Heroku Cloud Deployment, Week 5 Project

Name: Blessed Adjei-Gyan

Batch Code: LISUM 39

Submission Date: 5th December , 2024

Submitted to: Data Glacier

Objective

The objective of this project is to train and deploy a machine learning model using a simple toy dataset. The model was developed, saved, and deployed as both an API and a web application on **Heroku** using free-tier resources. A detailed PDF document, including snapshots of each deployment step, was created to document the process. The project demonstrates the complete workflow from model training to cloud deployment, ensuring accessibility and usability through a web-based interface.

Key Components

1. **Machine Learning Model**:
    - A pre-trained Logistic Regression model (titanic_model.pkl) was used to make survival predictions.
    - The model was trained using features such as:
        - Passenger Class (Pclass)
        - Gender (Sex)
        - Age
        - Number of Siblings/Spouse aboard (SibSp)
        - Number of Parents/Children aboard (Parch)
        - Fare paid (Fare).
2. **Flask API**:
    - A Flask application was built to host the predictive model.
    - The /predict endpoint accepts POST requests with passenger data in JSON format and returns predictions in JSON form.

- The home route (/) serves as a connection point for static files and basic API confirmation.

3. **Web Interface**:
    - An interactive HTML front-end allows users to input passenger data through a form and receive predictions in real time.
    - The background of the page features an image of the Titanic to enhance user experience.
    - JavaScript handles form submission and interacts with the Flask API for predictions.
4. **File Organization**:
    - The Titanic image file is stored in a static folder and linked in the HTML file to ensure proper rendering.
    - Flask was configured to serve static files and correctly route requests to the titanic.html front-end.

Steps Taken

1. **Data Preparation and Model Training**:
    - A Logistic Regression model was trained using the Titanic dataset, exported as titanic_model.pkl.
2. **Backend Development**:
    - A Flask app was created to load the model, define prediction logic, and serve the front-end.
3. **Front-End Design**:
    - A responsive HTML file was developed to capture user inputs and display predictions.
4. **Integration and Deployment**:
    - The Flask app was tested locally, with HTML and static assets linked properly.
    - End-to-end functionality was confirmed with accurate predictions based on sample inputs.

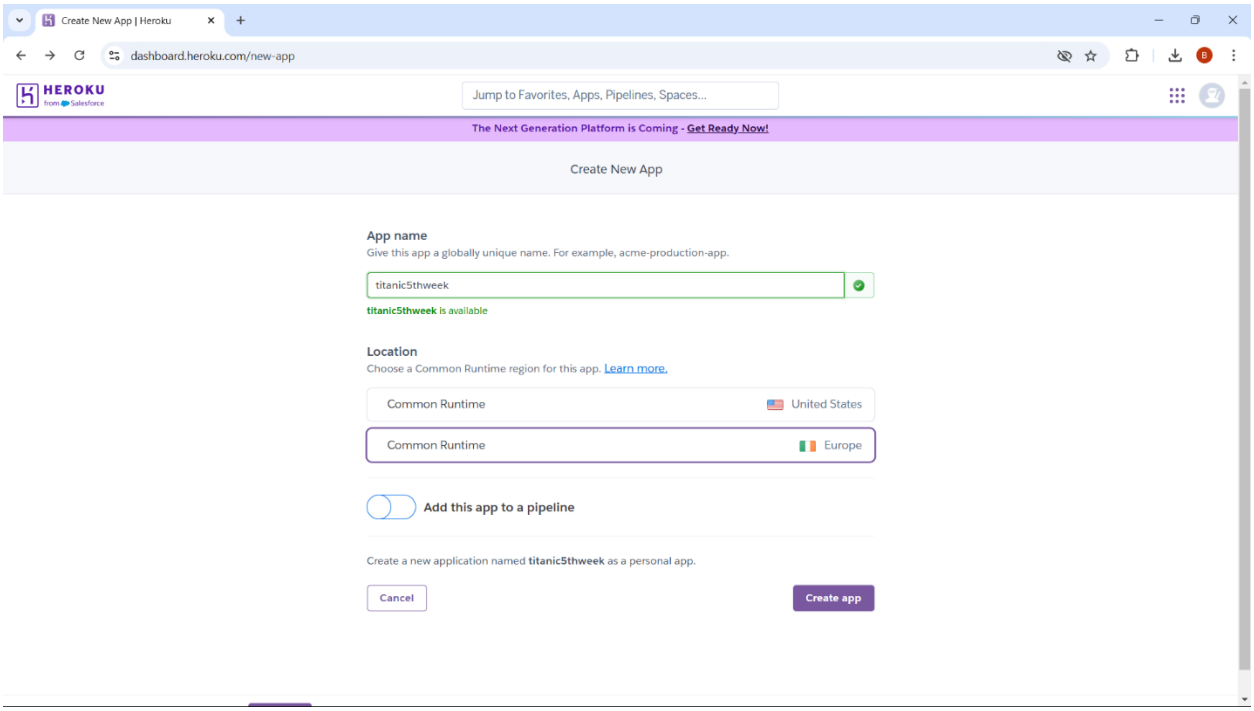
Sample Input and Output

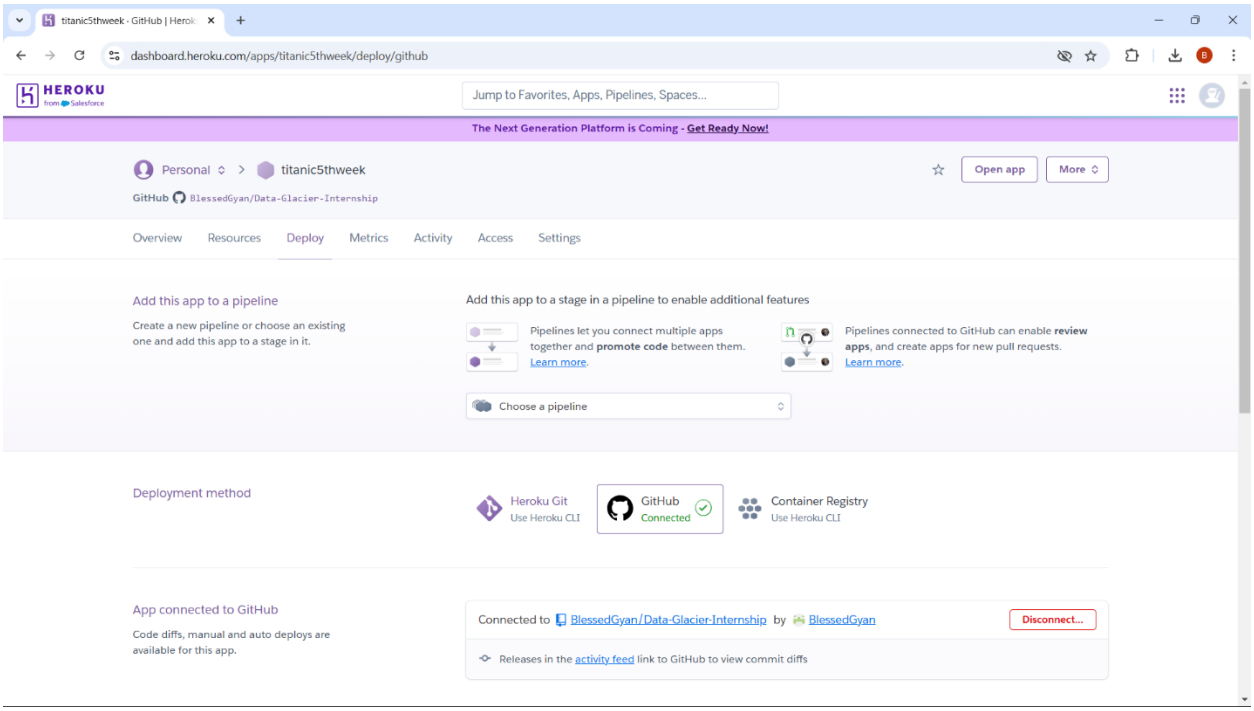
Input:

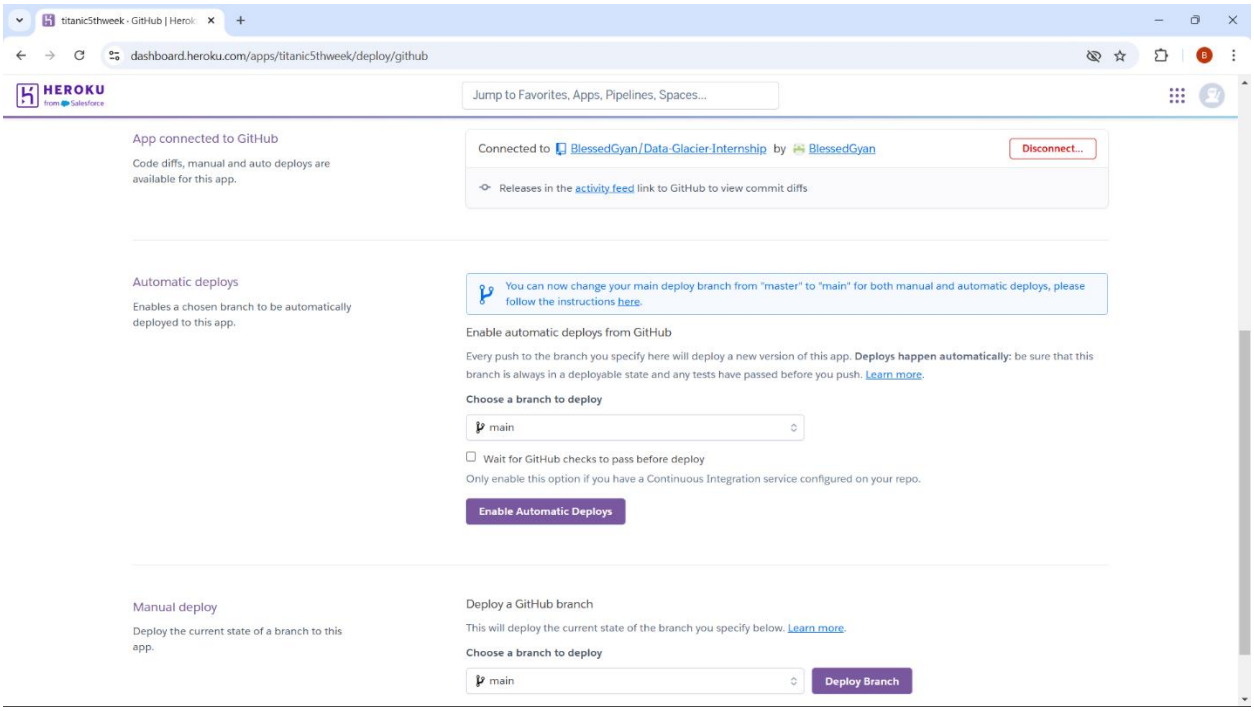
- Pclass: 1
- Sex: Female (0)
- Age: 25
- SibSp: 0
- Parch: 1
- Fare: 50.0

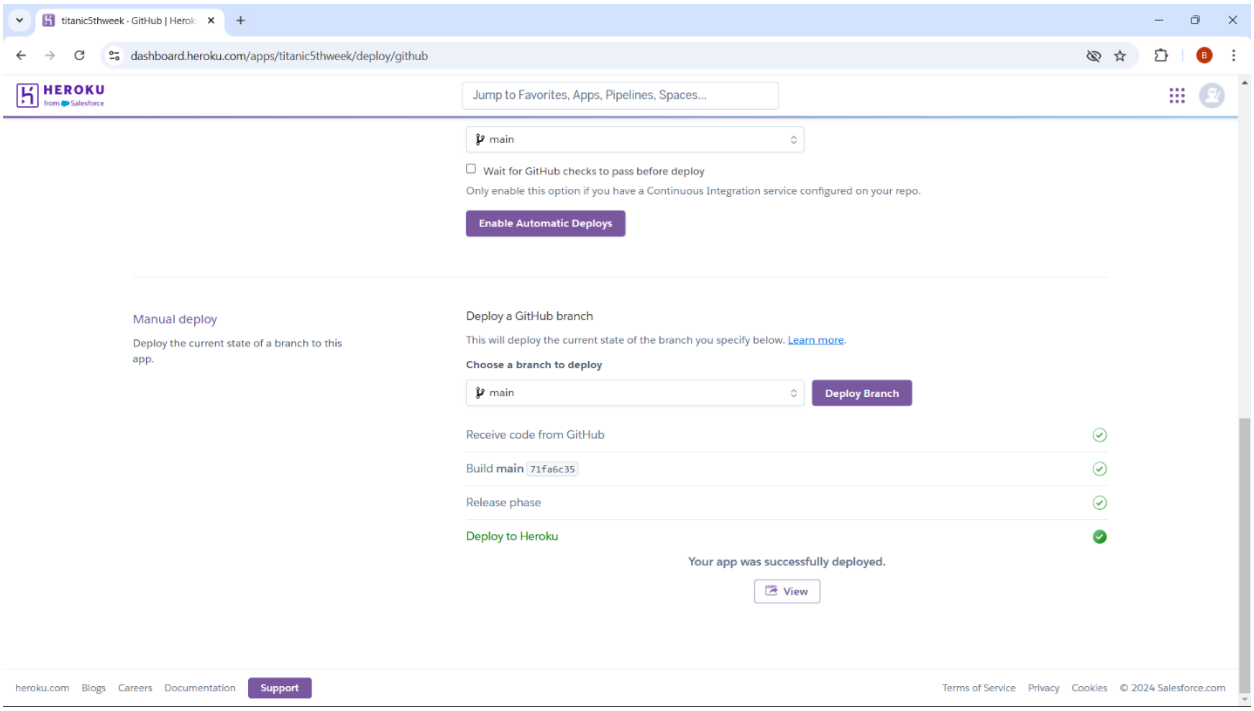
Output:

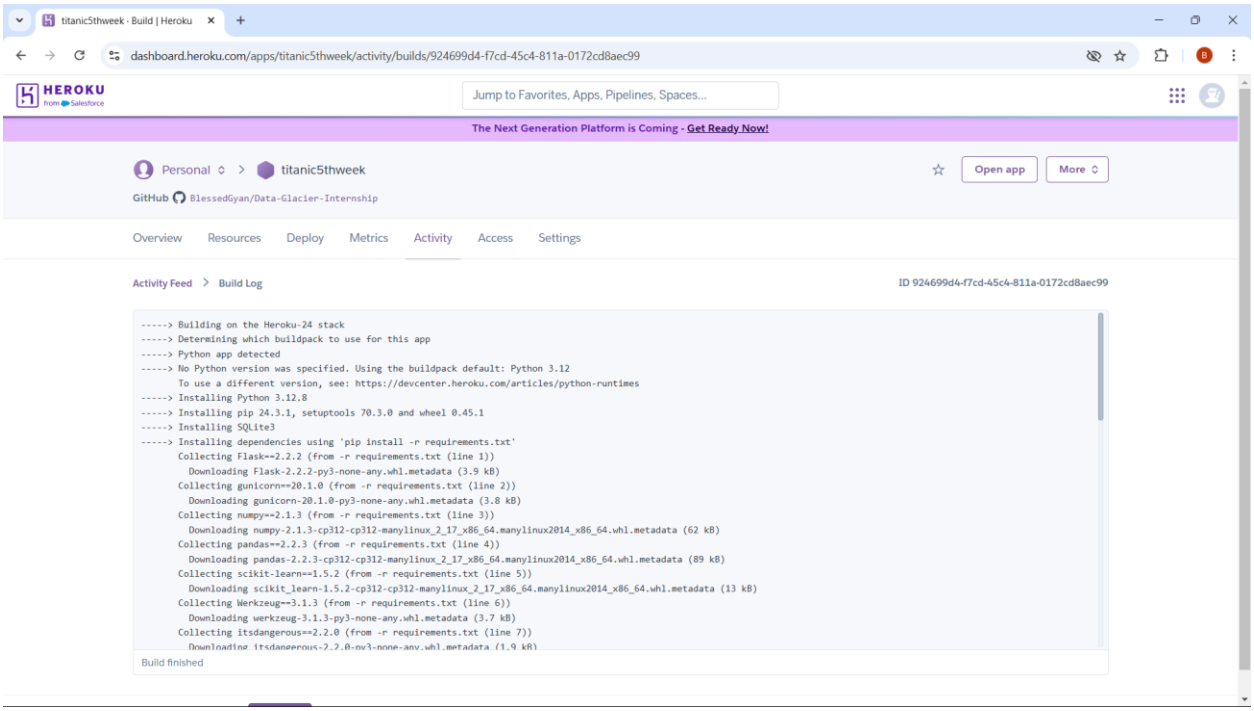
- JSON Response: { "Survived": 1 }
- Display: "Survived: Yes"

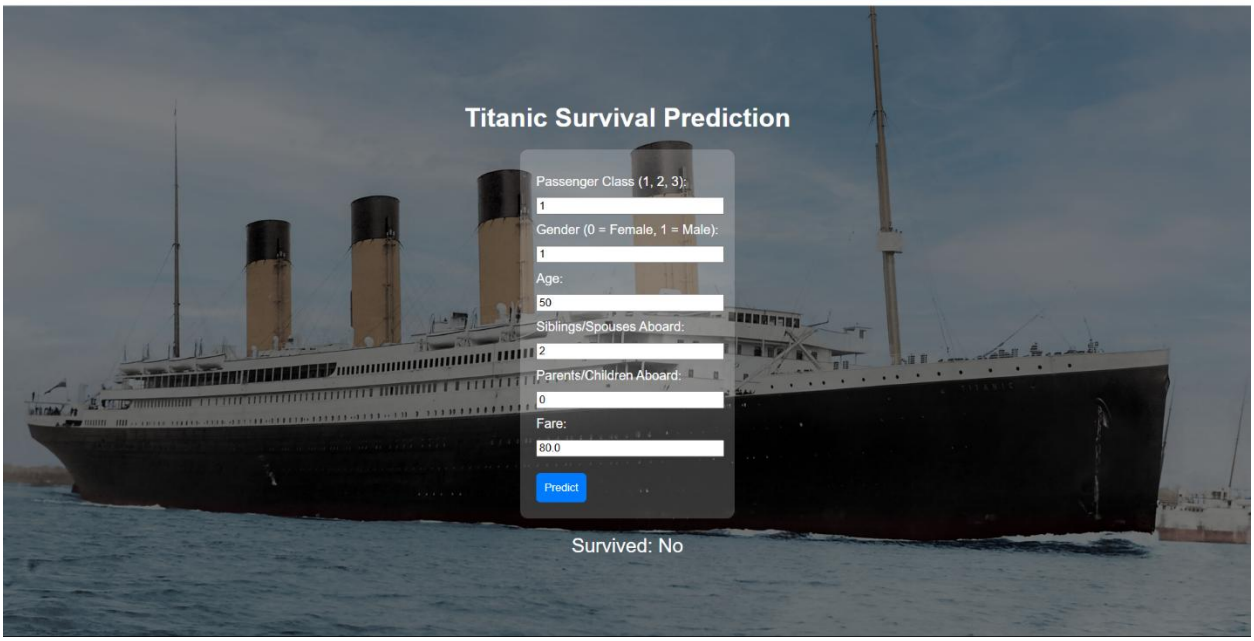


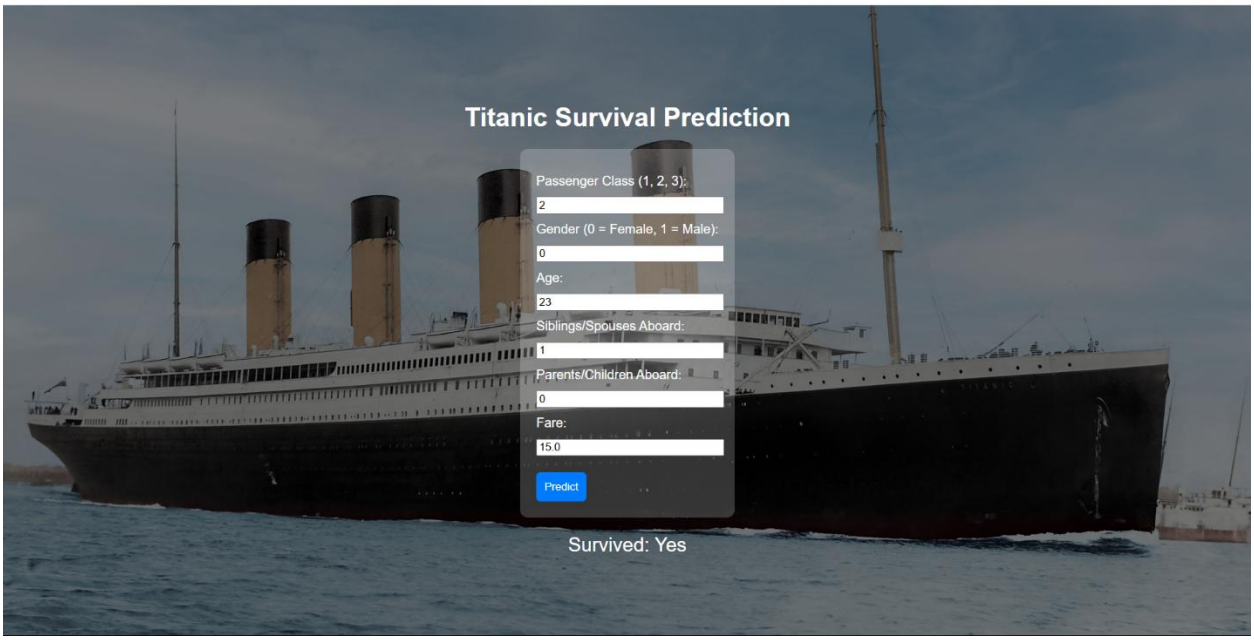

titanic5thweek · GitHub | Heroku
dashboard.heroku.com/apps/titanic5thweek/deploy/github
HEROKU
Jump to Favorites, Apps, Pipelines, Spaces...
The Next Generation Platform is Coming - Get Ready Now!
Personal > titanic5thweek
GitHub BlessedGyan/Data-Glacier-Internship
Open app
More
Overview
Resources
Deploy
Metrics
Activity
Access
Settings
Add this app to a pipeline
Create a new pipeline or choose an existing one and add this app to a stage in it.
Add this app to a stage in a pipeline to enable additional features
Pipelines let you connect multiple apps together and promote code between them. Learn more.
Pipelines connected to GitHub can enable review apps, and create apps for new pull requests. Learn more.
Choose a pipeline
Deployment method
Heroku Git
Use Heroku CLI
GitHub
Connected
Container Registry
Use Heroku CLI
App connected to GitHub
Code diffs, manual and auto deploys are available for this app.
Connected to BlessedGyan/Data-Glacier-Internship by BlessedGyan
Disconnect...
Releases in the activity feed link to GitHub to view commit diffs


titanic5thweek · GitHub | Heroku
dashboard.heroku.com/apps/titanic5thweek/deploy/github
HEROKU
Jump to Favorites, Apps, Pipelines, Spaces...
App connected to GitHub
Code diffs, manual and auto deploys are available for this app.
Connected to BlessedGyan/Data-Glacier-Internship by BlessedGyan
Disconnect...
Releases in the activity feed link to GitHub to view commit diffs
Automatic deploys
Enables a chosen branch to be automatically deployed to this app.
You can now change your main deploy branch from "master" to "main" for both manual and automatic deploys, please follow the instructions here.
Enable automatic deploys from GitHub
Every push to the branch you specify here will deploy a new version of this app. Deploys happen automatically: be sure that this branch is always in a deployable state and any tests have passed before you push. Learn more.
Choose a branch to deploy
main
Wait for GitHub checks to pass before deploy
Only enable this option if you have a Continuous Integration service configured on your repo.
Enable Automatic Deploys
Manual deploy
Deploy the current state of a branch to this app.
Deploy a GitHub branch
This will deploy the current state of the branch you specify below. Learn more.
Choose a branch to deploy
main
Deploy Branch


titanic5thweek · GitHub | Herok
dashboard.heroku.com/apps/titanic5thweek/deploy/github
HEROKU
from Salesforce
Jump to Favorites, Apps, Pipelines, Spaces...
main
Wait for GitHub checks to pass before deploy
Only enable this option if you have a Continuous Integration service configured on your repo.
Enable Automatic Deploys
Manual deploy
Deploy the current state of a branch to this app.
Deploy a GitHub branch
This will deploy the current state of the branch you specify below. Learn more.
Choose a branch to deploy
main
Deploy Branch
Receive code from GitHub
Build main 71fa6c35
Release phase
Deploy to Heroku
Your app was successfully deployed.
View
heroku.com Blogs Careers Documentation Support
Terms of Service Privacy Cookies © 2024 Salesforce.com


titanic5thweek · Build | Heroku
dashboard.heroku.com/apps/titanic5thweek/activity/builds/924699d4-f7cd-45c4-811a-0172cd8aec99
HEROKU
from Salesforce
Jump to Favorites, Apps, Pipelines, Spaces...
The Next Generation Platform is Coming - Get Ready Now!
Personal > titanic5thweek
Open app
More
GitHub BlessedGyan/Data-Glacier-Internship
Overview Resources Deploy Metrics Activity Access Settings
Activity Feed > Build Log
ID 924699d4-f7cd-45c4-811a-0172cd8aec99
-----> Building on the Heroku-24 stack
-----> Determining which buildpack to use for this app
-----> Python app detected
-----> No Python version was specified. Using the buildpack default: Python 3.12
To use a different version, see: https://devcenter.heroku.com/articles/python-runtimes
-----> Installing Python 3.12.8
-----> Installing pip 24.3.1, setuptools 70.3.0 and wheel 0.45.1
-----> Installing SQLite3
-----> Installing dependencies using 'pip install -r requirements.txt'
Collecting Flask==2.2.2 (from -r requirements.txt (line 1))
Downloading Flask-2.2.2-py3-none-any.whl.metadata (3.9 kB)
Collecting gunicorn==20.1.0 (from -r requirements.txt (line 2))
Downloading gunicorn-20.1.0-py3-none-any.whl.metadata (3.8 kB)
Collecting numpy==2.1.3 (from -r requirements.txt (line 3))
Downloading numpy-2.1.3-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (62 kB)
Collecting pandas==2.2.3 (from -r requirements.txt (line 4))
Downloading pandas-2.2.3-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (89 kB)
Collecting scikit-learn==1.5.2 (from -r requirements.txt (line 5))
Downloading scikit_learn-1.5.2-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (13 kB)
Collecting Werkzeug==3.1.3 (from -r requirements.txt (line 6))
Downloading werkzeug-3.1.3-py3-none-any.whl.metadata (3.7 kB)
Collecting itsdangerous==2.2.0 (from -r requirements.txt (line 7))
Build finished





Conclusion

The deployment was **successful**, and the model is now accessible via both API and web application on **Heroku**. The process provided hands-on experience in model deployment, cloud computing, and API integration. This project showcases an efficient approach to deploying machine learning models in a cloud environment, making them scalable and accessible.