Link to website : https://dvukman-portfolio.onrender.com

To deploy your Flask project on Render using a GitHub repository, follow these steps:

Step 1: Prepare Your Project for Deployment

1. Ensure a requirements.txt File Exists:

   This file is crucial because Render uses it to install your project dependencies.
   If you don't have a requirements.txt, create one by running the following command inside your virtual environment:

   ```
   pip freeze > requirements.txt
   ```

2. Procfile:

   Create a Procfile at the root of your project. This tells Render how to run your Flask app.
   Add the following line to the Procfile:

   ```
   web: gunicorn app:app
   ```

   This assumes that your main Flask application is in app.py and your Flask app instance is called app.

3. Configure Your Flask App for Deployment:

   Modify app.py to check if it's running on Render or a local machine:

   ```
   if __name__ == "__main__":
       app.run(host='0.0.0.0', port=int(os.environ.get("PORT", 5000)))
   ```

Step 2: Push Your Project to GitHub

1. Create a GitHub Repository:

   If you don't already have a GitHub repository for your project, create one by visiting GitHub and clicking the "New" button.

2. Push Your Code:

Initialize a git repository in your project directory (if you haven't already):
git init

3. Add your files:
   git add .

4. Commit your files:
   git commit -m "Initial commit"

5. Push the code to your GitHub repository:
   git remote add origin https://github.com/YOUR_USERNAME/YOUR_REPO_NAME.git
   git push -u origin master

Step 3: Deploy the Project on Render

1. Create a Render Account:

   Go to Render and create an account if you don't have one.

2. Create a New Web Service:

   After logging in, click "New" at the top and select "Web Service".
   Connect your Render account to your GitHub account.
   Select the GitHub repository that contains your Flask project.

3. Configure the Deployment Settings:

- Name: Choose a name for your service (this will be part of your URL).
- Environment: Select Python.
- Build Command: Leave it empty or set it to pip install -r requirements.txt.
- Start Command: Set this to:

  gunicorn app:app

4. Deploy the Web Service:

   Click Create Web Service.
   Render will now start building your project and install dependencies from requirements.txt.

5. Wait for the Deployment:

   Once the deployment is complete, Render will provide you with a URL where you can access your Flask app.

Step 4: Test and Monitor

1. Test the Application:

   After deployment, visit the URL provided by Render and test your Flask app to ensure it’s working as expected.

2. Monitor Logs:

   You can monitor the logs in the Render dashboard to see if there are any errors or warnings during deployment.