Table of Contents

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Product Functions 4

1.4 Definitions, Acronyms and Abbreviations 4

1.5 References 5

1.6 Feasibility Study 6

1.7 Technologies to be used 9

1.8 Survey of Technology 10

1.9 Planning & Scheduling 12

1.10 Pert Chart 12

1.11 Pert Chart and Giant Chart 13

1.12 Software and Hardware Requirements 14

1.13 Overview 15

2. Overall Description 16

2.1 Use-Case Model Survey 17

2.2 WEB Architecture diagram 18

2.3 ER Diagram 19

3. Specific Requirements 20

3.1 Use-Case Reports 20

3.2 Use-Case / Work Flow Diagram 21

3.3 Data Flow Diagram 30

3.4 Use-Case Diagrams 36

3.5 Sequence Diagram 39

3.6 Class Diagram 42

4. SYSTEM DESIGN 43

4.1 Basic Modules 43

4.2 Schema Design 44

4.3 Security Measures 49

4.5 Database Security 50

5. IMPLEMENTATION AND TESTING 51

5.1 User Interface Design 51

5.2 Coding Details and Code Efficiency 62

5.3 Coding 63

5.4 Testing Technique and Strategies 138

6. RESULTS AND DISCUSSION 142

6.1 Test Case Results 142

6.2 Test Case Result Screens 143

7. CONCLUSIONS 146

7.1 Conclusion 146

7.2 Limitations of the System 146

7.3 Future Scope of the Project 146

BIBLIOGRAPHY 147

APPENDIX A 148

APPENDIX B 149

APPENDIX C 149

1. Introduction

## Purpose

To develop a fully functional and user interactive tool for Accountant to enhance and help various users to manage and arrange their documents efficiently according to their work and by company . To Centralize the documents on server to facilitate effective and efficient file search so that they can handle the files easily and search them efficiently.

## Scope

* Create different users with varied roles and scopes.
* Increase efficiency, save time and money by electronically storing, organizing and accessing documents on your computer or network
* Manage all Documents at centralized location.
* can easily find documents according to the user need.
* Only registered employees can check the documents
* Admin has rights to create users.
* head can register company and upload documents
* only admin can delete the files which was wrongly added or incorrect details files so that others cant edit that file.
* This application can be deployed on an Ethernet server so that outsiders can't hack these files and it will be secure. Only authorized user can see the files.
* File can be upload with details so that it would be easy to find.
* File can be upload only with the name of registered company and their category Like(A,B,C,D,E) and who created that document must be registered user.

**1.3 Product Functions**

The product should be able to create users and head. In addition, it should be possible to add or remove files from server . The user should be able to **Search File by (User, company name and Year)**.

If the user modifies/deletes files through the Windows Explorer interface, the product should be able to update its records to catch these changes. The product utilizes a search capability, to search within labels for files and to search for labels themselves

* **File Search by (User, company name and Year)**
* **Download File**
* **File Upload**
* **Add company**
* **Add User**
* **File Delete**

(a) Add User: The product should be able to create a new user or head.

(b) File Upload: Only 4 specific (Excel,ppt,Doc,Pdf) File format should be uploaded .

(c) Add Company: The product should be able to register company.

(d) File Delete: Only admin should be able to delete the file.

(e) Download File: File should be downloaded only by authorized users

(f) Authorization: Menu should be display according to users permissions.

## Definitions, Acronyms and Abbreviation

**ODMS** – Online Document Management System.

**Admin** – Administrator.

**DM** – Document Manager.

**HTML** – Hyper Text Markup Language.

**XHTML** – Extensible Hypertext Markup Language.

**HTTP** – Hypertext Transfer Protocol.

**ASP** – Active Server Pages.

**JS** - Java Script.

**IIS** – Internet Information Services (Server of windows platorm).

**SQLDB** - Structured Query Language Data Base.

**RDBMS** - Relational Data Base Management System

**Doc** – document.

**xls** -Excel Style Sheet

**doc** - Word Document

**ppt**- Power point presentation

**Pdf**- Portable Document Format  (Acrobat reader file)

## References

## IEEE SRS Format

## TGMC-2008 Sample Synopsis Format.

Problem Definition Provided By TGMC-2008

**1.6 FEASIBILITY STUDY**

Feasibility is the determination of whether or not a project is worth doing the process followed making this determination is called feasibility study. This of determines if a project can and should be taken. Once it has been determined that a project is feasible, the analyst can go ahead and prepare the project specification which finalizes project requirements. Generally, feasibility studies are undertaken within right time constraints and normally culminate in a written and oral feasibility report. The contents and recommendations of such a study will be used as a sound basis for deciding whether to proceed, postpone or cancel the project. Thus, since the feasibility study may lead to the commitment of large resources, it becomes necessary that it should be conducted competently and that no fundamental and that no fundamental errors of judgment are made.

There are following types of inter-related feasibility

* Technical feasibility
* Operational feasibility
* Economic feasibility
* Social feasibility
* Management feasibility
* Legal feasibility
* Time feasibility

* Technical feasibility:

This is concerned with specifying equipment and software and hardware that will successfully satisfy the user requirement. The technical needs off the system may vary considerably, but might include:

* The facility to produce output in a given time.
* Response time under certain conditions.
* Ability to process a certain volume of transaction at a particular speed.
* Facility to communicate data to distant location.

In examining technical feasibility, configuration of the system is given more importance than the actual make of hardware. The configuration should give the complete picture about the system requirements. What speeds of input and output should be achieved at particular quality of printing.

According to the definition of technical feasibility the compatibility between front-end and back-end is very important. In our project the compatibility of both is very good. The compatibility of VISUAL STUDIO 2010 and SQL SERVER 2008 is very good. The speed of output is very good, when we enter the data and click button then the response time is very fast and give result very quickly. I never find difficulty when used complex query or heavy transaction. The speed of transaction is always smooth and constant. This software provides facility to communicate data to distant location.

We use VISUAL STUDIO 2010. The designing of front-end of any project is very important so we select VISUAL STUDIO 2010 as front-end due to following reason:

* Easy implementation of code.
* Well define interface with database.
* Well define hand shaking of SQL SERVER 2008.
* Easy debugging.

I have selected SQL SERVER 2008 because of the following number of reasons.

* Able to handle large data.
* Security.
* Robust RDBMS.
* Backup & recovery.

With the help of above support we remove defect of existing software. To ensure that system does not halt in case of undesired situation or events. Problem affected of any module does not affect any module of the system. A change of hardware does not produce problem.

* Operational Feasibility:

It is mainly related to human organizational and political aspects. The points to be considered are:

* What changes will be brought with the system?
* What organization structures are disturbed?

What new skills will be required? Do the existing staff members have these skills? If not, can they be trained in due course of time?

At present stage all the work is done by manual. So, throughput and response time is too much. Major problem is lack of security check that was must to be applied. Finding out the detail regarding any information was very difficult, because data store in different books and each books at different places. In case of any problem, no one can solve the problem until the master of this field is not present.

## 1.7 Technologies to be used

* ASP.Net Active server Pages
* HTML
* Java Script
* C#
* ADO.Net
* MS sql server
* CSS

**1.8 SURVEY OF TECHNOLOGIES :**

To develop this package different types of tools and database are used which are as follows:

1.ASP.NET 3.5

2.Framwork 3.0

3.Microsoft-sql server 2008

4. Microsoft word processor

**ASP.NET 3.5:**

ASP.NET 3.5 is a Microsoft technology for building the web based application and services. An ASP.NET application consists of forms, controls, classes and procedures.

Forms are windows upon which you build your user interface are the building blocks of the user interface. Controls also called activeX conrols, are interface tools , such as labels ,textbox and command buttons, that you use to display information to the user, gather information from the user, and respond to user actions. Classes are templates from which you can create your own objects at run time.

Procedures are small routines you write that are callable from anywhere in your application. These routines will perform a function for you that you write once but can call many times.

An application is made up of forms, modules and classes. A form is made up of properties, events and controls. Controls are also made up of properties and events

Some Features Of ASP.NET

Language interoperability : a one language code is to interact with other language.

EDP(Event driven programming language): it is a Event Driven Programing language(i.e. we write code on its event and drive the program).

Rapid Application Development(RAID).

Allows the creation of COM components such as activeX controls.

Includes good debugging facilities.

**Microsoft-sql server 2008 :**

Microsoft SQL Server fulfills these responsibilities :

· Reduction and redundancy :

Centralized control of the dba avoids the unnecessary duplication of data and effectively Reduces the total amount of data storage required. It also eliminates the extra processing and of the inconsistencies eg. Same format of grade card for all student.

Sharing data :Any number of application program of user’s can share the same database .

For example we can access the list of study centers.

· Data integrity :

Data integrity means that the data contains in the database both accurate and consistent.

· Data security :

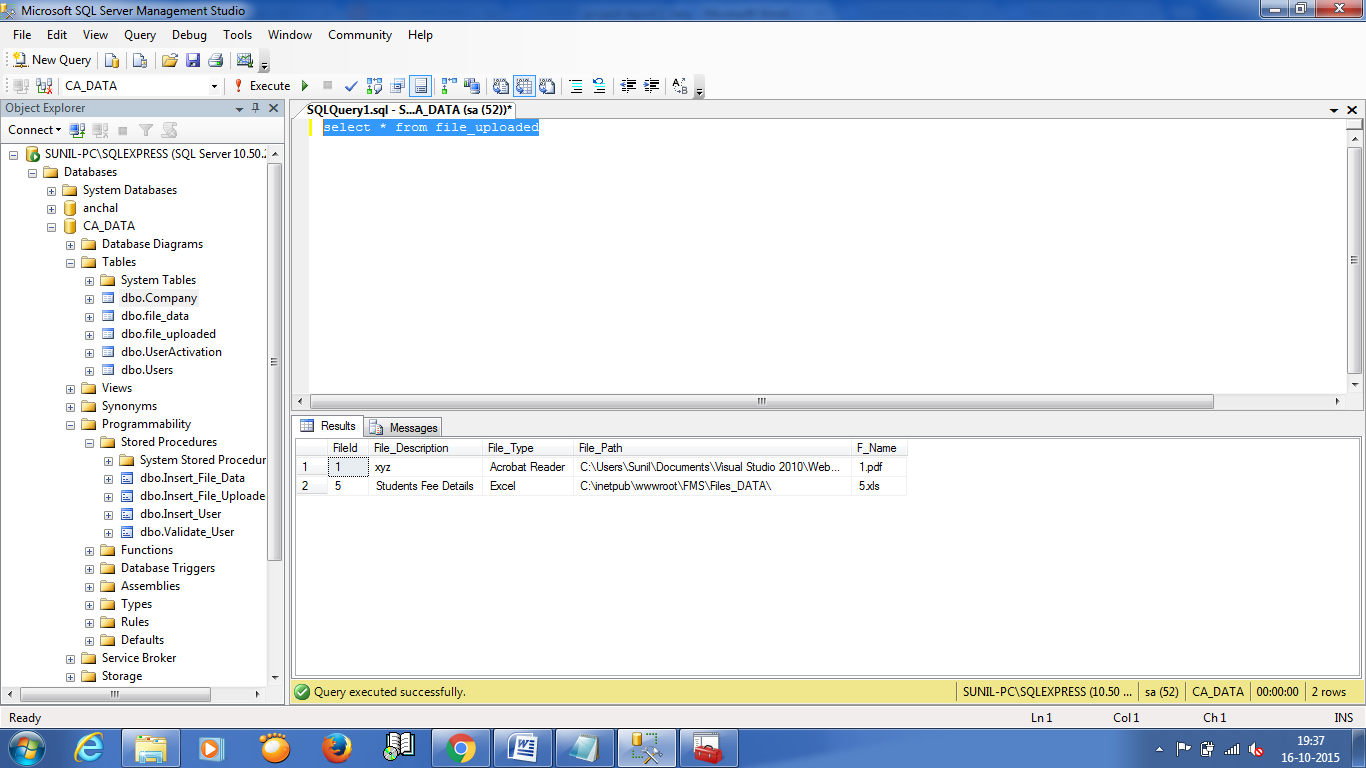
In this facility the confidential data must not be accessed by unauthorized person.

· Rapidly Accessing Data :

SQL Server provides rapid access to data by utilizing indexes and storing frequently accessed data in memory.

· SQL Server Enterprise Manager :

The Enterprise Manager is the central console from which most SQL Server database-management tasks can be coordinated. SQL Enterprise Manager provides a single interface from which all servers in a company can be managed.



**1.9 Planning & Scheduling**

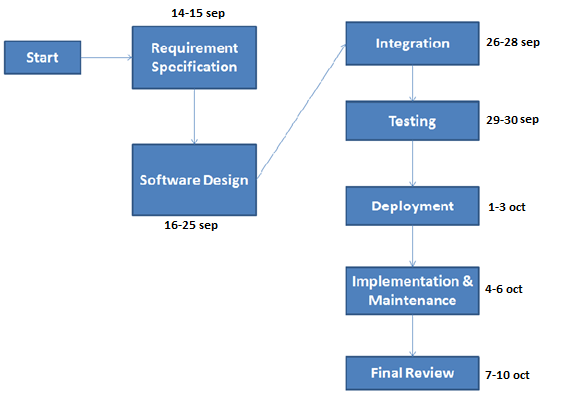
Studying the existing system. This will involve the amount of information, the manpower equipment space and other resources used for the information storage.

Cost benefit analysis study will be done.

The database structure and information storage retrieval process will also be studied and a new system for faster response using operation research technique will be designed.

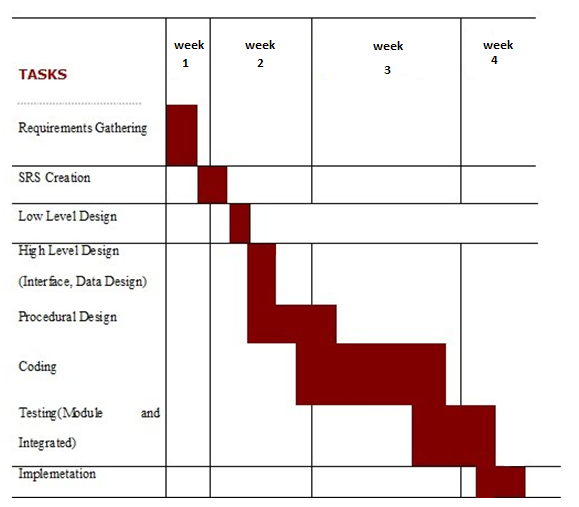
Planning and scheduling plan:

**1.10 Pert chart :**



**1.11 Gantt chart :**

Gantt chart are a project control technique that can be used for several purpose including scheduling and planning. Gantt chart is also known bar chart with each box representing an activity.



**1.12 Software and Hardware Requirement**

**For development, software’s used are:**

Operating System : Windows-7, service pack-2, Microsoft IIS 7.0

Platform : .Net

Technology : Asp.net

Language : C#(C sharp)

Backend : SQL server 2008

For Development : visual studio 2010

For Design : HTML, CSS

**For deployment, software’s used are :**

Operating system : Windows-7, service pack-2

Framework : .Net 3.5

IIS : Configured

Backend : sql server 2008

Hardware Requirements

For Development, Hardware’s used are:

1. Pentium IV(dual core processor operating at 3.6 GHz)

2. 40GB hard disk

3. 1GB RAM

**· For Deployment, Hardware’s used are :**

Minimum

1. Pentium III or better processor

2. 10 GB hard disk

3. 256MB RAM

4. 10-100 MBPS of Network card

Recommended

1. Pentium IV

2. 20GB hard disk

3. 1GB RAM

4. 100-512 MBPS of network card

## 1.13 Overview

This project is a tool to help in managing documents. It is more useful in current Accounting Documents situation. Because of a large number of files to be managed it is not easy to manage it efficiently if done manually. File should be at centralized location so that all the Authorized users can access documents efficiently.

## 2. Overall Description

**2.1 Use case model survey :**

1. Admin: - is responsible for performing:
   1. **Document** **Search: -** can search document by the Company name , Uploader name and by year.
   2. **Download** **File** : - File can be downloaded after searching.
   3. **Document** **Upload** : - Only registered Company file can be uploaded and only behalf of registered user and file should be document (xls,doc,ppt,pdf).
   4. **Add** **company** : - can register the company so that they can manage the documents according to the company.
   5. **Add** **User** : - can create two types of users HEAD and USER.
   6. **File** **Delete** : -can delete the file which is containing wrong details or which was uploaded by mistake, except admin nobody can delete the file bcoz of security reason.
2. Head : - is responsible for dealing with all the proceedings of the project.
   1. **Document** **Search: -** can search document by the company name , Uploader name and by year.
   2. **Download** **File** : - file can be downloaded after searching.
   3. **Document** **Upload** : - Only registered Company file can be uploaded and only behalf of registered user and file should be document (xls,doc,ppt,pdf).
   4. **Add** **company** : - can register the company so that they can manage the documents according to the company.
3. User: -
   1. **Document** **Search: -** can search document by the company name , Uploader name and by year.
   2. **Download** **File** : - file can be downloaded after searching.

## 2.2 Web Architecture Diagram

**HTTP/HTTPS**

**WAS**

**TCP/IP**

**HTML CLIENT**

**(Customer)**

**Client Software**

**(System Users)**

**DB SERVER**

Client Side Application Side DB Server Side

It uses three tier architecture dig where client side application side and DB server are three diff components that improve security in system.

## 2.3 ER Diagram



# 3. Specific Requirements

## 3.1 Use-Case Reports

**Administrator**- is responsible for registering the Company.

1. **Document** **Search: -** can search document by the Company name , Uploader name and by year.
2. **Download** **File** : - File can be downloaded after searching.
3. **Document** **Upload** : - Only registered Company file can be uploaded and only behalf of registered user and file should be document (xls,doc,ppt,pdf).
4. **Add** **company** : - can register the company so that they can manage the documents according to the company.
5. **Add** **User** : - can create two types of users HEAD and USER.
6. **File** **Delete** : -can delete the file which is containing wrong details or which was uploaded by mistake, except admin nobody can delete the file bcoz of security reason.

**Name of Use Case –** Document Search.

**Description –** Administrator can Search the document he wants to view.

**Pre Condition –**The Admin should be logged in the system.

**Normal Flow of Events –**

* A form will open, necessary information will be entered
* A query will be fired to the database.
* A Search will be made for the desired File.
* Relevant output will be given to the user.

**3.2 USE CASE / WORK FLOW DIAGRAM**

**USE CASE DIAGRAM**

Details entered

Required File displayed

Search performed

**Work Flow Diagram**

Details filled in.

Required File displayed

Searching

**Name of Use Case –** Download File

**Description –** Desired file after being searched is being downloaded.

**Pre Condition** The Admin should be logged in the system.

**Normal Flow of Events –**

* A Search for file is made.
* File if found is downloaded.

**USE CASE DIAGRAM**

Search is performed for required file

If file is found then it is downloaded

**WORK FLOW DIAGRAM**

Searching

Downloading file

Head : - is responsible for dealing with all the proceedings of the project.

* **Document** **Search: -** can search document by the company name , Uploader name and by year.
* **Download** **File** : - file can be downloaded after searching.
* **Document** **Upload** : - Only registered Company file can be uploaded and only behalf of registered user and file should be document (xls,doc,ppt,pdf).
* **Add** **company** : - can register the company so that they can manage the documents according to the company.

**Name of Use Case –** Document Search

**Description –** A search will be performed to find out the required document.

**Pre Condition –**

* The Head should be logged in the system.

**Normal Flow of Events –**

* A form will be opened necessary information would be entered.

* Validation of details will be done
* Search for the desired document will be performed.

**USE CASE DIAGRAM**

Enter the Desired details

Perform validation

Search the desired file.

**WORK FLOW DIAGRAM**

Enter the Desired details

Team Members

Perform validation

Search the desired file.

**Name of Use Case –** Document upload

**Description –** User can upload the document he wants to be stored at the server.

**Pre Condition –**

* The user should be logged in the system.

**Normal Flow of Events –**

* A form will be opened necessary information would be entered
* A message would be displayed.
* The user have to choose his company.
* The user have to choose the type of file he wants to upload.

**USE CASE DIAGRAM**

Input Details

Choose Company

Choose the type of File

**Name of Use Case –** Add Company

**Description –** The head can add a new company to database.

**Pre Condition –**

* The head should be logged in..

**Normal Flow of Events –**

* A company would be created.
* The Employees from the company can upload their Documents.

**USE CASE DIAGRAM**

Create a Company

Accessibility to upload documents

**WORK FLOW DIAGRAM**

Creating a Company

Accessibility to upload documents

# User – Can Search or Download any file for which it is authorized.

* **Document** **Search: -** can search document by the company name , Uploader name and by year.
* **Download** **File** : - file can be downloaded after searching.

**Use Case**: Document search.

**Pre Condition –**

* The Head should be logged in the system.

**Normal Flow of Events –**

* A form will be opened necessary information would be entered.

* Validation of details will be done
* Search for the desired document will be performed.

**USE CASE DIAGRAM**

Enter the Desired details

Perform validation

Search the desired file.

**WORK FLOW DIAGRAM**

Enter the Desired details

Team Members

Perform validation

Search the desired file.

**Name of Use Case –** Document upload

**Description –** User can upload the document he wants to be stored at the server.

**Pre Condition –**

* The user should be logged in the system.

**Normal Flow of Events –**

* A form will be opened necessary information would be entered
* A message would be displayed.
* The user have to choose his company.
* The user have to choose the type of file he wants to upload.

**USE CASE DIAGRAM**

Input Details

Choose Company

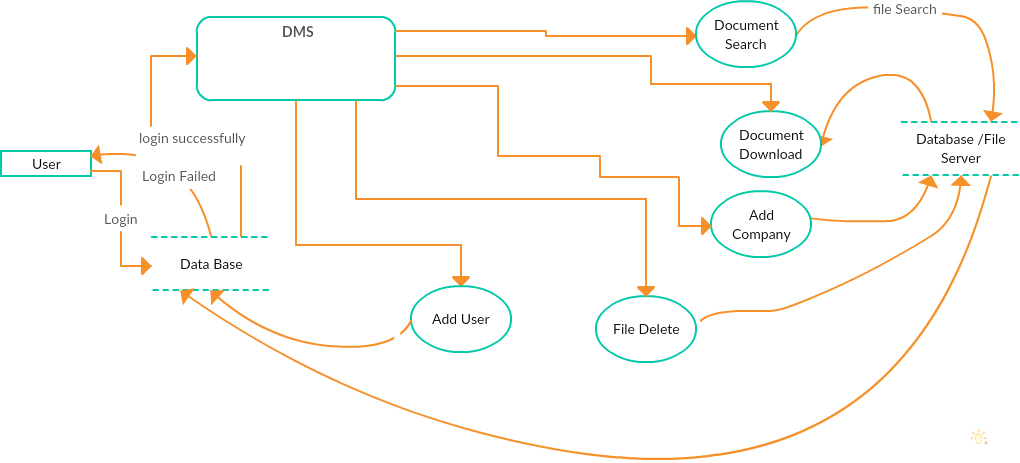
Choose the type of File

**3.3 DATA FLOW DIAGRAM :**

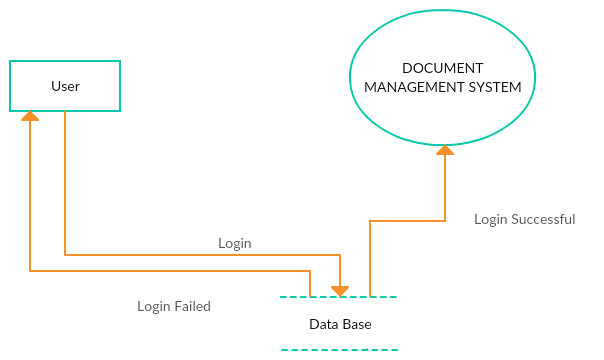
**CONTEXT LEVEL**

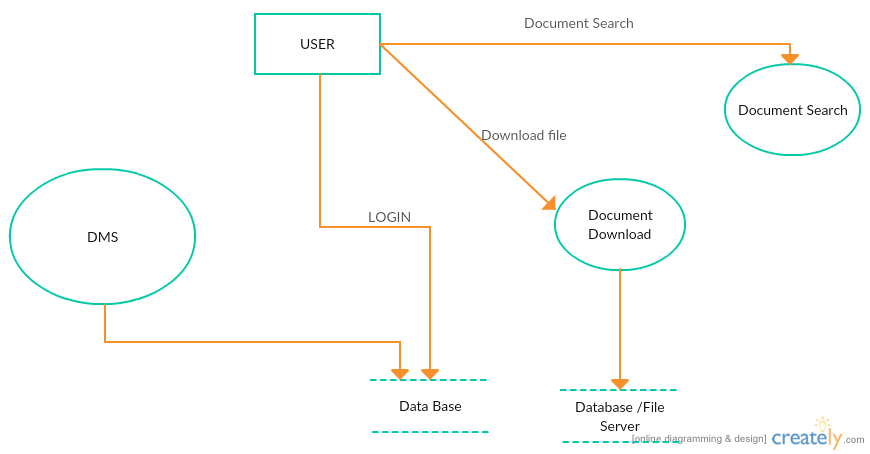
****

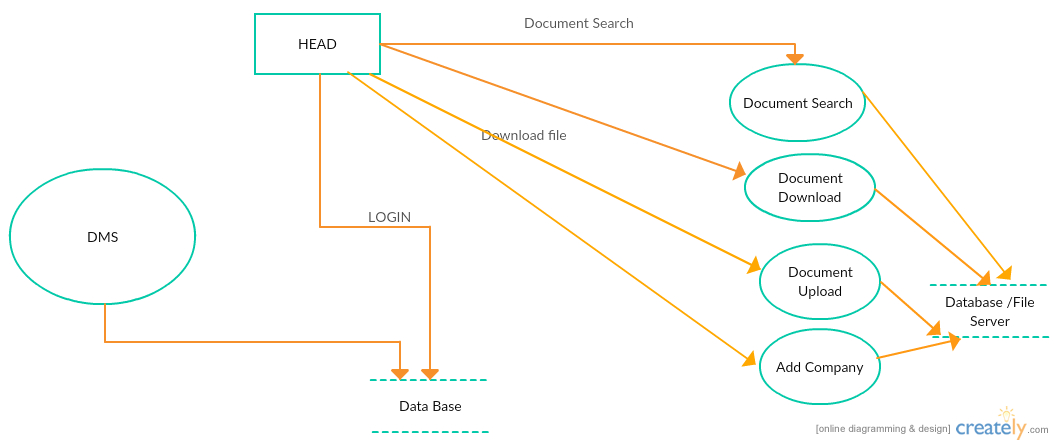
**LEVEL 1**

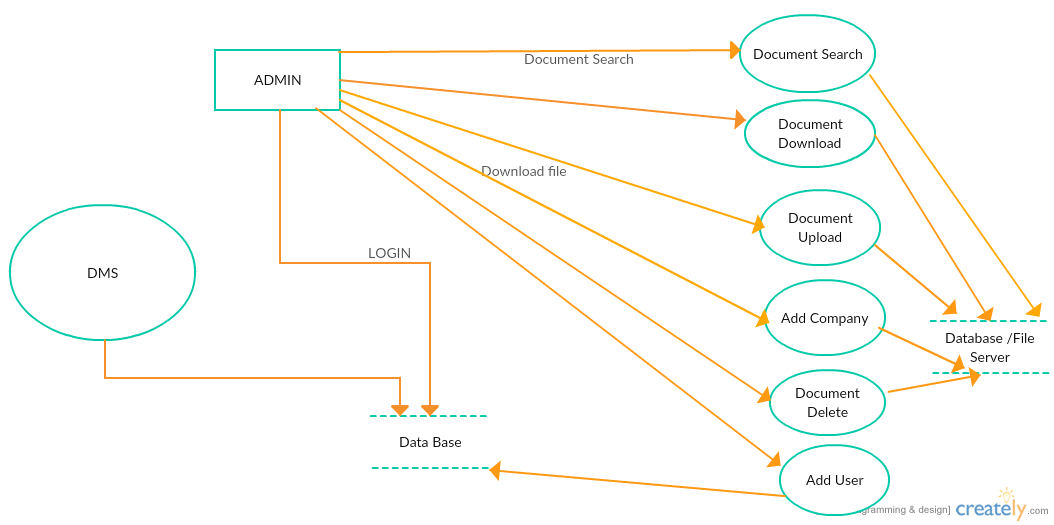
****

**LEVEL 2**

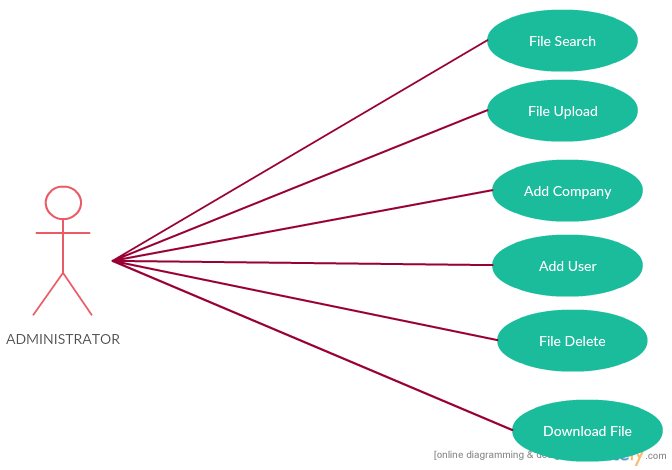
****

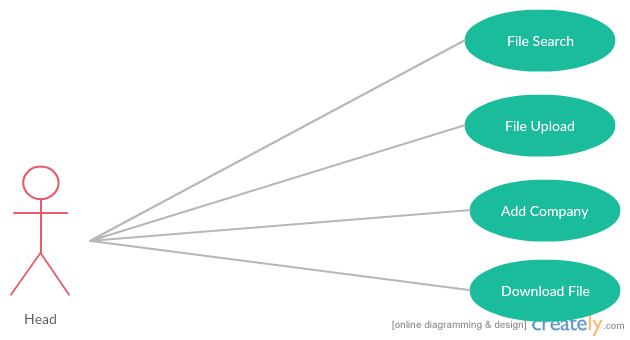
****

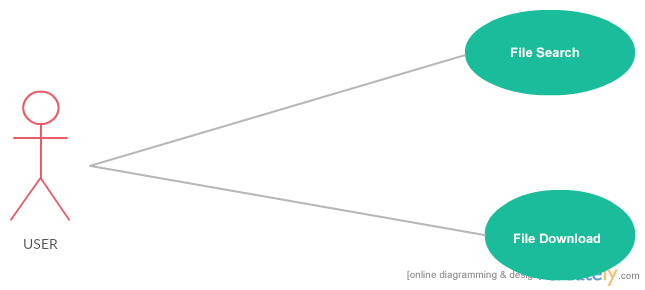
****

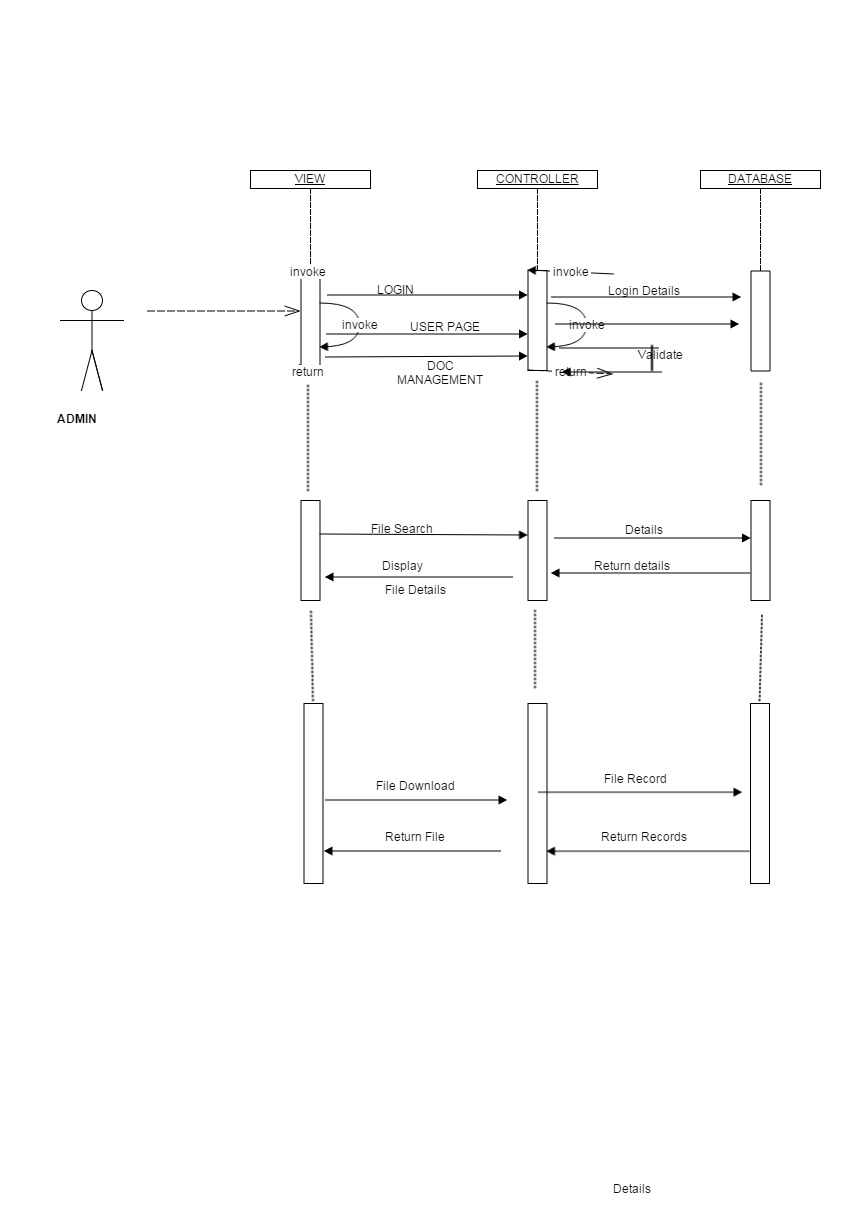
****

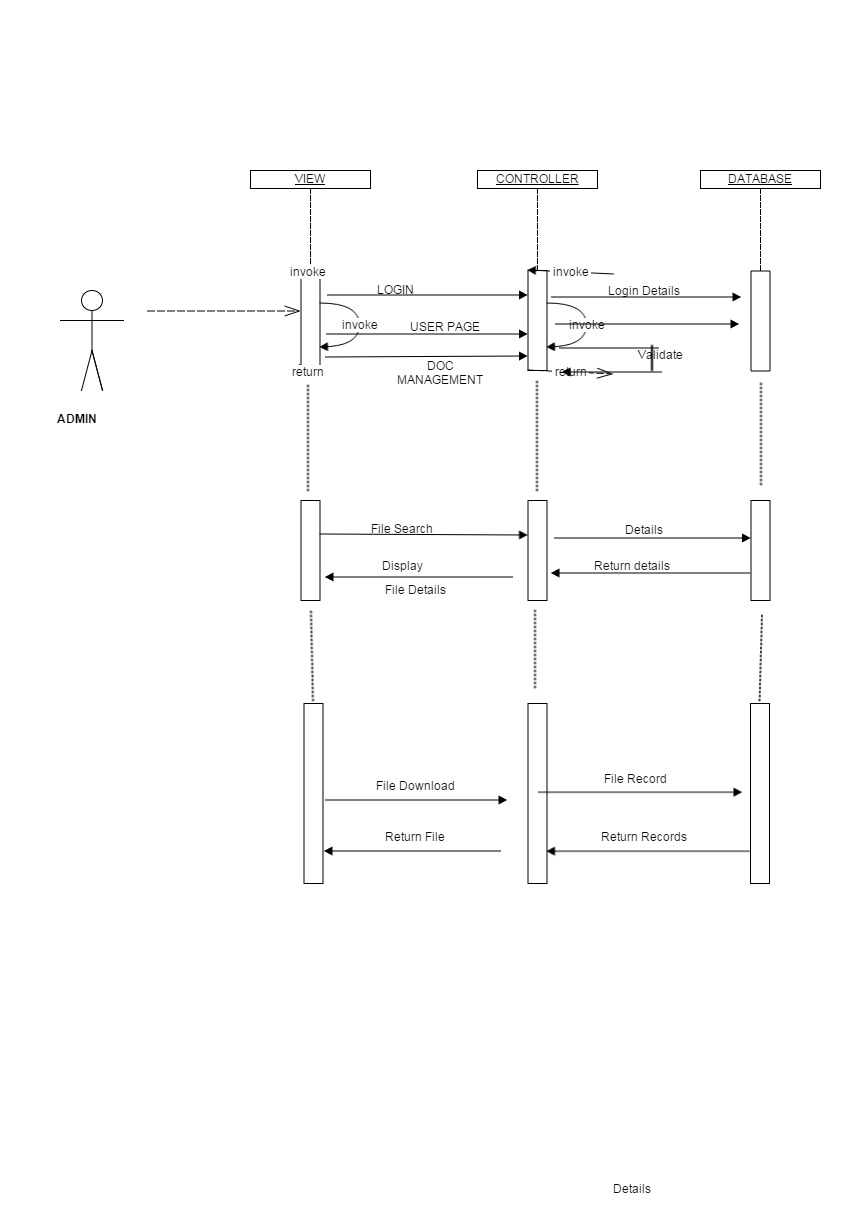
**3.4 USE CASE DIAGRAMS**

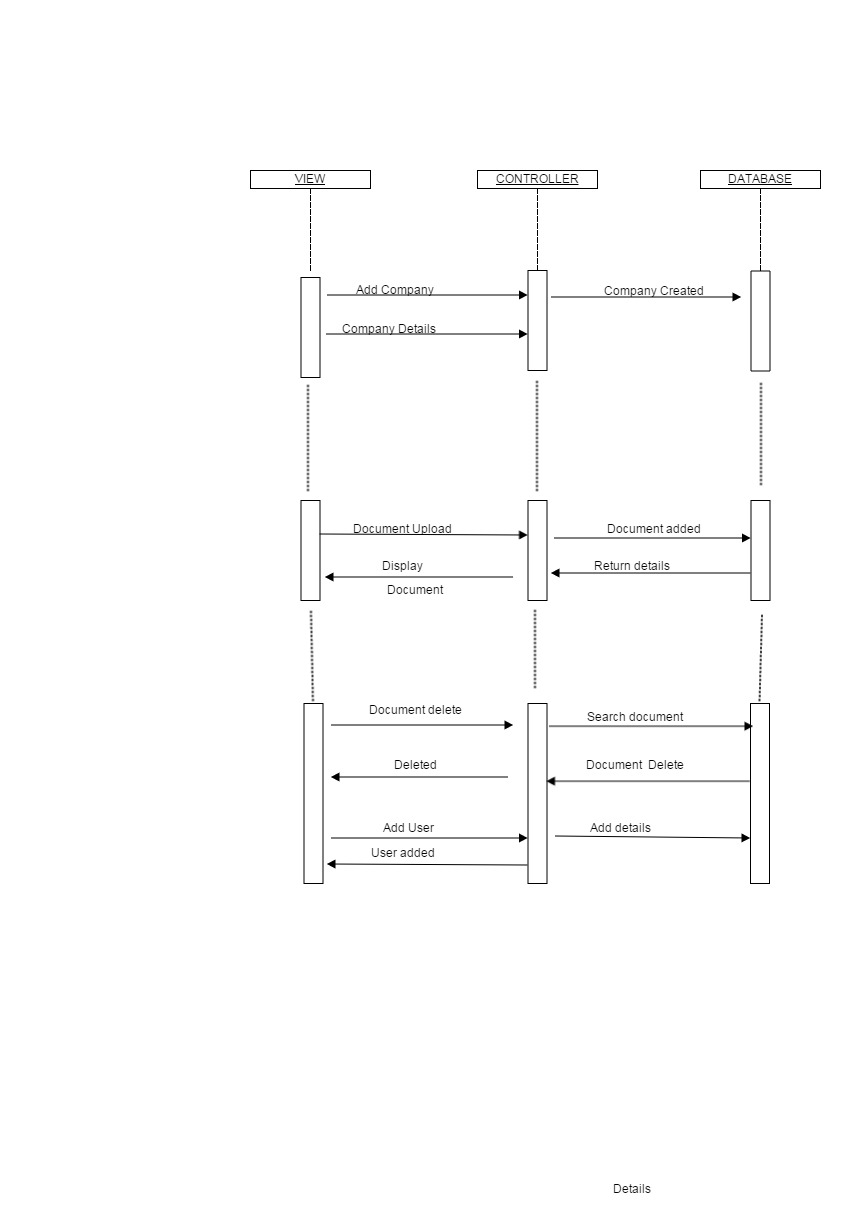
****

****

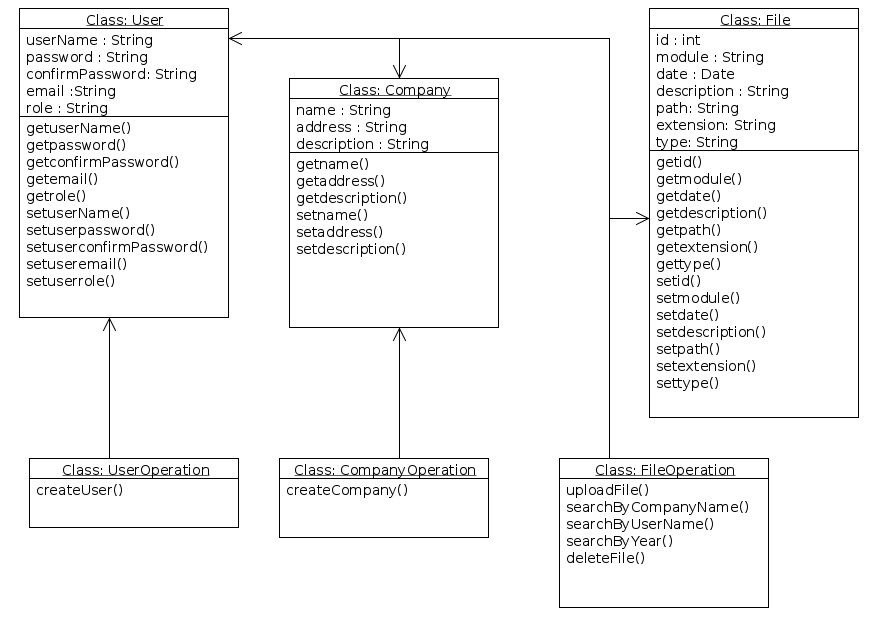
****

**3.5 Sequence Diagram**



****

**3.6 Class Diagram :**

****

**4 SYSTEM DESIGN**

4.1 Basic Modules :

* User Login
* FIle Search
* File Delete
* Create User
* Download File
* Upload File

**User Login** : This is the task which we have to achieve for authenticate the user.

**FIle Search** : in the project we need the searching mechanism to search efficiently.

**File Delete** : we need to delete the unwanted file which was uploaded by mistake .

**Create User** : This task should be achieve to create different kinds of users so that user can enter in application by their different kind of roles.

**Download File** : For downloading a particular file

**Upload File** : for uploading the file on server with unique number.

4.2 Schema Design

USE [master]

GO

CREATE DATABASE [CA\_DATA]

GO

USE [CA\_DATA]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE TABLE [dbo].[Users](

[UserId] [int] IDENTITY(1,1) NOT NULL,

[Username] [nvarchar](20) NOT NULL,

[Password] [nvarchar](20) NOT NULL,

[Roles] [nvarchar] (10) NOT NULL,

[Email] [nvarchar](30) NOT NULL,

[CreatedDate] [datetime] NOT NULL,

[LastLoginDate] [datetime] NULL,

CONSTRAINT [PK\_Users] PRIMARY KEY CLUSTERED

(

[UserId] 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

USE [CA\_DATA]

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

CREATE PROCEDURE [dbo].[Insert\_User]

@Username NVARCHAR(20),

@Password NVARCHAR(20),

@Roles nvarchar(10),

@Email NVARCHAR(30)

AS

BEGIN

SET NOCOUNT ON;

IF EXISTS(SELECT UserId FROM Users WHERE Username = @Username)

BEGIN

SELECT -1 -- Username exists.

END

ELSE IF EXISTS(SELECT UserId FROM Users WHERE Email = @Email)

BEGIN

SELECT -2 -- Email exists.

END

ELSE

BEGIN

INSERT INTO [Users]

([Username]

,[Password]

,[Roles]

,[Email]

,[CreatedDate])

VALUES

(@Username

,@Password

,@Roles

,@Email

,GETDATE())

SELECT SCOPE\_IDENTITY() -- UserId

END

END

GO

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

GO

CREATE PROCEDURE [dbo].[Validate\_User]

@Username NVARCHAR(20),

@Password NVARCHAR(20)

AS

BEGIN

SET NOCOUNT ON;

DECLARE @UserId INT, @LastLoginDate DATETIME

SELECT @UserId = UserId, @LastLoginDate = LastLoginDate

FROM Users WHERE Username = @Username AND [Password] = @Password

IF @UserId IS NOT NULL

BEGIN

SELECT Roles from Users where UserId=@UserId

END

ELSE

BEGIN

SELECT '-1' -- User invalid.

END

END

INSERT INTO Users values('Mudassar', '12345','Admin', 'mudassar@aspsnippets.com', GETDATE(), NULL)

go

create table Company

(

Company\_id int IDENTITY(1,1) constraint PK\_Company primary key,

Name nvarchar(100) not null,

Addres nvarchar(500) not null,

Com\_Description nvarchar(500) not null

)

go

CREATE TABLE file\_data(

FileId int IDENTITY(1,1) constraint PK\_File primary key,

UserId int foreign key references Users(UserId) not null,

Username nvarchar(20) NOT NULL,

Company\_id int foreign key references Company(Company\_id) not null,

Company\_Name nvarchar(100)Not null,

Module nvarchar (5)not null constraint chk check (Module in('A','B','C','D','E')),

CreatedDate date NOT NULL,

file\_year nvarchar(4)not null

)

go

create table file\_uploaded(

FileId int foreign key references file\_data(FileId) not null,

File\_Description nvarchar(100),

File\_Type nvarchar(30) Not null,

File\_Path nvarchar(100) not null,

F\_Name nvarchar(20) not null

)

go

CREATE PROCEDURE [dbo].[Insert\_File\_Data]

@UserId int,

@Username NVARCHAR(20),

@Company\_id int,

@Company\_Name nvarchar(100),

@Module nvarchar (5),

@CreatedDate date,

@file\_year nvarchar(4)

AS

BEGIN

set nocount on;

INSERT INTO [file\_data]

([UserId]

,[Username]

,[Company\_id]

,[Company\_Name]

,[Module]

,[CreatedDate]

,[file\_year])

VALUES

(@UserId

,@Username

,@Company\_id

,@Company\_Name

,@Module

,@CreatedDate

,@file\_year)

SELECT SCOPE\_IDENTITY()

END

go

CREATE PROCEDURE [dbo].[Insert\_File\_Uploaded]

@FileId int,

@File\_Description nvarchar(100),

@File\_Type nvarchar(30),

@File\_Path nvarchar(100),

@F\_Name nvarchar(20)

AS

BEGIN

INSERT INTO [file\_Uploaded]

([FileId]

,[File\_Description]

,[File\_Type]

,[File\_Path]

,[F\_Name])

VALUES

(@FileId

,@File\_Description

,@File\_Type

,@File\_Path

,@F\_Name)

END

**4.2 SECURITY MEASURES:-**

The security measures imposed in the software are:

(1) A login password is provided in the software. User must login to activate the application.

(2) User cannot change the password. To change password he must contact the administrator.

(3) The user/password are given through SQL SERVER 2008 Server. If this is installed on NT 4.0 then it is highly secured.

(4) Data security, correctness integrity is checked up before saving, update or delete if errors found the procedure is aborted.

(5) A primary key & foreign key concept is implemented for avoiding incorrect data entry or intentional or accidental delete or modification of data.

(6) When user tries to delete the data then this first check for its reference used by other data, if found the deletion aborted.

(7) I am also providing various securities at user level or at forms.

**4.5 Database Security**

Database security concerns the use of a broad range of information security controls to protect databases (potentially including the data, the database applications or stored functions, the database systems, the database servers and the associated network links) against compromises of their confidentiality, integrity and availability. It involves various types or categories of controls, such as technical, procedural/administrative and physical.  Security risks to database systems include, for example:

I have use following type of checks:

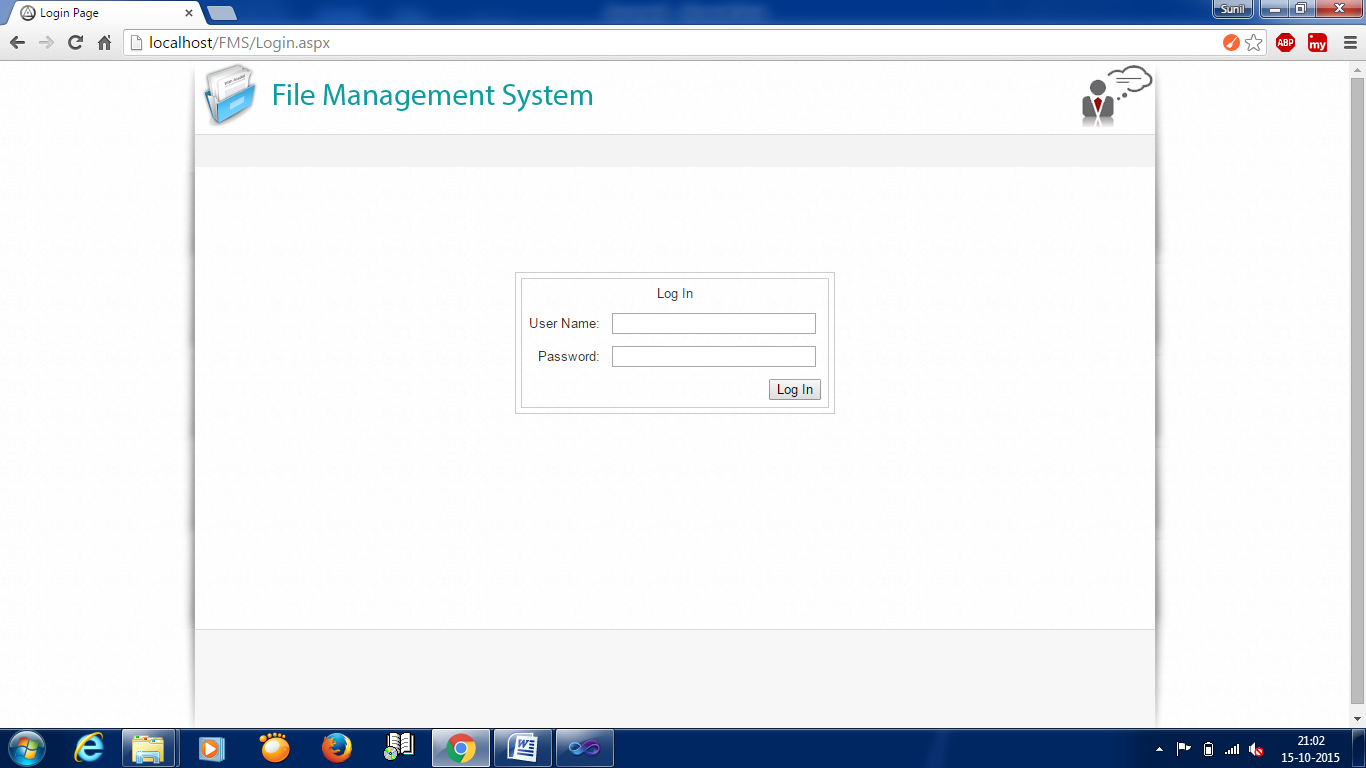
a. Data type

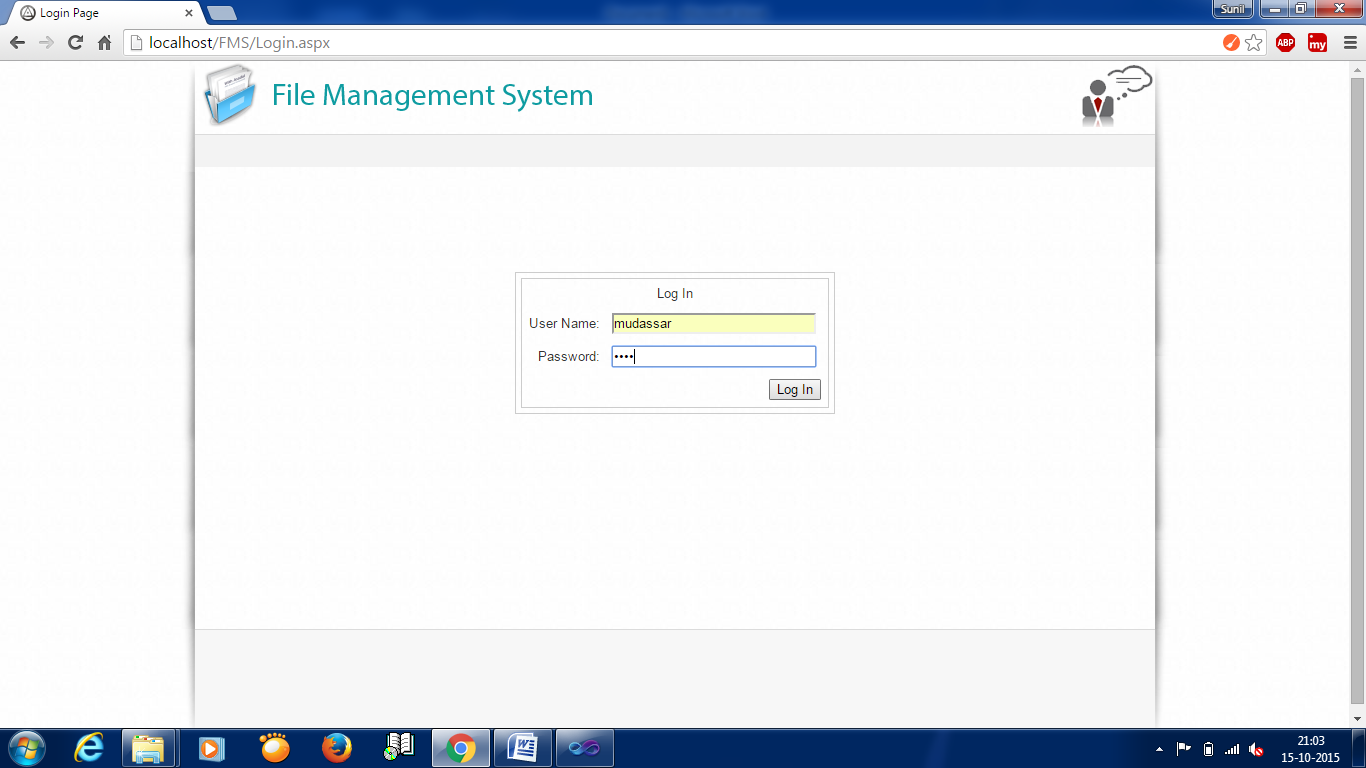
b. Length

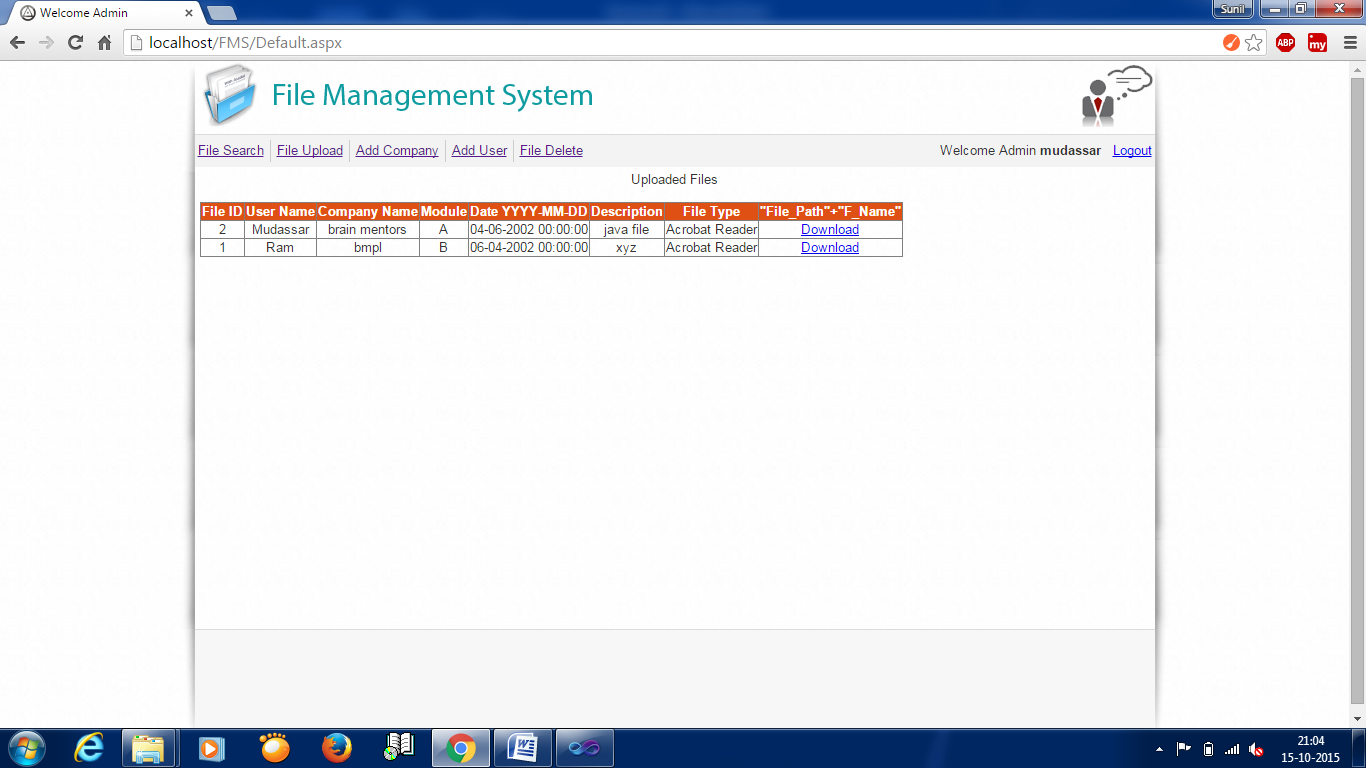
c. Constraints

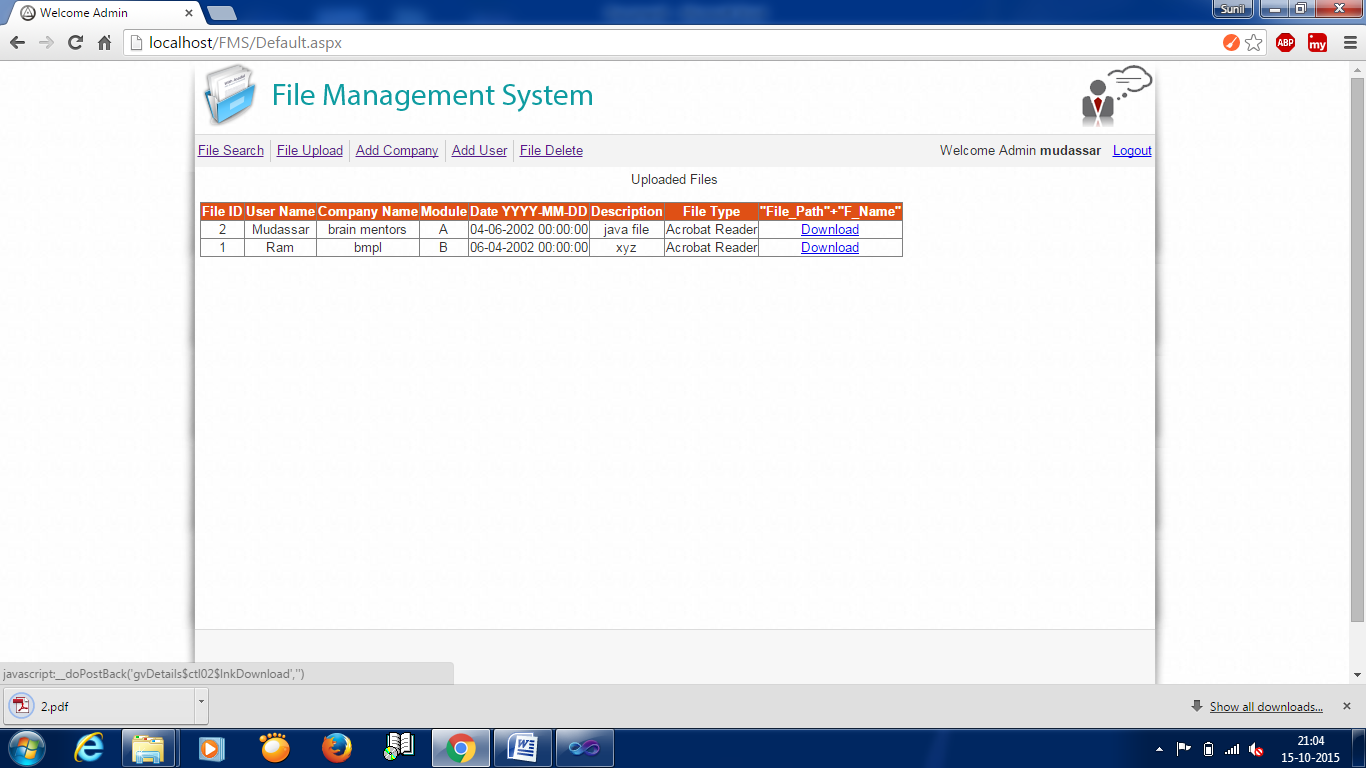
d. Blank field

e. Format

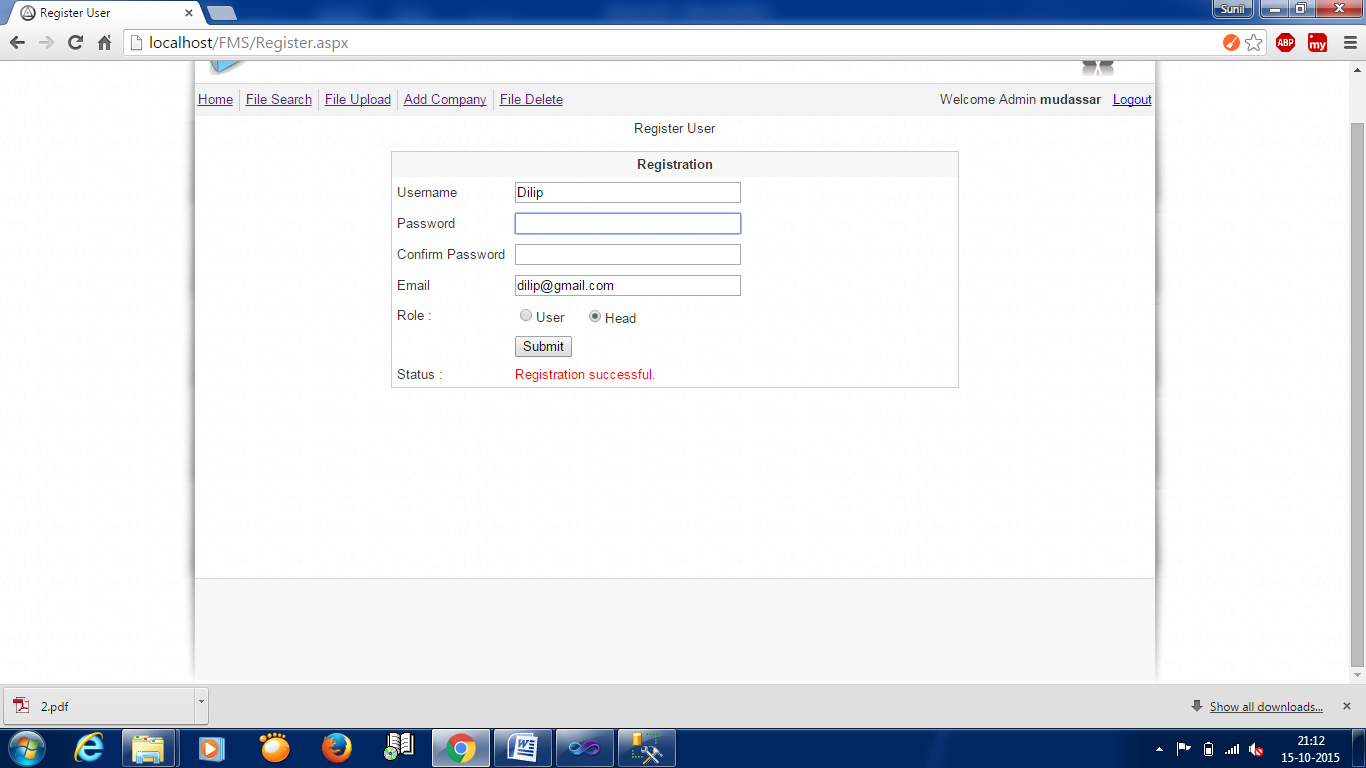
**5.1 User Interface :**

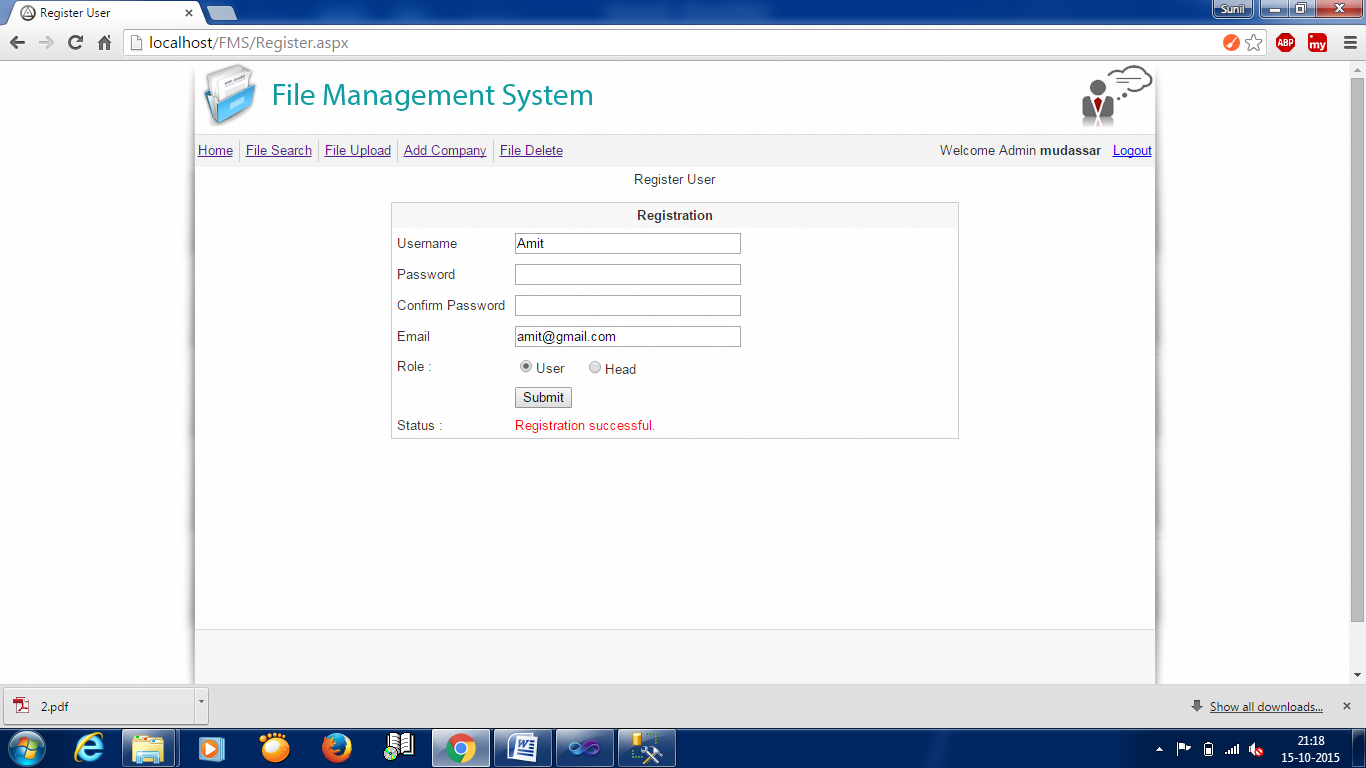


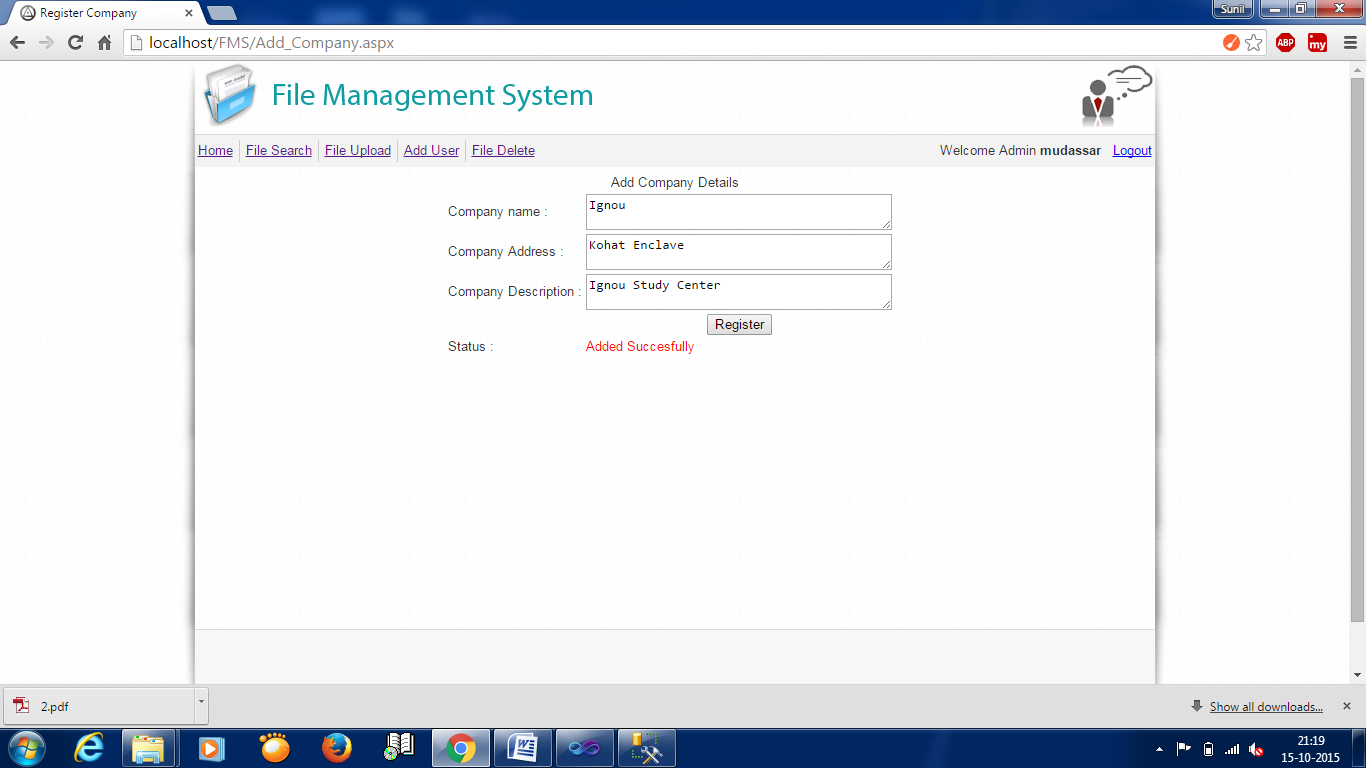


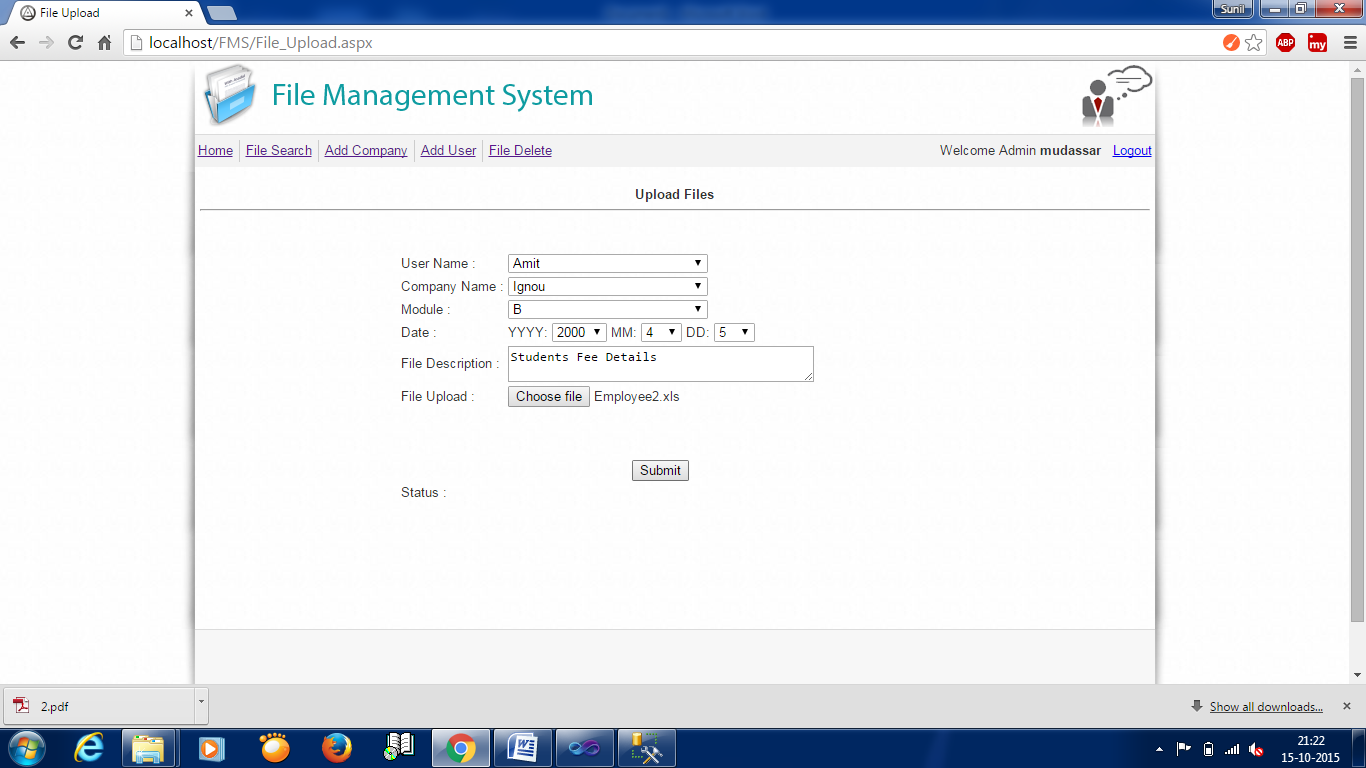


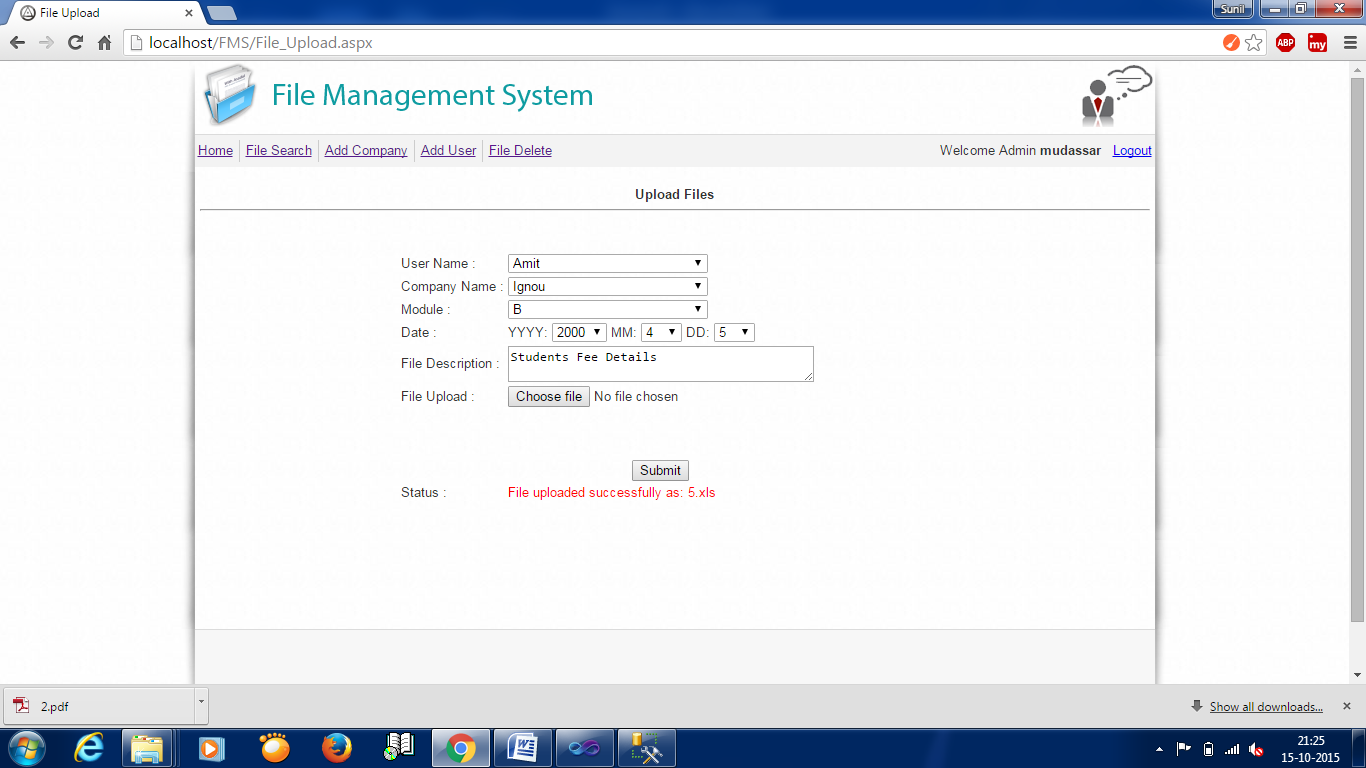


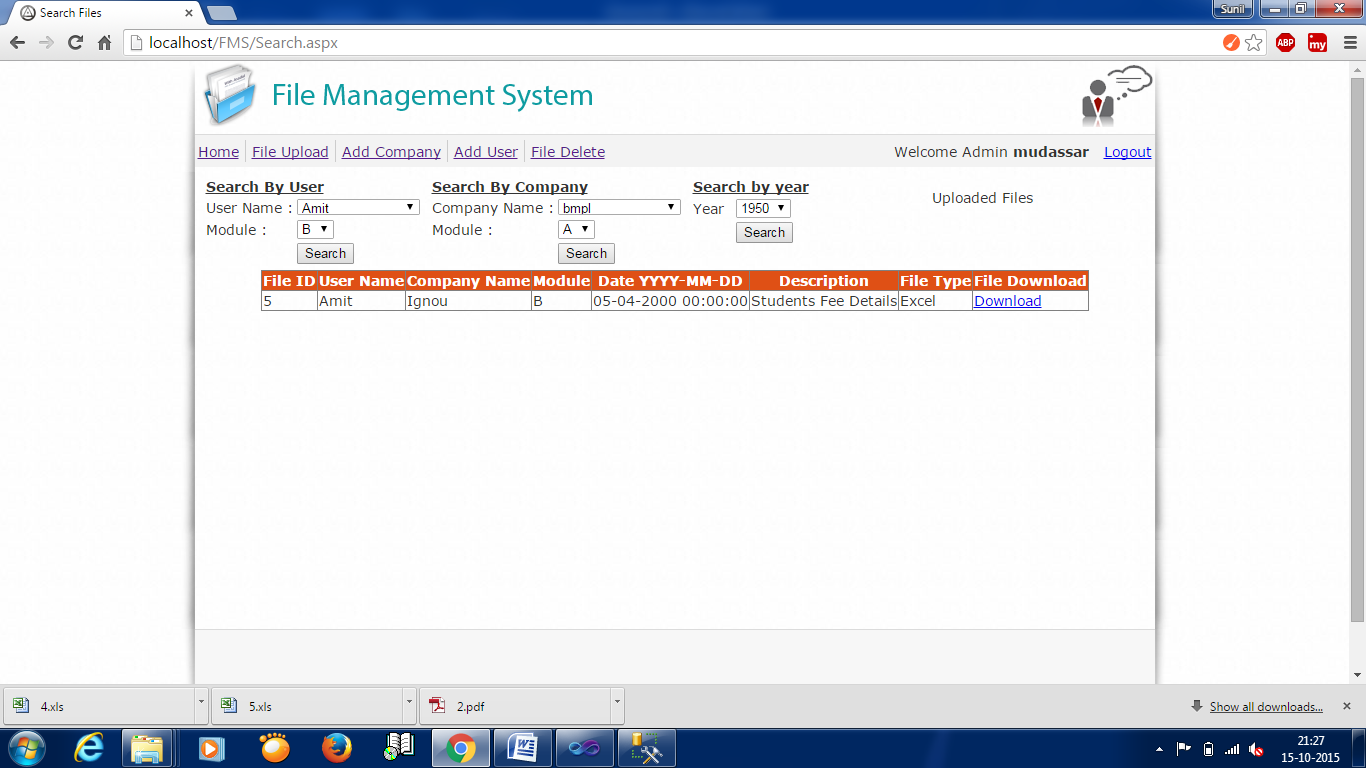


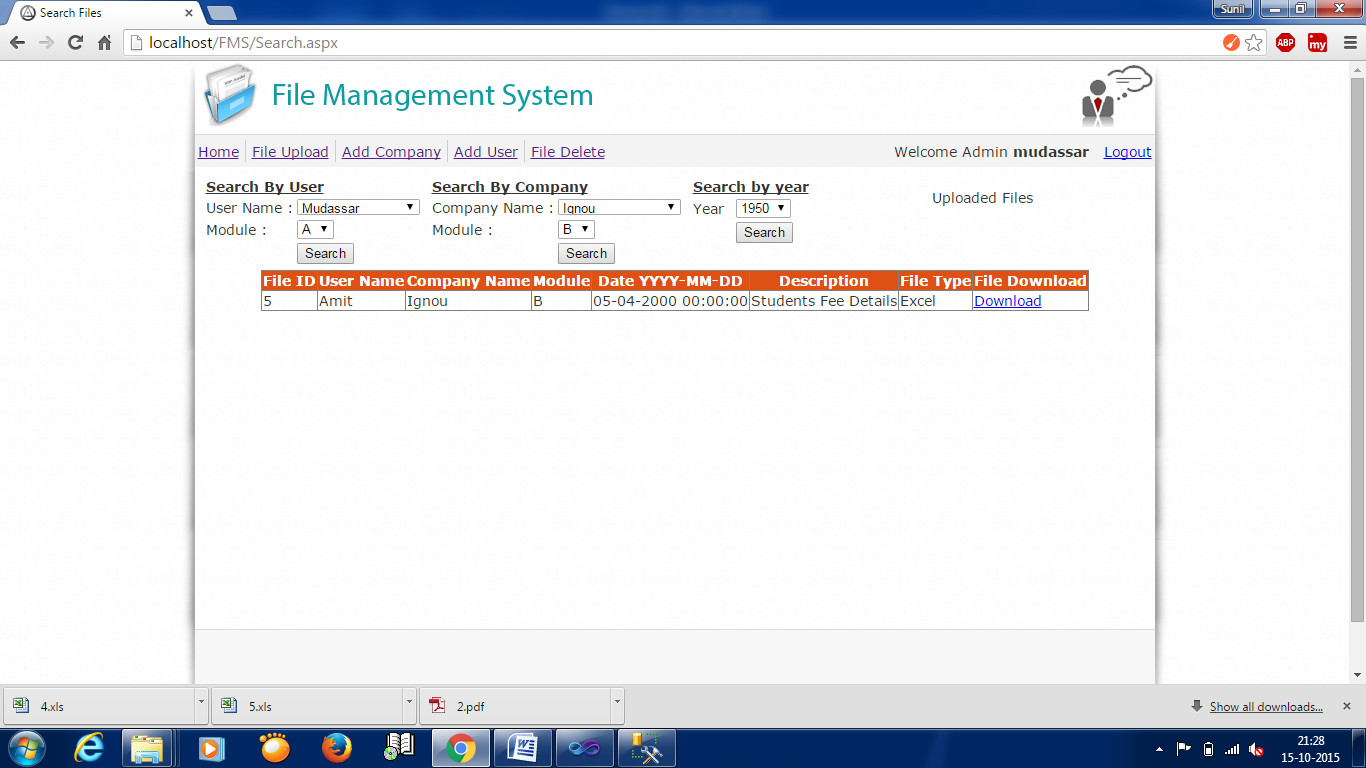


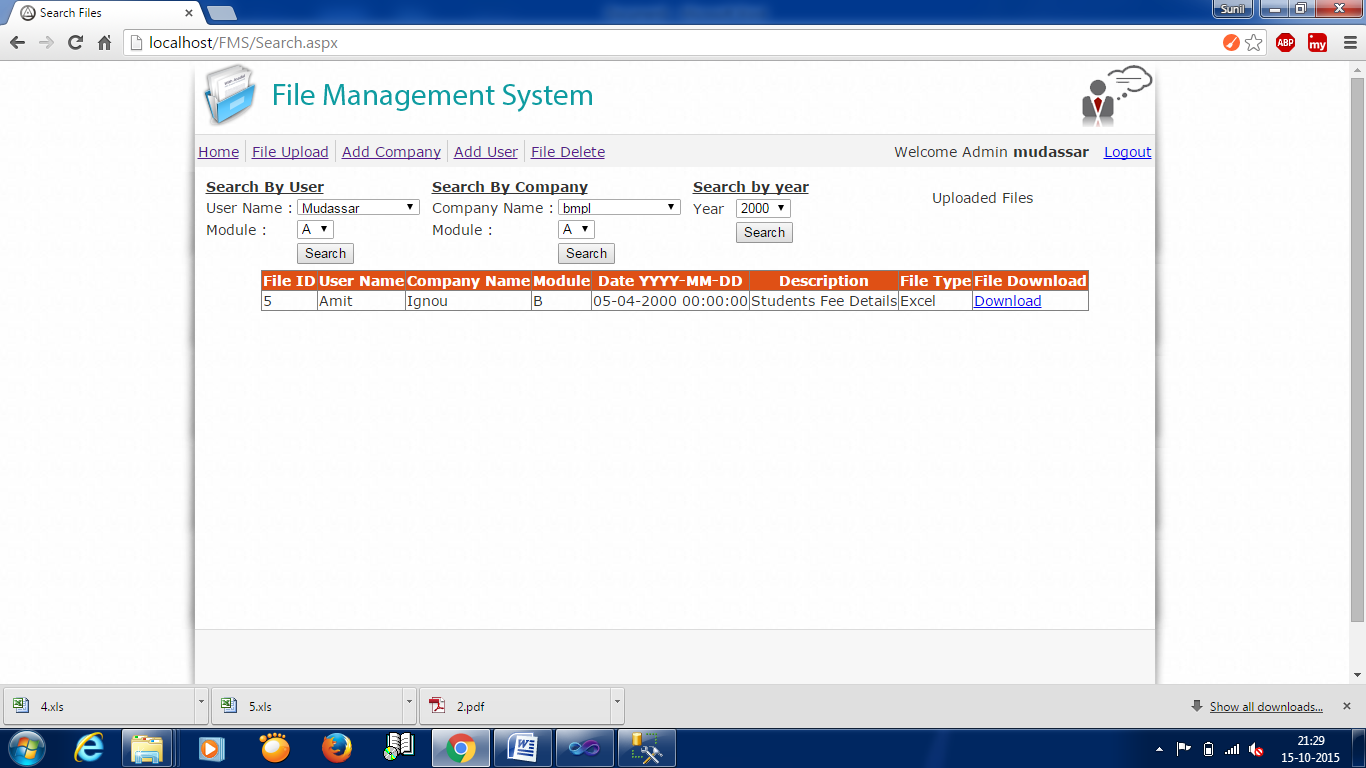


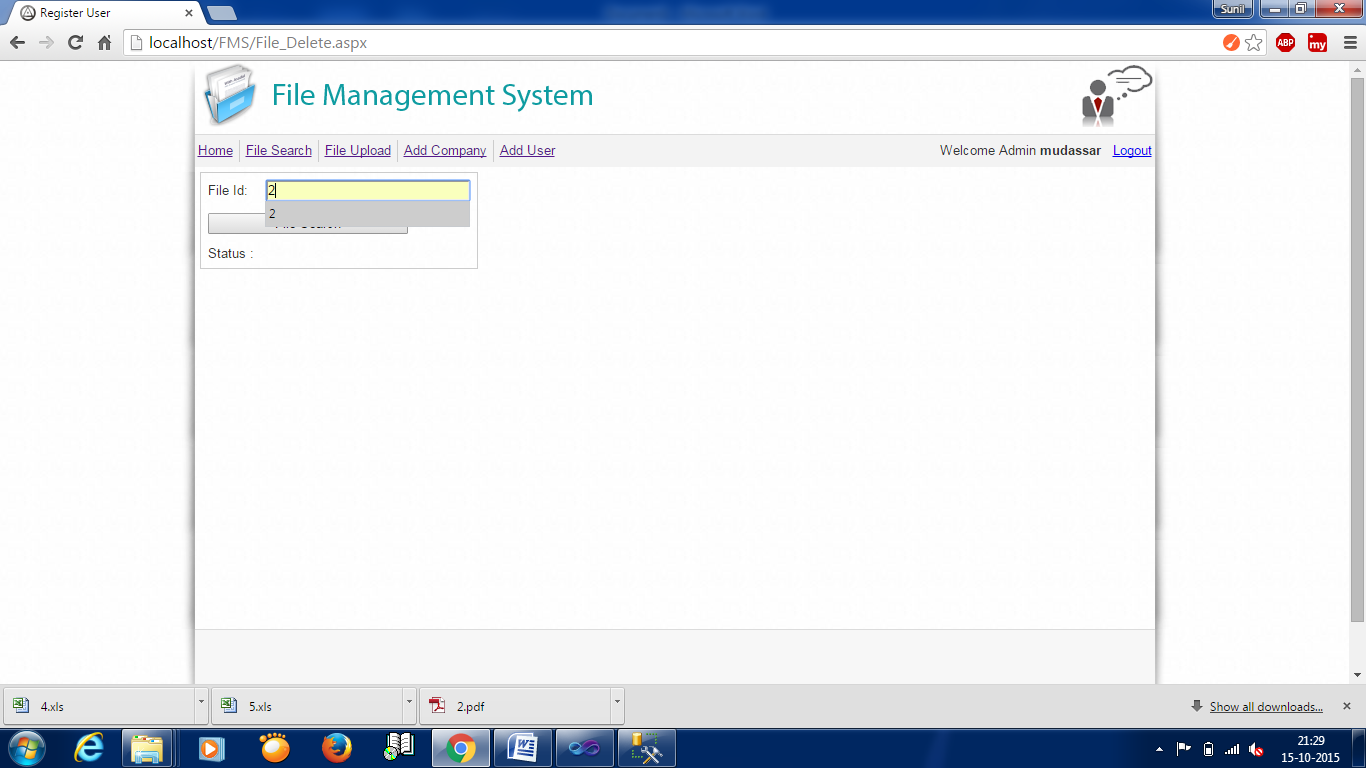


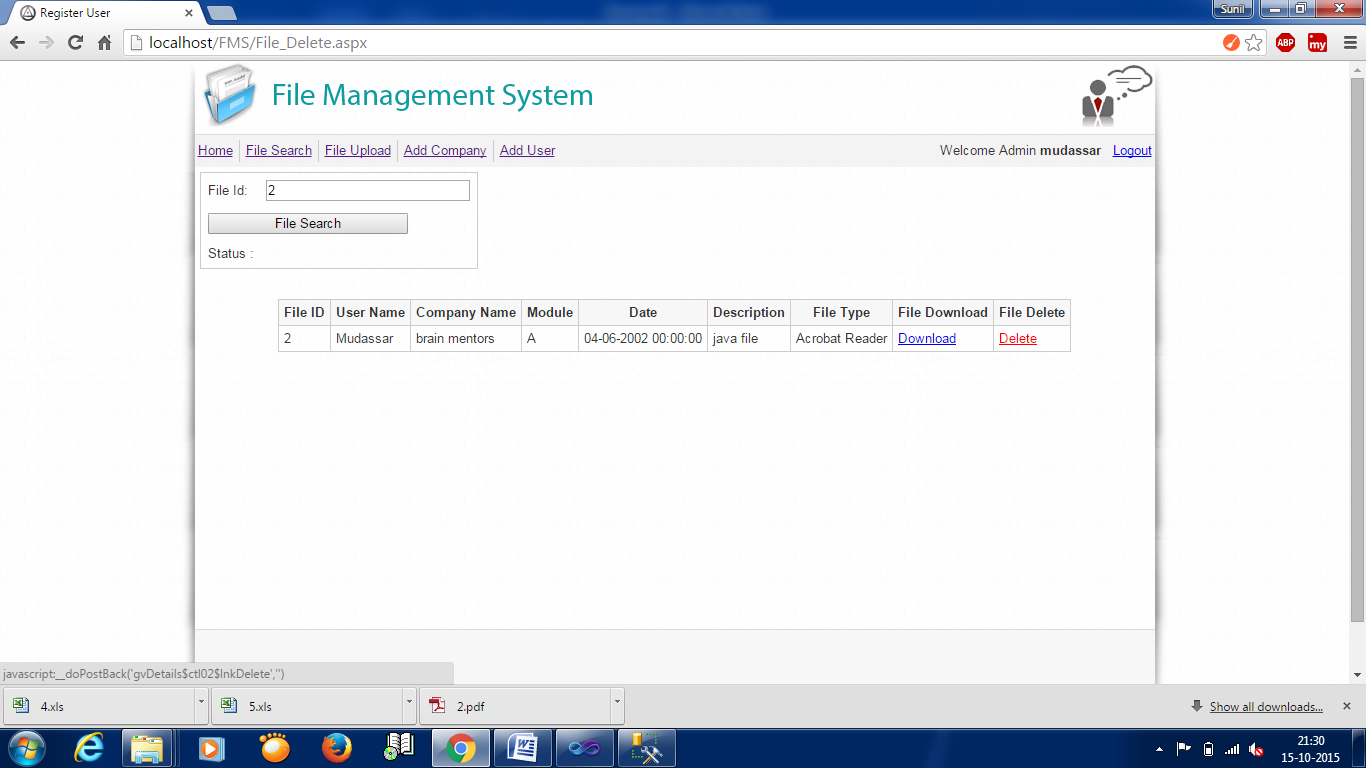


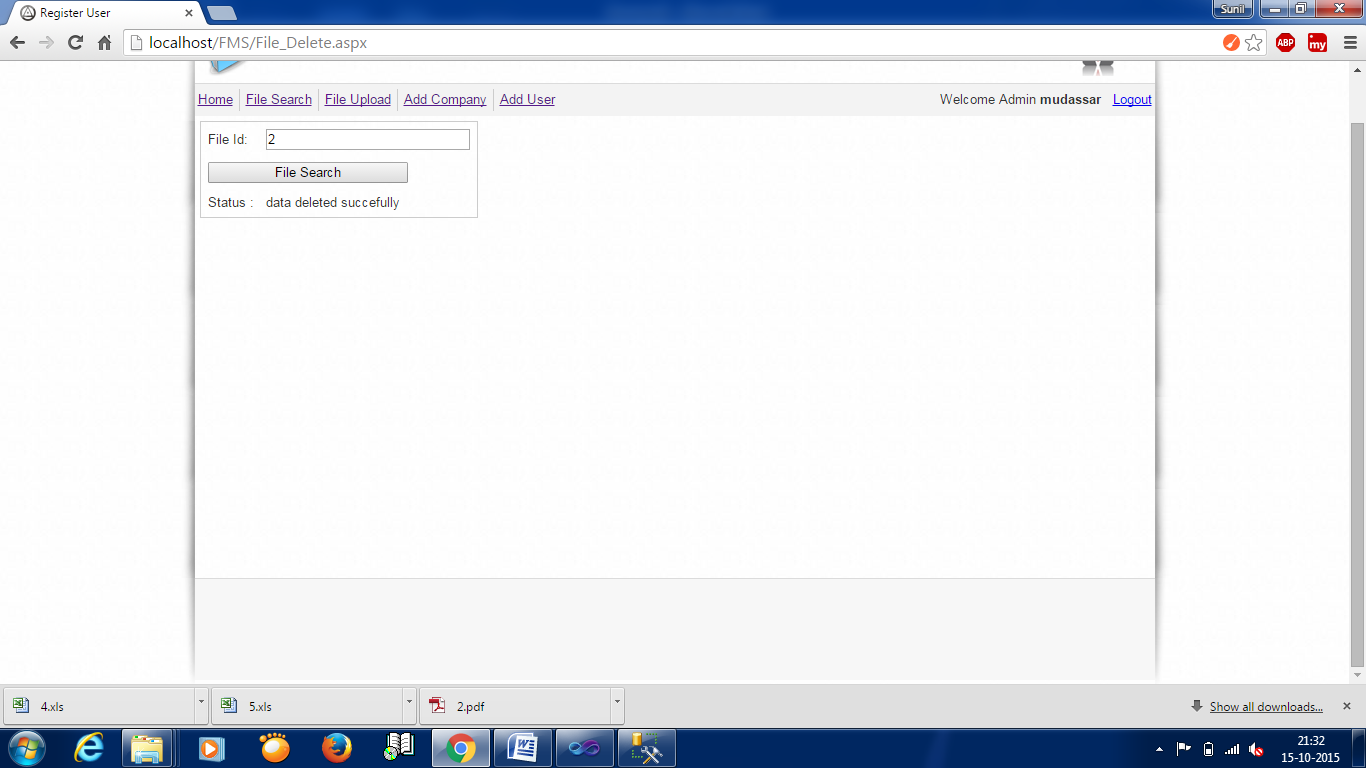


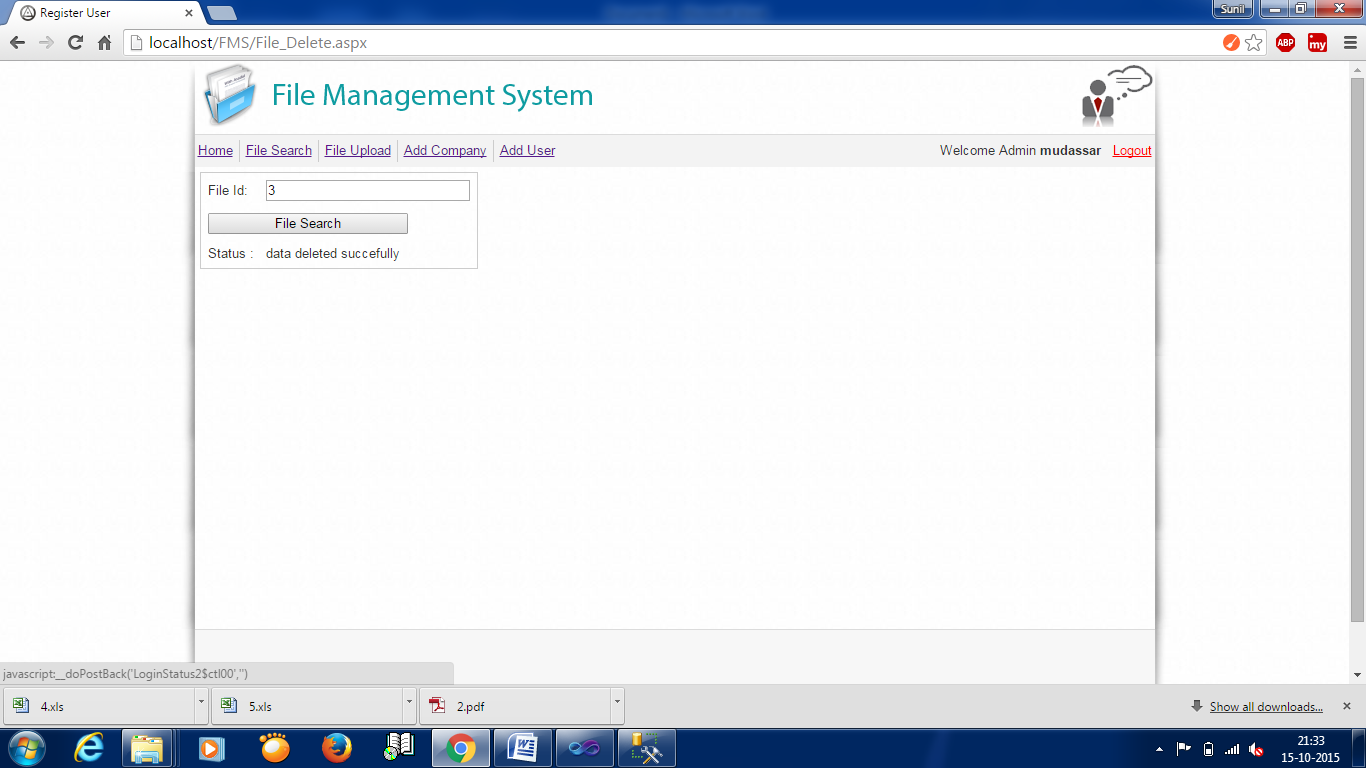


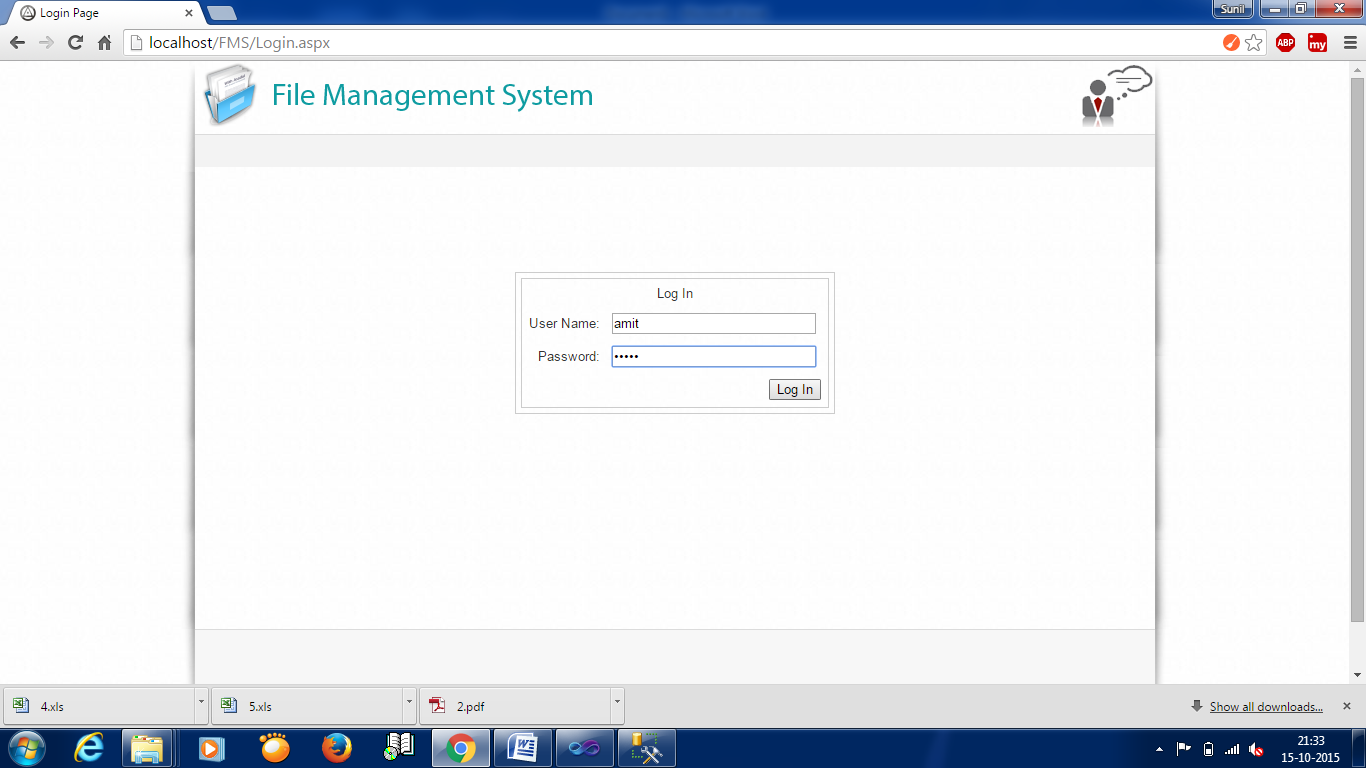
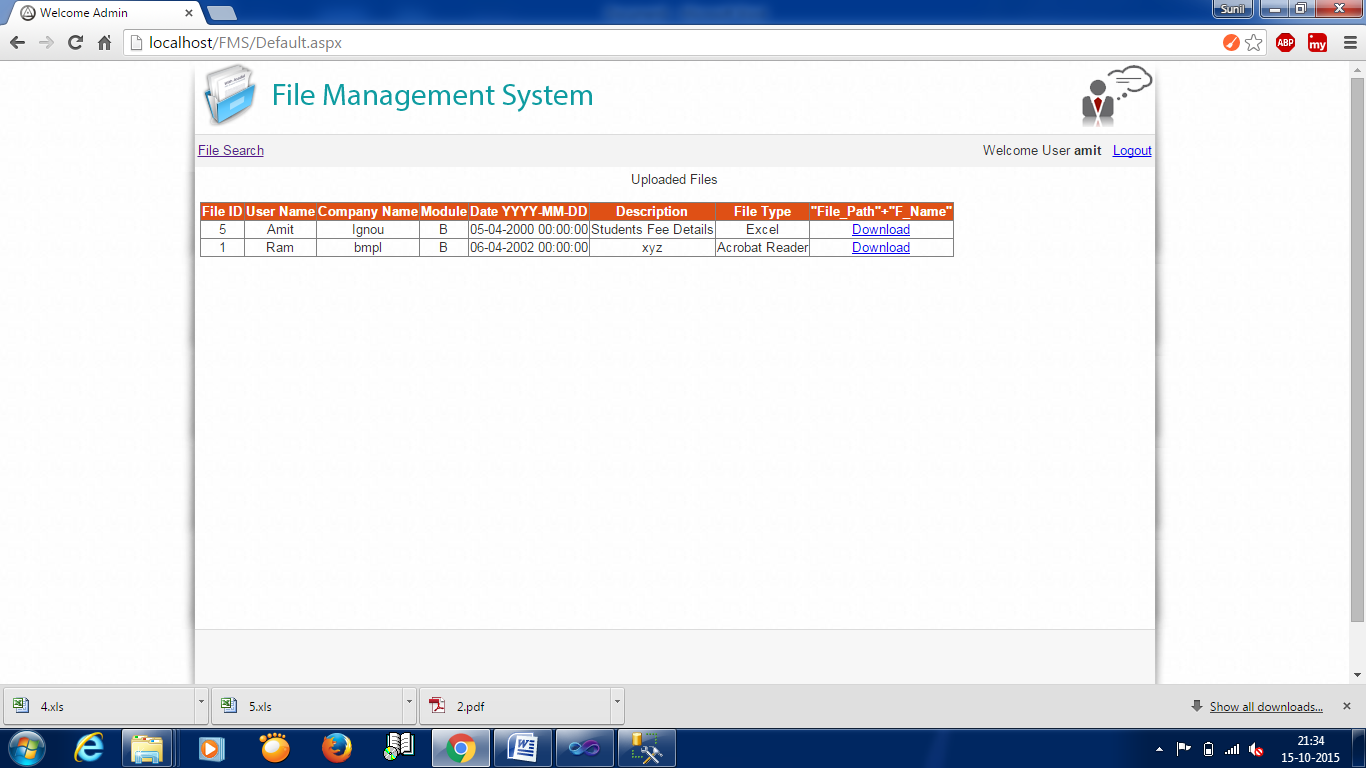


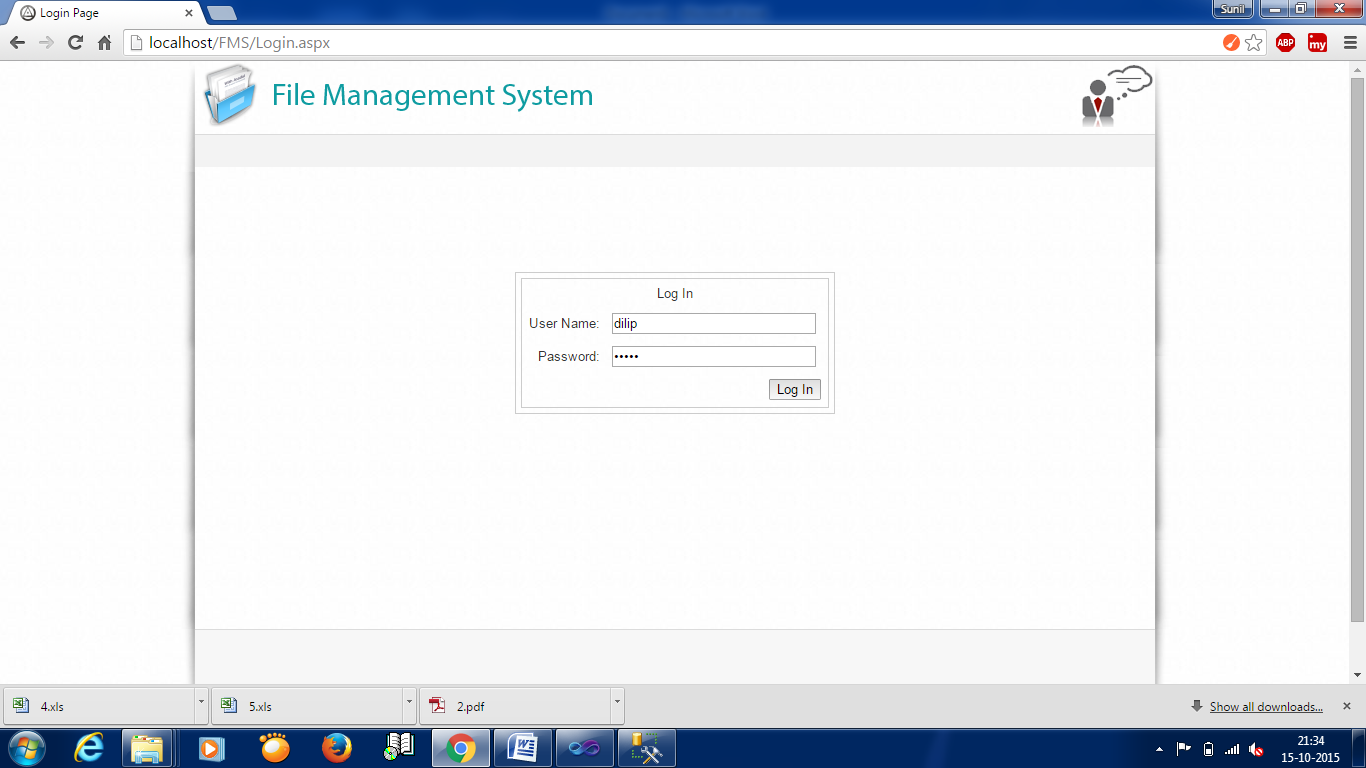


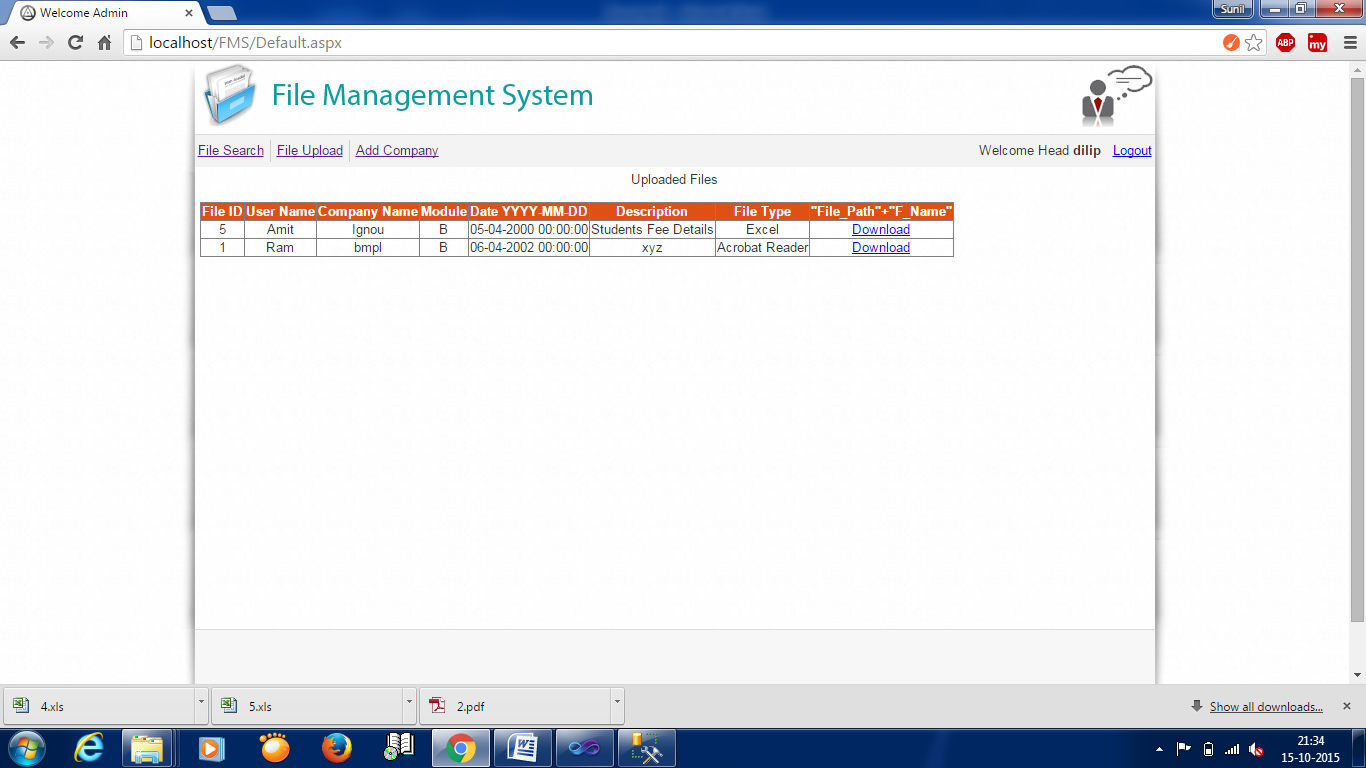












**5.2 CODING DETAILS AND CODE EFFICIENCY**

CODE EFFICIENCY:

Due to modular concept of visual basic this software also inherits this concept in the coding. Most of codes are efficient enough to perform quick, accurate retrieval of data, validation and showing outputs. In this software, most of codes are used which were designed and tested by famous vendors, such as, Microsoft, Crystal Corporation etc. We have used active x technology, which help user and vendor to design software, which provides better, accurate design and reusability code, such as, ADO technology.

CODE OPTIMIZATION:

Most of codes are reused to reduce repeated coding and the result set are reused where needed due to its modular concept, it is possible to reduce coding. The following points represent the code optimization.

1. Use of modules

2. Fixed type variable

3. Short and meaning full name

4. Disconcerted record

5. Connection established once

6. In built function

7. Different scope of variable for different purpose

8. Maximum use of independent procedure

**9. Use of function**

Use of Module

With the help of module we avoid the repetition of code for example: we use connection string in the module and avoid many times of repetition of same code because we use many forms and each form must connect the database.

· Fixed Type Variable

I am using fixed type variable not a variant for example: when we want input date I have used date picker in numeric value, use integer or float, for character use string.

· Short and Meaning Full Name

I have use short and meaning full name for example name for name adds for address etc.

· Disconnect Record

The disconnection of record set close with in its scope because it may produce some error.

· In Built Function

I have used in built function of ASP.net. I.e. input box, msgbox etc.

**5.3 Coding :**

**Default.aspx Page**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="\_Default" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head id="Head1" runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>Welcome Admin</title>

<style type="text/css">

body

{

font-family: Arial;

font-size: 10pt;

}

</style>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server" >

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

<div class="menu\_left"> <asp:HyperLink ID="Search" runat="server" NavigateUrl="~/Search.aspx" Visible="true">File Search</asp:HyperLink></div>

<asp:Label ID="LUpload" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Upload" runat="server" NavigateUrl="~/File\_Upload.aspx" Visible="false">File Upload</asp:HyperLink></div>

<asp:Label ID="LCompany" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Company" runat="server" NavigateUrl="~/Add\_Company.aspx" Visible="false">Add Company</asp:HyperLink></div>

<asp:Label ID="LRegister" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Register" runat="server" NavigateUrl="~/Register.aspx" Visible="false"> Add User</asp:HyperLink></div>

<asp:Label ID="LFDelete" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Delete" runat="server" NavigateUrl="~/File\_Delete.aspx" Visible="false"> File Delete</asp:HyperLink></div>

<div class="menu\_right"><asp:Label ID="Label2" runat="server" Text="Welcome User" Visible="false"></asp:Label><asp:Label ID="Label3" runat="server" Text="Welcome Head" Visible="false"></asp:Label><asp:Label ID="Label4" runat="server" Text="Welcome Admin" Visible="false"></asp:Label> <asp:LoginName ID="LoginName2" runat="server" Font-Bold="true" />&nbsp;&nbsp;&nbsp;<asp:LoginStatus ID="LoginStatus2" runat="server" onloggingout="LoginStatus1\_LoggingOut" /></div>

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<center> Uploaded Files </center>

<br />

<center style="text-align:center">

<asp:GridView ID="gvDetails" runat="server" AutoGenerateColumns="false" DataKeyNames="FileId">

<HeaderStyle BackColor="#df5015" Font-Bold="true" ForeColor="White" />

<Columns>

<asp:BoundField DataField="FileId" HeaderText="File ID" />

<asp:BoundField DataField="Username" HeaderText="User Name" />

<asp:BoundField DataField="Company\_Name" HeaderText="Company Name" />

<asp:BoundField DataField="Module" HeaderText="Module" />

<asp:BoundField DataField="CreatedDate" HeaderText="Date YYYY-MM-DD" />

<asp:BoundField DataField="File\_Description" HeaderText="Description" />

<asp:BoundField DataField="File\_Type" HeaderText="File Type" />

<asp:TemplateField HeaderText="File\_Path"+"F\_Name">

<ItemTemplate><asp:LinkButton ID="lnkDownload" runat="server" Text="Download" OnClick="lnkDownload\_Click">

</asp:LinkButton>

</ItemTemplate>

</asp:TemplateField>

</Columns>

</asp:GridView>

</center>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

</div>

</div>

</div>

</form>

</body>

</html>

**Default.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Web.Security;

using System.Collections;

using System.Data.SqlClient;

using System.Configuration;

using System.IO;

public partial class \_Default : System.Web.UI.Page

{

protected void Page\_Init(object sender, EventArgs e)

{

if (!this.Page.User.Identity.IsAuthenticated)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else if (Session["role"] == null)

{

FormsAuthentication.SignOut();

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() == "User")

{

Label2.Visible = true;

}

else if (Session["role"].ToString() == "Head")

{

Label3.Visible = true;

LUpload.Visible = true;

Upload.Visible = true;

LCompany.Visible = true;

Company.Visible = true;

}

else if (Session["role"].ToString() == "Admin")

{

Label4.Visible = true;

LUpload.Visible = true;

Upload.Visible = true;

LCompany.Visible = true;

Company.Visible = true;

LRegister.Visible = true;

Register.Visible = true;

LFDelete.Visible = true;

Delete.Visible = true;

}

}

}

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

BindGridviewData();

}

}

protected void LoginStatus1\_LoggingOut(object sender, LoginCancelEventArgs e)

{

FormsAuthentication.SignOut();

Session["role"] = null;

Response.Redirect("Login.aspx");

}

private void BindGridviewData()

{

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try

{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select top 5 fu.FileId, fd.Username, fd.Company\_Name, fd.Module, fd.CreatedDate, fu.File\_Description, fu.File\_Type, fu.File\_Path ,fu.F\_Name from file\_data fd join file\_uploaded fu on fd.FileId=fu.FileId order by fd.FileId desc";

cmd.Connection = con;

con.Open();

gvDetails.DataSource = cmd.ExecuteReader();

gvDetails.DataBind();

con.Close();

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

protected void lnkDownload\_Click(object sender, EventArgs e)

{

try{

LinkButton lnkbtn = sender as LinkButton;

GridViewRow gvrow = lnkbtn.NamingContainer as GridViewRow;

int fileid = Convert.ToInt32(gvDetails.DataKeys[gvrow.RowIndex].Value.ToString());

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select File\_Path, F\_Name from file\_uploaded where FileId=@id";

cmd.Parameters.AddWithValue("@id", fileid);

cmd.Connection = con;

con.Open();

SqlDataReader dr = cmd.ExecuteReader();

if (dr.Read())

{

FileInfo fileInfo = new FileInfo(dr[0].ToString() + dr[1].ToString());

Response.Clear();

Response.AddHeader("Content-Disposition", "attachment;filename=" + fileInfo.Name);

Response.AddHeader("Content-Length", fileInfo.Length.ToString());

Response.ContentType = "application/octet-stream";

Response.Flush();

Response.WriteFile(fileInfo.FullName);

Response.End();

}

con.Close();

dr.Close();

}

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

}

}

**Login.aspx Page**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Login.aspx.cs" Inherits="Login" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head id="Head1" runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>Login Page</title>

<style type="text/css">

body

{

font-family: Arial;

font-size: 10pt;

}

input[type=text], input[type=password]

{

width: 200px;

}

table

{

border: 1px solid #ccc;

}

table th

{

background-color: #F7F7F7;

color: #333;

font-weight: bold;

}

table th, table td

{

padding: 5px;

border-color: #ccc;

}

</style>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server">

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<div style="margin-top:100px;" align="center">

<asp:Login ID="Login1" runat="server" OnAuthenticate="ValidateUser"

DisplayRememberMe="False">

</asp:Login>

</div>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

<div style="width:159px; float:right; margin-top:22px"> </div> </div>

</div>

</div>

</form>

</body>

</html>

**Login.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Configuration;

using System.Data.SqlClient;

using System.Web.Security;

public partial class Login : System.Web.UI.Page

{

protected void ValidateUser(object sender, EventArgs e)

{

//int userId = 0;

string userId;

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand("Validate\_User"))

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@Username", Login1.UserName);

cmd.Parameters.AddWithValue("@Password", Login1.Password);

cmd.Connection = con;

con.Open();

userId = Convert.ToString(cmd.ExecuteScalar());

con.Close();

}

switch (userId)

{

case "-1":

Login1.FailureText = "Username and/or password is incorrect.";

break;

case "-2":

Login1.FailureText = "Account has not been activated.";

break;

default:

Session["role"] = userId;

FormsAuthentication.RedirectFromLoginPage(Login1.UserName, Login1.RememberMeSet);

break;

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

}

**Add\_Company.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Add\_Company.aspx.cs" Inherits="Add\_Company" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>Register Company</title>

<style type="text/css">

body

{

font-family: Arial;

font-size: 10pt;

}

</style>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server">

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

<div class="menu\_left"> <asp:HyperLink ID="Home" runat="server" NavigateUrl="~/Default.aspx" Visible="true">Home</asp:HyperLink></div>

<asp:Label ID="LSearch" runat="server" CssClass="line" Visible="true"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Search" runat="server" NavigateUrl="~/Search.aspx" Visible="true">File Search</asp:HyperLink></div>

<asp:Label ID="LUpload" runat="server" CssClass="line" Visible="true"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Upload" runat="server" NavigateUrl="~/File\_Upload.aspx" Visible="true">File Upload</asp:HyperLink></div>

<asp:Label ID="LRegister" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Register" runat="server" NavigateUrl="~/Register.aspx" Visible="false"> Add User</asp:HyperLink></div>

<asp:Label ID="LFDelete" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Delete" runat="server" NavigateUrl="~/File\_Delete.aspx" Visible="false"> File Delete</asp:HyperLink></div>

<div class="menu\_right"><asp:Label ID="Label3" runat="server" Text="Welcome Head" Visible="false"></asp:Label><asp:Label ID="Label4" runat="server" Text="Welcome Admin" Visible="false"></asp:Label> <asp:LoginName ID="LoginName2" runat="server" Font-Bold="true" />&nbsp;&nbsp;&nbsp;<asp:LoginStatus ID="LoginStatus2" runat="server" onloggingout="LoginStatus1\_LoggingOut" /></div>

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<center>

<table>

<tr>

<td colspan="3" align="center">

Add Company Details

</td>

</tr>

<tr>

<td>

Company name :

</td>

<td>

<asp:TextBox ID="TextBox1" runat="server" TextMode="MultiLine" Width="300px"></asp:TextBox>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"

ErrorMessage="\*" ForeColor="Red" ControlToValidate="TextBox1"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>

Company Address :

</td>

<td>

<asp:TextBox ID="TextBox2" runat="server" TextMode="MultiLine" Width="300px"></asp:TextBox>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"

ErrorMessage="\*" ForeColor="Red" ControlToValidate="TextBox2"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>

Company Description :

</td>

<td>

<asp:TextBox ID="TextBox3" runat="server" TextMode="MultiLine" Width="300px"></asp:TextBox>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"

ErrorMessage="\*" ForeColor="Red" ControlToValidate="TextBox3"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td></td>

<td align="center">

<asp:Button ID="Button1" runat="server" Text="Register"

onclick="Button1\_Click" /></td>

<td></td>

</tr>

<tr>

<td> Status :</td>

<td colspan="2">

<asp:Label ID="Label1" runat="server" Text="Label" ForeColor="Red" Visible="false"></asp:Label></td>

</tr>

</table>

</center>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

<div style="width:159px; float:right; margin-top:22px"> </div> </div>

</div>

</div>

</form>

</body>

</html>

**Add\_Company.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Web.Security;

using System.Collections;

using System.Data.SqlClient;

using System.Configuration;

public partial class Add\_Company : System.Web.UI.Page

{

static String dbstring = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

static SqlConnection con = new SqlConnection(dbstring);

protected void Page\_Init(object sender, EventArgs e)

{

if (!this.Page.User.Identity.IsAuthenticated)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else if (Session["role"] == null)

{

FormsAuthentication.SignOut();

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() != "Admin" && Session["role"].ToString() != "Head")

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else if (Session["role"].ToString() == "Admin")

{

LRegister.Visible = true;

Register.Visible = true;

Label4.Visible = true;

LFDelete.Visible = true;

Delete.Visible = true;

}

else if (Session["role"].ToString() == "Head")

{

Label3.Visible = true;

}

}

}

protected void LoginStatus1\_LoggingOut(object sender, LoginCancelEventArgs e)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Response.Redirect("Login.aspx");

}

protected void Button1\_Click(object sender, EventArgs e)

{

try

{

con.Open();

SqlCommand cmd = new SqlCommand("insert Company values(@name,@address,@description)", con);

SqlParameter sql1 = cmd.Parameters.Add("@name", SqlDbType.VarChar, 80);

sql1.Value = TextBox1.Text;

SqlParameter sql2 = cmd.Parameters.Add("@address", SqlDbType.VarChar, 80);

sql2.Value = TextBox2.Text;

SqlParameter sql3 = cmd.Parameters.Add("@description", SqlDbType.VarChar, 80);

sql3.Value = TextBox3.Text;

int result = cmd.ExecuteNonQuery();

if (result > 0)

{

Label1.Text = "Added Succesfully ";

Label1.Visible = true;

}

con.Close();

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

**File\_Delete.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="File\_Delete.aspx.cs" Inherits="File\_Delete" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head id="Head1" runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>Register User</title>

<style type="text/css">

body

{

font-family: Arial;

font-size: 10pt;

}

input

{

width: 200px;

}

table

{

border: 1px solid #ccc;

}

table th

{

background-color: #F7F7F7;

color: #333;

font-weight: bold;

}

table th, table td

{

padding: 5px;

border-color: #ccc;

}

</style>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server">

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

<div class="menu\_left"> <asp:HyperLink ID="Home" runat="server" NavigateUrl="~/Default.aspx" Visible="true">Home</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="HyperLink1" runat="server" NavigateUrl="~/Search.aspx">File Search</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="HyperLink2" runat="server" NavigateUrl="~/File\_Upload.aspx">File Upload</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="HyperLink3" runat="server" NavigateUrl="~/Add\_Company.aspx">Add Company</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="Register" runat="server" NavigateUrl="~/Register.aspx"> Add User</asp:HyperLink></div>

<div class="menu\_right">Welcome Admin <asp:LoginName ID="LoginName2" runat="server" Font-Bold="true" />&nbsp;&nbsp;&nbsp;<asp:LoginStatus ID="LoginStatus2" runat="server" onloggingout="LoginStatus1\_LoggingOut" /></div>

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<table>

<tr>

<td>

File Id:

</td>

<td>

<asp:TextBox ID="TextBox1" runat="server"></asp:TextBox>

</td>

</tr>

<tr>

<td colspan="2">

<asp:Button ID="Button1" runat="server" Text="File Search" onclick="Button1\_Click" />

</td>

</tr>

<tr>

<td>

Status :

</td>

<td>

<asp:Label ID="Label1" runat="server" Text="Label" Visible="false"></asp:Label>

</td>

</tr>

</table>

<br />

<br />

<center>

<asp:GridView ID="gvDetails" runat="server" AutoGenerateColumns="false" DataKeyNames="FileId">

<HeaderStyle BackColor="#df5015" Font-Bold="true" ForeColor="White" />

<Columns>

<asp:BoundField DataField="FileId" HeaderText="File ID" />

<asp:BoundField DataField="Username" HeaderText="User Name" />

<asp:BoundField DataField="Company\_Name" HeaderText="Company Name" />

<asp:BoundField DataField="Module" HeaderText="Module" />

<asp:BoundField DataField="CreatedDate" HeaderText="Date" />

<asp:BoundField DataField="File\_Description" HeaderText="Description" />

<asp:BoundField DataField="File\_Type" HeaderText="File Type" />

<asp:TemplateField HeaderText="File Download">

<ItemTemplate><asp:LinkButton ID="lnkDownload" runat="server" Text="Download" OnClick="lnkDownload\_Click">

</asp:LinkButton>

</ItemTemplate>

</asp:TemplateField>

<asp:TemplateField HeaderText="File Delete">

<ItemTemplate><asp:LinkButton ID="lnkDelete" runat="server" Text="Delete" OnClick="lnkDelete\_Click">

</asp:LinkButton>

</ItemTemplate>

</asp:TemplateField>

</Columns>

</asp:GridView>

</center>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

<div style="width:159px; float:right; margin-top:22px"> </div> </div>

</div>

</div>

</form>

</body>

</html>

**File\_Delete.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Configuration;

using System.Data.SqlClient;

using System.Web.Security;

using System.IO;

public partial class File\_Delete : System.Web.UI.Page

{

protected void Page\_Init(object sender, EventArgs e)

{

if (IsPostBack)

{

}

if (!this.Page.User.Identity.IsAuthenticated)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() != "Admin")

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

}

}

protected void LoginStatus1\_LoggingOut(object sender, LoginCancelEventArgs e)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Response.Redirect("Login.aspx");

}

protected void lnkDownload\_Click(object sender, EventArgs e)

{

try{

LinkButton lnkbtn = sender as LinkButton;

GridViewRow gvrow = lnkbtn.NamingContainer as GridViewRow;

int fileid = Convert.ToInt32(gvDetails.DataKeys[gvrow.RowIndex].Value.ToString());

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

using (SqlCommand cmd = new SqlCommand())

{

try

{

cmd.CommandText = "select File\_Path, F\_Name from file\_uploaded where FileId=@id";

cmd.Parameters.AddWithValue("@id", fileid);

cmd.Connection = con;

con.Open();

SqlDataReader dr = cmd.ExecuteReader();

if (dr.Read())

{

FileInfo fileInfo = new FileInfo(dr[0].ToString() + dr[1].ToString());

Response.Clear();

Response.AddHeader("Content-Disposition", "attachment;filename=" + fileInfo.Name);

Response.AddHeader("Content-Length", fileInfo.Length.ToString());

Response.ContentType = "application/octet-stream";

Response.Flush();

Response.WriteFile(fileInfo.FullName);

Response.End();

}

con.Close();

dr.Close();

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

}

private void BindGridviewFile()

{

try

{

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select fu.FileId, fd.Username, fd.Company\_Name, fd.Module, fd.CreatedDate, fu.File\_Description, fu.File\_Type, fu.File\_Path ,fu.F\_Name ,fu.F\_Name from file\_data fd join file\_uploaded fu on fd.FileId=fu.FileId where fd.FileId=@ID";

cmd.Parameters.AddWithValue("@ID", TextBox1.Text);

cmd.Connection = con;

con.Open();

gvDetails.DataSource = cmd.ExecuteReader();

gvDetails.DataBind();

con.Close();

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

}

protected void lnkDelete\_Click(object sender, EventArgs e)

{

try{

LinkButton lnkbtn = sender as LinkButton;

GridViewRow gvrow = lnkbtn.NamingContainer as GridViewRow;

int fileid = Convert.ToInt32(gvDetails.DataKeys[gvrow.RowIndex].Value.ToString());

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select F\_Name from file\_uploaded where FileId=@id";

cmd.Parameters.AddWithValue("@id", fileid);

cmd.Connection = con;

con.Open();

SqlDataReader dr = cmd.ExecuteReader();

if (dr.Read())

{

string fname = dr[0].ToString();

con.Close();

con.Open();

SqlCommand del = new SqlCommand("delete file\_uploaded where FileId=@dae; delete file\_data where FileId=@dae;", con);

SqlParameter sql2 = del.Parameters.Add("@dae", SqlDbType.Int);

sql2.Value = fileid;

int result = del.ExecuteNonQuery();

if (result > 0)

{

Label1.Text = "data deleted succefully";

Label1.Visible = true;

string completePath = Server.MapPath("~/Files\_DATA/" + fname); if (System.IO.File.Exists(completePath))

{

System.IO.File.Delete(completePath);

}

dr.Close();

con.Close();

}

else

{

Label1.Text = "there is no data for this date";

Label1.Visible = true;

}

con.Close();

}

else {

con.Close();

}

}

//gvDetails.DeleteRow(0);

BindGridviewFile();

gvDetails.EmptyDataText = "File Not Found";

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

}

protected void Button1\_Click(object sender, EventArgs e)

{

BindGridviewFile();

}

}

**File\_Upload.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="File\_Upload.aspx.cs" Inherits="File\_Upload" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>File Upload</title>

<style type="text/css">

body

{

font-family: Arial;

font-size: 10pt;

}

</style>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server">

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

<div class="menu\_left"> <asp:HyperLink ID="Home" runat="server" NavigateUrl="~/Default.aspx" Visible="true">Home</asp:HyperLink></div>

<asp:Label ID="LSearch" runat="server" CssClass="line" Visible="true"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Search" runat="server" NavigateUrl="~/Search.aspx" Visible="true">File Search</asp:HyperLink></div>

<asp:Label ID="LCompany" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Company" runat="server" NavigateUrl="~/Add\_Company.aspx" Visible="false">Add Company</asp:HyperLink></div>

<asp:Label ID="LRegister" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Register" runat="server" NavigateUrl="~/Register.aspx" Visible="false"> Add User</asp:HyperLink></div>

<asp:Label ID="LFDelete" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Delete" runat="server" NavigateUrl="~/File\_Delete.aspx" Visible="false"> File Delete</asp:HyperLink></div>

<div class="menu\_right"><asp:Label ID="Label3" runat="server" Text="Welcome Head" Visible="false"></asp:Label><asp:Label ID="Label4" runat="server" Text="Welcome Admin" Visible="false"></asp:Label> <asp:LoginName ID="LoginName2" runat="server" Font-Bold="true" />&nbsp;&nbsp;&nbsp;<asp:LoginStatus ID="LoginStatus2" runat="server" onloggingout="LoginStatus1\_LoggingOut" /></div>

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<center>

<br />

<b>Upload Files</b>

<hr />

<br />

<br />

<table>

<tr>

<td colspan="3">

</td>

</tr>

<tr>

<td>

User Name :

</td>

<td>

<asp:DropDownList ID="DropDownList1" runat="server" Width="200px">

</asp:DropDownList>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"

ErrorMessage="RequiredFieldValidator" ControlToValidate="DropDownList1"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>

Company Name :

</td>

<td>

<asp:DropDownList ID="DropDownList2" runat="server" Width="200px">

</asp:DropDownList>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"

ErrorMessage="RequiredFieldValidator" ControlToValidate="DropDownList2"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>

Module :

</td>

<td>

<asp:DropDownList ID="DropDownList3" runat="server" Width="200px">

<asp:ListItem Value="A" Text="A"></asp:ListItem>

<asp:ListItem Value="B" Text="B"></asp:ListItem>

<asp:ListItem Value="C" Text="C"></asp:ListItem>

<asp:ListItem Value="D" Text="D"></asp:ListItem>

<asp:ListItem Value="E" Text="E"></asp:ListItem>

</asp:DropDownList>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"

ErrorMessage="RequiredFieldValidator" ControlToValidate="DropDownList3" ></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>

Date :

</td>

<td>

YYYY:

<asp:DropDownList ID="DropDownList6" runat="server" AutoPostBack="True"

onselectedindexchanged="DropDownList6\_SelectedIndexChanged"

ontextchanged="DropDownList6\_SelectedIndexChanged">

</asp:DropDownList>

MM:

<asp:DropDownList ID="DropDownList5" runat="server" Enabled="false"

AutoPostBack="True"

onselectedindexchanged="DropDownList5\_SelectedIndexChanged">

</asp:DropDownList>

DD:

<asp:DropDownList ID="DropDownList4" runat="server" Enabled="false"

AutoPostBack="True">

</asp:DropDownList>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server"

ErrorMessage="RequiredFieldValidator" ControlToValidate="DropDownList3" ></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>

File Description :

</td>

<td>

<asp:TextBox ID="TextBox1" runat="server" TextMode="MultiLine" Width="300px"></asp:TextBox>

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator5" runat="server"

ErrorMessage="RequiredFieldValidator" ControlToValidate="TextBox1"></asp:RequiredFieldValidator>

</td>

</tr>

<tr>

<td>File Upload :</td>

<td>

<asp:FileUpload ID="FileUpload1" runat="server" />

</td>

<td>

<asp:Label ID="Label1" runat="server" Text="Label" ForeColor="Red" Visible="false"></asp:Label>

</td>

</tr>

<tr>

<td colspan="3">

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

</td>

</tr>

<tr>

<td>

</td>

<td align="center">

<asp:Button ID="Button1" runat="server" Text="Submit" onclick="Button1\_Click" />

</td>

<td>

</td>

</tr>

<tr>

<td>

Status :

</td>

<td colspan="2">

<asp:Label ID="Label2" runat="server" Text="Label" ForeColor="Red" Visible="false"></asp:Label>

</td>

</tr>

</table></center>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

<div style="width:159px; float:right; margin-top:22px"> </div> </div>

</div>

</div>

</form>

</body>

</html>

**File\_Upload.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Web.Security;

using System.Collections;

using System.Data.SqlClient;

using System.Configuration;

using System.IO;

public partial class File\_Upload : System.Web.UI.Page

{

static String dbstring = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

static SqlConnection con = new SqlConnection(dbstring);

protected void Page\_Init(object sender, EventArgs e)

{

if (!this.Page.User.Identity.IsAuthenticated)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() != "Admin" && Session["role"].ToString() != "Head")

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else if (Session["role"].ToString() == "Admin")

{

LCompany.Visible = true;

Company.Visible = true;

LRegister.Visible = true;

Register.Visible = true;

Label4.Visible = true;

LFDelete.Visible = true;

Delete.Visible = true;

}

else if (Session["role"].ToString() == "Head")

{

Label3.Visible = true;

LCompany.Visible = true;

Company.Visible = true;

}

}

}

int year, month;

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

DateTime tnow = DateTime.Now;

ArrayList AlYear = new ArrayList();

int i;

for (i = 1960; i <= 2040; i++)

AlYear.Add(i);

ArrayList AlMonth = new ArrayList();

for (i = 1; i <= 12; i++)

AlMonth.Add(i);

if (!this.IsPostBack)

{

DropDownList6.DataSource = AlYear;

DropDownList6.DataBind();

DropDownList6.SelectedValue = tnow.Year.ToString();

DropDownList5.DataSource = AlMonth;

DropDownList5.DataBind();

DropDownList5.SelectedValue = tnow.Month.ToString();

year = Int32.Parse(DropDownList6.SelectedValue);

month = Int32.Parse(DropDownList5.SelectedValue);

BindDays(year, month);

DropDownList4.SelectedValue = tnow.Day.ToString();

}

DropDownList6.SelectedItem.Text = System.DateTime.Now.Year.ToString();

try

{

//ListItem li = new ListItem();

con.Open();

SqlCommand cmd1 = new SqlCommand("select UserId,Username from Users", con);

SqlCommand cmd = new SqlCommand("select Company\_id,Name from Company", con);

SqlDataReader ddlValues1;

ddlValues1 = cmd1.ExecuteReader();

DropDownList1.DataSource = ddlValues1;

DropDownList1.DataValueField = "UserId";

DropDownList1.DataTextField = "Username";

DropDownList1.DataBind();

ddlValues1.Close();

SqlDataReader ddlValues;

ddlValues = cmd.ExecuteReader();

DropDownList2.DataSource = ddlValues;

DropDownList2.DataValueField = "Company\_id";

DropDownList2.DataTextField = "Name";

DropDownList2.DataBind();

ddlValues.Close();

con.Close();

}

catch (Exception z)

{

Response.Write(z.Message);

}

finally

{

con.Close();

}

}

}

private bool CheckLeap(int year)

{

if ((year % 4 == 0) && (year % 100 != 0) || (year % 400 == 0))

return true;

else return false;

}

//binding every month day

private void BindDays(int year, int month)

{

int i;

ArrayList AlDay = new ArrayList();

switch (month)

{

case 1:

case 3:

case 5:

case 7:

case 8:

case 10:

case 12:

for (i = 1; i <= 31; i++)

AlDay.Add(i);

break;

case 2:

if (CheckLeap(year))

{

for (i = 1; i <= 29; i++)

AlDay.Add(i);

}

else

{

for (i = 1; i <= 28; i++)

AlDay.Add(i);

}

break;

case 4:

case 6:

case 9:

case 11:

for (i = 1; i <= 30; i++)

AlDay.Add(i);

break;

}

DropDownList4.DataSource = AlDay;

DropDownList4.DataBind();

}

protected void DropDownList6\_SelectedIndexChanged(object sender, EventArgs e)

{

year = Int32.Parse(DropDownList6.SelectedValue);

month = Int32.Parse(DropDownList5.SelectedValue);

BindDays(year, month);

SetFocus(DropDownList6);

DropDownList5.Enabled = true;

}

protected void DropDownList5\_SelectedIndexChanged(object sender, EventArgs e)

{

year = Int32.Parse(DropDownList6.SelectedValue);

month = Int32.Parse(DropDownList5.SelectedValue);

BindDays(year, month);

SetFocus(DropDownList5);

DropDownList4.Enabled = true;

}

protected void LoginStatus1\_LoggingOut(object sender, LoginCancelEventArgs e)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Response.Redirect("Login.aspx");

}

protected bool check()

{

Label2.Visible = true;

if (DropDownList1.Text == "")

{

Label2.Text = "please select the User Name ";

return false;

}

else if (DropDownList2.Text == "")

{

Label2.Text = "please select the Company Name ";

return false;

}

else if (DropDownList3.Text == "")

{

Label2.Text = "please select the Module ";

return false;

}

else if (DropDownList4.Text == "" || DropDownList5.Text == "" || DropDownList6.Text == "" || DropDownList5.Enabled == false || DropDownList4.Enabled == false)

{

Label2.Text = "please select the Date ";

return false;

}

else if (TextBox1.Text == "")

{

Label2.Text = "please Enter the Description ";

return false;

}

else if (!FileUpload1.HasFile)

{

Label1.Text = "\*";

Label1.Visible = true;

Label2.Text = "please select the file first";

return false;

}

else

{

Label1.Visible = false;

Label2.Visible = false;

return true;

}

}

protected string filetype(string extension)

{

if (extension == ".pdf")

return "Acrobat Reader";

else if (extension == ".doc" || extension == ".docx")

return "Word";

else if (extension == ".xls" || extension == ".xlsx")

return "Excel";

else if (extension == ".ppt" || extension == ".pptx")

return "Power Point";

else

{

return "ERROR";

}

}

protected void Button1\_Click(object sender, EventArgs e)

{

if (check())

{

int FileId = 0, res = 0;

string Ftype, Fname;

string uploadFolder = Request.PhysicalApplicationPath + "Files\_DATA\\";

string extension = Path.GetExtension(FileUpload1.PostedFile.FileName);

Ftype = filetype(extension);

if (Ftype != "ERROR")

{

try

{

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

using (SqlCommand cmd = new SqlCommand("Insert\_File\_Data"))

{

using (SqlDataAdapter sda = new SqlDataAdapter())

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@UserId", DropDownList1.SelectedItem.Value);

cmd.Parameters.AddWithValue("@Username", DropDownList1.SelectedItem.Text);

cmd.Parameters.AddWithValue("@Company\_id", DropDownList2.SelectedItem.Value);

cmd.Parameters.AddWithValue("@Company\_Name", DropDownList2.SelectedItem.Text);

cmd.Parameters.AddWithValue("@Module", DropDownList3.SelectedItem.Value);

cmd.Parameters.AddWithValue("@CreatedDate", DropDownList5.SelectedItem.Text + "/" + DropDownList4.SelectedItem.Text + "/" + DropDownList6.SelectedItem.Text);

cmd.Parameters.AddWithValue("@file\_year", DropDownList6.SelectedItem.Text);

cmd.Connection = con;

con.Open();

FileId = Convert.ToInt32(cmd.ExecuteScalar());

con.Close();

sda.Dispose();

}

}

using (SqlCommand cmd = new SqlCommand("Insert\_File\_Uploaded"))

{

using (SqlDataAdapter sda = new SqlDataAdapter())

{

Fname = FileId.ToString() + extension;

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@FileId", FileId);

cmd.Parameters.AddWithValue("@File\_Description", TextBox1.Text);

cmd.Parameters.AddWithValue("@File\_Type", Ftype);

cmd.Parameters.AddWithValue("@File\_Path", uploadFolder);

cmd.Parameters.AddWithValue("@F\_Name", Fname);

cmd.Connection = con;

con.Open();

res = Convert.ToInt32(cmd.ExecuteScalar());

con.Close();

sda.Dispose();

FileUpload1.SaveAs(uploadFolder + FileId.ToString() + extension);

Label2.Text = "File uploaded successfully as: " + Fname;

Label2.Visible = true;

}

}

}

}

catch (SqlException ex)

{

Label1.Text = ex.ToString();

Label1.Visible = true;

con.Close();

}

finally

{

con.Close();

}

}

else

{

Label2.Text = "Unsupported file selected";

Label2.Visible = true;

}

}

else

{

ClientScript.RegisterStartupScript(GetType(), "alert", "alert(Error!);", true);

}

}

}

**Register.aspx**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Register.aspx.cs" Inherits="Register" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head id="Head1" runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>Register User</title>

<style type="text/css">

body

{

font-family: Arial;

font-size: 10pt;

}

table

{

border: 1px solid #ccc;

}

table th

{

background-color: #F7F7F7;

color: #333;

font-weight: bold;

}

table th, table td

{

padding: 5px;

border-color: #ccc;

}

</style>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server">

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

<div class="menu\_left"> <asp:HyperLink ID="Home" runat="server" NavigateUrl="~/Default.aspx" Visible="true">Home</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="HyperLink1" runat="server" NavigateUrl="~/Search.aspx">File Search</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="HyperLink2" runat="server" NavigateUrl="~/File\_Upload.aspx">File Upload</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="HyperLink3" runat="server" NavigateUrl="~/Add\_Company.aspx">Add Company</asp:HyperLink></div>

<div class="line"></div>

<div class="menu\_left"> <asp:HyperLink ID="Delete" runat="server" NavigateUrl="~/File\_Delete.aspx" > File Delete</asp:HyperLink></div>

<div class="menu\_right">Welcome Admin <asp:LoginName ID="LoginName2" runat="server" Font-Bold="true" />&nbsp;&nbsp;&nbsp;<asp:LoginStatus ID="LoginStatus2" runat="server" onloggingout="LoginStatus1\_LoggingOut" /></div>

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<center> Register User </center>

<br />

<center>

<table border="0" cellpadding="0" cellspacing="0">

<tr>

<th colspan="3" class="style5">

Registration

</th>

</tr>

<tr>

<td>

Username

</td>

<td class="style2">

<asp:TextBox ID="txtUsername" runat="server" style="margin-right: 53px"

Width="222px" />

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator1" ErrorMessage="Required" ForeColor="Red" ControlToValidate="txtUsername"

runat="server" />

</td>

</tr>

<tr>

<td>

Password

</td>

<td class="style2">

<asp:TextBox ID="txtPassword" runat="server" TextMode="Password"

style="margin-right: 53px"

Width="222px" />

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator2" ErrorMessage="Required" ForeColor="Red" ControlToValidate="txtPassword"

runat="server" />

</td>

</tr>

<tr>

<td>

Confirm Password

</td>

<td class="style2">

<asp:TextBox ID="txtConfirmPassword" runat="server" TextMode="Password"

style="margin-right: 53px"

Width="222px" />

</td>

<td>

<asp:CompareValidator ID="CompareValidator1" ErrorMessage="Passwords do not match." ForeColor="Red" ControlToCompare="txtPassword"

ControlToValidate="txtConfirmPassword" runat="server" />

</td>

</tr>

<tr>

<td>

Email

</td>

<td class="style2">

<asp:TextBox ID="txtEmail" runat="server" style="margin-right: 53px"

Width="222px" />

</td>

<td>

<asp:RequiredFieldValidator ID="RequiredFieldValidator3" ErrorMessage="Required" Display="Dynamic" ForeColor="Red"

ControlToValidate="txtEmail" runat="server" />

<asp:RegularExpressionValidator ID="RegularExpressionValidator1" runat="server" Display="Dynamic" ValidationExpression="\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*"

ControlToValidate="txtEmail" ForeColor="Red" ErrorMessage="Invalid email address." />

</td>

</tr>

<tr>

<td>

Role :

</td>

<td>

<asp:RadioButton ID="RadioButton1" runat="server" GroupName="name" Text="User" Checked="true" /> &nbsp;&nbsp;&nbsp;&nbsp;<asp:RadioButton ID="RadioButton2" runat="server" GroupName="name" style="vertical-align:middle" Text="Head"/>

</td>

<td class="style1">

</td>

</tr>

<tr>

<td class="style3">

</td>

<td class="style4">

<asp:Button ID="Button1" Text="Submit" runat="server" OnClick="RegisterUser" />

</td>

<td class="style3">

</td>

</tr>

<tr>

<td>

Status :

</td>

<td colspan="2">

<asp:Label ID="Label1" runat="server" Text="Label" Visible="false" ForeColor="Red"></asp:Label>

</td>

</tr>

</table>

</center>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

<div style="width:159px; float:right; margin-top:22px"> </div> </div>

</div>

</div>

</form>

</body>

</html>

**Register.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Configuration;

using System.Data.SqlClient;

using System.Web.Security;

public partial class Register : System.Web.UI.Page

{

protected void Page\_Init(object sender, EventArgs e)

{

if(IsPostBack)

{

Label1.Visible = false;

}

if (!this.Page.User.Identity.IsAuthenticated)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() != "Admin")

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

}

}

protected string groupcheck()

{

if ( RadioButton1.Checked)

{

return RadioButton1.Text.Trim();

}

else

{

return RadioButton2.Text.Trim();

}

}

protected void RegisterUser(object sender, EventArgs e)

{

int userId = 0;

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand("Insert\_User"))

{

using (SqlDataAdapter sda = new SqlDataAdapter())

{

cmd.CommandType = CommandType.StoredProcedure;

cmd.Parameters.AddWithValue("@Username", txtUsername.Text.Trim());

cmd.Parameters.AddWithValue("@Password", txtPassword.Text.Trim());

cmd.Parameters.AddWithValue("@Email", txtEmail.Text.Trim());

cmd.Parameters.AddWithValue("@Roles", groupcheck());

cmd.Connection = con;

con.Open();

userId = Convert.ToInt32(cmd.ExecuteScalar());

con.Close();

}

}

string message = string.Empty;

string message1 = string.Empty;

switch (userId)

{

case -1:

message = "Username already exists.\\nPlease choose a different username.";

message1 = "Username already exists.";

break;

case -2:

message = "Supplied email address has already been used.";

message1 = "Supplied email address has already been used.";

break;

default:

message = "Registration successful.\\nUser Id: " + userId.ToString();

message1 = "Registration successful.";

break;

}

Label1.Visible = true;

Label1.Text = message1;

ClientScript.RegisterStartupScript(GetType(), "alert", "alert('" + message + "');", true);

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

protected void LoginStatus1\_LoggingOut(object sender, LoginCancelEventArgs e)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Response.Redirect("Login.aspx");

}

}

**Search.aspx page**

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Search.aspx.cs" Inherits="Search" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="shortcut icon"

href="img/pix.png"/>

<style type="text/css"></style>

<link href="img/style.css" rel="stylesheet" type="text/css" />

<title>Search Files</title>

</head>

<script type="text/javascript" language="javascript">

function DisableBackButton() {

window.history.forward(-1)

}

DisableBackButton();

window.onload = DisableBackButton;

window.onpageshow = function (evt) { if (evt.persisted) DisableBackButton() }

window.onunload = function () { void (0) }

</script>

<body>

<form id="form1" runat="server">

<div class="wrapper">

<div class="top-shadow">

<div class="top">

<div class="logo"><img src="img/FMS.png" width="409" height="66" alt="FMS" /> </div>

<div class="top\_right"> <img src="img/clients\_icon.png" width="75" height="66" alt="client" /></div>

</div>

<div class="menu\_header">

<div class="menu\_left"> <asp:HyperLink ID="Home" runat="server" NavigateUrl="~/Default.aspx" Visible="true">Home</asp:HyperLink></div>

<asp:Label ID="LUpload" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Upload" runat="server" NavigateUrl="~/File\_Upload.aspx" Visible="false">File Upload</asp:HyperLink></div>

<asp:Label ID="LCompany" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Company" runat="server" NavigateUrl="~/Add\_Company.aspx" Visible="false">Add Company</asp:HyperLink></div>

<asp:Label ID="LRegister" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Register" runat="server" NavigateUrl="~/Register.aspx" Visible="false"> Add User</asp:HyperLink></div>

<asp:Label ID="LFDelete" runat="server" CssClass="line" Visible="false"></asp:Label>

<div class="menu\_left"> <asp:HyperLink ID="Delete" runat="server" NavigateUrl="~/File\_Delete.aspx" Visible="false"> File Delete</asp:HyperLink></div>

<div class="menu\_right"><asp:Label ID="Label2" runat="server" Text="Welcome User" Visible="false"></asp:Label><asp:Label ID="Label3" runat="server" Text="Welcome Head" Visible="false"></asp:Label><asp:Label ID="Label4" runat="server" Text="Welcome Admin" Visible="false"></asp:Label> <asp:LoginName ID="LoginName2" runat="server" Font-Bold="true" />&nbsp;&nbsp;&nbsp;<asp:LoginStatus ID="LoginStatus2" runat="server" onloggingout="LoginStatus1\_LoggingOut" /></div>

</div>

</div>

<div class="middle\_Shadow">

<div class="middle\_body">

<div class="menu\_left">

<table>

<tr>

<td colspan="2">

<u><b>Search By User</b></u>

</td>

</tr>

<tr>

<td>

User Name :

</td>

<td>

<asp:DropDownList ID="DropDownList1" runat="server" Width="123px" Height="16px">

</asp:DropDownList>

</td>

</tr>

<tr>

<td>

Module :

</td>

<td>

<asp:DropDownList ID="DropDownList3" runat="server">

<asp:ListItem>A</asp:ListItem>

<asp:ListItem>B</asp:ListItem>

<asp:ListItem>C</asp:ListItem>

<asp:ListItem>D</asp:ListItem>

<asp:ListItem>E</asp:ListItem>

</asp:DropDownList>

</td>

</tr>

<tr>

<td>

</td>

<td>

<asp:Button ID="Button1" runat="server" Text="Search" onclick="Button1\_Click" />

</td>

</tr>

</table>

</div>

<div class="menu\_left">

<table>

<tr>

<td colspan="2">

<u><b>Search By Company</b></u>

</td>

</tr>

<tr>

<td>

Company Name :

</td>

<td>

<asp:DropDownList ID="DropDownList2" runat="server" Width="123px" Height="16px">

</asp:DropDownList>

</td>

</tr>

<tr>

<td>

Module :

</td>

<td>

<asp:DropDownList ID="DropDownList4" runat="server">

<asp:ListItem>A</asp:ListItem>

<asp:ListItem>B</asp:ListItem>

<asp:ListItem>C</asp:ListItem>

<asp:ListItem>D</asp:ListItem>

<asp:ListItem>E</asp:ListItem>

</asp:DropDownList>

</td>

</tr>

<tr>

<td>

</td>

<td>

<asp:Button ID="Button2" runat="server" Text="Search" onclick="Button2\_Click" />

</td>

</tr>

</table>

</div>

<div class="menu\_left">

<table>

<tr>

<td colspan="2">

<u><b>Search by year</b></u>

</td>

</tr>

<tr>

<td>

Year

</td>

<td>

<asp:DropDownList ID="DropDownList5" runat="server">

</asp:DropDownList>

</td>

</tr>

<tr>

<td>

</td>

<td>

<asp:Button ID="Button3" runat="server" Text="Search" onclick="Button3\_Click" />

</td>

</tr>

</table>

</div>

<br/>

<div>

<center> Uploaded Files </center>

<br />

<center>

<asp:GridView ID="gvDetails" runat="server" AutoGenerateColumns="false" DataKeyNames="FileId">

<HeaderStyle BackColor="#df5015" Font-Bold="true" ForeColor="White" />

<Columns>

<asp:BoundField DataField="FileId" HeaderText="File ID" />

<asp:BoundField DataField="Username" HeaderText="User Name" />

<asp:BoundField DataField="Company\_Name" HeaderText="Company Name" />

<asp:BoundField DataField="Module" HeaderText="Module" />

<asp:BoundField DataField="CreatedDate" HeaderText="Date YYYY-MM-DD" />

<asp:BoundField DataField="File\_Description" HeaderText="Description" />

<asp:BoundField DataField="File\_Type" HeaderText="File Type" />

<asp:TemplateField HeaderText="File Download">

<ItemTemplate><asp:LinkButton ID="lnkDownload" runat="server" Text="Download" OnClick="lnkDownload\_Click">

</asp:LinkButton>

</ItemTemplate>

</asp:TemplateField>

</Columns>

</asp:GridView>

</center>

</div>

</div>

</div>

<div class="footer\_shadow">

<div class="footer">

<div style="width:133px; float:left; padding-left:5px" align="center"></div>

<div style="width:159px; float:right; margin-top:22px">&nbsp;</div> </div>

</div>

</div>

</form>

</body>

</html>

**Search.aspx.cs**

using System;

using System.Collections.Generic;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

using System.Data;

using System.Web.Security;

using System.Collections;

using System.Data.SqlClient;

using System.Configuration;

using System.IO;

public partial class Search : System.Web.UI.Page

{

static String dbstring = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

static SqlConnection con = new SqlConnection(dbstring);

protected void Page\_Init(object sender, EventArgs e)

{

if (!this.Page.User.Identity.IsAuthenticated)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() != "Admin" && Session["role"].ToString() != "User" && Session["role"].ToString() != "Head")

{

FormsAuthentication.SignOut();

Session["data"] = null;

Session["role"] = "-1";

FormsAuthentication.RedirectToLoginPage();

}

else

{

if (Session["role"].ToString() == "User")

{

Label2.Visible = true;

}

else if (Session["role"].ToString() == "Head")

{

Label3.Visible = true;

LUpload.Visible = true;

Upload.Visible = true;

LCompany.Visible = true;

Company.Visible = true;

}

else if (Session["role"].ToString() == "Admin")

{

Label4.Visible = true;

LUpload.Visible = true;

Upload.Visible = true;

LCompany.Visible = true;

Company.Visible = true;

LRegister.Visible = true;

Register.Visible = true;

LFDelete.Visible = true;

Delete.Visible = true;

}

}

}

}

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

ArrayList al= new ArrayList();

for (int i = 1950; i < 2051; i++)

al.Add(i);

DropDownList5.DataSource = al;

DropDownList5.DataBind();

try

{

ListItem li = new ListItem();

con.Open();

SqlCommand cmd1 = new SqlCommand("select UserId,Username from Users", con);

SqlCommand cmd = new SqlCommand("select Company\_id,Name from Company", con);

SqlDataReader ddlValues1;

ddlValues1 = cmd1.ExecuteReader();

DropDownList1.DataSource = ddlValues1;

DropDownList1.DataValueField = "UserId";

DropDownList1.DataTextField = "Username";

DropDownList1.DataBind();

ddlValues1.Close();

SqlDataReader ddlValues;

ddlValues = cmd.ExecuteReader();

DropDownList2.DataSource = ddlValues;

DropDownList2.DataValueField = "Company\_id";

DropDownList2.DataTextField = "Name";

DropDownList2.DataBind();

ddlValues.Close();

con.Close();

}

catch (Exception z)

{

Response.Write(z.Message);

}

finally

{

con.Close();

}

}

}

protected void LoginStatus1\_LoggingOut(object sender, LoginCancelEventArgs e)

{

FormsAuthentication.SignOut();

Session["data"] = null;

Response.Redirect("Login.aspx");

}

protected void Button1\_Click(object sender, EventArgs e)

{

BindGridviewData();

}

private void BindGridviewData()

{

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select fu.FileId, fd.Username, fd.Company\_Name, fd.Module, fd.CreatedDate, fu.File\_Description, fu.File\_Type, fu.File\_Path ,fu.F\_Name from file\_data fd join file\_uploaded fu on fd.FileId=fu.FileId where fd.Module=@Module and fd.UserId=@UserId order by fd.FileId desc";

cmd.Parameters.AddWithValue("@UserId", DropDownList1.SelectedItem.Value);

cmd.Parameters.AddWithValue("@Module", DropDownList3.SelectedItem.Value);

cmd.Connection = con;

con.Open();

gvDetails.DataSource = cmd.ExecuteReader();

gvDetails.DataBind();

con.Close();

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

protected void lnkDownload\_Click(object sender, EventArgs e)

{

try{

LinkButton lnkbtn = sender as LinkButton;

GridViewRow gvrow = lnkbtn.NamingContainer as GridViewRow;

int fileid = Convert.ToInt32(gvDetails.DataKeys[gvrow.RowIndex].Value.ToString());

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select File\_Path, F\_Name from file\_uploaded where FileId=@id";

cmd.Parameters.AddWithValue("@id", fileid);

cmd.Connection = con;

con.Open();

SqlDataReader dr = cmd.ExecuteReader();

if (dr.Read())

{

FileInfo fileInfo = new FileInfo(dr[0].ToString() + dr[1].ToString());

Response.Clear();

Response.AddHeader("Content-Disposition", "attachment;filename=" + fileInfo.Name);

Response.AddHeader("Content-Length", fileInfo.Length.ToString());

Response.ContentType = "application/octet-stream";

Response.Flush();

Response.WriteFile(fileInfo.FullName);

Response.End();

}

con.Close();

dr.Close();

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

}

protected void Button2\_Click(object sender, EventArgs e)

{

BindGridviewDatacom();

}

private void BindGridviewDatacom()

{

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select fu.FileId, fd.Username, fd.Company\_Name, fd.Module, fd.CreatedDate, fu.File\_Description, fu.File\_Type, fu.File\_Path ,fu.F\_Name from file\_data fd join file\_uploaded fu on fd.FileId=fu.FileId where fd.Module=@Module and fd.Company\_id=@comId order by fd.FileId desc";

cmd.Parameters.AddWithValue("@comid", DropDownList2.SelectedItem.Value);

cmd.Parameters.AddWithValue("@Module", DropDownList4.SelectedItem.Value);

cmd.Connection = con;

con.Open();

gvDetails.DataSource = cmd.ExecuteReader();

gvDetails.DataBind();

con.Close();

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

protected void Button3\_Click(object sender, EventArgs e)

{

BindGridviewDatayear();

}

private void BindGridviewDatayear()

{

string constr = ConfigurationManager.ConnectionStrings["constr"].ConnectionString;

using (SqlConnection con = new SqlConnection(constr))

{

try{

using (SqlCommand cmd = new SqlCommand())

{

cmd.CommandText = "select fu.FileId, fd.Username, fd.Company\_Name, fd.Module, fd.CreatedDate, fu.File\_Description, fu.File\_Type, fu.File\_Path ,fu.F\_Name from file\_data fd join file\_uploaded fu on fd.FileId=fu.FileId where fd.file\_year=@year";

cmd.Parameters.AddWithValue("@year", DropDownList5.SelectedItem.Value);

cmd.Connection = con;

con.Open();

gvDetails.DataSource = cmd.ExecuteReader();

gvDetails.DataBind();

con.Close();

}

}

catch (Exception ex)

{

Response.Write(ex.Message);

}

finally

{

con.Close();

}

}

}

}

**5.4 TESTING TECHNIQUE AND STRATEGIES**

There are following rules that can serve well as testing objectives:

i. Testing is a process of executing a program with the intent of finding an error.

ii. A good test case is one that has a high probability of finding an as-yet-undiscovered error.

iii. A successful test is one that uncovers as-yet-undiscovered error.

There are two types of testing techniques:

· White box testing.

· Black box testing.

White box testing:

White box test focus on the program control structure. Test cases are derived to ensure that all statement in the program has been executed at least once during testing and that all logical condition has been exercised. Basic path testing, a white box testing, makes use of program graph to derive the set of linearly independent test that will ensure coverage.

Cyclomatic Complexity:

This is software metric that provides a quantitative measure of the logical complexity of a program.

Condition Testing:

Condition testing is tests case design method that exercise the logical conditions contain in a program module. A simple condition is a Boolean variable or a relational expression.

Branch Testing:

I have used Branch testing is probably the for compound condition, the true and false (in project i.e. null values) for each branch.

Data Flow Testing:

I have used data flow testing due to check the path of program according to the locations of definitions and uses of variables in the program.

Loop Testing:

In our project I have use only simple loop. And I have use m pass through the loop where m<n.

BLACK BOX TESTING:

Black box testing focuses on the functional requirements of the software. That is, black-box testing enable the software engineer to derive set of input conditions that will fully exercise all functional requirements for a program.

Graph-Based Testing Method:

I have used graph-based testing method for removing errors associated with relationships. The first step in this testing is to understand the objects that are modeled in software and the relationship that connect these objects.

Equivalence Partitioning:

This testing is used for the following reason:

§ Specific numeric values

§ Range of values

§ Set of related values

§ Boolean condition

For example: Check in phone number, code generation, voucher type, bill type, password etc.

Boundary Value Analysis:

Boundary value analysis is a test case design technique that complements equivalence partitioning. Rather than selecting any element of equivalence class the selection of test cases at the edges of the class. Rather than focusing solely on the input condition,

The point of equivalence partitioning as;

1. An input condition specifies a range boundary by values a and b, test cases should be design with values a and b and just above and just below a and b.

2. An input condition specific a number of values, test cases should be developed that exercise the minimum and maximum number.

**TESTING STRATEGIES**

System testing

During system testing the system is used experimentally to ensure that the software does not fail i.e. it will run according to its specifications and in the way users expect. Special test data are input for the processing and the results examine. A limited number of users try to use it in unforeseen ways. It is preferable to discover any surprise before the organization implements the system and depend upon it. In many organizations persons other than those who wrote the original programs to ensure more complete and unbiased testing and more reliable testing perform testing. The norms that were followed during the phase were that after the developer of the software has satisfied regarding every aspect of the software under consideration he is required to release the program source code. A setup name release is used to copy the name file from the developers’ user area to a project area in the directory named with developer user name. Here all the final testing used to be done by persons other than the developer himself .if some changes were desired in the program the developer were required to use another setup. Retrieve, which copied back the latest version of the program to developer areas. As in this system data is entered at different levels I considered providing various types of checks like range check, validity check, completeness check etc. in different data entry screens according to the requirements. Since the user are not familiar to the new system the data screens were designed in such a way that were-.

§ Consistent

§ Easy to use

§ Has a fast response time

The following convention were used while designing the various screen:

Unit testing:

In unit testing I have testing a single program module in an isolated environment. Testing of the processing procedures is the main focus.

Integration testing:

Because of interfaces among the system modules, I use integration testing. In other word’s it ensures that the data moving between the modules is handled as intended.

System testing:

System testing is the testing of the system against its initial objective. It is done either in a simulated environment or in live environment.

Test review:

Test review is the process that ensures that testing is carried out as planned. Test review decides whether or not the program is ready to be shipper out for implementation

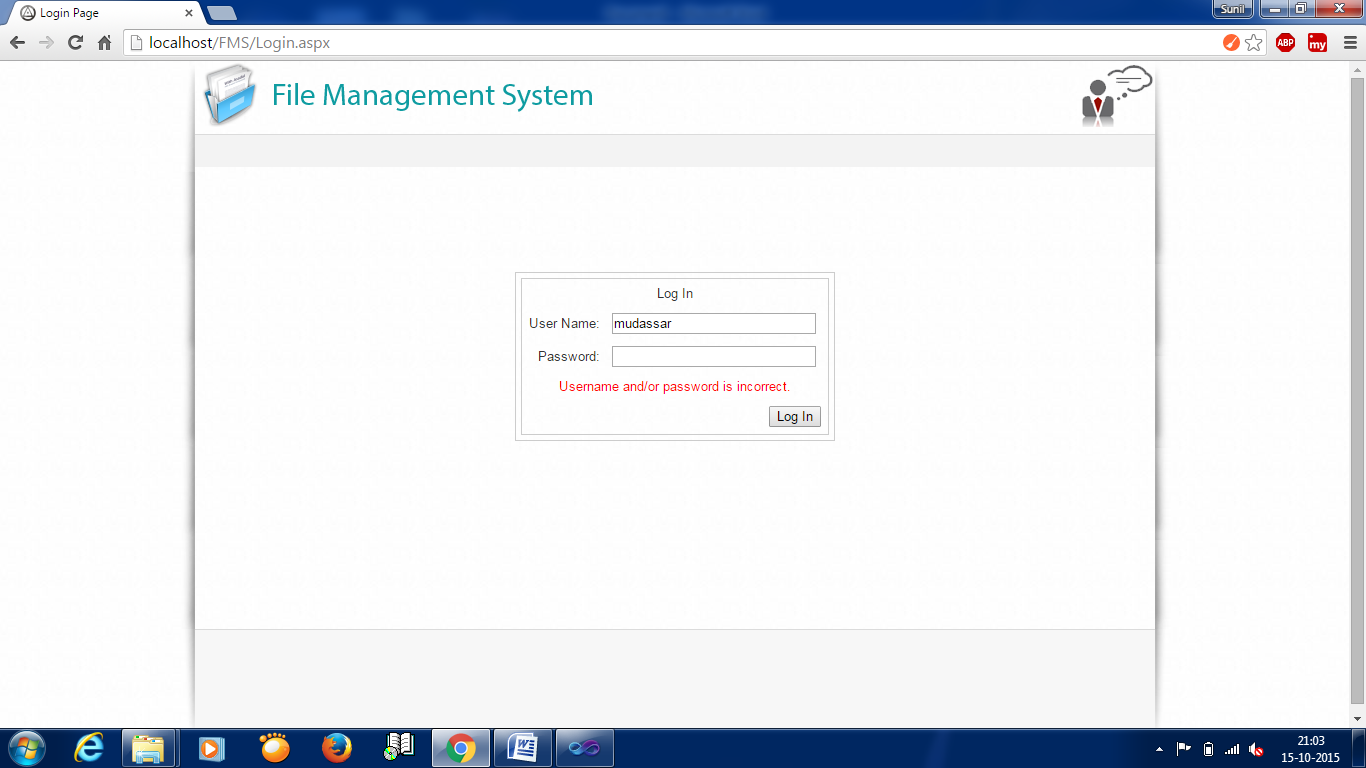
Security testing:

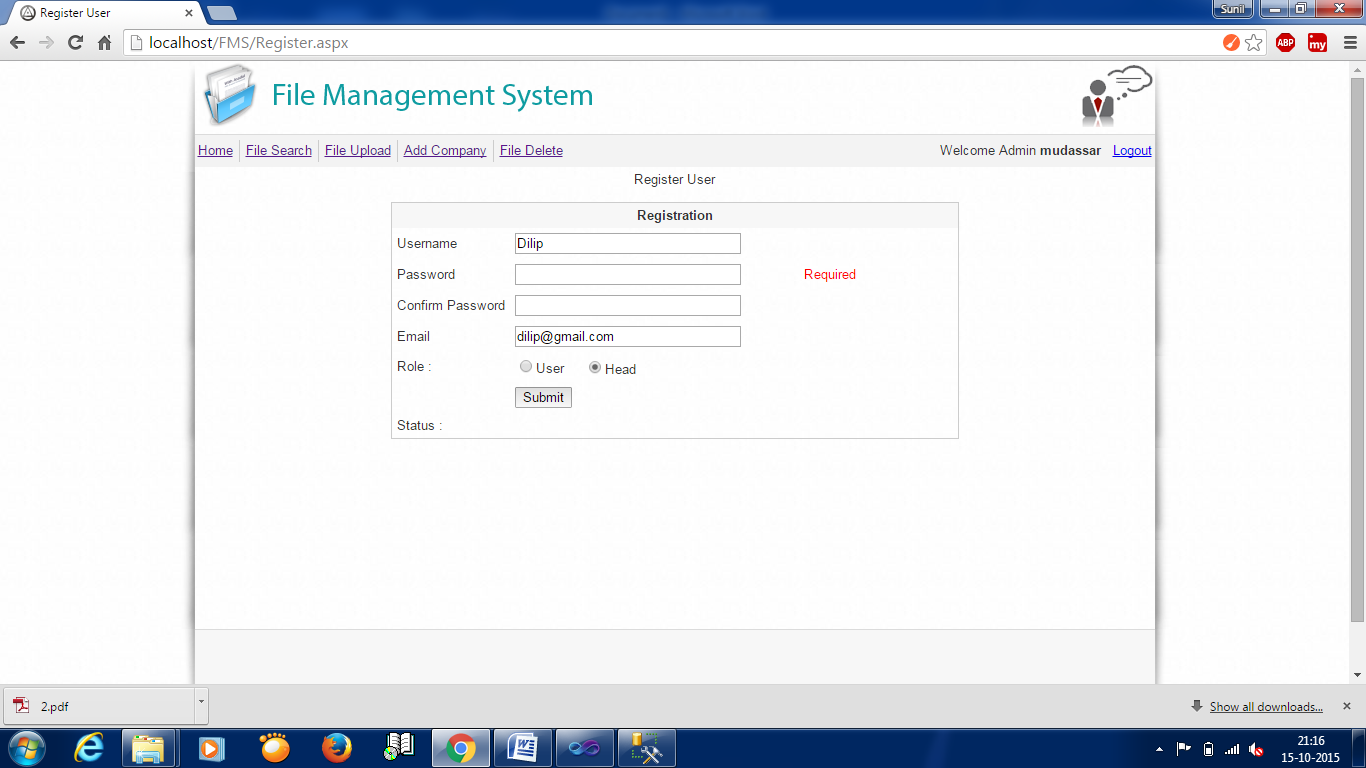
Security testing attempt to verify that protection mechanism built into a system will, in fact, protect it form penetration mechanisms.

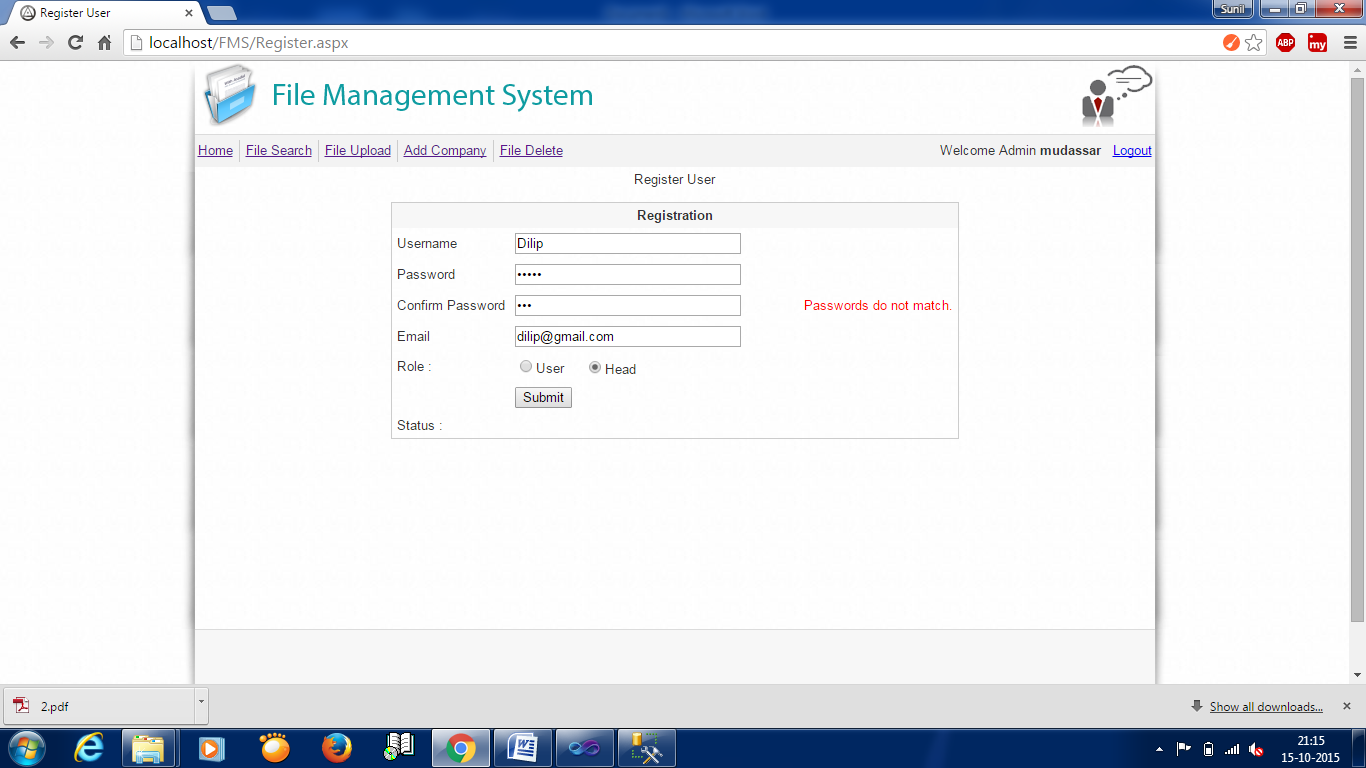
**6. Results and Discussion**

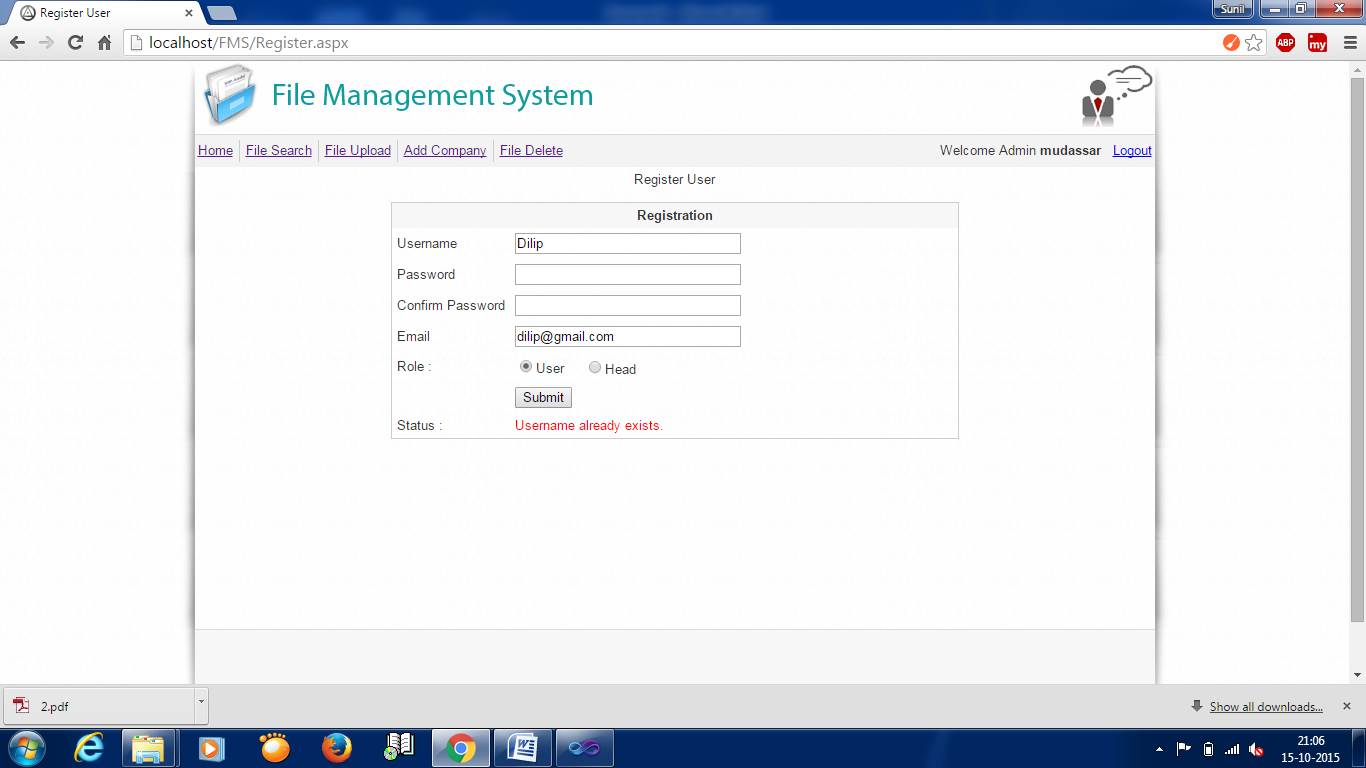
**6.1 Test Results :**

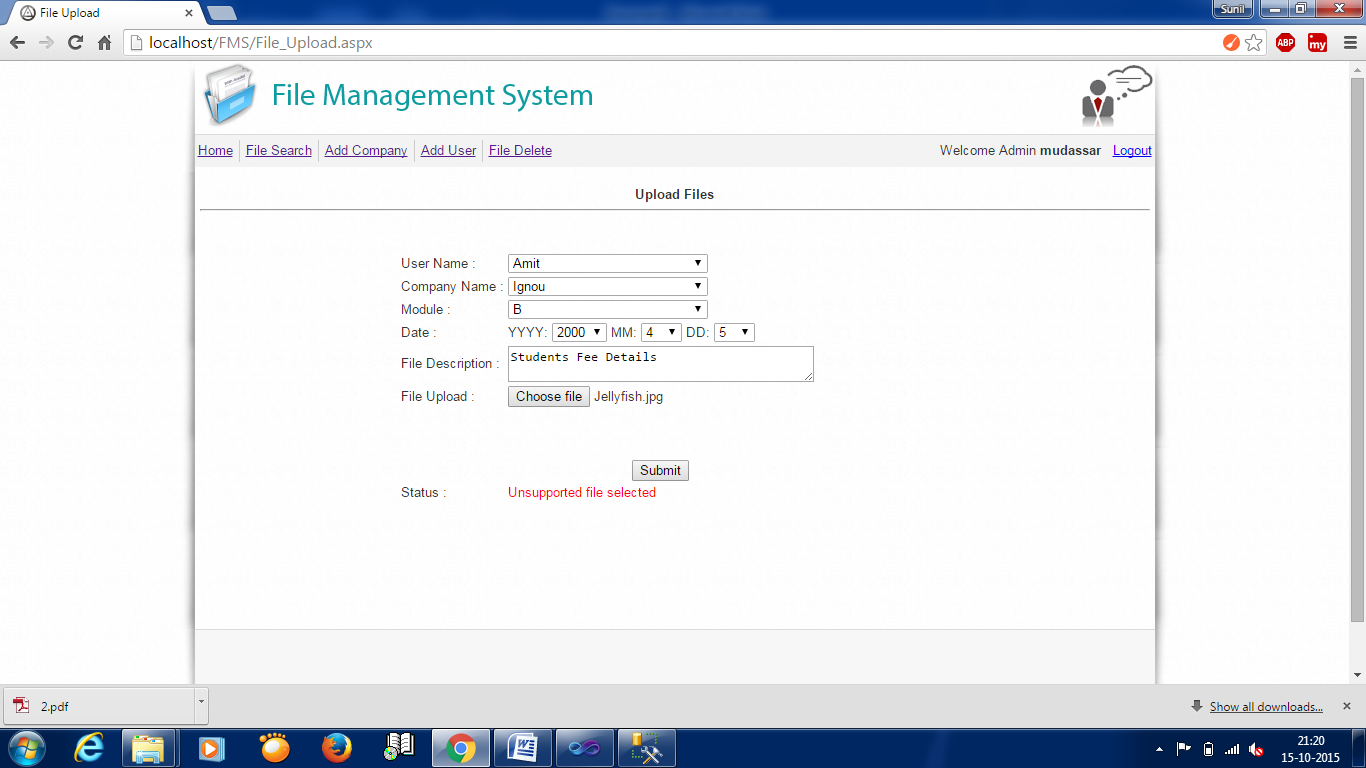
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Case ID**: 1.0  **Test Priority (Low/Medium/High)**: Med  **Module Name**: Final Check  **Test Title:** Check Validation and Functionality  **Description**: Complete Project Testing | | | **Test Designed by**: Sunil  **Test Designed date**: 30/09/2015  **Test Executed by**: Sunil  **Test Execution date**: 01/10/2015 | | |
| 1. | Go to webiste Enter Invalid User Name and Password | Id: Mud  Pwd:123 | Invalid User name and Password | Invalid User name and Password | pass |
| 2. | Go to Register Page and Create User without Enter Password | Password Field Blank | Password is Required | Password is Required | pass |
| 3. | Go to Register Page and Create User Re-Enter Password Check | Enter Different Password In Re-match password | Password do not match | Password do not match | Pass |
| 4. | Go to Register Page and Create User With Existing User Name | Enter Already Registered User Name | Username Already Exist | Username Already Exist | Pass |
| 5. | Go to Upload File Page and try to upload File Unsupported File | Browse : image file | Unsupported File Selected | Unsupported File Selected | Pass |

**6.2 Test Case Result Screens :**









**7.1 Conclusion :**

The Computer Based Management is an asset for CHARTED ACCOUNTANT FIRM Looking forward to a more efficient and streamlined processing system and better coordination between customers and the finance leading to desired results in terms of quality, service, profits, market, share and competitive ness.

**7.2 limitation of System :**

1. Unusable Site During Upgrades

### The Need to Deal with Comment Spam in Blogs

### The Software Uses More Resources on Your Web Server

### Changing a Web Host Requires Different Steps

### Potential Security Risks through the Online Scripts

**7.3 Future Scope of the Project:**

It is unreasonable to consider a computer based information system complete or finished; the system continues to evolve throughout its life cycle, even if it’s successful. It is the case with this system too. Due to the creative nature of the design, there remain some lapse-mistaken communications between the users and the developers. So, certain aspects of the system must be modified as operational experience is gained with it. As users work with the system, they develop ideas for change and enhancements.

**11. Bibliography**

The books taken as guidance for the completion of the project are :

* + - * Ignou Study Material
      * The complete reference C# – forth edition
      * By : Herbert Schildt
      * Publisher : Tata McGraw Hill Education private limited.
      * Software Engineering.
      * By : Roger S.Pressman.
      * Publisher : McGraw Hill.
      * The C# Handbook
      * By : Patrick Naughton.
      * Publisher : McGraw Hill Education
      * System Analysis and design.
      * By : Elias M.Awad.
      * Publisher : Galgotia.
      * An integrated approach to software engineering.
      * By : Pankaj Jalote.
      * Publisher : Narosa publication.
      * Fundamental of Java programming.
      * By : IGNOU Study material.
      * Web Resource :
* [www.java.about.com](http://www.java.about.com)
* [www.oracledocs.com](http://www.oracledocs.com)
* [www.javascript.com](http://www.javascript.com)

[www.ignou.ac.in](http://www.ignou.ac.in)

**APPENDIX A**

ABOUT THE OPERATING SYSTEM

Windows is the world’s most popular operating system and one reason for this is its Graphical Use Interface (GUI). Windows lets users issue commands by clicking icons and work with programs within easily manipulated screens called (appropriately) windows. Windows 7 represents the marriage of the windows operating system and Internet accesses. This unique melding of form and function known as Web integration helps the user to perform routine computer tasks such as writing a letter while maintaining seamless access to the information we need from the Internet. Web integration also changes the way we interact with the windows operating system. Command and navigation procedures, as well as the look of the Windows 7 interface, all more closely resemble their counterparts on the web. Windows 7 lets the user to manage the files and folders that contain them using the methodology of the Internet and the World Wide Web. User can use multiple monitors with a single computer dramatically increasing the size of the workplace. Installing new hardware is easy because Windows 7supports the Universal Serial Bus standard allowing to plug in new hardware and use it immediately without restarting computer.

More reliable: User can support online website for answers to common questions and to keep copies of windows up-to-date. Windows 7 tools can help regularly and test hard disk and system files and even automatically fix some problems. The troubleshooters and the Dr. Watson diagnostic tool also help to solve computer problem.

Faster: By using the maintenance wizard we can easily improve computers speed and efficiency. The power management feature allows newer computers to go into hibernation mode and awaken instantly instead of requiring shutting down and restarting computer. We can use the FAT32 file system to store files more efficiently and save hard disk space.

True web integration: The Internet connection wizard makes connecting to the web simple. Using the web –style Active Desktop can view web pages as the desktop wallpaper. In Microsoft Outlook @ Express we can send e-mail and post messages to Internet news groups.

More entertaining: Windows 7 supports DV, digital audio and VRML so can play high quality movies and audio on computer as well as see the full effect of web pages that use virtual reality features. Can also watch television broadcasts and check TV program listings by using Microsoft Web TV for windows.

**APPENDIX B**

ABOUT VISUAL STUDIO 2010

The need of today’s software development in a G.U.I based front-end tool, which can connect to relational database engines. This gives the programmer the opportunity to develop client/server based commercial applications.

These applications give users the power and ease of use of a G.U.I with the multi-user capabilities of NT based RDBMS engines like SQL SERVER 2008.

From the array of G.U.I based front-end tools I select Developer 2000 because as we know that developer 2000 is a product of SQL SERVER 2008 corporation and it has best compatibility with SQL SERVER 2008 and most of all the security in VISUAL STUDIO 2010 is as same as in SQL SERVER 2008 database.

SQL SERVER 2008 VISUAL STUDIO 2010 offers a host of technical advantages over many other front-end tools.

**APPENDIX C**

Introduction to the SQL SERVER 2008 Server

This chapter provides an overview of the SQL SERVER 2008 server. The topics include:

Introduction to Databases and Information Management

Database Structure and Space Management

Memory Structure and Processes

The Object-Relational Model for Database Management

Data Concurrency and Consistency

Distributed Processing and Distributed Databases

Startup and Shutdown Operations

Database Security