CSE445/598

Tutorial on Deploying Web Services into ASU WebStrar Server

2025

Important links:

Map Network Drive: MND Access: \\webstrar.fulton.asu.edu\websiteX,
where X = 1, 2, 3, …, N is your server site number.

After service deployment: HTTP Access: http://webstrarX.fulton.asu.edu/page0/Default.aspx

Table of Contents

Foreword from CSE445/59 Instructor – Yinong Chen

Creating a Web server for a large class is unprecedented because of the complexity, the security concerns of ASU ETS Department, and the cost of setting up and maintaining such a server. With numerous attempts and long hours of collaborative work between ETS staff, SCAI tech director (Lincoln), my TAs, and myself, we managed to get such a job done for the first time in Spring 2009. I owe all of them for their tireless effort and for having the server up and running. Numerous problems occurred in the first implementation in Spring 2009. In Fall 2021, the server is outsourced to AWS cloud computing environment. The server should work in the same way as it was hosted by ASU ETS. The professionally hosted server should perform even better. However, I urge all of you for your patience if the sites are slow or is not working as well as we have planned and wished. Please plan ahead and do not try to deploy your assignments in the last hour or in the last minute.

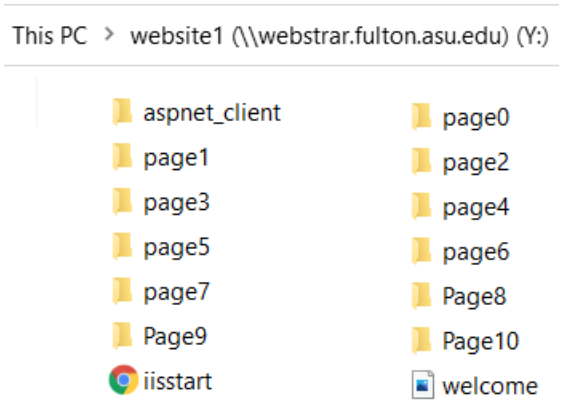
Dealing with a server is much more complex that running your program on your local computer. But this is a process that you need to learn and experience in this course. Please use the site responsibly at all the time and report any problems to me and to the TA immediately.

1. WebStrar Server and its Sites

The Web server WebStrar is hosted on AWS Cloud using the latest technology of Server on Demand (SoD) based on virtualization, server farm, and cloud computing concepts.

To test if your server work, you can enter this address: http://webstrarX.fulton.asu.edu/
where, X is the server site number assigned to your team. It should show you "Welcome IIS 7", which means your server is up and running. Note, http user's access to the server is different from developer's access: \\webstrar.fulton.asu.edu\websiteX, which will be explained later in this document.

WebStrar supports multiple independent sites, and its individual sites are http://webstrarX.fulton.asu.edu/pageY, where, X should be replaced by the site number 1, 2, 3, …, N that has been assigned to your team, and Y is the page number. Each site has 11 pages: page0, page1, page2, …, page10, as shown in the following figure.



Each group in CSE445/598 will be assigned a server site, and each member will be given the access to all the 11 pages (folders) in the site assigned. The team members can decide how they want to use the pages.

The suggestion is to deploy the main page that contains the description and the table of services outside the folders. In each page, you can deploy multiple Web services (service operations) or Web applications. Each member must plan properly to pack all the required services and elective services into three papers. If you have three members in the team, each member should have at least three folders. How do you deploy all your services into the three folders? You can use one folder for the TryIt test pages, one page for all the required services, and one page for all the elective services. In order to simply the deployment effort, you may need to wrap multiple services (operations) into one service, with each service containing multiple operations, and combine multiple TryIt pages into one page.

You can access each page from a Web browser. For example, if the site number is X = 15 and the page number is Y = 7, you can access the application deployed in the site at:

http://webstrar15.fulton.asu.edu/page7/Default.aspx, where the .aspx page or .svc has to be directly in one of the existing pages, page0, page1, page10, and it cannot be in a sub-folder that you crated.

Notice that these 11 folders (pages) have been converted into IIS **Application** folders that support .svc services and ASPX applications by the administrator in ASU ETS. We only have FTP (MND) access to the servers, and we cannot remote login to administrate the servers. Thus, please NEVER delete any pre-created folders/pages in the site. They are registered with WAS. If you delete a folder, your service will never work, even if you recreate the folder. The recreated folder is NOT registered to WAS. The entire site has to be reset in order to restore the site. It may take a few weeks for the administrator perason to complete the additional task, depending on the length of the job queue. I do not have admin access to the server.

When you deploy the service, you do not deploy the service-root folder. You copy the files and sub-folders in the service-root folder into one of the pre-created folders.

All the web services and web applications deployed into the server are accessible by the public. The services you deployed into WebStrar may be copied to another permanent server at ASU, as WebStrar will be reset for the next class at the end of the semester. You will still be the owner of the services and applications, but the services will be accessible by the public for free forever. If you do not want your services to be copied into another server at the end of the course, you must let the TA and instructor know and we will have the services deleted.

Before you put your Web services or Web applications into the server, you should test them on your own IIS (or at least on your .Net Development server IIS Express. In order to know if your IIS work, you need to test your IIS by putting a simple application in the IIS. For example,

You can put the following index.html file IIS wwwroot folder. You can also place the same file into the WebStrar root folder and the other pages (page0 through page7) folders where you plan to deploy your services:

```
<html>
<B>Today is a good day</B>
```

```
</html>
```

Then, you can put the following Default.aspx file with an embedded computation statement into the folders:

```
<html> <body>
Hello CSE445<br>
<B><% Response.Write(DateTime.Now.ToLongDateString())%> is a good day</B>
</body> </html>
```

You can use any text editor to type this code and name it Default.aspx.

WebStrar support .Net Framework 4.5 and above applications. Use Visual Studio 2022 to create your .Net Framework 4.5 applications.

2. Setting Up Connection on Windows

2.1 ASU VPN Connection sslvpn

To connect to Webstrar as a developer, you must install and use ASU VPN ssslvpn. You can search ASU website for sslvpn or ASU VPN

You will find how to install ASU VPN at: https://asu.my.salesforce-sites.com/kb/articles/FAQ/How-do-I-Install-Cisco-AnyConnect-SSLVPN. Install and enter the connection address: sslvpn.asu.edu

Then, you can use your ASURITE ID or ASUAD ID and password to connect to connect. The second password required is: **push**. After you connected to ASU VPN, you can access WebStrar server.

You need to use the two-phase DUO verification. A second password is required, which is simply **push**, and it pushes the DUO verification to your phone. Of course, you need to install the DUO App on your phone. Instruction for using ASU DUO can be found at: https://getprotected.asu.edu/services/identity-and-access-management/duo-two-factor-authentication

2.2 Map Network Drive

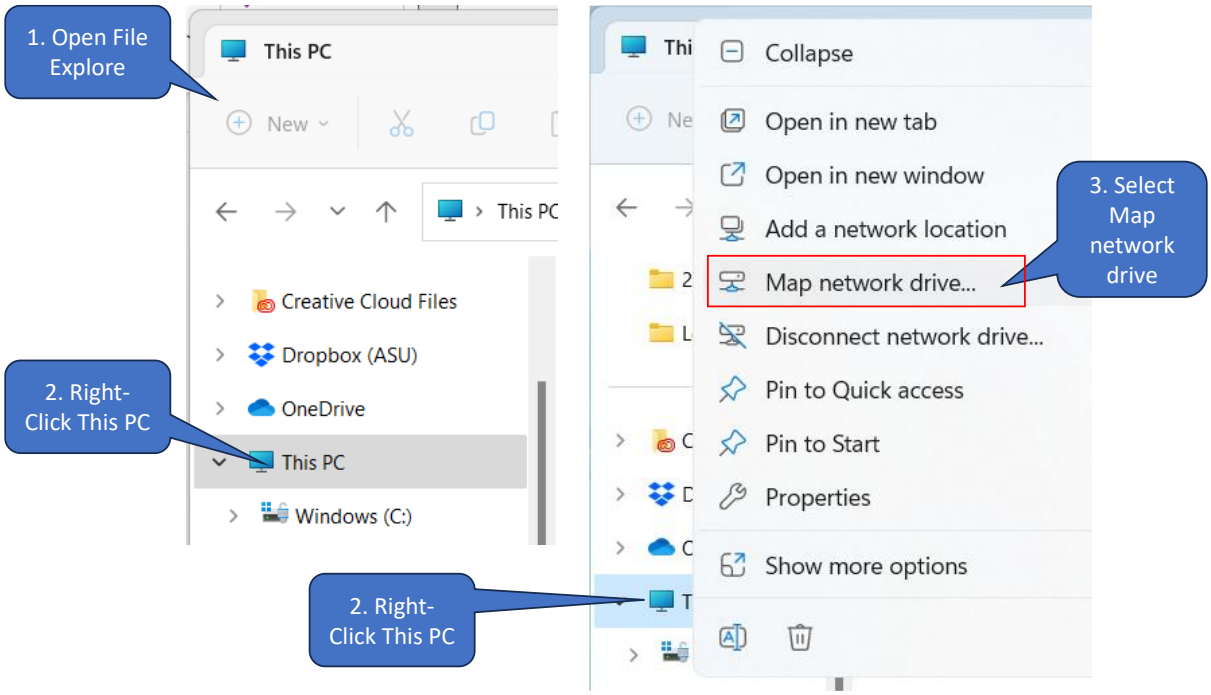
Connect to ASU VPN before mapping your network drive.

You can create a shortcut to your WebStrar server site, or any hard drive shared on your network by mapping that location. When you map a network drive, it will show up as a new drive under This PC in File Explorer, so you can quickly access the shared files by dragging and dropping, just like you access your local hard drive. Here's how to make that network drive shortcut in Windows.

For Windows 11, you follow these steps:

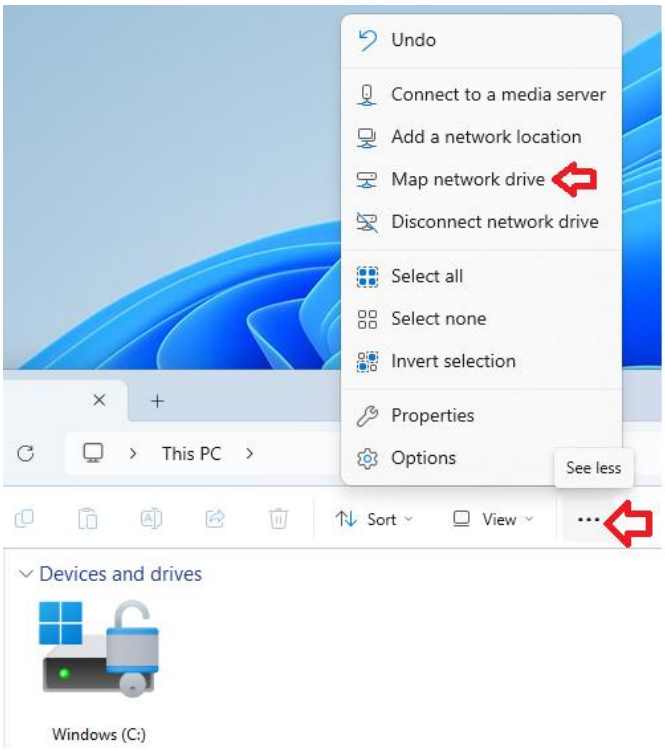
(1) Open file explorer and click This PC;
(2) Click the three dots and then select Map network drive, as shown in the following screeshot.

For Windows 11, you use the following steps:



Another possible configuration of your Windows 11 could look as follows:

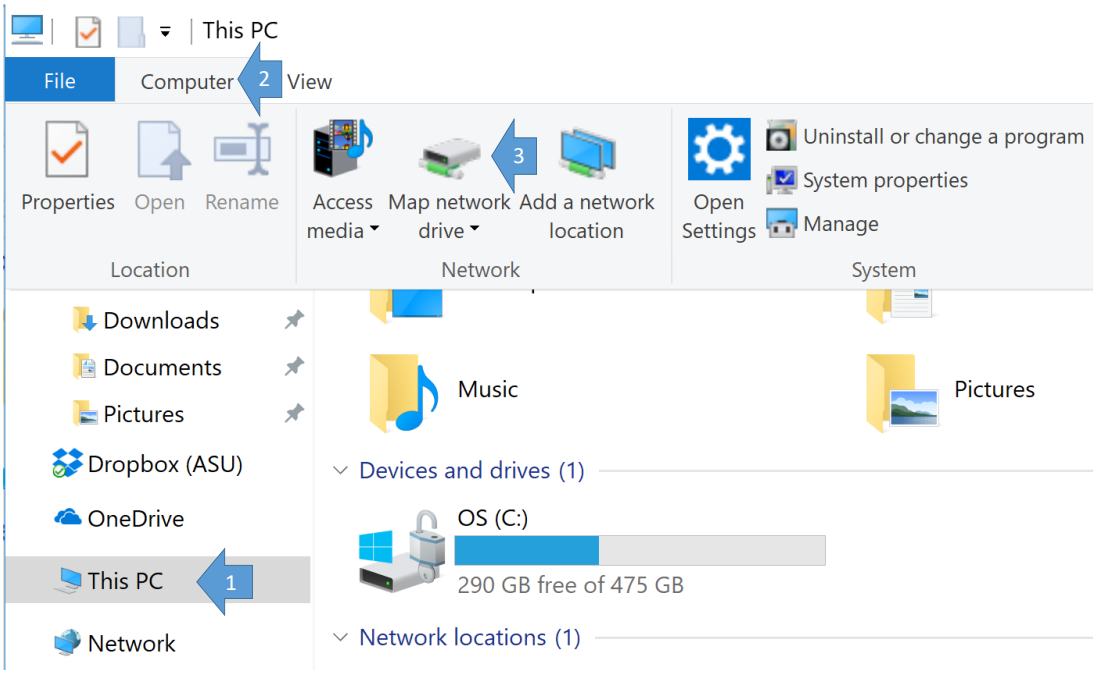
(1) Open file explorer and click This PC;
(2) Click the three horizontal dots, as shown in the figure below.
(3) Select Map Network Drive.



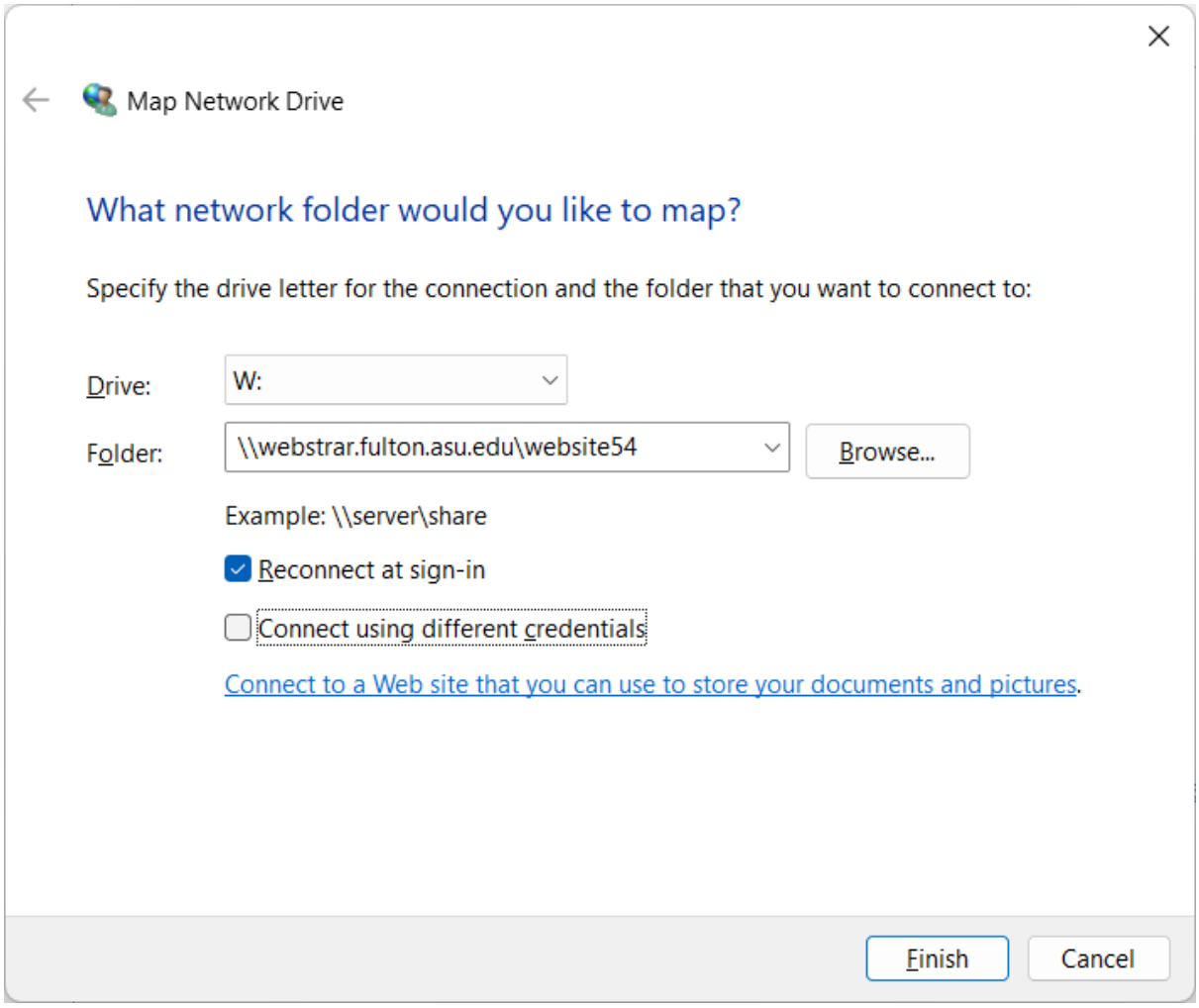
For Windows 10, you follow these steps:

(1) Open file explorer and click This PC;
(2) Click Computer
(3) Click Map Network Drive.

These steps are marked in the following figure.



After selecting Map network drive, you will see this window:



Enter your server site: \\webstrar.fulton.asu.edu\websiteX, where X = 1, 2, 3, …, N, and then click Finish.

For connecting for the first time, you may need to use the following process.

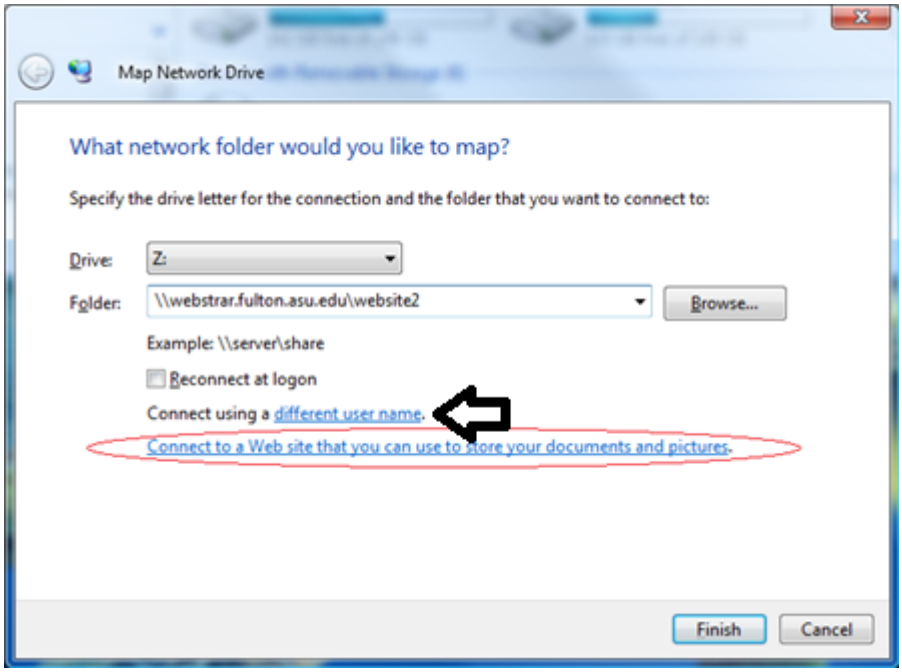
1. As shown in the "**Map Network Drive**" dialog, click on the "Connect to a Web site that you can use to store your documents and pictures" link.

   The network address likes the form:

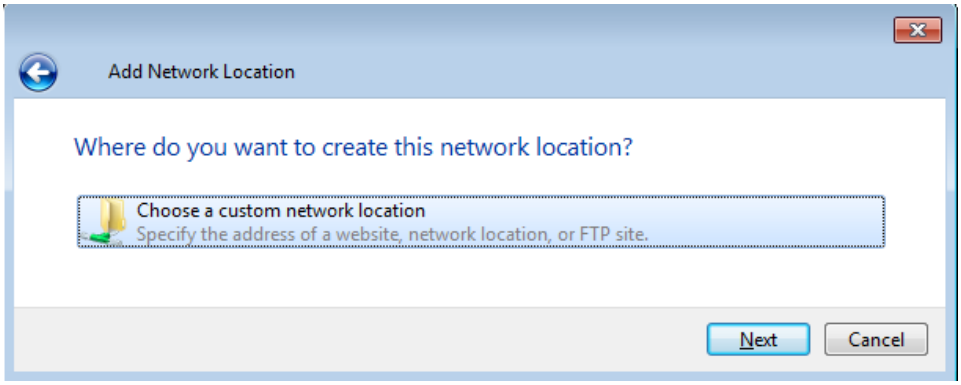
   \\webstrar.fulton.asu.edu\websiteX, where X = 1, 2, 3, …, N

   You should replace X with the site number assigned to your group.

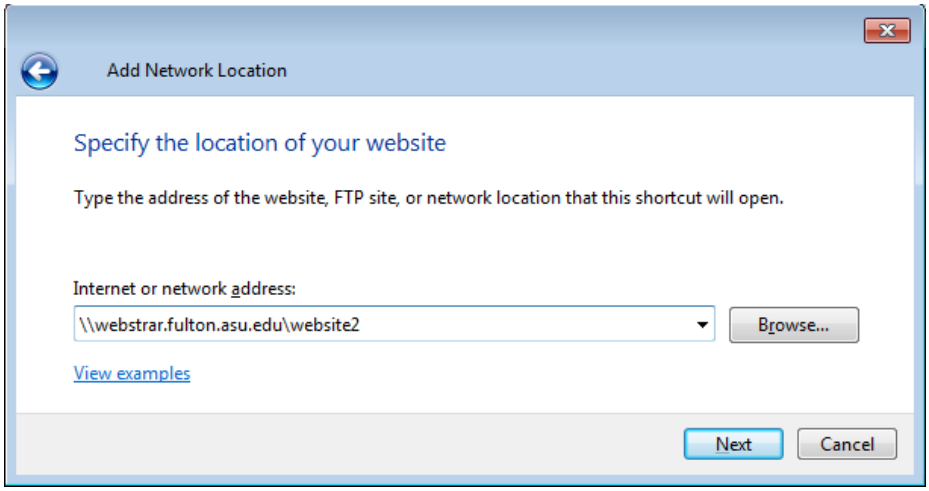
2. Type a name for the network location.
3. Choose: Connect use a different user name



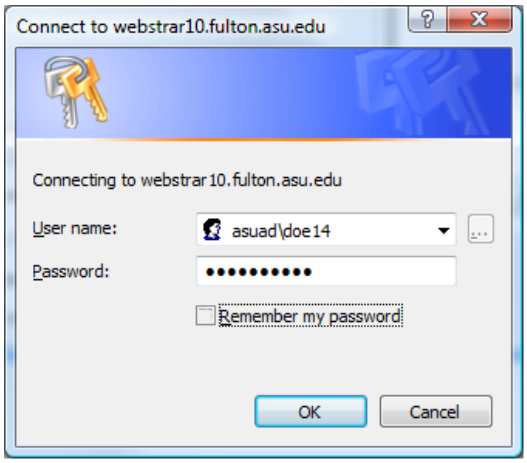
Double click to “Choose a custom network location”:



Enter the address again:

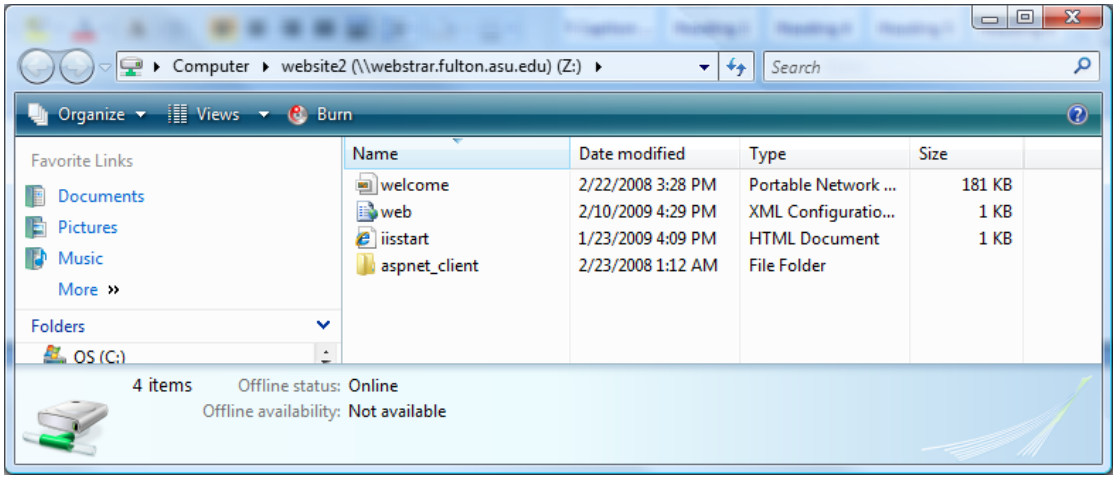


You will be asked to enter your ID and password. Assume your ASUAD ID is doe14, then, you should enter asuad\doe14, as shown below. Note, the User name is your MyASU login ID and password. DO NOT use your numerical student ID number.



- To maintain security, **do not** check "Remember my password."
- **Note:** You may be prompted to enter your User name and password two or three times.

Once you see the files, you are connected to the server site.



3. Service Publishing and Deployment

If you need to deploy web services and a web application that calls (accesses) the services, you must deploy the services first. You can obtain the service addresses (URLs). Then, you remove the localhost services and add the server-deployed services addresses into your web application. The deployment of web services and web applications must follow the following guidelines.

The folders page 0, page1, …, page10 have been converted to application and are registered to WAS (Web Administrative Services) of IIS. DO NOT delete these folders. You copy your service files into one of the folders.

If you do not have enough folders to deploy all your services and TryIt pages, wrap all the WSDL services in one big service and make each service an operation of the big service. Wrap all the RESTful services in one folder. Wrap all the TryIt pages into one page and deploy it in one folder.

3.1 Deploy Source Code

Copy (Deploy) **the files** in the solution folder into a pre-created folder in WebStrar, such as page0, page1, …, page10. Do not copy the entire root folder into pageX. Another layer of folder will make your services not accessible.

3.2 Deploy Pre-Compiled Code

You will use this option, only if the option in 3.1 does not work.

- Choose "Build → Publish Web App → Click on Custom and enter a New Custom Profile. Choose a name for your services. → Choose Publish method and choose File System and choose a target location to save your precompiled application. Do not choose directly to **publish** to server option. It would override the folders that had been registered to IIS WAS. Then, your server would not work. If this happened, you would have to file a ticket to ETS to reset your server. It could take weeks to have the request completed. The instructor does not have the privilege to reset the server.
- Choose a location and a folder, e.g., "PreComplierService" outside your project folder to publish the files.
- Copy (Deploy) **the files** in the folder PreComplierService into a pre-created folder in WebStrar, such as page0, page1. Do not copy the entire folder in pageX. Another layer of folder will make your services not accessible.

4. Trouble Shooting

4.1 HTTP Status Code Error Message

A server normally has different restrictions and limitations from you localhost. If you receive an error when accessing the services that you have deployed in the server, you can check the HTTP status code and error message code at: http://support.microsoft.com/kb/943891

For example:

400 - Bad request. The request could not be understood by the server due to malformed syntax. The client should not repeat the request without modifications. This means that your request syntax, most likely, the way you express the variables, is incorrect. You must follow the UriTemplate that you defined in the operation attributes to make the request.

An error code can have sub code. For example, 404 is an error “Not Found”. If it gives sub code, it may have different meanings. For example:

404.13 - Content length too large,

404.14 - Request URL too long,

404.15 - Query string too long

Some of limitations, such as input and output sizes, can be defined in the service or application configuration. You can change the value to bigger value.

```
<requestLimits maxAllowedContentLength="1000000" />
```

4.2 Uploading too Slow

If your network connection is slow, and it takes a long time for you to upload a Web service or application to WebStrar (or to Canvas), you can take the following steps to speed up uploading.

1. Zip your uploading the umbrella folder for checking its file size.
2. If the size is very large, you must have included many unnecessary libraries and utilities into your folder.
3. Use trial and error to delete the suspected unnecessary files. For example, the folder may have included language packages to support many different languages, and you just need to keep English language for the deployment.
4. After some deletions, test your services or applications and make sure they still work on your localhost before you deploy them.

The above-mentioned method steps also apply for WebStrar deployment and Canvas submission.

4.3 Viewstate MAC Failed Error

If you receive the following message:

Validation of viewstate MAC failed. If this application is hosted by a Web Farm or cluster, ensure that <machineKey> configuration specifies the same validationKey and validation algorithm. AutoGenerate cannot be used in a cluster."

You can follow these steps to create a machine key in your Web.config file:

Step 1:

Copy the following powershell script file and paste into a Notepad file.

```
# Generates a <machineKey> element that can be copied + pasted into a Web.config f
ile.
function Generate-MachineKey {
  [CmdletBinding()]
  param (
    [ValidateSet("AES", "DES", "3DES")]
    [string]$decryptionAlgorithm = 'AES',
    [ValidateSet("MD5", "SHA1", "HMACSHA256", "HMACSHA384", "HMACSHA512")]
    [string]$validationAlgorithm = 'HMACSHA256'
  )
  process {
    function BinaryToHex {
        [CmdLetBinding()]
        param($bytes)
        process {
            $builder = new-object System.Text.StringBuilder
            foreach ($b in $bytes) {
              $builder = $builder.AppendFormat([System.Globalization.CultureInf
o]::InvariantCulture, "{0:X2}", $b)
            }
            $builder
        }
    }
    switch ($decryptionAlgorithm) {
      "AES" { $decryptionObject = new-object System.Security.Cryptography.AesCrypt
oServiceProvider }
      "DES" { $decryptionObject = new-object System.Security.Cryptography.DESCrypt
oServiceProvider }
      "3DES" { $decryptionObject = new-object System.Security.Cryptography.TripleD
ESCryptoServiceProvider }
    }
    $decryptionObject.GenerateKey()
    $decryptionKey = BinaryToHex($decryptionObject.Key)
    $decryptionObject.Dispose()
    switch ($validationAlgorithm) {
      "MD5" { $validationObject = new-object System.Security.Cryptography.HMACMD5
 }
      "SHA1" { $validationObject = new-object System.Security.Cryptography.HMACSHA
1 }
      "HMACSHA256" { $validationObject = new-object System.Security.Cryptography.H
MACSHA256 }
```

```
      "HMACSHA385" { $validationObject = new-object System.Security.Cryptography.H
MACSHA384 }
      "HMACSHA512" { $validationObject = new-object System.Security.Cryptography.H
MACSHA512 }
    }
    $validationKey = BinaryToHex($validationObject.Key)
    $validationObject.Dispose()
    [string]::Format([System.Globalization.CultureInfo]::InvariantCulture,
      "<machineKey decryption=`"{0}`" decryptionKey=`"{1}`" validation=`"{2}`" val
idationKey=`"{3}`" />",
      $decryptionAlgorithm.ToUpperInvariant(), $decryptionKey,
      $validationAlgorithm.ToUpperInvariant(), $validationKey)
  }
}
```

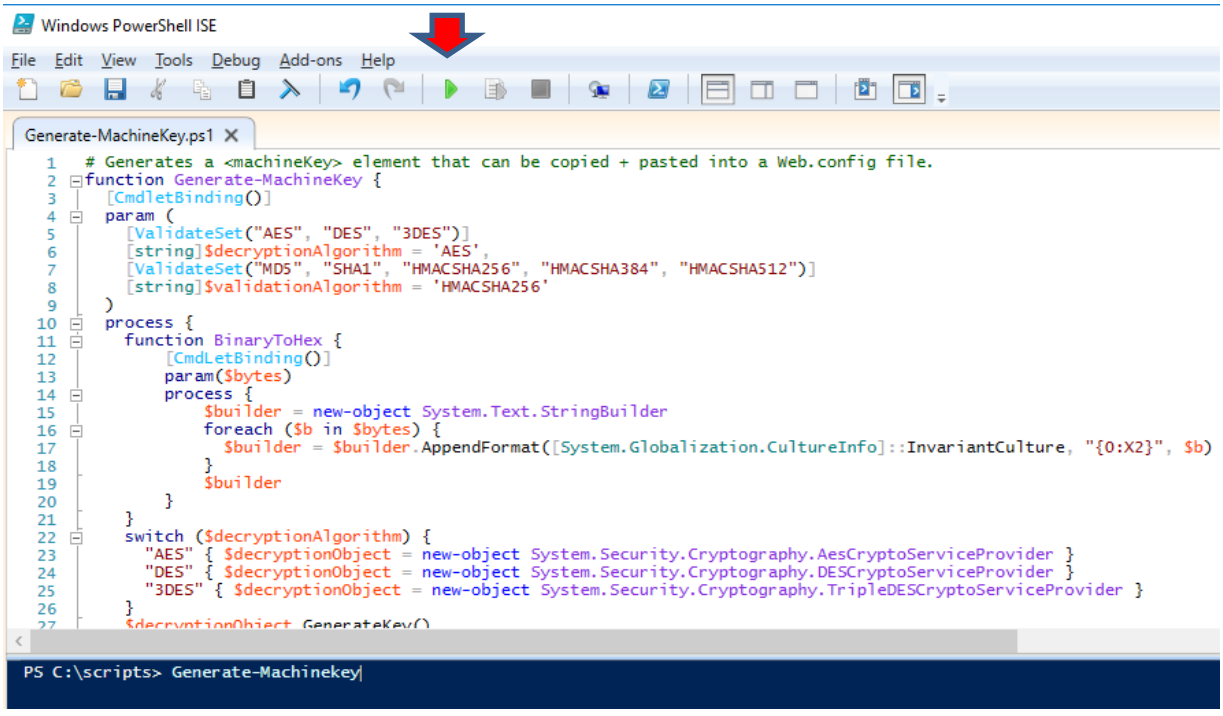
Step 2:

Save the file in the name **Generate-MachineKey.ps1**, and save it into a directory, for example: C:\scripts

Make sure the name is NOT **Generate-MachineKey.ps1.txt**

Step 3:

Right-click the file and choose Edit. It should be open by Windows Powershell ISE Editor, as shown in the following screenshot:



Step 4:

(1) Click the green triangle button  to run the script. Note, if the script failed to run, please read Step 5 and then come back to continue this step.

(2) Type the following command at the prompt:

PS C:\scripts> **Generate-Machinekey** <enter>

It should generate the machine key element that can be copy and paste into Web.config file:

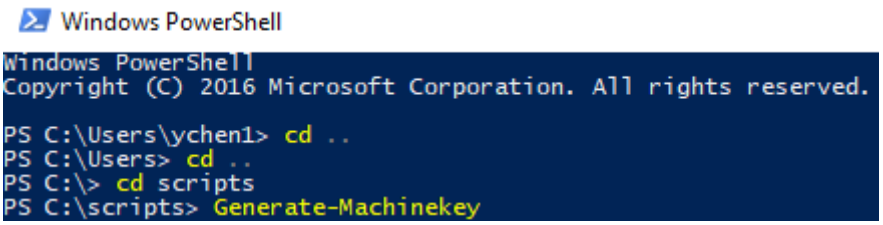
```
    <machineKey decryption="AES" decryptionKey="xxxxxxxxxxxxxxxxxxxxxxxxxx"
validation="HMACSHA256" validationKey="yyyyyyyyyyyyyyyyyyyyyyyyyyyy" />
```

Copy and paste the element in the element <system.web> in Web.config file, as shown beblow:

```
  <system.web>
    <machineKey decryption="AES" decryptionKey="xxxxxxxxxxxxxxxxxxxxxxxxxx"
validation="HMACSHA256" validationKey="yyyyyyyyyyyyyyyyyyyyyyyyyyyy" />
  </system.web>
```

When you copy and paste, make sure no space or break are added into the keys.

Another option is to type PowerShell in Windows search box and start PowerShell. Then, use Unix/DOS commands to change directories to the location where you have your script Generate-Machinekey.ps1, as shown in the screenshot below:



If the above steps do not work out for you, please read the comprehensive solutions for all kinds of situations given at:

https://support.microsoft.com/en-us/kb/2915218?wa=wsignin1.0

Another link for resolving this problem is:

https://support.microsoft.com/en-us/kb/312906

You will need to add an explicit <machineKey> element to the application's Web.config file, the developer tells ASP.NET not to use the auto-generated cryptographic key. See Appendix A of the site (https://support.microsoft.com/en-us/kb/2915218?wa=wsignin1.0) for instructions on how to generate a <machineKey> element. After this element is added to the Web.config file, redeploy the application to each server in the farm. The valid values for validationKey should be between 20 and 64. It creates keys from 40 to 128 bytes in length. The output from the code is an entire <machineKey>element that you can copy and paste into a Machine.config file.

Step 5:

If you have a permission issue in in running your script, you need to run the following command first "Set-ExecutionPolicy RemoteSigned" in Windows PowerShell (as admin) in order to run the script given in the previous step. For more information on the command, read: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.security/set-executionpolicy?view=powershell-6

4.4 Setting Up Web.config File for Displaying .svc service

If you experience an error

404 - File or directory not found.

The resource you are looking for might have been removed, had its name changed, or is temporarily unavailable.

This error occurs even if the file is correctly deployed in the right place. You need to add a handler in the Web.config file:

```
<handlers>
     <add name=".svc" verb="*" path="*.svc"
type="System.ServiceModel.Activation.ServiceHttpHandlerFactory,
System.ServiceModel.Activation, Version=4.0.0.0, Culture=neutral,
PublicKeyToken=31bf3856ad364e35" />
    </handlers>
```

Which should be placed in the element <system.webServer>

```
<system.webServer>
   <modules runAllManagedModulesForAllRequests="true"/>
   <handlers>
     <add name=".svc" verb="*" path="*.svc" type="System.ServiceModel.Activation.ServiceHttpHandlerFactory,
          System.ServiceModel.Activation,
          Version=4.0.0.0, Culture=neutral, PublicKeyToken=31bf3856ad364e35" />
   </handlers>
   <directoryBrowse enabled="true"/>
 </system.webServer>
```

The reason for this addition is:

WebStrar is now hosted by AWS that uses new version of OS and IIS, which applies newer security settings. Running a Service.svc file from the server is not permitted by default and need to override the new security feature.

4.5 Setting Up Web.config File for Allowing Public Access

First, in order to allow anyone to access your Web service and Web application, the server must enable “Anonymous” access.

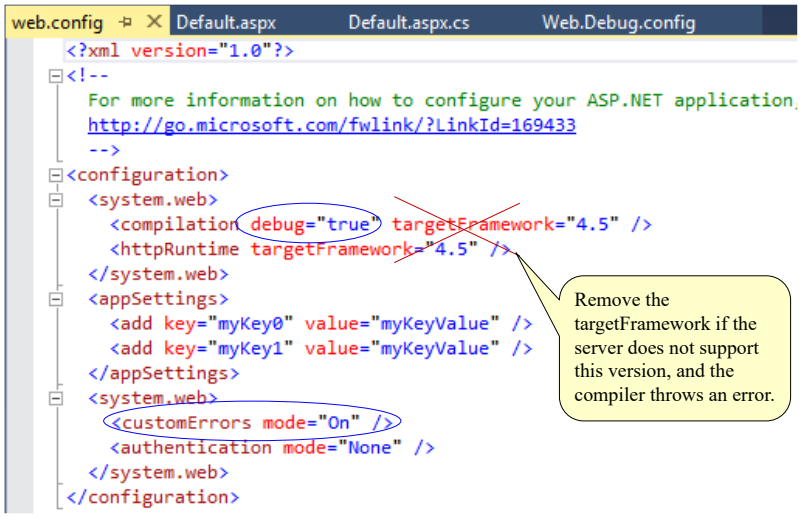
When you developing a Web service or a Web application in ASP .Net, a file called Web.config will be generated, which resides in the same directory as the service or the application. This file may contain

the authentication and authorization settings of the Web service or a Web application, depending on the initial setting of your system. For both applications and services deployed into the web server, you may need to define the permission to the public by editing the Web.config file. You must change the setting from <authentication mode="~~Windows~~"/> to <authentication mode="Forms"/> and add an authentication clause <allow users ="*" />, as shown below.

```
<configuration>
    <system.web>
        <authentication mode="Forms"/>
        <authorization>
            <allow users ="*" />
        </authorization>
        …
    </system.web>
</configuration>
```

We will discuss Forms-based security in Chapter 6. For this assignment, you may also set <authentication mode="None"/>

Furthermore, you need to define the correct .Net Framework in your configuration. For example, WebStrar support .Net 4.0, and you **cannot** use .Net 4.5 or 4.7, as shown in the figure below. WebStrar runs IIS 7, and you cannot use IIS 8.



4.6 Connection Issue

If you cannot access WebStrar, please try the followings steps one at a time.

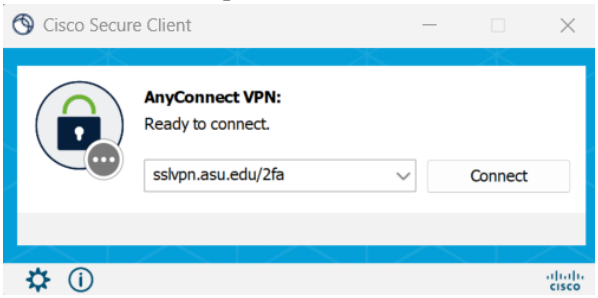
1) Make sure you install ASU VPN following the instructions at:

   https://asu.my.salesforce-sites.com/kb/articles/FAQ/How-do-I-Install-Cisco-AnyConnect-SSLVPN/?l=en_US&fs=RelatedArticle

   or

   https://math.asu.edu/resources/computer-resources/how-install-sslvpn-windowsmac

Then, you connect through ASU VPN: sslvpn.asu.edu/2fa. WebStrar will not connect without ASU VPN from off campus.



2) Delete the previous connected network drive and follow the tutorial to re-create the network drive. Once your IP address changes, the previous connection will no longer work.
3) If it still does not work, go to an ASU lab computer lab and plug in a physical Ethernet cable to your computer.
4) If it still does not work, use an ASU lab computer instead of your computer.
5) If it still does not work, change your ASUAD password and then try again. ASU requires you to change password after certain period of time. If you do not change, some network services will be disabled.
6) If it still does not work, check with your team members. If none of your team members cannot access the serve, then post the problem to the discussion board or send email to Charles Phillips <Charles.A.Phillips@asu.edu>. He is in charge of WebStrar management. WebStrar is managed him. Instructor and TA do not have more access privilege than you do.

4.7 Issues shared among students:

Below are the instructions that guide you step by step to set up and access the Web server as a developer.

==========

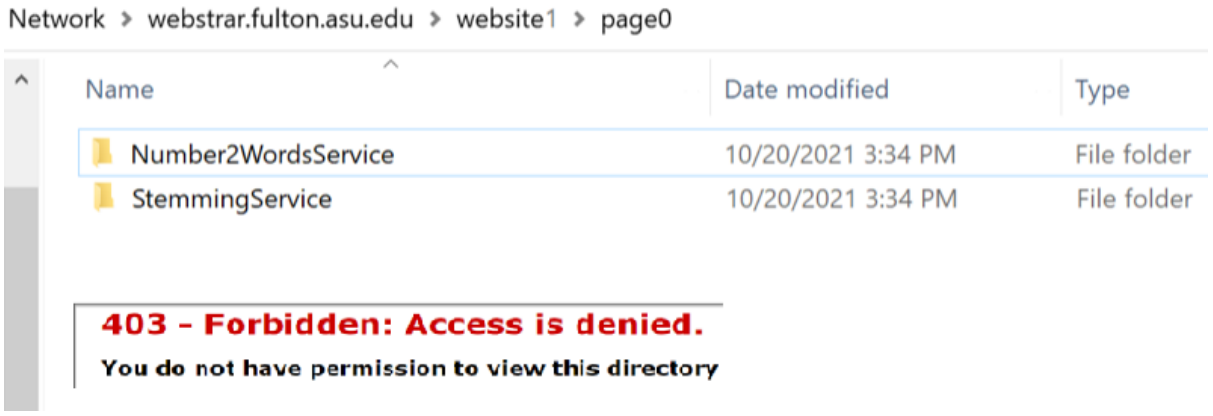
I was able to log in with the VPN. However, when I try to map the network drive, it cannot find the server.

I solved the problem based on this help page I used: https://thegeekpage.com/fix-windows-10-error-code-0x80070035-the-network-path-was-not-found/#Fix_1_%E2%80%93_Enable_SMB_V10

==========

I deployed the service as follow in the figure. When I access my service, I got:

403 – Forbidden: Access Denied.



The reason is that service file .svc cannot be in a sub-folder. The solution is:

(1) Put these two services in page0 and page1, instead in page0 only.
(2) Copy the contents of the folder Number2WordsService into page0 without adding a folder between the service file and page0.
(3) Do the same for StemmingService: Copy the contents of the folder StemmingService into page1 without adding a folder between the .svc service file and page1.
(4) When you deploy your .aspx pages, do the same: The .aspx page must be in page0, page1, …. You cannot add a folder between the page and the registered pages page0, page1, ….

If you do not have enough pages, you can combine your two services as one service with two operations. Then, you just need to use one page for the deployment.

==========

If you run into the following error when opening VPN anyconnect:

Failed to initialize connection subsystem

Go here and follow the directions at

http://christierney.com/2015/02/11/cisco-anyconnect-failed-to-initialize-connection-subsystem/

I was able to connect before, but then, I can no longer to connect.
I just figured out what my problem was. I went to the command prompt and ran:

net use

Which showed me that I still had a connection to the share drive from before that I thought I had removed. Then, I ran

net use /delete \\webstrar.fulton.asu.edu\website21
Then I was able to remap it again successfully via the instructions in the assignment document.

Note, if your computer uses wireless to connect to the internet, the Map Network Drive may not connect. In this case, try to use a computer with cable connection.

I finally got it to work yesterday but today the network drive was showing as disconnected. I did a "net use /delete" (DOS command) and then I was able to map it again, but just doing a Windows disconnect didn't seem to work to allow to re-add. If you are having issues, try these DOS commands:
"net use /delete \\webstrar.fulton.asu.edu\websiteXX" (XX is your site)
"net use Z: \\webstrar.fulton.asu.edu\websiteXX /user:asuad\<ASURITE>"(XX is your site and <ASURITE> is your ASUAD id.

Do not use wireless connection when you connect to WebStrar. It does not connect reliably, if it connects all.

I got a server error when I can call my mySite.aspx page in the folder page0: **"Server Error 404 - File or directory not found. The resource you are looking for might have been removed, had its name changed or is temporarily unavailable."** Even though the mySite.aspx page is there.

Solution: You may need to check your Web.config file to make sure the server redirect your request to the right place. For example:

Change Web.Config System.WebServer keys:

```
<system.webServer>
    <modules runAllManagedModulesForAllRequests="true">
      <remove name="UrlRoutingModule"/>
      <add name="UrlRoutingModule" type="System.Web.Routing.UrlRoutingModule,
System.Web, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a" />
   </modules>

    <handlers>
     <add name="UrlRoutingHandler" preCondition="integratedMode" verb
="*" path="UrlRouting.axd" type="System.Web.HttpForbiddenHandler,
System.Web, Version=2.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a"/>
    </handlers>
  </system.webServer>
```

You can also try to disable ViewState in the Web.config file:

- View State is used by the system. In .Net 4 and later, you can disable View State in Web.config by adding an attribute, e.g.:

  <asp:LoginView ID="HeadLoginView" runat="server" EnableViewState="false">

When running on local IIS, my web app's Timer works. However, when I upload my app to WebStrar, my Timer does not work anymore. When my Timer trys to tick, the following error is output to the browser's error console:

http://webstrar2.fulton.asu.edu/page2/bundles/MsAjaxJs?v=c42ygB2U07n37m_Sfa8ZbLGVu4Rr2gsBo7MvUEnJeZ81

GET 404 Not Found

I found a working solution here:

http://forums.asp.net/t/1885420.aspx?Asp+net+4+5+msajaxjs+and+webformjs+not+found+browser+looking+for+bundles

This also fixes the issue where CSS is not loaded on the WebStrar deployment of the web app.

Make sure that the debug parameter is false when you deploy the code, it's most likely set to true by default; switching it to false solved the problem for me.