

**INDIRA GANDHI NATIONAL OPEN UNIVERSITY**

**MCSP-060: PROJECT REPORT**

**DIGITAL LEARNING APP**

by

**AKSHAY KUMAR MISHRA**

**Enrollment No: 159858172**

Under Guidance

Of

**AMIT SINGH**

Submitted to the School of Computer and Information Sciences, IGNOU

In partial fulfillment of the requirements for the award of the degree

**MASTERS of COMPUTER APPLICATIONS (MCA)**

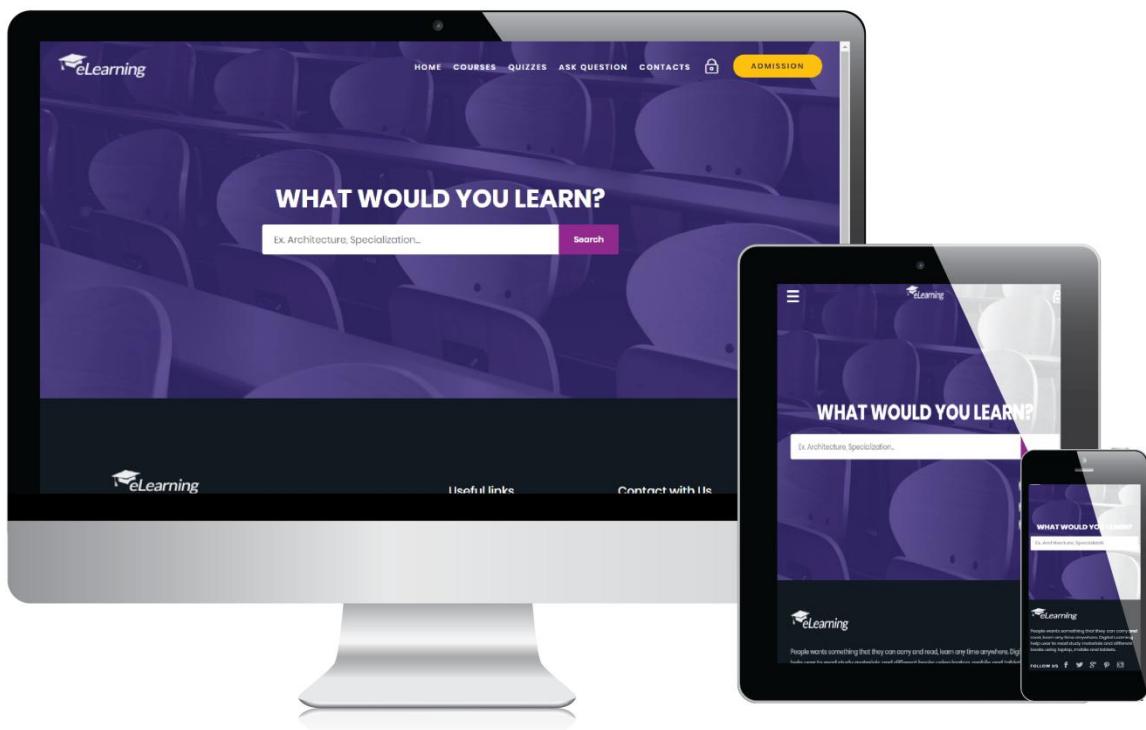
**2017**



**Indira Gandhi National Open University**

**MAIDAN GARHI, NEW DELHI-110068**

# DIGITAL LEARNING APPLICATION

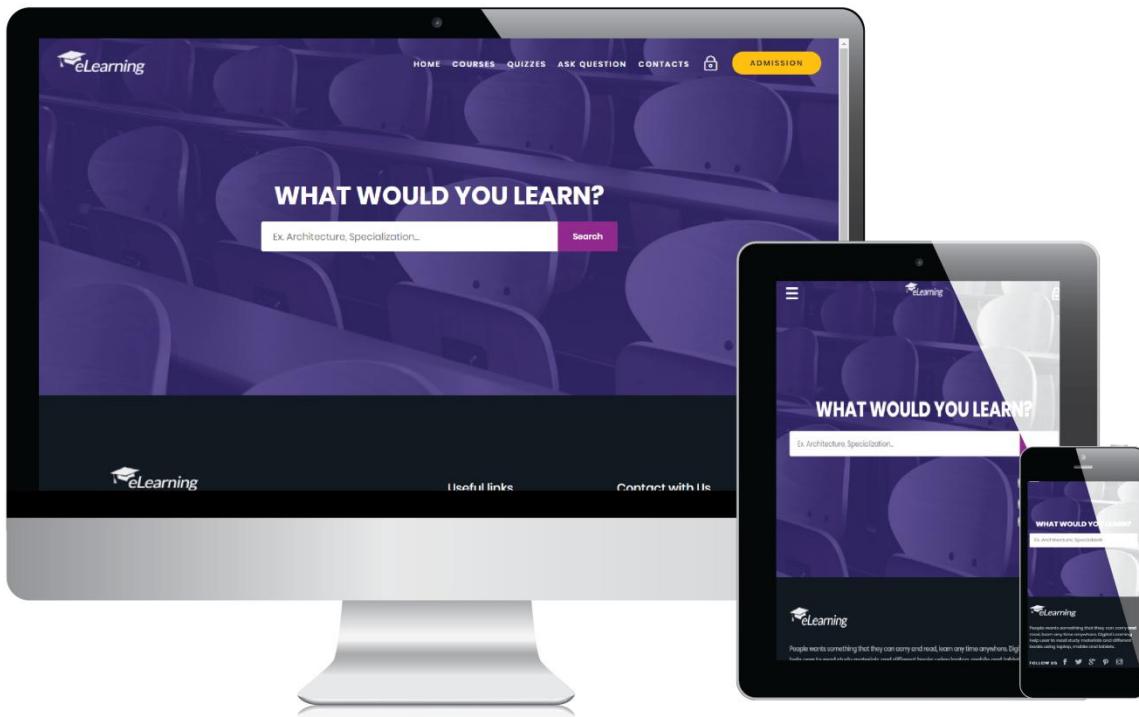


# CONTENTS

Digital

Learning

Application



.....	2
1. Overview .....	10
1.1 Introduction.....	10
1.2 Objectives.....	10
2. Survey of Technologies .....	11
2.1 Project Category.....	11
2.1.1 Front-End with ASP.NET MVC.....	11
2.1.2 Database (MS-SQL Server).....	11
2.2 Software and Hardware Requirements .....	12
3. System ANALYSIS .....	14



3.1 Identification of need .....	14
3.2 High level system requirement: .....	14
3.3 Requirement Specification .....	14
3.4 Project Planning and Scheduling .....	15
3.4.1 Pert Chart.....	15
3.4.2 Gantt Chart .....	17
Completed Tasks .....	18
3.6 Preliminary Product Descriptions .....	18
3.6.1 Module details .....	18
3.7 Object Oriented Models.....	20
3.7.1 Data Flow Diagram .....	20
3.7.2 ER Diagram .....	20
3.7.3 Use Case Diagram.....	22
3.7.4 Class Diagram (Updated) .....	23
Table Defintions .....	26
Users .....	26
Entity.....	26
Comments.....	27
Articles .....	27
Cart .....	27
Chapter .....	28
Course .....	28
Enrollment .....	28
Questions.....	29
Quizes .....	29
QuizesData.....	29
QuizesGroupLk.....	29



QuizesOption .....	30
QuizesQuestionsLk .....	30
QuizesResponse.....	30
Transactions.....	30
<b>Database Table Queries .....</b>	<b>31</b>
Articles .....	31
Cart .....	32
Chapters.....	33
Comments.....	33
Courses .....	34
Enrollments.....	34
Entity.....	36
Questions.....	36
Quizes .....	37
QuizesData.....	37
QuizesGroupLk.....	38
QuizesOptions.....	38
QuizesQUESTionsLk .....	39
QuizesReponse .....	39
Transactions.....	39
Users .....	40
<b>4. User Screens &amp; Code.....</b>	<b>41</b>
Home Page .....	42
Login .....	42
Register.....	43
Courses .....	43
Courses Detail.....	44



Chapters .....	44
Cart .....	45
Quiz.....	45
Ask Question .....	46
Admin Course Screen .....	46
Admin Chapters.....	47
Admin Questions.....	47
Views .....	49
Login.cshtml.....	49
ForgotPassword.cshtml .....	49
ChangePassword.cshtml.....	51
VerifyEmail.cshtml .....	51
ResetPasswordConfirmation.cshtml .....	52
Error.cshtml .....	52
Lockout.cshtml.....	52
Login Master (_Login.cshtml) .....	52
Register.cshtml .....	54
Public Master (_Layout.cshtml).....	55
Public Pages Stylesheet .....	58
Edit Course.....	85
Course Detail .....	87
Blog > Index.cshtml.....	92
Blog > Details.cshtml .....	93
Questions > index.cshtml .....	96
Ask.cshtml.....	97
Reset Password (ResetPassword.cshtml).....	98
Admin Master (_LayoutAdmin.cshtml) .....	100



Stylesheet for Admin Master.....	102
Supporting Javascript file.....	109
Controllers Admin .....	110
ChaptersController.cs .....	110
CoursesController.cs.....	112
DashboardController.cs.....	114
QuestionsController.cs.....	115
QuizesController.cs .....	117
ReportsController.cs .....	119
Controllers Public .....	120
CourseController.cs .....	120
QuestionsController.cs .....	123
QuizesController.cs.....	126
BlogController.cs .....	129
ChaptersController.cs .....	132
QuizesController.cs.....	135
AccountController.cs.....	137
ManageController.cs.....	146
Models.....	153
User.cs .....	153
Transaction.cs.....	154
Quiz.cs.....	155
Entity.cs .....	156
DigiApp.cs .....	157
Course.cs.....	158
Question.cs .....	159
Enrollment.cs.....	160



Cart.cs .....	161
Article.cs .....	162
QuizData.cs .....	163
QuizOptions.cs.....	163
QuizQuestionsLK.cs.....	164
QuizResponse.cs .....	164
Chapter.cs .....	165
Comment.cs.....	165
App_Start Helpers .....	166
BundleConfig.cs .....	166
FilterConfig.cs .....	166
IdentityConfig.cs .....	167
RouteConfig.cs.....	169
Startup.Auth.cs .....	169
WebApiConfig.cs.....	170
Global.asax.cs .....	170
enAreaRegistration.cs.....	171
adminAreaRegistration.cs .....	171
Package.Config.....	173
Web.config.....	174
5. Testing and Maintenance .....	176
5.1 Unit Test Screen .....	176
Test Code Article .....	176
Login Test .....	176
Validation .....	178
Cost Estimation & summary .....	179
Code Metrics .....	179



6. Reports .....	195
7. Future Scope .....	195
8. Conclusion.....	197
9. Bibliography .....	198
10. References .....	198



# **1. OVERVIEW**

## **1.1 Introduction**

“Quality and value: The true purpose of digital learning” From the ancient time we all learning new skills, books and different study materials using printed paper materials. In modern era, People don't have enough time to sit on a place and go for library to learn something. People wants something that they can carry and read, learn any time anywhere. Digital Learning help user to read study materials and different books using laptop, mobile and tablets.

## **1.2 Objectives**

In Digital Learning Application we have following objectives:

- Manage Learners.
- Track all courses and materials received.
- Manage courses and fees.
- Keep track of course status and notify learner.
- Manage login and role.
- It must work fast.
- It must be reliable.
- It must be economical.



## 2. SURVEY OF TECHNOLOGIES

### 2.1 Project Category

**Web Application** using Visual Studio IDE for (Visual C# and asp.net 4.5 MVC)

#### 2.1.1 Front-End with ASP.NET MVC

In ASP.NET MVC, since you develop the applications using C#, you can create simple classes of the data and then work with the data using those classes inside your application without having to worry about any side-code.

ADO.NET is a part of ASP.NET MVC (.NET Framework) that lets you work with data. You might be familiar with LINQ that enables you to work with data, slicing and dicing of the data is a great plug point in ASP.NET MVC.

C# and other .NET languages let you create your own modules, use some library tools from MSDN (Microsoft Developer Network) and create your own web applications. ASP.NET is vast enough for you to develop any sort of application, and you don't even have to worry about the platform, thanks to the Mono Project, you can now write the ASP.NET applications on the non-Windows OS too. Before Mono Project, PHP was the multi-platform language, but now, ASP.NET too is a multi-platform language.

#### Key Benefits

- More Reliable
- More Secure
- Cheaper
- Shorter code, more function!
- User Friendly
- Good Performance

#### 2.1.2 Database (MS-SQL Server)

**Information representation:** - In SQL server, data is represented in terms of rows and columns of a table. Data stored as a table can be easily visualized.



**Unique definition of rows:** - The relation model requires each row to be uniquely defined by at least one or more columns of a table. The unique row requirement ensures that each row in the table can be accessed and changed independently from other rows of the table. It means changes made to a row in the table do not affect the other rows.

**Systematic treatment of Null Values:** - SQL server like most RDBMS treats Null values, zeros and blanks differently. While creating a table, one can specify whether a field allows Null or not.

**High level Update, Insert and Delete:** - In SQL server, if a record is updated or deleted in the master table the corresponding records in the referencing table is also updated or dropped.

## 2.2 Software and Hardware Requirements

### Hardware Requirements for Deployment Server:

Processor	Pentium 4 or higher version
Speed	2.5 GHz
RAM	Minimum 2 GB RAM
Disk	NTFS file system-formatted partition with a minimum of 3 GB of free space
Network	56 kilobits per second (Kbps) connection between client computers and server

### Software Used:

IIS Web Server	IIS 7+
Microsoft Project Report	2003
Database	SQL Server 2012
Asp.net Runtime environment	4.5
Firewall	Port Security & Network security



A code-authoring environment

VS 2015, SQL Management Studio

SMTP Server

To send e-mail messages



### **3. SYSTEM ANALYSIS**

#### **3.1 Identification of need**

It pops in my mind when I started Ignou MCA, its very hard to attend all classes while working 9 to 5. I tried to find all study material in compact version, But I tried lots of websites but their no such material. I discussed this problem with many coworker and Ignou alumni but they told same, I thought to start a platform where people connect and ask questions, learn related topics in compact version.

So I started analysis of the this project and in many way and found that its good if I start a website with eLearning capabilities. This website has lots of opportunity for ignou learners they can find all courses in future to learn. I planned this to launch on a real live website.

#### **3.2 High level system requirement:**

- Admin/Author creates the courses.
- Learners enroll and start learning.
- Learner pay fees if opted paid course.
- Learners attempt quiz and check their progress.
- Learners can also share their understanding about particular topic.
- Learner can ask questions and can answer.

#### **3.3 Requirement Specification**

I gone through the lots of website and gathered information and requirement related digital learning system. I read lots of article to understand the meaning of digital learning, features and problems during development.

Depending on the results of the initial investigation the survey is expanded to a more detailed feasibility study. A feasibility study is a test of a system proposal according to its workability, impact on the organization, ability to meet user needs, and effective use of resources. The result of the feasibility study is a formal proposal. This is simply report formal document detailing the nature and scope of proposed system.

##### **3.3.1 Detailed Functional Requirements**

- Admin/Author creates the courses.



- Learners enroll and start learning.
- Learner pay fees if opted paid course.
- Learners attempt quiz and check their progress.
- Learners can also share their understanding about particular topic.
- Learner can ask questions and can answer.

### **3.3.2 Technical Specifications**

- Make it platform and device agnostic.
- Keep it short
- Learning goal should be visible
- Make it challenging and fun
- Make it social
- Keep it diverse

## **3.4 Project Planning and Scheduling**

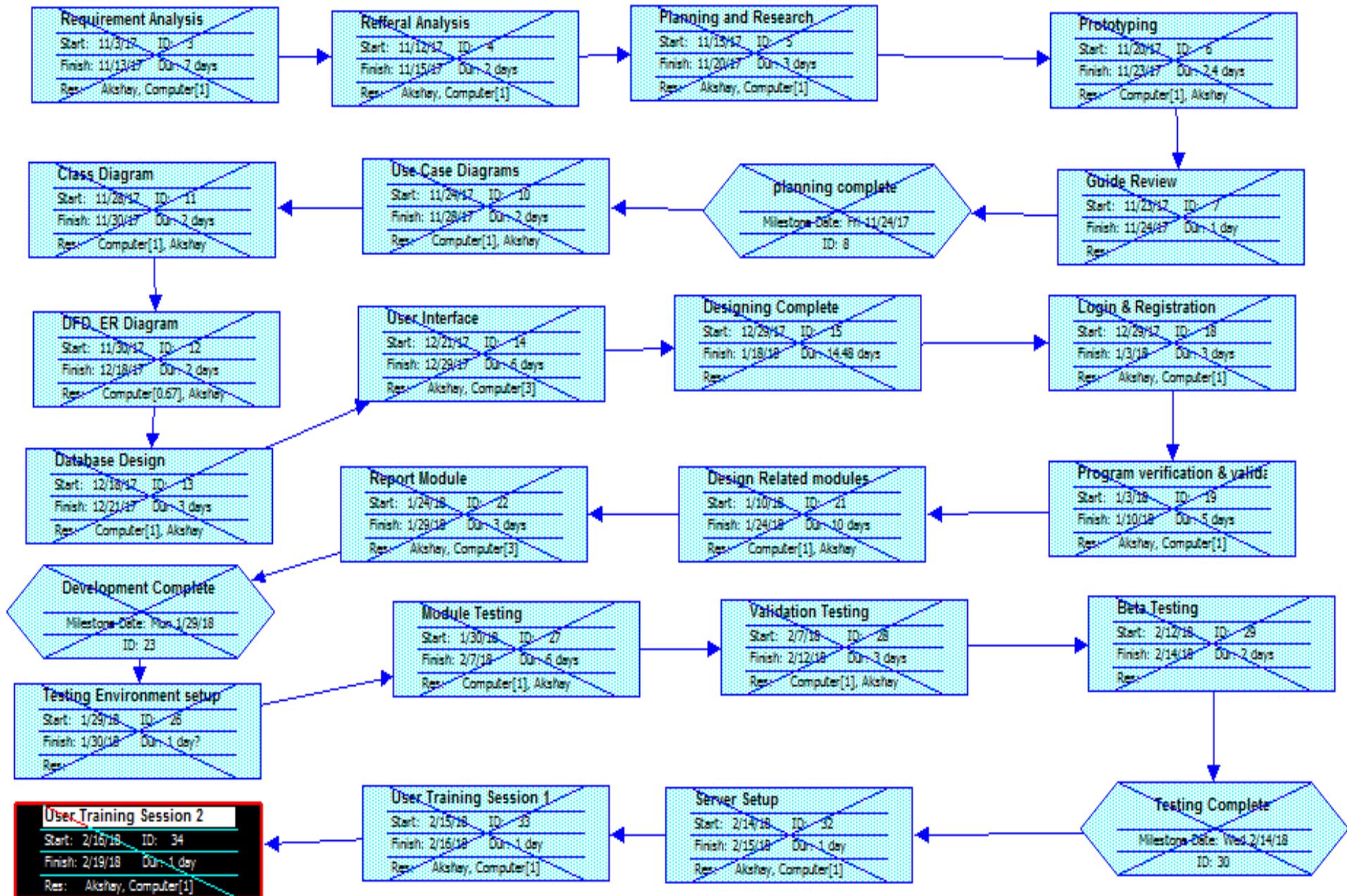
Our planning was about all kinds of resources required and estimation regarding total cost. We also finalized how much time it will take to implement the software with in the given constraint of the cost, effort and resources. In resources we also considered human resources, reusable software resources and environmental resources.

### **3.4.1 Pert Chart**

Our first step to decide what our tasks are performed which tasks depends on which. Following is the list of task to be performed.

- Prepare System Analysis
- Prepare Feasibility Study
- Prepare Software and hardware Requirement Study
- Prepare Software Design
- Identifying Reusable Components
- Database Preparation
- Coding
- Implementation System Security
- Testing Software Installation Software
- Maintenance





### 3.4.2 Gantt Chart

	Task Name	Duration	Start	Finish	Predecessors	Cost	% Complete
1	<input checked="" type="checkbox"/> Digital Learning	66.52 days?	Fri 11/3/17	Mon 2/19/18		\$523.64	99%
2	<input checked="" type="checkbox"/> Planning and Analysis	15.52 days	Fri 11/3/17	Fri 11/24/17		\$119.20	100%
3	Requirement Analysis	7 days	Fri 11/3/17	Mon 11/13/17		\$57.00	100%
4	Refferal Analysis	2 days	Sun 11/12/17	Wed 11/15/17	3	\$17.00	100%
5	Planning and Research	3 days	Wed 11/15/17	Mon 11/20/17	4	\$25.00	100%
6	Prototyping	2.4 days	Mon 11/20/17	Thu 11/23/17	5	\$20.20	100%
7	Guide Review	1 day	Thu 11/23/17	Fri 11/24/17	6	\$0.00	100%
8	planning complete	0 days	Fri 11/24/17	Fri 11/24/17	7	\$0.00	100%
9	<input checked="" type="checkbox"/> Designing	29.48 days	Fri 11/24/17	Thu 1/18/18		\$129.44	100%
10	Use Case Diagrams	2 days	Fri 11/24/17	Tue 11/28/17	8	\$17.00	100%
11	Class Diagram	2 days	Tue 11/28/17	Thu 11/30/17	10	\$17.00	100%
12	DFD, ER Diagram	2 days	Thu 11/30/17	Mon 12/18/17	11	\$19.44	100%
13	Database Design	3 days	Mon 12/18/17	Thu 12/21/17	12	\$25.00	100%
14	User Interface	6 days	Thu 12/21/17	Fri 12/29/17	13	\$51.00	100%
15	Designing Complete	14.48 days	Fri 12/29/17	Thu 1/18/18	14	\$0.00	100%
16	<input checked="" type="checkbox"/> Development	21 days	Fri 12/29/17	Mon 1/29/18		\$174.00	100%
17	<input checked="" type="checkbox"/> Module Designing	21 days	Fri 12/29/17	Mon 1/29/18		\$174.00	100%
18	Login & Registration	3 days	Fri 12/29/17	Wed 1/3/18	15	\$25.00	100%
19	Program verification & validation	5 days	Wed 1/3/18	Wed 1/10/18	18	\$41.00	100%
20	<input checked="" type="checkbox"/> Course & Syllabus system	10 days	Wed 1/10/18	Wed 1/24/18		\$81.00	100%
21	Design Related modules	10 days	Wed 1/10/18	Wed 1/24/18	19	\$81.00	100%
22	Report Module	3 days	Wed 1/24/18	Mon 1/29/18	21	\$27.00	100%
23	Development Complete	0 days	Mon 1/29/18	Mon 1/29/18	22	\$0.00	100%
24	<input checked="" type="checkbox"/> Testing	12 days?	Mon 1/29/18	Wed 2/14/18		\$74.00	100%
25	<input checked="" type="checkbox"/> Alpha Testing	10 days?	Mon 1/29/18	Mon 2/12/18		\$74.00	100%
26	Testing Environment setup	1 day?	Mon 1/29/18	Tue 1/30/18	23	\$0.00	100%
27	Module Testing	6 days	Tue 1/30/18	Wed 2/7/18	26	\$49.00	100%
28	Validation Testing	3 days	Wed 2/7/18	Mon 2/12/18	27	\$25.00	100%
29	Beta Testing	2 days	Mon 2/12/18	Wed 2/14/18	28	\$0.00	100%
30	Testing Complete	0 days	Wed 2/14/18	Wed 2/14/18	29	\$0.00	100%

# Completed Tasks

Completed Tasks as of Sun 4/1/18  
mcsp60-Project

ID	Task Name	Duration	Start	Finish	% Comp.	Cost
<b>November 2017</b>						
3	Requirement Analysis	7 days	Fri 11/3/17	Mon 11/13/17	100%	\$57.00
4	Referral Analysis	2 days	Sun 11/12/17	Wed 11/15/17	100%	\$17.00
5	Planning and Research	3 days	Wed 11/15/17	Mon 11/20/17	100%	\$25.00
6	Prototyping	2.4 days	Mon 11/20/17	Thu 11/23/17	100%	\$20.20
7	Guide Review	1 day	Thu 11/23/17	Fri 11/24/17	100%	\$0.00
8	planning complete	0 days	Fri 11/24/17	Fri 11/24/17	100%	\$0.00
10	Use Case Diagrams	2 days	Fri 11/24/17	Tue 11/28/17	100%	\$17.00
11	Class Diagram	2 days	Tue 11/28/17	Thu 11/30/17	100%	\$17.00
12	DFD, ER Diagram	2 days	Thu 11/30/17	Mon 12/18/17	100%	\$19.44
<b>December 2017</b>						
12	DFD, ER Diagram	2 days	Thu 11/30/17	Mon 12/18/17	100%	\$19.44
13	Database Design	3 days	Mon 12/18/17	Thu 12/21/17	100%	\$25.00
14	User Interface	6 days	Thu 12/21/17	Fri 12/29/17	100%	\$51.00
15	Designing Complete	14.48 days	Fri 12/29/17	Thu 1/18/18	100%	\$0.00
18	Login & Registration	3 days	Fri 12/29/17	Wed 1/3/18	100%	\$25.00
<b>January 2018</b>						
15	Designing Complete	14.48 days	Fri 12/29/17	Thu 1/18/18	100%	\$0.00
18	Login & Registration	3 days	Fri 12/29/17	Wed 1/3/18	100%	\$25.00
19	Program verification & validation	5 days	Wed 1/3/18	Wed 1/10/18	100%	\$41.00
21	Design Related modules	10 days	Wed 1/10/18	Wed 1/24/18	100%	\$81.00
22	Report Module	3 days	Wed 1/24/18	Mon 1/29/18	100%	\$27.00
23	Development Complete	0 days	Mon 1/29/18	Mon 1/29/18	100%	\$0.00
28	Testing Environment setup	1 day?	Mon 1/29/18	Tue 1/30/18	100%	\$0.00
27	Module Testing	6 days	Tue 1/30/18	Wed 2/7/18	100%	\$49.00
<b>February 2018</b>						
27	Module Testing	6 days	Tue 1/30/18	Wed 2/7/18	100%	\$49.00
28	Validation Testing	3 days	Wed 2/7/18	Mon 2/12/18	100%	\$25.00
29	Beta Testing	2 days	Mon 2/12/18	Wed 2/14/18	100%	\$0.00
30	Testing Complete	0 days	Wed 2/14/18	Wed 2/14/18	100%	\$0.00
32	Server Setup	1 day	Wed 2/14/18	Thu 2/15/18	100%	\$9.00
33	User Training Session 1	1 day	Thu 2/15/18	Fri 2/16/18	100%	\$9.00

## 3.6 Preliminary Product Descriptions

### 3.6.1 Module details

The System modules are:

- Learners module
- Course Author module
- Admin module

#### 3.6.1.1 Learners module

- **Before login:** Learners can browse courses, See Answers, Share blogs etc.
- **After Login:** Enroll for a Course, Ask questions, Answer question

#### 3.6.1.2 Author module

- Create course
- See enrolled students
- Answer them, Help them in learning

#### 3.6.1.3 Admin module

- Manage Setting



- Manage Authors, Courses, Learners
- Manage financial data
- See different reports



## 3.7 Object Oriented Models

### 3.7.1 Data Flow Diagram

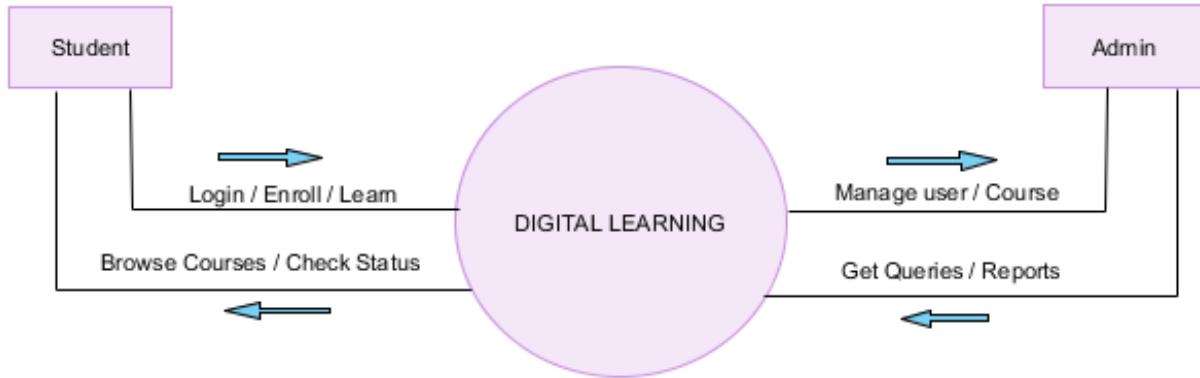


Figure 1 Data Flow Diagram

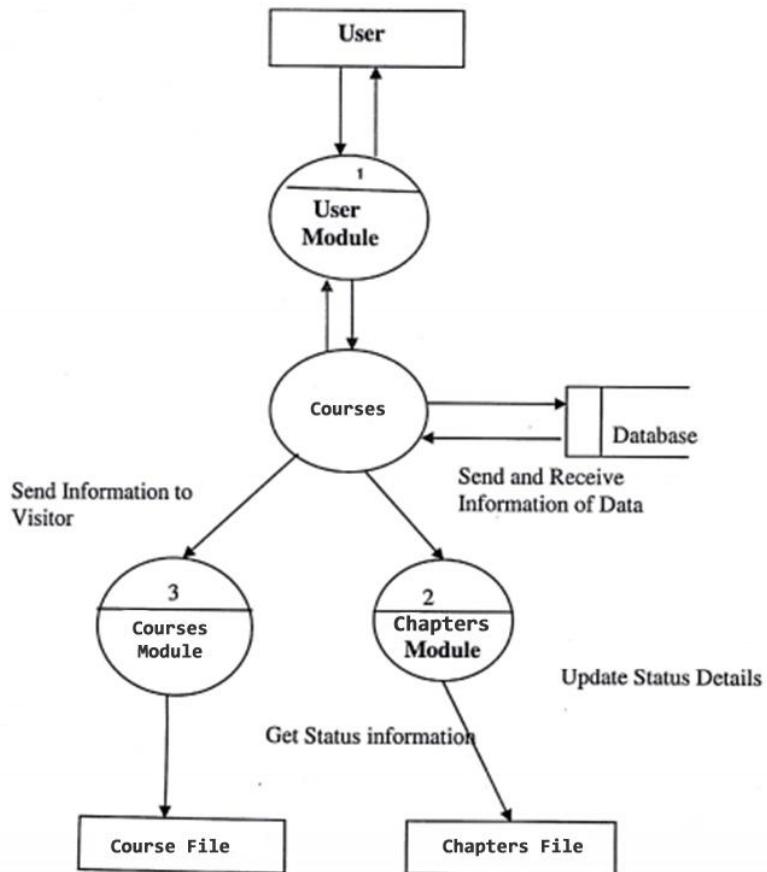
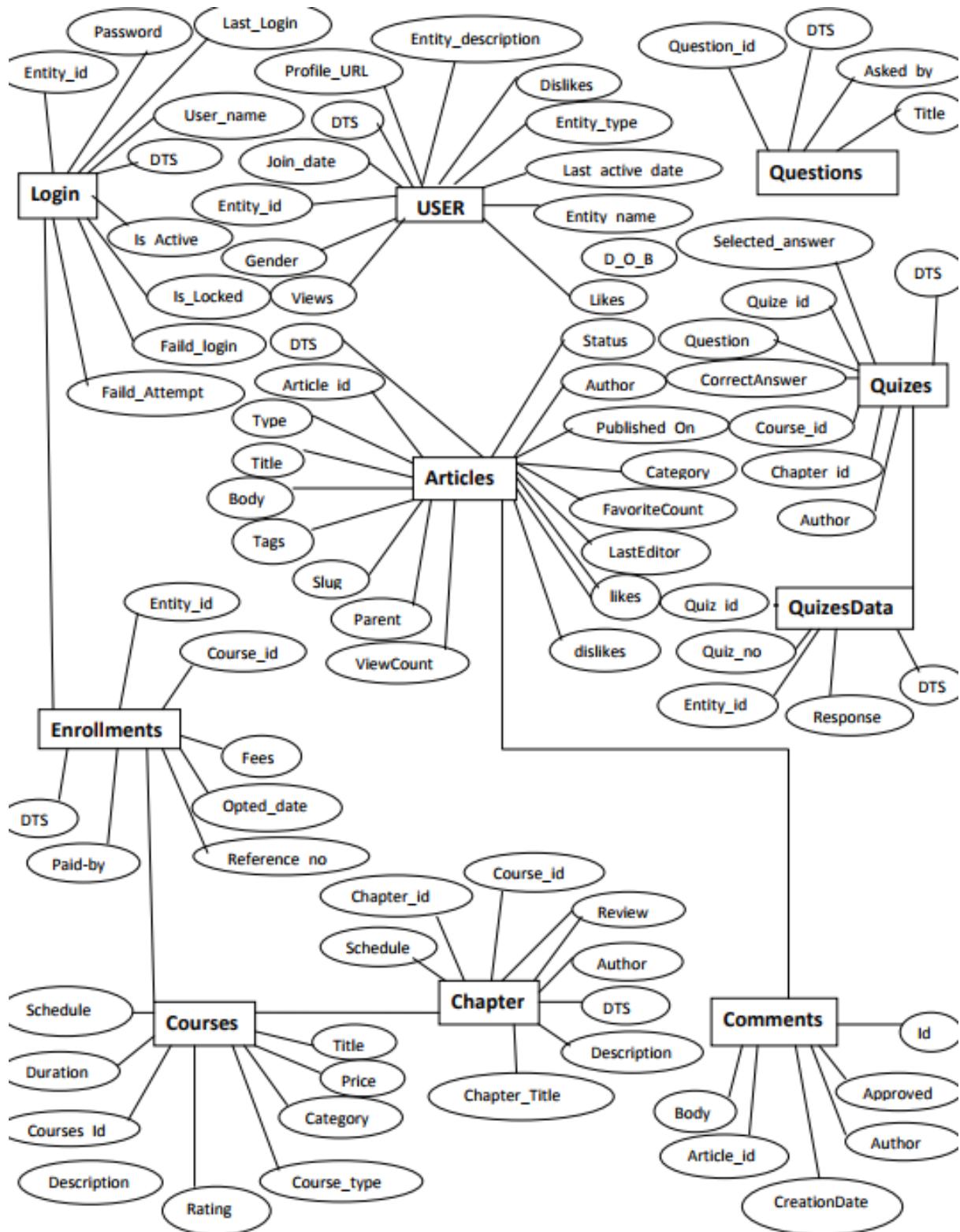


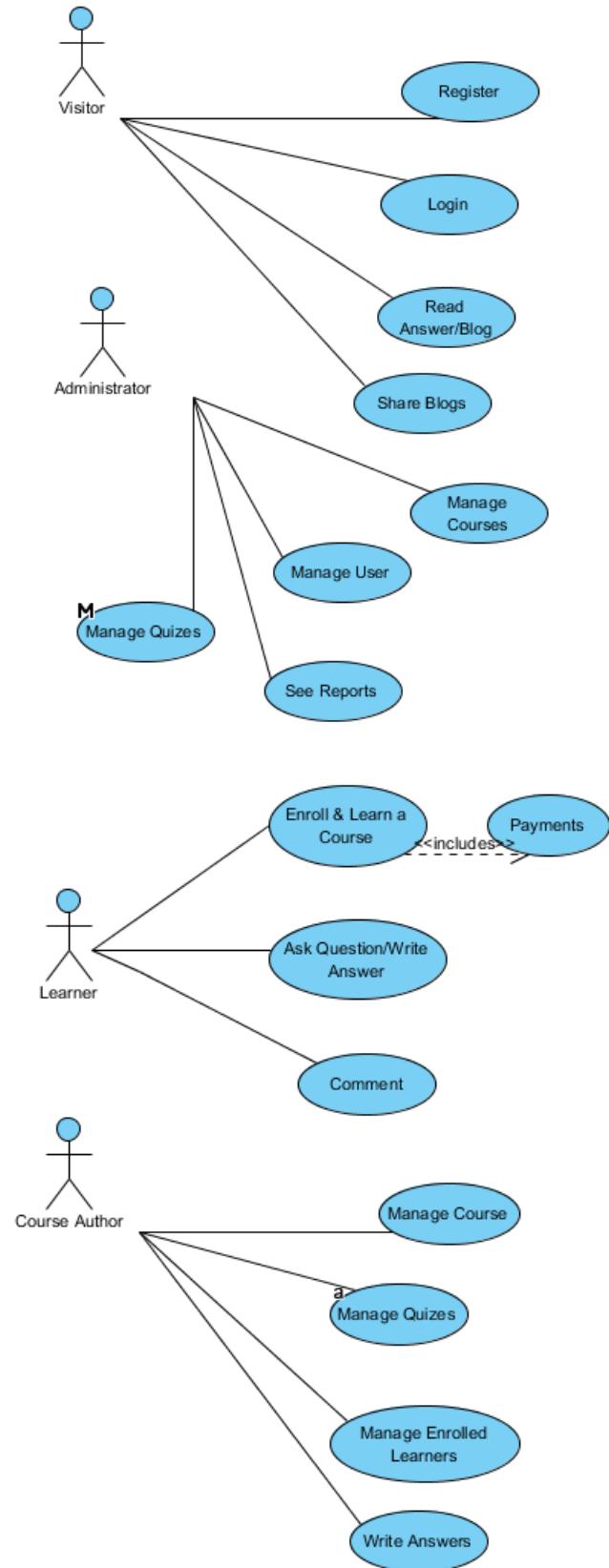
Figure 2 Level 1 DFD

### 3.7.2 ER Diagram

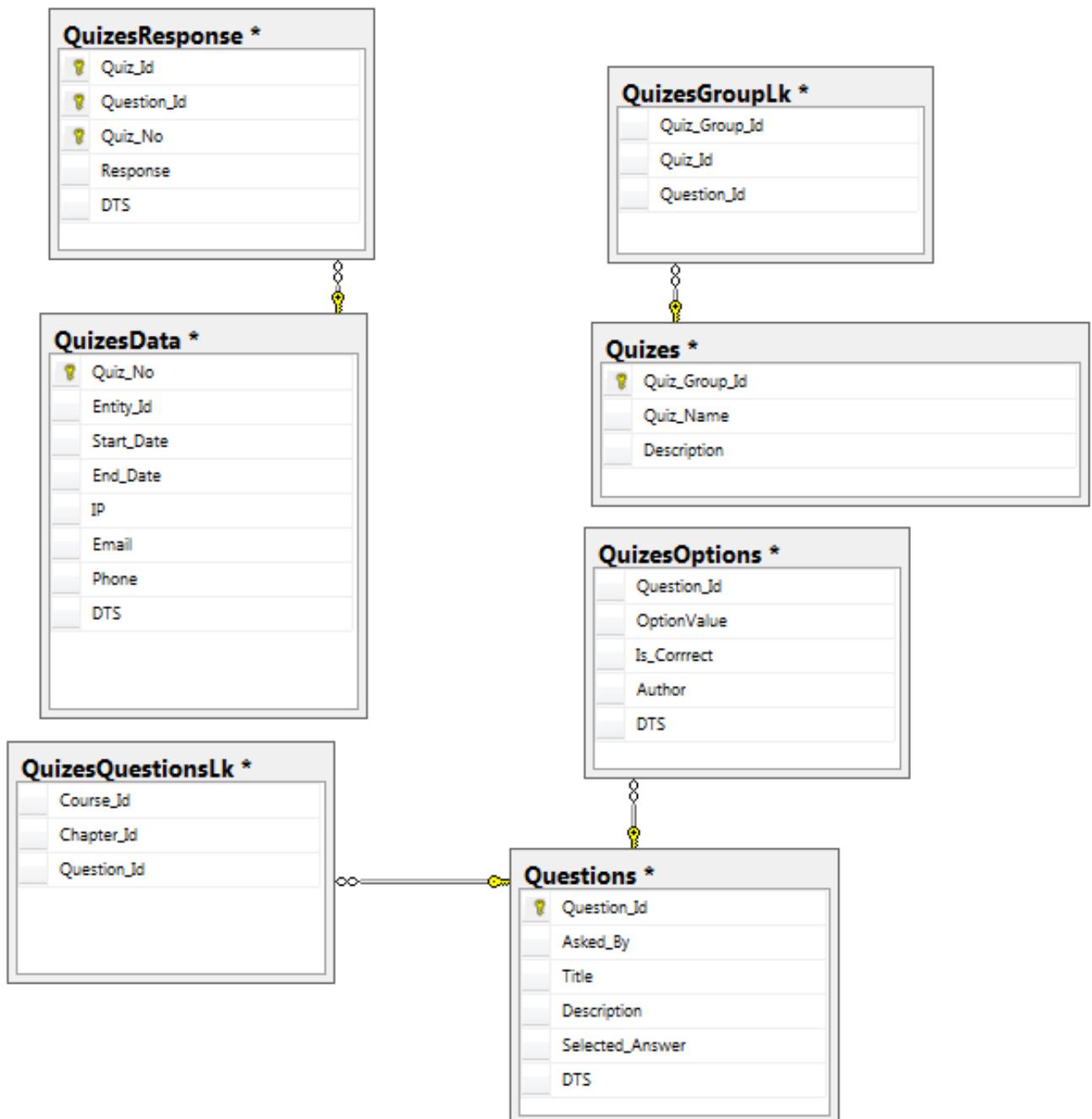


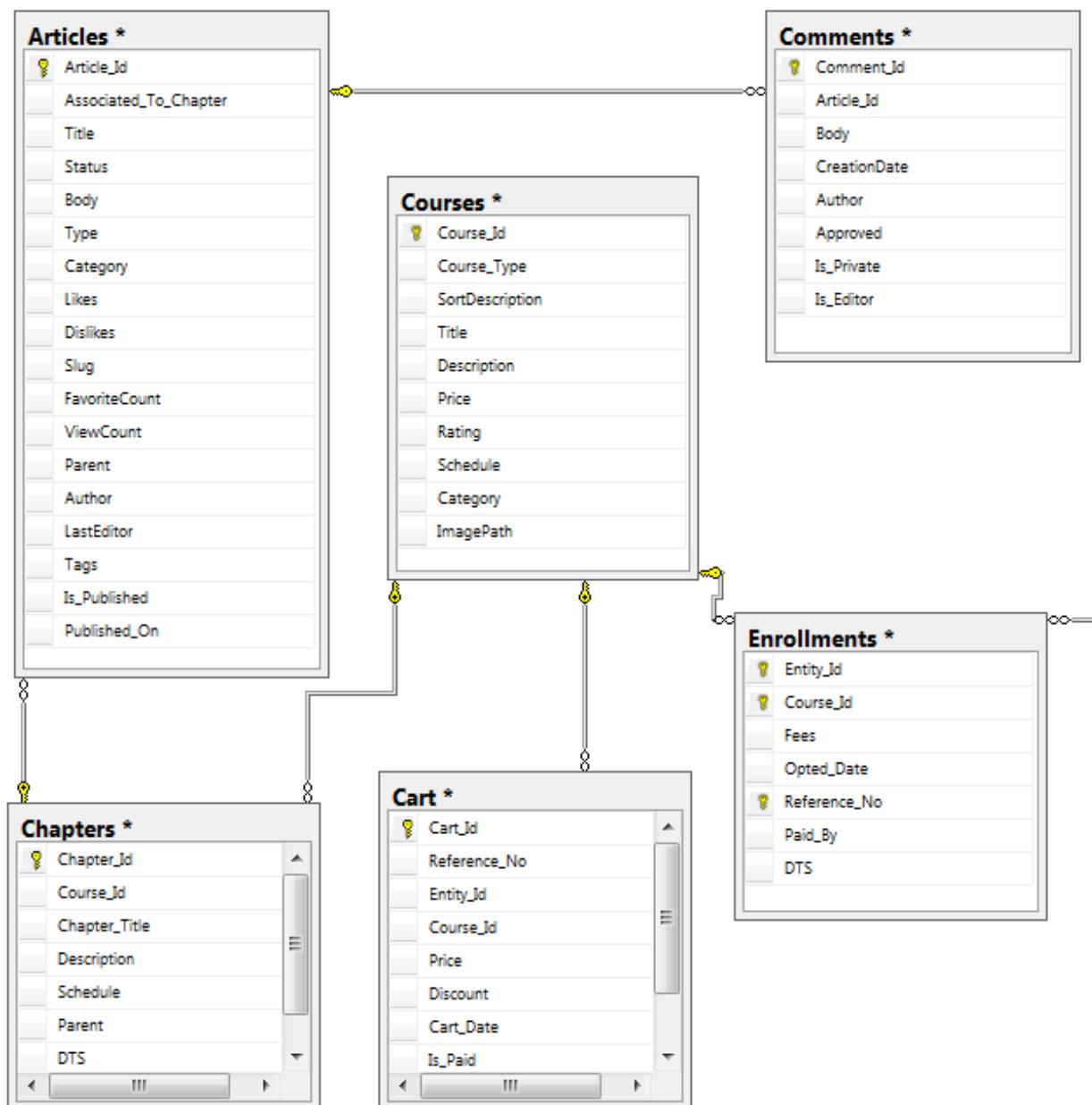


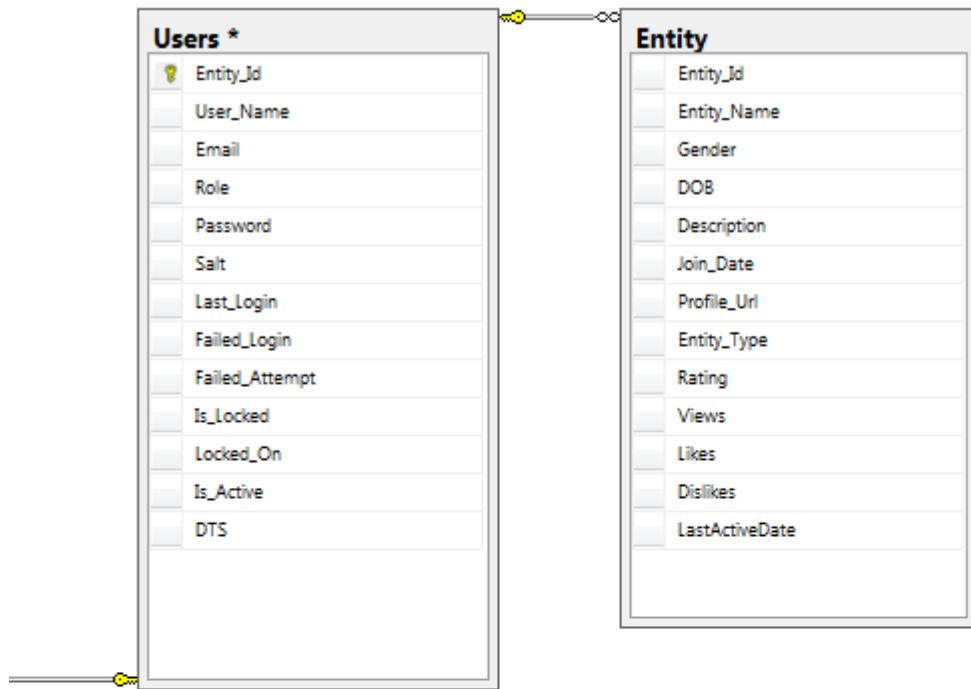
### 3.7.3 Use Case Diagram



### 3.7.4 Class Diagram (Updated)







Transactions	
Entity_Id	Transaction_Id
Mobile_No	Reference_No
Email	Website
Callback_Url	Response



## Table Definitions

### Users

Column Name	Data Type	Description
Entity_Id	Int	
User_Name	Nvarchar(50)	
Email	Nvarchar(250)	
Role	Int	
Password	Nvarchar(550)	Encrypted Password
Salt	Nvarchar(250)	
Last_Login	Datetime	
Failed_Login	Datetime	
Failed_Attempt	Int	
Is_Locked	Bit	
Locked_On	Datetime	
Is_Active	Bit	
DTS	datetime	

### Entity

Column Name	Data Type	Description
Entity_Id	Int	
Entity_Name	Nvarchar(50)	
Gender	Char(6)	
DOB	Datetime	
Description	Nvarchar(500)	About User
Join_date	Datetime	
Profile_Url	Nvarchar(500)	
Entity_Type	Char(10)	
Rating	Int	
Views	Int	
Likes	Int	
Dislikes	Int	
LastActiveDate	datetime	



## Comments

Column Name	Data Type	Description
Comment_Id	Int	Auto-Increment
Article_Id	Int	FK Article table
Body	Nvarchar(500)	
CreationDate	Datetime	
Author	Int	FK Entity table
Approved	Bit	
Is_Private	Bit	
Is_Editor	Bit	

## Articles

Column Name	Data Type	Description
Article_Id	int	Auto increment
Title	Nvarchar(255)	
Status	int	
Body	nvarchar(MAX)	
Type	int	
Category	int	
Likes	int	
Dislikes	int	
Slug	Nvarchar(500)	
FavoriteCount	int	
Views	int	
CreatedBy	int	
LastEditor	int	
Tags	Nvarchar(255)	
Is_Submitted	bit	
Submitted_On	datetime	
Is_Published	bit	
Published_On	datetime	
Is_Rejected	bit	
Rejected_On	datetime	

## Cart

Column Name	Data Type	Descriptions
Cart_Id	int	Identity
Reference_No	nvarchar(20)	
Entity_Id	int	
Course_Id	int	
Price	money	
Discount	money	



<b>Cart_Date</b>	datetime	
<b>Is_Paid</b>	bit	
<b>Payment_Date</b>	datetime	

## Chapter

Column Name	Data Type	Descriptions
<b>Chapter_Id</b>	int	Identity
<b>Course_Id</b>	int	
<b>Chapter_Title</b>	nvarchar(250)	
<b>Description</b>	nvarchar(MAX)	
<b>Schedule</b>	nvarchar(500)	
<b>Parent</b>	int	
<b>DTS</b>	datetime	

## Course

Column Name	Data Type	Descriptions
<b>Course_Id</b>	int	Identity
<b>Course_Type</b>	int	
<b>SortDescription</b>	nvarchar(500)	
<b>Description</b>	nvarchar(MAX)	
<b>Price</b>	money	
<b>Rating</b>	int	
<b>Schedule</b>	nvarchar(500)	
<b>Category</b>	nvarchar(50)	
<b>ImagePath</b>	nvarchar(500)	

## Enrollment

Column Name	Data Type	Descriptions
<b>Entity_Id</b>	int	Identity
<b>Course_Id</b>	int	
<b>Fees</b>	money	
<b>Opted_Date</b>	datetime	
<b>Reference_No</b>	nvarchar(50)	
<b>Paid_By</b>	nvarchar(50)	
<b>DTS</b>	datetime	



## Questions

Column Name	Data Type	Descriptions
Question_Id	int	Identity
Asked_By	int	
Title	nvarchar(500)	
Description	nvarchar(MAX)	
Selected_Answer	int	
DTS	datetime	

## Quizes

Column Name	Data Type	Descriptions
Quiz_Group_Id	int	Identity
Quiz_Name	nvarchar(50)	
Description	nvarchar(500)	

## QuizesData

Column Name	Data Type	Descriptions
Quiz_No	nvarchar(25)	Identity
Entity_Id	int	
Start_Date	datetime	
End_Date	datetime	
IP	nvarchar(15)	
Email	nvarchar(50)	
Phone	nvarchar(15)	
DTS	datetime	

## QuizesGroupLk

Column Name	Data Type	Descriptions
Quiz_Group_Id	int	Identity
Quiz_Id	int	
Question_Id	int	



## **QuizesOption**

<b>Column Name</b>	<b>Data Type</b>	<b>Descriptions</b>
<b>Question_Id</b>	int	Identity
<b>OptionValue</b>	nvarchar(500)	
<b>Is_Correct</b>	bit	
<b>Author</b>	int	
<b>DTS</b>	datetime	

## **QuizesQuestionsLk**

<b>Column Name</b>	<b>Data Type</b>	<b>Descriptions</b>
<b>Course_Id</b>	int	Identity
<b>Chapter_Id</b>	int	
<b>Question_Id</b>	bit	

## **QuizesResponse**

<b>Column Name</b>	<b>Data Type</b>	<b>Descriptions</b>
<b>Quiz_Id</b>	int	Identity
<b>Question_Id</b>	int	
<b>Quiz_No</b>	nvarchar(25)	
<b>Response</b>	int	
<b>DTS</b>	datetime	

## **Transactions**

<b>Column Name</b>	<b>Data Type</b>	<b>Descriptions</b>
<b>Transaction_Id</b>	int	Identity
<b>Reference_No</b>	nvarchar(20)	
<b>Mobile_No</b>	nvarchar(15)	
<b>Entity_Id</b>	int	
<b>Txn_Amt</b>	money	
<b>Email</b>	nvarchar(250)	
<b>Website</b>	nvarchar(500)	
<b>Callback_Url</b>	nvarchar(500)	
<b>Response</b>	nvarchar(500)	



# Database Table Queries

## Articles

```
CREATE TABLE [dbo].[Articles](
    [Article_Id] [int] NOT NULL,
    [Associated_To_Chapter] [int] NULL,
    [Title] [nvarchar](255) NULL,
    [Status] [int] NULL,
    [Body] [nvarchar](max) NULL,
    [Type] [int] NULL,
    [Category] [int] NULL,
    [Likes] [int] NULL,
    [Dislikes] [int] NULL,
    [Slug] [nvarchar](144) NULL,
    [FavoriteCount] [int] NULL,
    [ViewCount] [int] NULL,
    [Parent] [int] NULL,
    [Author] [int] NULL,
    [LastEditor] [int] NULL,
    [Tags] [nvarchar](512) NULL,
    [Is_Published] [bit] NULL,
    [Published_On] [datetime] NULL,
    CONSTRAINT [PK_Articles] PRIMARY KEY CLUSTERED
    (
        [Article_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] TEXTIMAGE_ON [PRIMARY]

GO
```



```
ALTER TABLE [dbo].[Articles] ADD CONSTRAINT [DF_Articles_Associated_To_Chapter] DEFAULT ((0)) FOR [Associated_To_Chapter]
```

```
GO
```

## Cart

```
CREATE TABLE [dbo].[Cart](
```

```
    [Cart_Id] [int] IDENTITY(1,1) NOT NULL,  
  
    [Reference_No] [nvarchar](20) NOT NULL,  
  
    [Entity_Id] [int] NOT NULL,  
  
    [Course_Id] [int] NOT NULL,  
  
    [Price] [money] NULL,  
  
    [Discount] [money] NULL,  
  
    [Cart_Date] [datetime] NULL,  
  
    [Is_Paid] [bit] NULL,  
  
    [Payment_Date] [datetime] NULL,
```

```
CONSTRAINT [PK_Cart_1] PRIMARY KEY CLUSTERED
```

```
(
```

```
    [Cart_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
```



## Chapters

```
CREATE TABLE [dbo].[Chapters]()

[Chapter_Id] [int] IDENTITY(1,1) NOT NULL,
[Course_Id] [int] NULL,
[Chapter_Title] [nvarchar](250) NULL,
[Description] [nvarchar](max) NULL,
[Schedule] [nvarchar](500) NULL,
[Parent] [int] NULL,
[DTS] [datetime] NULL,
CONSTRAINT [PK_Chapters_1] PRIMARY KEY CLUSTERED
(
[Chapter_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] TEXTIMAGE_ON [PRIMARY]
```

GO

## Comments

```
CREATE TABLE [dbo].[Comments](
[Comment_Id] [int] NOT NULL,
[Article_Id] [int] NULL,
[Body] [nvarchar](500) NULL,
[CreationDate] [datetime] NULL,
[Author] [int] NULL,
```



```

[Approved] [bit] NULL,
[Is_Private] [bit] NULL,
[Is_Editor] [bit] NULL,
CONSTRAINT [PK_Comments] PRIMARY KEY CLUSTERED
(
    [Comment_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

```

## Courses

```

USE [DigiAppDB]
GO

/******** Object: Table [dbo].[Courses]  Script Date: 4/1/2018 3:09:40 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Courses](
    [Course_Id] [int] IDENTITY(1,1) NOT NULL,
    [Course_Type] [int] NULL,
    [SortDescription] [nvarchar](250) NULL,
    [Title] [nvarchar](500) NULL,
    [Description] [nvarchar](max) NULL,
    [Price] [money] NULL,
    [Rating] [int] NULL,
    [Schedule] [nvarchar](500) NULL,
    [Category] [nvarchar](50) NULL,
    [ImagePath] [nvarchar](500) NULL,
CONSTRAINT [PK_Courses] PRIMARY KEY CLUSTERED
(
    [Course_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] TEXTIMAGE_ON [PRIMARY]

GO

```

## Enrollments

```

USE [DigiAppDB]
GO

/******** Object: Table [dbo].[Enrollments]  Script Date: 4/1/2018 3:09:50 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON

```



GO

```
CREATE TABLE [dbo].[Enrollments](
    [Entity_Id] [int] NOT NULL,
    [Course_Id] [int] NOT NULL,
    [Fees] [money] NULL,
    [Opted_Date] [datetime] NULL,
    [Reference_No] [nvarchar](50) NOT NULL,
    [Paid_By] [nvarchar](50) NULL,
    [DTS] [datetime] NULL,
CONSTRAINT [PK_Enrollments] PRIMARY KEY CLUSTERED
(
    [Entity_Id] ASC,
    [Course_Id] ASC,
    [Reference_No] 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



## Entity

```
USE [DigiAppDB]
GO

/***** Object: Table [dbo].[Entity]  Script Date: 4/1/2018 3:10:07 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

SET ANSI_PADDING ON
GO

CREATE TABLE [dbo].[Entity](
    [Entity_Id] [int] NOT NULL,
    [Entity_Name] [nvarchar](50) NULL,
    [Gender] [char](6) NULL,
    [DOB] [datetime] NULL,
    [Description] [nvarchar](500) NULL,
    [Join_Date] [datetime] NULL,
    [Profile_Url] [nvarchar](500) NULL,
    [Entity_Type] [char](10) NULL,
    [Rating] [int] NULL,
    [Views] [int] NULL,
    [Likes] [int] NULL,
    [Dislikes] [int] NULL,
    [LastActiveDate] [datetime] NULL
) ON [PRIMARY]

GO

SET ANSI_PADDING OFF
GO

ALTER TABLE [dbo].[Entity] WITH CHECK ADD CONSTRAINT [FK_Entity_Users] FOREIGN KEY([Entity_Id])
REFERENCES [dbo].[Users] ([Entity_Id])
GO

ALTER TABLE [dbo].[Entity] CHECK CONSTRAINT [FK_Entity_Users]
GO
```

## Questions

```
USE [DigiAppDB]
GO

/***** Object: Table [dbo].[Questions]  Script Date: 4/1/2018 3:10:24 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Questions](
    [Question_Id] [int] IDENTITY(1,1) NOT NULL,
    [Asked_By] [int] NULL,
```



```

[Title] [nvarchar](500) NULL,
[Description] [nvarchar](max) NULL,
[Selected_Answer] [int] NULL,
[DTS] [datetime] NULL,
CONSTRAINT [PK_Questions] PRIMARY KEY CLUSTERED
(
    [Question_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] TEXTIMAGE_ON [PRIMARY]

GO

```

## Quizes

```

USE [DigiAppDB]
GO

/******** Object: Table [dbo].[Quizes]  Script Date: 4/1/2018 3:10:54 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Quizes](
    [Quiz_Group_Id] [int] IDENTITY(1,1) NOT NULL,
    [Quiz_Name] [nvarchar](50) NULL,
    [Description] [nvarchar](250) NULL,
CONSTRAINT [PK_Quizes] PRIMARY KEY CLUSTERED
(
    [Quiz_Group_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

```

## QuizesData

```

USE [DigiAppDB]
GO

/******** Object: Table [dbo].[QuizesData]  Script Date: 4/1/2018 3:11:22 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[QuizesData](
    [Quiz_No] [nvarchar](25) NOT NULL,
    [Entity_Id] [int] NULL,
    [Start_Date] [datetime] NULL,

```



```

[End_Date] [datetime] NULL,
[IP] [nvarchar](15) NULL,
[Email] [nvarchar](50) NULL,
[Phone] [nvarchar](15) NULL,
[DTS] [datetime] NULL,
CONSTRAINT [PK_QuizesData] PRIMARY KEY CLUSTERED
(
    [Quiz_No] 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

```

## QuizesGroupLk

```

USE [DigiAppDB]
GO

/******** Object: Table [dbo].[QuizesGroupLk]  Script Date: 4/1/2018 3:11:40 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[QuizesGroupLk](
    [Quiz_Group_Id] [int] NULL,
    [Quiz_Id] [int] NULL,
    [Question_Id] [int] NULL
) ON [PRIMARY]

GO

```

## QuizesOptions

```

USE [DigiAppDB]
GO

/******** Object: Table [dbo].[QuizesOptions]  Script Date: 4/1/2018 3:11:50 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[QuizesOptions](
    [Question_Id] [int] NOT NULL,
    [OptionValue] [nvarchar](500) NULL,
    [Is_Correct] [bit] NULL,
    [Author] [int] NULL,
    [DTS] [datetime] NULL
) ON [PRIMARY]

GO

```



## **QuizesQuestionsLk**

```
USE [DigiAppDB]
GO

/******** Object: Table [dbo].[QuizesQuestionsLk]  Script Date: 4/1/2018 3:12:09 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[QuizesQuestionsLk](
    [Course_Id] [int] NULL,
    [Chapter_Id] [int] NULL,
    [Question_Id] [int] NOT NULL
) ON [PRIMARY]

GO
```

## **QuizesResponse**

```
USE [DigiAppDB]
GO

/******** Object: Table [dbo].[QuizesResponse]  Script Date: 4/1/2018 3:12:27 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[QuizesResponse](
    [Quiz_Id] [int] NOT NULL,
    [Question_Id] [int] NOT NULL,
    [Quiz_No] [nvarchar](25) NOT NULL,
    [Response] [int] NULL,
    [DTS] [datetime] NULL,
CONSTRAINT [PK_QuizesResponse] PRIMARY KEY CLUSTERED
(
    [Quiz_Id] ASC,
    [Question_Id] ASC,
    [Quiz_No] 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
```

## **Transactions**

```
USE [DigiAppDB]
GO

/******** Object: Table [dbo].[Transactions]  Script Date: 4/1/2018 3:12:44 PM *****/
```



```

SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

CREATE TABLE [dbo].[Transactions](
    [Transaction_Id] [int] IDENTITY(1,1) NOT NULL,
    [Reference_No] [nvarchar](20) NULL,
    [Mobile_No] [nvarchar](15) NULL,
    [Entity_Id] [int] NULL,
    [Txn_Amt] [money] NULL,
    [Email] [nvarchar](50) NULL,
    [Website] [nvarchar](500) NULL,
    [Callback_Url] [nvarchar](500) NULL,
    [Response] [nvarchar](500) NULL,
CONSTRAINT [PK_Transactions] PRIMARY KEY CLUSTERED
(
    [Transaction_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

```

## Users

```

USE [DigiAppDB]
GO

/**************** Object: Table [dbo].[Users]  Script Date: 4/1/2018 3:13:09 PM *****/
SET ANSI_NULLS ON
GO

SET QUOTED_IDENTIFIER ON
GO

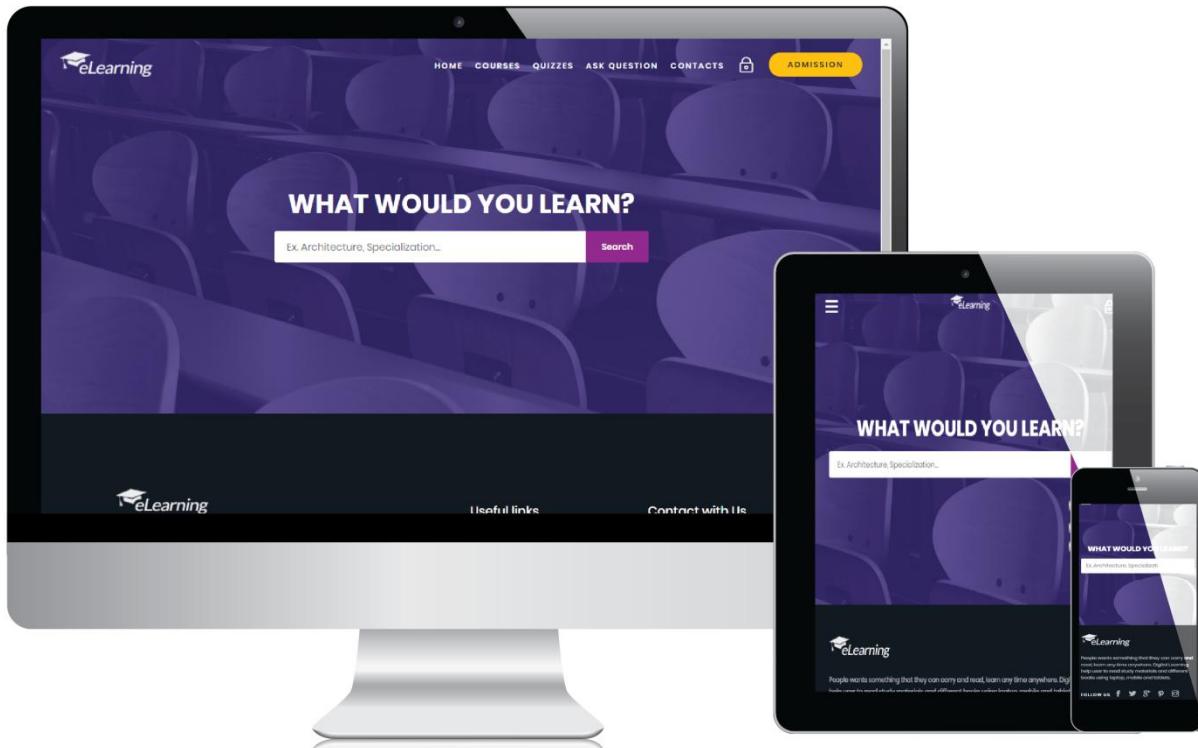
CREATE TABLE [dbo].[Users](
    [Entity_Id] [int] IDENTITY(1,1) NOT NULL,
    [User_Name] [nvarchar](50) NOT NULL,
    [Email] [nvarchar](250) NOT NULL,
    [Role] [int] NULL,
    [Password] [nvarchar](550) NULL,
    [Salt] [nvarchar](250) NULL,
    [Last_Login] [datetime] NULL,
    [Failed_Login] [datetime] NULL,
    [Failed_Attempt] [int] NULL,
    [Is_Locked] [bit] NULL,
    [Locked_On] [datetime] NULL,
    [Is_Active] [bit] NULL,
    [DTS] [datetime] NULL,
CONSTRAINT [PK_Users] PRIMARY KEY CLUSTERED
(
    [Entity_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

## 4. USER SCREENS & CODE



## UI DESIGNS



## Home Page

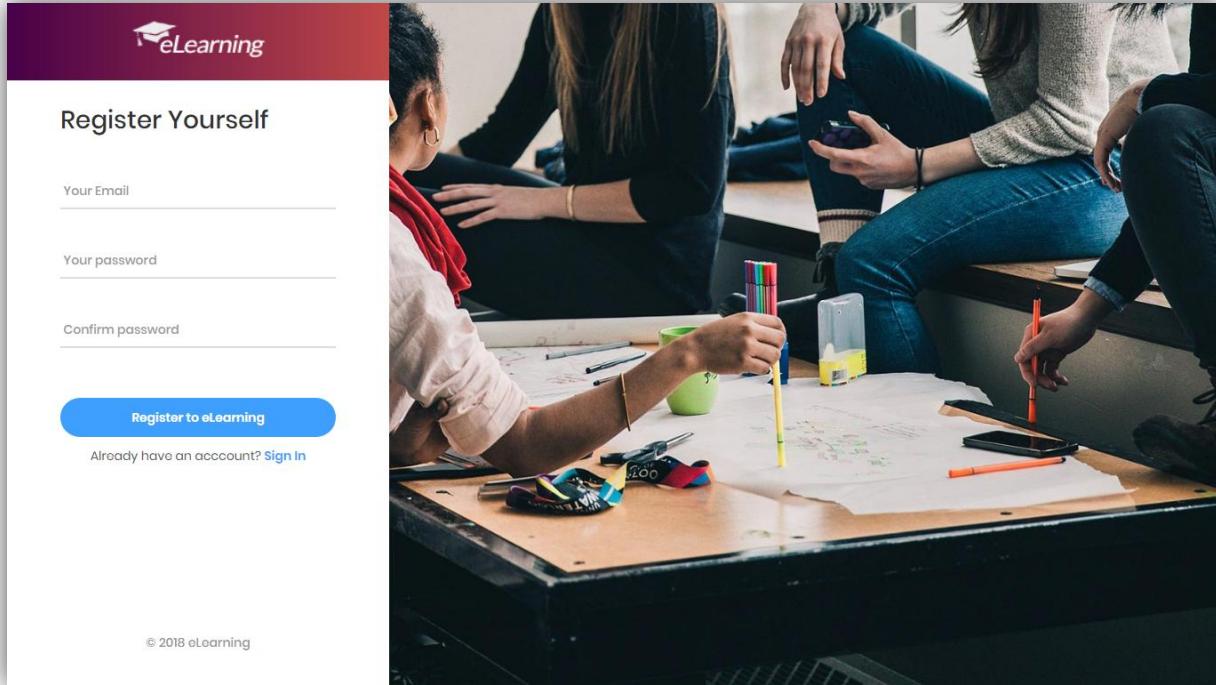
The image shows the homepage of an eLearning platform. At the top left is the logo 'eLearning' with a graduation cap icon. The top right features a navigation bar with links: HOME, COURSES, QUIZZES, ASK QUESTION, CONTACTS, a lock icon, and ADMISSION. Below the navigation is a large search bar with placeholder text 'Ex. Architecture, Specialization...' and a purple 'Search' button. The background is a blurred image of rows of empty blue theater-style seats. At the bottom, there's a dark footer bar with the 'eLearning' logo on the left, and 'Useful links' and 'Contact with Us' on the right.

## Login

The image shows the login page of the eLearning platform. It has a red header bar with the 'eLearning' logo. The main area contains fields for 'Your email' and 'Your password', both with placeholder text. Below these is a 'Forgot password?' link and a blue 'Login to eLearning' button. To the right of the form is a photograph of three students sitting at a desk, working together on a project with papers, pens, and a smartphone. At the bottom left of the page is a copyright notice: '© 2018 elearning'.

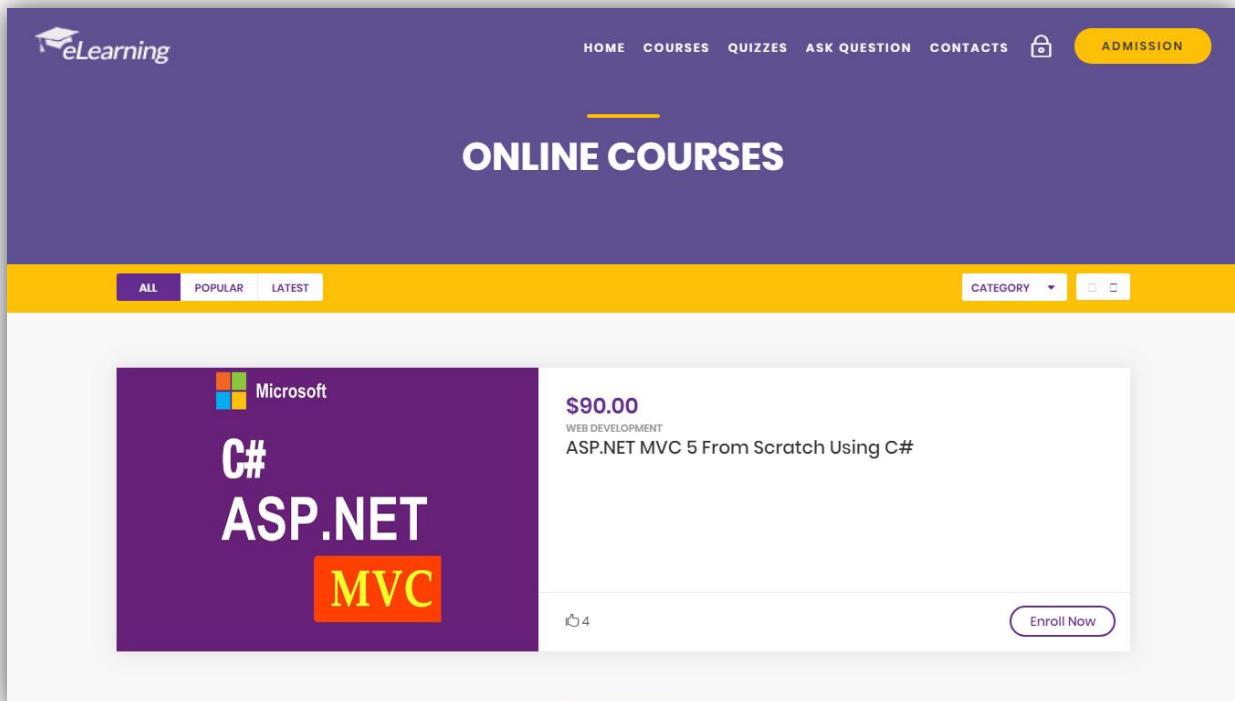


## Register



The image shows a screenshot of an eLearning registration page. At the top left is the logo 'eLearning' with a graduation cap icon. Below it is a large heading 'Register Yourself'. There are three input fields: 'Your Email', 'Your password', and 'Confirm password'. A blue button labeled 'Register to eLearning' is centered below the fields. Below the button is a link 'Already have an account? [Sign In](#)'. At the bottom left is a copyright notice '© 2018 eLearning'. To the right of the form is a photograph of several people sitting around a table, working on papers and using markers.

## Courses



The image shows a screenshot of an eLearning website's course listing page. The header features the 'eLearning' logo and navigation links for HOME, COURSES, QUIZZES, ASK QUESTION, CONTACTS, and ADMISSION. Below the header is a purple banner with the text 'ONLINE COURSES'. The main content area has a yellow header bar with buttons for 'ALL', 'POPULAR', and 'LATEST', and dropdown menus for 'CATEGORY' and search. A course card is displayed prominently, featuring the Microsoft logo, the text 'C# ASP.NET MVC', and a price of '\$90.00'. The course is described as 'WEB DEVELOPMENT' and 'ASP.NET MVC 5 From Scratch Using C#'. It has 4 reviews indicated by a '4' next to a thumbs-up icon. A blue 'Enroll Now' button is at the bottom right of the card.



## Courses Detail

The screenshot shows a course detail page for "ASP.NET MVC 5 FROM SCRATCH USING C#". The top navigation bar includes links for HOME, COURSES, QUIZZES, ASK QUESTION, CONTACTS, and ADMISSION. Below the title, there are tabs for Description, Lessons, and Reviews. A section titled "What will you learn" lists benefits like learning the benefits of using ASP.NET MVC, developing web applications using the MVC design pattern, working with controllers, and working with models. To the right, a promotional box shows a price of \$90.00 with a 60% discount, a "Purchase" button, and a play icon. The overall background is purple.

## Chapters

The screenshot shows a lessons page for a course. The top navigation bar is identical to the previous page. The main content area is titled "Lessons" and shows a list of 18 lessons with a total duration of 01:02:10. The lessons are categorized into sections: "Introduction", "Generative Modeling Review", "Variational Autoencoders", and "Gaussian Mixture Model Review". Below these, individual lessons are listed with their names and durations: Health Science (00:59), Health and Social Care (00:59), History (00:59), Healthcare Science (00:59), and Audiology (00:59). To the right, a promotional box shows a price of \$90.00 with a 60% discount, a "Purchase" button, and a play icon. A "What's included" section lists mobile support and lesson archive.



## Cart

The screenshot shows a navigation bar with links for HOME, COURSES, ADMISSION, and a lock icon. Below the navigation is a progress bar with three steps: 'YOUR CART' (highlighted with a yellow dot), 'PAYMENT', and 'FINISH!'. The main content area displays a table with one item: 'C# MVC5 Course' with a discount of 0% and a price of 69.00 INR. To the right, a summary shows a 'TOTAL' of 69.00 INR. Buttons for 'Checkout' and 'Continue Shopping' are visible.

ITEM	DISCOUNT	PRICE	ACTIONS
C# MVC5 Course	0%	69.00 INR	

**TOTAL 69.00 INR**

After purchase you will receive the link to start the course. Best of Luck

**Checkout**

[← Continue Shopping](#)

## Quiz

The quiz interface includes a user profile picture, a welcome message 'Welcome : aki', and details 'Quiz Selected : Programming C#' and 'Date : 3/31/2018 8:18:56 PM'. The first question asks: 'Which of the following variable types can be assigned a value directly in C#?' with options: Value types, Reference types, Pointer types, and All of the above. The second question asks: 'Which of the following is correct about value type variables in C#?' with options: The value types directly contain data., int, char, and float, which stores numbers, alphabets, and floating point numbers, respectively are value types., When you declare an int type, the system allocates memory to store the value., and All of the above. The correct answer is highlighted in a grey box.

**Questions :**

**Q1.** Which of the following variable types can be assigned a value directly in C#?

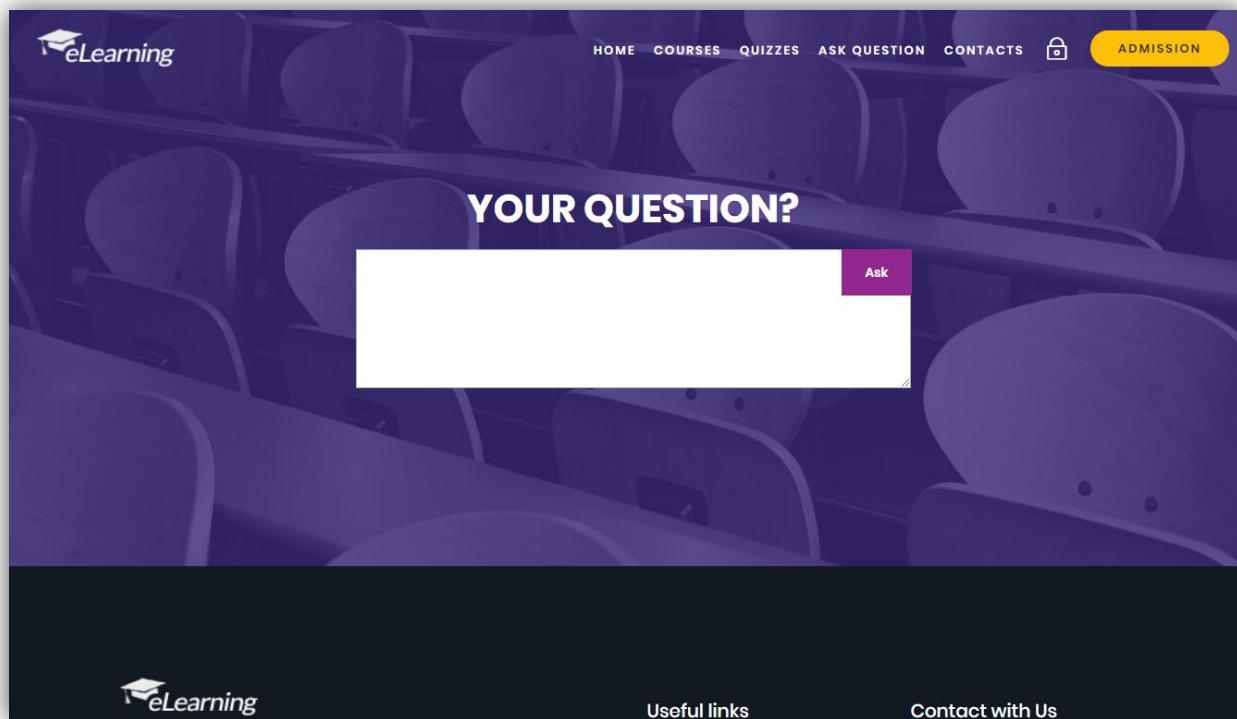
Value types  
 Reference types  
 Pointer types  
 All of the above.

**Q2.** Which of the following is correct about value type variables in C#?

The value types directly contain data.  
 int, char, and float, which stores numbers, alphabets, and floating point numbers, respectively are value types.  
 When you declare an int type, the system allocates memory to store the value.  
 All of the above

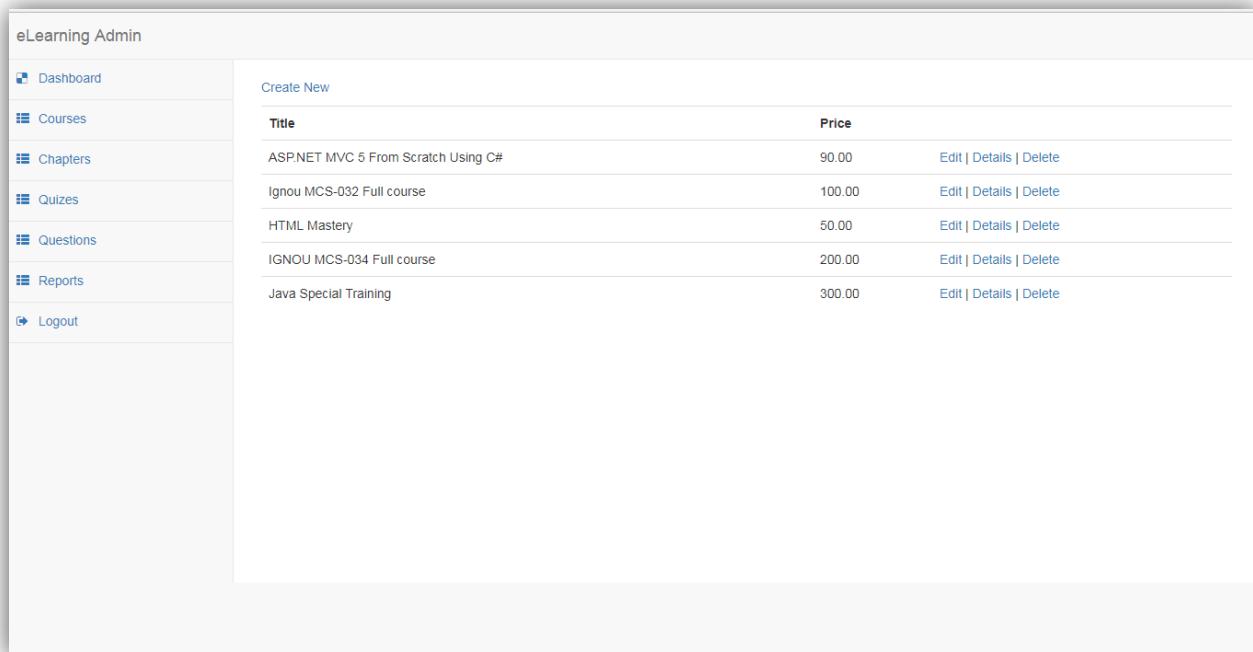


## Ask Question



The screenshot shows the 'Ask Question' page of an eLearning website. The background features a dark image of rows of empty blue theater-style seats. At the top left is the 'eLearning' logo. The top navigation bar includes links for HOME, COURSES, QUIZZES, ASK QUESTION, CONTACTS, a lock icon, and ADMISSION. The main heading 'YOUR QUESTION?' is centered above a large white input field with a purple 'Ask' button in the top right corner. Below the input field is a dark horizontal bar containing the 'eLearning' logo, a 'Useful links' link, and a 'Contact with Us' link.

## Admin Course Screen



The screenshot shows the 'eLearning Admin' course management interface. On the left is a sidebar with navigation links: Dashboard, Courses, Chapters, Quizzes, Questions, Reports, and Logout. The main content area is titled 'Create New' and displays a table of existing courses:

Title	Price	Action
ASP.NET MVC 5 From Scratch Using C#	90.00	Edit   Details   Delete
Ignou MCS-032 Full course	100.00	Edit   Details   Delete
HTML Mastery	50.00	Edit   Details   Delete
IGNOU MCS-034 Full course	200.00	Edit   Details   Delete
Java Special Training	300.00	Edit   Details   Delete



## Admin Chapters

eLearning Admin

Create New			
Chapter Title	Description	DTS	
ASP.NET MVC - The Big Picture		3/31/2018 1:37:42 PM	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Course Introduction		3/31/2018 1:38:02 PM	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
ASP.NET MVC Request Processing	ASP.NET MVC Request Processing	3/31/2018 1:38:02 PM	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Creating Our First ASP.NET MVC Application	Creating Our First ASP.NET MVC Application	3/31/2018 1:38:19 PM	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Creating a Sample MVC Application	Creating a Sample MVC Application	3/31/2018 1:38:54 PM	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>

[Dashboard](#)  
[Courses](#)  
[Chapters](#)  
[Quizzes](#)  
[Questions](#)  
[Reports](#)  
[Logout](#)

## Admin Questions

eLearning Admin

### Questions

Create New	
Title	DTS
Which of the following variable types can be assigned a value directly in C#?	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Which of the following is correct about value type variables in C#?	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Which of the following converts a type to a double type in C#?	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Which of the following operator determines whether an object is of a certain type in C#?	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
Which of the following access specifier in C# allows a class to hide its member variables and member functions from other functions and objects?	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>
C# supports multiple inheritance.	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>

[Dashboard](#)  
[Courses](#)  
[Chapters](#)  
[Quizzes](#)  
[Questions](#)  
[Reports](#)  
[Logout](#)



# **Project Codes**



## Views

### Login.cshtml

```
@using WebApp.Models
@model LoginViewModel
 @{
    ViewBag.Title = "Log in";
    Layout = "~/Views/Shared/_Login.cshtml";
}
<div id="preloader" style="display: none;">
    <div data-loader="circle-side" style="display: none;"></div>
</div><div id="login">
    <aside>
        <figure>
            <a href="/Home"><img src "~/Content/img/logo.png" width="149" height="42" data-retina="true" alt=""></a>
        </figure>
        <form>

            <div class="form-group">
                <span class="input">
                    <input class="input_field" type="email" autocomplete="off" name="email">
                    <label class="input_label">
                        <span class="input_label-content">Your email</span>
                    </label>
                </span>

                <span class="input">
                    <input class="input_field" type="password" autocomplete="new-password" name="password">
                    <label class="input_label">
                        <span class="input_label-content">Your password</span>
                    </label>
                </span>
                <small><a href="#0">Forgot password?</a></small>
            </div>
            <a href="#" class="btn_1 rounded full-width add_top_60">Login to eLearning</a>
            <div class="text-center add_top_10">New to eLearning? <strong><a href="/Account/Register">Sign up!</a></strong></div>
        </form>
        <div class="copy">© 2018 elearning</div>
    </aside>
</div>
```

### ForgotPassword.cshtml

```
@model WebApp.Models.ForgotPasswordViewModel
 @{
    ViewBag.Title = "Forgot your password?";
}

<h2>@ViewBag.Title.</h2>
```



```
@using (Html.BeginForm("ForgotPassword", "Account", FormMethod.Post, new { @class = "form-horizontal", role = "form" }))
{
    @Html.AntiForgeryToken()
    <h4>Enter your email.</h4>
    <hr />
    @Html.ValidationSummary("", new { @class = "text-danger" })
    <div class="form-group">
        @Html.LabelFor(m => m.Email, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.TextBoxFor(m => m.Email, new { @class = "form-control" })
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-offset-2 col-md-10">
            <input type="submit" class="btn btn-default" value="Email Link" />
        </div>
    </div>
}

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
}
```



## ChangePassword.cshtml

```
@model WebApp.Models.ChangePasswordViewModel
@{
    ViewBag.Title = "Change Password";
}

<h2>@ViewBag.Title.</h2>

@using (Html.BeginForm("ChangePassword", "Manage", FormMethod.Post, new { @class = "form-horizontal", role = "form" }))
{
    @Html.AntiForgeryToken()
    <h4>Change Password Form</h4>
    <hr />
    @Html.ValidationSummary("", new { @class = "text-danger" })
    <div class="form-group">
        @Html.LabelFor(m => m.OldPassword, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.PasswordFor(m => m.OldPassword, new { @class = "form-control" })
        </div>
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.NewPassword, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.PasswordFor(m => m.NewPassword, new { @class = "form-control" })
        </div>
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.ConfirmPassword, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.PasswordFor(m => m.ConfirmPassword, new { @class = "form-control" })
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-offset-2 col-md-10">
            <input type="submit" value="Change password" class="btn btn-default" />
        </div>
    </div>
}
@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
}
```

## VerifyEmail.cshtml

```
@{
    ViewBag.Title = "Confirm Email";
}

<h2>@ViewBag.Title.</h2>
<div>
    <p>
        Thank you for confirming your email. Please @Html.ActionLink("Click here to Log in", "Login", "Account", routeValues: null, htmlAttributes: new { id = "loginLink" })
    </p>
</div>
```



## ResetPasswordConfirmation.cshtml

```
@{
    ViewBag.Title = "Reset password confirmation";
}

<hgroup class="title">
    <h1>@ViewBag.Title.</h1>
</hgroup>
<div>
    <p>
        Your password has been reset. Please @Html.ActionLink("click here to log in",
"Login", "Account", routeValues: null, htmlAttributes: new { id = "loginLink" })
    </p>
</div>
```

## Error.cshtml

```
@model System.Web.Mvc.HandleErrorInfo

 @{
    ViewBag.Title = "Error";
}

<h1 class="text-danger">Error.</h1>
<h2 class="text-danger">An error occurred while processing your request.</h2>
```

## Lockout.cshtml

```
@model System.Web.Mvc.HandleErrorInfo

 @{
    ViewBag.Title = "Locked Out";
}

<hgroup>
    <h1 class="text-danger">Locked out.</h1>
    <h2 class="text-danger">This account has been locked out, please try again later.</h2>
</hgroup>
```

## Login Master (\_Login.cshtml)

```
@{
    Layout = null;
}

<html lang="en">
<head>
```



```

<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<title>Login</title>

<!-- BASE CSS -->

<link href="~/Content/css/bootstrap.min.css" rel="stylesheet">
<link href="~/Content/css/style.css" rel="stylesheet">
<link href="~/Content/css/vendors.css" rel="stylesheet">
<link href="~/Content/css/all_icons.min.css" rel="stylesheet">
<!-- YOUR CUSTOM CSS -->
<link href="~/Content/css/custom.css" rel="stylesheet">

<style id="theia-sticky-sidebar-stylesheet-TSS">
    .theiaStickySidebar:after {
        content: "";
        display: table;
        clear: both;
    }
</style>
</head>

<body id="login_bg">
    <nav id="mm-menu" class="fake_menu mm-menu mm-offcanvas mm-keyboardfocus mm-hasnavbar-bottom-1 mm-pagedim-black" aria-hidden="true">
        <button class="mm-tabstart" tabindex="0" type="button" aria-hidden="true"></button>
        <div class="mm-panels"></div><div class="mm-navbars-bottom">
            <div class="mm-navbar mm-navbar-size-1"> </div>
        </div><button class="mm-tabend" tabindex="0" type="button" aria-hidden="true"></button>
    </nav>

    <nav id="menu" class="fake_menu"></nav>

    <div id="mm-0" class="mm-page mm-slideout">
        @RenderBody();
    </div>
    <!-- End Preload -->
    <!-- /login -->
    <!-- COMMON SCRIPTS -->
    <script src="~/Content/js/jquery-2.2.4.min.js"></script>
    <script src="~/Content/js/common_scripts.js"></script>
    <script src="~/Content/js/main.js"></script>
    <script src="~/Content/js/validate.js"></script>

</body>
</html>

```



## Register.cshtml

```
@model WebApp.Models.RegisterViewModel
@{
    ViewBag.Title = "Register";
    Layout = "~/Views/Shared/_Login.cshtml";
}



<div id="preloader" style="display: none;">
        <div data-loader="circle-side" style="display: none;"></div>
    </div><div id="login">
        <aside>
            <figure>
                <a href="/Home"><img src("~/Content/img/logo.png" width="149" height="42" data-retina="true" alt=""></a>
            </figure>
            <h3>Register Yourself</h3>
            @using (Html.BeginForm("Register", "Account", FormMethod.Post, new { @class = "form-horizontal", role = "form" }))
            {
                @Html.AntiForgeryToken()
                @Html.ValidationSummary("", new { @class = "text-danger" })

                <div class="form-group">

                    <span class="input">
                        @Html.TextBoxFor(m => m.Email, new { @class = "input_field" })
                        <label class="input_label">
                            <span class="input_label-content">Your Email</span>
                        </label>
                    </span>

                    <span class="input">
                        @Html.PasswordFor(m => m.Password, new { @class = "input_field" })
                        <label class="input_label">
                            <span class="input_label-content">Your password</span>
                        </label>
                    </span>

                    <span class="input">
                        @Html.PasswordFor(m => m.ConfirmPassword, new { @class = "input_field" })
                        <label class="input_label">
                            <span class="input_label-content">Confirm password</span>
                        </label>
                    </span>

                    <div id="pass-info" class="clearfix"></div>
                </div>
                <input type="submit" class="btn_1 rounded full-width add_top_30" value="Register to eLearning" />
                <div class="text-center add_top_10">Already have an account? <strong><a href="/Account/login">Sign In</a></strong></div>
            }
            <div class="copy">© 2018 eLearning</div>
        </aside>
    </div>


```



## Public Master (\_Layout.cshtml)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="eLearning a A way of learning">
    <meta name="author" content="Ansonika">
    <title>eLearning</title>
    <!-- Favicons-->
    <link rel="shortcut icon" href="~/Content/img/favicon.ico" type="image/x-icon">
    <link rel="apple-touch-icon" type="image/x-icon" href="~/Content/img/apple-touch-icon-57x57-precomposed.png">
        <link rel="apple-touch-icon" type="image/x-icon" sizes="72x72" href="~/Content/img/apple-touch-icon-72x72-precomposed.png">
            <link rel="apple-touch-icon" type="image/x-icon" sizes="114x114" href="~/Content/img/apple-touch-icon-114x114-precomposed.png">
                <link rel="apple-touch-icon" type="image/x-icon" sizes="144x144" href="~/Content/img/apple-touch-icon-144x144-precomposed.png">
                    <!-- BASE CSS -->
                    <link href="~/Content/css/bootstrap.min.css" rel="stylesheet">
                    <link href="~/Content/css/style.css" rel="stylesheet">
                    <link href="~/Content/css/vendors.css" rel="stylesheet">
                    <link href="~/Content/css/all_icons.min.css" rel="stylesheet">
                    <!-- YOUR CUSTOM CSS -->
                    <link href="~/Content/css/custom.css" rel="stylesheet">
    <style>
        section.header-section {
            height: 80px;
            background: #000;
        }
    </style>
</head>
<body>

    <div id="page">

        <header class="header menu_2">
            <div id="preloader"><div data-loader="circle-side"></div></div>
            <div id="logo">
                <a href="/home"></a>
            </div>
            <ul id="top_menu">
                <li><a href="/Account/login" class="login">Login</a></li>
                <li class="hidden_tablet"><a href="/admission" class="btn_1 rounded">Admission</a></li>
            </ul>
            <!-- /top_menu -->
            <a href="#menu" class="btn_mobile">
                <div class="hamburger hamburger--spin" id="hamburger">
                    <div class="hamburger-box">
                        <div class="hamburger-inner"></div>
                    </div>
                </div>
            </a>
            <nav id="menu" class="main-menu">
                <ul>
                    <li>
                        <span><a href="/Home">Home</a></span>

```



```

        </li>
        <li>
            <span><a href="/en/Courses">Courses</a></span>
        </li>
        <li>
            <span><a href="/en/Quizes">Quizzes</a></span>
        </li>
        <li>
            <span><a href="/en/Questions/Ask">Ask Question</a></span>
        </li>
        <li>
            <span><a href="/Home/Contacts">Contacts</a></span>
        </li>

    </ul>
</nav>

<div class="search-overlay-menu">
    <span class="search-overlay-close"><span class="closebt"><i class="ti-close"></i></span></span>
    <form role="search" id="searchform" method="get">
        <input value="" name="q" type="search" placeholder="Search..." />
        <button type="submit">
            <i class="icon_search"></i>
        </button>
    </form>
</div>
</header>
<!-- /header --&gt;

&lt;main&gt;
    @RenderBody()
&lt;/main&gt;
<!-- /main --&gt;
&lt;footer&gt;
    &lt;div class="container margin_120_95"&gt;
        &lt;div class="row"&gt;
            &lt;div class="col-lg-5 col-md-12 p-r-5"&gt;
                &lt;p&gt;&lt;img src "~/Content/img/logo.png" width="149" height="42" data-retina="true" alt=""&gt;&lt;/p&gt;
                &lt;p&gt;People wants something that they can carry and read, learn any time anywhere.
                    Digital Learning help user to read study materials and different books using laptop, mobile and tablets.
                &lt;/p&gt;
                &lt;div class="follow_us"&gt;
                    &lt;ul&gt;
                        &lt;li&gt;Follow us&lt;/li&gt;
                        &lt;li&gt;&lt;a href="#"&gt;&lt;i class="ti-facebook"&gt;&lt;/i&gt;&lt;/a&gt;&lt;/li&gt;
                        &lt;li&gt;&lt;a href="#"&gt;&lt;i class="ti-twitter-alt"&gt;&lt;/i&gt;&lt;/a&gt;&lt;/li&gt;
                        &lt;li&gt;&lt;a href="#"&gt;&lt;i class="ti-google"&gt;&lt;/i&gt;&lt;/a&gt;&lt;/li&gt;
                        &lt;li&gt;&lt;a href="#"&gt;&lt;i class="ti-pinterest"&gt;&lt;/i&gt;&lt;/a&gt;&lt;/li&gt;
                        &lt;li&gt;&lt;a href="#"&gt;&lt;i class="ti-instagram"&gt;&lt;/i&gt;&lt;/a&gt;&lt;/li&gt;
                    &lt;/ul&gt;
                &lt;/div&gt;
            &lt;/div&gt;
            &lt;div class="col-lg-3 col-md-6 ml-lg-auto"&gt;
</pre>

```



```

<h5>Useful links</h5>
<ul class="links">
    <li><a href="#">Home</a></li>
    <li><a href="#">Courses</a></li>
    <li><a href="#">Quizes</a></li>
    <li><a href="#">Ask Question</a></li>
    <li><a href="#">Contacts</a></li>
</ul>
</div>
<div class="col-lg-3 col-md-6">
    <h5>Contact with Us</h5>
    <ul class="contacts">
        <li><a href="whatsapp://+919506168811"><i class="ti-mobile"></i> + 91 950 616 8811</a></li>
        <li><a href="#"><i class="ti-email"></i> learn@ignougroup.com</a></li>
    </ul>
</div>
<!--/row-->
<hr>
<div class="row">
    <div class="col-md-8">
        <ul id="additional_links">
            <li><a href="#">Terms and conditions</a></li>
            <li><a href="#">Privacy</a></li>
        </ul>
    </div>
    <div class="col-md-4">
        <div id="copy">© 2018 DigiLearn | A Digital Learning Application</div>
        </div>
    </div>
</div>
<!--/footer-->
<!--/page -->
<!-- COMMON SCRIPTS -->

<script src="~/Content/js/jquery-2.2.4.min.js"></script>
<script src="~/Content/js/common_scripts.js"></script>
<script src="~/Content/js/main.js"></script>
<script src="~/Content/js/validate.js"></script>

</body>
</html>

```



## Public Pages Stylesheet

```
@import url(https://fonts.googleapis.com/css?family=Poppins:300,400,400i,500,600,700,800);
@import url(./css/menu_2.css);
#reccomended .owl-item,ul#grid_home li a.img_container img{
    -webkit-backface-visibility:hidden;
    -moz-backface-visibility:hidden;
    -ms-backface-visibility:hidden;
    -o-backface-visibility:hidden
}
.follow_us ul li:first-child,a.grid_item .info small,ul#top_menu li{
    letter-spacing:2px;
    text-transform:uppercase
}
#hero_video div,.table.cart-list td{
    vertical-align:middle
}
.filter_type ul,.list_articles ul,.profile
ul,.sbOptions,ol,ul,ul#cat_nav,ul#grid_home,ul.list_3,ul.list_ok,ul.list_teachers,ul.socia
l_teacher{
    list-style:none
}
.input_label,html *{
    -webkit-font-smoothing:antialiased;
    -moz-osx-font-smoothing:grayscale
}
.bs-wizard>.bs-wizard-step.disabled a.bs-wizard-dot,.input_label{
    pointer-events:none
}
#menu.fake_menu{
    display:none!important;
    visibility:hidden!important
}
#preloader{
    position:fixed;
    top:0;
    left:0;
    right:0;
    width:100%;
    height:100%;
    bottom:0;
    background-color:#fff;
    z-index:9999999999!important
}
[data-loader=circle-side]{
    position:absolute;
    width:50px;
    height:50px;
    top:50%;
    left:50%;
    margin-left:-25px;
    margin-top:-25px;
    -webkit-animation:circle infinite .95s linear;
    -moz-animation:circle infinite .95s linear;
    -o-animation:circle infinite .95s linear;
    animation:circle infinite .95s linear;
    border:2px solid #662d91;
    border-top-color:rgba(0,0,0,.2);
    border-right-color:rgba(0,0,0,.2);
    border-bottom-color:rgba(0,0,0,.2);
    -webkit-border-radius:100%;
    -moz-border-radius:100%;
    -ms-border-radius:100%;
    border-radius:100%
```



```

}

.main_title,.main_title_2{
    text-align:center;
    position:relative
}
@-webkit-keyframes circle{
    0%{
        -webkit-transform:rotate(0);
        -moz-transform:rotate(0);
        -ms-transform:rotate(0);
        -o-transform:rotate(0);
        transform:rotate(0)
    }
    100%{
        -webkit-transform:rotate(360deg);
        -moz-transform:rotate(360deg);
        -ms-transform:rotate(360deg);
        -o-transform:rotate(360deg);
        transform:rotate(360deg)
    }
}
@-moz-keyframes circle{
    0%{
        -webkit-transform:rotate(0);
        -moz-transform:rotate(0);
        -ms-transform:rotate(0);
        -o-transform:rotate(0);
        transform:rotate(0)
    }
    100%{
        -webkit-transform:rotate(360deg);
        -moz-transform:rotate(360deg);
        -ms-transform:rotate(360deg);
        -o-transform:rotate(360deg);
        transform:rotate(360deg)
    }
}
@-o-keyframes circle{
    0%{
        -webkit-transform:rotate(0);
        -moz-transform:rotate(0);
        -ms-transform:rotate(0);
        -o-transform:rotate(0);
        transform:rotate(0)
    }
    100%{
        -webkit-transform:rotate(360deg);
        -moz-transform:rotate(360deg);
        -ms-transform:rotate(360deg);
        -o-transform:rotate(360deg);
        transform:rotate(360deg)
    }
}
@keyframes circle{
    0%{
        -webkit-transform:rotate(0);
        -moz-transform:rotate(0);
        -ms-transform:rotate(0);
        -o-transform:rotate(0);
        transform:rotate(0)
    }
    100%{
        -webkit-transform:rotate(360deg);
        -moz-transform:rotate(360deg);

```



```

        -ms-transform:rotate(360deg);
        -o-transform:rotate(360deg);
        transform:rotate(360deg)
    }
}
body{
    background:#f8f8f8;
    font-size:.875rem;
    line-height:1.6;
    font-family:Poppins,Helvetica,sans-serif;
    color:#555
}
.box_grid a.wish_bt:after,.box_list a.wish_bt:after,.sbToggle:after,footer ul.links li a:after{
    font-family:ElegantIcons
}
h1,h2,h3,h4,h5,h6{
    color:#333
}
p{
    margin-bottom:30px
}
strong{
    font-weight:600
}
hr{
    margin:30px 0;
    border-color:#eddede
}
ol,ul{
    margin:0 0 25px;
    padding:0
}
.main_title{
    margin-bottom:45px
}
.main_title span{
    content:"";
    display:block;
    width:60px;
    height:3px;
    margin:auto auto 20px;
    opacity:1;
    background-color:#999;
    visibility:hidden;
    -webkit-transform:scaleX(0);
    transform:scaleX(0);
    -webkit-transition:all .6s;
    transition:all .6s
}
.main_title.animated span{
    visibility:visible;
    -webkit-transform:scaleX(1);
    transform:scaleX(1)
}
.main_title_2{
    margin-bottom:45px
}
.main_title_2 span{
    width:120px;
    height:2px;
    background-color:#e1e1e1;
    display:block;
    margin:auto
}

```



```

}

.main_title_2 span em{
    width:60px;
    height:2px;
    background-color:#662d91;
    display:block;
    margin:auto
}
@media (max-width:767px){
    .main_title_2{
        margin-bottom:10px
    }
    .main_title_2 h2{
        font-size:26px;
        font-size:1.625rem
    }
}
.main_title_2 h2{
    margin:25px 0 0
}
.main_title_2 p{
    font-size:24px;
    font-size:1.5rem;
    font-weight:300;
    line-height:1.4
}
@media (max-width:767px){
    .main_title_2 p{
        font-size:18px;
        font-size:1.125rem
    }
}
h3.slide_typo{
    white-space:nowrap;
    color:#fff!important;
    font-weight:400;
    font-size:60px;
    font-size:3.75rem
}
.slide_typo_2{
    color:#fff;
    font-size:24px;
    font-size:1.5rem;
    white-space:nowrap;
    text-transform:uppercase
}
a{
    color:#3f9fff;
    text-decoration:none;
    -moz-transition:all .3s ease-in-out;
    -o-transition:all .3s ease-in-out;
    -webkit-transition:all .3s ease-in-out;
    -ms-transition:all .3s ease-in-out;
    transition:all .3s ease-in-out;
    outline:0
}
.btn_1,a.btn_1,a.btn_explore{
    text-align:center;
    -moz-transition:all .3s ease-in-out;
    -o-transition:all .3s ease-in-out;
    -webkit-transition:all .3s ease-in-out;
    -ms-transition:all .3s ease-in-out
}
a:focus,a:hover{

```



```

color:#111;
text-decoration:none;
outline:0
}
.btn_1,a.btn_1{
border:none;
background:#3f9fff;
outline:0;
cursor:pointer;
display:inline-block;
text-decoration:none;
padding:15px 30px;
color:#fff;
font-weight:600;
line-height:1;
transition:all .3s ease-in-out;
-webkit-border-radius:5px;
-moz-border-radius:5px;
-ms-border-radius:5px;
border-radius:5px
}
.btn_1:hover,a.btn_1:hover{
background:#0054a6;
color:#fff
}
.btn_1.full-width,a.btn_1.full-width{
display:block;
width:100%;
text-align:center;
margin-bottom:5px
}
.btn_1.medium,a.btn_1.medium{
font-size:16px;
font-size:1rem;
padding:18px 40px
}
.btn_1.rounded,a.btn_1.rounded{
-webkit-border-radius:25px!important;
-moz-border-radius:25px!important;
-ms-border-radius:25px!important;
border-radius:25px!important
}
.btn_1.outline,a.btn_1.outline{
border:2px solid #3f9fff;
color:#3f9fff;
padding:11px 40px;
background:0 0
}
.btn_1.outline:hover,a.btn_1.outline:hover{
background:#0054a6;
color:#fff;
border-color:#0054a6
}
.a.btn_explore{
background:#fff;
display:block;
width:60px;
height:60px;
-webkit-border-radius:50%;
-moz-border-radius:50%;
-ms-border-radius:50%;
border-radius:50%;
line-height:65px;
position:absolute;

```



```

bottom:40px;
left:50%;
margin-left:-30px;
font-size:24px;
font-size:1.5rem;
transition:all .3s ease-in-out
}
.button_plus,.closebt{
width:30px;
height:30px
}
a#moveDown,a.btn_explore:hover{
bottom:30px
}
.closebt,header.header.sticky{
-moz-transition:all .3s ease-in-out;
-o-transition:all .3s ease-in-out;
-webkit-transition:all .3s ease-in-out;
-ms-transition:all .3s ease-in-out
}
.closebt{
transition:all .3s ease-in-out;
cursor:pointer;
position:absolute;
font-size:24px;
font-size:1.5rem;
top:10px;
right:15px;
color:#fff;
opacity:.6
}
.closebt:hover{
-webkit-transform:rotate(90deg);
-moz-transform:rotate(90deg);
-ms-transform:rotate(90deg);
-o-transform:rotate(90deg);
transform:rotate(90deg);
opacity:1
}
.btn_home_align{
text-align:right
}
.button_plus{
position:relative
}
.button_plus:after,.button_plus:before{
content:"";
position:absolute;
background-color:#fff;
transition:transform .25s ease-out
}
.button_plus:before{
top:0;
left:50%;
width:4px;
height:100%;
margin-left:-2px
}
.button_plus:after{
top:50%;
left:0;
width:100%;
height:4px;
margin-top:-2px
}

```



```

}
.button_plus:hover{
  cursor:pointer
}
.button_plus:hover:before{
  transform:rotate(90deg)
}
.button_plus:hover:after{
  transform:rotate(180deg)
}
}
header{
  position:fixed;
  left:0;
  top:0;
  width:100%;
  padding:20px 30px;
  z-index:99999;
  border-bottom:1px solid rgba(255,255,255,0)
}
@media (max-width:767px){
  .btn_home_align{
    text-align:center
  }
  header{
    padding:15px 15px 5px
  }
}
header.header.sticky{
  transition:all .3s ease-in-out;
  background:#480048;
  background:-webkit-linear-gradient(left,#480048,#C04848);
  background:linear-gradient(to right,#480048,#C04848);
  padding:15px 20px
}
#newsletter input[type=submit],footer ul li a{
  -moz-transition:all .3s ease-in-out;
  -o-transition:all .3s ease-in-out;
  -webkit-transition:all .3s ease-in-out;
  -ms-transition:all .3s ease-in-out
}
header.header.sticky_menu_active{
  border-bottom:1px solid rgba(255,255,255,.2)
}
ul#top_menu{
  float:right;
  margin:0;
  font-size:13px;
  font-size:.8125rem
}
ul#top_menu li{
  display:inline-block;
  margin:2px 20px 0 0;
  font-weight:600;
  line-height:1
}
ul#top_menu li:last-child{
  margin-right:0;
  position:relative;
  top:0
}
@media (max-width:991px){
  header.header.sticky{
    padding:15px 15px 5px
  }
}

```



```

header.header.sticky_menu_active{
    padding:15px 15px 0;
    background:#480048;
    background:-webkit-linear-gradient(left,#480048,#C04848);
    background:linear-gradient(to right,#480048,#C04848)
}
ul#top_menu li{
    margin:5px 0 0 10px
}
ul#top_menu li a{
    color:#fff
}
ul#top_menu li a:hover{
    color:#fff;
    opacity:.7
}
ul#top_menu li a.login,ul#top_menu li a.search-overlay-menu-btn{
    display:block;
    width:25px;
    height:25px;
    text-indent:-9999px;
    position:relative;
    top:-5px
}
ul#top_menu li a.login:before,ul#top_menu li a.search-overlay-menu-btn:before{
    font-family:ElegantIcons;
    font-size:24px;
    font-size:1.5rem;
    color:#fff;
    text-indent:0;
    position:absolute;
    left:0;
    top:0;
    font-weight:400;
    line-height:1
}
#hero_in,footer ul li a,main{
    position:relative
}
ul#top_menu li a.login:before{
    content:"\7e"
}
ul#top_menu li a.search-overlay-menu-btn:before{
    content:"\55"
}
ul#top_menu li a.btn_1{
    background-color:#FFC107;
    color:#333;
    padding:13px 30px
}
ul#top_menu li a.btn_1:hover{
    background-color:#3f9fff;
    color:#fff;
    opacity:1
}
#logo{
    float:left;
    animation-delay:.2s;
    -webkit-animation-delay:.2s;
    -moz-animation-delay:.2s
}
@media (max-width:991px){
    #logo{

```



```

        float:none;
        position:absolute;
        top:0;
        left:0;
        width:100%;
        text-align:center
    }
    #logo img{
        width:auto;
        height:30px;
        margin-top:10px
    }
}
#hero_in{
    width:100%;
    height:450px;
    overflow:hidden;
    text-align:center;
    color:#fff
}
@media (max-width:767px){
    #hero_in{
        height:350px
    }
}
#hero_in.general{
    border-bottom:5px solid #FFC107
}
#hero_in.general:before{
    background:url(..../img/bg_general.jpg) center center no-repeat;
    -webkit-background-size:cover;
    -moz-background-size:cover;
    -o-background-size:cover;
    background-size:cover
}
#hero_in.general .wrapper{
    background-color:#372675;
    background-color:rgba(57,39,121,.8)
}
#hero_in.courses:before{
    background:url(..../img/bg_courses.jpg) center center no-repeat;
    -webkit-background-size:cover;
    -moz-background-size:cover;
    -o-background-size:cover;
    background-size:cover
}
#hero_in.courses .wrapper{
    background-color:#372675;
    background-color:rgba(57,39,121,.8)
}
#hero_in.contacts:before{
    background:url(..../img/bg_contacts.jpg) center center no-repeat;
    -webkit-background-size:cover;
    -moz-background-size:cover;
    -o-background-size:cover;
    background-size:cover
}
#hero_in.contacts .wrapper{
    background-color:#000;
    background-color:rgba(0,0,0,.8)
}
#hero_in:before{
    animation:pop-in 5s .3s cubic-bezier(0,.5,0,1) forwards;
    content:"";

```



```

    opacity:0;
    position:absolute;
    top:0;
    right:0;
    bottom:0;
    left:0;
    z-index:-1;
    -webkit-filter:grayscale(1);
    filter:grayscale(1)
}
#hero_in .wrapper{
    display:flex;
    flex-direction:column;
    justify-content:center;
    height:100%
}
#hero_in .wrapper h1{
    color:#fff;
    font-size:42px;
    font-size:2.625rem;
    text-transform:uppercase;
    font-weight:700;
    margin-top:30px
}
@media (max-width:767px){
    #hero_in .wrapper h1{
        font-size:32px;
        font-size:2rem
    }
}
#hero_in .wrapper h1 span{
    width:80px;
    height:4px;
    background-color:#FFC107;
    display:block;
    margin:auto auto 20px;
    border-radius:10px;
    opacity:1
}
#hero_in .wrapper p{
    font-weight:300;
    margin-bottom:0;
    font-size:24px;
    font-size:1.5rem
}
#hero_in .wrapper p strong{
    font-weight:600
}
@keyframes pop-in{
    0%{
        opacity:0
    }
    100%{
        opacity:1;
        -webkit-transform:scale(1.1);
        -moz-transform:scale(1.1);
        -ms-transform:scale(1.1);
        -o-transform:scale(1.1);
        transform:scale(1.1)
    }
}
ul#top_menu{
    animation-delay:.6s;
    -webkit-animation-delay:.6s;

```



```

        -moz-animation-delay:.6s
    }
    #hero_in h1{
        animation-delay:.8s;
        -webkit-animation-delay:.8s;
        -moz-animation-delay:.8s
    }
    #hero_in p{
        animation-delay:1.5s;
        -webkit-animation-delay:1.5s;
        -moz-animation-delay:1.5s;
        animation-duration:.3s;
        -webkit-animation-duration:.3s;
        -moz-animation-duration:.3s
    }
    #hero_in a.btn_explore{
        animation-delay:2s;
        -webkit-animation-delay:2s;
        -moz-animation-delay:2s
    }
    main{
        background-color:#f8f8f8;
        z-index:1
    }
    footer{
        background-color:#121921;
        color:#fbfbfb;
        color:rgba(255,255,255,.7)
    }
    footer h5{
        color:#fff;
        margin:25px 0
    }
    @media (max-width:575px){
        footer h5{
            margin:25px 0 10px
        }
    }
    footer ul li a i,footer ul.contacts li a i{
        margin-right:10px
    }
    footer ul li{
        margin-bottom:5px
    }
    footer ul li a{
        transition:all .3s ease-in-out;
        display:inline-block;
        color:#fff;
        opacity:.7
    }
    footer ul li a:hover{
        color:#3f9fff;
        opacity:1
    }
    footer ul li a i{
        color:#fff
    }
    footer ul.links li a:hover{
        -webkit-transform:translate(5px,0);
        -moz-transform:translate(5px,0);
        -ms-transform:translate(5px,0);
        -o-transform:translate(5px,0);
        transform:translate(5px,0)
    }
}

```



```

footer ul.links li a:hover:after{
    opacity:1;
    color:#3f9fff
}
footer ul.links li a:after{
    content:"\24";
    position:absolute;
    margin-left:5px;
    top:1px;
    opacity:0;
    -moz-transition:all .5s ease;
    -o-transition:all .5s ease;
    -webkit-transition:all .5s ease;
    -ms-transition:all .5s ease;
    transition:all .5s ease
}
footer hr{
    opacity:.1
}
footer #copy{
    text-align:right;
    font-size:13px;
    font-size:.8125rem;
    opacity:.7
}
@media (max-width:767px){
    footer #copy{
        text-align:left;
        margin:5px 0 20px
    }
}
footer.fake-footer{
    display:none;
    visibility:hidden
}
#newsletter .loader{
    position:absolute;
    right:-20px;
    top:11px;
    color:#ffff;
    font-size:12px;
    font-size:.75rem
}
#hero_video,#newsletter .form-group,.header-video{
    position:relative
}
#newsletter h6{
    color:#ffff;
    margin:15px 0
}
#newsletter input[type=email]{
    border:0;
    height:40px;
    -webkit-border-radius:0;
    -moz-border-radius:0;
    -ms-border-radius:0;
    border-radius:0;
    padding-left:15px;
    font-size:14px;
    font-size:.875rem;
    background-color:#1b2632;
    color:#ffff
}
#newsletter input[type=submit]{

```



```

position: absolute;
right: 0;
color: #ffff;
font-size: 13px;
font-size: .8125rem;
font-weight: 600;
top: 0;
border: 0;
padding: 0 12px;
height: 40px;
cursor: pointer;
-webkit-border-radius: 0 3px 3px 0;
-moz-border-radius: 0 3px 3px 0;
-ms-border-radius: 0 3px 3px 0;
border-radius: 0 3px 3px 0;
background-color: #92278f;
transition: all .3s ease-in-out;
outline: 0
}
#custom-search-input input[type=submit], ul#additional_links li a{
-moz-transition: all .3s ease-in-out;
-o-transition: all .3s ease-in-out;
-webkit-transition: all .3s ease-in-out;
-ms-transition: all .3s ease-in-out
}
#newsletter input[type=submit]:hover{
background-color: #392779
}
.follow_us{
margin-top: 15px;
animation-delay: 1.1s;
-webkit-animation-delay: 1.1s;
-moz-animation-delay: 1.1s
}
.follow_us h5{
color: #ffff
}
.follow_us ul li{
display: inline-block;
margin-right: 10px;
font-size: 20px;
font-size: 1.25rem
}
.follow_us ul li:first-child{
color: #ffff;
font-weight: 500;
font-size: 13px;
font-size: .8125rem
}
.follow_us ul li a{
color: #ffff;
opacity: .7
}
.follow_us ul li a:hover, ul#additional_links li a:hover{
opacity: 1
}
ul#additional_links{
margin: 0;
color: #555;
font-size: 13px;
font-size: .8125rem
}
ul#additional_links li{
display: inline-block;

```



```

        margin-right:15px
    }
    ul#additional_links li:first-child{
        margin-right:20px
    }
    ul#additional_links li:last-child:after{
        content:""
    }
    ul#additional_links li a{
        color:#fff;
        opacity:.5;
        transition:all .3s ease-in-out
    }
    ul#additional_links li:after{
        content:"|";
        font-weight:300;
        position:relative;
        left:10px
    }
    .wow{
        visibility:hidden
    }
    a#moveDown{
        position:fixed;
        right:30px;
        z-index:9999;
        font-size:32px;
        font-size:2rem;
        color:#fff;
        animation-duration:6s;
        -webkit-animation-duration:6s;
        -moz-animation-duration:6s;
        animation-iteration-count:infinite;
        -webkit-animation-iteration-count:infinite;
        -moz-animation-iteration-count:infinite
    }
    .header-video{
        overflow:hidden;
        background:#000
    }
    @media (max-width:991px){
        .header-video{
            height:450px!important
        }
    }
    #hero_video,iframe{
        height:100%;
        width:100%
    }
    #hero_video{
        background-size:cover;
        color:#fff;
        font-size:16px;
        font-size:1rem;
        display:table;
        z-index:99;
        text-align:center
    }
    #hero_video div{
        display:table-cell;
        text-align:center;
        padding:0 10%
    }
    #hero_video div h3{

```



```

color:#fff;
font-size:52px;
font-size:3.25rem;
margin:0;
text-transform:uppercase;
font-weight:800
}
@media (max-width:767px){
#hero_video div h3{
    font-size:21px;
    font-size:1.3125rem
}
}
@media (max-width:575px){
#hero_video div h3{
    font-size:23px;
    font-size:1.4375rem
}
}
#hero_video div p{
    font-weight:300;
    margin:10px 0 0;
    padding:0 20%;
    font-size:24px;
    font-size:1.5rem;
    line-height:1.4
}
#hero_video div p strong{
    font-weight:600
}
video{
    position:absolute;
    top:0;
    bottom:0;
    left:0;
    right:0;
    opacity:.4;
    width:100%
}
.header-video--media,.teaser-video{
    width:100%;
    height:auto
}
@media (max-width:1024px){
#hero_video{
    background:url(..../img/bg_home_fallback_video.jpg) #662d91;
    -webkit-background-size:cover;
    -moz-background-size:cover;
    -o-background-size:cover;
    background-size:cover
}
#hero_video div{
    background-color:#000;
    background-color:rgba(0,0,0,.7)
}
.header-video{
    background:0 0
}
}
.hero_single{
    width:100%;
    height:100vh;
    position:relative;
    overflow:hidden;
}

```



```

    text-align:center;
    margin:0;
    color:#ffff
}
@media (max-width:767px){
    #hero_video div p{
        padding:0;
        font-size:18px;
        font-size:1.125rem
    }
    .hero_single{
        height:450px
    }
}
.hero_single:before{
    background:url(..../img/home_section_2.jpg) center center no-repeat fixed;
    -webkit-background-size:cover;
    -moz-background-size:cover;
    -o-background-size:cover;
    background-size:cover
}
.hero_single .wrapper{
    background-color:#632c8d;
    background-color:rgba(102,45,145,.8);
    display:flex;
    flex-direction:column;
    justify-content:center;
    height:100%
}
.hero_single.start_bg_zoom:before{
    animation:pop-in 5s .3s cubic-bezier(0,.5,0,1) forwards;
    content:"";
    opacity:0;
    position:absolute;
    top:0;
    right:0;
    bottom:0;
    left:0;
    z-index:-1
}
.hero_single .wrapper h3{
    color:#ffff;
    font-size:52px;
    font-size:3.25rem;
    margin:0;
    text-transform:uppercase;
    font-weight:700
}
@media (max-width:767px){
    .hero_single .wrapper h3{
        font-size:21px;
        font-size:1.3125rem
    }
}
@media (max-width:575px){
    .hero_single .wrapper h3{
        font-size:23px;
        font-size:1.4375rem
    }
}
.hero_single .wrapper p{
    font-weight:300;
    margin:10px 0 0;
    padding:0 20%;
}

```



```

        font-size:24px;
        font-size:1.5rem;
        line-height:1.4
    }
.hero_single .wrapper p strong{
    font-weight:600
}
.hero_single.short,.hero_single.version_2{
    height:600px
}
@media (max-width:767px){
    .hero_single .wrapper p{
        padding:0;
        font-size:18px;
        font-size:1.125rem
    }
    .hero_single.version_2{
        height:450px
    }
}
.hero_single.version_2:before{
    background:url(..../img/home_section_1.jpg) center center no-repeat fixed;
    -webkit-background-size:cover;
    -moz-background-size:cover;
    -o-background-size:cover;
    background-size:cover
}
.hero_single.version_2 .wrapper{
    background-color:#372675;
    background-color:rgba(57,39,121,.8)
}
.hero_single.version_2 .wrapper h3{
    font-size:42px;
    font-size:2.625rem
}
@media (max-width:767px){
    .hero_single.version_2 .wrapper h3{
        font-size:21px;
        font-size:1.3125rem
    }
}
@media (max-width:575px){
    .hero_single.version_2 .wrapper h3{
        font-size:23px;
        font-size:1.4375rem
    }
}
.hero_single.version_2 .wrapper p{
    font-weight:400;
    margin:5px 0 0;
    padding:0;
    font-size:21px;
    font-size:1.3125rem
}
.hero_single.version_2 .wrapper p strong{
    font-weight:600
}
@media (max-width:767px){
    .hero_single.version_2 .wrapper p{
        padding:0;
        font-size:16px;
        font-size:1rem
    }
}

```



```

#custom-search-input{
    padding:0;
    width:600px;
    margin:20px auto 0;
    position:relative
}
@media (max-width:991px){
    #custom-search-input{
        width:auto
    }
}
#custom-search-input .search-query{
    width:100%;
    height:50px;
    padding-left:20px;
    border:0;
    -webkit-border-radius:3px;
    -moz-border-radius:3px;
    -ms-border-radius:3px;
    border-radius:3px;
    font-size:16px;
    font-size:1rem;
    color:#333
}
#custom-search-input .search-query:focus{
    outline:0
}
@media (max-width:575px){
    #custom-search-input .search-query{
        padding-left:15px
    }
}
#custom-search-input input[type=submit]{
    position:absolute;
    transition:all .3s ease-in-out;
    right:-1px;
    color:#ffff;
    font-weight:600;
    font-size:14px;
    font-size:.875rem;
    top:0;
    border:0;
    padding:0 25px;
    height:50px;
    cursor:pointer;
    outline:0;
    -webkit-border-radius:0 3px 3px 0;
    -moz-border-radius:0 3px 3px 0;
    -ms-border-radius:0 3px 3px 0;
    border-radius:0 3px 3px 0;
    background-color:#92278f
}
#custom-search-input input[type=submit]:hover{
    background-color:#FFC107
}
@media (max-width:575px){
    #custom-search-input input[type=submit]{
        text-indent:-999px;
        background:url(..../img/search.svg) center center no-repeat #92278f
    }
}
.features{
    background-color:#121921;
    color:#ffff;
}

```



```

padding:30px 0;
text-align:center
}
.features ul{
margin:0;
padding:0
}
.features ul li{
float:left;
width:33.33%;
position:relative
}
.features ul li span{
opacity:.7
}
.features ul li h4{
font-size:18px;
font-size:1.125rem;
margin:0;
color:#FFC107
}
.features ul li i{
color:#fff;
font-size:46px;
font-size:2.875rem;
margin-bottom:10px
}
@media (max-width:767px){
.features ul li{
width:100%;
float:none;
margin-bottom:35px
}
.features ul li:last-child{
margin-bottom:5px
}
}
ul#grid_home{
padding:0;
margin:0;
width:100%
}
ul#grid_home li{
position:relative;
width:33.33%;
float:left
}
ul#grid_home li:nth-child(1) .short_info{
background-color:#372675;
background-color:rgba(57,39,121,.8)
}
ul#grid_home li:nth-child(2) .short_info{
background-color:#8e268b;
background-color:rgba(146,39,143,.8)
}
ul#grid_home li:nth-child(3) .short_info{
background-color:#392779
}
ul#grid_home li a.img_container{
position:relative;
overflow:hidden;
z-index:0;
display:block;
width:100%
}

```



```

        height:280px
    }
    @media (max-width:767px){
        ul#grid_home li{
            width:100%;
            float:none
        }
        ul#grid_home li a.img_container{
            width:100%;
            float:none;
            height:250px
        }
    }
    ul#grid_home li a.img_container img{
        position:absolute;
        left:50%;
        top:50%;
        -webkit-transform:translate(-50%,-50%) scale(1);
        -moz-transform:translate(-50%,-50%) scale(1);
        -ms-transform:translate(-50%,-50%) scale(1);
        -o-transform:translate(-50%,-50%) scale(1);
        transform:translate(-50%,-50%) scale(1);
        -moz-transition:all .3s ease-in-out;
        -o-transition:all .3s ease-in-out;
        -webkit-transition:all .3s ease-in-out;
        -ms-transition:all .3s ease-in-out;
        transition:all .3s ease-in-out;
        width:100%;
        max-width:100%;
        height:auto;
        backface-visibility:hidden;
        -webkit-filter:grayscale(1);
        filter:grayscale(1)
    }
    @media (max-width:1199px){
        ul#grid_home li a.img_container img{
            height:100%;
            width:auto;
            max-width:inherit
        }
    }
    @media (max-width:767px){
        ul#grid_home li a.img_container img{
            width:100%;
            height:auto
        }
    }
    ul#grid_home li a.img_container:hover img{
        -webkit-transform:translate(-50%,-50%) scale(1.1);
        -moz-transform:translate(-50%,-50%) scale(1.1);
        -ms-transform:translate(-50%,-50%) scale(1.1);
        -o-transform:translate(-50%,-50%) scale(1.1);
        transform:translate(-50%,-50%) scale(1.1)
    }
    ul#grid_home li a.img_container .short_info{
        position:absolute;
        left:0;
        top:0;
        width:100%;
        box-sizing:border-box;
        color:#fff;
        padding:0 25px;
        height:100%;
        display:flex;

```



```

flex-direction:column;
justify-content:center;
-moz-transition:all .4s ease;
-o-transition:all .4s ease;
-webkit-transition:all .4s ease;
-ms-transition:all .4s ease;
transition:all .4s ease;
text-align:center
}
ul#grid_home li a.img_container .short_info h3{
color:#fff;
font-size:24px;
font-size:1.5rem;
margin:0 0 5px;
padding:0;
font-weight:400
}
ul#grid_home li a.img_container .short_info h3 strong{
font-weight:700;
text-transform:uppercase;
display:block;
font-size:28px;
font-size:1.75rem
}
ul#grid_home li a.img_container .short_info div{
position:absolute;
bottom:0;
left:0;
-webkit-transform:translateY(-25px);
-moz-transform:translateY(-25px);
-ms-transform:translateY(-25px);
-o-transform:translateY(-25px);
transform:translateY(-25px);
text-align:center;
width:100%;
visibility:hidden;
display:table;
opacity:0;
-webkit-transition:all .6s;
transition:all .6s
}
ul#grid_home li a.img_container .short_info div .btn_1{
padding:10px 22px;
font-size:13px;
font-size:.8125rem
}
@media (max-width:767px){
ul#grid_home li a.img_container .short_info h3{
margin-top:-20px
}
ul#grid_home li a.img_container .short_info div{
opacity:1;
visibility:visible;
-webkit-transform:translateY(-40px);
-moz-transform:translateY(-40px);
-ms-transform:translateY(-40px);
-o-transform:translateY(-40px);
transform:translateY(-40px)
}
}
ul#grid_home li a.img_container .short_info: hover div{
visibility:visible;
opacity:1;
-webkit-transform:translateY(-40px);

```



```

        -moz-transform:translateY(-40px);
        -ms-transform:translateY(-40px);
        -o-transform:translateY(-40px);
        transform:translateY(-40px)
    }
    ul#grid_home li:last-child .img_container:hover .short_info .btn_1{
        background-color:#FFC107;
        -moz-transition:all .5s ease;
        -o-transition:all .5s ease;
        -webkit-transition:all .5s ease;
        -ms-transition:all .5s ease;
        transition:all .5s ease
    }
    ul#grid_home li:last-child .img_container:hover .short_info .btn_1:hover{
        background-color:#3f9fff
    }
    #reccomended{
        margin-top:40px
    }
    @media (max-width:767px){
        #reccomended{
            margin-top:0
        }
    }
    #reccomended .item{
        margin:0 15px
    }
    #reccomended .owl-item{
        opacity:.5;
        transform:scale(.85);
        backface-visibility:hidden;
        -webkit-transform:translateZ(0) scale(.85,.85);
        transition:all .3s ease-in-out 0s;
        overflow:hidden
    }
    #reccomended .owl-item.active.center,a.box_news figure img{
        -webkit-backface-visibility:hidden;
        -moz-backface-visibility:hidden;
        -ms-backface-visibility:hidden;
        -o-backface-visibility:hidden
    }
    a.box_news figure img,a.box_news:hover h4{
        -moz-transition:all .3s ease-in-out;
        -o-transition:all .3s ease-in-out;
        -webkit-transition:all .3s ease-in-out;
        -ms-transition:all .3s ease-in-out
    }
    #reccomended .owl-item.active.center{
        opacity:1;
        backface-visibility:hidden;
        -webkit-transform:translateZ(0) scale(1,1);
        transform:scale(1)
    }
    #reccomended .owl-item.active.center .item .title h4,#reccomended .owl-item.active.center
    .item .views{
        opacity:1
    }
    .owl-theme .owl-dots{
        margin-top:25px!important
    }
    @media (max-width:767px){
        .owl-theme .owl-dots{
            margin-top:0!important;
            margin-bottom:20px
    }

```



```

        }
    }
.owl-theme .owl-dots .owl-dot.active span,.owl-theme .owl-dots .owl-dot:hover span{
    background:#3f9fff!important
}
.owl-theme .owl-dots .owl-dot span{
    width:13px!important;
    height:13px!important
}
.a.box_news{
    position:relative;
    display:block;
    padding-left:230px;
    color:#555;
    margin-bottom:30px;
    min-height:150px
}
.a.box_news figure{
    width:200px;
    height:150px;
    overflow:hidden;
    position:absolute;
    left:0;
    top:0
}
@media (max-width:767px){
    a.box_news{
        min-height:inherit;
        padding-left:0
    }
    a.box_news figure{
        position:relative;
        width:auto
    }
}
.a.box_news figure img{
    width:250px;
    height:auto;
    position:absolute;
    left:50%;
    top:50%;
    -webkit-transform:translate(-50%,-50%) scale(1.1);
    -moz-transform:translate(-50%,-50%) scale(1.1);
    -ms-transform:translate(-50%,-50%) scale(1.1);
    -o-transform:translate(-50%,-50%) scale(1.1);
    transform:translate(-50%,-50%) scale(1.1);
    transition:all .3s ease-in-out;
    backface-visibility:hidden
}
#carousel .owl-item,.animated a.grid_item figure img{
    -webkit-backface-visibility:hidden;
    -moz-backface-visibility:hidden;
    -ms-backface-visibility:hidden;
    -o-backface-visibility:hidden
}
@media (max-width:767px){
    a.box_news figure img{
        width:100%;
        max-width:100%;
        height:auto
    }
}
.a.box_news figure figcaption{
    background-color:#FFC107;

```



```

text-align:center;
position:absolute;
right:0;
top:0;
display:inline-block;
color:#ffff;
width:60px;
height:60px;
padding:15px;
line-height:1
}
.a.box_news figure figcaption strong{
display:block;
font-size:21px;
font-size:1.3125rem
}
.a.box_news h4{
font-size:18px;
font-size:1.125rem;
margin-bottom:10px
}
@media (max-width:767px){
.a.box_news h4{
margin-bottom:5px
}
}
.a.box_news:hover img{
-webkit-transform:translate(-50%, -50%) scale(1);
-moz-transform:translate(-50%, -50%) scale(1);
-ms-transform:translate(-50%, -50%) scale(1);
-o-transform:translate(-50%, -50%) scale(1);
transform:translate(-50%, -50%) scale(1)
}
.a.box_news:hover h4{
color:#3f9fff;
transition:all .3s ease-in-out
}
.box_grid figure a img,a.grid_item figure img{
-o-transition:all .3s ease-in-out;
-webkit-transition:all .3s ease-in-out;
-ms-transition:all .3s ease-in-out
}
.a.box_news ul{
margin:0;
padding:10px 0 0
}
@media (max-width:767px){
.a.box_news ul{
padding:0
}
}
.a.box_news ul li{
display:inline-block;
font-weight:600;
text-transform:uppercase;
color:#999;
font-size:12px;
font-size:.75rem;
margin-right:5px
}
.a.box_news ul li:last-child{
margin-right:0
}
.a.box_news ul li:last-child:after{

```



```

        content: '';
        margin: 0
    }

    @keyframes reveal_top_to_bottom{
        0%{
            -webkit-transform:translateY(-100%);
            -moz-transform:translateY(-100%);
            -ms-transform:translateY(-100%);
            -o-transform:translateY(-100%);
            transform:translateY(-100%)
        }
        50%{
            -webkit-transform:translateY(0);
            -moz-transform:translateY(0);
            -ms-transform:translateY(0);
            -o-transform:translateY(0);
            transform:translateY(0)
        }
        100%{
            -webkit-transform:translateY(100%);
            -moz-transform:translateY(100%);
            -ms-transform:translateY(100%);
            -o-transform:translateY(100%);
            transform:translateY(100%)
        }
    }
    @keyframes color{
        from{
            opacity:0
        }
        to{
            opacity:1
        }
    }
    .search-overlay-menu{
        position:fixed;
        top:0;
        left:0;
        width:100%;
        height:100%;
        background-color:#000;
        background-color:rgba(0,0,0,.9);
        -webkit-transition:transform .3s ease 0s,opacity .3s ease 0s,visibility 0s ease .3s;
        -moz-transition:transform .3s ease 0s,opacity .3s ease 0s,visibility 0s ease .3s;
        -ms-transition:transform .3s ease 0s,opacity .3s ease 0s,visibility 0s ease .3s;
        -o-transition:transform .3s ease 0s,opacity .3s ease 0s,visibility 0s ease .3s;
        transition:transform .3s ease 0s,opacity .3s ease 0s,visibility 0s ease .3s;
        opacity:0;
        visibility:hidden;
        z-index:999999;
        text-align:center
    }
    .search-overlay-menu.open{
        opacity:1;
        visibility:visible;
        -webkit-transition:opacity .3s ease 0s,transform .3s ease 0s;
        -moz-transition:opacity .3s ease 0s,transform .3s ease 0s;
        -ms-transition:opacity .3s ease 0s,transform .3s ease 0s;
        -o-transition:opacity .3s ease 0s,transform .3s ease 0s;
        transition:opacity .3s ease 0s,transform .3s ease 0s
    }
    .search-overlay-menu.open form{
        -webkit-transform:translateY(0);

```



```

        -moz-transform:translateY(0);
        -ms-transform:translateY(0);
        -o-transform:translateY(0);
        transform:translateY(0)
    }
.search-overlay-menu form{
    height:70px;
    position:absolute;
    top:50%;
    margin:-35px auto 0;
    width:60%;
    left:20%;
    -webkit-transform:translateY(20px);
    -moz-transform:translateY(20px);
    -ms-transform:translateY(20px);
    -o-transform:translateY(20px);
    transform:translateY(20px);
    -moz-transition:all .3s ease-in-out 0s;
    -o-transition:all .3s ease-in-out 0s;
    -webkit-transition:all .3s ease-in-out 0s;
    -ms-transition:all .3s ease-in-out 0s;
    transition:all .3s ease-in-out 0s
}
.search-overlay-menu form button{
    background:0 0;
    border:none;
    font-size:32px;
    font-size:2rem;
    position:absolute;
    right:0;
    top:15px;
    color:#c6c6c6;
    outline:0;
    cursor:pointer
}
@media (max-width:767px){
    .hidden_mobile{
        display:none!important
    }
    .search-overlay-menu form button{
        font-size:24px;
        font-size:1.5rem;
        top:8px
    }
    .search-overlay-menu form{
        width:80%;
        left:10%;
        height:50px;
        margin:-25px auto 0
    }
}
.search-overlay-menu input[type=search]{
    height:100%;
    width:100%;
    color:#ffff;
    background:0 0;
    border-top:0 solid transparent;
    border-bottom:2px solid #909090!important;
    border-left:0 solid transparent;
    border-right:0 solid transparent;
    font-size:42px;
    font-size:2.625rem;
    font-weight:300;
    outline:0;
}

```



```

        padding:10px 0
    }
    @media (max-width:767px){
        .search-overlay-menu input[type=search]{
            font-size:24px;
            font-size:1.5rem
        }
    }
    #cookie-bar{
        background:rgba(0,0,0,.8);
        height:auto;
        line-height:24px;
        color:#eee;
        text-align:center;
        padding:15px 0;
        font-size:13px;
        font-size:.8125rem;
        font-weight:500
    }
    #cookie-bar p{
        margin:0;
        padding:0
    }
    #cookie-bar a{
        color:#fff;
        display:inline-block;
        -webkit-border-radius:3px;
        -moz-border-radius:3px;
        -ms-border-radius:3px;
        border-radius:3px;
        text-decoration:none;
        padding:2px 10px;
        margin-left:8px
    }
    #cookie-bar .cb-enable{
        background:#92278f
    }
    #cookie-bar .cb-enable:hover{
        background:#fff;
        color:#92278f
    }
    #cookie-bar .cb-disable{
        background:#900
    }
    #cookie-bar .cb-disable:hover{
        background:#b00
    }
    #cookie-bar .cb-policy{
        background:#392779
    }
    #cookie-bar .cb-policy:hover{
        background:#fff;
        color:#392779
    }
    #cookie-bar.fixed{
        position:fixed;
        top:0;
        left:0;
        width:100%;
        z-index:999999
    }
    #cookie-bar.fixed.bottom{
        bottom:0;
        top:auto
    }

```



```

    }
    @media (max-width:767px){
        #cookie-bar a{
            display:block;
            max-width:150px;
            margin:auto auto 5px
        }
        #cookie-bar{
            padding:5px 0
        }
    }
}

```

## Edit Course

```

@model WebApp.Models.Course

@{
    ViewBag.Title = "Edit";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>Edit</h2>

@using (Html.BeginForm())
{
    @Html.AntiForgeryToken()

    <div class="form-horizontal">
        <h4>Course</h4>
        <hr />
        @Html.ValidationSummary(true, "", new { @class = "text-danger" })
        @Html.HiddenFor(model => model.Course_Id)

        <div class="form-group">
            @Html.LabelFor(model => model.Course_Type, htmlAttributes: new { @class =
"control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Course_Type, new { htmlAttributes = new {
@class = "form-control" } })
                @Html.ValidationMessageFor(model => model.Course_Type, "", new { @class =
"text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.SortDescription, htmlAttributes: new { @class =
"control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.SortDescription, new { htmlAttributes = new
{ @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.SortDescription, "", new {
@class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.Title, htmlAttributes: new { @class = "control-
label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.Title, new { htmlAttributes = new { @class
= "form-control" } })
                @Html.ValidationMessageFor(model => model.Title, "", new { @class = "text-
danger" })
            </div>
        </div>
    </div>
}

```



```

        </div>
    </div>

    <div class="form-group">
        @Html.LabelFor(model => model.Description, htmlAttributes: new { @class =
"control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Description, new { htmlAttributes = new {
@class = "form-control" } })
            @Html.ValidationMessageFor(model => model.Description, "", new { @class =
"text-danger" })
        </div>
    </div>

    <div class="form-group">
        @Html.LabelFor(model => model.Price, htmlAttributes: new { @class = "control-
label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Price, new { htmlAttributes = new { @class
= "form-control" } })
            @Html.ValidationMessageFor(model => model.Price, "", new { @class = "text-
danger" })
        </div>
    </div>

    <div class="form-group">
        @Html.LabelFor(model => model.Rating, htmlAttributes: new { @class = "control-
label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Rating, new { htmlAttributes = new { @class
= "form-control" } })
            @Html.ValidationMessageFor(model => model.Rating, "", new { @class =
"text-danger" })
        </div>
    </div>

    <div class="form-group">
        @Html.LabelFor(model => model.Schedule, htmlAttributes: new { @class =
"control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Schedule, new { htmlAttributes = new {
@class = "form-control" } })
            @Html.ValidationMessageFor(model => model.Schedule, "", new { @class =
"text-danger" })
        </div>
    </div>

    <div class="form-group">
        @Html.LabelFor(model => model.Category, htmlAttributes: new { @class =
"control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.Category, new { htmlAttributes = new {
@class = "form-control" } })
            @Html.ValidationMessageFor(model => model.Category, "", new { @class =
"text-danger" })
        </div>
    </div>

    <div class="form-group">
        @Html.LabelFor(model => model.ImagePath, htmlAttributes: new { @class =
"control-label col-md-2" })
        <div class="col-md-10">
            @Html.EditorFor(model => model.ImagePath, new { htmlAttributes = new {
@class = "form-control" } })

```



```

        @Html.ValidationMessageFor(model => model.ImagePath, "", new { @class =
"text-danger" })
    </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>

```

## Course Detail

@model WebApp.Models.Course

```

@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
<style>
    section#description li {
        list-style-type: square;
        line-height: 30px;
    }
</style>

<section id="hero_in" class="courses start_bg_zoom">
    <div class="wrapper">
        <div class="container">
            <h1 class="fadeInUp animated"><span></span>@Html.DisplayFor(model =>
model.Title)</h1>
        </div>
    </div>
</section>
<!--/hero_in-->
<div class="bg_color_1" style="transform: none;">
    <nav class="secondary_nav_sticky_horizontal" style="">
        <div class="container">
            <ul class="clearfix">
                <li><a href="#description" class="active">Description</a></li>
                <li><a href="#lessons">Lessons</a></li>
                <li><a href="#reviews">Reviews</a></li>
            </ul>
        </div>
    </nav>
    <div class="container margin_60_35" style="transform: none;">
        <div class="row" style="transform: none;">
            <div class="col-lg-8">

```



```

<section id="description">
    <h2>What will you learn</h2>
    <p>@Html.Raw(Model.Description)</p>

</section>
<!-- /section -->

<section id="lessons">
    <div class="intro_title">
        <h2>Lessons</h2>
        <ul>
            <li>18 lessons</li>
            <li>01:02:10</li>
        </ul>
    </div>
    <div id="accordion_lessons" role="tablist" class="add_bottom_45">
        <div class="card">
            <div class="card-header" role="tab" id="headingOne">
                <h5 class="mb-0">
                    <a data-toggle="collapse" href="#collapseOne" aria-expanded="false" aria-controls="collapseOne" class="collapsed"><i class="indicator ti-plus"></i> Introduction</a>
                </h5>
            </div>
            <div id="collapseOne" class="collapse" role="tabpanel" aria-labelledby="headingOne" data-parent="#accordion_lessons" style="">
                <div class="card-body">
                    <div class="list_lessons">
                        <ul>
                            <li><a href="https://www.youtube.com/watch?v=Ldgd_gUcqCw" class="video">Health Science</a><span>00:59</span></li>
                            <li><a href="https://www.youtube.com/watch?v=Ldgd_gUcqCw" class="video">Health and Social Care</a><span>00:59</span></li>
                            <li><a href="#" class="txt_doc">Audiology</a><span>00:59</span></li>
                        </ul>
                    </div>
                </div>
            </div>
            <!-- /card -->
            <div class="card">
                <div class="card-header" role="tab" id="headingTwo">
                    <h5 class="mb-0">
                        <a class="collapsed" data-toggle="collapse" href="#collapseTwo" aria-expanded="false" aria-controls="collapseTwo">
                            <i class="indicator ti-plus"></i> Generative Modeling Review
                        </a>
                    </h5>
                </div>
                <div id="collapseTwo" class="collapse" role="tabpanel" aria-labelledby="headingTwo" data-parent="#accordion_lessons" style="">
                    <div class="card-body">
                        <div class="list_lessons">
                            <ul>
                                <li><a href="https://www.youtube.com/watch?v=Ldgd_gUcqCw" class="video">Health Science</a><span>00:59</span></li>
                            </ul>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</section>

```



```

                <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Health and Social Care</a><span>00:59</span></li>
                    <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">History</a><span>00:59</span></li>
                        <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Healthcare Science</a><span>00:59</span></li>
                            <li><a href="#" class="txt_doc">Audiology</a><span>00:59</span></li>
                                </ul>
                            </div>
                        </div>
                    </div>
                <!-- /card -->
                <div class="card">
                    <div class="card-header" role="tab" id="headingThree">
                        <h5 class="mb-0">
                            <a class="collapsed" data-toggle="collapse" href="#collapseThree" aria-expanded="false" aria-controls="collapseThree">
                                <i class="indicator ti-plus"></i>Variational Autoencoders
                            </a>
                        </h5>
                    </div>
                    <div id="collapseThree" class="collapse" role="tabpanel" aria-labelledby="headingThree" data-parent="#accordion_lessons" style="">
                        <div class="card-body">
                            <div class="list_lessons">
                                <ul>
                                    <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Health Science</a><span>00:59</span></li>
                                    <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Health and Social Care</a><span>00:59</span></li>
                                        <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">History</a><span>00:59</span></li>
                                            <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Healthcare Science</a><span>00:59</span></li>
                                                <li><a href="#" class="txt_doc">Audiology</a><span>00:59</span></li>
                                                    </ul>
                                                </div>
                                            </div>
                                        </div>
                                    </div>
                                <!-- /card -->
                                <div class="card">
                                    <div class="card-header" role="tab" id="headingFourth">
                                        <h5 class="mb-0">
                                            <a class="" data-toggle="collapse" href="#collapseFourth" aria-expanded="true" aria-controls="collapseFourth">
                                                <i class="indicator ti-minus"></i>Gaussian Mixture Model Review
                                            </a>
                                        </h5>
                                    </div>

```



```


- <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Health Science</a><span>00:59</span></li>
- <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Health and Social Care</a><span>00:59</span></li>
- <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">History</a><span>00:59</span></li>
- <li><a href="https://www.youtube.com/watch?v=LDgd_gUcqCw" class="video">Healthcare Science</a><span>00:59</span></li>
- <li><a href="#" class="txt_doc">Audiology</a><span>00:59</span></li>


```

```

</div>
<!-- /col -->

<aside class="col-lg-4" id="sidebar" style="position: relative; overflow: visible; box-sizing: border-box; min-height: 1px;">
```

```

    <div class="theiaStickySidebar" style="padding-top: 0px; padding-bottom: 1px; position: static; transform: none; top: 0px; left: 879.5px;">
        <div class="box_detail">
            <figure>
                <a href="#" class=""><i class="arrow_triangle-right"></i><span></span></a>
            </figure>
            <div class="price">
                $@Html.DisplayFor(model => model.Price) <span class="original_price">60% discount</span>
            </div>
            <a href="cart-1.html" class="btn_1 full-width">Purchase</a>
        <div id="list_feat">
            <h3>What's includes</h3>
            <ul>
                <li><i class="icon_mobile"></i>Mobile support</li>
                <li><i class="icon_archive_alt"></i>Lesson archive</li>
            </ul>
        </div>
    </div><div class="resize-sensor" style="position: absolute; left: 0px; top: 0px; right: 0px; bottom: 0px; overflow: hidden; z-index: -1; visibility: hidden;"><div class="resize-sensor-expand" style="position: absolute; left: 0; top: 0; right: 0; bottom: 0; overflow: hidden; z-index: -1; visibility: hidden;"><div style="position: absolute; left: 0px; top: 0px; transition: 0s; width: 390px; height: 2134px;"></div></div><div class="resize-sensor-shrink" style="position: absolute; left: 0; top: 0; right: 0; bottom: 0; overflow: hidden; z-index: -1; visibility: hidden;"><div
```



```
style="position: absolute; left: 0; top: 0; transition: 0s; width: 200%; height: 200%"></div></div></div>
    </div>
</aside>
</div>
<!-- /row -->
</div>
<!-- /container -->
</div>
```



## Blog > Index.cshtml

```
@model IEnumerable<WebApp.Models.Article>

@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<section id="hero_in" class="courses">
    <div class="wrapper">
        <div class="container">
            <h1 class="fadeInUp"><span></span>Blog</h1>
        </div>
    </div>
</section>

<div class="container margin_60_35">

    @foreach (var item in Model)
    {

        <div class="box_list wow">
            <div class="row no-gutters">
                <div class="col-lg-5">
                    <figure class="block-reveal">
                        <div class="block-horizontal"></div>
                        <a href="@Url.Action("Details", "Courses", new { id = item.Article_Id })"></a>
                        <div class="preview"><span>Preview course</span></div>
                    </figure>
                </div>
                <div class="col-lg-7">
                    <div class="wrapper">
                        @*<a href="#" class="wish_bt"></a>*@
                        <small>@Html.DisplayFor(modelItem => item.Author) |<br/>
                        @Html.DisplayFor(modelItem => item.Published_On) </small>
                        <h3> @Html.DisplayFor(modelItem => item.Title) </h3>
                        <p> @Html.Raw(item.Body.Substring(0, Math.Min(item.Body.Length, 100))) </p>
                        @*<div class="rating"><i class="icon_star voted"></i><i class="icon_star voted"></i><i class="icon_star voted"></i><i class="icon_star voted"></i><small>(145)</small></div>*@
                        <div>
                            <ul>
                                <li><i class="icon_like"></i> @Html.DisplayFor(modelItem => item.Likes) </li>
                                <li>@Html.ActionLink("Read More", "Details", new { id = item.Article_Id }) </li>
                            </ul>
                        </div>
                    </div>
                </div>
            <!-- /box_list -->
        }
    <!-- /container -->
</div>
```



## Blog > Details.cshtml

```
@model WebApp.Models.Article

@{
    ViewBag.Title = "Details";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>Details</h2>

<div>
    <h4>Article</h4>
    <hr />
    <dl class="dl-horizontal">
        <dt>
            @Html.DisplayNameFor(model => model.Associated_To_Chapter)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Associated_To_Chapter)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.Title)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Title)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.Status)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Status)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.Body)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Body)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.Type)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Type)
        </dd>

        <dt>
            @Html.DisplayNameFor(model => model.Category)
        </dt>

        <dd>
            @Html.DisplayFor(model => model.Category)
        </dd>
    </dl>
</div>
```



```

</dd>

<dt>
    @Html.DisplayNameFor(model => model.Likes)
</dt>

<dd>
    @Html.DisplayFor(model => model.Likes)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Dislikes)
</dt>

<dd>
    @Html.DisplayFor(model => model.Dislikes)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Slug)
</dt>

<dd>
    @Html.DisplayFor(model => model.Slug)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.FavoriteCount)
</dt>

<dd>
    @Html.DisplayFor(model => model.FavoriteCount)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.ViewCount)
</dt>

<dd>
    @Html.DisplayFor(model => model.ViewCount)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Parent)
</dt>

<dd>
    @Html.DisplayFor(model => model.Parent)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Author)
</dt>

<dd>
    @Html.DisplayFor(model => model.Author)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.LastEditor)
</dt>

<dd>
    @Html.DisplayFor(model => model.LastEditor)

```



```
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Tags)
</dt>

<dd>
    @Html.DisplayFor(model => model.Tags)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Is_Published)
</dt>

<dd>
    @Html.DisplayFor(model => model.Is_Published)
</dd>

<dt>
    @Html.DisplayNameFor(model => model.Published_On)
</dt>

<dd>
    @Html.DisplayFor(model => model.Published_On)
</dd>

</dl>
</div>
<p>
    @Html.ActionLink("Edit", "Edit", new { id = Model.Article_Id }) |
    @Html.ActionLink("Back to List", "Index")
</p>
```



## Questions > index.cshtml

```
@model IEnumerable<WebApp.Models.Question>

@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<style type="text/css">
    .question-summary {
        padding: 12px 0 10px 0;
        border-bottom: 1px solid #e4e6e8;
    }

    .question-summary {
        overflow: hidden;
        padding: 15px 0;
        border-bottom: 1px solid #eff0f1;
    }

    #questions {
        margin-bottom: 20px;
    }

    #questions, #answers {
        clear: both;
        width: 728px;
    }

    .statscontainer {
        width: 78px;
        margin-right: 8px;
        margin-left: 8px;
    }

    .summary {
        float: left;
        width: 630px;
    }

    .summary h3 {
        font-size: 15px;
        line-height: 1.4;
        margin-bottom: .5em;
    }

    .started {
        width: 200px;
        float: right;
        line-height: 18px;
    }

    .float-right, .fr {
        float: right !important;
    }
</style>
<section class="header-section">
    <div class="wrapper">

        </div>
    </section>
    <br />
    <br />
```



```

<div class="container">
    <p>
        @Html.ActionLink("Ask Question", "Ask", null, new { @class = "btn btn-success" })
    </p>
    <div id="questions">

        @foreach (var item in Model)
        {
            <div class="question-summary">

                <div class="summary">
                    <h3><a href="#" class="question-hyperlink">@Html.DisplayFor(modelItem => item.Title)</a></h3>
                    <div class="excerpt">
                        @Html.DisplayFor(modelItem => item.Description)
                    </div>

                    <div class="started fr">
                        <div class="user-info user-hover">
                            <div class="user-action-time">
                                asked <span title="@Html.DisplayFor(modelItem => item.DTS)" class="relativetime">@Html.DisplayFor(modelItem => item.DTS)</span>
                            </div>
                            <div class="user-details">
                                <a href="#">@Html.DisplayFor(modelItem => item.Asked_By)</a>
                            </div>
                        </div>
                    </div>
                </div>
            </div>
        }
    </div>
</div>

```

## Ask.cshtml

```

@model WebApp.Models.Question

@if
    ViewBag.Title = "Create";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
<style type="text/css">
    #custom-search-input textarea {
        width: 100%;
        height: 150px;
    }
</style>
<section class="hero_single version_2">
    <div class="wrapper">
        <div class="container">
            <h3>Your question?</h3>

            @using (Html.BeginForm())
            {

```



```
    @Html.AntiForgeryToken()

    <div id="custom-search-input">
        <div class="input-group">

            @Html.ValidationSummary(true, "", new { @class = "text-danger" })

            @Html.TextAreaFor(model => model.Title, new { htmlAttributes =
new { @class = "search-query", @style = "width: 100%;" } })
            @Html.ValidationMessageFor(model => model.Title, "", new {
@class = "text-danger" })

        </div>
    </div>
</div>
</div>
</section>
```

## Reset Password (ResetPassword.cshtml)

```
@model WebApp.Models.ResetPasswordViewModel
@{
    ViewBag.Title = "Reset password";
}

<h2>@ViewBag.Title.</h2>

@using (Html.BeginForm("ResetPassword", "Account", FormMethod.Post, new { @class = "form-horizontal", role = "form" }))
{
    @Html.AntiForgeryToken()
    <h4>Reset your password.</h4>
    <hr />
    @Html.ValidationSummary("", new { @class = "text-danger" })
    @Html.HiddenFor(model => model.Code)
    <div class="form-group">
        @Html.LabelFor(m => m.Email, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.TextBoxFor(m => m.Email, new { @class = "form-control" })
        </div>
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.Password, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.PasswordFor(m => m.Password, new { @class = "form-control" })
        </div>
    </div>
}
```



```
        </div>
    </div>
    <div class="form-group">
        @Html.LabelFor(m => m.ConfirmPassword, new { @class = "col-md-2 control-label" })
        <div class="col-md-10">
            @Html.PasswordFor(m => m.ConfirmPassword, new { @class = "form-control" })
        </div>
    </div>
    <div class="form-group">
        <div class="col-md-offset-2 col-md-10">
            <input type="submit" class="btn btn-default" value="Reset" />
        </div>
    </div>
}
```



## Admin Master (\_LayoutAdmin.cshtml)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">
    <title>@ViewBag.Title - eLearning</title>
    <!-- Bootstrap Core CSS -->
    <link href="~/Content/admin/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet"
/>
    <link href="~/Content/admin/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
    <!-- MetisMenu CSS -->
    <link href="~/Content/admin/vendor/metisMenu/metisMenu.min.css" rel="stylesheet">
    <!-- Custom CSS -->
    <link href="~/Content/admin/dist/css/sb-admin-2.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="~/Content/admin/vendor/font-awesome/css/font-awesome.min.css"
rel="stylesheet" type="text/css">

</head>
<body>
    <div id="wrapper">
        <!-- Navigation -->
        <nav class="navbar navbar-default navbar-static-top" role="navigation"
style="margin-bottom: 0">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".navbar-collapse">
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a class="navbar-brand" href="index.html">eLearning Admin</a>
            </div>
            <div class="navbar-default sidebar" role="navigation">
                <div class="sidebar-nav navbar-collapse">
                    <ul class="nav" id="side-menu">
                        <li>
                            <a href="/admin/dashboard"><i class="fa fa-delicious"></i>
Dashboard</a>
                        </li>
                        <li>
                            <a href="/admin/courses"><i class="fa fa-th-list"></i> Courses
                        </a>
                    </ul>
                </div>
            </div>
        </nav>
    </div>

```



```

        </li>
        <li>
            <a href="/admin/chapters"><i class="fa fa-th-list"></i>
Chapters <a>

        </li>
        <li>
            <a href="/admin/quizes"><i class="fa fa-th-list"></i> Quizes
</a>

        </li>
        <li>
            <a href="/admin/questions"><i class="fa fa-th-list"></i>
Questions </a>

        </li>
        <li>
            <a href="/admin/Reports"><i class="fa fa-th-list"></i> Reports
</a>

        </li>
        <li>
            <a href="#"><i class="fa fa-sign-out"></i> Logout</a>

            <!-- /.nav-second-level -->
        </li>
    </ul>
</div>
<!-- /.sidebar-collapse -->
</div>
<!-- /.navbar-static-side -->
</div>
<!-- #page-wrapper -->
</div>

<script src "~/Content/admin/vendor/bootstrap/js/bootstrap.min.js"></script>
<!-- Metis Menu Plugin JavaScript -->
<script src "~/Content/admin/vendor/metisMenu/metisMenu.min.js"></script>

<!-- Custom Theme JavaScript -->
<script src "~/Content/admin/dist/js/sb-admin-2.js"></script>

<style>
    ul#side-menu .fa {
        width: 20px;
        height: 20px;
    }
</style>
</body>
</html>

```



## Stylesheet for Admin Master

```
/*
body {
    background-color: #f8f8f8;
}
#wrapper {
    width: 100%;
}
#page-wrapper {
    padding: 0 15px;
    min-height: 568px;
    background-color: white;
}
@media (min-width: 768px) {
    #page-wrapper {
        position: inherit;
        margin: 0 0 0 250px;
        padding: 0 30px;
        border-left: 1px solid #e7e7e7;
    }
}
.navbar-top-links {
    margin-right: 0;
}
.navbar-top-links li {
    display: inline-block;
}
.navbar-top-links li:last-child {
    margin-right: 15px;
}
.navbar-top-links li a {
    padding: 15px;
    min-height: 50px;
}
.navbar-top-links .dropdown-menu li {
    display: block;
}
.navbar-top-links .dropdown-menu li:last-child {
    margin-right: 0;
}
.navbar-top-links .dropdown-menu li a {
    padding: 3px 20px;
    min-height: 0;
}
.navbar-top-links .dropdown-menu li a div {
    white-space: normal;
}
.navbar-top-links .dropdown-messages,
.navbar-top-links .dropdown-tasks,
.navbar-top-links .dropdown-alerts {
    width: 310px;
    min-width: 0;
}
.navbar-top-links .dropdown-messages {
    margin-left: 5px;
}
.navbar-top-links .dropdown-tasks {
    margin-left: -59px;
}
.navbar-top-links .dropdown-alerts {
    margin-left: -123px;
}
```



```

.navbar-top-links .dropdown-user {
  right: 0;
  left: auto;
}
.sidebar .sidebar-nav.navbar-collapse {
  padding-left: 0;
  padding-right: 0;
}
.sidebar .sidebar-search {
  padding: 15px;
}
.sidebar ul li {
  border-bottom: 1px solid #e7e7e7;
}
.sidebar ul li a.active {
  background-color: #eeeeee;
}
.sidebar .arrow {
  float: right;
}
.sidebar .fa.arrow:before {
  content: "\f104";
}
.sidebar .active > a > .fa.arrow:before {
  content: "\f107";
}
.sidebar .nav-second-level li,
.sidebar .nav-third-level li {
  border-bottom: none !important;
}
.sidebar .nav-second-level li a {
  padding-left: 37px;
}
.sidebar .nav-third-level li a {
  padding-left: 52px;
}
@media (min-width: 768px) {
  .sidebar {
    z-index: 1;
    position: absolute;
    width: 250px;
    margin-top: 51px;
  }
  .navbar-top-links .dropdown-messages,
  .navbar-top-links .dropdown-tasks,
  .navbar-top-links .dropdown-alerts {
    margin-left: auto;
  }
}
.btn-outline {
  color: inherit;
  background-color: transparent;
  transition: all .5s;
}
.btn-primary.btn-outline {
  color: #428bca;
}
.btn-success.btn-outline {
  color: #5cb85c;
}
.btn-info.btn-outline {
  color: #5bc0de;
}
.btn-warning.btn-outline {

```



```

    color: #f0ad4e;
}
.btn-danger.btn-outline {
    color: #d9534f;
}
.btn-primary.btn-outline:hover,
.btn-success.btn-outline:hover,
.btn-info.btn-outline:hover,
.btn-warning.btn-outline:hover,
.btn-danger.btn-outline:hover {
    color: white;
}
.chat {
    margin: 0;
    padding: 0;
    list-style: none;
}
.chat li {
    margin-bottom: 10px;
    padding-bottom: 5px;
    border-bottom: 1px dotted #999999;
}
.chat li.left .chat-body {
    margin-left: 60px;
}
.chat li.right .chat-body {
    margin-right: 60px;
}
.chat li .chat-body p {
    margin: 0;
}
.panel .slidedown .glyphicon,
.chat .glyphicon {
    margin-right: 5px;
}
.chat-panel .panel-body {
    height: 350px;
    overflow-y: scroll;
}
.login-panel {
    margin-top: 25%;
}
.flot-chart {
    display: block;
    height: 400px;
}
.flot-chart-content {
    width: 100%;
    height: 100%;
}
table.dataTable thead .sorting,
table.dataTable thead .sorting_asc,
table.dataTable thead .sorting_desc,
table.dataTable thead .sorting_asc_disabled,
table.dataTable thead .sorting_desc_disabled {
    background: transparent;
}
table.dataTable thead .sorting_asc:after {
    content: "\f0de";
    float: right;
    font-family: fontawesome;
}
table.dataTable thead .sorting_desc:after {
    content: "\f0dd";
}

```



```

    float: right;
    font-family: fontawesome;
}
table.dataTables thead .sorting:after {
    content: "\f0dc";
    float: right;
    font-family: fontawesome;
    color: rgba(50, 50, 50, 0.5);
}
.btn-circle {
    width: 30px;
    height: 30px;
    padding: 6px 0;
    border-radius: 15px;
    text-align: center;
    font-size: 12px;
    line-height: 1.428571429;
}
.btn-circle.btn-lg {
    width: 50px;
    height: 50px;
    padding: 10px 16px;
    border-radius: 25px;
    font-size: 18px;
    line-height: 1.33;
}
.btn-circle.btn-xl {
    width: 70px;
    height: 70px;
    padding: 10px 16px;
    border-radius: 35px;
    font-size: 24px;
    line-height: 1.33;
}
.show-grid [class^="col-"] {
    padding-top: 10px;
    padding-bottom: 10px;
    border: 1px solid #ddd;
    background-color: #eee !important;
}
.show-grid {
    margin: 15px 0;
}
.huge {
    font-size: 40px;
}
.panel-green {
    border-color: #5cb85c;
}
.panel-green > .panel-heading {
    border-color: #5cb85c;
    color: white;
    background-color: #5cb85c;
}
.panel-green > a {
    color: #5cb85c;
}
.panel-green > a:hover {
    color: #3d8b3d;
}
.panel-red {
    border-color: #d9534f;
}
.panel-red > .panel-heading {

```



```

border-color: #d9534f;
color: white;
background-color: #d9534f;
}
.panel-red > a {
  color: #d9534f;
}
.panel-red > a:hover {
  color: #b52b27;
}
.panel-yellow {
  border-color: #f0ad4e;
}
.panel-yellow > .panel-heading {
  border-color: #f0ad4e;
  color: white;
  background-color: #f0ad4e;
}
.panel-yellow > a {
  color: #f0ad4e;
}
.panel-yellow > a:hover {
  color: #df8a13;
}
.timeline {
  position: relative;
  padding: 20px 0 20px;
  list-style: none;
}
.timeline:before {
  content: " ";
  position: absolute;
  top: 0;
  bottom: 0;
  left: 50%;
  width: 3px;
  margin-left: -1.5px;
  background-color: #eeeeee;
}
.timeline > li {
  position: relative;
  margin-bottom: 20px;
}
.timeline > li:before,
.timeline > li:after {
  content: " ";
  display: table;
}
.timeline > li:after {
  clear: both;
}
.timeline > li:before,
.timeline > li:after {
  content: " ";
  display: table;
}
.timeline > li:after {
  clear: both;
}
.timeline > li > .timeline-panel {
  float: left;
  position: relative;
  width: 46%;
  padding: 20px;
}

```



```

border: 1px solid #d4d4d4;
border-radius: 2px;
-webkit-box-shadow: 0 1px 6px rgba(0, 0, 0, 0.175);
box-shadow: 0 1px 6px rgba(0, 0, 0, 0.175);
}
.timeline > li > .timeline-panel:before {
  content: " ";
  display: inline-block;
  position: absolute;
  top: 26px;
  right: -15px;
  border-top: 15px solid transparent;
  border-right: 0 solid #ccc;
  border-bottom: 15px solid transparent;
  border-left: 15px solid #ccc;
}
.timeline > li > .timeline-panel:after {
  content: " ";
  display: inline-block;
  position: absolute;
  top: 27px;
  right: -14px;
  border-top: 14px solid transparent;
  border-right: 0 solid #fff;
  border-bottom: 14px solid transparent;
  border-left: 14px solid #fff;
}
.timeline > li > .timeline-badge {
  z-index: 100;
  position: absolute;
  top: 16px;
  left: 50%;
  width: 50px;
  height: 50px;
  margin-left: -25px;
  border-radius: 50% 50% 50% 50%;
  text-align: center;
  font-size: 1.4em;
  line-height: 50px;
  color: #fff;
  background-color: #999999;
}
.timeline > li.timeline-inverted > .timeline-panel {
  float: right;
}
.timeline > li.timeline-inverted > .timeline-panel:before {
  right: auto;
  left: -15px;
  border-right-width: 15px;
  border-left-width: 0;
}
.timeline > li.timeline-inverted > .timeline-panel:after {
  right: auto;
  left: -14px;
  border-right-width: 14px;
  border-left-width: 0;
}
.timeline-badge.primary {
  background-color: #2e6da4 !important;
}
.timeline-badge.success {
  background-color: #3f903f !important;
}
.timeline-badge.warning {

```



```
background-color: #f0ad4e !important;
}
.timeline-badge.danger {
  background-color: #d9534f !important;
}
.timeline-badge.info {
  background-color: #5bc0de !important;
}
.timeline-title {
  margin-top: 0;
  color: inherit;
}
.timeline-body > p,
.timeline-body > ul {
  margin-bottom: 0;
}
.timeline-body > p + p {
  margin-top: 5px;
}
@media (max-width: 767px) {
  ul.timeline:before {
    left: 40px;
  }
  ul.timeline > li > .timeline-panel {
    width: calc(10%);
    width: -moz-calc(10%);
    width: -webkit-calc(10%);
  }
  ul.timeline > li > .timeline-badge {
    top: 16px;
    left: 15px;
    margin-left: 0;
  }
  ul.timeline > li > .timeline-panel {
    float: right;
  }
  ul.timeline > li > .timeline-panel:before {
    right: auto;
    left: -15px;
    border-right-width: 15px;
    border-left-width: 0;
  }
  ul.timeline > li > .timeline-panel:after {
    right: auto;
    left: -14px;
    border-right-width: 14px;
    border-left-width: 0;
  }
}
```



## Supporting Javascript file

```
$(function() {
    $('#side-menu').metisMenu();
});

//Loads the correct sidebar on window load,
//collapses the sidebar on window resize.
// Sets the min-height of #page-wrapper to window size
$(function() {
    $(window).bind("load resize", function() {
        var topOffset = 50;
        var width = (this.window.innerWidth > 0) ? this.window.innerWidth :
this.screen.width;
        if (width < 768) {
            $('div.navbar-collapse').addClass('collapse');
            topOffset = 100; // 2-row-menu
        } else {
            $('div.navbar-collapse').removeClass('collapse');
        }

        var height = ((this.window.innerHeight > 0) ? this.window.innerHeight :
this.screen.height) - 1;
        height = height - topOffset;
        if (height < 1) height = 1;
        if (height > topOffset) {
            $("#page-wrapper").css("min-height", (height) + "px");
        }
    });
}

var url = window.location;
// var element = $('ul.nav a').filter(function() {
//     return this.href == url;
// }).addClass('active').parent().parent().addClass('in').parent();
var element = $('ul.nav a').filter(function() {
    return this.href == url;
}).addClass('active').parent();

while (true) {
    if (element.is('li')) {
        element = element.parent().addClass('in').parent();
    } else {
        break;
    }
}
});
```



## Controllers Admin

### ChaptersController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.admin.Controllers
{
    public class ChaptersController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: admin/Chapters
        public async Task<ActionResult> Index()
        {
            return View(await db.Chapters.ToListAsync());
        }
        public void RefreshAll()
        {
            foreach (var entity in db.ChangeTracker.Entries())
            {
                entity.Reload();
            }
        }
        // GET: admin/Chapters/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Chapter chapter = await db.Chapters.FindAsync(id);
            if (chapter == null)
            {
                return HttpNotFound();
            }
            return View(chapter);
        }

        // GET: admin/Chapters/Create
        public ActionResult Create()
        {
            //RefreshAll();
            //List<Course> Courses = db.Courses.Where(x => x.Course_Id > 0).ToList();
            //List<SelectListItem> lstItems = new List<SelectListItem>();
            //foreach(Course m in Courses)
            //{
            //    lstItems.Add(new SelectListItem { Text = m.Title, Value =
m.Course_Id.ToString() });
            //}
            // ViewBag.Courses = lstItems;
            return View();
        }
    }
}
```



```

// POST: admin/Chapters/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 async Task<ActionResult> Create([Bind(Include =
"Course_Id,Chapter_Title,Description,Schedule,Parent")] Chapter chapter)
{
    if (ModelState.IsValid)
    {
        chapter.DTS = DateTime.Now;
        chapter.Chapter_Id = 0;
        db.Chapters.Add(chapter);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    return View(chapter);
}

// GET: admin/Chapters/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Chapter chapter = await db.Chapters.FindAsync(id);

    if (chapter == null)
    {
        return HttpNotFound();
    }
    //else
    //{
    //    chapter.Courses = db.Courses.Where(x=> x.Course_Id > 0).ToList();
    //}
    return View(chapter);
}

// POST: admin/Chapters/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 async Task<ActionResult> Edit([Bind(Include =
"Chapter_Id,Course_Id,Chapter_Title,Description,Schedule,Parent")] Chapter chapter)
{
    if (ModelState.IsValid)
    {
        chapter.DTS = DateTime.Now;

        db.Entry(chapter).State = EntityState.Modified;
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
    return View(chapter);
}

// GET: admin/Chapters/Delete/5
public async Task<ActionResult> Delete(int? id)
{

```



```

        if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Chapter chapter = await db.Chapters.FindAsync(id);
    if (chapter == null)
    {
        return HttpNotFound();
    }
    return View(chapter);
}

// POST: admin/Chapters/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<ActionResult> DeleteConfirmed(int id)
{
    Chapter chapter = await db.Chapters.FindAsync(id);
    db.Chapters.Remove(chapter);
    await db.SaveChangesAsync();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
}
}

```

## CoursesController.cs

```

using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.admin.Controllers
{
    public class CoursesController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: admin/Courses
        public async Task<ActionResult> Index()
        {
            return View(await db.Courses.ToListAsync());
        }

        // GET: admin/Courses/Details/5
        public async Task<ActionResult> Details(int? id)
        {

```



```

        if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Course course = await db.Courses.FindAsync(id);
    if (course == null)
    {
        return HttpNotFound();
    }
    return View(course);
}

// GET: admin/Courses/Create
public ActionResult Create()
{
    return View();
}

// POST: admin/Courses/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 async Task<ActionResult> Create([Bind(Include =
"Course_Id,Course_Type,Duration,Title,Description,Price,Rating,Schedule,Category,ImagePath
")] Course course)
{
    if (ModelState.IsValid)
    {
        db.Courses.Add(course);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    return View(course);
}

// GET: admin/Courses/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Course course = await db.Courses.FindAsync(id);
    if (course == null)
    {
        return HttpNotFound();
    }
    return View(course);
}

// POST: admin/Courses/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, ValidateInput(false)]
[ValidateAntiForgeryToken]
public async Task<ActionResult> Edit([Bind(Include =
"Course_Id,Course_Type,Duration,Title,Description,Price,Rating,Schedule,Category,ImagePath
")] Course course)
{
    if (ModelState.IsValid)

```



```

        {
            db.Entry(course).State = EntityState.Modified;
            await db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
        return View(course);
    }

    // GET: admin/Courses/Delete/5
    public async Task<ActionResult> Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Course course = await db.Courses.FindAsync(id);
        if (course == null)
        {
            return HttpNotFound();
        }
        return View(course);
    }

    // POST: admin/Courses/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> DeleteConfirmed(int id)
    {
        Course course = await db.Courses.FindAsync(id);
        db.Courses.Remove(course);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
}
}

```

## DashboardController.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace WebApp.Areas.admin.Controllers
{
    public class DashboardController : Controller
    {
        // GET: admin/Dashboard
        public ActionResult Index()
        {
            return View();
        }
    }
}

```



```
}
```

## QuestionsController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.admin.Controllers
{
    public class QuestionsController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: admin/Questions
        public async Task<ActionResult> Index()
        {
            return View(await db.Questions.ToListAsync());
        }

        // GET: admin/Questions/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Question question = await db.Questions.FindAsync(id);
            if (question == null)
            {
                return HttpNotFound();
            }
            return View(question);
        }

        // GET: admin/Questions/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: admin/Questions/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 async Task<ActionResult> Create([Bind(Include =
"Question_Id,Asked_By,Title,Description,Selected_Answer")] Question question)
        {
            if (ModelState.IsValid)
            {
                question.DTS = DateTime.Now;
```



```

        db.Questions.Add(question);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    return View(question);
}

// GET: admin/Questions/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Question question = await db.Questions.FindAsync(id);
    if (question == null)
    {
        return HttpNotFound();
    }
    return View(question);
}

// POST: admin/Questions/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 async Task<ActionResult> Edit([Bind(Include =
"Question_Id,Asked_By,Title,Description,Selected_Answer,DTS")] Question question)
{
    if (ModelState.IsValid)
    {
        db.Entry(question).State = EntityState.Modified;
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
    return View(question);
}

// GET: admin/Questions/Delete/5
public async Task<ActionResult> Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Question question = await db.Questions.FindAsync(id);
    if (question == null)
    {
        return HttpNotFound();
    }
    return View(question);
}

// POST: admin/Questions/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<ActionResult> DeleteConfirmed(int id)
{
    Question question = await db.Questions.FindAsync(id);
    db.Questions.Remove(question);
    await db.SaveChangesAsync();
}

```



```
        return RedirectToAction("Index");
    }

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}

}
```

## QuizesController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.admin.Controllers
{
    public class QuizesController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: admin/Quizes
        public async Task<ActionResult> Index()
        {
            return View(await db.Quizes.ToListAsync());
        }

        // GET: admin/Quizes/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Quizes quizes = await db.Quizes.FindAsync(id);
            if (quizes == null)
            {
                return HttpNotFound();
            }
            return View(quizes);
        }

        // GET: admin/Quizes/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: admin/Quizes/Create
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Create([Bind(Include = "Id,Title,Content,Answer,Score")] Quizes quizes)
        {
            if (ModelState.IsValid)
            {
                db.Quizes.Add(quizes);
                db.SaveChanges();
                return RedirectToAction("Index");
            }
            return View(quizes);
        }
    }
}
```



```

// 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 async Task<ActionResult> Create([Bind(Include =
"Quiz_Group_Id,Quiz_Name,Description")] Quizes quizzes)
{
    if (ModelState.IsValid)
    {
        db.Quizes.Add(quizzes);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    return View(quizzes);
}

// GET: admin/Quizes/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Quizes quizzes = await db.Quizes.FindAsync(id);
    if (quizzes == null)
    {
        return HttpNotFound();
    }
    return View(quizzes);
}

// POST: admin/Quizes/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 async Task<ActionResult> Edit([Bind(Include =
"Quiz_Group_Id,Quiz_Name,Description")] Quizes quizzes)
{
    if (ModelState.IsValid)
    {
        db.Entry(quizzes).State = EntityState.Modified;
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
    return View(quizzes);
}

// GET: admin/Quizes/Delete/5
public async Task<ActionResult> Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Quizes quizzes = await db.Quizes.FindAsync(id);
    if (quizzes == null)
    {
        return HttpNotFound();
    }
    return View(quizzes);
}

```



```

    }

    // POST: admin/Quizes/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> DeleteConfirmed(int id)
    {
        Quizes quizzes = await db.Quizes.FindAsync(id);
        db.Quizes.Remove(quizzes);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
}
}

```

## ReportsController.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace WebApp.Areas.admin.Controllers
{
    public class ReportsController : Controller
    {
        // GET: admin/Reports
        public ActionResult Index()
        {
            return View();
        }
    }
}

```



## Controllers Public

### CourseController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.en.Controllers
{
    public class CoursesController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: Courses
        public async Task<ActionResult> Index()
        {
            return View(await db.Courses.ToListAsync());
        }

        // GET: Courses/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Course course = await db.Courses.FindAsync(id);
            if (course == null)
            {
                return HttpNotFound();
            }
            return View(course);
        }

        // GET: Courses/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: Courses/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 async Task<ActionResult> Create([Bind(Include =
"Course_Id,Course_Type,SortDescription,Title,Description,Price,Rating,Schedule,Category")]
        Course course)
        {
            if (ModelState.IsValid)
            {
                db.Courses.Add(course);
                await db.SaveChangesAsync();
                return RedirectToAction("Index");
            }
        }
    }
}
```



```

        }

        return View(course);
    }

    // GET: Courses/Edit/5
    public async Task<ActionResult> Edit(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Course course = await db.Courses.FindAsync(id);
        if (course == null)
        {
            return HttpNotFound();
        }
        return View(course);
    }

    // POST: Courses/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 async Task<ActionResult> Edit([Bind(Include =
    "Course_Id,Course_Type,SortDescription,Title,Description,Price,Rating,Schedule,Category")]
    Course course)
    {
        if (ModelState.IsValid)
        {
            db.Entry(course).State = EntityState.Modified;
            await db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
        return View(course);
    }

    // GET: Courses/Delete/5
    public async Task<ActionResult> Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Course course = await db.Courses.FindAsync(id);
        if (course == null)
        {
            return HttpNotFound();
        }
        return View(course);
    }

    // POST: Courses/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> DeleteConfirmed(int id)
    {
        Course course = await db.Courses.FindAsync(id);
        db.Courses.Remove(course);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
}

```



```
protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```



## QuestionsController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.en.Controllers
{
    public class QuestionsController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: en/Questions
        public async Task<ActionResult> Index()
        {
            return View(await db.Questions.ToListAsync());
        }

        // GET: en/Questions/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Question question = await db.Questions.FindAsync(id);
            if (question == null)
            {
                return HttpNotFound();
            }
            return View(question);
        }

        // GET: en/Questions/Create
        public ActionResult Create()
        {
            return View();
        }

        // GET: Questions/Create
        public ActionResult Ask()
        {
            return View();
        }

        // POST: en/Questions/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 async Task<ActionResult> Create([Bind(Include =
        "Question_Id,Asked_By,Title,Description,Selected_Answer,DTS")] Question question)
        {
            if (ModelState.IsValid)
            {
                db.Questions.Add(question);
                await db.SaveChangesAsync();
                return RedirectToAction("Index");
```



```

        }

        return View(question);
    }

    // GET: en/Questions/Edit/5
    public async Task<ActionResult> Edit(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Question question = await db.Questions.FindAsync(id);
        if (question == null)
        {
            return HttpNotFound();
        }
        return View(question);
    }

    // POST: en/Questions/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 async Task<ActionResult> Edit([Bind(Include =
    "Question_Id,Asked_By,Title,Description,Selected_Answer,DTS")] Question question)
    {
        if (ModelState.IsValid)
        {
            db.Entry(question).State = EntityState.Modified;
            await db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
        return View(question);
    }

    // GET: en/Questions/Delete/5
    public async Task<ActionResult> Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Question question = await db.Questions.FindAsync(id);
        if (question == null)
        {
            return HttpNotFound();
        }
        return View(question);
    }

    // POST: en/Questions/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> DeleteConfirmed(int id)
    {
        Question question = await db.Questions.FindAsync(id);
        db.Questions.Remove(question);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
}

```



```
protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```



## QuizesController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.en.Controllers
{
    public class QuizesController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: Quizes
        public async Task<ActionResult> Index()
        {
            return View(await db.Quizes.ToListAsync());
        }

        // GET: Quizes/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Quizes quiz = await db.Quizes.FindAsync(id);
            if (quiz == null)
            {
                return HttpNotFound();
            }
            return View(quiz);
        }

        // GET: Quizes/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: Quizes/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 async Task<ActionResult> Create([Bind(Include = "Quiz_Name,Description")]
Quizes quiz)
        {
            if (ModelState.IsValid)
            {
                db.Quizes.Add(quiz);
                await db.SaveChangesAsync();
                return RedirectToAction("Index");
            }
            return View(quiz);
        }
    }
}
```



```

// GET: Quizes/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Quizes quiz = await db.Quizes.FindAsync(id);
    if (quiz == null)
    {
        return HttpNotFound();
    }
    return View(quiz);
}

// POST: Quizes/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 async Task<ActionResult> Edit([Bind(Include =
"Quiz_Group_Id,Quiz_Name,Description")] Quizes quiz)
{
    if (ModelState.IsValid)
    {
        db.Entry(quiz).State = EntityState.Modified;
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
    return View(quiz);
}

// GET: Quizes/Delete/5
public async Task<ActionResult> Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Quizes quiz = await db.Quizes.FindAsync(id);
    if (quiz == null)
    {
        return HttpNotFound();
    }
    return View(quiz);
}

// POST: Quizes/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<ActionResult> DeleteConfirmed(int id)
{
    Quizes quiz = await db.Quizes.FindAsync(id);
    db.Quizes.Remove(quiz);
    await db.SaveChangesAsync();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
}

```



```
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```



## BlogController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.en.Controllers
{
    public class BlogController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: en/Blog
        public async Task<ActionResult> Index()
        {
            return View(await db.Articles.ToListAsync());
        }

        // GET: en/Blog/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Article article = await db.Articles.FindAsync(id);
            if (article == null)
            {
                return HttpNotFound();
            }
            return View(article);
        }

        // GET: en/Blog/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: en/Blog/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 async Task<ActionResult> Create([Bind(Include =
        "Article_Id,Associated_To_Chapter,Title,Status,Body>Type,Category,Likes,Dislikes,Slug,Favo
        riteCount,ViewCount,Parent,Author,LastEditor,Tags,Is_Published,Published_On")] Article
        article)
        {
            if (ModelState.IsValid)
            {
                db.Articles.Add(article);
                await db.SaveChangesAsync();
                return RedirectToAction("Index");
            }
        }
    }
}
```



```

        return View(article);
    }

    // GET: en/Blog/Edit/5
    public async Task<ActionResult> Edit(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Article article = await db.Articles.FindAsync(id);
        if (article == null)
        {
            return HttpNotFound();
        }
        return View(article);
    }

    // POST: en/Blog/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 async Task<ActionResult> Edit([Bind(Include =
"Article_Id,Associated_To_Chapter,Title,Status,Body>Type,Category,Likes,Dislikes,Slug,Favo
riteCount,ViewCount,Parent,Author,LastEditor,Tags,Is_Published,Published_On")] Article
article)
    {
        if (ModelState.IsValid)
        {
            db.Entry(article).State = EntityState.Modified;
            await db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
        return View(article);
    }

    // GET: en/Blog/Delete/5
    public async Task<ActionResult> Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Article article = await db.Articles.FindAsync(id);
        if (article == null)
        {
            return HttpNotFound();
        }
        return View(article);
    }

    // POST: en/Blog/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> DeleteConfirmed(int id)
    {
        Article article = await db.Articles.FindAsync(id);
        db.Articles.Remove(article);
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
}

```



```
protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```



## ChaptersController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.admin.Controllers
{
    public class ChaptersController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: admin/Chapters
        public async Task<ActionResult> Index()
        {
            return View(await db.Chapters.ToListAsync());
        }
        public void RefreshAll()
        {
            foreach (var entity in db.ChangeTracker.Entries())
            {
                entity.Reload();
            }
        }
        // GET: admin/Chapters/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Chapter chapter = await db.Chapters.FindAsync(id);
            if (chapter == null)
            {
                return HttpNotFound();
            }
            return View(chapter);
        }

        // GET: admin/Chapters/Create
        public ActionResult Create()
        {

            return View();
        }

        [HttpPost]
        [ValidateAntiForgeryToken]
        public async Task<ActionResult> Create([Bind(Include =
"Course_Id,Chapter_Title,Description,Schedule,Parent")] Chapter chapter)
        {
            if (ModelState.IsValid)
            {
                chapter.DTS = DateTime.Now;
                chapter.Chapter_Id = 0;
                db.Chapters.Add(chapter);
            }
        }
    }
}
```



```

        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }

    return View(chapter);
}

// GET: admin/Chapters/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Chapter chapter = await db.Chapters.FindAsync(id);

    if (chapter == null)
    {
        return HttpNotFound();
    }
    //else
    //{
    //    chapter.Courses = db.Courses.Where(x=> x.Course_Id > 0).ToList();
    //}
    return View(chapter);
}

[HttpPost]
[ValidateAntiForgeryToken]
public async Task<ActionResult> Edit([Bind(Include =
"Chapter_Id,Course_Id,Chapter_Title,Description,Schedule,Parent")] Chapter chapter)
{
    if (ModelState.IsValid)
    {
        chapter.DTS = DateTime.Now;

        db.Entry(chapter).State = EntityState.Modified;
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
    return View(chapter);
}

// GET: admin/Chapters/Delete/5
public async Task<ActionResult> Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Chapter chapter = await db.Chapters.FindAsync(id);
    if (chapter == null)
    {
        return HttpNotFound();
    }
    return View(chapter);
}

// POST: admin/Chapters/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<ActionResult> DeleteConfirmed(int id)
{

```



```
Chapter chapter = await db.Chapters.FindAsync(id);
db.Chapters.Remove(chapter);
await db.SaveChangesAsync();
return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```



## QuizesController.cs

```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Threading.Tasks;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApp.Models;

namespace WebApp.Areas.admin.Controllers
{
    public class QuizesController : Controller
    {
        private ApplicationDbContext db = new ApplicationDbContext();

        // GET: admin/Quizes
        public async Task<ActionResult> Index()
        {
            return View(await db.Quizes.ToListAsync());
        }

        // GET: admin/Quizes/Details/5
        public async Task<ActionResult> Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Quizes quizes = await db.Quizes.FindAsync(id);
            if (quizes == null)
            {
                return HttpNotFound();
            }
            return View(quizes);
        }

        // GET: admin/Quizes/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: admin/Quizes/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 async Task<ActionResult> Create([Bind(Include =
        "Quiz_Group_Id,Quiz_Name,Description")] Quizes quizes)
        {
            if (ModelState.IsValid)
            {
                db.Quizes.Add(quizes);
                await db.SaveChangesAsync();
                return RedirectToAction("Index");
            }
            return View(quizes);
        }
    }
}
```



```

// GET: admin/Quizes/Edit/5
public async Task<ActionResult> Edit(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Quizes quizes = await db.Quizes.FindAsync(id);
    if (quizes == null)
    {
        return HttpNotFound();
    }
    return View(quizes);
}

// POST: admin/Quizes/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 async Task<ActionResult> Edit([Bind(Include =
"Quiz_Group_Id,Quiz_Name,Description")] Quizes quizes)
{
    if (ModelState.IsValid)
    {
        db.Entry(quizes).State = EntityState.Modified;
        await db.SaveChangesAsync();
        return RedirectToAction("Index");
    }
    return View(quizes);
}

// GET: admin/Quizes/Delete/5
public async Task<ActionResult> Delete(int? id)
{
    if (id == null)
    {
        return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
    }
    Quizes quizes = await db.Quizes.FindAsync(id);
    if (quizes == null)
    {
        return HttpNotFound();
    }
    return View(quizes);
}

// POST: admin/Quizes/Delete/5
[HttpPost, ActionName("Delete")]
[ValidateAntiForgeryToken]
public async Task<ActionResult> DeleteConfirmed(int id)
{
    Quizes quizes = await db.Quizes.FindAsync(id);
    db.Quizes.Remove(quizes);
    await db.SaveChangesAsync();
    return RedirectToAction("Index");
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
}

```



```
        db.Dispose();
    }
    base.Dispose(disposing);
}
}
```

## AccountController.cs

```
using System;
using System.Globalization;
using System.Linq;
using System.Security.Claims;
using System.Threading.Tasks;
using System.Web;
using System.Web.Mvc;
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.Owin;
using Microsoft.Owin.Security;
using WebApp.Models;

namespace WebApp.Controllers
{
    [Authorize]
    public class AccountController : Controller
    {
        private ApplicationSignInManager _signInManager;
        private ApplicationUserManager _userManager;

        public AccountController()
        {
        }

        public AccountController(ApplicationUserManager userManager,
ApplicationSignInManager signInManager )
        {
            UserManager = userManager;
            SignInManager = signInManager;
        }

        public ApplicationSignInManager SignInManager
        {
            get
            {
                return _signInManager ?? 
HttpContext.GetOwinContext().Get<ApplicationSignInManager>();
            }
            private set
            {
                _signInManager = value;
            }
        }

        public ApplicationUserManager UserManager
        {
            get
            {
                return _userManager ?? 
HttpContext.GetOwinContext().GetUserManager<ApplicationUserManager>();
            }
        }
    }
}
```



```

        }
        private set
        {
            _userManager = value;
        }
    }

    //
    // GET: /Account/Login
    [AllowAnonymous]
    public ActionResult Login(string returnUrl)
    {
        ViewBag.ReturnUrl = returnUrl;
        return View();
    }

    //
    // POST: /Account/Login
    [HttpPost]
    [AllowAnonymous]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> Login(LoginViewModel model, string returnUrl)
    {
        if (!ModelState.IsValid)
        {
            return View(model);
        }

        // This doesn't count login failures towards account lockout
        // To enable password failures to trigger account lockout, change to
        shouldLockout: true
        var result = await SignInManager.PasswordSignInAsync(model.Email,
model.Password, model.RememberMe, shouldLockout: false);
        switch (result)
        {
            case SignInStatus.Success:
                return RedirectToLocal(returnUrl);
            case SignInStatus.LockedOut:
                return View("Lockout");
            case SignInStatus.RequiresVerification:
                return RedirectToAction("SendCode", new { ReturnUrl = returnUrl,
RememberMe = model.RememberMe });
            case SignInStatus.Failure:
            default:
                ModelState.AddModelError("", "Invalid login attempt.");
                return View(model);
        }
    }

    //
    // GET: /Account/VerifyCode
    [AllowAnonymous]
    public async Task<ActionResult> VerifyCode(string provider, string returnUrl, bool
rememberMe)
    {
        // Require that the user has already logged in via username/password or
        external login
        if (!await SignInManager.Has Been Verified Async())
        {
            return View("Error");
        }
        return View(new VerifyCodeViewModel { Provider = provider, ReturnUrl =
returnUrl, RememberMe = rememberMe });
    }
}

```



```

//  

// POST: /Account/VerifyCode  

[HttpPost]  

[AllowAnonymous]  

[ValidateAntiForgeryToken]  

public async Task<ActionResult> VerifyCode(VerifyCodeViewModel model)  

{  

    if (!ModelState.IsValid)  

    {  

        return View(model);  

    }  

    // The following code protects for brute force attacks against the two factor  

    codes.  

    // If a user enters incorrect codes for a specified amount of time then the  

    user account  

    // will be locked out for a specified amount of time.  

    // You can configure the account lockout settings in IdentityConfig  

    var result = await SignInManager.TwoFactorSignInAsync(model.Provider,  

model.Code, isPersistent: model.RememberMe, rememberBrowser: model.RememberBrowser);  

    switch (result)  

    {  

        case SignInStatus.Success:  

            return RedirectToLocal(model.ReturnUrl);  

        case SignInStatus.LockedOut:  

            return View("Lockout");  

        case SignInStatus.Failure:  

        default:  

            ModelState.AddModelError("", "Invalid code.");  

            return View(model);  

    }  

}  

//  

// GET: /Account/Register  

[AllowAnonymous]  

public ActionResult Register()  

{  

    return View();  

}  

//  

// POST: /Account/Register  

[HttpPost]  

[AllowAnonymous]  

[ValidateAntiForgeryToken]  

public async Task<ActionResult> Register(RegisterViewModel model)  

{  

    if (ModelState.IsValid)  

    {  

        var user = new User { User_Name = model.Email, Email = model.Email };  

    }  

    // If we got this far, something failed, redisplay form  

    return View(model);  

}  

//  

// GET: /Account/ConfirmEmail  

[AllowAnonymous]  

public async Task<ActionResult> ConfirmEmail(string userId, string code)  

{

```



```

        if (userId == null || code == null)
        {
            return View("Error");
        }
        var result = await UserManager.ConfirmEmailAsync(userId, code);
        return View(result.Succeeded ? "ConfirmEmail" : "Error");
    }

    //
    // GET: /Account/ForgotPassword
    [AllowAnonymous]
    public ActionResult ForgotPassword()
    {
        return View();
    }

    //
    // POST: /Account/ForgotPassword
    [HttpPost]
    [AllowAnonymous]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> ForgotPassword(ForgotPasswordViewModel model)
    {
        if (ModelState.IsValid)
        {
            var user = await UserManager.FindByNameAsync(model.Email);
            if (user == null || !(await UserManager.IsEmailConfirmedAsync(user.Id)))
            {
                // Don't reveal that the user does not exist or is not confirmed
                return View("ForgotPasswordConfirmation");
            }

            // For more information on how to enable account confirmation and password
reset please visit http://go.microsoft.com/fwlink/?LinkId=320771
                // Send an email with this link
                // string code = await
UserManager.GeneratePasswordResetTokenAsync(user.Id);
                // var callbackUrl = Url.Action("ResetPassword", "Account", new { userId =
user.Id, code = code }, protocol: Request.Url.Scheme);
                // await UserManager.SendEmailAsync(user.Id, "Reset Password", "Please
reset your password by clicking <a href=\"" + callbackUrl + "\">here</a>");
                // return RedirectToAction("ForgotPasswordConfirmation", "Account");
            }

            // If we got this far, something failed, redisplay form
            return View(model);
        }
    }

    //
    // GET: /Account/ForgotPasswordConfirmation
    [AllowAnonymous]
    public ActionResult ForgotPasswordConfirmation()
    {
        return View();
    }

    //
    // GET: /Account/ResetPassword
    [AllowAnonymous]
    public ActionResult ResetPassword(string code)
    {
        return code == null ? View("Error") : View();
    }

```



```

//  

// POST: /Account/ResetPassword  

[HttpPost]  

[AllowAnonymous]  

[ValidateAntiForgeryToken]  

public async Task<ActionResult> ResetPassword(ResetPasswordViewModel model)  

{  

    if (!ModelState.IsValid)  

    {  

        return View(model);  

    }  

    var user = await UserManager.FindByNameAsync(model.Email);  

    if (user == null)  

    {  

        // Don't reveal that the user does not exist  

        return RedirectToAction("ResetPasswordConfirmation", "Account");  

    }  

    var result = await UserManager.ResetPasswordAsync(user.Id, model.Code,  

model.Password);  

    if (result.Succeeded)  

    {  

        return RedirectToAction("ResetPasswordConfirmation", "Account");  

    }  

    AddErrors(result);  

    return View();  

}  

//  

// GET: /Account/ResetPasswordConfirmation  

[AllowAnonymous]  

public ActionResult ResetPasswordConfirmation()  

{  

    return View();  

}  

//  

// POST: /Account/ExternalLogin  

[HttpPost]  

[AllowAnonymous]  

[ValidateAntiForgeryToken]  

public ActionResult ExternalLogin(string provider, string returnUrl)  

{  

    // Request a redirect to the external login provider  

    return new ChallengeResult(provider, Url.Action("ExternalLoginCallback",  

"Account", new { ReturnUrl = returnUrl }));  

}  

//  

// GET: /Account/SendCode  

[AllowAnonymous]  

public async Task<ActionResult> SendCode(string returnUrl, bool rememberMe)  

{  

    var userId = await SignInManager.GetVerifiedUserIdAsync();  

    if (userId == null)  

    {  

        return View("Error");  

    }  

    var userFactors = await UserManager.GetValidTwoFactorProvidersAsync(userId);  

    var factorOptions = userFactors.Select(purpose => new SelectListItem { Text =  

purpose, Value = purpose }).ToList();  

    return View(new SendCodeViewModel { Providers = factorOptions, ReturnUrl =  

returnUrl, RememberMe = rememberMe });  

}

```



```

//  

// POST: /Account/SendCode  

[HttpPost]  

[AllowAnonymous]  

[ValidateAntiForgeryToken]  

public async Task<ActionResult> SendCode(SendCodeViewModel model)  

{  

    if (!ModelState.IsValid)  

    {  

        return View();  

    }  

    // Generate the token and send it  

    if (!await SignInManager.SendTwoFactorCodeAsync(model.SelectedProvider))  

    {  

        return View("Error");  

    }  

    return RedirectToAction("VerifyCode", new { Provider = model.SelectedProvider,  

ReturnUrl = model.ReturnUrl, RememberMe = model.RememberMe });
}  

//  

// GET: /Account/ExternalLoginCallback  

[AllowAnonymous]  

public async Task<ActionResult> ExternalLoginCallback(string returnUrl)  

{  

    var loginInfo = await AuthenticationManager.GetExternalLoginInfoAsync();  

    if (loginInfo == null)  

    {  

        return RedirectToAction("Login");
    }  

    // Sign in the user with this external login provider if the user already has  

    a login  

    var result = await SignInManager.ExternalSignInAsync(loginInfo, isPersistent:  

false);
    switch (result)
    {
        case SignInStatus.Success:
            return RedirectToLocal(returnUrl);
        case SignInStatus.LockedOut:
            return View("Lockout");
        case SignInStatus.RequiresVerification:
            return RedirectToAction("SendCode", new { ReturnUrl = returnUrl,
RememberMe = false });
        case SignInStatus.Failure:
        default:
            // If the user does not have an account, then prompt the user to  

            // create an account
            ViewBag.ReturnUrl = returnUrl;
            ViewBag.LoginProvider = loginInfo.Login.LoginProvider;
            return View("ExternalLoginConfirmation", new
ExternalLoginConfirmationViewModel { Email = loginInfo.Email });
    }
}  

//  

// POST: /Account/ExternalLoginConfirmation  

[HttpPost]  

[AllowAnonymous]  

[ValidateAntiForgeryToken]  

public async Task<ActionResult>  

ExternalLoginConfirmation(ExternalLoginConfirmationViewModel model, string returnUrl)
{

```



```

    if (User.Identity.IsAuthenticated)
    {
        return RedirectToAction("Index", "Manage");
    }

    if (ModelState.IsValid)
    {
        // Get the information about the user from the external login provider
        var info = await AuthenticationManager.GetExternalLoginInfoAsync();
        if (info == null)
        {
            return View("ExternalLoginFailure");
        }
        var user = new ApplicationUser { UserName = model.Email, Email =
model.Email };
        var result = await UserManager.CreateAsync(user);
        if (result.Succeeded)
        {
            result = await UserManager.AddLoginAsync(user.Id, info.Login);
            if (result.Succeeded)
            {
                await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
                return RedirectToLocal(returnUrl);
            }
        }
        AddErrors(result);
    }

    ViewBag.ReturnUrl = returnUrl;
    return View(model);
}

// POST: /Account/LogOff
[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult LogOff()
{
    AuthenticationManager.SignOut(DefaultAuthenticationTypes.ApplicationCookie);
    return RedirectToAction("Index", "Home");
}

// GET: /Account/ExternalLoginFailure
[AllowAnonymous]
public ActionResult ExternalLoginFailure()
{
    return View();
}

protected override void Dispose(bool disposing)
{
    if (disposing)
    {
        if (_userManager != null)
        {
            _userManager.Dispose();
            _userManager = null;
        }

        if (_signInManager != null)
        {
            _signInManager.Dispose();
    }
}

```



```

        _signInManager = null;
    }
}

base.Dispose(disposing);
}

#region Helpers
// Used for XSRF protection when adding external logins
private const string XsrfKey = "XsrfId";

private IAuthenticationManager AuthenticationManager
{
    get
    {
        return HttpContext.GetOwinContext().Authentication;
    }
}

private void AddErrors(IdentityResult result)
{
    foreach (var error in result.Errors)
    {
        ModelState.AddModelError("", error);
    }
}

private ActionResult RedirectToLocal(string returnUrl)
{
    if (Url.IsLocalUrl(returnUrl))
    {
        return Redirect(returnUrl);
    }
    return RedirectToAction("Index", "Home");
}

internal class ChallengeResult : HttpUnauthorizedResult
{
    public ChallengeResult(string provider, string redirectUri)
        : this(provider, redirectUri, null)
    {
    }

    public ChallengeResult(string provider, string redirectUri, string userId)
    {
        LoginProvider = provider;
        RedirectUri = redirectUri;
        UserId = userId;
    }

    public string LoginProvider { get; set; }
    public string RedirectUri { get; set; }
    public string UserId { get; set; }

    public override void ExecuteResult(ControllerContext context)
    {
        var properties = new AuthenticationProperties { RedirectUri = RedirectUri };
        if (UserId != null)
        {
            properties.Dictionary[XsrfKey] = UserId;
        }
        context.HttpContext.GetOwinContext().Authentication.Challenge(properties,
LoginProvider);
    }
}

```



```
    }
}
#endifregion
}

}
```



## ManageController.cs

```
using System;
using System.Linq;
using System.Threading.Tasks;
using System.Web;
using System.Web.Mvc;
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.Owin;
using Microsoft.Owin.Security;
using WebApp.Models;

namespace WebApp.Controllers
{
    [Authorize]
    public class ManageController : Controller
    {
        private ApplicationSignInManager _signInManager;
        private ApplicationUserManager _userManager;

        public ManageController()
        {
        }

        public ManageController(ApplicationUserManager userManager,
ApplicationSignInManager signInManager)
        {
            UserManager = userManager;
            SignInManager = signInManager;
        }

        public ApplicationSignInManager SignInManager
        {
            get
            {
                return _signInManager ??
HttpContext.GetOwinContext().Get<ApplicationSignInManager>();
            }
            private set
            {
                _signInManager = value;
            }
        }

        public ApplicationUserManager UserManager
        {
            get
            {
                return _userManager ??
HttpContext.GetOwinContext().GetUserManager<ApplicationUserManager>();
            }
            private set
            {
                _userManager = value;
            }
        }

        //
        // GET: /Manage/Index
        public async Task<ActionResult> Index(ManageMessageId? message)
        {
            ViewBag.StatusMessage =
                message == ManageMessageId.ChangePasswordSuccess ? "Your password has been
changed."
```



```

        : message == ManageMessageId.SetPasswordSuccess ? "Your password has been
set."
        : message == ManageMessageId.SetTwoFactorSuccess ? "Your two-factor
authentication provider has been set."
        : message == ManageMessageId.Error ? "An error has occurred."
        : message == ManageMessageId.AddPhoneSuccess ? "Your phone number was
added."
        : message == ManageMessageId.RemovePhoneSuccess ? "Your phone number was
removed."
        : "";

    var userId = User.Identity.GetUserId();
    var model = new IndexViewModel
    {
        HasPassword = HasPassword(),
        PhoneNumber = await UserManager.GetPhoneNumberAsync(userId),
        TwoFactor = await UserManager.GetTwoFactorEnabledAsync(userId),
        Logins = await UserManager.GetLoginsAsync(userId),
        BrowserRemembered = await
AuthenticationManager.TwoFactorBrowserRememberedAsync(userId)
    };
    return View(model);
}

// POST: /Manage/RemoveLogin
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<ActionResult> RemoveLogin(string loginProvider, string
providerKey)
{
    ManageMessageId? message;
    var result = await UserManager.RemoveLoginAsync(User.Identity.GetUserId(), new
UserLoginInfo(loginProvider, providerKey));
    if (result.Succeeded)
    {
        var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
        if (user != null)
        {
            await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
        }
        message = ManageMessageId.RemoveLoginSuccess;
    }
    else
    {
        message = ManageMessageId.Error;
    }
    return RedirectToAction("ManageLogins", new { Message = message });
}

// GET: /Manage/AddPhoneNumber
public ActionResult AddPhoneNumber()
{
    return View();
}

// POST: /Manage/AddPhoneNumber
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<ActionResult> AddPhoneNumber(AddPhoneNumberViewModel model)
{

```



```

        if (!ModelState.IsValid)
        {
            return View(model);
        }
        // Generate the token and send it
        var code = await
UserManager.GenerateChangePhoneNumberTokenAsync(User.Identity.GetUserId(), model.Number);
        if (UserManager.SmsService != null)
        {
            var message = new IdentityMessage
            {
                Destination = model.Number,
                Body = "Your security code is: " + code
            };
            await UserManager.SmsService.SendAsync(message);
        }
        return RedirectToAction("VerifyPhoneNumber", new { PhoneNumber = model.Number
});
    }

    //
    // POST: /Manage/EnableTwoFactorAuthentication
    [HttpPost]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> EnableTwoFactorAuthentication()
    {
        await UserManager.SetTwoFactorEnabledAsync(User.Identity.GetUserId(), true);
        var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
        if (user != null)
        {
            await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
        }
        return RedirectToAction("Index", "Manage");
    }

    //
    // POST: /Manage/DisableTwoFactorAuthentication
    [HttpPost]
    [ValidateAntiForgeryToken]
    public async Task<ActionResult> DisableTwoFactorAuthentication()
    {
        await UserManager.SetTwoFactorEnabledAsync(User.Identity.GetUserId(), false);
        var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
        if (user != null)
        {
            await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
        }
        return RedirectToAction("Index", "Manage");
    }

    //
    // GET: /Manage/VerifyPhoneNumber
    public async Task<ActionResult> VerifyPhoneNumber(string phoneNumber)
    {
        var code = await
UserManager.GenerateChangePhoneNumberTokenAsync(User.Identity.GetUserId(), phoneNumber);
        // Send an SMS through the SMS provider to verify the phone number
        return phoneNumber == null ? View("Error") : View(new
VerifyPhoneNumberViewModel { PhoneNumber = phoneNumber });
    }

    //

```



```

// POST: /Manage/VerifyPhoneNumber
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<ActionResult> VerifyPhoneNumber(VerifyPhoneNumberViewModel
model)
{
    if (!ModelState.IsValid)
    {
        return View(model);
    }
    var result = await
UserManager.ChangePhoneNumberAsync(User.Identity.GetUserId(), model.PhoneNumber,
model.Code);
    if (result.Succeeded)
    {
        var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
        if (user != null)
        {
            await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
        }
        return RedirectToAction("Index", new { Message =
ManageMessageId.AddPhoneSuccess });
    }
    // If we got this far, something failed, redisplay form
    ModelState.AddModelError("", "Failed to verify phone");
    return View(model);
}

//
// GET: /Manage/RemovePhoneNumber
public async Task<ActionResult> RemovePhoneNumber()
{
    var result = await UserManager.SetPhoneNumberAsync(User.Identity.GetUserId(),
null);
    if (!result.Succeeded)
    {
        return RedirectToAction("Index", new { Message = ManageMessageId.Error });
    }
    var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
    if (user != null)
    {
        await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
    }
    return RedirectToAction("Index", new { Message =
ManageMessageId.RemovePhoneSuccess });
}

//
// GET: /Manage/ChangePassword
public ActionResult ChangePassword()
{
    return View();
}

//
// POST: /Manage/ChangePassword
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<ActionResult> ChangePassword(ChangePasswordViewModel model)
{
    if (!ModelState.IsValid)
    {

```



```

        return View(model);
    }
    var result = await UserManager.ChangePasswordAsync(User.Identity.GetUserId(),
model.OldPassword, model.NewPassword);
    if (result.Succeeded)
    {
        var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
        if (user != null)
        {
            await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
        }
        return RedirectToAction("Index", new { Message =
ManageMessageId.ChangePasswordSuccess });
    }
    AddErrors(result);
    return View(model);
}

// GET: /Manage/SetPassword
public ActionResult SetPassword()
{
    return View();
}

// POST: /Manage/SetPassword
[HttpPost]
[ValidateAntiForgeryToken]
public async Task<ActionResult> SetPassword(SetPasswordViewModel model)
{
    if (ModelState.IsValid)
    {
        var result = await UserManager.AddPasswordAsync(User.Identity.GetUserId(),
model.NewPassword);
        if (result.Succeeded)
        {
            var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
            if (user != null)
            {
                await SignInManager.SignInAsync(user, isPersistent: false,
rememberBrowser: false);
            }
            return RedirectToAction("Index", new { Message =
ManageMessageId.SetPasswordSuccess });
        }
        AddErrors(result);
    }

    // If we got this far, something failed, redisplay form
    return View(model);
}

// GET: /Manage/ManageLogins
public async Task<ActionResult> ManageLogins(ManageMessageId? message)
{
    ViewBag.StatusMessage =
        message == ManageMessageId.RemoveLoginSuccess ? "The external login was
removed."
        : message == ManageMessageId.Error ? "An error has occurred."
        : "";
    var user = await UserManager.FindByIdAsync(User.Identity.GetUserId());
}

```



```

        if (user == null)
        {
            return View("Error");
        }
        var userLogins = await UserManager.GetLoginsAsync(User.Identity.GetUserId());
        var otherLogins =
AuthenticationManager.GetExternalAuthenticationTypes().Where(auth => userLogins.All(u1 =>
auth.AuthenticationType != u1.LoginProvider)).ToList();
        ViewBag.ShowRemoveButton = user.PasswordHash != null || userLogins.Count > 1;
        return View(new ManageLoginsViewModel
        {
            CurrentLogins = userLogins,
            OtherLogins = otherLogins
        });
    }

    //
    // POST: /Manage/LinkLogin
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult LinkLogin(string provider)
    {
        // Request a redirect to the external login provider to link a login for the
current user
        return new AccountController.ChallengeResult(provider,
Url.Action("LinkLoginCallback", "Manage"), User.Identity.GetUserId());
    }

    //
    // GET: /Manage/LinkLoginCallback
    public async Task<ActionResult> LinkLoginCallback()
    {
        var loginInfo = await AuthenticationManager.GetExternalLoginInfoAsync(XsrfKey,
User.Identity.GetUserId());
        if (loginInfo == null)
        {
            return RedirectToAction("ManageLogins", new { Message =
ManageMessageId.Error });
        }
        var result = await UserManager.AddLoginAsync(User.Identity.GetUserId(),
loginInfo.Login);
        return result.Succeeded ? RedirectToAction("ManageLogins") :
RedirectToAction("ManageLogins", new { Message = ManageMessageId.Error });
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing && _userManager != null)
        {
            _userManager.Dispose();
            _userManager = null;
        }

        base.Dispose(disposing);
    }
}

#region Helpers
// Used for XSRF protection when adding external logins
private const string XsrfKey = "XsrfId";

private IAuthenticationManager AuthenticationManager
{
    get
    {

```



```

        return HttpContext.GetOwinContext().Authentication;
    }

    private void AddErrors(IdentityResult result)
    {
        foreach (var error in result.Errors)
        {
            ModelState.AddModelError("", error);
        }
    }

    private bool HasPassword()
    {
        var user = UserManager.FindById(User.Identity.GetUserId());
        if (user != null)
        {
            return user.PasswordHash != null;
        }
        return false;
    }

    private bool HasPhoneNumber()
    {
        var user = UserManager.FindById(User.Identity.GetUserId());
        if (user != null)
        {
            return user.PhoneNumber != null;
        }
        return false;
    }

    public enum ManageMessageId
    {
        AddPhoneSuccess,
        ChangePasswordSuccess,
        SetTwoFactorSuccess,
        SetPasswordSuccess,
        RemoveLoginSuccess,
        RemovePhoneSuccess,
        Error
    }

#endregion
}
}

```



## Models

### User.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class User
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Entity_Id { get; set; }

        [Required]
        [StringLength(50)]
        public string User_Name { get; set; }

        [Required]
        [StringLength(250)]
        public string Email { get; set; }

        public int? Role { get; set; }

        [StringLength(550)]
        public string Password { get; set; }

        [StringLength(250)]
        public string Salt { get; set; }

        public DateTime? Last_Login { get; set; }

        public DateTime? Failed_Login { get; set; }

        public int? Failed_Attempt { get; set; }

        public bool? Is_Locked { get; set; }

        public DateTime? Locked_On { get; set; }

        public bool? Is_Active { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```



## Transaction.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Transaction
    {
        [Key]
        public int Transaction_Id { get; set; }

        [StringLength(20)]
        public string Reference_No { get; set; }

        [StringLength(15)]
        public string Mobile_No { get; set; }

        public int? Entity_Id { get; set; }

        [Column(TypeName = "money")]
        public decimal? Txn_Amt { get; set; }

        [StringLength(50)]
        public string Email { get; set; }

        [StringLength(500)]
        public string Website { get; set; }

        [StringLength(500)]
        public string Callback_Url { get; set; }

        [StringLength(500)]
        public string Response { get; set; }
    }
}
```



## Quiz.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Quizes
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
        public int Quiz_Group_Id { get; set; }

        [StringLength(50)]
        public string Quiz_Name { get; set; }

        [StringLength(250)]
        public string Description { get; set; }
    }
}
```



## Entity.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("Entity")]
    public partial class Entity
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Entity_Id { get; set; }

        [StringLength(50)]
        public string Entity_Name { get; set; }

        [StringLength(6)]
        public string Gender { get; set; }

        public DateTime? DOB { get; set; }

        [StringLength(500)]
        public string Description { get; set; }

        public DateTime? Join_Date { get; set; }

        [StringLength(500)]
        public string Profile_Url { get; set; }

        [StringLength(10)]
        public string Entity_Type { get; set; }

        public int? Rating { get; set; }

        public int? Views { get; set; }

        public int? Likes { get; set; }

        public int? Dislikes { get; set; }

        public DateTime? LastActiveDate { get; set; }
    }
}
```



## DigiApp.cs

```
namespace WebApp.Models
{
    using System;
    using System.Data.Entity;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Linq;

    public partial class DigiApp : DbContext
    {
        public DigiApp()
            : base("name=DigiApp")
        { }

        public virtual DbSet<Article> Articles { get; set; }
        public virtual DbSet<Comment> Comments { get; set; }
        public virtual DbSet<Enrollment> Enrollments { get; set; }
        public virtual DbSet<Entity> Entities { get; set; }
        public virtual DbSet<QuizesData> QuizesDatas { get; set; }
        public virtual DbSet<Quizes> Quizes { get; set; }
        public virtual DbSet<QuizesResponse> QuizesResponses { get; set; }
        public virtual DbSet<QuizesOption> QuizesOptions { get; set; }
        public virtual DbSet<QuizesQuestionsLk> QuizesQuestionsLks { get; set; }
        public virtual DbSet<User> Users { get; set; }
        public virtual DbSet<Question> Questions { get; set; }
        public virtual DbSet<Cart> Carts { get; set; }
        public virtual DbSet<Transaction> Transactions { get; set; }
        public virtual DbSet<Chapter> Chapters { get; set; }
        public virtual DbSet<Course> Courses { get; set; }
        protected override void OnModelCreating(DbModelBuilder modelBuilder)
        {
            modelBuilder.Entity<Course>()
                .Property(e => e.Price)
                .HasPrecision(19, 4);

            modelBuilder.Entity<Enrollment>()
                .Property(e => e.Fees)
                .HasPrecision(19, 4);

            modelBuilder.Entity<Entity>()
                .Property(e => e.Gender)
                .IsFixedLength()
                .IsUnicode(false);

            modelBuilder.Entity<Entity>()
                .Property(e => e.Entity_Type)
                .IsFixedLength()
                .IsUnicode(false);

            modelBuilder.Entity<Cart>()
                .Property(e => e.Price)
                .HasPrecision(19, 4);

            modelBuilder.Entity<Cart>()
                .Property(e => e.Discount)
                .HasPrecision(19, 4);

            modelBuilder.Entity<Transaction>()
                .Property(e => e.Txn_Amt)
                .HasPrecision(19, 4);
        }
    }
}
```



## Course.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("Courses")]
    public partial class Course
    {
        [Key]
        public int Course_Id { get; set; }

        public int? Course_Type { get; set; }

        [StringLength(250)]
        public string SortDescription { get; set; }

        [StringLength(500)]
        public string Title { get; set; }

        public string Description { get; set; }

        [Column(TypeName = "money")]
        public decimal? Price { get; set; }

        public int? Rating { get; set; }

        [StringLength(500)]
        public string Schedule { get; set; }

        [StringLength(50)]
        public string Category { get; set; }

        [StringLength(500)]
        public string ImagePath { get; set; }
    }
}
```



## Question.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Question
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
        public int Question_Id { get; set; }

        public int? Asked_By { get; set; }

        [StringLength(500)]
        public string Title { get; set; }

        public string Description { get; set; }

        public int? Selected_Answer { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```



## Enrollment.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Enrollment
    {
        [Key]
        [Column(Order = 0)]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Entity_Id { get; set; }

        [Key]
        [Column(Order = 1)]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Course_Id { get; set; }

        [Column(TypeName = "money")]
        public decimal? Fees { get; set; }

        public DateTime? Opted_Date { get; set; }

        [Key]
        [Column(Order = 2)]
        [StringLength(50)]
        public string Reference_No { get; set; }

        [StringLength(50)]
        public string Paid_By { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```

## Cart.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("Cart")]
    public partial class Cart
    {
        [Key]
        public int Cart_Id { get; set; }

        [Required]
        [StringLength(20)]
        public string Reference_No { get; set; }

        public int Entity_Id { get; set; }

        public int Course_Id { get; set; }

        [Column(TypeName = "money")]
        public decimal? Price { get; set; }

        [Column(TypeName = "money")]
        public decimal? Discount { get; set; }

        public DateTime? Cart_Date { get; set; }

        public bool? Is_Paid { get; set; }

        public DateTime? Payment_Date { get; set; }
    }
}
```



## Article.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Article
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Article_Id { get; set; }

        public int? Associated_To_Chapter { get; set; }

        [StringLength(255)]
        public string Title { get; set; }

        public int? Status { get; set; }

        public string Body { get; set; }

        public int? Type { get; set; }

        public int? Category { get; set; }

        public int? Likes { get; set; }

        public int? Dislikes { get; set; }

        [StringLength(144)]
        public string Slug { get; set; }

        public int? FavoriteCount { get; set; }

        public int? ViewCount { get; set; }

        public int? Parent { get; set; }

        public int? Author { get; set; }

        public int? LastEditor { get; set; }

        [StringLength(512)]
        public string Tags { get; set; }

        public bool? Is_Published { get; set; }

        public DateTime? Published_On { get; set; }
    }
}
```



## QuizData.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("QuizesData")]
    public partial class QuizesData
    {
        [Key]
        [StringLength(25)]
        public string Quiz_No { get; set; }

        public int? Entity_Id { get; set; }

        public DateTime? Start_Date { get; set; }

        public DateTime? End_Date { get; set; }

        [StringLength(15)]
        public string IP { get; set; }

        [StringLength(50)]
        public string Email { get; set; }

        [StringLength(15)]
        public string Phone { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```

## QuizOptions.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class QuizesOption
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Question_Id { get; set; }

        [StringLength(500)]
        public string OptionValue { get; set; }

        public bool? Is_Correct { get; set; }

        public int? Author { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```



```
}
```

## QuizQuestionsLK.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("QuizesQuestionsLK")]
    public partial class QuizesQuestionsLk
    {
        public int? Course_Id { get; set; }

        public int? Chapter_Id { get; set; }

        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Question_Id { get; set; }
    }
}
```

## QuizResponse.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    [Table("QuizesResponse")]
    public partial class QuizesResponse
    {
        [Key]
        [Column(Order = 0)]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Quiz_Id { get; set; }

        [Key]
        [Column(Order = 1)]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Question_Id { get; set; }

        [Key]
        [Column(Order = 2)]
        [StringLength(25)]
        public string Quiz_No { get; set; }

        public int? Response { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```



## Chapter.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Chapter
    {
        [Key]
        public int Chapter_Id { get; set; }

        public int? Course_Id { get; set; }

        [StringLength(250)]
        public string Chapter_Title { get; set; }

        public string Description { get; set; }

        [StringLength(500)]
        public string Schedule { get; set; }

        public int? Parent { get; set; }

        public DateTime? DTS { get; set; }
    }
}
```

## Comment.cs

```
namespace WebApp.Models
{
    using System;
    using System.Collections.Generic;
    using System.ComponentModel.DataAnnotations;
    using System.ComponentModel.DataAnnotations.Schema;
    using System.Data.Entity.Spatial;

    public partial class Comment
    {
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.None)]
        public int Comment_Id { get; set; }

        public int? Article_Id { get; set; }

        [StringLength(500)]
        public string Body { get; set; }

        public DateTime? CreationDate { get; set; }

        public int? Author { get; set; }

        public bool? Approved { get; set; }
    }
}
```



```

        public bool? Is_Private { get; set; }

        public bool? Is_Editor { get; set; }
    }
}

```

## App\_Start Helpers

### BundleConfig.cs

```

using System.Web;
using System.Web.Optimization;

namespace WebApp
{
    public class BundleConfig
    {

        public static void RegisterBundles(BundleCollection bundles)
        {
            bundles.Add(new ScriptBundle("~/bundles/jquery").Include(
                        "~/Scripts/jquery-{version}.js"));

            bundles.Add(new ScriptBundle("~/bundles/jqueryval").Include(
                        "~/Scripts/jquery.validate*"));

            bundles.Add(new ScriptBundle("~/bundles/modernizr").Include(
                        "~/Scripts/modernizr-*"));

            bundles.Add(new ScriptBundle("~/bundles/bootstrap").Include(
                        "~/Scripts/bootstrap.js",
                        "~/Scripts/respond.js"));

            bundles.Add(new StyleBundle("~/Content/css").Include(
                        "~/Content/bootstrap.css",
                        "~/Content/site.css"));
        }
    }
}

```

### FilterConfig.cs

```

using System.Web;
using System.Web.Mvc;

namespace WebApp
{
    public class FilterConfig
    {
        public static void RegisterGlobalFilters(GlobalFilterCollection filters)
        {
            filters.Add(new HandleErrorAttribute());
        }
    }
}

```



```
}
```

## IdentityConfig.cs

```
using System;
using System.Collections.Generic;
using System.Data.Entity;
using System.Linq;
using System.Security.Claims;
using System.Threading.Tasks;
using System.Web;
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.EntityFramework;
using Microsoft.AspNet.Identity.Owin;
using Microsoft.Owin;
using Microsoft.Owin.Security;
using WebApp.Models;

namespace WebApp
{
    public class EmailService : IIdentityMessageService
    {
        public Task SendAsync(IdentityMessage message)
        {
            // Plug in your email service here to send an email.
            return Task.FromResult(0);
        }
    }

    public class SmsService : IIdentityMessageService
    {
        public Task SendAsync(IdentityMessage message)
        {
            // Plug in your SMS service here to send a text message.
            return Task.FromResult(0);
        }
    }

    // Configure the application user manager used in this application. UserManager is
    // defined in ASP.NET Identity and is used by the application.
    public class ApplicationUserManager : UserManager<ApplicationUser>
    {
        public ApplicationUserManager(IUserStore<ApplicationUser> store)
            : base(store)
        {
        }

        public static ApplicationUserManager
Create(IdentityFactoryOptions<ApplicationUserManager> options, IOwinContext context)
        {
            var manager = new ApplicationUserManager(new
UserStore<ApplicationUser>(context.Get<ApplicationDbContext>()));
            // Configure validation logic for usernames
            manager.UserValidator = new UserValidator<ApplicationUser>(manager)
            {
                AllowOnlyAlphanumericUserNames = false,
                RequireUniqueEmail = true
            };

            // Configure validation logic for passwords
        }
    }
}
```



```

        manager.PasswordValidator = new PasswordValidator
    {
        RequiredLength = 6,
        RequireNonLetterOrDigit = true,
        RequireDigit = true,
        RequireLowercase = true,
        RequireUppercase = true,
    };

    // Configure user lockout defaults
    manager.UserLockoutEnabledByDefault = true;
    manager.DefaultAccountLockoutTimeSpan = TimeSpan.FromMinutes(5);
    manager.MaxFailedAccessAttemptsBeforeLockout = 5;

    // Register two factor authentication providers. This application uses Phone
    and Emails as a step of receiving a code for verifying the user
    // You can write your own provider and plug it in here.
    manager.RegisterTwoFactorProvider("Phone Code", new
PhoneNumberTokenProvider<ApplicationUser>
{
    MessageFormat = "Your security code is {0}"
});
    manager.RegisterTwoFactorProvider("Email Code", new
EmailTokenProvider<ApplicationUser>
{
    Subject = "Security Code",
    BodyFormat = "Your security code is {0}"
});
    manager.EmailService = new EmailService();
    manager.SmsService = new SmsService();
    var dataProtectionProvider = options.DataProtectionProvider;
    if (dataProtectionProvider != null)
    {
        manager.UserTokenProvider =
            new
DataProtectorTokenProvider<ApplicationUser>(dataProtectionProvider.Create("ASP.NET
Identity"));
    }
    return manager;
}

// Configure the application sign-in manager which is used in this application.
public class ApplicationSignInManager : SignInManager<ApplicationUser, string>
{
    public ApplicationSignInManager(ApplicationUserManager userManager,
IAuthenticationManager authenticationManager)
        : base(userManager, authenticationManager)
    {

    }

    public override Task<ClaimsIdentity> CreateUserIdentityAsync(ApplicationUser user)
    {
        return user.GenerateUserIdentityAsync(( ApplicationUserManager )UserManager);
    }

    public static ApplicationSignInManager
Create(IdentityFactoryOptions<ApplicationSignInManager> options, IWinContext context)
    {
        return new
ApplicationSignInManager(context.GetUserManager<ApplicationUserManager>(),
context.Authentication);
    }
}

```



```
}
```

## RouteConfig.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.Routing;

namespace WebApp
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Home", action = "Index", id =
UrlParameter.Optional }
            );
        }
    }
}
```

## Startup.Auth.cs

```
using System;
using Microsoft.AspNet.Identity;
using Microsoft.AspNet.Identity.Owin;
using Microsoft.Owin;
using Microsoft.Owin.Security.Cookies;
using Microsoft.Owin.Security.Google;
using Owin;
using WebApp.Models;

namespace WebApp
{
    public partial class Startup
    {

        public void ConfigureAuth(IAppBuilder app)
        {

            app.CreatePerOwinContext(ApplicationDbContext.Create);

            app.CreatePerOwinContext<ApplicationUserManager>(ApplicationUserManager.Create);

            app.CreatePerOwinContext<ApplicationSignInManager>(ApplicationSignInManager.Create);

            app.UseCookieAuthentication(new CookieAuthenticationOptions
            {
```



```

        AuthenticationType = DefaultAuthenticationTypes.ApplicationCookie,
        LoginPath = new PathString("/Account/Login"),
        Provider = new CookieAuthenticationProvider
    {

        OnValidateIdentity =
SecurityStampValidator.OnValidateIdentity<ApplicationUserManager, ApplicationUser>(
            validateInterval: TimeSpan.FromMinutes(30),
            regenerateIdentity: (manager, user) =>
user.GenerateUserIdentityAsync(manager))
    }
});  

app.UseExternalSignInCookie(DefaultAuthenticationTypes.ExternalCookie);

        app.UseTwoFactorSignInCookie(DefaultAuthenticationTypes.TwoFactorCookie,
TimeSpan.FromMinutes(5));

app.UseTwoFactorRememberBrowserCookie(DefaultAuthenticationTypes.TwoFactorRememberBrowserCookie);

    }
}
}

```

## WebApiConfig.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web.Http;

namespace WebApp
{
    public static class WebApiConfig
    {
        public static void Register(HttpConfiguration config)
        {
            // Web API configuration and services

            // Web API routes
            config.MapHttpAttributeRoutes();

            config.Routes.MapHttpRoute(
                name: "DefaultApi",
                routeTemplate: "api/{controller}/{id}",
                defaults: new { id = RouteParameter.Optional }
            );
        }
    }
}

```

## Global.asax.cs

```
using System;
```



```

using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Http;
using System.Web.Mvc;
using System.Web.Optimization;
using System.Web.Routing;

namespace WebApp
{
    public class MvcApplication : System.Web.HttpApplication
    {
        protected void Application_Start()
        {
            AreaRegistration.RegisterAllAreas();
            GlobalConfiguration.Configure(WebApiConfig.Register);
            FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);
            RouteConfig.RegisterRoutes(RouteTable.Routes);
            BundleConfig.RegisterBundles(BundleTable.Bundles);
        }
    }
}

```

### **enAreaRegistration.cs**

```

using System.Web.Mvc;

namespace WebApp.Areas.en
{
    public class enAreaRegistration : AreaRegistration
    {
        public override string AreaName
        {
            get
            {
                return "en";
            }
        }

        public override void RegisterArea(AreaRegistrationContext context)
        {
            context.MapRoute(
                "en_default",
                "en/{controller}/{action}/{id}",
                new { action = "Index", id = UrlParameter.Optional }
            );
        }
    }
}

```

### **adminAreaRegistration.cs**

```

using System.Web.Mvc;

namespace WebApp.Areas.admin
{
    public class adminAreaRegistration : AreaRegistration
    {

```



```
{  
    public override string AreaName  
    {  
        get  
        {  
            return "admin";  
        }  
    }  
  
    public override void RegisterArea(AreaRegistrationContext context)  
    {  
        context.MapRoute(  
            "admin_default",  
            "admin/{controller}/{action}/{id}",  
            new { action = "Index", id = UrlParameter.Optional }  
        );  
    }  
}
```



## Package.Config

```
<?xml version="1.0" encoding="utf-8"?>
<packages>
    <package id="Antlr" version="3.4.1.9004" targetFramework="net45" />
    <package id="bootstrap" version="3.0.0" targetFramework="net45" />
    <package id="EntityFramework" version="6.1.3" targetFramework="net45" />
    <package id="jQuery" version="1.10.2" targetFramework="net45" />
    <package id="jQuery.Validation" version="1.11.1" targetFramework="net45" />
    <package id="Microsoft.AspNet.Identity.Core" version="2.2.1" targetFramework="net45" />
    <package id="Microsoft.AspNet.Identity.EntityFramework" version="2.2.1"
targetFramework="net45" />
    <package id="Microsoft.AspNet.Identity.Owin" version="2.2.1" targetFramework="net45" />
    <package id="Microsoft.AspNet.Mvc" version="5.2.3" targetFramework="net45" />
    <package id="Microsoft.AspNet.Razor" version="3.2.3" targetFramework="net45" />
    <package id="Microsoft.AspNet.Web.Optimization" version="1.1.3" targetFramework="net45"
/>
    <package id="Microsoft.AspNet.WebApi" version="5.2.3" targetFramework="net45" />
    <package id="Microsoft.AspNet.WebApi.Client" version="5.2.3" targetFramework="net45" />
    <package id="Microsoft.AspNet.WebApi.Core" version="5.2.3" targetFramework="net45" />
    <package id="Microsoft.AspNet.WebApi.WebHost" version="5.2.3" targetFramework="net45" />
    <package id="Microsoft.AspNet.WebPages" version="3.2.3" targetFramework="net45" />
    <package id="Microsoft.CodeDom.Providers.DotNetCompilerPlatform" version="1.0.0"
targetFramework="net45" />
    <package id="Microsoft.jQuery.Unobtrusive.Validation" version="3.2.3"
targetFramework="net45" />
    <package id="Microsoft.Net.Compilers" version="1.0.0" targetFramework="net45"
developmentDependency="true" />
    <package id="Microsoft.Owin" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Owin.Host.SystemWeb" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Owin.Security" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Owin.Security.Cookies" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Owin.Security.Facebook" version="3.0.1" targetFramework="net45"
/>
    <package id="Microsoft.Owin.Security.Google" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Owin.Security.MicrosoftAccount" version="3.0.1"
targetFramework="net45" />
    <package id="Microsoft.Owin.Security.OAuth" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Owin.Security.Twitter" version="3.0.1" targetFramework="net45" />
    <package id="Microsoft.Web.Infrastructure" version="1.0.0.0" targetFramework="net45" />
    <package id="Modernizr" version="2.6.2" targetFramework="net45" />
    <package id="Newtonsoft.Json" version="6.0.4" targetFramework="net45" />
    <package id="Owin" version="1.0" targetFramework="net45" />
    <package id="Respond" version="1.2.0" targetFramework="net45" />
    <package id="WebGrease" version="1.5.2" targetFramework="net45" />
</packages>
```



## Web.config

```
<?xml version="1.0" encoding="utf-8"?>

<configuration>
  <configSections>
    <section name="entityFramework"
    type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework,
    Version=6.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089"
    requirePermission="false" />
  </configSections>
  <connectionStrings>
    <add name="DefaultConnection" connectionString="Data Source=Akshay-pc;Initial
    Catalog=DigiAppDB;Integrated Security=True;Connect
    Timeout=30;Encrypt=False;TrustServerCertificate=True;ApplicationIntent=ReadWrite;MultiSubn
    etFailover=False" providerName="System.Data.SqlClient" />
  </connectionStrings>
  <appSettings>
    <add key="webpages:Version" value="3.0.0.0" />
    <add key="webpages:Enabled" value="false" />
    <add key="ClientValidationEnabled" value="true" />
    <add key="UnobtrusiveJavaScriptEnabled" value="true" />
  </appSettings>
  <system.web>
    <authentication mode="None" />
    <compilation debug="true" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
  </system.web>
  <system.webServer>
    <modules>
      <remove name="FormsAuthentication" />
    </modules>
    <handlers>
      <remove name="ExtensionlessUrlHandler-Integrated-4.0" />
      <remove name="OPTIONSVerbHandler" />
      <remove name="TRACEVerbHandler" />
      <add name="ExtensionlessUrlHandler-Integrated-4.0" path="*.*" verb="*"
    type="System.Web.Handlers.TransferRequestHandler"
    preCondition="integratedMode,runtimeVersionv4.0" />
    </handlers>
  </system.webServer>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <dependentAssembly>
        <assemblyIdentity name="Microsoft.Owin.Security" publicKeyToken="31bf3856ad364e35"
    />
        <bindingRedirect oldVersion="1.0.0.0-3.0.1.0" newVersion="3.0.1.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="Microsoft.Owin.Security.OAuth"
    publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="1.0.0.0-3.0.1.0" newVersion="3.0.1.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="Microsoft.Owin.Security.Cookies"
    publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="1.0.0.0-3.0.1.0" newVersion="3.0.1.0" />
      </dependentAssembly>
      <dependentAssembly>
        <assemblyIdentity name="Microsoft.Owin" publicKeyToken="31bf3856ad364e35" />
        <bindingRedirect oldVersion="1.0.0.0-3.0.1.0" newVersion="3.0.1.0" />
      </dependentAssembly>
    </assemblyBinding>
  </runtime>

```



```

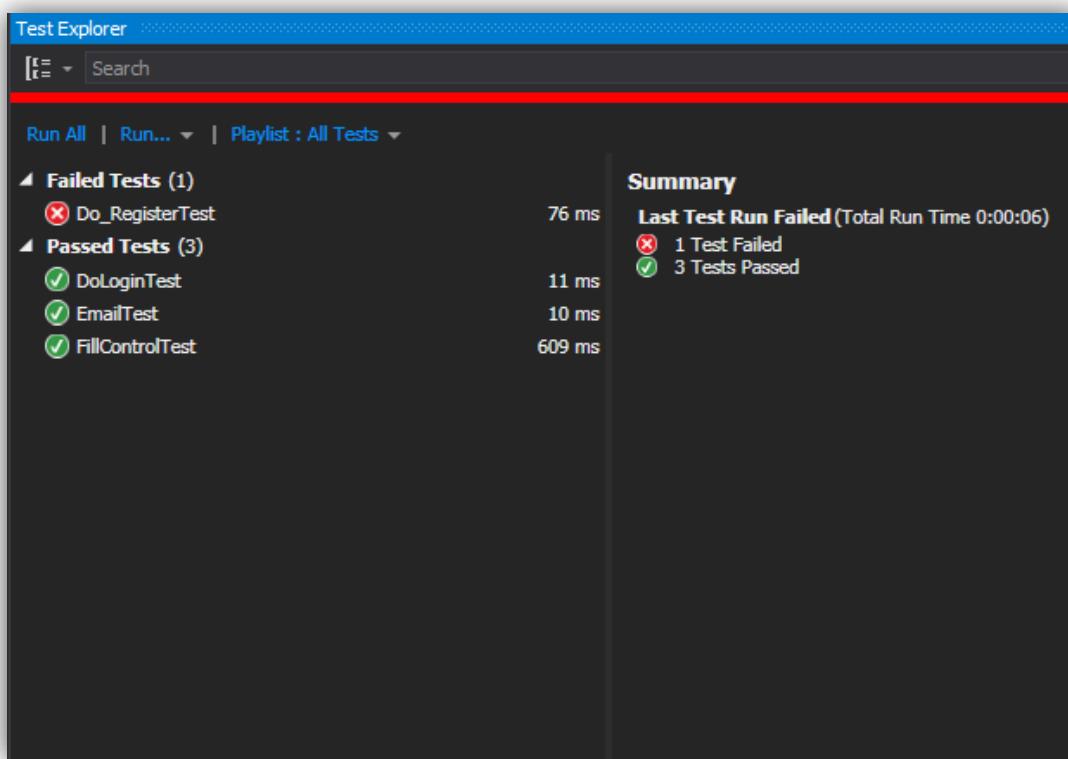
<dependentAssembly>
    <assemblyIdentity name="Newtonsoft.Json" culture="neutral"
publicKeyToken="30ad4fe6b2a6aeed" />
    <bindingRedirect oldVersion="0.0.0.0-6.0.0.0" newVersion="6.0.0.0" />
</dependentAssembly>
<dependentAssembly>
    <assemblyIdentity name="System.Web.Optimization" publicKeyToken="31bf3856ad364e35"
/>
    <bindingRedirect oldVersion="1.0.0.0-1.1.0.0" newVersion="1.1.0.0" />
</dependentAssembly>
<dependentAssembly>
    <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35" />
    <bindingRedirect oldVersion="0.0.0.0-1.5.2.14234" newVersion="1.5.2.14234" />
</dependentAssembly>
<dependentAssembly>
    <assemblyIdentity name="System.Web.Helpers" publicKeyToken="31bf3856ad364e35" />
    <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
</dependentAssembly>
<dependentAssembly>
    <assemblyIdentity name="System.Web.Mvc" publicKeyToken="31bf3856ad364e35" />
    <bindingRedirect oldVersion="1.0.0.0-5.2.3.0" newVersion="5.2.3.0" />
</dependentAssembly>
<dependentAssembly>
    <assemblyIdentity name="System.Web.WebPages" publicKeyToken="31bf3856ad364e35" />
    <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
</dependentAssembly>
</assemblyBinding>
</runtime>
<entityFramework>
    <defaultConnectionFactory
type="System.Data.Entity.Infrastructure.LocalDbConnectionFactory, EntityFramework">
        <parameters>
            <parameter value="mssqllocaldb" />
        </parameters>
    </defaultConnectionFactory>
    <providers>
        <provider invariantName="System.Data.SqlClient"
type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer" />
    </providers>
</entityFramework>
<system.codedom>
    <compilers>
        <compiler language="c#;cs;csharp" extension=".cs"
type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.CSharpCodeProvider,
Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" warningLevel="4" compilerOptions="/langversion:6
/nowarn:1659;1699;1701" />
        <compiler language="vb;vbs;visualbasic;vbscript" extension=".vb"
type="Microsoft.CodeDom.Providers.DotNetCompilerPlatform.VBCodeProvider,
Microsoft.CodeDom.Providers.DotNetCompilerPlatform, Version=1.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" warningLevel="4" compilerOptions="/langversion:14
/nowarn:41008 /define:_MYTYPE="Web"; /optionInfer+" />
    </compilers>
</system.codedom>
</configuration>

```



# 5. TESTING AND MAINTENANCE

## 5.1 Unit Test Screen



### Test Code Article

```
[TestClass()]
public class ArticlesTests
{
    [TestMethod()]
    public void FillControlTest()
    {
        Articles objArticles = new Articles();
        objArticles = objArticles.Get_Articles_By_Article_Id(1008);
        Assert.IsNotNull(objArticles);
    }
}
```

### Login Test

```
[TestClass()]
public class LoginPageTests
{
    [TestMethod()]
    public void DoLoginTest()
```



```

    {
        string Email = "nowgray@gmail.com";
        string Password = "123456";

        Users objUser = new Users();
        bool Actual = objUser.USERS_Login(Email, Password);

        Assert.IsTrue(Actual);
    }

    [TestMethod()]
    public void EmailTest()
    {
        string Email = "nowgray@gmail.com";

        Users objUser = new Users();
        bool Actual = objUser.Users_GetByEmail(Email) != null ? true : false;

        Assert.IsTrue(Actual);
    }
}

```

- Verify that the login screen is having option to enter username and password with submit button and option of forgot password
- Verify that user is able to login with valid username and password
- Verify that user is not able to login with invalid username and password
- Verify that validation message gets displayed in case user leaves username or password field as blank
- Verify that validation message is displayed in case user exceeds the character limit of the user name and password fields
- Verify that there is reset button to clear the field's text
- Verify if there is checkbox with label "remember password" in the login page
- Verify that the password is in encrypted form when entered
- Verify that there is limit on the total number of unsuccessful attempts
- For security point of view, in case of incorrect credentials user is displayed the message like "incorrect username or password" instead of exact message pointing at the field that is incorrect. As message like "incorrect username" will aid hacker in bruteforcing the fields one by one
- Verify the timeout of the login session
- Verify if the password can be copy-pasted or not
- Verify that once logged in, clicking back button doesn't logout user
- Verify if SQL Injection attacks works on login page
- Verify if XSS vulnerability work on login page



## Validation

### Register Yourself

The Email field is required. ↗  
The Password field is required. ↗

Your Email

---

Your password

---

Confirm password

---

**Register to eLearning**

Already have an account? [Sign In](#)



# COST ESTIMATION & SUMMARY

mcsp60-Project			
as of Sun 4/1/18			
<b>Dates</b>			
Start: Fri 11/3/17 Finish: Mon 2/19/18			
Baseline Start: NA Baseline Finish: NA			
Actual Start: Fri 11/3/17 Actual Finish: NA			
Start Variance: 0 days Finish Variance: 0 days			
<b>Duration</b>			
Scheduled: 66.52 days? Remaining: 0.49 days?			
Baseline: 0 days Actual: 66.03 days			
Variance: 66.52 days? Percent Complete: 99%			
<b>Work</b>			
Scheduled: 501.97 hrs Remaining: 4.8 hrs			
Baseline: 0 hrs Actual: 497.17 hrs			
Variance: 501.97 hrs Percent Complete: 99%			
<b>Costs</b>			
Scheduled: \$523.64 Remaining: \$0.60			
Baseline: \$0.00 Actual: \$523.04			
<b>Task Status</b>			
Tasks not yet started: 0		<b>Resource Status</b>	
Tasks in progress: 3		Work Resources: 1	
Tasks completed: 31		Overallocated Work Resources: 0	
Total Tasks: 34		Material Resources: 1	
		Total Resources: 2	

## Code Metrics

- Scope
- Project Namespace
- Type
- Member
- Maintainability Index
- Cyclomatic Complexity
- Depth of Inheritance
- Class Coupling



- Lines of Code

Complete chart is attached



Scope	Project	Namespace	Type	Member	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
Project	WebApp (Debug)				89	604	4	260	985
Name space	WebApp (Debug)	WebApp			89	24	4	79	56
Type	WebApp (Debug)	WebApp	ApplicationSignInManager		90	3	2	9	5
Member	WebApp (Debug)	WebApp	ApplicationSignInManager	ApplicationSignInManager(ApplicationUserManager, IAuthenticationManager)	95	1		4	1
Member	WebApp (Debug)	WebApp	ApplicationSignInManager	Create(IdentityFactoryOptions<ApplicationSignInManager>, IOwinContext) : ApplicationSignInManager	83	1		4	2
Member	WebApp (Debug)	WebApp	ApplicationSignInManager	CreateUserIdentityAsync(ApplicationUser) : Task<ClaimsIdentity>	85	1		5	2
Type	WebApp (Debug)	WebApp	ApplicationUserManager		63	3	3	21	16
Member	WebApp (Debug)	WebApp	ApplicationUserManager	ApplicationUserManager(IUserStore<ApplicationUser>)	98	1		3	1
Member	WebApp (Debug)	WebApp	ApplicationUserManager	Create(IdentityFactoryOptions<ApplicationUserManager>, IOwinContext) : ApplicationUserManager	55	2		19	15
Type	WebApp (Debug)	WebApp	BundleConfig		75	2	1	5	6
Member	WebApp (Debug)	WebApp	BundleConfig	BundleConfig()	100	1		0	1
Member	WebApp (Debug)	WebApp	BundleConfig	RegisterBundles(BundleCollection) : void	68	1		5	5
Type	WebApp (Debug)	WebApp	DigiAppDBDataSet		100	0	3	10	0

Type	WebApp (Debug)	WebApp	DigiAppDBDataSet .EntityDataTable		100	0	4	3	0
Type	WebApp (Debug)	WebApp	DigiAppDBDataSet .EntityRow		100	0	2	1	0
Type	WebApp (Debug)	WebApp	DigiAppDBDataSet .UsersDataTable		100	0	4	3	0
Type	WebApp (Debug)	WebApp	DigiAppDBDataSet .UsersRow		100	0	2	1	0
Type	WebApp (Debug)	WebApp	EmailService		94	2	1	3	3
Member	WebApp (Debug)	WebApp	EmailService	EmailService()	100	1		0	1
Member	WebApp (Debug)	WebApp	EmailService	SendAsync(IdentityMessage) : Task	86	1		2	2
Type	WebApp (Debug)	WebApp	FilterConfig		95	2	1	2	2
Member	WebApp (Debug)	WebApp	FilterConfig	FilterConfig()	100	1		0	1
Member	WebApp (Debug)	WebApp	FilterConfig	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1		2	1
Type	WebApp (Debug)	WebApp	MvcApplication		79	2	2	12	6
Member	WebApp (Debug)	WebApp	MvcApplication	Application_Start() : void	73	1		11	5
Member	WebApp (Debug)	WebApp	MvcApplication	MvcApplication()	100	1		1	1
Type	WebApp (Debug)	WebApp	RouteConfig		90	2	1	4	3
Member	WebApp (Debug)	WebApp	RouteConfig	RegisterRoutes(RouteCollection) : void	82	1		4	2
Member	WebApp (Debug)	WebApp	RouteConfig	RouteConfig()	100	1		0	1
Type	WebApp	WebApp	SmsService		94	2	1	3	3



	(Debug)								
<b>Mem ber</b>	WebApp (Debug)	WebApp	SmsService	SendAsync(IdentityMessage) : Task	86	1		2	2
<b>Mem ber</b>	WebApp (Debug)	WebApp	SmsService	SmsService()	100	1		0	1
<b>Type</b>	WebApp (Debug)	WebApp	Startup		75	5	1	20	10
<b>Mem ber</b>	WebApp (Debug)	WebApp	Startup	Configuration(IAppBuilder) : void	98	1		1	1
<b>Mem ber</b>	WebApp (Debug)	WebApp	Startup	ConfigureAuth(IAppBuilder) : void	62	3		20	8
<b>Mem ber</b>	WebApp (Debug)	WebApp	Startup	Startup()	100	1		0	1
<b>Type</b>	WebApp (Debug)	WebApp	WebApiConfig		82	1	1	5	2
<b>Mem ber</b>	WebApp (Debug)	WebApp	WebApiConfig	Register(HttpConfiguration) : void	82	1		5	2
<b>Name space</b>	WebApp (Debug)	WebApp.Areas.admin			92	3	2	4	4
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin	adminAreaRegistrat ion		92	3	2	4	4
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin	adminAreaRegistrat ion	adminAreaRegistration()	100	1		1	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin	adminAreaRegistrat ion	AreaName.get() : string	91	1		0	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin	adminAreaRegistrat ion	RegisterArea(AreaRegistrationContext) : void	90	1		3	1
<b>Name space</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers			81	51	3	52	169
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController		75	14	3	27	43
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	ChaptersController()	92	1		2	1



<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Create() : ActionResult	89	1		2	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Create(Chapter) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Delete(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Details(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Dispose(bool) : void	82	2		3	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Edit(Chapter) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Edit(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	Index() : Task<ActionResult>	74	1		6	4
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ChaptersController	RefreshAll() : void	77	3		8	3
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController		73	11	3	22	40
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	CoursesController()	92	1		2	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Create() : ActionResult	89	1		2	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Create(Course) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Delete(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5
<b>Mem</b>	WebApp	WebApp.Areas.admin	CoursesController	Details(int?) : Task<ActionResult>	71	1		7	5



ber	(Debug)	.Controllers								
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Dispose(bool) : void	82	2		3	3	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Edit(Course) : Task<ActionResult>	71	1		11	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Edit(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	CoursesController	Index() : Task<ActionResult>	74	1		6	4	
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	DashboardControl ler		96	2	3	2	3	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	DashboardControl ler	DashboardController()	100	1		1	1	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	DashboardControl ler	Index() : ActionResult	89	1		2	2	
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er		73	11	3	21	40	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Create() : ActionResult	89	1		2	2	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Create(Question) : Task<ActionResult>	71	1		10	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Delete(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Details(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Dispose(bool) : void	82	2		3	3	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Edit(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Edit(Question) : Task<ActionResult>	71	1		10	5	



<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	Index() : Task<ActionResult>	74	1		6	4
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuestionsControll er	QuestionsController()	92	1		2	1
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController		73	11	3	21	40
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Create() : ActionResult	89	1		2	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Create(Quizes) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Delete(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Details(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Dispose(bool) : void	82	2		3	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Edit(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Edit(Quizes) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	Index() : Task<ActionResult>	74	1		6	4
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	QuizesController	QuizesController()	92	1		2	1
<b>Type</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ReportsController		96	2	3	2	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ReportsController	Index() : ActionResult	89	1		2	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.admin .Controllers	ReportsController	ReportsController()	100	1		1	1
<b>Name</b>	WebApp	WebApp.Areas.en			92	3	2	4	4



<b>space</b>	(Debug)								
<b>Type</b>	WebApp (Debug)	WebApp.Areas.en	enAreaRegistration		92	3	2	4	4
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en	enAreaRegistration	AreaName.get() : string	91	1		0	2
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en	enAreaRegistration	enAreaRegistration()	100	1		1	1
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en	enAreaRegistration	RegisterArea(AreaRegistrationContext) : void	90	1		3	1
<b>Name space</b>	WebApp (Debug)	WebApp.Areas.en.Controllers			74	45	3	45	162
<b>Type</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController		73	11	3	21	40
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	BlogController()	92	1		2	1
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Create() : ActionResult	89	1		2	2
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Create(Article) : Task<ActionResult>	71	1		10	5
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Delete(int?) : Task<ActionResult>	71	1		7	5
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Details(int?) : Task<ActionResult>	71	1		7	5
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Dispose(bool) : void	82	2		3	3
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Edit(Article) : Task<ActionResult>	71	1		10	5
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Edit(int?) : Task<ActionResult>	71	1		7	5
<b>Member</b>	WebApp (Debug)	WebApp.Areas.en.Controllers	BlogController	Index() : Task<ActionResult>	74	1		6	4



Type	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController		73	11	3	21	40
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	CoursesController()	92	1		2	1
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Create() : ActionResult	89	1		2	2
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Create(Course) : Task<ActionResult>	71	1		10	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Delete(int?) : Task<ActionResult>	71	1		7	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Details(int?) : Task<ActionResult>	71	1		7	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Dispose(bool) : void	82	2		3	3
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Edit(Course) : Task<ActionResult>	71	1		10	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Edit(int?) : Task<ActionResult>	71	1		7	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	CoursesController	Index() : Task<ActionResult>	74	1		6	4
Type	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er		76	12	3	21	42
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Ask() : ActionResult	89	1		2	2
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Create() : ActionResult	89	1		2	2
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Create(Question) : Task<ActionResult>	71	1		10	5
Mem ber	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Delete(int?) : Task<ActionResult>	71	1		7	5
Mem	WebApp	WebApp.Areas.en.Co	QuestionsControll	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5



ber	(Debug)	ntrollers	er							
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Details(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Dispose(bool) : void	82	2		3	3	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Edit(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Edit(Question) : Task<ActionResult>	71	1		10	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	Index() : Task<ActionResult>	74	1		6	4	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuestionsControll er	QuestionsController()	92	1		2	1	
Type	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController		73	11	3	21	40	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Create() : ActionResult	89	1		2	2	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Create(Quizes) : Task<ActionResult>	71	1		10	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Delete(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Details(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Dispose(bool) : void	82	2		3	3	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Edit(int?) : Task<ActionResult>	71	1		7	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Edit(Quizes) : Task<ActionResult>	71	1		10	5	
<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	Index() : Task<ActionResult>	74	1		6	4	



<b>Mem ber</b>	WebApp (Debug)	WebApp.Areas.en.Co ntrollers	QuizesController	QuizesController()	92	1		2	1
<b>Name space</b>	WebApp (Debug)	WebApp.Controllers			79	69	4	74	170
<b>Type</b>	WebApp (Debug)	WebApp.Controllers	AccountController		75	40	3	54	103
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	AccountController()	100	1		1	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	AccountController(ApplicationUserManager, ApplicationSignInManager)	82	1		3	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	AddErrors(IdentityResult) : void	76	3		7	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	AuthenticationManager.get() : IAuthenticationManager	84	1		4	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ConfirmEmail(string, string) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	Dispose(bool) : void	68	4		6	8
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ExternalLogin(string, string) : ActionResult	82	1		8	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ExternalLoginCallback(string) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ExternalLoginConfirmation(ExternalLoginConfirm ationViewModel, string) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ExternalLoginFailure() : ActionResult	89	1		3	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ForgotPassword() : ActionResult	89	1		3	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ForgotPassword(ForgotPasswordViewModel) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ForgotPasswordConfirmation() : ActionResult	89	1		3	2
<b>Mem</b>	WebApp	WebApp.Controllers	AccountController	Login(LoginViewModel, string) :	71	1		10	5



<b>ber</b>	(Debug)			Task<ActionResult>					
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	Login(string) : ActionResult	74	2		11	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	LogOff() : ActionResult	78	1		5	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	RedirectToLocal(string) : ActionResult	75	2		3	4
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	Register() : ActionResult	89	1		3	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	Register(RegisterViewModel) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ResetPassword(ResetPasswordViewModel) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ResetPassword(string) : ActionResult	84	2		3	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	ResetPasswordConfirmation() : ActionResult	89	1		3	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	SendCode(SendCodeViewModel) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	SendCode(string, bool) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	SignInManager.get() : ApplicationSignInManager	83	2		4	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	SignInManager.set(ApplicationSignInManager) : void	95	1		1	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	UserManager.get() : ApplicationUserManager	83	2		4	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	UserManager.set(ApplicationUserManager) : void	95	1		1	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	VerifyCode(string, string, bool) : Task<ActionResult>	68	1		7	6
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	AccountController	VerifyCode(VerifyCodeViewModel) : Task<ActionResult>	71	1		10	5



Type	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult		89	10	4	7	15
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	ChallengeResult(string, string)	94	1		0	1
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	ChallengeResult(string, string, string)	78	1		1	4
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	ExecuteResult(ControllerContext) : void	72	2		6	4
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	LoginProvider.get() : string	98	1		0	1
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	LoginProvider.set(string) : void	95	1		0	1
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	RedirectUri.get() : string	98	1		0	1
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	RedirectUri.set(string) : void	95	1		0	1
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	UserId.get() : string	98	1		0	1
Member	WebApp (Debug)	WebApp.Controllers	AccountController .ChallengeResult	UserId.set(string) : void	95	1		0	1
Type	WebApp (Debug)	WebApp.Controllers	BlogController		73	11	3	21	40
Member	WebApp (Debug)	WebApp.Controllers	BlogController	BlogController()	92	1		2	1
Member	WebApp (Debug)	WebApp.Controllers	BlogController	Create() : ActionResult	89	1		2	2
Member	WebApp (Debug)	WebApp.Controllers	BlogController	Create(Article) : Task<ActionResult>	71	1		10	5
Member	WebApp (Debug)	WebApp.Controllers	BlogController	Delete(int?) : Task<ActionResult>	71	1		7	5
Member	WebApp (Debug)	WebApp.Controllers	BlogController	DeleteConfirmed(int) : Task<ActionResult>	71	1		9	5
Mem	WebApp	WebApp.Controllers	BlogController	Details(int?) : Task<ActionResult>	71	1		7	5



<b>ber</b>	(Debug)								
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	BlogController	Dispose(bool) : void	82	2		3	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	BlogController	Edit(Article) : Task<ActionResult>	71	1		10	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	BlogController	Edit(int?) : Task<ActionResult>	71	1		7	5
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	BlogController	Index() : Task<ActionResult>	74	1		6	4
<b>Type</b>	WebApp (Debug)	WebApp.Controllers	HomeController		79	8	3	12	12
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	HomeController	About() : ActionResult	74	2		10	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	HomeController	Contact() : ActionResult	74	2		10	3
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	HomeController	HomeController()	100	1		1	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	HomeController	Index() : ActionResult	89	1		2	2
<b>Mem ber</b>	WebApp (Debug)	WebApp.Controllers	HomeController	Layout() : ActionResult	74	2		10	3
<b>Name space</b>	WebApp (Debug)	WebApp.DigiAppDBD ataSetTableAdapters			100	1	3	14	1
<b>Type</b>	WebApp (Debug)	WebApp.DigiAppDBD ataSetTableAdapters	EntityTableAdapte r		100	0	3	9	0
<b>Type</b>	WebApp (Debug)	WebApp.DigiAppDBD ataSetTableAdapters	TableAdapterManager		100	1	3	9	1
<b>Mem ber</b>	WebApp (Debug)	WebApp.DigiAppDBD ataSetTableAdapters	TableAdapterManager	TableAdapterManager()	100	1		1	1
<b>Type</b>	WebApp (Debug)	WebApp.DigiAppDBD ataSetTableAdapters	UsersTableAdapte r		100	0	3	9	0
<b>Name space</b>	WebApp (Debug)	WebApp.Models	AddPhoneNumber		93	408	4	56	419



	WebApp (Debug)		ViewModel							
<b>Totals</b>						59916	2416	62332	2375	3940



## 6. REPORTS

The screenshot shows a software application window titled "User Activity Log". At the top, there is a toolbar with various icons and a status bar indicating "1 of 1" and "100%". Below the title, the table is displayed:

Entity Name	Gender	DOB	Join Date	Last Active Date
Admin	M	1/1/1991	6/23/2017	6/23/2017
Shiv Bhanu	M	1/4/1993	7/8/2017	7/8/2017
Akshay Mishra	M	4/5/1992	7/8/2017	7/8/2017
Vikram	M	4/7/1996	7/8/2017	7/8/2017

## 7. FUTURE SCOPE

Now a day's there is a big demand by IGNOU student of this type of learning website, where they can find specific information, Currently its limited for **Ignou MCA**, later on I'll update it for all students from Ignou and other universities. I have also plan for offering teachers to come and teach students directly using my website.

I am also planning to launch the **mobile application** very soon.

The controls used in the Front End can be modified and the new web pages can be modified and implemented whenever required.

This system electrically matches according to the portfolio and description and shows the related information on the web. Software scope describes the data and control to be processed, function performance, constraints, interfaces and reliability. We can implement easily this

application. Reusability is possible as and when we require in this application. We can update it in its next version. We can add new features as and when we require. There is flexibility in all the modules. Scope of this document is to put down the requirements, clearly identifying the information needed by the user, the source of the information and outputs expected from the system.

## **8. CONCLUSION**

DIGITAL LEARNING SYSTEM allows learners to register, enroll, learn desired topics and courses. I have decided to develop a project solution that addresses the challenges and allows for a website to remain competitive.

I am using Asp.Net 4.0 MVC and C#, keeping in mind the specifications of the system. For designing the system we have used simple data flow diagrams.

Overall the project teaches us the essential skills like:

- Using system analysis and design techniques like data flow diagram in designing the system.
- Managing users and roles into a software system.
- Web application development and deployment to server.
- Understanding the database handling and query processing using SQL Server.
- Managing project and documentation.



## **9. BIBLIOGRAPHY**

The books, which I referred and which really helped me in building this project in time, are as follows:

- 'Internet and web design' by ISRD Group
- Youtube.com
- IGNOU Books

## **10. REFERENCES**

- [www.wikipedia.org](http://www.wikipedia.org) (Software Engineering Concept)
- [www.ignougroup.com](http://www.ignougroup.com) (Help topics related to project)
- [isites.harvard.edu](http://isites.harvard.edu) (Concept of Project Designing)

