**CHAPTER I**

**INTRODUCTION**

The System smart-parking in vehicular networks with the private parking spots provided from suppliers and processed by authority, cruising drivers can find extra parking spots besides public lot with less fuel and time, suppliers will improve the utility of their spots and make a profit, and public traffic congestion is further reduced.

The situation is getting even worse in developing countries where the number of vehicles has been increasing without sufficient investment in parking facilities. Some governments try to mitigate these problems through building extra parking lots, deploying roadside sensors, and establishing parking guidance systems. While the effect of such centralized approach is obvious and immediate, the limited construction space, expensive investment, and the consequent maintenance cost inhibit a widespread. adoption. Therefore, the parking problem cannot be solved efficiently only through public infrastructure or management.

Different from the traditional solutions, we have observed that a large proportion of parking spots is owned by the private sector and beyond the direct control of local transport authority. Such parking spots in private sector (e.g., residential space, workplace) always remain vacant when spot suppliers (we will use supplier for short) are on a trip or off duty. In addition, suppliers usually spend much money on buying and maintaining these private spots. Hence, they are willing to offer their parking spots for a parking fee as an economical compensation for their expenses. These motivate us to think how much time will be saved for cruising drivers and how much trafﬁc congestion will be relieved and if the information of private parking spots can be initiatively provided by suppliers the public, especially the cruising drivers.

1.1 Statement of the problem

Cruising for a vacant and economical parking spot causes not only time-consuming and frustrating driving experiences, but fuel waste and air pollution. Public parking spots in crowded cities are scarce and expensive. Some governments try to solve these problems through building extra parking lots, deploying road-side sensors, and establishing parking guidance systems. While the effect of such centralized approach is obvious and immediate, the limited construction space, expensive investment, and the consequent maintenance cost inhibit a widespread adoption. Therefore, the parking problem cannot be solved efficiently only through public infrastructure or management.

On the contrary, private parking spots usually have low utilization rates, and the spot suppliers are willing to provide their extra parking resources due to a maintenance cost by charging parking fees. Given this situation, it is imperative to call for a smart parking system that collects and provides private parking spots (e.g., around the home or workplace) to ease public parking concerns.

**CHAPTER 2**

**SYSTEM ANALYSIS**

During system analysis we analysed that the parking in downtown areas has been a problem for years. The situation is getting even worse in developing countries where the number of vehicles has been increasing without sufficient investment in parking facilities. Some governments try to mitigate these problems through building extra parking lots, deploying roadside sensors, and establishing parking guidance systems. While the effect of such centralized approach is obvious and immediate, the limited construction space, expensive investment, and the consequent maintenance cost inhibit a widespread. adoption. Therefore, the parking problem cannot be solved efficiently only through public infrastructure or management.

Different from the traditional solutions, we have observed that a large proportion of parking spots are owned by the private sector and beyond the direct control of local transport authority. Such parking spots in private sector (e.g., residential space, workplace) always remain vacant when spot suppliers

(We will use supplier for short) are on a trip or off duty. In addition, suppliers usually spend much money on buying and maintaining these private spots. Hence, they are willing to offer their parking spots for a parking fee as an economical compensation for their expenses. These motivate us to think how much time will be saved for cruising drivers and how much traffic congestion will be relieved and if the information of private parking spots can be initiatively provided by suppliers the public, especially the cruising drivers.

The property registered by the suppliers are approved by a trusted authority.The drivers can search and book their nearest parking spot based on their type of vehicle,security of parking spot etc.Successfull parking is acknowledged with an sms .Sms involves the parking slot area, parking hours, date and parking fee etc. To prevent illegal suppliers or drivers from submitting invalid messages to the authority, registration is necessary for all entities.

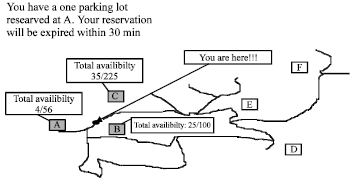


Fig: Availability of Parking Slots

2.1 Present System

In existing system, Public parking spots in crowded cities are scarce and expensive. In existing system, to solve the parking problems try to building extra parking lots, deploying road-side sensors, and establishing parking guidance systems. While the effect of such centralized approach is obvious and immediate, the limited construction space, expensive investment, and the consequent maintenance cost inhibit a widespread adoption. Therefore, the parking problem cannot be solved efficiently only through public infrastructure or management. In 2012, San Francisco conduct a pilot test for smart parking. They use real-time data and "demand-responsive parking to help make parking easier to find." Their program is called SF park. Rates ranged from 25 cents to $7 per hour.Libelium has introduced an upgraded version of smart parking sensor node able to detect available parking slots. It has also extended coverage to three more regions: Australia, Asia-Pacific and Latin America.

2.2 Disadvantages of Present System:

1.Take more time Time and frustrating driving experiences.

2. No security protection,

        3. Limited construction space and expensive investment

2.3 Proposed System

In proposed system, with the private parking spots provided from suppliers and processed by authority, cruising drivers can find extra parking spots besides public lot with less fuel and time, suppliers will improve the utility of their spots and make a profit, and public traffic congestion is further reduced.

2.4 Advantages in Proposed System

1. Improve smart-parking with timely and accurate parking information.

2. Ensure Security.

3. Sms system and Time consuming

5. Quick result retrieval and User friendly

2.5 Feasibility Study

Feasibility analysis is the first step during the system study phase. During the initial investigation phase, the problem is clearly formulated and the area that needs special attention is identified. The phase identifies the particular problem to be solved depending upon the system goals.

When the system is designed, the first step is the study of requirement analysis. The study of requirement analysis is done through feasibility study. Feasibility study is a test of a system proposed according to workability, impact on the organizations, ability to meet the needs and effective use of resources.

A feasible study is not warranted for a system in which economic justification is observed, technical risk is law, few local problems are expected and no reasonable alternatives exist. An alternative is made of whether the identified user needs may be satisfied using our recent software using our recent software and hardware technology. The study decides if the proposed system will be cost effective from the business point of view and it can be developed in the existing budgetary constraints

2.5.1 Technical Feasibility

Our system mainly works on the basis of web based application. It can also available in the other operating system through its site. Technical feasibility includes Risk Resource availability and technologies .Our system works on the current technology and may available and run in various different platforms.

2.5.2 Economic Feasibility

It is found that the benefit from our system would be more than the cost and time involved in its development. In our system the implementation cost over production is economically feasible. By using this system, the suppliers of the system will get parking fees for their private property. If the booking of parking spot will be cancelled after payment then drivers can refund their money and suppliers can credit their service fees. By estimate the profit of the proposed system will exceed the cost of development. Thus the proposed system is economically feasible.

Economic analysis is the most frequently used techniques for evaluating the effectiveness of the proposed system more commonly known as cost/benefit analysis; the procedure is to determine the benefits and savings that are expected from a proposed system and compare them with costs.

2.5.3 Operational Feasibility

The proposed system satisfies the basic needs of every user,for both suppliers and drivers, besides all features. The basic requirements of the supplier is contribute the property, update the property information and register complaints and the drivers can search the parking places and also can do booking and payment and can place complaints.Besides all features in the proposed system , it is ensured that the system solves users requirements.Thus the system is operationally feasible.

The proposed system would be beneficial only if it can be turned into an information system that will meet the operational requirements. One of the main problems faced during the development of a system is getting acceptance from user.

2.5.4 Behavioral Feasibility

It is behaviorally feasible because our system can be accessed and used even for a normal user. Only the contents of the page will change on navigation instead of loading a whole page. This will make the performance faster and efficient. Thus it is very convenient for every users to do booking faster. The interface of the system is designed in such a way that it almost similar to existing systems. Thus it will be user-friendly.

**CHAPTER 3**

**SYSTEM SPECIFICATION**

3.1 Hardware Requirements

* 1 GHz or Faster Processor
* 1 GB of available hard disk space
* 1 GB of RAM
* 5400 RPM hard drive

3.2 Software Requirements

     Development Environment Software Requirements:

* IDE : Net beans
* Frontend : PHP
* Backend : My SQL
* Web Server : Apache
* Web Browser : IE/Chrome/Safari/Opera
* Operating System : Windows 7 ,8,10
* UI Design : HTML, CSS, Bootstrap, JavaScript
  + Engine

**CHAPTER 4**

**SYSTEM DESIGN**

4.1 Context Level Diagram

Admin

Supplier

Driver

Yes/no

booking

Yes/no

Approve/deny

Scheduling

Approve/deny

Authority

Query manage

notification

Fig: 4.1.1 Context level diagram

4.2 Data Flow Diagram

Driver details

Driver

id

Plot status

login

Registration request

Supplier details

Property

details

Supplier\_reg

Driver\_reg

Property\_reg

Supplier

id

Property

id

Driver request

Driver

request

Sms

notification

booking details

booking request

booking

sms

Driver request

status

plot spot

plot details

Driver request

Fig:4.2.1 Level 1 DFD

driver

notification

Notification to driver

Supplier request

notification

supplier

Supplier

details

Driver details

Property details

property

notification

plotspot

Plot

details

Noti

fication

Property details

Fig: 4.2.2 Level 2 DFD

4.3 E-R Diagram

n

1

1

n

1

n

1

Supplier

Server

authority\_supplier

Driver\_

Server

Admin

Driver\_ parking

Parking Slot

Driver

Approv\_

info

1

4.4 Database Design

Table 4.4.1 Driver

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Did | Int | 11 | Primary key | Unique Id |
| Vno | Varchar | 10 | Not null | Vehicle Number |
| Name | Varchar | 20 | Not null | Driver name |
| Email | Varchar | 50 | Not null | Email Id |
| Phone | Int | 11 | Not null | Phone number |
| Dob | Date |  | Not null | Date |
| Address | Varchar | 50 | Not null | Address |
| Gender | Varchar | 5 | Not null | Gender |
| Aadhaar | Int | 11 | Not null | Aadhar Number |
| Pwd | Varchar | 20 | Not null | Password |
| Rpwd | Varchar | 20 | Not null | Repeat Password |

Table 4.4.2 Feedback

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Name | Varchar | 20 | Not null | Name |
| Email | Varchar | 25 | Not null | Email Id |
| Phone | Int | 11 | Not null | Phone number |
| Msg | Varchar | 1000 | Not null | Message |

Table 4.4.3 fwheel

| Column name | Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Pid | Int | 11 | Foreign Key | Property Id |
| Tot | Int | 11 | Not null | Total Spot |
| Row | Int | 11 | Not null | Number of rows |
| Col | Int | 11 | Not null | Number of Columns |
| Features | varchar | 25 | Not null | Features |
| Status | Int | 11 | Not null | Status |

Table 4.4.4 Login

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Username | Text |  | Primary Key | Username |
| Password | Text |  | Not null | Password |
| Type | Text |  | Not null | Type |
| Status | Text |  | Not null | Status |

Table 4.4.5 District

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| District | Varchar | 20 | Not null | District |

Table 4.4.6 Map

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Pincode | Int | 11 | Not null | Pincode |
| Long | Float | 11 | Not null | Longitude |
| Lat | Float | 11 | Not null | Latitude |

Table 4.4.7 plotspot

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Pid | int(11) | 11 | Foreign Key | Plot id |
| Type | varchar(8) | 8 | Not null | Type |
| Row | int(11) | 11 | Not null | Rows |
| Col | int(11) | 11 | Not null | Columns |
| Status | int(11) | 11 | Not null | Status |

Table 4.4.8 plotspot2

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Pid | int(11) | 11 | Foreign Key | Plot id |
| Type | varchar(8) | 8 | Not null | Type |
| Row | int(11) | 11 | Not null | Rows |
| Col | int(11) | 11 | Not null | Columns |
| Status | int(11) | 11 | Not null | Status |

Table 4.4.9 Property

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Pid | Int | 11 | Primary Key | Property Id |
| District | Varchar | 20 | Not null | district |
| Location | Varchar | 20 | Not null | Location |
| Address | Varchar | 20 | Not null | Address |
| Pincode | Int | 11 | Not null | Pincode |
| Status | Int | 11 | Not null | Status |

Table 4.4.10 Supplier

| Column Name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Sid | Int | 11 | Primary Key | Supplier Id |
| Name | Varchar | 20 | Not null | Supplier Name |
| Email | Varchar | 20 | Not null | Email Id |
| Phone | Int | 11 | Not null | Phone number |
| Dob | Date |  | Not null | date |
| Address | Varchar | 50 | Not null | Address |
| Gender | Varcha | 5 | Not null | gender |
| Aadhaar | Int | 11 | Not null | Aadhaar number |
| Pwd | Varchar | 20 | Not null | Password |
| Rpwd | Varchar | 20 | Not null | Repeat Password |

Table 4.4.11 twheel

| Column name | Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Pid | Int | 11 | Foreign Key | Property Id |
| Tot | Int | 11 | Not null | Total Spot |
| Row | Int | 11 | Not null | Number of rows |
| Col | Int | 11 | Not null | Number of Columns |
| Status | Int | 11 | Not null | Status |

Table 4.4.12 Booking

| Column name | Data Type | Size | Constraints | Description |
| --- | --- | --- | --- | --- |
| Uid | Varchar | 50 | Primary Key | User Id |
| Pid | Int | 11 | Foreign Key | Property Id |
| Row | Int | 11 | Not null | Number of rows |
| Col | Int | 11 | Not null | Number of columns |
| Date | Varchar | 20 | Not null | Date of booking |
| Hrs | Int | 11 | Not null | Number of hours |

4.5 Normalization

Designing a database is a complex task and the normalization theory is a useful aid in this design process. A bad database design may lead to certain undesirable situations such as:

1. Repetition of information
2. Inability to represent certain information
3. Loss of information

To minimize these anomalies, normalization may be used. In this system ,here there is only one database named asap

First Normal Form(1NF)

A relation is in first normal form (1 NF),if and only if all its attributes are based on a signal domain. The objective of normalizing a table is to remove its repeating groups and ensure that all entries of the resulting table have at most value. The objective of 1 NF is to divide the database into logical units called tables. When each table has been designed, primary key is assigned to most or all tables.

Second Normal Form (2 NF)

A table is said to be in second normal form (2NF),when it is in 1NF and it satisfies functional dependency. Functional dependency means that every non-primary attributes is fully dependent on a key. The objective of 2NF is to take data that is partly dependent on the primary key, enter that data into another table.

Now consider the database of this system. In this there are a total 8 tables and all the tables are in second normal form. That is all of this tables satisfies second normal form. No repeated information is stored in any table.All of table have a separate primary key some of all an auto number.

Benefits of Normalization

1. To permit simple retrieval of data in response to query and report requests.
2. Help to simplify the structure of tables
3. To structure the data so that there is no repetition of data, that helps in saving space.
4. To simplify the maintenance of data through updates, insertions and deletions
5. To reduce the need to restructure data when new application requirement arise
6. Data consistency within the database
7. Much more flexible database design
8. A better handle on database security
9. Data modification normalize are reduced.

4.6 Design of each subsystem

View

Complaints

View

Booking details

SUPPLIER

AUTHORITY

Register

Property Registration

Receive

Payment

Feedback

Approve /deny

Property

Approve/deny

four wheeler

plot

Approve/deny

Two wheeler

plot

View Feedback

SMART PARKING

ADMIN

DRIVER

Approve

Driver

Approve

Supplier

Reset

booking

Register

Search Plot

Booking

Feedback

View Feedback

4.7 UML Diagrams

4.7.1 Use Case Diagram

Admin

Driver

Supplier

Authority

4.7.2 Sequence Diagram

Available spot

Plot search

:User

:Admin

:Plot

:Slot

: Booking& sms

availability

Yes/no

Booking notification

Sms notification

Fig:Sequence diagram for booking parking area

4.7.3 Class Diagram

User

-uid

+name

+vno

-aadhaar

-accno

+email

+register()

+login()

+search()

+booking()

-payment()

+addProperty()

+delete()

Approve

-uid

-pwd

+decription

+status

+approve()

+delete()

slot

-sid

+description

+type

-pid

+row

+col

+status

+addSlot()

+editSlot()

+deleteSot()

+searchSot()

property

-pid

-sid

+address

+location

+rent/sqrfeet

+availability()

+register()

-notification()

registration

-uid

-name

-address

-aadaar

-type

-pwd

+register()

-notification()

Booking

-bookingid

-sid

-pid

-vid

-notification()

1..\*

1..\*

\*..1

\*..1

1..\*

1..1

**CHAPTER 5**

**CODING**

5.1 Features of 1anguage

PHP

PHP is mainly focused on server-side scripting, so you can do anything any other CGI program Can do, Such as collect from data, generate dynamic page content, or send receive cookies. But PHP can do much more.

There are three main areas where PHP scripts arc used.

Server-side scripting. This is the most traditional and main target field for PHP. You need three things to make this work, The PHP parser (CGI or server module). a webserver and a web browser. You need to run installation. You can access the PHP program output with a web browser, viewing the PHP page through the server. All these can run on your home machine if you are Just experimenting with PHP programming.

The web server, with a connected PHP Command line scripting. You can make a PHP script to run it without any server or browser. You only need the PHP parser to use it this way. This type of usage is Ideal for scripts regularly executed using cron(on Linux) or Task Scheduler (on Windows). These seripts can also be uscd for simple text processing tasks. Writing desktop applications. PHP is probably not the very best language to create a desktop application with a GUI, but if you know PHP very well, and would you like to use some advanced PHP features in your client-side applications you can also use PHP-GTK to write such programs. You can also have the ability to write cross plat form applications this way. PHP-GTK is an extension to PHP, not available in the main distribution.

One of the strongest and most significant features in PHP is its support for a wide range of databases. Writing a database-enabled web page is incredibly simple. The following databases are currently supported:

When someone visits your PH webpage your web Server processes the PHP code then sees which part it needs to Show to visitors (content and picture s) and hides other stuff (file operations, math calculations, etc) then translates your PHP into the HTML. After the translation into HTMIL. It sends the webpage to your visitor's web browser.

DATABASE

MySQL is [free and open-source software](https://en.wikipedia.org/wiki/Free_and_open-source_software) under the terms of the [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License), and is also available under a variety of [proprietary](https://en.wikipedia.org/wiki/Proprietary_software) licenses. MySQL was owned and sponsored by the [Swedish](https://en.wikipedia.org/wiki/Sweden) company [MySQL AB](https://en.wikipedia.org/wiki/MySQL_AB), which was bought by [Sun Microsystems](https://en.wikipedia.org/wiki/Sun_Microsystems) (now [Oracle Corporation](https://en.wikipedia.org/wiki/Oracle_Corporation)).[[8]](https://en.wikipedia.org/wiki/MySQL#cite_note-sunacquire-8) In 2010, when Oracle acquired Sun, Widenius [forked](https://en.wikipedia.org/wiki/Fork_(software_development)) the [open-source](https://en.wikipedia.org/wiki/Open-source) MySQL project to create [MariaDB](https://en.wikipedia.org/wiki/MariaDB" \o "MariaDB).

MySQL is a database management system that allows you to manage relational databases. It is open source software backed by Oracle. It means you can use MySQL without paying a dime. Also, if you want, you can change its source code to suit your needs.Even though MySQL is open source software, you can buy a commercial license version from Oracle to get premium support services.MySQL is pretty easy to master in comparison with other database software like Oracle Database, or Microsoft SQL Server.MySQL can run on various platforms UNIX, Linux, Windows, etc. You can install it on a server or even in a desktop. Besides, MySQL is reliable, scalable, and fast.

MySQL is a component of the [LAMP](https://en.wikipedia.org/wiki/LAMP_(software_bundle)) [web application](https://en.wikipedia.org/wiki/Web_application) [software stack](https://en.wikipedia.org/wiki/Software_stack) (and [others](https://en.wikipedia.org/wiki/List_of_AMP_packages)), which is an acronym for [Linux](https://en.wikipedia.org/wiki/Linux), [Apache](https://en.wikipedia.org/wiki/Apache_HTTP_Server), MySQL, [Perl](https://en.wikipedia.org/wiki/Perl)/[PHP](https://en.wikipedia.org/wiki/PHP)/[Python](https://en.wikipedia.org/wiki/Python_(programming_language)). MySQL is used by many database-driven web applications, including [Drupal](https://en.wikipedia.org/wiki/Drupal), [Joomla](https://en.wikipedia.org/wiki/Joomla" \o "Joomla), [phpBB](https://en.wikipedia.org/wiki/PhpBB" \o "PhpBB), and [WordPress](https://en.wikipedia.org/wiki/WordPress" \o "WordPress).

MySQL  is the  world's most  popular open source  database software, with  over 100 million copies of  its software downloaded or distributed  throughout its history. With its superior  speed, reliability, and ease of use, MySQL  has become the preferred choice for Web, Web 2.0,  SaaS, ISV, Telecom companies and forward-thinking corporate  IT Managers because it eliminates the major problems associated  with downtime, maintenance and administration for modern, online applications.

Many  of the  world's largest  and fastest-growing  organizations use MySQL  to save time and money powering  their high-volume Web sites, critical  business systems, and packaged software — including  industry leaders such as Yahoo!, Alcatel-Lucent, Google,  Nokia, YouTube, Wikipedia, and Booking.com.

The  flagship  MySQL offering  is MySQL Enterprise,  a comprehensive set of  production-tested software,  proactive monitoring tools, and  premium support services available  in an affordable annual subscription.

MySQL  is a key  part of LAMP  (Linux, Apache,  MySQL, PHP / Perl / Python),  the fast-growing open source  enterprise software stack. More  and more companies are using LAMP  as an lock-in.

The  MySQL  database  is owned,  developed and  supported by Sun   Microsystems, one of  the world's largest contributors  to open source software. MySQL was  originally founded and developed in Sweden  by two Swedes and a Finn : David Axmark, Allan  Larsson and Michael "Monty" Wideners, who had worked  together since the 1980's. The best and the most-used  database in the world for online applications. Available and  affordable for all Continuously improved while remaining fast,  secure and reliable.

HTML with Bootstrap framework

Hypertext Markup Language(HTML) is the standard markup language for creating web pages and web applications.With cascading style sheets(CSS) and javaScript,it forms a triad of cornerstone technologies for the World Wide Web.

Web browsers receive HTML documents from a web server or trom loc and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

Bootstrap is a web framework that focuses on simplifying the development of informative web pages (as opposed to [web apps](https://en.wikipedia.org/wiki/Web_Apps)). The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all [HTML elements](https://en.wikipedia.org/wiki/HTML_element). The result is a uniform appearance for prose, tables and form elements across [web browsers](https://en.wikipedia.org/wiki/Web_browser). In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent pull quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components in the form of [jQuery](https://en.wikipedia.org/wiki/JQuery" \o "JQuery) plugins. They provide additional user interface elements such as [dialog boxes](https://en.wikipedia.org/wiki/Dialog_box), [tooltips](https://en.wikipedia.org/wiki/Tooltip), and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

Bootstrap 1s a front-end framework that is developed to support creating dynamic websites and web applications. It is one of the most preferred front-end frameworKS As it aids an easy and fast processing to develop a website. It supports all major browsers and fast loading responsive web pages.

Bootstrap consists of HTML and CSS-based design templates for various interface components and is aimed to ease web development. By updating the CSs, you can adapt to modem trends quickly. The developers should concentrate more on interaction components as the bootstrap itself will take care of standard views of data. which can be altered later if you wish to.

Bootstrap is compatible with almost all the latest version browsers such as Internet Explorer. Google Chrome. Opera. Firefox. and Safari. It supports the responsive web design and dynamically adjusts the layout of web pages by considering the characteristics of the device used.

Generally, every web project you work on will need to be responsive and work properly on all the major browsers, and likely have some fallbacks for older browsers. Bootstrap has [a huge open source community](https://github.com/twbs/bootstrap) that works on covering this so you don't have to. Additionally, when multiple developers all know the same system, they can work in better harmony - and it also makes it easier for newcomers on a project to get up to speed.

[The grid](https://getbootstrap.com/examples/grid/) is probably one of the most essential aspects of the framework. It's the basis on which the entire layout is created. Beyond that, Bootstrap's [core CSS](https://getbootstrap.com/css/) will also add helpful styling to forms, tables, buttons, lists, and images, as well as fully functioning navigation bars, while the [core JavaScript](https://getbootstrap.com/javascript/) will add helpful code for creating modals, carousels, alerts, popups, dropdowns, and accordions.

JavaScript

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.Client-side JavaScript is the most common form of the language. The script should be included in or referenced by an HTML document for the code to be interpreted by the browser.

It means that a web page need not be a static HTML, but can include programs that interact with the user, control the browser, and dynamically create HTML content.

The JavaScript client-side mechanism provides many advantages over traditional CGI server-side scripts. For example, you might use JavaScript to check if the user has entered a valid e-mail address in a form field.

The JavaScript code is executed when the user submits the form, and only if all the entries are valid, they would be submitted to the Web Server.

JavaScript can be used to trap user-initiated events such as button clicks, link navigation, and other actions that the user initiates explicitly or implicitly.

5.2 Functional Description

Smart parking contains several functions used to perform some specified tasks.Some functions are described below.

1.registerSupplier()

This function is used to create registered supplier.

Public void registerSupplier(int id,char name,int aadha)

{

//$post holds the values from the form that has to be inserted to the database.

}

2.registerDriver()

This function is used to create registered driver.

Public void registerDriver()

{

//$post holds the values from the form that has to be inserted to the database.

}

3.addPropety()

This function is used to create registered property.

Public void addProperty()

{

//$post holds the values from the form that has to be inserted to the database.

}

4.search()

This function is used to search parking area.

Public void search()

{

//$post holds the values from the form that has to selected from the database and displays some values.

}

5.addBooking()

This function is used to book a parking slot.This will be done with a sms.

Public void addBooking()

{

//$post holds the values from the form that has to be inserted to the database.

}

5.addFeedback()

This function is used to add feedback

Public void addFeedback()

{

//$post holds the values from the form that has to be inserted to the database.

}

**CHAPTER 6**

**TESTING**

SMART PARKING follows the SDLC V-model testing

First we collect and understood the product requirements from the customers perspective. This phase involves detailed communication with the customer to understand his expectations and exact requirement. This is a very important activity and needs to be managed well, as most of the customers are not sure about what they need. The acceptance test design planning is done at this stage as exactly they business requirements can be used as an input for acceptance testing.

As we know that, Software Testing is an empirical investigation conducted to provide stakeholders of information about the quality of the product or service under test with respect the contest in which it is intended to operate. Software Testing also provides an objective ,independent view of the software to allow the business to appreciate and understand the risks at implementation of the software.

V-Model

The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as Verification and Validation model.

The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. This is a highly-disciplined model and the next phase starts only after completion of the previous phase.

V-Model- DesignUnder the V-Model, the corresponding testing phase of the development phase is planned in parallel. So, there are Verification phases on one side of the V and Validation phases on the other side. The Coding Phase joins the two sides of the V- Model.

System Design

Once we have the clear and detailed product requirements, it 1s time the complete system. The system design will have the understanding and detailing the complete hardware and communication setup for the product under develop The system test plan is developed based on the system design. Doing this at an earlier stage leaves more time for the actual test execution later.

Architectural Design

Architectural specifications are understood and designed in this phase. Usually more than one technical approach is proposed and based on the technical and financial feasibility the final decision is taken. The system design is broken down further modules taking up different functionality. This is also referred to as High Level Design (HLD).

The data transfer and communication between the internal modules and with the outside world (other systems) is clearly understood and defined in this stage. w1tn this information, integration tests can be designed and documented during this stage.

Module Design

In this phase, the detailed internal design for all the system modules is specified, referred to as Low Level Design (LLD). It is important that the design is compatible with the other modules in the system architecture and the other external systems. The unit tests are an essential part of any development process and helps eliminate the maximum faults and errors at a very early stage. These unit tests can be designed at this stage based on the internal module designs.

Coding Phase

The actual coding of the system modules designed in the design phase is taken up in the Coding phase. The best suitable programming language is decided based on the system and architectural requirements.

The coding is performed based on the coding guidelines and standards. The goes through numerous code reviews and is optimized for best build is checked into the repository.

Validation Phases

The different Validation Phases in a V-Model are explained in detail below

Unit Testing

Unit tests designed in the module design phase are executed on the code during this validation phase. Unit testing is the testing at code level and helps  eliminate bugs at an early stage, though all defects cannot be uncovered by testing.

 Integration Testing

Integration testing is associated with the architectural design phase. Integration tests are performed to test the coexistence and communication of the internal within the system.

System Testing

System testing is directly associated with the system design phase. System tests check the entire system functionality and the communication of the system under development with external systems. Most of the software and hardware compatibility issues can be uncovered during this system test execution.

Acceptance Testing

Acceptance testing is associated with the business requirement analysis phase and involves testing the product in user environment. Acceptance tests uncover the compatibility issues with the other systems available in the user environment. It also discovers the non-functional issues such as load and performance defects in the actual user environment.

Security Testing

Security testing is a testing technique to determine if an information system protects data and maintains functionally as intended. It takes this test for user authentication. We test login process with the help of security testing. As the result of this testing, unauthorized persons cannot enter the system.

**CHAPTER7**

**IMPLEMENTATION**

7.1 IMPLEMENTATION OF THE PROPOSED SYSTEM

The implementation is the final stage and it's an important phase. It involves the individual programming, system testing, user training and the operational running of developed proposed system that constitute the application subsystems. Once major task of preparing for implementation is education of users, which should really have been taken place much earlier in the project when they he investigation and design work. During this implementation phase system actually takes physical shape.

7.2 INSTALLATION PROCEDURE

Install server software

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the internal users and the operation of the system. In many organization someone who will not be operating it, will commission the software development project. The people who are not sure that the software meant to make their job easier. In the initial stage. they doubt about the software but we have to ensure that the resistance does not build up as one has to make sure that the active user must be aware of the benefits of using the system proper guidance imparted to the user so that he is comfortable in using the application. Their confidence in the software is built up.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server objects 1s not running on the server, the actual processes will not take place.

The task of installing a new system has degree of difficulties. Since thehardware is already running, only new software system has to be installed, in this cases there arise some issues. They are,

Is there area big enough?

Is the layout appropriate?

Is the environment appropriate?

Does it require an additional hardware?

Is this site secure?

The following components are needed for the installation procedure.

1. Web Server (Wamp/Xampp)
2. PHP 7.1+
3. Database 5.6+ (MySQLi suggested)
4. Required PHP libraries/modules

After the installation create the files in the web server root folder and then create the database then with the web browser run the system. The interface between all the sub systems is checked.

CHAPTER 8

CONCLUSION

The primary objective of the proposed system is to easily find out the nearest parking slots. Thus project is intended to computerize the booking procedures. The newly developed system consumes less processing time and all the details are updated and processed immediately. This study has proposed a parking system that performance for minimizes the cost of moving to the parking space and reducing the number of users that fail to a parking space. This smart parking system provides better performance .This system provides a fast and systematic approach for finding the nearest parking slot through Just a click. The system consist of a mobile application that allows users to access various parking slots. In existing it take too much time for finding a parking slot.

Varoius Features

1. Time saving.

2. Easy to find the nearest available parking slot for booking through map

3. Easy to make better budget plan based on rate of parking area

4. Users will get notifications through a mobile application on

booking is completed.

5. Interface of the system uses only less time to load.

6. Performance of the system will not degrade even in slow internet connections

.

.

**CHAPTER 9**

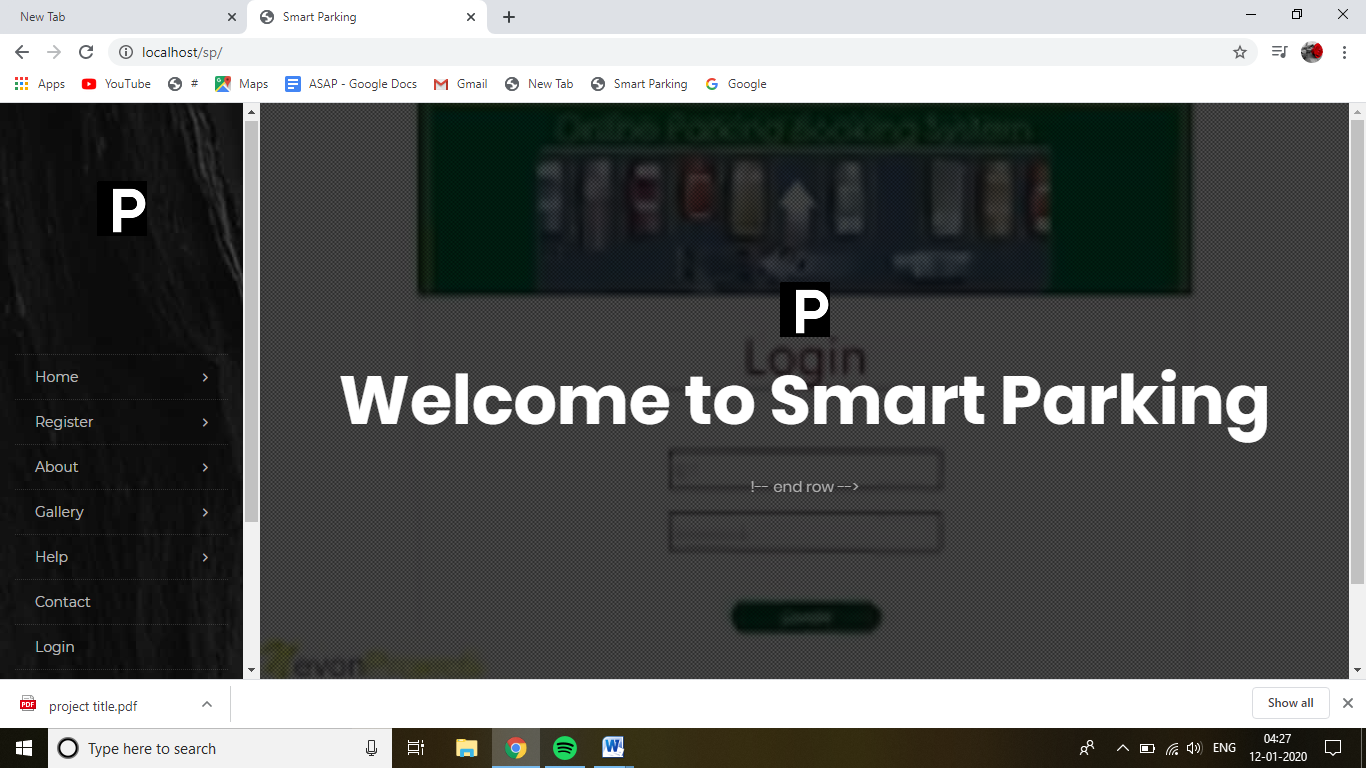
**FUTURE ENHANCEMENTS**

For the future work, a supplier and driver can anonymously send a supplying report and a parking query to the authority. Meanwhile, a trusted authority is able to disclose a user’s identity if a dispute occurs and users achieve anonymous payment with E-cash. The main future enhancement is the security measure to ensure that the users’ do not misuse the parking system can be implemented.

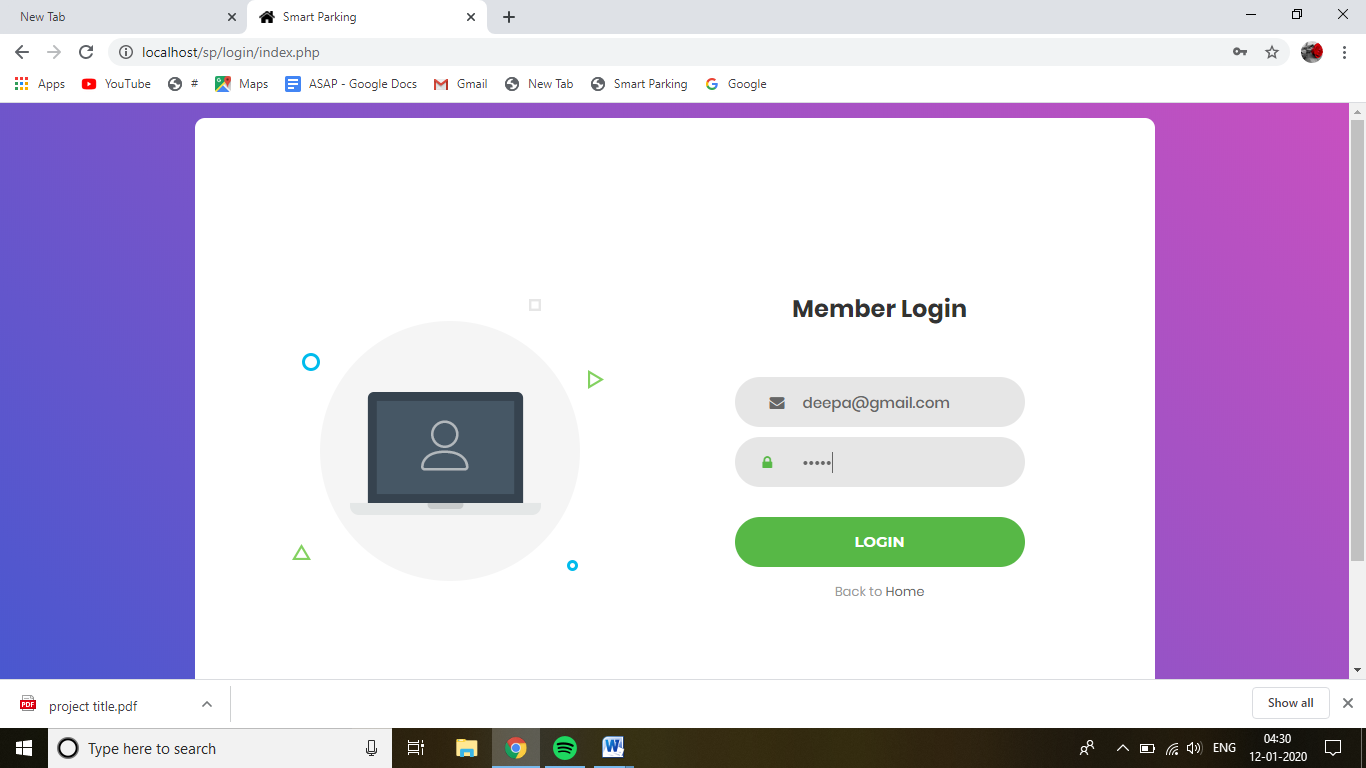
We will consider detecting location attack from drivers in advance, meaning a driver may send a parking query to the authority long he arrives at the destination area. In future our system also supports ﬁnding public parking spot. We will design a privacy preserving smart-parking and payment scheme can be implemented.

**APPENDIX**

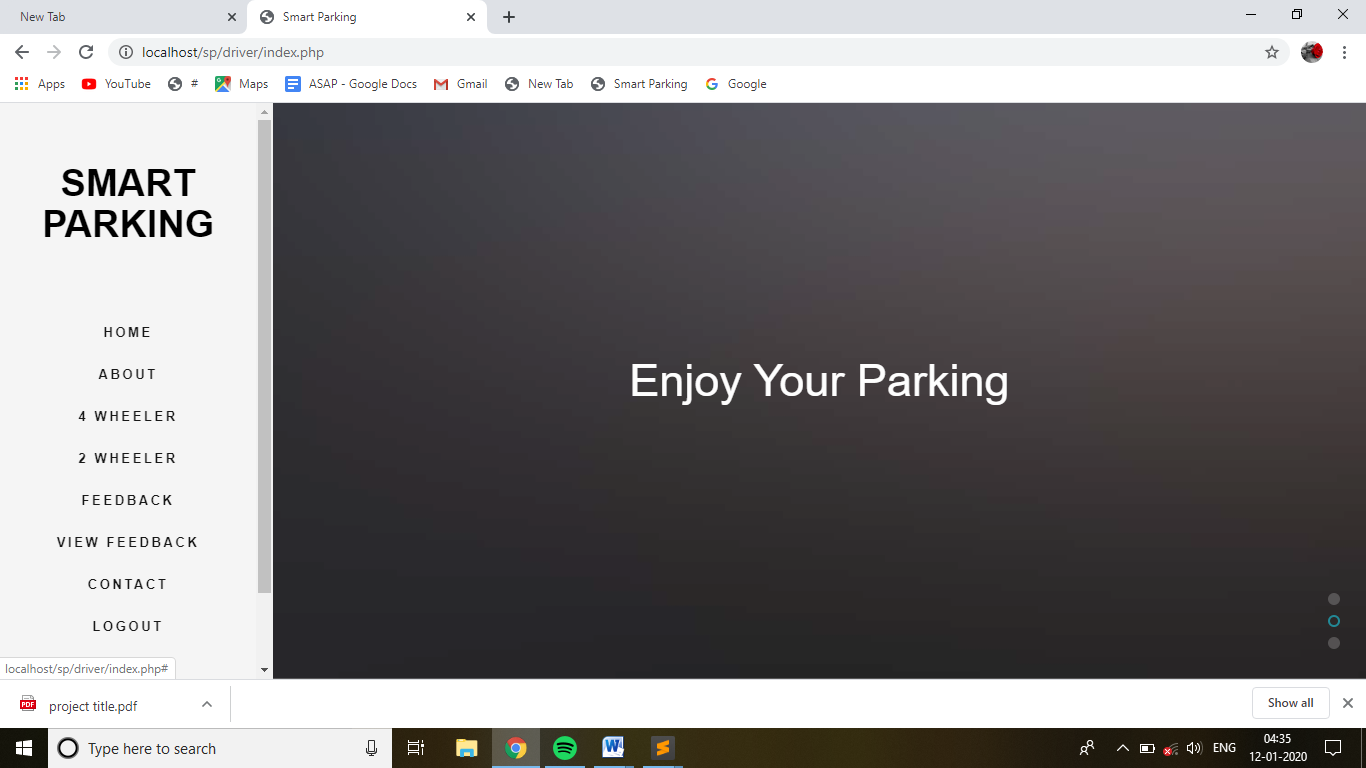
Index



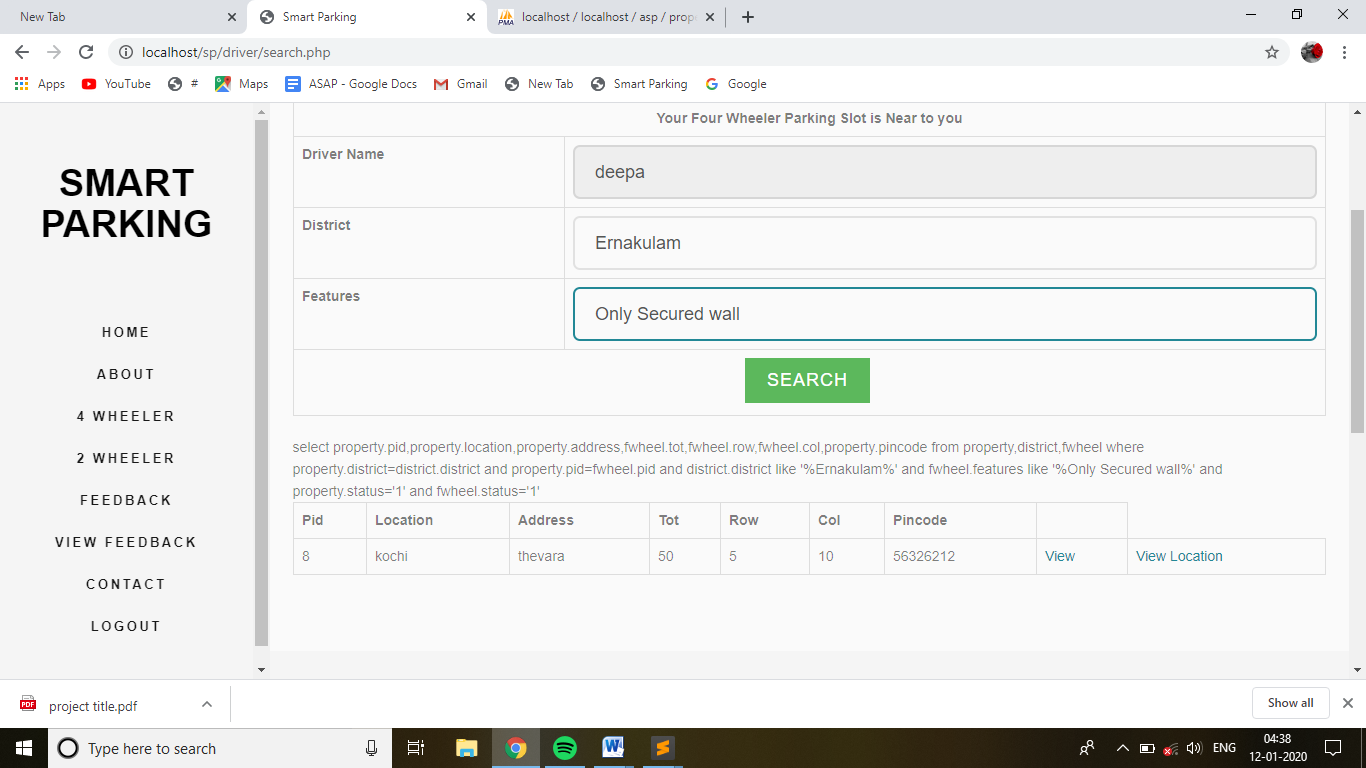
Login



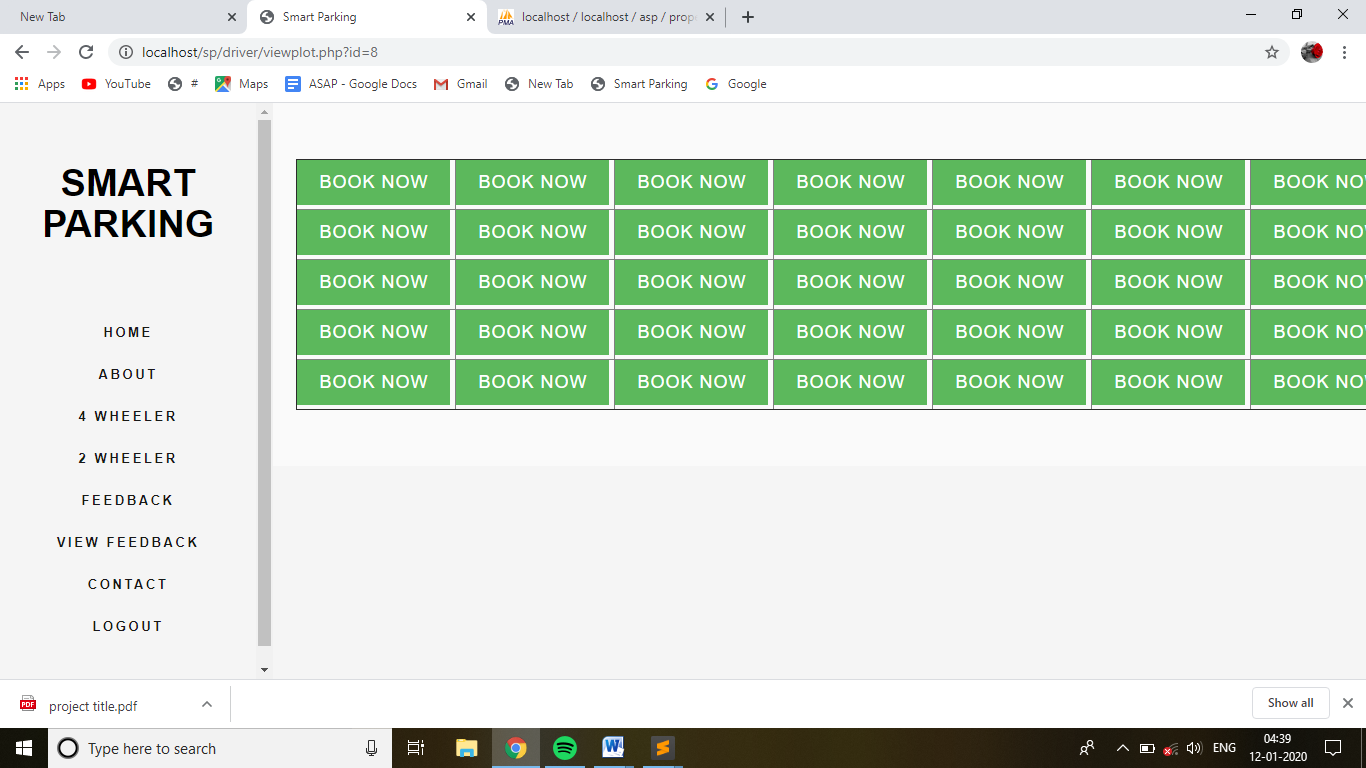
Driver Home



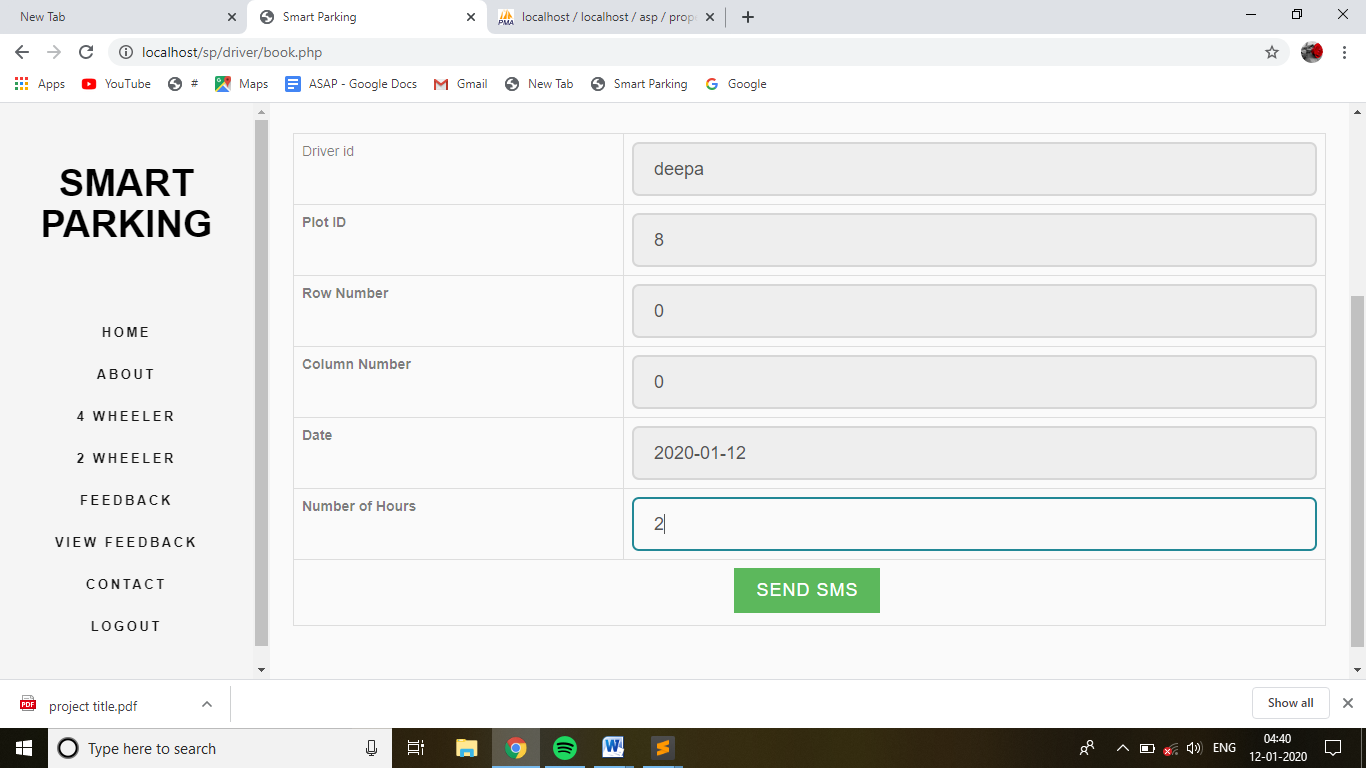
Search parking slot



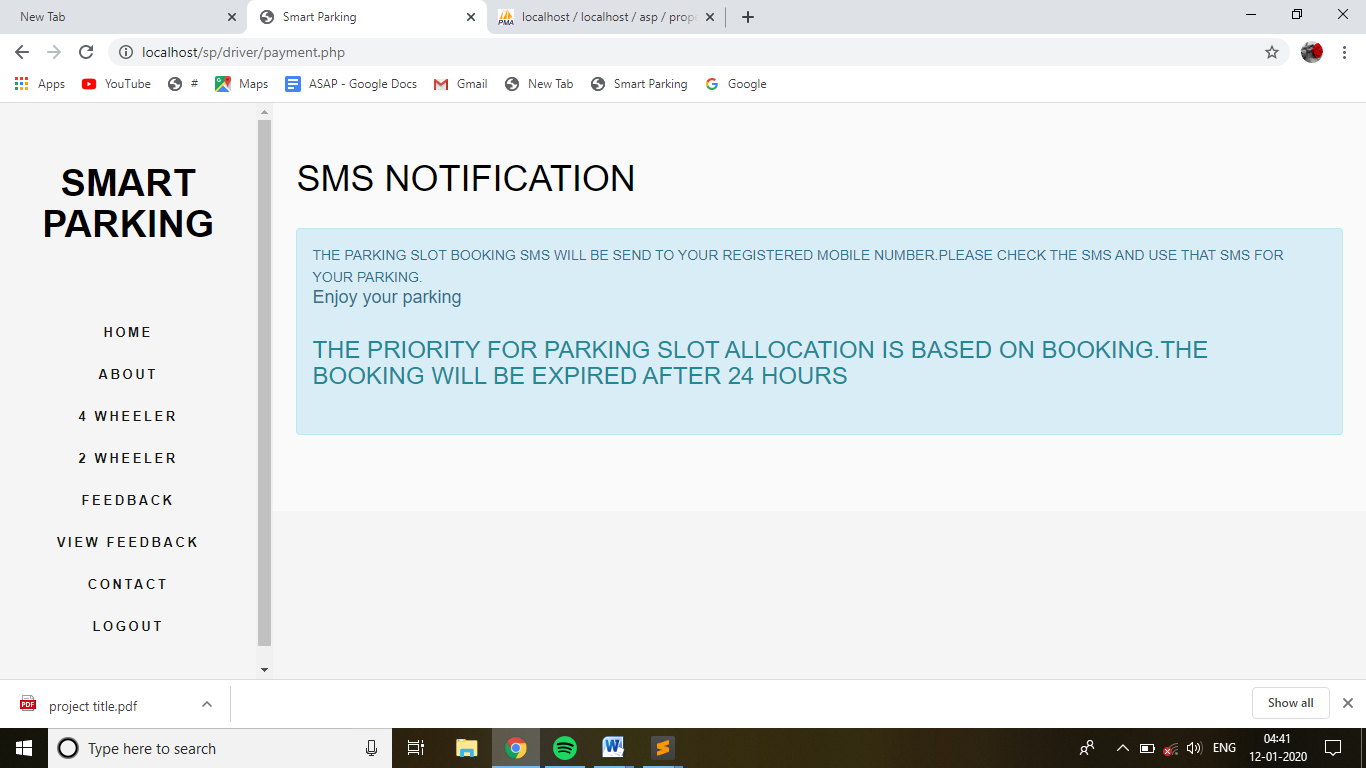
Viewplot



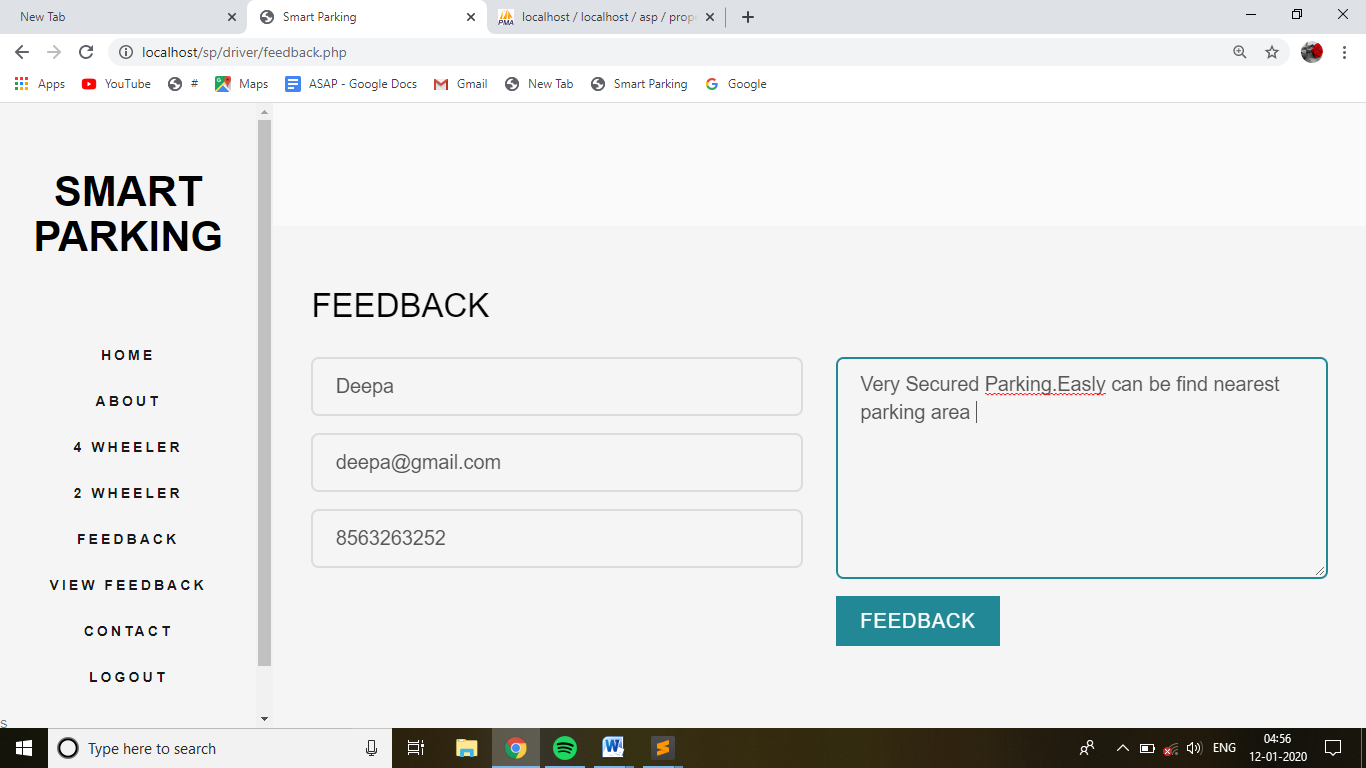
Booking



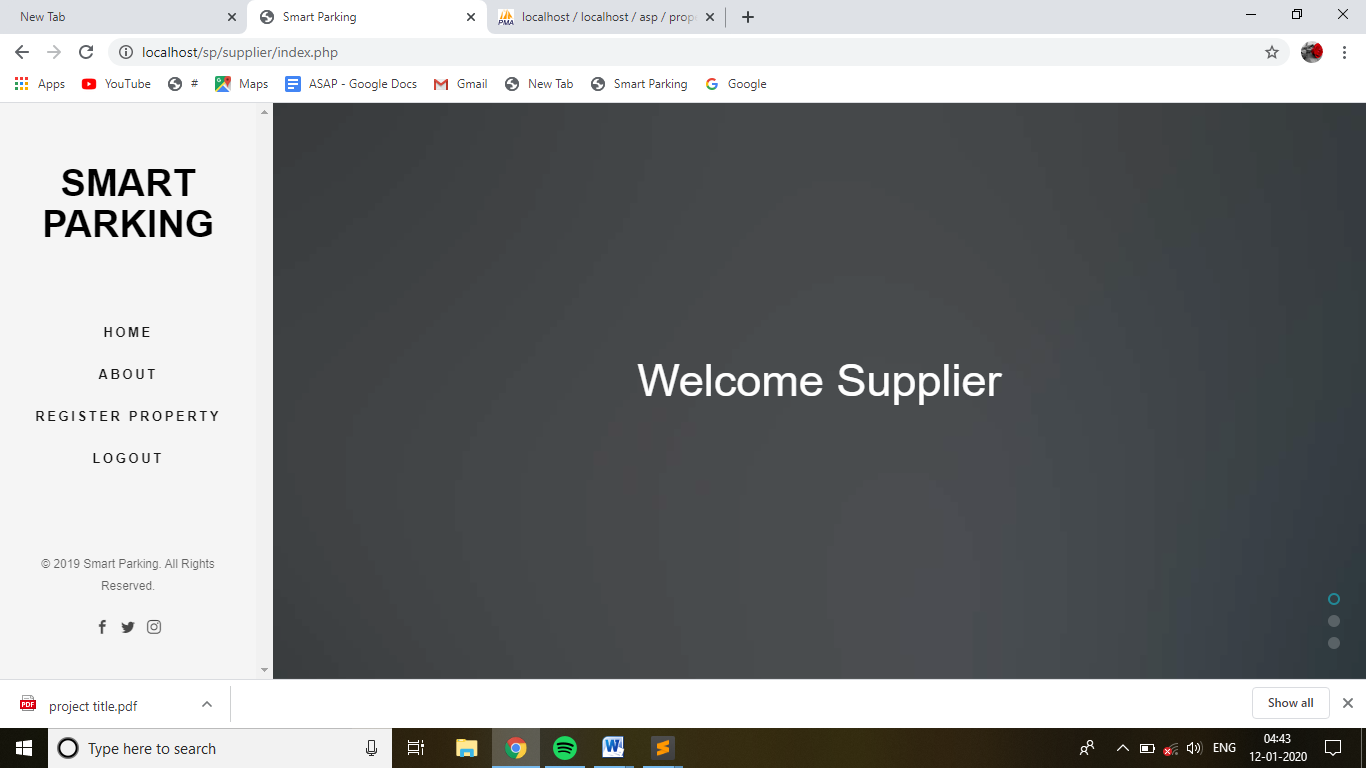
SMS notification



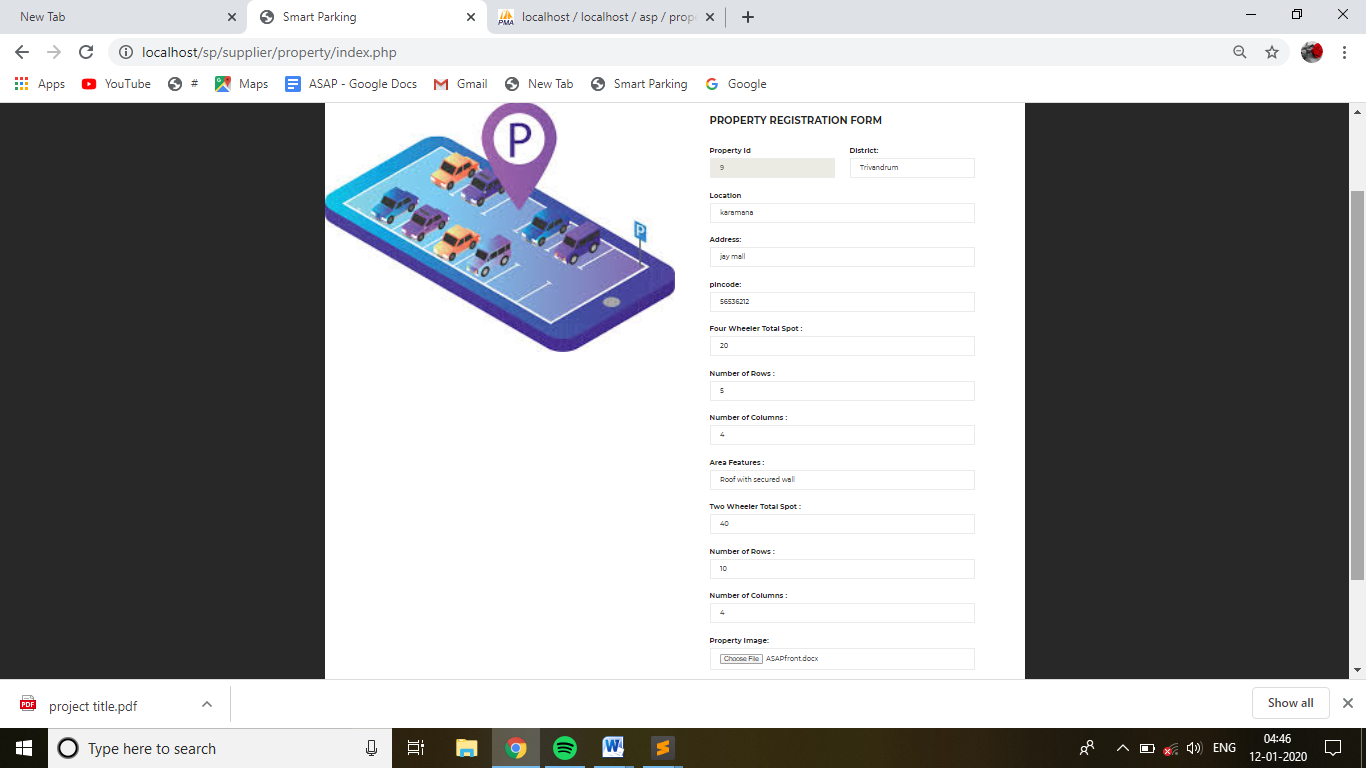
Feedback



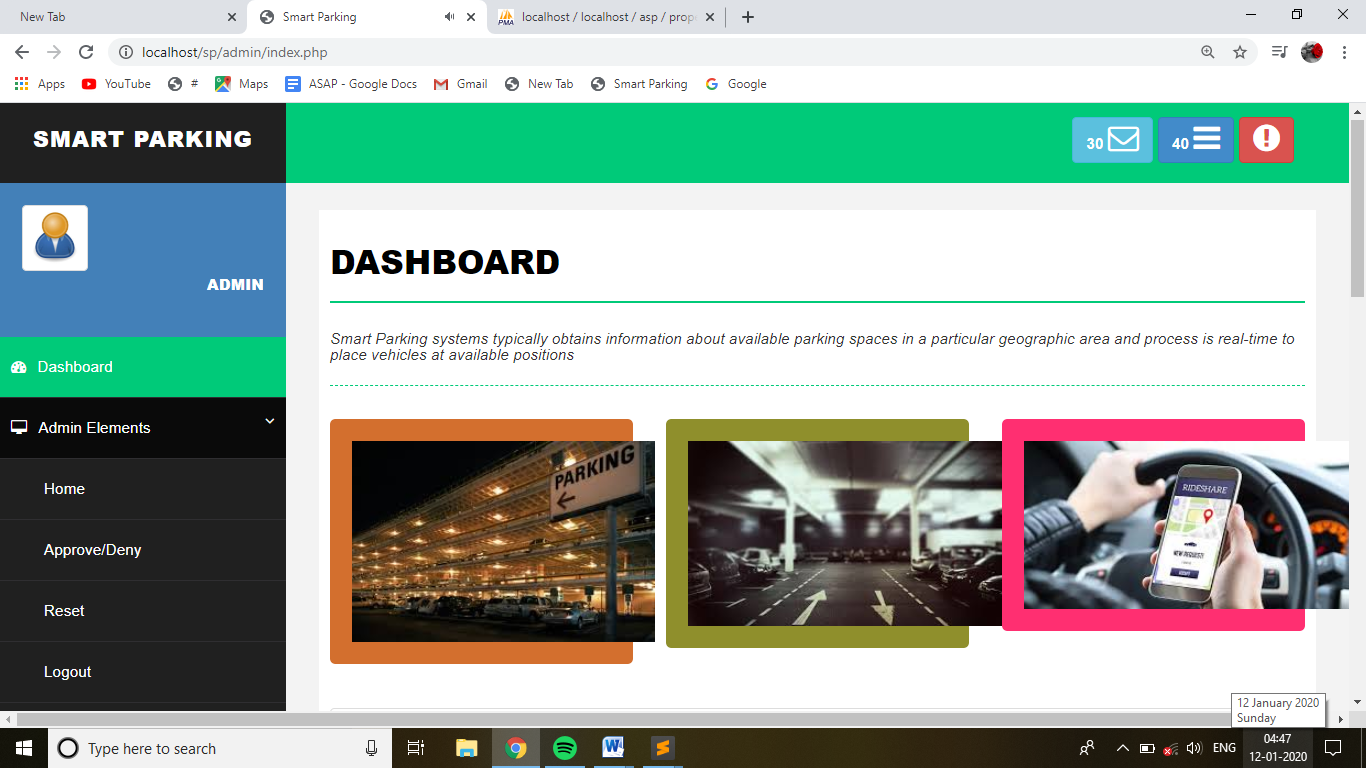
Supplier Home



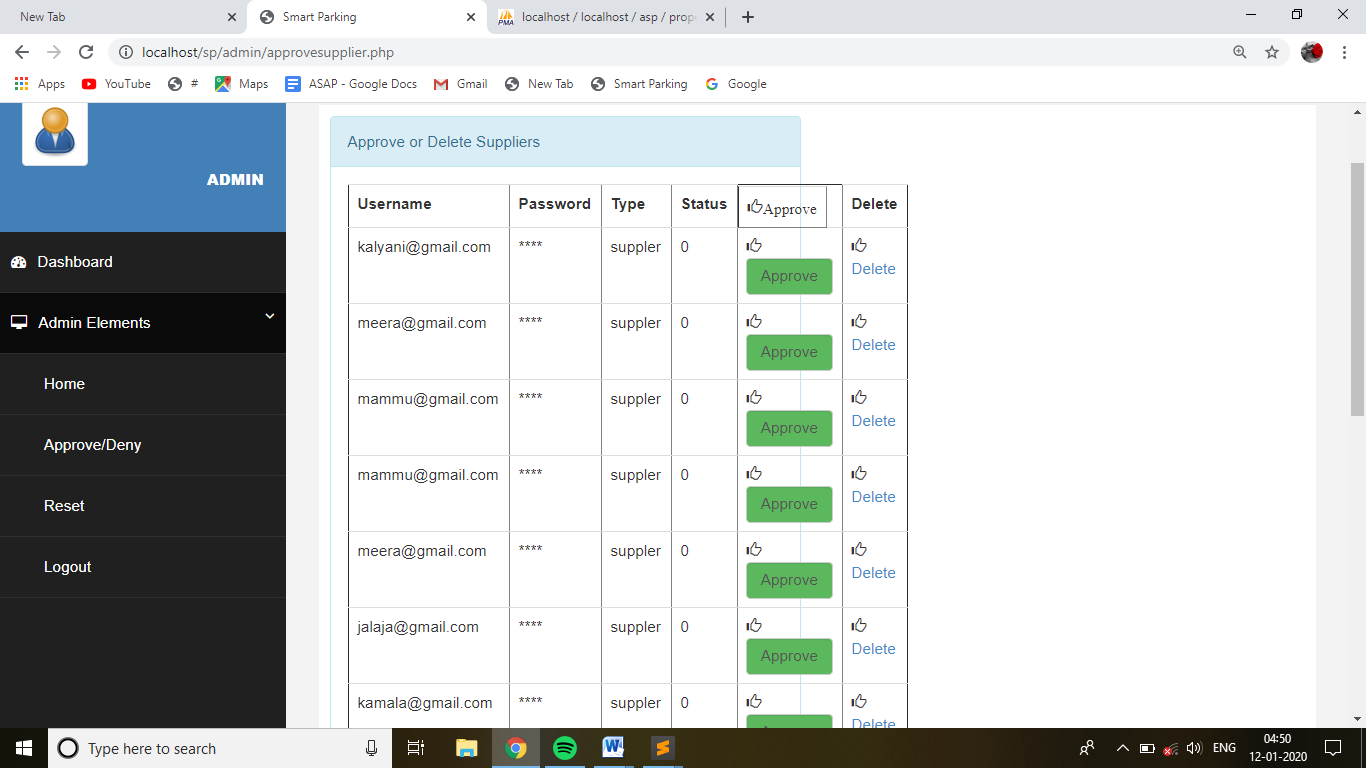
Register Property



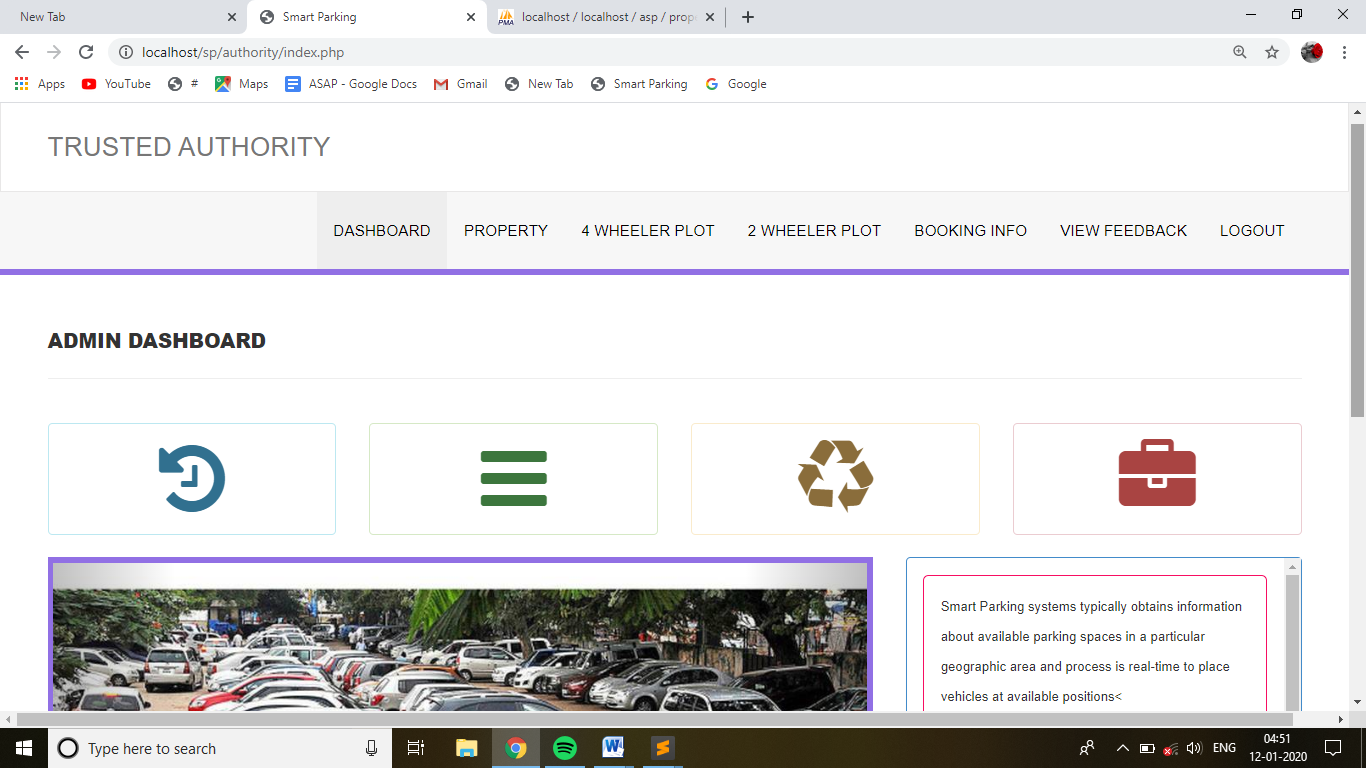
Admin Dashboard



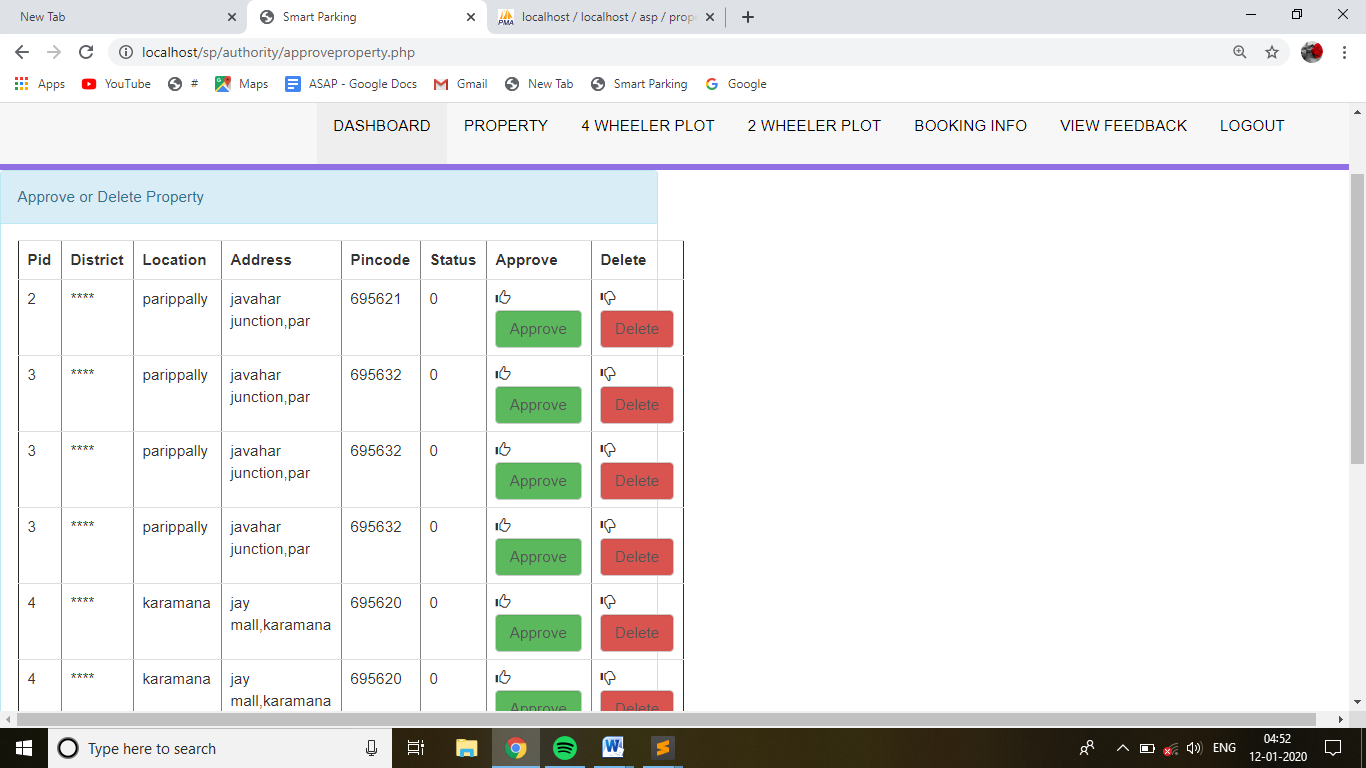
Approve/deny



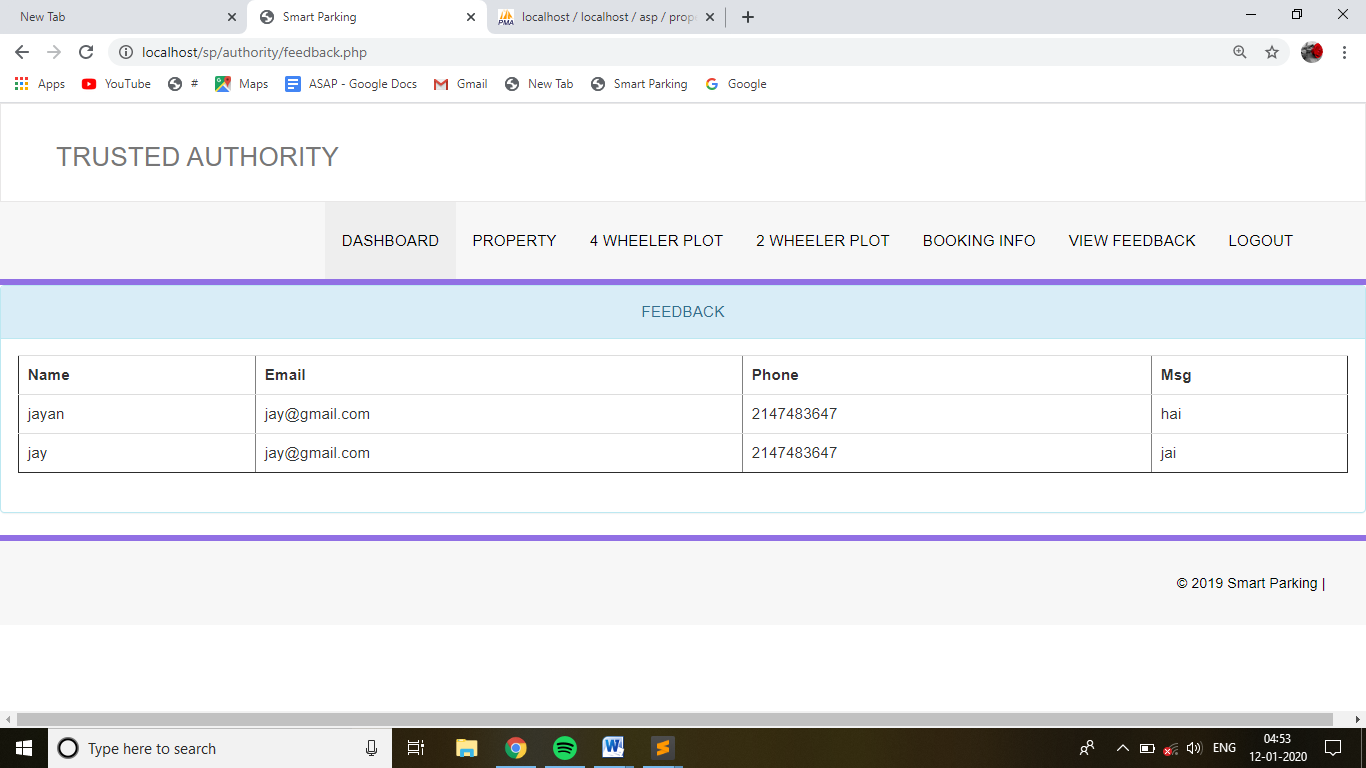
Authority Dashboard



Approve/deny property



View Feedback



**BIBILIOGRAPHY**

Fundamentals of Software Engineering-Rajib Mall-

Page No:167-202

Page No:248-279

<https://www.w3schools.com/css>

<https://www.w3schools.com/sql>

<https://www.w3schools.com/booTsTrap/default.asp>