EC2 Flask App Deployment – DevOPS Project

This project demonstrates how to deploy a simple Python Flask web application on an AWS EC2 instance using a virtual environment and `systemd`. It's part of a hands-on DevOps learning path focused on cloud-native deployment, automation, and environment management.

📅 Tech Stack

- **Python 3.12+**
- **Flask**
- **AWS EC2 (Ubuntu)**
- **Virtual Environment (``)**
- **systemd (for persistent background service)**
- **GitHub + SSH Authentication**

🗜️ Project Features

- Flask app hosted on an AWS EC2 Ubuntu instance
- Virtual environment for Python package isolation
- systemd for auto-starting app on EC2 reboot
- SSH-based Git push/pull to GitHub repo
- Uses either port 80 or 5000
- Easily extendable to Docker, ECS, CodePipeline, S3, and CloudWatch

---

📁 Folder Structure

```
DevOPS/
├── app.py
├── requirements.txt
├── venv/               # Virtual environment (not committed)
├── setup.sh (optional automation script)
└── README.md
```

🪑 Setup Instructions (On EC2)

🕰️1. Clone the Repository Using SSH

```
git clone git@github.com:VasanthKumar1698/DevOPS.git
cd DevOPS
```

> ⚠️Use **SSH** to avoid GitHub credential issues (no need for username/password or PAT).

🕰️2. Install Python Virtual Environment Support

```
sudo apt update
sudo apt install python3.12-venv -y
```

🕰️3. Create and Activate Virtual Environment

```
python3 -m venv venv
source venv/bin/activate
```

🕰️4. Install Required Python Packages

```
pip install -r requirements.txt
```

> If `requirements.txt` doesn't exist, just run:
>
> ```
> pip install flask
> ```

🕰️5. Run Flask App on Port 5000 (for development)

Edit `app.py` if needed:

```
app.run(host='0.0.0.0', port=5000)
```

Then run:

```
python3 app.py
```

🔗Access the app:

```
http://<your-ec2-public-ip>:5000
```

🚻Make sure port **5000** is allowed in your EC2 **inbound rules**.

🔐Run Flask on Port 80 (Optional)

If you want to use port 80:

1. Update `app.py`:

   ```
   app.run(host='0.0.0.0', port=80)
   ```

2. Run it using `sudo`:

   ```
   sudo venv/bin/python app.py
   ```

⚙️Set Up `systemd` for Auto-Restart

1. Create the service file:

```
sudo nano /etc/systemd/system/flaskapp.service
```

1. Paste this content:

```
[Unit]
Description=Flask App
After=network.target

[Service]
User=vasanth
WorkingDirectory=/home/vasanth/DevOPS
ExecStart=/home/vasanth/DevOPS/venv/bin/python app.py
Restart=always
```

```
[Install]
WantedBy=multi-user.target
```

1. Enable and start the service:

```
sudo systemctl daemon-reload
sudo systemctl enable flaskapp
sudo systemctl start flaskapp
```

1. Check status:

```
sudo systemctl status flaskapp
```

🔓 Optional: Automation Script (`setup.sh`)

Create a bash file to automate the setup:

```
#!/bin/bash
sudo apt update
sudo apt install python3.12-venv -y
python3 -m venv venv
source venv/bin/activate
pip install -r requirements.txt
```

Make it executable:

```
chmod +x setup.sh
./setup.sh
```

🛁 Add `.gitignore`

```
echo "venv/" >> .gitignore
echo "__pycache__/" >> .gitignore
git add .gitignore
git commit -m "Add .gitignore for venv and cache"
git push
```

🔗 Sample `app.py`

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def home():
    return "Hello from Flask on AWS EC2!"

if __name__ == '__main__':
    app.run(host='0.0.0.0', port=5000)
```

📦 `requirements.txt`

```
flask
```

🔧 Troubleshooting

Issue	Fix
`Permission denied`	Use `sudo` if running on port 80
`ModuleNotFoundError: flask`	Activate venv and `pip install flask`
`venv` not created	Install `python3.12-venv` package
GitHub push fails	Use SSH or Personal Access Token

📊 What's Next (DevOps Progression)

Stage	AWS Service	Purpose
✅ Done	EC2 + Flask	Deploy app
⏭️ Next	S3	Host static website / store logs
🔐 IAM	Secure EC2 / S3 access via roles	
🈂️ CloudWatch	Monitor EC2 and app logs	
⟳ CodePipeline	CI/CD deployment from GitHub	

Stage	AWS Service	Purpose
🐳 Docker + ECS	Containerize and scale app	

Author

Vasanth Kumar Palla\ ♈ Missouri, USA\ 🆖 DevOps | Cloud | System Admin\ Ⓜ vasanthkumar16.palla@gmail.com\ 🌐 GitHub Profile

---

📄 License