**Home-Service** Project Report

SUKHPREET SINGH



About Project:

This project is mainly developed for the keeping record of Home Service located at different locations. The project entitled “Home Service in Asp.Net” set of platform for the project leaders and administrator to send messages to keep constant interaction with the programmers. In the proposed system is the administrator can sit at his system and communicate with the project leaders, programmers and can also maintain the login time, logout time. He can chat about the project going on with both the programmers and project leaders. This project also maintains offline messages. The administrator sees the IP address of the system of project leader/programmer he is interested and chat with the people .The project leader gives the instant report about the project to the administrator. Through this Communication System, Whenever user logs-in into the system then login time is recorded in the administrator’s system automatically. This project makes the most of the tedious and mundane deskwork to minimize. This facilitates the user to concentrate more on the project and less on the managing of these tasks. The software is fully integrated with CRM (Customer Relationship Management) as well as CMS (Content Management System) solution and developed in a manner that is easily manageable, time saving and relieving one from manual works.

About Asp-mvc project

For my project, we want to deploy a demo server and publish my ASP.NET MVC 5 app to it. I'm familiar with developing applications in this environment, but I have no experience in deploying it to an actual server. Our sys admin does not have extensive experience with ASP.NET applications or IIS, but is reasonably well-versed in Windows Server.

Now, I can't really find what system requirements there are for the server - only for some older versions of ASP.NET MVC (e.g. version 2 or 3).

I am not asking in terms of capacity - I need to know the actual software version requirements:

What Windows Server version is the minimum?

Which IIS version

Any other software we'll need?

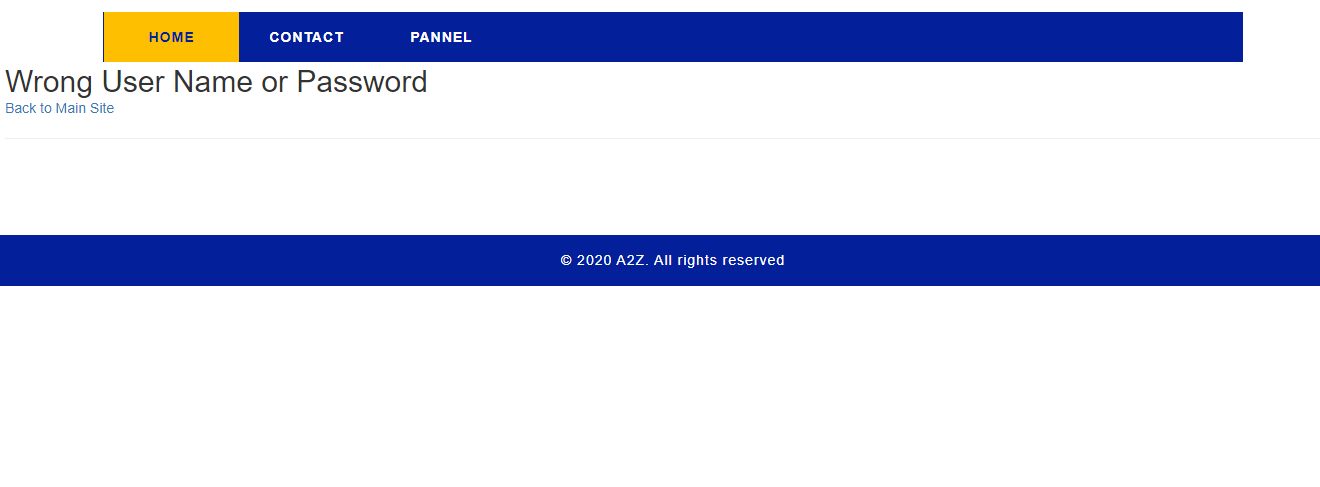
Admin Duties

After logging into admin panel admin can add new staff, customer details, customer service and view the contact information.

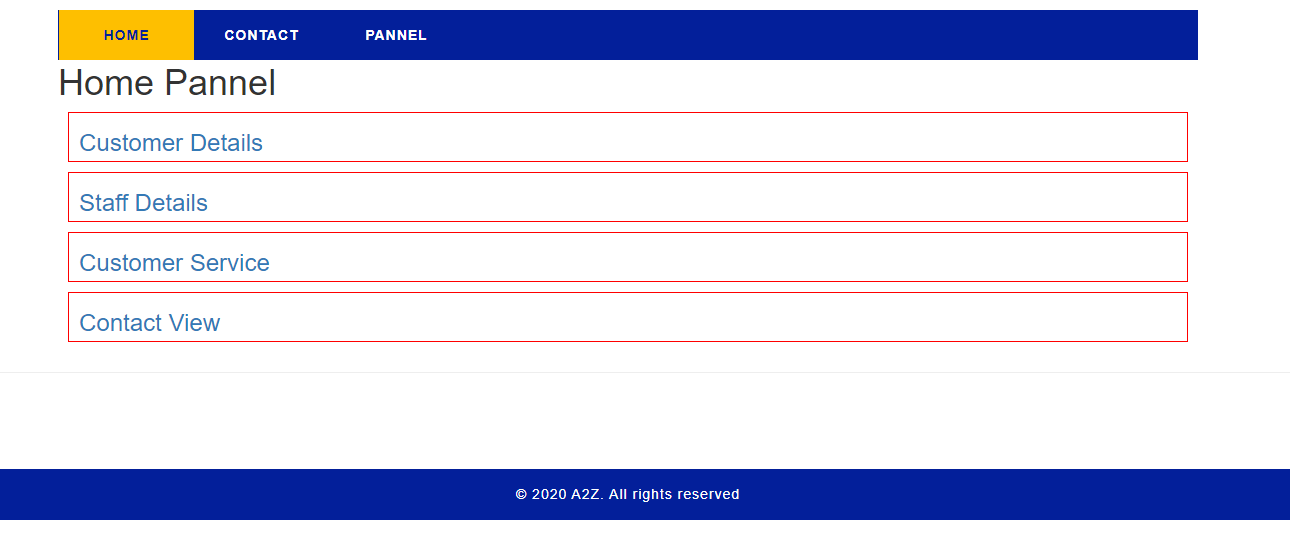
Below are given login details of the admin Username Password

|  |  |
| --- | --- |
| Username | Password |
| Admin | Admin |

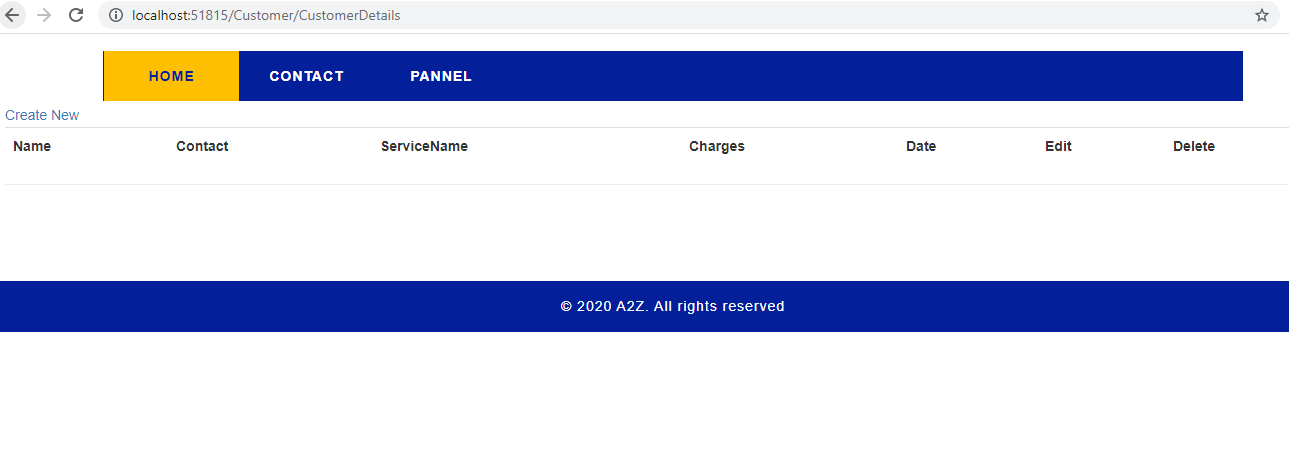


If someone enter wrong username and detail

After successfully logging in to the account



Customer Detail Page



Adding, updating, deleting coding below

// POST: Customer/Create

[HttpPost]

public ActionResult Create([Bind(Exclude = "id")] Customer addDetails)

{

if (!ModelState.IsValid)

return View();

instance\_obj.Customers.Add(addDetails);

instance\_obj.SaveChanges();

//Response.Redirect("StudentAdmission",true);

return RedirectToAction("CustomerDetails");

}

// GET: Customer/Edit/5

public ActionResult Edit(int id)

{

var IdToEdit = (from m in instance\_obj.Customers where m.id == id select m).First();

return View(IdToEdit);

}

// POST: Customer/Edit/5

[HttpPost]

public ActionResult Edit(Customer IdToEdit)

{

var orignalRecord = (from m in instance\_obj.Customers where m.id == IdToEdit.id select m).First();

if (!ModelState.IsValid)

return View(orignalRecord);

instance\_obj.Entry(orignalRecord).CurrentValues.SetValues(IdToEdit);

instance\_obj.SaveChanges();

return RedirectToAction("CustomerDetails");

}

// GET: Customer/Delete/5

public ActionResult Delete(Customer IdToDelete)

{

var d = instance\_obj.Customers.Where(x => x.id == IdToDelete.id).FirstOrDefault();

instance\_obj.Customers.Remove(d);

instance\_obj.SaveChanges();

return RedirectToAction("CustomerDetails");

}

// POST: Customer/Delete/5

[HttpPost]

public ActionResult Delete(int id, FormCollection collection)

{

try

{

// TODO: Add delete logic here

return RedirectToAction("Index");

}

catch

{

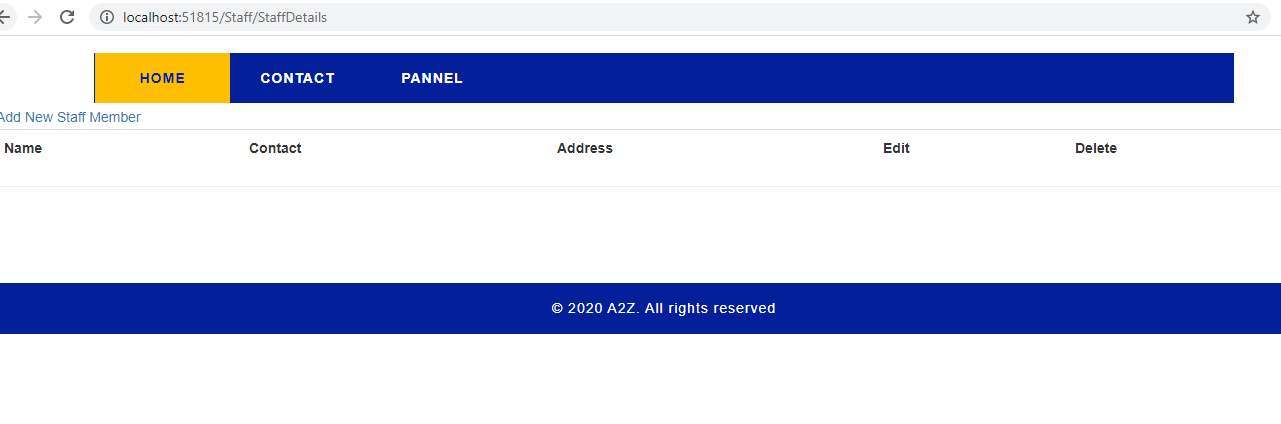
return View();

}

}

}

Add New Staff



Adding, updating and deleting staff members

public ActionResult Create()

{

return View();

}

// POST: Staff/Create

[HttpPost]

public ActionResult Create([Bind(Exclude = "id")] Staff addDetails)

{

if (!ModelState.IsValid)

return View();

instance\_obj.Staffs.Add(addDetails);

instance\_obj.SaveChanges();

return RedirectToAction("StaffDetails");

}

// GET: Staff/Edit/5

public ActionResult Edit(int id)

{

var IdToEdit = (from m in instance\_obj.Staffs where m.id == id select m).First();

return View(IdToEdit);

}

// POST: Staff/Edit/5

[HttpPost]

public ActionResult Edit(Staff IdToEdit)

{

var orignalRecord = (from m in instance\_obj.Staffs where m.id == IdToEdit.id select m).First();

if (!ModelState.IsValid)

return View(orignalRecord);

instance\_obj.Entry(orignalRecord).CurrentValues.SetValues(IdToEdit);

instance\_obj.SaveChanges();

return RedirectToAction("StaffDetails");

}

// GET: Staff/Delete/5

public ActionResult Delete(Staff IdToDelete)

{

var d = instance\_obj.Staffs.Where(x => x.id == IdToDelete.id).FirstOrDefault();

instance\_obj.Staffs.Remove(d);

instance\_obj.SaveChanges();

return RedirectToAction("StaffDetails");

}

// POST: Staff/Delete/5

[HttpPost]

public ActionResult Delete(int id, FormCollection collection)

{

try

{

// TODO: Add delete logic here

return RedirectToAction("Index");

}

catch

{

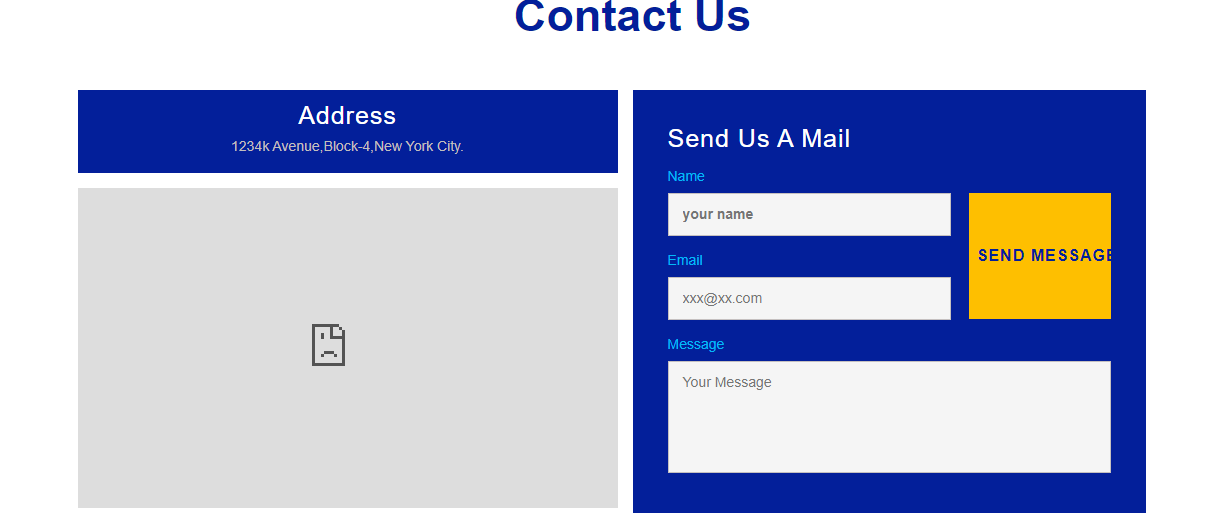
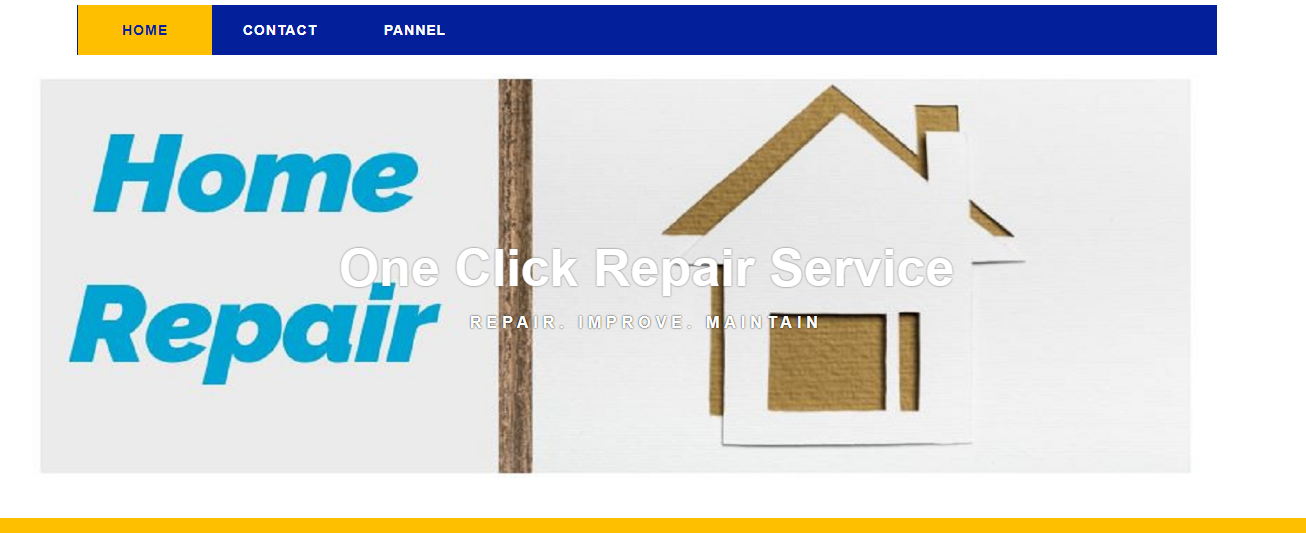
return View();

}

}

}

Contact page



Index Page:

