Project:

CI/CD Pipeline for .NET Application

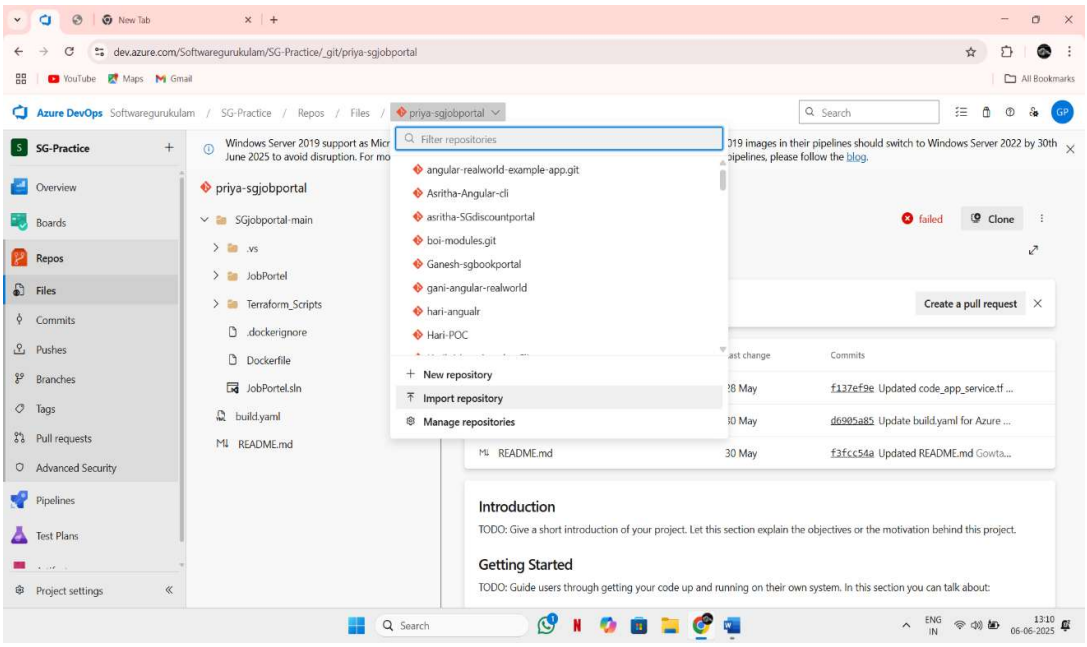
✅ Pre-Requisites

- Azure DevOps Project
- Windows VM (Azure or on-prem)
- IIS Installed & configured
- .NET Framework or .NET Core Runtime installed on VM
- Azure DevOps Agent installed on the VM using Deployment Group
- App hosted in Azure Repos

✅ Step-by-Step Procedure

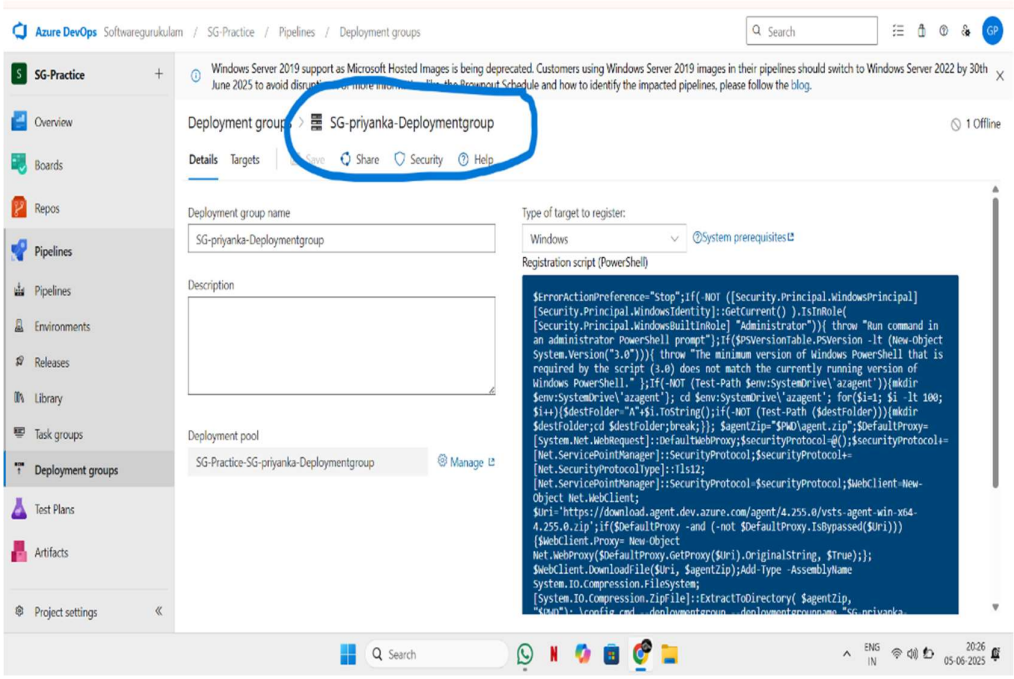
🔹 Step 1: Import a folder containing .NET files into Azure repos

- Import a folder that contains all files related to .NET app into the Azure repos through option Import Repository.



Step 2: Create a Deployment Group in Azure DevOps

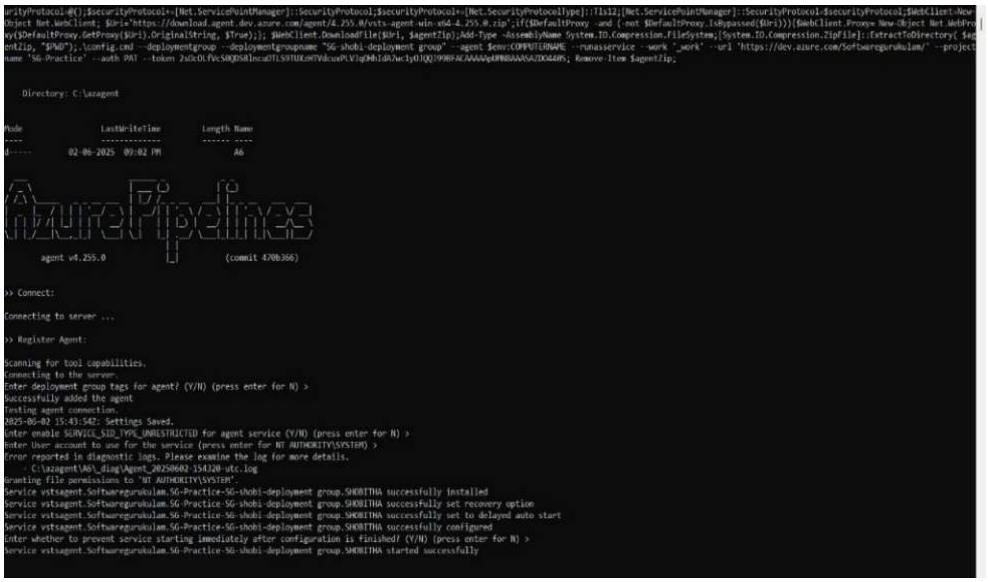
1. Go to your Azure DevOps Project → **Pipelines > Deployment Groups**
2. Click **Add a deployment group**
    - Name: SG-priyanka-Deploymentgroup
3. Copy the **PowerShell script** shown (Agent installation script)



Step 3: Set Up VM and Register the Agent

On your **Windows VM**:

1. Open **PowerShell as Administrator**
2. Paste and run the **PowerShell script** from the deployment group page
3. Ensure agent is connected and shown as **online** in Azure DevOps

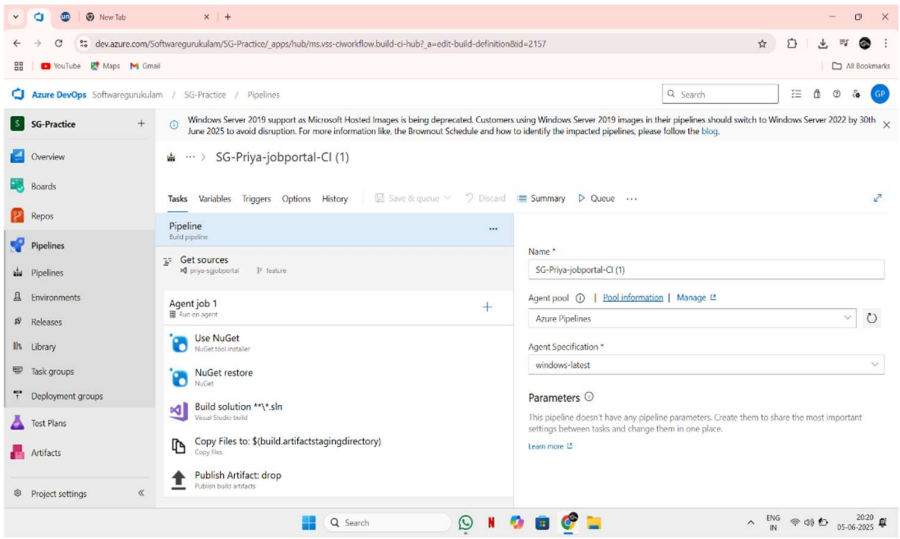


◆ Step 4: Set Up CI Pipeline (Build)

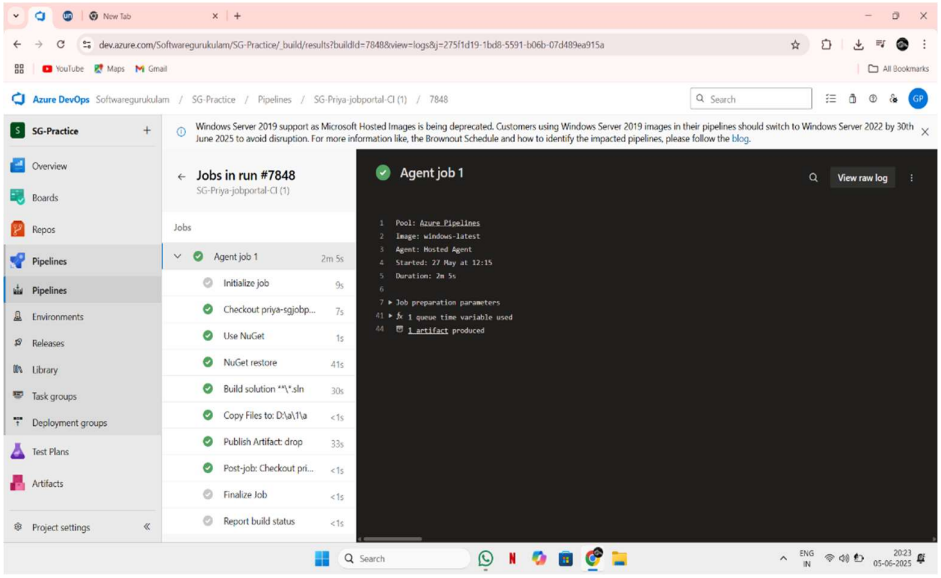
1. Go to **Pipelines > Builds**
2. Click **New Pipeline** → Use **Classic Editor**
3. Choose your repo, then select **Empty Job**
4. In build pipeline select agent specification as windows-latest
5. Change the pipeline name to SG-Priya-jobportal-CI (1)
6. In agent job change agent pool to Azure pipelines and agent specification as windows-latest
7. Add tasks to agent job in order:
    - **NuGet tool installer**
    - **NuGet:** give NuGet restore in command
    - **Visual Studio build:**
        - platform: any cpu
        - configuration: release
    - **Copy files:**
        - source folder: $(agent.builddirectory)

- content: ** (or any specified type required like **/*.tf for terraform files)
- target folder: $(build.artifactstagingdirectory)

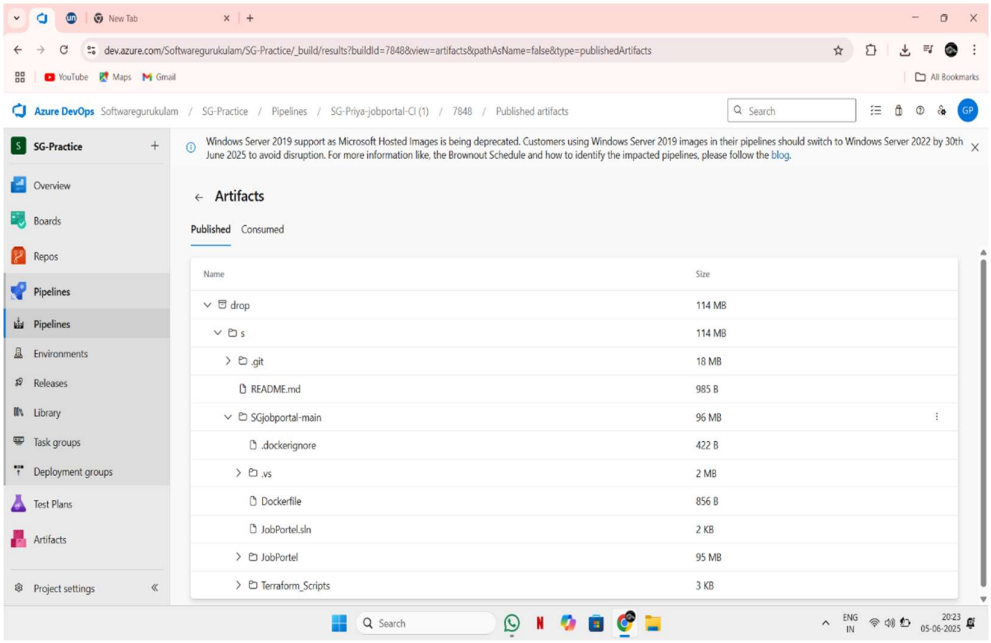
- **Publish Build Artifacts**:
  - Path to publish: $(Build.ArtifactStagingDirectory)
  - Artifact name: drop



8. Save and Queue build



9. Published artifacts:

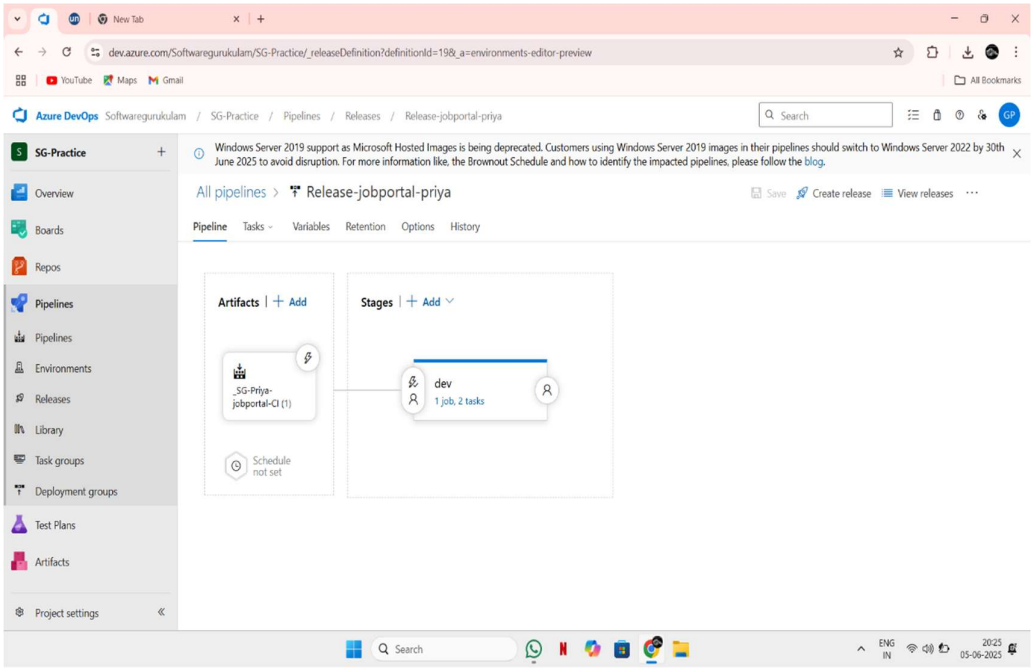


◆ Step 5: Set Up CD Pipeline (Release) with Deployment Group

1. Go to **Pipelines > Releases**
2. Create a **New Release Pipeline** → Start with **Empty Job**
3. Add an **Artifact** from the build pipeline
4. Enable **CD Trigger** (⚡ icon)

In the Stage (dev):

5. Select the template as IIS website deployment
6. the stage name as dev
7. Inside the stage dev it will take the default web site of VM. If you want you can create a custom web site and add bindings in the stage.
8. Under **Agent Job**, change:
    - **Deployment group** → SG-priyanka-Deploymentgroup



◆ Step 6: Add Tasks to Deploy to IIS

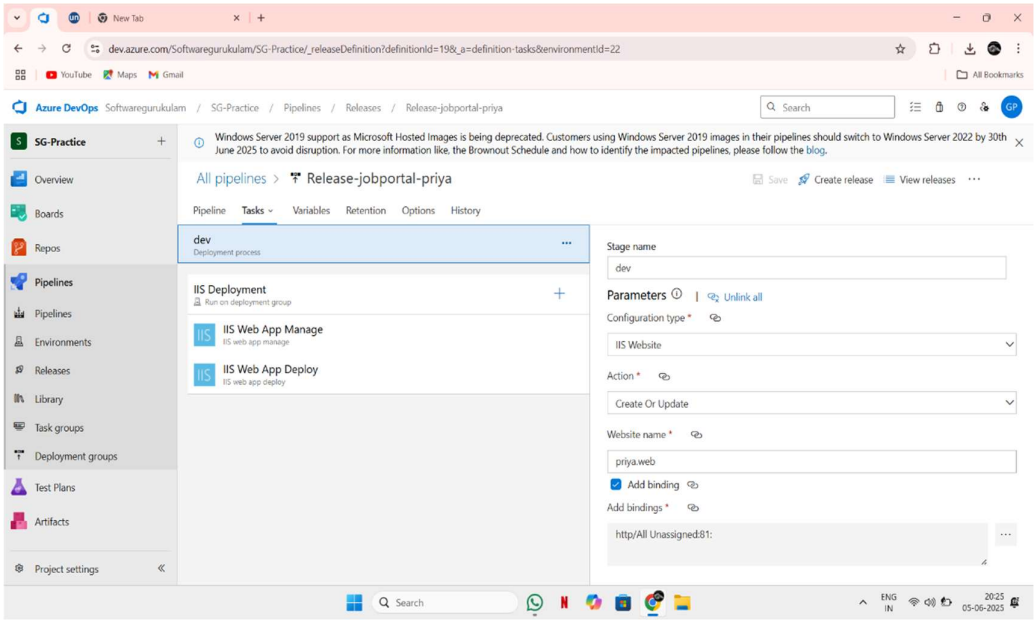
Inside the **IIS agent Job**, add these tasks:

1. IIS Web App Management

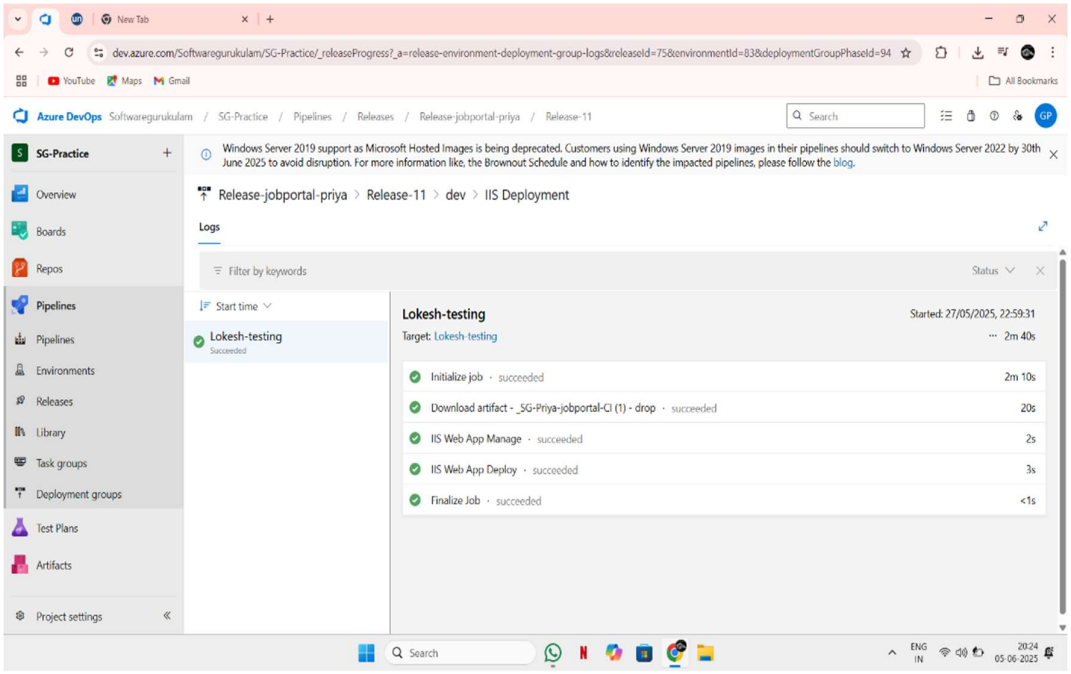
- Task: **IIS Web App Manage**
- **Web site Name:** Priya.web

2. IIS Web App Deployment

- Task: **IIS Web App Deploy**
- **Web Site Name**: priya.web (Default Web Site or custom site)
- **Package or Folder**: $(System.DefaultWorkingDirectory)/_SG-Priya-jobportal-CI (1)/drop/WebPublish
- Ensure IIS site is pre-created or handled via PowerShell



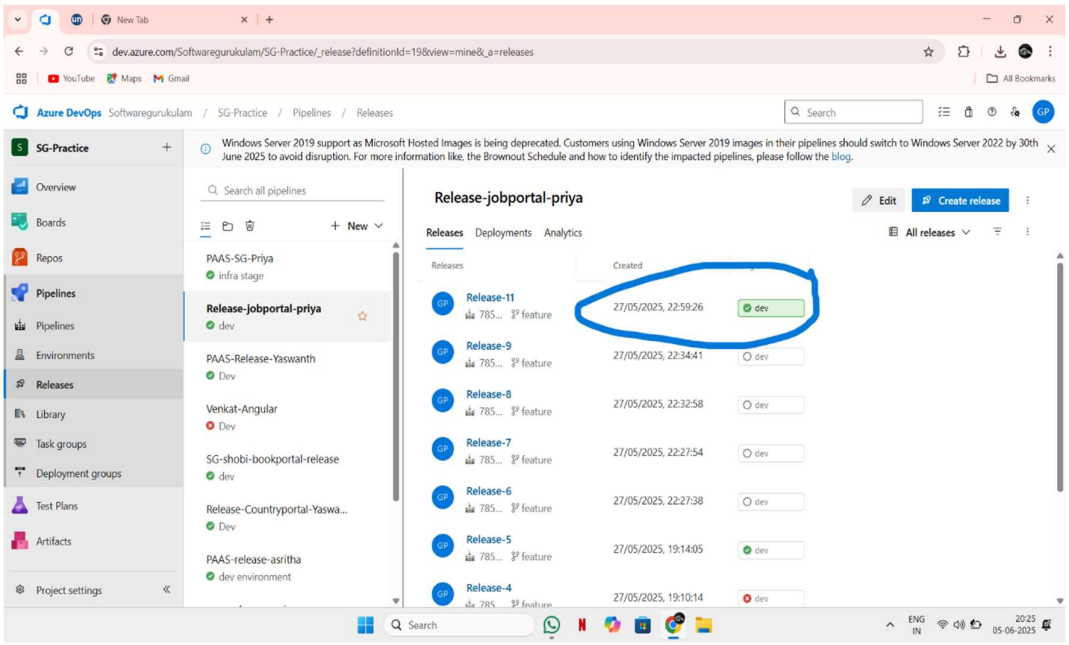
3. Save and Create Release



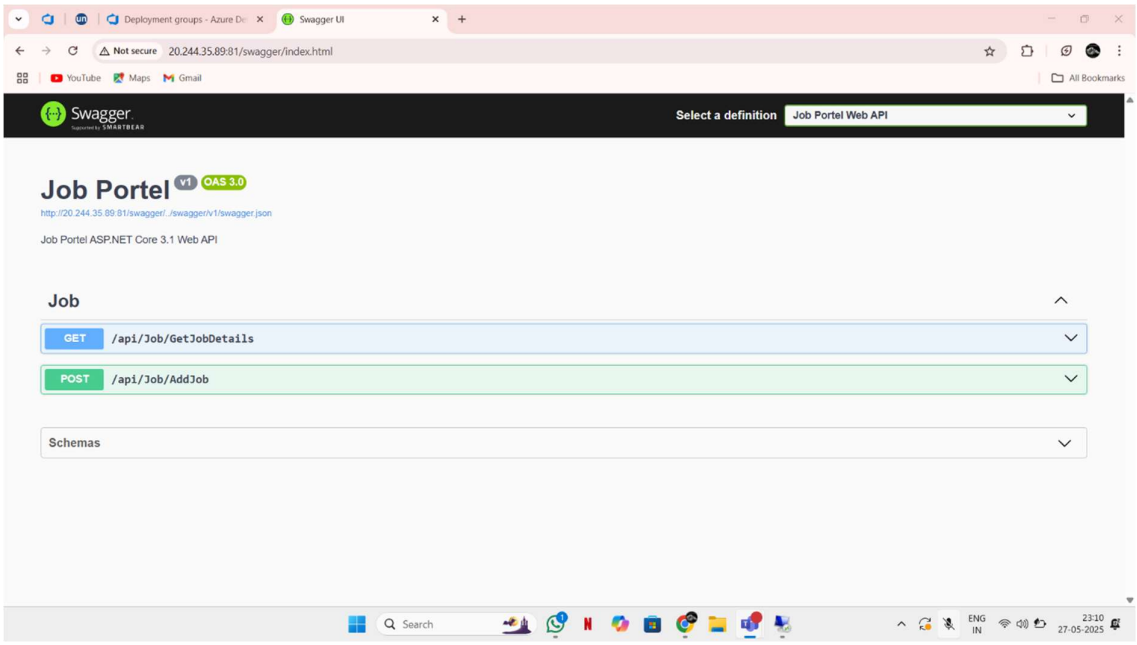
◆ Step 7: Trigger and Verify Deployment

1. Commit changes to your repo → triggers CI

2. Release pipeline picks artifact → deploys to IIS



3.Open your VM's public IP in browser and verify app is running.



✓ **The application is successfully deployed.**