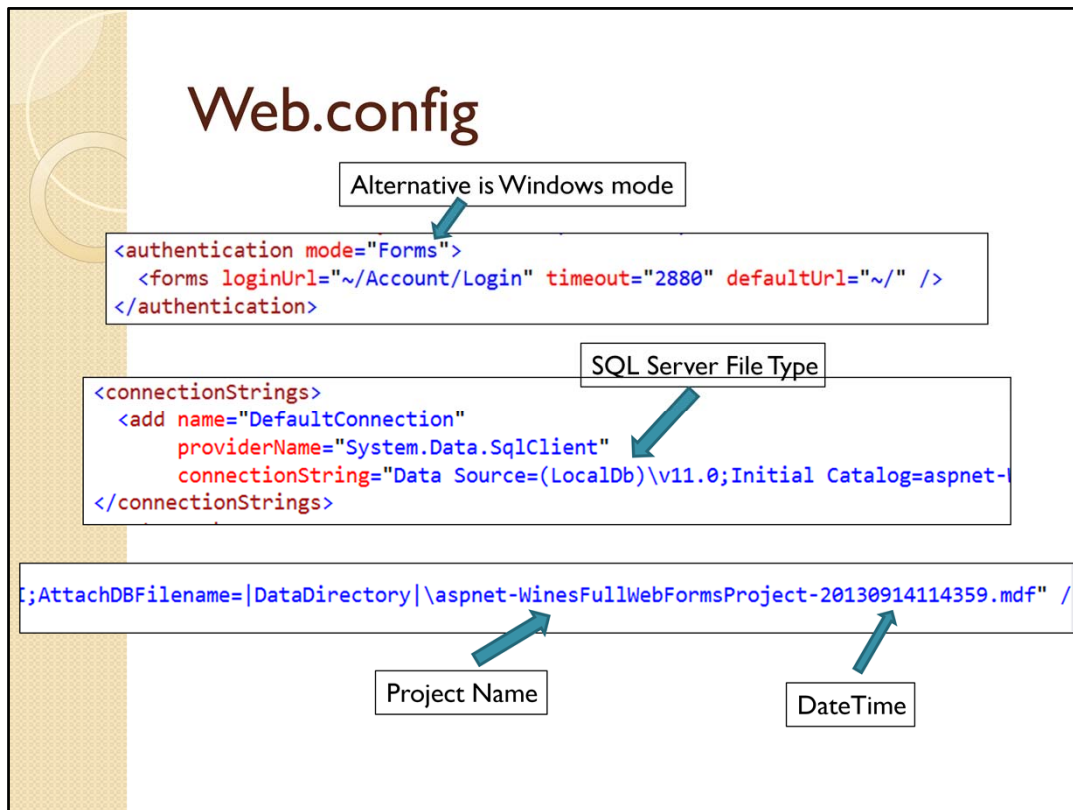


# Security - Forms Authentication

By Dr Derek Peacock



By default security is built in to a ASP.NET Web Forms application or an ASP.NET MVC application, and the settings are already in the root level web.config file. Forms mode is best used with internet applications and Windows mode for intranet applications. In forms mode the use must enter a username and password to access the pages where access is limited to authorised users.

By default usernames and passwords will be stored in a local SQL Server LocalDb database file stored in the App\_Data folder. This database is automatically created when the first user registers.

All these settings can be changed.

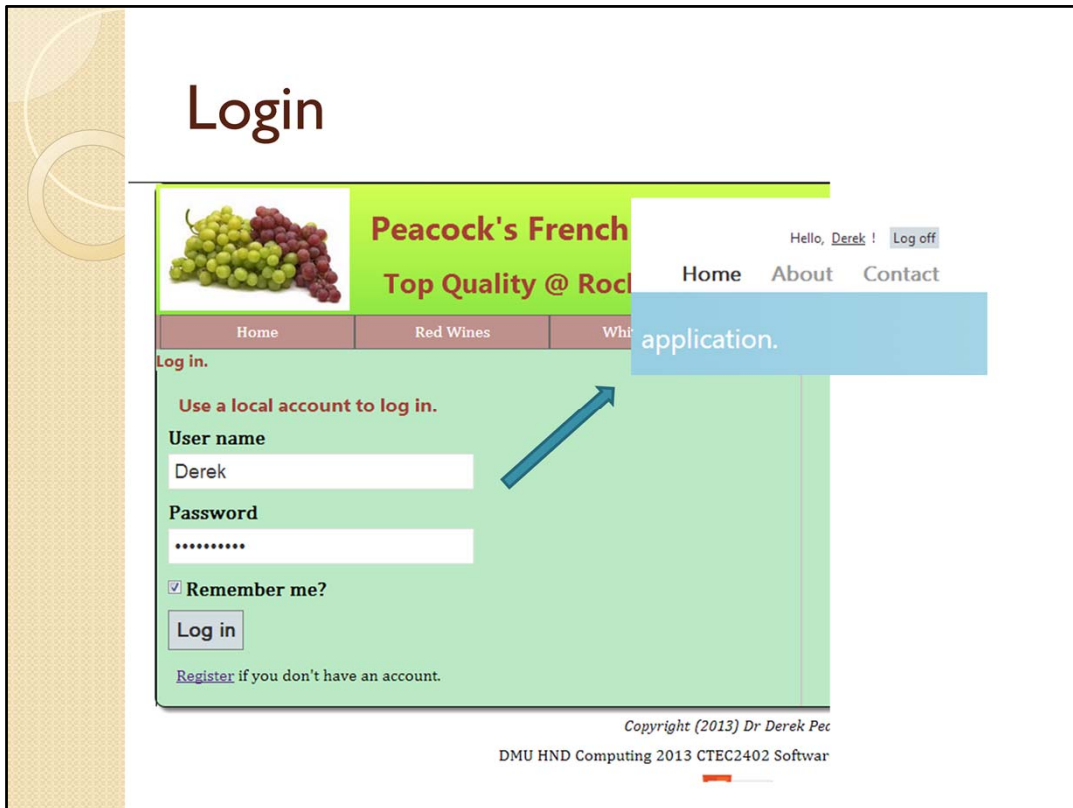
# Initialise the Database

The screenshot shows a web application interface for 'Peacock's French Wine Application'. At the top right, there are links for 'Register' and 'Log in', and a navigation menu with 'Home', 'About', and 'Contact'. The main header features a green banner with a cluster of grapes and the text 'Peacock's French Wine Application. Top Quality @ Rock Bottom Prices'. Below this is a navigation bar with 'Home', 'Red Wines', 'White Wines', and 'Sparkling Wines'. The central content area is a registration form with the following fields and instructions:

- Register.**
- Use the form below to create a new account.
- Passwords are required to be a minimum of 6 characters in length.
- User name:** Derek
- Email address:** derek.peacock@yahoo.com
- Password:** [masked with asterisks]
- Confirm password:** [masked with asterisks]
- Register** button

At the bottom of the page, there is a copyright notice: 'Copyright (2013) Dr Derek Peacock DMU HND Computing 2013 CTEC2402 Software Development'. A blue arrow points from the 'Register' button to the text 'application.' in the header.

To create the SQL Server database to store user names and passwords, run your application and select Register. By default passwords need to be 6 or more characters.



Once registered, the users name replaces the register link in the top right hand corner of the web page. To login again once log offed click on the Log In link.

# Data Validation



**Peacock's French Wines**  
**Top Quality @ Rock Bottom**

Home	Red Wines	White Wines
------	-----------	-------------

Log in.

**Use a local account to log in.**  
**Your login attempt was not successful. Please try again.**

**User name**  
Derekp

**Password**  
[Redacted]

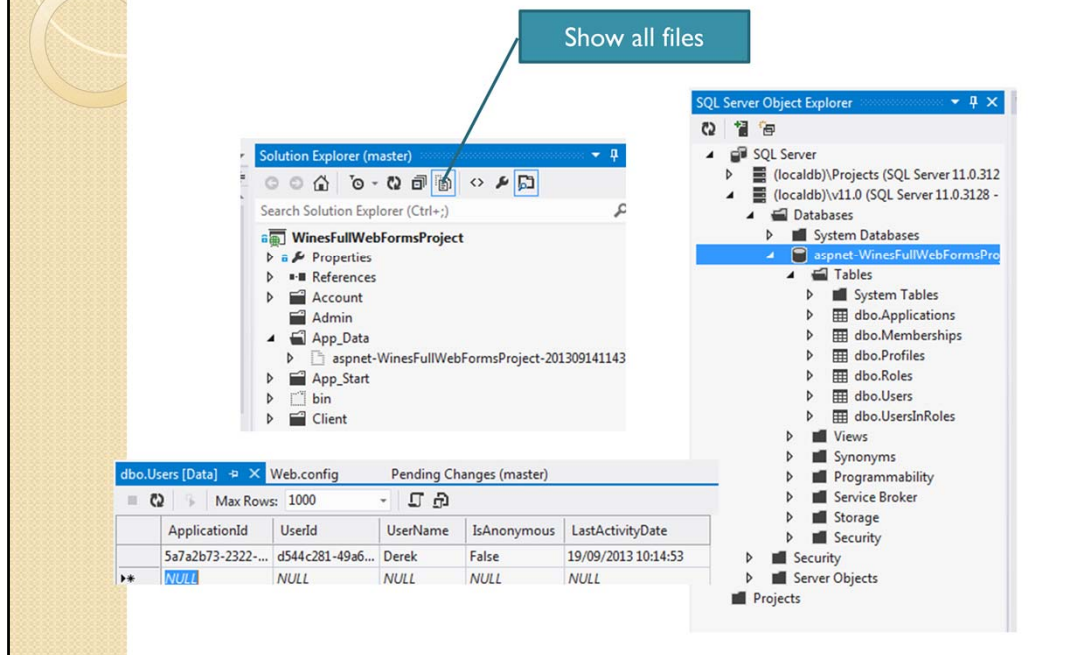
Remember me?

[Register](#) if you don't have an account.

Copyright (2012)

The register and login pages have built in data validation

# Accessing the Database File

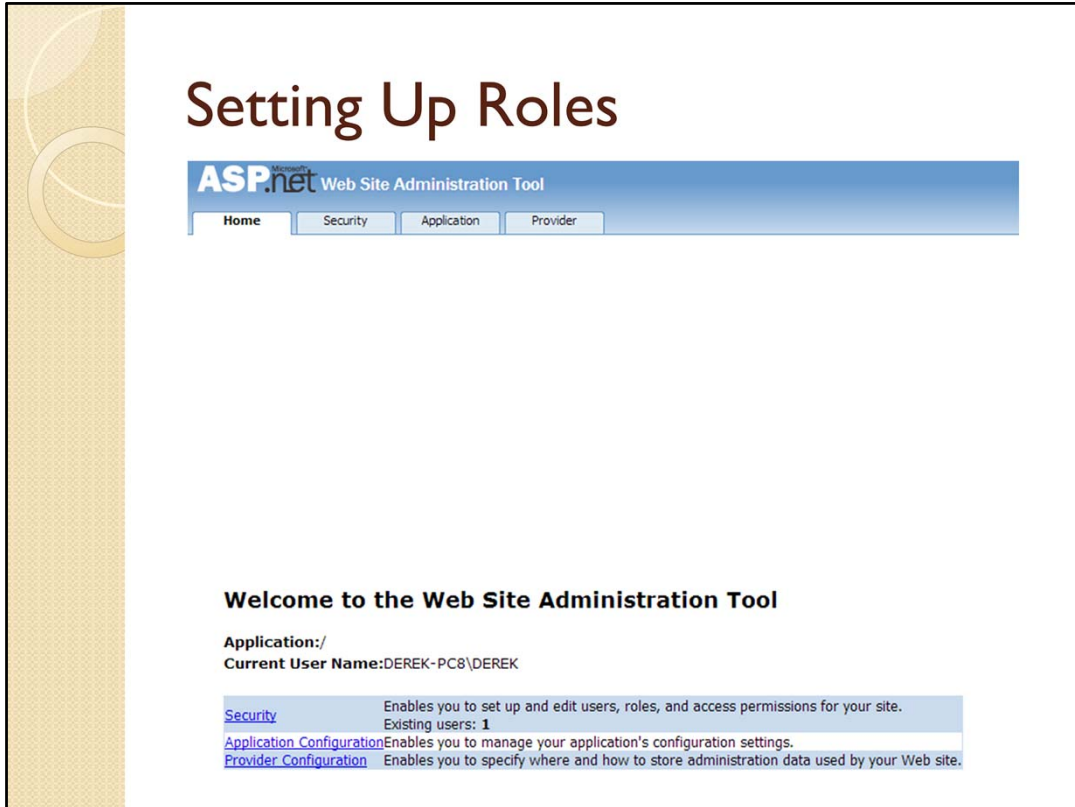


For some reason the new database file is not automatically part of the project. This means the file is not displayed in the solution explorer, unless you **display all files**. You may need to refresh files first. To view the contents of the SQL Server database file, you need to view the SQL Object Inspector.

# Membership Settings

```
<membership defaultProvider="DefaultMembershipProvider">
  <providers>
    <add name="DefaultMembershipProvider"
      type="System.Web.Providers.DefaultMembershipProvider,
      connectionStringName="DefaultConnection"
      enablePasswordRetrieval="false"
      enablePasswordReset="true"
      requiresQuestionAndAnswer="false"
      requiresUniqueEmail="false"
      maxInvalidPasswordAttempts="5"
      minRequiredPasswordLength="6"
      minRequiredNonalphanumericCharacters="0"
      passwordAttemptWindow="10" applicationName="/" />
  </providers>
</membership>
```

The membership security system can be configured in various ways. The standard settings are shown above. The only one I would be tempted to change for sure is that I believe each user should have a unique email address in order to avoid confusion.



In order to assign different roles to different users, goto Project => ASP.NET Configuration, and then select security.



# Enable Roles

**Roles**  
Roles are not enabled  
[Enable roles](#)  
[Create or Manage roles](#)

**Roles**  
Existing roles: **0**  
[Disable Roles](#)  
[Create or Manage roles](#)

You can optionally add roles, or groups, that enable you to allow or deny groups of users access to specific folders in your Web site. For example, you might create roles such as "managers," "sales," or "members," each with different access to specific folders.

**Create New Role**  
New role name:

Before you can create roles, roles need to be enabled and then different roles such as admin and client can be created and added.

# Roles Added

You can optionally add roles, or groups, that enable you to allow or deny groups of users access to specific folders in your Web site. For example, you might create roles such as "managers," "sales," or "members," each with different access to specific folders.

## Create New Role

New role name:

Role Name	Add/Remove Users	
Admin	<a href="#">Manage</a>	<a href="#">Delete</a>
Client	<a href="#">Manage</a>	<a href="#">Delete</a>

Once roles have been added then you can click **Manage** to assign a role to a user.

# Roles Assigned

Use this page to manage the members in the specified role. To add a user to the role, search for the user name and

Role: **Admin**

## Search for Users

Search By:  for:

Wildcard characters \* and ? are permitted.

[A](#) [B](#) [C](#) [D](#) [E](#) [F](#) [G](#) [H](#) [I](#) [J](#) [K](#) [L](#) [M](#) [N](#) [O](#) [P](#) [Q](#) [R](#) [S](#) [T](#) [U](#) [V](#) [W](#) [X](#) [Y](#) [Z](#) [All](#)

User name	User Is In Role
Derek	<input checked="" type="checkbox"/>

```
<roleManager enabled="true" defaultProvider="DefaultRoleProvider">
  <providers>
    <add connectionStringName="DefaultConnection" applicationName="/"
      name="DefaultRoleProvider" type="System.Web.Providers.DefaultRoleP
    </providers>
  </roleManager>
```

Once roles have been created, they can be assigned to chosen users, or all users. Enabling roles is a setting written into the web.config file.

# Database Content

	ApplicationId	RoleId	RoleName	Description
	5a7a2b73-2322-...	a31d2979-890d...	Client	NULL
	5a7a2b73-2322-...	039c8655-83e8-...	Admin	NULL
▶*	NULL	NULL	NULL	NULL

	UserId	RoleId
	d544c281-49a6-...	039c8655-83e8-...
▶*	NULL	NULL

The membership database has been greatly simplified in Visual Studio 2012 compared to earlier versions. It is however easier to use the ASP.NET configuration pages, although descriptions for roles could be entered directly.