



Flask ML Model Deployment on Azure

U.S.A Cites Taxi Fare Price Prediction

Name: Monisha Shree Senthil Nathan

Batch Code: LISUM32

Submission: 08-05-2024

Submitted to: Data Glacier

Contents

CLOUD DEPLOYMENT	3
1. Create a docker image for ML Flaks application	3
1. Install Docker	3
2. Create Requirements.txt file	3
3. Create a Docker File	3
4. Build Docker Image.....	4
5. Check docker image	5
6. Run the Docker Container Locally	5
7. Login into Docker (<i>Only for First time</i>).....	5
8. Create a repo in Docker Hub.....	5
9. Create a tag	5
10. Push the tag with image	5
11. Verify if tag is updated in Docker hub	6
2. Create Web App Resource	6
12.1 Student Subscription	6
12.2 PAY-AS-YOU-GO subscription	8
3. Deployment	8

CLOUD DEPLOYMENT

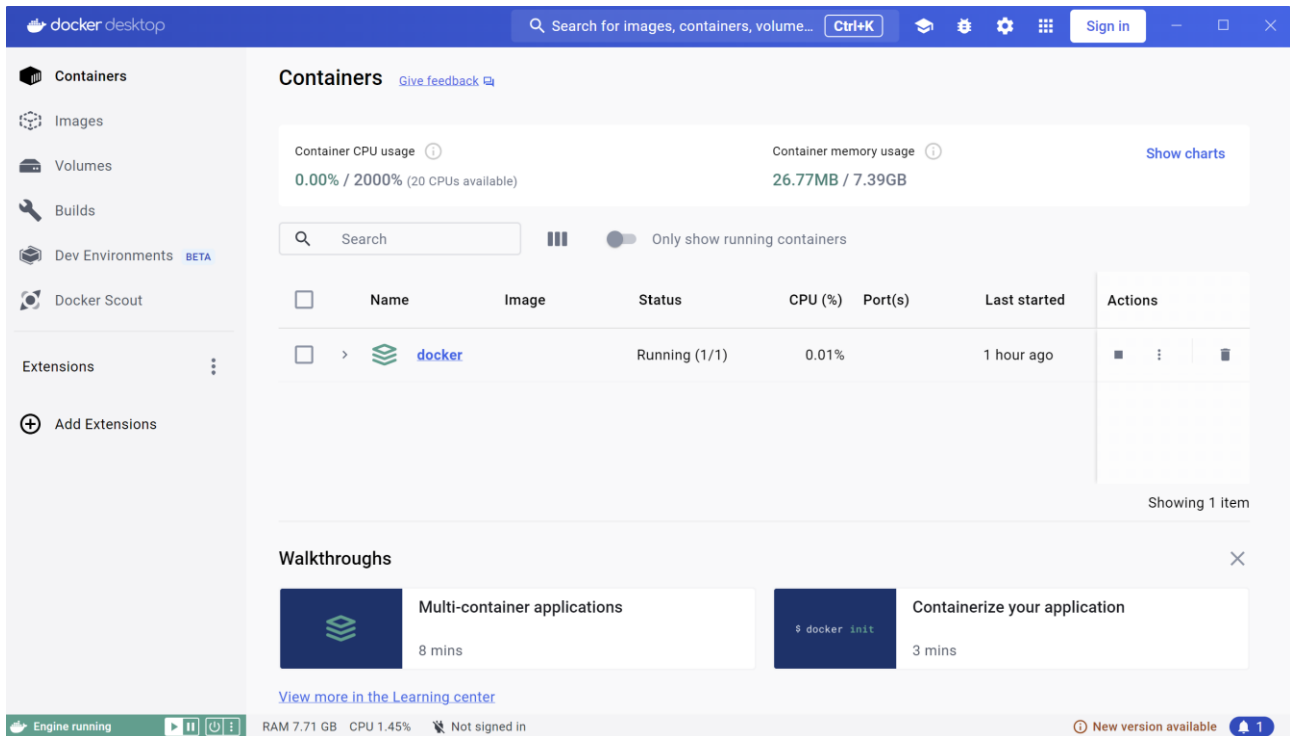
I have chosen Microsoft Azure to host my ML Flask application.

1. Create a docker image for ML Flaks application

1. Install Docker

<https://www.docker.com/products/docker-desktop/>

Once installed, start the docker service



2. Create Requirements.txt file

- `pip install pipreqs`
- `C:\Virtual_Internships\FarePricePrediction`
- `cd ..`
- `pipreqs FarePricePrediction\`

3. Create a Docker File

In VSC

- New file → Dockerfile (without any extension)

This Automatically creates a docker file.

Dockerfile content

```
# Use the Fedora base image
FROM fedora:latest

# Install Python and pip
RUN yum -y install python3 && \
    yum -y install python3-pip && \
    yum -y clean all

# Set the working directory in the container
```

```

WORKDIR /app

# Copy the current directory contents into the container at /app
COPY . /app

# Install any needed dependencies specified in requirements.txt
RUN pip3 install --no-cache-dir -r requirements.txt

# Expose the port Flask runs on
EXPOSE 5000

#set environment variable
ENV NAME OpentoAll

# Run the app model
CMD ["python3", "app.py"]

```

Note:

The use of python as base image created a large size docker image.

So instead of python:3.12.1 , I have used fedora:latest

Results:

The file size has been reduced from 3.56 GB to 2.7GB

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
app	latest	05540488b0d7	15 hours ago	2.7GB
monishashree/cab_price_prediction	2.0	05540488b0d7	15 hours ago	2.7GB
monishashree/cab_price_prediction	1.0	9a565bbf0d89	15 hours ago	3.56GB
monishashree/cab_price_prediction	latest	ae8694b29202	21 hours ago	2.95GB
postgres	latest	d60dc4bd84c0	2 months ago	431MB

4. Build Docker Image

In VSC terminal, run the below code

➤ **docker image build -t app .**

```

[+] Building 85.9s (9/9) FINISHED                                docker:default
=> [internal] load build definition from Dockerfile              0.1s
=> => transferring dockerfile: 367B                             0.0s
=> [internal] load metadata for docker.io/library/python:3.12.2 0.8s
=> [internal] load .dockerignore                                0.0s
=> => transferring context: 2B                                    0.0s
=> [1/4] FROM docker.io/library/python:3.12.2@sha256:19973e1796237522ed1fcc1357c766770b47dc15854eafdda055b65953fe5ec1 0.0s
=> [internal] load build context                                0.9s
=> => transferring context: 1.04MB                                0.8s
=> CACHED [2/4] WORKDIR /app                                    0.0s
=> [3/4] COPY . /app                                           28.5s
=> [4/4] RUN pip install -r requirements.txt                    49.6s
=> exporting to image                                           5.9s
=> => exporting layers                                           5.9s
=> => writing image sha256:ae8694b2920217cc3a51d079ffe5183ff768c35cfdc56cdb305ce0c26d1331a3 0.0s
=> => naming to docker.io/library/app                            0.0s

```

5. Check docker image

➤ `C:\>docker images`

```
C:\>docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
app           latest    ae8694b29202   3 minutes ago  2.95GB
postgres      latest    d60dc4bd84c0   2 months ago  431MB
```

6. Run the Docker Container Locally

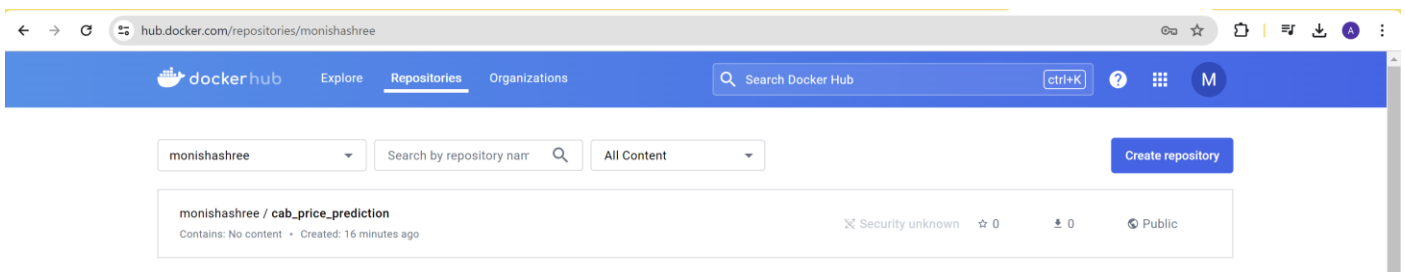
➤ `docker run -d -p 5000:5000 app`

7. Login into Docker (Only for First time)

- `C:\>docker login docker.io`
- Username and password

Login Succeeded

8. Create a repo in Docker Hub



9. Create a tag

- `C:\>docker tag app monishashree/cab_price_prediction`

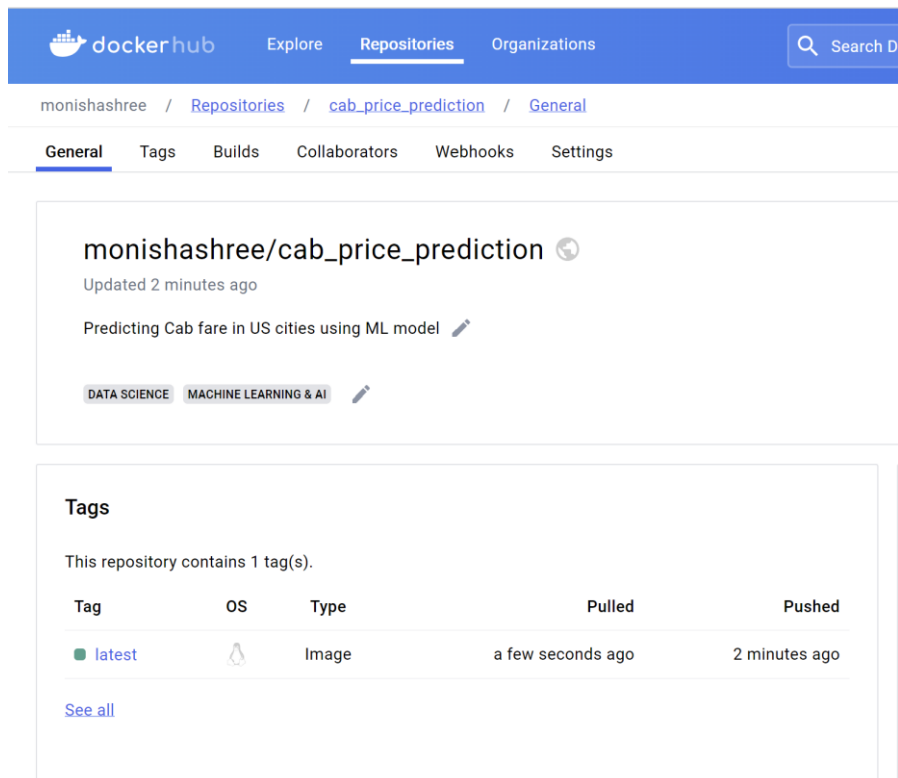
10. Push the tag with image

- `C:\>docker push monishashree/cab_price_prediction`

```
C:\>docker tag app monishashree/cab_price_prediction

C:\>docker push monishashree/cab_price_prediction
Using default tag: latest
The push refers to repository [docker.io/monishashree/cab_price_prediction]
8b9efe906b47: Pushed
485bbc9e0128: Pushed
99d0c42ff449: Pushed
9adbc4b1428d: Mounted from library/python
f52093e4f67d: Mounted from library/python
1193f41e6b14: Mounted from library/python
e077e19b6682: Mounted from library/python
21e1c4948146: Mounted from library/python
68866beb2ed2: Mounted from library/python
e6e2ab10dba6: Mounted from library/python
0238a1790324: Mounted from library/python
latest: digest: sha256:f80ad930745734173fb11c643941459369657074b77cd3307487558366b53a5a size: 2639
```

11. Verify if tag is updated in Docker hub



The screenshot shows the Docker Hub repository page for `monishashree/cab_price_prediction`. The page is under the 'General' tab. It displays the repository name, a description 'Predicting Cab fare in US cities using ML model', and two tags: 'DATA SCIENCE' and 'MACHINE LEARNING & AI'. Below this, the 'Tags' section shows a table with one tag, 'latest', which is an 'Image' type, pulled 'a few seconds ago' and pushed '2 minutes ago'. The table has columns for Tag, OS, Type, Pulled, and Pushed.

Tag	OS	Type	Pulled	Pushed
latest		Image	a few seconds ago	2 minutes ago

2. Create Web App Resource

12.1 Student Subscription

12.1.1 Basics

Home > Create a resource > Marketplace > Web App for Containers >

Create Web App

Basics Database Deployment Docker Container Networking Monitoring Tags Review + create

App Service Web Apps lets you quickly build, deploy, and scale enterprise-grade web, mobile, and API apps running on any platform. Meet rigorous performance, scalability, security and compliance requirements while using a fully managed platform to perform infrastructure maintenance. [Learn more](#)

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Azure for Students Starter ▼

Resource Group * ⓘ DefaultResourceGroup-EUS ▼
[Create new](#)

Instance Details

Name * FlaskModel ✓
.azurewebsites.net

Publish * ☐ Code ☒ Container ☐ Static Web App

Operating System * ☒ Linux ☐ Windows

Region * East US ▼

i Not finding your App Service Plan? Try a different region or select your App Service Environment.

12.1.2 Choose the default configuration

Pricing plans

App Service plan pricing tier determines the location, features, cost and compute resources associated with your app.
[Learn more](#)

Linux Plan (East US) * ⓘ

(New) ASP-DefaultResourceGroupEUS-915f

Create new

Pricing plan

Free F1 (Shared infrastructure)

Zone redundancy

An App Service plan can be deployed as a zone redundant service in the regions that support it. This is a deployment time only decision. You can't make an App Service plan zone redundant after it has been deployed [Learn more](#)

Zone redundancy

☐ Enabled: Your App Service plan and the apps in it will be zone redundant. The minimum App Service plan instance count will be three.

☒ Disabled: Your App Service Plan and the apps in it will not be zone redundant. The minimum App Service plan instance count will be one.

Review + create

< Previous

Next : Database >

12.1.3 Docker setup

Basics Database Deployment Docker Container Networking Monitoring Tags Review + create

Pull container images from Azure Container Registry, Docker Hub or a private Docker repository. App Service will deploy the containerized app with your preferred dependencies to production in seconds.

Options

Single Container

Image Source

Docker Hub

Docker hub options

Access Type *

Public

Image and tag *

monishashree/cab_price_prediction:latest

Startup Command ⓘ

12.1.4 Tags

Create Web App ...

Basics Database Deployment Networking Monitoring Tags Review + create

Tags are name/value pairs that enable you to categorize resources and view consolidated billing by applying the same tag to multiple resources and resource groups.

Note that if you create tags and then change resource settings on other tabs, your tags will be automatically updated.

Name ⓘ	Value ⓘ	Resource
PORT	5000	Web App 
		Web App

Leave other fields as default. Give create and review.

12.2 PAY-AS-YOU-GO subscription

Since my model size is large, my Docker image size is also large ie. more than 1 GB. In Free service plan(F1) only 1 GB memory is free. Therefore, I chose pay-as-you-go subscription to create a webapp resource with premium service plan that offered 4GB.

The screenshot shows the 'Create Web App' wizard in the Microsoft Azure portal, specifically the 'Project Details' step. The 'Subscription' is set to 'Pay-As-You-Go' and the 'Resource Group' is '(New) PayASyouGo'. Under 'Instance Details', the 'Name' is 'farepricepredict', 'Publish' is set to 'Container', 'Operating System' is 'Linux', and 'Region' is 'East US'. The 'Pricing plans' section shows 'Linux Plan (East US)' with a selected plan of '(New) ASP-PayASyouGo-8eb3'. The 'Zone redundancy' section has 'Disabled' selected. Navigation buttons at the bottom include 'Review + create', '< Previous', and 'Next: Database >'.

3. Deployment

- Click Review + create.
- Deployment will be started.

The screenshot displays the 'Overview' page for a Microsoft.Web.WebApp-Portal resource in the Azure portal. The deployment status is 'Deployment is in progress'. Key details include: Deployment name: Microsoft.Web.WebApp-Portal-34fb81eb-88b2, Subscription: Pay-As-You-Go, Resource group: PayASyouGo, Start time: 5/8/2024, 7:34:18 PM, and Correlation ID: 18854361-642a-4caa-b936-193f25fb0297. A table for 'Deployment details' is currently empty with the message 'No results.' The right sidebar contains links for 'Microsoft Defender for Cloud', 'Free Microsoft tutorials', and 'Work with an expert'.

- Once deployment is completed, we can find the website link under Default Domain.

- Click on the URL: flaskmodel.azurewebsites.net

The screenshot displays the Microsoft Azure portal interface for a web application named 'farepricepredict'. The left sidebar shows the navigation menu with categories like Overview, Deployment, Performance, and Settings. The main content area is divided into several sections:

- Essentials:** Provides key information about the app, including the resource group (PayASyouGo), status (Running), location (East US), subscription ID, and tags (PORT: 5000).
- Properties:** A tabbed interface showing details about the web app, domains, and hosting.
 - Web app:** Name (farepricepredict), Publishing model (Container), and Container Image (index.docker.io/monishashree/cab_price_prediction:latest).
 - Domains:** Default domain (farepricepredict.azurewebsites.net) and an option to add a custom domain.
 - Hosting:** Plan Type (App Service plan), Name (ASP-PayASyouGo-8eb3), and Operating System (Linux).
- Deployment Center:** Shows deployment logs and a link to view logs.
- Application Insights:** A link to enable application insights.
- Networking:** Displays virtual IP addresses, outbound IP addresses, and virtual network integration status.

- Finally, the webapp is hosted on Azure!

The screenshot shows the web application 'U.S.A Cities Taxi Fare Price Prediction' running on a yellow background. The application features a central form with the following fields:

- City:** A dropdown menu with 'ATLANTA GA' selected.
- Company:** A dropdown menu with 'Pink Cab' selected.
- Gender:** A dropdown menu with 'Male' selected.
- Km to Travel:** An input field.

Below the form is a green button labeled 'Predict'. The background of the application features a large, stylized black outline of a taxi cab inside a circular frame, with the text 'KNOW YOUR RIDE PRICE BEFORE YOU BOARD' and 'WE'RE ALWAYS HERE FOR YOU!' at the bottom.

KNOW YOUR RIDE PRICE BEFORE YOU BOARD
WE'RE ALWAYS HERE FOR YOU!

Attention!!

Do not forget to delete the resources as well as the service plan if you choose pay-to-go subscription