

# Design and Implementation of File Sharing Server

Firas Abdullah Thweny Al-Saedi <sup>#1</sup>, Zaiab Dheya'a Al-Taweel <sup>\*2</sup>

<sup>#1,2</sup> Computer Engineering Department, Al-Nahrain University, Baghdad, Iraq

**Abstract** — this paper discusses building of a file sharing server which enables the process of copying files from web server to clients' computers through online internet connection. This is implemented by creating a web application and hosting it on the local server. Within this web application, the users can upload and download files of different types like videos, audios, books. ASP.NET technology with Microsoft Visual Studio platform was used to create the web application. The server was tested in a real environment and its performance was reliable.

**Keywords** — File Sharing Server, Content Server.

## I. INTRODUCTION

Today there is a real problem that users need to download files of different types, this process take a lot of the bandwidth of internet available for the users to get access to remote servers; this will make the internet service slow. A good solution for this problem is to implement a file sharing system within a local network which will reduce request time, meaning that files on the server are accessed easily and quickly. So, this will result in increasing the internet bandwidth available for the user.

Before moving into the subject, it must known what client-server architecture is ?. This term refers to two processes or two applications which will be communicating with each other to exchange some information. One of the two processes acts as a client process and the other acts as a server.

The client process is the processes which make a request for information. After getting the response, this process may terminate or may perform some other processing. For example: Internet browser works as a client application which sends a request to web server, where the web application resides to get one HTML web page. The server process is the processes which take a request from the clients, performs required processing on the request, gathers required information and sends it back to the requestor client. Once done, it becomes ready to serve another client.

That is known as 2-tier client server architecture, but the most commonly used architecture with the web application is the 3-tier client server database architecture that is shown in Fig. 1 which include one additional tier, the database tier which interacts with persistent data usually stored in a database server [1].

In other words, the clients use browsers to interact with the web application pages which are graphical user interface (GUI) that display information about the uploaded files in an acceptable form for the users. These uploaded files are stored in the web server while the information about these files in addition to users' information are stored in the database server.

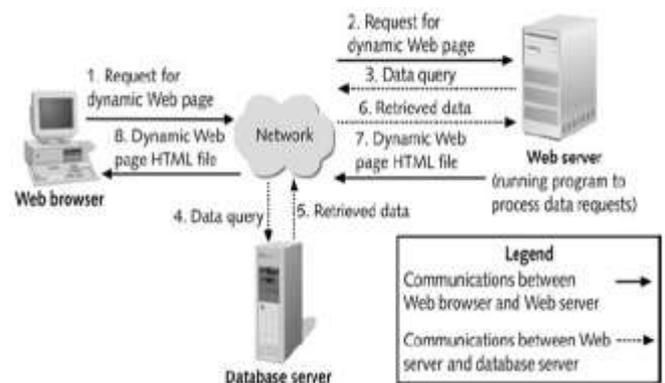


Fig. 1 The 3- tier architecture

## II. WEB APPLICATION LAYERS AND IMPLEMENTATION REQUIREMENTS

The 3-tier architecture web application, as mentioned previously, contains three distinct layers which are presentation (client side), business (server side), and data layers (database).

Microsoft visual studio platform was used to work with ASP.NET, it include all the tools that are required to create ASP.NET Web application including Visual Web Developer Express, Internet Information Services (IIS), SQL Server Express, and the .NET Framework [2][3].

Here is a brief description of each layer and the tool that was used to implement it.

- A. Business Layer (Server-side code): In ASP.NET, server-side code uses the .NET Framework and it is written using C# language [4]. Server-side processing is used to interact with permanent storage like databases or files. The server will also render pages to the client and process user input. Server-side processing happens when a page is first requested and when pages are posted back to the server.

Examples of server-side processing are user validation, saving and retrieving data, and navigating to other pages.

- B. Presentation Layer (Client-side code): In contrast to server-side code, client-side scripts are embedded on the client's web page and processed on the client's internet browser. Client-side scripts are written in some type of scripting language like JavaScript and interact directly with the page's HTML elements like text boxes, buttons, list-boxes and tables. HTML and CSS (cascading style sheets) are also used in the client side code [5][6]. In order that the client-side code could work, the client's internet browser must support these languages.
- C. Data Layer (Database): Database tables are implemented using SQL Server Compact which is a database engine that facilitates ASP.NET Web development and deployment. The database is created using SQL which is the standard language for accessing and manipulating databases. SQL consists of a data definition language and a data manipulation language. The scope of SQL includes data insert, query, update and delete, schema creation and modification, and data access control.

After designing the database and building the web pages and in order to run the web application, it must be hosted on a web server, and in this case the web server is the Internet Information Services (IIS) which provides a secure, easy-to-manage, modular and extensible platform for reliably hosting websites, services and applications. With IIS, choice and control are achieved without giving up reliability or security, and in order to connect the database with the web pages ADO.NET technology is used.

ADO.NET (ActiveX Data Objects) is a set of classes that expose data access services to the .NET programmer. ADO.NET provides consistent access to data sources such as Microsoft SQL Server [7].

### III. WEB APPLICATION ANALYSIS

Within this web application, anyone can browse it and download any file he/she wants, a client also can create an account with a username and password and then he/she will be able to upload files in addition to downloading them.

The admin of the web application verifies the files before they are made available to the user, the admin checks if the file is acceptable or not and then he decides to enable the file or not.

Fig. 2 shows a simple flowchart to illustrate the main function of the web application which is file upload.

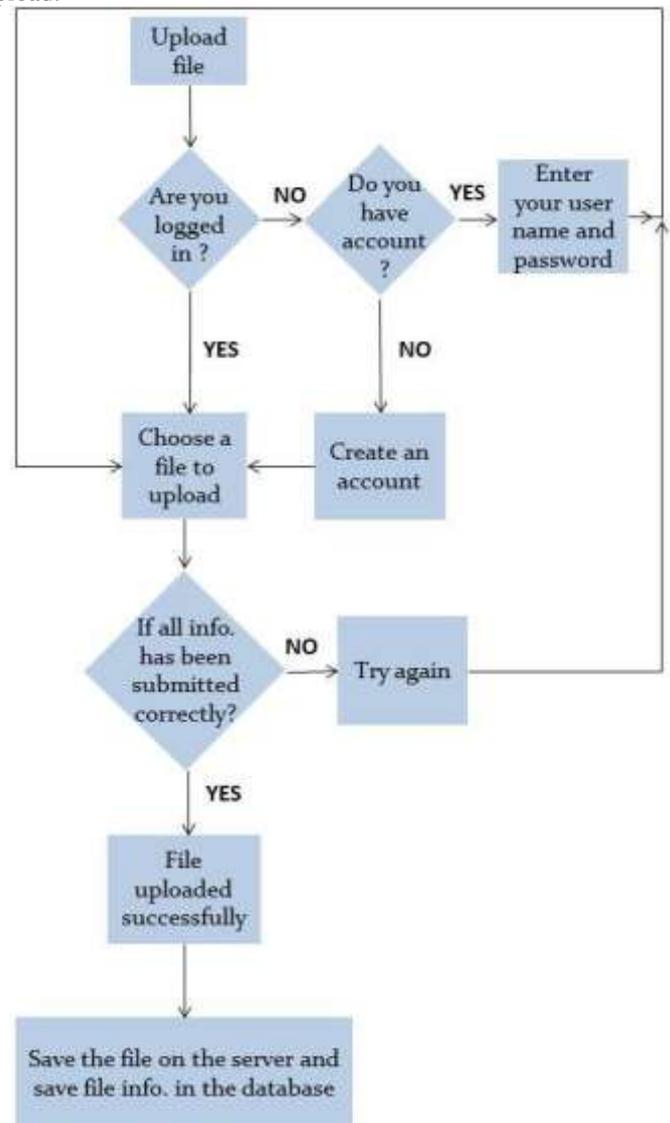


Fig. 2 File upload function flowchart

Here is a description of the three privilege levels

- A. Anonymous Users: This web application is made accessible by anonymous users (any one can get access to view the content of the web application) within this level, a user can view and download any file, but he is not allowed to upload files or leave comments.
- B. Registered Level: Any one can register within the web application and create an account with user name and a password, and then he/she can log in to the web application with his/her username and password. A user within this level can download, upload files and leave comments on other files.

C. The Administration Level: The admin of the web application has a verification role. When a user upload a file it will not made available to other users at once, before that the admin must check the file, and if it is acceptable then the admin enable the file on the web application and it will be available to the users. The verification function of the admin is shown as flowchart in Fig. 3.

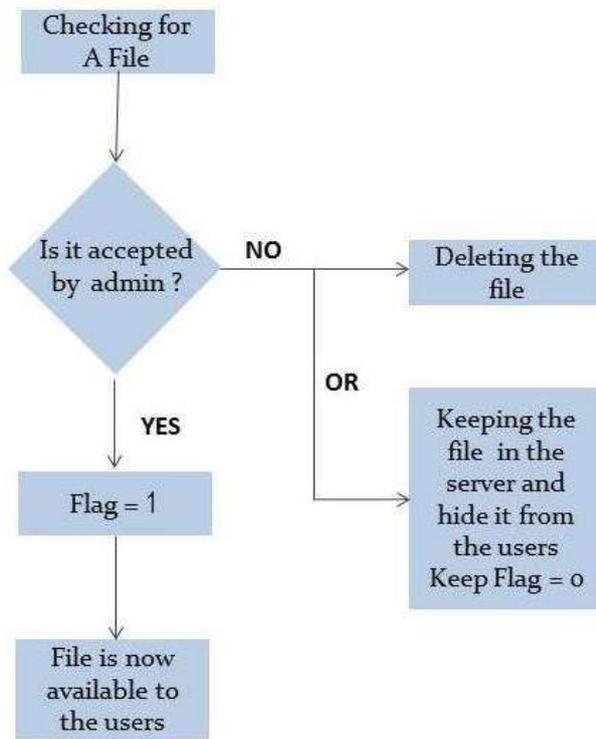


Fig. 3 Admin verification function flowchart

- 1) Application: Membership store user information uniquely for each application. This enables multiple ASP.NET applications to use the same data source without running into a conflict if duplicate user names are created
- 2) Membership: Stores membership data. Each record in this table corresponds to one membership user.
- 3) Role: Stores the roles data. Each record in Roles table corresponds to one role.
- 4) UserInRole: It is used to map roles to users.
- 5) User: Stores data about users.

The relationship between these tables is show in Fig.

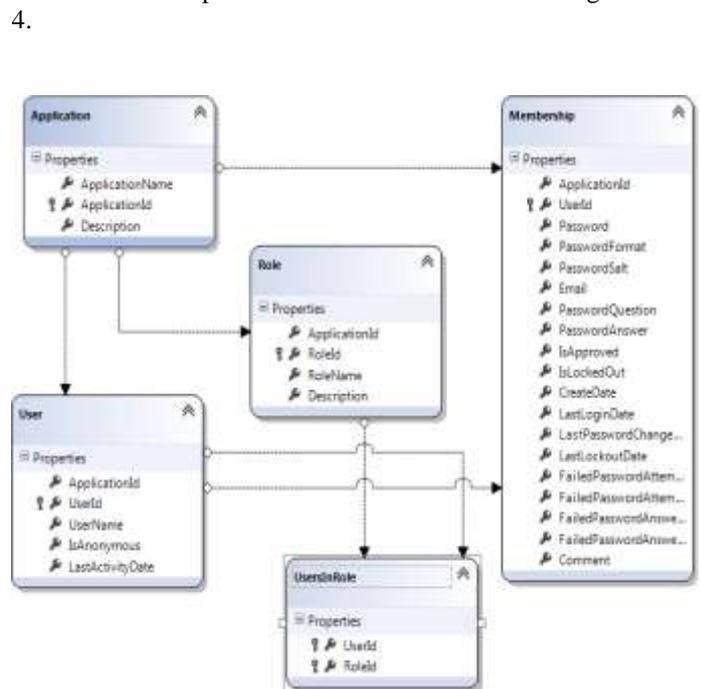


Fig. 4 Membership tables and relationships between them

#### IV. WEB APPLICATION DESIGN

In order to implement the web application, there are two steps, the first step is to design the database (represented by the data layer) which includes tables of information about users and uploaded files, and the second step is to design the web pages which is represented by both the server side and the client side.

##### A. Database Design:

Database includes two sections, the first section is the membership which includes tables and contains information about registered users. ASP.NET membership gives a built-in way to validate and store user credentials [7]. ASP.NET membership therefore helps in managing user authentication in the Web sites. ASP.NET membership can be used with ASP.NET forms authentication by using with the ASP.NET login controls to create a complete system for authenticating users. The membership tables are:

The other section presents the main tables of the web application that includes all other tables that stores data about files uploaded on the web application. These tables are:

- 1) Categories Table: Includes all files category in the web application.
- 2) SubCategories: Includes all files subcategories that are related to the categories table by a foreign key.

The relationship between these two tables is shown in Fig. 5.

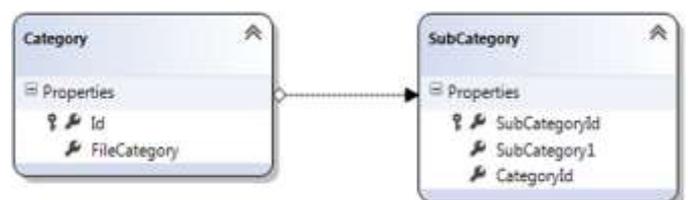


Fig.5 Relationship between Categories and SubCategories tables

- 3) UploadedFiles Table: Stores all the information of files that were uploaded by users.
- 4) Likes Table: Users can like the uploaded files (something like that on Facebook), and this process required a table to store information about each file and the number of likes related for it.
- 5) Comments Table: Users can add comments on the uploaded files, this process also required additional table to store the comments.
- 6) FavoriteFiles Table: A user can add any file he wants to his favorite files, and this will be added to his page on the web application.

The relationships between the above tables are explained in Fig. 6.

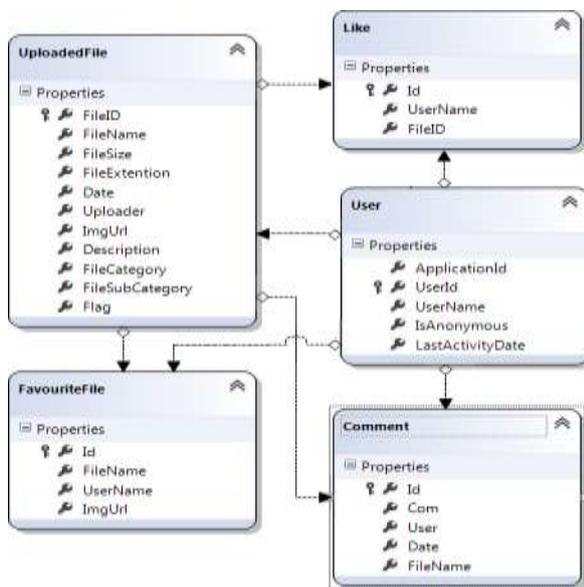


Fig.6 Relationships between the main tables

#### B. Web Pages Design:

The pages were designed using ASP.NET technology with C# as code behind file language, HTML5 as client side programming language and CSS for layout formatting as mentioned previously in this paper.

The web application pages are:

- 1) The Start Page: Is the first page, that appears when a user runs the web application as shown in Fig. 7.
- 2) Register Page: If the user is browsing the web application for the first time and he don't have an account yet, then he/she can create an account with a username and password by clicking on Register at the right top corner of the page. In this page the user can enter his/her information including username, e-mail and password. These information are then stored in the database tables, and the user account is

created. Then the user can use that information to log into the web application.



Fig. 7 The start page

- 3) Log in Page: If the user has an account then he can log in to the web application by clicking on Log In at the left top corner and entering his user name and password. After logging in the user will be directed to the home page.
- 4) Home Page: This page presents all the uploaded files sorted by date of uploading as shown in Fig. 8. Even if the user doesn't have an account he/she is still able to view this page.

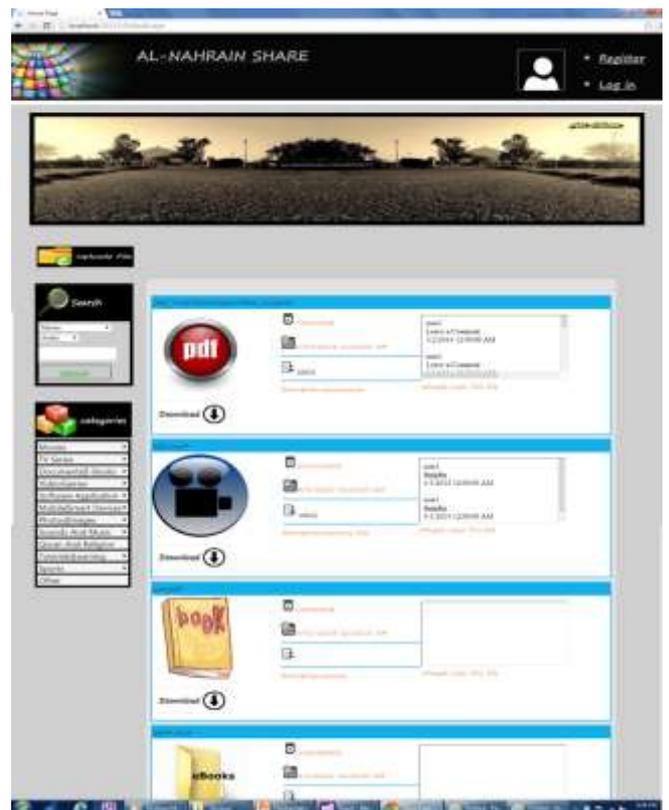


Fig. 8 The home page

The user can upload file from his/her computer to the web application by clicking on the upload file button which will redirect the user to the file upload page as shown in Fig. 9. Unauthenticated

(anonymous users) are not allowed to navigate to this page.

- 5) File Upload Page: Within this page, the user can upload a file and specify the file category and subcategory by selecting them from the dropdown lists, in addition to that, the user must upload an image for the file, he/she also can add a description for that file. As the user clicks on upload file button, the file will be saved in a file store called pending files store on the server and all the information of that file will be stored on the database tables. The file now is not available for the users; the admin must verify the file in order to be available on the web application. And for this purpose the admin must login with his username and password to verify the pending files.



Fig. 9 File upload page

- 6) Admin Verification Page: In this page all pending files are displayed, the admin can download the file, check whether it's acceptable or not, If it's acceptable the admin will click on enable button, if not he will click on delete button. In addition to that, the admin can update the file information by changing the file name, category and description.

A table has been implemented for each file, and it is arranged in such a manner that displays the file image in addition to the file name, size, the username who uploads the file and the date of uploading. By clicking on the file name, the user will be directed to the file details page and by clicking on the download button that is appeared

under the file image, the file will be downloaded on the client computer. Also there is the comment table on the right side of each file table which includes the comment of that file. Under this table, there is a label that tells the number of likes for that file, a text box for the user to enter a comment, in addition to three buttons that are: leave a comment, like and add to my favorite. If the user doesn't have an account he doesn't able to like a file or leave a comment, so these three buttons and the text box are not displayed for the anonymous users.

Another interesting feature is that the video is displayed on the page itself, this is implemented with the video tag of HTML5 that provides all video controls like stop and continue, sound control and full screen options.

The web application includes a web page for each sub category that presents all the files that are classified under that sub category. The menu list that appears on the left side of the pages presents the main categories of the uploaded files. By moving the mouse over any one of them a new sub list will appear, that presents the sub categories within that category. By clicking on the items of the menu list, the user will be directed to the page of the selected item.

The search engine is a very important part of the web application; it appears on the right side of the page. Where the user can select the type of files he wants to search from the dropdown lists, or by default, the search process will be done on the whole files then the user can enter the filename he wants to find within the web application.

## V. CONCLUSIONS

This paper discussed the implementation of a file sharing server in the form of a web application to get a simple, easy, and fast method for the users to download the most requested files locally through the Internet. This paper also discussed the technologies and programming tools used for the project design which are ASP.NET and Microsoft Visual Studio respectively.

## REFERENCES

- [1] <http://www.seguetech.com/blog/2013/05/01/client-side-server-side-code-difference>
- [2] Pro ASP.NET 4.5 in C#, Fifth Edition, Published by Adam Freeman.
- [3] <http://www.asp.net/get-started/websites>
- [4] The C# Programming Language, Fourth Edition, Anders Hejlsberg Mads Torgersen, Scott Wiltamuth ,Peter Golde
- [5] HTML, XHTML, and CSS Bible, Fifth Edition, Published by Wiley Publishing, Inc.
- [6] Beginning CSS: Cascading Style Sheets for Web Design, Second Edition Published by Wiley Publishing, Inc.
- [7] <http://msdn.microsoft.com/enus/library/vstudio/>