

A
Project Report On
Cloud Storage Service
A Document Manager (developed using ASP.net)

In Partial Fulfillment of the Requirements
For the Degree of
B.Tech
in
Computer Science

Submitted To:
Dr. Deepak Dahiya
Professor
Computer Science & IT
JUIT,Solan

Submitted By:
Suchita Aggarwal
101257



May-2014

Department of Computer Science and IT
Jaypee University of Information Technology

Table of Content

| | |
|---|-------|
| Certificate | I |
| Acknowledgement | II |
| List of Figures | III |
| Chapter 1: Introduction..... | 1-5 |
| 1.1 Abstract | |
| 1.2 Topic of the System | |
| 1.3 Topic Background | |
| 1.4 Objective of the System | |
| Chapter 2: Project Review..... | 6-9 |
| 2.1 Front End | |
| 2.2 Back End | |
| 2.3 Cloud Hosting Platform | |
| Chapter 3: Project work..... | 9-13 |
| 3.1 Definition of Problem | |
| 3.2 H/W & S/W Requirements | |
| 3.3 System Analysis | |
| 3.4 Initial Investigation | |
| 3.5 Information Gathering | |
| 3.6 Feasibility Study | |
| 3.7 Cost & Benefit Analysis | |
| Chapter 4: Project Design..... | 14-48 |
| 4.1 UML Diagrams | |
| 4.2 Snapshots | |
| 4.3 Code | |
| 4.4 Testing | |
| 4.5 Implementation | |
| 4.6 Maintenance | |
| Chapter 5: Conclusion & Future scope..... | 49 |
| Chapter 6: References..... | 50 |

Certificate

This is to certify that the work titled “Document Manager” submitted By Suchita Aggarwal in partial fulfillment for the award of degree of B.Tech Computer Science Engineering of Jaypee University of Information Technology, Waknaghat has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

(Signature of Supervisor)

Name of Supervisor: Prof. Dr. Deepak Dahiya
Professor, Dept. of CSE and ICT

ACKNOWLEDGEMENT

I owe a great thanks to many people who have been helping and supporting us during this project. Our deepest thanks to Dr. Deepak Dahiya, the Project Guide for guiding and correcting us at every step of our work with attention and care. He has taken pain to go through the project and make necessary correction as and when needed. Thanks and appreciation to the helpful people at college for their support. We would also thank our university and my faculty members without whom this project would have been a distant reality. We also extend our heartfelt thanks to our family and well-wishers.

Signature of the Student.....

Name of the Student – Suchita

Date -

List of figures

| | |
|------------------------------------|----|
| Fig1 Design of application..... | 16 |
| Fig2 User Interaction Diagram..... | 16 |
| Fig3 Use Case Diagram..... | 17 |
| Fig4 Sequence Diagram..... | 17 |
| Fig5 Layers Of Architecture..... | 48 |

Chapter 1: Introduction

1.1 ABSTRACT

Cloud computing is a style of computing in which virtualized and standard resources, software and data are provided as a service over the Internet. Cloud computing is a new method to add capabilities to a computer without licensing new software, investing in new hardware or infrastructure or training new personnel. Applications are purchased, licensed and run over the network instead of users desktop. It provides common business applications online that are that are accessed from a web browser, while the software and data are stored on the servers. The services are accessible anywhere in the world, with the cloud appearing as a single point of access for all the computing needs of consumers. New advances in processors , virtualization technology, disk storage, broadband internet access and fast, inexpensive servers have all combined to make cloud computing a compelling paradigm. Cloud computing allows users and companies to pay for and use the software and storage that they need, when they need then and as wireless broadband connection options grow, where they need them. This type of software deployment is called Software as a Service (SaaS) .

Advantages and Issues Related to Cloud Computing for Web Application

There are several advantages to cloud computing and some of them are presented below:

Price:

It is easy to see that start up enterprise companies do not have to invest huge sums of money into setting up infrastructure such as huge application servers, data servers, database administrators, people resources for managing such critical systems including back up and recovery, etc. Instead, enterprise companies pay for services based on usage.

Simplicity:

It is simple to use and set up all the services without having to worry about resource management and other hassles that come with infrastructure set up and management.

Reliability:

Network and data access are guaranteed to be reliably maintained as the

service provides are experts in maintaining the infrastructure and such reliability is backed by some kind of “money back guarantees” or penalties for the providers in the event if they have a down time.

Flexibility:

Service consumers have the flexibility to “outsource” parts of the infrastructure and can still maintain to some extent proprietary data at their own site.

Collaboration:

Since all the applications are on the cloud, it becomes a natural fit for consumers to effectively collaborate on a common

project or application.

1.2 Topic of the system:

Project “Document Manager” is aimed at developing a web application on cloud to help document management of an individual or a group .It is web –based application that can be accessed by all authorized user. It allows users to upload files that could then be accessed over the internet from a different computer or other devices by the same user or possibly shared by other users after proper authentication. This claims to add and maintain document of an individual which will be accessible anywhere , anytime. It will provide an online way of accessing the documents .It provides online procedure for add new PDF files and make them share to other single or in group.

It will act as a complete resource for user to gain access of document details and lists and provisions to delete and download all available documents and will also enable sharing after authenticating users.

Here admin can create edit and delete user name and password .Admin is able to change the password of an existing user.

Significance Of The Work Done

To get correct knowledge of applications on cloud can turn out to be very useful. Demand of managing soft document and it’s easy to mention record of all Items. guidance in stylish electronic way is bouncing high. A

discussion with my project guide produced enthusiastic response and I decided to take the challenge. Also this topic is chosen as it is going to give me a lifetime experience studying and developing a full-fledged Web Application as per today trends and help me chocking out a nice career for myself. In current Competitive Environment every user wants to keep their records in an online database. The basic motive for this is just to reduce the paper work, avoiding usages of hard ware storage unit and increase the efficiency.

1.3 Topic background

Overview Of The Project:

The project is based on designing an application of managing documents and deploying it on cloud. The project includes the following features:

Admin should be able to

- Create user name password.
- Add documents for users.
- Remove file and document.
- Show list of PDF document.
- Organization administration is able to see the user records.
- Data of all the files added.
- A complete Training kit of Microsoft .Net Framework 2.0 Web Based client development.
- It provides complete security assurance.
- Designed in a manner to facilitate easy user friendly navigation
- User and admin log as per there credentials.
- It supports PDF document formats.

1.4 Objectives Of The System:

The following objectives have been set:

- Adequate validation checks for Information.
- Adequate security of data.
- Facility to update data from time to time.
- Prompt and specific retrieval of data.
- Flexibility in the system according to the changing environment.
- Controlling redundancy in storing the same data multiple times.
- Accuracy, timeliness and comprehensiveness of the system output.
- Stability and operability by people of average intelligence.

Identification Of Need:

Document Manager is an attempt to automate the traditional method of keeping records of documents manually and also to give the facility of accessing their documents anywhere and anytime. Traditional system had following disadvantages:

- 1) All the details of the user and doc were being entered on paper manually which was an extremely cumbersome as well as time consuming process.
- 2) Redundancy of data and non-availability of documents at other places.

Considering above drawbacks in, there was a need felt for having an automated data Management System in place which could address all the above issues and provide viable solutions.

Preliminary Investigation:

The basic purpose behind Preliminary Investigation is to first clarify, understand and evaluate the Project Scope and need of developing and working on the project.

It means making an informed decision about feasibility of the work by gathering enough information before starting working. This sort of investigation provides us with a through picture of the kind of software and hardware requirements which are most feasible for the system, plus the environment in which the entire project has to be installed and made operational.

Scope Of The Project:

Completion of the development process will result in a software package that will provide user-friendly environment, which is very easy to work with, even for people with very little knowledge of form submit application of organization.

Management of various tasks is incorporated in the package and will deliver the required information in a very easy to use and easy to access manner.

This package will provide accuracy, efficiency, speed and easiness to the end user. Since the system is verified with valid as well as invalid data and is run with an insight into the necessary modifications that may require in the future, it can be maintained successfully without much hassle.

Benefits of the proposed system:

The benefits of the proposed system must also be evaluated. Benefits may be categorized as tangible or intangible.

Tangible benefits which are measured in money terms consist of the saving of

- Time.
- Certain operating costs.

Intangible benefits are more difficult to estimate and justify. They are often impossible to give a money value to. These may include:

- Satisfaction of the companies and Employees
- Efficiency.

Methodology Adopted:

Prototyping Model has been used for software development according to which a throwaway prototype of the proposed system, based on the currently known requirements, is given to the user so that he has a fair idea about how the proposed system is going to be like. This will help him in deciding the interface, input and output requirements.

It can be easily adjudged that inputs and outputs are big in number, can increase exponentially and may create a big chaos if not restricted properly. As the user spends some time on the prototype, he will become more precise about his own input and output requirements. This prototype will provide him with an environment analogous to the proposed system's environment.

Because of object oriented support in .NET, various concepts (like reusability, polymorphism, isolation etc.) are already there but for the efficient management of system components, Component based Software Engineering will also be exercised which will help in a resultant library of components, the benefit of which will be reusability and fast development.

Because of lack of hierarchical structure in object-oriented approach, there is no meaning of Bottom-up or Top-down testing. Testing will begin from the most rudimentary levels of the system and will move towards higher level components which will be based on design phase rather than coding phase. In little words, it can be said that 'CLUSTER Testing' will be exercised to scrutinize all the parts and their associative functionality.

Chapter 2:Project Review

Working Environment

2.1Understanding .Net Frame Work:

Definition:

The .Net framework can be defined as a language neutral platform designed to provide a number of rich and powerful application development tools and technologies. It has an extensive class library that provides wide-ranging support for data access.

.Net framework has an independent code execution and management environment called the Common Language Runtime (CLR) which ensures that code is safe to run, and provides an abstract layer on top of the operating system, which allows the elements of the .Net framework to run on many operating systems and devices.

Why .Net platform chosen:

The windows platform is inherently complex and may pose several problems for programmers. However the .Net platform is used as it provides solutions to many such problems and offers an Internet centric platform.A

1)Multiple platform Support:

2) Using .Net there is just one simple object oriented way of accessing the functionality of the .Net Framework.

.

Testing Technologies Used:

As already mentioned, the whole development process is object oriented and implementation will also be in object oriented environment i.e. .NET. So, anticipated testing techniques automatically imply to be object oriented. In object oriented approach, the semantic constructs appear at all the stages, so it is better to find the problem in early stages so that we start testing parallel to development process and will be taken in consideration at each stage.

Grossly, the testing strategy is divided in the following steps:

Step 1: Object oriented analysis and object oriented design models will be checked for consistency.

Step 2: Unit testing will be exercised in context to object orientation in which each class with their methods and operations are scrutinized.

Step 3: Integration testing at various stages of testing will be exercised to find associative errors. Regression testing will be utilized but at the same time, it will also be based on object-oriented view. Specifically, Cluster testing is a good choice.

Step 4: Finally, termination of testing will be supported by Validation testing and System testing which includes requirements, review, stress testing, security testing, recovery testing etc.

2.2 Microsoft SQL Server™ 2008

Microsoft SQL Server 2008 is a full-featured relational database management system (RDBMS) that offers a variety of administrative tools to ease the burdens of database development, maintenance and administration. In this article, we'll cover six of the more frequently used tools: Enterprise Manager, Query Analyzer, SQL Profiler, Service Manager, Data Transformation Services and Books Online.

FEATURES:

User-Defined Functions: SQL Server 2008 introduces the long-awaited support for user-defined functions. User-defined functions can take zero or more input parameters and return a single value—either a scalar value like the system-defined functions, or a table result. Table-valued functions can be used anywhere table or view expressions can be used in queries, and they can perform more complex logic than is allowed in a view.

Indexed Views: Views are often used to simplify complex queries, and they can contain joins and aggregate functions. In SQL Server 2000 Enterprise or Developer Edition, you can define indexes on views to improve query performance against the view. When creating an index on a view, the result set of the view is stored and indexed in the database. Existing applications can take advantage of the performance improvements without needing to be modified.

Indexed views can improve performance for the following types of queries:

Joins and aggregations that process many rows

Join and aggregation operations that are performed frequently within many queries

Decision support queries that rely on summarized, aggregated data that is infrequently updated.

Distributed Partitioned Views: SQL Server 2008 expands the ability to create partitioned views by allowing you to horizontally partition tables across multiple SQL Servers. The feature helps you scale out one database server to multiple database servers, while making the data appear as if it comes from a single table on a single SQL Server. In addition, partitioned views are now able to be updated.

New Data types:SQL Server 2008 introduces three new data types. Two of these can be used as data types for local variables, stored procedure parameters and return values, user-defined function parameters and return values, or table columns:

Text in Row Data:SQL Server 2008 provides a new text in row table option that allows small text and image data values to be placed directly in the data row, instead of requiring a separate data page. This can reduce the amount of space required to store small text and image data values, as well as reduce the amount of I/O required to retrieve rows containing small text and image data values.

Cascading RI Constraints:SQL Server 2008 provides the ability to specify the action to take when a column referenced by a foreign key constraint is updated or deleted. You can still abort the update or delete if related foreign key records exist by specifying the NO ACTION option, or you can specify the new CASCADE option, which will cascade the update or delete operation to the related foreign key records.

Multiple SQL Server Instances:SQL Server 2008 provides support for running multiple instances of SQL Server on the same system. This allows you to simultaneously run one instance of SQL Server 6.5 or 7.0 along with one or more instances of SQL Server 2008. Each SQL Server instance runs independently of the others and has its own set of system and user databases, security configuration, and so on. Applications can connect to the different instances in the same way they connect to different SQL Servers on different machines.

XML Support:Server 2000 provides native support for XML. This new feature provides the ability to do the following:

Return query result sets directly in XML format.

Retrieve data from an XML document as if it were a SQL Server table.

Access SQL Server through a URL using HTTP. Through Internet Information Services (IIS), you can define a virtual root that gives you HTTP access to the data and XML functionality of SQL Server 2008.

2.3Microsoft Windows Azure

Microsoft Azure is a cloud computing platform and infrastructure, created by Microsoft, for building, deploying and managing applications and services through a global network of Microsoft-managed datacenters. Websites can be built in ASP.NET,PHP and can be deployed. It is used to create scalable applications and services. Supports multi-tier scenarios and automated deployments. Microsoft Azure uses a specialized operating system, called Microsoft Azure, to run its "fabric layer" — a cluster hosted at Microsoft's datacenters that manages computing and storage resources of the computers and provisions the resources (or a subset of them) to applications running on top of Microsoft Azure.

Chapter 3 Project work

3.1-Definition of Problem

The problem here is of designing and implementing a web application will be hosted on cloud through which users can log in ,create their account and use the application for management of their documents.Organisation system documents have become quite popular with so many educational institutes,corporate and public offices.Today such a application is used in almost every sphere of work.he application will provide a very user friendly method to store,add,delete,share and access a user documents anywhere with a great amount of ease.

System Feasibility:

Prior to stating whether the system we have to develop is feasible or not we believe that we should emphasize on what is implied by the word “Feasibility”. Feasibility is the measure of how beneficial or practical the development of the system will be to the organization. It is a preliminary survey for the systems investigation. It aims to provide information to facilitate a later in-depth investigation.

Functional Requirements:

- Application will help manage documents of users.
- Users will be able to access their documents anytime anywhere.
- Users will be able to share documents after proper authentication.

Administrator will be able to add/delete users.

- Administrator will keep record of all users and their activities.

Non Functional Requirements:

They are the quality requirements that stipulate how well a software does what it has to do.

Performance

No. of terminals to be supported is dependent on the server that we will use at the time of deployment.The web application server used should provide good performance and ability to manage performance with techniques such as support for caching.Document management,uploading will be fast.

Availability

Application has 24*7 availability. It can be accessed for 24 hours a day.

Application will be accessible anywhere anytime.

Usability

The website should be user friendly and should require least effort to operate.

Portability

The application is made using technologies which are platform independent and can be transported to other servers with minimum effort.

Flexibility

It is effort required to modify operational program. The whole application is made using independent modules so that any changes done in 1 module should not effect the other one and new modules can be added easily to increase functionality.

3.2-Hardware and Software Requirement

.NET Framework System Requirements:

To ensure adequate performance, .NET Framework has the following minimum and recommended system requirements for client and server applications:

Operating System Requirements:

The .NET Framework is supported on the following platforms:

| Scenario | Operating System |
|-----------------|---|
| Client | MS Windows 98 |
| | MS Windows 98 Second Edition |
| | MS Windows Millennium Edition |
| | MS Windows NT 4.0 Workstation with service pack 6.0a or later |
| | MS Windows NT 4.0 Server with service pack 6.0a or later |
| | MS Windows 2000 Professional |
| | MS Windows 2000 Server |
| | MS Windows 2000 Advanced Server |
| | MS Windows 2000 Datacenter Server |
| | MS Windows XP Home Edition Professional |
| | MS Windows Server 2003 Family |

Note: On all these systems, MS Internet Explorer 5.01 or later and MS Windows Installer 2.0 or later are also required.

| | |
|--------|---|
| Server | MS Windows 2000 Professional |
| | MS Windows 2000 Server with service pack 2.0 |
| | MS Windows 2000 Advanced Server with service pack 2.0 |
| | MS Windows 2000 |
| | Datacenter Server with service pack 2.0 |
| | MS Windows XP Professional |
| | MS Windows Server 2003 Family |

Additional Software Requirements:

To use additional features such as ASP.Net, COM+ services and SQL Server .NET Data Provider, you will need the following additional software's:

| Scenario | Feature | Required Software |
|----------------------------|---|---|
| Client Provider | SQL Server .Net Data Access to system management Instrumentation (WMI) (installed with OS on Windows 2000, Windows Millennium Edition and Windows XP) | MS Data Access Components (MDAC) 2.6 or later Windows Management |
| COM+ Services | | Windows 2000 with service Pack 2.0 |
| Server Provider | SQL Server .Net Data | MS Data Access Components (MDAC) 2.7 |
| ASP.Net Services (IIS) 5.0 | | MS Internet Information |

Hardware Requirements:

| Scenario | Processor Recommended | RAM Required | RAM Recommended |
|-----------------|------------------------------|-----------------------|------------------------|
| Client | Pentium 500 MHz* | Pentium 1 GHz 64 MB* | 128 MB or higher |
| Server | Pentium 667 MHz* | Pentium 1 GHz 128 MB* | 256 MB or higher |

3.4-INITIAL INVESTIGATION

This is the first step in system development life cycle. For making a project, it is very important to understand the overall working of the system. To know about the overall working of the system initial investigation is carried out. Initial investigation helps to identify the need. To make a candidate system request is given by the user either to change the system, improve or enhance the existing or to make the new system. Requests are coming because of limitation in the present system or here may not system available.

3.5Information Gathering:

Kind of information needed: -

Before one determines where to go and what tool to use, first requirement is to figure out what information to gather. The basic information required is how organization of personnel or official contacts as well as appointment takes place.

Sources of information

Information is gathered from two main sources namely personal and written documents within the police stations.

Two main sources of information are: -

- 1) External sources.
- 2) Internal sources.

Information gathering tools: -

There are no general rules for using the tools for the information gathering. An important rule is that information is acquired accurately, methodically under right Conditions and with minimum interruption to the user staff.

Following are the common information gathering tools: -

Review procedure manuals

Onsite observation

Interview

Questionnaires

Work measurement

Sampling

Meeting

3.6 Feasibility Study

The dictionary meaning of feasibility is “the work study, we need to consider the economic, technical and behavioral factors in the system development. Feasibility study is procedure that identifies, describes and evaluates candidate system and selects the best system for the job.

Depending on the result of the initial investigation, the survey is expanded to a more feasibility study. A feasibility study is a test of system proposal according to its workability impact on the organization, ability to meet the user needs, and effective use of resources.

Each of these questions must be answered carefully. They revolve around the investigation and evaluation of the problem, identification and description of candidate system, specification of performance and not cost of each system, and the final selection of the best system. The objective of the feasibility study is not to solve the problem but to acquire a sense of its scope.

The result of the feasibility study is a formal proposal. This is simply a report – a formal document detailing the nature and scope of proposed system. Feasibility study involves many steps. After all the steps feasibility report is prepared, which is given to the management for the approval of the project. The proposal summarizes what is known and what is going to be done. Therefore feasibility study also plays major role in the system development life cycle.

Feasibility study is the test of the system proposal according to its, workability, impact on the organization ability to meet the user needs and effective use of resources. The

objective of feasibility is not solving the problem but a sense of its scope. During the feasibility study the problem definition is crystallized and aspects of the problem to be included in the system are determined. Consequently the cost and the benefits are estimated with greater accuracy at this stage.

An initial investigation culminates in a proposal that determine whether an alternative system is feasible. This is the step to determine what the candidate system is to do by defining its expected performance. Thus, a feasibility study is carried out to select the best system that meets performance requirements. This entails identification, description and evaluation of candidate system and selection for the best system for the job. This then addresses the system performance definition and expounded on the feasibility study as a second major step in system development life cycle.

Feasibility considerations

There are three essential considerations that are involved in the feasibility analysis.

Technical Feasibility

Economic Feasibility

Behavioral Feasibility

Technical Feasibility:

Technical feasibility centers on the existing computer system (hardware, software etc) and to what extent it can support the proposed edition. For example, if the current system is operating 80% capacity –and arbitrary ceiling – then running another application could overload the system or required additional hardware. This involves financial consideration to accommodate technical enhancements. If the budget is a serious constraint, then the project is judged not feasible. The proposed system was found to be technically feasible in all aspects.

Economic Feasibility:

Economic analysis is the most frequently method used for evaluating the effectiveness of a candidate system. More commonly known as cost and benefit analysis, the procedure is to determine the benefits and saving that are expected from a candidate system and compare them with cost. If benefits outweigh costs then the decision is made to design and implement the system. The proposed system is economically feasible and it fulfills all the requirements of the existing with much more accuracy. Lots of paper work will be reduced and the time consumed will become minimum. In short it will over weigh the existing system in costs compared to benefits.

Behavioral Feasibility :

People are inherently resistant to change, and computers have been known to facilitate change. An estimate should be made of how strong a reaction the user staff is likely to have towards the development of computerized system.

Steps in feasibility analysis: -

It involves the eight steps:

1) Form a project team appoints a project leader.

- 2) Prepare system flowcharts.
- 3) Enumerate potential candidate system.
- 4) Describe the identity characteristics of candidate system.
- 5) Determine and evaluate performance and cost effectiveness of each candidate system.
- 6) Weigh system performance and cost data.
- 7) Select the best candidate system.
- 8) Prepare a report final project directive to management.

3.7-Cost and Benefit Analysis

Cost and Benefit Analysis is a procedure that gives a picture of the various costs, benefits and rules associated with a system. To what extent benefits outweigh costs is the function of cost/benefit analysis.

The various costs involved in this software are:

Hardware Cost: Various hardware requirements have already been mentioned and the hardware cost may be ascertained on that basis.

Personnel Cost: It requires only one person i.e. administrator to work on the system

Facility Cost: Since it is an online application, no facility costs are involved.

Operating Cost: No day-to-day costs are involved. However, the application needs to be registered which is a onetime expenditure and the domain name needs to be renewed on a yearly basis.

A system is also expected to provide benefits. The first task is to identify each benefit and then assign a monetary value to it. Benefits may be tangible or intangible, direct or indirect. Some of them are:

User friendly interface

Properly organized system

Proper work flow

Information will be well used

Chapter-4 PROJECT DESIGN

The most creative and challenging of the system life cycle is system design. The term design describes the final system and the process by which it is developed. It refers to the technical specifications that will be applied to implement the candidate system. It also includes the construction of programs and program testing. The key question is here: how should the problem be solved.

The physical design maps out the details of the physical system, plans the system implementation, devises a test and implementation plan, and specifies any new hardware and software. Design includes the construction of programs and program testing.

The first step is to determine how the output is to be produced and in what format. Second input data and master files (data bases) have to be design to meet the requirement of the proposed output. The operational phases are handled through program construction and testing, including a list of program needed to meet the system's objective and complete documentation. The final report prior to the implementation phase includes procedural flow charts, record lay out and workable plan for implementing the candidate system.

Various steps are involved in system design. These steps are as follows: -

- 1) Problem definition.**
- 2) Input – output specification.**
- 3) Database design.**
- 4) Modular program design.**
- 5) Preparation of source code.**
- 6) Testing and debugging**

Design the physical system

- a) Specify input output media.
- b) Design the database.
- c) Design physical information flow through the system and a physical walks through.

4.1-Data Flow Diagram

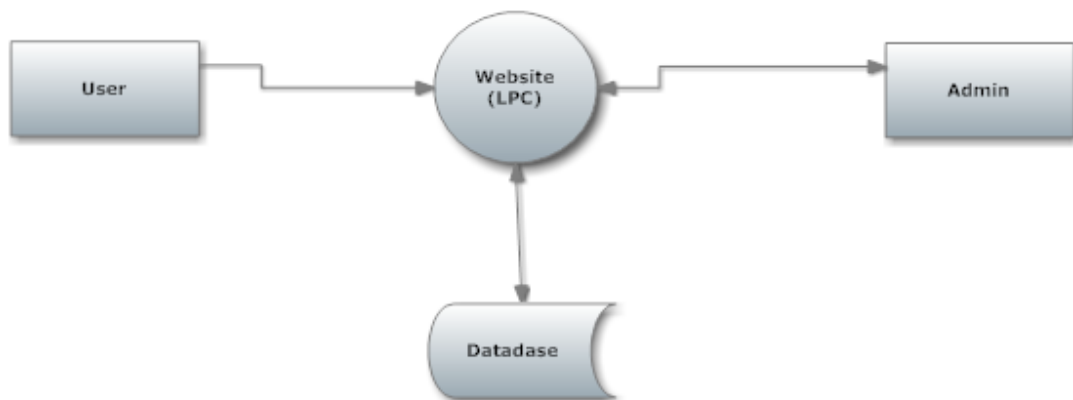


Fig 1: Design of application

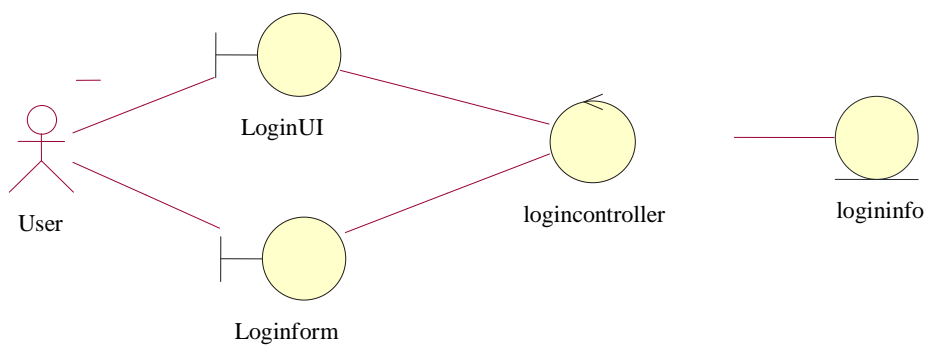


Fig 2: User interaction diagram

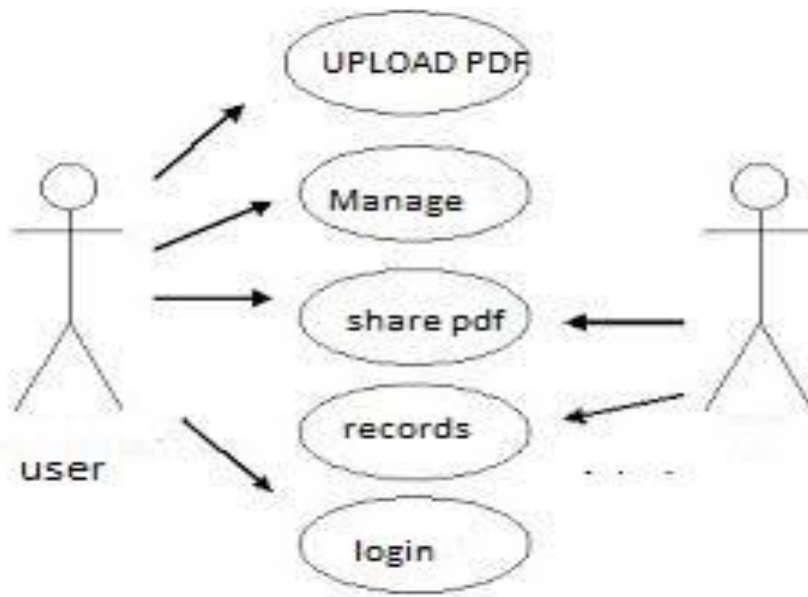


Fig 3: Use Case Diagram

Sequence Diagram:

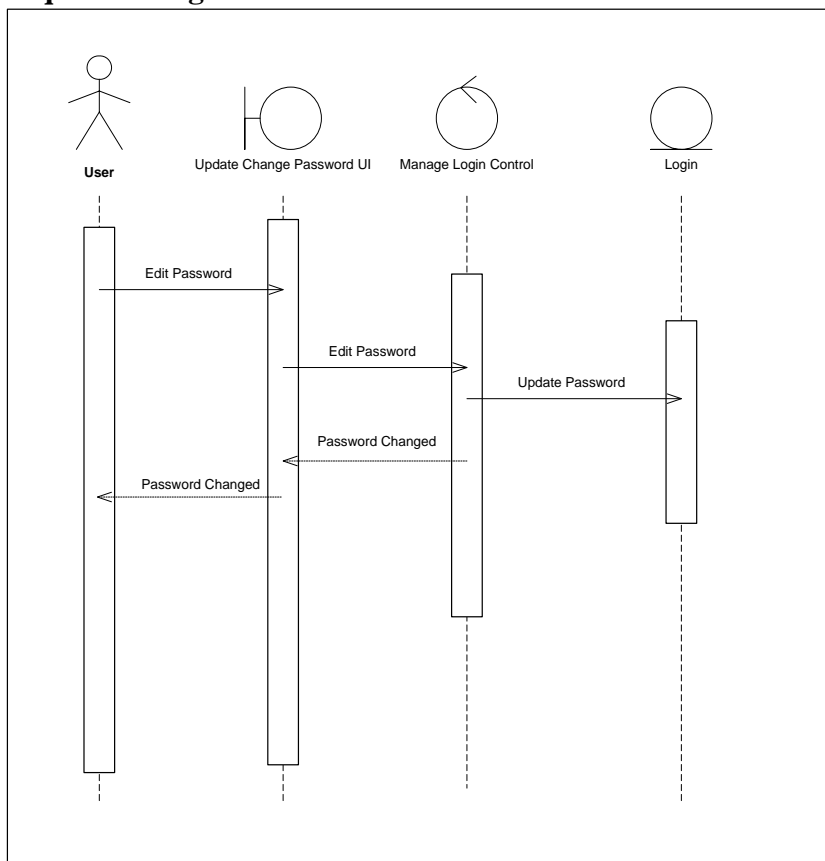


Fig 4: Sequence Diagram

Verification & Validation:

Validation refers to the process of using software in a live environment using real data. The process of validation refers to a set of activities that ensure that the software that has been built is matching to customer requirement. Validation is successful when software functions in a manner that can be reasonably expected by the customer.

Suitable validation checks have been put wherever need was felt so as to avoid wrong data input. Coding has been done so as to avoid wrong entries in the tables. For example Numeric characters are not allowed in the Employee's name. Various modules have different process logic, which involves sorting of data on different attributes and selection of required attribute depending upon conditions have been decided. Most of these are SQL queries.

The various kinds of validations performed in our system are as follows:

Date Validation: The validation on date data type has been specified to be of the format DD/MM/YY. Any other format is unacceptable.

Amount Validation: There is a validation on amount that is entered in rupees in the following format. "00,000,000.00" E.g. 15,65,789.00

From Date to Date: The "From Date" always has to be less than the "To Date", e.g. From 1 Nov, 2003 to 4 Nov, 2003 is correct and it cannot be other way round.

Number Field Validation: The field specified with Number as then their data-type will not accept Character or any other data type.

User Authentication: When a user logs on to the system to access data from the database, the password needs to be checked for user authentication.

Password change Validation: Only authorized users are allowed to change the password and the process requires asking the old password before changing it to the new one.

Security Features

Authentication is the process of identifying users. Authorization is the process of granting access to users based on identity. Together, authentication and authorization provide the means to keep your application secure from intruders.

The kind of security measures used in our System are as follows:

User Authentication:

Only the Valid users, who are present in the pre defined list of Users stored in User Master table are granted permissions to enter the System . Any user apart from Record of Users are denied access.

User Authorization:

A privilege is a right to execute a particular type of SQL statement or to access another user's object. Some examples of privileges include the right to:

Connect to the database (create a session)

Create a table

Select rows from another user's table

Execute another user's stored procedure

Privileges are granted to users so that they can accomplish tasks required for their jobs. A privilege should be granted only to a user who absolutely requires it to accomplish necessary task. Excessive granting can compromise security.

Table Level Security:

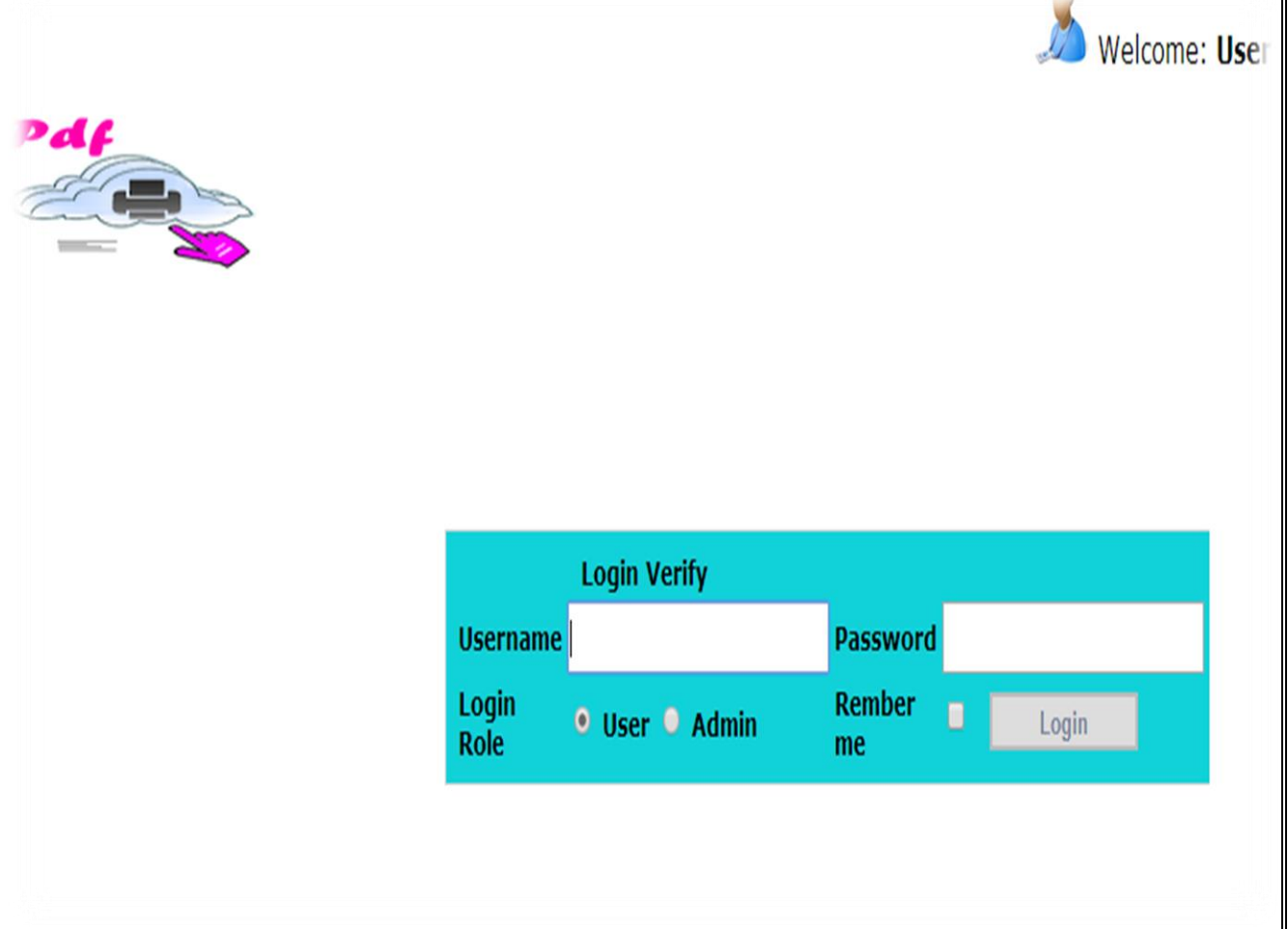
Schema object privileges for tables allow table security at the level of Data Manipulation Language and Data Definition Language operations.

We can grant privileges to use the various DML statements like DELETE, INSERT, SELECT, and UPDATE to perform operations on a table or view. We grant these privileges only to users and roles that need to query or manipulate a table's data.

With selective INSERT, a privileged user can insert a row with values for theselected columns. All other columns receive NULL or the column's default value. With selective UPDATE, a user can update only specific column values of a row. Selective INSERT and UPDATE privileges are used to restrict a user's access to sensitive data.

4.2SNAPSHOTS

Login Page :



Welcome: User

Pdf

Login Verify

Username Password

Login Role User Admin

Remember me

Login

Home Page :



Welcome: **admin**



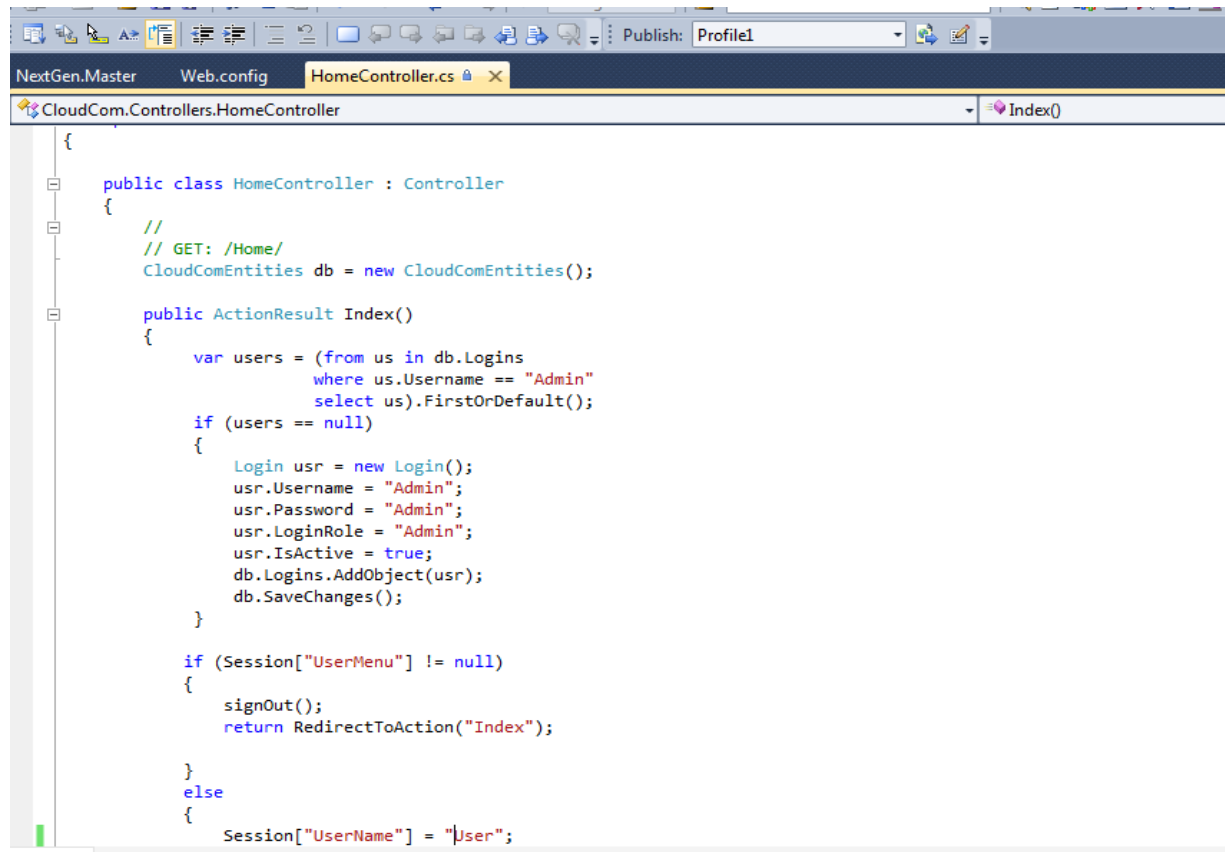
[HOME](#) [ADD USER](#) [DOCUMENTS](#) [SIGN OUT](#)

- admin
- beta
- all
- second
- second1



[Change Photo](#)

Source:



```
NextGen.Master Web.config HomeController.cs x
CloudCom.Controllers.HomeController Index()
{
    public class HomeController : Controller
    {
        //
        // GET: /Home/
        CloudComEntities db = new CloudComEntities();

        public ActionResult Index()
        {
            var users = (from us in db.Logins
                        where us.Username == "Admin"
                        select us).FirstOrDefault();
            if (users == null)
            {
                Login usr = new Login();
                usr.Username = "Admin";
                usr.Password = "Admin";
                usr.LoginRole = "Admin";
                usr.IsActive = true;
                db.Logins.AddObject(usr);
                db.SaveChanges();
            }

            if (Session["UserMenu"] != null)
            {
                signOut();
                return RedirectToAction("Index");
            }
            else
            {
                Session["UserName"] = "User";
            }
        }
    }
}
```

Document Menu:



Welcome: admin




[HOME](#) [ADD USER](#) [DOCUMENTS](#) [SIGN OUT](#)

| Filename | Date | Action |
|---|------------|--|
| file1 | 11/26/2013 | Download Share |
| | 11/26/2013 | Download Share |
| | 11/26/2013 | Download Share |
| 395853_388226587860096_364836220199133_1752163_1618053633_n.jpg | 11/26/2013 | Download Share |
| 24122011250.jpg | 11/26/2013 | Download Share |
| HOSPITAL MANAGEMENT system.docx | 11/26/2013 | Download Share |
| SW2014_SP1.0_fixed_SPRs.pdf | 5/12/2014 | Download Share |

[Upload New](#)

Add Document :



 Welcome: **beta**

[HOME](#) [DOCUMENTS](#) [SIGN OUT](#)

Upload

Filename Attachment No file chosen

Browse Document:

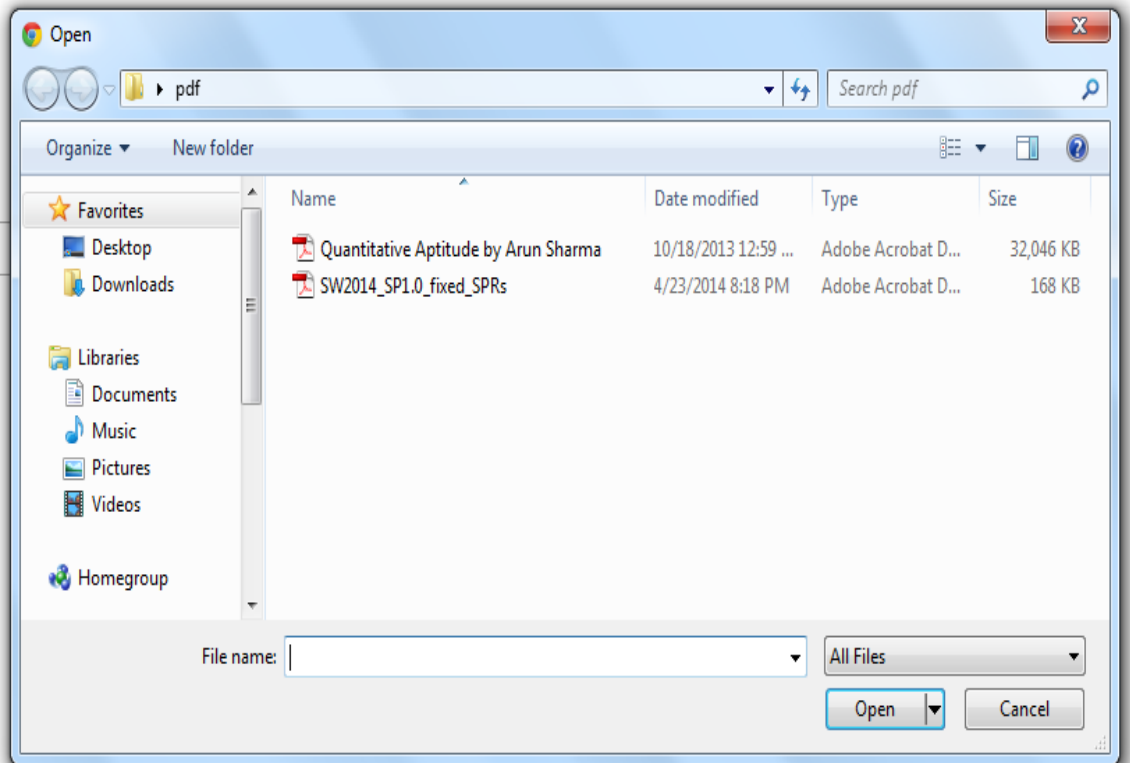
Welcome: **beta**



HOME DOCUMENTS SIGN OUT

Upload

Filename



Add New User:



Welcome: **admin**

[HOME](#) [ADD USER](#) [DOCUMENTS](#) [SIGN OUT](#)

Add user

Fill User Name
Please fill a unique user name.

This is the create user page. Here you can create the user and update there details.

On the left hand side you will enter the username and password. If the user already exist you can update there information at right hand side. Otherwise new user will create.

| | |
|------------|---------------------------------------|
| User Name | <input type="text"/> |
| Password | <input type="password"/> |
| Login Role | <input type="text" value="User"/> |
| | <input type="submit" value="Submit"/> |

User List:



- ◊ admin
- beta
- ◊ all
- ◊ second
- ◊ second1

Document Detail:

| Filename | Date |
|---|------------|
| file1 | 11/26/2013 |
| | 11/26/2013 |
| | 11/26/2013 |
| 395853_388226587860096_364836220199133_1752163_1618053633_n.jpg | 11/26/2013 |
| 24122011250.jpg | 11/26/2013 |
| HOSPITAL MANAGEMENT system.docx | 11/26/2013 |
| SW2014_SP1.0_fixed_SPRs.pdf | 5/12/2014 |

Create / Edit User Detail:

Add user

Fill User Name
Please fill a unique user name.

This is the create user page. Here you can create the user and update there details.

On the left hand side you will enter the username and password. If the user already exist you can update there information at right hand side. Otherwise new user will create.

User Name

Password

Login Role

Your user name is valid.
Now Fill your basic detail.

| | |
|-----------|---|
| Name | <input type="text" value="beta"/> |
| Address | <input type="text" value="delhi"/> |
| Zip | <input type="text" value="122346"/> |
| Phone | <input type="text" value="9988665577"/> |
| Email | <input type="text" value="absd@gmail.com"/> |
| Sex | <input type="text" value="Mail"/> |
| Education | <input type="text" value="B tech"/> |
| Country | <input type="text" value="India"/> |

Sign Out Menu :



Welcome: **beta**

[HOME](#) [DOCUMENTS](#) [SIGN OUT](#)

DATABASE

```

/***** Object: Table [dbo].[UsersToFiles]    Script Date: 12/01/2013 05:54:39 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[UsersToFiles](
    [UserID] [int] NOT NULL,
    [FileID] [int] NOT NULL,
    CONSTRAINT [PK_UsersToFiles] PRIMARY KEY CLUSTERED
(
    [FileID] ASC,
    [UserID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = O
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[roomdetails]    Script Date: 12/01/2013 05:54:39 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[roomdetails](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [ROOMNAME] [varchar](25) NULL,
    [RIN] [varchar](20) NULL,
    [MODERATOR] [varchar](25) NULL,
    CONSTRAINT [PK_roomdetails] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = O
) ON [PRIMARY]

```

Jquery Java Script Code:

```

    <%> %>

    Welcome: <b>
    <%if (Session["UserName"] != null)
    { %>
    <%=Session["UserName"].ToString()%>
    <%> %>
    </b>
    </div>
    </div>
    <div id="header">
    <div class="headerLogo">
    
    </div>
    <div class="headerMenu">
    <div id="menu">
    <%if (Session["UserMenu"] != null)
    { %>
    <%=Session["UserMenu"].ToString()%>
    <%> %>
    </div>
    </div>
    </div>
    <div id="conntent" class="conntentColorLess">

    <asp:ContentPlaceHolder ID="MainContent" runat="server">

    </asp:ContentPlaceHolder>
    </div>
    <div id="footer">
    <div id="footermenu">

```

4.3 Implementation code

Validation check query

```

function validateEmail(email) {
    var re = /^[^<>()[]\.\,\;\s@"]+(\.[^<>()[]\.\,\;\s@"]+)*|(\".+\")@((\[[0-9]{1,3}\. [0-9]{1,3}\. [0-9]{1,3}\. [0-9]{1,3}\])|(([a-zA-Z-0-9]+\.)+[a-zA-Z]{2,}))$/;
    return re.test(email);
}
function validateZipCode(elementValue) {
    var zipCodePattern = /\d{6}$|\d{5}-\d{4}$/;
    return zipCodePattern.test(elementValue);
}

```

```

}

function regIsDigit(fData)
{
    var reg = /^[0-9]+$/;
    return (reg.test(fData));
}

// checks that an input string is an integer, with an optional +/- sign character.

function isInteger (s) {
var isInteger_re = /^s*(\+|-)?\d+s*$/;
    return String(s).search (isInteger_re) != -1
}

function validateEmails(obj) {
    // $("#result").text("");
    var email = $(obj).val();
    if (validateEmail(email)) {
        // $("#result").text(email + " is valid :");
        $(obj).css("background-color", "green");

        return true;
    } else {
        // $("#result").text(email + "is not valid :(");
        $(obj).css("background-color", "red");
        $(obj).focus();

        return false;
    }
}

function validateRequired(obj) {
    // $("#result").text("");
    var email = $(obj).val();
    if (email.length>=1) {
        // $("#result").text(email + " is valid :");
        $(obj).css("background-color", "green");
        return true;
    } else {
        // $("#result").text(email + "is not valid :(");
        $(obj).css("background-color", "red");
        return false;
    }
}

```

```

}

function validateZip(obj) {
    // $("#result").text("");
    var zip = $(obj).val();
    if (validateZipCode(zip)) {
        // $("#result").text(email + " is valid :");
        $(obj).css("background-color", "green");
        return true;
    } else {
        // $("#result").text(email + "is not valid :(");
        $(obj).css("background-color", "red");
        return false;
    }
}
}

```

```

function validatePhone(obj) {
    // $("#result").text("");
    var zip = $(obj).val();
    if (regIsDigit(zip)) {
        // $("#result").text(email + " is valid :");
        $(obj).css("background-color", "green");
        return true;
    } else {
        // $("#result").text(email + "is not valid :(");
        $(obj).css("background-color", "red");
        return false;
    }
}
}

```

Database Code

```

CREATE DATABASE [CloudCom]
CONTAINMENT = NONE
ON PRIMARY
( NAME = N'CloudCom', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\CloudCom.mdf' , SIZE =
3136KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )
LOG ON
( NAME = N'CloudCom_log', FILENAME = N'C:\Program Files\Microsoft SQL
Server\MSSQL11.MSSQLSERVER\MSSQL\DATA\CloudCom_log.ldf' , SIZE =
784KB , MAXSIZE = 2048GB , FILEGROWTH = 10%)
GO
ALTER DATABASE [CloudCom] SET COMPATIBILITY_LEVEL = 110
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))

```

```

begin
EXEC [CloudCom].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [CloudCom] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [CloudCom] SET ANSI_NULLS OFF
GO
ALTER DATABASE [CloudCom] SET ANSI_PADDING OFF
GO
ALTER DATABASE [CloudCom] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [CloudCom] SET ARITHABORT OFF
GO
ALTER DATABASE [CloudCom] SET AUTO_CLOSE ON
GO
ALTER DATABASE [CloudCom] SET AUTO_CREATE_STATISTICS ON
GO
ALTER DATABASE [CloudCom] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [CloudCom] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [CloudCom] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [CloudCom] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [CloudCom] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [CloudCom] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [CloudCom] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [CloudCom] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [CloudCom] SET ENABLE_BROKER
GO
ALTER DATABASE [CloudCom] SET AUTO_UPDATE_STATISTICS_ASYNC
OFF
GO
ALTER DATABASE [CloudCom] SET DATE_CORRELATION_OPTIMIZATION
OFF
GO
ALTER DATABASE [CloudCom] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [CloudCom] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [CloudCom] SET PARAMETERIZATION SIMPLE

```

```

GO
ALTER DATABASE [CloudCom] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [CloudCom] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [CloudCom] SET RECOVERY SIMPLE
GO
ALTER DATABASE [CloudCom] SET MULTI_USER
GO
ALTER DATABASE [CloudCom] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [CloudCom] SET DB_CHAINING OFF
GO
ALTER DATABASE [CloudCom] SET FILESTREAM(
NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [CloudCom] SET TARGET_RECOVERY_TIME = 0
SECONDS
GO
USE [CloudCom]
GO
/***** Object: Table [dbo].[addressbook]   Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[addressbook](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [NAME] [varchar](40) NULL,
    [LoginID] [int] NOT NULL,
    [DOB] [varchar](14) NULL,
    [EMAIL] [varchar](40) NULL,
    [PHONE] [varchar](16) NULL,
    [CITY] [varchar](30) NULL,
    [OUSERID] [varchar](25) NULL,
    [Mobile] [varchar](50) NULL,
    [Address] [varchar](500) NULL,
    [Website] [varchar](100) NULL,
    [Company] [varchar](50) NULL,
    [Post] [varchar](50) NULL,
    [Photo] [varchar](500) NULL,
    CONSTRAINT [PK_addressbook] PRIMARY KEY CLUSTERED
(
    [ID] ASC

```



```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[advices] Script Date: 5/13/2014 12:16:39 PM *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[advices](  
    [NAME] [varchar](50) NULL,  
    [PHONE] [varchar](14) NULL,  
    [EMPID] [varchar](6) NULL,  
    [ADDRESS] [varchar](40) NULL,  
    [MESSAGE] [varchar](500) NULL  
) ON [PRIMARY]
```

```
GO
```

```
SET ANSI_PADDING OFF
```

```
GO
```

```
/***** Object: Table [dbo].[Chat] Script Date: 5/13/2014 12:16:39 PM *****/
```

```
SET ANSI_NULLS ON
```

```
GO
```

```
SET QUOTED_IDENTIFIER ON
```

```
GO
```

```
SET ANSI_PADDING ON
```

```
GO
```

```
CREATE TABLE [dbo].[Chat](  
    [ID] [int] IDENTITY(1,1) NOT NULL,  
    [Sender] [int] NOT NULL,  
    [Reciver] [int] NOT NULL,  
    [Message] [text] NULL,  
    [Time] [varchar](50) NULL,  
    [isRead] [bit] NOT NULL,  
    CONSTRAINT [PK_Chat] PRIMARY KEY CLUSTERED  
(  
    [ID] ASC
```

```
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,  
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,  
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]  
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

```

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[clientonline]  Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[clientonline](
    [LoginID] [int] NOT NULL,
    [ROOMID] [int] NOT NULL,
    [DATEOFLOG] [varchar](25) NULL,
    [TIMEOFLOG] [varchar](25) NULL
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Empdetails]  Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Empdetails](
    [EMPID] [int] IDENTITY(1,1) NOT NULL,
    [KEYS] [varchar](10) NULL,
    [AC_STATUS] [varchar](1) NULL,
    [DOJ] [varchar](10) NULL,
    [USERID] [varchar](25) NULL,
    CONSTRAINT [PK_empdetails] PRIMARY KEY CLUSTERED
(
    [EMPID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO

```

/***** Object: Table [dbo].[Files] Script Date: 5/13/2014 12:16:39 PM *****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

SET ANSI_PADDING ON

GO

```
CREATE TABLE [dbo].[Files](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Filename] [varchar](max) NULL,
    [Filepath] [varchar](max) NULL,
    [Date] [varchar](50) NULL,
    [UserID] [int] NULL,
    [IsProtected] [bit] NOT NULL,
    CONSTRAINT [PK_Files] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
```

GO

SET ANSI_PADDING OFF

GO

/***** Object: Table [dbo].[FilesToLogins] Script Date: 5/13/2014 12:16:39 PM

*****/

SET ANSI_NULLS ON

GO

SET QUOTED_IDENTIFIER ON

GO

```
CREATE TABLE [dbo].[FilesToLogins](
    [FileID] [int] NOT NULL,
    [UserID] [int] NOT NULL,
    CONSTRAINT [PK_FilesToLogins] PRIMARY KEY CLUSTERED
(
    [FileID] ASC,
    [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

/***** Object: Table [dbo].[GroupChat] Script Date: 5/13/2014 12:16:39 PM

*****/

SET ANSI_NULLS ON

```

GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[GroupChat](
    [Id] [int] IDENTITY(1,1) NOT NULL,
    [Message] [varchar](max) NULL,
    [SenderName] [varchar](50) NULL,
    [Time] [varchar](50) NULL,
    [sendTo] [int] NULL,
    CONSTRAINT [PK_GroupChat] PRIMARY KEY CLUSTERED
(
    [Id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Groups]    Script Date: 5/13/2014 12:16:39 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Groups](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Name] [varchar](50) NULL,
    [CreatedBy] [int] NULL,
    CONSTRAINT [PK_Groups] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Logins]    Script Date: 5/13/2014 12:16:39 PM *****/
SET ANSI_NULLS ON
GO

```

```

SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Logins](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [Username] [varchar](25) NOT NULL,
    [Password] [varchar](20) NULL,
    [IsActive] [bit] NOT NULL,
    [IsLogin] [bit] NOT NULL,
    [LoginRole] [varchar](50) NOT NULL,
    [UserPhoto] [varchar](500) NULL,
    CONSTRAINT [PK_password] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Mails]    Script Date: 5/13/2014 12:16:39 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Mails](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [FromID] [int] NOT NULL,
    [ToID] [int] NULL,
    [MESSAGE] [text] NULL,
    [TIMING] [varchar](50) NULL,
    [READ] [bit] NOT NULL,
    [SUBJECT] [varchar](25) NULL,
    [Attachment] [text] NOT NULL,
    CONSTRAINT [PK_mail] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

```

```

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Members]   Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Members](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [GroupID] [int] NULL,
    [MemberID] [int] NULL,
    CONSTRAINT [PK_Members] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
/***** Object: Table [dbo].[roomdetails]   Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[roomdetails](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [ROOMNAME] [varchar](25) NULL,
    [RIN] [varchar](20) NULL,
    [MODERATOR] [varchar](25) NULL,
    CONSTRAINT [PK_roomdetails] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Security]   Script Date: 5/13/2014 12:16:39 PM *****/

```

```

SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[Security](
    [LoginID] [int] NOT NULL,
    [QUES] [varchar](100) NULL,
    [ANS] [varchar](50) NULL
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[ShareFiles]  Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[ShareFiles](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [FileID] [int] NOT NULL,
    [ToShareID] [int] NOT NULL,
    [UploaderID] [int] NOT NULL,
    [ShareBy] [int] NOT NULL,
    [Date] [varchar](50) NULL,
    CONSTRAINT [PK_ShareFiles] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[Table_1]  Script Date: 5/13/2014 12:16:39 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON

```

```

GO
CREATE TABLE [dbo].[Table_1](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [FileName] [varchar](max) NULL,
    [Path] [varchar](max) NULL,
    [UploadID] [int] NULL,
    CONSTRAINT [PK_Table_1] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[UsersDetail]    Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
SET ANSI_PADDING ON
GO
CREATE TABLE [dbo].[UsersDetail](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [NAME] [varchar](60) NULL,
    [ADDRESS] [varchar](120) NULL,
    [ZIP] [varchar](6) NULL,
    [PHONE] [varchar](12) NULL,
    [EMAIL] [varchar](50) NULL,
    [SEX] [varchar](10) NULL,
    [EDUCATION] [varchar](30) NULL,
    [COUNTRY] [varchar](25) NULL,
    [CITY] [varchar](20) NULL,
    [STATE] [varchar](25) NULL,
    [DOB] [varchar](12) NULL,
    [LoginID] [int] NOT NULL,
    CONSTRAINT [PK_userdetails] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

```



```

GO
SET ANSI_PADDING OFF
GO
/***** Object: Table [dbo].[UsersToFiles]  Script Date: 5/13/2014 12:16:39 PM
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[UsersToFiles](
    [UserID] [int] NOT NULL,
    [FileID] [int] NOT NULL,
    CONSTRAINT [PK_UsersToFiles] PRIMARY KEY CLUSTERED
(
    [FileID] ASC,
    [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
ALTER TABLE [dbo].[Chat] ADD CONSTRAINT [DF_Chat_isRead] DEFAULT
((0)) FOR [isRead]
GO
ALTER TABLE [dbo].[Files] ADD CONSTRAINT [DF_Files_IsProtected]
DEFAULT ((0)) FOR [IsProtected]
GO
ALTER TABLE [dbo].[Logins] ADD CONSTRAINT [DF_password_IsActive]
DEFAULT ((1)) FOR [IsActive]
GO
ALTER TABLE [dbo].[Logins] ADD CONSTRAINT [DF_Logins_IsLogin]
DEFAULT ((0)) FOR [IsLogin]
GO
ALTER TABLE [dbo].[Logins] ADD CONSTRAINT [DF_Logins_LoginRole]
DEFAULT ('User') FOR [LoginRole]
GO
ALTER TABLE [dbo].[Mails] ADD CONSTRAINT [DF_mail_READ] DEFAULT
((0)) FOR [READ]
GO
ALTER TABLE [dbo].[addressbook] WITH CHECK ADD CONSTRAINT
[FK_addressbook_Logins] FOREIGN KEY([LoginID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[addressbook] CHECK CONSTRAINT
[FK_addressbook_Logins]
GO

```

```

ALTER TABLE [dbo].[Chat] WITH CHECK ADD CONSTRAINT
[FK_Chat_Logins] FOREIGN KEY([Sender])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Chat] CHECK CONSTRAINT [FK_Chat_Logins]
GO
ALTER TABLE [dbo].[Chat] WITH CHECK ADD CONSTRAINT
[FK_Chat_Logins1] FOREIGN KEY([Reciver])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Chat] CHECK CONSTRAINT [FK_Chat_Logins1]
GO
ALTER TABLE [dbo].[clientonline] WITH CHECK ADD CONSTRAINT
[FK_clientonline_Logins] FOREIGN KEY([LoginID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[clientonline] CHECK CONSTRAINT
[FK_clientonline_Logins]
GO
ALTER TABLE [dbo].[clientonline] WITH CHECK ADD CONSTRAINT
[FK_clientonline_roomdetails] FOREIGN KEY([ROOMID])
REFERENCES [dbo].[roomdetails] ([ID])
GO
ALTER TABLE [dbo].[clientonline] CHECK CONSTRAINT
[FK_clientonline_roomdetails]
GO
ALTER TABLE [dbo].[Files] WITH CHECK ADD CONSTRAINT
[FK_Files_Logins] FOREIGN KEY([UserID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Files] CHECK CONSTRAINT [FK_Files_Logins]
GO
ALTER TABLE [dbo].[FilesToLogins] WITH CHECK ADD CONSTRAINT
[FK_FilesToLogins_Files] FOREIGN KEY([FileID])
REFERENCES [dbo].[Files] ([ID])
GO
ALTER TABLE [dbo].[FilesToLogins] CHECK CONSTRAINT
[FK_FilesToLogins_Files]
GO
ALTER TABLE [dbo].[FilesToLogins] WITH CHECK ADD CONSTRAINT
[FK_FilesToLogins_Logins] FOREIGN KEY([UserID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[FilesToLogins] CHECK CONSTRAINT
[FK_FilesToLogins_Logins]
GO

```

```

ALTER TABLE [dbo].[Groups] WITH CHECK ADD CONSTRAINT
[FK_Groups_Logins] FOREIGN KEY([CreatedBy])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Groups] CHECK CONSTRAINT [FK_Groups_Logins]
GO
ALTER TABLE [dbo].[Mails] WITH CHECK ADD CONSTRAINT
[FK_Mails_Logins] FOREIGN KEY([FromID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Mails] CHECK CONSTRAINT [FK_Mails_Logins]
GO
ALTER TABLE [dbo].[Mails] WITH CHECK ADD CONSTRAINT
[FK_Mails_Logins1] FOREIGN KEY([ToID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Mails] CHECK CONSTRAINT [FK_Mails_Logins1]
GO
ALTER TABLE [dbo].[Security] WITH CHECK ADD CONSTRAINT
[FK_security_Logins] FOREIGN KEY([LoginID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Security] CHECK CONSTRAINT [FK_security_Logins]
GO
ALTER TABLE [dbo].[ShareFiles] WITH CHECK ADD CONSTRAINT
[FK_ShareFiles_Files] FOREIGN KEY([FileID])
REFERENCES [dbo].[Files] ([ID])
GO
ALTER TABLE [dbo].[ShareFiles] CHECK CONSTRAINT [FK_ShareFiles_Files]
GO
ALTER TABLE [dbo].[ShareFiles] WITH CHECK ADD CONSTRAINT
[FK_ShareFiles_Logins] FOREIGN KEY([ToShareID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[ShareFiles] CHECK CONSTRAINT
[FK_ShareFiles_Logins]
GO
ALTER TABLE [dbo].[ShareFiles] WITH CHECK ADD CONSTRAINT
[FK_ShareFiles_Logins1] FOREIGN KEY([UploaderID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[ShareFiles] CHECK CONSTRAINT
[FK_ShareFiles_Logins1]
GO
ALTER TABLE [dbo].[ShareFiles] WITH CHECK ADD CONSTRAINT
[FK_ShareFiles_Logins2] FOREIGN KEY([ShareBy])
REFERENCES [dbo].[Logins] ([ID])

```

```

GO
ALTER TABLE [dbo].[ShareFiles] CHECK CONSTRAINT
[FK_ShareFiles_Logins2]
GO
ALTER TABLE [dbo].[Table_1] WITH CHECK ADD CONSTRAINT
[FK_Table_1_Logins] FOREIGN KEY([UploadID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[Table_1] CHECK CONSTRAINT [FK_Table_1_Logins]
GO
ALTER TABLE [dbo].[UsersDetail] WITH CHECK ADD CONSTRAINT
[FK_UsersDetail_Logins] FOREIGN KEY([LoginID])
REFERENCES [dbo].[Logins] ([ID])
GO
ALTER TABLE [dbo].[UsersDetail] CHECK CONSTRAINT
[FK_UsersDetail_Logins]
GO
USE [master]
GO
ALTER DATABASE [CloudCom] SET READ_WRITE
GO

```

4.4-Testing Methodology Used

As already mentioned, the whole development process is object oriented and implementation is also done in object oriented environment i.e. asp.net. So, anticipated testing technique automatically implies to be object oriented. In object oriented approach, the semantic constructs appear at all the stages, so it is better to find the problem in early stages so that we start testing parallel to development process and will be taken in consideration at each stage.

Grossly, the testing strategy is divided in the following steps:

Step 1: Object oriented analysis and object oriented design models were checked for consistency.

Step 2: Unit testing was exercised in context to object orientation in which each class with their methods and operations were scrutinized.

Step 3: Integration testing at various stages of testing was exercised to find associative errors. Regression testing was utilized but at the same time, it was also based on object oriented view. Specifically, Cluster testing is a good choice.

Step 4: Finally, termination of testing was supported by Validation testing and System testing which includes requirements review, stress testing, security testing, recovery testing etc.

These different levels of testing attempt to detect different types of faults. The relation of the faults introduced in different phases, and the different levels of testing are shown:

1)UNIT TESTING:

The first level of testing is unit testing. In this different modules are tested against the specifications produced during design for the modules. Unit testing is essential for verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules.

2)INTEGRATION TESTING:

The next level of testing is often called integration testing. In this many tested modules are combined into sub-systems, which are then tested .the goal here is to see if the modules can be integrated properly, the emphasis being on testing interfaces between modules. This activity can be considered as testing the design, and hence the emphasis on testing module interactions.

3) SYSTEM TESTING:

The next level of testing is system testing. Here the entire software system is tested. The reference document for this process is requirement document, and the goal is to see if the software meets its requirements. This is essentially a validation exercise.

4.5-System Implementation

Implementation implies what configurations are required to implement the given software at the end user's site.

There must be .Net Framework 2.0 loaded on all the computers that will provide support to the computer on which the application is running. Then, install the installation on the system using its installer.

There must be SQL Server 2005 installed on the system to provide the support of database.

To deploy the application on cloud there must be microsoft azure installed.

Layers Of Project

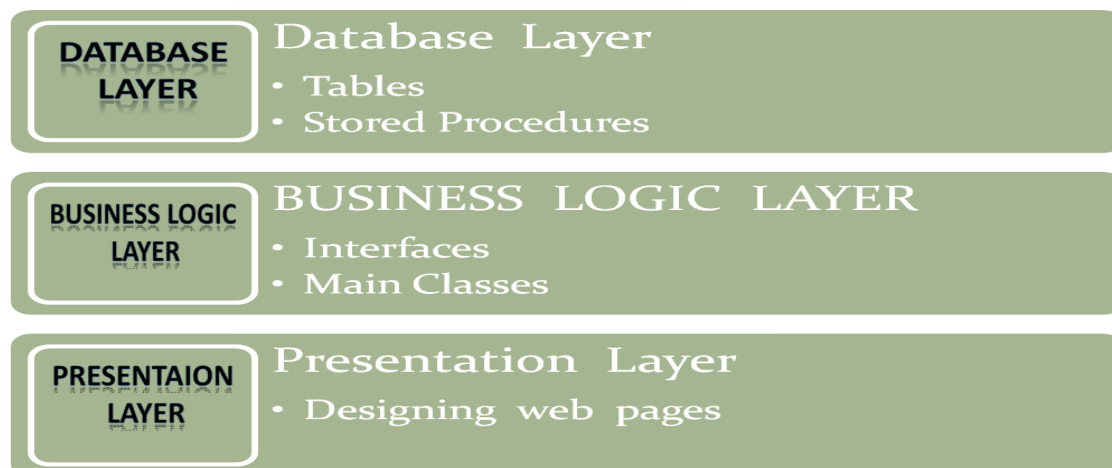


Fig 5:Layers of Architecture

Our project includes two panels:

Admin Panel

User Panel

Database Layer:

It includes:

Tables

Stored Procedures

Data Flow diagram

Number of modules and their description:

According to the requirement login is provided to Limited users and administrator. This module is integrated with Restriction of unauthorized access. Each user can only access those data, which are required for his/her work.

Records:- This module is for checking all types of records related to the registered users.

Administrator:- This module has only access to the administrator.the administrator only can check this module and make the required modifications as per requirements and he can add and delete the users.

Document uploader:-This module is designed to upload documents for easy access by the user.

Login & Register module:-New users can register on the application and login to upload ,view and share the document

Chapter-5
CONCLUSION & FUTURE SCOPE

DEGREE OF SUCCESS:

All the requirements given in the specification are met.
The project is completed within specified time and budget.

EXPERIENCE GAINED:

Got knowledge of new language asp.net with c#.
Got experience of developing design documents in Smart Draw for the first time.
Learnt the importance of *ADVANCED OBJECT ORIENTED METHODOLOGIES* and about various elements of it.
Got the experience on cloud computing project.

LIMITATIONS:

Currently supports only pdf format ,the application needs to be extended for other formats of documents also

ENHANCEMENTS:

- 1)In future system can conduct an online document creation doc , pdf etc.
- 2)User can create a document that can simultaneously available to other user with it is shared and all can edit in real time.
- 3)Edit pdf page.
- 4)Open pdf file.

Chapter 6
REFERENCES

- 1) R. Hunter, The why of cloud, <http://www.gartner.com/DisplayDocument?doc cd=226469&ref= g noreg>, 2012.
- 2) M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali, Cloud computing: Distributed internet computing for IT and scientific research, *Internet Computing*, vol.13, no.5, pp.10-13, Sept.-Oct. 2009.
- 3) P. Mell and T. Grance, The NIST definition of cloud computing, <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>, 2012.
- 4) Microsoft Academic Research, Cloud computing, <http://libra.msra.cn/Keyword/6051/cloudcomputing?query=cloud%20computing>, 2012.
- 5) Google Trends, Cloud computing, <http://www.google.com/trends/explor#q=c loud%20computing>, 2012.
- 6) N. G. Shivaratri, P. Krueger, and M. Singhal, Load distributing for locally distributed systems, *Computer* vol. 25, no. 12, pp. 33-44, Dec. 1992.