

Smart Parking System

by

Md.Naeem

ID: CSE1701010087

Sumon Ahmed

ID: CSE1701010035

Selim Reza

ID: CSE1701010074

Mahmud Hasan

ID: CSE1701010068

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SONARGAON UNIVERSITY (SU)**

October 2020

APPROVAL

The thesis and project titled “**A Comprehensive Study on Smart Parking system Uses and Preventive Measures**” submitted by Md.Naeem (CSE1701010087), Sumon Ahmed (CSE1701010035), Selim Reza(CSE1701010074) and Mahmud Hasan (CSE1701010068) to the Department of Computer Science and Engineering, Sonargaon University (SU), has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

Board of Examiners

Sadia Tasnim Barsha

Lecturer

Department of Computer Science and Engineering
Sonargaon University (SU)

Supervisor

(Examiner Name & Signature)

Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 1

(Examiner Name & Signature)

Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 2

(Examiner Name & Signature)

Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 3

DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of **Sadia Tasnim Barsha**, Lecturer, Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this Thesis and thereof has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

Signature

(Sadia Tasnim Barsha)
Supervisor

Md.Naeem
ID: CSE1701010087

Sumon Ahmed
ID: CSE1701010035

Selim Reza
ID: CSE1701010074

Mahmud Hasan
ID: CSE17010100

ABSTRACT

Humanity is currently experiencing one of the short periods of transition thanks to novel sensing solutions for smart cities that bring the future to today. Overpopulation of cities demands the development of solid strategic planning's that uses infrastructure, innovation, and technology to adapt to rapid changes. To improve mobility in cities with a larger and larger vehicle fleet, a novel sensing solution that is the cornerstone of a smart parking system, the smart vehicular presence sensor (SPIN-V, in its Spanish abbreviation), is presented. The SPIN-V is composed of a small single-board computer, distance sensor, camera, LED indicator, buzzer, and battery and devoted to obtain the status of a parking space. This smart mobility project involves three main elements, namely the SPIN-V, a mobile application, and a monitoring center, working together to monitor, control, process, and display the parking space information in real-time to the drivers. In addition, the design and implementation of the three elements of the complete architecture are presented.

ACKNOWLEDGMENT

At the very beginning, we would like to express my deepest gratitude to the Almighty Allah for giving us the ability and the strength to finish the task successfully within the schedule time.

We are very lucky that we had the kind association as well as supervision of **Sadia Tasnim Barsha**, Lecturer, Department of Computer Science and Engineering, Sonargaon University whose hearted and valuable support with best concern and direction acted as necessary recourse to carry out our project.

We would like to convey our special gratitude to **Prof. Dr. M. A Mabud**, Dean, Faculty of Science and Engineering for his kind concern and precious suggestions.

We are also thankful to all our teachers during our whole education, for exposing us to the beauty of learning.

Finally, our deepest gratitude and love to my parents for their support, encouragement, and endless love.

LIST OF ABBREVIATIONS

| | |
|-------|---|
| AJAX | Asynchronous JavaScript and XML |
| CSS | Cascading Style Sheet |
| GUI | Graphical User Interface |
| HTML | Hyper Text Markup Language |
| IOS | I-Phone Operating System |
| RDBMS | Relational Database Management System |
| SQL | Structured Query Language |
| SPS | Smart Parking System |
| UI | User Interface |
| XAMPP | Cross-platform, Apache, MySQL, PHP and Perl |

TABLE OF CONTENTS

| <u>Title</u> | <u>Page No.</u> |
|--|----------------------------|
| CHAPTER 1 | 1 – 5 |
| INTRODUCTION AND PURPOSES OF THE PROJECT | |
| 1.1 Introduction..... | 1 |
| 1.2 Objectives | 1-3 |
| 1.3 Parking Slot Problems | 3 |
| 1.4 Advantage..... | 3-5 |
| 1.5 Purpose of the Proposed System..... | 5 |
| 1.6 Project Scope | 5 |
| CHAPTER 2 | 6-7 |
| FEATURES AND FUNCTIONS OF THE PROJECT | |
| 2.1 Features of the project..... | 6 |
| 2.2 Function of the project | 6-7 |
| CHAPTER 3 | 8-20 |
| TOOLS AND METHODOLOGY | |
| 3.1 Tools..... | 8 |
| 3.1.1 MySQL | 8-10 |
| 3.1.2 Ajax | 11 |
| 3.1.3 JQuery | 11 |
| 3.1.4 Javascript | 12 |
| 3.1.5 Bootstrap | 12 |
| 3.1.6 HTML | 12 |
| 3.1.7 CSS | 13 |
| 3.1.8 PHP | 13-14 |
| 3.1.9 Photoshop | 14 |
| 3.1.10 Illustrator | 14 |

| | | |
|-------------------------------------|--|-------|
| 3.1.11 | Adobe Dreamweaver | 14 |
| 3.1.12 | SQLyog | 14-16 |
| 3.3 | Server Specification..... | 16 |
| 3.3.1 | XAMPP..... | 17-20 |
| CHAPTER 4 | | 21-27 |
| LOGICAL DESIGN AND SYSTEM WORK FLOW | | |
| 4.1 | Work Flow Diagram | 21 |
| 4.2 | Database Schema | 22 |
| 4.3 | Database Table | 22-26 |
| 4.4 | Database function and stored procedure | 27 |
| CHAPTER 5 | | 28-37 |
| SYSTEM AND INTERFACE DESIGN | | |
| 5.1 | System Design | 28 |
| 5.2 | File and folder organogram..... | 28-29 |
| 5.3 | System menu..... | 29-30 |
| 5.4 | System Home Page..... | 31 |
| 5.5 | System design snapshots | 32-35 |
| 5.6 | Report snapshots | 36-37 |
| CHAPTER 6 | | 38-41 |
| SECURITY AND IMPLEMENTATION | | |
| 6.1 | System Security..... | 38 |
| 6.1.1 | User Management..... | 38 |
| 6.1.2 | User Authentication..... | 38 |
| 6.1.3 | Admin Authentication..... | 38 |
| 6.1.4 | Pages Permission..... | 38 |
| 6.1.5 | Change Password..... | 38 |
| 6.2 | System Requirements..... | 38 |
| 6.2.1 | Minimum Hardware Requirements..... | 39 |

| | | |
|------------------------------------|------------------------------------|-------|
| 6.2.2 | Minimum Software Requirements..... | 39 |
| 6.3 | Result of analysis | 40 |
| 6.3.1 | Results of Implementation | 40 |
| 6.3.2 | Results of testing | 40 |
| 6.3.3 | Achievements of the project | 40-41 |
| CHAPTER 7 | | 42-43 |
| CONCLUSION AND FUTURE WORKS | | |
| 7.1 | Introduction | 42 |
| 7.2 | Recommendation | 42 |
| 7.3 | Scope of Further Development | 42 |
| 7.4 | Conclusion | 43 |
| REFERENCES | | 44 |

LIST OF TABLES

| <u>Table No.</u> | <u>Title</u> | <u>Page No.</u> |
|-------------------------|-------------------------------|------------------------|
| Table 4.3.1 | Customer Type | 22 |
| Table 4.3.2 | Customer Info | 23 |
| Table 4.3.3 | Customer Vehicle Info | 23 |
| Table 4.3.4 | Floor Info | 23-24 |
| Table 4.3.5 | Parking Slot Info | 24 |
| Table 4.3.6 | Vehicle Type | 24 |
| Table 4.3.7 | Parking Info | 25 |
| Table 4.3.8 | Charge Info | 25 |
| Table 4.3.9 | Payment Received | 26 |
| Table 6.2.1 | Minimum Hardware Requirements | 39 |
| Table 6.2.2 | Minimum Software Requirements | 39 |

LIST OF FIGURES

| <u>Figure No.</u> | <u>Title</u> | <u>Page No.</u> |
|--------------------------|--------------------------------------|------------------------|
| Fig.4.1 | Work Flow Diagram | 21 |
| Fig.4.2 | Database Schema | 22 |
| Fig.5.4 | Home page | 31 |
| Fig.5.5.1 | Create System User | 32 |
| Fig.5.5.2 | User Permission | 32 |
| Fig.5.5.3 | Customer Info Entry | 33 |
| Fig.5.5.4 | Manual Parking Info | 33 |
| Fig.5.5.5 | Manual Parking Out | 34 |
| Fig.5.5.6 | Cash Receive | 34 |
| Fig.5.5.7 | Real Time Parking Slot | 35 |
| Fig.5.6.1 | Customer Ledger | 36 |
| Fig.5.6.2 | All Customer Summary | 36 |
| Fig.5.6.3 | Day Wise Parking Report | 36 |
| Fig.5.6.4 | Month Wise Parking Details | 37 |
| Fig.5.6.5 | Day Wise Charge Collection Details | 37 |
| Fig.5.6.6 | Month Wise Charge Collection Details | 37 |

CHAPTER 1

INTRODUCTION & PURPOSES OF THE PROJECT

1.1 Introduction

As an important component of traffic system, parking management system is playing an important role and affecting people's daily life. By detecting and processing the information from parking lots, smart parking system allows drivers to obtain real-time parking information and alleviates parking contentions. The use of intelligent systems have become the most prevalent among the world contribute to the implementation of daily business in a more efficient and flexible reveal. In this research project, we study state-of-the-art parking policies in smart parking systems, and show that the smart parking system needs to be "smarter". Our design goals of the smart parking systems include: simplify the operations of parking systems, improve drivers' satisfaction, increase parking revenue, and alleviate traffic congestion. Moreover, we propose a reservation-based system, built on advanced sensing and web based communication technology, with the objectives to alleviate the parking contention, balance the benefits between parking service providers and drivers, coordinate among service providers, differentiate the needs of individual drivers, and reduce the amount of traffic searching for parking as well. To achieve the design goals, a powerful tool to model the behavior of both service providers and drivers is required. Meanwhile, we need to design control signals to guide the parking selection of large scale, autonomous drivers.

1.2 Objectives

- To develop an intelligent, user friendly automated car parking system which reduces the manpower and traffic congestion.
- To offer safe and secure parking slots within limited area.
- To minimize the effects of on street parking upon road safety and congestion.
- To reduce dependence on the car, particularly in town centers.
- To help maintain the vitality of town centers and to minimize out of town developments.
- To ensure that car parking provision and enforcement are broadly self-financing through the Decriminalization of Parking Enforcement (DPE) process.
- To reduce, where possible, environmental damage caused by cars and car ownership, particularly in residential areas.

- To reduce, where possible, competition for road space, between residents and other groups.
- If through town center regeneration and development more shoppers/visitors are attracted to the town centers surrounding roads will become increasingly congested. Some reduction in potential congestion may be achieved by encouraging through traffic to use other routes and by assessing the design and access to developments in terms of minimum access through the town center core.
- To ensure that any on street parking close to town centers is of short duration to ensure reasonable customer turnover for shops and commercial premises and to prevent longer stay parking by nonusers of town center facilities. That the duration of stay is lengthened commensurate with the distance from the commercial core of the towns; to implement schemes to deal with local on street parking "issues", in an agreed priority order based upon objective criteria, for locations where genuine road safety, congestion or residents problems occur.
- To monitor the effects of implemented schemes on adjacent areas with subsequent action only being taken where the same criteria are met.
- Where residents compete for road parking space in their own streets with other groups (local workers, commuters etc.) new schemes will give greater priority to residents.
- Where local businesses compete for road parking space in their local industrial areas with other groups, (commuters etc.) new schemes will give greater priority to local businesses.
- To keep the implementation of controlled parking zones and application for special parking area status (via the highway authority) under review, in association with the highway authority and the police.
- To maintain a priority list for the provision of parking spaces in areas where residents' parking is causing environmental damage and to review the priority list on an annual basis.
- To produce and implement an annual program of parking provision utilizing the available budgets, drawing locations for investigation and action from the above-mentioned priority list.

1.3 Parking Slot Problems

- Difficulty in Finding Vacant Spaces Quickly finding a vacant space in a multilevel parking lot is difficult if not impossible, especially on weekends or public holidays, Finding spaces during weekends or public holidays can take more than 10 minutes for about 66% of visitors. Stadiums or shopping malls are crowded at peak periods, and difficulty in finding vacant slots at these places is a major problem for customers. Insufficient car park spaces \lead to traffic congestion and driver frustration.
- If a car is parked in such a way that it occupies two parking slots rather than one, this is called improper parking .Improper parking can happen when a driver is not careful about another driver’s rights. This is tackled by the development of automated smart car parking system.

1.4 Advantage

Smart Parking involves the use of low cost sensors, real-time data and applications that allow users to monitor available and unavailable parking spots. The goal is to automate and decrease time spent manually searching for the optimal parking floor, spot and even lot. Some solutions will encompass a complete suite of services such as online payments, parking time notifications and even car searching functionalities for very large lots. A parking solution can greatly benefit both the user and the lot owner. Here are some of the top benefits:

- **Optimized parking** – Users find the best spot available, saving time, resources and effort. The parking lot fills up efficiently and space can be utilized properly by commercial and corporate entities.
- **Reduced traffic** – Traffic flow increases as fewer cars are required to drive around in search of an open parking space.
- **Reduced pollution** – Searching for parking burns around one million barrels of oil a day. An optimal parking solution will significantly decrease driving time, thus lowering the amount of daily vehicle emissions and ultimately reducing the global environmental footprint.
- **Enhanced User Experience** – A smart parking solution will integrate the entire user experience into a unified action. Driver’s payment, spot identification,

location search and time notifications all seamlessly become part of the destination arrival process.

- **New Revenue Streams** – Many new revenue streams are possible with smart parking technology. For example, lot owners can enable tiered payment options dependent on parking space location. Also, reward programs can be integrated into existing models to encourage repeat users.
- **Integrated Payments and POS** – Returning users can replace daily, manual cash payments with account invoicing and application payments from their phone. This could also enable customer loyalty programs and valuable user feedback.
- **Increased Safety** – Parking lot employees and security guards contain real-time lot data that can help prevent parking violations and suspicious activity. License plate recognition cameras can gather pertinent footage. Also, decreased spot-searching traffic on the streets can reduce accidents caused by the distraction of searching for parking.
- **Real-Time Data and Trend Insight** – Over time, a smart parking solution can produce data that uncovers correlations and trends of users and lots. These trends can prove to be invaluable to lot owners as to how to make adjustments and improvements to drivers.
- **Decreased Management Costs** – More automation and less manual activity saves on labor cost and resource exhaustion.
- **Increased Service and Brand Image** – A seamless experience can really skyrocket a corporate or commercial entities brand image to the user. Whether the destination is a retail store, an airport or a corporate business office, visitors will surely be impressed with the cutting edge technology and convenience factors.

1.5 Purpose of the Proposed System

- Smart Parking System is a unique parking management solution. Suitable for all types of parking areas, it digitizes end-to end parking processes including multi-tenant, multi-level parking.
- It is integrated with visitor management systems and access control hardware. Extremely useful for shared parking spaces, the solution automates day-to-day processes such as auto-identifying appropriate parking slots - be it reserved or pay-and-park, auto-generating parking tickets, levying penalties, and many more.

- Most parking spaces are manually managed. This makes managing multi-tenant, multi-level car parking chaotic - a headache for parking lot owners and providers.
- Smart Parking system digitizes the end-to-end management of parking spaces and all parking processes related to visitors, vehicles, & payments.
- It eliminates manual monitoring and tracking of vehicles. With Smart Parking system, you can define and customize areas like paid, visitor, staff, tenants, etc.
- Allocation of slots becomes easy. Live countdown of occupied versus unoccupied slots, filled versus unfilled slots, etc. can be displayed on a LED screen and on the phone.

1.6 Project Scope

As we all know that the world is advancing day new technologies come and go, many new methods are been introduced almost daily, therefore, the demand of the new systems have been increased in every organization old systems have been replaced by new systems, the question is why? Its because every organization demands excellent quality work more profit to be earned in short span of time, to gain as much market trust as they can, and not to forget as quick they can so they can complete with other organizations.

CHAPTER 2

FEATURES AND FUNCTIONS OF THE PROJECT

2.1 Features of the Project

- Real-time monitoring of parking space
- Real-time monitoring of all floor
- Parking space optimizing
- **Track All Vehicles:** Know when vehicles enter and exit, along with the dwelling time.
- **Deep Analytics:** View vehicle's history, identify repeat offenders, and more!
- **Custom Rules:** Create your own rules, tag vehicles as authorized or blacklist, and more!
- **Customize spaces:** Customize & reserve parking space as per company, staff, pay-&-park, etc.

- **Slots per vehicle type** : Allocate slots as per two-wheeler and four-wheeler vehicles to utilize all available space.
- **View live reports:** No more manual checking of empty space, view it on your phone/application along with reports.
- **Auto-generated token:** QR code-based token are instantly auto-generated for a visitor.
- **On spot or monthly pay** : Parking passes generated can be on pay-&-park or a monthly pay basis.
- **Access Control System** : Integrate parking token with access control system – scan number plate using mobile, manually, etc.

2.2 Functions of the Project

- **User management:** Create unlimited user and manage all user access. Also control their activity.
- **User permission controlling:** Menu permissions can be given to the user as desired and it is possible to monitor which user has been given which menu permission.
- **Vehicle controlling:** What types of vehicles can enter the parking lot can be controlled.
- **Parking slot controlling:** Real time can be seen in some parking slots are open and full.
- **Customer controlling:** The customer can be controlled through this functionality. And the customer can be kept inactive or activated if desired.
- **Customer payment system controlling:** Daily, monthly and yearly payments can be taken from the customer.

CHAPTER 3

TOOLS AND METHODOLOGY

3.1 Required Tools:

- This project is a web application that is developed for cross platform.
- Database Design (MySQL)
- Front end Design (Ajax, JQuery, Javascript, Bootstrap, HTML5 And CSS3)
- Back end Design (Customize PHP Framework)
- Design purpose Photoshop and Illustrator
- IDE : Adobe Dreamweaver
- GUI tool for the RDBMS MySQL : *SQLyog*

3.1.1 MySQL

What is MySQL?

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

It is developed, marketed, and supported by MySQL AB, a Swedish company, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is My Ess Que Ell. However, you can pronounce it in your way. Many small and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

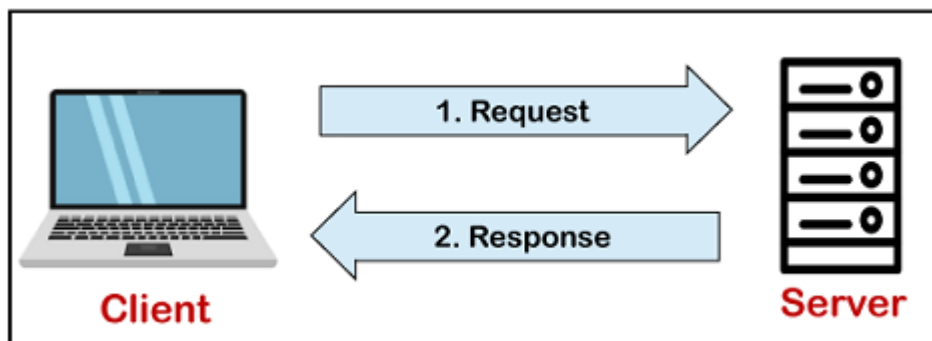
MySQL is a Relational Database Management System (RDBMS) software that provides many things, which are as follows:

- It allows us to implement database operations on tables, rows, columns, and indexes.
- It defines the database relationship in the form of tables (collection of rows and columns), also known as relations.

- It provides the Referential Integrity between rows or columns of various tables.
- It allows us to updates the table indexes automatically.
- It uses many SQL queries and combines useful information from multiple tables for the end-users.

How MySQL Works?

MySQL follows the working of Client-Server Architecture. This model is designed for the end-users called clients to access the resources from a central computer known as a server using network services. Here, the clients make requests through a graphical user interface (GUI), and the server will give the desired output as soon as the instructions are matched. The process of MySQL environment is the same as the client-server model.



The core of the MySQL database is the MySQL Server. This server is available as a separate program and responsible for handling all the database instructions, statements, or commands. The working of MySQL database with MySQL Server are as follows:

1. MySQL creates a database that allows you to build many tables to store and manipulate data and defining the relationship between each table.
2. Clients make requests through the GUI screen or command prompt by using specific SQL expressions on MySQL.
3. Finally, the server application will respond with the requested expressions and produce the desired result on the client-side.

A client can use any MySQL GUI. But, it is making sure that your GUI should be lighter and user-friendly to make your data management activities faster and easier. Some

of the most widely used MySQL GUIs are MySQL Workbench, SequelPro, DBVisualizer, and the Navicat DB Admin Tool. Some GUIs are commercial, while some are free with limited functionality, and some are only compatible with MacOS. Thus, you can choose the GUI according to your needs.

Reasons for popularity

MySQL is becoming so popular because of these following reasons:

- MySQL is an open-source database, so you don't have to pay a single penny to use it.
- MySQL is a very powerful program that can handle a large set of functionality of the most expensive and powerful database packages.
- MySQL is customizable because it is an open-source database, and the open-source GPL license facilitates programmers to modify the SQL software according to their own specific environment.
- MySQL is quicker than other databases, so it can work well even with the large data set.
- MySQL supports many operating systems with many languages like PHP, PERL, C, C++, JAVA, etc.
- MySQL uses a standard form of the well-known SQL data language.
- MySQL is very friendly with PHP, the most popular language for web development.
- MySQL supports large databases, up to 50 million rows or more in a table. The default file size limit for a table is 4GB, but you can increase this (if your operating system can handle it) to a theoretical limit of 8 million terabytes (TB).

3.1.2 Ajax

AJAX stands for Asynchronous JavaScript and XML. AJAX is a new technique for creating better, faster, and more interactive web applications with the help of XML, HTML, CSS, and Java Script.

- Ajax uses XHTML for content, CSS for presentation, along with Document Object Model and JavaScript for dynamic content display.

- Conventional web applications transmit information to and from the sever using synchronous requests. It means you fill out a form, hit submit, and get directed to a new page with new information from the server.
- With AJAX, when you hit submit, JavaScript will make a request to the server, interpret the results, and update the current screen. In the purest sense, the user would never know that anything was even transmitted to the server.
- XML is commonly used as the format for receiving server data, although any format, including plain text, can be used.
- AJAX is a web browser technology independent of web server software.
- A user can continue to use the application while the client program requests information from the server in the background.
- Intuitive and natural user interaction. Clicking is not required, mouse movement is a sufficient event trigger.
- Data-driven as opposed to page-driven.

3.1.3 JQuery

Query is a lightweight, "write less, do more", JavaScript library.

The purpose of jQuery is to make it much easier to use JavaScript on your website.

jQuery takes a lot of common tasks that require many lines of JavaScript code to accomplish, and wraps them into methods that you can call with a single line of code.

jQuery also simplifies a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.

The jQuery library contains the following features:

- HTML/DOM manipulation
- CSS manipulation
- HTML event methods
- Effects and animations
- AJAX
- Utilities

3.1.4 Javascript

JavaScript is a text-based programming language used both on the client-side and **server**-side that allows you to make web pages interactive. Where HTML and CSS are languages that give structure and style to web pages, JavaScript gives web pages interactive elements that engage a user.

3.1.5 Bootstrap

- Bootstrap is a free front-end framework for faster and easier web development
- Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins
- Bootstrap also gives you the ability to easily create responsive designs

3.1.6 HTML

- HTML stands for Hyper Text Markup Language
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- HTML elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

3.1.7 CSS

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files.

3.1.8 PHP

What is PHP?

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

What Can PHP Do?

- PHP can generate dynamic page content
- PHP can create, open, read, write, delete, and close files on the server
- PHP can collect form data
- PHP can send and receive cookies
- PHP can add, delete, modify data in your database
- PHP can be used to control user-access
- PHP can encrypt data

With PHP you are not limited to output HTML. You can output images, PDF files, and even Flash movies. You can also output any text, such as XHTML and XML.

Why PHP?

- PHP runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.)
- PHP is compatible with almost all servers used today (Apache, IIS, etc.)
- PHP supports a wide range of databases
- PHP is free. Download it from the official PHP resource: www.php.net
- PHP is easy to learn and runs efficiently on the server side

3.1.9 Photoshop

Adobe Photoshop is the predominant photo editing and manipulation software on the market. Its uses range from the full-featured editing of large batches of photos to creating intricate digital paintings and drawings that mimic those done by hand.

3.1.10 Illustrator

Adobe Illustrator is a professional vector-based design and drawing program. Used as part of a larger design workflow, Illustrator allows for the creation of everything from single design elements to entire compositions. Designers use Illustrator to create posters, symbols, logos, patterns, icons, etc.

3.1.11 Adobe Dreamweaver

Developed by [Adobe Systems](#), Dreamweaver is a proprietary web development tool first created by Macromedia in 1997. Dreamweaver helps developers build web sites for the Internet or an intranet, letting developers design, code and manage websites as well as mobile content. It is currently available for OS X and Windows.

Dreamweaver combines a visual design surface, known as Live View, and a code editor with features such as code collapsing, code completion and syntax highlighting.

3.1.12 SQLyog

MySQL manager and admin tool.

SQLyog is the most powerful manager, admin and GUI tool for MySQL, combining the features of MySQL Query Browser, Administrator, phpMyAdmin and other MySQL Front Ends and MySQL GUI tools in a single intuitive interface. SQLyog is a fast, easy to use and compact graphical tool for managing your MySQL databases. SQLyog was developed for all who use MySQL as their preferred RDBMS. Whether you enjoy the control of handwritten SQL or prefer to work in a visual environment, SQLyog makes it easy for you to get started and provides you with tools to enhance your MySQL experience.

- **MySQL manager and admin tool:** SQLyog provides you with powerful means to manage your MySQL databases. It features the simplicity of MySQL Front, with the power of EMS MySQL Admin. SQLyog provides detailed profile information for every SQL statement executed.
- **Session Restore:** SQLyog is a powerful MySQL manager that restores your previous session the way you left it. Be it a system crash or accidentally closing your MySQL client.

- **Schema and Data Sync:** Find and fix schema mismatching while syncing data from one database to another. Replicate data periodically to a different server at scheduled intervals.
- **Compressed backups with scheduling:** Find the plethora of options in mysqldump daunting? Just back up with a single click interface in SQLyog MySQL GUI tool.
- **Autocomplete and SQL formatting:** Save hours of typing. Write queries 10x faster. Inherited spaghetti SQL? Make it neat with a click. Improve readability and maintenance of SQL.
- **Query Builder:** Build complex queries using drag-n-drop interface. Visually create SQL statements without the need to remember column names.
- **Schema Optimizer and Index Analyzer:** Determine the optimal column types. Fit more data into memory thereby improving response times. Eliminate Redundant Indexes by improving INSERTs and UPDATEs
- **Object Browser:** The Object browser window gives you complete details of the server you are connected to in a tree format. It allows you to perform different operations on selected objects such as Open Table, Create Table, Alter Table, Manage Indexes, Relationships/Foreign Keys, Backup/Export, Import and more.
- **Favorites:** SQLyog allows easy access to frequently used SQL scripts. The script files can be stored as files from the SQLyog interface or link to an existing file anywhere where windows can access it - on a local drive or a shared network drive. The SQL scripts and the file links can be organized in folder and subfolders.

SQLyog gives you a choice between four different connectivity options. No matter whatever your firewall/proxy/ISP settings are, you should be able to smartly manage your MySQL Server(s). The various connectivity options (in the order of responsiveness and network efficiency) are:

- **Direct connection using MySQL C API:** This is by far the fastest way to connect and manage your MySQL server. This is the preferred method if you are not concerned about your MySQL traffic being spoofed. However, this requires direct access to your MySQL server. Most ISPs block the MySQL port or disallow remote access.

- **SSH Tunneling*:** Use this option if you need to encrypt the contents of your MySQL traffic, but still want the efficiency of the MySQL C API. If your host supports SSH tunneling, SQLyog can use the tunnel to communicate with MySQL.
- **HTTP Tunneling*:** This option saves your day if your ISP/MySQL provider disallows remote access, either by blocking the MySQL port or by giving access to “localhost” only. The response is slow compared to the previous options since the data is XML encoded and HTTP is stateless by nature. However, you can use all the cool features of SQLyog.
- **HTTPS Tunneling*:** This option is similar to HTTP Tunneling but adds another layer of security by establishing a secure channel.
- **SSL Encryption*:** With this option all communication between SQLyog and the MySQL server is encrypted. The MySQL server must support this option. Not all servers do.

3.2 Server Specifications

- Web server : XAMPP
- PHP Version: For MySQL & MariaDB Editions, PHP 5.4 or above with PHP XML extension enabled.
- Database: MySQL 5.1 or above, MariaDB 10.0 or above.

3.2.1 XAMPP

What is XAMPP?

XAMPP is an abbreviation where X stands for Cross-Platform, A stands for Apache, M stands for MYSQL, and the Ps stand for PHP and Perl, respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl.

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable

environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl is a programming language used for web development, PHP is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL. The detailed description of these components is given below.

Components of XAMPP

As defined earlier, XAMPP is used to symbolize the classification of solutions for different technologies. It provides a base for testing of projects based on different technologies through a personal server. XAMPP is an abbreviated form of each alphabet representing each of its major components. This collection of software contains a web server named Apache, a database management system named MariaDB and scripting/programming languages such as PHP and Perl. X denotes Cross-platform, which means that it can work on different platforms such as Windows, Linux, and macOS.

Many other components are also part of this collection of software and are explained below.

1. **Cross-Platform:** Different local systems have different configurations of operating systems installed in it. The component of cross-platform has been included to increase the utility and audience for this package of Apache distributions. It supports various platforms such as packages of Windows, Linus, and MAC OS.
2. **Apache:** It is an HTTP a cross-platform web server. It is used worldwide for delivering web content. The server application has made free for installation and used for the community of developers under the aegis of Apache Software Foundation. The remote server of Apache delivers the requested files, images, and other documents to the user.
3. **MariaDB:** Originally, MySQL DBMS was a part of XAMPP, but now it has been replaced by MariaDB. It is one of the most widely used relational DBMS, developed by MySQL. It offers online services of data storage, manipulation, retrieval, arrangement, and deletion.
4. **PHP:** It is the backend scripting language primarily used for web development. PHP allows users to create dynamic websites and applications. It can be installed on every platform and supports a variety of database management systems. It was

implemented using C language. PHP stands for Hypertext Processor. It is said to be derived from Personal Home Page tools, which explains its simplicity and functionality.

5. Perl: It is a combination of two high-level dynamic languages, namely Perl 5 and Perl 6. Perl can be applied for finding solutions for problems based on system administration, web development, and networking. Perl allows its users to program dynamic web applications. It is very flexible and robust.
6. phpMyAdmin: It is a tool used for dealing with MariaDB. Its version 4.0.4 is currently being used in XAMPP. Administration of DBMS is its main role.
7. OpenSSL: It is the open-source implementation of the Secure Socket Layer Protocol and Transport Layer Protocol. Presently version 0.9.8 is a part of XAMPP.
8. XAMPP Control Panel: It is a panel that helps to operate and regulate upon other components of the XAMPP. Version 3.2.1 is the most recent update. A detailed description of the control panel will be done in the next section of the tutorial.
9. Webalizer: It is a Web Analytics software solution used for User logs and provide details about the usage.
10. Mercury: It is a mail transport system, and its latest version is 4.62. It is a mail server, which helps to manage the mails across the web.
11. Tomcat: Version 7.0.42 is currently being used in XAMPP. It is a servlet based on JAVA to provide JAVA functionalities.
12. Filezilla: It is a File Transfer Protocol Server, which supports and eases the transfer operations performed on files. Its recently updated version is 0.9.41.

XAMPP Format Support

XAMPP is supported in three file formats:

- .EXE- It is an extension used to denote executable files making it accessible to install because an executable file can run on a computer as any normal program.
- .7z - 7zip file- This extension is used to denote compressed files that support multiple data compression and encryption algorithms. It is more favored by a formalist, although it requires working with more complex files.
- .ZIP- This extension supports lossless compression of files. A Zipped file may contain multiple compressed files. The Deflate algorithm is mainly used for

compression of files supported by this format. The .ZIP files are quite tricky to install as compared to .EXE

Thus .EXE is the most straightforward format to install, while the other two formats are quite complicated and complex to install.

Prerequisites

Before going through XAMPP tutorial in-depth, you must have a fundamental knowledge of web development languages like HTML, and PHP.

Audience

Our XAMPP tutorial is designed for the aspirants who want to test their website or application on a localhost webserver. This tutorial will help those who want to build their career as a front end or web developer.

Problem

We assure you that it will resolve all your queries related to XAMPP, and you will not find any problem in this tutorial. Still, if there is any mistake or discrepancy, please post the problem in the contact form.

CHAPTER 4

LOGICAL DESIGN AND SYSTEM WORK FLOW

4.1 Work Flow Diagram

The following Diagram can describe how work flow from one step to another

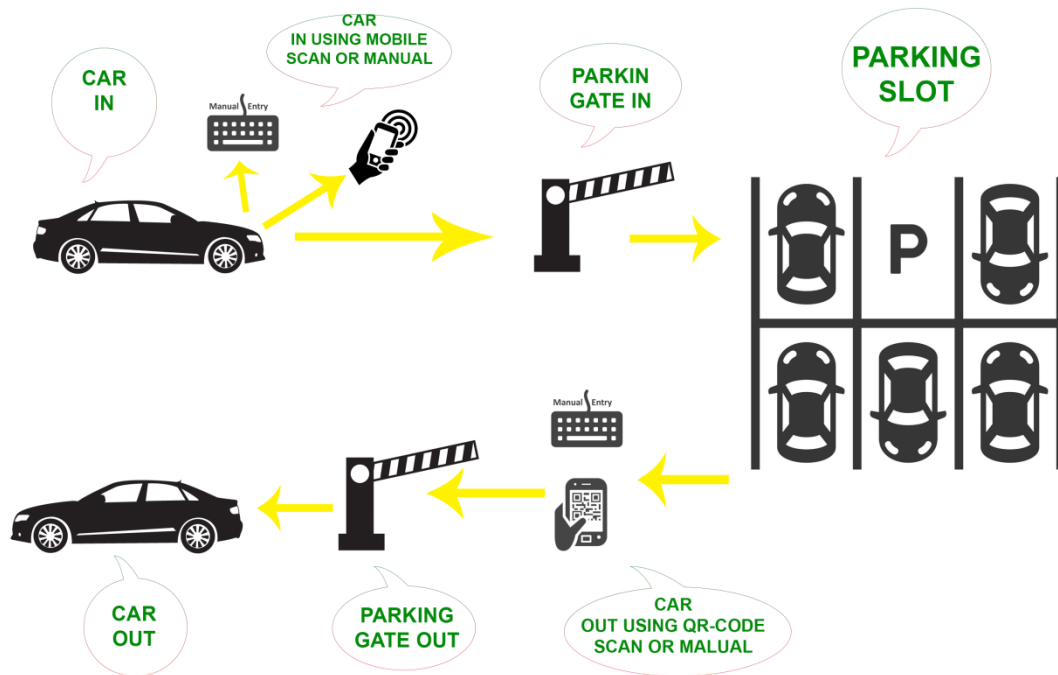


Fig: 4.1 Work Flow Diagram

4.2 Database Schema relationship

The following diagram shows the relationship between table

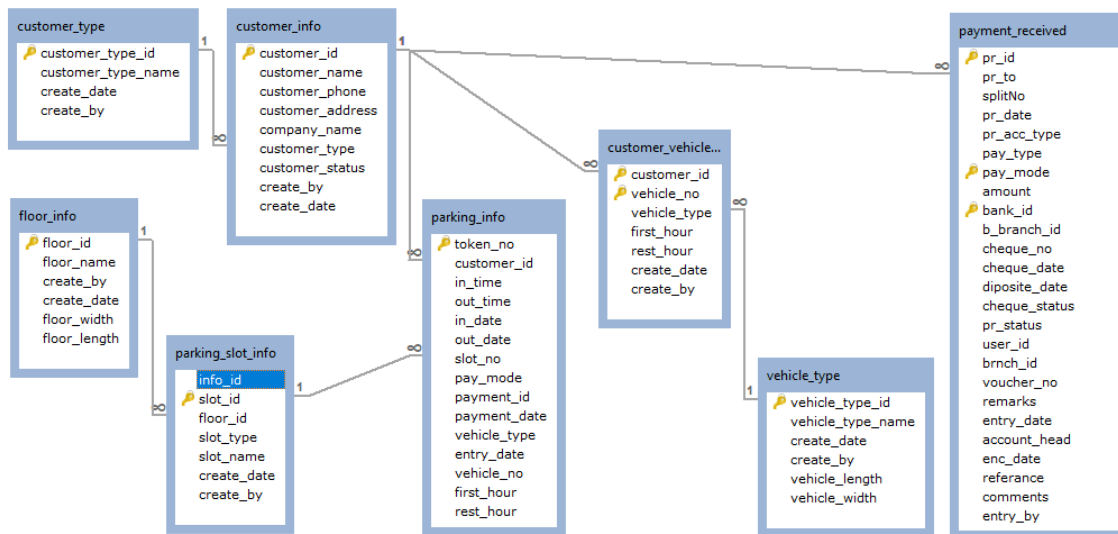


Fig: 4.2 Database Schema

4.3 Database Table

| | Field | Type | Comment |
|-----------|--------------------|----------------------|---------|
| PK | customer_type_id | varchar(20) NOT NULL | |
| | customer_type_name | varchar(30) NULL | |
| | create_date | datetime NULL | |
| | create_by | varchar(25) NULL | |

Table: 4.3.1 Customer Type

| | Field | Type | Comment |
|-----------|----------------|----------------------|---------|
| PK | customer_id | varchar(15) NOT NULL | |
| | customer_name | varchar(30) NULL | |
| | customer_phone | varchar(30) NULL | |

| | |
|------------------|-------------------|
| customer_address | varchar(255) NULL |
| company_name | varchar(20) NULL |
| customer_type | varchar(10) NULL |
| customer_status | int(11) NULL |
| create_by | varchar(20) NULL |
| create_date | datetime NULL |

Table: 4.3.2 Customer Info

| | Field | Type | Comment |
|-----------|--------------|----------------------|---------|
| PK | customer_id | varchar(15) NOT NULL | |
| PK | vehicle_no | varchar(20) NOT NULL | |
| | vehicle_type | varchar(10) NULL | |
| | first_hour | double NULL | |
| | rest_hour | double NULL | |
| | create_date | datetime NULL | |
| | create_by | varchar(25) NULL | |

Table: 4.3.3 Customer Vehicle Info

| | Field | Type | Comment |
|-----------|--------------|----------------------|---------|
| PK | floor_id | varchar(10) NOT NULL | |
| | floor_name | varchar(30) NULL | |
| | create_by | varchar(20) NULL | |
| | create_date | datetime NULL | |
| | floor_width | float NULL | |
| | floor_length | float NULL | |

Table: 4.3.4 Floor Info

| | Field | Type | Comment |
|-----------|--------------|----------------------|----------------|
| | info_id | varchar(20) NULL | |
| PK | slot_id | varchar(20) NOT NULL | |
| | floor_id | varchar(10) NULL | |
| | slot_type | varchar(20) NULL | |
| | slot_name | varchar(50) NULL | |
| | create_date | datetime NULL | |
| | create_by | varchar(30) NULL | |

Table: 4.3.5 Parking Slot Info

| | Field | Type | Comment |
|-----------|-------------------|----------------------|----------------|
| PK | vehicle_type_id | varchar(10) NOT NULL | |
| | vehicle_type_name | varchar(20) NULL | |
| | create_date | datetime NULL | |
| | create_by | varchar(30) NULL | |
| | vehicle_length | float NULL | |
| | vehicle_width | float NULL | |

Table: 4.3.6 Vehicle Type

| | Field | Type | Comment |
|-----------|--------------|----------------------|----------------|
| PK | token_no | varchar(30) NOT NULL | |
| | customer_id | varchar(20) NULL | |
| | in_time | time NULL | |
| | out_time | time NULL | |
| | in_date | date NULL | |
| | out_date | date NULL | |
| | slot_no | varchar(10) NULL | |

| | |
|--------------|------------------|
| pay_mode | varchar(15) NULL |
| payment_id | varchar(20) NULL |
| payment_date | datetime NULL |
| vehicle_type | varchar(20) NULL |
| entry_date | datetime NULL |
| vehicle_no | varchar(30) NULL |
| first_hour | float NULL |
| rest_hour | float NULL |

Table: 4.3.7 Parking Info

| Field | Type | Comment |
|----------------|----------------------|---------|
| charge_info_id | varchar(15) NOT NULL | |
| customer_id | varchar(20) NULL | |
| vehicle_id | varchar(20) NULL | |
| first_hour | float NULL | |
| rest_hour | float NULL | |
| perking_time | varchar(20) NULL | |
| charge_amaout | float NULL | |
| charge_date | datetime NULL | |
| charge_status | varchar(20) NULL | |
| user_id | varchar(20) NULL | |

Table: 4.3.8 Charge Info

| Field | Type | Comment |
|-------------|----------------------|-------------------|
| pr_id | varchar(20) NOT NULL | |
| pr_to | varchar(100) NULL | |
| splitNo | varchar(25) NULL | |
| pr_date | datetime NULL | |
| pr_acc_type | varchar(20) NULL | |
| pay_type | varchar(20) NULL | RECEIVED/PAYMENT |
| pay_mode | varchar(20) NOT NULL | CASH/CHEQUE/BEFTN |

| | | | |
|---------------|--------------|----------|-----------------------------------|
| amount | double | NULL | |
| bank_id | varchar(10) | NOT NULL | Cheque Bank |
| b_branch_id | varchar(20) | NULL | Cheque Bank Branch |
| cheque_no | varchar(20) | NULL | |
| cheque_date | datetime | NULL | |
| diposite_date | datetime | NULL | |
| cheque_status | varchar(25) | NULL | Received/Deposited/Honor/DisHonor |
| pr_status | varchar(20) | NULL | Pending/Complete |
| user_id | varchar(20) | NULL | Create By |
| brnch_id | varchar(50) | NULL | Branch Filter |
| voucher_no | varchar(20) | NULL | Voucher No For This Pay_Receive |
| remarks | varchar(200) | NULL | Pay_receive Verify Remarks |
| entry_date | datetime | NULL | |
| account_head | varchar(30) | NULL | Chart Of Account For Bank |
| enc_date | datetime | NULL | |
| referance | varchar(50) | NULL | |
| comments | varchar(255) | NULL | |
| entry_by | varchar(20) | NULL | |

Table: 4.3.9 Payment Received

4.4 Database Function and Stored Procedure

- **Stored Procedure**

- delete_node: this Stored Procedure use for the delete higherkey model single node.
- delete_sub_tree: this Stored Procedure use for the delete higherkey model sub tree.
- insert_node: this Stored Procedure use for the insert higherkey model node.
- make_id: this Stored Procedure use for the create a won customize id which is needed ex: CUS-00001, VEH-00015 etc.
- move_tree: this Stored Procedure use for the move higherkey model tree one node to another node.

- **Function**

- Age : this function return Age from a birthday which is important for this project
- STRIP_NON_DIGIT: this function return word from a number .
- vehicleCharge: this function calculate parking charge from a specific customer vehicle in time and out time
- cus_opening_balance: this function calculate opening balance charge from a specific customer

CHAPTER 5

SYSTEM AND INTERFACE DESIGN

5.1 System Design

Our entire system has been developed in PHP, Ajax and JQuery programming languages. Through data collection and research studies, we have seen a variety of challenges in the current parking management system, which often affect its good performance and efficiency. I have described the current system below.

5.2 File and Folder path

- Smart_parking
 - css: This folder contains all types of css files that have been used in this project.

- fonts: This folder contains all types of font files that have been used in this project.
- images: This folder contains all types of image files that have been used in this project.
- js: This folder contains all types of js files that have been used in this project.
- site:
 - account: This folder contains all the PHP files and class files of the accounts. Through which many kinds of functionality have been done in the project.
 - monitoring: This folder contains all the PHP files and class files of the monitoring. Through which many kinds of functionality have been done in the project.
 - parking: This folder contains all the PHP files and class files of the parking. Through which many kinds of functionality have been done in the project.
 - report: This folder contains all the PHP files and class files of the report.
 - settings: This folder contains all the PHP files and class files of the settings. Through which many kinds of functionality have been done in the project.
- sLIB: This folder contains all types of Library files that have been used in this project.

5.3 System Menu

- Settings
 - Global Information
 - User/Administration
 - Create User
 - User Permission
 - Parking Settings
 - Parking Floor info
 - Parking Slot info
 - Customer Settings

- Customer Info
 - Customer Type
 - Vehicle Settings
 - Vehicle Type
 - Vehicle Pricing
- Task
 - Manual Parking Entry
 - Manual Parking Out
- Monitoring
 - Real Time Parking
- Account
 - Cash Receive
- Reports
 - Customer Report
 - Customer Ledger
 - All Customer Summary
 - Parking Report
 - Parking Report Day Wise
 - Parking Report Month Wise
 - Accounts Report
 - Parking Charge Collection Day Wise
 - Parking Charge Collection Month Wise
 - Parking Charge Explorar
 - Charge Report
 - Daily Charge Report
 - Monthly Charge Report
 - Day Wise Charge Report
 - Yearly Charge Report

5.4 System Home page

This is system home page. Lots of information has been shown here

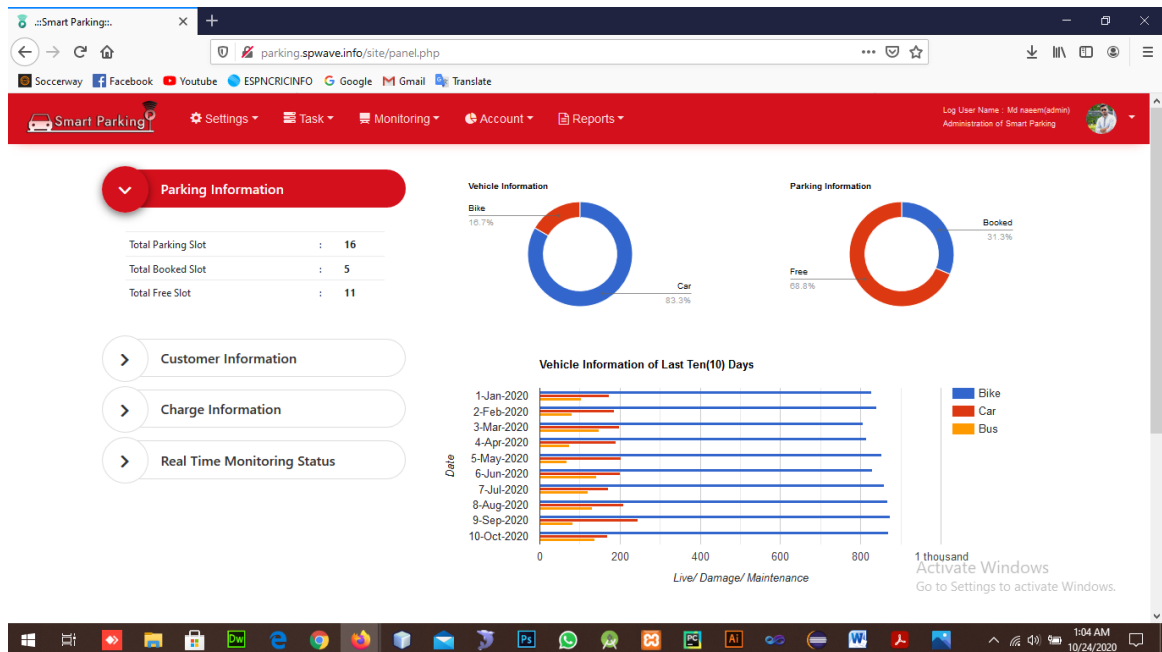


Fig: 5.4 Home page

- Perking Information
 - Total Parking Slot
 - Total Booked Slot
 - Total Free Slot

- Customer Information
 - Total Customer
 - Daily Customer
 - Monthly Customer
 - Yearly Customer
- Charge Information
 - Total Charge
 - Cash Customer
 - Dues
- Also use pie chart and bar chart
 - Pie chart represents overall Vehicle Information and Parking Information
 - Bar chart represents Vehicle Information of Last Ten(10) Days

5.5 System Design Snapshots

Some important form snapshots given below

The screenshot shows a web application interface for user management. The top section is titled 'Create System User' and contains a form with the following fields:

- User Name: admin
- Full Name: Md naeem
- Address: khilgaon
- Password: *23AE809DDACAF96AF0FD78ED04
- Contact No: 01822874962
- User Status: Active (selected), In Active
- Photo: Browse... No file selected. (Only Image Format Allowed)

Below the form are buttons for '+ New', 'Save', 'Save Change', and 'Delete'. The bottom section is titled 'System User List' and displays a table with the following data:

| SrNo. | User ID | Name | Full Name | Contact No | Address | Status |
|-------|------------|--------|---------------------|-------------|-------------|--------|
| 1 | USR-000001 | admin | Md naeem | 01822874962 | khilgaon | Active |
| 2 | USR-000002 | ali | Mohammad Ali | 01773907245 | kawranbazar | Active |
| 3 | USR-000003 | sumon | Sumon Ahmed | 0112556656 | motijheel | Active |
| 4 | USR-000004 | selim | Selim Reza | 0171000000 | Gazipur | Active |
| 5 | USR-000005 | mahmud | Mahmud Hasan | 0171000000 | SU | Active |
| 6 | USR-000006 | barsha | Sadia Tasnim Barsha | 0171000000 | SU | Active |

The table includes a search bar, a 'Show 10 entries' dropdown, and pagination controls at the bottom indicating 'Showing 1 to 6 of 6 entries'.

Fig : 5.5.1 Create System User

Settings/User/Administration/User Permission

Search

User Permission

User Permission For Smart Parking.

Select / Deselect All Menu Permission

▼ Settings

- All Settings
- Settings
- Global Information
- User/Administration
- Create User
- User Permission
- Parking Settings
- Parking Floor info
- Parking Slot info
- Customer Settings
- Customer Info
- Customer Rules
- Customer Type
- Vehicle Settings
- Vehicle Type
- Vehicle Pricing Policy

Sp Wave

Fig : 5.5.2 User Permission

Settings / Customer

Search

Customer

Customer Info Entry

Customer Info

Active Inactive

Vehicle Info

| Sl | Vehicle Type | Vehicle No | First Hour | Rest Hour |
|----|--------------|------------|------------|-----------|
| 1 | Car | 154565 | 20 | 10 |
| 2 | Bike | 757575 | 15 | 10 |

Customer List

| SN | Customer Name | Phone | Address | Customer Type |
|----|---------------|-------------|---------|---------------|
| 1 | Cash Customer | | | Daily |
| 2 | ali | 01773101010 | Dhaka | Monthly |
| 3 | Arif | 1234 | aaa | Monthly |

Fig: 5.5.3 Customer Info Entry

Settings / Parking Search

Manual Parking Info

Manual Parking Info Entry

Manual Parking Info

| | | | |
|-----------------|--|----------------|--|
| • Customer Name | <input type="text" value="Cash Customer"/> | • In Date | <input type="text" value="22-Oct-2020"/> |
| • Vehicle No | <input type="text" value="2020201121110 Car"/> | • In Time | <input type="text" value="06:54:46"/> |
| • Vehicle Type | <input type="text" value="Car"/> | • First Hour | <input type="text" value="0"/> |
| • Slot No | <input type="text" value="Select a Slot"/> | • Rest of Hour | <input type="text" value="0"/> |

Parking List

Show entries Search:

| SN | Token No | Customer Name | In Date | In Time | Slot No | Vehicle Type |
|----|------------------------|---------------|-------------|----------|------------|--------------|
| 1 | 2020102220202011211101 | Cash Customer | 22-Oct-2020 | 06:54:46 | PRK-000004 | Car |
| 2 | 2020101832432000000001 | Cash Customer | 18-Oct-2020 | 14:52:33 | PRK-000010 | Car |
| 3 | 2020101143525320000001 | Cash Customer | 11-Oct-2020 | 17:56:02 | PRK-000013 | Car |
| 4 | 2020101115456500000001 | ali | 11-Oct-2020 | 17:53:07 | PRK-000017 | Car |
| 5 | 2020101111456100000001 | Arif | 11-Oct-2020 | 17:53:07 | PRK-000002 | Bike |
| 6 | 2020100915456500000001 | ali | 09-Oct-2020 | 17:44:27 | PRK-000015 | Car |

Showing 1 to 6 of 6 entries Previous 1 Next

Fig: 5.5.4 Manual Parking Info

Task / Manual Parking Out Cash Customer 43532532

Customer Info

Customer Name : Cash Customer

Customer Type : Daily

Vehicle No : 43532532

Vehicle Type : Car

Customer Balance : 1,000.00

Perking Info

Perking Slot : A-7

In : 2020-10-11 17:56:02

Out : 2020-10-24 01:21:43

Total Hours : 295:25:41

Payment Info

First Hour : 20 Tk

Rest of the Hours : 10 Tk

Charge Amount : 2970 Tk

Discount :

Total Amount :

Fig: 5.5.5 Manual Parking Out

Account / Cash Receive Search

Cash Receive

All cash receive entry for smart parking system

Cash Receive Info

•Customer Name

•Amount

•Discount

•Net Receive

Cash Receive List

Show entries Search:

| SN | Date | Customer Name | Amount | Discount |
|----|---------------------|---------------|--------|----------|
| 1 | 2020-10-09 17:26:03 | CUS-000002 | 100 | 10 |
| 2 | 2020-10-09 17:27:03 | CUS-000002 | 200 | 40 |

Showing 1 to 2 of 2 entries Previous | 1 | Next

Fig: 5.5.6 Cash Receive

Real Time Parking Slot

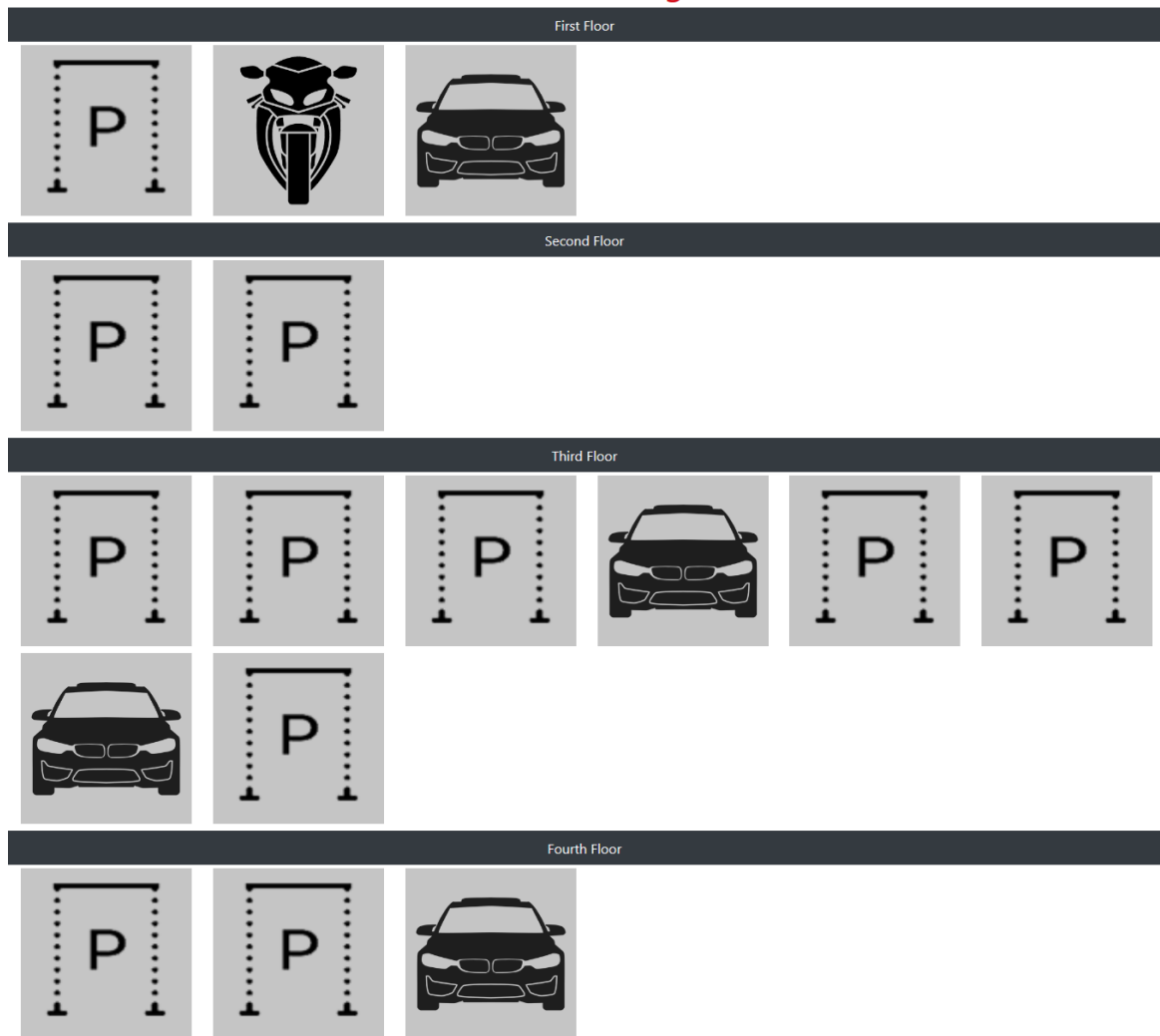


Fig: 5.5.7 Real Time Parking Slot

5.6 Report Snapshots

Some important report snapshots given below

Customer Ledger [Generate](#) [Print](#)

Customer Ledger
01-Oct-2020 to 24-Oct-2020

Customer Name : ali
Address : Dhaka
Mobile : 01773101010 Date: 24-Oct-2020

| Sl | Date | Particulars | Reference | Debit(TK) | Credit(TK) | Balance(TK) |
|--------------|-------------------------|-----------------|-----------|---------------|--------------|-----------------------|
| 1 | 01-Oct-2020 00:00:01 AM | Opening Balance | | - | - | 0.00 |
| 2 | 09-Oct-2020 17:10:03 PM | Cash Receive | | 100.00 | 0.00 | 100.00 |
| 3 | 09-Oct-2020 17:10:03 PM | Discount | | 0.00 | 10.00 | 90.00 |
| 4 | 09-Oct-2020 17:10:03 PM | Cash Receive | | 200.00 | 0.00 | 290.00 |
| 5 | 09-Oct-2020 17:10:03 PM | Discount | | 0.00 | 40.00 | 250.00 |
| 6 | 09-Oct-2020 17:10:12 PM | Parking Charge | | 0.00 | 20.00 | 230.00 |
| Total | | | | 300.00 | 70.00 | Advance 230.00 |

Fig: 5.6.1 Customer Ledger

All Customer Summary [Generate](#) [Print](#)

All Customer Summary
24-Oct-2020

| Sl | Customer Name | Parking | Charge | Payment | Discount | Balance |
|--------------|---------------|----------|--------------|---------------|--------------|---------------|
| 1 | Cash Customer | 3 | 0.00 | 0.00 | 0.00 | 0.00 |
| 2 | ali | 2 | 20.00 | 300.00 | 50.00 | 230.00 |
| 3 | Arif | 1 | 0.00 | 0.00 | 0.00 | 0.00 |
| Total | | 6 | 20.00 | 300.00 | 50.00 | 230.00 |

Fig: 5.6.2 All Customer Summary

Day Wise Parking Report [Generate](#) [Print](#) [XL](#)

Day Wise Parking Report
01-Oct-2020 to 24-Oct-2020

| Sl | Date | Customer Name | Vehicle Type | Vehicle No | Slot | In Time | Out Time |
|----|-------------|---------------|--------------|---------------|---------------------|----------|----------|
| 1 | 09-Oct-2020 | ali | Car | 154565 | Fourth Floor (A-1) | 17:44:27 | 17:45:12 |
| 2 | 11-Oct-2020 | Cash Customer | Car | 43532532 | Third Floor (A-7) | 17:56:02 | - |
| 3 | 11-Oct-2020 | ali | Car | 154565 | Fourth Floor (A-3) | 17:53:07 | - |
| 4 | 11-Oct-2020 | Arif | Bike | 114561 | First Floor (A-1) | 17:53:07 | - |
| 5 | 18-Oct-2020 | Cash Customer | Car | 32432 | Third Floor (A-4) | 14:52:33 | - |
| 6 | 22-Oct-2020 | Cash Customer | Car | 2020201121110 | First Floor (A-3) | 06:54:46 | - |

Fig: 5.6.3 Day Wise Parking Report

Month Wise Parking Report Start Date: 01-Oct-2020 End Date: 31-Oct-2020 [Generate](#) [Print](#) [XL](#)

Month Wise Parking Details
01-Oct-2020 to 31-Oct-2020

| SL | Month | Total Customer | Total IN | Total Out | Total Payment |
|----|----------|----------------|----------|-----------|---------------|
| 1 | Oct-2020 | 6 | 6 | 1 | 0 |

Copyright © 2020 - Smart Parking System

Fig: 5.6.4 Month Wise Parking Details

Day Wise Charge Collection Start Date: 01-Oct-2020 End Date: 24-Oct-2020 [Generate](#) [Print](#) [XL](#)

Day Wise Charge Collection Details
01-Oct-2020 to 24-Oct-2020

| SL | Customer Name | Date | Pay Mode | Amount |
|---------------|---------------|-------------|----------|---------------|
| 1 | ali | 09-Oct-2020 | CASH | 100.00 |
| 2 | ali | 09-Oct-2020 | CASH | 200.00 |
| Total: | | | | 300.00 |

Copyright © 2020 - Smart Parking System

Fig: 5.6.5 Day Wise Charge Collection Details

Month Wise Charge Collection Start Date: 01-Oct-2020 End Date: 31-Oct-2020 [Generate](#) [Print](#) [XL](#)

Smart Parking
Month Wise Charge Collection Details

| SL | Month | Amount |
|---------------|----------|---------------|
| 1 | Oct-2020 | 350.00 |
| Total: | | 350.00 |

Copyright © 2020 - Smart Parking System

Fig: 5.6.6 Month Wise Charge Collection Details

CHAPTER 6

SECURITY AND IMPLEMENTATION

6.1 System Security

6.1.1 User Management

The user system is including add, modify and delete user. The system admin can be able to manage all user.

6.1.2 User Authentication

Users need to provide their username and Password every time as they need to access the system.

6.1.3 Admin Authentication

Administrator need to provide administrator username and Password every time as they need to access the system.

6.1.4 Pages permission

- Admin: Admin can access all pages.
- User: User can access their permitted pages

6.1.5 Change Password

Each of the users can update their account's password if they feel their password is not secure anymore.

6.2 System Requirements:

The term system requirements means the minimum hardware and software requirement of a computer. If the computer full fill this minimum requirement then the system will run smoothly from this computer. The requirement specification given below

6.2.1 Minimum Hardware Requirements:

| Hardware | Minimum Requirement | Reason |
|---|---------------------|---|
| Processor speed | 2.0 GHz | Accommodate most PCs |
| Memory of user PC | 1 GB RAM | Relatively fast |
| Disk Space of user PC | 100 GB | Adequate Storage capacity |
| Bandwidth(network connection) | 15 Mbps | Relatively Good |
| Disk space of server (Not essential) | 200 GB | Adequate Storage for database and application |

Table 6.2.1 :Minimum Hardware Requirements

A fast processor is required because there is need to handle large amounts of data queries. 8 GB memory is required on the server for faster performance because it runs many processes simultaneously, while the memory in the computer should be relatively fast so as to run the processes required. 200 GB storage capacity in the server's hard disk is necessary for storage of huge amounts of data while the disk space in a user computer should be modest enough. Good network connection is vital because the application is majorly internet based and there is need for faster communication and retrieval of information.

6.2.2 Minimum Software Requirements:

| Software | Minimum Requirement | Reason |
|--------------------------------------|--|--|
| Operating System for computer | Window 7, Windows 8.1, Windows 8, windows XP, Linux, MAC OS, | Globally distributed and widely accessed |
| Database Management System | MySQL | Easy to use and scalable |
| Browser | Opera, Google Chrome, IE, Mozilla Firefox. | Standard browser |

Table 6.2.2: Minimum Software Requirements

Shows the software requirements for the web application that define the prerequisites needed for the optimal functioning of the web application. Each of the following operating system can handle the application, windows 7, windows 8, windows

8.1, Linux, those operating systems were chosen because they are affordable and readily available. MySQL was used in the development of the databases and is relatively cheap, easy to use and scalable. Browsers: any browser including opera, Google chrome, Microsoft Internet explorer and Mozilla Firefox.

6.3 Result of Analysis:

It shows overall progress of a project. The analysis result focus on Results of Implementation, Results of Testing, Achievements of the project, and Limitations of the project. By this analysis we may able to know how much a project efficient or not.

6.3.1 Results of Implementation:

It demonstrates that it will have a web interface where user of the system (User /system admin) can login, where customer parking a vehicle and parking out vehicle also payment daily, monthly and yealy.

6.3.2 Results of Testing:

Cross browser testing was done me to ensure that the web application looks the same in major browsers that is Google chrome, Mozilla Firefox, Opera and Internet explorer. The web project is consistent (looks exactly the same) in Google chrome, Mozilla Firefox and Opera but the looks vary slightly in internet explorer.

6.3.3 Achievements of the project:

- Admin can create unlimited user
- Admin can add permission all created user
- Able to entry customer and multiple vehicle
- Entry vehicle in parking slot vehicle wise
- Perking out the vehicle
- Customer can payment on spot/ monthly and yearly
- System Administrators can add parking floor and slot
- System Administrator can add vehicle type customer type
- Admin and user can see real time perking slot monitoring
- Admin and user can see various report

CHAPTER 7

CONCLUSION AND FUTURE PLANS

7.1 Introduction

This chapter discusses about what has been done, recommendations on the system's operations, future scope of this system and the conclusions that are to be made regarding the system's functioning in the current status and that of the future where some improvements are to be made on the system.

7.2 Recommendation:

- I recommend that the system be used by other companies because I believe that the system can effectively and efficiently park a vehicle.
- I recommend that more research should be done such that the vehicles' details should be able to be scanned during entering in the parking place.
- More research should also be done to extend this project to include tracking a vehicle by using GPS for the purpose of maintaining the security.
- I recommend that through making the web application more responsive the customer will be able to park vehicles on mobile devices easier.

7.3 Scope of Further Development

- We will definitely take this project a step further if we get the chance in the future.
- We will make our parking system more automated by adding new features.
- I will address all the limitations of this system.

- CC TV integration when the car enters, exits and into the parking slot.
- We will definitely create user apps.
- I will further improve the payment system.

7.4 Conclusion

- The system benefits of smart parking go well beyond avoiding the needless circling of city blocks. It also enables cities to develop fully integrated multimodal intelligent transportation systems that don't rely on cars in the first place.
- Developing smart parking solutions within a city requires data standardization and management; mobile phone integration; hardware and software innovation; and coordination among various stakeholders (on and off street parking facility owners, business owners, municipalities, transportation authorities, customers, and software developers).
- The technical solutions and stakeholders are the same data structures and development groups integral to making a smart phone-enabled, multimodal, fully
- integrated transportation solution a reality. In essence, technical enablers and multi-stakeholder coordination effort behind development of a local smart parking solution creates a launch pad toward full transportation system integration.

REFERENCES

- [1] Researchgate: Available https://www.researchgate.net/publication/320356747Design_and_Implementation_of_Smart_Car_Parking_System
- [2] Researchgate: Available https://www.researchgate.net/publication/318486476_Smart_Car_Parking_System_using_Adriano_UNO
- [3] Tutorialspoint: Available <https://www.tutorialspoint.com/index.htm>
- [4] W3schools : Available <https://www.w3schools.com/>