[Document title]

[Document subtitle]

Abstract

[Draw your reader in with an engaging abstract. It is typically a short summary of the document.
When you're ready to add your content, just click here and start typing.]

Bond
[Email address]

Table of Contents

Abstract

This document is to demonstrate why our website is here for. Through our research we have found out that there is a demand for car parking spaces in Dublin because there is an increase of people driving to the city centre. According to our research, Ireland and especially Dublin, has become one of the greatest economies in Europe and it has the potential to grow even more and has become a centre for business and creativity. Our goal was to develop a website that can be accessible to anyone in Dublin. In order to build this software and to keep track of our website, we had to use work frame to set targets in order to finish a good prototype.We found that to find a spot to park has become more difficult and there are many studies that show that you can easily spend up to €200 monthly to make sure you have a spot when you need it. Our website allows the user to be flexible with their parking arrangements. They do not need to book a weekly or monthly pass and can tailor their booking exactly to their needs. They can also easily cancel a booking if their plans change on short notice. The user will receive a confirmation of theirbooking instantly. This can help any driver save time, create less traffic and change the way on-street and private parking is managed around Dublin.

Introduction

For the past few years, Ireland has been growing economically.You can notice this simply by walking around Dublin City centre.Finding a parking spot can be a major problem in an urban area, especially for a well-developed city such as Dublin. As the number of people increases so does the number of vehicles.

Our car parking system helps a user book a parking spot quickly and easily in a commercial area.The system is designed in such a way that a user can find a parking spot from the comfort of their home. This process can be done on either a smartphone or a laptop.This can be done according to their liking. Once a user has created a login, they have almost total control over how they book their spot such as what level they would like to park on and the duration they would like to book for i.e. an hour, a full day, a week or even a month.

The main objective of our project is to build a parking system that will help the user save time in finding a parking spot in this crowded world we live in. We have developed an entirely cloud-based parking system that can be deployed on a large scale very quickly with minimal infrastructure.

Background to the Problem and Proposed Solution

Project aims

We are aiming to build a car parking website based on user needs. According to the Central statistics Office," there was an increased number of car drivers in the state. In the State overall, 61.4 per cent of working commuters drove to work in 2016. Car drivers increased from 402,878 in 1986 to 1,152,631 in 2016, an increase of 749,753 drivers or 186 per cent." The problem of finding a car park space has become so bad that car park spaces are available to rent on daft.ie for up to €230 a month (https://www.daft.ie/dublin/parking-spaces/, accessed 10/05/2020). Having a website where a person can find an online car park space is a brilliant and useful idea. Long term rentals without contractual commitment allow users to tailor their usage exactly to their needs. As working from home is becoming more common, a person who might only travel to the office 2-3 days a week can book a spot only for those days and not waste money having to buy a weekly or monthly pass or pay a very high hourly rate as in other car parks around the city.

Learning the principles of the internet and how software works in a realistic way has influenced the scope and direction of our project.There are free resources and free program software used in college that will be a starting point for our team to begin our construction of the car parking system website.

One of the goals is to reduce the amount of traffic and tosimplify the user's life when searching for a spot in Dublin.

A surprising fact about parking

"The world first parking meter according to B4 Parking blog the planet's first installation of a parking meter was Park-O-Meter. Designed to solve the lack of sufficient parking space in America's urban areas as increasing numbers of cars crowded into the business district each day, Park-O-Meter was installed in Oklahoma on 16th July 1935. Back in the day, parking cost $0, 05 per hour – sounds beyond ridiculous today, but that was an expensive parking spot for the year 1935."

Parky, 2017

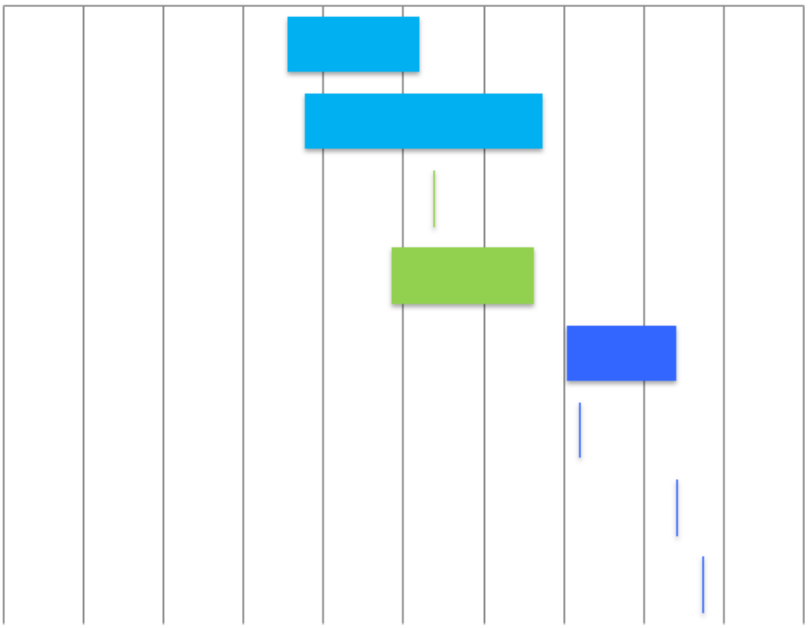
The cost of parking has been slowly rising around the country for years. Rates in certain areas of Dublin city centre increased by 10% in July 2019 (Irish Times, 2019). The parking meter was a revolutionary piece of technology in its day. However, as technology has advanced over the years, a technological solution is needed in this modern-day world. Our project overcomes the problem of finding a parking space in commercial areas that unnecessarily consumes time. On street parking is very expensive and there is a shortage.

Proposed Solution

We are designing a smart parking booking system that provides customers an easy way of reserving a parking space online. Users can book a space for a specific time slot. This system also provides additional featuressuch as cancelling the booking. Users can cancel parking booking space anytime. Users can even make payment online via credit card. After making payment users are notified about the booking via email along with unique parking number.Research shows that more than 66% of drivers do not mind paying for car parking facilities during their working hours (Bilodeau, 2010). Thisadds value to the car parking business, which was a factor in our choice to develop an intelligent car parking service.

When the user logs in then selectsa spot, they will get a message on their email and/or mobile saying their spot has been booked. This system helps with traffic management in a smart city as well because users can see from the website whether parking is available or not. This influences their choice of whether to go into the city that day or not.

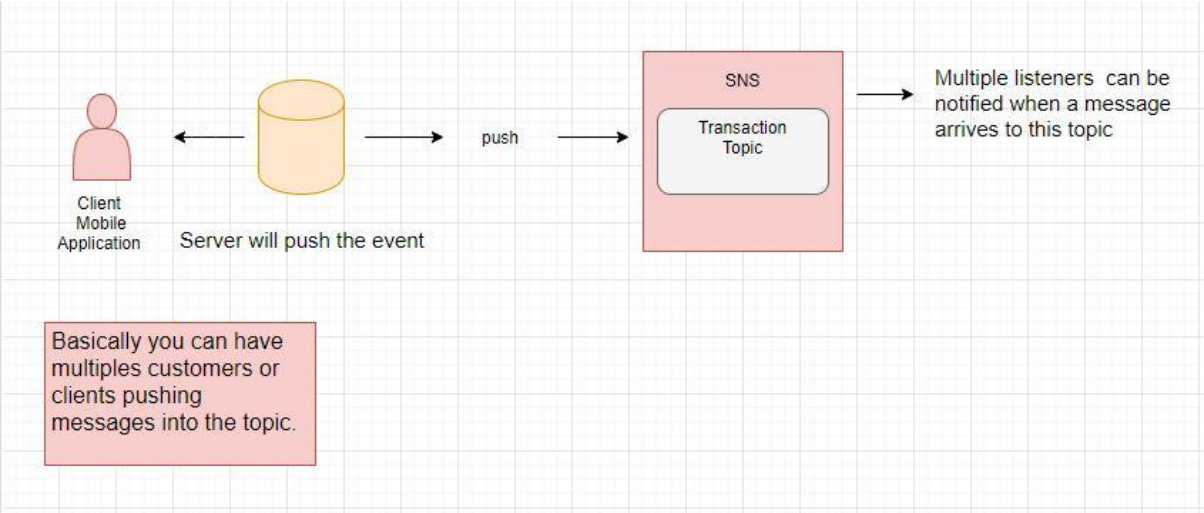
Project Planning (Team, and Roles and Responsibilities with Project Timeline)



	Presentation	Project Submission	April Prototype	Coronavirus Pandemic	Project Planning	Project Proposal	Team Meetings	Research for Project Proposal
START DATE	06/04	05/19	03/19	03/12	11/23	12/20	09/30	09/20
DURATION in days	1	1	1	68	89	1	149	82

Technology Selection:Technologies we have used in this project

Amazon Simple Notification Service (SNS) Cloud



In cloud computing, everything is connected to an infrastructure. We are using an IPA with command tool. Amazon SNS is a messaging service that is a push-based system, like SQL, using many to many concept messaging. Using SNS, the system can send out a large number of messages to a large number of subscribers. The Amazon SNS provides a secure data transfer. The messages will always be encrypted and will be in a secure channel such as https.

Figure 1: Creating an account

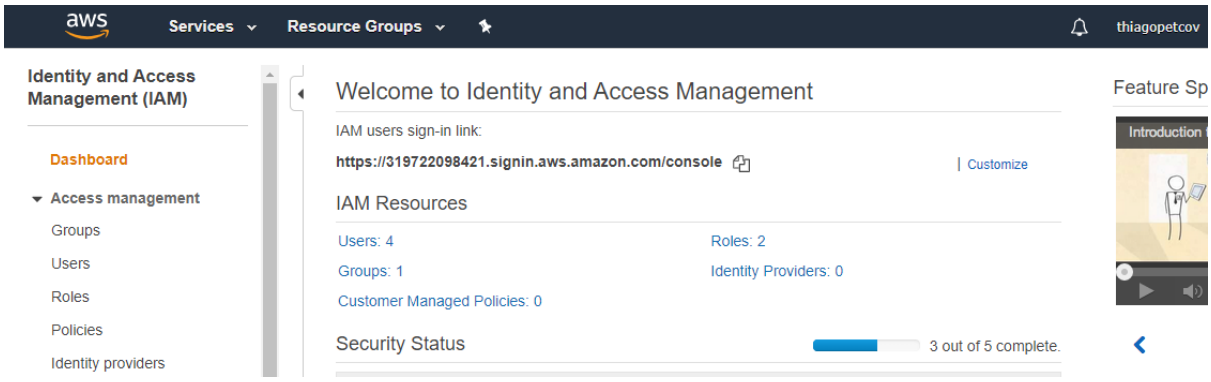


Figure 2: Creating the SNS

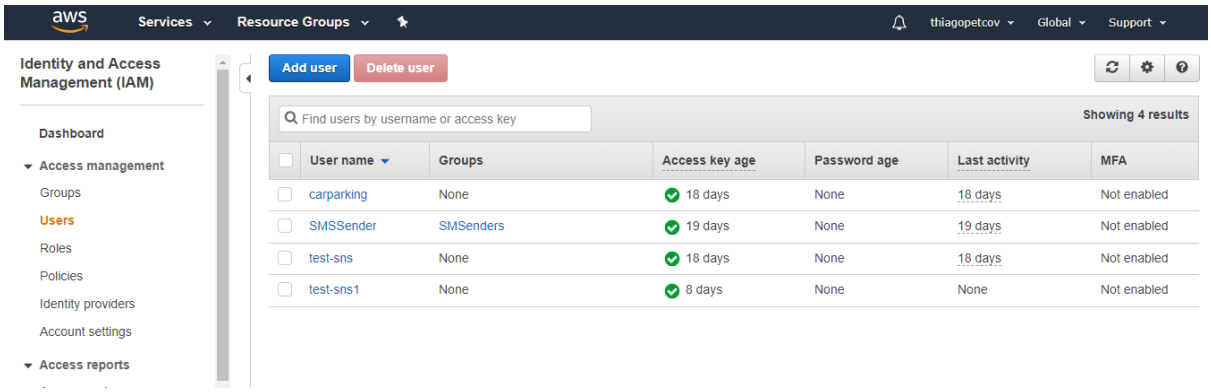


Figure 3: Testing user details

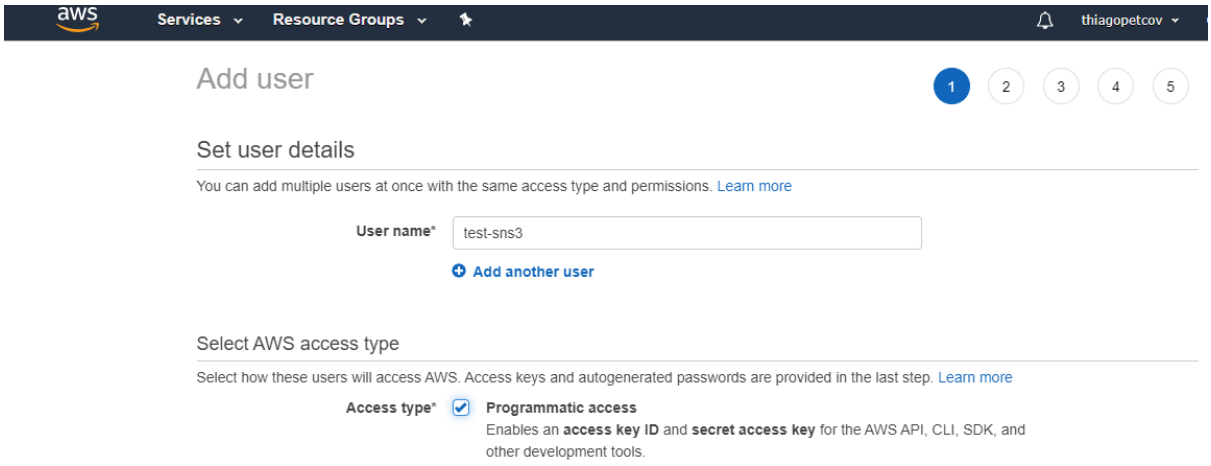
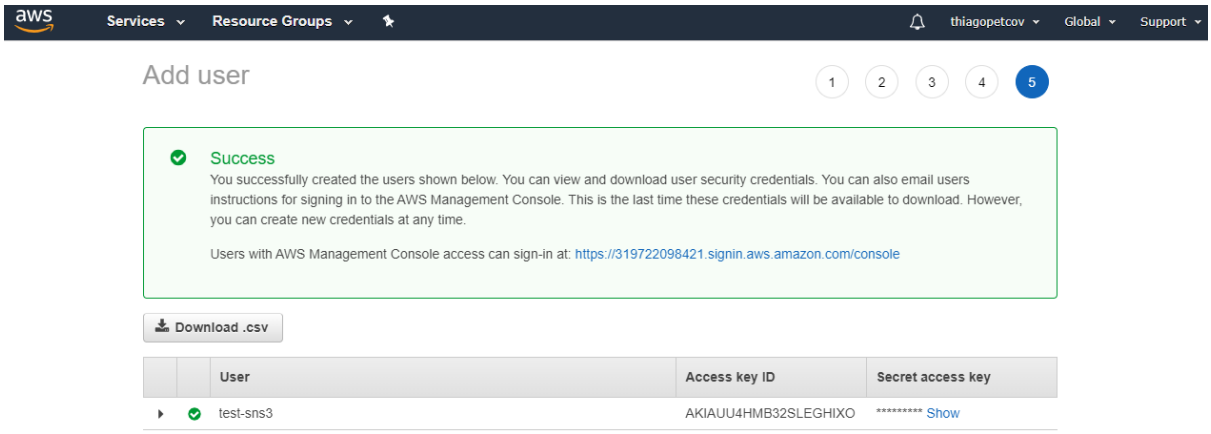
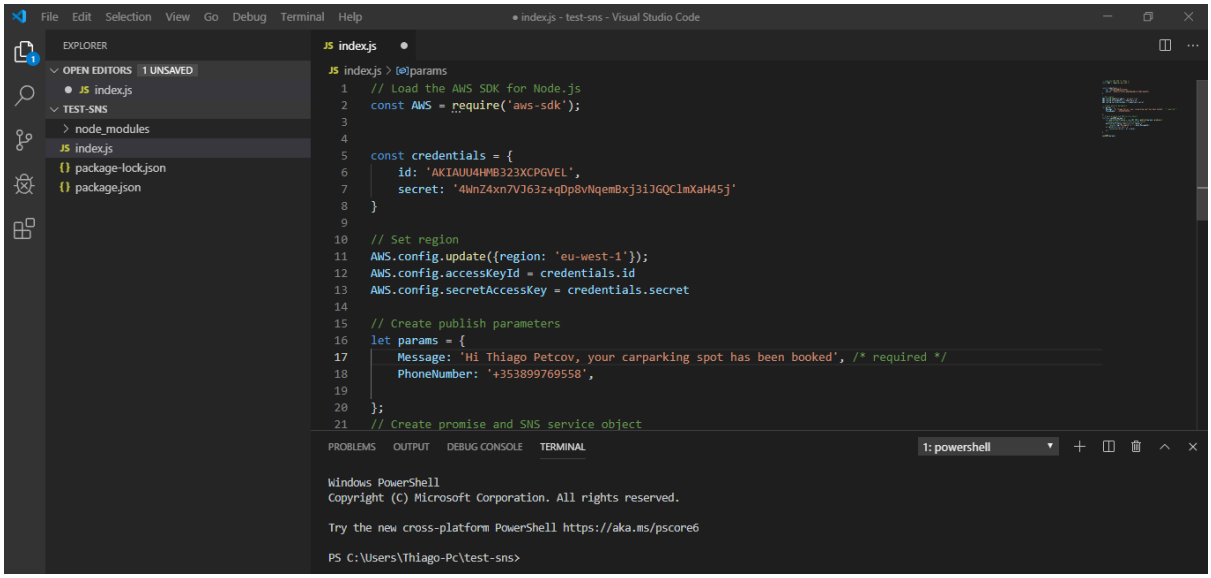
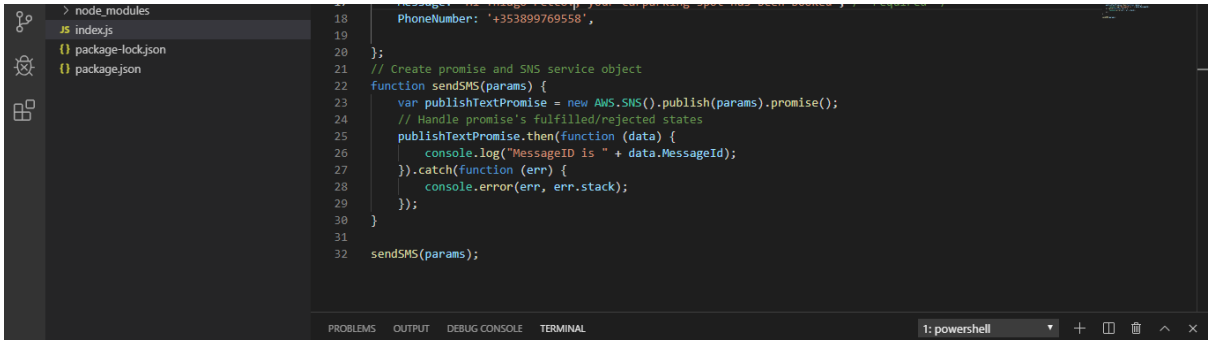


Figure 4: Generating access keys



Algorithm responsible for running the Simple mail Notification





Benefits that we derive from implementing a Cloud Computing solution:

Cost

Cloud computing is the future. In a cloud service, you pay for what you need. Any company hoping to use our website does not need to buy servers or any physical hardware to use our website. This can reduce overall costs for a company because everything is provided by the cloud provider.

Scalability

We imagine a launch of our website for a small to medium car park in Dublin city centre but once the brand has been established, cloud computing allows us to scale up operations to different areas around Ireland and also to different types of parking i.e. private businesses, on-street etc.

Performance

Cloud computing services are substantial and run on a worldwide network. Often data centres are well secure and are upgraded regularly. A company can reduce their physical network therefore doing more with less. This is more economical in the long run.

Other benefits include security, speed, productivity and reliability.

The languages used and suitable logic

HTML: Hypertext mark-up language. Html will define the web content, web images and web links and will gather all the content together.

The reason to work with HTML and CSS was because we are more familiar with these technologies and we have already completed some projects using them. We could have used bootstrap which is a powerful toolkit including a collection of HTML, CSS, and JavaScript tools for creating and building web pages and web applications. However, we are familiar with working individually with each language and decided that would be more suitable for this project as everyone could work on separate areas.

CSS: Cascading style sheets. CSS works like an artist, in my words, or designer to make the images well organised, colours organised and the titles the same size and style.

JSON: stands for Java Script Object Notation. It is a format that you can send data from client to server and from server to client. In order to run the SNS messages from Amazon, there was a need to use JSon and java script was the best language to work with. We had some issues implementing that because the logic of our code was built in PHP and we are using Apache xampp as a local host

The reason to store our database on XAMPP and run it from there was because is easy to use and free to install.You can use an apache distribution containing MySQL and PHP.

HTML, CSS, and Java Script work together as a team nowadays and they are responsible for maintaining the front end user interface which is the client side.

PHP is mostly doing the work.All variables, arrays, functions and switching pages are written in PHP. As a group, we decided at the beginning that it would be more useful and easy to understand and work within.We could have used python to build our prototype but we just started to learn at the end the semester in college.We would have many small problems if we decided to learn on the goto build our website.

We could have used java object oriented as we have been learning that since year one. We thought that specific language would bring more trouble shooting problems and would delay our deadline for the prototype.

Apache interprets PHP code which is a server side language and it will run on the top of our webserver.We chose this language because we can use Apache. This is built into the XAMPP system. This streamlines the system.

MySQL is used for the database.The advantage of using XAMPP from the control panel is that it allows you to control your server. It was just simpler to use. From the configuration button, you can manage your database easily and Apache, PHP or MySQL.

We could have used MySQL workbench but we usually had problems with port Numbers on the configuration. On xampp everything was working simple and well.

The Building Process: How we put the prototype together

Software development life cycle, agile methodology

We decided we needed a guide of some sort to point where we could take action to start this project. We needed a project management framework. I researched the following technologies: Agile, Scrum, Waterfall and Kanban. I found that agile technology suited our purpose the best. I was already familiar with Agile because we had some lectures about it in our first year in college.

Systems Design and Analysis: Wireframes and prototyping

Our development process for this project includes:

- Gathering information related to car park system
- Planning the process for design
- Gather sample examples
- Coding
- Debugging code
- Maintenance and updating

SUBMIT First the user has to select the Option ‘Register’.

Then the user would get to the page to sign in with an email and password.

The user then selects the spot they want to book.

The user can select the entry date and the exit date. Then the user can submit their personal information

Then they press to submit the relevant information and their selection will be displayed on the screen.

To cancel a booking, in case the user is not happy with the booking choice on the booking table, the user can select one or multiple options and delete them from the table.

System Requirements

For the application to run on a computer, a device is expected to meet these minimum requirements.We have two categories,hardware and software

Hardware Requirements

Processor speed

- 1.6GHZ Note: Most PCs have this processor speed.

Memory of user PC

- 500MB+ RAM

Bandwidth(network connection)

- 15Mbps. Note: A simple internet connection will do.

Software Requirements

Operating System

- Window 7,Windows
- 8.1,Linux,MacOs ,Windows
- 8, Windows xp
- All of these are widely available

Database Management System

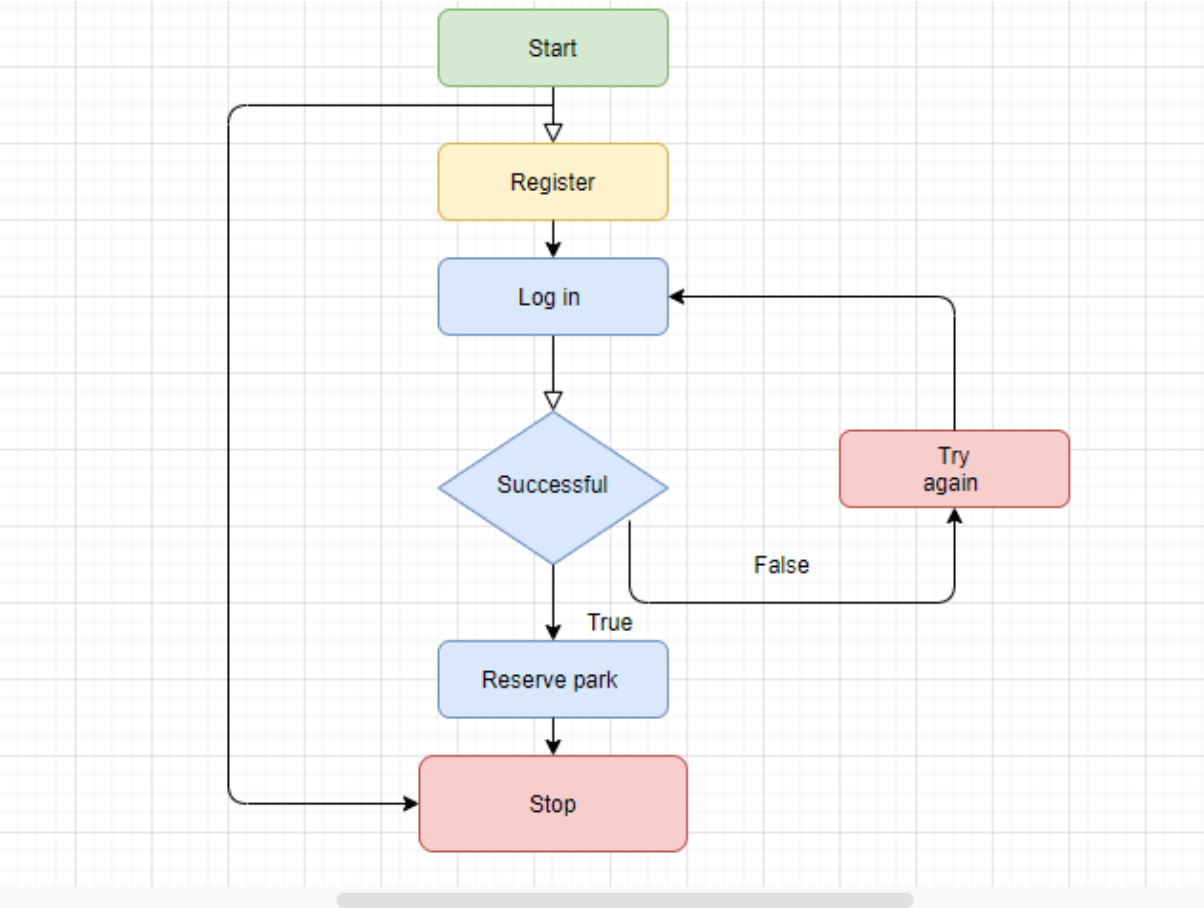
- MySql. This is free and easy to use

Browser

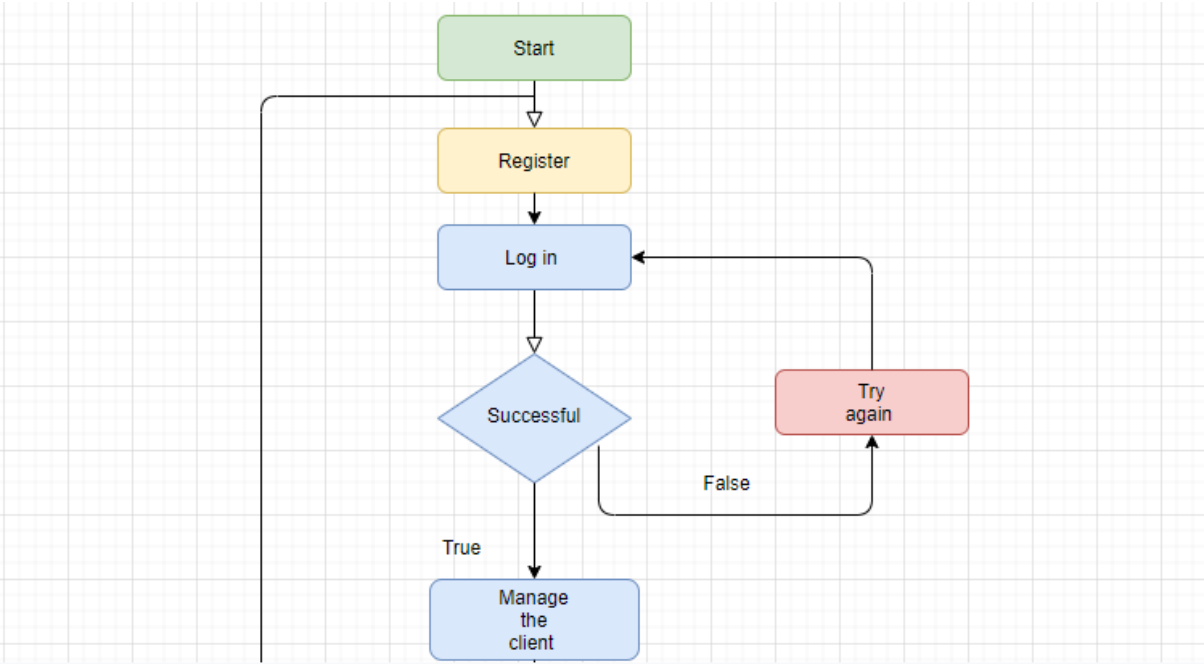
- Google Chrome
- Mozilla Firefox

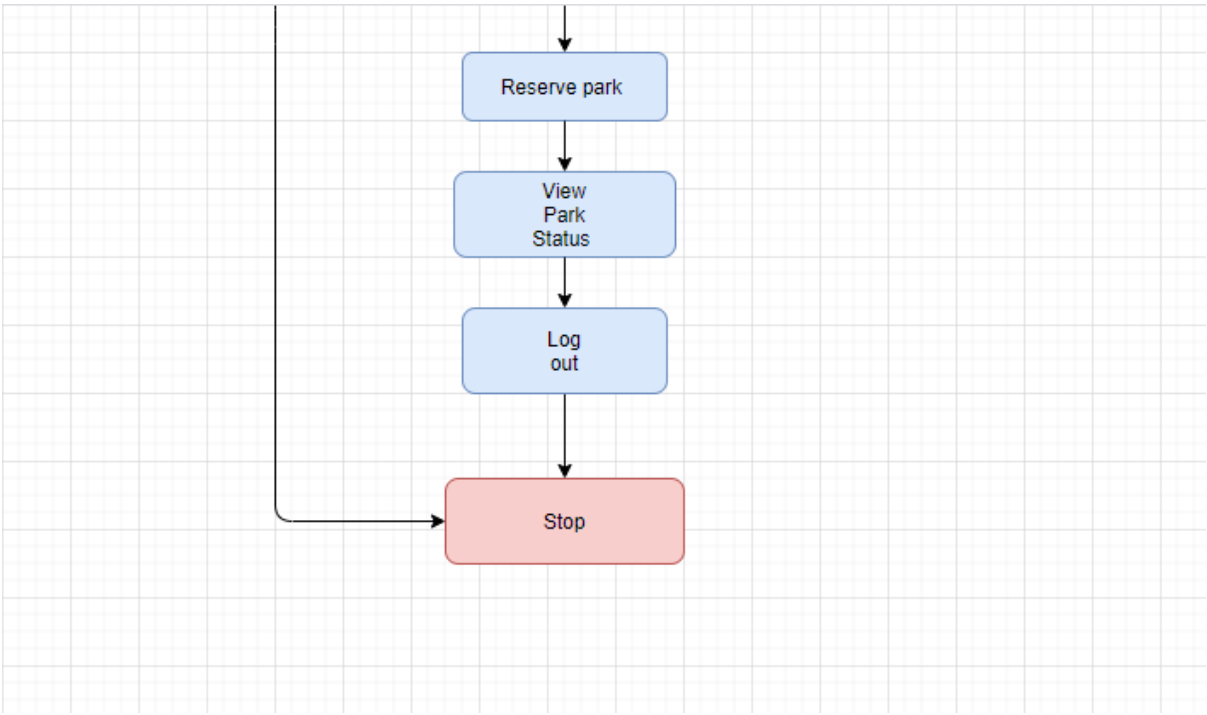
Flow chart to demonstrate how the system works

System Flow Chart: User side

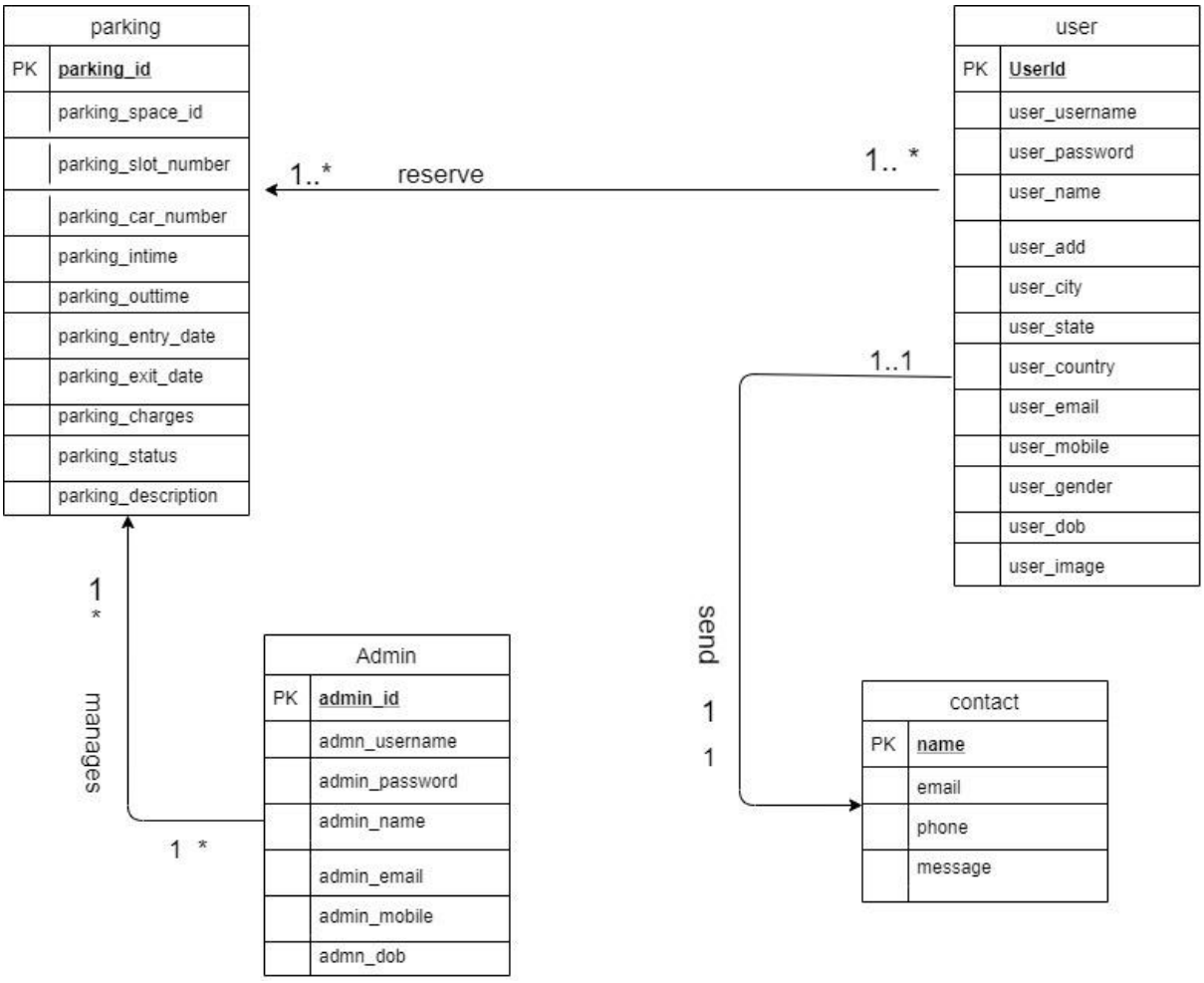


System Flow Chart: User side.





Entity Relationship Diagram of the system



Database Schemas

Below are the tables contained in the Online Parking Reservation system database

Table 1: Describes the Parking Table

Colum Name	Data Type	Description	Size	Allow Null Value
Parking_id	Int	Primary Key For The Table	11	No
Parking_space_id	Varchar	Name Of The Park Space	Default	No
Parking_slot_number	Varchar	Parking space number	Default	No
Parking_car_number	Varchar	Car number	Default	No
Parking_intime	Varchar	Time the client arrive in parking spot	Default	No
Parking_outtime	Varchar	Time the client leaves the parking spot	Default	Yes
Parking_entry_date	Varchar	Date the client arrive in parking spot	Default	No
Parking_exit_date	Varchar	Date the client arrive in parking spot	Default	Yes
Parking_charges	Varchar	Indicate the amount paid	Default	Yes
Parking_status	Varchar	Shows if the spot is available or not available	Default	Yes
Parking_description	Varchar	Indicates the location of the spot	Default	Yes

Table 2: Describes the User Table

Colum Name	Data Type	Description	Size	Allow Null Value
user_id	int	Primary key for the table	11	No
user_username	text	Users profile name	Default	No
user_password	varchar	The users password	Default	No
user_name	varchar	First and last name of the user	Default	No
user_add	varchar	Users address	Default	No
user_city	varchar	Users city of residence	Default	No
user_state	varchar	Users state	Default	No
user_country	varchar	Users country	Default	No
user_email	varchar	Users email	Default	No
user_mobile	varchar	Users mobile number	Default	No
user_gender	varchar	Users gender	Default	No
user_dob	date	Users date of birth	Default	
user_image	varchar	Profile picture of the user	Default	

Table 3: Describes the Admin Table

Colum Name	Data Type	Description	Size	Allow Null Value
Admin_id	int	Primary key for the table	11	No
Admin_username	text	Users profile name	Default	No
Admin_password	varchar	The users password	Default	No
Admin_name	varchar	First and last name of the user	Default	No
Admin_email	varchar	Users email	Default	No
Admin_mobile	varchar	Users mobile number	Default	No
Admin_dob	date	Users date of birth	Default	

Table 4: Describes the Contact Table

Colum Name	Data Type	Description	Size	Allow Null Value
name	text	The name of the client or user who wants to send the feedback	11	No
email	text	The email of the client or user who wants to send the feedback	Default	No
phone	text	The phone number of the client or user who wants to send the feedback	Default	No
message	text	The feedback which needs to be sent	Default	No

Functionalities

Admin – Account

Can add parking space for the users

Can see the total capacity of the parking space

Can see the booked parking spaces which are booked by the users

Can assign the user a parking spot

Can add users to the system

Capable to delete the system users from the system database

Can search the car information by the car number

Client – Account

Can Book the parking space

Can delete the booked space

Website requirements

The car system must save information about entering a new car.

Information about the parking system is intended to help internal staff retain parking information and find it using a variety of inquiries.

Should monitor the parking space.

Should have ability to Edit/ update and delete the data record.

It should also search car by there car number provided by users.

Should also require a security system for data security of the website database.

What the system is expected to do

Our website is a platform where any person with a driving license can find a place to park.

All a customer needs to create a login is to enter their email and create a password. They then have an account that they can reuse whenever they need. Once the customer has created a login, this collected information is used to access the next page of the website where the booking will take place.

The second page of the website collects more data. The customer will fill out a simple car park form in order to choose a parking spot. The website uses this information, including entry and exit times to complete the booking. Values are added for Day, Month, Year, Hour and minutes.

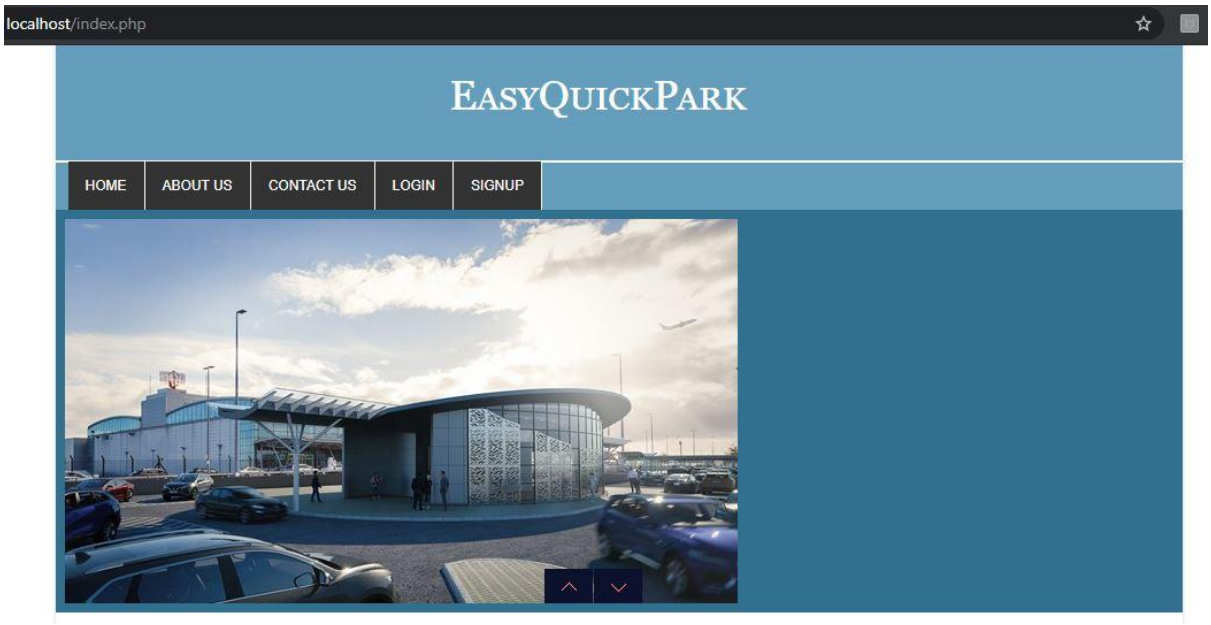
The website is required to collect personal data for the final procedure such as First Name, Last name, email, and phone number.

On the same page there will be a submit bottom.

The website will have a service from Amazon cloud. Once the user submits their information, the chosen spot will be sent to them by email or phone. The message will include a unique parking number as well as the relevant date and time information.

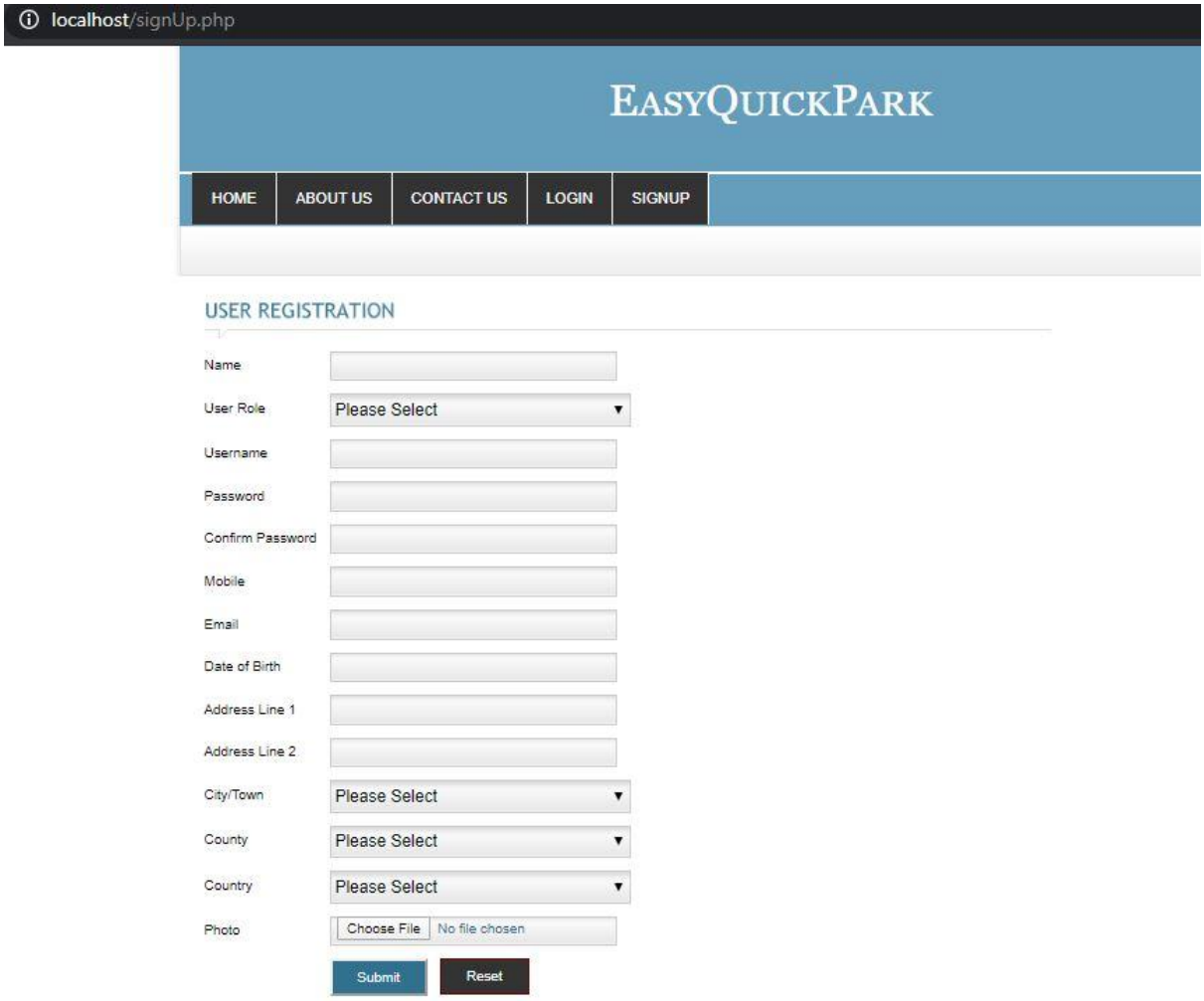
Finished Prototype

Index.php page



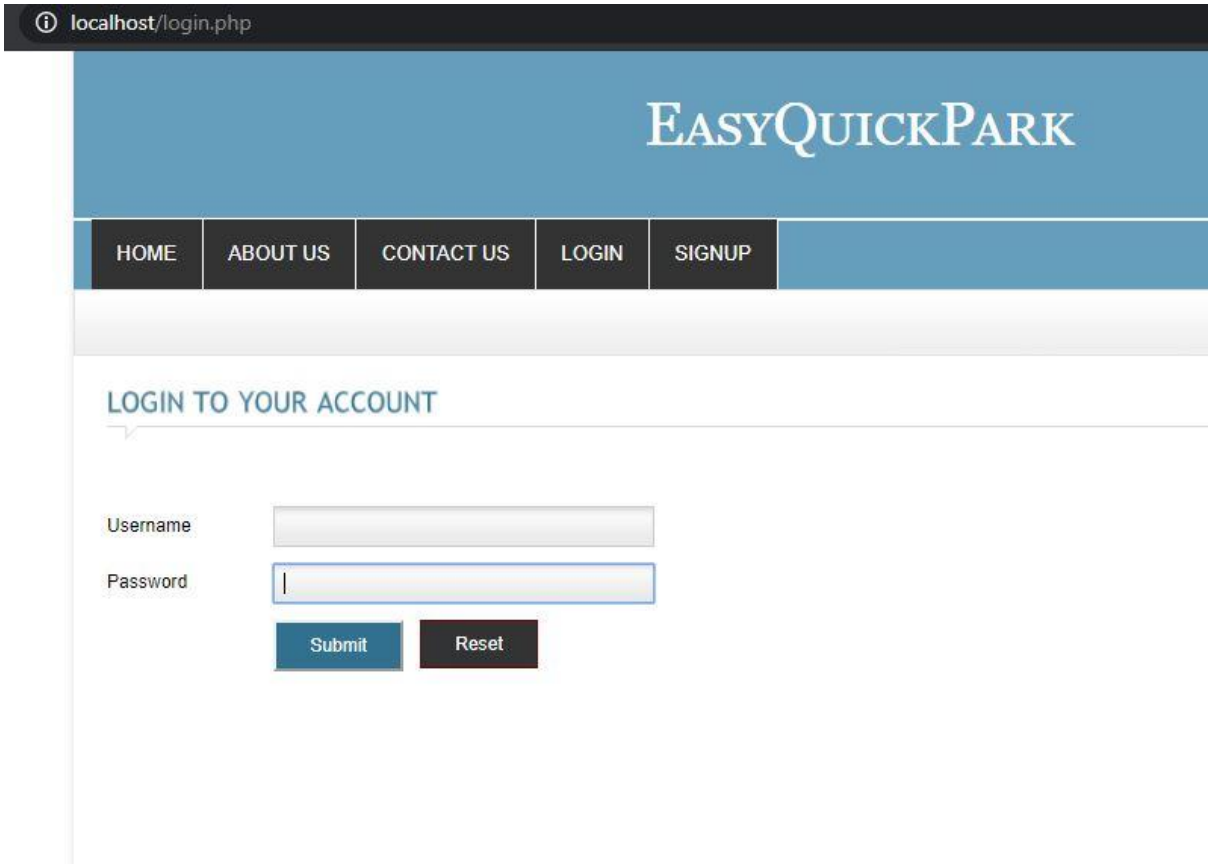
This is the first thing the user sees when visiting our web app.This page also includes the offers we provide, that goes hand in hand with the parking.

Register.php page



The picture above shows the registration page where new users are able to create their accounts. Users should fill all the required information before they can be approved for registration.

Login.php page

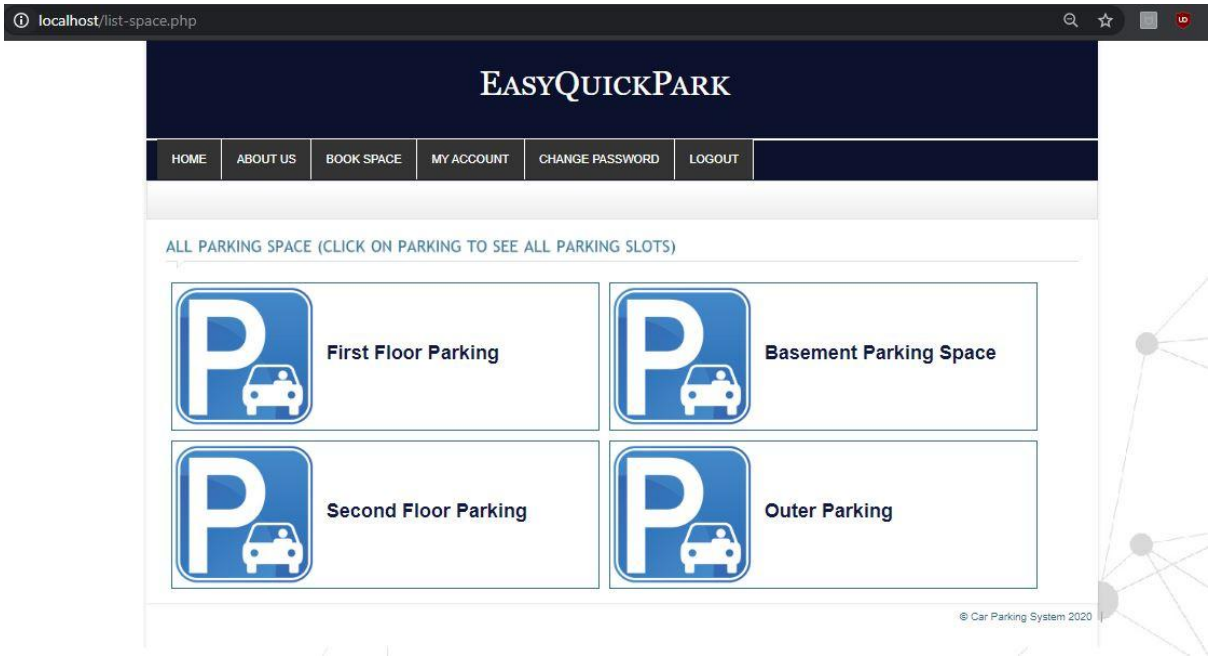


Code snippet:verification of login details

```
####Function check user######
function check_login()
{
        $user_user=$_REQUEST[user_user];
        $user_password=$_REQUEST[user_password];
        $SQL="SELECT * FROM user WHERE user_username = '$user_user' AND user_password =
'$user_password'";
        $rs = mysql_query($SQL) or die(mysql_error());
        if(mysql_num_rows($rs))
        {
                $_SESSION[login]=1;
                $_SESSION['user_details'] = mysql_fetch_assoc($rs);
                if($_SESSION['user_details']['user_level_id'] == 2) {
```

```
                $SQL="SELECT * FROM branch_type WHERE bt_hod =
'".$_SESSION['user_details']['user_id']."'";
                $rs = mysql_query($SQL) or die(mysql_error());
                $_SESSION['branch_details'] = mysql_fetch_assoc($rs);
            }
            header("Location:../index.php");
        }
        else
        {
            header("Location:../login.php?msg=Invalid User and Password.");
        }
}
```

List-space.php page



Here the user chooses where they want to park their car when booking a spot. They can choose if they want to park their car on the first floor, second floor, basement or the outer parking space.

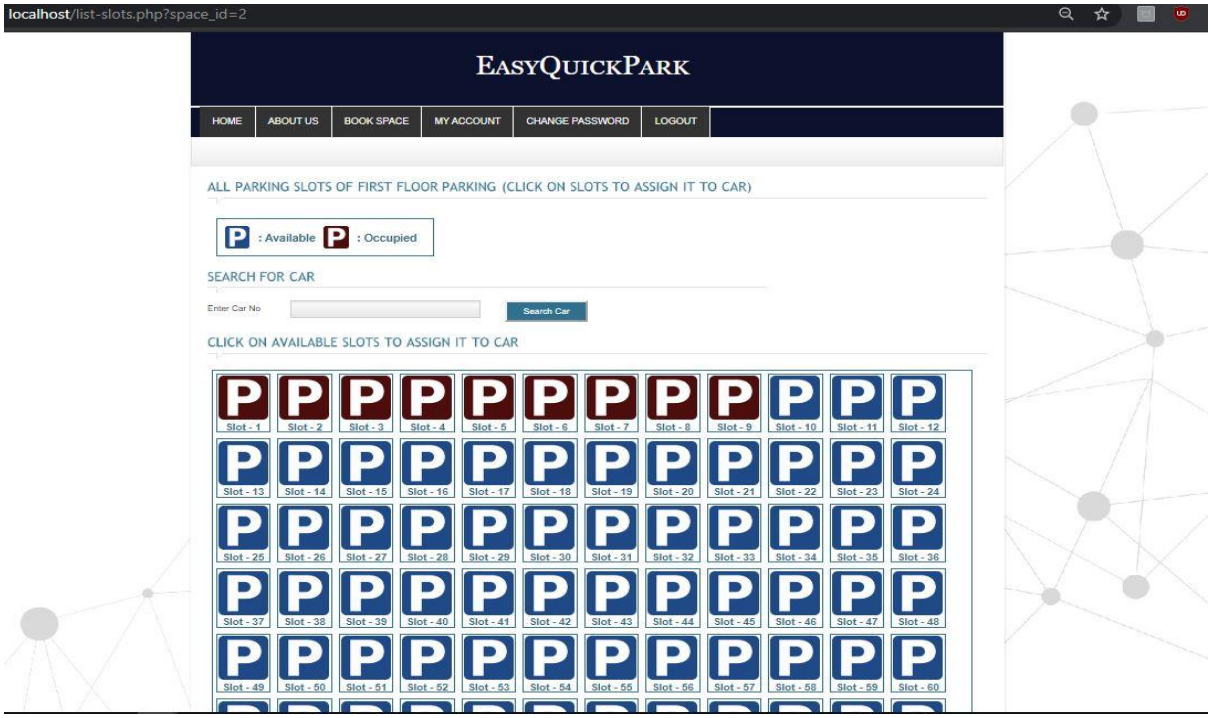
Code snippet :

```
function save_parking()
    {
        $R=$_REQUEST;
        if($R[parking_id])
        {
```

```
                #when user update parking
                $statement = "UPDATE `parking` SET";
                $cond = "WHERE `parking_id` = '$R[parking_id]'";
                $msg = "Parking deallocated Successfully.";

        }
        else
        {
                #insert into the parking table in database
                $statement = "INSERT INTO `parking` SET";
                $cond = "";
                #$msg="Parking Alloted Successfully.";
                 $msg= '<a href="payment.php">parking alloted successfully, please click here for payment process </a>';

        }

        $SQL=  $statement."
                `parking_space_id` = '$R[parking_space_id]',
                `parking_slot_number` = '$R[parking_slot_number]',
                `parking_car_no` = '$R[parking_car_no]',
                `parking_intime` = '$R[parking_intime]',
                `parking_outtime` = '$R[parking_outtime]',
                `parking_entry_date` = '$R[parking_entry_date]',
                `parking_status` = '$R[parking_status]',
                `parking_exit_date` = '$R[parking_exit_date]',

                `parking_email` = '$R[parking_email]',
                `parking_phonenumber` = '$R[parking_phonenumber]',

                `parking_charges` = '$R[parking_charges]',
```

```
                `parking_description` = '$R[parking_description]'".
                 $cond;
        $rs = mysql_query($SQL) or die(mysql_error());
        header("Location:../list-slots.php?msg=$msg&space_id=$R[parking_space_id]");


    }
```

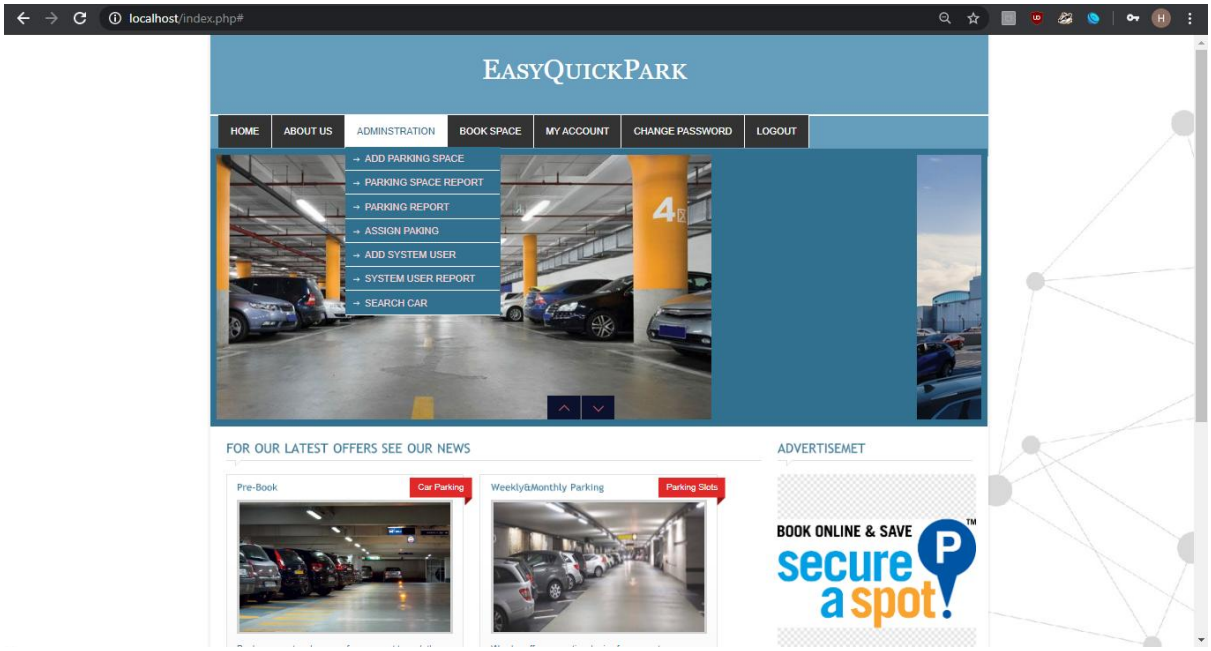
List-slots.php page



This page is shown to the user after they have chosen where to park their car in the (list-space.php) page .The slots that are red if they are occupied slots , while the blue slots are free slots which the user can select.

NB: The red slots are not clickable only the blue are.

Admin.php page



The admin page is the one responsible for addition of slots, deletion of users and maintaining the system at large.

Code snippet: deletion of a user by the admin

```
function delete_user()
{
        $SQL="SELECT * FROM user WHERE user_id = $_REQUEST[user_id]";
        $rs=mysql_query($SQL);
        $data=mysql_fetch_assoc($rs);

        /////////Delete the record//////////
        $SQL="DELETE FROM user WHERE user_id = $_REQUEST[user_id]";
        mysql_query($SQL) or die(mysql_error());

        //////////Delete the image/////////
        if($data[user_image])
        {
                unlink("../uploads/".$data[user_image]);
        }
        header("Location:../user-report.php?msg=Deleted Successfully.");
```

}

?>

This line show the syntax of the where the user chooses where he can park their car when booking for a spot, they can choose if they want to park their car in the first floor, second floor, basement or the outer parking space

Code snippet:

```
function save_space()
    {
        $R=$_REQUEST;

        if($R[space_id])
        {
            $statement = "UPDATE `space` SET";
            $cond = "WHERE `space_id` =
'$R[space_id]'";
            $msg = "Data Updated Successfully.";
        }
        else
        {
            $statement = "INSERT INTO `space` SET";
            $cond = "";
            $msg="Data saved successfully.";
        }
        $SQL=  $statement."
                `space_title` = '$R[space_title]',
                `space_total_parkings` =
'$R[space_total_parkings]',
                `space_description` =
'$R[space_description]'".
                 $cond;
        $rs = mysql_query($SQL) or die(mysql_error());
        header("Location:../space-report.php?
msg=$msg");
    }
```

Testing and Evaluation

Trouble shutting difficulty

At the very beginning we had an idea to include a Google Maps Platform. For our first project, according to our needs back then, we spoke to our supervisor about adding a functionality where the user can search for directions on our webpage. Then this search engine could be a great tool andwould bring our project to a better level. Unfortunately, our first proposal was not consideredcreative enough.We had to come with another idea in a short amount time for a new project.Despite the issues, we decided to keep the idea for API. We came up with our current final project: a car parking system but as we decided to find another API suitable for this project, I researched and I find on amazon cloud a system where the user can get messages from the server.

Conclusion (did you achieve your goals/objectives)

Appendix: Division of Labour and Tasks among members of the team:

Bibliography

https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/#cloud-computing-models

Bilodeau, V.P. Intelligent Parking Technology Adoption. Ph.D. Thesis, University of Southern Queensland: Queensland, Australia, 2010

https://www.cso.ie/en/releasesandpublications/ep/p-cp6ci/p6cii/p6mtw/. Accessed 14/04/2020

https://www.daft.ie/dublin/parking-spaces/. Accessed 10/05/2020

https://www.guru99.com/web-application-testing.html. Accessed 21/04/2020

Khanna A. and Anand R. (2016) *IoT based Smart Parking System.* Available from: https://www.researchgate.net/publication/303842610 . Accessed 07/03/2020

Parky (2017) "8 Surprising facts you didn't know about parking!" Available online: https://www.b4parking.com/blog/8-surprising-facts-you-didnt-know-about-parking/Accessed 7th May 2020

Power, J. (2019) "Councillors approve Dublin City Centre Parking Fee increases" Irish Times Availabe online: https://www.irishtimes.com/news/environment/councillors-approve-dublin-city-centre-parking-fee-increases-1.3782162. Accessed 12/04/2020

https://xbsoftware.com/blog/website-development-process-full-guide/

Zhanlin Ji, Ivan Ganchev 1, MáirtínO'Droma, Li Zhao and Xueji Zhang (2014) *A Cloud-Based Car Parking Middleware for IoT-Based Smart Cities: Design and Implementation.* Sensors. Available from: https://www.mdpi.com/1424-8220/14/12/22372. Accessed 08/03/2020