



Deployment with Flask/FastAPI

cloud platforms

So... You Built a Web App. How Do You Get It Out Into the World?

A casual lecture for normal humans who don't speak Python-ese.

Alright folks, gather around.

You've built something cool, maybe a tiny website, maybe an app that lets users track their cat's mood swings, maybe a tool that reminds you to drink water because none of us are hydrating like adults.

Now we hit the next big question:

“How do I get this thing onto the internet so other people can actually use it?”

Today we're talking about deploying your app using **Flask**, **FastAPI**, and those magical cloud platforms that pretend they're simple but definitely have the power to fry your brain if you stare too long (looking at you, AWS).

Don't worry. We're keeping it simple, fun, and beginner-friendly.

Part 1 , What the Heck Is Flask or FastAPI?

Think of Flask and FastAPI as tiny restaurant kitchens.

- Flask = a small, cozy taco truck.
- FastAPI = the electric modern food truck with neon lights that makes tacos **fast**.

Both let you serve “routes,” which are basically the buttons on a website.

Click a button, you're ordering something.

Flask/FastAPI cooks it and hands it back.

The problem?

Your cute taco truck is currently parked **in your living room** (your laptop).

No one else can walk in and order tacos.

Deployment is basically:

“Let’s roll the truck outside so the world can eat.”

Part 2 , What Does ‘Deploy’ Mean? (The human version)

Deployment is just the process of putting your app on a computer **that never sleeps, never shuts down, and doesn’t randomly restart for Windows updates.**

These computers live in “the cloud.”

Not the fluffy ones, think giant warehouse of machines humming like an army of bees.

When your app is deployed:

- Anyone can visit it through a URL
- It keeps running even when you’re asleep
- It doesn’t crash because your laptop battery got to 1%

So deployment =

Moving your app from ‘my laptop’ → ‘the internet’

Part 3 , Deploying With Flask or FastAPI (The Simple Version)

Flask and FastAPI apps need two things to survive “the outside world”:

1. A server to run on (the cloud computer)

Amazon, Google, Microsoft, or Heroku will rent you one by the hour.

It’s like Airbnb for apps.

2. A “WSGI/ASGI server”

This sounds scary.

It’s not.

It’s just a tool that takes your Python app and makes it strong enough to handle actual website visitors.

- Flask uses WSGI (like Gunicorn or uWSGI)
- FastAPI uses ASGI (like Uvicorn or Hypercorn)

Think of these as the “bouncers” who let people into your taco truck one at a time so nobody knocks the table over.

Part 4 , Using Cloud Platforms (aka The Easy Button, Mostly)

Let's talk about the friendly cloud platforms, the ones that don't bite too hard.

1. Heroku

The beginner's favorite.

Heroku is the “microwave oven” of deployment:

- You press a button
- Your app gets hot
- Magic happens
- You pretend you did something impressive

You push your code to Heroku, and it basically says,
“Cool, I got this,”
and does everything for you.

Perfect for simple apps.

2. Render

Render is like Heroku's younger, fitter cousin who eats organic.

- Free tier
- Very beginner-friendly
- Handles Flask/FastAPI easily

You give it your GitHub project and it deploys automatically.

3. AWS / Google Cloud / Azure

These are the “IKEA of cloud platforms”:

- Lots of power
- Endless options
- You will cry at least once while assembling them
- But once it's built, it's rock solid

Perfect if you want:

- Millions of users
- Enterprise-grade reliability
- Enough settings to feel like you're piloting a spaceship

But they're overkill for beginners unless you enjoy pain.

4. PythonAnywhere

Very simple, very friendly.

Like a retirement home for Python apps:

- Calm
- Slow-paced
- Reliable
- Won't surprise you

Great for small hobby projects.

Part 5 , A Simple Deployment Recipe (No Tech Jargon)

Here's deployment in human language:

Step 1: Package your app

Put your app files together and list the ingredients (dependencies).

Step 2: Choose a cloud home

Heroku if you want easy.

AWS if you want advanced.

Render if you want modern and simple.

Step 3: Add a server (Gunicorn/Uvicorn)

This is like adding a motor to your food truck.

Now it can handle real traffic.

Step 4: Send your files to the cloud

Usually through:

GitHub

A zip upload

Or a magic button that says "Deploy"

Step 5: Press Start

The platform turns on your app.

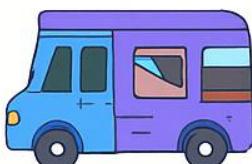
Boom. You're live.

Whw, I Built a Web App. How Do I Get It on the Internet

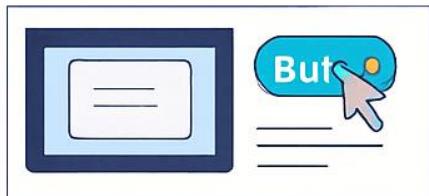
WHAT THE HECK IS FLASK OR FASTAPI?



FLASK



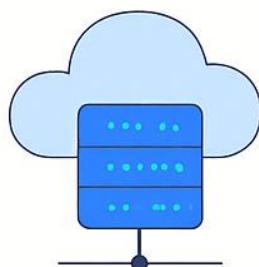
FASTAPI



Click a button, you're ordering something.

Flask/FastAPI cooks it and hands it back.

WHAT DOES “DEPLOY” MEAN?



When your app is deployed:

- Anyone can visit it through a URL
- It keeps running even when you're asleep
- It doesn't crash because your laptop battery got to 1%

DEPLOYING WITH FLASK OR FASTAPI

Two things you need

- A server to run on
- A WSGI/ASGI server
- fool that makes your Python app strong enough to hang actual website visitors

A WSGI/ASGI server



UCVICORI

Gunicorn



FUICKRO

Uvicorn

USING CLOUD PLATFORMS

Heroku

A microwave oven of dep

Render

Simple and friendly

AWS/GC

Endless, of endless option

Python Anywhere

Simple

Part 6 , How to Explain Deployment to a Friend

If someone asks what deployment is, just say:

“It’s like taking a project off your computer and putting it on the internet so people can actually see and use it.”

If they say, “Oh, that sounds complicated,” reply:

“It is. But luckily other people built tools so I don’t have to understand all the complicated parts.”

And they will nod respectfully.

Final Thoughts

Deployment used to be a terrifying thing that only wizards in basements could do.

But today?

Platforms like Heroku, Render, and even AWS make it surprisingly doable, even if you:

- don’t code
- don’t know what a server is
- panic when Python prints an error message

With Flask or FastAPI, plus a friendly cloud platform, you can go from

“I built something cool” → “The world can use it.”

And trust me, seeing your app live online for the first time?

That feeling hits harder than caffeine at 7 AM.

n hear it.”  