

## To complete the deployment need to following three step:

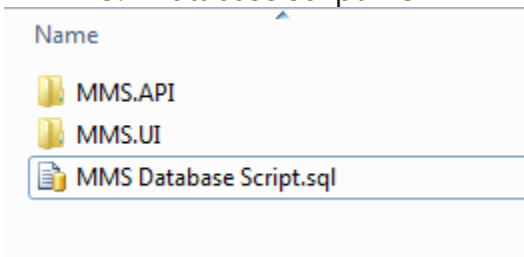
1. Database creation in MS SQL Server.
2. API (Backend) deployment in IIS.
3. UI (Frontend) deployment in IIS.

## Prerequisite

4. IIS configure in windows.
5. MS SQL Server in window.
6. Hosting bundle .net core 2.1 in window.

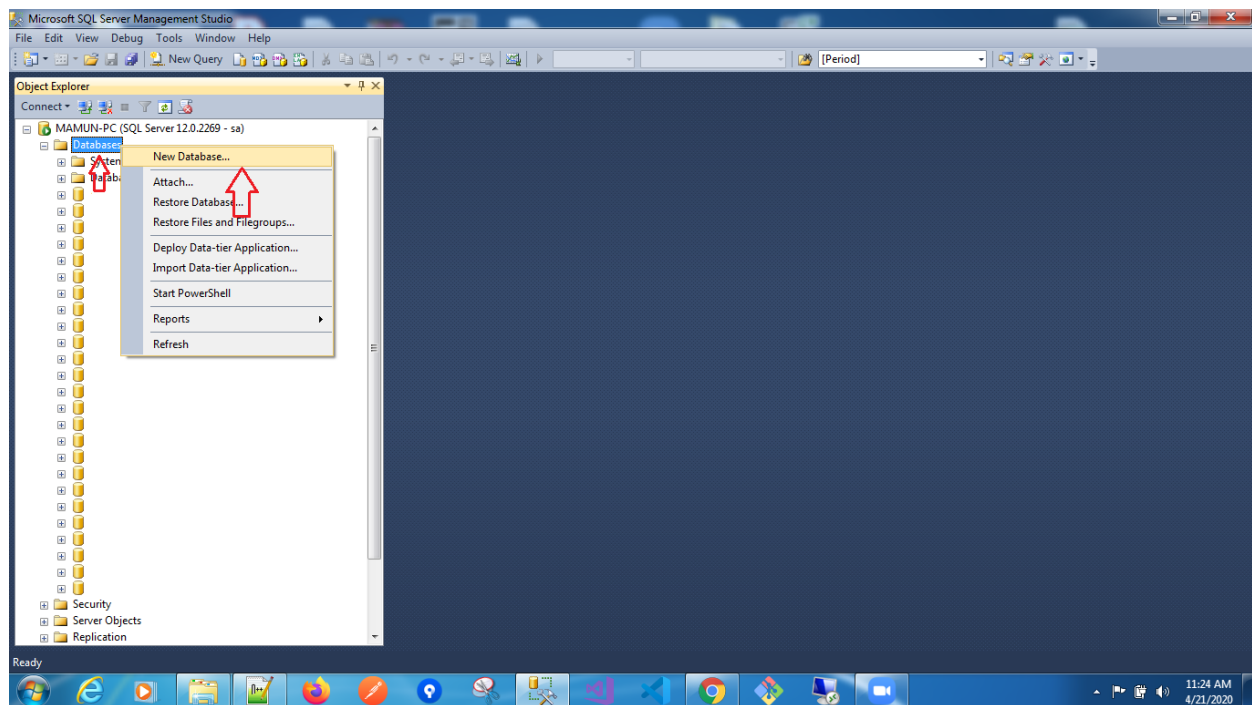
## Deployment Contents

1. MMS.API
2. MMS.UI
3. Database script file

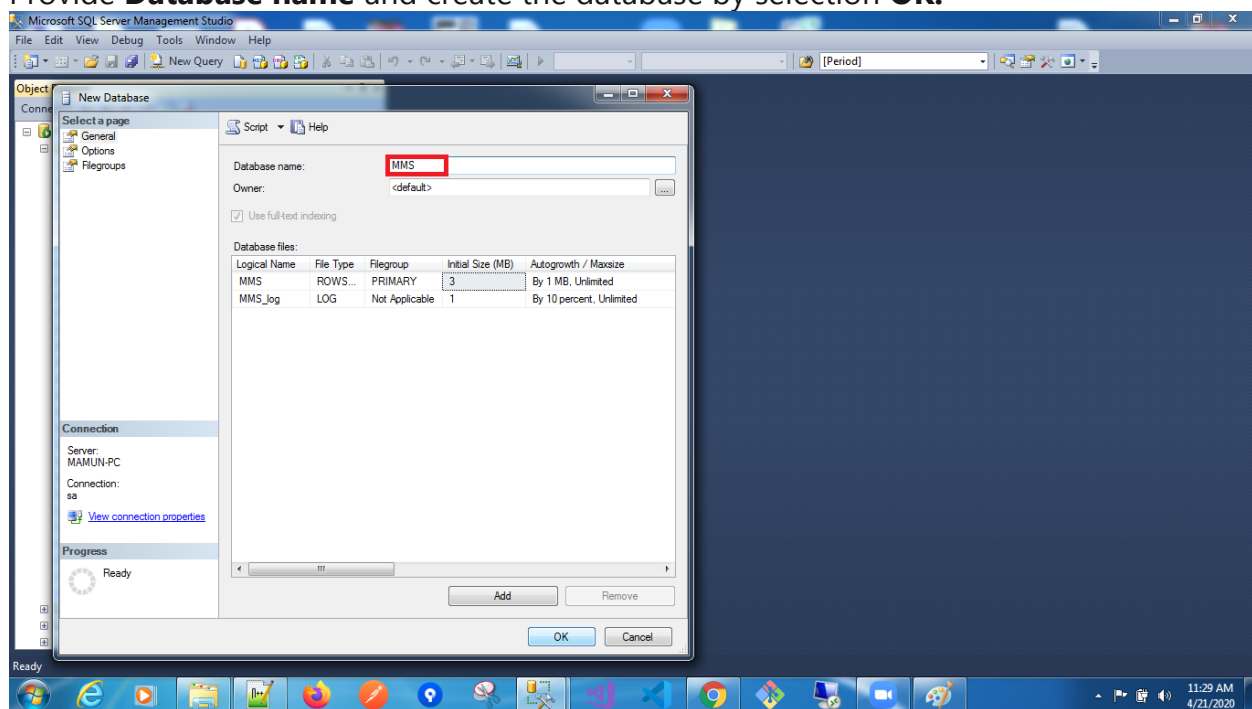


## Database create in MS SQL Server

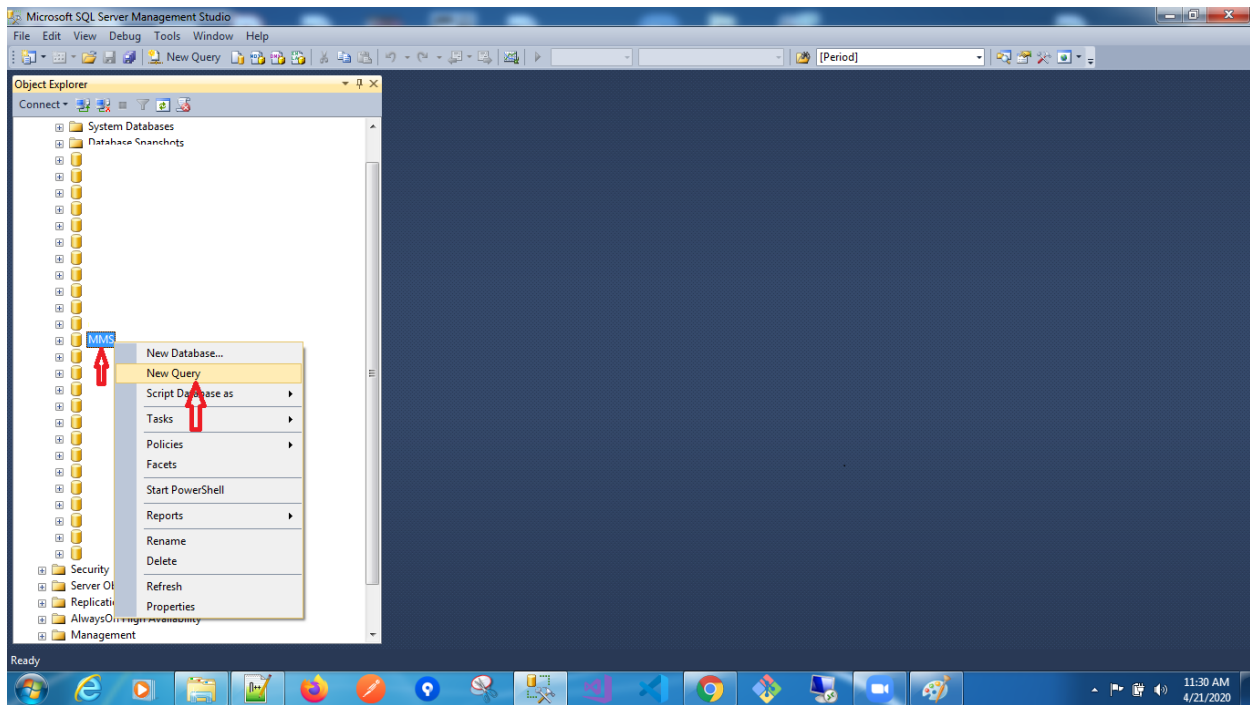
1. Open SQL Server Management Studio.
2. In Object Explore Manager right-click the **Databases** folder. Click **New Database** from the contextual menu.



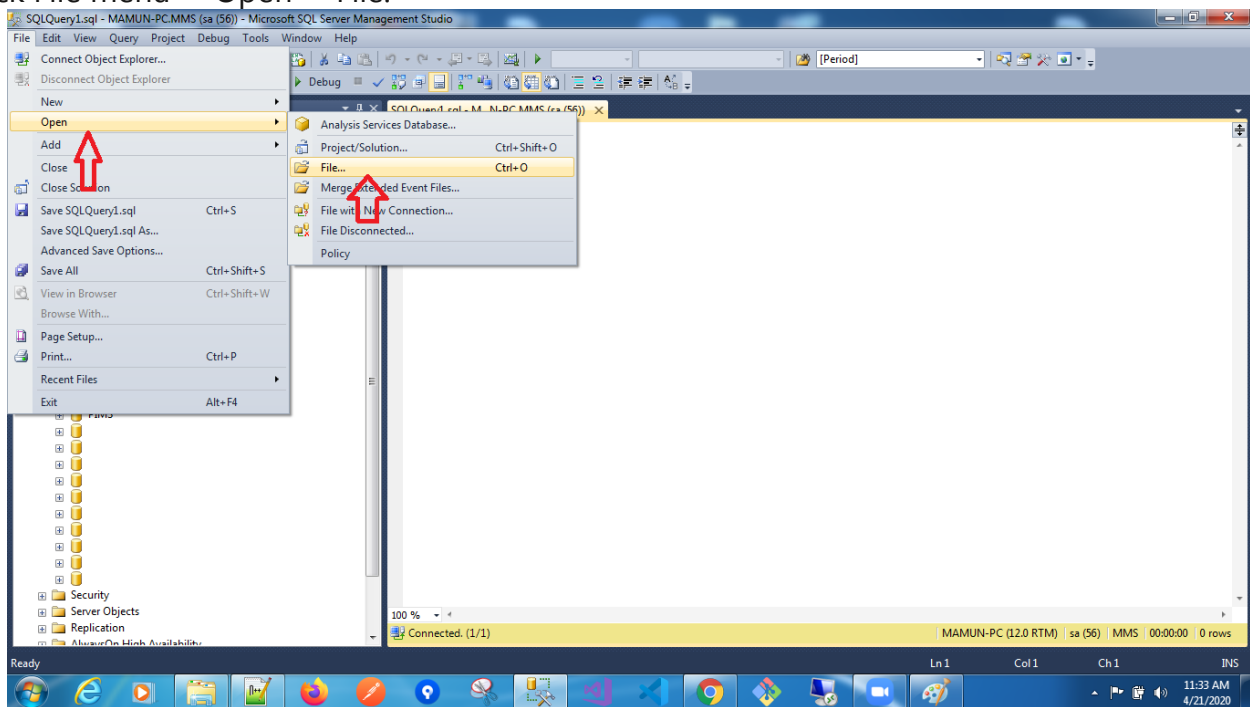
3. Provide **Database name** and create the database by selection **OK**.



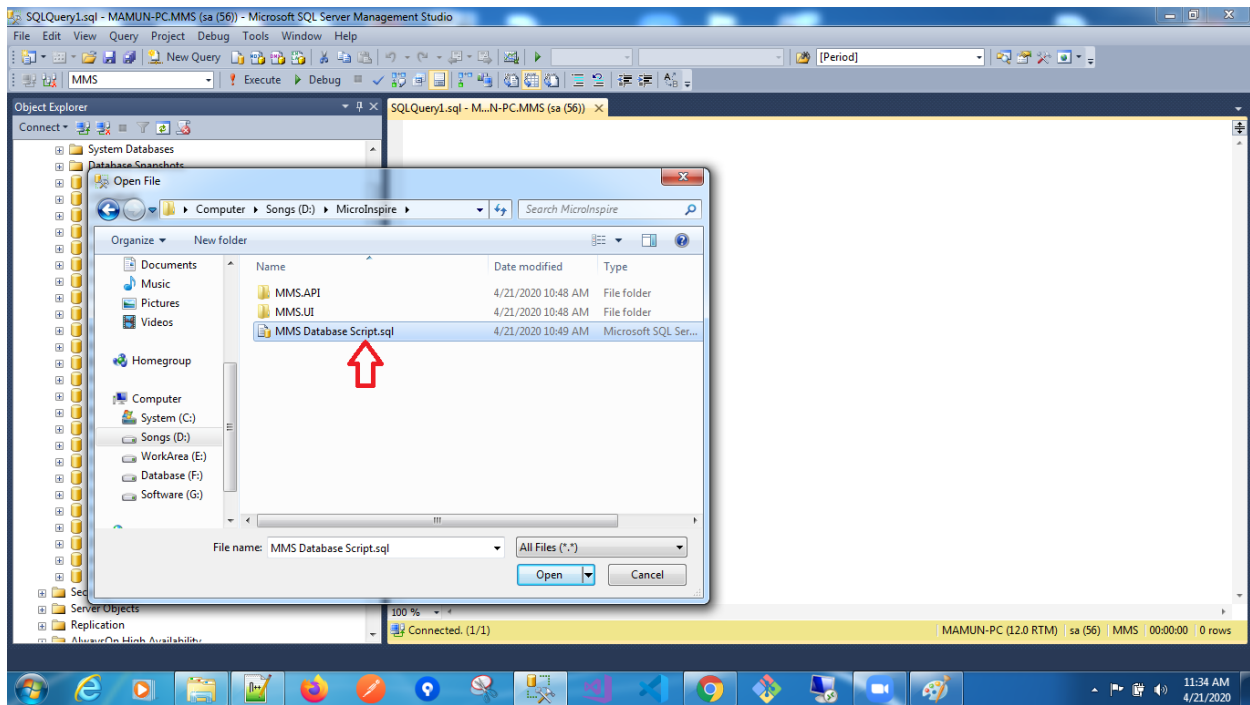
4. In Manager, open **Databases** node in the **Object Explore Manager** panel. Right-click the **Created Database**. Click **New Query** from the contextual menu.



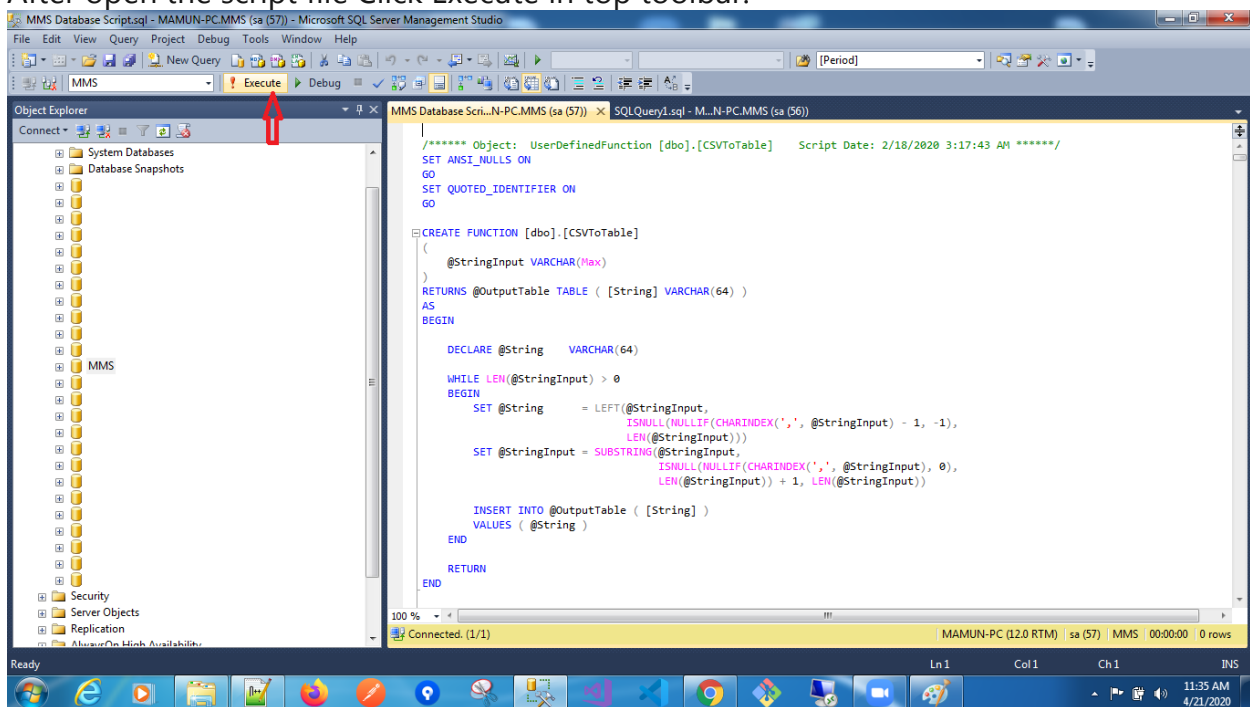
5. Click File menu=>Open=>File.



6. Select the provided database script file. Then click Open.



7. After open the script file Click Execute in top toolbar.



# Deploy API (Backend) in IIS

If not install hosting bundle .net core 2.1 in window.

Please download & install from bellow link:

<https://dotnet.microsoft.com/download/dotnet-core/2.1>

Download .NET Core 2.1 (Linux, macOS, Windows)

dotnet.microsoft.com/download/dotnet-core/2.1

This site uses cookies for analytics, personalized content and ads. By continuing to browse this site, you agree to this use. [Learn more](#)

Microsoft | .NET About Learn Architecture Docs Downloads Community [Get Started](#) All Microsoft

Release information	Build apps - SDK	Run apps - Runtime																		
<b>v2.1.17</b> <a href="#">Release notes</a> <b>Released</b> 2020-03-24	<p>ⓘ This release contains multiple SDKs. If you're using Visual Studio, look for the SDK that supports the version you're using. If you're not using Visual Studio, install the first SDK listed.</p> <p><b>SDK 2.1.805</b></p> <p><b>Visual Studio support</b> Visual Studio 2019 (v16.2 or later)</p> <p><b>Included runtimes</b> .NET Core Runtime 2.1.17 ASP.NET Core Runtime 2.1.17</p> <p><b>Language support</b> C# 7.3 F# 4.5</p> <table border="1"><thead><tr><th>OS</th><th>Installers</th><th>Binaries</th></tr></thead><tbody><tr><td>Linux</td><td><a href="#">Package manager instructions</a></td><td><a href="#">ARM32</a>   <a href="#">ARM64</a>   <a href="#">x64 Alpine</a>   <a href="#">x64 RHEL</a>   <a href="#">x64 Ubuntu</a></td></tr></tbody></table>	OS	Installers	Binaries	Linux	<a href="#">Package manager instructions</a>	<a href="#">ARM32</a>   <a href="#">ARM64</a>   <a href="#">x64 Alpine</a>   <a href="#">x64 RHEL</a>   <a href="#">x64 Ubuntu</a>	<p><b>ASP.NET Core Runtime 2.1.17</b></p> <p>The ASP.NET Core Runtime enables you to run existing web/server applications. <b>On Windows, we recommended installing the Hosting Bundle, which includes the .NET Core Runtime and IIS support.</b></p> <p><b>IIS runtime support (ASP.NET Core Module v2)</b> 12.1.20052.17</p> <table border="1"><thead><tr><th>OS</th><th>Installers</th><th>Binaries</th></tr></thead><tbody><tr><td>Linux</td><td><a href="#">Package manager instructions</a></td><td><a href="#">ARM32</a>   <a href="#">x64 Alpine</a>   <a href="#">x64</a></td></tr><tr><td>macOS</td><td></td><td><a href="#">x64</a></td></tr><tr><td>Windows</td><td><a href="#">x64</a>   <a href="#">x86</a>   <b>Hosting Bundle</b></td><td><a href="#">x64</a>   <a href="#">x86</a></td></tr></tbody></table> <p><b>.NET Core Runtime 2.1.17</b></p> <p>The .NET Core Runtime contains just the components needed to run a console app. Typically, you'd also install the ASP.NET</p>	OS	Installers	Binaries	Linux	<a href="#">Package manager instructions</a>	<a href="#">ARM32</a>   <a href="#">x64 Alpine</a>   <a href="#">x64</a>	macOS		<a href="#">x64</a>	Windows	<a href="#">x64</a>   <a href="#">x86</a>   <b>Hosting Bundle</b>	<a href="#">x64</a>   <a href="#">x86</a>
OS	Installers	Binaries																		
Linux	<a href="#">Package manager instructions</a>	<a href="#">ARM32</a>   <a href="#">ARM64</a>   <a href="#">x64 Alpine</a>   <a href="#">x64 RHEL</a>   <a href="#">x64 Ubuntu</a>																		
OS	Installers	Binaries																		
Linux	<a href="#">Package manager instructions</a>	<a href="#">ARM32</a>   <a href="#">x64 Alpine</a>   <a href="#">x64</a>																		
macOS		<a href="#">x64</a>																		
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <b>Hosting Bundle</b>	<a href="#">x64</a>   <a href="#">x86</a>																		

1. On the hosting system, create a folder to contain the app's published folders and files. In a following step, the folder's path is provided to IIS as the physical path to the app.
2. In IIS Manager, open the server's node in the **Connections** panel. Right-click the **Sites** folder. Select **Add Website** from the contextual menu.
3. Provide a **Site name** and set the **Physical path** to the app's deployment folder. Provide the **Binding** configuration and create the website by selecting **OK**:

**Add Web Site**

Site name:  Application pool:

Content Directory

Physical path:

Pass-through authentication

Binding

Type:  IP address:  Port:

Host name:

Example: www.contoso.com or marketing.contoso.com

☒ Start Web site immediately

Or <http://localhost:8086>

Add Web Site

Site name: MicroinspireApi Application pool: MicroinspireApi Select...

Content Directory

Physical path: D:\MicroInspire\MMS.API ...

Pass-through authentication

Connect as... Test Settings...

Binding

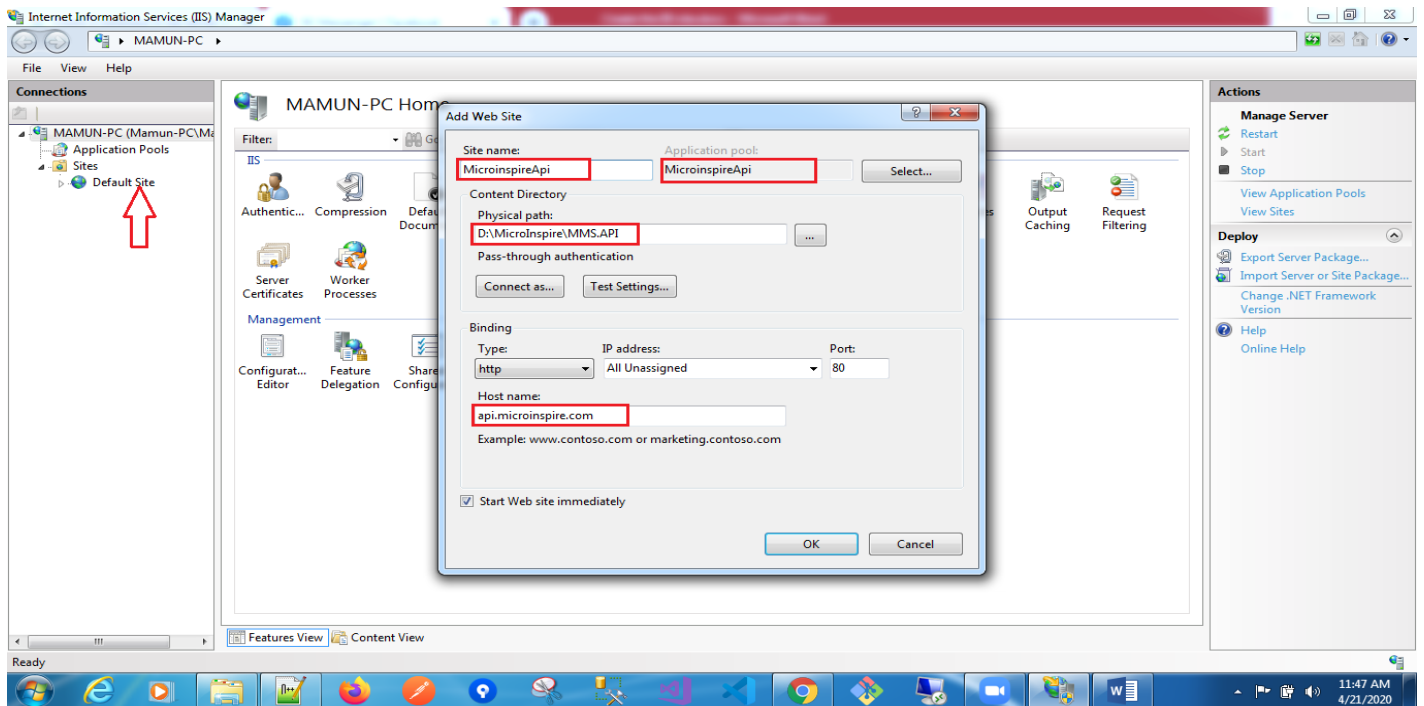
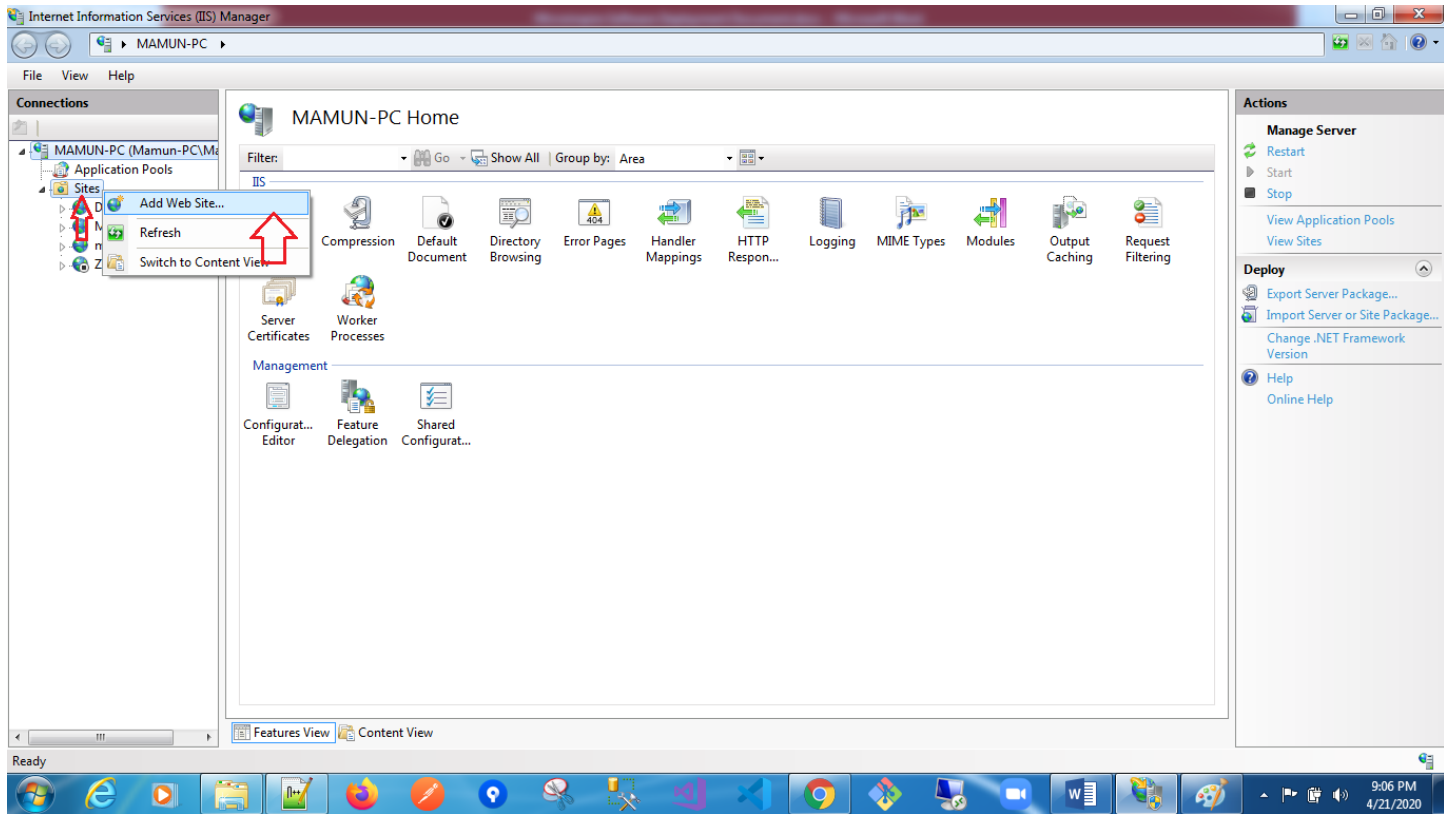
Type: http IP address: All Unassigned Port: 8086

Host name:

Example: www.contoso.com or marketing.contoso.com

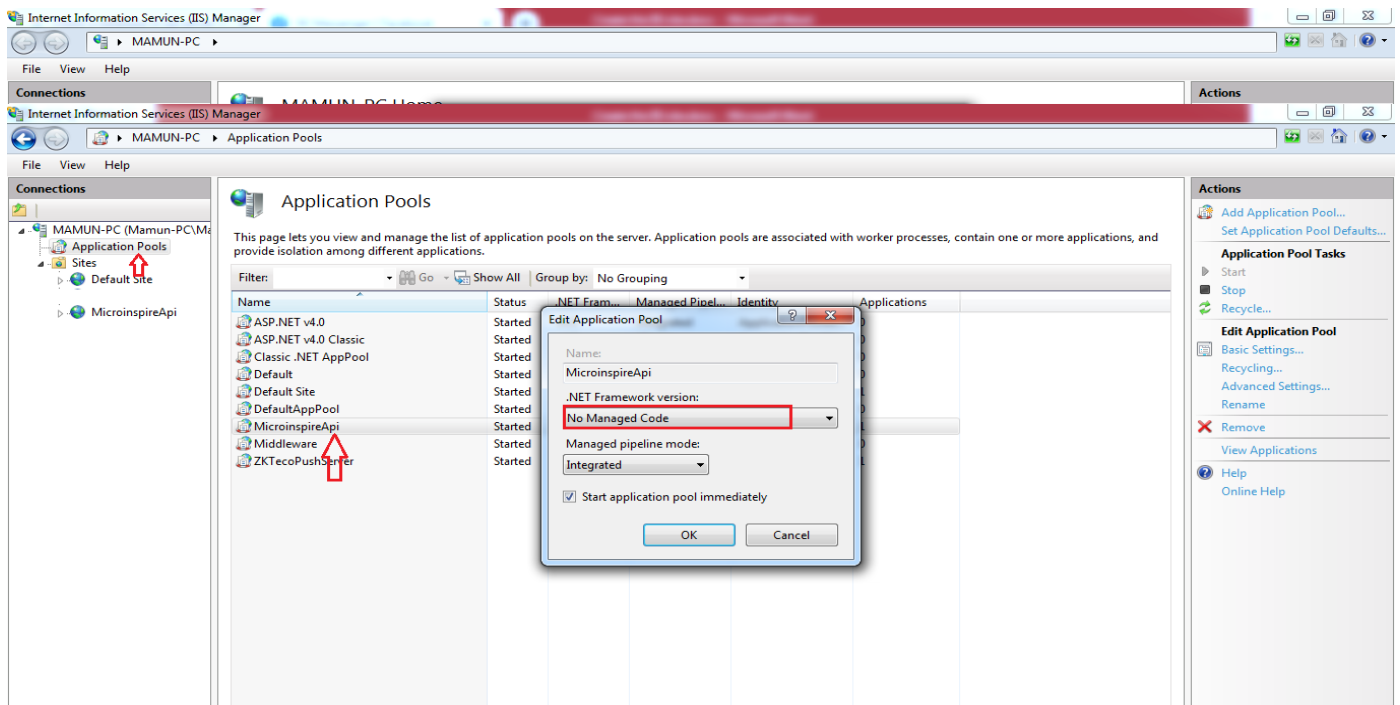
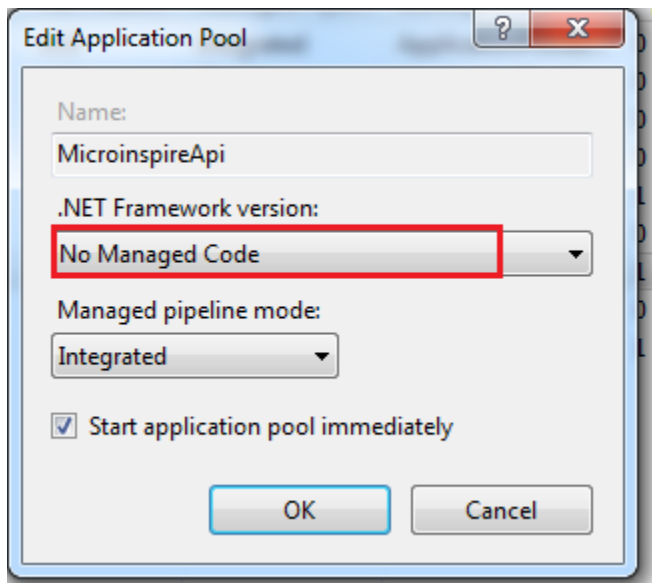
☒ Start Web site immediately

OK Cancel





4. Under the server's node, select **Application Pools**.
5. Right-click the site's app pool and select **Basic Settings** from the contextual menu.
6. In the **Edit Application Pool** window, set the **.NET CLR version** to **No Managed Code**:



## Deploy UI (Frontend) in IIS

1. On the hosting system, create a folder to contain the app's published folders and files. In a following step, the folder's path is provided to IIS as the physical path to the app.
2. In IIS Manager, open the server's node in the **Connections** panel. Right-click the **Sites** folder. Select **Add Website** from the contextual menu.
3. Provide a **Site name** and set the **Physical path** to the app's deployment folder. Provide the **Binding** configuration and create the website by selecting **OK**:
4. Select DefaultAppPool

The screenshot shows the 'Add Web Site' dialog box in IIS Manager. The dialog is titled 'Add Web Site' and contains the following fields and controls:

- Site name:** microinspireUI
- Application pool:** DefaultAppPool
- Content Directory:**
  - Physical path:** D:\MicroInspire\MMS.UI
  - Pass-through authentication:** (unchecked)
  - Connect as...** (button)
  - Test Settings...** (button)
- Binding:**
  - Type:** http
  - IP address:** All Unassigned
  - Port:** 80
  - Host name:** microinspire.com
  - Example:** www.contoso.com or marketing.contoso.com
- ☒ **Start Web site immediately**
- OK** (button)
- Cancel** (button)

Or <http://localhost:8085>

**Add Web Site**

Site name:  Application pool:

Content Directory

Physical path:

Pass-through authentication

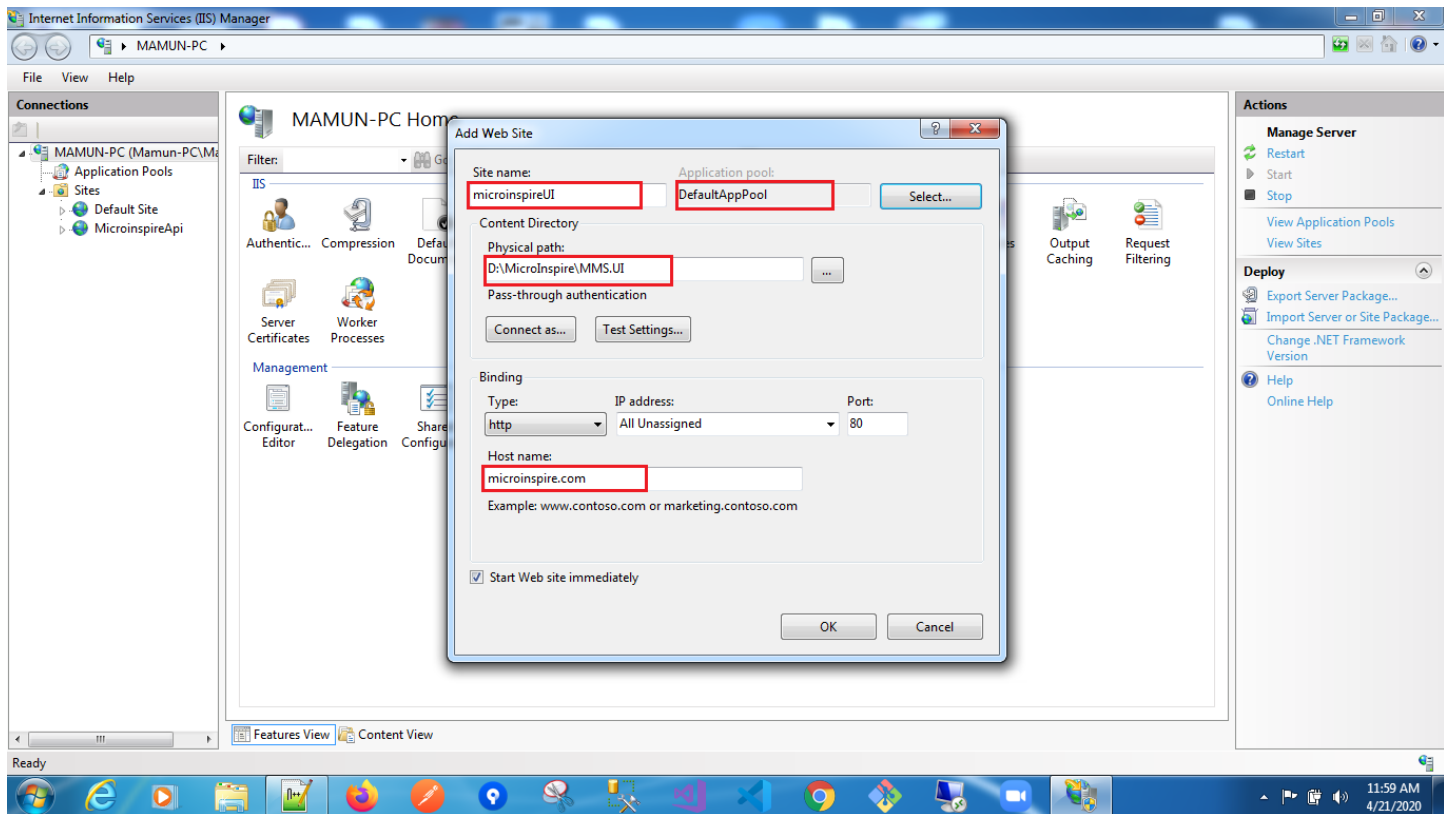
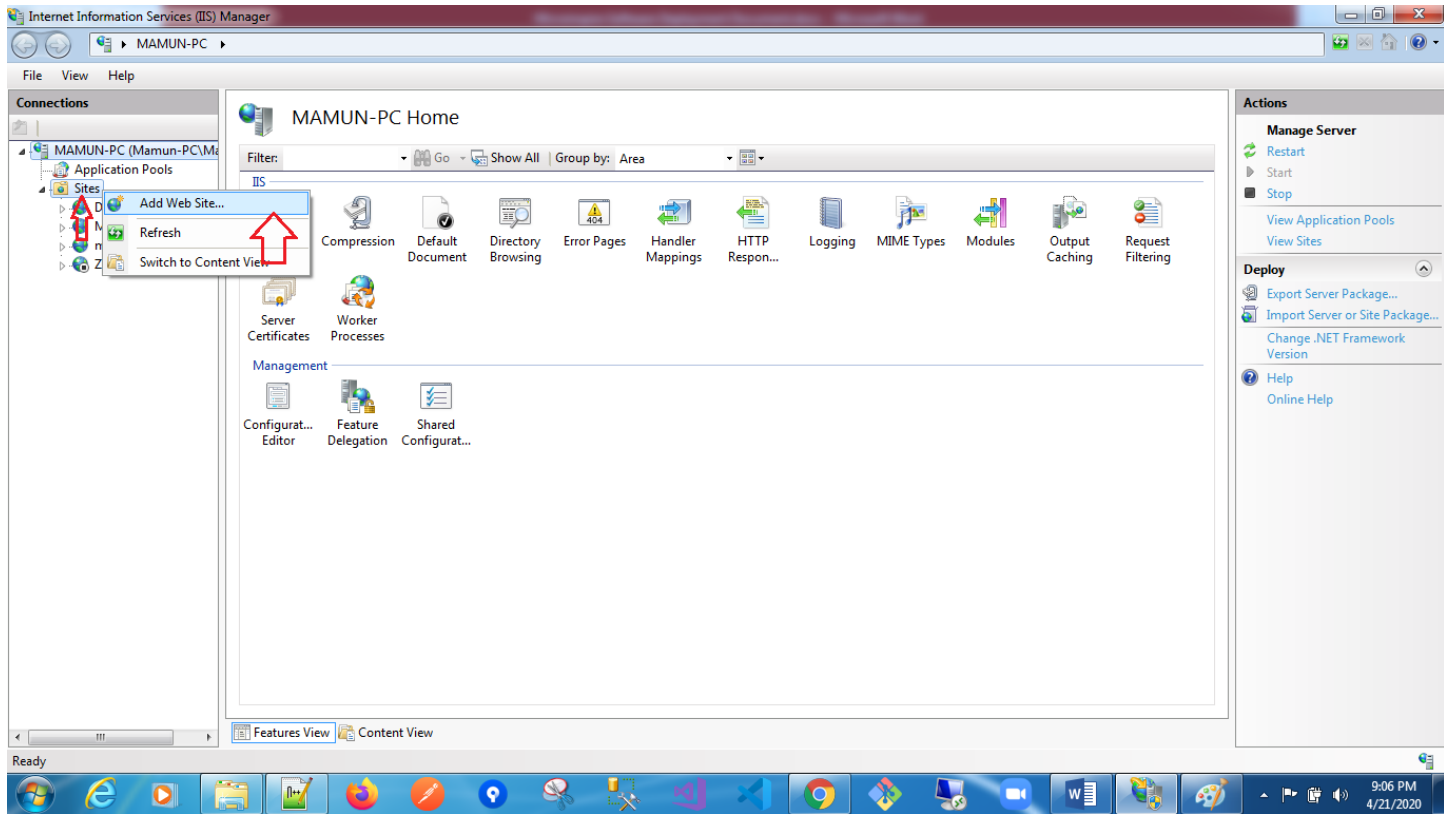
Binding

Type:  IP address:  Port:

Host name:

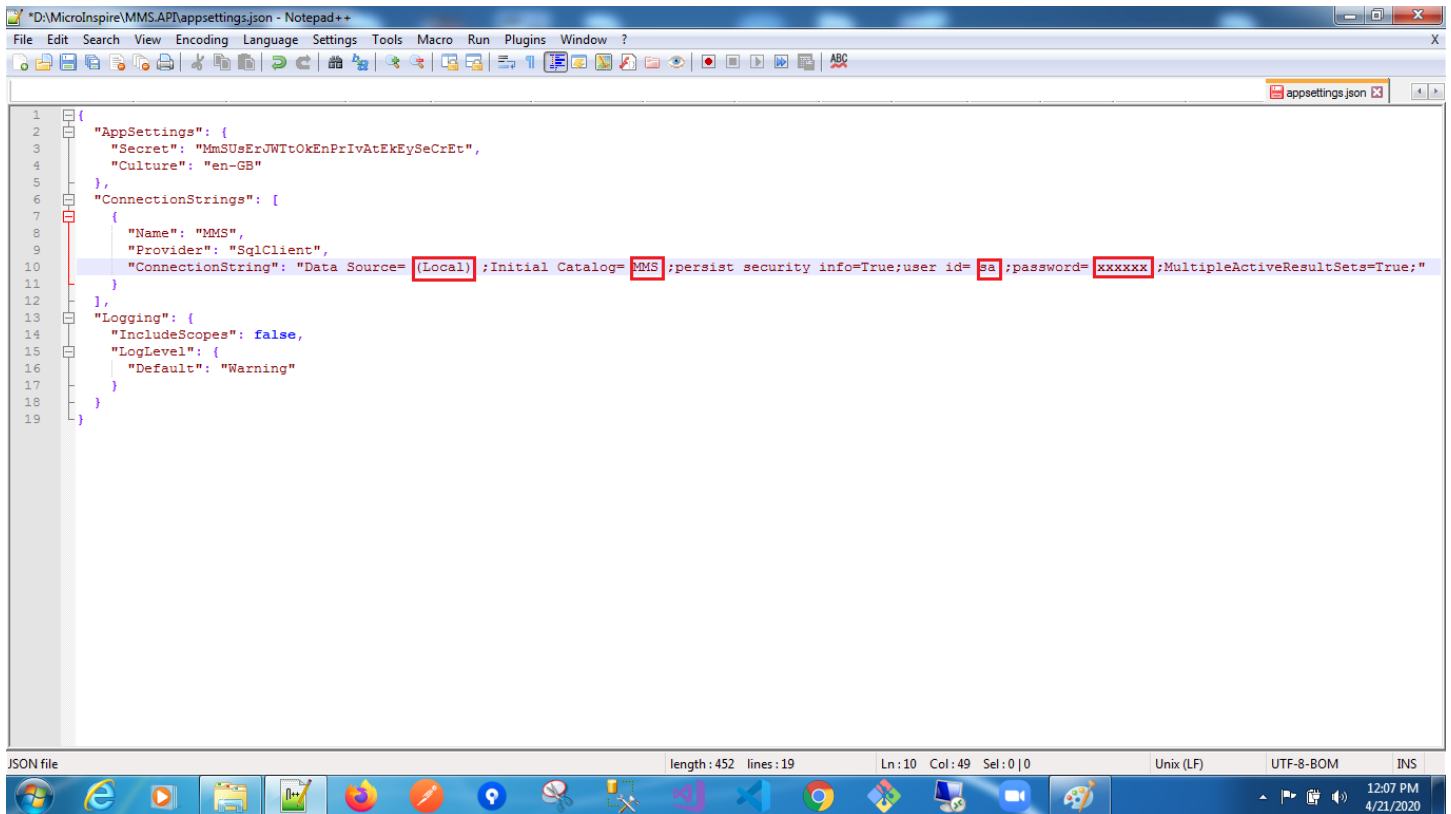
Example: www.contoso.com or marketing.contoso.com

☒ Start Web site immediately



To change SQL Server database connection string open Deploymentfolder\MMS.API\appsettings.json.

Set "Data Source"," Initial Catalog","user id","password".

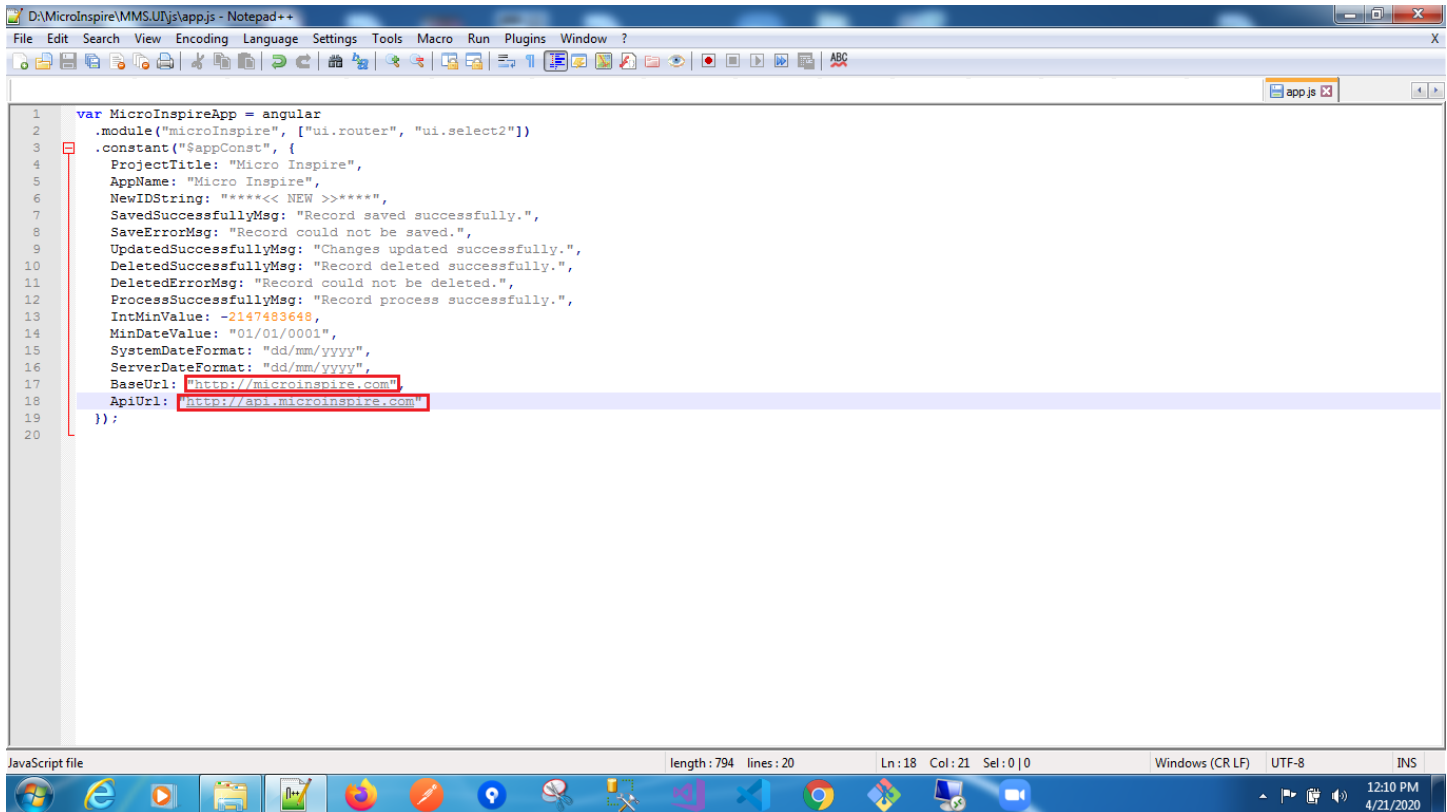


```
1 {
2   "AppSettings": {
3     "Secret": "MmSU5ErJWt0kEnPrIvAtEkEySeCrEt",
4     "Culture": "en-GB"
5   },
6   "ConnectionStrings": [
7     {
8       "Name": "MMS",
9       "Provider": "SqlClient",
10      "ConnectionString": "Data Source=(Local);Initial Catalog=MMS;persist security info=True;user id=sa;password=xxxxxx;MultipleActiveResultSets=True;"
11    }
12  ],
13  "Logging": {
14    "IncludeScopes": false,
15    "LogLevel": {
16      "Default": "Warning"
17    }
18  }
19 }
```

JSON file length: 452 lines: 19 Ln: 10 Col: 49 Sel: 0 | 0 Unix (LF) UTF-8-BOM INS 12:07 PM 4/21/2020

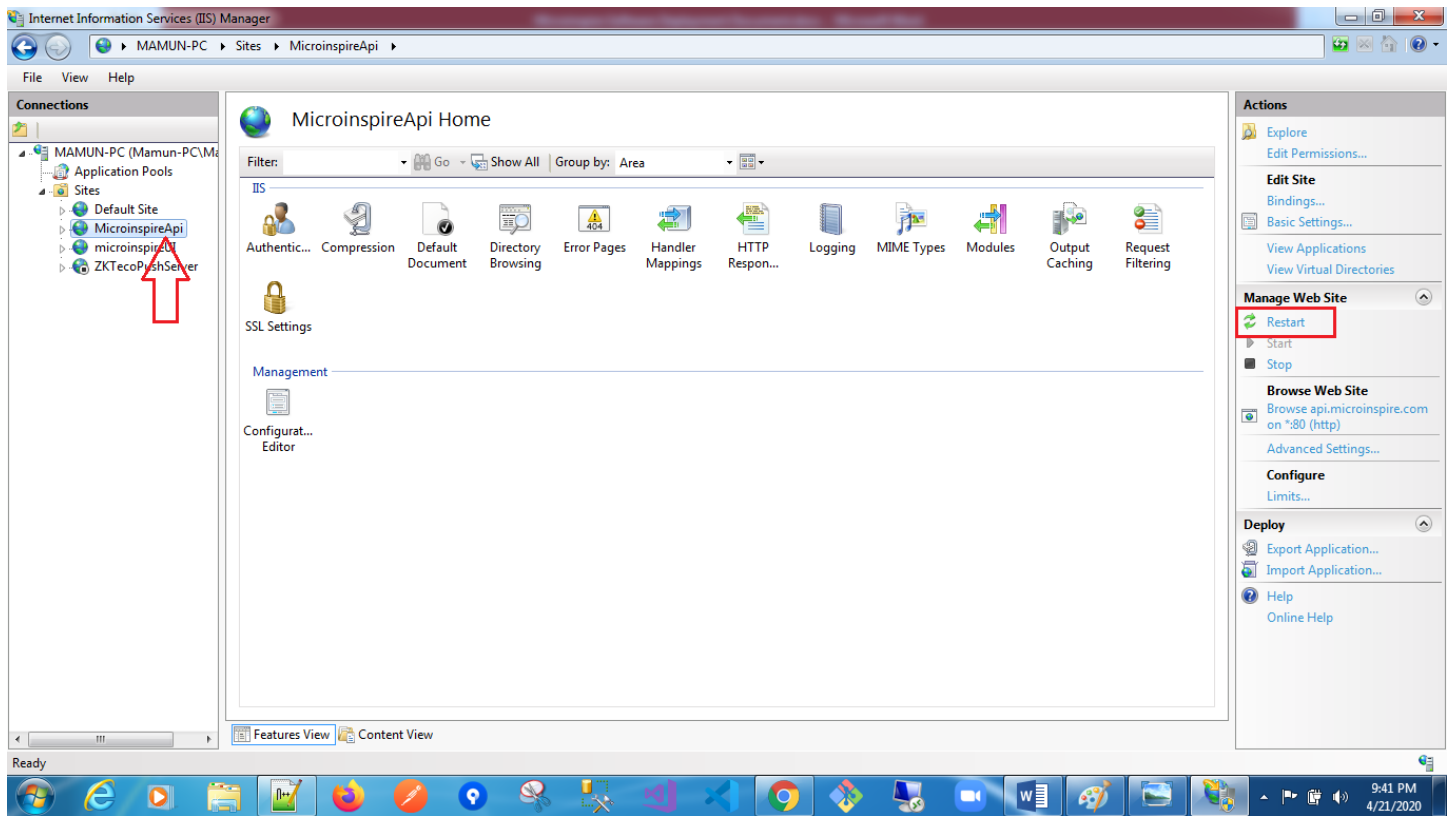
To change BaseUrl & ApiUrl open DeploymentFolder\MMS.UI\js\app.js.

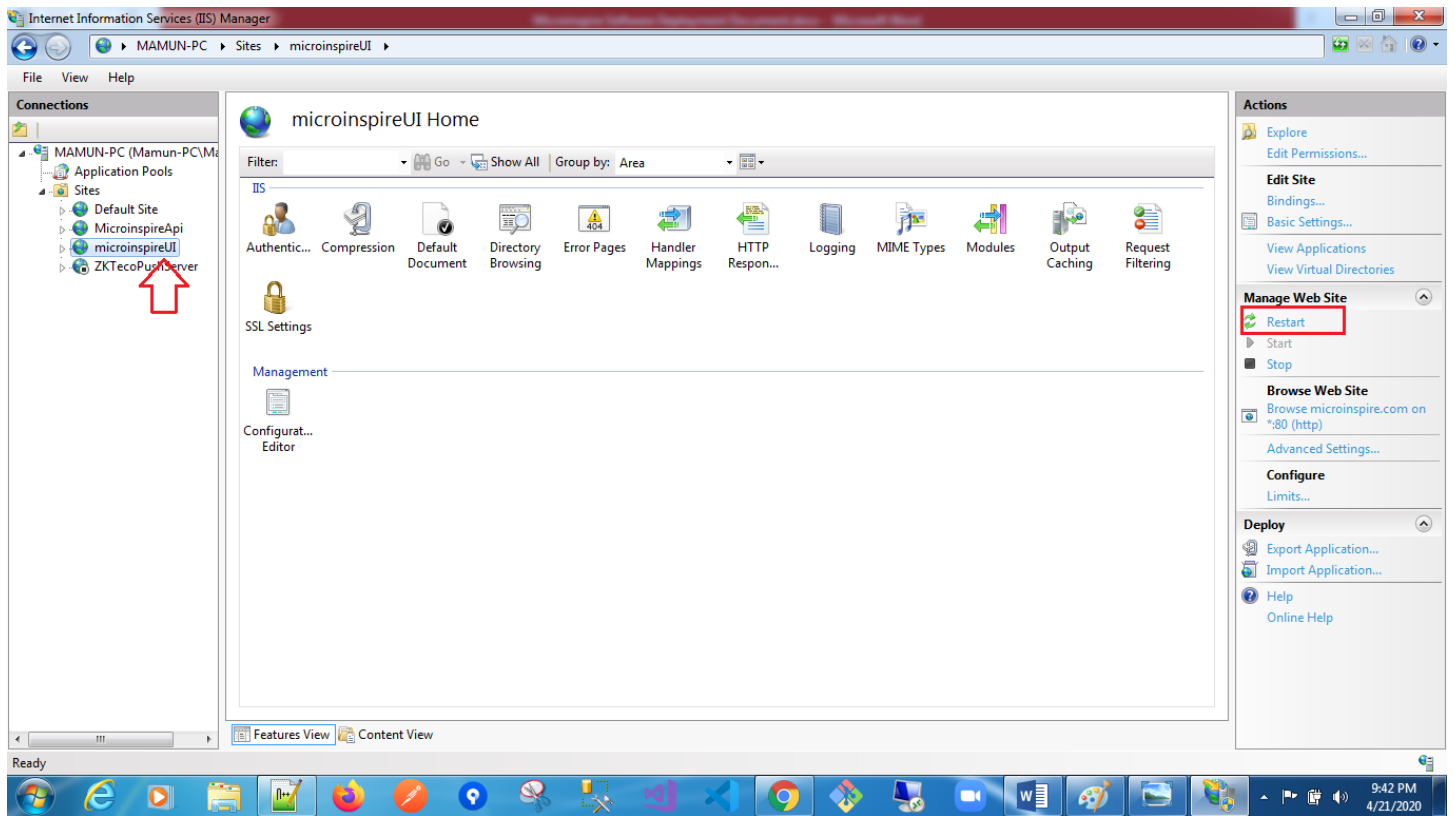
1. BaseUrl = UI (Frontend) Deployment Url. Like: <http://localhost:8085> Or <http://microinspire.com>
2. ApiUrl = API (Backend) Deployment Url. Like: <http://localhost:8086> Or [api.microinspire.com](http://api.microinspire.com)



```
1 var MicroInspireApp = angular
2   .module("microInspire", ["ui.router", "ui.select2"])
3   .constant("$appConst", {
4     ProjectTitle: "Micro Inspire",
5     AppName: "Micro Inspire",
6     NewIDString: "*****< NEW >*****",
7     SavedSuccessfullyMsg: "Record saved successfully.",
8     SaveErrorMsg: "Record could not be saved.",
9     UpdatedSuccessfullyMsg: "Changes updated successfully.",
10    DeletedSuccessfullyMsg: "Record deleted successfully.",
11    DeletedErrorMsg: "Record could not be deleted.",
12    ProcessSuccessfullyMsg: "Record process successfully.",
13    IntMinValue: -2147483648,
14    MinDateValue: "01/01/0001",
15    SystemDateFormat: "dd/mm/yyyy",
16    ServerDateFormat: "dd/mm/yyyy",
17    BaseUrl: "http://microinspire.com"
18    ApiUrl: "http://api.microinspire.com"
19  });
20
```

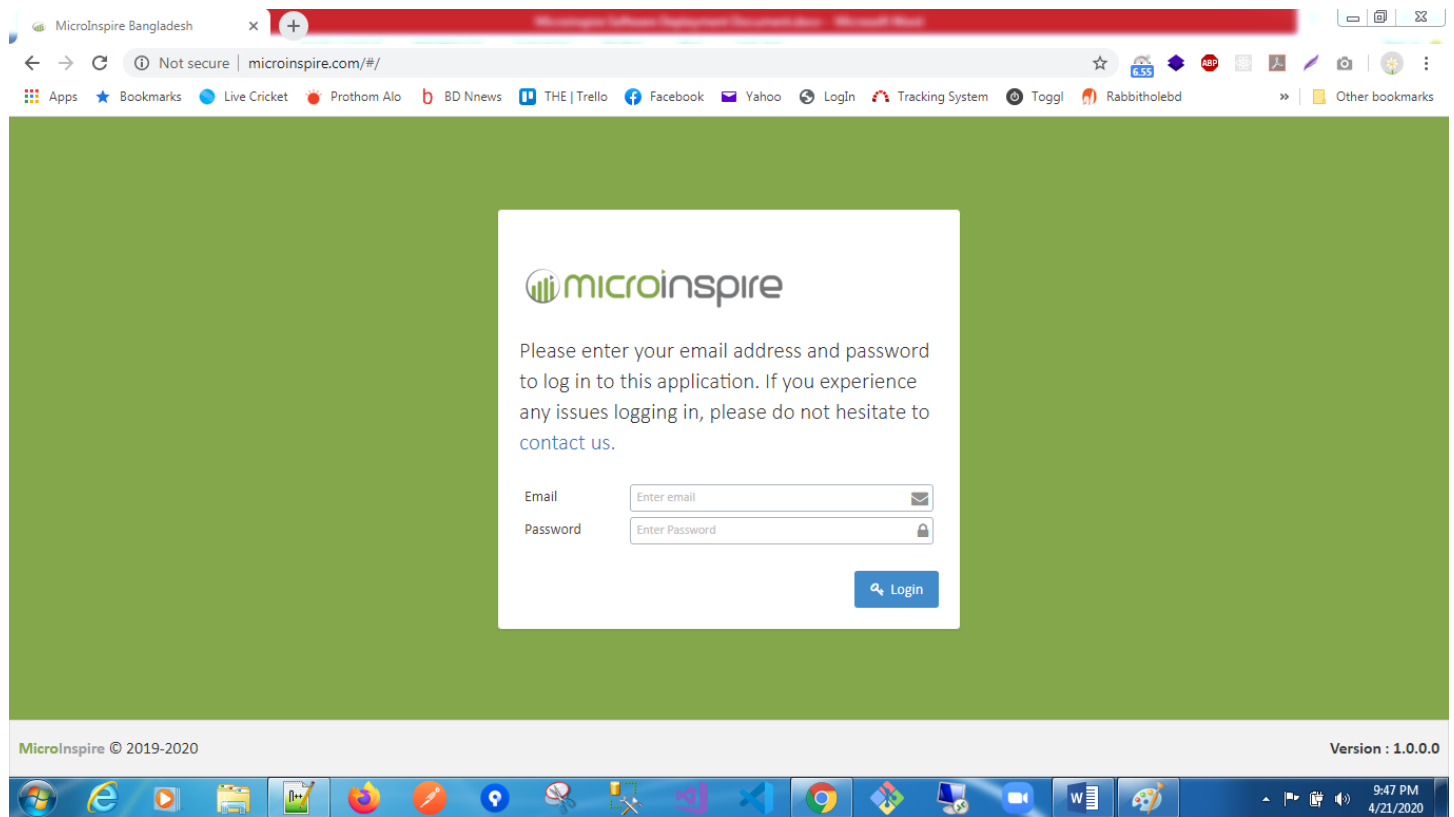
Finally restart two site from IIS.







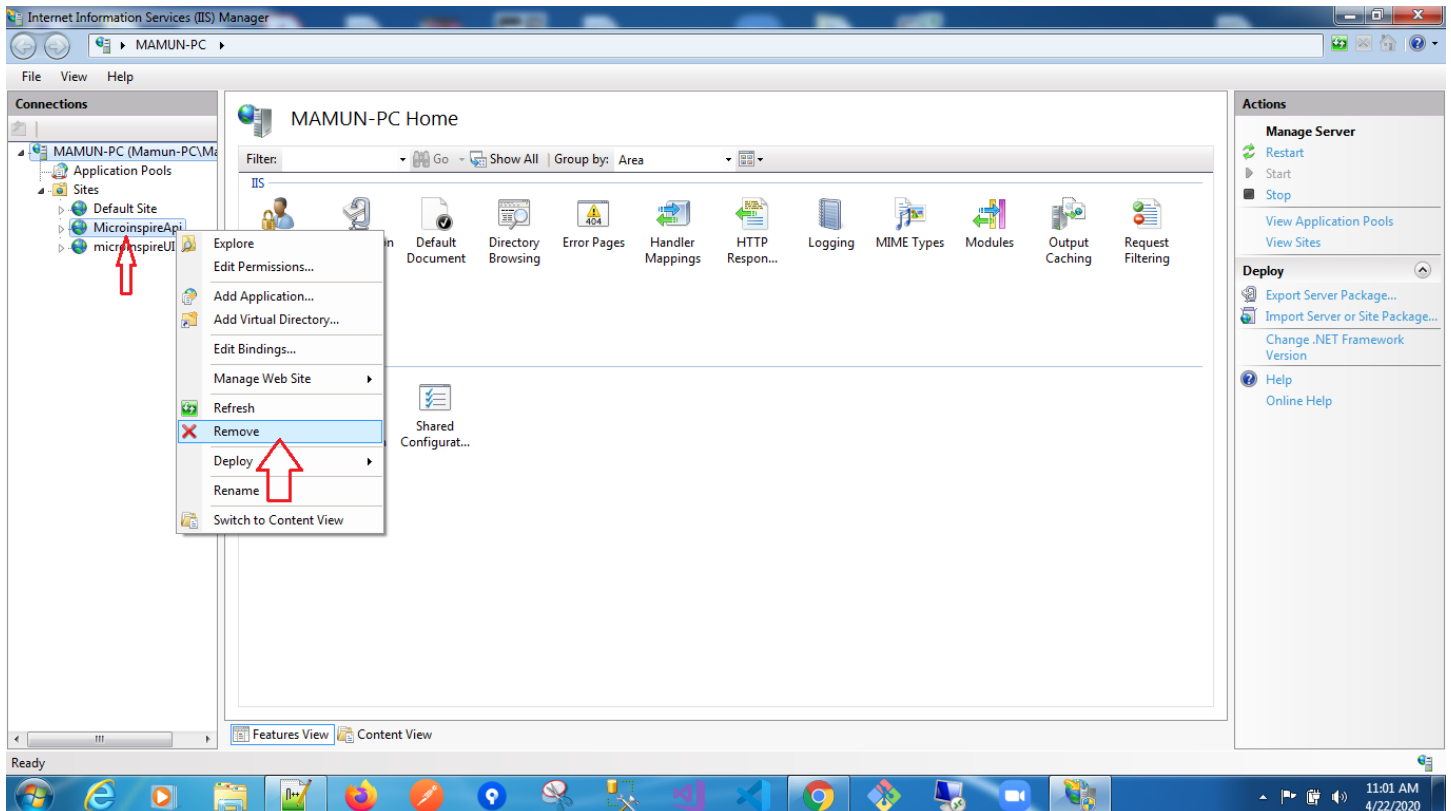
Open browser then browse <http://microinspire.com/#/> & Login with provided email,password.



# To complete the un-deployment need to following step.

## Step1:

1. In IIS Manager, open the server's node in the **Connections** panel. Open the **Sites** folder.
2. Right-click **MicroinspireApi** website and click **Remove** from the contextual menu.



## Step2:

1. In IIS Manager, open the server's node in the **Connections** panel. Open the **Sites** folder.
2. Right-click **MicroinspireUI** website and click **Remove** from the contextual menu.

