Deploy Flask App to IIS Windows Server 2019

Please follow the steps below to deploy a python flask app on windows server IIS.

1. Install python

<https://www.python.org/downloads/>

1. Install flask

Open cmd as administrator and type.

Pip install flask

1. Install wfastcgi

Open cmd as administrator and type.

Pip install wfastcgi

1. Enable wfastcgi

Open cmd as administrator and type.

wfastcgi-enable

1. Install git (Only if you prefer to use git, which is a recommended for better code distribution)

<https://git-scm.com/>

1. Clone the flask application from repository or just copy the files over manually

There are multiple ways to do this, but the fastest way to take a one-time copy of your git repository, for more persistence in code management you can use a git UI like Github Desktop or check out the git documentation. To do this the fast way follow the steps below.

Open cmd as administrator and type

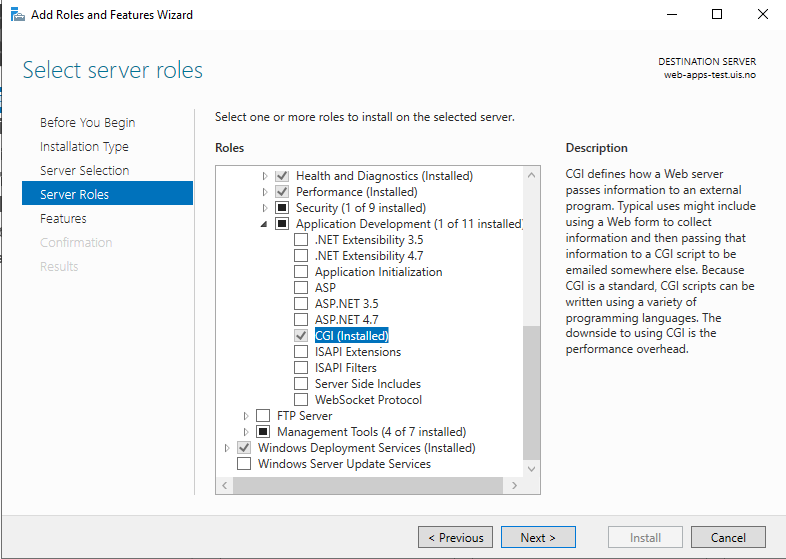
Cd C:\inetpub\wwwroot\

Git clone <url for your git repository>

1. Create a production website on IIS

Open Server Manager and follow the steps below.

Enable the CGI feature on IIS.



Open IIS.

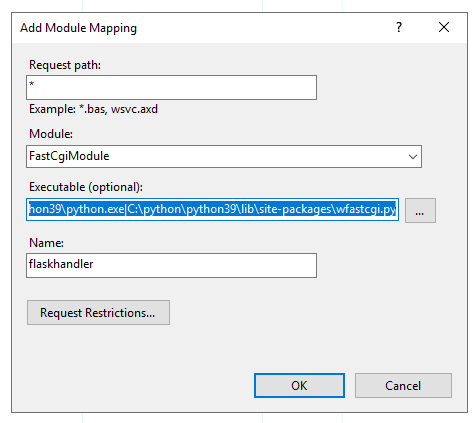
Right-click on the server name and select add website.

Enter the site name, physical path(the path of the folder where you cloned your flask project), and the site building.

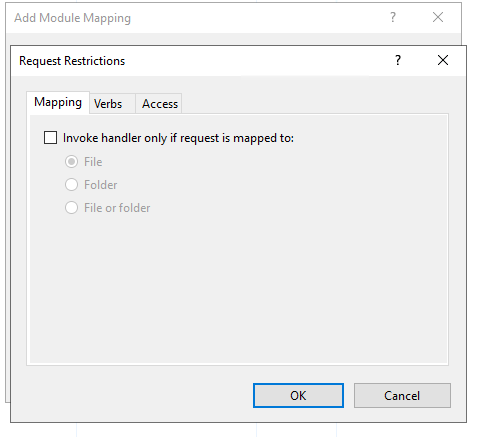
From the features view of your site, enter Handler Mappings, and to the right choose Add Module Mapping.

Put in the following inputs, in executable you need the path of your python executable file and wfastcgi module file, in my case it was under the following paths and it is retrieved in cmd after enabling wfastcgi early on in this documentation.

C:\python\python39\python.exe|C:\python\python39\lib\site-packages\wfastcgi.py



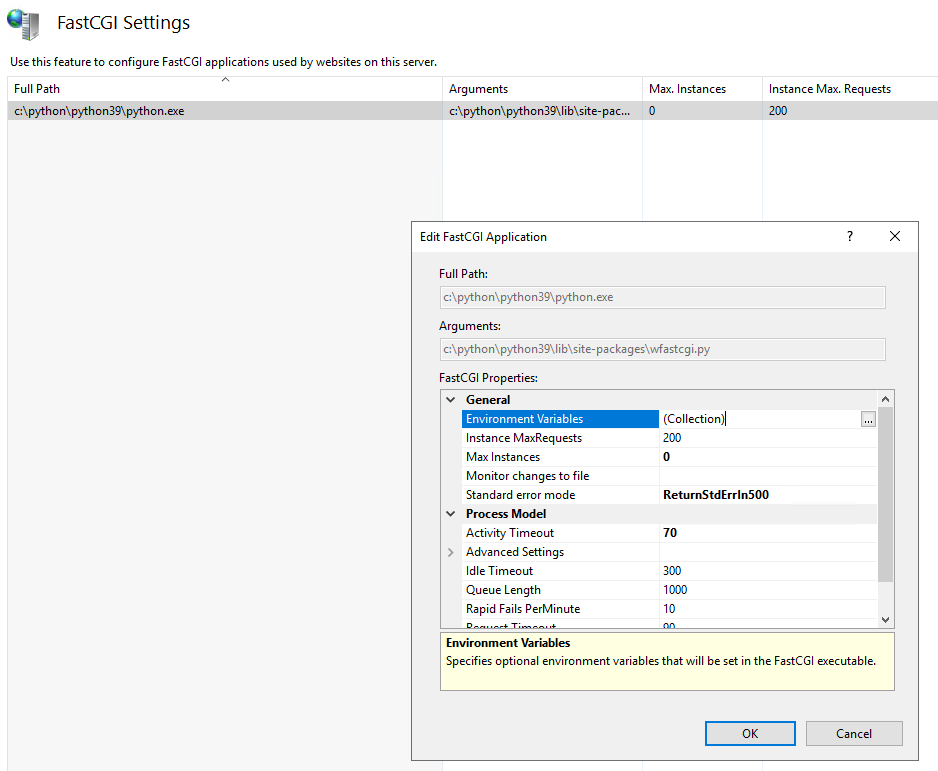
Click on Request Restrictions in the same window above and uncheck Invoke handler only if request is mapped to:



Confirm and agree to the rest of the pop confirmation windows.

Go back and select the server name, then select fast CGI settings from the feature view pane.

Double-click on your python full path, and enter the collection of your Environment Variables under General.



Set the following paths and parameters

PYTHONPATH (Your local python-flask repository path).

WSGI\_HANDLER (\*.app) where \* equals the name of the flask application file, often it is called app.py or main.py and located in PYTHONPATH.

If your application is expected to process requests from clients in extended period of time, you can change the Activity Timeout parameter to 3600 seconds for example.

Remember to

* Install all packages that your flask application needs before you start your webserver.
* Add any required environment variables by your application
* Add the needed permissions to folders and physical path

1. Redirect HTTP traffic to HTTPS

Install the URL Rewrite extension

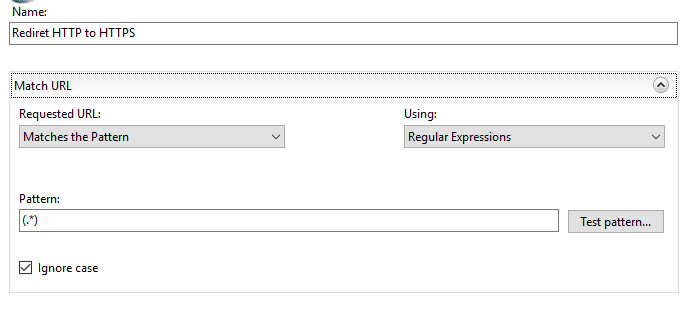
<https://www.iis.net/downloads/microsoft/url-rewrite>

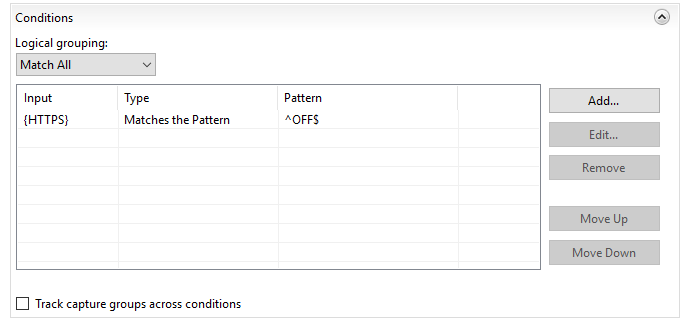
You might have to close the Server Manager and IIS, then open them again to see the URL Rewrite extension module on your Features View Pane

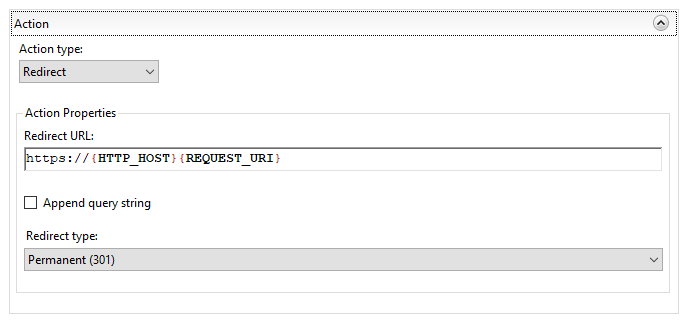
Open IIS and enter URL Rewrite

Go to Add Rule(s), then choose Blank rule

Make sure you form the rule as described in the screenshots below







Click on apply to the right put the redirect rule into effect.