



# AUTHENTICATED WEB MANAGEMENT SYSTEM

**Masters Project Report (CPEG 597)**  
December 2005

*Submitted to*  
Prof. Ausif Mahmood

*By*  
Kavya P Basa

**ID. 655795**

# Abstract

In an era where web development is taking priority over other areas of development, the management of web based resources is still done in old ways, i.e., by closing the site down and making changes and uploading the changes back to the web server. There is a need for a more versatile management of the resource wherein the changes can be done to the content over the site from its original location over the web server.

This project is based on an application that provides extended support for maintenance of the web management from its hosted environment. Web Manager is an ASP.NET Server Web Management System. The application is controlled by two users based upon the access levels allotted to them. The admin user can do every task with the application. Every user is specified with a minimum of one home directory (The application on which the person has the privilege to work with). Using Web Manager, only the sites which are authorized to a user can be managed.

The web pages, pages related content can be newly created as well as modifications can be done on the existing documents, which will reflect a change on the outlook of the site. A single user can have authority to manage number of sites but only the site which is set as a home directory can be managed at a given moment of time. It also provides the features of uploading and downloading the files needed for management.

# **CONTENTS**

**1. INTRODUCTION**

**2. PROJECT ANALYSIS**

**3. SELECTED SOFTWARE**

**4. SOFTWARE REQUIREMENT**

**5. PROJECT DESIGN**

**6. OUTPUT SCREENS**

**7. CONCLUSION**

**8. REFERENCES**

## INTRODUCTION

In an era where having a website of our own is a matter of prestige for some and necessity for others. Management of these websites is still an area which is ignored or done in traditional manner. In a changing world no corporate company or any website owner would like to host a static content over their site. A structural mechanism to manage these web sites is provided by a system called Web Manager.

The structural mechanism provided in this system requires web server administrator who takes care of all the web sites hosted with web server. The administrator using site explore provides a gateway to the site owners to access the exact location of the web server where the site is placed. Using the Web Manager editing capability, changes can be made to the home directory as well as the web pages available. Even the code content can be viewed and edited by the authorized user. This provides a direct interaction to the site area from any corner of the world.

***Problems in the current web management system:*** In the current web management systems to create new content or the pages with newly made changes it requires the shutting down the website temporarily and updates the new contents with old ones. As every section of the web server is not allowed to be accessed from remote location this requires greater involvement of the Web Server administrator. The involvement of the web server administrator will raise the maintenance of the web site. The user (site owner) cannot see the code of the web page from the current location, for that it requires closing the site download of the older content and uploading the newer one.

***Required features for the existing web management system:*** The system should be able to provide interactive interface to the user through which he can interact with different areas of application easily. The information of the entire web sites hosted should be stored in a single

database that maintains the information regarding the users and their privileges. User can interact with the interface only in the manner the privileges provided by the web server administrator. This can be strictly enforced by maintaining the proper privilege set, with respect to the site's home directory.

***Purpose of the project:*** The purpose of the project is to develop a web based application for managing the web sites from any location. The system provides help in managing different web sites at once but one at a time. Provides an interface to change code without shutting down the code. Provides a provision to change the content and the functionality of the webpage without shifting the page from web server to the changed area. It makes the managing simpler for normal user and also speeds up the processing with in the system. Also provides security to the web access to the integral parts of the site.

## **PROJECT ANALYSIS**

The Web management has many users and depending on their privileges and features they are allowed to access the server to get the required task to be performed. Based on the access levels to the server the system can be divided in two modules, Administrator and User.

### **Administrator:**

In this application administrator of the web server is a person who manages the web sites on the web server. The job of the web server administrator includes managing the proper functioning of the web sites as well as management of the Web Manager. An admin has all the privileges on site explore and can create any number of users or administrator as situation

demands. An administrator creates a user with a privilege i.e, privilege such as deletions, updating, uploading files, viewing files adding new directories and files etc. The allotted privilege will work as a part of the security to the content which the user or the site owner is going to use.

Every user created is bonded with a home directory upon which the privileges will be allotted and on which the owner can work with. If the owner owns more than one website, only one web site can be managed with the user which is marked as the home directory. The user has the privilege to change the working directory. The privilege set can be seen in the admin toolbar with which the settings to the application privilege set can be managed. By default an administrator has all the privileges on a set site.

**User:**

The user of the Web Manager is the registered user with a particular site. Every user is bonded to at least one web site to which he/she can connect from a remote location. The working of a user depends on the privileges the user is assigned with by the administrator.

If all the privileges are provided to the user a user can create new documents or directories on the web server; view all the documents which are marked for viewing, can edit; delete or rename an existing documents if enabled with the specified feature; upload new files from the home computer to the home directory of the web server; can change the current working directory to other home directory (If more than two sites are owned by single registered and also manage the file settings.

The access control for data by the user needs to be authenticated. This is done by providing access control identifiers like; the *user* identification is that which is required by the server for access to its file system. This command will normally be the first command transmitted by the user after the control connections are made (some servers may require this); the *password* command must be immediately preceded by the user name command, and, for some sites, completes the user's identification for access control. As password information is quite sensitive, it is in general masked (encrypted).

### **Server:**

In information technology, a server is a computer program that provides services to other computer programs (and their users) in the same or other computers. A Web server is a program that, using the client/server model and the World Wide Web's Hypertext Transfer Protocol (HTTP), serves the files that form Web pages to Web users (whose computers contain HTTP clients that forward their requests). Every computer on the Internet that contains a Web site must have a Web server program. Two leading Web servers are Apache, the most widely-installed Web server, and Microsoft's Internet Information Server (IIS).

### **Data Base:**

A database management, or DBMS, gives the user access to their data and helps them transform the data into information. Such database management systems include dBase, paradox, IMS, MS Access and MS Access. These systems allow users to create, update and extract information from their database.

## SELECTED SOFTWARE

**Microsoft.Net Framework:** It simplifies the application development in the highly distributed environment of the Internet. It helps the developer to experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.

**Client Application Development:** These are the types of applications that display windows or forms on the desktop, enabling a user to perform a task. The Windows Form classes contained in the .NET Framework are designed to be used for GUI development. The .NET Framework integrates the developer interface, making coding simpler and more consistent.

Windows Form controls have semi-trusted access to a user's computer, this means that binary or natively executing code can access some of the resources on the user's system (such as GUI elements and limited file access) without being able to access or compromise other resources. Because of code access security, many applications that once needed to be installed on a user's system can now be safely deployed through the Web.

ACTIVE X DATA OBJECTS.NET (ADO.NET): ADO.NET is an evolution of the ADO data access model that directly addresses user requirements for developing scalable applications. ADO.NET is the data access model that comes with the .NET Framework. ADO.NET provides the classes required to communicate with any database source (including Oracle, Sybase, Microsoft Access, Xml, and even text files). ADO.NET is a Data Access Providers in .NET. A DATAPROVIDER is a set of classes that can be used to access, retrieve and manipulate data from the database. Both OleDb and SqlClient data providers have their own set of classes, but they have the same concepts. They are divided into two sets of classes:

*Classes for communicate with database:*

The first category of classes is used to communicate with database and send or retrieve data from the database. The second category of the classes will be used as a carrier of data.

Classes for communicating with data:



The **Connection**, **Command**, **DataReader**, and **DataAdapter** objects are the core elements of the ADO.NET provider model.

Object	Description	SqlClient Objects	OleDb Objects
<b>Connection</b>	Establishes a connection to a specific data source.	SqlConnection	OleDbConnection
<b>Command</b>	Executes a command against a data source.	SqlCommand	OleDbCommand
<b>DataReader</b>	Reads a forward-only, read-only stream of data from a data source.	SqlDataReader	OleDbDataReader
<b>DataAdapter</b>	Populates a <b>DataSet</b> and resolves updates with the data source.	SqlDataAdapter	OleDbDataAdapter

*Classes for holding data:*

The following are the main classes used to hold data in Ado.NET:

- **DataSet:** A DataSet is an in-memory representation of the database. It contains DataTables (and more...)
- **DataTable:** DataTable represents a database table. It contains DataRow (and more...)
- **DataRow:** DataRow represents a record in a database table. It is a collection of all fields in a record.

When dealing with connections to a database, there are two different options: MS Access .NET Data Provider (System.Data.SqlClient) and OLE DB .NET Data Provider (System.Data.OleDb). In this project *MS Access .NET Data Provider* is used. These are written to talk directly to Microsoft MS Access. As this application use MS Access database so OleDb objects are used.

### **Server Application Development:**

This application is a basic network schema with managed code running in different server environments. Servers such as *IIS (Internet Information Services)* and *MS Access* can perform standard operations while the application logic executes through the managed code.

*Server-side managed code (Active Server Pages .Net):* ASP.NET is the hosting environment that enables developers to use the .NET Framework to target Web-based applications. However,

ASP.NET is more than just a runtime host; it is a complete architecture for developing Web sites and Internet-distributed objects using managed code. Both Web Forms and XML Web services use IIS and ASP.NET as the publishing mechanism for applications, and both have a collection of supporting classes in the .NET Framework.

**MS Access:** MS Access is a very light weight database provided by Microsoft for applications with less number of users and relatively small quantity of data. MS Access saves data into database files with the extension .mdb. Every table in MS Access has a field or a combination of fields that uniquely identifies each record in the table. The Unique identifier is called the *Primary Key*.

MS Access's networking and distributed database capabilities to access data stored on remote server with the same ease as if the information was stored on a single local computer. A single SQL statement can access data at multiple sites. Data can be stored where system requirements such as performance, security or availability dictate.

## SOFTWARE REQUIREMENT

**Required Software:** Windows XP Professional operating system; Visual Studio .Net 2002 Enterprise Edition; Internet Information Server 5.0; Visual Studio .Net Framework (Minimal for Deployment); MS Access (data base) is used for this application.

## DATA

### CONTROLS

**Data Storage using MS Access Tables:** Depending on the requirements the project, two tables would be sufficient to store the entire data about **Users (client)** and their **Privileges**. The below tables are normalized to avoid any anomalies during the course of data entry.

**Users:**

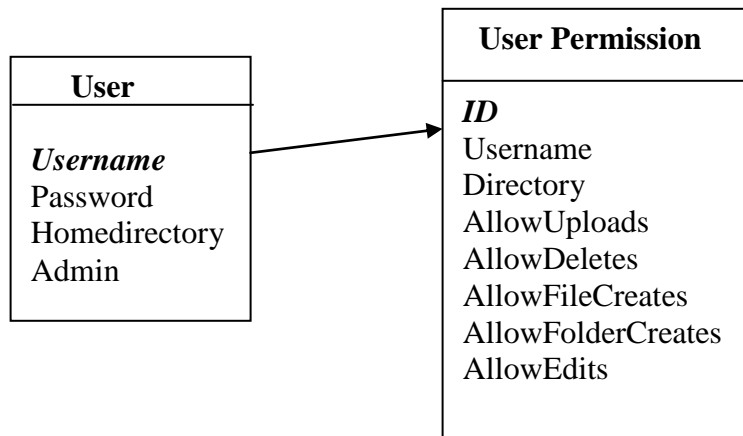
S. No	Field Name	Data Type	Description
1	User Name	Text(10)	Name of user(Primary Key)
2	Password	Text(10)(password)	Password of the user
3	Home Directory	Text(50)	Directory of the user
4	Admin	Yes/no	Is the user a admin or not

**User Permission:**

S. No	Field Name	Data Type	Description
1	ID	Number(9)	Number of users(Auto Number)
2	User Name	Text(10)	Name of the user(Foreign Key)
3	Directory	Text(50)	Directory of the user
4	Permission	Yes/No	Is the permission given to the user
5	Allow Upload	Yes/No	Is the user allowed to upload
6	Allow Deleted	Yes/No	Is the user allowed to delete
7	Allow Folder Creates	Yes/No	Is user allowed to create folder
8	Allow File Creates	Yes/No	Is user allowed to create file
9	Allow Edits	Yes/No	Is user allowed to edit

**PROJECT DESIGN**

**ENTITY-RELATION DIAGRAM:**



User and User permission are the two data tables in the data base. These tables are related by the primary key, 'User name'. Depending on the user and the assigned privileges user permissions will be dependent. Every user is assigned by a particular id number with which they are authenticated.

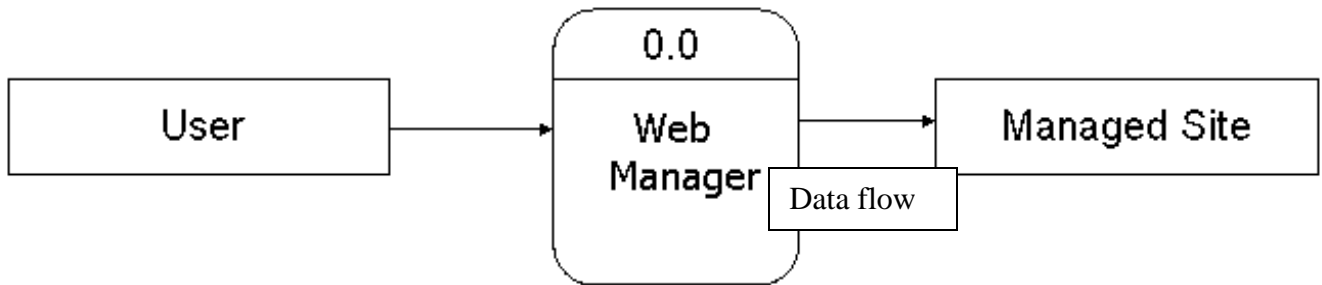
#### **DATA FLOW DIAGRAM:**

A data flow diagram is graphical tool used to describe and analyze movement of data flow through a system. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow diagrams. A full description of a system actually consists of a set of data flow diagrams.

To move data from one *Data store* to another a process must be used. Data cannot move directly from a *Source* (origin) to *Sink* (destination) it must be moved by a process. A *Data flow* has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. It cannot go directly back to the same process it leads. There must be at least one other process that handles the data flow produce

some other data flow returns the original data into the beginning process. A Data flow to a data store means update (delete or change).A data flow from a data store means retrieve or use.

***First level:***

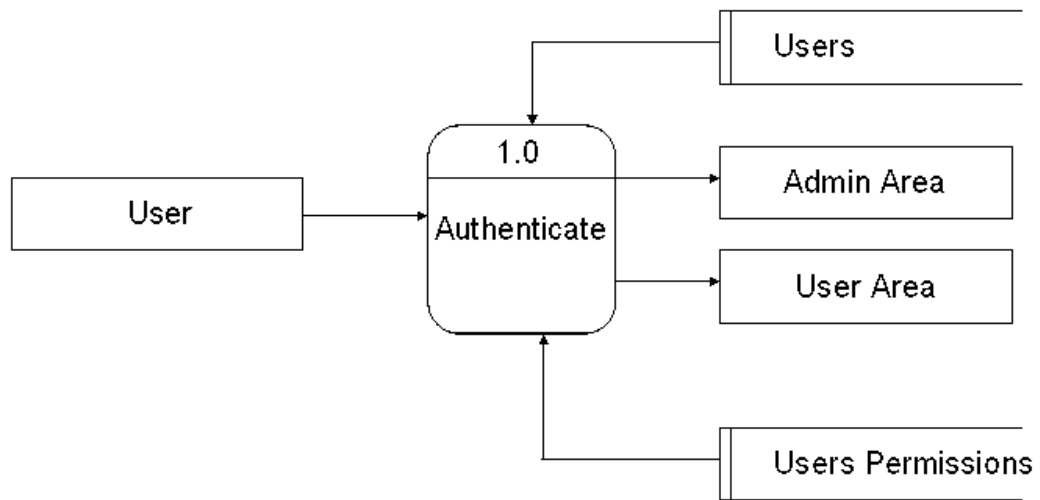


*Web manager:* Process that transforms data flow

*User:* Source of data

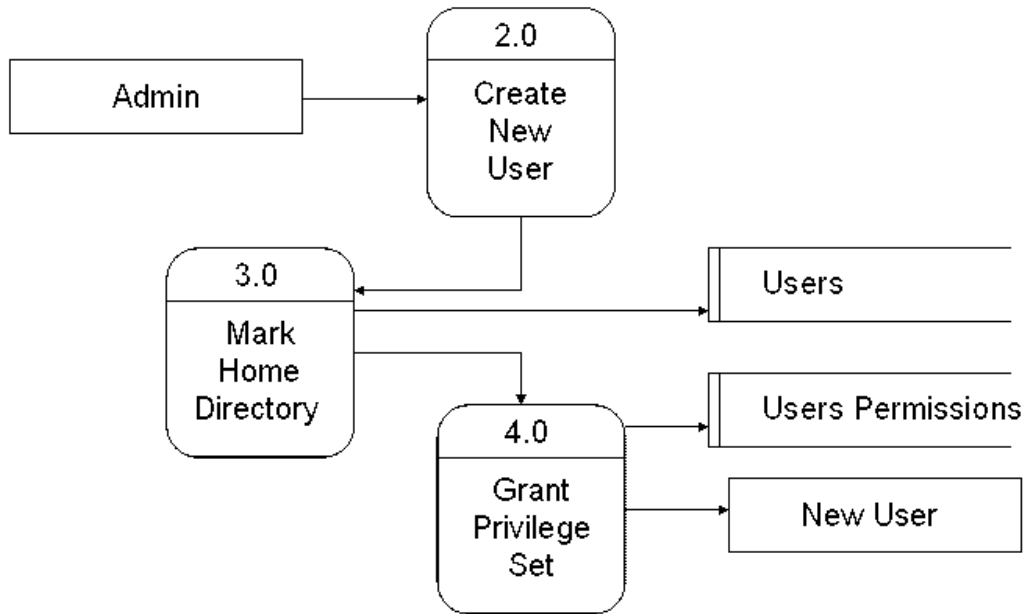
*Managed site:* Destination of data

***Second level:***

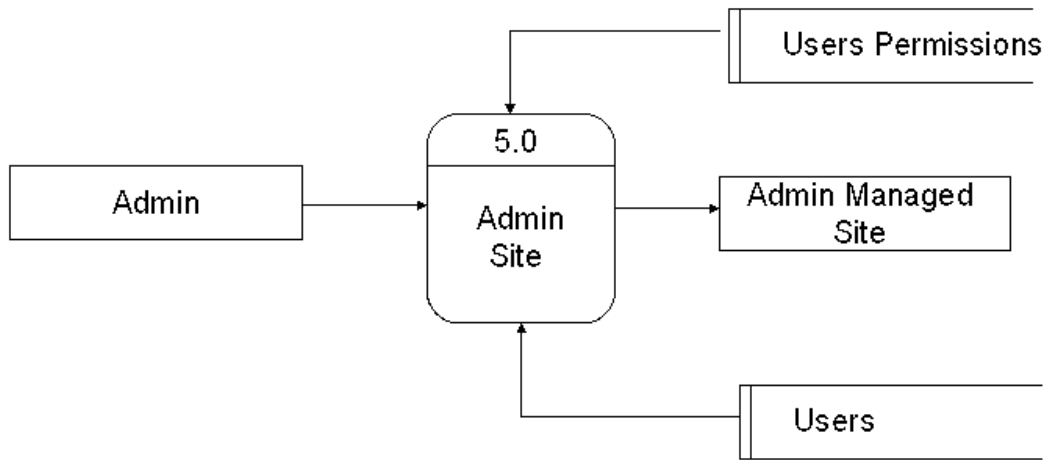


*User and User Permissions: Data store*

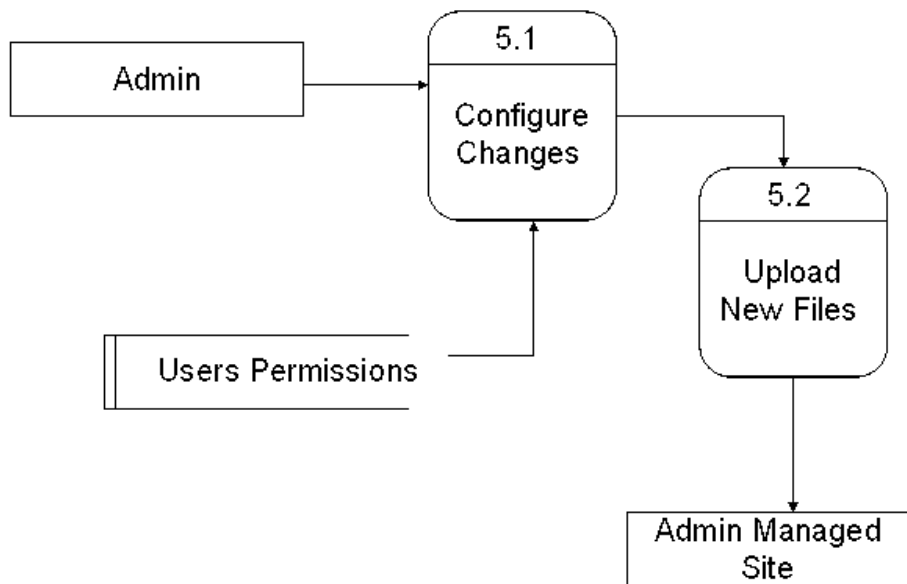
*Third level :(Create a new user)*



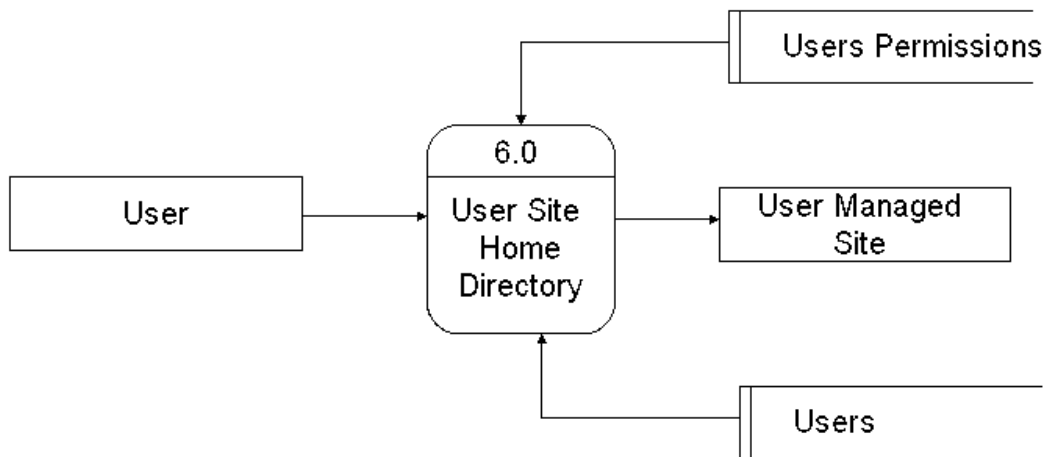
*Forth level:*



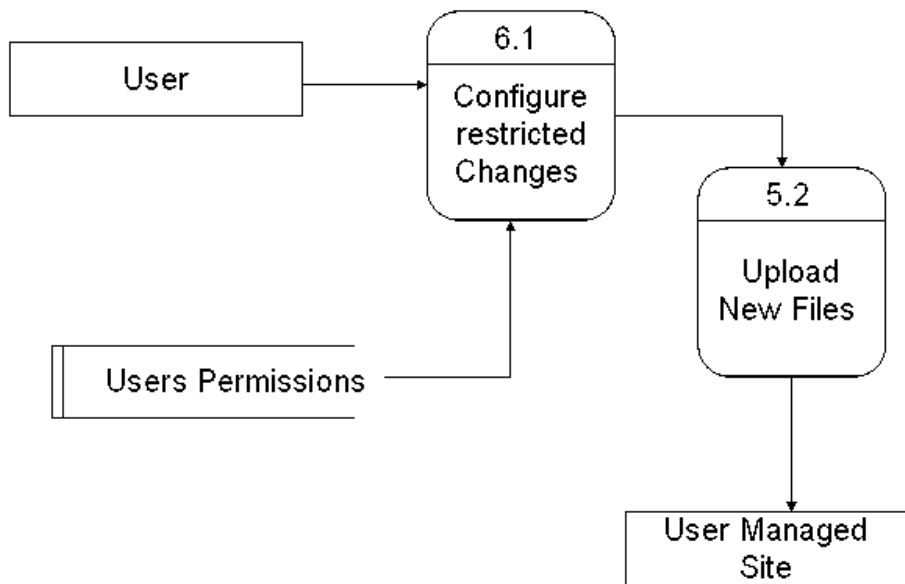
*Fifth level:*



*Sixth level:*



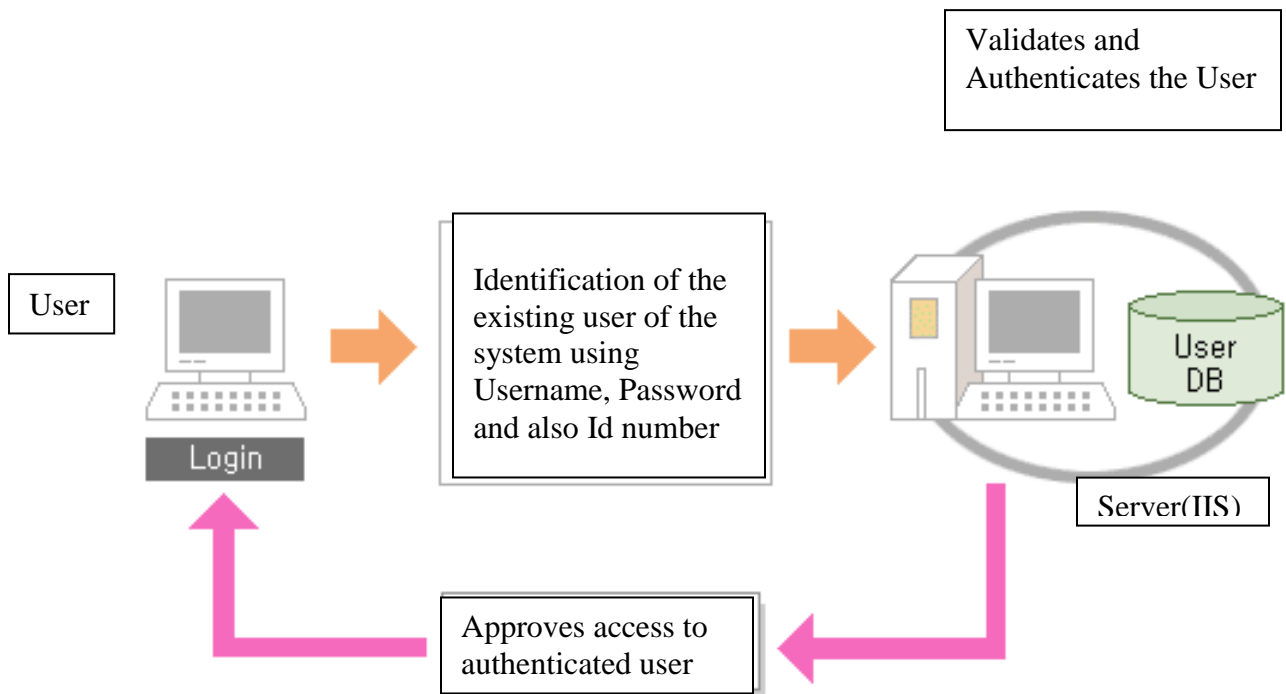
*Seventh level:*





**Working Process:** When a user tries to login to this site. This application checks for the validity and authentication of the user. If it's an authenticated user it checks for if it is an administrator or a user. If the user is an administrator he enjoys all the privileges in the system. He can create a new user, view all the users, can perform create, delete, upload functions on files and folders. If it's just a user then depending on the provided privileges like allow to delete, allow to upload, allow to create etc. Those functions are performed by them depending on their access. They can also change their settings like password.

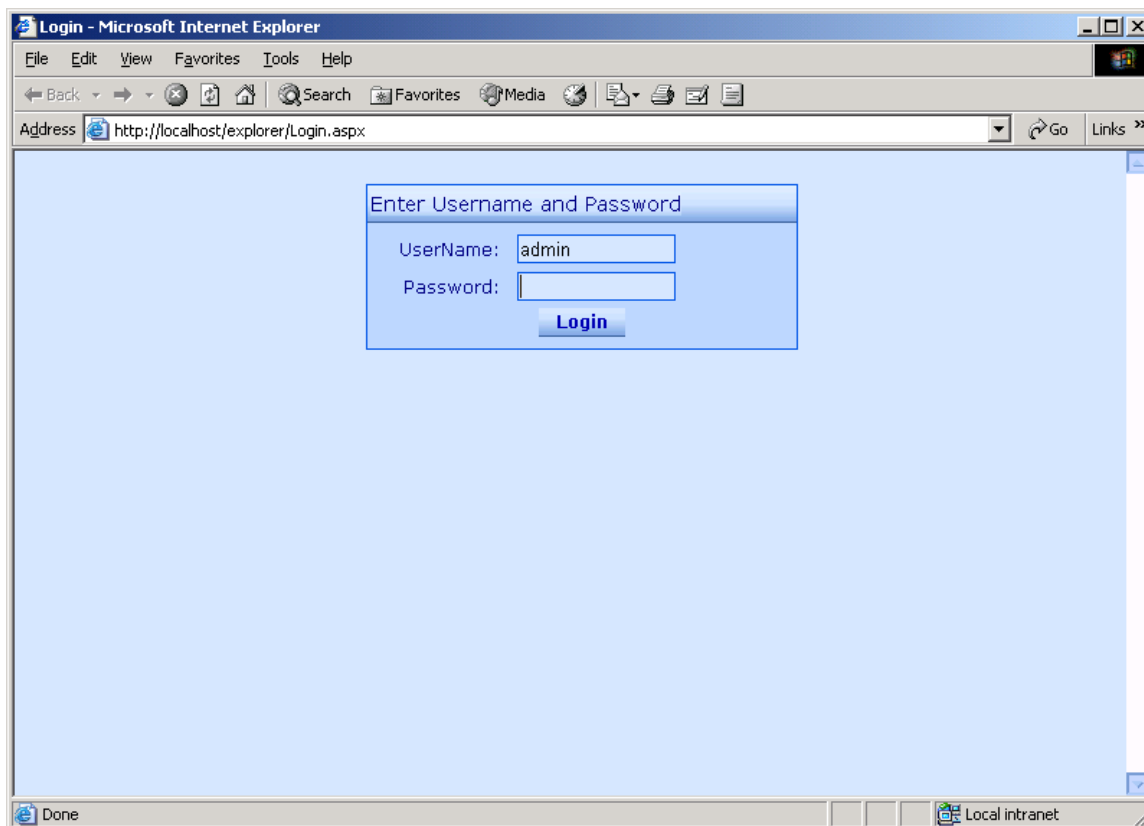
**Representation of the Authenticated Web Management System:**



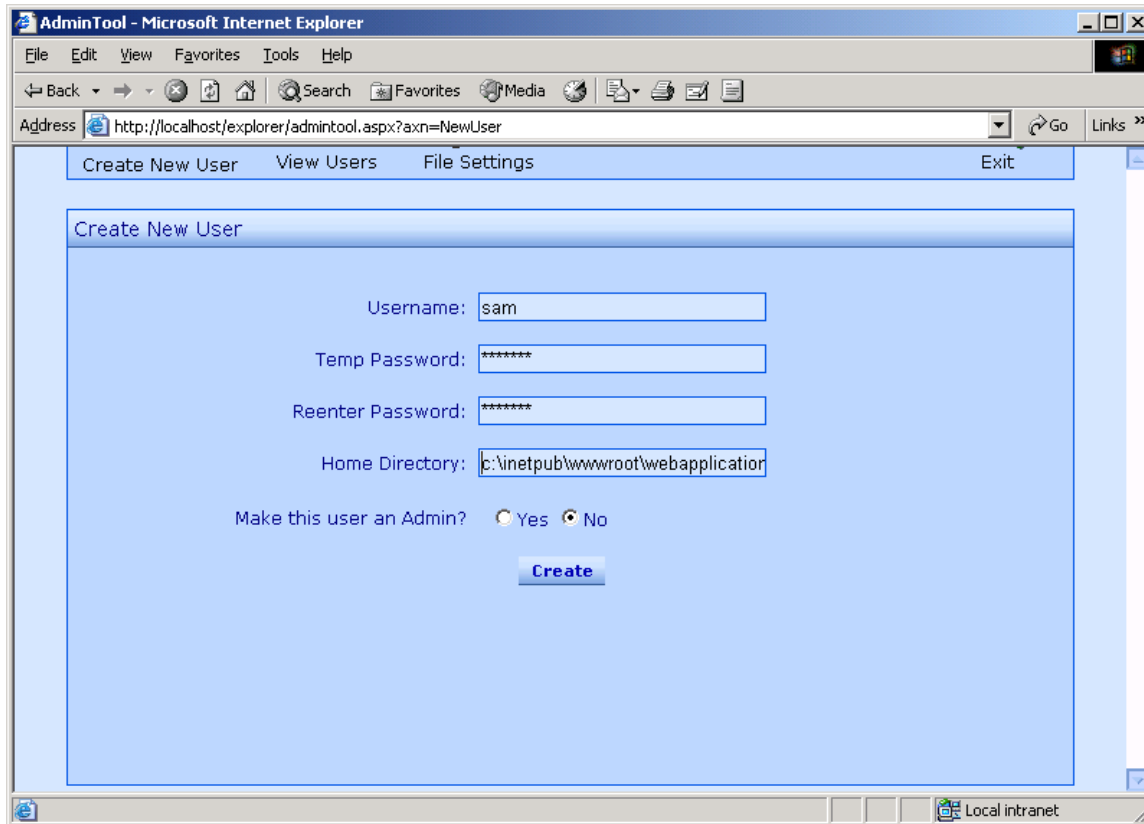
The application is run successfully and the output snapshots and given below.

## OUTPUT SCREENS

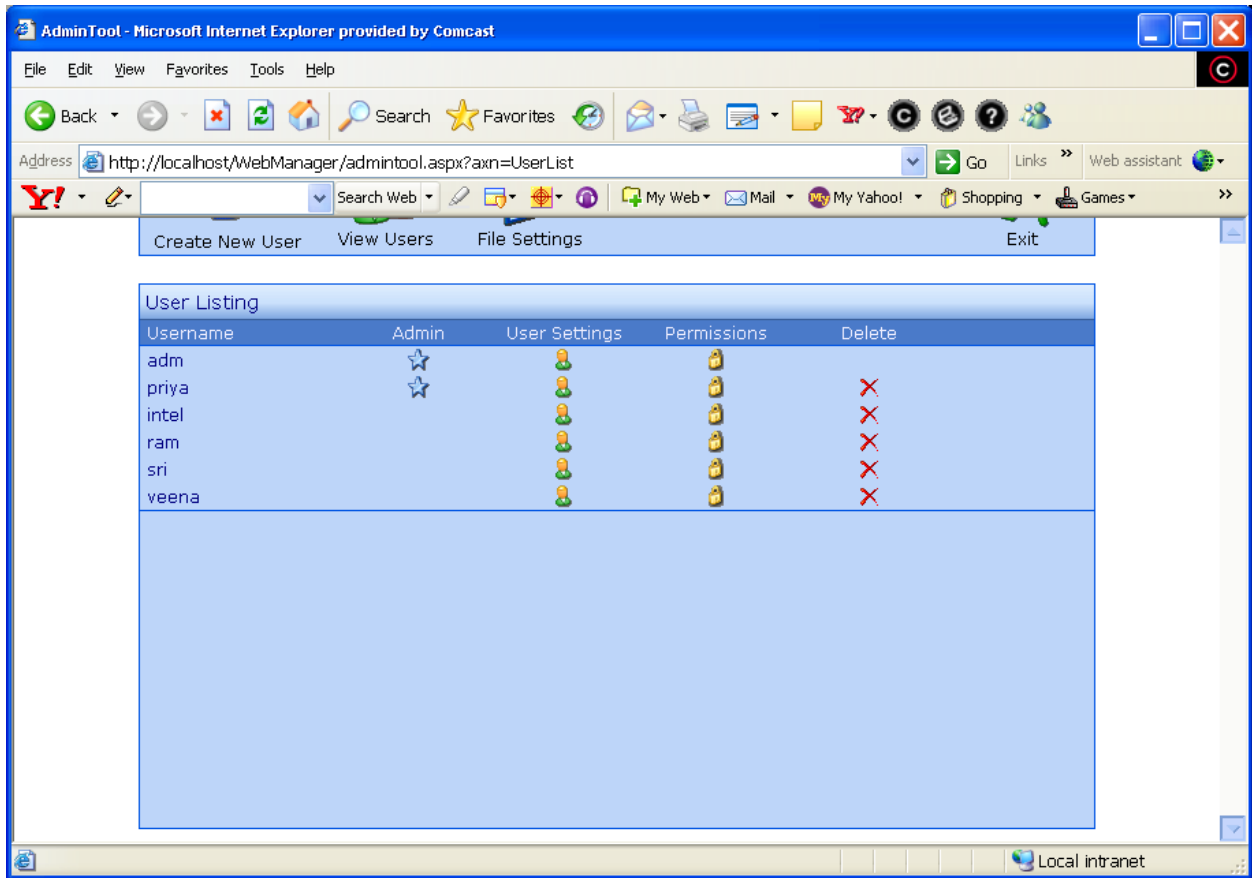
### Admin Login:



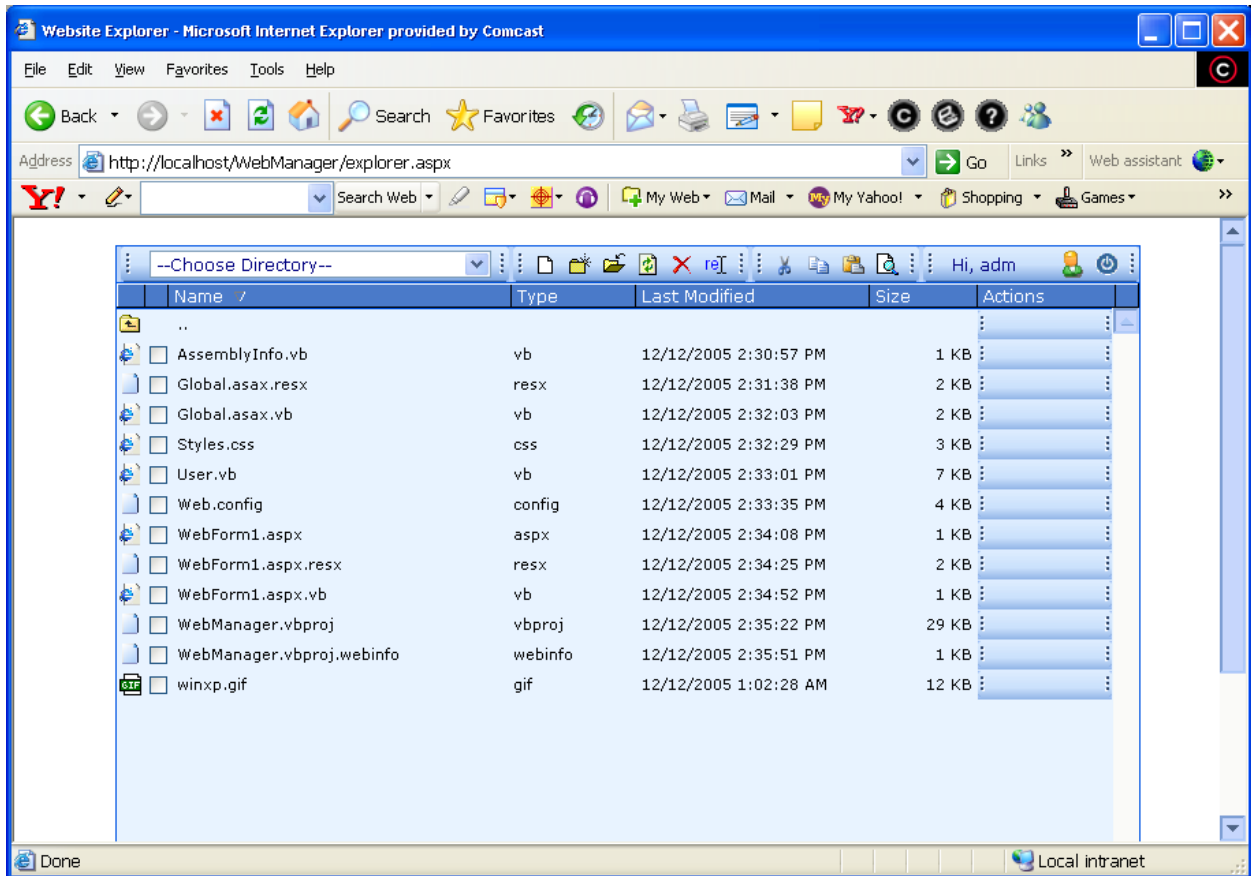
### New User Creation:



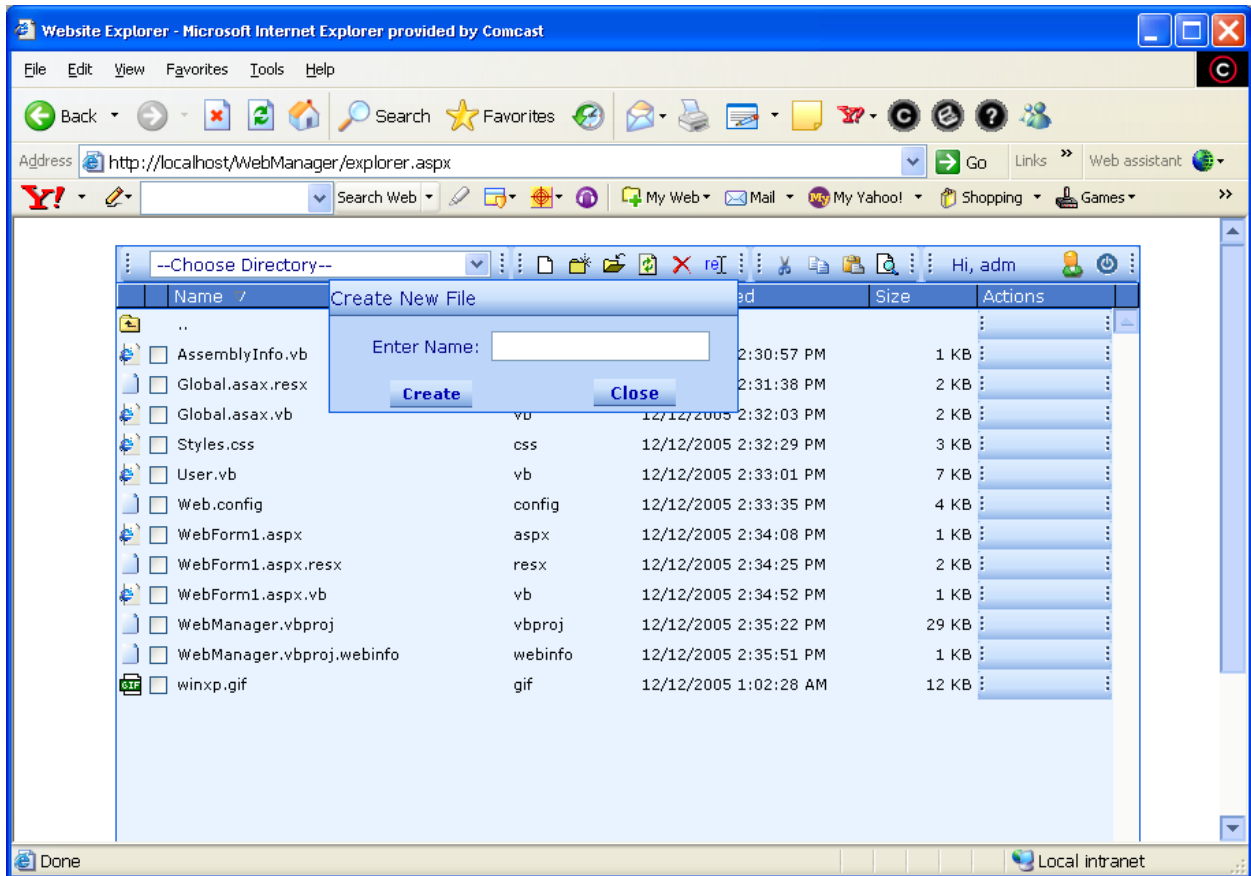
**Users with privileges:**



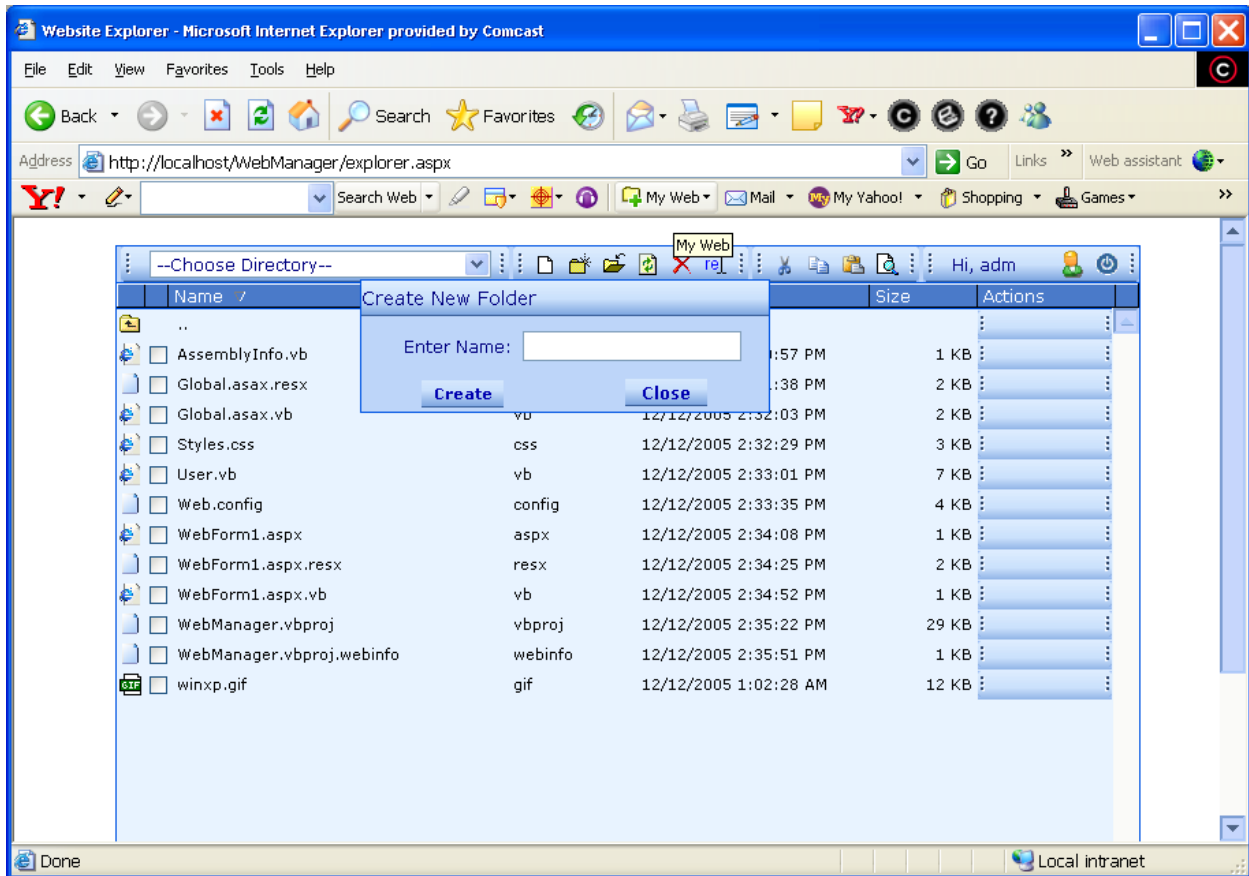
**Administrator Area:**



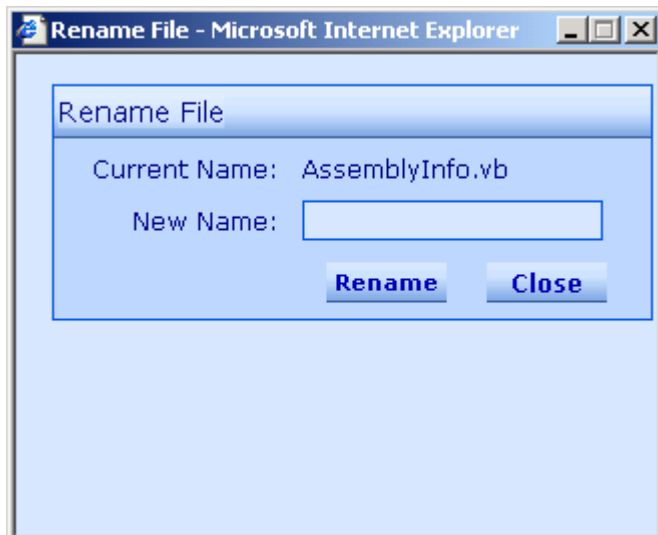
## Creating New File:



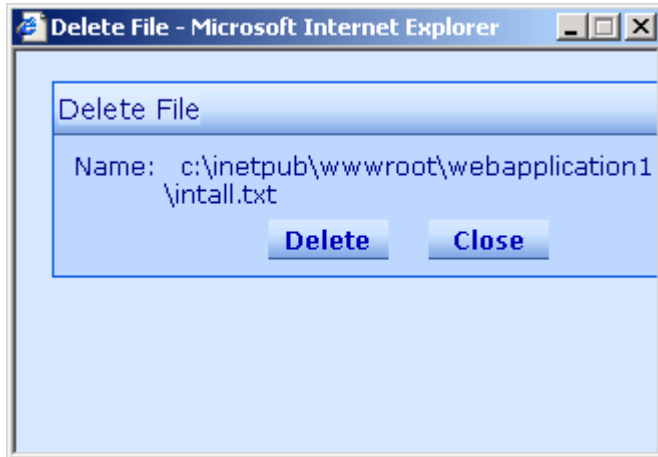
### New Folder Creation:



### **Renaming a File:**

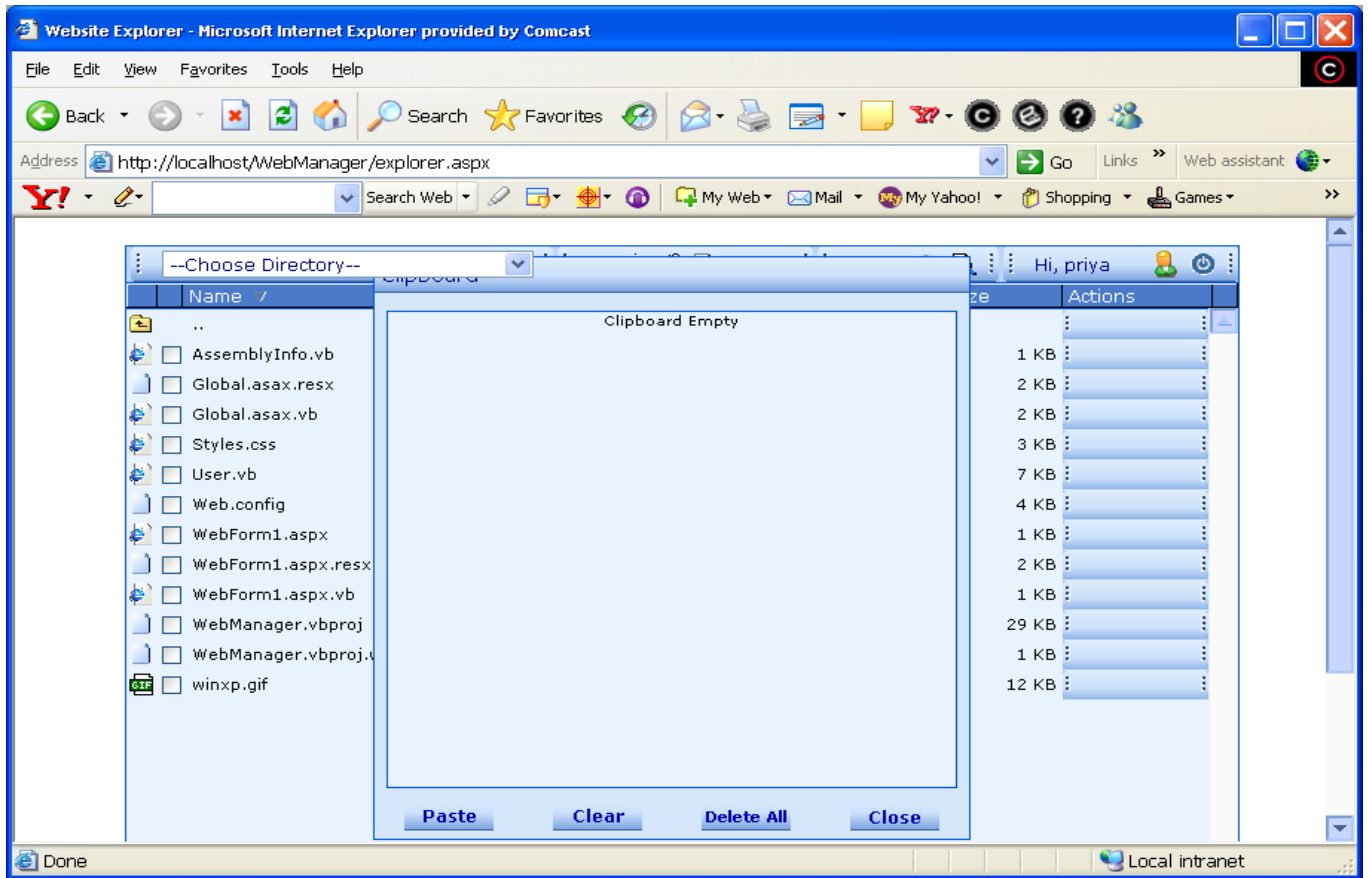


**Deleting a File:**

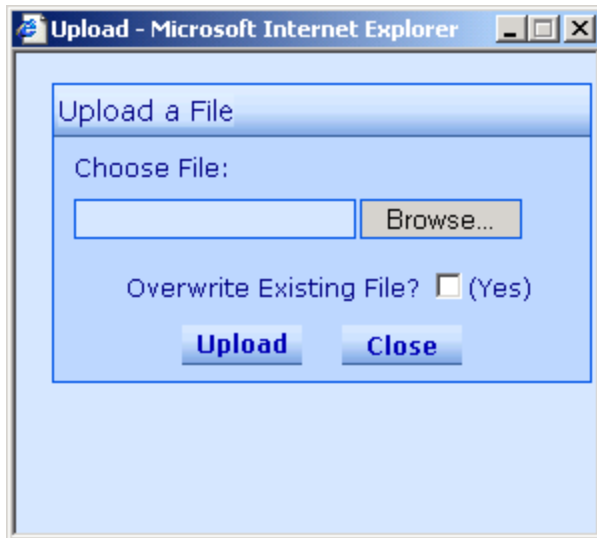


**Clipboard Area:**

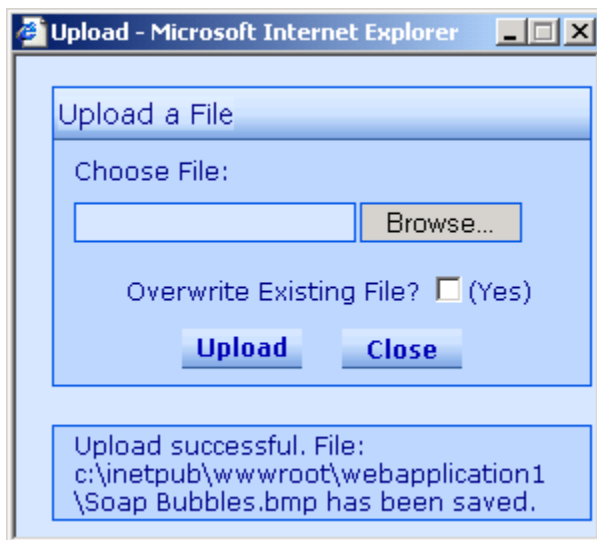




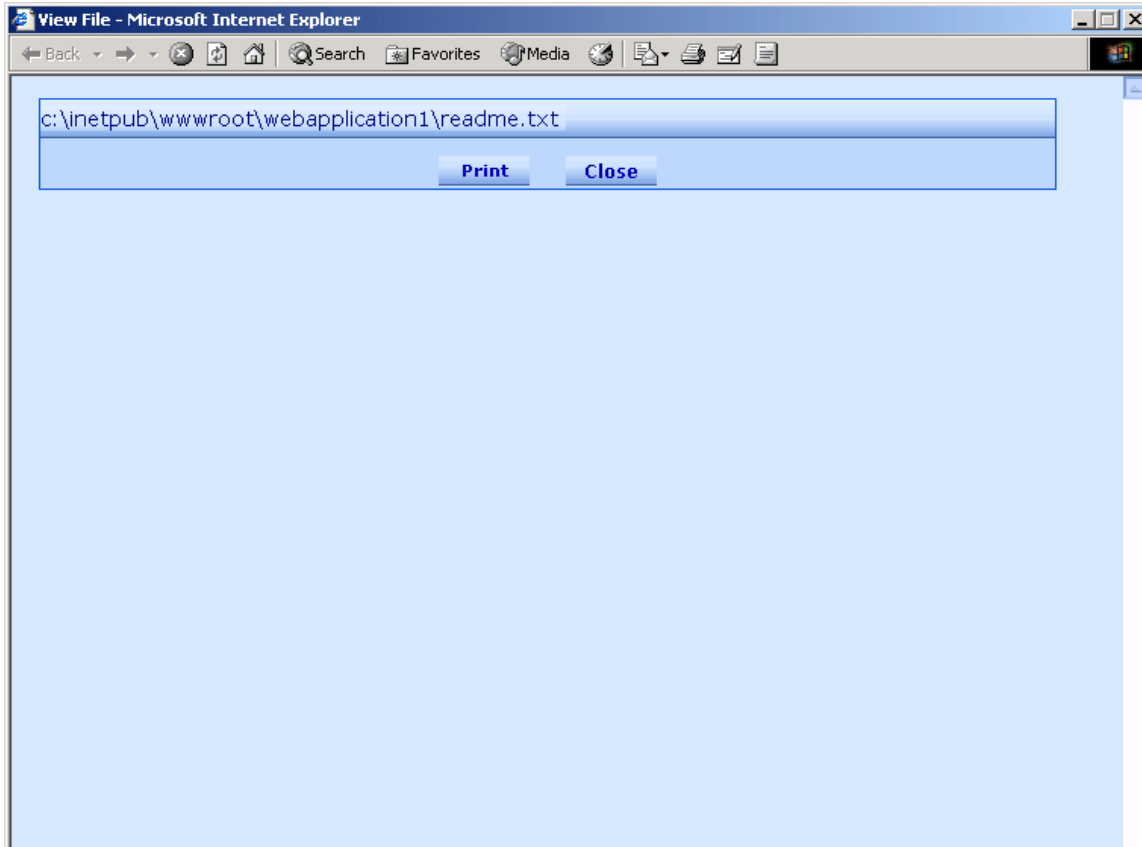
### Uploading Files:



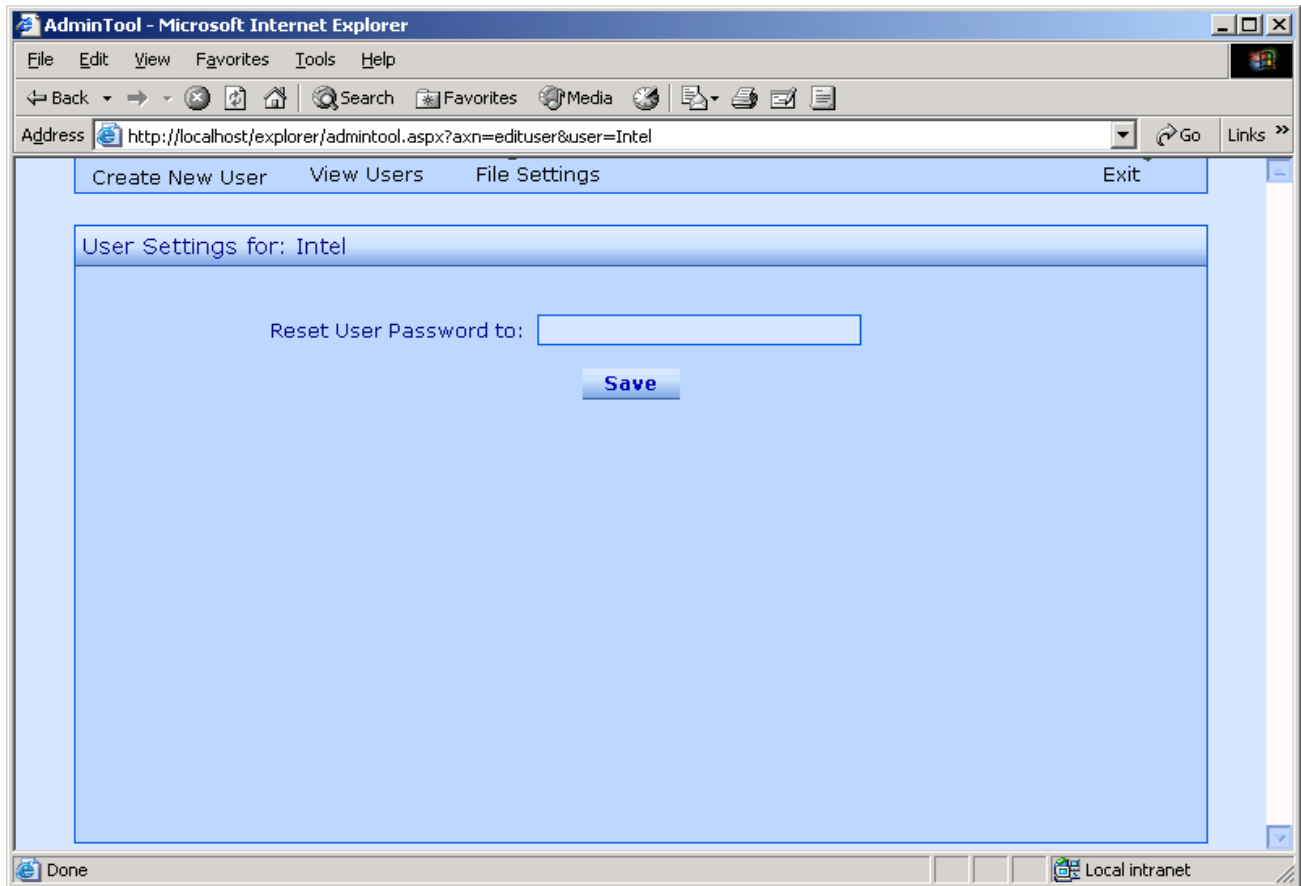
**Upload Status:**



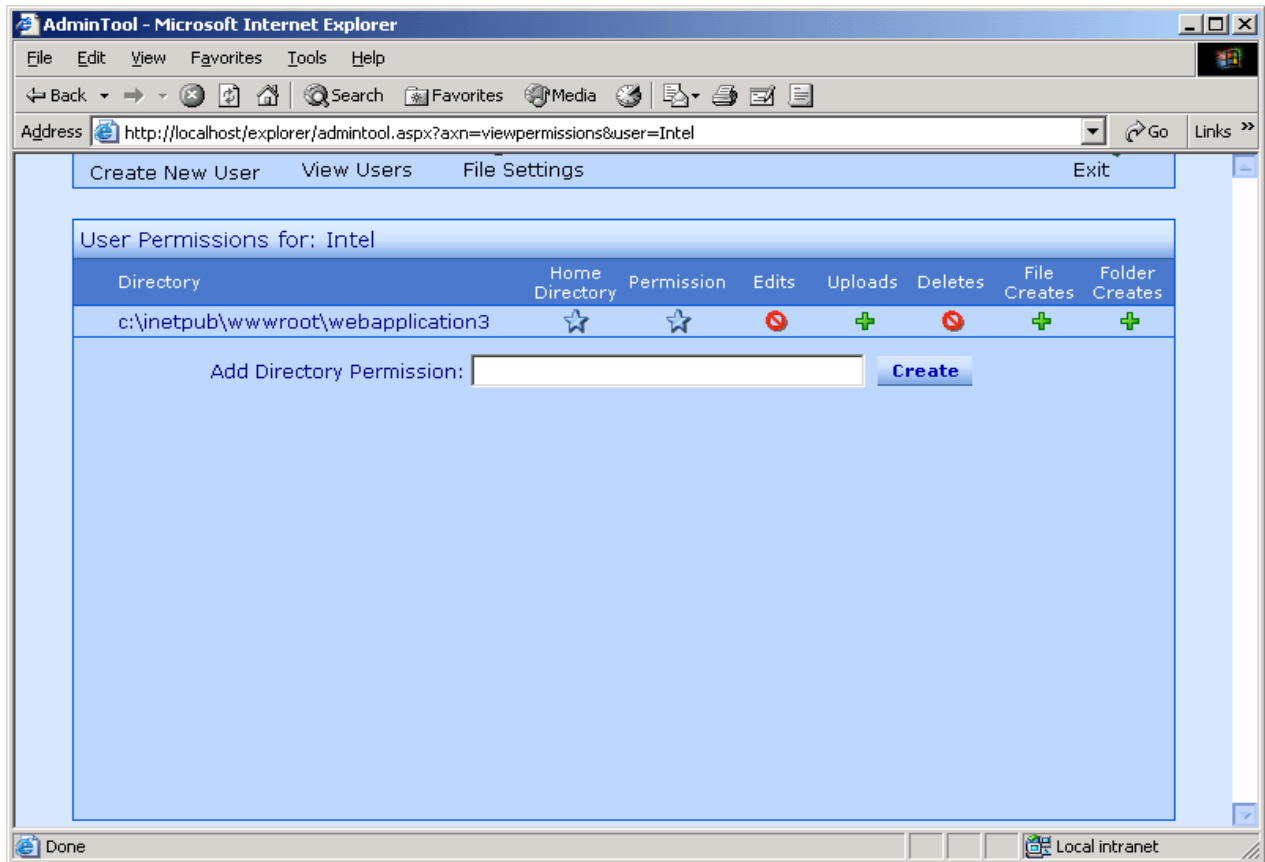
**Printing:**



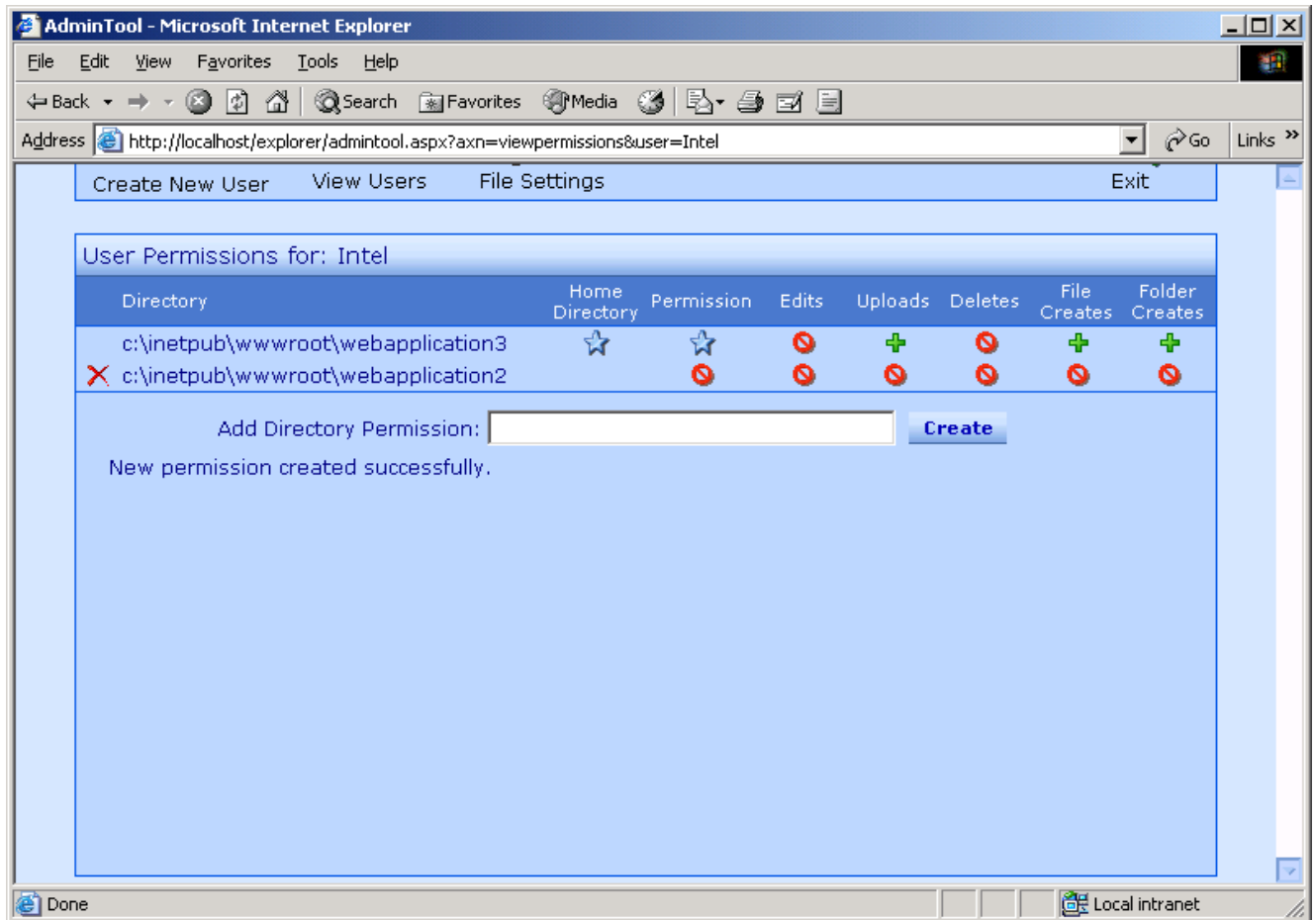
**User Settings:**



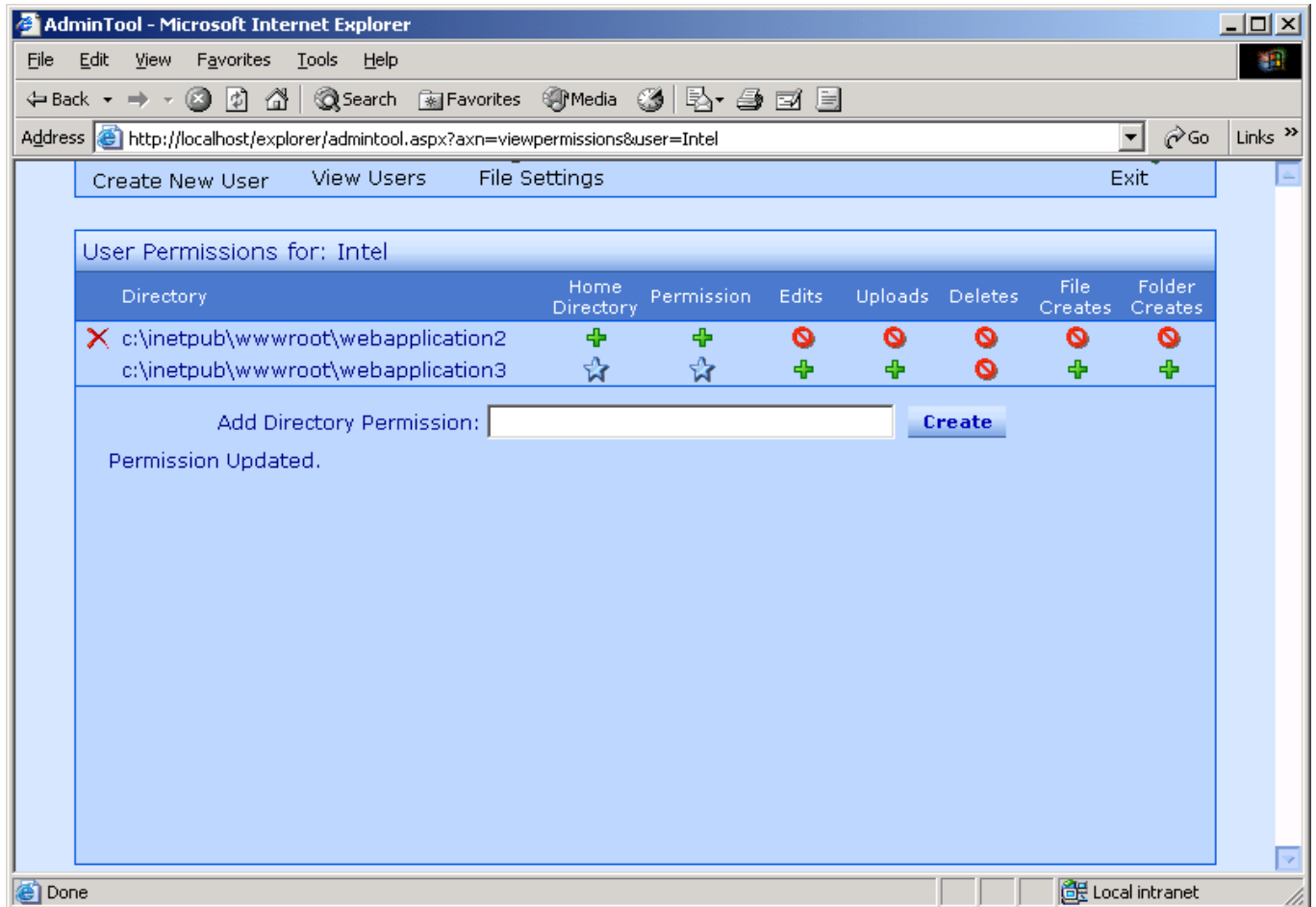
**User Permissions:**



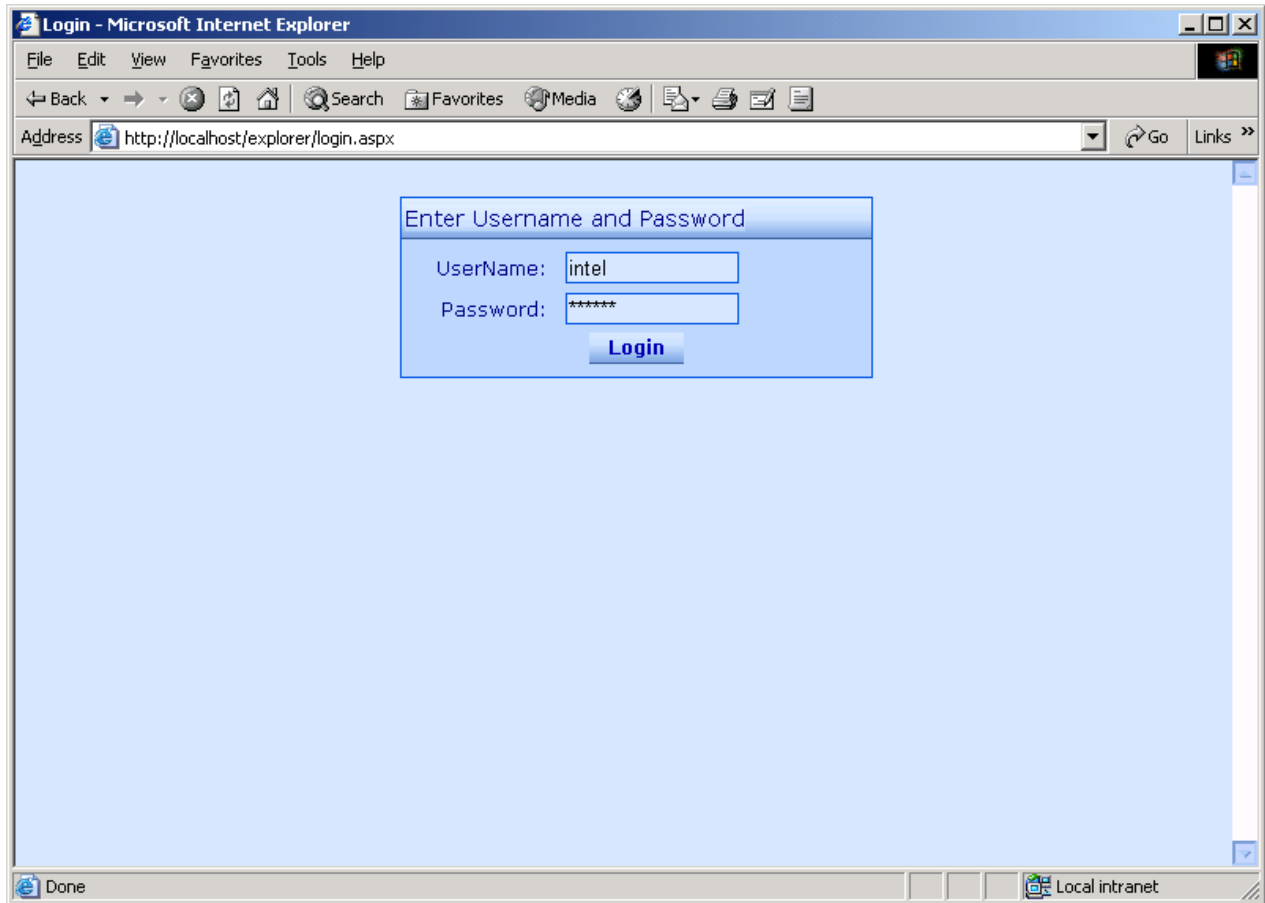
### **Home Directory Creation:**



### **Granting Privileges:**



**User Area:**



## CONCLUSION



In an era where web development is taking priority over other areas of development, the management of web based resources is being neglected. In addition to creating the content of the Web site, it is often needed to make changes, like adding or removing pages or code, or making configuration changes. Proper Web Site Management is one of the keys to having a successful website. This project is mainly designed to manage web sites even from a remote location depending on the privileges provided to the user.

With this application more secured web management system is obtained which provides authentication not only to the administrators but also to the user. It is easy to use, since it uses the **GUI** provided in the user dialog. User friendly screens are provided. The usage of software increases the efficiency, decreases the effort. It is relevant to real time entity. This application provides most of the features required to manage the site contents hosted on a web server. It also focuses on providing excellent environment for managing web sites.

## **REFERENCES**

**VISUAL BASIC.NET** (Wrox Publications)  
By Richard Blair

**ASP.Net Professional** (Wrox Publications)  
By Alex Homer

**Head First HTML with CSS & XHTML**  
By Eric Freeman, Elisabeth Freeman

**An Introduction to DataBase Systems**  
By C.J.Date

**MSDN Help**

<http://kiwi.bridgeport.edu/cs555>

<http://www.dotnetspider.com>

[http://www.bambooweb.com/articles/c/o/Content\\_management\\_system.html](http://www.bambooweb.com/articles/c/o/Content_management_system.html)

<http://www.wdvl.com/Authoring/Tutorials>

<http://www.w3schools.com>

[http://searchwebservices.techtarget.com/sDefinition/0,,sid26\\_gci213606,00.html](http://searchwebservices.techtarget.com/sDefinition/0,,sid26_gci213606,00.html)

<http://www.htmlgoodies.com/primers/database/article.php/3478121>