Hosting your Web Application on Azure

Azure Machine Learning provides a powerful platform for running and deploying large language models (LLMs). This guide will walk you through the steps to deploy your LLM and create an endpoint for your LLM in Azure Machine Learning.

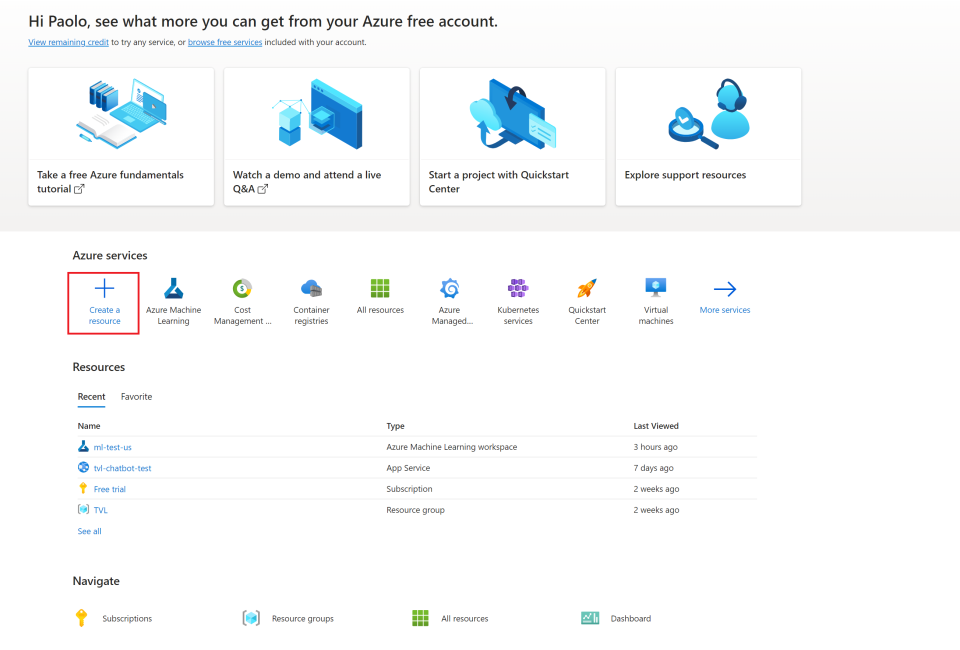
Prerequisites

- An active Azure account - Setting Up a free Azure Account - NLP Community - Confluence (iabg.de)
- An API-accessible LLM, either from closed-source providers (GPT, Claude, Mistral Large etc.) OR a self-deployed LLM - Deploying Large Language Model (LLM) on Azure

Steps to Host your Web Application

- **Sign in to Azure Portal:**
  - Open your web browser and navigate to the Azure Portal: https://portal.azure.com.
  - Sign in with your Azure account credentials

- **From the home page, select "Create a resource" and then "Web App":**



Hi Paolo, see what more you can get from your Azure free account.

View remaining credit to try any service, or browse free services included with your account.

Take a free Azure fundamentals tutorial

Watch a demo and attend a live Q&A

Start a project with Quickstart Center

Explore support resources

Azure services

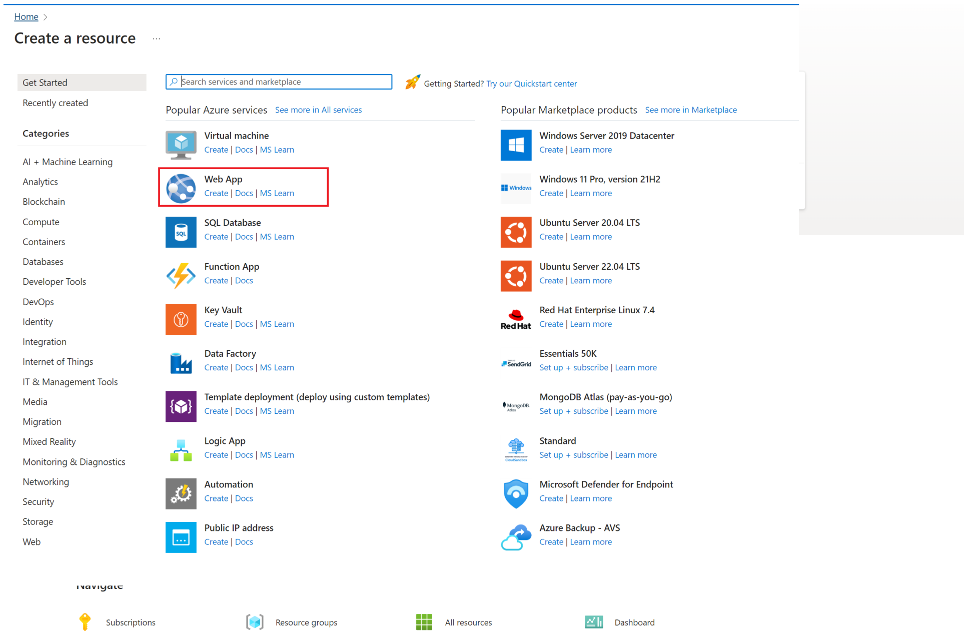
Resources

Recent Favorite

Name	Type	Last Viewed
ml-test-us	Azure Machine Learning workspace	3 hours ago
tvl-chatbot-test	App Service	7 days ago
Free trial	Subscription	2 weeks ago
TVL	Resource group	2 weeks ago

See all

Navigate



Home >

Create a resource ...

Popular Azure services See more in All services

- Virtual machine — Create | Docs | MS Learn
- Web App — Create | Docs | MS Learn
- SQL Database — Create | Docs | MS Learn
- Function App — Create | Docs
- Key Vault — Create | Docs | MS Learn
- Data Factory — Create | Docs | MS Learn
- Template deployment (deploy using custom templates) — Create | Docs | MS Learn
- Logic App — Create | Docs | MS Learn
- Automation — Create | Docs
- Public IP address — Create | Docs

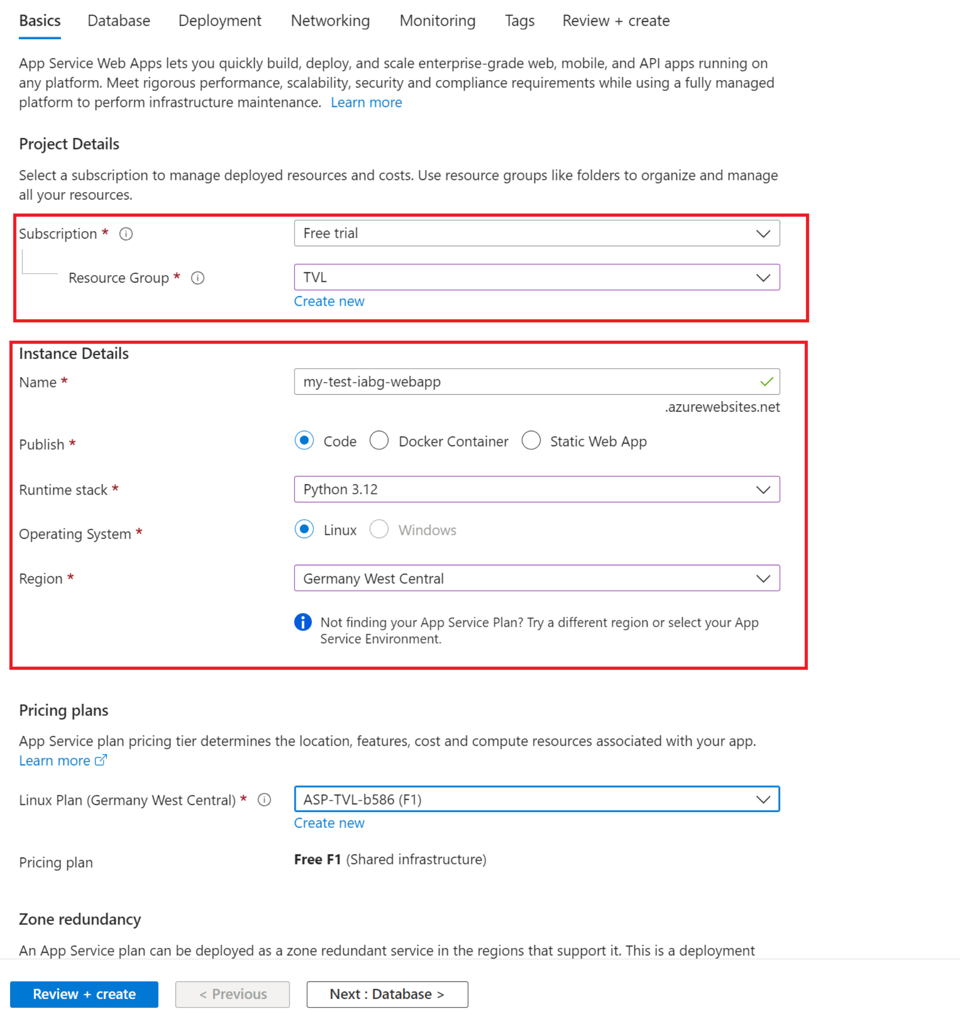
Popular Marketplace products See more in Marketplace

- Windows Server 2019 Datacenter — Create | Learn more
- Windows 11 Pro, version 21H2 — Create | Learn more
- Ubuntu Server 20.04 LTS — Create | Learn more
- Ubuntu Server 22.04 LTS — Create | Learn more
- Red Hat Enterprise Linux 7.4 — Create | Learn more
- Essentials 50K — Set up + subscribe | Learn more
- MongoDB Atlas (pay-as-you-go) — Set up + subscribe | Learn more
- Standard — Set up + subscribe | Learn more
- Microsoft Defender for Endpoint — Create | Learn more
- Azure Backup - AVS — Create | Learn more

- **Fill in the initial configuration for your Web App:**

For this tutorial, we are deploying a Python web-based application directly from a code repository. Other options (e.g. Docker deployment, other programming languages) can also be specified here.

Under "Basics", select your free trial subscription and a resource group (if you have created one). Specify a name for your web app (which will also be used as the final URL), select "Code" as publish mode, your prefered Python version and Region (e.g. "Germany West Central").



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 * ⓘ: Free trial

Resource Group * ⓘ: TVL
Create new

Instance Details

Name *: my-test-iabg-webapp ✓
.azurewebsites.net

Publish *: ● Code ○ Docker Container ○ Static Web App

Runtime stack *: Python 3.12

Operating System *: ● Linux ○ Windows

Region *: Germany West Central

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

Pricing plans

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

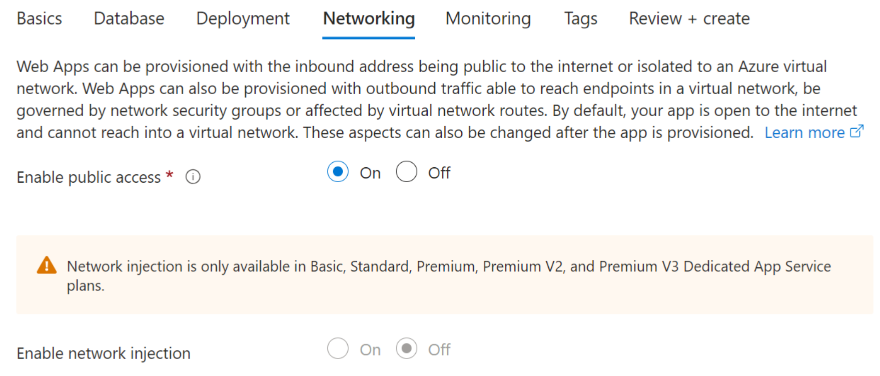
Linux Plan (Germany West Central) * ⓘ: ASP-TVL-b586 (F1)
Create new

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

You can also customize other aspects of your applications in the other tabs. For example, if you want your app to be publicly accessible, ensure that "Enable public access" is turned on under "Networking":



Web Apps can be provisioned with the inbound address being public to the internet or isolated to an Azure virtual network. Web Apps can also be provisioned with outbound traffic able to reach endpoints in a virtual network, be governed by network security groups or affected by virtual network routes. By default, your app is open to the internet and cannot reach into a virtual network. These aspects can also be changed after the app is provisioned. Learn more

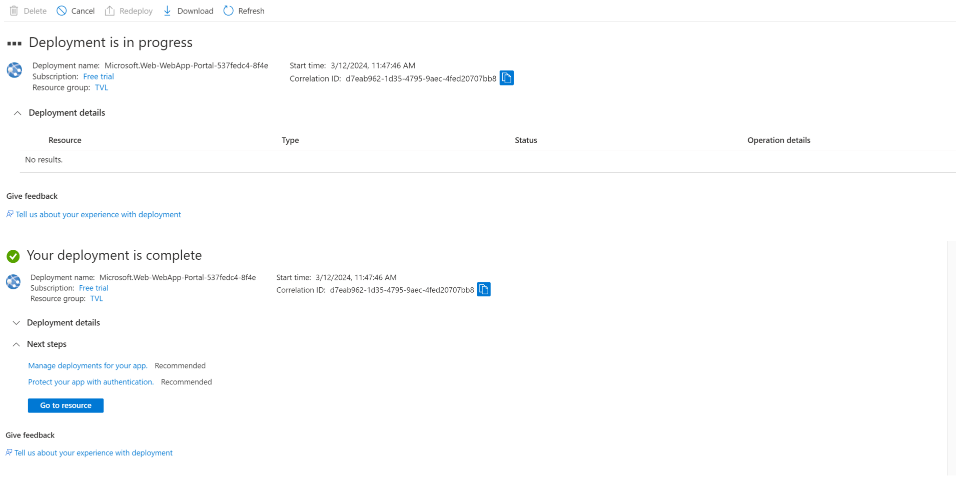
Enable public access * ⓘ: ● On ○ Off

⚠ Network injection is only available in Basic, Standard, Premium, Premium V2, and Premium V3 Dedicated App Service plans.

Enable network injection: ○ On ● Off

In general, the default options for the advanced tabs are sufficient.
When you are done, click on "Review + create" in the lower part of the screen.

- **Wait for the App to be deployed**



Deployment is in progress

Deployment name: Microsoft.Web-WebApp-Portal-537fedc4-8f4e
Subscription: Free trial
Resource group: TVL

Start time: 3/12/2024, 11:47:46 AM
Correlation ID: d7eab962-1d35-4795-9aec-4fed20707bb8

Deployment details

Resource	Type	Status	Operation details
No results.			

Give feedback

Tell us about your experience with deployment

Your deployment is complete

Deployment name: Microsoft.Web-WebApp-Portal-537fedc4-8f4e
Subscription: Free trial
Resource group: TVL

Start time: 3/12/2024, 11:47:46 AM
Correlation ID: d7eab962-1d35-4795-9aec-4fed20707bb8

Deployment details

Next steps

Manage deployments for your app. Recommended

Protect your app with authentication. Recommended

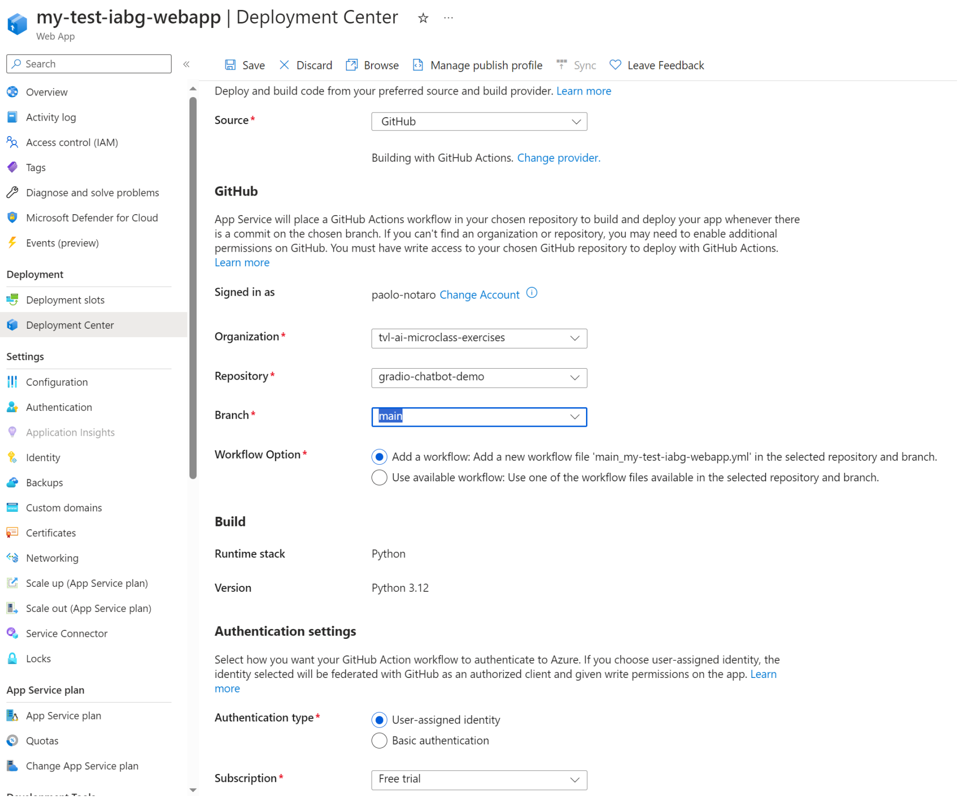
Give feedback

Tell us about your experience with deployment

when deployment is complete, click on "Go to resource". This page gives you an overview on everything related to your current WebApp deployment (usage, logs, monitoring, accessibility, etc..)

- **Connect a repository to your Web App**

  Under "Deployment Center" in the left pane, you can specify the source of your app code. Fill in the Source provider (in our example "GitHub"), repository name, branch and other details.
  This code repository will be copied in the app container. You can access it later on via SSH (see Debugging).



Deploy and build code from your preferred source and build provider. Learn more

Source * GitHub

Building with GitHub Actions. Change provider.

GitHub

App Service will place a GitHub Actions workflow in your chosen repository to build and deploy your app whenever there is a commit on the chosen branch. If you can't find an organization or repository, you may need to enable additional permissions on GitHub. You must have write access to your chosen GitHub repository to deploy with GitHub Actions. Learn more

Signed in as paolo-notaro Change Account

Organization * tvl-ai-microclass-exercises

Repository * gradio-chatbot-demo

Branch * main

Workflow Option *
● Add a workflow: Add a new workflow file 'main_my-test-iabg-webapp.yml' in the selected repository and branch.
○ Use available workflow: Use one of the workflow files available in the selected repository and branch.

Build

Runtime stack Python

Version Python 3.12

Authentication settings

Select how you want your GitHub Action workflow to authenticate to Azure. If you choose user-assigned identity, the identity selected will be federated with GitHub as an authorized client and given write permissions on the app. Learn more

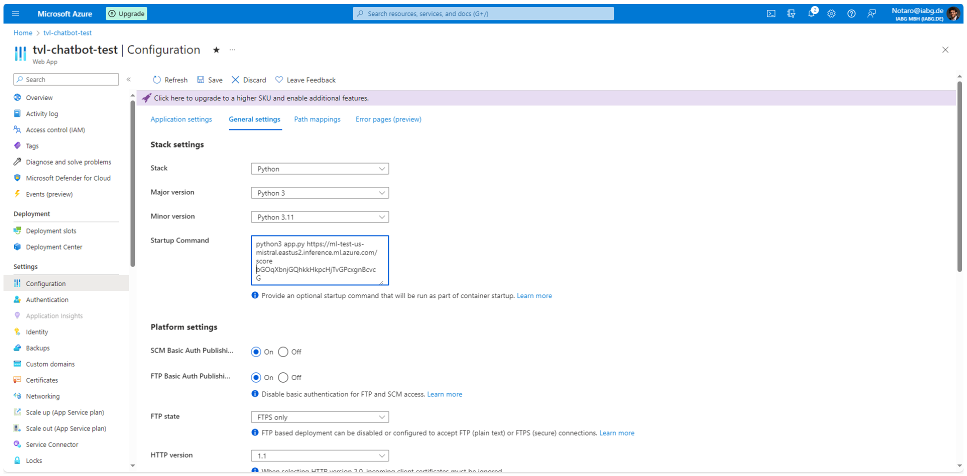
Authentication type *
● User-assigned identity
○ Basic authentication

Subscription * Free trial

- **Set up an entrypoint for your Web App**

  Under "Settings" in the left pane, select the "General Settings" tab. Then, specify your entrypoint options.

In the example below, we specified a Python start-up script with CLI arguments (LLM API endpoint URL, LLM API key).



tvl-chatbot-test | Configuration

Web App

Click here to upgrade to a higher SKU and enable additional features.

Stack settings

Stack: Python

Major version: Python 3

Minor version: Python 3.11

Startup Command:

```
python3 app.py https://ml-test-us-mistral.eastus2.inference.ml.azure.com/score bGOqXbnjGQhkkHkpcHjTvGPcxgnBcvcG
```

Provide an optional startup command that will be run as part of container startup. Learn more

Platform settings

SCM Basic Auth Publishi... ● On ○ Off

FTP Basic Auth Publishi... ● On ○ Off

Disable basic authentication for FTP and SCM access. Learn more

FTP state: FTPS only

FTP based deployment can be disabled or configured to accept FTP (plain text) or FTPS (secure) connections. Learn more

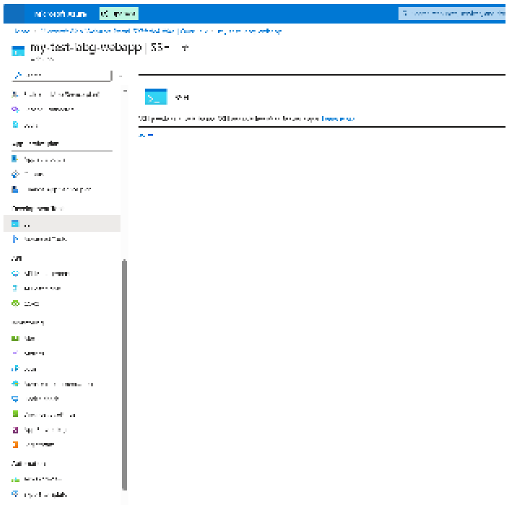
HTTP version: 1.1

When selecting HTTP version 2.0, incoming client certificates must be ignored

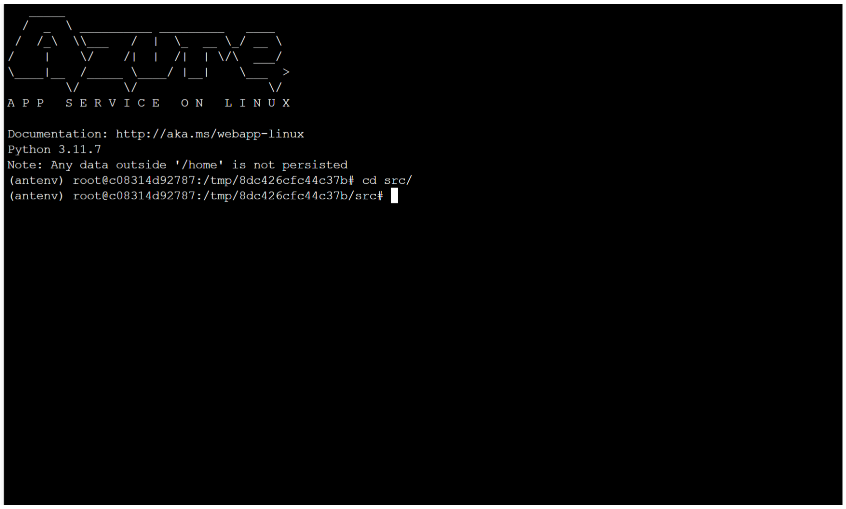
Debugging your Web App

SSH Connection

You can use the SSH connection option from the left pane to connect inside your app container. There you have all your application code and computing resources. You can also set up a debugger via SSH tunnel (e.g. in VSCode) or try running tests directly.



my-test-iabg-webapp | SS



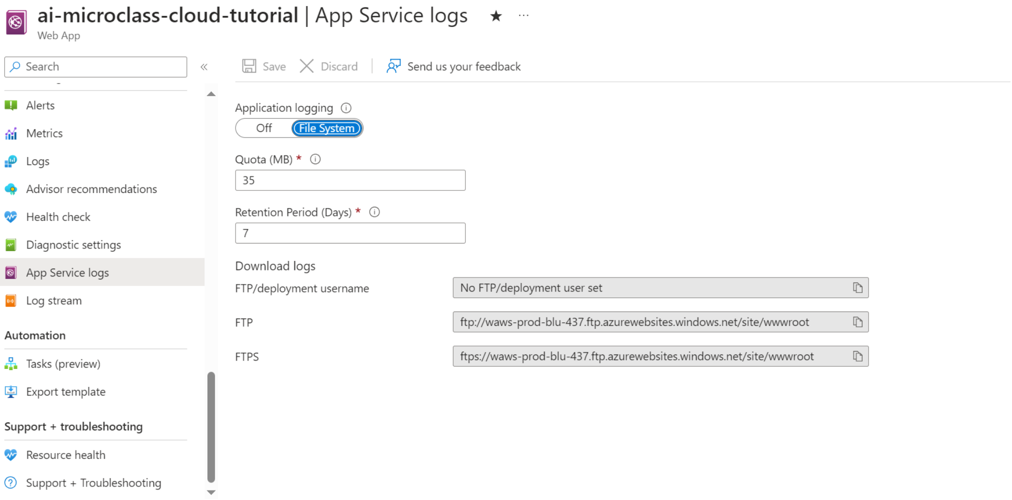
```
   _____
  /  _  \ __________ _________   ____
 /  /_\  \\___   /  |  \_  __ \_/ __ \
/    |    \/    /|  |  /|  | \/\  ___/
\____|__  /_____ \____/ |__|    \___  >
        \/      \/                  \/
A P P   S E R V I C E   O N   L I N U X

Documentation: http://aka.ms/webapp-linux
Python 3.11.7
Note: Any data outside '/home' is not persisted
(antenv) root@c08314d92787:/tmp/8dc426cfc44c37b# cd src/
(antenv) root@c08314d92787:/tmp/8dc426cfc44c37b/src#
```

Streaming Logs

Another possibility is to turn on logs in the Azure WebApp configs. In this way you can access the console stream of your program and gather diagnostic information.

To do so, select "App Service logs" from the left pane, and turn on Application logging in Filesystem mode (specify retention period and storage quota).



ai-microclass-cloud-tutorial | App Service logs

Web App

Application logging: Off [✓ File System]

Quota (MB) *: 35

Retention Period (Days) *: 7

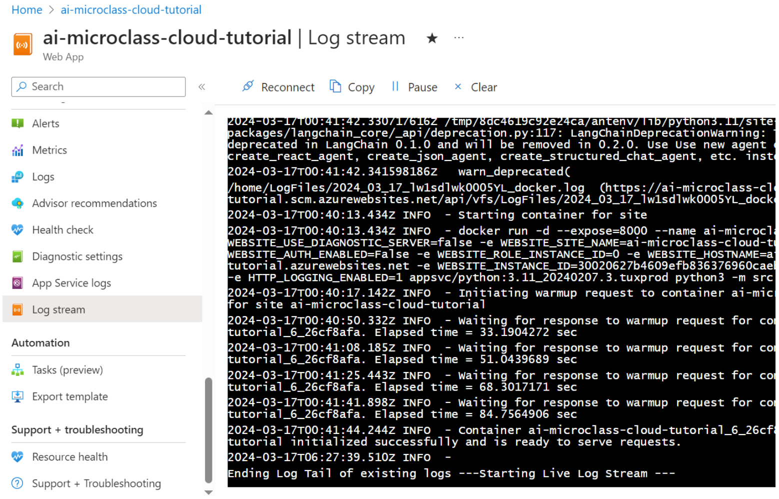
Download logs

FTP/deployment username: No FTP/deployment user set

FTP: ftp://waws-prod-blu-437.ftp.azurewebsites.windows.net/site/wwwroot

FTPS: ftps://waws-prod-blu-437.ftp.azurewebsites.windows.net/site/wwwroot

After this step, you can go to "Log stream" and access the application logs:



```
2024-03-17T00:41:42.330717616Z /tmp/8dc4619c92e24ca/antenv/lib/python3.11/site
packages/langchain_core/_api/deprecation.py:117: LangChainDeprecationWarning:
deprecated in LangChain 0.1.0 and will be removed in 0.2.0. Use Use new agent
create_react_agent, create_json_agent, create_structured_chat_agent, etc. inst
2024-03-17T00:41:42.341598186Z   warn_deprecated(
/home/LogFiles/2024_03_17_lw1sdlwk0005YL_docker.log  (https://ai-microclass-cl
tutorial.scm.azurewebsites.net/api/vfs/LogFiles/2024_03_17_lw1sdlwk0005YL_dock
2024-03-17T00:40:13.434Z INFO  - Starting container for site
2024-03-17T00:40:13.434Z INFO  - docker run -d --expose=8000 --name ai-microcl
WEBSITE_USE_DIAGNOSTIC_SERVER=false -e WEBSITE_SITE_NAME=ai-microclass-cloud-t
WEBSITE_AUTH_ENABLED=False -e WEBSITE_ROLE_INSTANCE_ID=0 -e WEBSITE_HOSTNAME=a
tutorial.azurewebsites.net -e WEBSITE_INSTANCE_ID=30020627b4609efb836376960cae
-e HTTP_LOGGING_ENABLED=1 appsvc/python:3.11_20240207.3.tuxprod python3 -m src
2024-03-17T00:40:17.142Z INFO  - Initiating warmup request to container ai-mic
for site ai-microclass-cloud-tutorial
2024-03-17T00:40:50.332Z INFO  - Waiting for response to warmup request for co
tutorial_6_26cf8afa. Elapsed time = 33.1904272 sec
2024-03-17T00:41:08.185Z INFO  - Waiting for response to warmup request for co
tutorial_6_26cf8afa. Elapsed time = 51.0439689 sec
2024-03-17T00:41:25.443Z INFO  - Waiting for response to warmup request for co
tutorial_6_26cf8afa. Elapsed time = 68.3017171 sec
2024-03-17T00:41:41.898Z INFO  - Waiting for response to warmup request for co
tutorial_6_26cf8afa. Elapsed time = 84.7564906 sec
2024-03-17T00:41:44.244Z INFO  - Container ai-microclass-cloud-tutorial_6_26cf
tutorial initialized successfully and is ready to serve requests.
2024-03-17T06:27:39.510Z INFO  -
Ending Log Tail of existing logs ---Starting Live Log Stream ---
```