**Deploy Your OCR Flask App on AWS EC2: A Step-by-Step Guide**

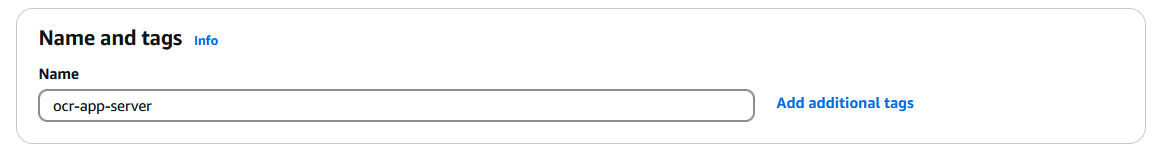
### ****STEP 1: Launch EC2 Instance****

#### ****Requirements:****

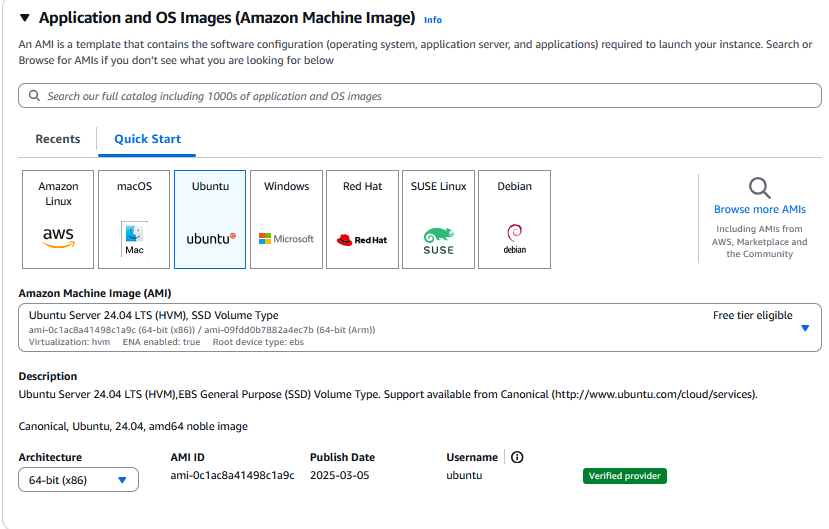
1. AWS account (either AWS Educate or regular)
2. Free-tier eligible instance
3. Basic terminal access (for SSH)

#### ****Steps:****

1. **Go to AWS EC2 Dashboard:**
2. Visit the [AWS EC2 Console](https://console.aws.amazon.com/ec2/" \t "_blank)
3. Click **Launch Instance** to create a new instance.
4. **Name Your Instance:**
5. Name your instance something like ocr-app-server.

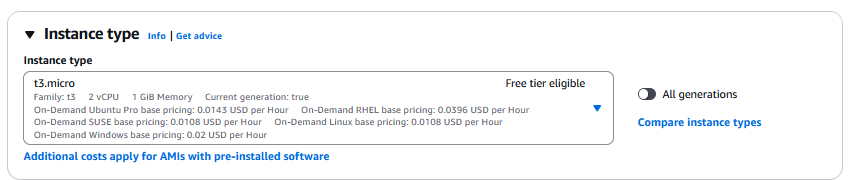


1. **Choose an OS:**
2. Select **Ubuntu**

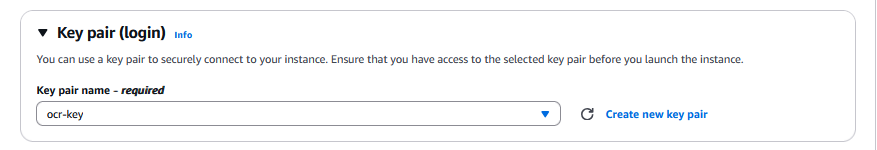


1. **Instance Type:**

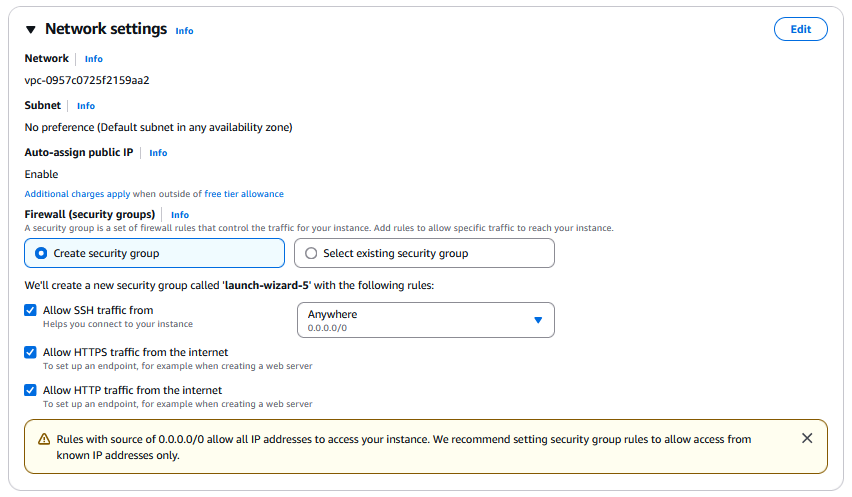
Select **t2.micro** (free-tier eligible).



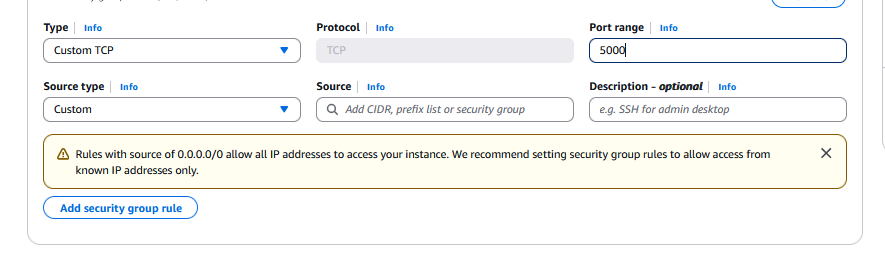
1. **Key Pair:**
2. Create a new key pair (e.g., ocr-key).



1. Download the .pem file and save it in a secure location (you’ll need it for SSH access).
2. **Configure Security Group:**

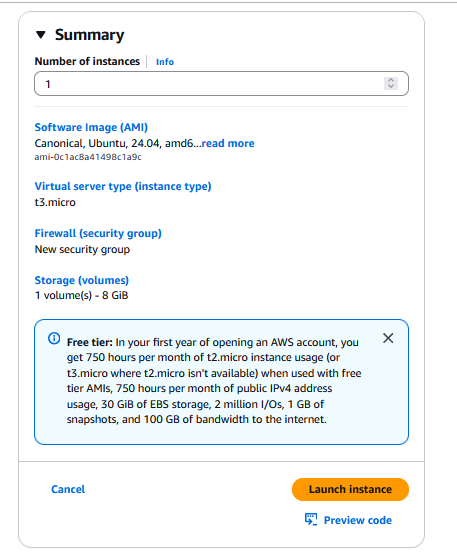


1. Now click on “edit”
2. Add the following rules new add security group rule:
3. **Custom TCP (port 5000)**: Flask runs on port 5000 by default.



1. **Launch:**

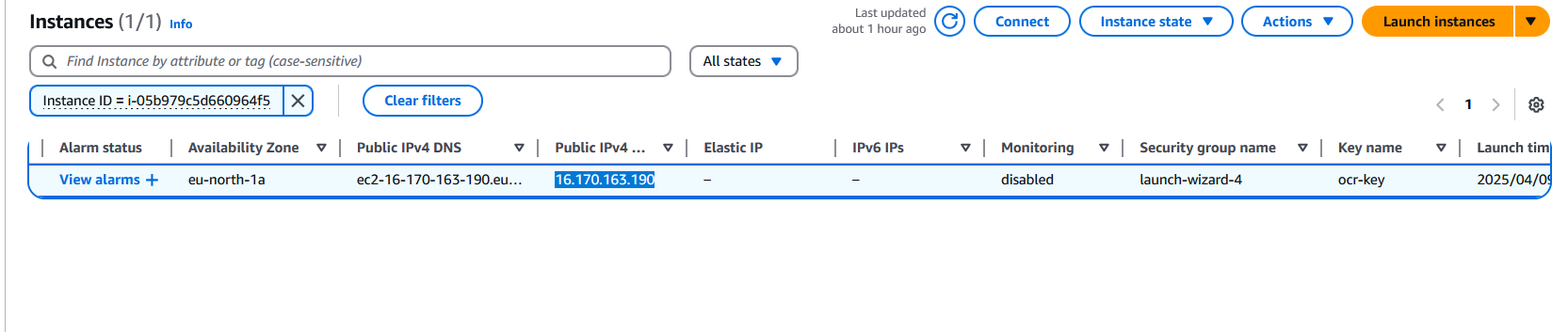
Click **Launch Instance** once everything is configured.



### ****STEP 2: Connect to EC2 via SSH****

#### ****Get the Public IPv4 Address:****

1. Go to **EC2 > Instances** in the AWS Console.
2. Copy the **Public IPv4 address** (e.g., 13.211.xx.xx).



#### ****Connect to EC2 via SSH: (search powershell, right click and run as admin)****

1. **Navigate to your .pem file directory:**

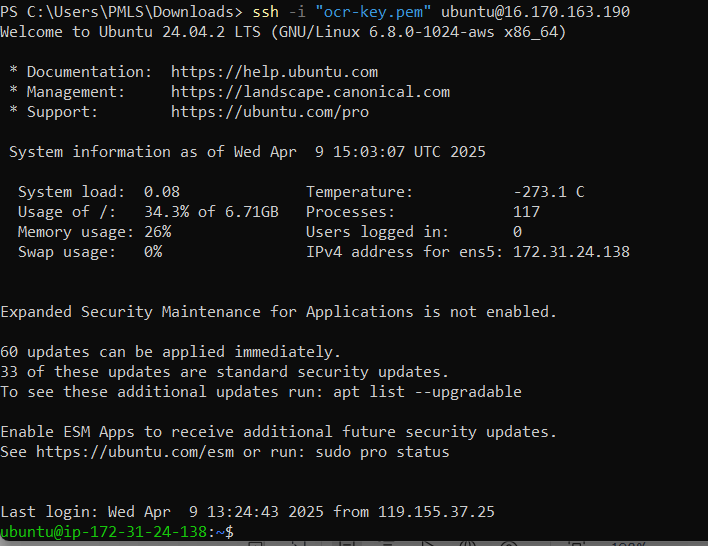
cd path/to/your/pem/file

C:\Users\PMLS\Downloads>(this contains the pem file)



1. **SSH into the instance:**

ssh -i "ocr-key.pem" ubuntu@<your-public-ip>



Replace <your-public-ip> with the actual IP (e.g., ubuntu@13.211.xx.xx). (in our case 16.170.163.190)

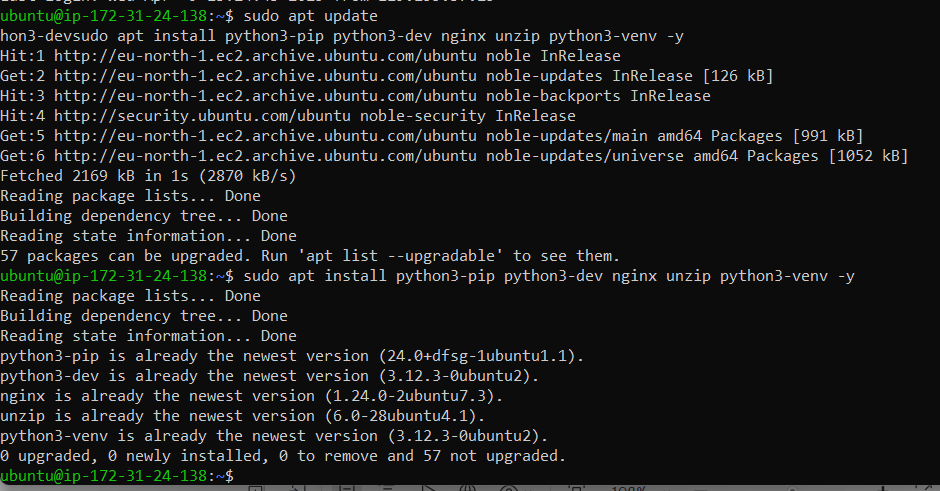
### ****STEP 3: Setup Environment on EC2****

Once logged in, set up the environment:

#### ****Update and Install Required Tools:****

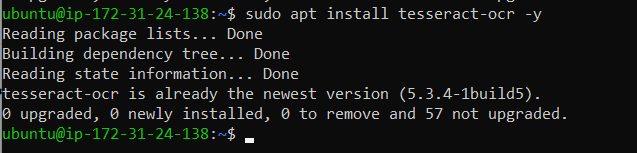
sudo apt update

sudo apt install python3-pip python3-dev nginx unzip python3-venv -y



#### ****Install Tesseract OCR:****

sudo apt install tesseract-ocr -y



### ****STEP 4: Upload Project to EC2 (USE NEW POWERSHELL)****

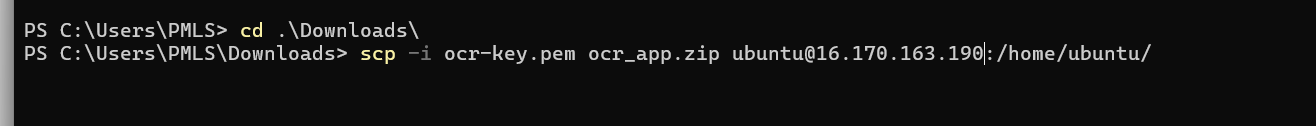
#### ****Upload the Project:****

1. On your local machine, zip your project (if not already zipped):



1. Upload the zip file to EC2 using scp:

*scp -i ocr-key.pem ocr\_app.zip ubuntu@<your-public-ip>:/home/ubuntu/*



#### ****Unzip and Install Dependencies: (BACK TO PREVIOUS POWERSHELL)****

1. SSH into EC2 (if not already connected).
2. Unzip the file:

*unzip ocr\_app.zip*

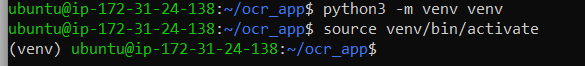


*cd ocr\_app*



1. **Create a Virtual Environment**:

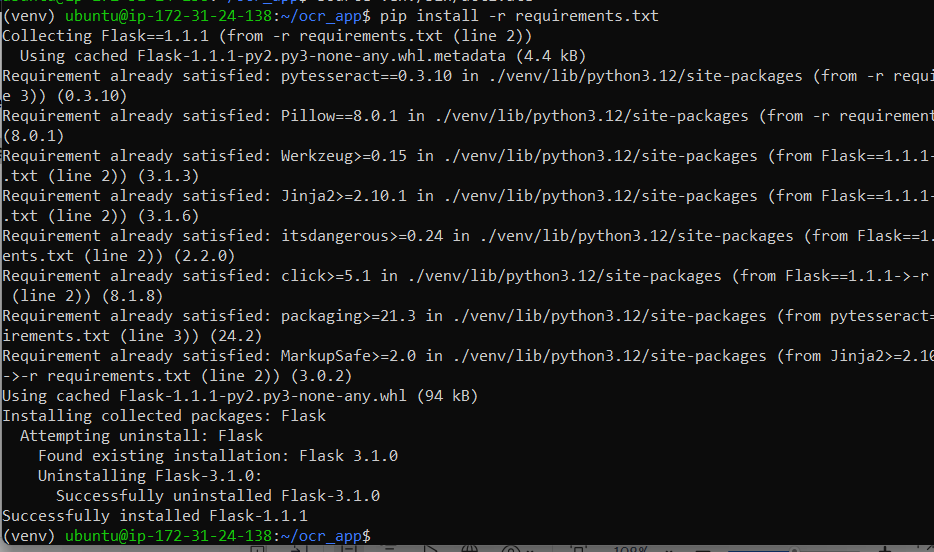
*python3 -m venv venv*

*source venv/bin/activate*

1. **Install the Dependencies:**

Once the virtual environment is active, install the required Python packages:

*pip install -r requirements.txt*



### ****STEP 5: Run the App****

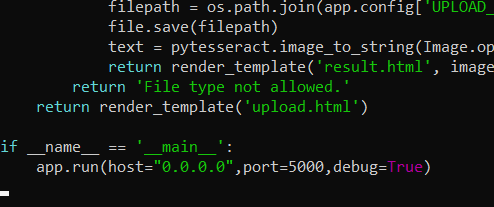
Now, you can run your Flask app:

1. Modify app.py to allow public access:
2. Find the line where the app runs:

app.run(host='127.0.0.1', port=5000)

1. Change it to:

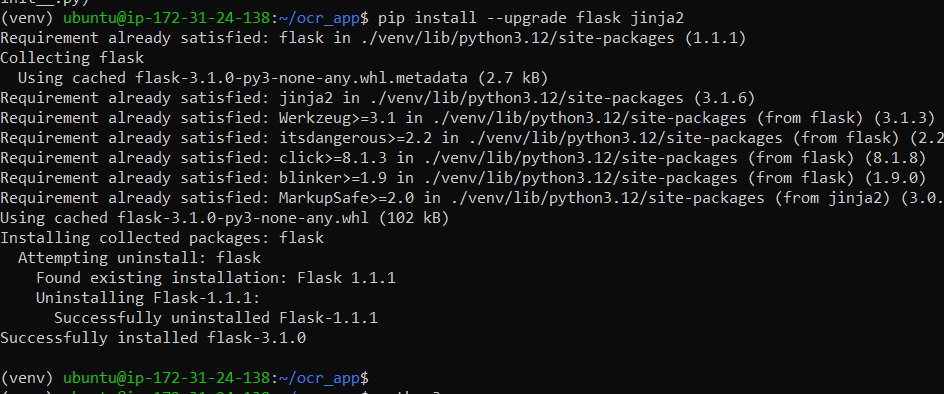
app.run(host='0.0.0.0', port=5000)



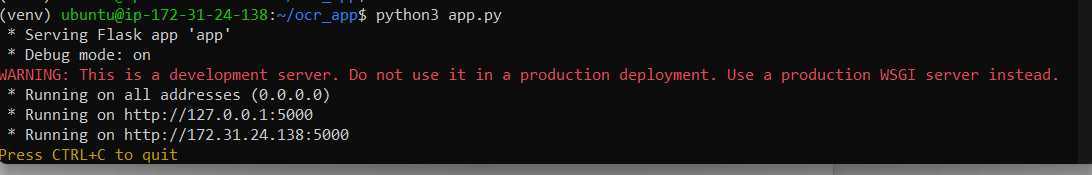
1. **Install dependency Run the Flask app:**

Dependency:

*pip install --upgrade flask jinja2*



*python3 app.py*



You should see something like:

Running on <http://127.0.0.1:5000>

### ****STEP 6: Access the App in Browser****

Go to your browser and visit the URL:

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

In our case:

<http://16.170.163.190:5000>

You should see your OCR app homepage.

### ****Optional Step: Deactivating the Virtual Environment****

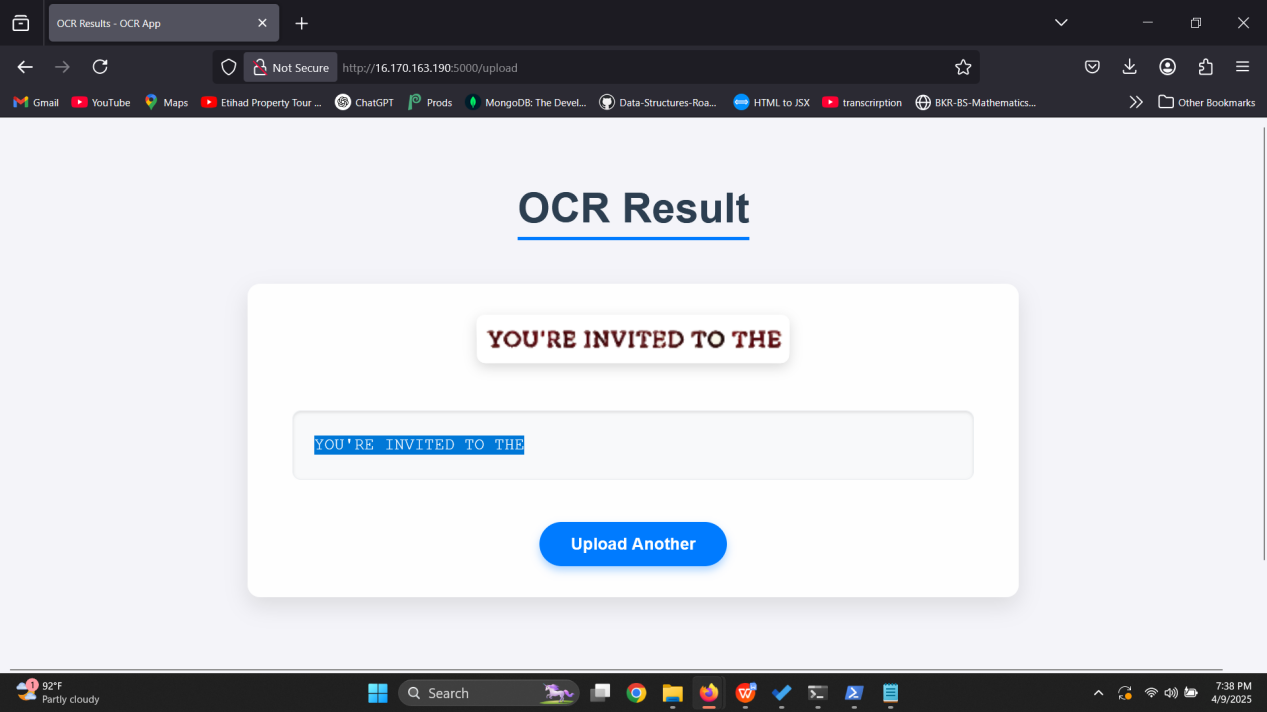
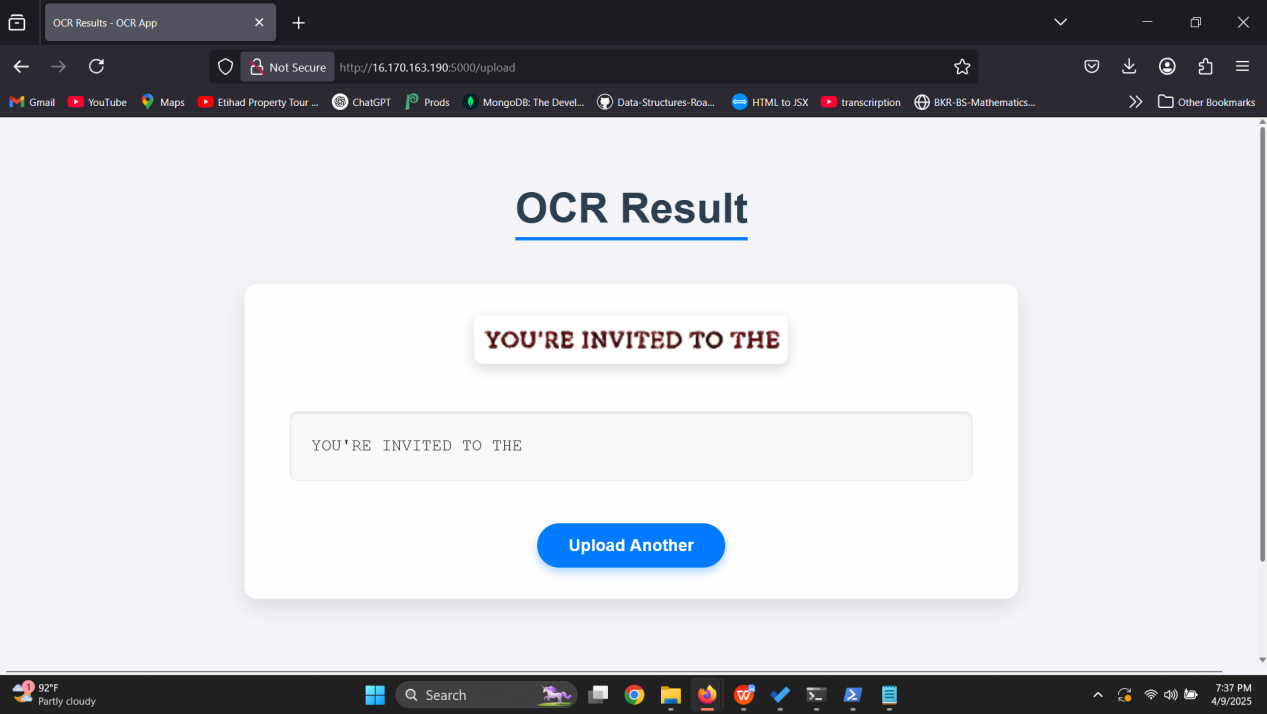
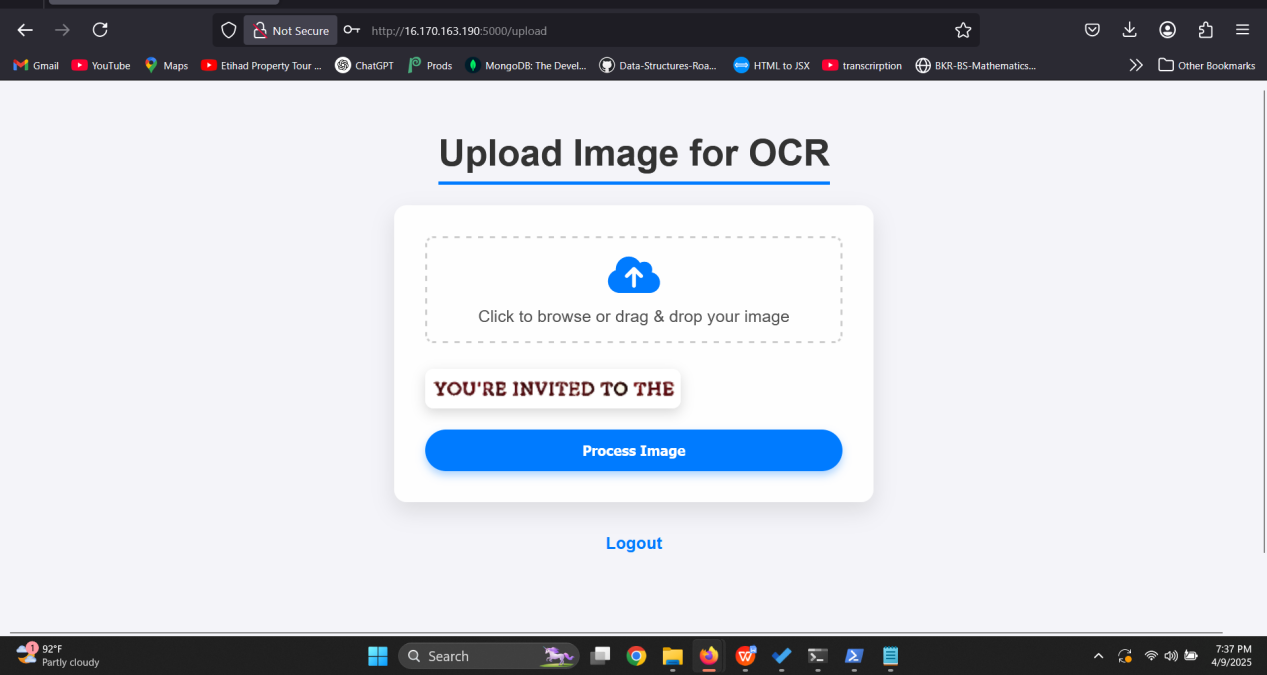
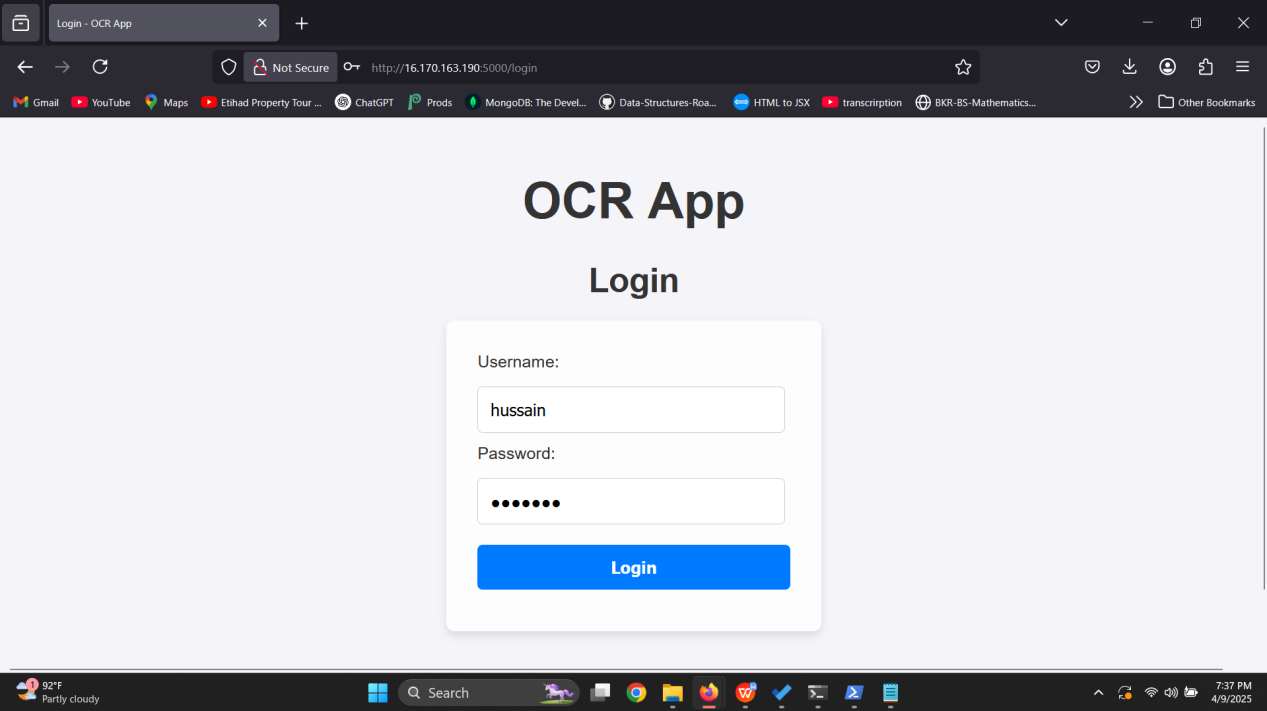
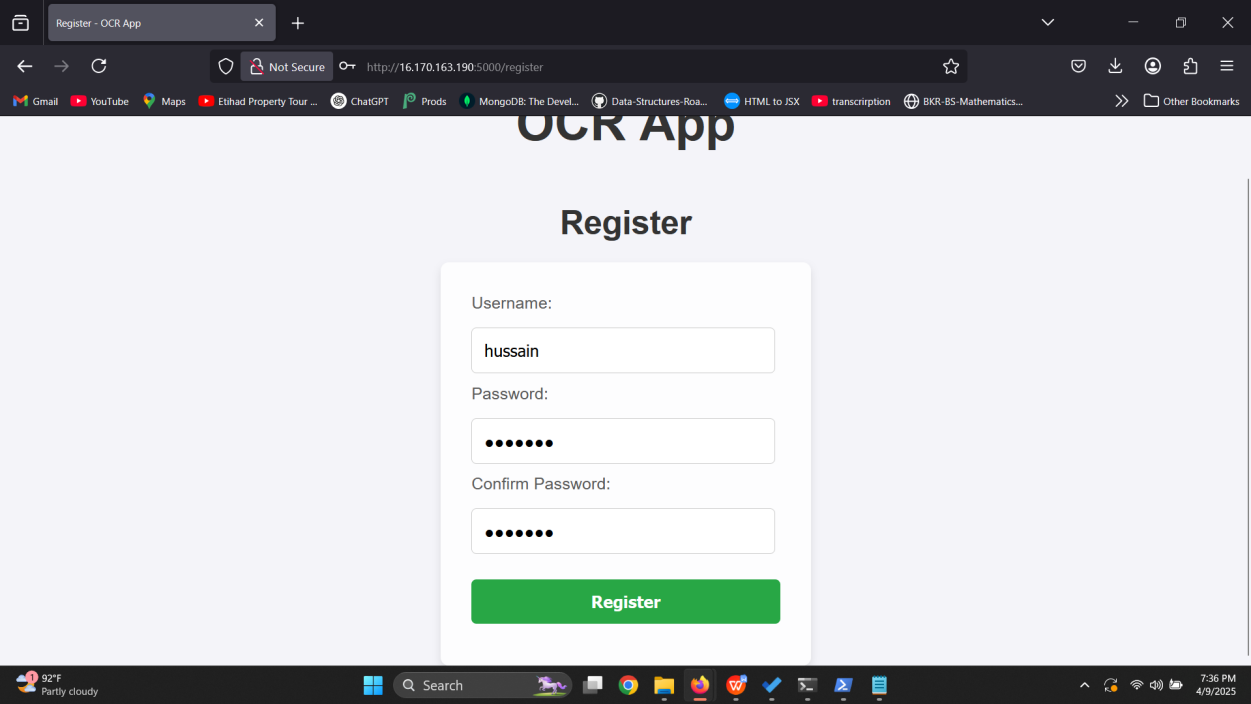
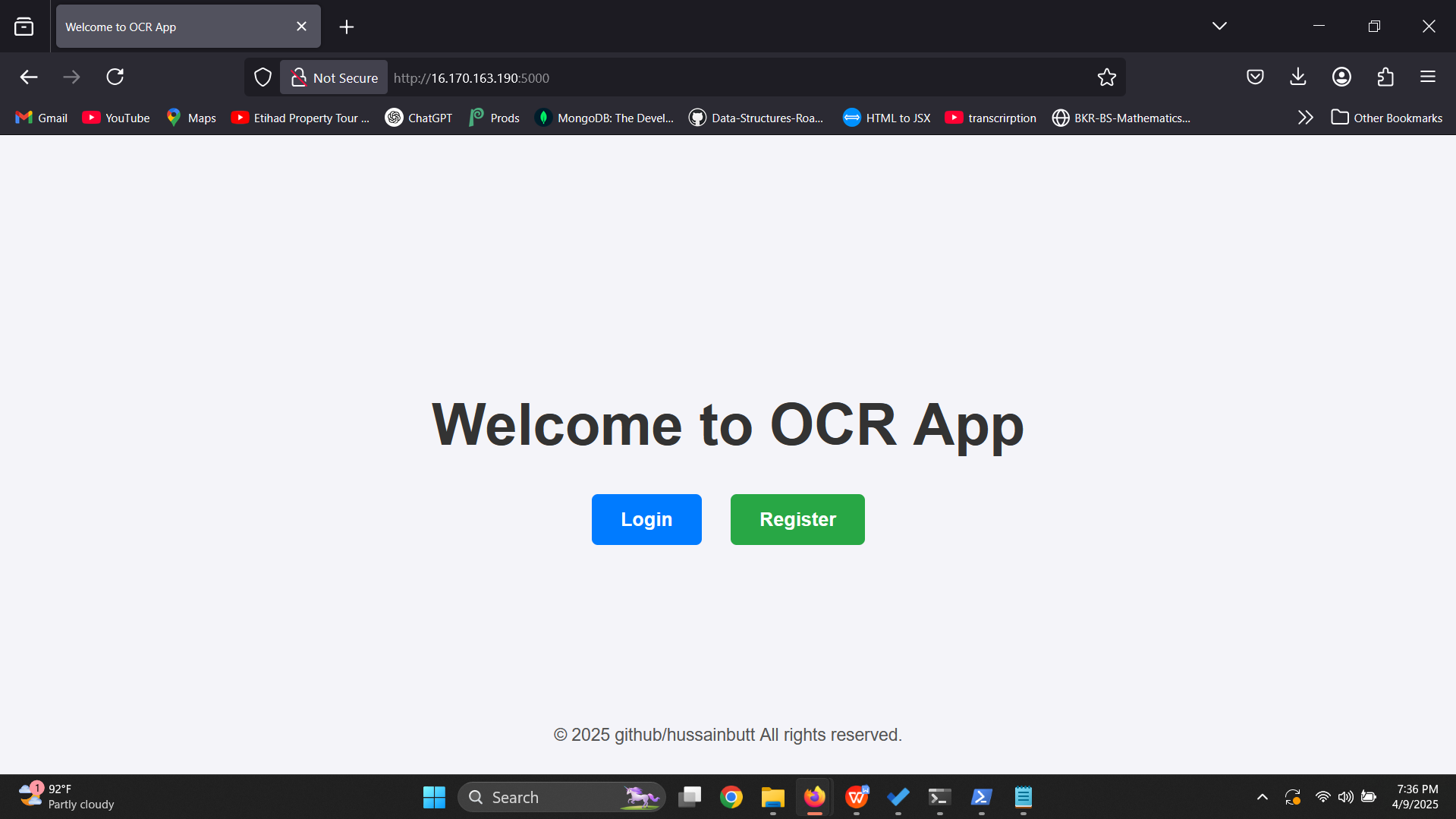
Once you're done working in the virtual environment, you can deactivate it by running:

*deactivate*

**App Screenshots**

**Code will be available at**

**<https://github.com/hussainbutt/ocr-app>**

****