

## Contents

<b>Azure DevOps Principles and Practices</b>	3
<b>1. Project Detail Documentation</b>	3
1.1. An overview of the project	3
1.2. Architectural Diagram	3
1.3. Instructions for running the Python project	3
1.4. Short description of how to improve the project in the future	8
1.5. Screenshots demonstrating key steps	8
<b>2. Project Estimate Document</b>	8
2.1. A link to a Trello board for the project	9
2.2. A link to a spreadsheet that includes the original and final project plan	10
<b>3. Professional, portfolio-ready demo of an Azure DevOps application</b>	11
3.1. A link to the screencast on YouTube	11
<b>Continuous Integration</b>	12
<b>4. Clone and build an application inside of Azure Cloud Shell</b>	12
4.1. Create the Cloud-Based Development Environment	12
4.1.1. Create a GIT repo	12
4.1.2. Setup SSH key	12
4.2. A screenshot showing the project cloned into Azure Cloud Shell	13
4.3. A screenshot showing the passing tests that are displayed after running the test	14
<b>5. Conduct a test run on an application in Azure Cloud Shell</b>	15
5.1. A screenshot showing the output of a test run	15
<b>6. Use GitHub Actions to test and lint a project while writing code</b>	15
6.1. The GitHub repo contains the GitHub Actions YAML file	16
6.2. The GitHub actions badge is shown in the README, like in this screenshot.	17
<b>Continuous Delivery</b>	19
<b>7. Load test an application using Locust</b>	19
7.1. A screenshot of the application running against a load test with locust	19
<b>8. Deploy an application using Azure Pipelines into Azure App Services</b>	19
8.1. A screenshot of Azure App Service	20

8.1. A screenshot of a successful prediction in Azure Cloud Shell..... 20

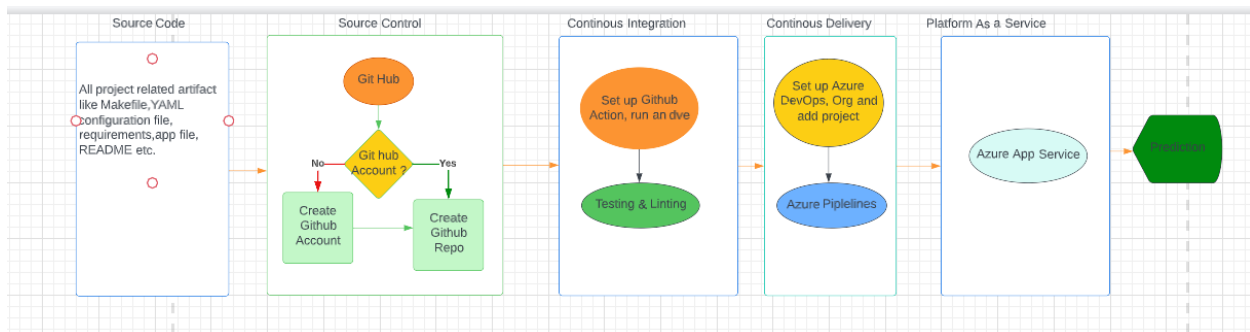
# 1. Azure DevOps Principles and Practices

## 1.1. Project Plan

### 1.1.1 An overview of the project

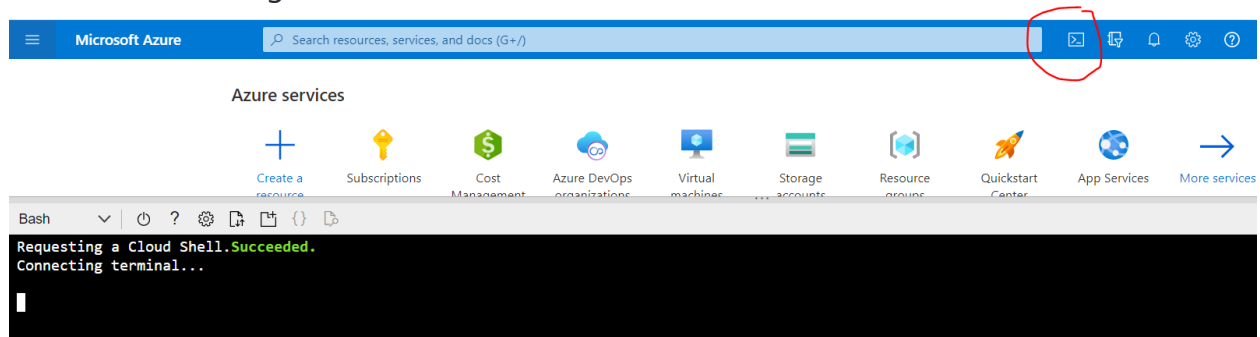
This project is a Python flask app that serves out predictions (inference) about housing prices through API calls which can be extended to any pre-trained machine learning model, such as those for image recognition and data labeling. The project provides pre-trained, sklearn model that has been trained to predict housing prices in Boston according to several features, such as average rooms in a home and data about highway access, teacher-to-pupil ratios, and so on. I will build a Github repository from scratch and create a scaffolding that will assist me in performing both Continuous Integration and Continuous Delivery of this project. I will use Github Actions along with a Makefile, requirements.txt and application code to perform an initial lint, test, and install cycle and then integrate this project with Azure Pipelines to enable Continuous Delivery to Azure App Service

## 1.2. Architectural Diagram



## 1.3. Instructions for running the Python project

1.3.1. Login into azure portal and click on Azure cloud shell (>). This will prompt to create a storage account and file share for first time, this is required to store Azure cloud shell related setting and metadata.

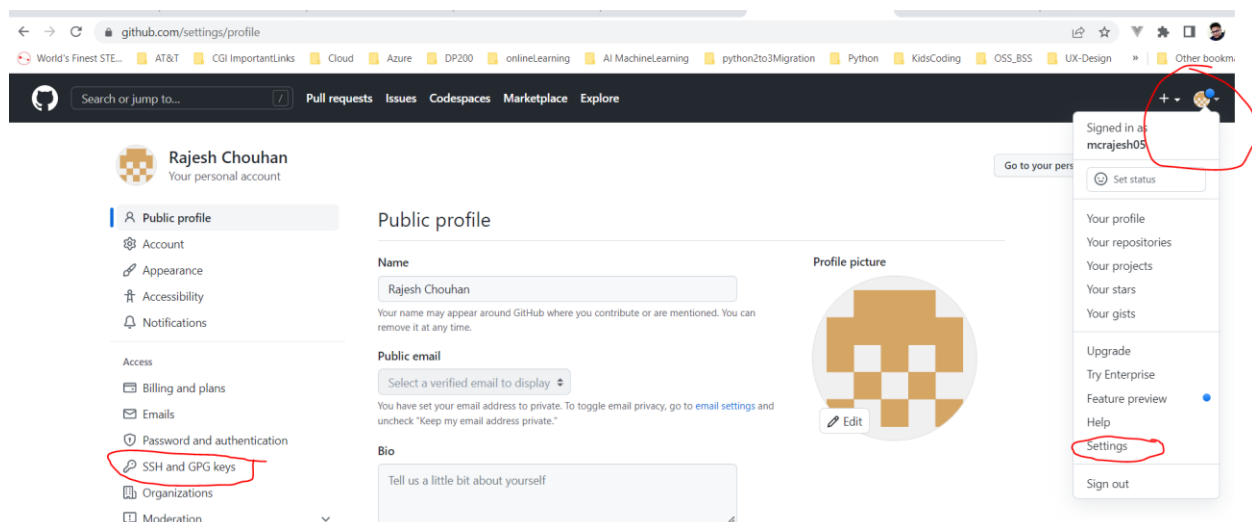


1.3.2. Create ssh-key and upload the public key GitHub account to facilitate secure password less communication.

Use command ssh-keygen



To Upload the public key GitHub account, login to GitHub account and click on profile and setting as shown below.













Click on SSH and GPG and New SSH key



**Rajesh Chouhan**  
Your personal account

Go

-  Public profile
-  Account
-  Appearance
-  Accessibility
-  Notifications

- Access
-  Billing and plans
-  Emails
-  Password and authentication
-  **SSH and GPG keys**
-  Organizations

## SSH keys

New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

### Authentication Keys



**devops-pay-as-you-go**

SHA256:nLYK71vWGXZFtF8TbXIIdEnZRNIJ81TyT5R5t13T2wk

Added on 13 hours ago

Last used within the last week — Read/write

Delete



**myLinuxVM**






SHA256:dKb081/F2Bd6r/8svTTHRZwFSNqkv/51LbMjdrDU56g







Added on 12 hours ago

Last used within the last week — Read/write

Delete

Fill the title with appropriate and copy the public key form cloud shell and past into key field. Save it.

-  Public profile
-  Account
-  Appearance
-  Accessibility
-  Notifications

- Access
-  Billing and plans
-  Emails
-  Password and authentication
-  **SSH and GPG keys**
-  Organizations
-  Moderation

Code, planning, and automation

 Repositories

## SSH keys / Add new

Title

Key type

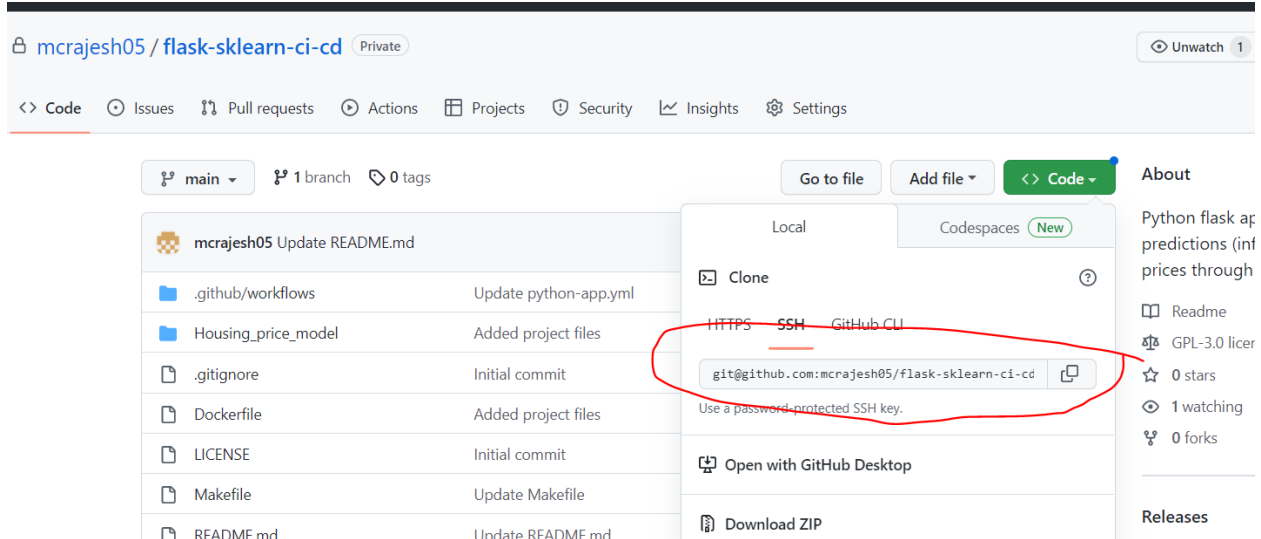
Authentication Key

Key

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp521', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256@openssh.com', or 'sk-ssh-ed25519@openssh.com'

Add SSH key

### 1.3.4. Clone the project into cloud shell. Grab the ssh URL.



In azure cloud shell type as shown the screen shot.

```
rajesh [ ~ ]$ git clone git@github.com:mcrajes05/flask-sklearn-ci-cd.git
Cloning into 'flask-sklearn-ci-cd'...
The authenticity of host 'github.com (20.207.73.82)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJhbpZisF/zLDA0zPMSvHdKr4UvCoqU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 84, done.
remote: Counting objects: 100% (84/84), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 84 (delta 33), reused 20 (delta 5), pack-reused 0
Receiving objects: 100% (84/84), 242.30 KiB | 386.00 KiB/s, done.
Resolving deltas: 100% (33/33), done.
rajesh [ ~ ]$ ls -la
total 64
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:45 .
drwxrwxrwx 3 root root 4096 Nov 15 05:27 .
drwx----- 5 rajesh rajesh 4096 Nov 15 05:28 .azure
-rw----- 1 rajesh rajesh 322 Nov 15 05:45 .bash_history
-rw-r--r-- 1 rajesh rajesh 178 Apr 23 2022 .bash_logout
-rw-r--r-- 1 rajesh rajesh 645 Apr 23 2022 .bash_profile
-rw-r--r-- 1 rajesh rajesh 741 Nov 15 05:27 .bashrc
lrwxrwxrwx 1 rajesh rajesh 22 Nov 15 05:27 clouddrive -> /usr/csuser/clouddrive
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:45 flask-sklearn-ci-cd
drwx----- 2 rajesh rajesh 4096 Nov 15 05:45 .ssh
-rw-r--r-- 1 rajesh rajesh 42 Nov 15 05:27 .tmux.conf
```

### 1.3.5. Create and virtual environment and activate it.

- Use command

```
python3 -m venv venv310
```

```
Python 3.9.14
rajesh [ ~ ]$ python3 -m venv .devops
```

- Use command to activate

source ./venv310/bin/activate

```

-rw-r--r-- 1 rajesh rajesh 22287 Jun 17 22:09 .zshrc
rajesh [ ~ ]$ source ./devops/bin/activate
(.devops) rajesh [ ~ ]$

```

1.3.6. Navigate to cloned project folder. Run “make all” . This will install all the necessary python libraries

1.3.7. To test locally run: python app.py

- Open new cloud shell and run: ./make\_prediction.sh

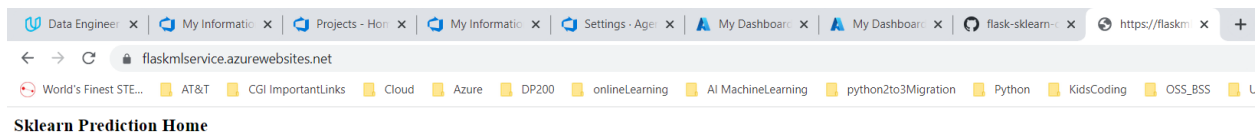
```

rajesh [ ~ ]$ source ./devops/bin/activate
(.devops) rajesh [ ~ ]$ cd flask-sklearn-ci-cd
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$ ./make_prediction.sh
Port: 5000
{
  "prediction": [
    20.353731771344123
  ]
}
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$

```

1.3.7. To deploy and test.

- Run the following command to deploy the app.  
az webapp up --name flaskmlservice --resource-group devops --runtime "PYTHON:3.10" --sku B
- Once the deployment is completed. Get the URL something like  
<https://flaskmlservice.azurewebsites.net/>
- open browser and open this URL. Verify if you get output as shown below.



Once you see the above output then run the command as shown below to see prediction.

```
-rwxrwxrwx 1 rajesh rajesh 358 Nov 15 05:45 run_kubernetes.sh
-rwxrwxrwx 1 rajesh rajesh 417 Nov 15 05:45 test_hello.py
-rwxrwxrwx 1 rajesh rajesh 333 Nov 15 05:45 upload_docker.sh
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$ ./make_predict_azure_app.sh
Port: 443
{"prediction": [20.353731771344123]}
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

## 1.4. Short description of how to improve the project in the future.

This entire project can work as scaffold for new any pre-trained machine learning model project. We just need to provide trained memory object to read and predict. For example, image recognition and data labeling, credit risk forecast, Predict net promoter score etc.

## 1.5. Screenshots demonstrating key steps

The screen shots are provided in 1, 2 and 3 section of this document along with explanation.

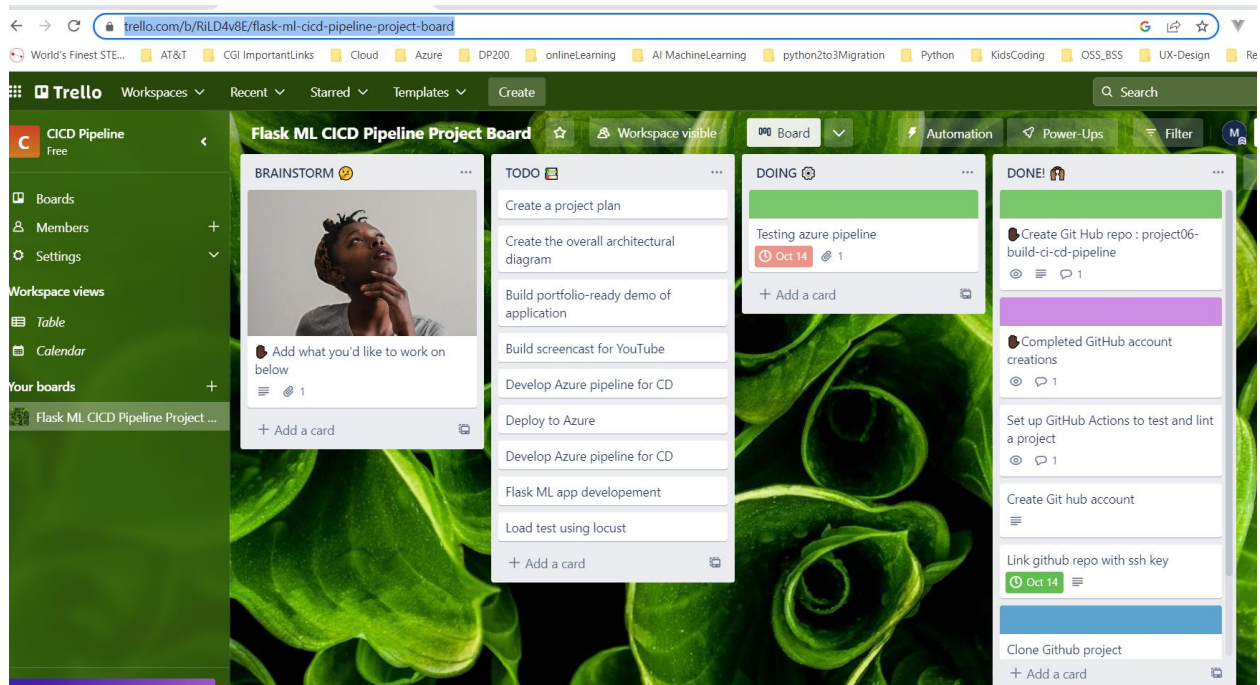
## 1.6. Project Estimate Document



	A	B	C	D
1	Work Itmes	Task	Story Point Estimation	
2				
3	Create Project Plan	Create a detail project plan		
4			3	
5				
6	Create Architeturat Diagram	Create the overall architectural diagram		
7				
8				
9	Create the Cloud-Based Development Environment	Create Git hub account		
10		Create Git hub repo : flask-sklearn-ci-cd	5	
11		Link Git hub to cloud shell/Local development account		
12		Clone the project		
13		Install required software/libraries		
14				
15	Create Project Scaffolding	Create the Makefile		
16		Create requirements.txt	1	
17		Create the Python Virtual Environment		
18		Create the project script file and test file.		
19				
20	Set up GitHub Actions to test and lint a project	Setup GitHub action		
21		Test gitub action	1	
22				
23				
24	Flask ML app developement	Clone the project into cloud shell		
25		create virtual enviroment	5	
26		Install the required python libraries		
27		Run the application to predict		
28				
29	Local Test	Run make all		
30		Run the application locally and test if it predict sucessfully	3	
31				
32				
33	Deploy the webapp and test	Deploy the web to using basic app plan		
34		Run the make_prediction_azure_app and verify the prediction resu	1	
35				
36	Setup Azure Devops	Create organisation in devops		
37		Create project in devops		
38				
39				
40	Create Azure pipeline for CD	Create project pipeline	5	
41		Setup Azure agent pool using virtual machine		
42		Run the pipeline		
43				
44				
45	Test Azure pipeline for CD	Test the piplene and fix any technical issue.	3	
46				
47				
48	Load Test	Load testing locust tool	1	
49				
50	Build portfolio-ready demo of application			
51		Demo the project	1	
52				
53	Build screencast for YouTube			
54		Create Screencast for youtube	1	
55				
56				
57				
58				
59				
60				
Project Management Plan				

## 1.7. A link to a Trello board for the project

<https://trello.com/b/RiLD4v8E/flask-ml-cicd-pipeline-project-board>



## 1.8. A link to a spreadsheet that includes the original and final project plan

H14						
	A	B	C	D	E	F
	Yearly Plan	Plan	Weekly Plan	Work Items	Task	
1				Create Project Plan	Create a detail project plan	
2						
3			9/16/2021			
4				Create Architectural Diagram	Create the overall architectural diagram	
5			9/23/2021			
6				Create the Cloud-Based Development Environment	Create Git hub account	
7					Create Git hub repo : flask-sklearn-ci-od	
8					Link Git hub to cloud shell/Local development account	
9			9/30/2021		Clone the project	
10					Install required software/libraries	
11						
12						
13				Create Project Scaffolding	Create the Makefile	
14			10/7/2021		Create requirements.txt	
15					Create the Python Virtual Environment	
16			10/14/2021		Create the project script file and test file.	
17						
18			10/21/2021	Set up GitHub Actions to test and lint a project	Setup GitHub action	
19					Test gitub action	
20						
21			10/21/2021		Deploy the web to : using basic app plan	
22			10/14/2021	Flask ML app development	Run the make_prediction_azure_app and verify the prediction result	
23			10/21/2021			
24			10/28/2021			
25			10/28/2021		Run make all	
26			10/21/2021	Local Test	Run the application locally and test if it predict sucessfully	
27			10/28/2021			
28			11/4/2021			
29			11/4/2021		Deploy the web to : using basic app plan	
30			10/28/2021	Deploy the webapp and test	Run the make_prediction_azure_app and verify the prediction result	
31			11/4/2021			
32	2021		11/11/2021			
33			11/11/2021		Create organisation in devops	
34			11/4/2021	Setup Azure Devops	Create project in devops	
35			11/11/2021			
36			11/18/2021			
37			11/18/2021		Create project pipeline	
38			11/11/2021	Create Azure pipeline for CD	Setup Azure agent pool using virtual machine	
39			11/18/2021		Run the pipeline	
40			11/25/2021			
41			11/25/2021		Test the piplene and fix any technical issue.	
42			11/18/2021	Test Azure pipeline for CD		
43			11/25/2021			
44			12/2/2021			
45			12/2/2021	Load Test	Load testing locust tool	
46			11/25/2021			
47			12/2/2021			
48			12/9/2021			
49			12/9/2021		Demo the project	
50			12/2/2021	Build portfolio-ready demo of application		
51			12/9/2021			
52			12/16/2021			
53			12/16/2021			
54			12/9/2021	Build screencast for YouTube	Create Screencast for youtube	
55			12/16/2021			
56			12/23/2021			
57			12/23/2021			
58			12/16/2021			
59			12/23/2021			
60						

</

**1.9. Professional, portfolio-ready demo of an Azure DevOps application.**

**1.10. A link to the screencast on YouTube**

## 2. Continuous Integration

### 2.1. Setup Azure Cloud Shell and clone application inside of Azure Cloud Shell.

#### 2.1.1. Create the Cloud-Based Development Environment

##### 2.1.1.1. Create a GIT repo

The screenshot shows a web browser with multiple tabs. The active tab is 'github.com/mcrjesh05/flask-sklearn-ci-cd/tree/main'. The page displays the repository structure with files like .gitignore, LICENSE, Makefile, README.md, and requirements.txt. The README.md content is visible, showing the title 'flask-sklearn-ci-cd' and a description: 'Python flask app that serves out predictions (inference) about housing prices through API calls'. The right sidebar contains metadata about the repository, including 'About', 'Releases', and 'Packages'.

File	Commit Message	Commit Hash	Time Ago
requirements.txt	Create requirements.txt	1a76e8d	2 minutes ago
README.md	Initial commit		21 minutes ago
Makefile	Create Makefile		3 minutes ago
LICENSE	Initial commit		21 minutes ago
.gitignore	Initial commit		21 minutes ago

**README.md**

### flask-sklearn-ci-cd

Python flask app that serves out predictions (inference) about housing prices through API calls

**About**  
Python flask app that serves out predictions (inference) about prices through API calls

**Releases**  
No releases published  
[Create a new release](#)

**Packages**  
No packages published  
[Publish your first package](#)

##### 2.1.1.2. Setup SSH key

Microsoft Azure Search resources, services, and docs (G+)

Home >

**myLinuxVM** Virtual machine

Search

Connect Start Restart Stop Capture Delete

Overview Activity log Access control (IAM) Tags

Location : Central India

Subscription (move) : [Pay-As-You-Go](#)

Subscription ID : 18393faf-fb05-40d8-b0ea-d522f51e82b9

Tags (edit) : [Click here to add tags](#)

Bash

```
created directory '/home/rajesh/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/rajesh/.ssh/id_rsa
Your public key has been saved in /home/rajesh/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:nLYK7lvWWGXZFtFBTbXIdEnZRNIJ81TyT5R5t13T2wk rajesh@cc-36a73aee-9f46b9f49-zzh74
The key's randomart image is:
----[RSA 3072]-----+
      .+B@^|
      ooE*&@|
      + oo.+%|
      . + . =+|
      S      .|
      = .      |
      . + o      |
      . + .      |
      .+..      |
-----[SHA256]-----+
rajesh [ ~ ]$ ls -la
```

2.1.1.3. A screenshot showing the project `flask-sklearn-ci-cd` cloned into Azure Cloud Shell.

```

rajesh [ ~ ]$ git clone git@github.com:mcrajesh05/flask-sklearn-ci-cd.git
Cloning into 'flask-sklearn-ci-cd'...
The authenticity of host 'github.com (20.207.73.82)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhbpZisF/zLDA0zPMSvHdkr4UvCOqU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
remote: Enumerating objects: 84, done.
remote: Counting objects: 100% (84/84), done.
remote: Compressing objects: 100% (74/74), done.
remote: Total 84 (delta 33), reused 20 (delta 5), pack-reused 0
Receiving objects: 100% (84/84), 242.30 KiB | 386.00 KiB/s, done.
Resolving deltas: 100% (33/33), done.
rajesh [ ~ ]$ ls -la
total 64
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:45 .
drwxrwxrwx 3 root root 4096 Nov 15 05:27 .
drwx----- 5 rajesh rajesh 4096 Nov 15 05:28 .azure
-rw----- 1 rajesh rajesh 322 Nov 15 05:45 .bash_history
-rw-r--r-- 1 rajesh rajesh 178 Apr 23 2022 .bash_logout
-rw-r--r-- 1 rajesh rajesh 645 Apr 23 2022 .bash_profile
-rw-r--r-- 1 rajesh rajesh 741 Nov 15 05:27 .bashrc
lrwxrwxrwx 1 rajesh rajesh 22 Nov 15 05:27 clouddrive -> /usr/csuser/clouddrive
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:45 flask-sklearn-ci-cd
drwx----- 2 rajesh rajesh 4096 Nov 15 05:45 .ssh
-rw-r--r-- 1 rajesh rajesh 42 Nov 15 05:27 .tmux.conf

```

#### 2.1.1.4. A screenshot showing creating virtual env and activating it

```

rajesh [ ~ ]$ python --version
Python 3.9.14
rajesh [ ~ ]$ python3 -m venv .devops
rajesh [ ~ ]$ ls -la
total 68
drwxr-xr-x 6 rajesh rajesh 4096 Nov 15 05:50 .
drwxrwxrwx 3 root root 4096 Nov 15 05:27 .
drwx----- 5 rajesh rajesh 4096 Nov 15 05:28 .azure
-rw----- 1 rajesh rajesh 386 Nov 15 05:50 .bash_history
-rw-r--r-- 1 rajesh rajesh 178 Apr 23 2022 .bash_logout
-rw-r--r-- 1 rajesh rajesh 645 Apr 23 2022 .bash_profile
-rw-r--r-- 1 rajesh rajesh 741 Nov 15 05:27 .bashrc
lrwxrwxrwx 1 rajesh rajesh 22 Nov 15 05:27 clouddrive -> /usr/csuser/clouddrive
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:50 .devops
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:45 flask-sklearn-ci-cd
drwx----- 2 rajesh rajesh 4096 Nov 15 05:45 .ssh
-rw-r--r-- 1 rajesh rajesh 42 Nov 15 05:27 .tmux.conf
-rw-r--r-- 1 rajesh rajesh 22287 Jun 17 22:09 .zshrc
rajesh [ ~ ]$ source ~/.devops/bin/activate
(.devops) rajesh [ ~ ]$

```

### 2.1.1.5. A screenshot showing the passing tests that are displayed after running the test

Run make all

```
.10.0)
Requirement already satisfied: MarkupSafe>=2.0 in /home/rajesh/.devops/lib/python3.9/site-packages (from Jinja2==3.0->Flask==2.2.2->-r requirements.txt (line 1)) (2.1.1)
Requirement already satisfied: six>=1.5 in /home/rajesh/.devops/lib/python3.9/site-packages (from python-dateutil==2.7.3->pandas==1.3.5->-r requirements.txt (line 2)) (1.16
.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /home/rajesh/.devops/lib/python3.9/site-packages (from packaging->pytest==7.2.0->-r requirements.txt (line 6)) (3
.0.9)
#hadolint Dockerfile #uncomment to explore linting Dockerfiles
pylint --disable=R,C,W1203,W0703 app.py

-----
Your code has been rated at 10.00/10 (previous run: 10.00/10, +0.00)

python -m pytest -vv test_hello.py
===== test session starts =====
platform linux -- Python 3.9.14, pytest-7.2.0, pluggy-1.0.0 -- /home/rajesh/.devops/bin/python
cachedir: .pytest_cache
rootdir: /home/rajesh/flask-sklearn-ci-cd
collected 1 item

test_hello.py::test_hello_subtract PASSED [100%]

===== 1 passed in 0.01s =====
#python -m pytest --nbval notebook.ipynb
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

## 2.2. Conduct a test run on an application in Azure Cloud Shell.

### 2.2.1. A screenshot showing the output of a test run.

```
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$ make test
python -m pytest -vv test_hello.py
===== test session starts =====
platform linux -- Python 3.9.14, pytest-7.2.0, pluggy-1.0.0 -- /home/rajesh/.devops/bin/python
cachedir: .pytest_cache
rootdir: /home/rajesh/flask-sklearn-ci-cd
collected 1 item

test_hello.py::test_hello_subtract PASSED [100%]

===== 1 passed in 0.06s =====
#python -m pytest --nbval notebook.ipynb
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

## 2.3. Use GitHub Actions to test and lint a project while writing code.

### 2.3.1. The GitHub repo contains the GitHub Actions YAML file

Data Engineer for Shell (IDA) - U...Home - Microsoft Azureproject06-build-ci-cd-pipeline/m...Create python-app.yml · mcrajes...

github.com/mcrajes05/flask-sklearn-ci-cd/commit/4895d53206f9f8901d6dc12aa0ef49b5bd3fb4f1

World's Finest STE...AT&T CGI ImportantLinksCloudAzureDP200onlineLearningAI MachineLearningpython2to3Migra

<> CodeIssuesPull requestsActionsProjectsSecurityInsightsSettings

✓ Create python-app.yml

main

mcrajes05 committed 3 minutes ago (Verified)

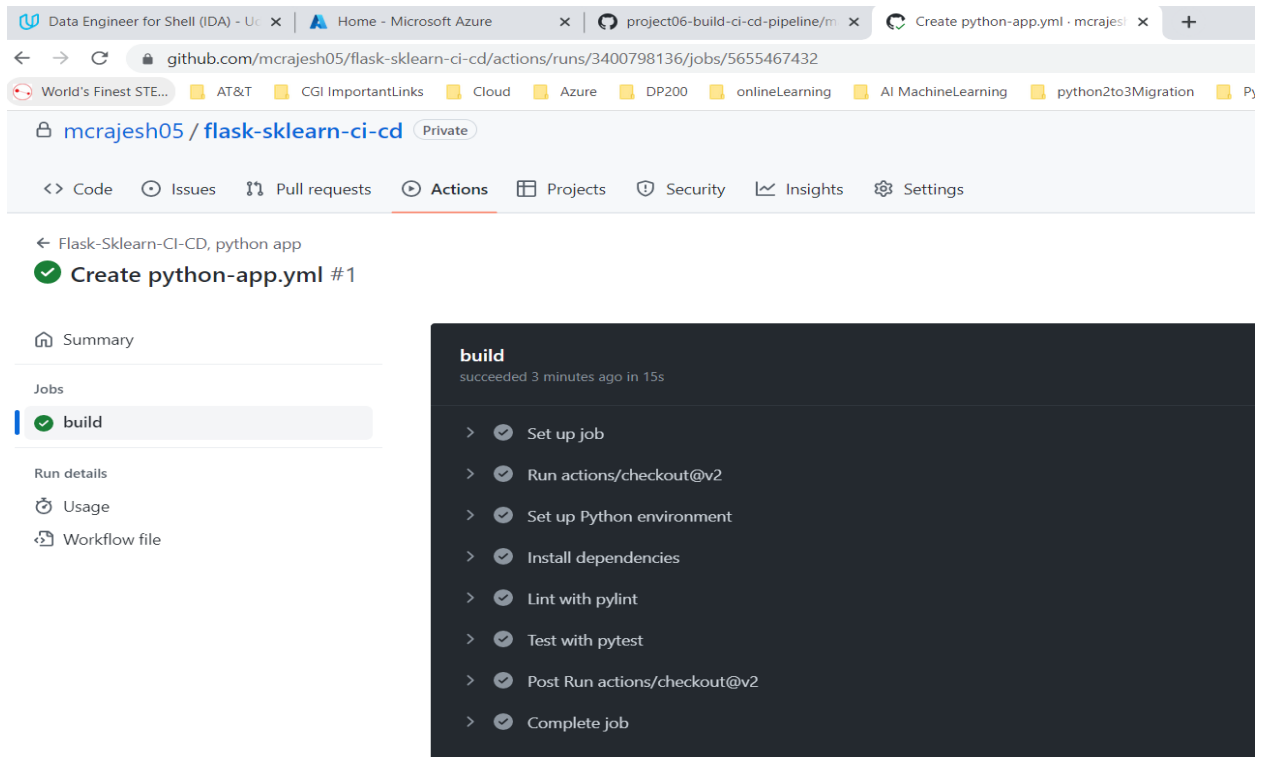
Showing 1 changed file with 30 additions and 0 deletions.

30 .github/workflows/python-app.yml

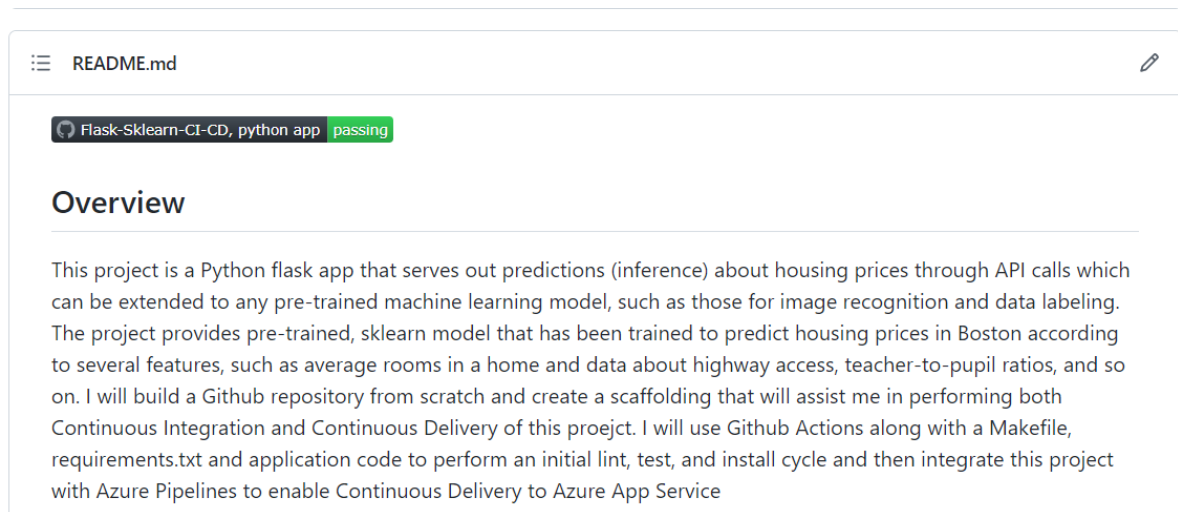
```
...  ... @@ -0,0 +1,30 @@
1 + # This workflow will install Python dependencies, run tests and lint with a single version of Python
2 +
3 + name: Flask-Sklearn-CI-CD, python app
4 +
5 + on:
6 +   push:
7 +     branches: [ "main" ]
8 +   pull_request:
9 +     branches: [ "main" ]
10 +
11 + jobs:
12 +   build:
13 +
14 +     runs-on: ubuntu-latest
15 +
16 +     steps:
17 +       - uses: actions/checkout@v2
18 +       - name: Set up Python environment
19 +         uses: actions/setup-python@v1
20 +         with:
21 +           python-version: 3.7
22 +       - name: Install dependencies
23 +         run: |
24 +           make install
25 +       - name: Lint with pylint
26 +         run: |
27 +           make lint
28 +       - name: Test with pytest
29 +         run: |
30 +           make test
```

0 comments on commit 4895d53





2.3.2. The GitHub actions badge is shown in the README, like in this screenshot.



World's Finest STE...

AT&T

CGI ImportantLinks

Cloud

Azure

DP200

onlineLearning

AI MachineLearning

python2to3Migration

Py

mcrajes05 / flask-sklearn-ci-cd

Private

<> Code

Issues

Pull requests

Actions

Projects

Security

Insights

Settings

← Flask-Sklearn-CI-CD, python app

✓

Create python-app.yml #1

Summary

Jobs

✓

build

Run details

Usage

Workflow file

build

succeeded 3 minutes ago in 15s

> ✓ Set up job

> ✓ Run actions/checkout@v2

> ✓ Set up Python environment

> ✓ Install dependencies

> ✓ Lint with pylint

> ✓ Test with pytest

> ✓ Post Run actions/checkout@v2

> ✓ Complete job

## 3. Continuous Delivery

### 3.1. Load test an application using Locust.

#### 3.1.1 A screenshot of the application running against a load test with locust

```
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$ locust -f locustfile.py --headless -u 100 --run-time 30 --host flaskmlservice.azurewebsites.net
[2022-11-16 16:25:21,704] cc-30170938-66fd9b7f98-rvhjj/INFO/locust.main: Run time limit set to 30 seconds
[2022-11-16 16:25:21,704] cc-30170938-66fd9b7f98-rvhjj/INFO/locust.main: Starting Locust 2.13.0

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
Aggregated 0 0(0.00%) 0 0 0 0 0.00 0.00

[2022-11-16 16:25:21,706] cc-30170938-66fd9b7f98-rvhjj/INFO/locust.runners: Ramping to 100 users at a rate of 1.00 per second

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
Aggregated 0 0(0.00%) 0 0 0 0 0.00 0.00

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
POST /predict 3 0(0.00%) 1705 1207 2191 1700 0.00 0.00
Aggregated 3 0(0.00%) 1705 1207 2191 1700 0.00 0.00

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
POST /predict 6 0(0.00%) 1897 1207 2191 2000 0.50 0.00
Aggregated 6 0(0.00%) 1897 1207 2191 2000 0.50 0.00

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
POST /predict 10 0(0.00%) 2312 1207 3699 2100 0.83 0.00

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
POST /predict 38 0(0.00%) 5596 1207 11054 5300 1.40 0.00
Aggregated 38 0(0.00%) 5596 1207 11054 5300 1.40 0.00

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
POST /predict 40 0(0.00%) 5892 1207 11570 5300 1.40 0.00
Aggregated 40 0(0.00%) 5892 1207 11570 5300 1.40 0.00

[2022-11-16 16:25:51,528] cc-30170938-66fd9b7f98-rvhjj/INFO/locust.main: --run-time limit reached, shutting down
[2022-11-16 16:25:51,577] cc-30170938-66fd9b7f98-rvhjj/INFO/locust.main: Shutting down (exit code 0)

Type Name # reqs # fails Avg Min Max Med req/s failures/s
-----
POST /predict 43 0(0.00%) 6296 1207 12167 5700 1.47 0.00
Aggregated 43 0(0.00%) 6296 1207 12167 5700 1.47 0.00

Response time percentiles (approximated)
Type Name 50% 66% 75% 80% 90% 95% 98% 99% 99.9% 99.99% 100% # r
eqs
---
POST /predict 5700 8200 9200 9600 11000 11000 12000 12000 12000 12000 12000
43
---
Aggregated 5700 8200 9200 9600 11000 11000 12000 12000 12000 12000 12000
43

(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

### 3.2. Deploy an application using Azure Pipelines into Azure App Services

### 3.2.1. A screenshot of Azure App Service

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the Microsoft Azure logo and a search bar. The main content area displays the 'devops' resource group overview. It includes a sidebar with navigation options like Overview, Activity log, Access control (IAM), Tags, Resource visualizer, Events, Settings, Deployments, Security, Policies, Properties, Locks, Cost Management, and Cost analysis. The main content area shows the subscription details, including the Subscription ID (18393faf-fb05-40d8-b0ea-d522f51e82b9) and Location (Central India). It also displays a list of resources, including App Service, App Service plan, Virtual machine, and Public IP address. A blue tooltip on the right side of the page reads: 'Switch between a list view of your resources and a summary chart view of resource counts. CLOSE'.

Sklearn Prediction Home

### 3.2.2. A screenshot of a successful prediction in Azure Cloud Shell

```
-rwxrwxrwx 1 rajesh rajesh 358 Nov 15 05:45 run_kubernetes.sh
-rwxrwxrwx 1 rajesh rajesh 417 Nov 15 05:45 test_hello.py
-rwxrwxrwx 1 rajesh rajesh 333 Nov 15 05:45 upload_docker.sh
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$ ./make_predict_azure_app.sh
Port: 443
{"prediction": [20.353731771344123]}
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

Prediction from Local setup

```
rajesh [ ~ ]$ source .devops/bin/activate
(devops) rajesh [ ~ ]$ cd flask-sklearn-ci-cd
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$ python app.py
* Serving Flask app 'app'
* Debug mode: on
Address already in use
Port 5000 is in use by another program. Either identify and stop that program, or start the server with a different port.
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$ ./make_prediction.sh
Port: 5000
{
  "prediction": [
    20.353731771344123
  ]
}
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$ ./make_predict_azure_app.sh
Port: 443
{"prediction": [20.353731771344123]}
(devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

## Agent Pool

dev.azure.com/mcarajesh05/flask-sklearn-ci-cd/\_settings/agentqueues?queueId=10&view=agents

World's Finest STE... AT&T CGI ImportantLinks Cloud Azure DP200 onlineLearning AI MachineLearning python2to3Migration Python KidsCoding OSS\_BSS UX-Design » Ot

mcarajesh05 / flask-sklearn-ci-cd / Settings / Agent pools / udacity

Project Settings  
flask-sklearn-ci-cd

General

- Overview
- Teams
- Permissions
- Notifications
- Service hooks
- Dashboards

Boards

- Project configuration
- Team configuration
- GitHub connections

udacity

Jobs Agents Details Security Analytics

Name	Last run	Current status	Agent version	Enabled
myLinuxVM Online		Idle	2.213.2	<input checked="" type="checkbox"/> On

Update all agents New agent

Home > Virtual machines > myLinuxVM

Virtual machines

Default Directory (vocareumvocareum.onmicrosoft...)

Create Switch to classic

Filter for any field...

myLinuxVM | Connect

Virtual machine

Search

Auto-shutdown

Backup

To improve security, enable just-in-time access on this VM. →

Private key path

~/ssh/<keyname>.pem

```
1 devopsagent devopsagent 13 Nov 2 14:08 .env
1 devopsagent devopsagent 128 Nov 3 03:14 .path
1 devopsagent devopsagent 69 Nov 2 14:13 .service
3 devopsagent devopsagent 4096 Nov 3 03:14 _diag
8 devopsagent devopsagent 4096 Nov 2 16:07 _work
14 devopsagent devopsagent 20480 Oct 11 14:42 bin
1 devopsagent devopsagent 2863 Oct 11 14:32 config.sh
1 devopsagent devopsagent 734 Oct 11 14:32 env.sh
6 devopsagent devopsagent 4096 Oct 11 14:33 externals
1 devopsagent devopsagent 9465 Oct 11 14:32 license.html
1 devopsagent devopsagent 2753 Oct 11 14:32 run-docker.sh
1 devopsagent devopsagent 2014 Oct 11 14:32 run.sh
1 devopsagent devopsagent 512 Nov 2 14:13 runsvc.sh
1 devopsagent devopsagent 4639 Nov 2 14:12 svc.sh
1 devopsagent devopsagent 108938851 Nov 2 15:29 vstsagent.tar.gz
devopsagent@myLinuxVM:~/myagent$ ./run.sh
Scanning for tool capabilities.
Connecting to the server.
Error reported in diagnostic logs. Please examine the log for more details.
- /home/devopsagent/myagent/_diag/Agent_20221103-032205-utc.log
A session for this agent already exists.
2022-11-03 03:22:12: Agent connect error: The task agent myLinuxVM already has an active session for owner myLinuxVM.. Retrying until reconnected.
```

## Lint Test

The screenshot shows the Azure DevOps interface for a pipeline named 'mcrajesh05.project06-build-ci-cd-pipeline'. The 'Jobs in run #20221103.5' section shows a list of build stages. The 'Running lint tests' job is highlighted with a red 'X' icon, indicating it failed. The job details pane on the right shows the error message: 'make: \*\*\* [Makefile:15: lint] Error 4'. The error message is: 'MyError [bash exited with code: 4]'. The job duration is 14s.

## Successful build

The screenshot shows the Azure DevOps interface for the same pipeline. The 'Jobs in run #20221103.10' section shows a list of build stages. The 'BuildJob' is highlighted with a green checkmark icon, indicating it succeeded. The job details pane on the right shows the job status as 'Succeeded'. The job duration is 39s. The job details pane also shows the job preparation parameters and the artifact produced.

--NEW screen shots

Local prediction

```
rajesh [ ~ ]$ source ~/.devops/bin/activate
(.devops) rajesh [ ~ ]$ cd flask-sklearn-ci-cd
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$ ./make_prediction.sh
Port: 5000
{
  "prediction": [
    20.353731771344123
  ]
}
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$
```

## Azure pipeline agent

```
rajesh@myLinuxVM:~/myagent$ ./config.sh
```

agent v2.213.2 (commit 4f90e68)

```
>> End User License Agreements:
```

Building sources from a TFVC repository requires accepting the Team Explorer Everywhere E repositories.

A copy of the Team Explorer Everywhere license agreement can be found at:  
/home/rajesh/myagent/license.html

Enter (Y/N) Accept the Team Explorer Everywhere license agreement now? (press enter for N)

```
>> Connect:
```

Enter server URL > https://dev.azure.com/mcarajesh05/  
Enter authentication type (press enter for PAT) >  
Enter personal access token > \*\*\*\*\*  
Connecting to server ...



```

rajesh@myLinuxVM:~/myagent$ sudo ./svc.sh start

/etc/systemd/system/vsts.agent.mcarajesh05.udacity.myLinuxVM.service
● vsts.agent.mcarajesh05.udacity.myLinuxVM.service - Azure Pipelines Agent (mcarajesh05.udacity.myLinuxVM)
   Loaded: loaded (/etc/systemd/system/vsts.agent.mcarajesh05.udacity.myLinuxVM.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2022-11-15 09:05:32 UTC; 19ms ago
     Main PID: 1887 (runsv.sh)
        Tasks: 2 (limit: 472)
       Memory: 604.0K
      CGroup: /system.slice/vsts.agent.mcarajesh05.udacity.myLinuxVM.service
              └─1887 /bin/bash /home/rajesh/myagent/runsv.sh

Nov 15 09:05:32 myLinuxVM systemd[1]: Started Azure Pipelines Agent (mcarajesh05.udacity.myLinuxVM).
rajesh@myLinuxVM:~/myagent$

```

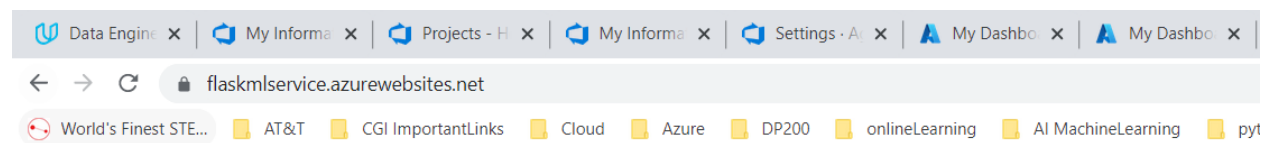
## Deploying app

```

(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$ az webapp up --name flaskmlservice --resource-group devops --runtime "PYTHON:3.10" --sku F1
The webapp 'flaskmlservice' doesn't exist
Creating AppServicePlan 'mcarajesh05_asp_1983' ...
Resource provider 'Microsoft.Web' used by this operation is not registered. We are registering for you.
Registration succeeded.
Creating webapp 'flaskmlservice' ...
Configuring default logging for the app, if not already enabled
Creating zip with contents of dir /home/rajesh/flask-sklearn-ci-cd ...
Getting scm site credentials for zip deployment
Starting zip deployment. This operation can take a while to complete ...
Deployment endpoint responded with status code 202
You can launch the app at http://flaskmlservice.azurewebsites.net
Setting 'az webapp up' default arguments for current directory. Manage defaults with 'az configure --scope local'
--resource-group/-g default: devops
--sku default: F1
--plan/-p default: mcarajesh05_asp_1983
--location/-l default: centralus
--name/-n default: flaskmlservice
{
  "URL": "http://flaskmlservice.azurewebsites.net",
  "appserviceplan": "mcarajesh05_asp_1983",
  "location": "centralus",
  "name": "flaskmlservice",
  "os": "Linux",
  "resourcegroup": "devops",
  "runtime_version": "PYTHON|3.10",
  "runtime_version_detected": "-",
  "sku": "FREE",
  "src_path": "//home//rajesh//flask-sklearn-ci-cd"
}
(.devops) rajesh [ ~/flask-sklearn-ci-cd ]$

```

## App is running



## Sklearn Prediction Home

## Prediction form deployed app



## Azure pipeline

dev.azure.com/mcarajesh05/flask-sklearn-ci-cd/\_build/results?buildId=2&view=logs&g=a314f22c-0d8d-5f4e-2e5a-7023d7141778&t=a314f22c-0d8d-5f4e-2e5a-7023d7141778

Azure DevOps mcarajesh05 / flask-sklearn-ci-cd / Pipelines / mcarajesh05.flask-sklearn-ci-cd / 20221115.2

flask-sklearn-ci-cd

Jobs in run #20221115.2  
mcarajesh05.flask-sklearn-ci-cd

Build stage

Job	Duration
BuildJob	1m 18s
Initialize job	<1s
Checkout mcarajesh05/fl...	9s
Downloading required...	31s
Installing required pyth...	3s
Running lint tests	11s
Archiving files	2s
Uploading package	11s
Post-job: Checkout mcr...	1s
Finalize Job	<1s

BuildJob

```
1 Pool: udacity
2 Queued: Just now [manage_parallel_jobs]
3 Agent: myLinuxVM
4 Started: Just now
5 Duration: 1m 18s
6
7 The agent request is already running or has already completed.
8 Job preparation parameters
9 1 Artifact produced
10 Job live console data:
11 Starting: BuildJob
12 Finishing: BuildJob
```

- Output of streamed log files from deployed application

portal.azure.com/#@mcarajesh05gmail.onmicrosoft.com/resource/subscriptions/18393faf-fb05-40d8-b0ea-d522f51e82b9/resourceGroups/devops/providers/Microsoft.Web/sites/fl...

Microsoft Azure

Home > Recent > devops > flaskmlservice

flaskmlservice | Log stream

App Service

Search

Reconnect Copy Pause Clear

API

- API Management
- API definition
- CORS

Monitoring

- Alerts
- Metrics
- Logs
- Advisor recommendations
- Health check
- Diagnostic settings
- App Service logs
- Log stream
- Process explorer

Automation

- Tasks (preview)
- Export template

```
WEBSITE_INSTANCE_ID=67f3482f905021a7c5e029b8a6698668c79fc4cb3d14f16ada1b625cb9a41c41 -e HTTP_LOGGING_ENABLED=1 -e WEBSITE_USE_DIAGNOSTIC_SERVER=True appsvc/python:3.10_20220826.8.tuxprod
2022-11-15T08:57:30.907Z INFO - Initiating warmup request to container flaskmlservice_0_89105d35 for site flaskmlservice
2022-11-15T08:57:47.339Z INFO - Waiting for response to warmup request for container flaskmlservice_0_89105d35. Elapsed time = 16.4213581 sec
2022-11-15T08:58:03.158Z INFO - Container flaskmlservice_0_89105d35 for site flaskmlservice initialized successfully and is ready to serve requests.
2022-11-15 08:57:48.545 [MainThread] [DEBUG] : Initializing AppServiceAppLogging
2022-11-15 08:57:48.546 [MainThread] [DEBUG] : Initialized AppServiceAppLogging
2022-11-15 08:57:48.548 [Thread-1 ()] [DEBUG] : Did not find any previously bound socket
2022-11-15 08:58:01.562 [Thread-3 ()] [DEBUG] : Waiting for the logs flag to be set
[2022_11_15_08_58_01] [appsvc_profiler.installer] [INFO] Code Profiler Installer is starting up
[2022_11_15_08_58_01] [appsvc_profiler.installer] [INFO] Cleaning up any existing status file which indicated signal handlers initialized status
[2022_11_15_08_58_01] [appsvc_profiler.installer] [DEBUG] APPSETTING_WEBSITE_ENABLE_DEFAULT_CODE_PROFILER : None
[2022_11_15_08_58_02] [appsvc_profiler.installer] [INFO] Attempting to install the default code profiler.
[2022_11_15_08_58_02] [appsvc_profiler.installer] [DEBUG] viztracer would save traces to /tmp/67f348_profiler_trace.json
[2022_11_15_08_58_02] [appsvc_profiler.installer] [INFO] Successfully installed code profiler.
[2022_11_15_08_58_02] [appsvc_profiler.installer] [INFO] Signal Handlers SIGUSR for needed code-profiler have been initialized for gunicorn process on instance 67f3482f905021a7c5e029b8a6698668c79fc4cb3d14f16ada1b625cb9a41c41
[2022_11_15_08_58_02] [appsvc_profiler.installer] [DEBUG] Code Profiler Installer is exiting as installation is completed
2022-11-15T10:24:14.856398782Z 169.254.129.1 - - [15/Nov/2022:10:24:14 +0000] "GET / HTTP/1.1" 200 32 "-" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36"
2022-11-15T10:25:35.983982766Z [2022-11-15 10:25:35.973] INFO in app: JSON payload: %s json_payload
2022-11-15T10:25:36.471507660Z [2022-11-15 10:25:36.470] INFO in app: inference payload DataFrame: %s inference_payload
2022-11-15T10:25:36.480203173Z [2022-11-15 10:25:36.471] INFO in app: Scaling Payload: %s payload
2022-11-15T10:25:37.626169805Z 169.254.129.1 - - [15/Nov/2022:10:25:37 +0000] "POST /predict HTTP/1.1" 200 36 "-" "curl/7.84.0"
```

## Project running on app service

### Git clone

```
(.devops) rajesh [ ~ ]$ ls -la
total 72
drwxr-xr-x 7 rajesh rajesh 4096 Nov 15 09:29 .
drwxrwxrwx 3 root root 4096 Nov 15 09:29 ..
drwx----- 5 rajesh rajesh 4096 Nov 15 05:28 .azure
-rw----- 1 rajesh rajesh 1709 Nov 15 10:34 .bash_history
-rw-r--r-- 1 rajesh rajesh 178 Apr 23 2022 .bash_logout
-rw-r--r-- 1 rajesh rajesh 645 Apr 23 2022 .bash_profile
-rw-r--r-- 1 rajesh rajesh 741 Nov 15 05:27 .bashrc
drwxr-xr-x 4 rajesh rajesh 4096 Nov 15 06:40 .cache
lrwxrwxrwx 1 rajesh rajesh 22 Nov 15 09:29 clouddrive -> /usr/csuser/clouddrive
drwxr-xr-x 5 rajesh rajesh 4096 Nov 15 05:50 .devops
drwxrwxrwx 8 rajesh rajesh 4096 Nov 15 09:58 flask-silearn-ci-co
drwx----- 2 rajesh rajesh 4096 Nov 15 06:03 .ssh
-rw-r--r-- 1 rajesh rajesh 42 Nov 15 05:27 .tmux.conf
-rw-r--r-- 1 rajesh rajesh 22287 Jun 17 22:09 .zshrc
(.devops) rajesh [ ~ ]$
```

