**E-PROJECT**

**REPORT**

VIMS

VEHICLE INSURANCE MANAGEMENT SYSTEM

***Hussain Abbas***

*Student1042319*

***BY***

***STUDENT OF***

***1707F***

***APTECH GARDEN***

***CENTRE***

**ACKNOWLEDGEMENT**

I would like to acknowledge all those who have given me moral support and helped in making the project a success.

I wish to express my gratitude to e Project Team at APTECH GARDEN CENTER who guided and helped me. Also, I would like to express my gratitude to all the staff members of my center for not only providing me opportunity to work with them on this project, but also for their support and encouragement throughout the process.

I also express my sincere gratitude to e Project Team from Aptech Garden Center and my project guide at the organization for its valuable guidance and support for completion of this project.

|  |  |
| --- | --- |
|  | Table of content |
| 1. | **Project Synopsis** |
| 1.1 | OBJECTIVE |
| 1.2 | PROBLEM STATEMENT |
| 2. | **Project Analysis** |
| 2.1 | TOOLS AND TECHNOLIGIES |
| 2.2 | STAGES OF PROJECT |
| 2.3 | REFERENCES |
| 3. | **Design Details** |
| 4 | **Source code** |
| 5. | **User Guide** |
| 6. | **Developer Guide** |
| 6.1 | REQUIREMENTS |
| 6.2 | STEPS TO OPEN |
| 6.3 | MODULE DESCRIPTION |

**PROJECT SYNOPSIS**

* OBJECTIVE:

You are expected to develop a ‘Vehicle Insurance Management System (VIMS)’ so that users can manage the details and admin functionalities.

* PROBLEM STATEMENT:

The product entitled “automation” aim at automation of many processes of Vehicle Insurance. Current manual system that is employed is extremely laborious and quite in adequate it only makes the process more difficult and hard. Main purpose of this software should be to reduce the manpower, paperwork and provide the security from unauthorized access. It should also help in producing difficult kinds of report required for different managerial aspects.

**PROJECT ANALYSIS**

* TOOLS AND TECHNOLOGIES:
* **TOOLS**:
* Visual Studio .Net / ASP
* IIS server
* .Net Framework
* Java Virtual Machine/ J2EE server
* Notepad/Java editor
* j2sdk1.4.1\_02 (or later).
* EJB Dev Kit
* Java enabled web server
* JSP / Servlets Dev. Kit
* **TECHNOLOGIES**:
* C#
* STAGES OF PROJECT:

|  |  |  |
| --- | --- | --- |
| **S.No** | **STAGES OF PROJECT** | **DAYS** |
| 1 | Layout Designing | 5 |
| 2 | Coding | 4 |
| 3 | Collecting Data | 2 |
| 4 | Testing | 2 |
| 5 | Report | 1 |

* REFERENCES:

--

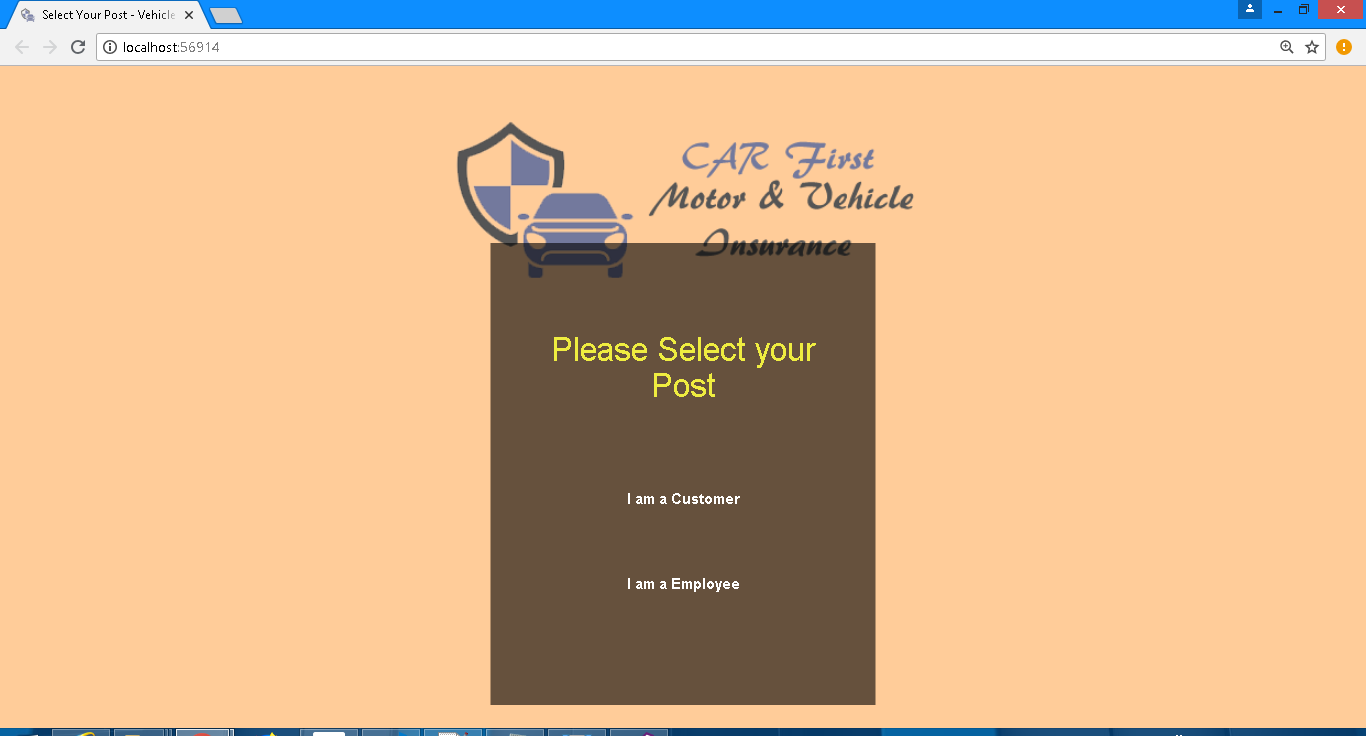
**Design Details**

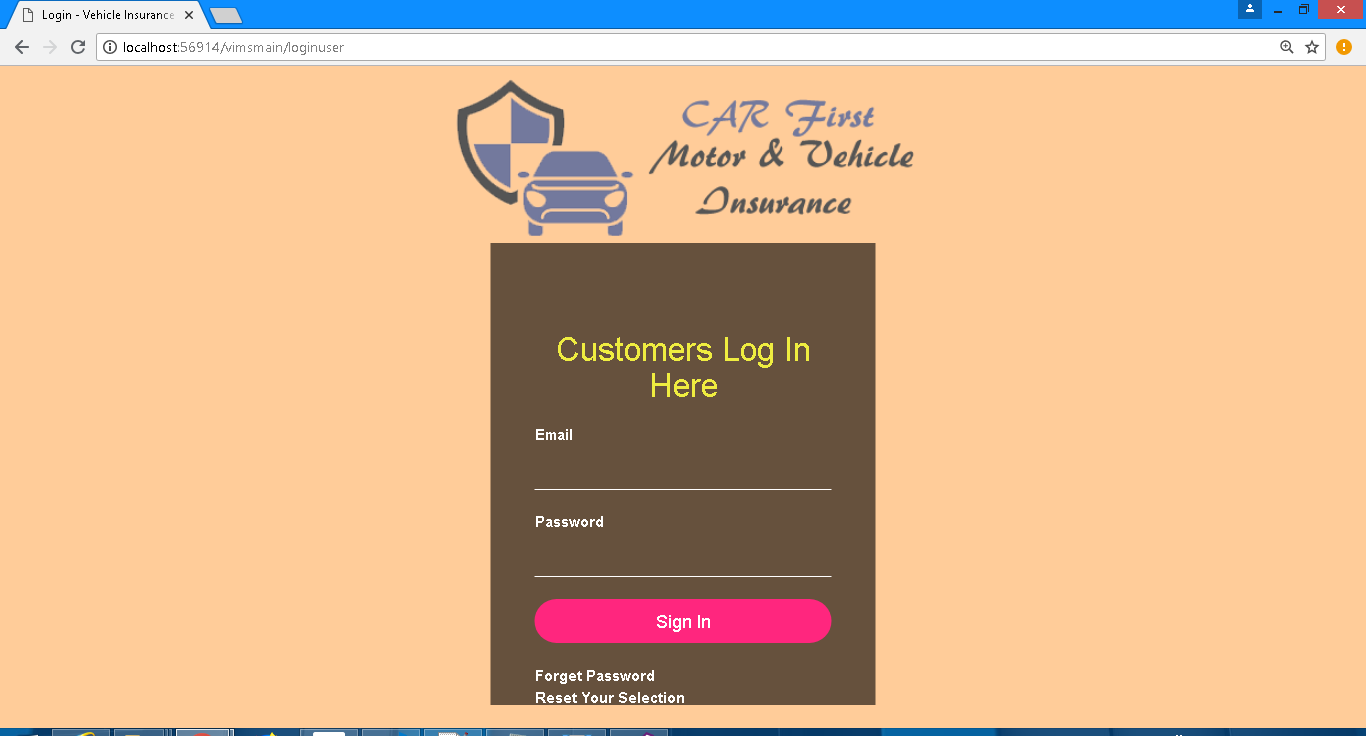
* + LAYOUT: The layout is made of tabs specified for different section it is a very good way to make the User Experience better and simplified.
  + COLOR: We have used colors to make the software look more attractive. We have mainly use baige and white color combination. We have gone for a minimalist look to make the website/webpages look more sleek and professional.

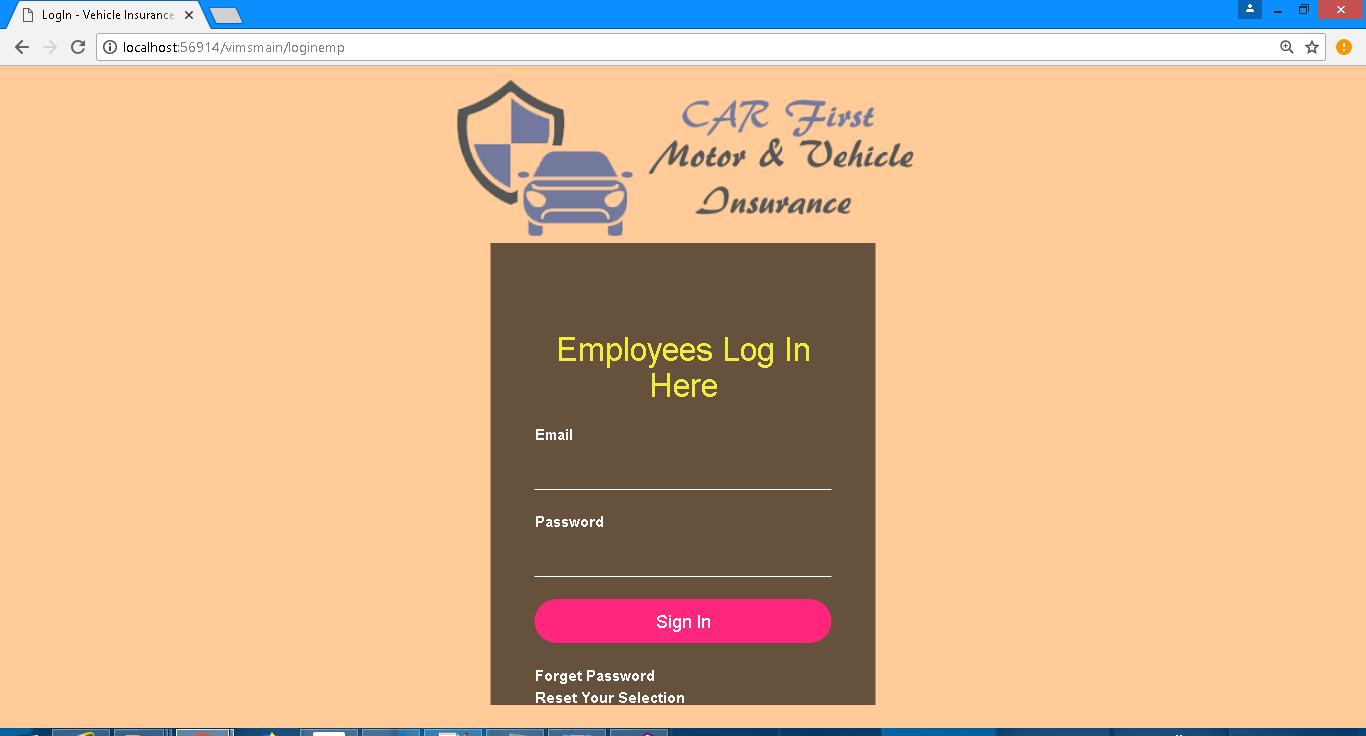
**SCREEN SHOTS**

SELECT YOUR POST SCREEN

Start screen of the software is the place where every user has to go through every time thay are going to select weather they are a customer or an employee.

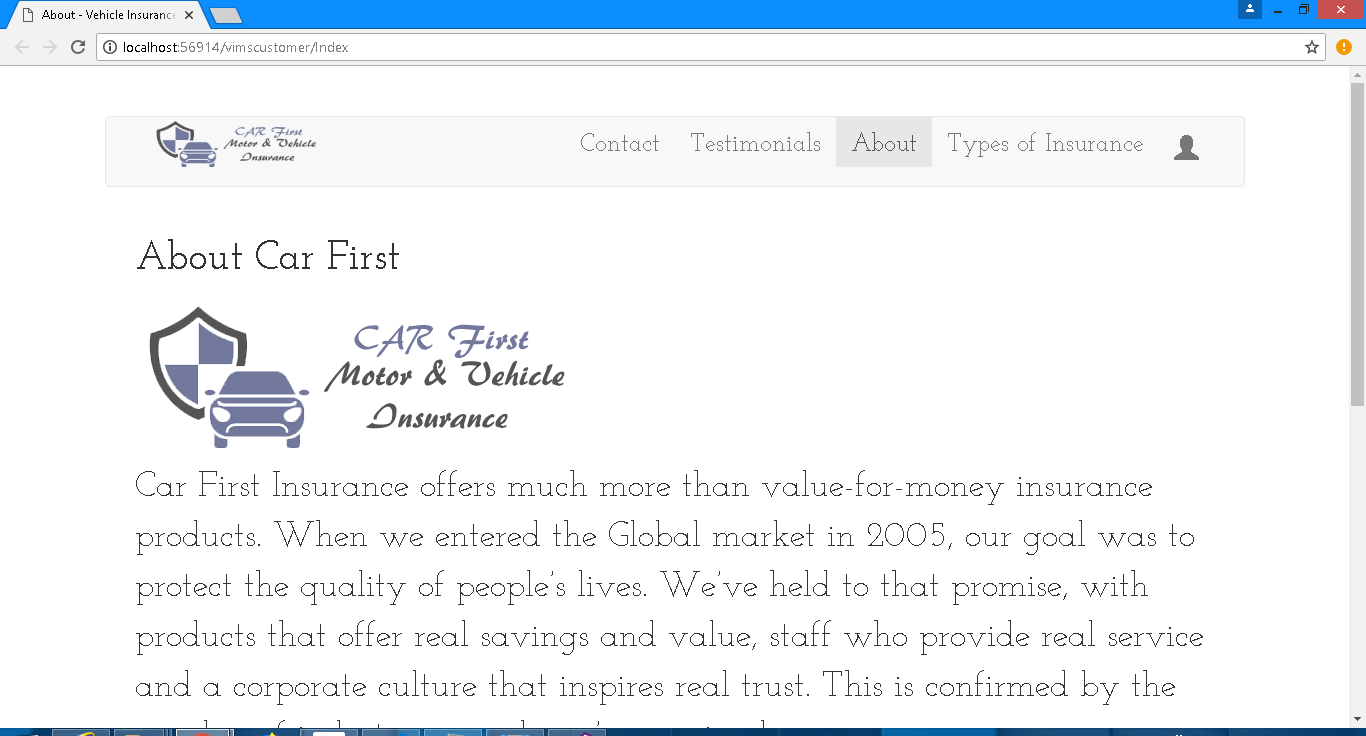


LOGIN SCREENS



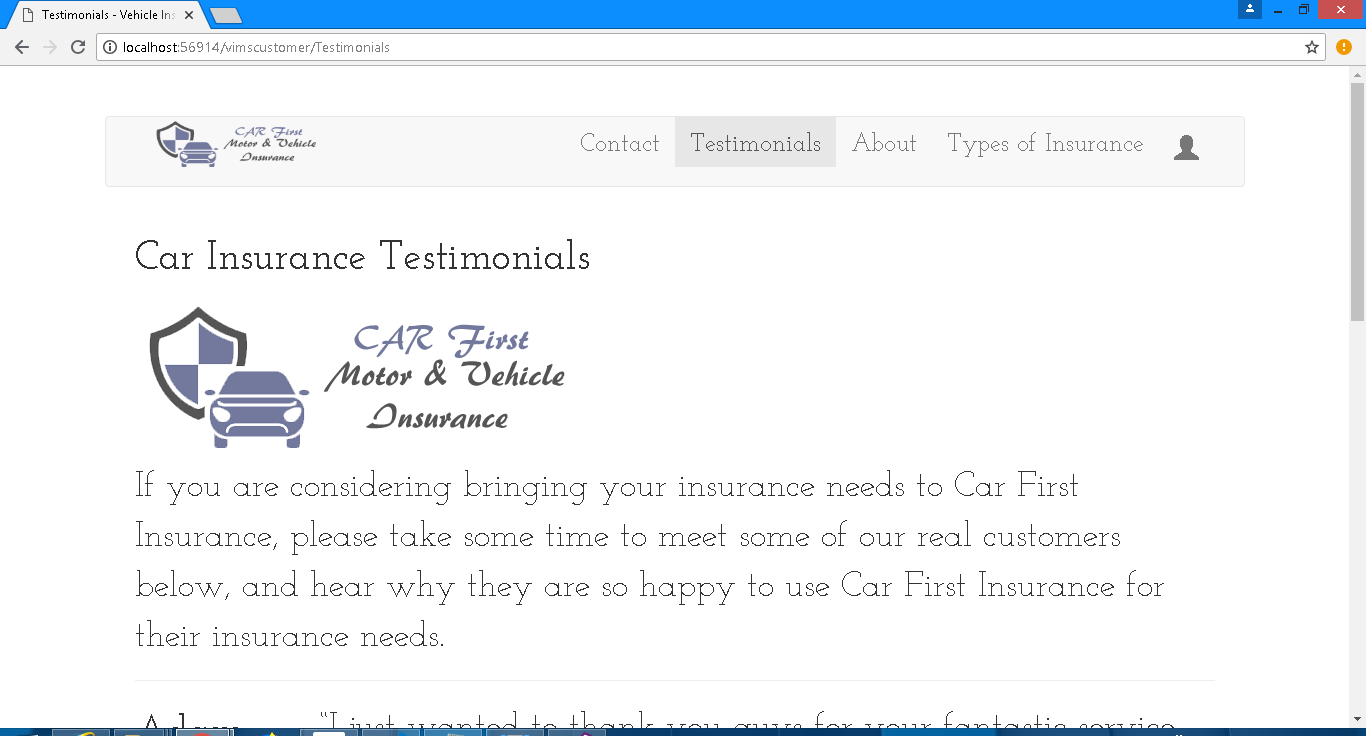
About Page

This page tells about the Insurance company.



Testimonial

This page tells us about the testimonials



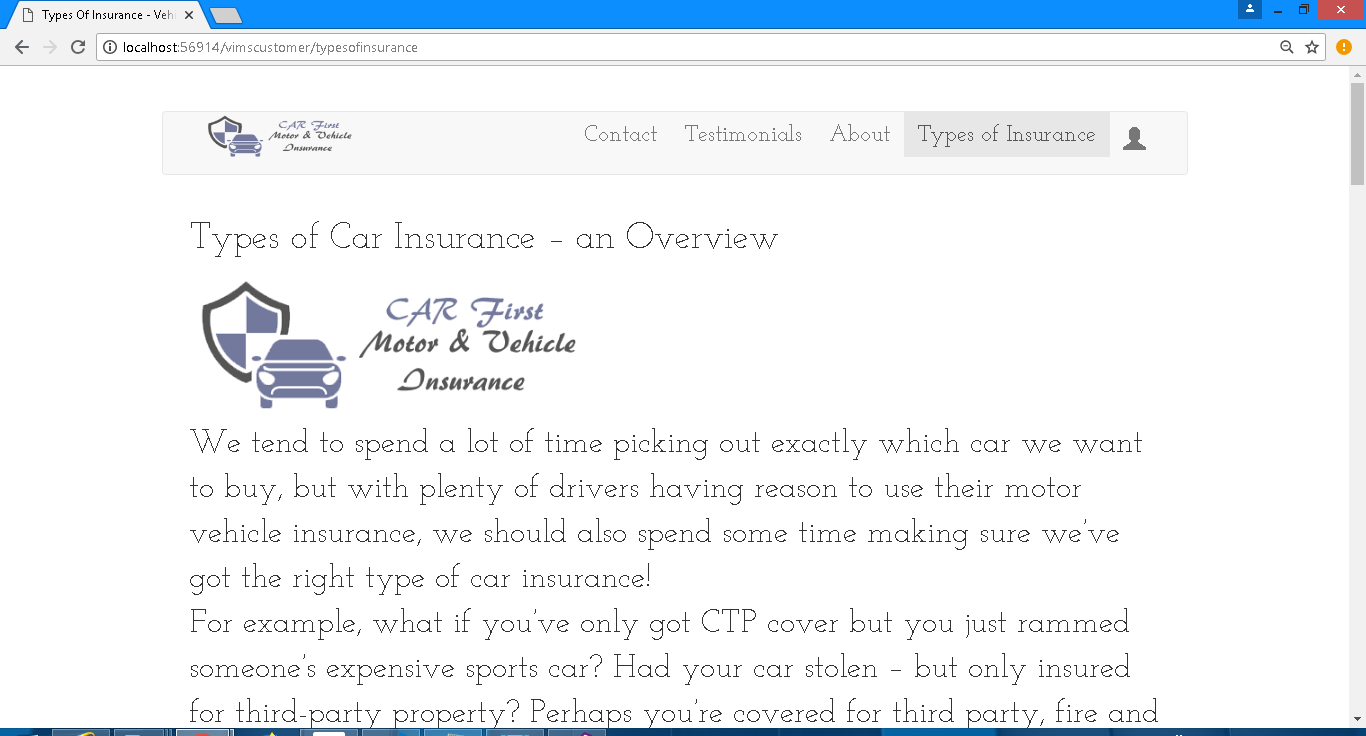
Contact

This page tells us about Contact details



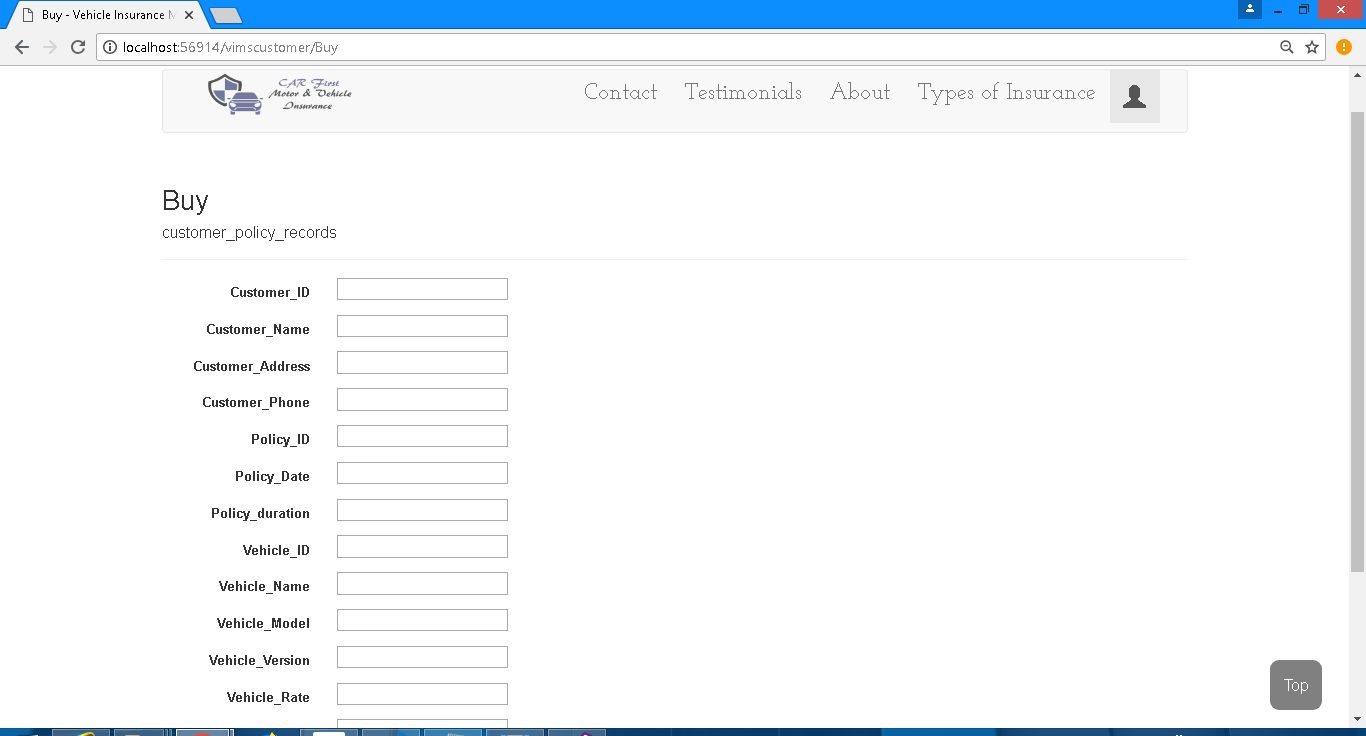
Types Of Insurance

This page tells us about types of insurance that the company offers.



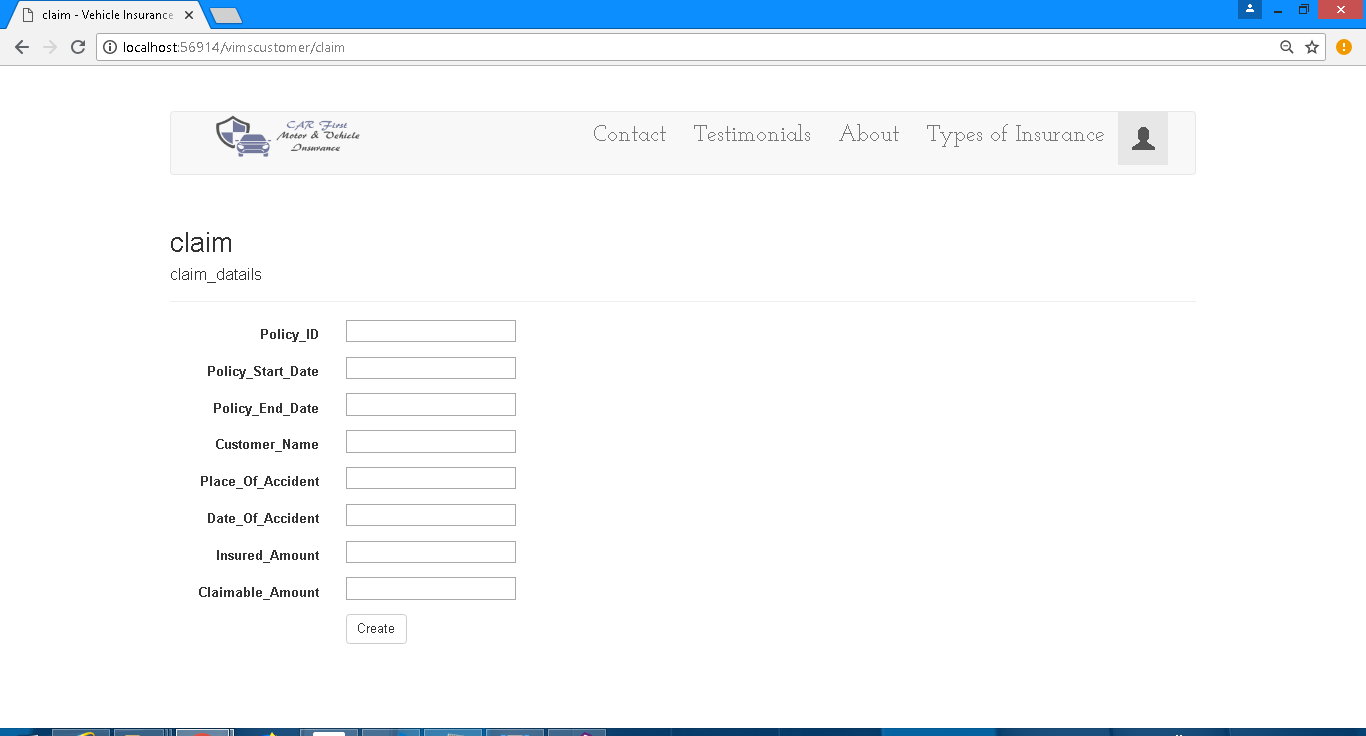
Buy A Policy

With this page a customer can buy a policy.

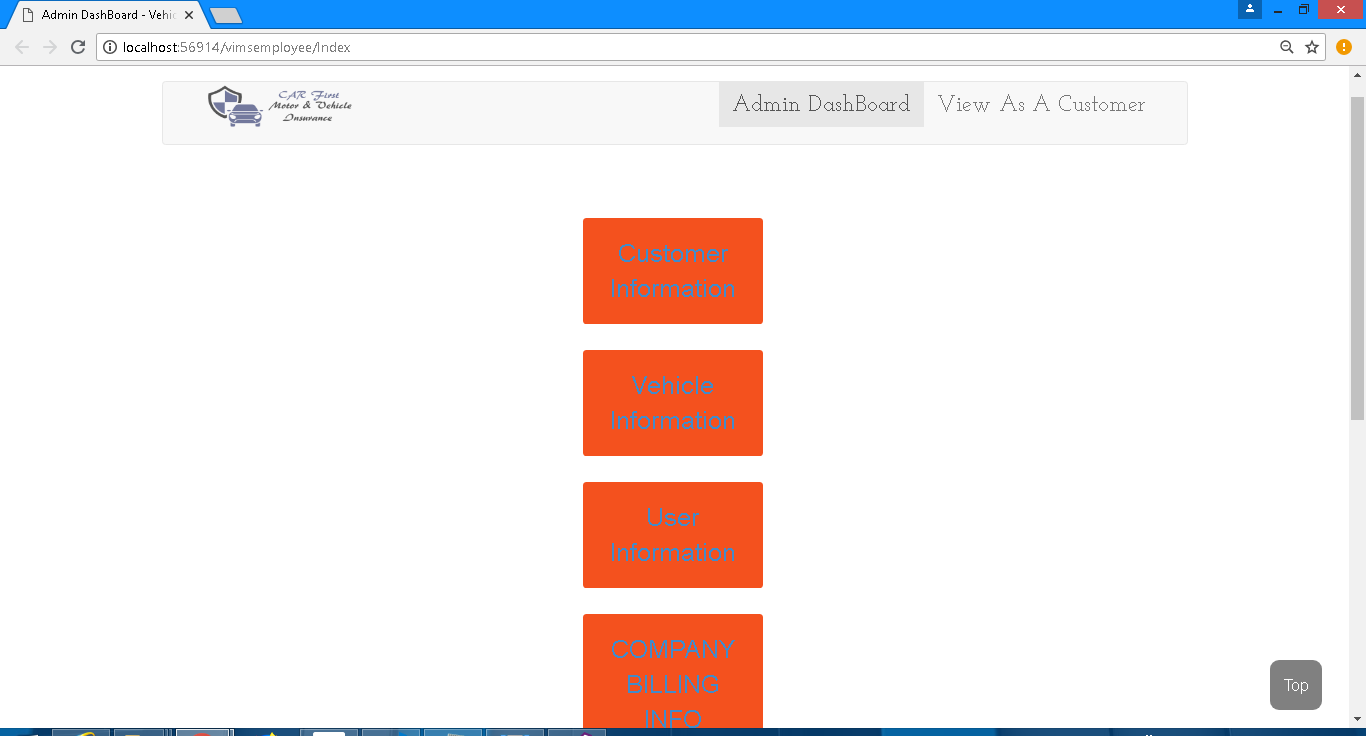


Make a Claim

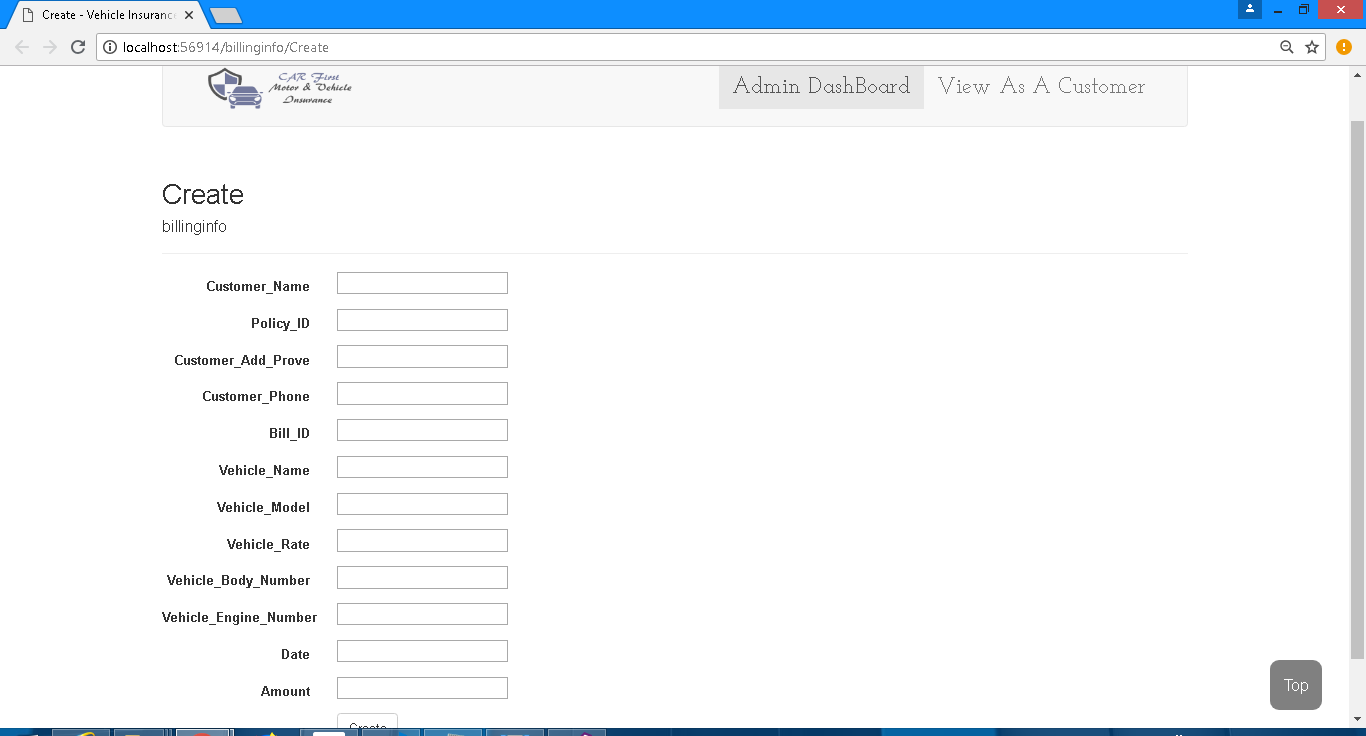
With this page a customer can make a claim



Admin DashBoard

With this page admin can manage all the records in one place

Management of Records(e.g Billing Info)



**SOURCE CODE**

SQL Query(please make sure to execute this first)

create database vims

use vims

CREATE TABLE userinfo(

     userID int PRIMARY KEY IDENTITY(1,1),

     userName varchar(255),

     userpassword varchar(255)

);

CREATE TABLE customerinfo(

     Customer\_ID int PRIMARY KEY IDENTITY(1,1),

     Customer\_Name varchar(255),

     Customer\_Address varchar(255),

     Customer\_Phone int

);

CREATE TABLE vehicleinfo(

     Vehicle\_ID int PRIMARY KEY IDENTITY(1,1),

     Vehicle\_Name varchar(255),

     Vehicles\_Owner\_Address varchar(255),

     Vehicle\_Model varchar(255),

Vehicle\_Version varchar(255),

Vehicle\_Rate int,

Vehicle\_Body\_Number varchar(255),

Vehicle\_Engine\_Number varchar(255),

Vehicle\_Number int

);

CREATE TABLE estimate(

     Customer\_ID int PRIMARY KEY,

     Estimate\_Number int,

Customer\_Name varchar(255),

     Customer\_Phone int,

Vehicle\_Name varchar(255),

     Vehicle\_Model varchar(255),

Vehicle\_Rate int,

Vehicle\_Waranty varchar(255),

Vehicle\_Policy\_type varchar(255),

);

CREATE TABLE customer\_policy\_records(

Customer\_ID int PRIMARY KEY,

     Customer\_Name varchar(255),

     Customer\_Address varchar(255),

     Customer\_Phone int,

Policy\_ID int,

Policy\_Date varchar(225),

Policy\_duration int,

Vehicle\_ID int,

     Vehicle\_Name varchar(255),

Vehicle\_Model varchar(255),

Vehicle\_Version varchar(255),

Vehicle\_Rate int,

Vehicle\_Warranty varchar(225),

Vehicle\_Body\_Number varchar(255),

Vehicle\_Engine\_Number varchar(255),

Customer\_Add\_Prove varchar(255)

);

CREATE TABLE billinginfo(

Customer\_ID int PRIMARY KEY,

     Customer\_Name varchar(255),

     Policy\_ID int,

Customer\_Add\_Prove varchar(255),

Customer\_Phone int,

Bill\_ID int,

     Vehicle\_Name varchar(255),

Vehicle\_Model varchar(255),

Vehicle\_Rate int,

Vehicle\_Body\_Number varchar(255),

Vehicle\_Engine\_Number varchar(255),

Date varchar(255),

Amount int

);

CREATE TABLE company\_expenses(

Expence\_ID int PRIMARY KEY IDENTITY(1,1),

Date\_of\_Expence varchar(255),

Type\_of\_Expence varchar(255),

Amount\_of\_Expence varchar(255),

);

CREATE TABLE claim\_datails(

Claim\_ID int PRIMARY KEY IDENTITY(1,1),

Policy\_ID int,

Policy\_Start\_Date varchar(255),

Policy\_End\_Date varchar(255),

Customer\_Name varchar(255),

Place\_Of\_Accident varchar(255),

Date\_Of\_Accident varchar(255),

Insured\_Amount int,

Claimable\_Amount int

);

INSERT INTO userinfo VALUES ('user', 'user');

INSERT INTO userinfo VALUES ('admin', 'admin');

Vimsmaincontroler

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace vimseproject.Controllers

{

public class vimsmainController : Controller

{

//

// GET: /vimsmain/

public ActionResult Index()

{

return View();

}

public ActionResult loginemp()

{

return View();

}

public ActionResult loginuser()

{

return View();

}

public ActionResult signupuser()

{

return View();

}

}

}

Vimsemployeecontroller.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace vimseproject.Controllers

{

public class vimsemployeeController : Controller

{

//

// GET: /vimsemployee/

public ActionResult Index()

{

return View();

}

public ActionResult Testimonials()

{

return View();

}

}

}

Vimscustomercontroller.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using vimseproject.Models;

namespace vimseproject.Controllers

{

public class vimscustomerController : Controller

{

vimsEntities db = new vimsEntities();

//

// GET: /vimscustomer/

public ActionResult Index()

{

return View();

}

public ActionResult Testimonials()

{

return View();

}

public ActionResult Contact()

{

return View();

}

public ActionResult typesofinsurance()

{

return View();

}

public ActionResult ChangePassword()

{

return View();

}

// add view of buy policy from ado.net

public ActionResult Buy()

{

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Buy([Bind(Include = "Customer\_ID,Customer\_Name,Customer\_Address,Customer\_Phone,Policy\_ID,Policy\_Date,Policy\_duration,Vehicle\_ID,Vehicle\_Name,Vehicle\_Model,Vehicle\_Version,Vehicle\_Rate,Vehicle\_Warranty,Vehicle\_Body\_Number,Vehicle\_Engine\_Number,Customer\_Add\_Prove")] customer\_policy\_records customer\_policy\_records)

{

if (ModelState.IsValid)

{

db.customer\_policy\_records.Add(customer\_policy\_records);

db.SaveChanges();

return RedirectToAction("Index");

}

return View();

}

// add view of claim policy from ado.net

public ActionResult claim()

{

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult claim([Bind(Include = "Claim\_ID,Policy\_ID,Policy\_Start\_Date,Policy\_End\_Date,Customer\_Name,Place\_Of\_Accident,Date\_Of\_Accident,Insured\_Amount,Claimable\_Amount")] claim\_datails claim\_datails)

{

if (ModelState.IsValid)

{

db.claim\_datails.Add(claim\_datails);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(claim\_datails);

}

public ActionResult changepass()

{

return View();

}

}

}

\*(AS THEIR WERE A LOT OF TABLES IN THE DATA BASE, THAT IS WHY WE ARE ONLY SHOWING THE SOURCE CODE OF ONE TABLE CONTROLLER AS AN EXAMPLE YOU CAN FIND THE OTHER CODE IN THE MAIN PROJECT FILE[.sln] )

vehicleinfocontroller.cs

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.Entity;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

using vimseproject.Models;

namespace vimseproject.Controllers

{

public class vehicleinfoController : Controller

{

private vimsEntities db = new vimsEntities();

// GET: /vehicleinfo/

public ActionResult Index()

{

return View(db.vehicleinfoes.ToList());

}

// GET: /vehicleinfo/Details/5

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

vehicleinfo vehicleinfo = db.vehicleinfoes.Find(id);

if (vehicleinfo == null)

{

return HttpNotFound();

}

return View(vehicleinfo);

}

// GET: /vehicleinfo/Create

public ActionResult Create()

{

return View();

}

// POST: /vehicleinfo/Create

// To protect from overposting attacks, please enable the specific properties you want to bind to, for

// more details see http://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include="Vehicle\_ID,Vehicle\_Name,Vehicles\_Owner\_Address,Vehicle\_Model,Vehicle\_Version,Vehicle\_Rate,Vehicle\_Body\_Number,Vehicle\_Engine\_Number,Vehicle\_Number")] vehicleinfo vehicleinfo)

{

if (ModelState.IsValid)

{

db.vehicleinfoes.Add(vehicleinfo);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(vehicleinfo);

}

// GET: /vehicleinfo/Edit/5

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

vehicleinfo vehicleinfo = db.vehicleinfoes.Find(id);

if (vehicleinfo == null)

{

return HttpNotFound();

}

return View(vehicleinfo);

}

// POST: /vehicleinfo/Edit/5

// To protect from overposting attacks, please enable the specific properties you want to bind to, for

// more details see http://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include="Vehicle\_ID,Vehicle\_Name,Vehicles\_Owner\_Address,Vehicle\_Model,Vehicle\_Version,Vehicle\_Rate,Vehicle\_Body\_Number,Vehicle\_Engine\_Number,Vehicle\_Number")] vehicleinfo vehicleinfo)

{

if (ModelState.IsValid)

{

db.Entry(vehicleinfo).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(vehicleinfo);

}

// GET: /vehicleinfo/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

vehicleinfo vehicleinfo = db.vehicleinfoes.Find(id);

if (vehicleinfo == null)

{

return HttpNotFound();

}

return View(vehicleinfo);

}

// POST: /vehicleinfo/Delete/5

[HttpPost, ActionName("Delete")]

[ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

vehicleinfo vehicleinfo = db.vehicleinfoes.Find(id);

db.vehicleinfoes.Remove(vehicleinfo);

db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

}

index.cshtml(vehicleinfo)

@model IEnumerable<vimseproject.Models.vehicleinfo>

@{

ViewBag.Title = "Index";

}

<style>

.button1 {

border-radius: 4px;

background-color: #f4511e;

border: none;

color: #FFFFFF;

text-align: center;

font-size: 28px;

padding: 20px;

width: 200px;

transition: all 0.5s;

cursor: pointer;

margin: 5px;

}

.button1 span {

cursor: pointer;

display: inline-block;

position: relative;

transition: 0.5s;

}

.button1 span:after {

content: '\00bb';

position: absolute;

opacity: 0;

top: 0;

right: -20px;

transition: 0.5s;

}

.button1:hover span {

padding-right: 25px;

}

.button1:hover span:after {

opacity: 1;

right: 0;

}

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="@Url.Action("Index", "vimsemployee")">Admin DashBoard</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">View As A Customer</a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<h2>Index</h2>

<p>

@Html.ActionLink("Create New", "Create")

</p>

<table class="table">

<tr>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Name)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicles\_Owner\_Address)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Model)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Version)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Rate)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Body\_Number)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Engine\_Number)

</th>

<th>

@Html.DisplayNameFor(model => model.Vehicle\_Number)

</th>

<th></th>

</tr>

@foreach (var item in Model) {

<tr>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Name)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicles\_Owner\_Address)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Model)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Version)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Rate)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Body\_Number)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Engine\_Number)

</td>

<td>

@Html.DisplayFor(modelItem => item.Vehicle\_Number)

</td>

<td>

@Html.ActionLink("Edit", "Edit", new { id=item.Vehicle\_ID }) |

@Html.ActionLink("Details", "Details", new { id=item.Vehicle\_ID }) |

@Html.ActionLink("Delete", "Delete", new { id=item.Vehicle\_ID })

</td>

</tr>

}

</table>

Details.cshtml

@model vimseproject.Models.vehicleinfo

@{

ViewBag.Title = "Details";

}

<style>

.button1 {

border-radius: 4px;

background-color: #f4511e;

border: none;

color: #FFFFFF;

text-align: center;

font-size: 28px;

padding: 20px;

width: 200px;

transition: all 0.5s;

cursor: pointer;

margin: 5px;

}

.button1 span {

cursor: pointer;

display: inline-block;

position: relative;

transition: 0.5s;

}

.button1 span:after {

content: '\00bb';

position: absolute;

opacity: 0;

top: 0;

right: -20px;

transition: 0.5s;

}

.button1:hover span {

padding-right: 25px;

}

.button1:hover span:after {

opacity: 1;

right: 0;

}

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="@Url.Action("Index", "vimsemployee")">Admin DashBoard</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">View As A Customer</a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<h2>Details</h2>

<div>

<h4>vehicleinfo</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Name)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Name)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicles\_Owner\_Address)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicles\_Owner\_Address)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Model)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Model)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Version)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Version)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Rate)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Rate)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Body\_Number)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Body\_Number)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Engine\_Number)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Engine\_Number)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Number)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Number)

</dd>

</dl>

</div>

<p>

@Html.ActionLink("Edit", "Edit", new { id = Model.Vehicle\_ID }) |

@Html.ActionLink("Back to List", "Index")

</p>

create.cshtml

@model vimseproject.Models.vehicleinfo

@{

ViewBag.Title = "Create";

}

<style>

.button1 {

border-radius: 4px;

background-color: #f4511e;

border: none;

color: #FFFFFF;

text-align: center;

font-size: 28px;

padding: 20px;

width: 200px;

transition: all 0.5s;

cursor: pointer;

margin: 5px;

}

.button1 span {

cursor: pointer;

display: inline-block;

position: relative;

transition: 0.5s;

}

.button1 span:after {

content: '\00bb';

position: absolute;

opacity: 0;

top: 0;

right: -20px;

transition: 0.5s;

}

.button1:hover span {

padding-right: 25px;

}

.button1:hover span:after {

opacity: 1;

right: 0;

}

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="@Url.Action("Index", "vimsemployee")">Admin DashBoard</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">View As A Customer</a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<h2>Create</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h4>vehicleinfo</h4>

<hr />

@Html.ValidationSummary(true)

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Name, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Name)

@Html.ValidationMessageFor(model => model.Vehicle\_Name)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicles\_Owner\_Address, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicles\_Owner\_Address)

@Html.ValidationMessageFor(model => model.Vehicles\_Owner\_Address)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Model, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Model)

@Html.ValidationMessageFor(model => model.Vehicle\_Model)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Version, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Version)

@Html.ValidationMessageFor(model => model.Vehicle\_Version)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Rate, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Rate)

@Html.ValidationMessageFor(model => model.Vehicle\_Rate)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Body\_Number, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Body\_Number)

@Html.ValidationMessageFor(model => model.Vehicle\_Body\_Number)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Engine\_Number, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Engine\_Number)

@Html.ValidationMessageFor(model => model.Vehicle\_Engine\_Number)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Number, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Number)

@Html.ValidationMessageFor(model => model.Vehicle\_Number)

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Create" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

edit.cshtml

@model vimseproject.Models.vehicleinfo

@{

ViewBag.Title = "Edit";

}

<style>

.button1 {

border-radius: 4px;

background-color: #f4511e;

border: none;

color: #FFFFFF;

text-align: center;

font-size: 28px;

padding: 20px;

width: 200px;

transition: all 0.5s;

cursor: pointer;

margin: 5px;

}

.button1 span {

cursor: pointer;

display: inline-block;

position: relative;

transition: 0.5s;

}

.button1 span:after {

content: '\00bb';

position: absolute;

opacity: 0;

top: 0;

right: -20px;

transition: 0.5s;

}

.button1:hover span {

padding-right: 25px;

}

.button1:hover span:after {

opacity: 1;

right: 0;

}

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="@Url.Action("Index", "vimsemployee")">Admin DashBoard</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">View As A Customer</a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<h2>Edit</h2>

@using (Html.BeginForm())

{

@Html.AntiForgeryToken()

<div class="form-horizontal">

<h4>vehicleinfo</h4>

<hr />

@Html.ValidationSummary(true)

@Html.HiddenFor(model => model.Vehicle\_ID)

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Name, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Name)

@Html.ValidationMessageFor(model => model.Vehicle\_Name)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicles\_Owner\_Address, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicles\_Owner\_Address)

@Html.ValidationMessageFor(model => model.Vehicles\_Owner\_Address)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Model, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Model)

@Html.ValidationMessageFor(model => model.Vehicle\_Model)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Version, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Version)

@Html.ValidationMessageFor(model => model.Vehicle\_Version)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Rate, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Rate)

@Html.ValidationMessageFor(model => model.Vehicle\_Rate)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Body\_Number, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Body\_Number)

@Html.ValidationMessageFor(model => model.Vehicle\_Body\_Number)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Engine\_Number, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Engine\_Number)

@Html.ValidationMessageFor(model => model.Vehicle\_Engine\_Number)

</div>

</div>

<div class="form-group">

@Html.LabelFor(model => model.Vehicle\_Number, new { @class = "control-label col-md-2" })

<div class="col-md-10">

@Html.EditorFor(model => model.Vehicle\_Number)

@Html.ValidationMessageFor(model => model.Vehicle\_Number)

</div>

</div>

<div class="form-group">

<div class="col-md-offset-2 col-md-10">

<input type="submit" value="Save" class="btn btn-default" />

</div>

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

Delete.cshtml

@model vimseproject.Models.vehicleinfo

@{

ViewBag.Title = "Delete";

}

<style>

.button1 {

border-radius: 4px;

background-color: #f4511e;

border: none;

color: #FFFFFF;

text-align: center;

font-size: 28px;

padding: 20px;

width: 200px;

transition: all 0.5s;

cursor: pointer;

margin: 5px;

}

.button1 span {

cursor: pointer;

display: inline-block;

position: relative;

transition: 0.5s;

}

.button1 span:after {

content: '\00bb';

position: absolute;

opacity: 0;

top: 0;

right: -20px;

transition: 0.5s;

}

.button1:hover span {

padding-right: 25px;

}

.button1:hover span:after {

opacity: 1;

right: 0;

}

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="@Url.Action("Index", "vimsemployee")">Admin DashBoard</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">View As A Customer</a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<h2>Delete</h2>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>vehicleinfo</h4>

<hr />

<dl class="dl-horizontal">

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Name)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Name)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicles\_Owner\_Address)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicles\_Owner\_Address)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Model)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Model)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Version)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Version)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Rate)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Rate)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Body\_Number)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Body\_Number)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Engine\_Number)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Engine\_Number)

</dd>

<dt>

@Html.DisplayNameFor(model => model.Vehicle\_Number)

</dt>

<dd>

@Html.DisplayFor(model => model.Vehicle\_Number)

</dd>

</dl>

@using (Html.BeginForm()) {

@Html.AntiForgeryToken()

<div class="form-actions no-color">

<input type="submit" value="Delete" class="btn btn-default" /> |

@Html.ActionLink("Back to List", "Index")

</div>

}

</div>

Index.cshtml(about page)

@{

ViewBag.Title = "About";

}

<style>

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="@Url.Action("Index", "vimscustomer")"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li><a href="@Url.Action("Contact", "vimscustomer")">Contact</a></li>

<li><a href="@Url.Action("Testimonials", "vimscustomer")">Testimonials</a></li>

<li class="active"><a href="#">About</a></li>

<li><a href="@Url.Action("typesofinsurance", "vimscustomer")">Types of Insurance</a></li>

<li><a href="@Url.Action("ChangePassword", "vimscustomer")"><span class="glyphicon glyphicon-user"></span></a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<div>

<div class="abouttext">

<p class="abouthead">About Car First </p><img img src="~/logo.png"><br>

<p class="about-text">

Car First Insurance offers much more than value-for-money insurance products. When we entered the Global market in 2005, our goal was to protect the quality of people’s lives. We’ve held to that promise, with products that offer real savings and value, staff who provide real service and a corporate culture that inspires real trust. This is confirmed by the number of industry awards we’ve received.

</p>

<center>

<img src="~/images1/carpicture.jpg" class="img-responsive mypicture">

</center>

</div>

Testimonials.cshtml

@{

ViewBag.Title = "Testimonials";

}

<style>

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="@Url.Action("Index", "vimscustomer")"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li><a href="@Url.Action("Contact", "vimscustomer")">Contact</a></li>

<li class="active"><a href="#">Testimonials</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">About</a></li>

<li><a href="@Url.Action("typesofinsurance", "vimscustomer")">Types of Insurance</a></li>

<li><a href="@Url.Action("ChangePassword", "vimscustomer")"><span class="glyphicon glyphicon-user"></span></a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<div>

<div class="abouttext">

<p class="abouthead">Car Insurance Testimonials</p><img img src="~/logo.png"><br>

<p class="about-text">

If you are considering bringing your insurance needs to Car First Insurance, please take some time to meet some of our real customers below, and hear why they are so happy to use Car First Insurance for their insurance needs.

</p>

<hr>

<div class="row">

<div class="col-md-2">

<p class="abouthead">Adam</p>

<img class="img-resonsive" src="~/images1/maletest.jpg">

</div>

<div class="col-md-10">

<p class="about-text">

“I just wanted to thank you guys for your fantastic service regarding the insurance on our car that was damaged during a recent storm. Your staff are professional, polite and well trained. Thanks again for making a difficult situation bearable.”

</p>

</div>

</div>

<hr>

<hr>

<div class="row">

<div class="col-md-2">

<p class="abouthead">Angi</p>

<img class="img-resonsive" src="~/images1/femaletest.jpg">

</div>

<div class="col-md-10">

<p class="about-text">

“We recently had to make a claim and wondered if the bad and ugly would surface. NOT SO! We were absolutely delighted with the way our claim was handled. Initially completing the claim online, we very quickly had a follow up call and emails and the claims personnel really made it a most positive experience.”

</p>

</div>

</div>

<hr>

<hr>

<div class="row">

<div class="col-md-2">

<p class="abouthead">Jhon</p>

<img class="img-resonsive" src="~/images1/maletest.jpg">

</div>

<div class="col-md-10">

<p class="about-text">

“I’ve converted our insurance over to Real Insurance in the last twelve months. As we have four policies, value was important to us. This ended up saving us around $350 on our two cars...I set everything up through the call centre, which was pretty easy and user friendly. I had been nervous to change insurer, but the whole process was simple and straight forward.”

</p>

</div>

</div>

<hr>

</div>

Contact.cshtml

@{

ViewBag.Title = "Contact";

}

<style>

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="@Url.Action("Index", "vimscustomer")"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="#">Contact</a></li>

<li><a href="@Url.Action("Testimonials", "vimscustomer")">Testimonials</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">About</a></li>

<li><a href="@Url.Action("typesofinsurance", "vimscustomer")">Types of Insurance</a></li>

<li><a href="@Url.Action("ChangePassword", "vimscustomer")"><span class="glyphicon glyphicon-user"></span></a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

</div>

<div class="abouttext">

<p class="abouthead">Our Address</p><img src="~/logo.png"><br>

<p class="about-text">

Phone<br>

Car, Home, Landlords: 13 19 48<br>

International enquiries: +61 2 8867 6890<br>

Monday to Friday, 8am to 7pm, AEST<br>

Saturday, 9am to 5pm, AEST<br>

<br>

Bicycle: 13 19 48<br>

Monday to Friday, 8:30am to 5pm, AEST<br>

<br>

VoIP/Skype<br>

Car, Home, Landlords: 02 8889 9499<br>

<br>

Email<br>

Car, Home, Landlords: customercare@realinsurance.com.au<br>

Travel: travelsales@realinsurance.com.au<br>

Bike: bike@realinsurance.com.au<br>

<br>

Mailing address<br>

Car, Home, Contents, Landlords: Locked Bag 2010, St Leonards NSW 1590<br>

Bike: RealBike Claims, PO Box 199, Chatswood NSW 2057<br>

</p>

<br><br>

<div class="controls">

<p class="abouthead">Send Us a Message</p><img src="~/logo.png"><br>

<div class="row">

<div class="col-md-6">

<div class="form-group">

<label for="form\_name">Firstname \*</label>

<input id="form\_name" type="text" name="name" class="form-control" placeholder="Please enter your firstname \*" required="required" data-error="Firstname is required.">

<div class="help-block with-errors"></div>

</div>

</div>

<div class="col-md-6">

<div class="form-group">

<label for="form\_lastname">Lastname \*</label>

<input id="form\_lastname" type="text" name="surname" class="form-control" placeholder="Please enter your lastname \*" required="required" data-error="Lastname is required.">

<div class="help-block with-errors"></div>

</div>

</div>

</div>

<div class="row">

<div class="col-md-6">

<div class="form-group">

<label for="form\_email">Email \*</label>

<input id="form\_email" type="email" name="email" class="form-control" placeholder="Please enter your email \*" required="required" data-error="Valid email is required.">

<div class="help-block with-errors"></div>

</div>

</div>

<div class="col-md-6">

<div class="form-group">

<label for="form\_need">Please specify your need \*</label>

<select id="form\_need" name="need" class="form-control" required="required" data-error="Please specify your need.">

<option value=""></option>

<option value="Request quotation">Request quotation</option>

<option value="Request order status">Request order status</option>

<option value="Request copy of an invoice">Request copy of an invoice</option>

<option value="Other">Other</option>

</select>

<div class="help-block with-errors"></div>

</div>

</div>

</div>

<div class="row">

<div class="col-md-12">

<div class="form-group">

<label for="form\_message">Message \*</label>

<textarea id="form\_message" name="message" class="form-control" placeholder="Message for me \*" rows="4" required="required" data-error="Please, leave us a message."></textarea>

<div class="help-block with-errors"></div>

</div>

</div>

<div class="col-md-12">

<input type="submit" class="btn btn-success btn-send" value="Send message">

</div>

</div>

<div class="row">

<div class="col-md-12">

</div>

</div>

</div>

Typesofinsurance.cshtml

@{

ViewBag.Title = "Types Of Insurance";

}

<style>

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="@Url.Action("Index", "vimscustomer")"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li><a href="contact.html">Contact</a></li>

<li><a href="@Url.Action("Testimonials", "vimscustomer")">Testimonials</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">About</a></li>

<li class="active"><a href ="#">Types of Insurance</a></li>

<li><a href="@Url.Action("ChangePassword", "vimscustomer")"><span class="glyphicon glyphicon-user"></span></a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<div>

<div class="abouttext">

<p class="abouthead">Types of Car Insurance – an Overview</p><img src="~/logo.png"><br>

<p class="about-text">

We tend to spend a lot of time picking out exactly which car we want to buy, but with plenty of drivers having reason to use their motor vehicle insurance, we should also spend some time making sure we’ve got the right type of car insurance!<br>

For example, what if you’ve only got CTP cover but you just rammed someone’s expensive sports car? Had your car stolen – but only insured for third-party property? Perhaps you’re covered for third party, fire and theft … but not flood damage. Without the right type of car insurance, being involved in a car accident can potentially set your finances (and lifestyle) back significantly. So – what are the difference types of motor vehicle insurance?

<br><br>

</p>

<p class="abouthead">

Types of Insurance

</p>

<br><br>

<p class="about-text1">

1. Compulsory Third Party (CTP)

</p>

<p class="about-text">

What does CTP insurance cover? CTP gives you protection against claims for compensation if you injure or kill someone in a motor vehicle accident. The specific conditions vary from state to state, so be sure to check how it works in your state

<br>

What it doesn’t cover: CTP doesn’t insure against the cost of repairs to any vehicles or property – yours or anyone else’s.

<br>

While it is the cheapest and most basic form of cover, keep in mind that having only CTP insurance leaves you at significant financial risk in the event that you damage a vehicle or other property belonging to someone else.

</p>

<br><br>

<p class="about-text1">

2. Third Party Property

</p>

<p class="about-text">

This is the most basic form of optional car insurance, and helps to provide you with financial cover if you cause damage to property rather than a person; for example someone’s prized car, boat or caravan.

<br>

What does third party insurance cover? Third Party Property covers damage caused by your car to other people’s property. It will also cover your legal costs if you become embroiled in legal proceedings as a result of the damage caused.

<br>

What it doesn’t cover: Third party property doesn’t cover the cost of any repairs or replacement of your own car, except under some limited circumstances through some insurers. So it’s similar to CTP in that it only provides cover for other people and their property, rather than you and yours as well.

</p>

<br><br>

<p class="about-text1">

3. Third Party, Fire and Theft

</p>

<p class="about-text">

If you don’t want to pay for top-of-the-line insurance but do want some level of protection for your own car, Third Party, Fire and Theft insurance is the next option.

<br>

What does third party insurance cover? – In addition to covering damage to the property of others, Third Party, Fire and Theft insurance also provides some limited cover for your own car in the event that it’s damaged or lost due to fire or theft.

<br>

What it doesn’t cover: This insurance doesn’t cover the cost of repairs to your vehicle if it is involved in a traffic accident.

</p>

<br><br>

<p class="about-text1">

4. Comprehensive Car Insurance

</p>

<p class="about-text">

This is the top-level insurance cover option and will certainly give you the greatest peace of mind – but it’s also the most expensive!

<br>

What does comprehensive car insurance cover? – It covers everything mentioned in the previous insurance options above, and in addition it covers accidental damage to your own car, irrespective of who is at fault. There are also a range of optional extras, including replacement vehicles and no-excess windscreen replacement. When they say comprehensive, they definitely mean it.

<br>

What it doesn’t cover: Even comprehensive insurance has limitations. It generally won’t cover damage caused by someone else driving your car, unless they are authorised to do so on your policy. It also won’t cover damage caused if the driver was over the legal alcohol limit or affected by drugs. Additionally your car must be roadworthy and you must hold a valid driver’s license.

</p>

<br><br>

<center>

<p class="about-text1">In summary: What’s covered by different types of car insurance</p>

<img src="~/images1/typesofinsurance.png" class="img-responsive mypicture">

</center>

</div>

Index.cshtml(employeecontroller)

@{

ViewBag.Title = "Admin DashBoard";

}

<style>

.button1 {

border-radius: 4px;

background-color: #f4511e;

border: none;

color: #FFFFFF;

text-align: center;

font-size: 28px;

padding: 20px;

width: 200px;

transition: all 0.5s;

cursor: pointer;

margin: 5px;

}

.button1 span {

cursor: pointer;

display: inline-block;

position: relative;

transition: 0.5s;

}

.button1 span:after {

content: '\00bb';

position: absolute;

opacity: 0;

top: 0;

right: -20px;

transition: 0.5s;

}

.button1:hover span {

padding-right: 25px;

}

.button1:hover span:after {

opacity: 1;

right: 0;

}

.navbaricon1 {

margin-top: -15px;

height: 54px;

width: 169px;

}

.nav {

}

.navcontents {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 25px;

}

.portfolioimgs {

margin-right: 60px;

margin-left: 60px;

}

.imgcolumn {

margin-top: 10px;

margin-bottom: 10px;

}

#myBtn {

display: none; /\* Hidden by default \*/

position: fixed; /\* Fixed/sticky position \*/

bottom: 20px; /\* Place the button at the bottom of the page \*/

right: 30px; /\* Place the button 30px from the right \*/

z-index: 99; /\* Make sure it does not overlap \*/

border: none; /\* Remove borders \*/

outline: none; /\* Remove outline \*/

background-color: grey; /\* Set a background color \*/

color: white; /\* Text color \*/

cursor: pointer; /\* Add a mouse pointer on hover \*/

padding: 15px; /\* Some padding \*/

border-radius: 10px; /\* Rounded corners \*/

font-size: 18px; /\* Increase font size \*/

}

#myBtn:hover {

background-color: #555; /\* Add a dark-grey background on hover \*/

}

.abouttext {

margin-right: 30px;

margin-left: 30px;

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.mypicture {

width: 1000px;

}

.mypiccaption {

font-size: 20px;

color: blue;

text-align: right;

}

.contactpageheader {

font-family: 'Josefin Slab', serif;

font-size: 55px;

}

.contactpage {

margin-left: 20px;

margin-right: 20px;

}

.contactpagetext {

font-family: 'Josefin Slab', serif;

font-size: 30px;

}

.abouthead {

font-size: 40px;

}

.about-text {

font-size: 35px;

}

.about-text1 {

font-size: 35px;

text-decoration-style: bold;

}

</style>

<script>

// When the user scrolls down 20px from the top of the document, show the button

window.onscroll = function () { scrollFunction() };

function scrollFunction() {

if (document.body.scrollTop > 20 || document.documentElement.scrollTop > 20) {

document.getElementById("myBtn").style.display = "block";

} else {

document.getElementById("myBtn").style.display = "none";

}

}

// When the user clicks on the button, scroll to the top of the document

function topFunction() {

document.body.scrollTop = 0; // For Safari

document.documentElement.scrollTop = 0; // For Chrome, Firefox, IE and Opera

}

</script>

<nav class="navbar navbar-default navv">

<div class="navcontents">

<div class="container-fluid">

<div class="navbar-header">

<button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">

<span class="icon-bar"></span>

<span class="icon-bar"></span>

<span class="icon-bar"></span>

</button>

<a class="navbar-brand" href="#"><img class="navbaricon1" src="~/logo.png"></a>

</div>

<div class="collapse navbar-collapse" id="myNavbar">

<ul class="nav navbar-nav navbar-right">

<li class="active"><a href="@Url.Action("Index", "vimsemployee")">Admin DashBoard</a></li>

<li><a href="@Url.Action("Index", "vimscustomer")">View As A Customer</a></li>

</i></a>

</ul>

</div>

</div>

</div>

</nav>

<br>

<button onclick="topFunction()" id="myBtn" title="Go to top">Top</button>

<center>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "customerinfo")">

Customer Information

</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "vehicleinfo")">

Vehicle Information

</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "userinfo")">

User Information

</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "billinginfo")">

COMPANY BILLING INFO

</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "estimate")">

Estimate</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "company\_expenses")">

COMPANY EXPENSES

</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "customer\_policy\_records")">

CUSTOMER POLICY RECORDS

</a>

</button>

<br /><br />

<button class="button1">

<a href="@Url.Action("Index", "claim\_datails")">

Claim details

</a>

</button>

<br /><br />

</center>

Index.cshtml(vimsmaincontroller)

@{

ViewBag.Title = "Select Your Post";

}

<style>

body {

margin: 0;

padding: 0;

background-color: #ffcc99;

background-size: cover;

font-family: sans-serif;

}

.loginBox {

margin-top: 70px;

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%,-50%);

width: 350px;

padding: 80px 40px;

box-sizing: border-box;

height: 420px;

background: rgba(0,0,0,.6);

}

.user {

width: 100px;

height: 100px;

border-radius: 50%;

overflow: hidden;

position: absolute;

top: calc(-100px/2);

left: calc(50% - 50px);

}

h2 {

margin: 0;

padding: 0 0 20px;

color: #efed40;

text-align: center;

}

.loginBox p {

margin: 0;

padding: 0;

font-weight: bold;

color: #fff;

}

.loginBox input {

width: 100%;

margin-bottom: 20px;

}

.loginBox input[type="text"],

.loginBox input[type="Password"] {

border: none;

border-bottom: 1px solid #fff;

outline: none;

height: 40px;

background: transparent;

color: #fff;

font-size: 16px;

}

::placeholder {

color: rgba(255,255,255,.5);

}

.loginBox input[type="submit"] {

border: none;

outline: none;

height: 40px;

color: #fff;

font-size: 16px;

background: #ff267e;

cursor: pointer;

border-radius: 20px;

}

.loginBox input[type="submit"]:hover {

background: #efed40;

color: #262626;

}

.loginBox a {

color: #fff;

font-size: 14px;

font-weight: bold;

text-decoration: none;

}

#sign {

font-family: sans-serif;

font-size: 25px;

float: right;

padding: 0px 38px 0px 38px;

background-color: royalblue;

color: yellow;

margin-right: 30px;

margin-top: 20px;

border-top-right-radius: 5px;

border-bottom-left-radius: 5px;

width: 100px;

}

#link {

text-decoration: none;

color: skyblue;

}

</style>

<body>

<center>

<br /><br />

<img src="~/logo.png" />

</center>

<br /><br />

<div class="loginBox">

<h2>Please Select your Post</h2>

<form>

<br /><br /><br />

<center>

<a href="@Url.Action("loginuser", "vimsmain")" class="elements">

I am a Customer

</a>

</center>

<br /><br /><br />

<center>

<a href="@Url.Action("loginemp", "vimsmain")" class="elements">

I am a Employee

</a>

</center>

</form>

</div>

</body>

Loginemp.cshtml

@model vimseproject.Models.userinfo

@{

ViewBag.Title = "LogIn";

}

<style>

body {

margin: 0;

padding: 0;

background-color: #ffcc99;

background-size: cover;

font-family: sans-serif;

}

.loginBox {

margin-top: 70px;

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%,-50%);

width: 350px;

padding: 80px 40px;

box-sizing: border-box;

height: 420px;

background: rgba(0,0,0,.6);

}

.user {

width: 100px;

height: 100px;

border-radius: 50%;

overflow: hidden;

position: absolute;

top: calc(-100px/2);

left: calc(50% - 50px);

}

h2 {

margin: 0;

padding: 0 0 20px;

color: #efed40;

text-align: center;

}

.loginBox p {

margin: 0;

padding: 0;

font-weight: bold;

color: #fff;

}

.loginBox input {

width: 100%;

margin-bottom: 20px;

}

.loginBox input[type="text"],

.loginBox input[type="Password"] {

border: none;

border-bottom: 1px solid #fff;

outline: none;

height: 40px;

background: transparent;

color: #fff;

font-size: 16px;

}

::placeholder {

color: rgba(255,255,255,.5);

}

.loginBox input[type="submit"] {

border: none;

outline: none;

height: 40px;

color: #fff;

font-size: 16px;

background: #ff267e;

cursor: pointer;

border-radius: 20px;

}

.loginBox input[type="submit"]:hover {

background: #efed40;

color: #262626;

}

.loginBox a {

color: #fff;

font-size: 14px;

font-weight: bold;

text-decoration: none;

}

#sign {

font-family: sans-serif;

font-size: 25px;

float: right;

padding: 0px 38px 0px 38px;

background-color: royalblue;

color: yellow;

margin-right: 30px;

margin-top: 20px;

border-top-right-radius: 5px;

border-bottom-left-radius: 5px;

width: 100px;

}

#link {

text-decoration: none;

color: skyblue;

}

</style>

<center><img src="~/logo.png"></center>

<br /><br /><br />

<div class="loginBox">

@using (Html.BeginForm("postemp", "vimsmain", FormMethod.Post)) {

<h2>Employees Log In Here</h2>

<p>Email</p>

@Html.TextBoxFor(model => model.userName, new { @id = "name" })

<p>Password</p>

@Html.TextBoxFor(model => model.userpassword, new { @id = "name" })

<input type="submit" onclick="showdata()" value="Sign In">

<a href="#">Forget Password</a>

<br>

<a href="@Url.Action("Index", "vimsmain")" class="elements">Reset Your Selection</a>

}

</div>

Loginuser.cshtml

@model vimseproject.Models.userinfo

@{

ViewBag.Title = "Login";

}

<style>

body {

margin: 0;

padding: 0;

background-color: #ffcc99;

background-size: cover;

font-family: sans-serif;

}

.loginBox {

margin-top: 70px;

position: absolute;

top: 50%;

left: 50%;

transform: translate(-50%,-50%);

width: 350px;

padding: 80px 40px;

box-sizing: border-box;

height: 420px;

background: rgba(0,0,0,.6);

}

.user {

width: 100px;

height: 100px;

border-radius: 50%;

overflow: hidden;

position: absolute;

top: calc(-100px/2);

left: calc(50% - 50px);

}

h2 {

margin: 0;

padding: 0 0 20px;

color: #efed40;

text-align: center;

}

.loginBox p {

margin: 0;

padding: 0;

font-weight: bold;

color: #fff;

}

.loginBox input {

width: 100%;

margin-bottom: 20px;

}

.loginBox input[type="text"],

.loginBox input[type="Password"] {

border: none;

border-bottom: 1px solid #fff;

outline: none;

height: 40px;

background: transparent;

color: #fff;

font-size: 16px;

}

::placeholder {

color: rgba(255,255,255,.5);

}

.loginBox input[type="submit"] {

border: none;

outline: none;

height: 40px;

color: #fff;

font-size: 16px;

background: #ff267e;

cursor: pointer;

border-radius: 20px;

}

.loginBox input[type="submit"]:hover {

background: #efed40;

color: #262626;

}

.loginBox a {

color: #fff;

font-size: 14px;

font-weight: bold;

text-decoration: none;

}

#sign {

font-family: sans-serif;

font-size: 25px;

float: right;

padding: 0px 38px 0px 38px;

background-color: royalblue;

color: yellow;

margin-right: 30px;

margin-top: 20px;

border-top-right-radius: 5px;

border-bottom-left-radius: 5px;

width: 100px;

}

#link {

text-decoration: none;

color: skyblue;

}

</style>

<center><img src="~/logo.png"></center>

<br /><br /><br /><br /><br /><br />

<div class="loginBox">

@using (Html.BeginForm("postuser", "vimsmain", FormMethod.Post)) {

<h2>Customers Log In Here</h2>

<p>Email</p>

@Html.TextBoxFor(model => model.userName, new { @id = "name" })

<p>Password</p>

@Html.TextBoxFor(model => model.userpassword, new { @id = "name" })

<input type="submit" onclick="showdata()" value="Sign In">

<a href="#">Forget Password</a>

<br>

<a href="@Url.Action("Index", "vimsmain")" class="elements">Reset Your Selection</a>

}

</div>

**USER GUIDE**

* DEVICES:

Desktop PC.

* OPERATING SYSTEM:

Windows.

* SPACE:

74.3 MB (78,004,224 bytes)

**DEVELOPERS GUIDE**

* REQUIREMENTS:
* Operating System: Windows.
* Tools: Microsoft Visual Studios.
* Technologies: C#.
* STEPS TO OPEN:

1. Install Visual Studios.
2. Extract the file with name ‘EPROJECT(2019)VIMS’.
3. Open Extracted Files
4. First execute the SQL query file which is named ‘execute this first’.
5. Then open the folder name ‘vimseproject’.
6. Then open the Visual Studio File named ‘vimseproject’.
7. When the solution/project is loaded preview run(start) program.

* MODULE DISCRIPTION:

This contains the detail description of module with comments so that the developer can understand the code.

Following is the coding to respective pages.

THE END