ONLINE PARKING BOOKING

MINI PROJECT REPORT

By

BHAVEN KAKADE

SAMEER MAHAJAN 40

AISHWARYA MUCHANDI

VATSAL PANCHAL

Department of Computer Engineering

A.P.Shah Institute Of Technology

Thane

October 2018

CERTIFICATE

This is to certify that

Ms. Aishwarya . S. Muchandi , RollNo. 44 and Moodle

Id. 16102056 and Semester V has completed mini project entitled **"Online Parking Management"** in the subject **WEB DESIGN** as laid down by University of Mumbai in satisfactory manner within premises of institute during academic year 2018-2019.

Date: 12/09/18

_____ _____

Subject In-Charge/Guide Head of Department

Acknowledgement

It takes us immense pleasure to present before you project on **Online Parking Booking.**

We Would like to express gratitude to all people who have helped during the making of this project. We would also like to thank all the faculties and especially our project guide for giving us this wonderful opportunity for working on this topic.We would like to express our gratitude to A.P.Shah Institute Of Technology for providing us with the resources and opportunity for this unique project.

And lastly we thank our family and friends for all the love and support during the making of this project without which it would have been difficult to implement this project.

Abstract

This project is a smart parking booking system that provides institutes students as well as faculties an easy way of reserving a parking space online using this web service. It overcomes the problem of finding a parking space in universities/institute areas that unnecessary consumes time. Hence, this project offers a web application based reservation system where students can view various parking spaces and select nearby or specific area of their choice to view whether space is available or not.If the booking space is available, then student can book it for specific time slot. The booked space will be marked and will not be

available for anyone else for the specified time..Students can also view previous parking booking details using the web portal.

At present, sometimes the students have to search around campus for parking space for their vehicle but this project reduces this work of searching around campus. This traditional method of searching around for parking space wastes a lot of fuel and time. This project saves the time and money by smartly providing parking space.

Table of Contents

Title

1.Certificate

2.Acknowledgement

3.Abstract

Introduction

The online parking booking system is one of the latest services most municipal corporations , malls, Corporate offices are adopted. Using this system, users can book their parking remotely from virtually anywhere in the world. This is made possible through the use of electronic payment system. Customers can pay with their credit cards, debit cards, digital wallets, etc. So this project is completely designed to work in digital way which will enable customers to go online and book their parking.

Due to great increase in awareness of internet and technologies associated with it, several groups and organisations are opting for digital way to go on with their businesses. In this digital age it is very convenient to book parking slot online rather than going out on searching spree which will indeed save time and money.

In view of the rapid development of information technology in almost all fields of operation it has become important to look into development of online booking system. Therefore , online parking booking system will help customers and management to:

1.Book parking slots from remote location.

2.Create a platform for online booking of parking slots.

3.Keep record on the slots booked by faculties and students.

Once a slot is booked from the web page, the record is placed in the database and then retrieved in real time by a desktop application.With this system all the information about the booked slots and the person who has booked is displayed in a precise manner

PROPOSED SYSTEM

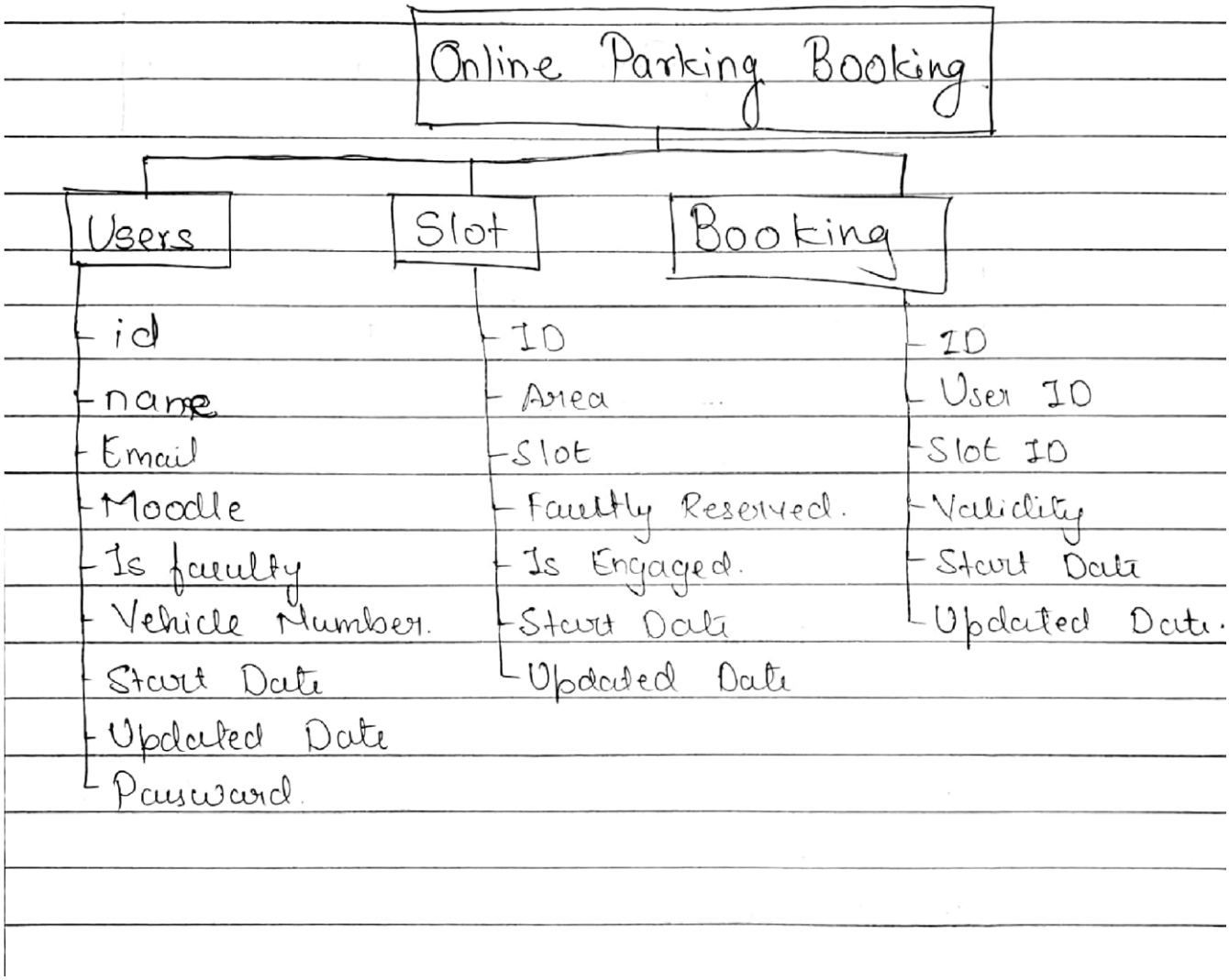
This project lays out a framework for a new system to be developed and brought to the market for maximum use and to create an avenue through the web

where users can log on to our server and make a selection of whatever goods or food they like and subsequently pay via the internet.

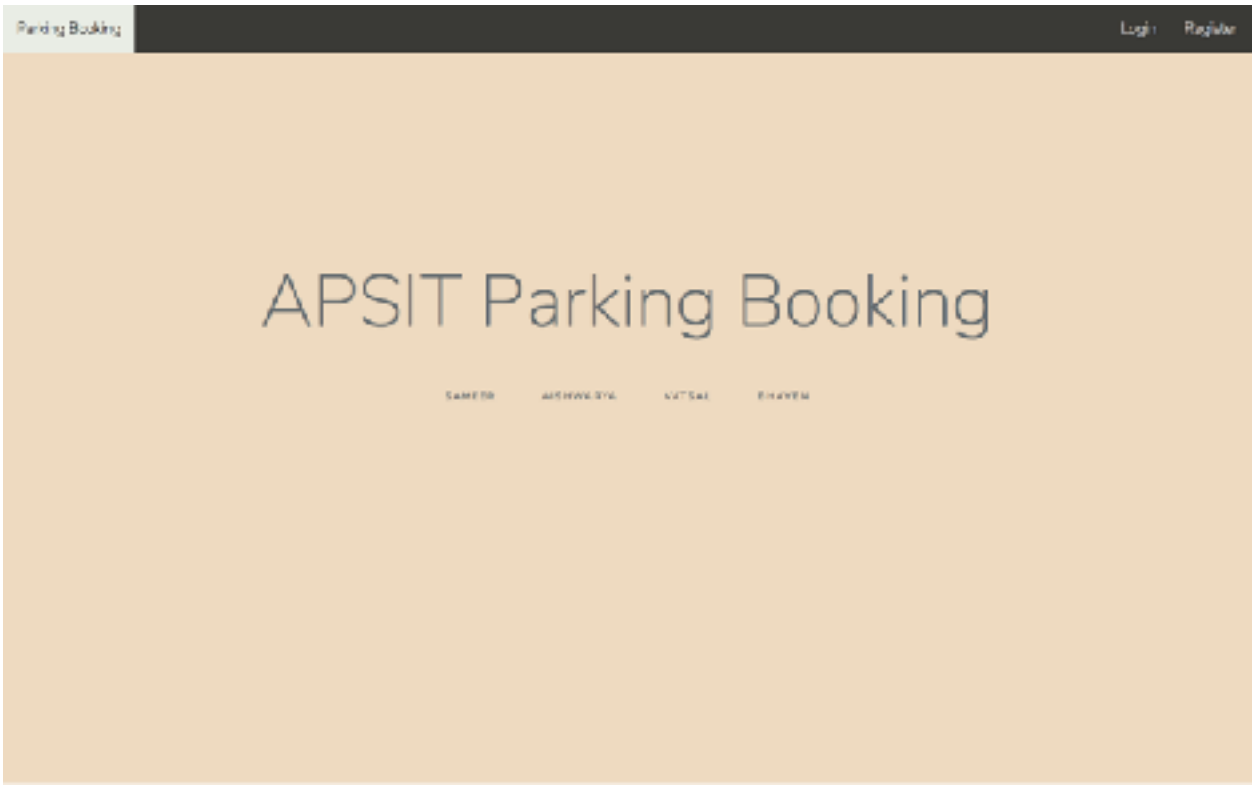
This online project enables the end users to register online, select the slot from the map displayed online, by just selecting the slot that the user want to book. The user will be provided a unique ID and a password to log in to their profile.

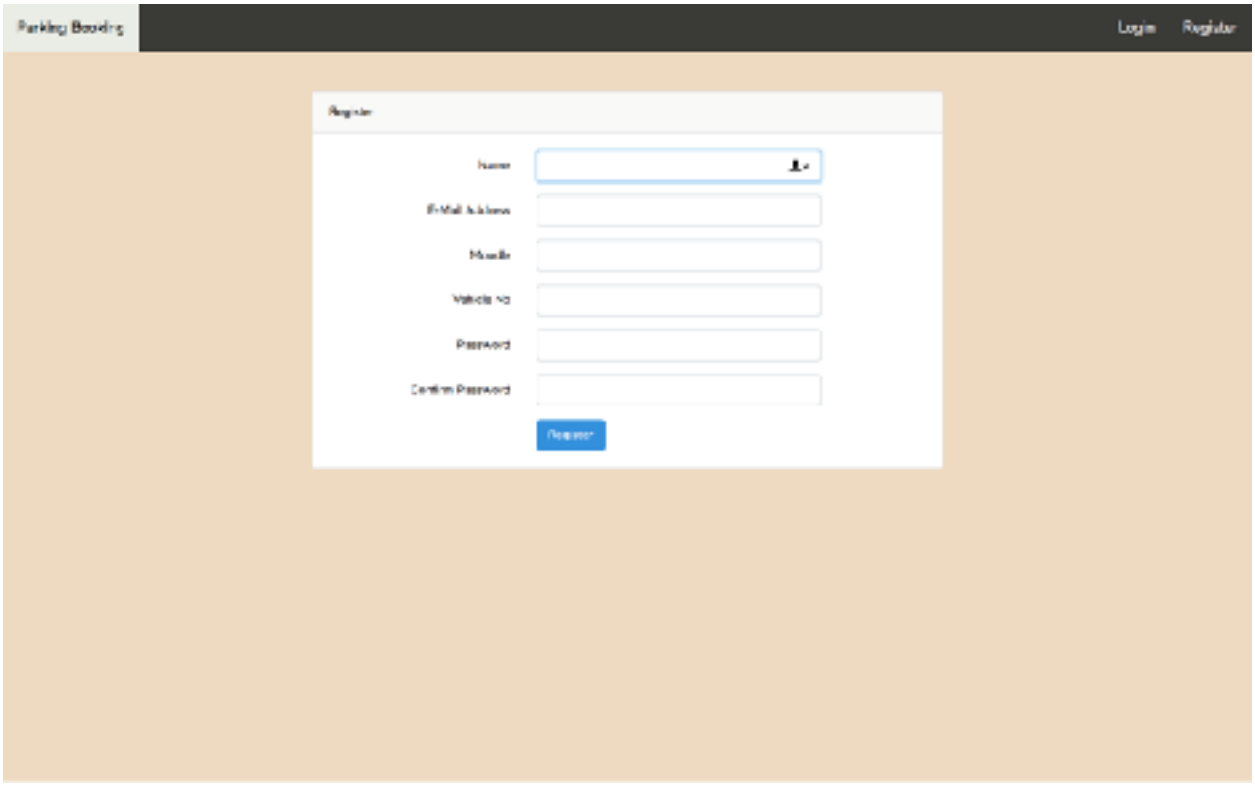
This will provide a user friendly environment between the customer and the management to book the slots online thus increasing the efficiency of the parking system. There will also be an option to select from either a monthly pass or daily booking. It will also help in easily locating the vehicle in any case by using the real time parking system by tracking the slot ID provided to the user.

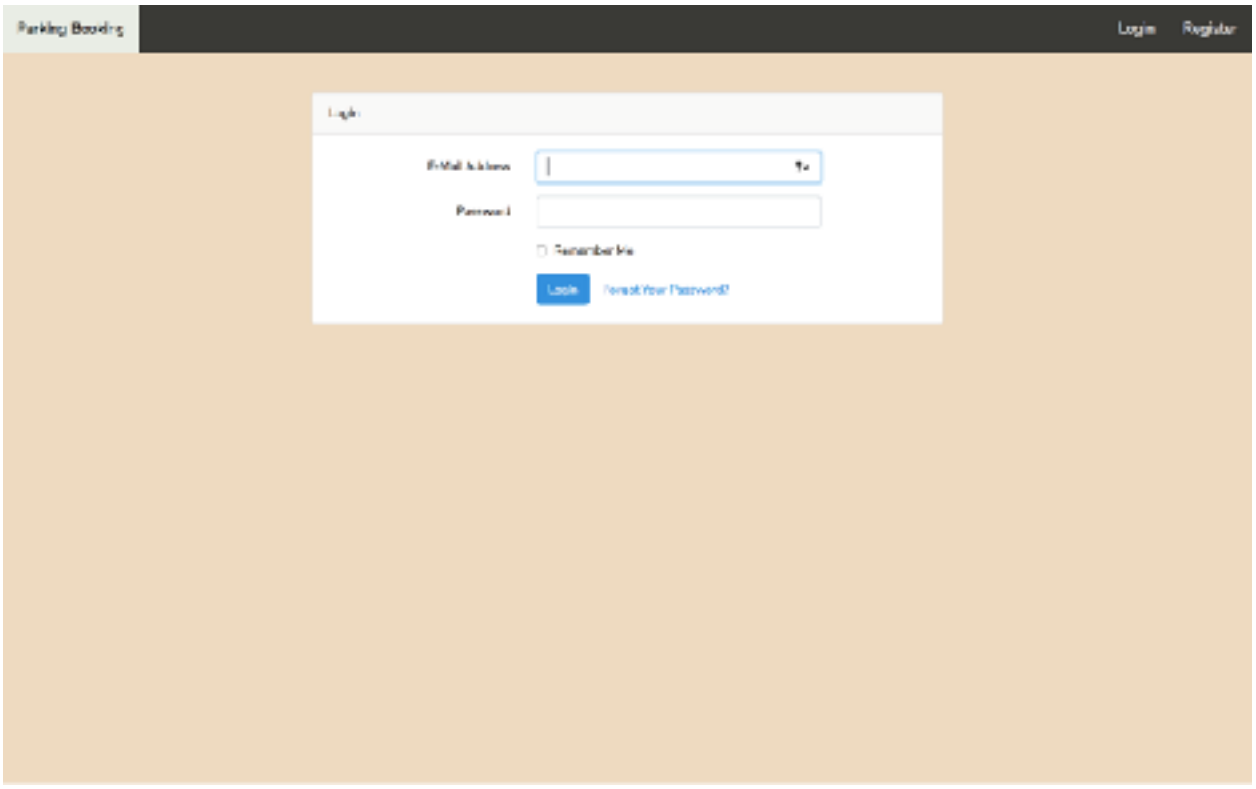
FLOW OF SYSTEM



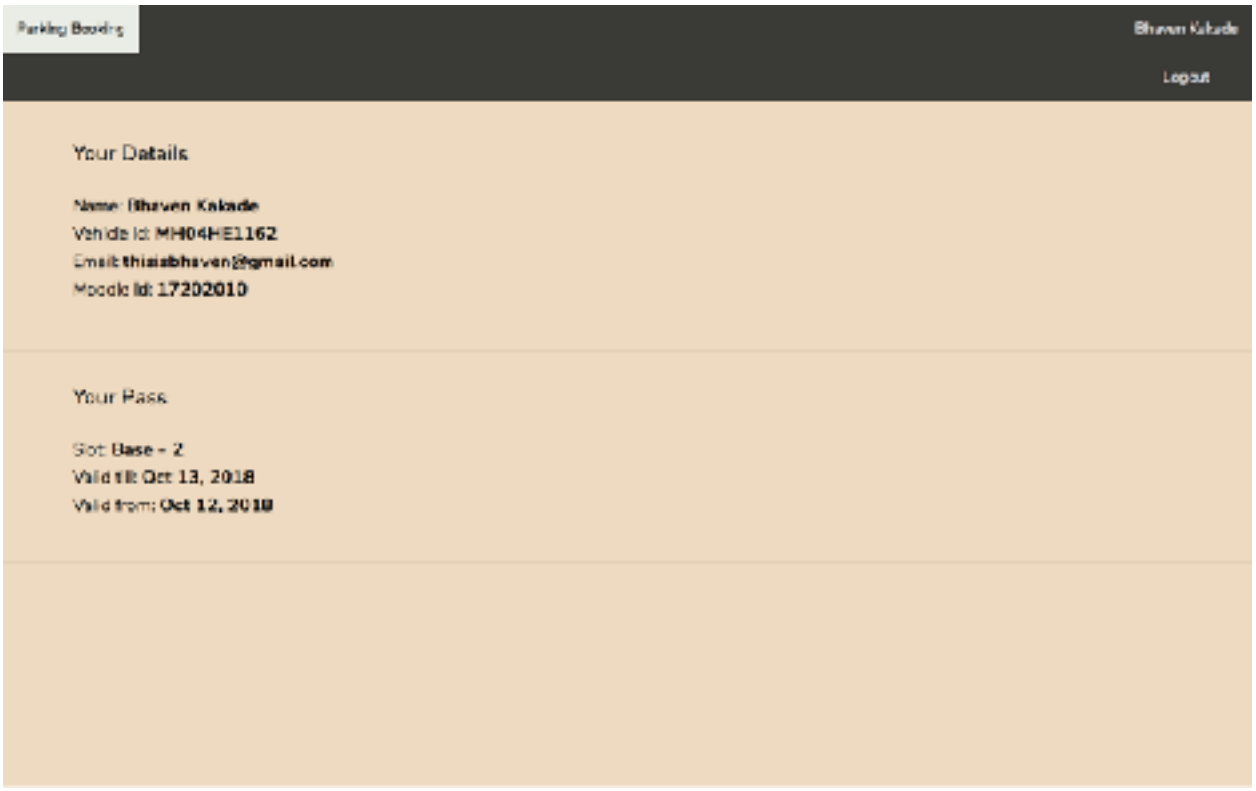
GRAPHICAL USER INTERFACE

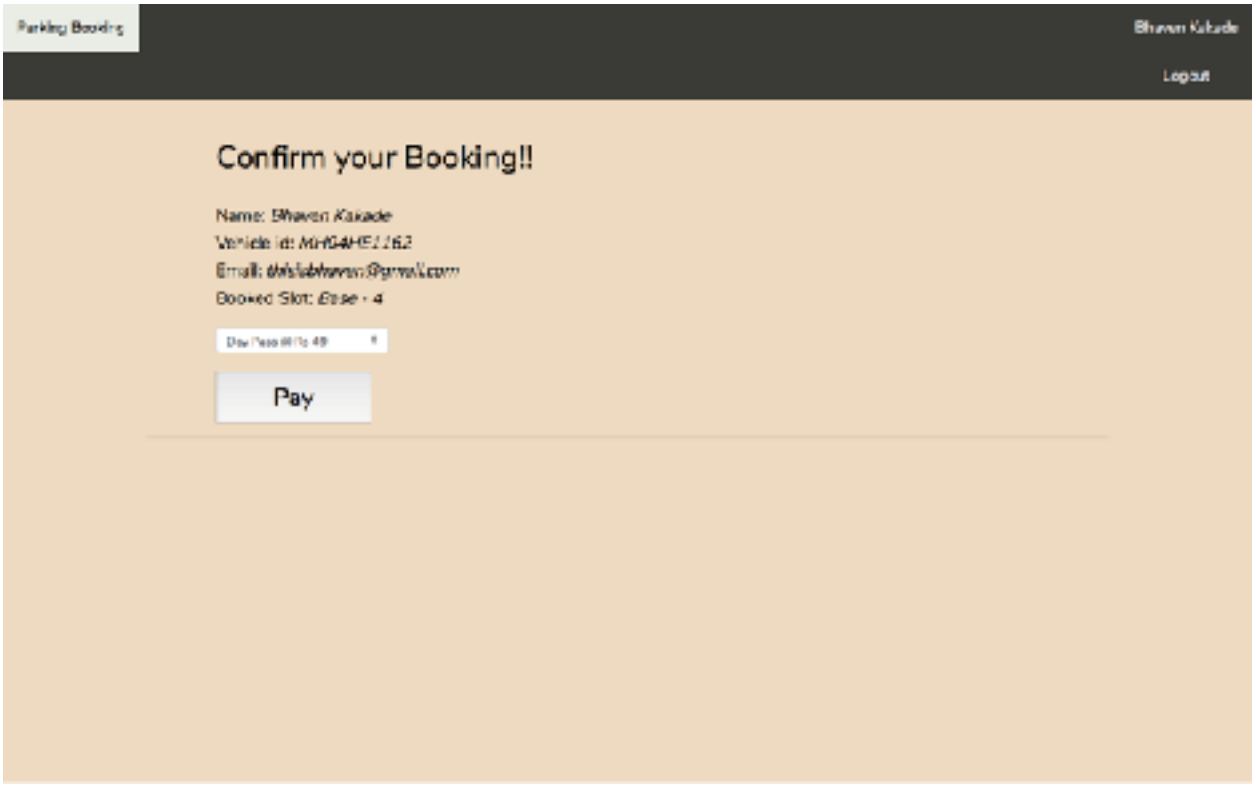

Parking Booking
Login
Register
APSIT Parking Booking


Parking Booking
Login
Register

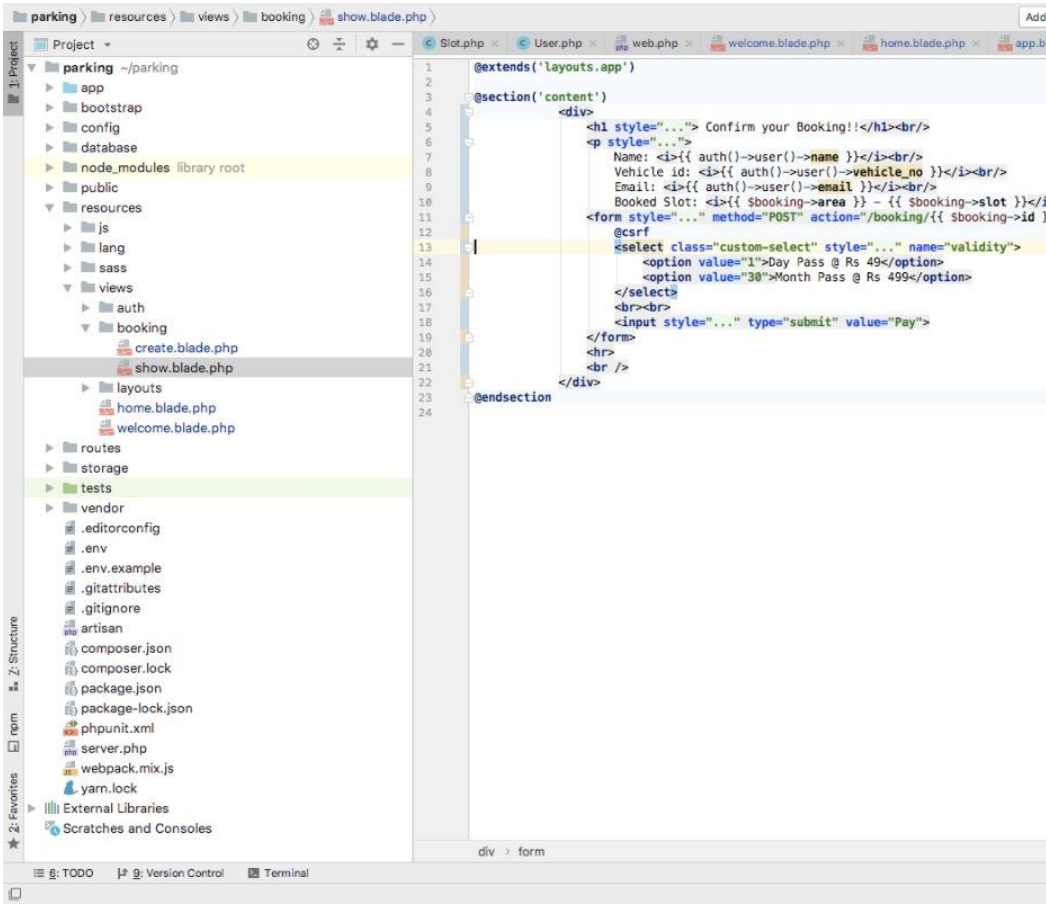

Parking Booking
Login
Register

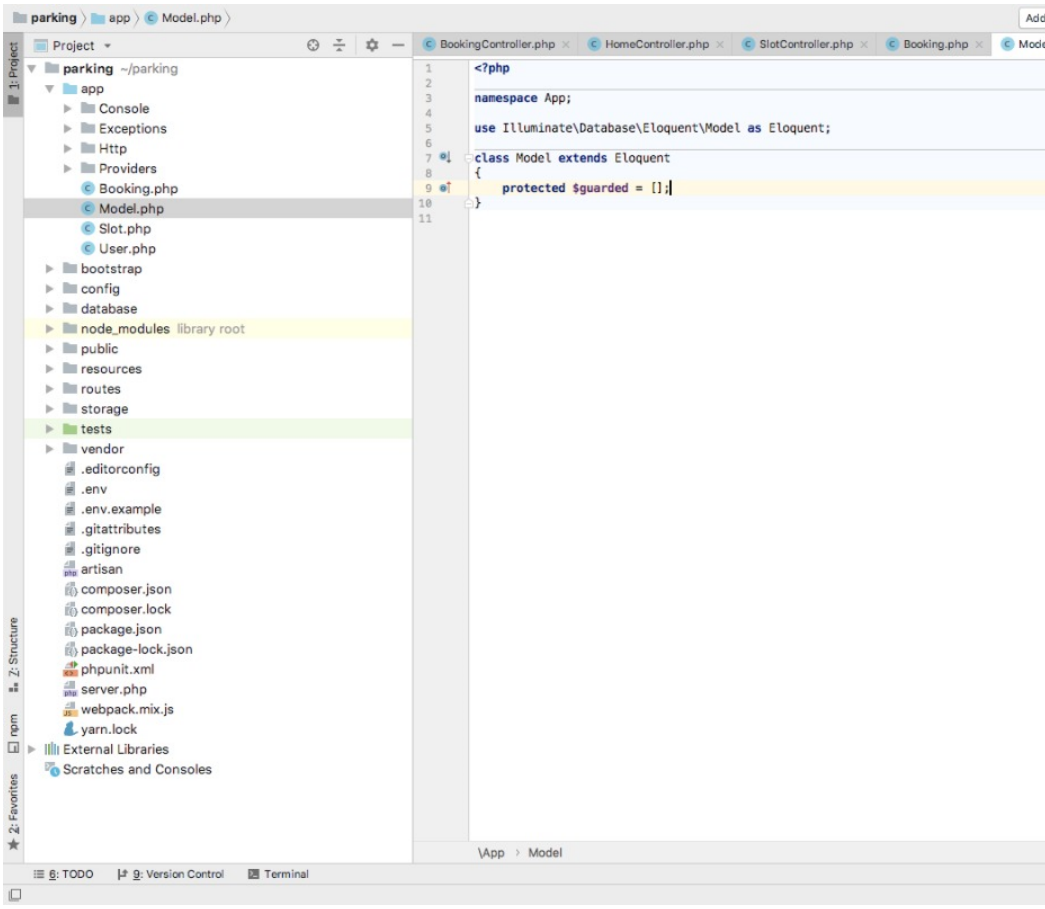
Parking Booking
Bhavan Kakade
Logout
Basement Parking
Front Gate Parking

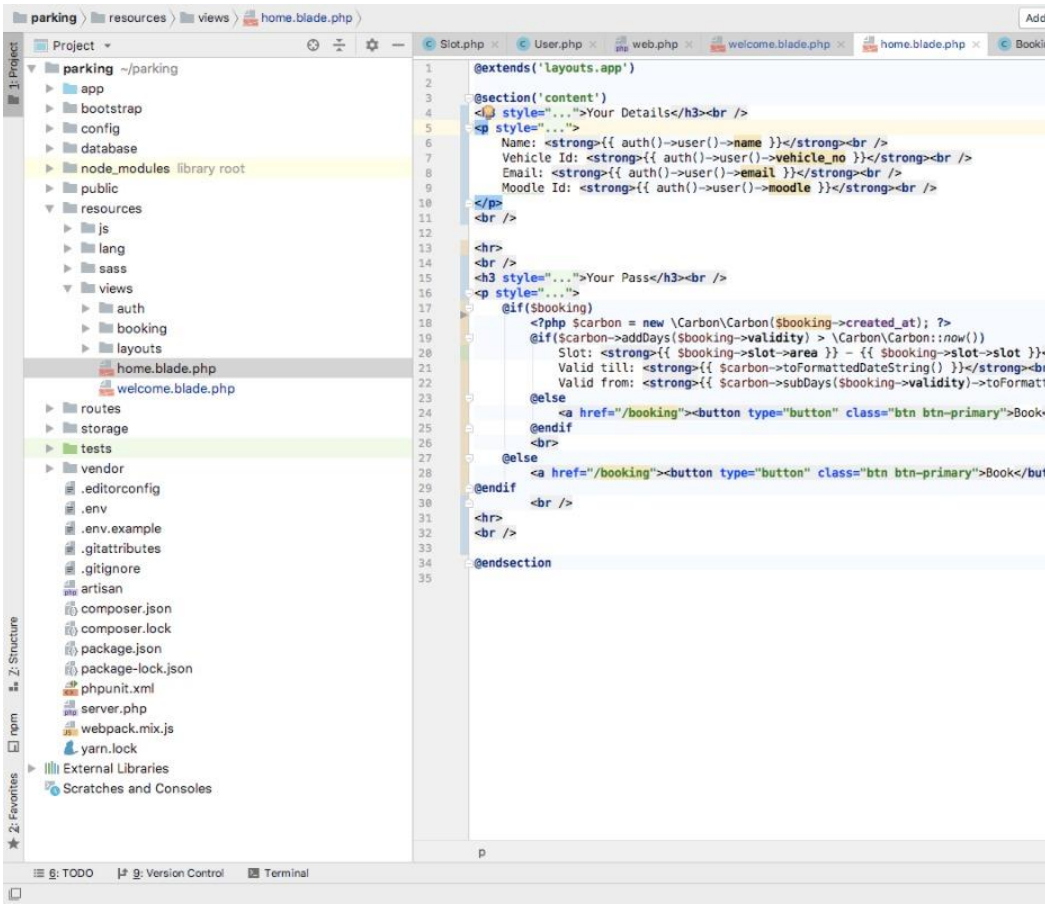

Parking Booking
Bhavan Kakade
Logout
Your Details
Name: Bhavan Kakade
Vehicle Id: MH04HE1162
Email: thisisbhavan@gmail.com
Moodle Id: 17202010
Your Pass
Slot: Base - 2
Valid till: Oct 13, 2018
Valid from: Oct 12, 2018


Parking Booking
Bhavan Kakade
Logout
Confirm your Booking!!
Name: Bhavan Kakade
Vehicle Id: MH04HE1162
Email: thisisbhavan@gmail.com
Booked Slot: Base - 4
Pay

Code Snippets






parking
resources
views
home.blade.php
Project
parking ~/parking
app
bootstrap
config
database
node_modules library root
public
resources
js
lang
sass
views
auth
booking
layouts
home.blade.php
welcome.blade.php
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
web.php
welcome.blade.php
home.blade.php
@extends('layouts.app')
@section('content')
<h3 style="...">Your Details</h3>

Name: {{ auth()->user()->name }}
Vehicle Id: {{ auth()->user()->vehicle_no }}
Email: {{ auth()->user()->email }}
Moodle Id: {{ auth()->user()->moodle }}

<hr>

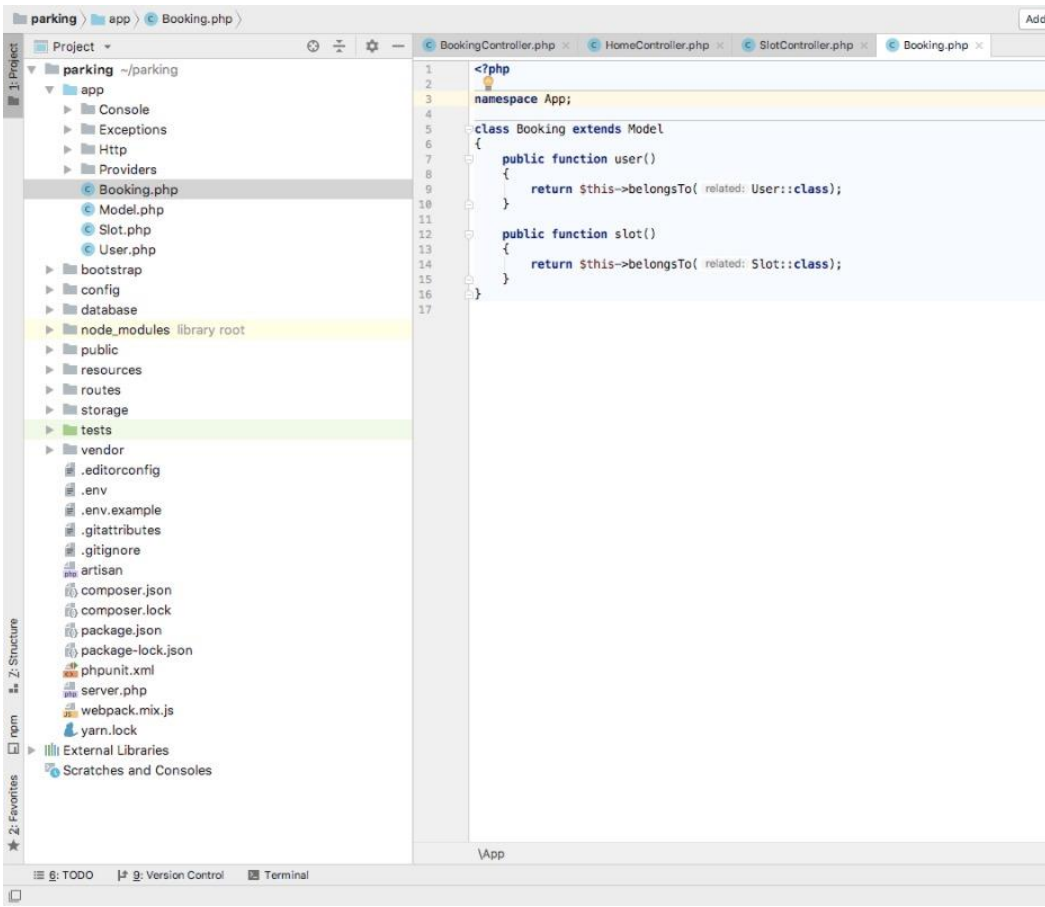
<h3 style="...">Your Pass</h3>

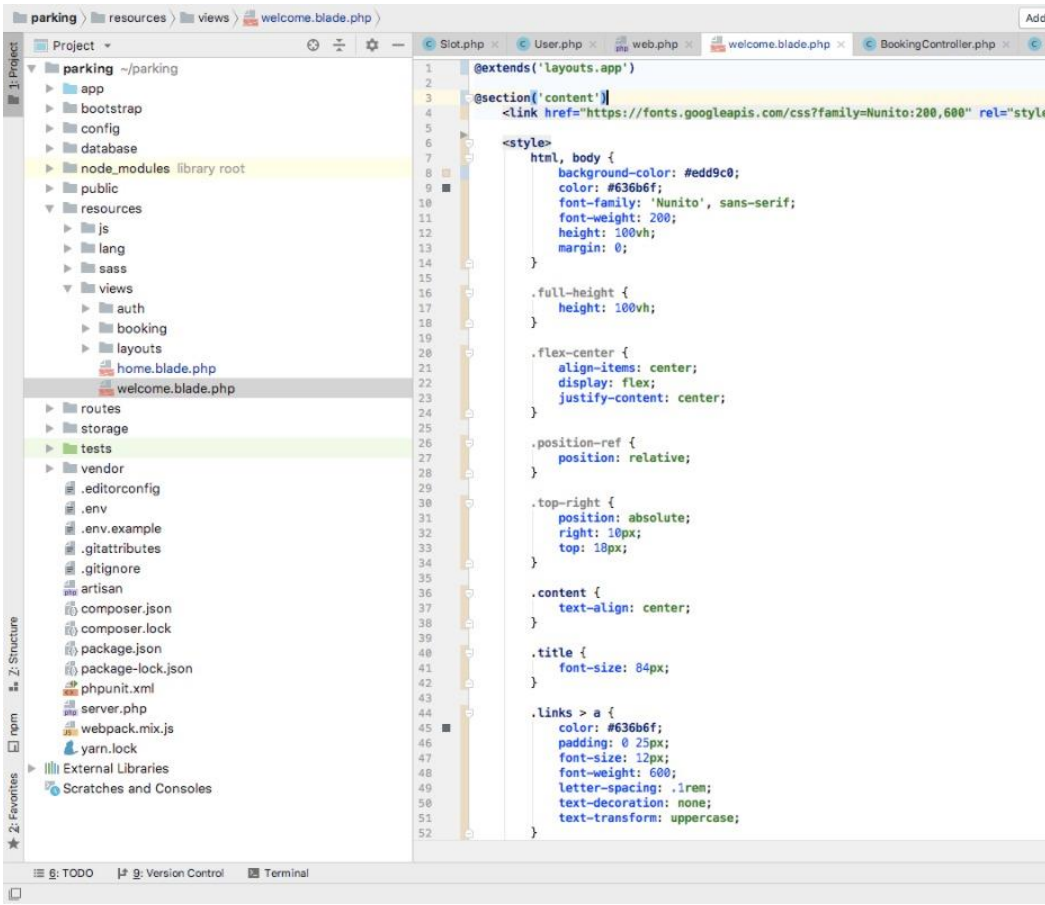
@if($booking)
<?php $carbon = new \Carbon\Carbon($booking->created_at); ?>
@if($carbon->addDays($booking->validity) > \Carbon\Carbon::now())
Slot: {{ $booking->slot->area }} - {{ $booking->slot->slot }}
Valid till: {{ $carbon->toFormattedDateString() }}<br
Valid from: {{ $carbon->subDays($booking->validity)->toFormatt
@else
<a href="/booking"><button type="button" class="btn btn-primary">Book
@endif

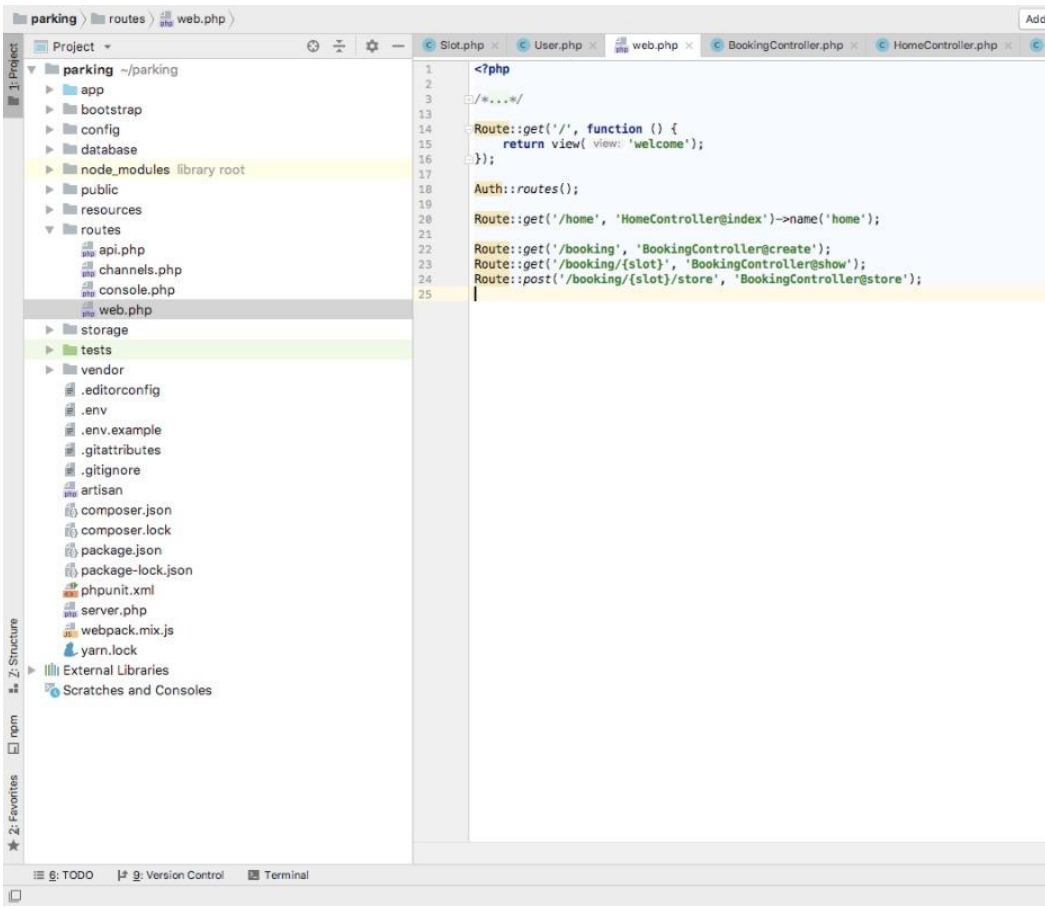
@else
<a href="/booking"><button type="button" class="btn btn-primary">Book</but
@endif

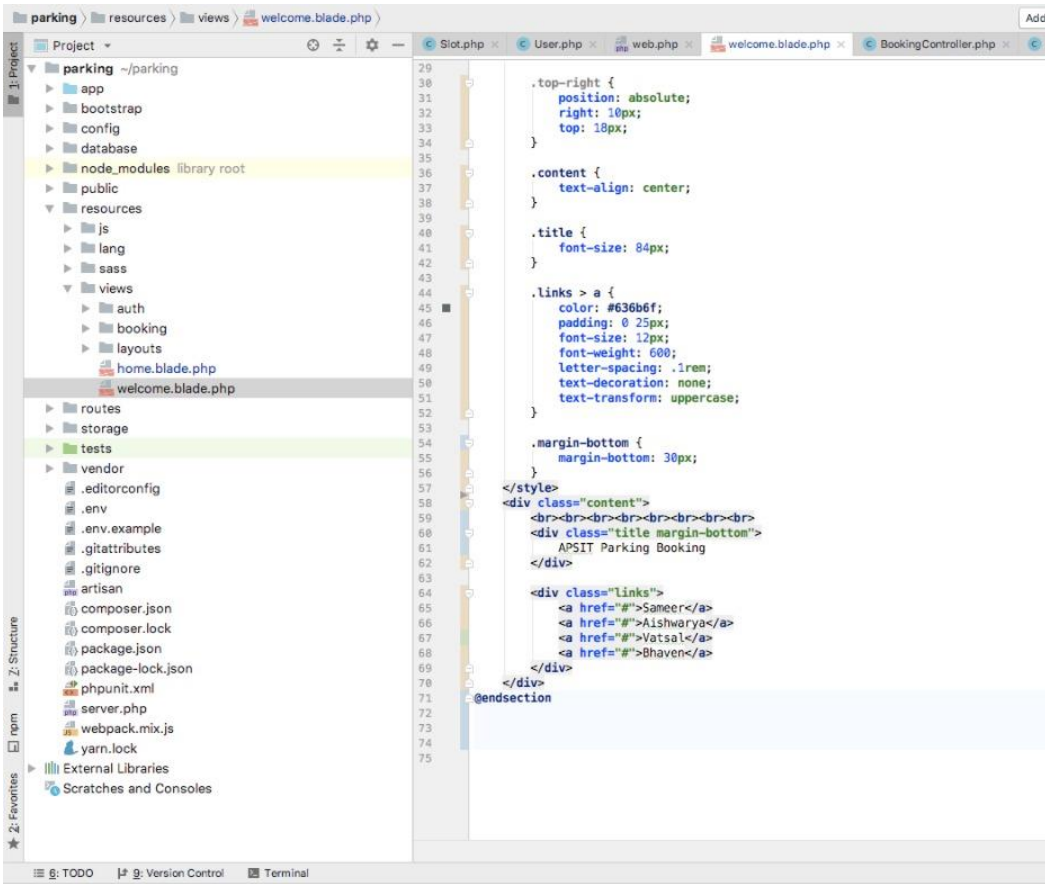
<hr>

@endsection
p
6: TODO
9: Version Control
Terminal


parking
app
Booking.php
Project
parking ~/parking
app
Console
Exceptions
Http
Providers
Booking.php
Model.php
Slot.php
User.php
bootstrap
config
database
node_modules library root
public
resources
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
BookingController.php
HomeController.php
SlotController.php
Booking.php
<?php
namespace App;
class Booking extends Model
{
public function user()
{
return $this->belongsTo(related: User::class);
}
public function slot()
{
return $this->belongsTo(related: Slot::class);
}
}
\App
6: TODO
9: Version Control
Terminal


parking › resources › views › welcome.blade.php
Project
parking ~/parking
app
bootstrap
config
database
node_modules library root
public
resources
js
lang
sass
views
auth
booking
layouts
home.blade.php
welcome.blade.php
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
web.php
welcome.blade.php
BookingController.php
@extends('layouts.app')
@section('content')
<link href="https://fonts.googleapis.com/css?family=Nunito:200,600" rel="styl
<style>
html, body {
background-color: #edd9c0;
color: #636b6f;
font-family: 'Nunito', sans-serif;
font-weight: 200;
height: 100vh;
margin: 0;
}
.full-height {
height: 100vh;
}
.flex-center {
align-items: center;
display: flex;
justify-content: center;
}
.position-ref {
position: relative;
}
.top-right {
position: absolute;
right: 10px;
top: 18px;
}
.content {
text-align: center;
}
.title {
font-size: 84px;
}
.links > a {
color: #636b6f;
padding: 0 25px;
font-size: 12px;
font-weight: 600;
letter-spacing: .1rem;
text-decoration: none;
text-transform: uppercase;
}
6: TODO
9: Version Control
Terminal

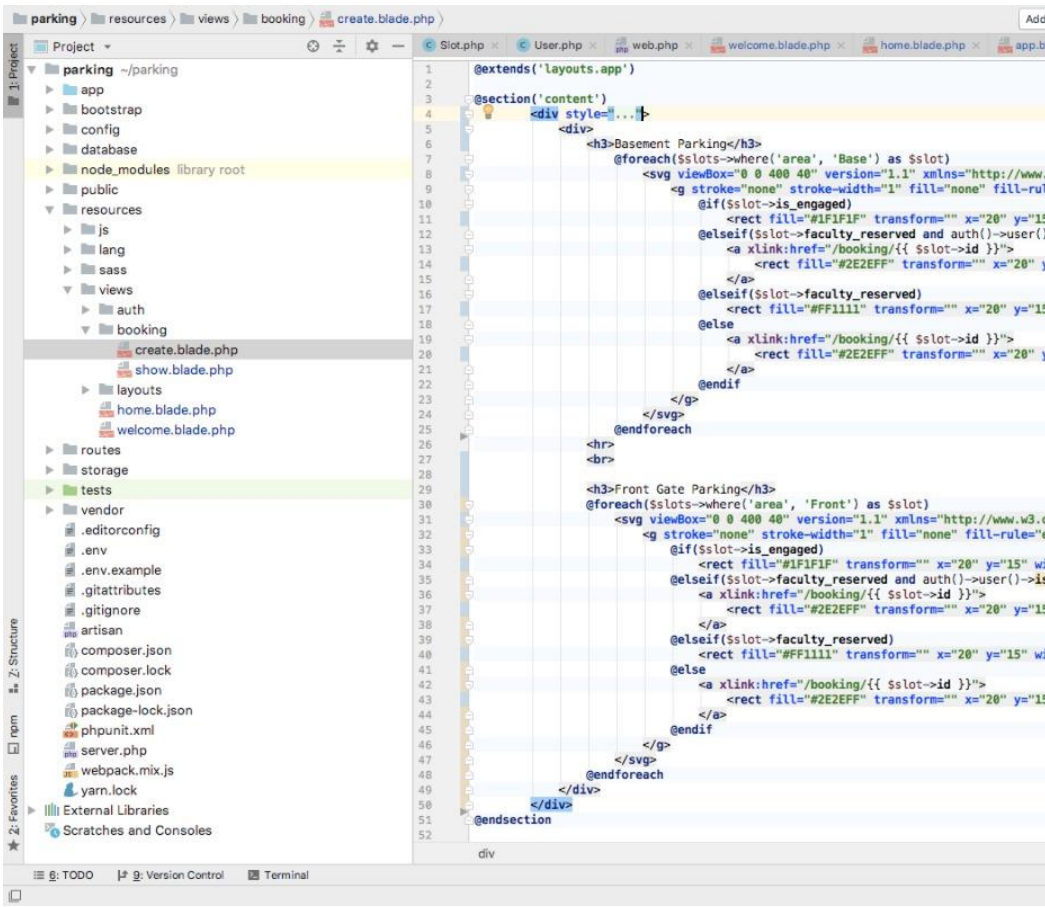

parking › routes › web.php
Project
parking ~/parking
app
bootstrap
config
database
node_modules library root
public
resources
routes
api.php
channels.php
console.php
web.php
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
web.php
BookingController.php
HomeController.php
<?php
/*...*/
Route::get('/', function () {
return view(view: 'welcome');
});
Auth::routes();
Route::get('/home', 'HomeController@index')->name('home');
Route::get('/booking', 'BookingController@create');
Route::get('/booking/{slot}', 'BookingController@show');
Route::post('/booking/{slot}/store', 'BookingController@store');
6: TODO
9: Version Control
Terminal


parking
resources
views
welcome.blade.php
Project
parking ~/parking
app
bootstrap
config
database
node_modules library root
public
resources
js
lang
sass
views
auth
booking
layouts
home.blade.php
welcome.blade.php
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
web.php
welcome.blade.php
BookingController.php
.top-right {
position: absolute;
right: 10px;
top: 18px;
}
.content {
text-align: center;
}
.title {
font-size: 84px;
}
.links > a {
color: #636b6f;
padding: 0 25px;
font-size: 12px;
font-weight: 600;
letter-spacing: .1rem;
text-decoration: none;
text-transform: uppercase;
}
.margin-bottom {
margin-bottom: 30px;
}
</style>

APSIT Parking Booking

<a href="#">Sameer</a>
<a href="#">Aishwarya</a>
<a href="#">Vatsal</a>
<a href="#">Bhaven</a>

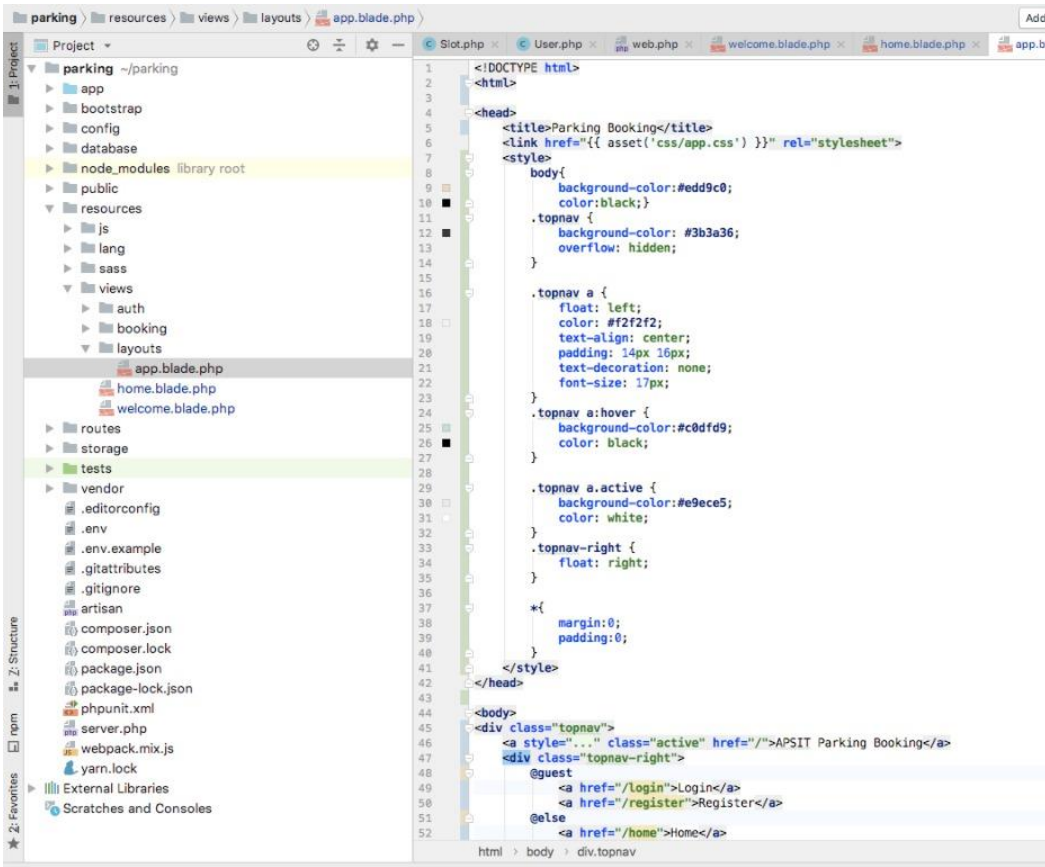
@endsection
6: TODO
9: Version Control
Terminal


parking
resources
views
booking
create.blade.php
Project
parking ~/parking
app
bootstrap
config
database
node_modules library root
public
resources
js
lang
sass
views
auth
booking
create.blade.php
show.blade.php
layouts
home.blade.php
welcome.blade.php
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
web.php
welcome.blade.php
home.blade.php
@extends('layouts.app')
@section('content')

<h3>Basement Parking</h3>
@foreach($slots->where('area', 'Base') as $slot)
<svg viewBox="0 0 400 40" version="1.1" xmlns="http://www.
<g stroke="none" stroke-width="1" fill="none" fill-rul
@if($slot->is_engaged)
<rect fill="#1F1F1F" transform="" x="20" y="15
@elseif($slot->faculty_reserved and auth()->user()
<a xlink:href="/booking/{{ $slot->id }}">
<rect fill="#2E2EFF" transform="" x="20" y
</a>
@elseif($slot->faculty_reserved)
<rect fill="#FF1111" transform="" x="20" y="15
@else
<a xlink:href="/booking/{{ $slot->id }}">
<rect fill="#2E2EFF" transform="" x="20" y
</a>
@endif
</g>
</svg>
@endforeach
<hr>

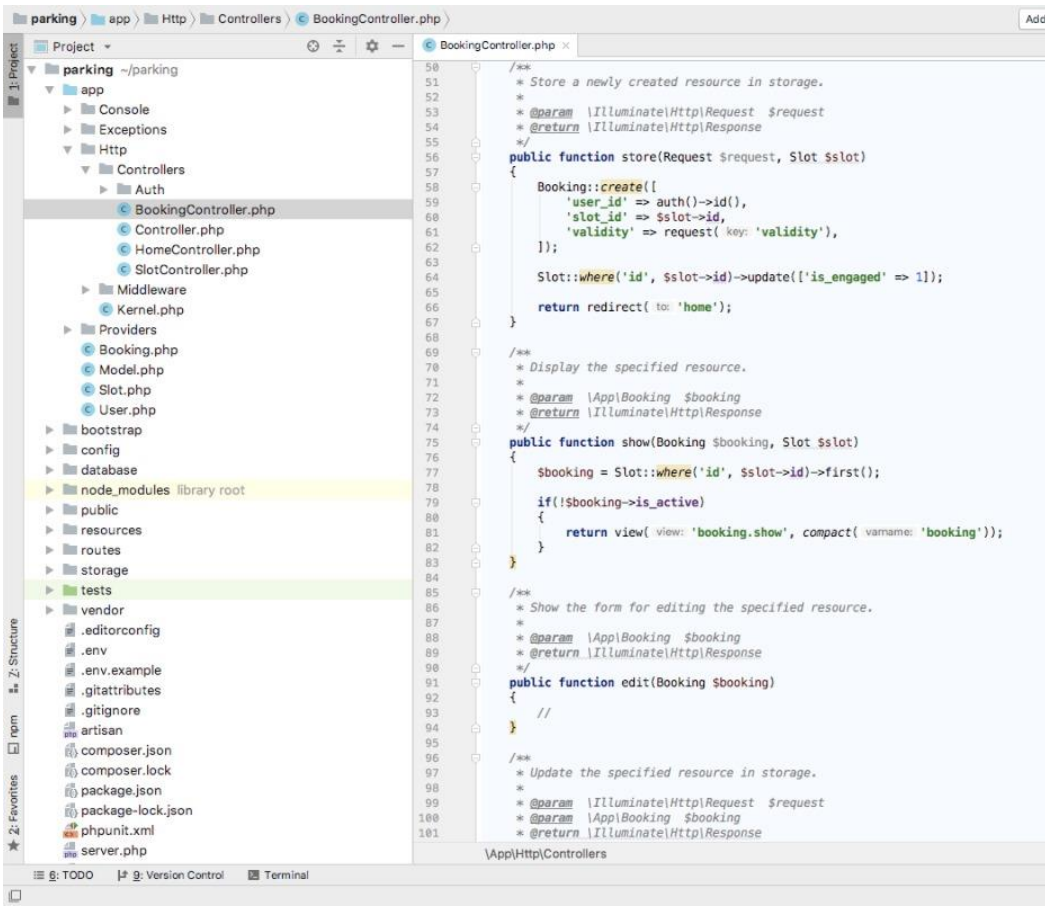
<h3>Front Gate Parking</h3>
@foreach($slots->where('area', 'Front') as $slot)
<svg viewBox="0 0 400 40" version="1.1" xmlns="http://www.w3.o
<g stroke="none" stroke-width="1" fill="none" fill-rule="
@if($slot->is_engaged)
<rect fill="#1F1F1F" transform="" x="20" y="15" w
@elseif($slot->faculty_reserved and auth()->user()->is
<a xlink:href="/booking/{{ $slot->id }}">
<rect fill="#2E2EFF" transform="" x="20" y="15
</a>
@elseif($slot->faculty_reserved)
<rect fill="#FF1111" transform="" x="20" y="15" w
@else
<a xlink:href="/booking/{{ $slot->id }}">
<rect fill="#2E2EFF" transform="" x="20" y="15
</a>
@endif
</g>
</svg>
@endforeach

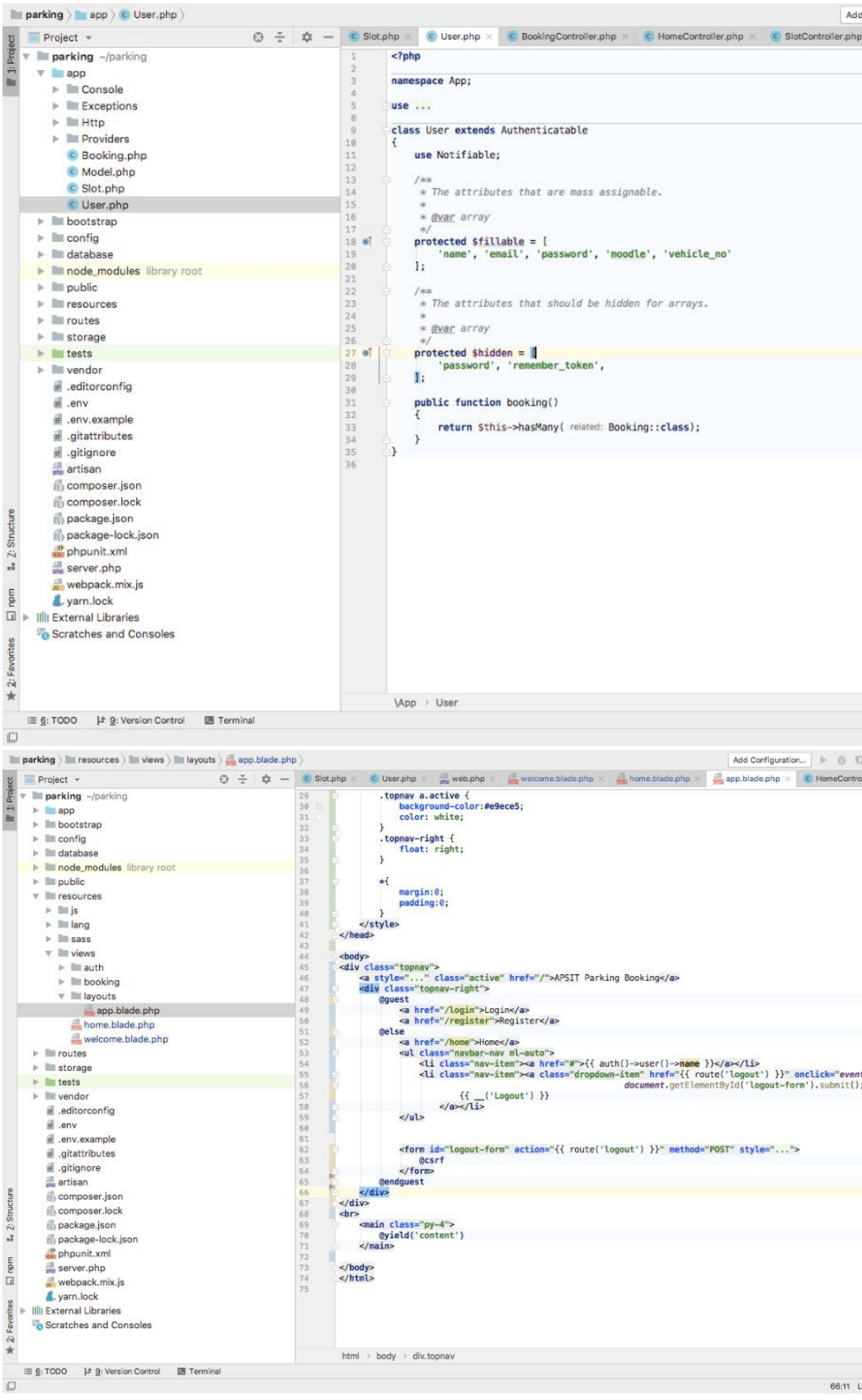
@endsection
div
6: TODO
9: Version Control
Terminal


parking › resources › views › layouts › app.blade.php
Project
parking ~/parking
app
bootstrap
config
database
node_modules library root
public
resources
js
lang
sass
views
auth
booking
layouts
app.blade.php
home.blade.php
welcome.blade.php
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
web.php
welcome.blade.php
home.blade.php
<!DOCTYPE html>
<html>
<head>
<title>Parking Booking</title>
<link href="{{ asset('css/app.css') }}" rel="stylesheet">
<style>
body{
background-color:#edd9c0;
color:black;}
.topnav {
background-color: #3b3a36;
overflow: hidden;
}
.topnav a {
float: left;
color: #f2f2f2;
text-align: center;
padding: 14px 16px;
text-decoration: none;
font-size: 17px;
}
.topnav a:hover {
background-color:#c0dfd9;
color: black;
}
.topnav a.active {
background-color:#e9ece5;
color: white;
}
.topnav-right {
float: right;
}
*{
margin:0;
padding:0;
}
</style>
</head>
<body>

<a style="..." class="active" href="/">APSIT Parking Booking</a>

@guest
<a href="/login">Login</a>
<a href="/register">Register</a>
@else
<a href="/home">Home</a>
html › body › div.topnav
6: TODO
9: Version Control
Terminal


parking › app › Http › Controllers › BookingController.php
Project
parking ~/parking
app
Console
Exceptions
Http
Controllers
Auth
BookingController.php
Controller.php
HomeController.php
SlotController.php
Middleware
Kernel.php
Providers
Booking.php
Model.php
Slot.php
User.php
bootstrap
config
database
node_modules library root
public
resources
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
BookingController.php
/**
* Store a newly created resource in storage.
*
* @param \Illuminate\Http\Request $request
* @return \Illuminate\Http\Response
*/
public function store(Request $request, Slot $slot)
{
Booking::create([
'user_id' => auth()->id(),
'slot_id' => $slot->id,
'validity' => request(key: 'validity'),
]);
Slot::where('id', $slot->id)->update(['is_engaged' => 1]);
return redirect(to: 'home');
}
/**
* Display the specified resource.
*
* @param \App\Booking $booking
* @return \Illuminate\Http\Response
*/
public function show(Booking $booking, Slot $slot)
{
$booking = Slot::where('id', $slot->id)->first();
if(!$booking->is_active)
{
return view(view: 'booking.show', compact(varname: 'booking'));
}
}
/**
* Show the form for editing the specified resource.
*
* @param \App\Booking $booking
* @return \Illuminate\Http\Response
*/
public function edit(Booking $booking)
{
//
}
/**
* Update the specified resource in storage.
*
* @param \Illuminate\Http\Request $request
* @param \App\Booking $booking
* @return \Illuminate\Http\Response
\App\Http\Controllers
6: TODO
9: Version Control
Terminal


parking app User.php
Project
parking ~/parking
app
Console
Exceptions
Http
Providers
Booking.php
Model.php
Slot.php
User.php
bootstrap
config
database
node_modules library root
public
resources
routes
storage
tests
vendor
.editorconfig
.env
.env.example
.gitattributes
.gitignore
artisan
composer.json
composer.lock
package.json
package-lock.json
phpunit.xml
server.php
webpack.mix.js
yarn.lock
External Libraries
Scratches and Consoles
Slot.php
User.php
BookingController.php
HomeController.php
SlotController.php
<?php
namespace App;
use ...
class User extends Authenticatable
{
use Notifiable;
/**
* The attributes that are mass assignable.
*
* @var array
*/
protected $fillable = [
'name', 'email', 'password', 'moodle', 'vehicle_no'
];
/**
* The attributes that should be hidden for arrays.
*
* @var array
*/
protected $hidden = [
'password', 'remember_token',
];
public function booking()
{
return $this->hasMany(related: Booking::class);
}
}
\App User
6: TODO
9: Version Control
Terminal
parking resources views layouts app.blade.php
Add Configuration...
Slot.php
User.php
web.php
welcome.blade.php
home.blade.php
app.blade.php
.topnav a.active {
background-color:#e9ece5;
color: white;
}
.topnav-right {
float: right;
}
*{
margin:0;
padding:0;
}
</style>
</head>
<body>

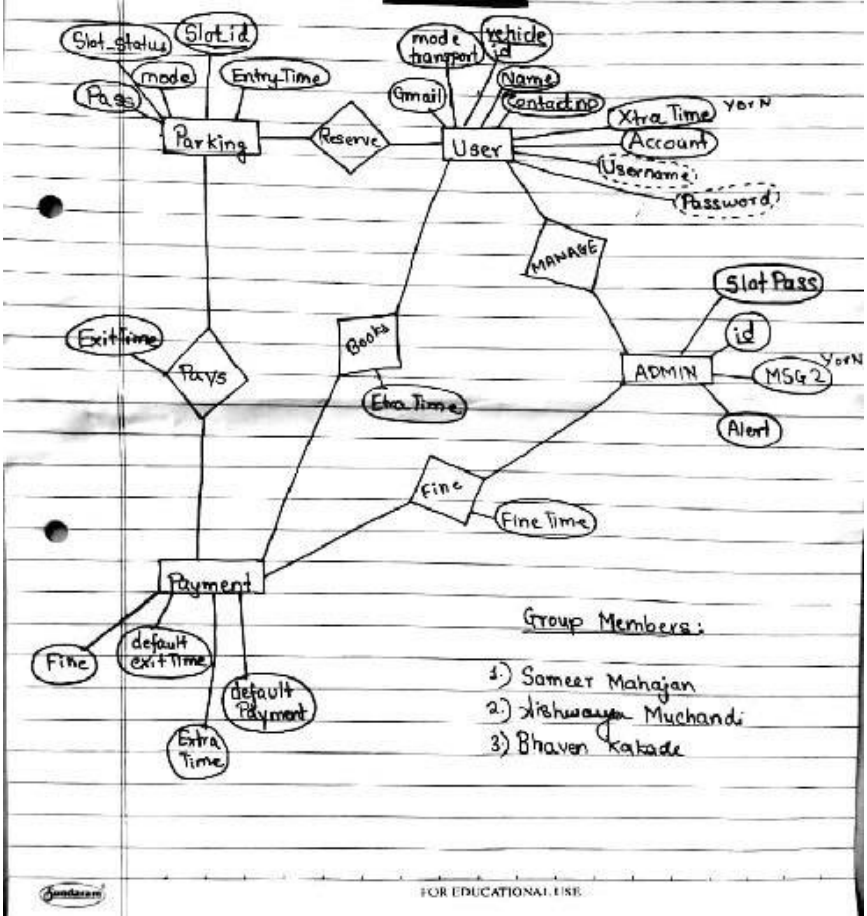
<a style="..." class="active" href="/">APSIT Parking Booking</a>

@guest
<a href="/login">Login</a>
<a href="/register">Register</a>
@else
<a href="/home">Home</a>
<ul class="navbar-nav ml-auto">
<li class="nav-item"><a href="#">{{ auth()->user()->name }}</a></li>
<li class="nav-item"><a class="dropdown-item" href="{{ route('logout') }}" onclick="event
document.getElementById('logout-form').submit();
{{ __('Logout') }}
</a></li>
</ul>
<form id="logout-form" action="{{ route('logout') }}" method="POST" style="...">
@csrf
</form>
@endguest

<main class="py-4">
@yield('content')
</main>
</body>
</html>
html body div.topnav
66:11

ONLINE PARKING SLOT.

ER DIAGRAM



FOR EDUCATIONAL USE

Conclusions