle. For some aged people or medically ill people it will get even hard. To get electric to fill generators people need to go to an electric recharge bunk station.

### PROPOSED SYSTEM

If you have launched an EV charging station, then taking your business online can be the best business decision you will ever make. Developing an EV charging station finder project will be your best investment if you are jumping into the EV charging station business. It will increase the visibility of your business and will generate more revenue. An EV charging station finder project allows the car owners to not wait in the long queue at the charging station. The users can book the spot in advance so that they don’t need to wait at the time of need. Thus, there will be no hassle for users to charge their electric vehicles.

### FEASIBILITY STUDY

Feasibility is conducted to identify the best system that meets all requirements. It is both necessary and important to evaluate the feasibility of a project at the earliest possible time. feasibility study includes an identification description, an evaluation of proposed system and selection of the best system for the job. During the system is to be carried out. this is to ensure that the proposed system is not A burden to the shop. The feasibility study should be relatively cheap and quick. the results should inform the decision of whether to go ahead with a more detailed analysis, some understanding of the major requirements for the system is essential. Four key considerations involved in the feasibility analysis are

* Operational feasibility
* Technical feasibility
* Economical feasibility
* Behavioral feasibility

### OPERATIONAL FEASIBILITY

The purpose of the operational feasibility is to determine whether the new system will be used if it is developed and implemented and whether there will be resistance from users that will undermine the possible application benefits .the aspect of study is to check the level of acceptance of the system by the user .this includes the process of training the user to use the system efficiently .the user must not feel threatened by the system, instead must accept it as a necessity .the level of acceptance by the user solely depends on the methods that are employed to educate the user about the system and to make him familiar with it .his level of confidence must be raised so that he is also able to make some constructive . The proposed system is an upgrade version of the current systems new fields have been implemented according to the user need, hence it ensures all the aspects.

The proposed system is very much user-friendly and the system is easily understood by simple training and it is operationally feasible to use by any users.

### TECHNICAL FEASIBILITY

A study of function, performance and constraints may improve the ability to create an acceptable system, technical feasibility is frequently the most difficult area to achieve at the stage of product Engineering process. technical feasibility is deals with the hardware as well as software requirements. the scope was whether the work for the project is done with the current equipment’s and the existing system technology has to be examined in the feasibility study. The result was found to be true. This feasibility is carried out to check the technical requirements of the system. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system. this is related to the technicality of the project. This evaluation determines whether the technology needed for proposed system is available or not .it deals with hardware as well as software requirements. that is, type of hardware, software and the methods required for running the system are analyzed .so it can be used in any windows so computer. This system requires very low system resources and it will work in almost all configurations. In the existing system all functions are doing manually. So, if they get this designed software, the problems can be avoided and thus the system will run smoothly.

In the proposed system, data can be easily stored and the managed using database management system software. the reports and the results for various queries can be generated easily. our proposed system is technically feasible to use by any users.

### ECONOMICAL FEASIBILITY

Economic feasibility is the most frequently used method for evaluating the effectiveness of the candidate system .it is very essential because the main goal of the proposed system is to have economically better result along with increased efficiency. A cost evaluation is weighed against the ultimate income or product. Economic justification is generally the bottom-line consideration that includes cost benefit analysis, long term corporate income strategies, and cost of resources needed for development and potential market growth. When compared to the advantage obtained from implementing the system its cost is affordable. Proposed system was developed with available resources. Since cost input for the software is almost nil the output of the software is always a profit. Hence software is economically feasible.

### BEHAVIOURAL FEASIBILITY

People are inherently resistant to change and computer is known for facilitating the changes .an estimate should be made of how strongly the user; staff reacts towards the development of the computerized system. In the existing system more manpower is required and time factor is more. The more manpower for managing many files for dynamic data replication and more time for search through these files is needed. but in the proposed system, both manpower and time factors are reduced and also unnecessary burden is reduced. thus, the remaining people are made to engage in some other important work. also, there is no need to wait in case of downloading the data for the users therefore, the system is behaviorally feasible.

### SYSTEM SPECIFI CATION

### PHP

PHP is a server-side scripting language designed for Web development but also used as a general-purpose programming language. PHP is now installed on more than 20 million Web sites and 1 million Web servers. While PHP originally stood for Personal Home Page, it is now said to stand for PHP: Hypertext Preprocessor, a recursive acronym.

PHP code is interpreted by a Web server with a PHP processor module which generates the resulting Web page: PHP commands can be embedded directly into an HTML source document rather than calling an external file to process death includes free and open-source libraries with the core build. PHP is a fundamentally Internet-aware system with modules built in for accessing File Transfer Protocol (FTP) servers, many database servers, embedded SQL libraries such as embedded PostgreSQL, MySQL, Microsoft SQL Server and SQLite, LDAP servers, and others. PHP is commonly used as the P in this bundle alongside Linux, Apache and MySQL, although the P may also refer to Python, Perl, or some mix of the three. Similar packages are also available for Windows and OS X, then called WAMP and MAMP, with the first letter standing for the respective operating system.

PHP is a server-side scripting language designed specifically for the Web. Within an HTML page, you can embed PHP code that will be executed each time the page is visited. Your PHP code is interpreted at the web server and generates HTML or other output that the visitor will see.

PHP was conceived in 1994 and was originally the work of one man, RasmusLerdorf. It was adopted by other talented people and has gone through four major rewrites to bring us the broad, mature product we see today. PHP is an Open-Source project, which means you have access to the source code and can use, alter, and redistribute it all without charge. PHP originally stood for Personal Home Page but was changed in line with the GNU recursive naming convention (GNU = Gnu’s Not Unix) and now stands for PHP Hypertext Preprocessor.

### ABOUT THE BACK END

### Hypertext Mark-up Language (HTML)

It is the standard [mark-up language](https://en.wikipedia.org/wiki/Markup_language) for documents designed to be displayed in a [web browser.](https://en.wikipedia.org/wiki/Web_browser) It can be assisted by technologies such as [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [scripting languages](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript.](https://en.wikipedia.org/wiki/JavaScript)

[Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a web page [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

[HTML elements](https://en.wikipedia.org/wiki/HTML_element) are the building blocks of HTML pages. With HTML constructs, [images](https://en.wikipedia.org/wiki/HTML_element#Images_and_objects) and other objects such as [interactive forms](https://en.wikipedia.org/wiki/Fieldset) may be embedded into the rendered page. HTML provides a means to create [structured documents](https://en.wikipedia.org/wiki/Structured_document) by denoting structural [semantics](https://en.wikipedia.org/wiki/Semantics) for text such as headings, paragraphs, lists, [links](https://en.wikipedia.org/wiki/Hyperlink), quotes and other items. HTML elements are delineated by *tags*, written using [angle](https://en.wikipedia.org/wiki/Bracket#Angle_brackets) [brackets.](https://en.wikipedia.org/wiki/Bracket#Angle_brackets)

Tags such as <imp /> and <input /> directly introduce content into the page. Other tags such as <p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags but use them to interpret the content of the page.

HTML can embed programs written in a [scripting language](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript), which affects the behavior and content of web pages. Inclusion of CSS defines the look and layout of content.

### JavaScript

It is a lightweight, interpreted, object-oriented language with [first-class](https://en.wikipedia.org/wiki/First-class_function) [functions](https://en.wikipedia.org/wiki/First-class_function), and is best known as the scripting language for Web pages, but it's [used](https://en.wikipedia.org/wiki/JavaScript#Uses_outside_Web_pages) [in many non-browser environments](https://en.wikipedia.org/wiki/JavaScript#Uses_outside_Web_pages) as well. It is a [prototype-based](https://en.wikipedia.org/wiki/Prototype-based_programming), multi- paradigm scripting language that is dynamic, and supports object-oriented, imperative, and functional programming styles.

JavaScript runs on the client side of the web, which can be used to design / program how the web pages behave on the occurrence of an event. JavaScript is an easy to learn and powerful scripting language, widely used for controlling web page behavior.

JavaScript can function as both a [procedural](https://en.wikipedia.org/wiki/Procedural_programming) and an [object oriented language](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Introduction_to_Object-Oriented_JavaScript). Objects are created programmatically in JavaScript, by attaching methods and Properties to otherwise empty objects at run time, as opposed to the syntactic class definitions common in compiled languages like C++ and Java. Once an object has been constructed it can be used as a blueprint (or prototype) for creating similar objects.

### Cascading Style Sheets (CSS)

It is a [style sheet](https://developer.mozilla.org/en-US/docs/DOM/stylesheet) language used to describe the presentation of a document written in [HTML](https://developer.mozilla.org/en-US/docs/Web/HTML) or [XML](https://developer.mozilla.org/en-US/docs/XML_introduction) (including XML dialects such as [SVG](https://developer.mozilla.org/en-US/docs/Web/SVG), [Math](https://developer.mozilla.org/en-US/docs/Web/MathML) or [XHTML](https://developer.mozilla.org/en-US/docs/Glossary/XHTML)). CSS describes how elements should be rendered on screen, on paper, in speech, or on other media.

### MYSQL

MySQL is an Oracle-backed open source relational database management system ([RDBMS](https://searchdatamanagement.techtarget.com/definition/RDBMS-relational-database-management-system)) based on Structured Query Language ([SQL](https://searchsqlserver.techtarget.com/definition/SQL)). MySQL runs on virtually all platforms, including [Linux](https://searchdatacenter.techtarget.com/definition/Linux-operating-system), [UNIX](https://searchdatacenter.techtarget.com/definition/Unix) and [Windows](https://searchwindowsserver.techtarget.com/definition/Windows). Although it can be used in a wide range of applications, MySQL is most often associated with web applications and online publishing.

MySQL is an important component of an open-source enterprise stack calle[d WAMP.](https://whatis.techtarget.com/definition/LAMP-Linux-Apache-MySQL-PHP) WAMP is a web development platform that uses Linux as the operating system, [Apache](https://whatis.techtarget.com/definition/Apache) as the web server, and MySQL as the relational database management system and [PHP](https://whatis.techtarget.com/definition/PHP-Hypertext-Preprocessor) as the object-oriented scripting language. (Sometimes [Perl](https://whatis.techtarget.com/definition/Perl) or [Python](https://whatis.techtarget.com/definition/Python) is used instead of PHP.)

Originally conceived by the Swedish company MySQL AB, MySQL was acquired by Sun Microsystems in 2008 and then by Oracle when it bought Sun in 2010. Developers can use MySQL under the GNU General Public License ([GPL](https://searchdatacenter.techtarget.com/definition/GNU-General-Public-License-GNU-GPL-or-simply-GPL)), but enterprises must obtain a commercial license from Oracle. Today, MySQL is the RDBMS behind many of the top websites in the world and countless corporate and consumer-facing web-based applications, including Facebook, Twitter and YouTube.

* MySQL is the database management system, or a database server.

### How MySQL works

MySQL is based on a [client-server](https://searchnetworking.techtarget.com/definition/client-server) model. The core of MySQL is MySQL server, which handles all of the database instructions (or commands). MySQL server is available as a separate program for use in a client-server networked environment and as a library that can be embedded (or linked) into separate applications.

MySQL operates along with several utility programs which support the administration of MySQL databases. Commands are sent to MySQL Server via the MySQL client, which is installed on a computer.

MySQL was originally developed to handle large databases quickly. Although MySQL is typically installed on only one machine, it is able to send the database to multiple locations, as users are able to access it via different MySQL client interfaces. These interfaces send SQL statements to the server and then display the results.

### MySQL Features

* + **Relational Database Management System (RDBMS):** MySQL is a relational database management system
  + **Easy to use** MySQL is easy to use. You have to get only the basic knowledge of SQL. You can build and interact with MySQL with only a few simple SQL statements.
  + **It is secure:** MySQL consist of a solid data security layer that protects sensitive data from intruders. Passwords are encrypted in MySQL.
  + **Client/ Server Architecture:** MySQL follows a client /server architecture. There is a database server (MySQL) and arbitrarily many clients (application programs), which communicate with the server; that is, they query data, save changes, etc.
  + **Free to download** MySQL is free to use and you can download it from MySQL official website.
  + **It is scalable:** MySQL 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, you can increase this number to a theoretical limit of 8 TB of data.
  + **Compatible on many operating systems:** MySQL is compatible to run on many operating systems, like Novell NetWare, Windows\* Linux\*, many varieties of UNIX\* (such as Sun\* Solaris\*, AIX, and DEC\* UNIX), OS/2, FreeBSD\*, and others. MySQL also provides a facility that the clients can run on the same computer as the server or on another computer (communication via a local network or the Internet).
  + **Allows roll-back:** MySQL allows transactions to be rolled back, commit and crash recovery.
  + **High Performance:** MySQL is faster, more reliable and cheaper because of its unique storage engine architecture.
  + **High Flexibility:** MySQL supports a large number of embedded applications which makes MySQL very flexible.
  + **High Productivity:** MySQL uses Triggers, Stored procedures and views which allows the developer to give a higher productivity.

### Core MySQL features

MySQL enables data to be stored and accessed across multiple storage engines, including Inorb, [CSV](https://searchwindowsserver.techtarget.com/definition/CSV-Cluster-Shared-Volumes), and NDB. MySQL is also capable of replicating data and partitioning tables for better performance and durability. MySQL users aren't required to learn new commands; they can access their data using standard SQL commands.

Before 2016, the main difference between MySQL and SQL was that the former could be used on multiple platforms, whereas the latter could only be used on Windows. Microsoft has since expanded SQL to support Linux, a change which went into effect in 2017. When MySQL is installed via Linux, its package management system requires custom configuration to adjust security and optimization settings.

MySQL also allows users to choose the most effective storage engine for any given table, as the program can utilize multiple storage engines for individual tables. One of MySQL's engines is Inorb. Inorb was designed for [high](https://searchdatacenter.techtarget.com/definition/high-availability) [availability](https://searchdatacenter.techtarget.com/definition/high-availability). Because of this, it is not as quick as other engines. SQL uses its own storage system, but it does maintain multiple safeguards against loss of data. Both systems are able to run in clusters for high availability.

SQL Server offers a wide variety of data analysis and reporting tools. SQL Server Reporting Services is the most popular one and is available as a free download. There are similar analysis tools for MySQL available from third-party software companies**,** such as Crystal Reports XI and Actuate BIRT.

### PHPMYADMIN

PhpMyAdmin is a (web application) client for MySQL. MySQL is server where your commands get executed and returns you data, it manages all about data while PhpMyAdmin is a web Application, with user friendly, easy to use GUI makes it easy to handle database, which is difficult to use on command line. phpMyAdmin is the web application written primarily in PHP. It’s used for managing MySQL database.

To be more specific, here is the detailed definition:

MySQL is the world’s most popular open-source database. With its proven performance, reliability, and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube, and all five of the top five websites. Additionally, it is an extremely popular choice as embedded database, distributed by thousands of ISVs and OEMs.

PhpMyAdmin is a free and open-source administration tool for MySQL and MariaDB. As a portable web application written primarily in PHP, it is one of the most popular MySQL administration tools, especially for web hosting services.

### What is Windows?

Windows 10 professional integrates the strengths of windows 2008 professional such as standards-based security, manageability, and reliability, with the best business features of windows 98 and windows Millennium Edition, such as plug and play, simplified user interface, and innovative support services. This combination creates the best desktop operating system for business.

It is more users friendly and a stable operating system equipped with much more added features. The operating system supports new technologies such as digital video disks, multiple monitors etc. along with plug and play and multi display features. It has a graphical user interface operating environment. Faster computing, easy access to remote information and control remote computers are some added features. Following are the common features of Windows 10.

Faster computing, easy access to remote information and control remote computers. Built-in networking and messaging facility.

* Easier to set up, add or remove.
* Increase system security and control.
* Support advanced networking and communication.

### HARDWARE SPECIFICATION

Selection of hardware configuration is very important task related to the software development. The processor should be powerful to handle all the operations. The hard disk should have the sufficient capacity to solve the database and the application.

### SYSTEM CONFIGURATION

### H/W SYSTEM CONFIGURATION

| Processor | Pentium –III |
| --- | --- |
| Speed | 1.1 GHz |
| RAM | 256 MB (min) |
| Hard Disk | 20 GB |
| Floppy Drive | 1.44 MB |
| Key Board | Standard Windows Keyboard |
| Mouse | Two or Three Button Mouse |
| Monitor | SVGA |

### S/W SYSTEM CONFIGURATION

| Operating System | Windows 95/98/2000/NT4.0. |
| --- | --- |
| Application Server | Wamp2.2e |
| Front End | HTML, PHP. |
| Scripts | JavaScript. |
| Server-side Script | PHP |
| Database | MySQL. |
| Database Connectivity | PhpMyAdmin. |

## SYSTEM DESIGN

### SYSTEM DESIGN

The term design describes the final system and the way in which it is developed. The system design is a solution, how to approach to the new system. This important phase is composed of several steps. An emphasis is on translating the performance requirements of our proposed system into design specification. Design goes through logical and physical stage of development. In the design phase the physical design producing the working system by defining a particular specification that helps to knowing exactly what the new system must do. The logical design determines the information flow into and of the system and require database. Design is a multistep process that focuses on data structure, software, architecture, procedural details, and interface between modules. The design process translates the requirements into the representation of the software. Computer software design changes continually because new methods, better analysis and broader understanding evolved. It provides the understanding and procedure details necessary for implementing the proposed system .an emphasis is on translating the performance requirement of our proposed system into design specification. Design goes through logical and physical stage. The system design is the last phase that indicate the final system and process of design phase. In the designed phase of maintenance management system, the database tables, input screens and output reports are designed. In table designing, redundancy is avoided. Design is the only way that we can accurately translate a system requirement into a software product. In our production management system, the all-input screens are designed as user friendly and understandable.

### INPUT DESIGN

Input design is the link that ties the information system into the world of its users. The input design involves determining what the input is, how the data should be performed, how to validate data, how to minimize data entry and how to provide a multi user facility, inaccurate input data are the most common cause of errors in data processing. Errors entered by data entry operator can be controlled by input design. Input design is the process of converting user originated input to a computer-based format. Input data are collected and organized into groups of similar data. Once identified, appropriate input media are selected for processing

All the input data re validated in the order and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, then it is transferred to the appropriate table in the database. A form is designed to enter the details should be user friendlier so that authorized user with even less knowledge can enter the data. The form is designed using v b tools like command boxes, text boxes, labels, option buttons, combo boxes etc. System analyst decodes the following input design details,

### OUTPUT DESIGN

Output design is very important concept in the computerized system, without reliable output the user may feel the entire system unnecessary and avoids using it. The proper output design is important in any system and facilitates effective decision making. The output design of this system includes various reports. output requirements are designed during system analysis. An application is successful only when it can provide efficient and effective reports.

The goal of the output design is to capture the output and get the data into a format suitable for the computer. It is very helpful to produce the clear, accurate and speedy information for end users.

A major form of the output is the harder copy from the pointer and screen reports. Printouts are designed around the output requirements of the user. Allowing the user to view the sample screen is important because the user is the ultimate judge of the quality of output. Output of this project is provided in the form of reports created using crystal report tool.

### DATABASE DESIGN

Database is a collection of interrelated data stores with minimum the overall objective in the development of the database technology has been to treat data as an organizational resource and has an integrated whole. Database management system allows data to be protected and organized separately from other resources. Database is an integrated collection of data. this is the difference between logical and physical data. The general objective is to make information access easy, quick, inexpensive and flexible for users. the database approach to system design places greater emphasis on the integration, integrity and independence of data. this involves the separation of logical and

Physical storage and vice versa. databases are normally implemented by using a package called dams.

### PROCESS DESIGN

Process design represents the structure of data and program components that are required to build a computer-based system. It considered the architectural style that the system will take, the structure and properties of the components constitute the system, and the interrelationships that occur among all architectural components of a system. Although a software engineer can design both and architecture, the job is often allocated to specialist when large, complex system are to be built. A database or data warehouse designer creates data architecture for a system. The ‘system architect ‘select an appropriate architectural style for the requirements derived during system engineering and software requirement analysis. Architectural design begins with data design and proceeds to the derivation of one or more representations of the architectural structure of the system. An architecture model encompassing data architecture and program structure is created during architectural design. In addition, component properties and the process by which it is developed. It refers to technical specifications that will be applied in implementing the system. It includes the construction of program and program testing. The input to design phase is software requirement specification

. Dad’s, e-r diagrams and structured diagrams depending on analysis. The output will be design specification. System design involves designing from layouts for input and reports for output.

### STRUCTURED DESIGN

Structured design deals with the data-flow in the system. It partitions a program into hierarchy of modules. The modules are organized in a top-down manner and the details will be at the bottom. The structured Design begins with a system specification that identifies inputs and outputs that described the functional of the Table.

### DATAFLOW DIAGRAM

Data flow diagram (DFD) is a graphical representation of the “flow” of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. A DED shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored.

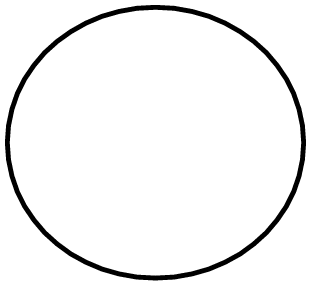
DFD is a designing tool used in the top-down approach to system Design. This context level DFD is next “exploded “, to produce a Level 1 DFD that shows some of the detail of the system being modelled. The Level 1 DFD shows how the system is divided into sub-systems (processes), each of which deals with one or more of the data flows to or from an external agent, and which together provide all of the functionality of the system as a whole. It also identifies internal data stores that must be present in order for the system to do its job and shows the flow of data between the various parts of the system.

* Function- An activity or a function that is performed for some specific reason; can be manual or computerized; ultimately each process should perform only one activity.
* Data Store- collection of data that is permanently stored.
* External Entity- A person, organization or system that is external to the system but interact with it.
* Data Flow- Single piece of data or logical collection of information like a bill.

The following are some DFD symbols used in the project.



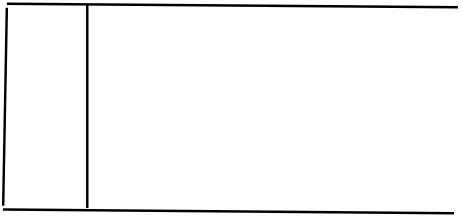
Rectangle: - It defines a source or destination of system data.



Circle: - It represents a process that transforms incoming data flow into outgoing data flow.



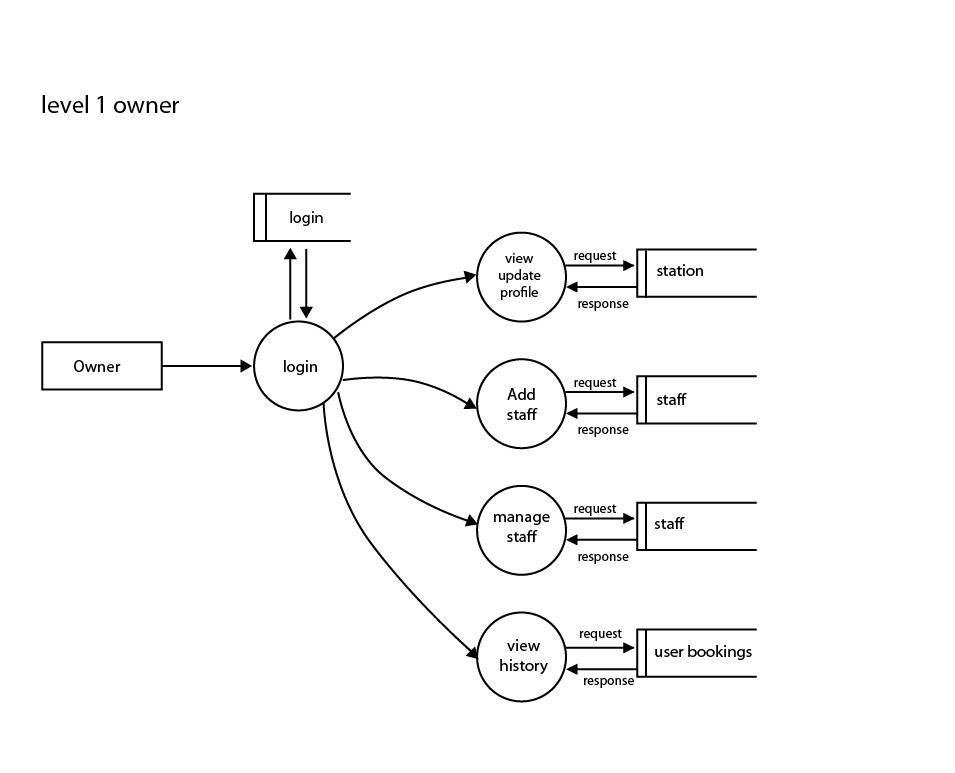
Arrow: - It defines data flow. It is a pipeline through which information flows.

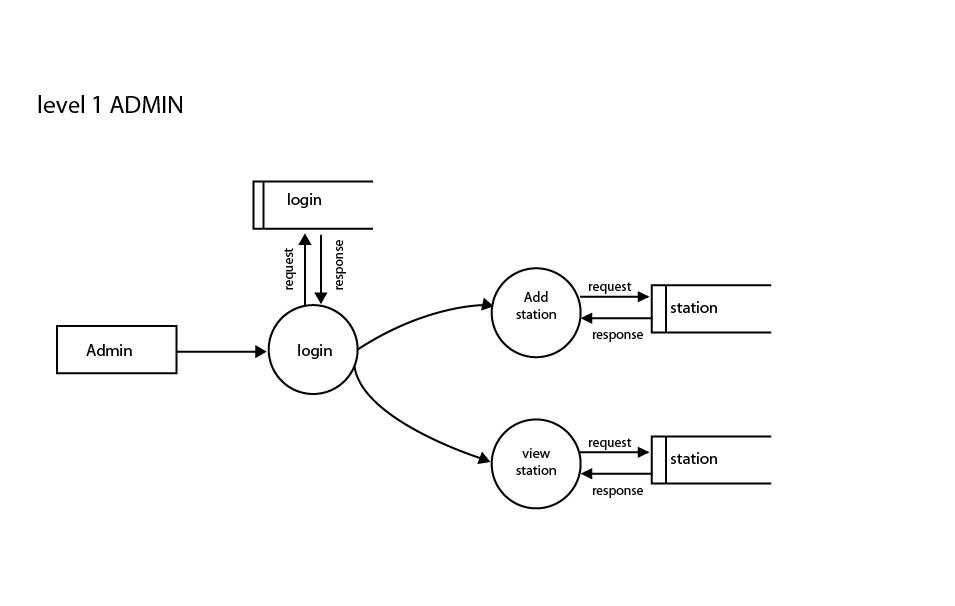
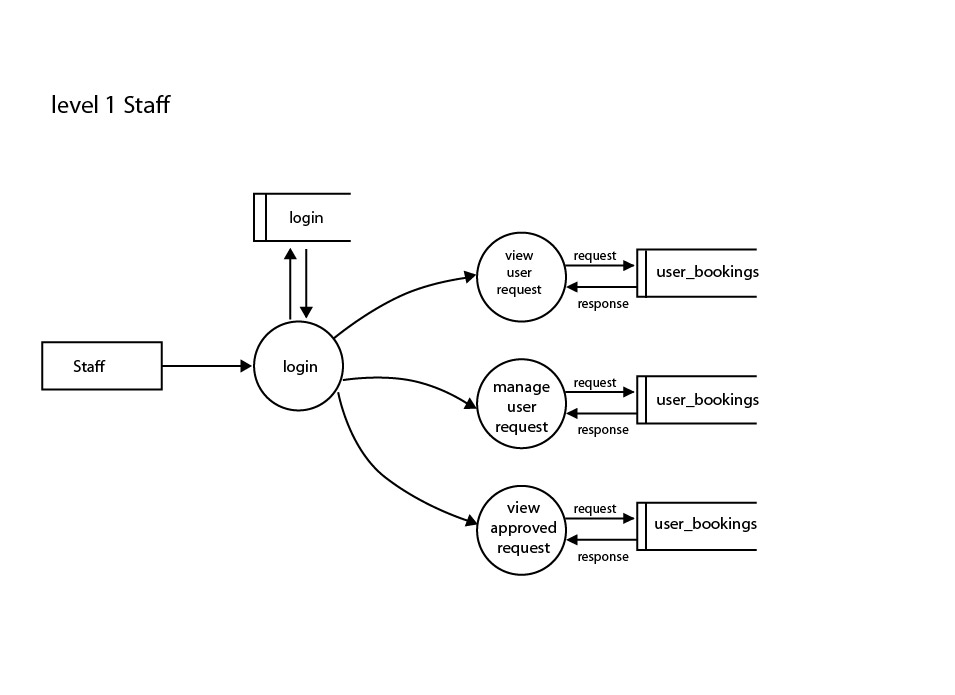


Open rectangle: - It is used to store data or a temporary repository of data.

### DATAFLOW DIAGRAM

### \\192.168.0.222\backup\STAFFS\ARCHANA\dfd 2021\ajin\electricox\level 0-01.jpg





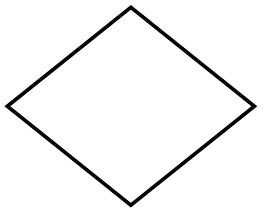
**ER Diagram**

The overall logical structure of a database can be expressed graphically by an E-R diagram. The relative simplicity and pictorial clarity of this diagramming technique may well account in large part for the widespread use of E-R model. Such diagram consists of the following major components:

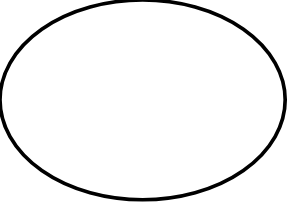
* Dashed ellipses: which denote derived attributes?
* Double ellipses: which represent multi valued attributes.
* Lines: which link attributes to entity sets and entity sets to relationship sets.  ¬
* Diamonds: which represents relationship sets?  ¬
* Ellipses: which represents attributes?  ¬
* Rectangles: which represent entity sets?  ¬



**Rectangle** which represents entity sets.

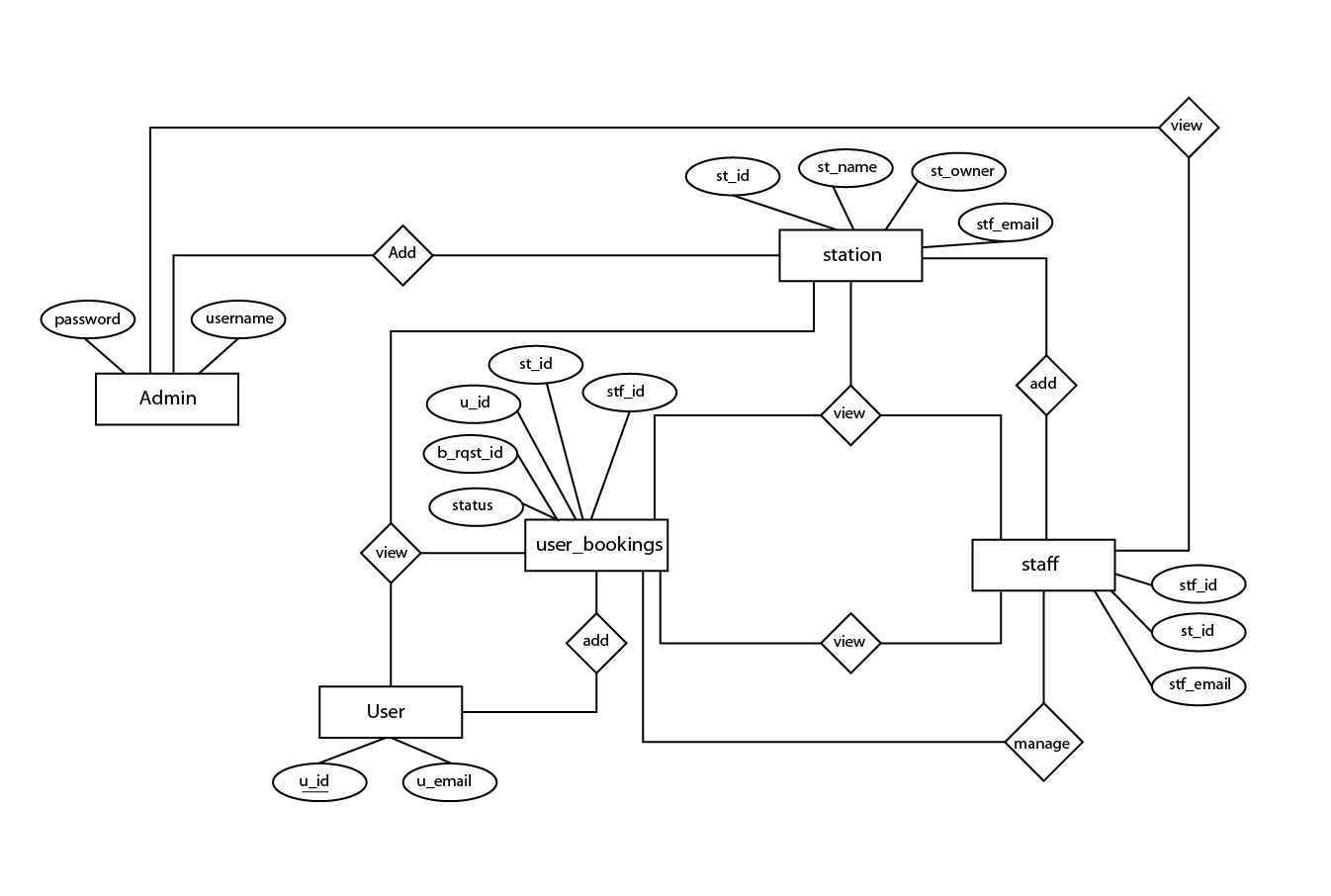


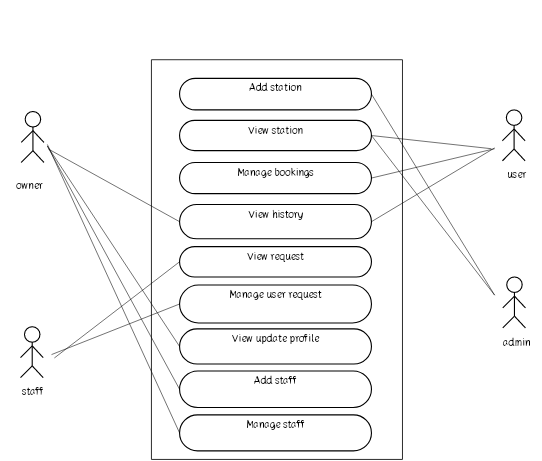
**Diamond**  which represents relationship sets.



**Ellipse** which represents attributes.

**Lines** which link attributes to entity sets and entity sets to relationship sets.





## SYSTEM DEVELOPMENT

### MODULE DESCRIPTION

**MODULES**

* **Admin**
* **Owner**
* **Station staff**
* **User**
* **Admin**

Admin has the overall control of the system. The main activities of admin were adding charging station, add location and update changes.

* **Owner**

The charging station is controlled by the owner. Owner activities are adding staff, removing staff, Add charging slots and update location for users.

* **Staff**

Staff module perform updating of availability of slots and send information to the user when charging is done.

* **User**

The user activities are booking slots for recharging, update location for finding nearest bunk and finding charging station.

### SOURCE CODE

<?php

session start ();

include("DB Connection/dbconnection.php");

?>

<!DOCTYPE html>

<html lang="zxx">

<head>

<title>Electricox :: Login</title>

<!-- Meta tag Keywords -->

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta charset="UTF-8" />

<meta name="keywords" content="Report Login Form Responsive web template, Bootstrap Web Templates, Flat Web Templates, Android Compatible web template, Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG, SonyEricsson, Motorola web design" />

<!-- //Meta tag Keywords -->

<link href="//fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@400;500;700;900&display=swap" rel="stylesheet">

<!--/Style-CSS -->

<link rel="stylesheet" href="login/css/style.css" type="text/css" media="all" />

<!--//Style-CSS -->

<link rel="stylesheet" href="login/css/font-awesome.min.css" type="text/css" media="all">

</head>

<body>

<!-- form section start -->

<section class="w3l-hotair-form my-tab-div-x-z">

<h1>Report Login Form</h1>

<div class="container">

<!-- /form -->

<div class="workinghny-form-grid">

<div class="main-hotair">

<div class="content-wthree">

<h2>Log In</h2>

<form action="#" method="post">

<input type="text" class="text" name="Email" placeholder="User Name" required="" autofocus>

<input type="password" class="password" name="Password" placeholder="User Password" required="" autofocus>

<button class="btn" type="submit" name="login">Log In</button>

</form>

<p class="account">Don't have an account? <a href="USER/userRegister.php">Register</a></p>

</div>

<div class="w3l\_form align-self">

<div class="left\_grid\_info">

<img src="login/images/1.png" alt="" class="img-fluid">

</div>

</div>

</div>

</div>

<!-- //form -->

</div>

<!-- copyright-->

<div class="copyright text-center">

<p class="copy-footer-29">© 2021 Report Login Form. All rights reserved | Design by <a href="#">LCC</a></p>

</div>

<!-- //copyright-->

<?php

if (isset($\_REQUEST['login'])) {

$email = $\_REQUEST['Email'];

$password = $\_REQUEST['Password'];

$qry = "SELECT \* FROM login WHERE `email` = '$email' AND `password` = '$password'";

$result = mysqli\_query($conn, $qry);

if ($result->num\_rows > 0) {

$data = $result->fetch\_assoc();

$uid = $data['reg\_id'];

$type = $data['type'];

$\_SESSION['uid'] = $uid;

$\_SESSION['type'] = $type;

if ($type == 'ADMIN') {

echo "<script>alert('Login Success'); window.location='ADMIN/adminHome.php'</script>";

} else if ($type == 'OWNER') {

echo "<script>alert('Login Success'); window.location='OWNER/ownerHome.php'</script>";

} else if ($type == 'STAFF') {

echo "<script>alert('Login Success'); window.location='STAFF/staffHome.php'</script>";

} else if ($type == 'USER') {

echo "<script>alert('Login Success'); window.location='USER/userHome.php'</script>";

} else {

echo "<script>alert('Login Failed')</script>";

}

} else {

echo "<script>alert('Invalid Email / Password'); window.location='index.php'</script>";

}

}

?>

</section>

<!-- //form section start -->

</body>

</html>

<?php

include("adminHeader.php");

?>

<!-- form section start -->

<section class="w3l-hotair-form my-tab-div-x">

<h1>Add Charging Station</h1>

<div class="container">

<!-- /form -->

<div class="workinghny-form-grid">

<div class="main-hotair">

<div class="content-wthree">

<h2>Add</h2>

<form action="#" method="post">

<input type="text" class="text" name="ST\_NAME" placeholder="Station Name" required autofocus>

<textarea placeholder="Address" name="ST\_ADDRESS" required></textarea>

<input type="text" name="ST\_LNO" placeholder="License Number" required>

<input type="text" class="text" name="NAME" placeholder="Owner Name" required>

<input type="number" name="AGE" placeholder="Age" required>

<input type="text" name="PHONE" placeholder="Phone" maxlength="10" pattern="[0-9][0-9]{9}" required>

<input value="E-MAIL" class="text" name="EMAIL" type="email" required onfocus="this.value = '';" onblur="if (this.value == '') {this.value = 'E-Mail';}" />

<input value="PASSWORD" class="password" name="PASSWORD" type="password" required onfocus="this.value = '';" onblur="if (this.value == '') {this.value = 'password';}" />

<div class="clear"></div>

<button class="btn" name="add\_station" type="submit">Add</button>

</form>

</div>

<div class="w3l\_form align-self">

<di id="map-x">

<img src="../login/images/1.png" alt="" class="img-fluid">

</div>

</div>

<!-- <div id="id-map" style="width: 600px; height: 500px;">

</div> -->

</div>

</div>

<!-- //form -->

</div>

<?php

if (isset($\_REQUEST['add\_station'])) {

$StName = $\_REQUEST['ST\_NAME'];

$StAddress = $\_REQUEST['ST\_ADDRESS'];

$StLicense = $\_REQUEST['ST\_LNO'];

$Name = $\_REQUEST['NAME'];

$Age = $\_REQUEST['AGE'];

$Phone = $\_REQUEST['PHONE'];

$Email = $\_REQUEST['EMAIL'];

$Password = $\_REQUEST['PASSWORD'];

$qryCheck = "SELECT COUNT(\*) AS cnt FROM `user\_reg` WHERE `u\_email` = '$Email' OR `u\_phone` = '$Phone'";

$qryOut = mysqli\_query($conn, $qryCheck);

$fetchData = mysqli\_fetch\_array($qryOut);

if ($fetchData['cnt'] > 0) {

echo "<script>alert('Already exist an Account with same Email / Phone Number');window.location = 'userRegister.php';</script>";

} else {

$qryReg = "INSERT INTO station(st\_name, st\_address, st\_l\_no, st\_owner, st\_age, st\_phone, st\_email) VALUES('$StName', '$StAddress', '$StLicense', '$Name', '$Age', '$Phone', '$Email')";

$qryLog = "INSERT INTO login(`reg\_id`, `email`, `password`, `type`) VALUES((select max(st\_id) from station), '$Email', '$Password', 'OWNER')";

// echo $qryReg . "&& " . $qryLog;

if ($conn->query($qryReg) == TRUE && $conn->query($qryLog) == TRUE) {

echo "<script>alert('Registration Success');window.location = 'addStation.php';</script>";

} else {

echo "<script>alert('Registration Failed');window.location = 'addStation.php';</script>";

}

}

}

?>

</section>

<!-- //form section start -->

<script>

// Initialize and add the map

function initMap() {

// The location of Uluru

const location = {

lat: -25.344,

lng: 131.036

};

// The map, centered at Uluru

const map = new google.maps.Map(document.getElementById("id-map"), {

zoom: 4,

center: location,

});

// The marker, positioned at Uluru

const marker = new google.maps.Marker({

position: location,

map: map,

});

}

</script>

<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCGSqedOUcnRGc1\_RIxL9QVxPP7xME7yv0&callback=initMap&libraries=&v=weekly" async></script>

<?php

include("../footer.php");

?>

<?php

session\_start();

include("../DBConnection/dbconnection.php");

?>

<!DOCTYPE html>

<html lang="zxx">

<head>

<title>Electricox :: Register</title>

<!-- Meta tag Keywords -->

<meta name="viewport" content="width=device-width, initial-scale=1">

<meta charset="UTF-8" />

<meta name="keywords" content="Report Login Form Responsive web template, Bootstrap Web Templates, Flat Web Templates, Android Compatible web template, Smartphone Compatible web template, free webdesigns for Nokia, Samsung, LG, SonyEricsson, Motorola web design" />

<!-- //Meta tag Keywords -->

<link href="//fonts.googleapis.com/css2?family=Noto+Sans+JP:wght@400;500;700;900&display=swap" rel="stylesheet">

<!--/Style-CSS -->

<link rel="stylesheet" href="../login/css/style.css" type="text/css" media="all" />

<!--//Style-CSS -->

<link rel="stylesheet" href="../login/css/font-awesome.min.css" type="text/css" media="all">

</head>

<body>

<!-- form section start -->

<section class="w3l-hotair-form my-tab-div-x-z">

<h1>Report Register Form</h1>

<div class="container">

<!-- /form -->

<div class="workinghny-form-grid">

<div class="main-hotair">

<div class="content-wthree">

<h2>Register</h2>

<form action="#" method="post">

<input type="text" class="text" name="NAME" placeholder="Name" required autofocus>

<input type="number" name="AGE" placeholder="Age" required>

<input type="text" name="PHONE" placeholder="Phone" maxlength="10" pattern="[0-9][0-9]{9}" required>

<textarea placeholder="Address" name="ADDRESS" required></textarea>

<input value="E-MAIL" class="text" name="EMAIL" type="email" required onfocus="this.value = '';" onblur="if (this.value == '') {this.value = 'E-Mail';}" />

<input value="PASSWORD" class="password" name="PASSWORD" type="password" required onfocus="this.value = '';" onblur="if (this.value == '') {this.value = 'password';}" />

<div class="clear"></div>

<button class="btn" name="register" type="submit">Register</button>

</form>

<p class="account">Already have an account? <a href="../index.php">Login</a></p>

</div>

<div class="w3l\_form align-self">

<div class="left\_grid\_info">

<img src="../login/images/1.png" alt="" class="img-fluid">

</div>

</div>

</div>

</div>

<!-- //form -->

</div>

<!-- copyright-->

<div class="copyright text-center">

<p class="copy-footer-29">© 2021 Report Login Form. All rights reserved | Design by <a href="#">LCC</a></p>

</div>

<!-- //copyright-->

<?php

if (isset($\_REQUEST['register'])) {

$Name = $\_REQUEST['NAME'];

$Age = $\_REQUEST['AGE'];

$Phone = $\_REQUEST['PHONE'];

$Address = $\_REQUEST['ADDRESS'];

$Email = $\_REQUEST['EMAIL'];

$Password = $\_REQUEST['PASSWORD'];

$qryCheck = "SELECT COUNT(\*) AS cnt FROM `user\_reg` WHERE `u\_email` = '$Email' OR `u\_phone` = '$Phone'";

$qryOut = mysqli\_query($conn, $qryCheck);

$fetchData = mysqli\_fetch\_array($qryOut);

if ($fetchData['cnt'] > 0) {

echo "<script>alert('Already exist an Account with same Email / Phone Number');window.location = 'userRegister.php';</script>";

} else {

$qryReg = "INSERT INTO user\_reg(u\_name, u\_age, u\_phone, u\_address, u\_email) VALUES('$Name', '$Age', '$Phone', '$Address', '$Email')";

$qryLog = "INSERT INTO login(`reg\_id`, `email`, `password`, `type`) VALUES((select max(u\_id) from user\_reg), '$Email', '$Password', 'USER')";

// echo $qryReg . "&& " . $qryLog;

if ($conn->query($qryReg) == TRUE && $conn->query($qryLog) == TRUE) {

echo "<script>alert('Registration Success');window.location = '../index.php';</script>";

} else {

echo "<script>alert('Registration Failed');window.location = 'userRegister.php';</script>";

}

}

}

?>

</section>

<!-- //form section start -->

</body>

</html>

<?php

include('userHeader.php');

$uid = $\_SESSION['uid'];

?>

<section class="w3l-hotair-form my-tab-div-x">

<h1>Charging Station</h1>

<div class="view-content">

<?php

$qry = "SELECT \* FROM `station`";

// $time = "SELECT "

$result = mysqli\_query($conn, $qry);

if ($result->num\_rows > 0) {

// echo date("H:i:s");

?>

<table width="60%">

<tr>

<th>Station Name</th>

<th>Location</th>

<th>License</th>

<th>Price per 1 kWh</th>

<th>Price per Hour</th>

<th>Parking Price</th>

<th>Opening Time</th>

<th>Closing Time</th>

<th>Owner</th>

<th>Phone</th>

<th>Email</th>

<th>Action</th>

</tr>

<?php

while ($row = mysqli\_fetch\_array($result)) {

echo "

<tr>

<td>" . $row['st\_name'] . "</td>

<td>" . $row['st\_address'] . "</td>

<td>" . $row['st\_l\_no'] . "</td>

<td>" . $row['st\_price\_one\_kwh'] . "</td>

<td>" . $row['st\_price\_per\_hour'] . "</td>

<td>" . $row['st\_parking\_price'] . "</td>

<td>" . $row['st\_open'] . "</td>

<td>" . $row['st\_close'] . "</td>

<td>" . $row['st\_owner'] . "</td>

<td>" . $row['st\_phone'] . "</td>

<td>" . $row['st\_email'] . "</td>

<td><a class='approve-btn' href='bookStation.php?stid=" . $row['st\_id'] . "&u\_id=" . $uid . "'>Book</a></td>

</tr>";

}

} else {

echo "<div class='no-data-div'><img src='../images/no\_data.png'><p>No Stations</p></div>";

}

?>

</table>

</div>

</section>

<?php

include('../footer.php');

?>

<?php

include('userHeader.php');

$uid = $\_SESSION['uid'];

?>

<section class="w3l-hotair-form my-tab-div-x">

<h1>User Request</h1>

<div class="view-content">

<?php

$qry = "SELECT ub.\*, st.\* FROM `user\_bookings` ub, `station` st, `user\_reg` u WHERE ub.`u\_id` = u.`u\_id` AND ub.`st\_id` = st.`st\_id` AND ub.`status` = 'REQUESTED' AND ub.`u\_id` = '$uid'";

// $time = "SELECT "

$result = mysqli\_query($conn, $qry);

if ($result->num\_rows > 0) {

// echo date("H:i:s");

?>

<table width="60%">

<tr>

<th>Station Name</th>

<th>Location</th>

<th>License</th>

<th>Price per 1 kWh</th>

<th>Price per Hour</th>

<th>Parking Price</th>

<th>Owner</th>

<th>Phone</th>

<th>Email</th>

<th>Status</th>

<th>Requested On</th>

</tr>

<?php

while ($row = mysqli\_fetch\_array($result)) {

echo "

<tr>

<td>" . $row['st\_name'] . "</td>

<td>" . $row['st\_address'] . "</td>

<td>" . $row['st\_l\_no'] . "</td>

<td>" . $row['st\_price\_one\_kwh'] . "</td>

<td>" . $row['st\_price\_per\_hour'] . "</td>

<td>" . $row['st\_parking\_price'] . "</td>

<td>" . $row['st\_owner'] . "</td>

<td>" . $row['st\_phone'] . "</td>

<td>" . $row['st\_email'] . "</td>

<td>" . $row['status'] . "</td>

<td>" . $row['date'] . "</td>

</tr>";

}

} else {

echo "<div class='no-data-div'><img src='../images/no\_data.png'><p>No Requests</p></div>";

}

?>

</table>

</div>

</section>

<?php

include('../footer.php');

?>

<?php

include("ownerHeader.php");

?>

<!-- form section start -->

<section class="w3l-hotair-form my-tab-div-x">

<h1>Add Station Staff</h1>

<div class="container">

<!-- /form -->

<div class="workinghny-form-grid">

<div class="main-hotair">

<div class="content-wthree">

<h2>Add</h2>

<form action="#" method="post">

<input type="text" class="text" name="NAME" placeholder="Staff Name" required>

<input type="number" name="AGE" placeholder="Age" required>

<textarea placeholder="Address" name="ADDRESS" required></textarea>

<input type="text" name="PHONE" placeholder="Phone" maxlength="10" pattern="[0-9][0-9]{9}" required>

<textarea placeholder="Qualification" name="QUAL" required></textarea>

<input type="number" name="EXP" placeholder="Experience (\*in years)" required>

<input value="E-MAIL" class="text" name="EMAIL" type="email" required onfocus="this.value = '';" onblur="if (this.value == '') {this.value = 'E-Mail';}" />

<input value="PASSWORD" class="password" name="PASSWORD" type="password" required onfocus="this.value = '';" onblur="if (this.value == '') {this.value = 'password';}" />

<div class="clear"></div>

<button class="btn" name="add\_staff" type="submit">Add</button>

</form>

</div>

<div class="w3l\_form align-self">

<di id="map-x">

<img src="../login/images/1.png" alt="" class="img-fluid">

</div>

</div>

<!-- <div id="id-map" style="width: 600px; height: 500px;">

</div> -->

</div>

</div>

<!-- //form -->

</div>

<?php

$uid = $\_SESSION['uid'];

if (isset($\_REQUEST['add\_staff'])) {

$Name = $\_REQUEST['NAME'];

$Age = $\_REQUEST['AGE'];

$Address = $\_REQUEST['ADDRESS'];

$Phone = $\_REQUEST['PHONE'];

$Qual = $\_REQUEST['QUAL'];

$Exp = $\_REQUEST['EXP'];

$Email = $\_REQUEST['EMAIL'];

$Password = $\_REQUEST['PASSWORD'];

$qryCheck = "SELECT COUNT(\*) AS cnt FROM `staff` WHERE `stf\_email` = '$Email' OR `stf\_phone` = '$Phone'";

$qryOut = mysqli\_query($conn, $qryCheck);

$fetchData = mysqli\_fetch\_array($qryOut);

if ($fetchData['cnt'] > 0) {

echo "<script>alert('Already exist an Account with same Email / Phone Number');window.location = 'userRegister.php';</script>";

} else {

$qryReg = "INSERT INTO staff(st\_id, stf\_name, stf\_age, stf\_phone, stf\_address, stf\_qualification, stf\_experience, stf\_email) VALUES('$uid', '$Name', '$Age', '$Phone', '$Address', '$Qual', '$Exp', '$Email')";

$qryLog = "INSERT INTO login(`reg\_id`, `email`, `password`, `type`) VALUES((select max(stf\_id) from staff), '$Email', '$Password', 'STAFF')";

// echo $qryReg . "&& " . $qryLog;

if ($conn->query($qryReg) == TRUE && $conn->query($qryLog) == TRUE) {

echo "<script>alert('New staff added Successfully');window.location = 'addStaff.php';</script>";

} else {

echo "<script>alert('Can't add Staff');window.location = 'addStaff.php';</script>";

}

}

}

?>

</section>

<!-- //form section start -->

<script>

// Initialize and add the map

function initMap() {

// The location of Uluru

const location = {

lat: -25.344,

lng: 131.036

};

// The map, centered at Uluru

const map = new google.maps.Map(document.getElementById("id-map"), {

zoom: 4,

center: location,

});

// The marker, positioned at Uluru

const marker = new google.maps.Marker({

position: location,

map: map,

});

}

</script>

<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCGSqedOUcnRGc1\_RIxL9QVxPP7xME7yv0&callback=initMap&libraries=&v=weekly" async></script>

<?php

include("../footer.php");

?>

<?php

include("ownerHeader.php");

$stid = $\_SESSION['uid'];

$qry = "SELECT st.\*, l.`password` FROM `station` st, `login` l WHERE st.`st\_id` = l.`reg\_id` AND l.`type` = 'OWNER' AND st.`st\_id` = '$stid'";

$result = mysqli\_query($conn, $qry);

$row = mysqli\_fetch\_array($result);

?>

<!-- form section start -->

<section class="w3l-hotair-form my-tab-div-x">

<h1>Charging Station Profile</h1>

<div class="container">

<!-- /form -->

<div class="workinghny-form-grid">

<div class="main-hotair">

<div class="content-wthree">

<h2>Update</h2>

<form action="#" method="post">

<div class="div-input-x">

<p>Station Name</p>

<input type="text" class="text" name="ST\_NAME" placeholder="Station Name" required autofocus value="<?php echo $row['st\_name'] ?>">

</div>

<div class="div-input-x">

<p>Station Address</p>

<textarea placeholder="Address" name="ST\_ADDRESS" required><?php echo $row['st\_address'] ?></textarea>

</div>

<div class="div-input-x">

<p>License Number</p>

<input disabled type="text" name="ST\_LNO" placeholder="License Number" required value="<?php echo $row['st\_l\_no'] ?>">

</div>

<div class="div-input-x">

<p>Price per 1 kWh</p>

<input type="text" name="ST\_PPONE" placeholder="Price per 1 kWh" required value="<?php echo $row['st\_price\_one\_kwh'] ?>">

</div>

<div class="div-input-x">

<p>Price per 1 hour</p>

<input type="text" name="ST\_PPHR" placeholder="Price per 1 hour" required value="<?php echo $row['st\_price\_per\_hour'] ?>">

</div>

<div class="div-input-x">

<p>Parking Price</p>

<input type="text" name="ST\_PP" placeholder="Parking price" required value="<?php echo $row['st\_parking\_price'] ?>">

</div>

<div class="div-input-x">

<p>Opening Time</p>

<input type="time" name="ST\_OPEN" placeholder="Opening Time" required value="<?php echo $row['st\_open'] ?>">

</div>

<div class="div-input-x">

<p>Closing Time</p>

<input type="time" name="ST\_CLOSE" placeholder="Closing Time" required value="<?php echo $row['st\_close'] ?>">

</div>

<div class="div-input-x">

<p>Owner Name</p>

<input type="text" class="text" name="NAME" placeholder="Owner Name" required value="<?php echo $row['st\_owner'] ?>">

</div>

<div class="div-input-x">

<p>Age</p>

<input type="number" name="AGE" placeholder="Age" required value="<?php echo $row['st\_age'] ?>">

</div>

<div class="div-input-x">

<p>Phone</p>

<input type="text" name="PHONE" placeholder="Phone" maxlength="10" pattern="[0-9][0-9]{9}" required value="<?php echo $row['st\_phone'] ?>">

</div>

<div class="div-input-x">

<p>Email</p>

<input class="text" name="EMAIL" type="email" required placeholder="Email" value="<?php echo $row['st\_email'] ?>" />

</div>

<div class="div-input-x">

<p>Password</p>

<input class="password" name="PASSWORD" type="password" placeholder="Password" required value="<?php echo $row['password'] ?>" />

</div>

<div class="clear"></div>

<button class="btn" name="update\_station" type="submit">Update</button>

</form>

</div>

<div class="w3l\_form align-self">

<di id="map-x">

<img src="../login/images/1.png" alt="" class="img-fluid">

</div>

</div>

<!-- <div id="id-map" style="width: 600px; height: 500px;">

</div> -->

</div>

</div>

<!-- //form -->

</div>

<?php

if (isset($\_REQUEST['update\_station'])) {

$StName = $\_REQUEST['ST\_NAME'];

$StAddress = $\_REQUEST['ST\_ADDRESS'];

$StLicense = $\_REQUEST['ST\_LNO'];

$StPPOne = $\_REQUEST['ST\_PPONE'];

$StPPhr = $\_REQUEST['ST\_PPHR'];

$StPP = $\_REQUEST['ST\_PP'];

$StOpen = $\_REQUEST['ST\_OPEN'];

$StClose = $\_REQUEST['ST\_CLOSE'];

$Name = $\_REQUEST['NAME'];

$Age = $\_REQUEST['AGE'];

$Phone = $\_REQUEST['PHONE'];

$Email = $\_REQUEST['EMAIL'];

$Password = $\_REQUEST['PASSWORD'];

$qryReg = "UPDATE `station` SET `st\_name` = '$StName', `st\_address` = '$StAddress', `st\_price\_one\_kwh` = '$StPPOne', `st\_price\_per\_hour` = '$StPPhr', `st\_parking\_price` = '$StPP', `st\_owner` = '$Name', `st\_age` = '$Age', `st\_phone` = '$Phone', `st\_email` = '$Email', `st\_open` = '$StOpen', `st\_close` = '$StClose' WHERE `st\_id` = '$stid'";

$qryLog = "UPDATE `login` SET `email` = '$Email' , `password` = '$Password' WHERE `reg\_id` = '$stid' AND `type` = 'OWNER'";

// echo $qryReg . "&& " . $qryLog;

if ($conn->query($qryReg) == TRUE && $conn->query($qryLog) == TRUE) {

echo "<script>alert('Profile Updated Successfully');window.location = 'profile.php';</script>";

} else {

echo "<script>alert('Update Failed');window.location = 'profile.php';</script>";

}

}

?>

</section>

<!-- //form section start -->

<script>

// Initialize and add the map

function initMap() {

// The location of Uluru

const location = {

lat: -25.344,

lng: 131.036

};

// The map, centered at Uluru

const map = new google.maps.Map(document.getElementById("id-map"), {

zoom: 4,

center: location,

});

// The marker, positioned at Uluru

const marker = new google.maps.Marker({

position: location,

map: map,

});

}

</script>

<script src="https://maps.googleapis.com/maps/api/js?key=AIzaSyCGSqedOUcnRGc1\_RIxL9QVxPP7xME7yv0&callback=initMap&libraries=&v=weekly" async></script>

<?php

include("../footer.php");

?>

<?php

include('staffHeader.php');

$uid = $\_SESSION['uid'];

?>

<section class="w3l-hotair-form my-tab-div-x">

<h1>User Request</h1>

<div class="view-content">

<?php

$qry = "SELECT ub.\*, u.\*, st.`st\_name` FROM `user\_bookings` ub, `station` st, `user\_reg` u, `staff` stf WHERE ub.`u\_id` = u.`u\_id` AND ub.`st\_id` = st.`st\_id` AND ub.`st\_id` = stf.`st\_id` AND ub.`status` = 'REQUESTED'";

// $time = "SELECT "

$result = mysqli\_query($conn, $qry);

if ($result->num\_rows > 0) {

// echo $uid;

?>

<table width="60%">

<tr>

<th>Customer</th>

<th>Phone</th>

<th>Email</th>

<th>Status</th>

<th>Requested On</th>

<th>Station</th>

<th>Action</th>

</tr>

<?php

while ($row = mysqli\_fetch\_array($result)) {

echo "

<tr>

<td>" . $row['u\_name'] . "</td>

<td>" . $row['u\_phone'] . "</td>

<td>" . $row['u\_email'] . "</td>

<td>" . $row['status'] . "</td>

<td>" . $row['date'] . "</td>

<td>" . $row['st\_name'] . "</td>

<td><a class='approve-btn' href='approveUserRequest.php?rqstid=" . $row['b\_rqst\_id'] . "&stf\_id=" . $uid . "'>Approve</a></td>

</tr>";

}

} else {

echo "<div class='no-data-div'><img src='../images/no\_data.png'><p>No Request</p></div>";

}

?>

</table>

</div>

</section>

<?php

include('../footer.php');

?>

<?php

include('staffHeader.php');

$uid = $\_SESSION['uid'];

?>

<section class="w3l-hotair-form my-tab-div-x">

<h1>Approved Request</h1>

<div class="view-content">

<?php

$qry = "SELECT ub.\*, u.\*, st.`st\_name` FROM `user\_bookings` ub, `station` st, `user\_reg` u, `staff` stf WHERE ub.`u\_id` = u.`u\_id` AND ub.`st\_id` = st.`st\_id` AND ub.`stf\_id` = stf.`stf\_id` AND ub.`st\_id` = stf.`st\_id` AND ub.`status` = 'APPROVED' AND ub.`stf\_id` = '$uid'";

// $time = "SELECT "

$result = mysqli\_query($conn, $qry);

if ($result->num\_rows > 0) {

// echo $uid;

?>

<table width="60%">

<tr>

<th>Customer</th>

<th>Phone</th>

<th>Email</th>

<th>Status</th>

<th>Requested On</th>

<th>Station</th>

<th>Action</th>

</tr>

<?php

while ($row = mysqli\_fetch\_array($result)) {

echo "

<tr>

<td>" . $row['u\_name'] . "</td>

<td>" . $row['u\_phone'] . "</td>

<td>" . $row['u\_email'] . "</td>

<td>" . $row['status'] . "</td>

<td>" . $row['date'] . "</td>

<td>" . $row['st\_name'] . "</td>

<td><a class='approve-btn' href='fullyCharged.php?rqstid=" . $row['b\_rqst\_id'] . "&stf\_id=" . $uid . "'>Fully Charged</a></td>

</tr>";

}

} else {

echo "<div class='no-data-div'><img src='../images/no\_data.png'><p>No Approved Request</p></div>";

}

?>

</table>

</div>

</section>

<?php

include('../footer.php');

?>

<?php

include('staffHeader.php');

$uid = $\_SESSION['uid'];

?>

<section class="w3l-hotair-form my-tab-div-x">

<h1>History</h1>

<div class="view-content">

<?php

$qry = "SELECT ub.\*, u.\*, st.`st\_name` FROM `user\_bookings` ub, `station` st, `user\_reg` u, `staff` stf WHERE ub.`u\_id` = u.`u\_id` AND ub.`st\_id` = st.`st\_id` AND ub.`stf\_id` = stf.`stf\_id` AND ub.`st\_id` = stf.`st\_id` AND ub.`status` = 'FULLY CHARGED' AND ub.`stf\_id` = '$uid'";

// $time = "SELECT "

$result = mysqli\_query($conn, $qry);

if ($result->num\_rows > 0) {

?>

<table width="60%">

<tr>

<th>Customer</th>

<th>Phone</th>

<th>Email</th>

<th>Station</th>

<th>Completed On</th>

<th>Status</th>

</tr>

<?php

while ($row = mysqli\_fetch\_array($result)) {

echo "

<tr>

<td>" . $row['u\_name'] . "</td>

<td>" . $row['u\_phone'] . "</td>

<td>" . $row['u\_email'] . "</td>

<td>" . $row['st\_name'] . "</td>

<td>" . $row['date'] . "</td>

<td>" . $row['status'] . "</td>

</tr>";

}

} else {

echo "<div class='no-data-div'><img src='../images/no\_data.png'><p>No Data</p></div>";

}

?>

</table>

</div>

</section>

<?php

include('../footer.php');

?>

## SYSTEM IMPLEMENTATION

## 

### SYSTEM TESTING

Testing is an important phase in software development. After completion, the system may work without any problem. But there should be several unknown or hidden errors in the system still remaining. The error chances may be injected into the system at any stage of the development. Even if there are techniques to detect and eliminate the errors, some errors may retain in the system. So, after the completion of coding, the system is to be executed with the only purpose of detecting maximum number of errors. The tester executes the system, and inputs different types of values those may cause error or some exceptional situation in the system. The error locations detected through the testing are to be corrected in the system then. So, the important and the only aim of testing is to detect and cure even a less possible of an error that may face in the future executions of the system. Testing is a set of activity that can be planned in advance and conducted systematically. Testing begins at the module level and work towards the integration of entire computers-based system. Nothing is completed without testing, as it is vital to the success of the system. System testing makes a logical assumption that if all parts of the system are corrected, the goal will be successfully achieved. Inadequate testing or non-testing may lead to errors that may not appear until months later.

### PURPOSE OF TESTING

Testing is the success of the system. System testing makes a logical assumption that if all part of the system is correct, the goal will be successfully achieved. The following points shows how testing is essential. Existence of program defects of inadequacies is inferred. Verifies whether the software behave as intended byitsdesigner. Checks conformance with requirements specification or user need.

Access the operational reliability of the system. Test the performance of the system. The performance of the system. Reflects the frequencies of actual user inputs. Find the fault which caused the output anomaly. Detect flaws and deficiencies in requirements. Exercise the program using data like the real data processed by the program. Test the system capabilities. Judges whether or not the program is usable in practice. Testing objectives There are several rules that can serve as testing objectives. They are; Testing is a process of executing a program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. A successful test is one that uncovers an undiscovered error.

If testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrates that software functions appear to the working according to the specifications, that performance requirement appear to have been met. These are three ways to test a program for correctness for implementation efficiency for computational complexity Test for correctness are supported to verify that a program does exactly what it was designed to do. This is much difficult that it may at first appear especially for large programs.

Tests for implementation efficiency attempt to find ways to make a correct program faster or use less storage. It is a code-refining process, which reexamines the implementation phase algorithm development. Tests for computational complexity amount to an experiment analysis of the complexity of an algorithm or an experiment comparison of two or more algorithms, which solve the same problem.

### TYPES OF TESTING

System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct. The goal will be successfully achieving. The candidate system is subject to a variety of tests. A series of tests are performed for the proposed system is ready for system acceptance testing. The various levels at which testing are conducted are,

* Unit testing
* Integration testing
* Sequential testing
* System testing
* Validation testing unit testing
* **UNIT TESTING**

In unit testing each program unit is tested individually.so any errors in a unit are debugged. Sample data is given for unit testing. The unit test results are recorded for future references. Unit testing focus verification efforts on the smallest unit of software design, the module. This is known as “module testing”. It comprises of the set test performed by an individual programmer prior to the integration of unit into the large system. The modules are tested separately, this testing is carried out programming stage itself.

In this step each module is found to be working satisfactory as regard to the expected out from module. The unit testing was done for every module in the software for various inputs, such they each line of code is at least once executed. This testing was carried out during the unit to a large system.

* **INTEGRATION TESTING**

Integration testing is a systematic technique for constructing the program structure while at the same time conducting test to uncover errors associated with interfacing.

* **PROGRAM TESTING**

Program testing checks for two types of errors; syntax and logic. A syntax error is a program statement that violates one or more rules of the language in which it is written. A logic error deals with incorrect data fields. When a program is tested, the actual output is compared with the expected output. All the modules are combined and tested as a whole. Here correction is difficult because the vast expenses of all errors uncovered are correct for the next testing steps. We follow bottom-up integration. Bottom-up integration testing as its name implies begin construction and sling with atomic modules. Because components are integrated from the bottom up, accessing required for the components subordinate to a given level is always available and need for stubs is eliminated.

* **SEQUENTIAL TESTING**

Sequential or series testing is checking the logic of one or more programs in the candidate system, where the output of one program will affect the processing done by another program.

* **SYSTEM TESTING**

System testing executing a program to check logic changes made in it and with the intension of finding errors-making the program fails. Effective testing does not guaranty reliability is a design consideration. This testing actually consists of a series of different test whose primary purpose is to fully exercise the computer based system.it begins where integration testing is completed and finally software is completely assembled as package, interfacing errors are uncovered and corrected.

* **ACCEPTANCE TESTING**

Acceptance testing is running the system with live data by the actual user. An acceptance test has the objective of selling the user in the validity and reliability of the system. A comprehensive test report is prepared. The report indicates the system’s tolerance, performance range, error rate and accuracy. It verifies the system procedures operate to system specification and the integrity of important data is maintained, performance of an acceptance test is actually the users show. User motivation is very important for the successful performance of the system. After that a comprehensive test report is prepared. This report shows the systems tolerance, performance range, error rate and accuracy.

* **INPUT TESTING**

Here system is tested with all verifiable combination of input. User may type data in situations like entering password, numerical details etc. The system is tested with all the causes and it responded with appropriate error message.

* **OUTPUT TESTING**

Here the output is tested to view where the screen is what which is desired. It is also checked whether it is to the satisfaction of the user. Changes that need to be done can be done after the result is seen.

### SYSTEM IMPLEMENTATION

A crucial phase in the system life cycle is the successful implementation of the new system design. Implementation involves creating computer compatible files, training the operating staff, installing hardware, terminals. In the system implementation, user training is crucial for minimizing resistance to change and giving the new system a chance to prove its worth. The objectives of the system implementation are to put the system into operation while holding costs, risks and personal irritation to minimum. Once the physical system has been designed in details, the next stage is to run the design into a working system and then to monitor the operation of the system to ensure that is continue to work efficiently and the operation of the system to ensure that is continues to work efficiently and effectively. The implementation stage of a is often very complex and time consuming because many more people are involved than in the earlier stages. The system implementation took place through various stages as follows,

* Implantation planning.
* Education and training.
* System testing.
* System implementation.
* Change over.

The implementation plan includes a description of all the activities that must occurs to implement the new system and to put it into operation. To achieve the objectives and benefits from computer-based system, it is essential for the people who will be confident of their role in the new jobs. After software is developed to meet user’s requirements, users test it for acceptance. The changes over phase are used to provide adaptability for the new system.

## FUTURE SCOPE OF PROJECT

### SYSTEM MAINTENANCE

Software maintenance is the process of modifying a software system or component after its delivery in order to correct faults improve the performance and other attributes, or to adapt to the changed environment. maintenance covers a wide range of activities including correcting the error and design coding, updating the documentation and test data, and upgrading the user hardware and software. maintenance is always necessary to keep the software usable and useful. Hardware also requires periodic maintenance to keep the system into its standards. After installation is completed and user start is adjusted to the changes created by the candidate system. evaluation and maintenance begin. If new information is consistent with design specification the changes have to be made. Hardware also requires periodic maintenance to keep in tune with design specifications. User priorities changes in organizational requirements or environmental factors also called for system enhancements. Maintenance covers wide range of activities, including correcting, coding and design errors, updating documentation and test data, and upgrading user support. any activities classified as maintenance are actually enhancements. Maintenance means restoring something to do its original condition. Unlike hardware, software does not wear out; it is corrected.in contrast, enhancement means adding, modifying or redeveloping the code to support changes in the specifications.it is necessary to keep up with changing user needs the operational environment. Maintenance means repairing processing or performance failures or making changes because of previously uncorrected problems or false assumptions. adaptive maintenance means changing the program function. Perfective maintenance means enhancing the performance or modifying the program to respond to the user’s additional or changing needs. Of these types, more time and money are spent on perfective than on corrective and adaptive maintenance together. Maintenance activities begin where conversion leaves off. Maintenance is handled by the same planning and control used in a formal system project. a major problem with Software maintenance is its labor-intensive nature. Documentation is as much a part of maintenance as it is of system development .to put maintenance in its proper perspective requires considerable skill and experience and is an important and is an important and ongoing aspect of system development .an additional factor in the success of the maintenance programmer is the work environment.

##### Maintenance Types

System maintenance can be classified into four types –

* Corrective Maintenance
* Adaptive Maintenance
* Perfective Maintenance
* Preventive Maintenance

##### Corrective Maintenance

Corrective Maintenance deals with the repair of faults or defects found in day- today system functions. A defect can result due to errors in software design, logic and coding. Design errors occur when changes made to the software are incorrect, incomplete, wrongly communicated, or the change request is misunderstood. Logical errors result from invalid tests and conclusions, incorrect implementation of design specifications, faulty logic flow, or incomplete implementation of design specifications, faulty logic flow, or incomplete test of data. All these errors, referred to as residual errors, prevent the software from confirming to its agreed specifications. Note that the need for corrective maintenance is usually initiated by big reports drawn by the users.

##### Adaptive Maintenance

Adaptive Maintenance is the implementation of changes in a part of the system, which has been affected by a change that occurred in some other part of the system. Adaptive Maintenance consists of adapting software to changes in the environment such as the hardware or the operating system. The term environment in this context refers to the conditions and the influences which act (from outside) on the system. For example, business rules, work patterns and government policies have a significant impact on the software system.

##### Perfective Maintenance

Perfective Maintenance mainly deals with implementing new or changed user requirements. Perfective Maintenance involves making functional enhancements to the system in addition to the activities to increase the system’s performance even when the changes have not been suggested by faults. This includes enhancing both the function and efficiency of the code and changing the functionalities of the system as per the users’ changing needs.

##### Preventive Maintenance

Preventive Maintenance involves performing activities to prevent the occurrence of errors. It tends to reduce the software complexity thereby improving program understand ability and increasing software maintainability. It comprises documentation updating, code optimization and code restructuring. Documentation updating involves modifying the documents affected by the changes in order to correspond to the present state of the system. Code optimization involves modifying the programs for faster execution or efficient use of storage space. Code restructuring involves transforming the program structure for reducing the complexity in source code and making it easier to understand.

### FUTURE ENHANCEMENT

Almost every project is subjected to change on depending on the client requirements. Since this system is subjected to change for each and every client, there is always a scope for further enhancement. The system and the architecture of the assessment system is a compatible one, so addition of new modules can be done without much difficulty.

The software is developed in visual basic which makes the system more reliable and compatible with other environments. The application proves better extensibility and flexibility for future enhancements. Any further requirement application is possible with the same feature guaranteed. It is a user –friendly system, which is very easy and convenient to use. The system is complete in the sense that it is operational and it is tested by entering data and getting reports in proper order. During the development of this project coding standards are followed for easy maintainability and extensibility. Though the new system provides a base for improving the efficiency of operations, there are lots of further enhancement that can be added to this project. Keeping this in view, provision has been made in the system to facilitate easy modification updating in future. Any modification will not affect the normal working of the system. The developed system is very interactive, coded in such a way to ensure maximum user friendliness and also allows flexibility for future. The system developed automates most needed activities in an organization. The new system can be combined with an existing system as well. More and better advanced separation system can be build on top of the proposed system as and when the need arises.This is one of the main special feature of the proposed system

## APPENDICES

### TABLE DESIGN

**Table: login**

| Field | Type | Comment |
| --- | --- | --- |
| l\_id | int(20) NOT NULL |  |
| reg\_id | varchar(50) NULL |  |
| email | varchar(100) NULL |  |
| password | varchar(100) NULL |  |
| type | varchar(100) NULL |  |

**Table: staff**

| Field | Type | Comment |
| --- | --- | --- |
| stf\_id | int(20) NOT NULL |  |
| st\_id | varchar(100) NULL |  |
| stf\_name | varchar(100) NULL |  |
| stf\_age | varchar(100) NULL |  |
| stf\_phone | varchar(100) NULL |  |
| stf\_address | varchar(100) NULL |  |
| stf\_qualification | varchar(100) NULL |  |
| stf\_experience | varchar(100) NULL |  |
| stf\_email | varchar(100) NULL |  |

**Table: station**

| Field | Type | Comment |
| --- | --- | --- |
| st\_id | int(20) NOT NULL |  |
| st\_name | varchar(100) NULL |  |
| st\_address | varchar(100) NULL |  |
| st\_l\_no | varchar(100) NULL |  |
| st\_price\_one\_kwh | varchar(100) NULL |  |
| st\_price\_per\_hour | varchar(100) NULL |  |
| st\_parking\_price | varchar(100) NULL |  |
| st\_owner | varchar(100) NULL |  |
| st\_age | varchar(100) NULL |  |
| st\_phone | varchar(100) NULL |  |
| st\_email | varchar(100) NULL |  |
| st\_status | varchar(100) NULL |  |
| st\_lat | varchar(100) NULL |  |
| st\_long | varchar(100) NULL |  |
| st\_open | varchar(100) NULL |  |
| st\_close | varchar(100) NULL |  |

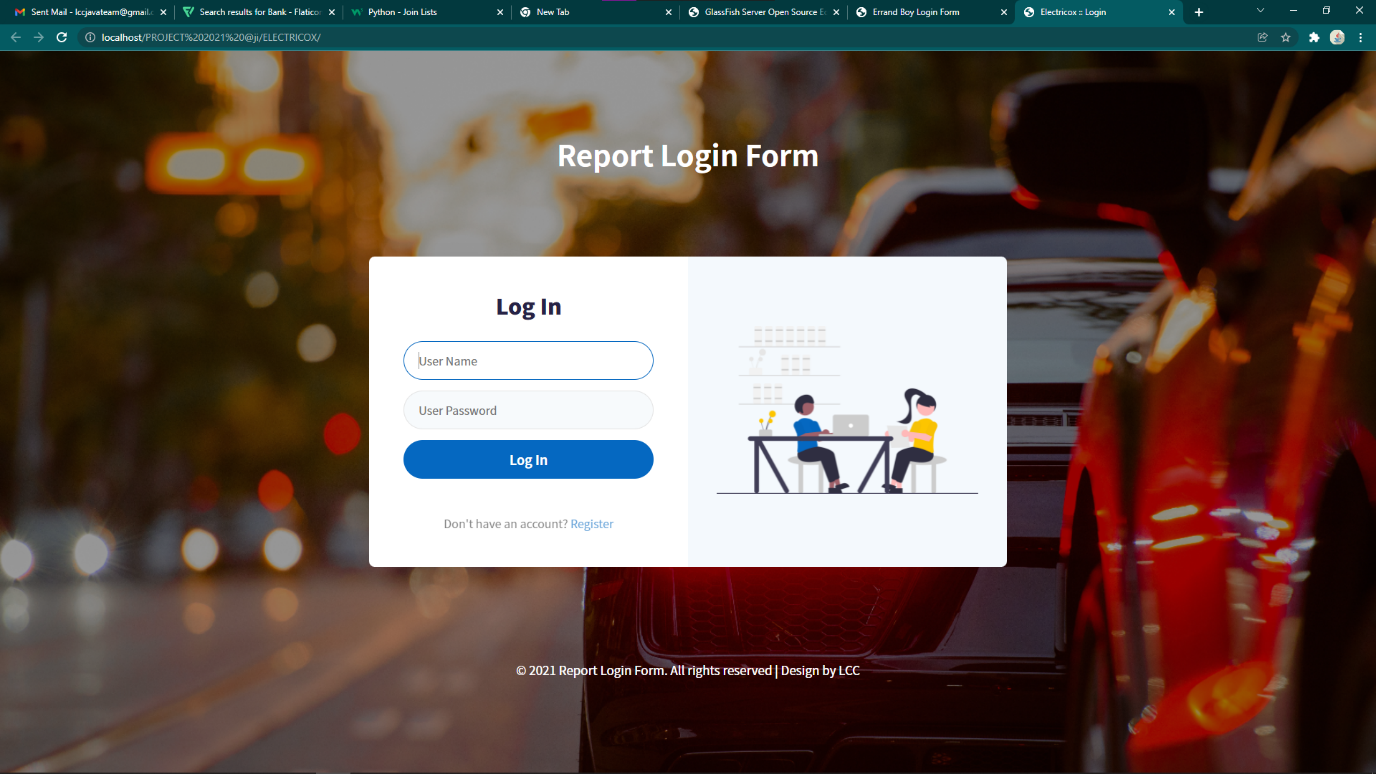
**Table: user\_bookings**

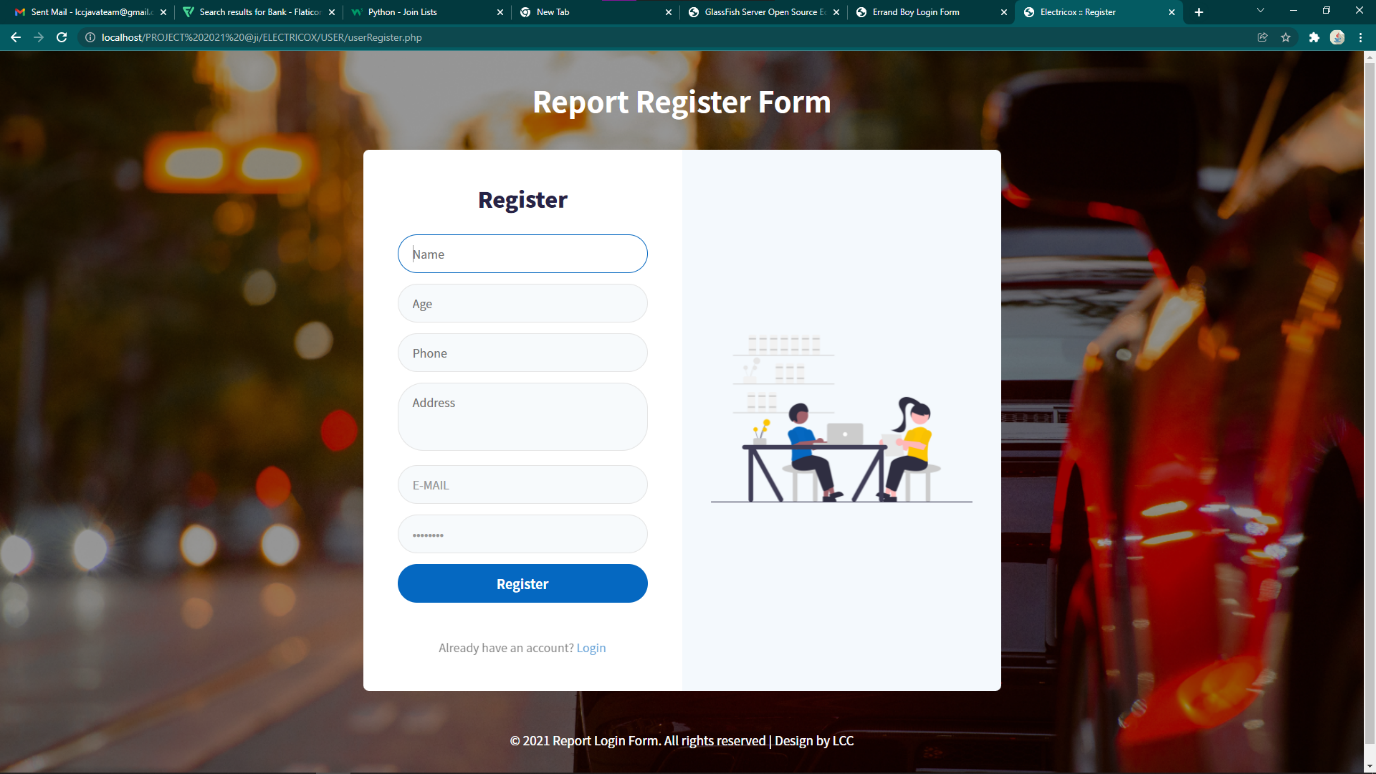
| Field | Type | Comment |
| --- | --- | --- |
| b\_rqst\_id | int(20) NOT NULL |  |
| u\_id | varchar(20) NULL |  |
| st\_id | varchar(20) NULL |  |
| stf\_id | varchar(100) NULL |  |
| status | varchar(100) NULL |  |
| date | varchar(100) NULL |  |

**Table: user\_reg**

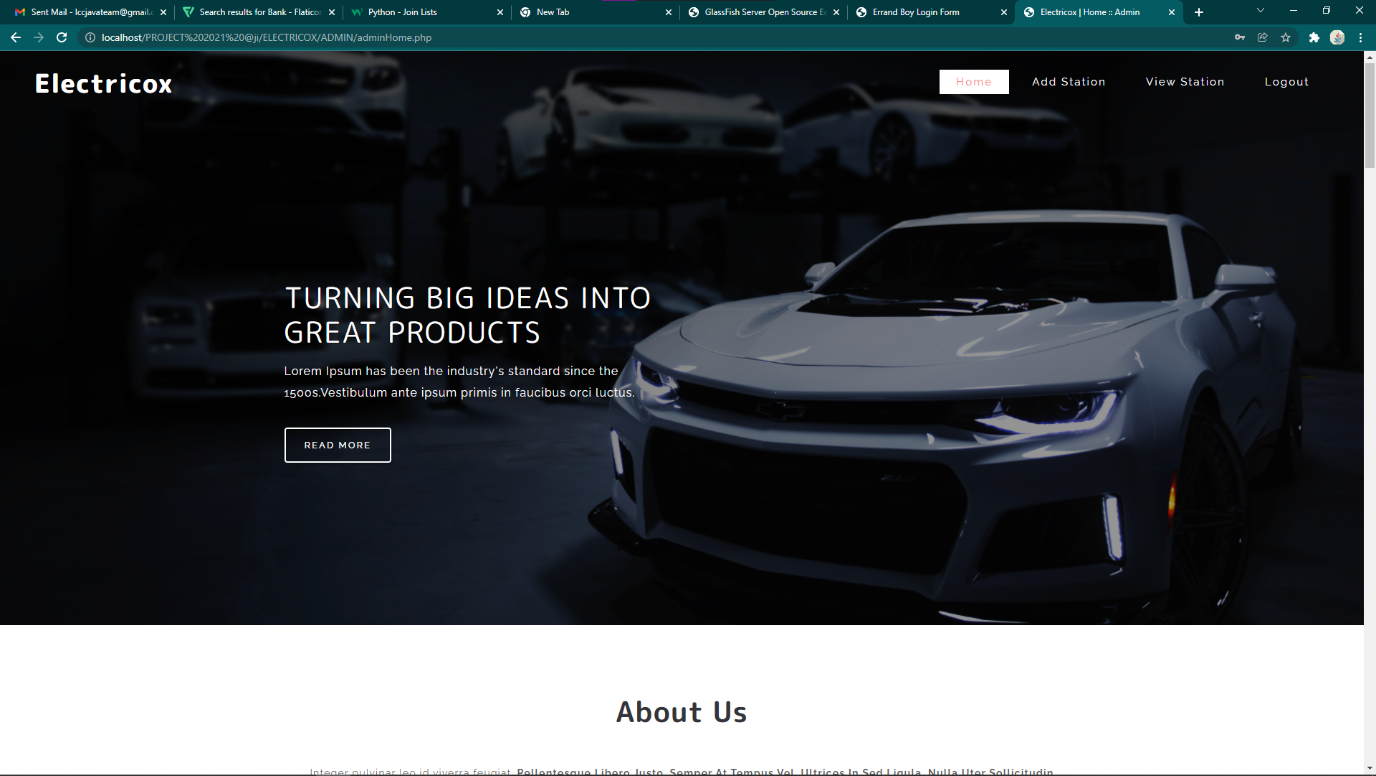
| Field | Type | Comment |
| --- | --- | --- |
| u\_id | int(20) NOT NULL |  |
| u\_name | varchar(100) NOT NULL |  |
| u\_age | varchar(100) NOT NULL |  |
| u\_phone | varchar(100) NOT NULL |  |
| u\_address | varchar(100) NOT NULL |  |
| u\_email | varchar(100) NOT NULL |  |

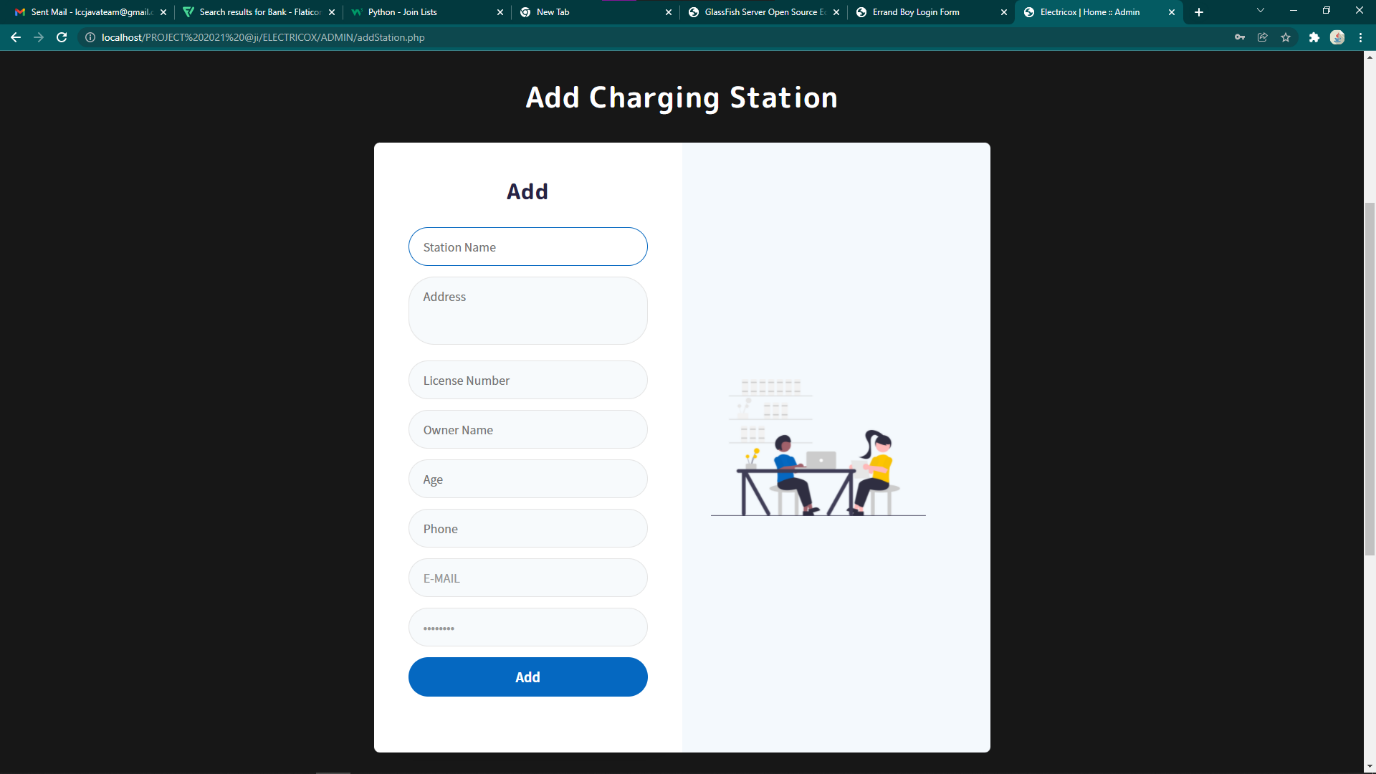
### SAMPLE INPUT SCREENS

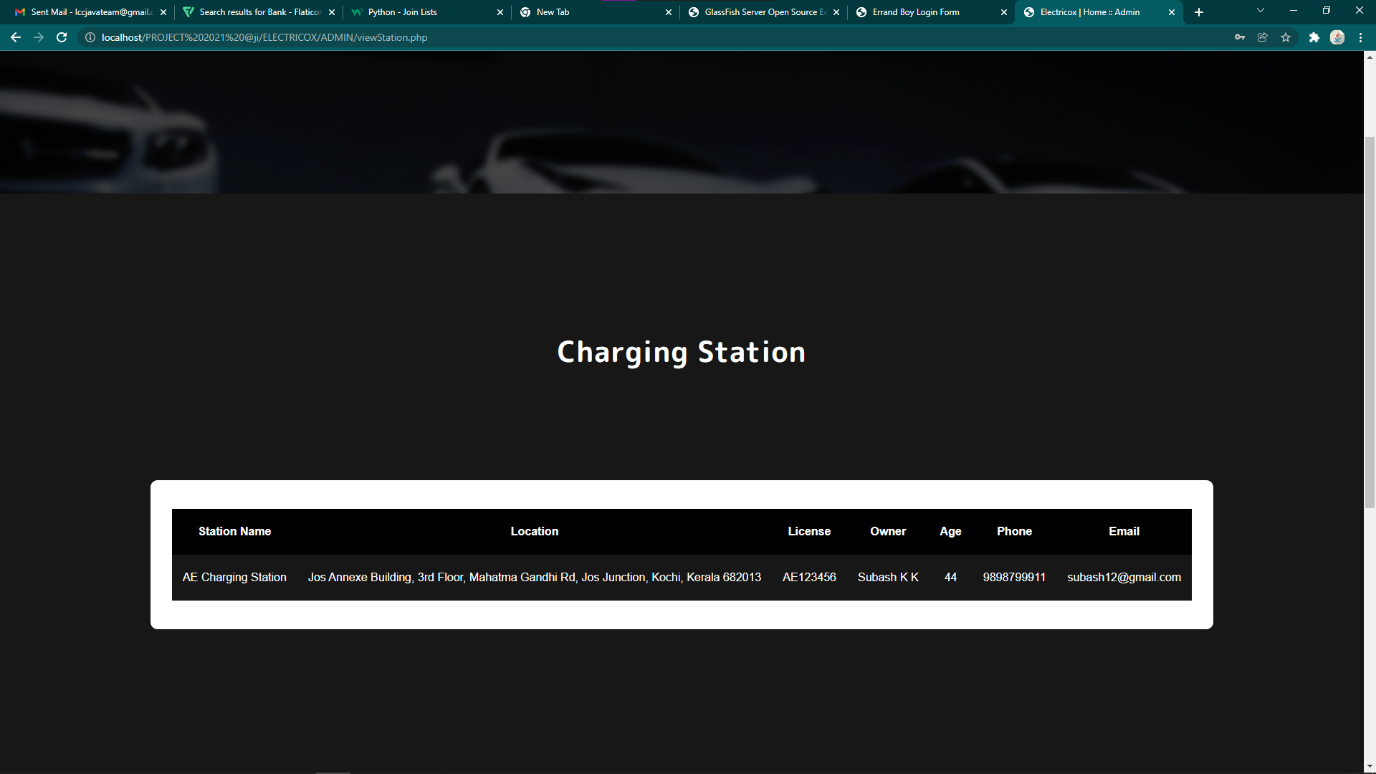
****

****

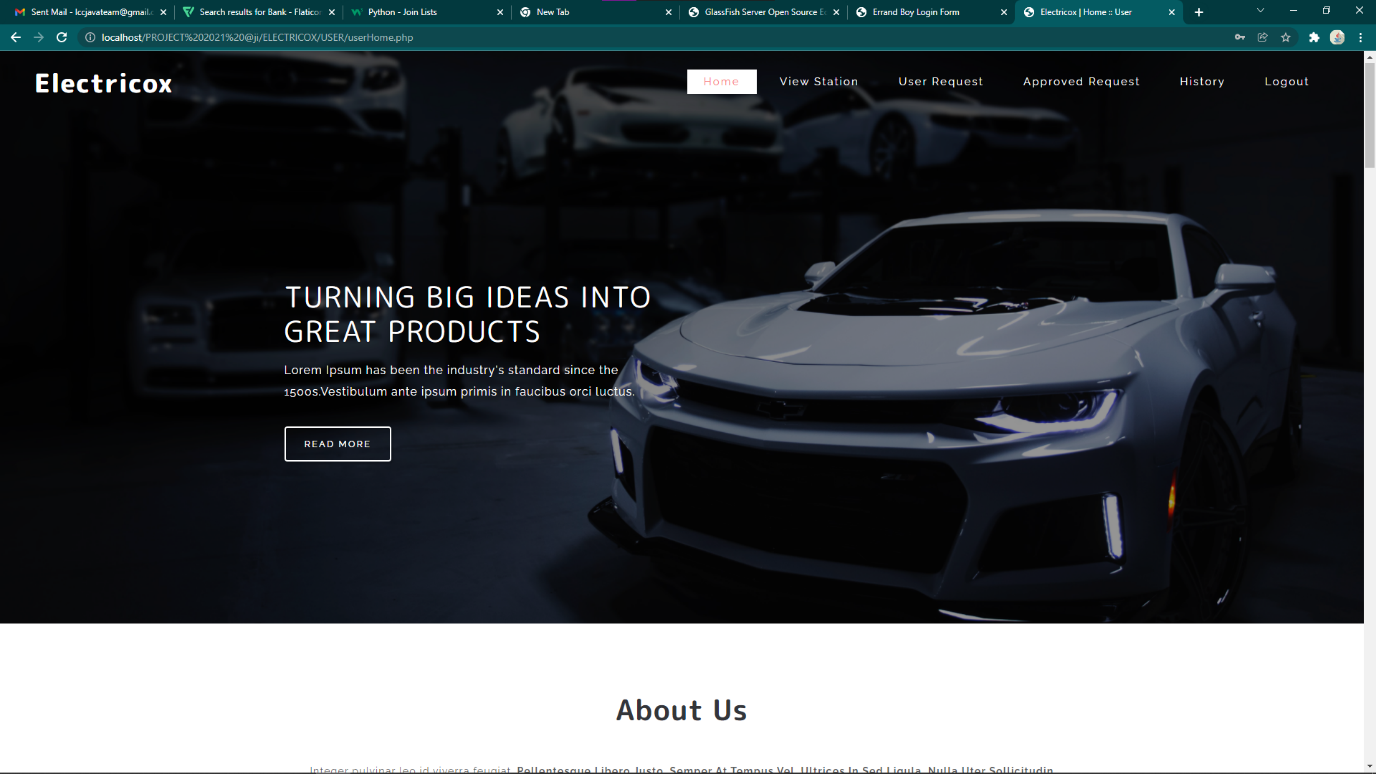
**Admin**

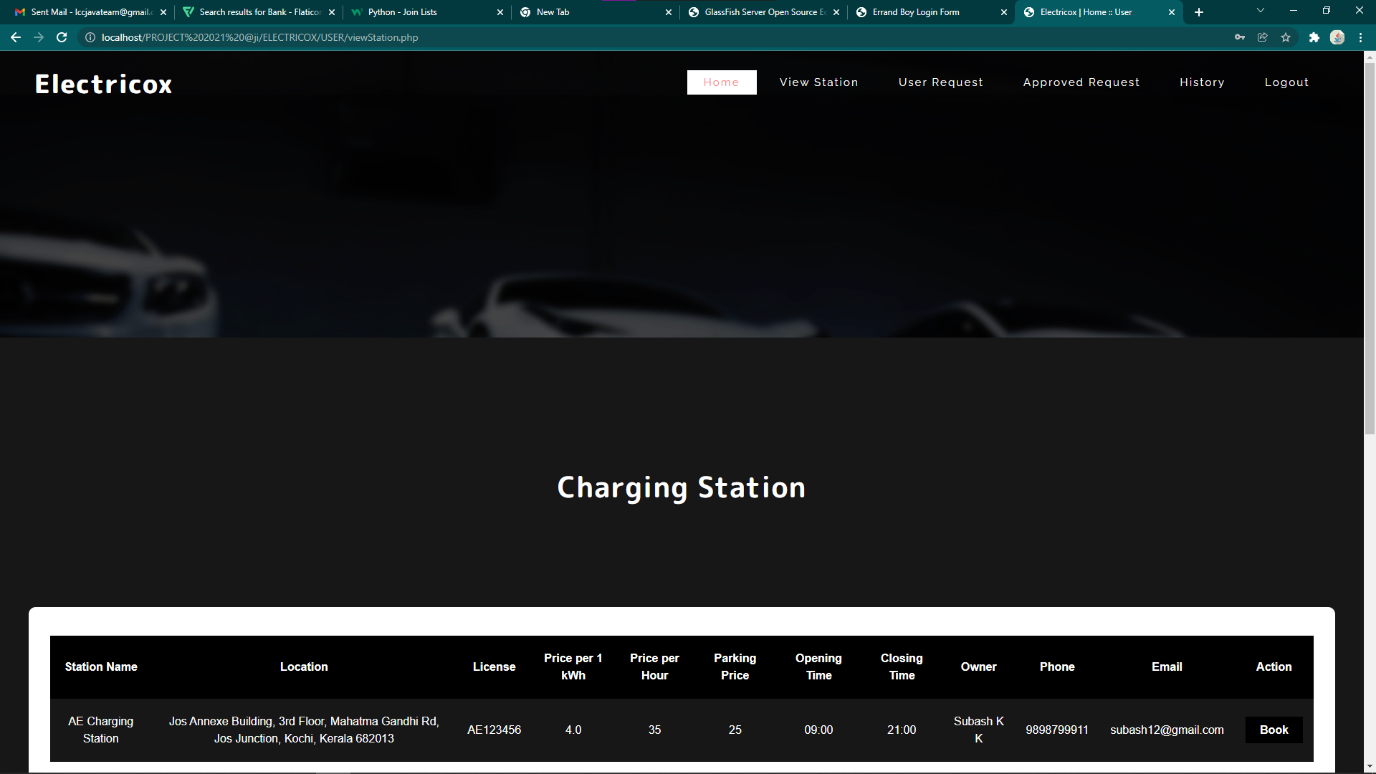
****

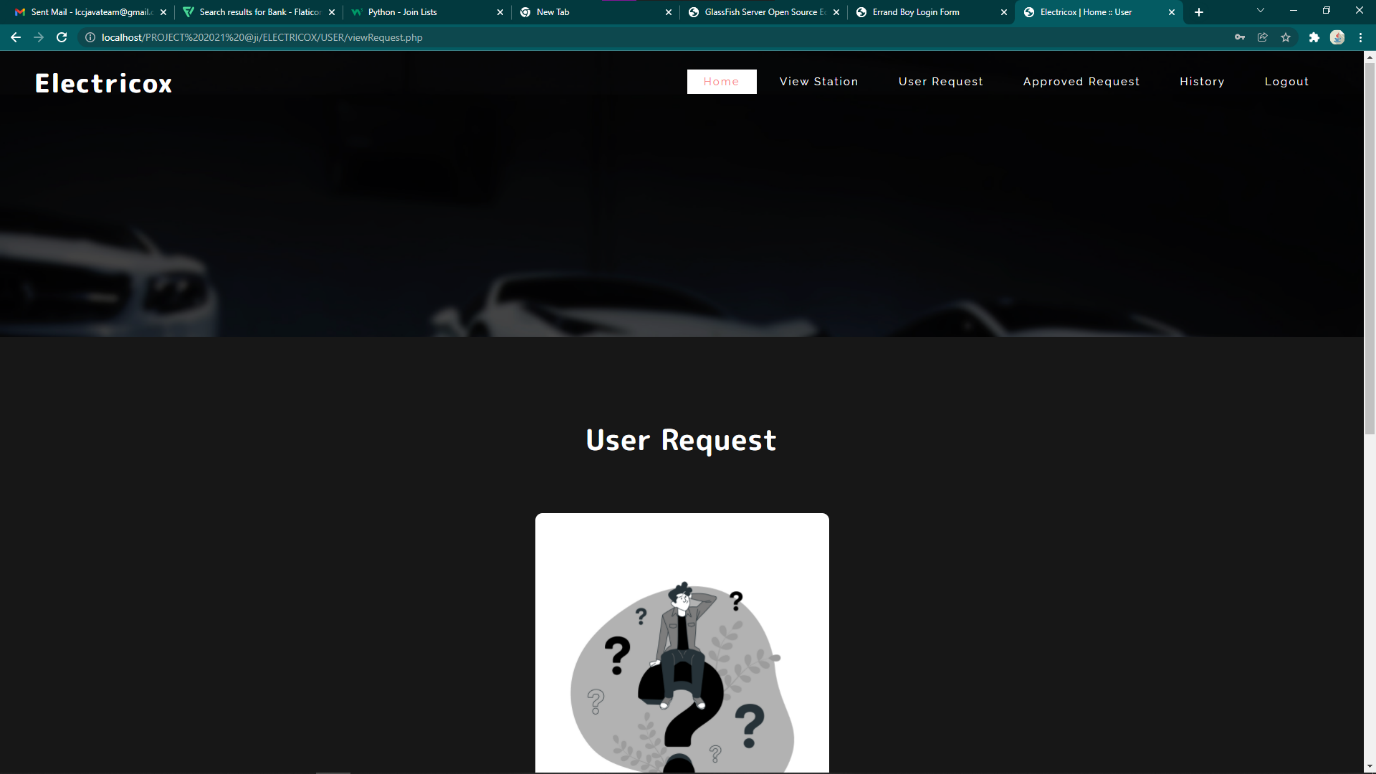
****

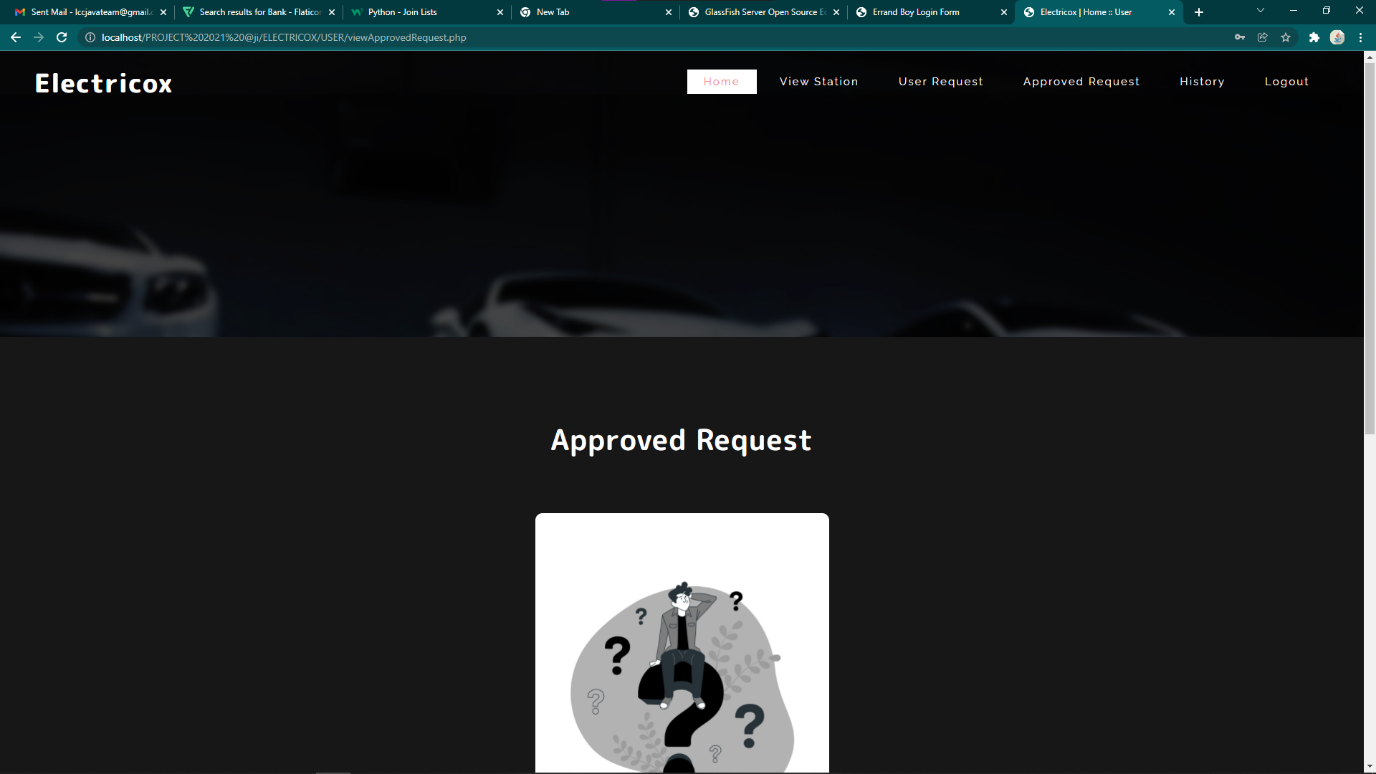
****

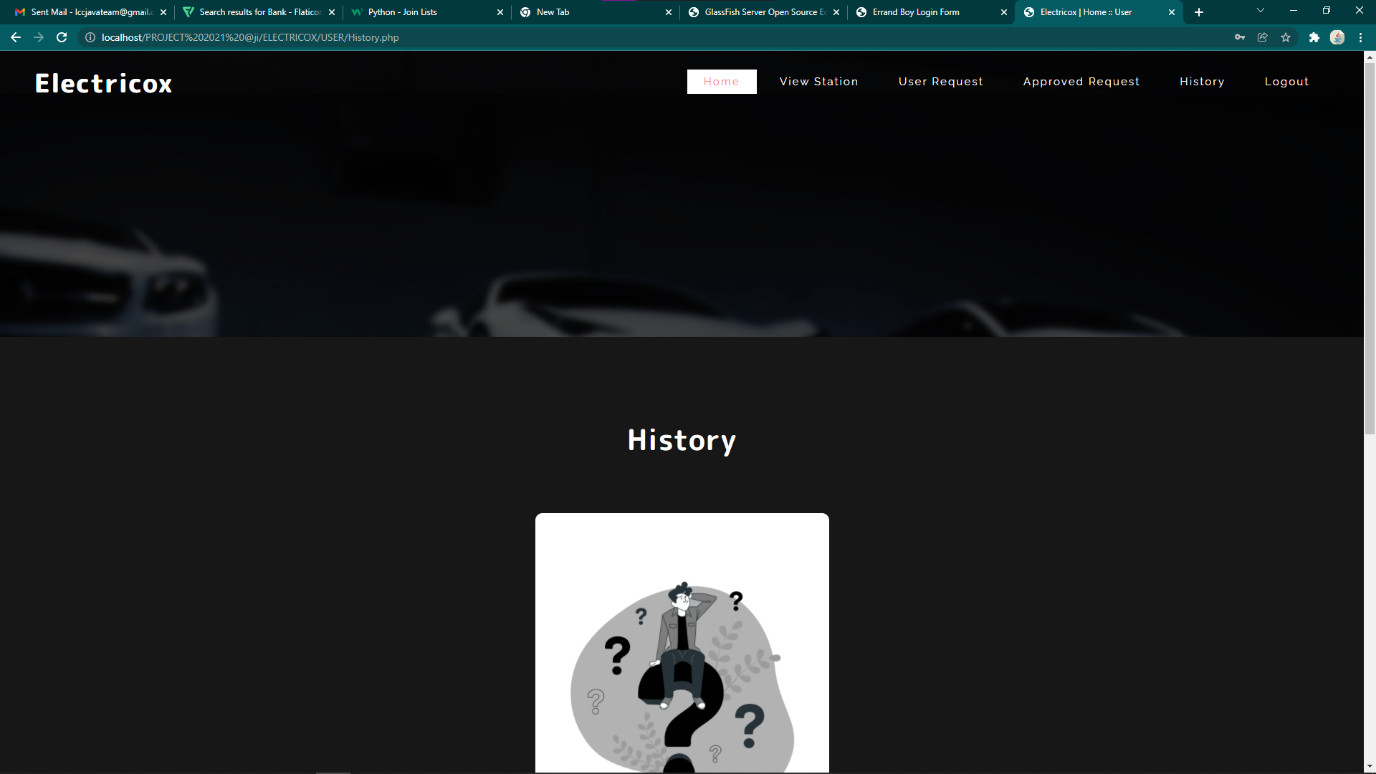
**User**

****

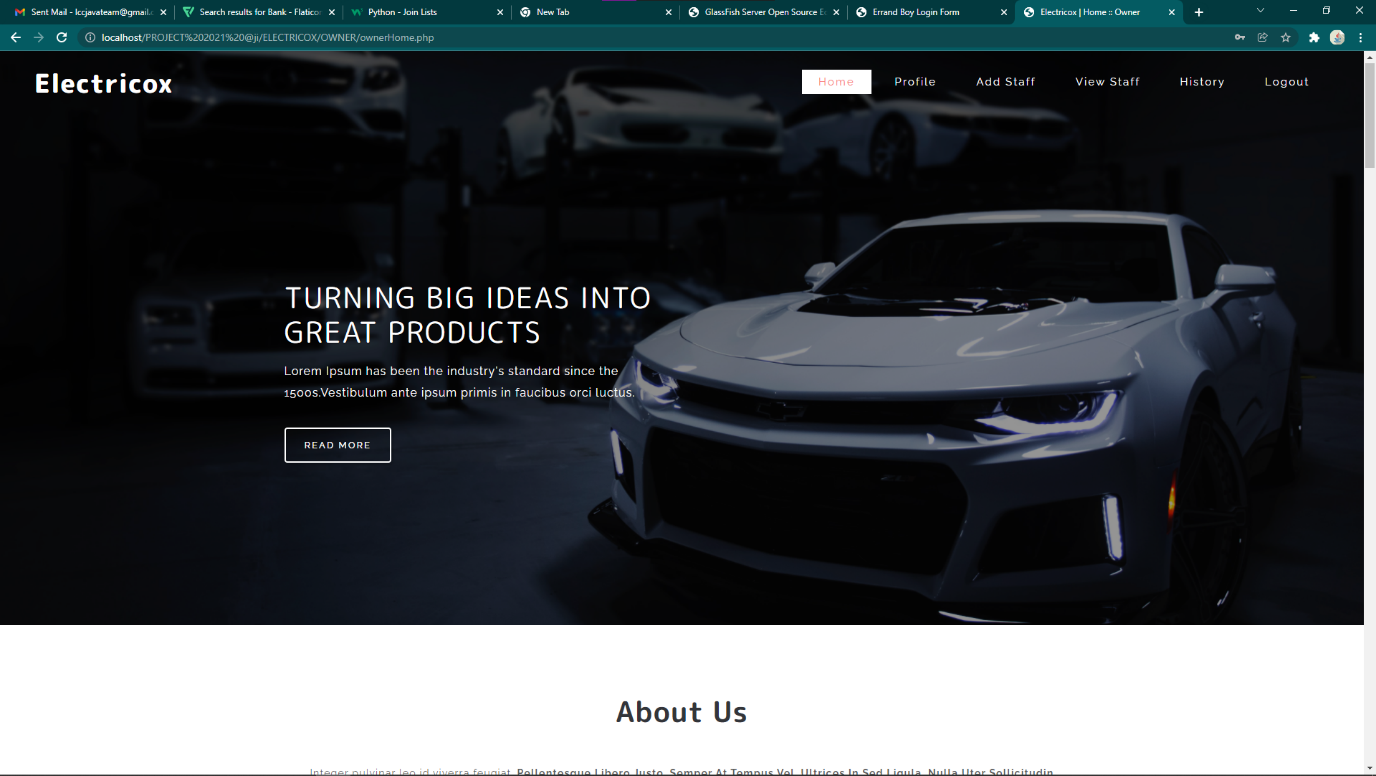
****

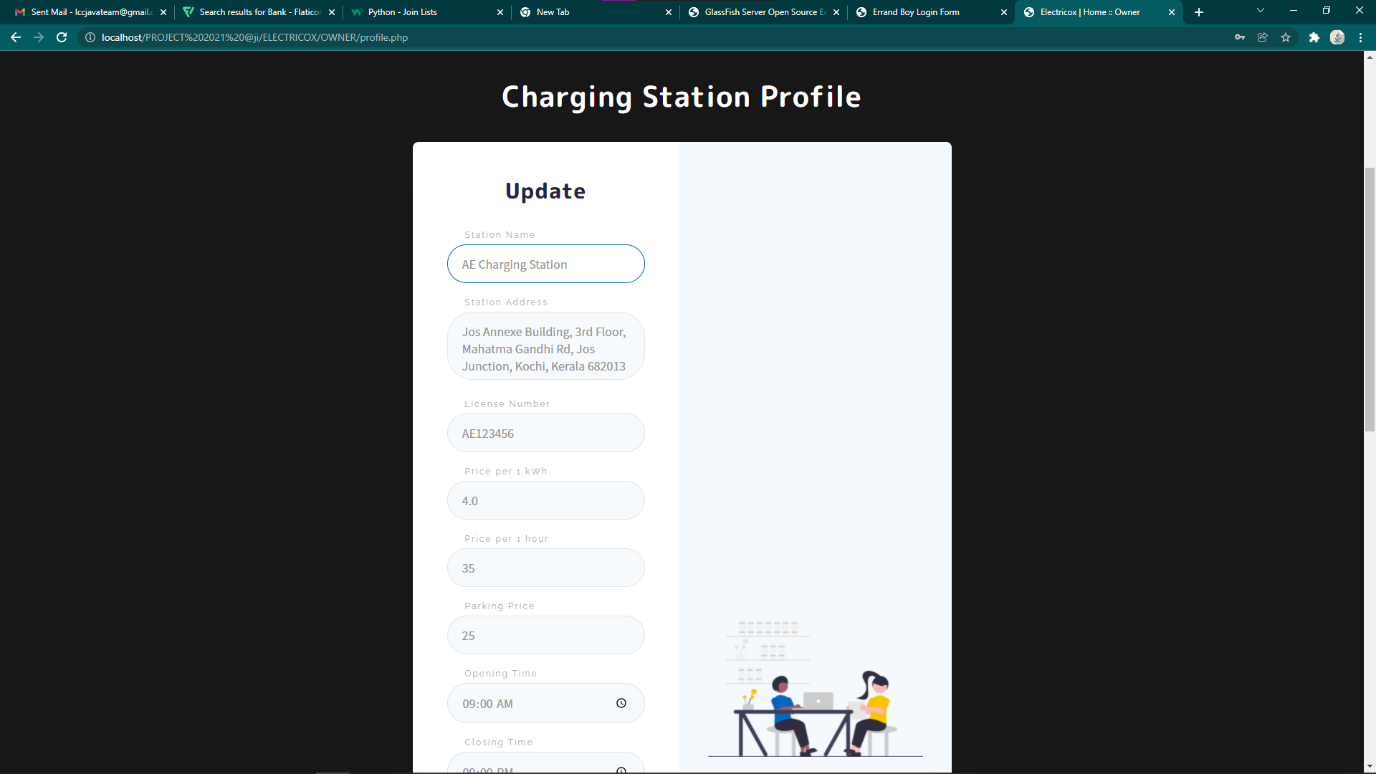
****

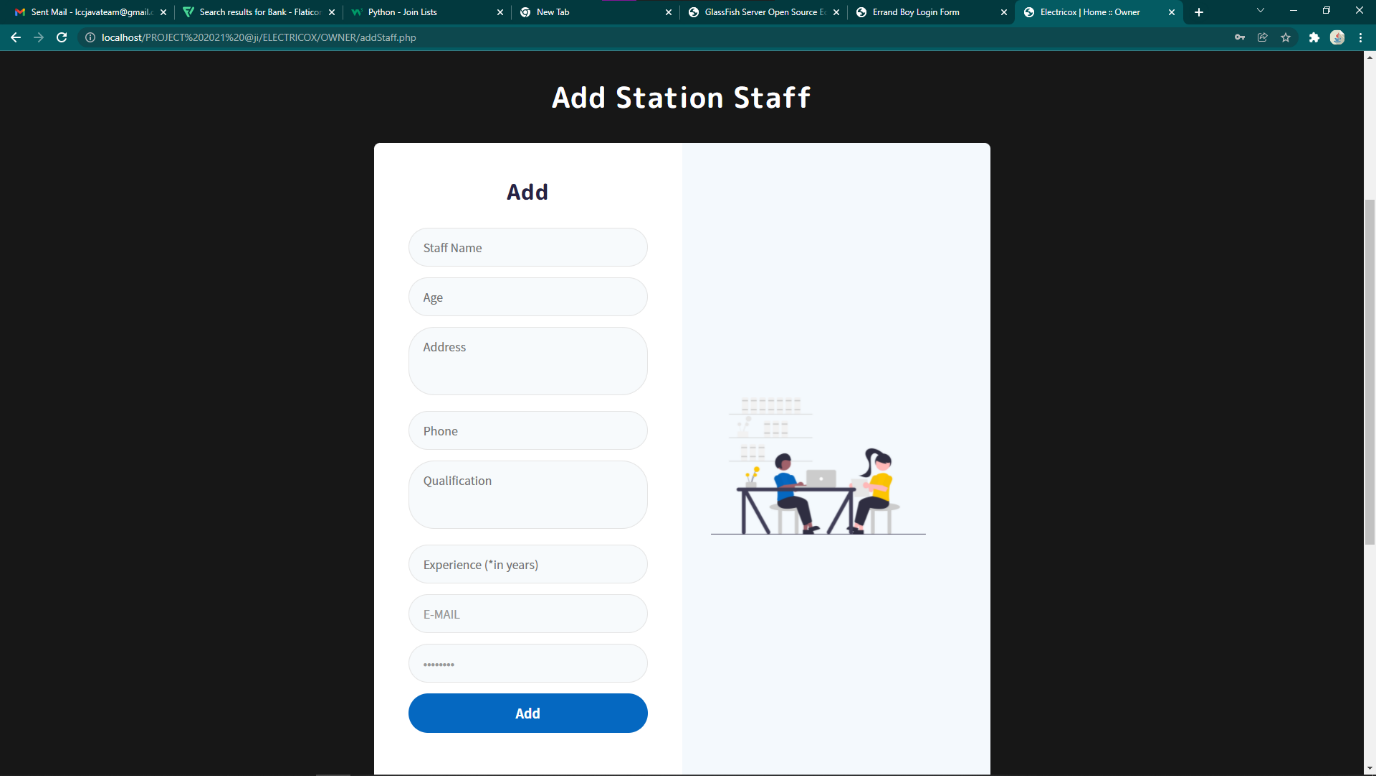
****

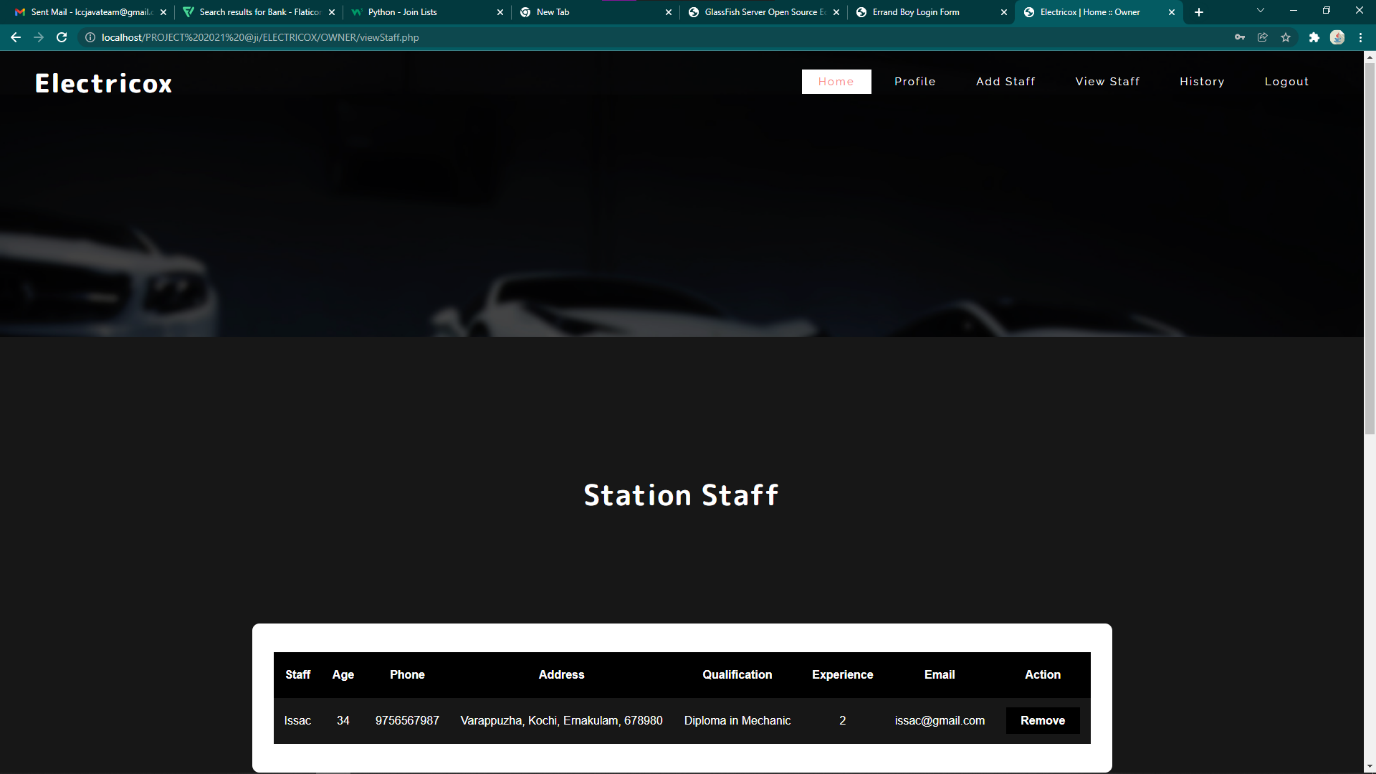
****

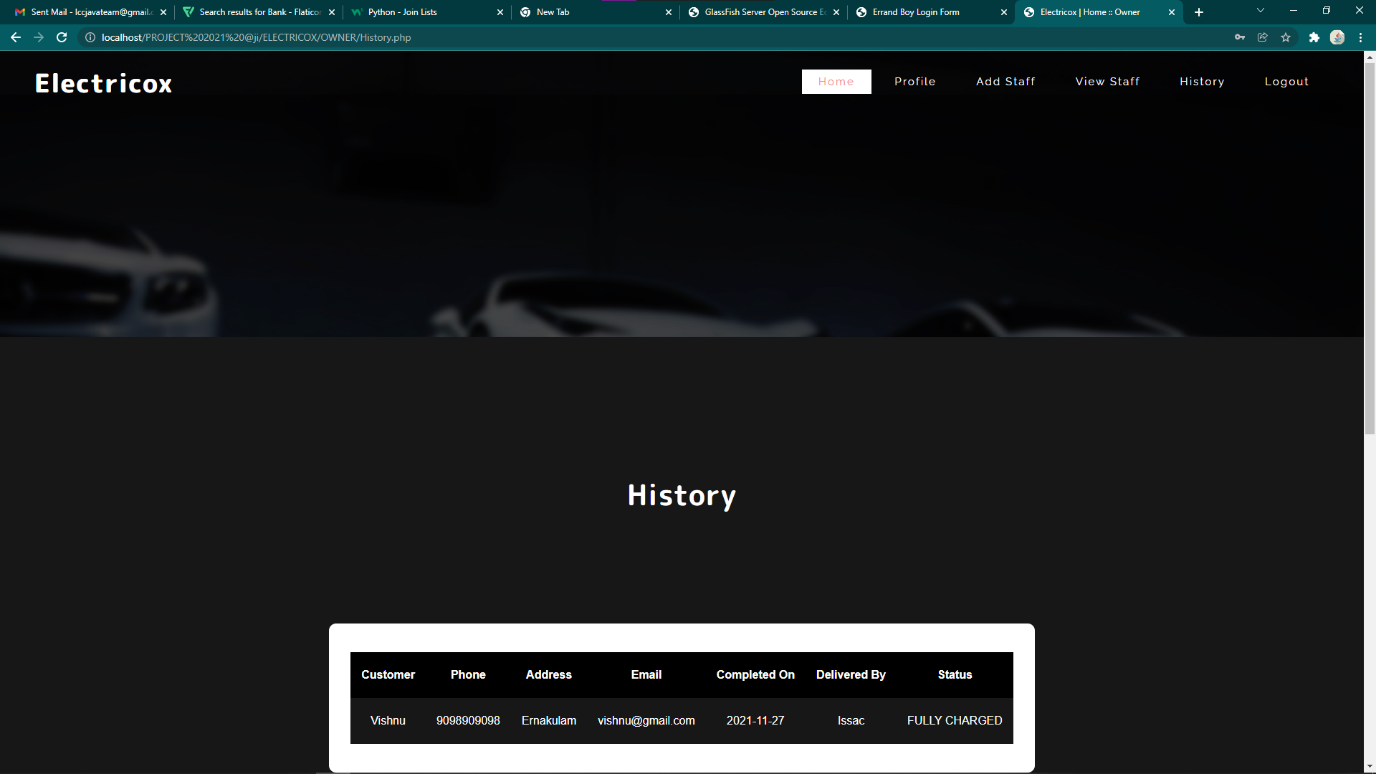
**Station**

****

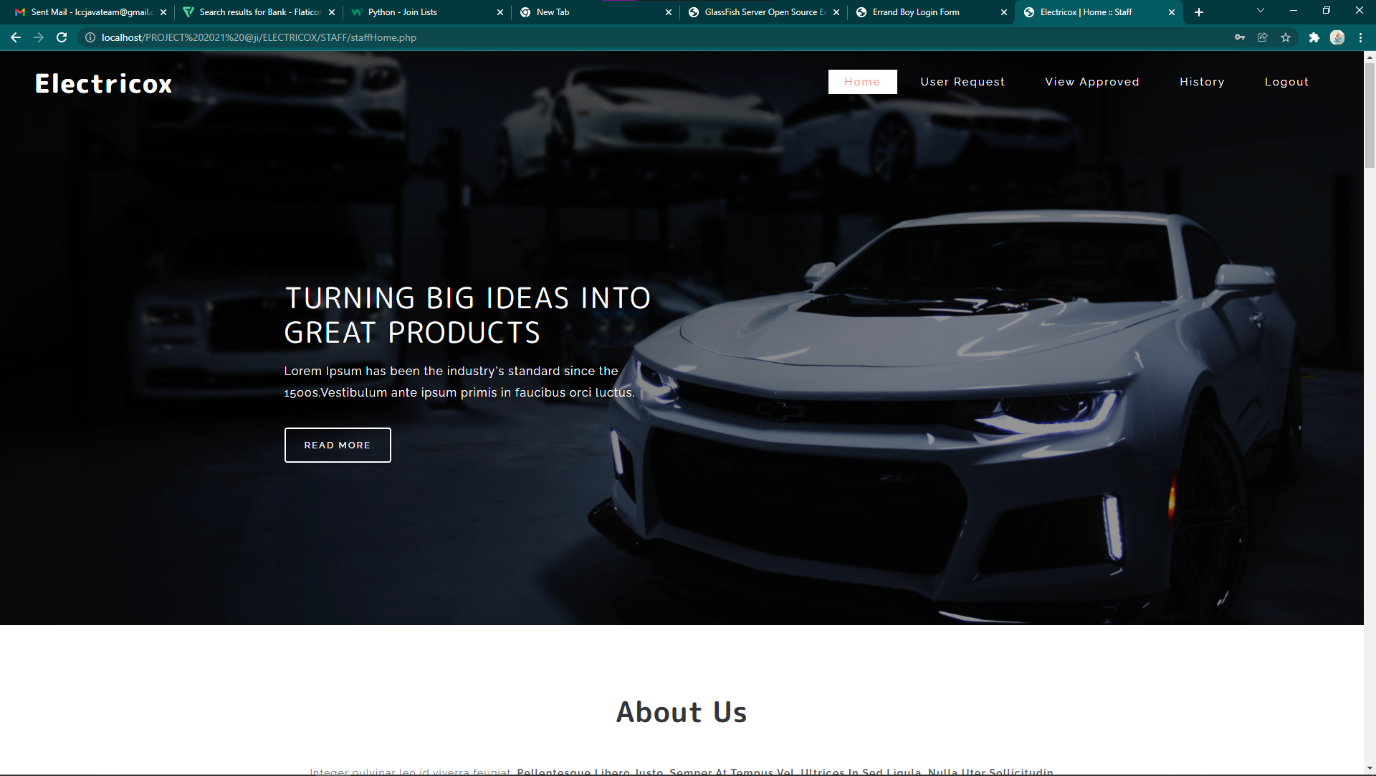
****

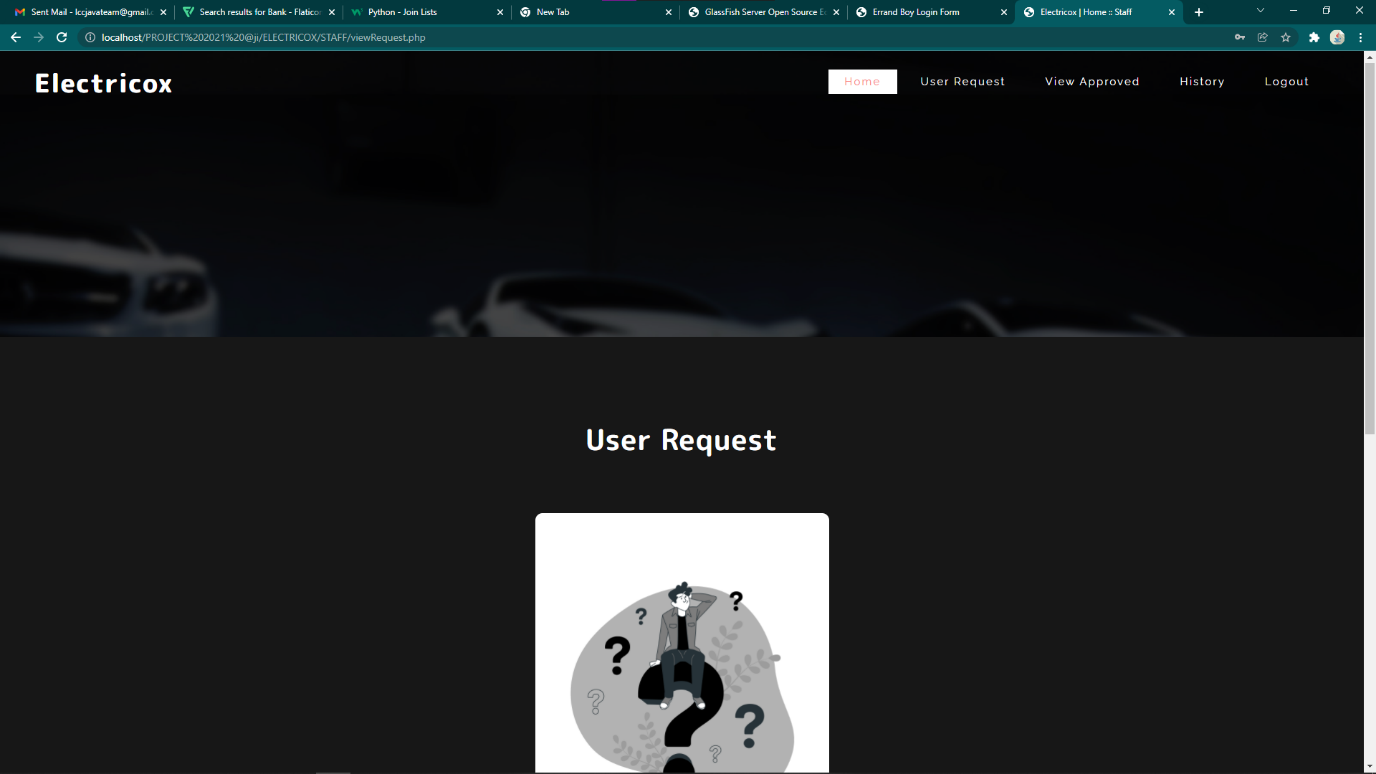
****

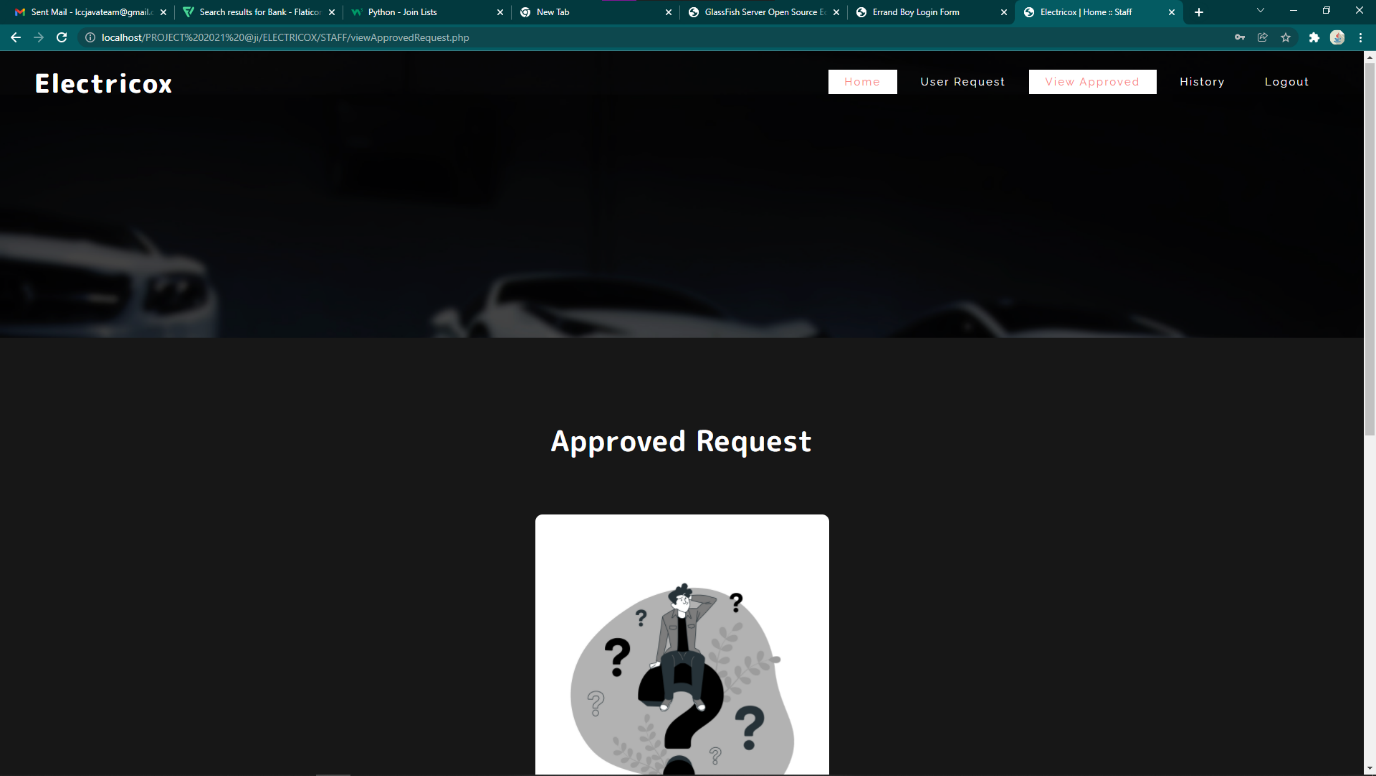
****

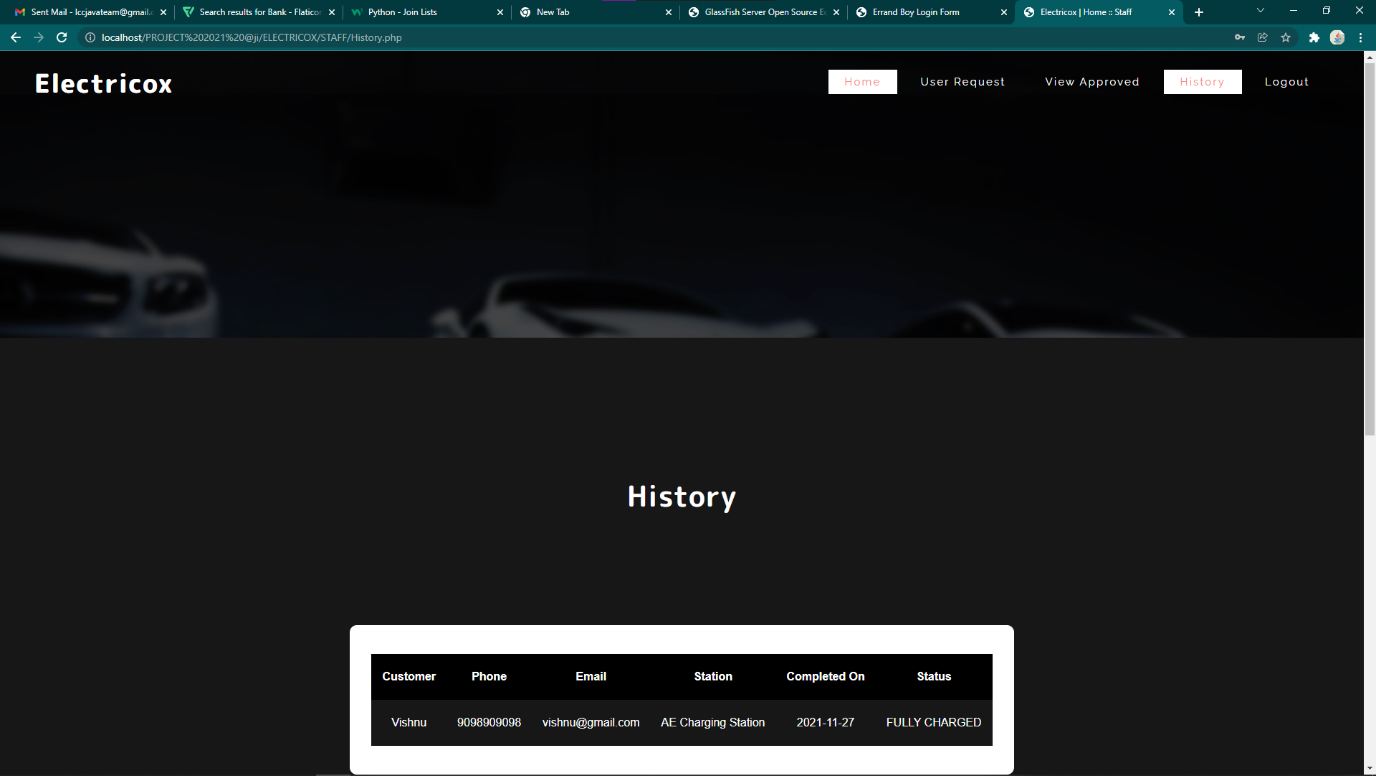
****

**Staff**

****

****

****

****

## CONCLUSION

### CONCLUSION

Charging availability was plotted relative to time of day and day of the week to show the range of percentage of EVSE connected to a vehicle over time. Charging demand was plotted to show the range of charging demand of the EVSE on the electric gird over time. In general, residential EVSE charging availability is low during the day, steadily increases in the evening, and remains high at night. Charging availability, which is a function of when individuals connect their vehicles to their EVSE, is consistent across EV Project regions.

## BIBLIOGRAPHY

### REFERENCES

Roger s pressman, software engineering - a practitioner’s approach - 7th edition, published by mcgraw-hill, 2008

Balaguruswamy, "programming in c#", tata mcgraw-hill, published 2007.

Andrew troelsen, "pro c# 2008 and the .net 3.5 platforms",

Apress, published 2008

Larman, craig, applying uml and patterns: an introduction to object-oriented analysis, pearson education, 3nd ed., 2004

Pressman, r.s., software engineering: a practitioner's approach, mghise, 7th edition, 2010