USE [pubs]

GO

/\*\*\*\*\*\* Object: Table [dbo].[url\_tbl] Script Date: 2/4/2020 12:05:31 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[url\_tbl](

[url\_id] [int] NOT NULL,

[Url] [varchar](100) NOT NULL,

[category] [int] NULL,

PRIMARY KEY CLUSTERED

(

[url\_id] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

ALTER TABLE [dbo].[url\_tbl] WITH CHECK ADD FOREIGN KEY([category])

REFERENCES [dbo].[Category\_Tbl] ([Category\_ID])

GO

## **Disable Lazy Loading**

We can disable lazy loading for a particular entity or a context. To turn off lazy loading for a particular property, do not make it virtual. To turn off lazy loading for all entities in the context, set its configuration property to false.

using System;

using System.Data.Entity;

using System.Data.Entity.Infrastructure;

using System.Data.Entity.Core.Objects;

using System.Linq;

public partial class SchoolDBEntities : DbContext

{

public SchoolDBEntities(): base("name=SchoolDBEntities")

{

this.Configuration.LazyLoadingEnabled = false;

}

protected override void OnModelCreating(DbModelBuilder modelBuilder)

{

}

}

CREATE TABLE [dbo].[Url\_tbl](

[Url\_ID] [int] IDENTITY(1,1) NOT NULL,

[Url] [varchar](100) NOT NULL,

[Category] [int] NULL,

PRIMARY KEY CLUSTERED

(

[Url\_ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

ALTER TABLE [dbo].[Url\_tbl] WITH CHECK ADD FOREIGN KEY([Category])

REFERENCES [dbo].[Category\_Tbl] ([Category\_ID])

GO

USE [pubs]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Url\_tbl] Script Date: 2/3/2020 10:22:02 AM \*\*\*\*\*\*/

set ANSI\_NULLS ON

set QUOTED\_IDENTIFIER ON

SET IDENTITY\_INSERT url\_tbl ON

GO

|  |  |
| --- | --- |
|  | **Cognizant Academy**  **URL-CATEGORY-ICT**  **DOTNET Integrated Case Study**  **ASP.NET MVC** |
| |  |  |  |  | | --- | --- | --- | --- | |  | **Prepared By / Last Updated By** | **Reviewed By** | **Approved By** | | **Name** | Venkateshwari Devi |  |  | | **Role** |  |  |  | | **Signature** |  |  |  | | **Date** |  |  |  | |
|  |

Table of Contents

[1.0 Introduction 3](#_Toc23440118)

[1.1 Purpose of this document 3](#_Toc23440119)

[1.2 Definitions & Acronyms 3](#_Toc23440120)

[1.3 Project Overview 3](#_Toc23440121)

[1.4 Scope 3](#_Toc23440122)

[1.5 Target Audience 3](#_Toc23440123)

[1.6 Hardware and Software Requirement 3](#_Toc23440124)

[1.6.1 Hardware Requirements 3](#_Toc23440125)

[1.6.2 Software Requirements 4](#_Toc23440126)

[2.0 Functional Requirements 4](#_Toc23440127)

[2.1 Functional Requirements 4](#_Toc23440128)

[2.2 Use case Diagram 4](#_Toc23440129)

[2.3 System Architecture Diagram 4](#_Toc23440130)

[3.0 Design Specification 5](#_Toc23440131)

[3.1 Pre-requisite Steps 5](#_Toc23440132)

[3.2 UI Design 5](#_Toc23440133)

[3.3 Data Design 5](#_Toc23440134)

[3.4 Component Design for identified Use cases 5](#_Toc23440135)

[3.5 General Design Constraints 6](#_Toc23440136)

[3.6 Requirements to Design Traceability 6](#_Toc23440137)

[4.0 Submission 6](#_Toc23440138)

[4.1 Code submission instructions 6](#_Toc23440139)

[5.0 Change Log 6](#_Toc23440140)

# Introduction

## Purpose of this document

The purpose of the Document is to elaborate the functional requirements and technical design specification of the problem statement to be solved.

## Definitions & Acronyms

|  |  |
| --- | --- |
| Definition / Acronym | Description |
|  |  |
|  |  |
|  |  |
|  |  |

## Project Overview

Browser allowed to as user or Admin . user able to Add Url for a type of category.

Admin able to add Category , for which user is able add URL’s.

## Scope

Based on ASP.NET MVC ,EntityFramework database first Model.

## Target Audience

Beginners

## Hardware and Software Requirement

### Hardware Requirements

|  |  |  |
| --- | --- | --- |
| # | Item | Specification/Version |
|  | Processor | Intel® coreTM i5 or higher |
|  | RAM | 8 GB |
|  | HARD DISK | OS partitation 200GB |

### Software Requirements

|  |  |  |
| --- | --- | --- |
| # | Item | Specification/Version |
|  | Operating System | Windows 10 |
|  | Application Software | Visual Studio2017 or higher |
|  | Database Server | MS Sql Server 2014 Expression Edition |

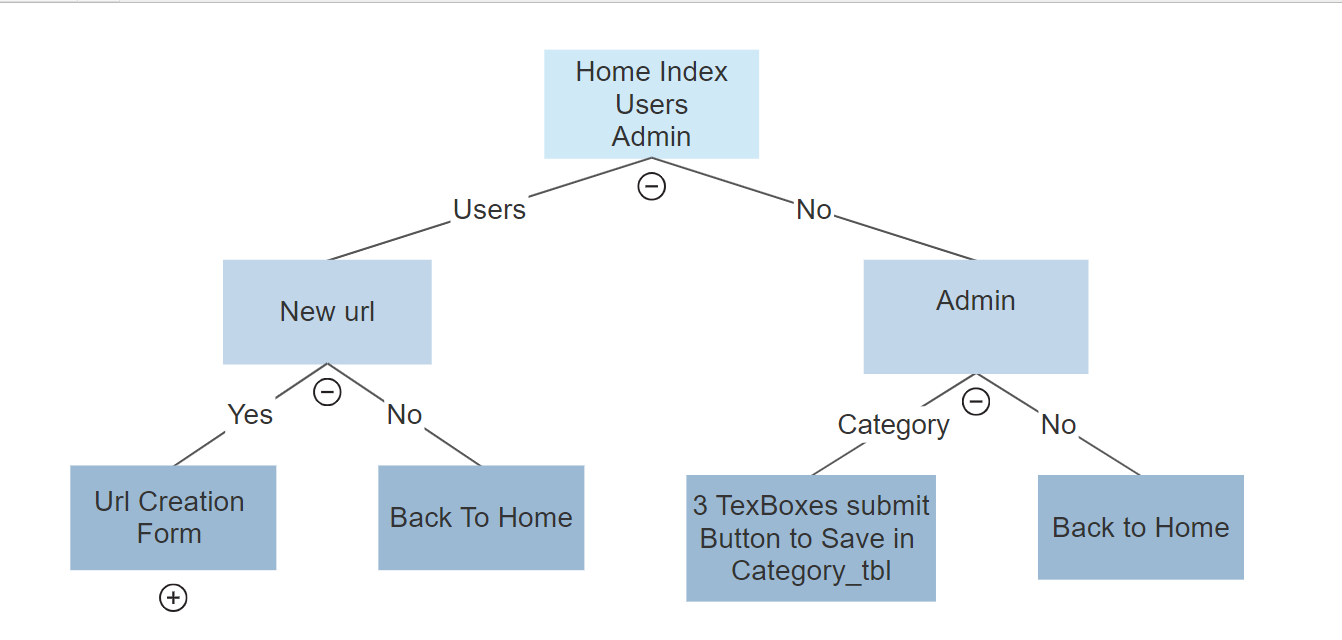
# Functional Requirements

## Functional Requirements

*<List the high level use cases or behaviors and actors of the system. This requirement # and name must be unique and must be referred elsewhere in the document. >*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Req. # | Req. Name | Req. Description | Actors / Users | Comments |
| <sample\_1> | Home Page | With Users, or Admin | Users  Admin | Normal Web page No Login Details required |
| USERS | Index page |  | Users | Page to add URL |
| ADMINS | Index page |  | ADMIN | Page to display existing list of categories |
| ADMINS | Create page |  | ADMIN | Add new Category |

## Use case Diagram



## System Architecture Diagram

* Understanding Requirements and Database Design.
* Creating Solution and Adding projects to It.
* Creating The business Objects.
* Creating Data Access Layer in EF.
* Creating Business Logic Layer in C#.Net.
* Creating Presentation Logic Layer in MVC5
* Designing Models, Controllers and Views.
* Business Rules Validations.
* Securing Your App.

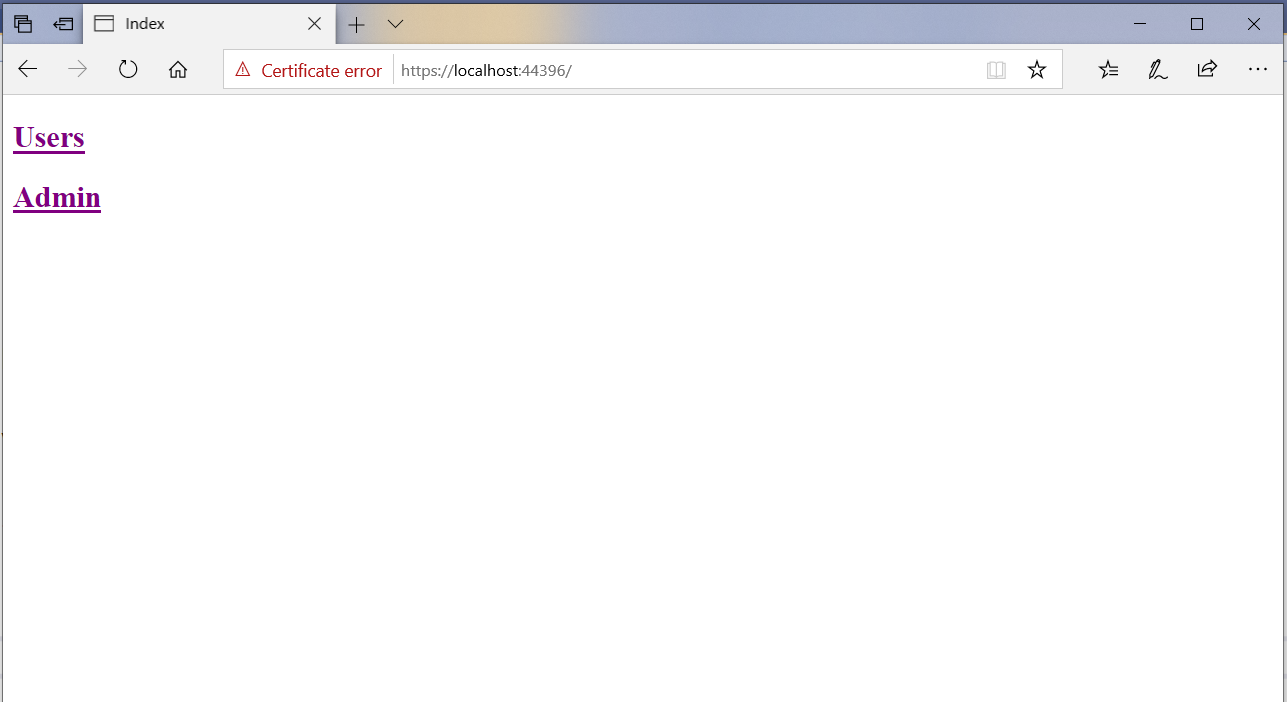
# Design Specification

## Pre-requisite Steps

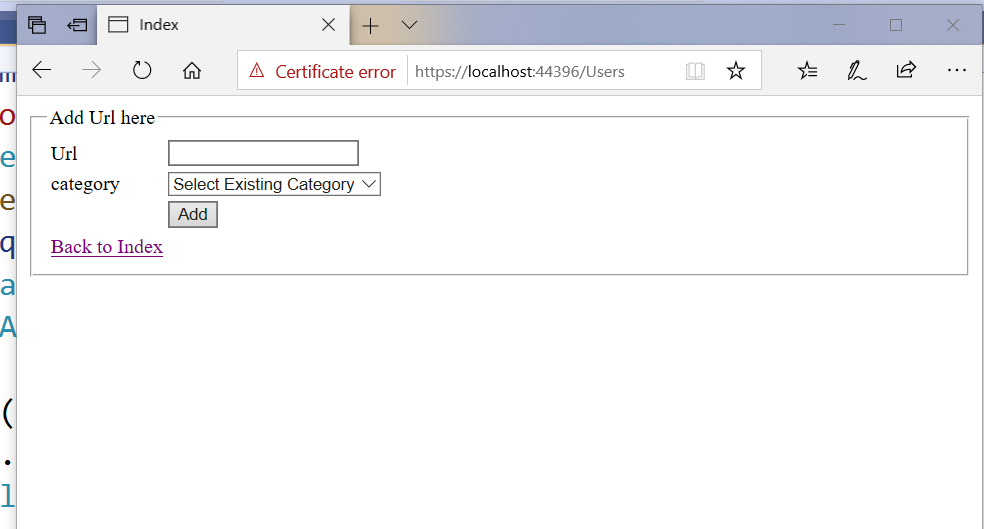
Link hub is a web portal where a user can submit their portal URL under a specific category to be shared on link hub. Admin can Add new category or Delete,Edit the URL submitted by the Admin. It will be available on the link hub portal under a specific category.

## UI Design

1. Home-index



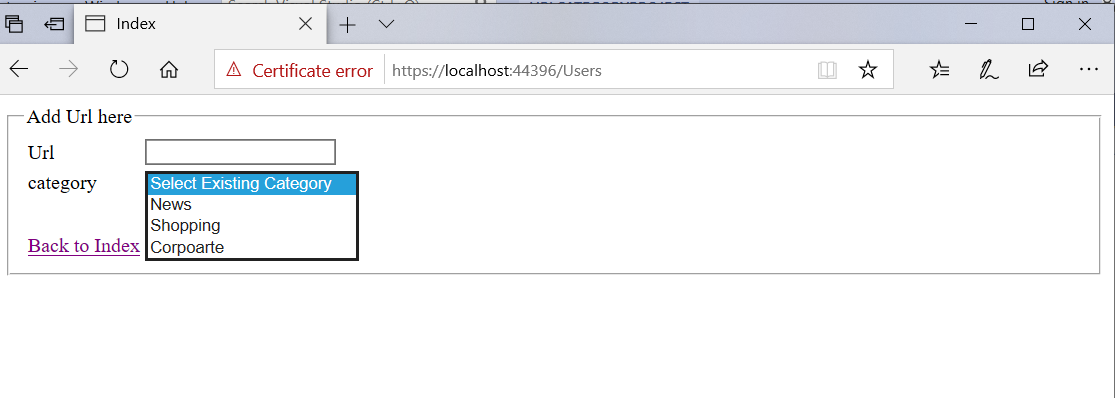
2)Users-index



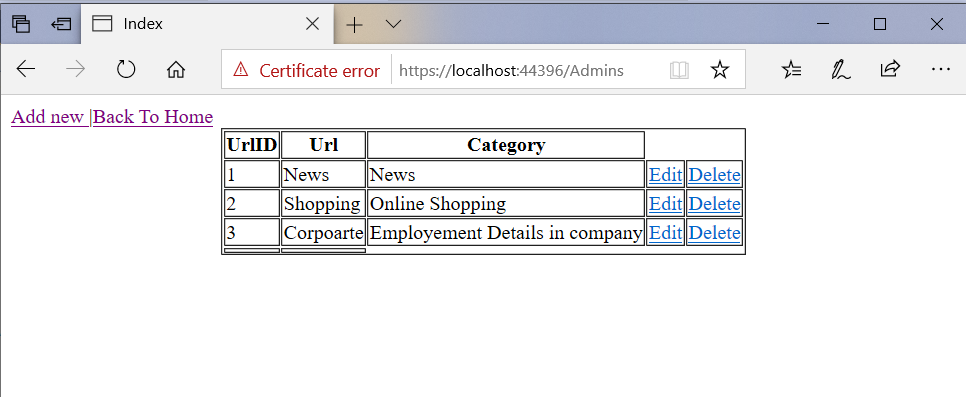
1. UsersController’s Index View should display existing categories column values from “Category.tbl”

In DropDownList such that user’s allowed to select existing Categories. Which is depicted in Below Screen.

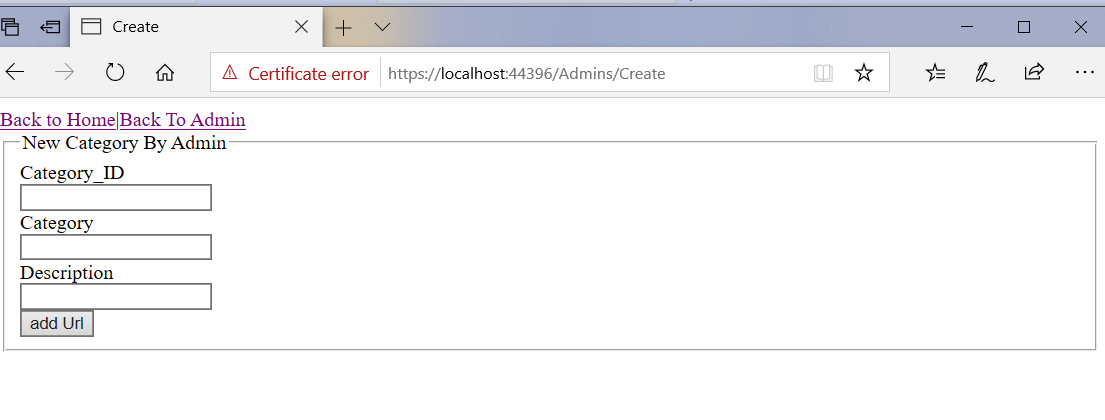
User is also allowed to browser back to Home page.



3)Admins-Index



🡪Admin – Create category

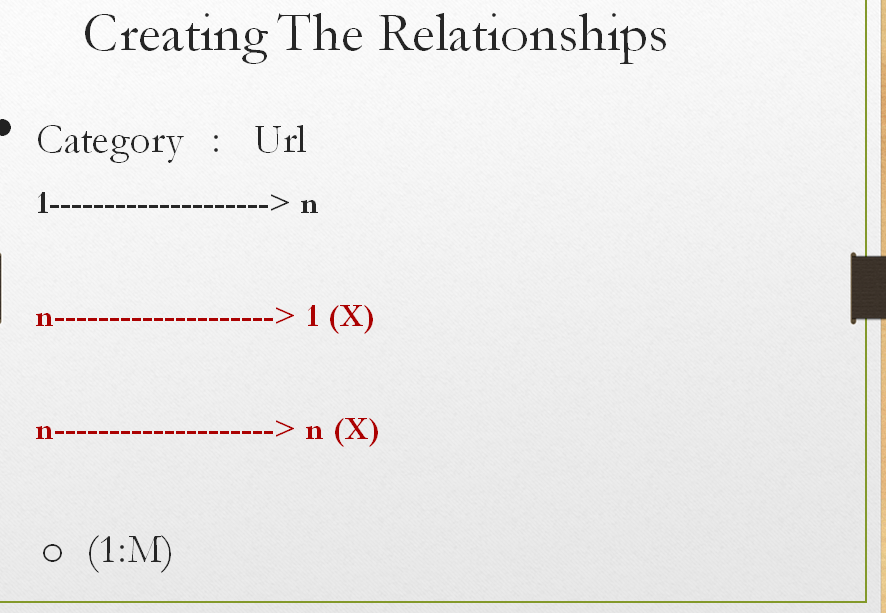


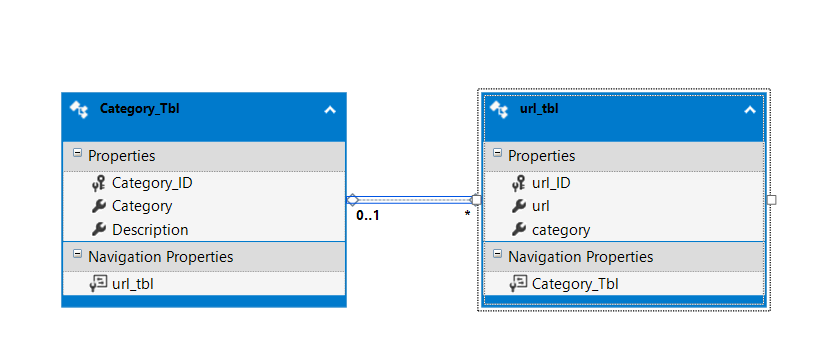
## Data Design

Identifying The Objects(Tables)

1)Url\_tbl

2)Category\_tbl





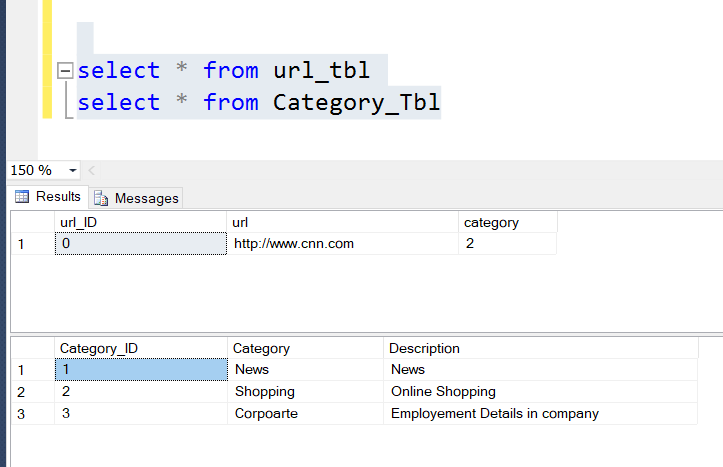
🡪Implementating Database

url\_tbl

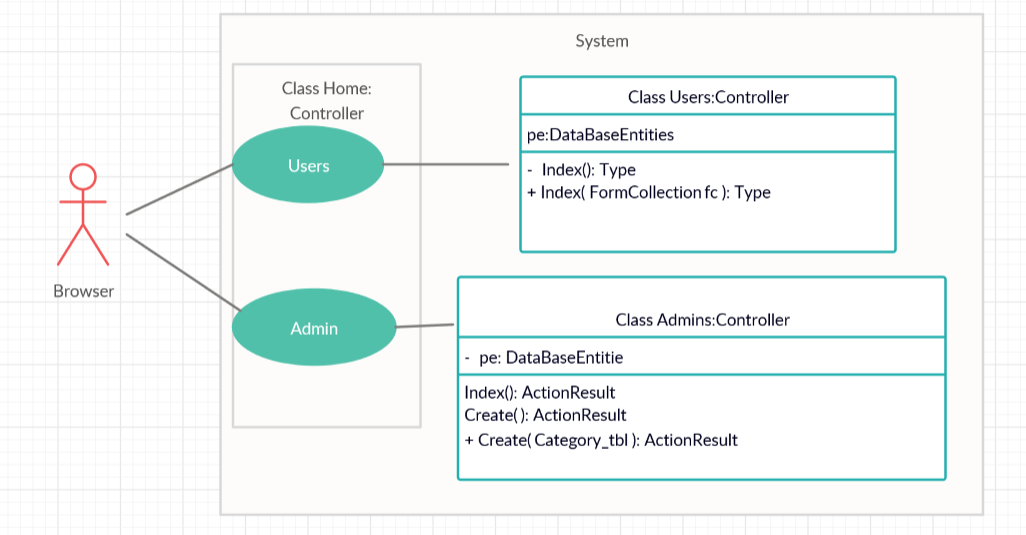
|  |  |
| --- | --- |
| Column | Type |
| url\_ID | Int ,Primary Key,Identity Column |
| url | Varchar(100) |
| Category | Int references Category\_tbl Category\_ID |

1. Category\_Tbl

|  |  |
| --- | --- |
| Column | Type |
| Category\_ID | Int ,Primary Key,Identity Column |
| Category | Varchar(100) not null |
| Description | Varchar(100) not null |

Insert few records into tables:

## Component Design for identified Use cases



<Repeat this table for all identified components>

|  |  |
| --- | --- |
| *Component Name* | *HomeController* |
| *Component Definition* | *Browser will access Index View of Home Controller* |
| *Type* | *Controller* |
| *Responsibilities* | *Will have users,Admin HyperLinks* |
| *Constraints* | *No Need of any Authentication Browser able to select any of the link.* |
| *Composition* | *Users Link will open a page where url ,Category are assigned and saved to Database Table “Url\_tbl” or back to Home Index View*  *Admin Link open a web page with display existing categories in a gridview Format. Or Back to Home Index View.* |
| *Interactions* |  |
| *Resources* | *UsersController will save browser’s input in “url\_tbl” table.*  *AdminController will save browser’s input in “Category\_tbl”* |
| *Processing* | 1. *Url\_tbl data comes from UsersController ,Index Method which has “FormCollection” Argument.* 2. *Category\_tbl data comes from AdminsController ‘s Create method which has “Categort\_tbl ” table as argument object of the table will save browser’s input into corresponding table.* |
| *Exceptional Conditions* | 1. *Url\_tbl , Category\_tbl columns defined as not null . so browser’s input must fecilitate the conditions..* 2. *UsersController ‘s index method should display “category” column values from “Category\_tbl”*   *In Index View’s DropDownList , such that browser is allowed to select existing Categories from DropDownList.*   1. *UsersController has another index method with “FormCollection” argument defined to save browser’s input into “Url\_tbl” table* |

## General Design Constraints

UsersController Index method to be defined with code first model to access “Category” Column Values from “Category\_tbl” table .

Index View has 2 fields ,textbox for “url”, DropDownList” for Category column Values .

These values to be defined as “Viewbag” object to be displayed in Index View.

## Requirements to Design Traceability

# Submission

## Code submission instructions

# Change Log

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Changes Made | | | |
| V1.0.0 | Initial baseline created on 26-01-2020 by k venkateshwari devi | | | |
| Vx.y.z | <Please refer the configuration control tool / change item status form if the details of changes are maintained separately. If not, the template given below needs to be followed> | | | |
| **Section No.** | **Changed By** | **Effective Date** | **Changes Effected** |
|  |  |  |  |

USE [pubs]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Category\_Tbl] Script Date: 1/26/2020 12:44:11 PM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Category\_Tbl](

[Category\_ID] [int] NOT NULL,

[Category] [char](25) NOT NULL,

[Description] [varchar](100) NOT NULL,

CONSTRAINT [PK\_Category\_Tbl] PRIMARY KEY CLUSTERED

(

[Category\_ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

SET ANSI\_PADDING OFF

GO

USE [pubs]

GO

/\*\*\*\*\*\* Object: Table [dbo].[Url\_tbl] Script Date: 1/27/2020 11:33:01 AM \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

SET ANSI\_PADDING ON

GO

CREATE TABLE [dbo].[Url\_tbl](

[Url\_ID] [int] IDENTITY(1,1) NOT NULL,

[Url] [varchar](100) NOT NULL,

[Category] [int] NULL,

PRIMARY KEY CLUSTERED

(

[Url\_ID] ASC

)WITH (PAD\_INDEX = OFF, STATISTICS\_NORECOMPUTE = OFF, IGNORE\_DUP\_KEY = OFF, ALLOW\_ROW\_LOCKS = ON, ALLOW\_PAGE\_LOCKS = ON) ON [PRIMARY]

) ON [PRIMARY]

GO

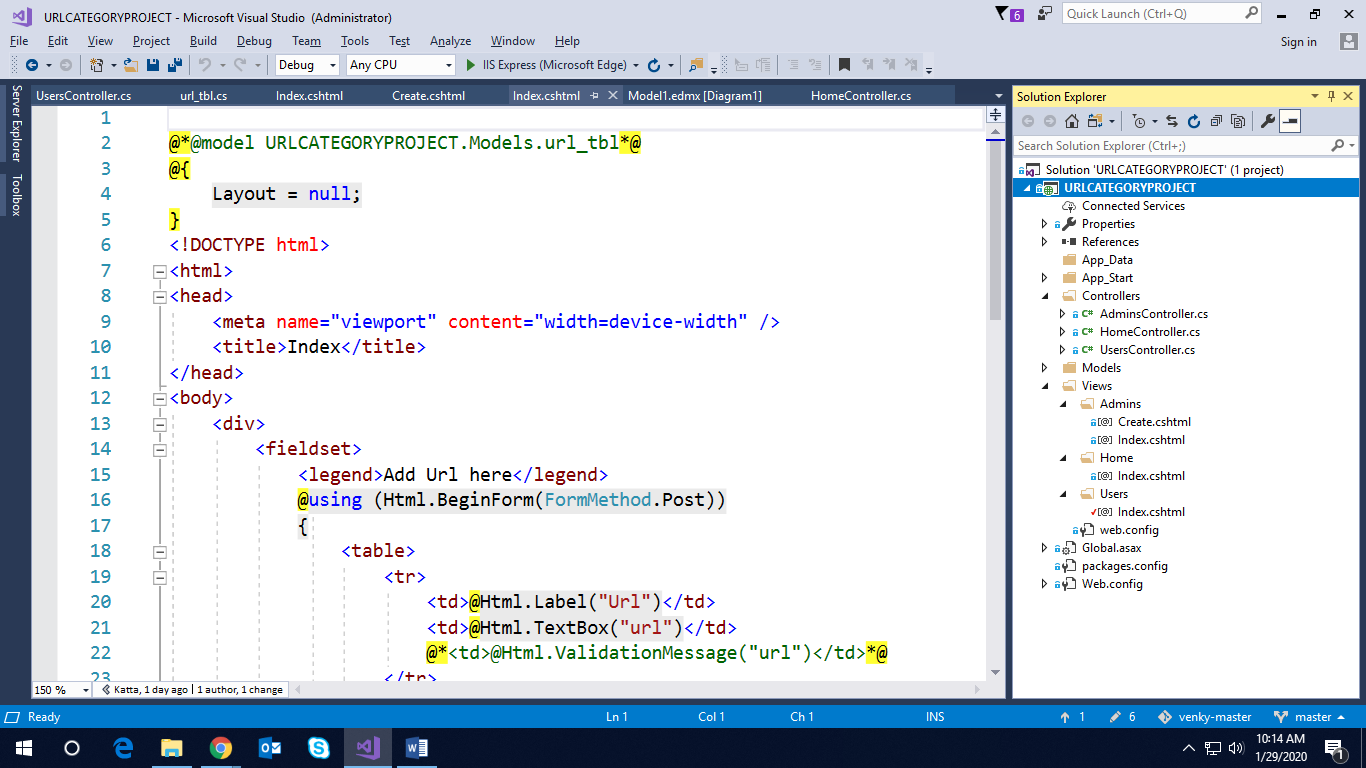
SET ANSI\_PADDING OFF

GO

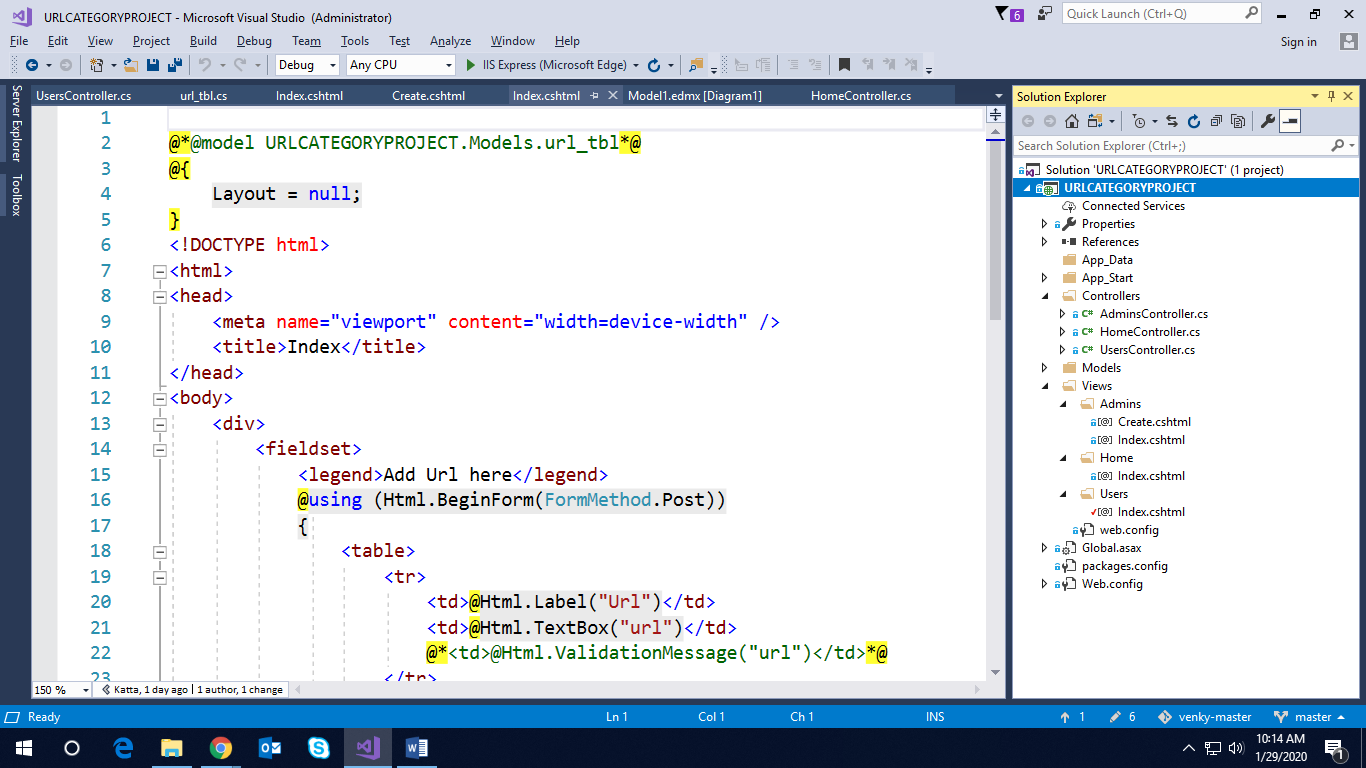
ALTER TABLE [dbo].[Url\_tbl] WITH CHECK ADD FOREIGN KEY([Category])

REFERENCES [dbo].[Category\_Tbl] ([Category\_ID])

GO



HOME:INDEX.Cshtml



@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

<title>Index</title>

</head>

<body>

<div>

<h2>@Html.ActionLink("Users","Index","Users")</h2>

<h2>@Html.ActionLink("Admin","Index","Admins")</h2>

</div>

</body>

</html>

AdminsController : Index.cshtml

@model List<URLCATEGORYPROJECT.Models.Category\_Tbl>

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

<title>Index</title>

</head>

<body>

<div style="align-items:center;align-content:center">

@Html.ActionLink("Add new ", "Create")|@Html.ActionLink("Back To Home", "Index", "Home")

</div>

<div> <table border="1" cellpadding="1" cellspacing="1" align="center">

<tr><th>UrlID</th><th>Url</th><th>Category</th></tr>

@foreach (var item in Model)

{

<tr>

<td>@item.Category\_ID</td>

<td>@item.Category</td>

<td>@item.Description</td>

<td>@Html.ActionLink("Edit","Update",new {id=item.Category\_ID })</td>

<td>@Html.ActionLink("Delete","Delete",new { id=item.Category\_ID})</td>

</tr>

}

<tr><td></td><td></td></tr>

</table>

</div>

</body>

</html>

AdminsController: Create.cshtml

@model URLCATEGORYPROJECT.Models.Category\_Tbl

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

<title>Create</title>

</head>

<body>

<div> <div><div><div><div>@Html.ActionLink("Back to Home","Index","Home")|@Html.ActionLink("Back To Admin","Index","Admins")</div></div></div></div>

<fieldset>

<legend>New Category By Admin</legend>

@using (Html.BeginForm())

{

<div>@Html.LabelFor(x => x.Category\_ID)</div>

<div>@Html.TextBoxFor(x => x.Category\_ID)</div>

<div>@Html.ValidationMessageFor(x => x.Category\_ID)</div>

<div>@Html.LabelFor(x => x.Category)</div>

<div>@Html.TextBoxFor(x => x.Category)</div>

<div>@Html.ValidationMessageFor(x => x.Category)</div>

<div>@Html.LabelFor(x => x.Description)</div>

<div>@Html.TextBoxFor(x => x.Description)</div>

<div>@Html.ValidationMessageFor(x => x.Description)</div>

<div></div>

<div></div>

<div></div>

<div><input type="submit" value="add Url" /></div>

}

</fieldset>

</div>

</body>

</html>

UsersController:Index.cshtml

@\*@model URLCATEGORYPROJECT.Models.url\_tbl\*@

@{

Layout = null;

}

<!DOCTYPE html>

<html>

<head>

<meta name="viewport" content="width=device-width" />

<title>Index</title>

</head>

<body>

<div>

<fieldset>

<legend>Add Url here</legend>

@using (Html.BeginForm(FormMethod.Post))

{

<table>

<tr>

<td>@Html.Label("Url")</td>

<td>@Html.TextBox("url")</td>

@\*<td>@Html.ValidationMessage("url")</td>\*@

</tr>

<tr>

<td>@Html.Label("category")</td>

<td>@Html.DropDownList("cate", "Select Existing Category ")</td>

</tr>

<tr><td></td><td><input type="submit" value="Add" /></td></tr>

<tr></tr>

<tr><td>@Html.ActionLink("Back to Index", "Index", "Home")</td></tr>

</table>

}

</fieldset>

</div>

<div>

</div>

</body>

</html>