9/8/2024

FastAPI Machine Learning application for Heart Disease Detection

Mohamad Thawfeek Mohammadhu Safran

ITBIN-2110-0095

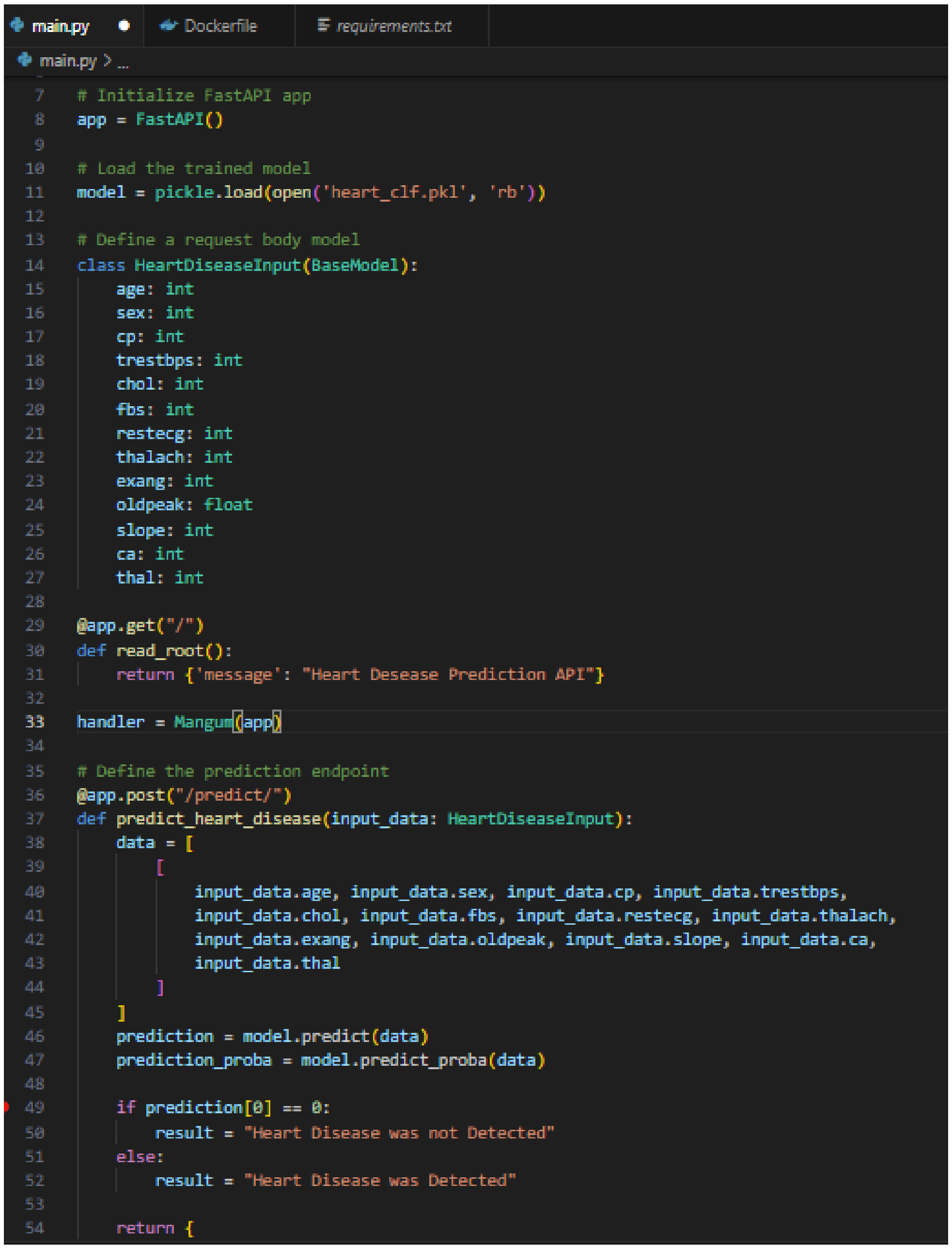
Horizon Campus

To deploy a FastAPI Machine Learning application for Heart Disease Detection on Google Cloud Run, follow these steps

1. Prepare Your FastAPI Application

Before deployment, ensure that your FastAPI app is functional and ready to serve the heart disease detection model. The app should:

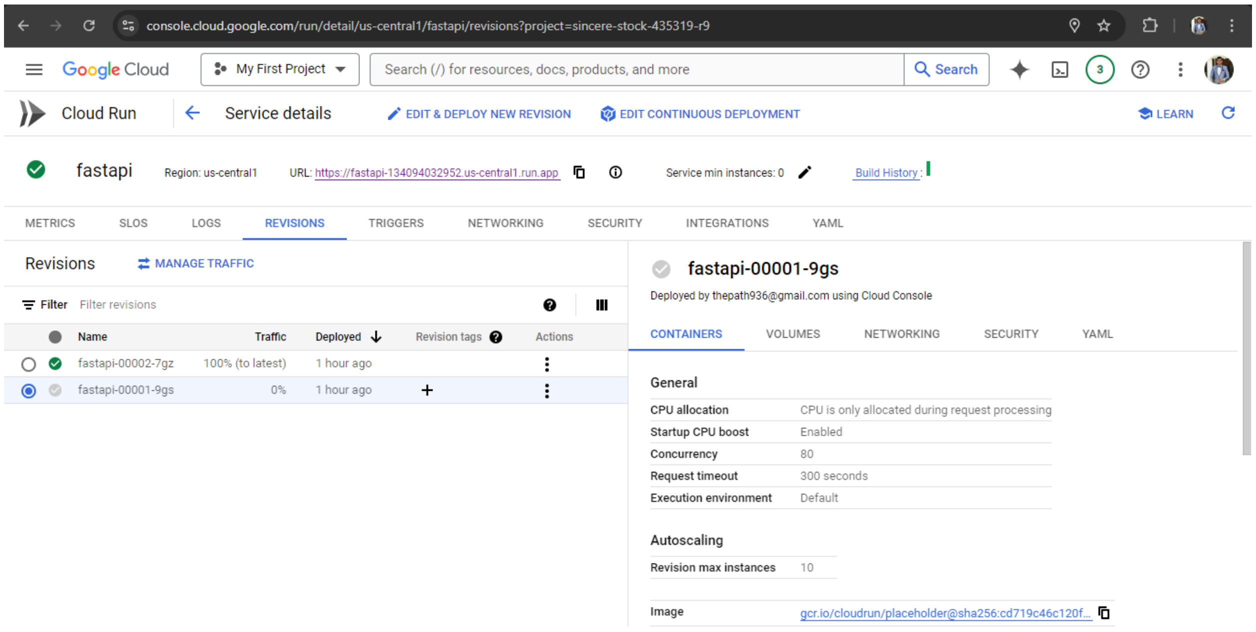
- Accept input data (e.g., patient attributes such as age, blood pressure, cholesterol).
- Use a pre-trained machine learning model to make predictions (e.g., classify if a patient has heart disease).
- Return the prediction in JSON format.



```python
# Initialize FastAPI app
app = FastAPI()

# Load the trained model
model = pickle.load(open('heart_clf.pkl', 'rb'))

# Define a request body model
class HeartDiseaseInput(BaseModel):
    age: int
    sex: int
    cp: int
    trestbps: int
    chol: int
    fbs: int
    restecg: int
    thalach: int
    exang: int
    oldpeak: float
    slope: int
    ca: int
    thal: int

@app.get("/")
def read_root():
    return {'message': "Heart Desease Prediction API"}

handler = Mangum(app)

# Define the prediction endpoint
@app.post("/predict/")
def predict_heart_disease(input_data: HeartDiseaseInput):
    data = [
        [
            input_data.age, input_data.sex, input_data.cp, input_data.trestbps,
            input_data.chol, input_data.fbs, input_data.restecg, input_data.thalach,
            input_data.exang, input_data.oldpeak, input_data.slope, input_data.ca,
            input_data.thal
        ]
    ]
    prediction = model.predict(data)
    prediction_proba = model.predict_proba(data)

    if prediction[0] == 0:
        result = "Heart Disease was not Detected"
    else:
        result = "Heart Disease was Detected"

    return {
```

2. Set Up Google Cloud

- Create a Google Cloud Project
- Go to the Google Cloud Console.
- Create a new project or select an existing on



console.cloud.google.com/run/detail/us-central1/fastapi/revisions?project=sincere-stock-435319-r9

fastapi Region: us-central1 URL: https://fastapi-134094032952.us-central1.run.app Service min instances: 0 Build History :

Revisions

Name	Traffic	Deployed	Revision tags	Actions
fastapi-00002-7gz	100% (to latest)	1 hour ago		
fastapi-00001-9gs	0%	1 hour ago		

fastapi-00001-9gs

Deployed by thepath936@gmail.com using Cloud Console

General

CPU allocation	CPU is only allocated during request processing
Startup CPU boost	Enabled
Concurrency	80
Request timeout	300 seconds
Execution environment	Default

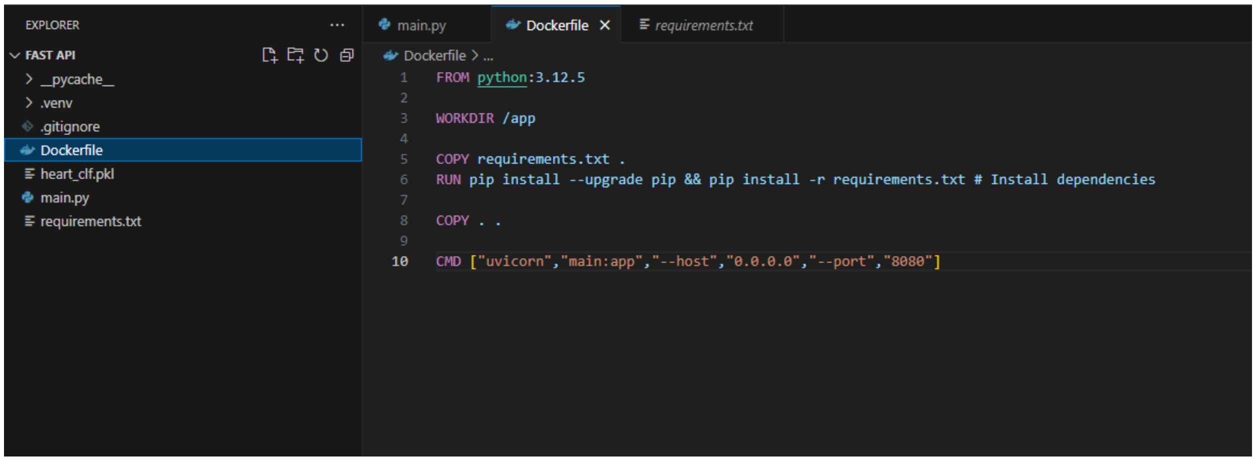
Autoscaling

Revision max instances	10

Image	gcr.io/cloudrun/placeholder@sha256:cd719c46c120f...

3. Dockerize the Application

Cloud Run requires Docker containers for deployment. Create a Dockerfile in project directory

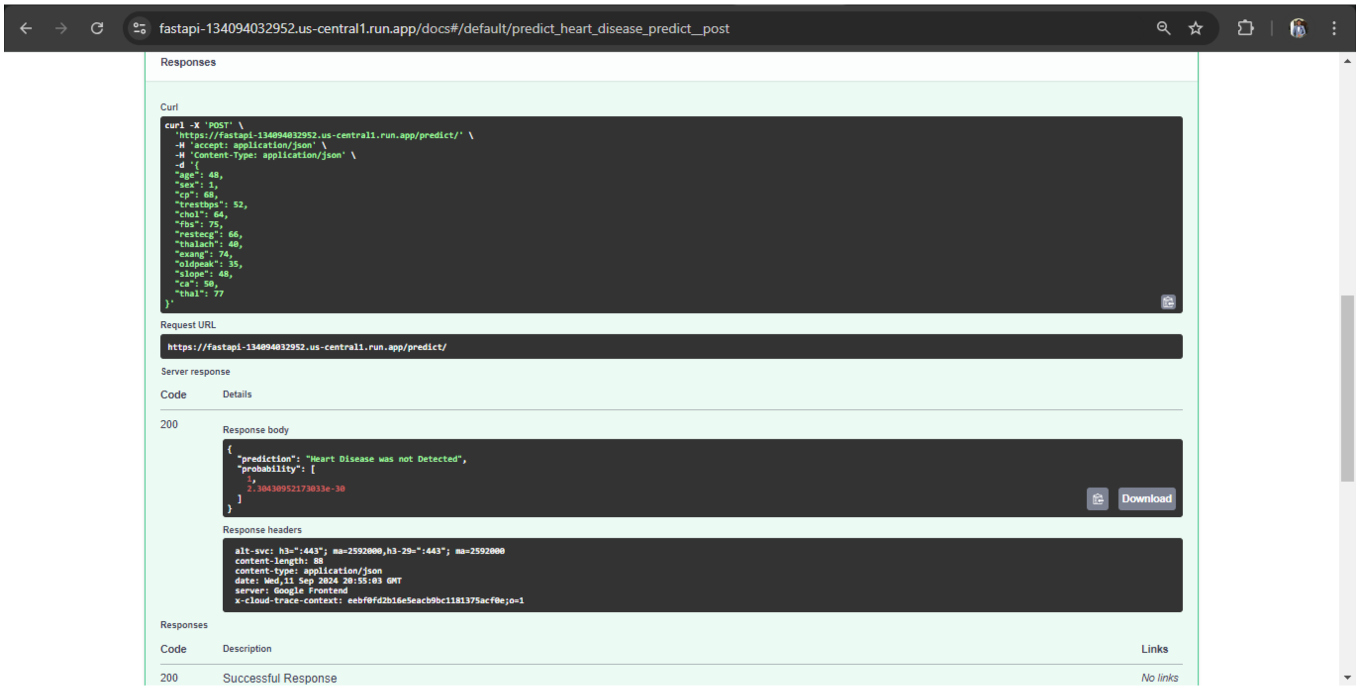


```
FROM python:3.12.5

WORKDIR /app

COPY requirements.txt .
RUN pip install --upgrade pip && pip install -r requirements.txt # Install dependencies

COPY . .

CMD ["uvicorn","main:app","--host","0.0.0.0","--port","8080"]
```

4. Deploy to Google Cloud Run

Push Docker Image to Google Container Registry

Output:



fastapi-134094032952.us-central1.run.app/docs#/default/predict_heart_disease_predict__post

Responses

Curl

```
curl -X 'POST' \
  'https://fastapi-134094032952.us-central1.run.app/predict/' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
  "age": 48,
  "sex": 1,
  "cp": 68,
  "trestbps": 52,
  "chol": 64,
  "fbs": 75,
  "restecg": 66,
  "thalach": 40,
  "exang": 74,
  "oldpeak": 35,
  "slope": 48,
  "ca": 50,
  "thal": 77
}'
```

Request URL

```
https://fastapi-134094032952.us-central1.run.app/predict/
```

Server response

Code	Details
200	Response body

```
{
  "prediction": "Heart Disease was not Detected",
  "probability": [
    1,
    2.30430952173033e-30
  ]
}
```

Response headers

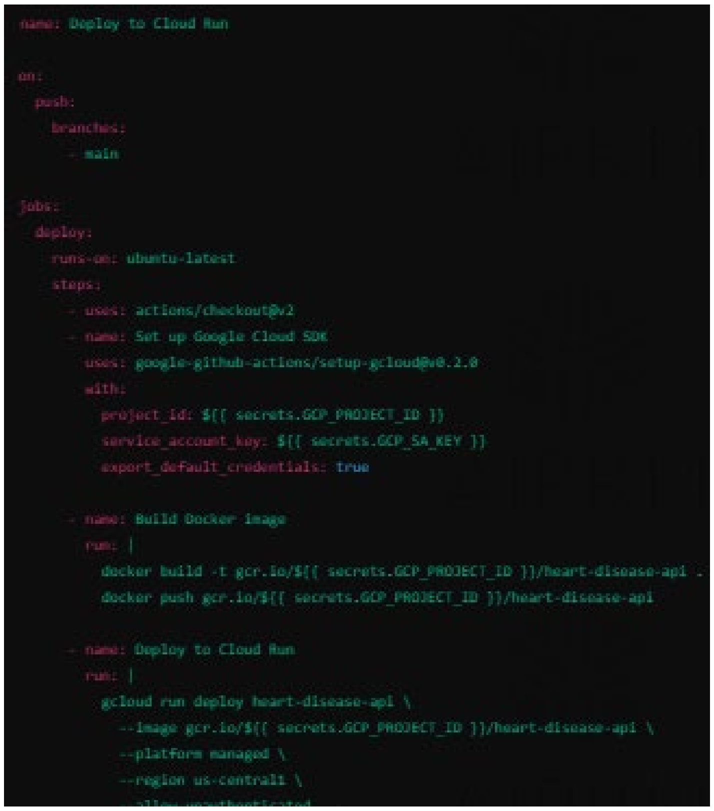
```
 alt-svc: h3=":443"; ma=2592000,h3-29=":443"; ma=2592000
 content-length: 88
 content-type: application/json
 date: Wed,11 Sep 2024 20:55:03 GMT
 server: Google Frontend
 x-cloud-trace-context: eebf0fd2b16e5eacb9bc1181375acf0e;o=1
```

Responses

Code	Description	Links
200	Successful Response	No links

5.Set Up CI/CD Pipeline (Optional)

- To automate the deployment, set up a CI/CD pipeline with GitHub Actions.
- Create a .github/workflows/deploy.yml file in your GitHub repository



```
name: Deploy to Cloud Run

on:
  push:
    branches:
      - main

jobs:
  deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v2
      - name: Set up Google Cloud SDK
        uses: google-github-actions/setup-gcloud@v0.2.0
        with:
          project_id: ${{ secrets.GCP_PROJECT_ID }}
          service_account_key: ${{ secrets.GCP_SA_KEY }}
          export_default_credentials: true

      - name: Build Docker image
        run: |
          docker build -t gcr.io/${{ secrets.GCP_PROJECT_ID }}/heart-disease-api .
          docker push gcr.io/${{ secrets.GCP_PROJECT_ID }}/heart-disease-api

      - name: Deploy to Cloud Run
        run: |
          gcloud run deploy heart-disease-api \
            --image gcr.io/${{ secrets.GCP_PROJECT_ID }}/heart-disease-api \
            --platform managed \
            --region us-central1 \
            --allow-unauthenticated
```

This is the URL of the deployed application: - https://fastapi-134094032952.us-central1.run.app

Published this on platforms Medium :- https://medium.com/@thepath936/how-to-deploy-a-fastapi-machine-learning-application-in-google-cloud-run-edd52c30b60e

This is my Github account: - https://github.com/SafranThawfeek/FastAPI.git