



relevant ecommerce™

ZNODE AZURE INSTALLATION GUIDE

April 2016

Znode Multifront Installation on Azure

Prerequisites

Required software:

1. SQL Server 2008 R2
2. .Net Framework 4.5
3. IIS 7.5
4. Microsoft report viewer 2010

Installation Steps:

1. Download the Installer from Znode.com
2. Run the Installer
3. Create the Database from the Znode_Multifront_Database.sql script.
4. Update the ZnodeMultifront Domain Name in the ZnodeDomain table
5. Update Znode.Engine.Admin and Znode.Engine.ApiWeb.config
6. Update Znode.Engine.MvcDemoWeb.config
7. Update Znode Background Service Config and install.
8. Build the projects
9. IIS Setup

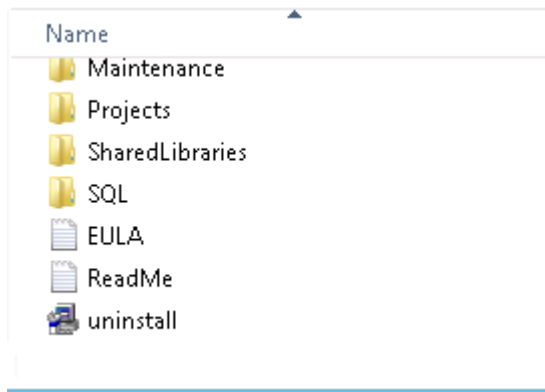
1. Download the Installer from Znode.com

2. Run ZNodeMultifront Installer

1. Download the installer file and double click.

NOTE: Be sure to select a desired location for Znode, it is recommended that you **do not** put Znode in the Program Files folder. We will reference this directory as **\$InstallationDirectory** throughout this document.

2. Follow the prompts.



Projects Folder contains all projects and libraries associated with the Multifront Solution:
Znode.Engine.Admin – Project associated with running the Merchant Admin Dashboard
Znode.Engine.MvcDemo – Project associated with running the Front End U/I of the demo site.
Znode.Engine.API – Project associated with running the Znode API

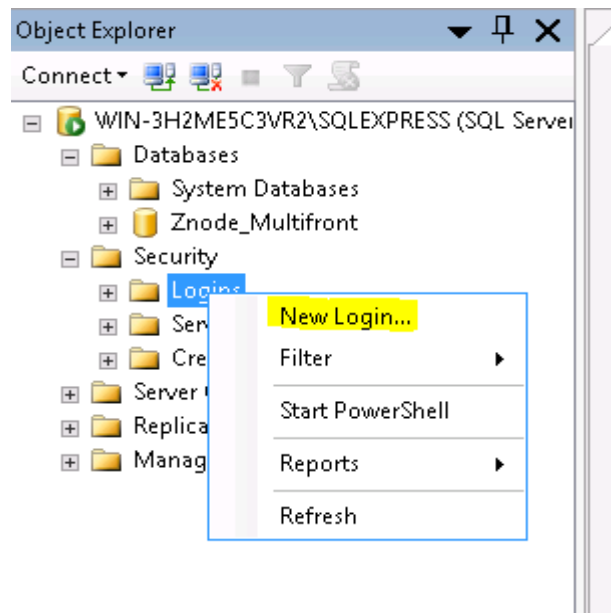
SQL contains the script for creation of the ZnodeMultifront database.

3. Database Setup

In order to render the ZnodeMultifront site, the ZnodeMultifront database must be created.

You can find the database creation script under the `$InstallationDirectory\SQL` folder.

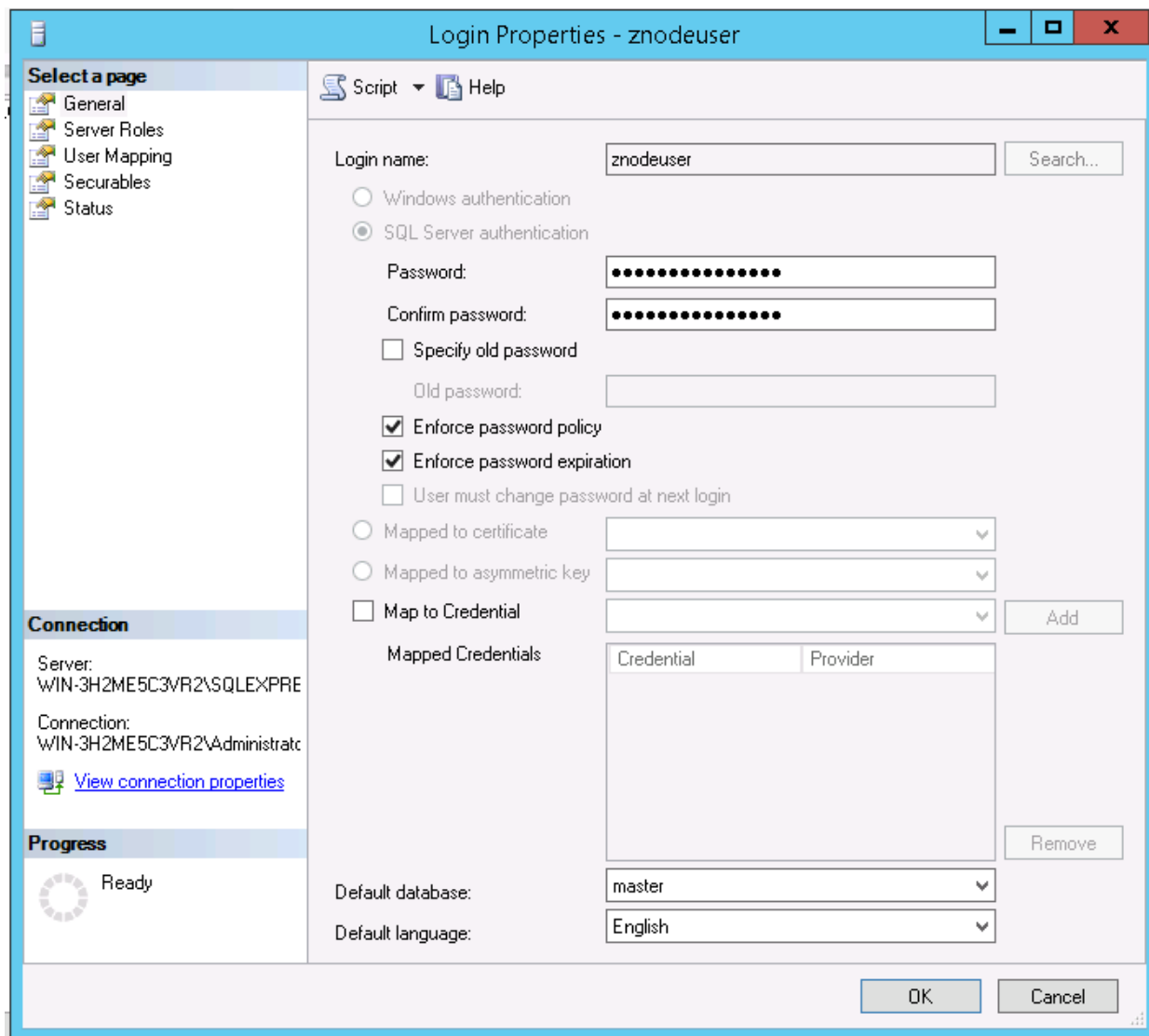
1. Open Microsoft SQL Server Management Studio. (IMPORTANT: Always run SQL Server as admin)
2. Run the `znode_multifront_database.sql` script (found in the SQL folder). This will create a new database called **znode_multifront**. You can change the name of the database prior to running the script, but you will need to update the remaining config files manually.
3. Create a SQL Login. Default web.config settings will use `userid = znodeuser` with `password = p@ssw0rd` in their connection string. (Note that you can create a different SQL login or use windows authentication --- in those cases you need to update the web.config appropriately).
4. Open Security folder and Right click the Logins folder and select New Login. This is optional you can also use existing `znodeuserdbo` user.



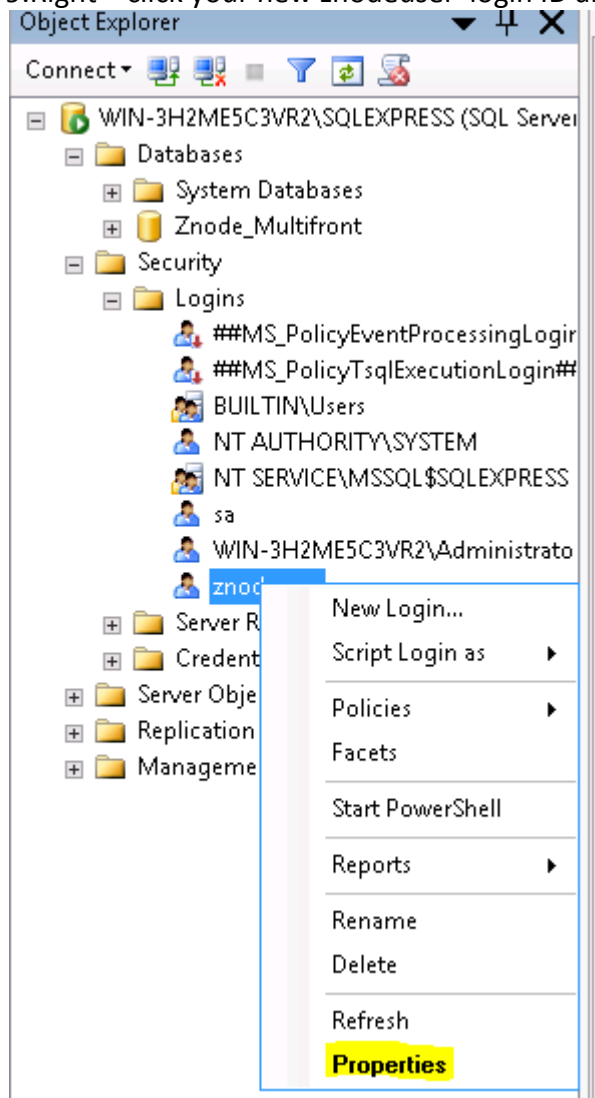
5. Enter Login name in the Login name field.
6. Select the SQL Server Authentication radio button.

7. Enter the password associated with your SQL username in the Password field and enter it again in the Confirm Password field.

8. Click OK. Important: Assign DBO permissions to the login for the database for both schema and roles.



9. Right---click your new znodeuser login ID and select Properties.



10. Select User Mapping in the upper left, and then select the znode_multifront database.

11. Scroll to and check the db owner role membership in the lower---right panel.

12. Confirm Public is already checked (it should be checked by default). If it's not, check the box now.

13. Click OK.

4. Update Domain Settings

In order to render the Znode/Multifront site, the Domain Name associated to your Multifront instance needs to be added to the Znode Domain table. (This is the Domain name of your website) This will be required for all stores. Only follow these steps if your website name is different from default websites.

To update Domain name in the Domain table:

1. Open SQL Server.
2. Go to the ZnodeDomain table in znode_multifront database
3. Add New domain name For Example: multifrontapi.azurewebsites.net

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the Object Explorer displays the database schema for 'MULTIFRONTMVC...bo.ZnodeDomain'. The 'dbo.ZnodeDomain' table is selected and highlighted in yellow. On the right, the table data is displayed in a grid view. The table has six columns: DomainID, PortalID, DomainName, IsActive, and ApiKey. The data includes various domain names such as 'localhost', 'admin.multifront.localhost.com', 'demo.multifront.localhost.com', 'api.multifront.localhost.com', 'mvcdemo.multifront.localhost.com', 'mvcnutsdemo.multifront.localhost.com', 'mvcspecialty.multifront.localhost.com', 'znodesitecore.localhost.com', 'nuts.localhost.com', 'multifrontdemo.azurewebsites.net', 'multifrontmvc.azurewebsites.net', 'multifrontadmin.azurewebsites.net', and 'multifrontapi.azurewebsites.net'. The last row is highlighted in yellow.

DomainID	PortalID	DomainName	IsActive	ApiKey
1	1	localhost	True	5D2B4C5E-D8B3...
2	1	admin.multifront.localhost.com	True	103CFDD6-F349...
3	1	demo.multifront.localhost.com	True	0F39E5C4-5925...
4	1	api.multifront.localhost.com	True	913837FD-36F5...
5	1	mvcdemo.multifront.localhost.com	True	4FB7C166-AF0F...
6	6	mvcnutsdemo.multifront.localhost.com	True	DCD0F4BD-1E83...
7	5	mvcspecialty.multifront.localhost.com	True	72EFB73B-0D7D...
8	1	znodesitecore.localhost.com	True	F6A0BFF3-1210...
9	6	nuts.localhost.com	True	f202217e-7bbd...
10	1	multifrontdemo.azurewebsites.net	True	0F39E5C4-5925...
11	1	multifrontmvc.azurewebsites.net	True	4FB7C166-AF0F...
12	1	multifrontadmin.azurewebsites.net	True	103CFDD6-F349...
13	1	multifrontapi.azurewebsites.net	True	913837FD-36F5...
*	NULL	NULL	NULL	NULL

5. Update Admin and APIWeb.configs

Update connection string and Lucene Index file location in below projects web.config files

1. Znode.Engine.Admin -path is \Projects\Znode.Engine.Admin\Web.config
2. Znode.Engine.API – path is \Projects\Znode.Engine.Api\Web.config
3. Znode.Engine.Demo – path is \Projects\Znode.Engine.Demo\Web.config Example of connection string is given below

```
<connectionStrings>  
<add name="ZNodeECommerceDB" connectionString="Data Source=localhost;Initial  
Catalog=znode_multifront;user id=YourUserId;password=YourPassword" />  
</connectionStrings>
```

Default Lucene Index file location and LuceneIndexReaderKey is given below. Below entry needs to match in all configs

```
<add key="LuceneIndexLocation" value="C:\LuceneData" /><add  
key="LuceneIndexReaderKey" value="Ajgsfdj342343kdksjsjncmflf1" />
```


6. Update MVCDemoWeb.configs

1. Update following in appSettings section of Znode.Engine.MVCDemo/web.config

```
<add key="ZnodeApiRootUri" value="http://api.multifront.localhost.com" />
<add key="ZnodeApiUriItemSeparator" value="," />
<add key="ZnodeApiUriKeyValueSeparator" value="~" />
<!-- Store Domain and Key : key=DomainName and Value=ApiKey-->
<add key="mvcdemo.multifront.localhost.com" value="4FB7C166-AF0F-43A2-
A02AEFB14D5ADAFF"
/>
<add key="mvcnutsdemo.multifront.localhost" value="DCD0F4BD-1E83-455B-A845-
4632C3EE27C8" />
<add key="store.multifront.localhost.com" value="a940eed6-5fd8-4970-
9ecefdf029e4ab8"
/>
```

Note: `api.multifront.localhost.com` is the default domain name of Znode API. Make sure it matches API domain name and API Key that has been configured in the ZNodeDomain database table.

7. Znode Background Service Installation

This is a Windows service that must be installed and running in order for the Lucene search index to be updated and maintained. The name of the executable is Znode.BackgroundService.exe and shows up in Service Manager as “Znode Background Service”. Each web server will have its own copy installed and running.

1. Build Znode.Multifront Solution under `$InstallationDirectory%\Projects`
2. Create a directory e.g. (C:\BackgroundService) where the BackgroundService files will be deployed.
3. Copy all files from `$InstallationDirectory\Projects\Znode.Engine.Task\bin\Debug` to C:\BackgroundService
4. Copy all files from `$InstallationDirectory\SharedLibraries\BackgroundService` to C:\BackgroundService **NOTE:** Installing the Background service from different locations will allow you to rebuild the project in the future without running into errors with the background service dll(s) being in use.

5. Open Znode.Background.Service.exe config in a text editor with the below sections updated:

```
<Znode.Engine.Tasks.Lucene>
<LogFoldervalue="C:\BackgroundService\Logs" />
<TransactionLogFilevalue="C:\BackgroundService\Logs\_transaction.log" />
<LuceneIndexLocationvalue="C:\LuceneData" />
<LuceneIndexIntervalvalue="30" />
<LuceneIndexReaderKeyvalue="913837FD-36F5-4B88-85A7-651E9976A775" />
<LuceneIndexReaderUrl
value="http://api.multifront.localhost.com/search/reloadindex" />
<OnStartWaitTimevalue="10" />
<ServerNamevalue="SERVER_NAME" />
</Znode.Engine.Tasks.Lucene>
```

a. `LogFolder`---> Location of log file

b. `TransactionLogFile`---> Location of Transaction logs

c. `LuceneIndexLocation`---> Location of Lucene Index file

NOTE :--- Make sure this location is same which has been provided in other web.config files (section 5).

d. `LuceneIndexReaderUrl`--->

- WebForms (<http://demo.multifront.localhost.com/UpdateLuceneReader.aspx>)
- MVC(<http://api.multifront.localhost.com/search/reloadindex>)

e. `ServerName`---> This is the name of the server where the Lucene background task is running

6. Update App Setting Section

```
<appSettings>
<add key="EnableWhatDoIHave" value="false" />
<add key="EnableDebugLog" value="true" />
<add key="DebugLogFile"
value="C:\BackgroundService\Logs\_debug.log" />
<add key="LuceneIndexLocation" value="C:\LuceneData" />
</appSettings>
```

Provide Same `LuceneIndexLocation`.

7. `Znode.Background.Service.Install.bat` is a file that will install the service and reference the location of the `Znode.Background.Service.exe`. Verify `binPath` is pointing to `Znode.Background.Service.exe`

- e.g. (`binPath= "C:\BackgroundService\Znode.BackgroundService.exe"`)

8. To run the batch script, right-click on the Znode.BackgroundService.Install.bat file and select Run as administrator.

8. Build Applications

To compile/build application:

Please update the **highlighted** path with your installer location

1. Open command prompt. (Start->Run->cmd)
2. Enter the following command to build Znode.Engine.Admin

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319\msbuild "C:\Installer\Znode Multifront Enterprise\Projects\Znode.Engine.Admin\Znode.Engine.Admin.csproj"
```

3. After successful build, enter the following command to build Znode.Engine.MvcDemo

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319\msbuild "C:\Installer\Znode Multifront Enterprise\Projects\Znode.Engine.MvcDemo\Znode.Engine.Mvcdemo.csproj"
```

4. After successful build, enter the following command to build Znode.Engine.Api

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319\msbuild "C:\Installer\Znode Multifront Enterprise\Projects\Znode.Engine.Api\Znode.Engine.Api.csproj"
```

5. After successful build, enter the following command to build Znode.Engine. Demo

```
C:\Windows\Microsoft.NET\Framework\v4.0.30319\msbuild "C:\Installer\Znode Multifront Enterprise\Projects\Znode.Engine.Demo\Znode.Engine.Demo.csproj"
```

```

Administrator: C:\Windows\system32\cmd.exe
Secure\Orders\OrderManagement\OrderDesk\QuickOrder.ascx.cs(695,38): warning CS0618: 'ZNode.Libraries.ECommerce.Catalog.ZNodeProductBase.SelectedAddOnItems' is obsolete: 'This property is obsolete. Use 'SelectedAddOns' property in the inherited base.' [C:\Znode\Znode Multifront Enterprise\Projects\Znode.Engine.Admin\Znode.Engine.Admin.csproj]
Secure\Orders\OrderManagement\OrderDesk\QuickOrder.ascx.cs(983,13): warning CS0618: 'ZNode.Libraries.ECommerce.Catalog.ZNodeProductBase.SelectedAddOnItems' is obsolete: 'This property is obsolete. Use 'SelectedAddOns' property in the inherited base.' [C:\Znode\Znode Multifront Enterprise\Projects\Znode.Engine.Admin\Znode.Engine.Admin.csproj]
Secure\Orders\OrderManagement\OrderDesk\QuickOrder.ascx.cs(1382,50): warning CS0618: 'ZNode.Libraries.ECommerce.Catalog.ZNodeProductBase.SelectedAddOnItems' is obsolete: 'This property is obsolete. Use 'SelectedAddOns' property in the inherited base.' [C:\Znode\Znode Multifront Enterprise\Projects\Znode.Engine.Admin\Znode.Engine.Admin.csproj]
Secure\Orders\OrderManagement\OrderDesk\QuickOrder.ascx.cs(1401,50): warning CS0618: 'ZNode.Libraries.ECommerce.Catalog.ZNodeProductBase.SelectedAddOnItems' is obsolete: 'This property is obsolete. Use 'SelectedAddOns' property in the inherited base.' [C:\Znode\Znode Multifront Enterprise\Projects\Znode.Engine.Admin\Znode.Engine.Admin.csproj]

1621 Warning(s)
0 Error(s)

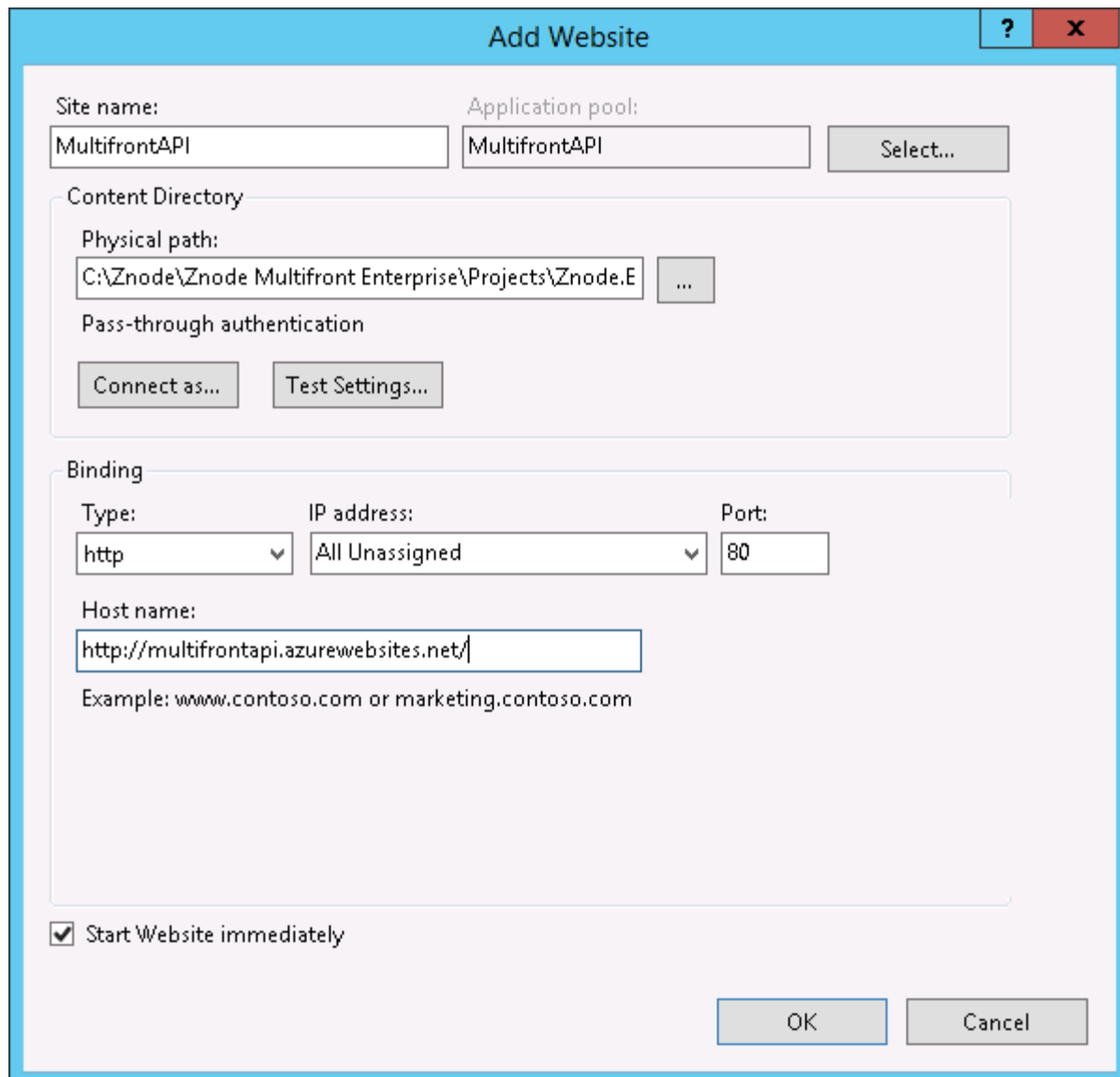
Time Elapsed 00:00:53.42

C:\Users\Administrator>C:\Windows\Microsoft.NET\Framework\v4.0.30319\msbuild "C:\Znode\Znode Multifront Enterprise\Projects\Znode.Engine.Admin\Znode.Engine.Admin.csproj"
    
```

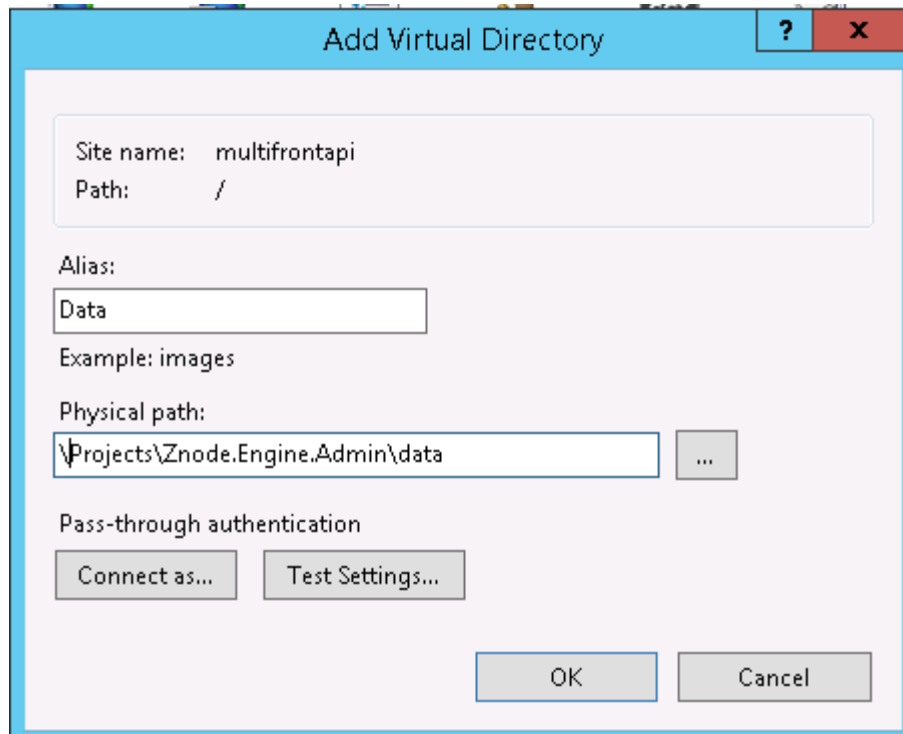
9. IIS Setup

To begin the configuration of Znode.Engine.Api :

1. Open IIS and select Sites.
2. Right click and click Add Web Site and enter the Sitename, select the physical path as (\Projects\Znode.Engine.Api) and enter the host name as follows:
Note: Make sure Host name is the same as Domain name for the API given in ZnodeDomain Table and provided in MvcDemoweb.config file



3. Click Ok, A new website is created in IIS. To give the permissions, select the site, right click and select Edit
4. Give Permissions for NETWORK SERVICE and IIS_USR.
5. Right click on API Website and Add Virtual Directory and point that virtual directory to Projects/Znode.Engine.Admin/data folder as shown in below screenshot.



6. Make an entry in the host file for API configuration.

- Hosts file is located at (C:\Windows\System32\drivers\etc\hosts)
- Create an entry for the API URL (i.e. 127.0.0.1 multifrontapi.azurewebsites.net)

7. Restart IIS.

MVC Site, Web Form and Site admin Set up

ZnodeMultifront provides several projects to allow control in flexibility to how the solution will be deployed in IIS. As such, your deployment steps by include changes such as binding, websites, applications, and application pools. These steps are intended as a generic guideline for running the solution in IIS, either in production, staging, or development machine.

Each project must be built or published before deploying.

Follow same steps for all websites

1. Open IIS Manager (inetmgr.exe)
2. Right Click "Sites"
 - a) "Add Website" a. Site name: Name of the website that will be displayed in IIS.
 - b) Physical Path: Will be one of the following for 3 different sites

- (1) Znode.Engine.Admin
 - (2) Znode.Engine.MvcDemo
 - (3) Znode.Engine.Demo
- c) Host name: Will be the domains being used for stores. These should match up with domains in the Znode Domain table of the ZnodeMultifront database.
- d) Physical path should point to the appropriate solution i. MVC Websites = Znode.Engine.MvcDemo ii. Admin Dashboard = Znode.Engine.Admin iii. Web Form Website = Znode.Engine.Demo.

Verify the Deployment of Multifront

After creating the database and updating the corresponding web.configfiles,set each of the following as Startup Projects and debug to confirm the website will load:

- Znode.Engine.Admin
- Znode.Engine.MvcDemo
- Znode.Engine.API

Note: Default username/password for Site Admin is admin/admin

MULTIFRONT®

Merchant Login

Log in to setup and manage your store.

Username

Password

LOGIN

Remember Me [Forgot your password?](#)

znode relevant ecommerce



API Overview

The Znode API is organized around REST and is designed to have predictable, resource-oriented URLs and to use HTTP response codes to indicate API errors. The API uses built-in HTTP features — such as headers, methods, and status codes — which can be understood by any off-the-shelf HTTP client. The Znode API only supports requests and responses in JSON format.

Authentication

The Znode API supports authentication by looking for the following values in the request header:

- **Znode-DomainName:** This is the domain for the store whose data will be retrieved during the request.
- **Znode-DomainKey:** This is the API key for the domain, as found in the ZNodeDomain table in the Multifront database.

Having these two values in the header also sets the context for the request. Meaning, if you set the Znode-DomainName header value as "www.mysite1.com", then all data retrieved during the request will be for the store related to that domain.

All requests to the Znode API require that **both** header values exist in the request; if either one doesn't exist, the request will fail. Also, if the Znode-DomainKey value doesn't match the one associated with the Znode-DomainName, the request will fail. If either of these two scenarios cause the request to fail, the response will immediately return with the following information:

- **HTTP status code:** 403 Forbidden
- **HTTP status description:** Domain name and key are either incorrect or missing from the request header.

Disabling Authentication

Authentication can be disabled by setting **ValidateDomainHeaders** to false in the <appSettings> section of the API's web.config file. However, in most cases, this value should be set to true in production environments.

```
<add key="ValidateDomainHeaders" value="true" />
```

During Development

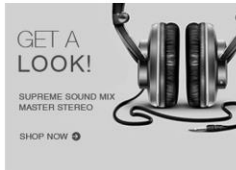
When developing a client that uses the Znode API, it will be useful at times to open endpoints in a browser in order to see data. To ensure this is possible, you need to add an entry in the ZNodeDomain table for the domain of the API itself. In this case, simply disabling authentication will have no effect.

For example, if you have the API deployed locally at "api.localhost.com", you should add an entry in the ZNodeDomain table where the DomainName field is "api.localhost.com" with the relevant portal ID. This will be the context for retrieving data when viewing endpoints in a browser.

IMPORTANT: This should be for development purposes only.

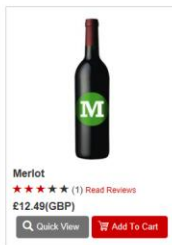
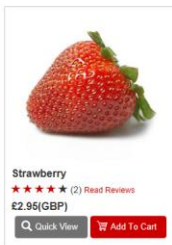


Our Special Offers of the Month



Enjoy free shipping on all orders from £180

Best Sellers



Subscribe



SHOP

- Nuts
- Cheese
- Flowers
- Organic Produce
- Wine
- Fruit
- Gifts
- Vegetables

HELP

- Login
- Register
- Contact Us
- FAQ
- Returns
- Shipping Info
- Site Map

STORE INFO

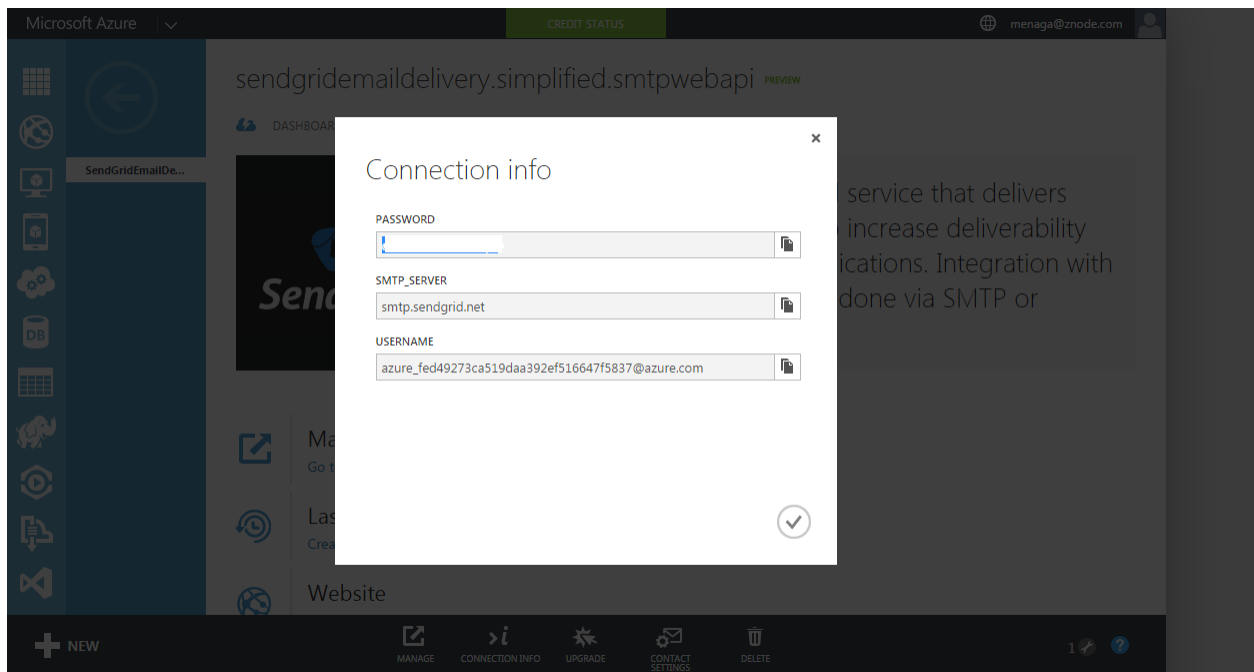
- About Us
- Find A Store
- Careers
- Press
- Gift Cards
- To Top

CUSTOMER SUPPORT

enquiries@maxwells.com
1-888-MY-STORE

Email Settings

1. Go to Setup->Stores
2. Click on Manage for the specific store.
3. Goto SMTP tab and click Edit
4. Enter the following settings in Edit SMTP settings. Your SMTP details will be found in SendGrid-.connection Info



Edit SMTP Settings for - Fine Foods

SMTP Settings

SMTP Port

SMTP Server

SMTP Server User Name

SMTP Server Password

Enable SSL for SMTP

SUBMIT **CANCEL**


Allow apps to communicate through Windows Firewall

To add, change, or remove allowed apps and ports, click Change settings.

What are the risks of allowing an app to communicate?



Allowed apps and features:

Name	Private	Public	
<input type="checkbox"/> iSCSI Service	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Key Management Service	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Netlogon Service	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Network Discovery	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Performance Logs and Alerts	<input type="checkbox"/>	<input type="checkbox"/>	
<input checked="" type="checkbox"/> Port 81-84	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
<input checked="" type="checkbox"/> Remote Desktop	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
<input type="checkbox"/> Remote Event Log Management	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Remote Event Monitor	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Remote Scheduled Tasks Management	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Remote Service Management	<input type="checkbox"/>	<input type="checkbox"/>	
<input type="checkbox"/> Remote Shutdown	<input type="checkbox"/>	<input type="checkbox"/>	

