Deploying a Flask Machine Learning App on Render

This guide explains how to deploy a Flask-based machine learning application on Render. Render allows you to host web services for free or at low cost, making it a good choice for student projects.

# Step 1. Prepare Your Project for Deployment

Before deploying, make sure your project folder has the following files:  
 - app.py → Flask application entry point  
 - model/ → trained model files (e.g., income\_model.pkl)  
 - preprocessing.py → preprocessing functions  
 - requirements.txt → all dependencies (Flask, pandas, joblib, scikit-learn, etc.)  
 - Procfile → tells Render how to start your app  
 - test\_payload.json → sample payload for testing  
  
Example Procfile content:  
web: gunicorn app:app  
  
This tells Render to run the Flask app using Gunicorn, a production-ready WSGI server.

# Step 2. Push Your Project to GitHub

Render requires your code to be in a Git repository. To push your project to GitHub:  
1. Create a new repository on https://github.com  
2. On your computer, open the terminal in your project folder.  
3. Run the following commands:  
 git init  
 git add .  
 git commit -m "Initial commit"  
 git branch -M main  
 git remote add origin https://github.com/your-username/your-repo-name.git  
 git push -u origin main  
  
Your code is now stored on GitHub and accessible to Render.

# Step 3. Create a Render Account

Go to https://render.com and sign up for a free account. You can use your GitHub account to sign in.

# Step 4. Create a New Web Service on Render

1. From the Render dashboard, click \*\*New + → Web Service\*\*.  
2. Connect your GitHub account if prompted.  
3. Choose your repository (the one containing your Flask app).  
4. Fill in the service details:  
 - Name: any descriptive name  
 - Region: pick the closest region  
 - Branch: main  
 - Build Command: pip install -r requirements.txt  
 - Start Command: gunicorn app:app  
5. Click \*\*Create Web Service\*\*.

# Step 5. Wait for Build and Deployment

Render will:  
 - Install Python and your dependencies from requirements.txt  
 - Use the Procfile and start command to run your Flask app  
 - Allocate a free web service URL for your project  
  
Example: https://your-app.onrender.com

# Step 6. Test Your Deployment

Once deployment is successful, open the Render URL in your browser.  
  
 - Visit / → should show the home page  
 - Visit /health → should show {"status": "ok"}  
 - Test /predict using curl or Postman:  
  
curl -X POST https://your-app.onrender.com/predict \  
 -H "Content-Type: application/json" \  
 --data @test\_payload.json

# Step 7. Common Deployment Issues

• Missing packages → double-check requirements.txt  
• Wrong start command → ensure Procfile says 'web: gunicorn app:app'  
• Model not found → make sure model/income\_model.pkl is pushed to GitHub  
• App crashes → check Render logs (click Logs in dashboard)

# Conclusion

With these steps, students can successfully deploy a Flask machine learning app on Render. They can share the public URL with others for testing and demonstration.