Project: Machine Learning Model Deployment Using Heroku

(A Cloud and API Deployment)

Data Science Virtual Internship Program

Name: Harold Wilson

Batch code: LISUM19

Submission Date : 5th April 2023

Submitted to: Data Glazier

Data Glacier

Your Deep Learning Partner

Agenda

- Executive Summary
- Business Understanding
- Data Understanding
- Data Preparation
- Data modelling
- Evaluation
- Deployment

Data Glacier

Your Deep Learning Partner

Executive Summary

The main idea of this project is to develop a machine learning algorithm using one of the common toy data set called the iris data and then deploying it on a webserver using Heroku.
Heroku is a container-based cloud Platform as a Service (PaaS) that enables users to deploy and manage their apps. It offers a very flexible platform which is easy to use, and offers users a simple and easy way to run apps on the web

Machine learning is the ability of computers to learn from a given set of data, and to mimic humans to deploy desired outcomes without being programmed.

Machine learning falls under three categories:

- Supervised learning
- Unsupervised Learning
- Reinforcement learning

Model deployment is the process of putting machine learning models into production. The reason why we deploy machine learning models is to make them available for users, developers or systems to interact with the applications thereby making good business decision with it's main goal of solving a problem
There are many ways to deploy machine Learning models, examples are StreamLit, AWS(Amazon Web Services), Flask to name a few. In this project, we will be using heroku to deploy models on to the web.

Data Glacier

Your Deep Learning Partner

Business Understanding

The main objective of this project is to examine the iris data using data mining techniques available, to find an optimal algorithm to correctly predict a plant specie based on their distinctive feature and measurements. In this work, there are four different classifier namely RandomForest, Decision Tree, Logistics Regression and K-nearest Neighbours that will be used to classify the iris dataset.

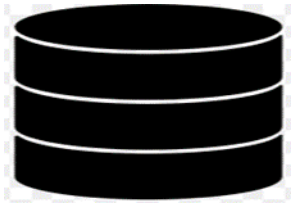
The idea is to automatically predict which class of species at a given time the iris plant belongs to based on their respectively measurement properties by using a machine learning algorithm derived from the iris data set.

Data Glacier

Your Deep Learning Partner

Data Understanding

Iris data set



The iris is an open access flower based dataset and is normally available on UCI dataset, kaggle or from the scikit-learn website . The dataset consist 150 observations with five different attributes/features i.e.,

- Sepal length
- Sepal width
- Petal length
- Petal width
- Class.

This is perhaps the best known data set to be used in pattern recognition settings. The data set contains 3 classes of 50 observations each, where each class refers to a type of iris plant. These types are:

- Iris setosa
- Iris virginica
- Iris versicolor

Data Glacier

Your Deep Learning Partner

Data Preparation

This is the most complex and tedious stage of the process where the data collected needs to be prepared in line with the business plan.

In this stage, the main objective includes cleaning the data, dealing with missing and unknown values, reducing data dimensionality, transforming data values, and sometimes reformatting the data to suite the desired mining solution. Other operations performed under this stage includes data aggregation, normalisation, and attribute creation i.e., making new variables to tackle specific business queries.

The features of the data set, are renamed for simplicity and to make the web application interface more readable.

This is a simple toy data set with no missing values, and no anomalies to the data set ,hence it provides a solid base to be used for the sake of this project

Data Glacier

Your Deep Learning Partner

Data modelling

In this stage, the sole concern is to select a data mining technique from a list of possible mining tasks, activities like generating a test design/modelling solution is sort and a model is built to see how the data performs.
Other activity considered in this stage is the setting of model parameters. Also results should be assessed by all stakeholders to make sure that model can meet data mining objectives

This the post-processing stage where these discovered patterns from mining the data are analysed. These results are evaluated, and the mining process is reviewed and checked against the outcome to determine the best results.

Four algorithms were chosen for this modelling stage namely RandomForest, Decision Tree, Logistics Regression and K-nearest Neighbours.

Steps taken for the modelling stage

- The feature of the data set are put into independent and dependent variables for the model building.
- The iris data was standardised using a standard scaler technique in python
- The iris data set was split into training and testing data in the ratio 0.70: 0.30, 70% for training the data and 30% for testing the outcome.

Data Glacier

Your Deep Learning Partner

Evaluation

Model Evaluation plays a crucial role in the model development process, where it helps to find the best model that fit and represents the data set in question. These models are assessed based on their performance. Data mining is an iterative process; hence the generated modelling results should be evaluated against the business objectives, and sometimes should the accuracy of the mining model be poor, the whole mining process needs to be revisited starting from the business point and understanding of the data.

The iris data was trained with four algorithms/classifiers namely RandomForest, Decision Tree, Logistics Regression and K-nearest Neighbours

Below are the outcome of the results of the classifiers

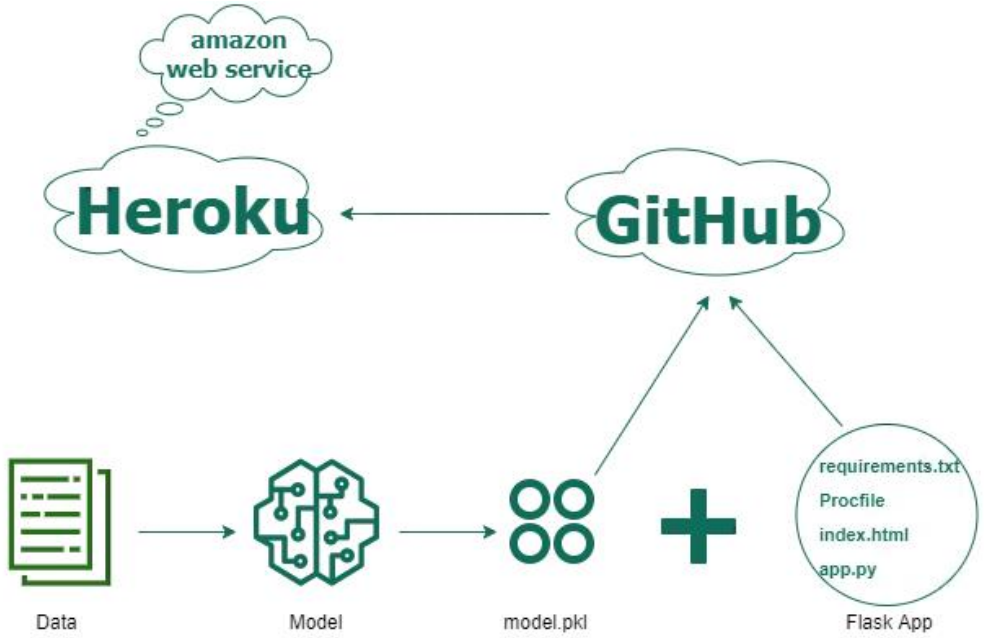
FLOWER CLASS PREDICTION		
Classifier	**Classifier Name**	**Accuracy_Score**
Classifier1	RandomForest RF	0.9555
Classifier2	Decision Tree DT	0.9555
Classifier3	Logistics Regression LR	0.9777
Classifier4	K-nearest neighbours KNN	0.9555

Clearly the Logistics Regression classifier out performed the rest so far with an accuracy score of 97%, while the rest achieved an accuracy score of 95%, hence this will be the best model classifier that could be used to learn from the iris data.

Data Glacier

Your Deep Learning Partner

Deployment



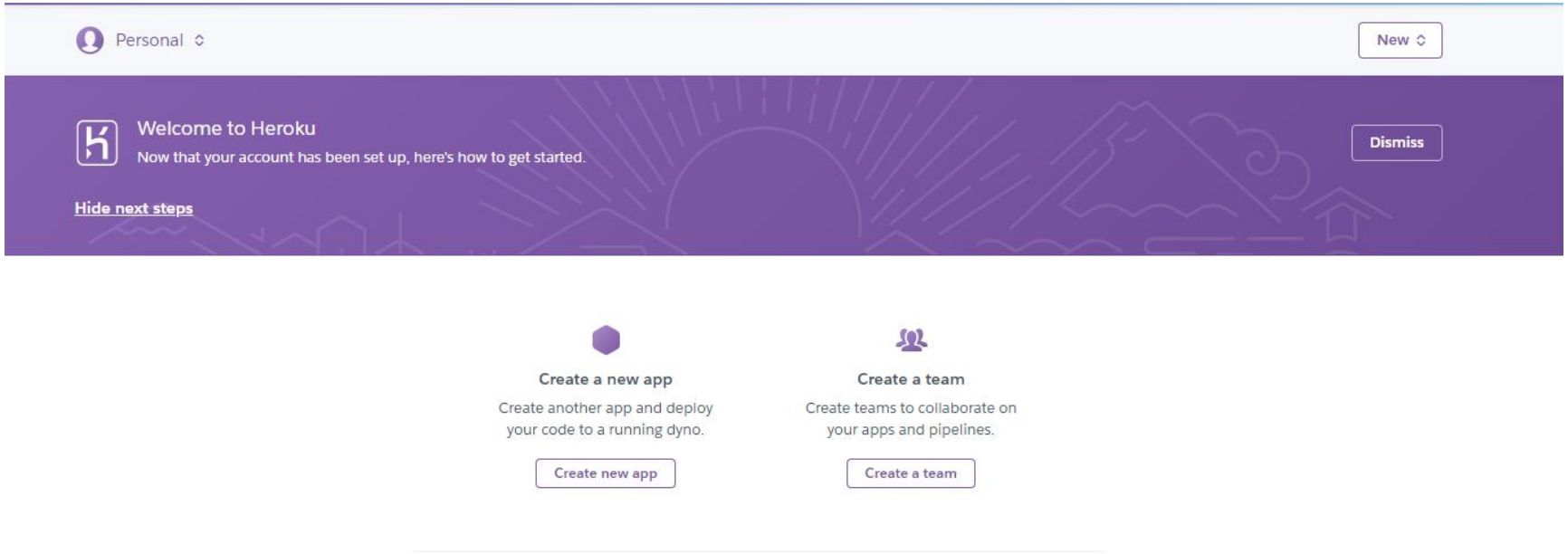
The above shows the different steps taken and the necessary files to have a successful model creation and deployment.

- Templates
- Index.html
- App.py
- Iris csv file
- Model.pkl
- Procfile
- Requirements.txt

Data Glacier

Your Deep Learning Partner

Deployment

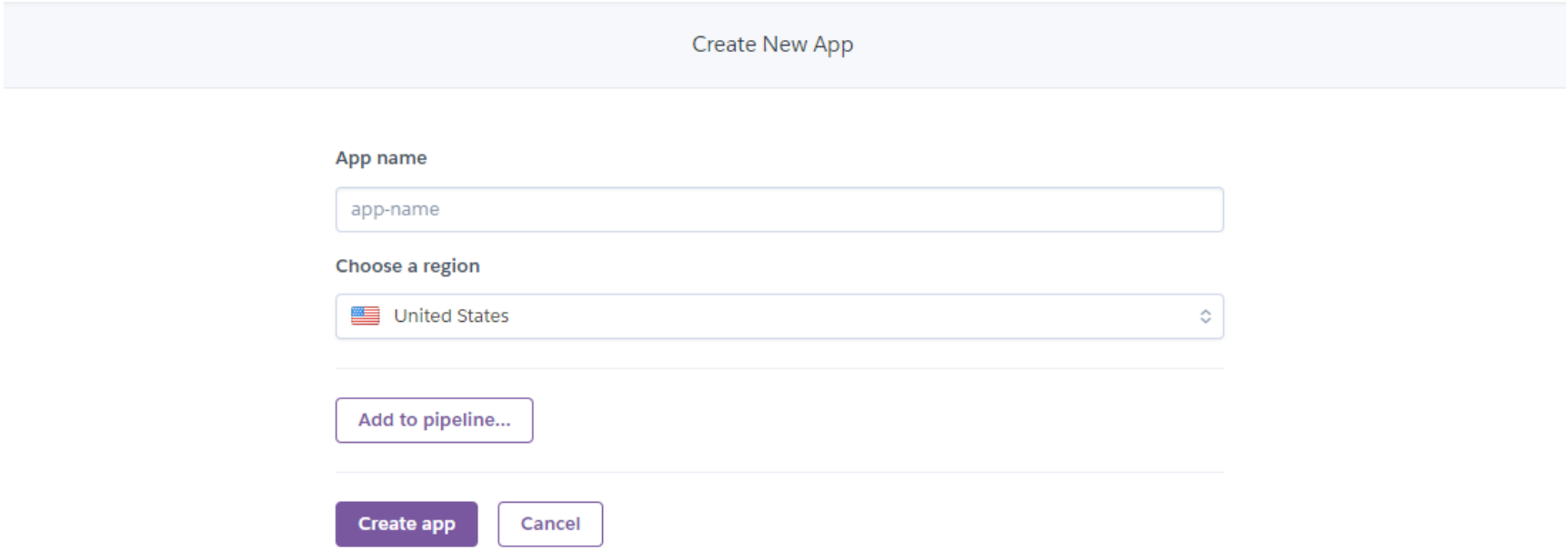


You need to create an account at https://signup.heroku.com/ and go through the sign in process. After the signing process, the above Heroku page opens up on a web browser.

Data Glacier

Your Deep Learning Partner

Deployment

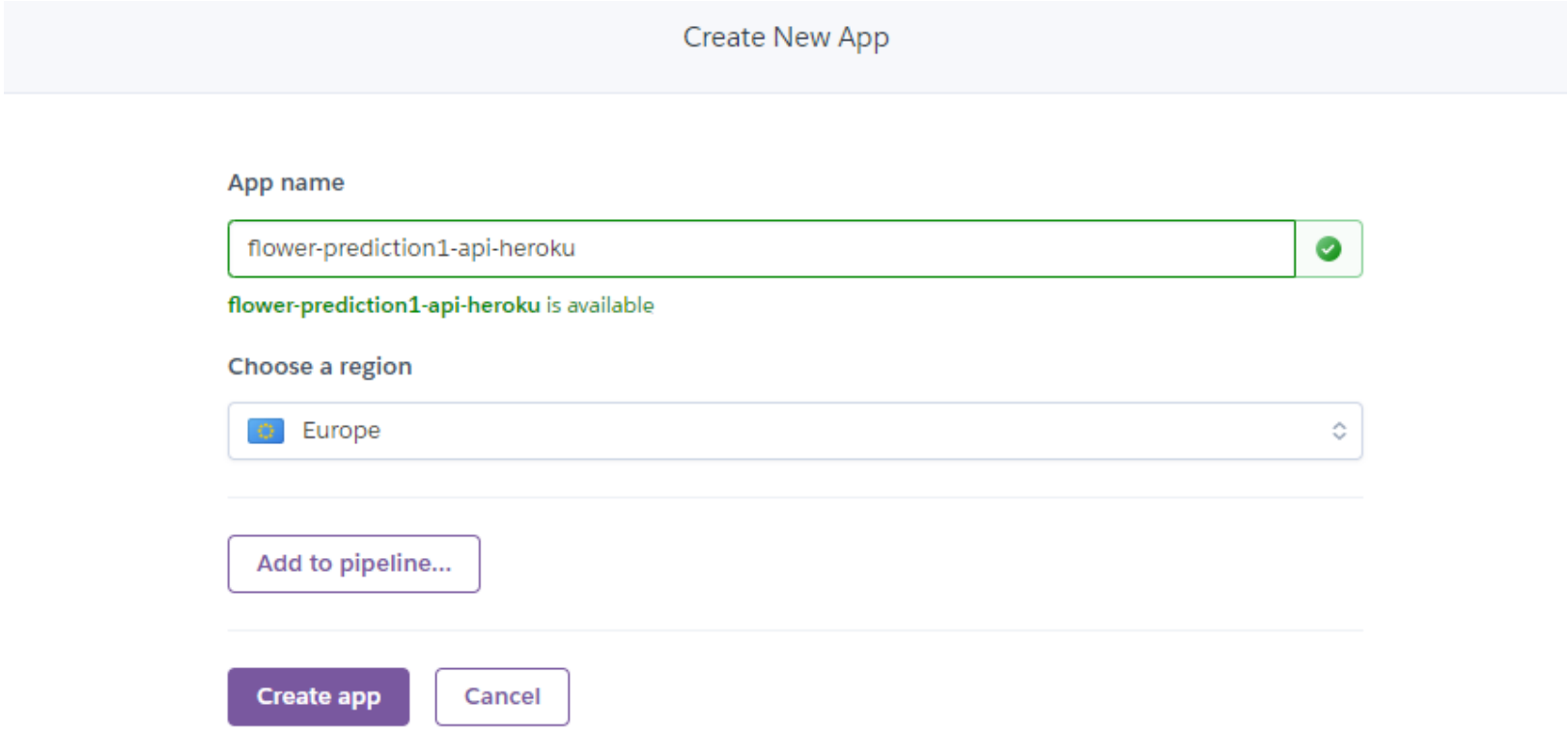


Select create new app , and then name the app and chose a region for your deployment.

Data Glacier

Your Deep Learning Partner

Deployment

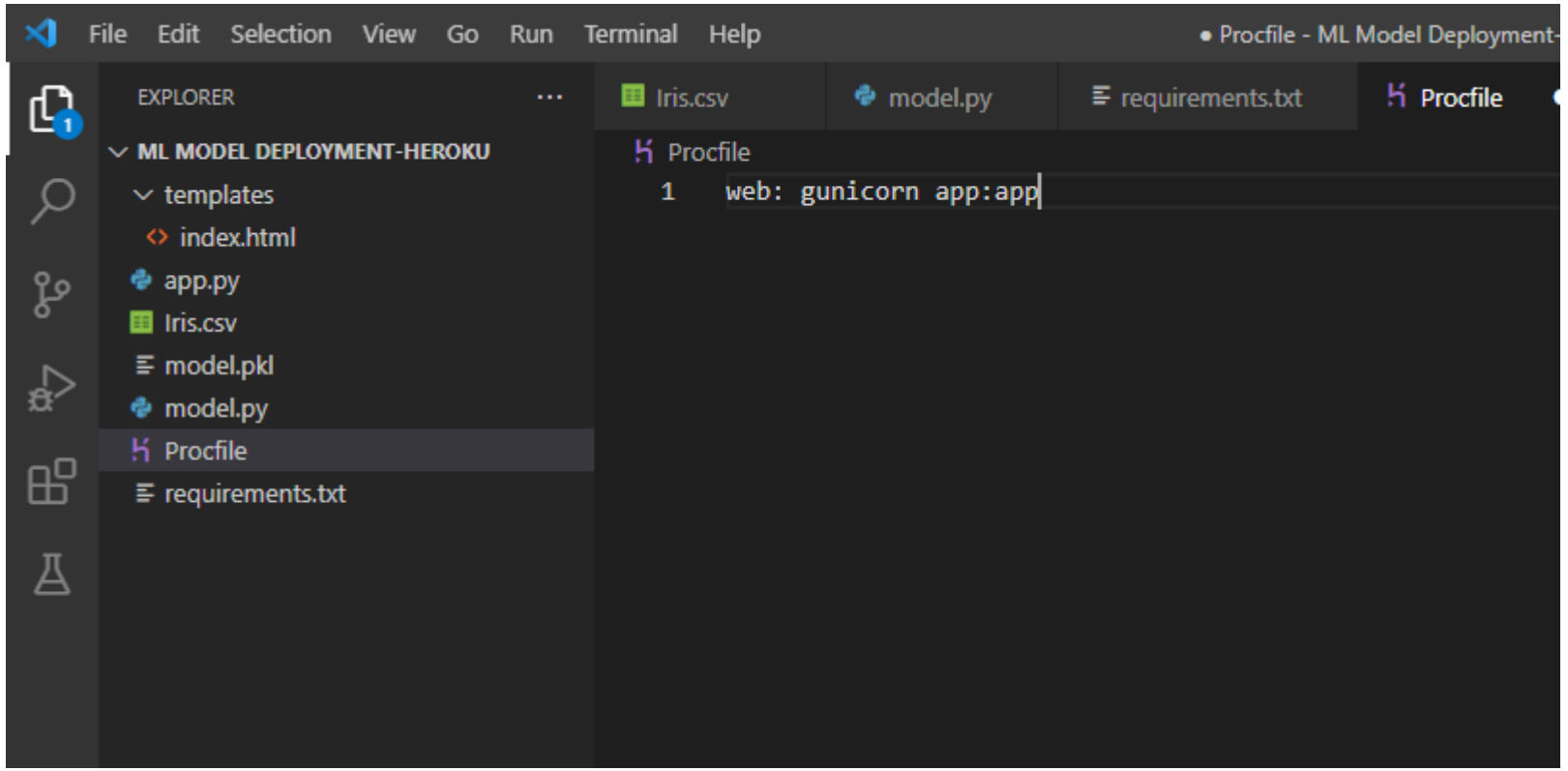


App name and region selected and chosen as shown above.

Data Glacier

Your Deep Learning Partner

Deployment

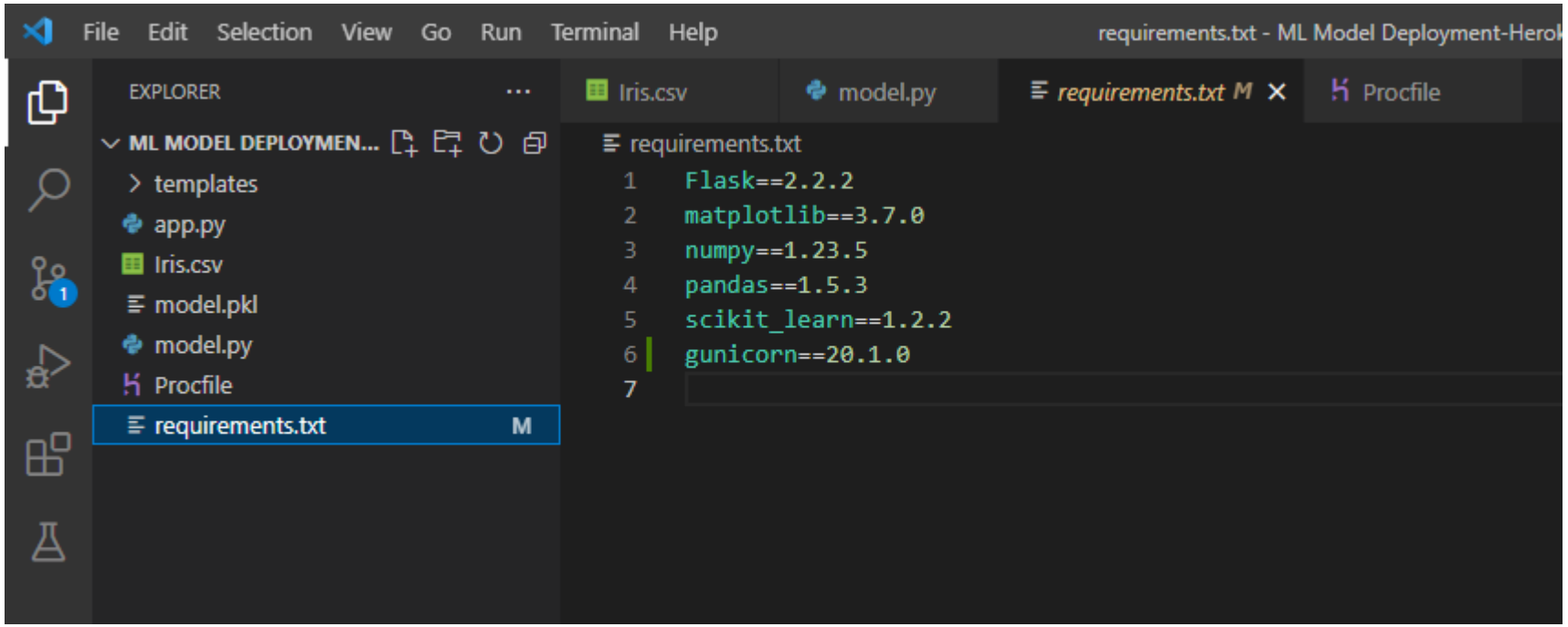


As one of the required files for deployment, we create a file called Procfile.
A Procfile is a simple text file that tells Heroku what kind of app that is going to be run and how to serve it to users. It also handles the commands that are executed by the app on start up.

Data Glacier

Your Deep Learning Partner

Deployment

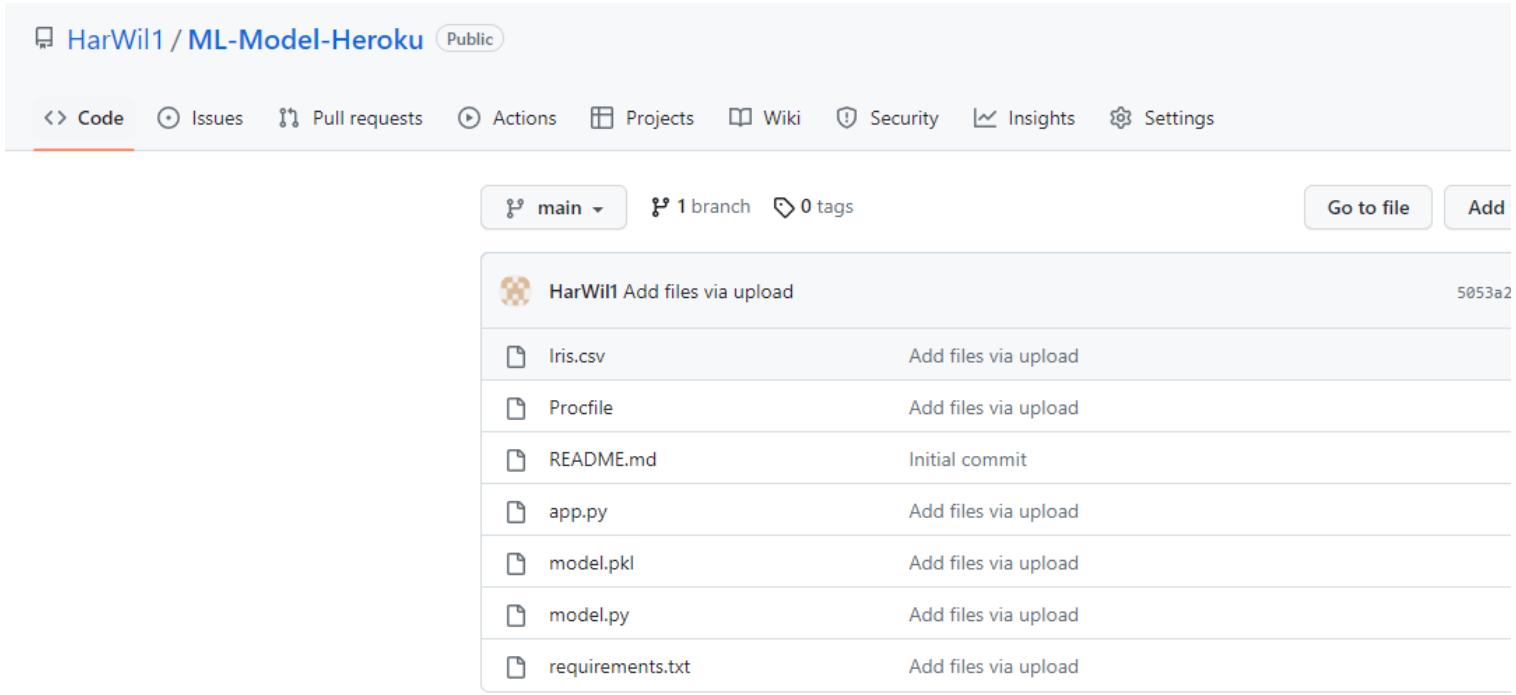


Another required file for deployment, is called the requirement .txt.
A requirement.txt: This file tells Heroku which packages to install necessary for your web app to run.

Data Glacier

Your Deep Learning Partner

Deployment

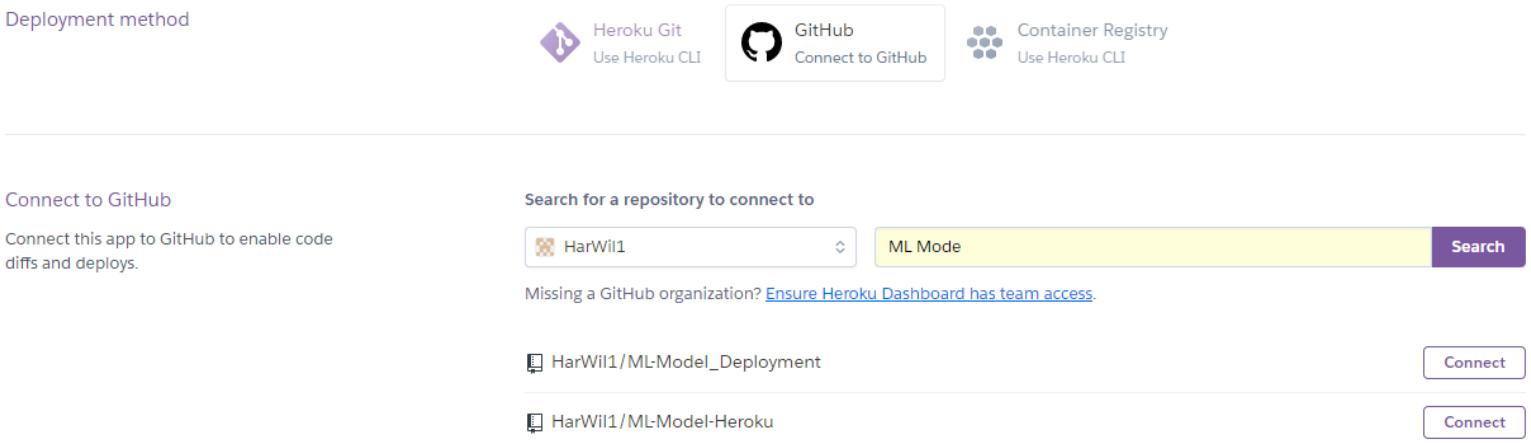


We then upload all the necessary files for the app deployment unto GitHub as shown above.

Data Glacier

Your Deep Learning Partner

Deployment

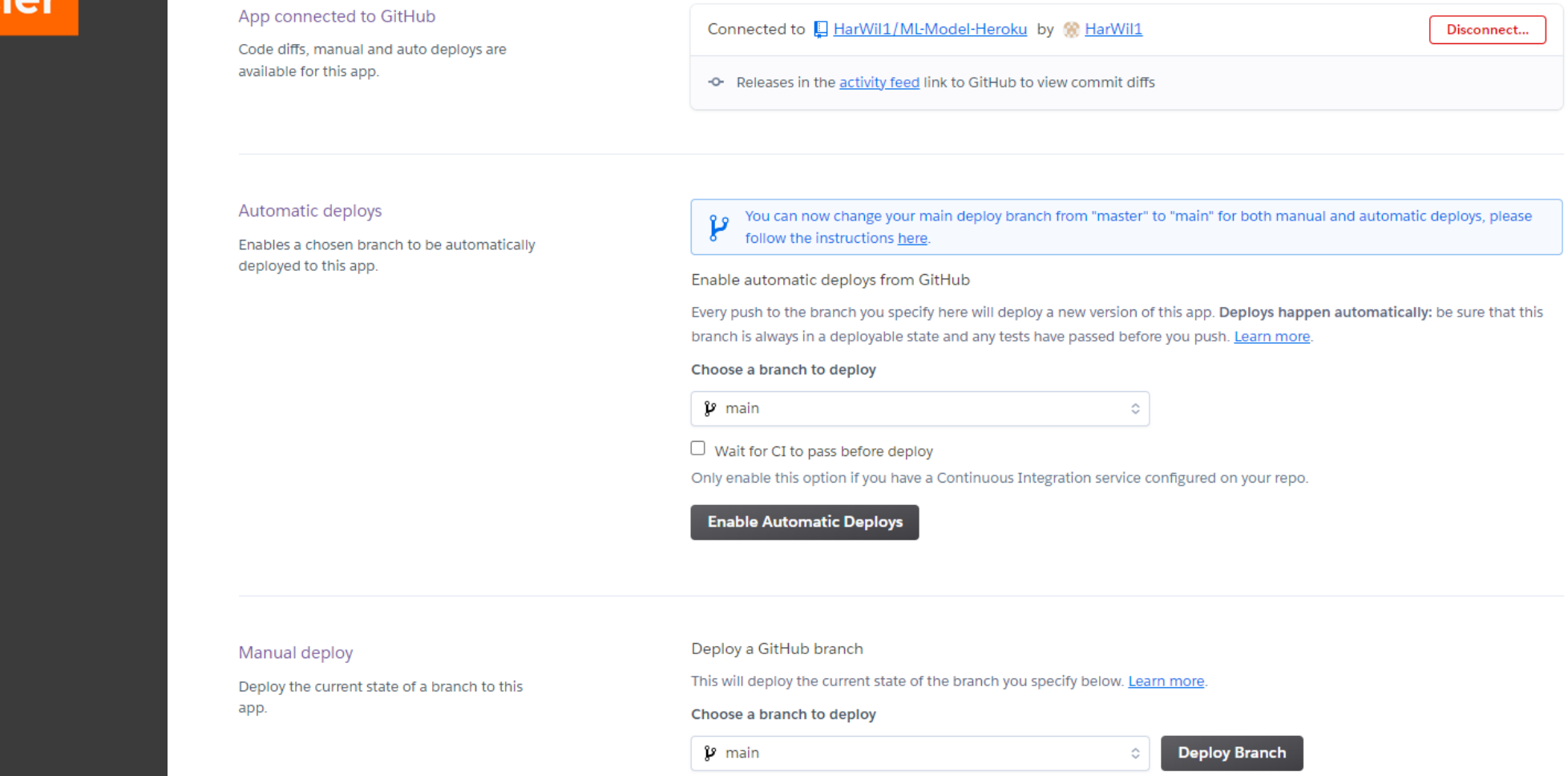


We then link our GitHub page where all the files for deployment resides with Heroku, and establish a connection between these two platforms as shown above.

Data Glacier

Your Deep Learning Partner

Deployment

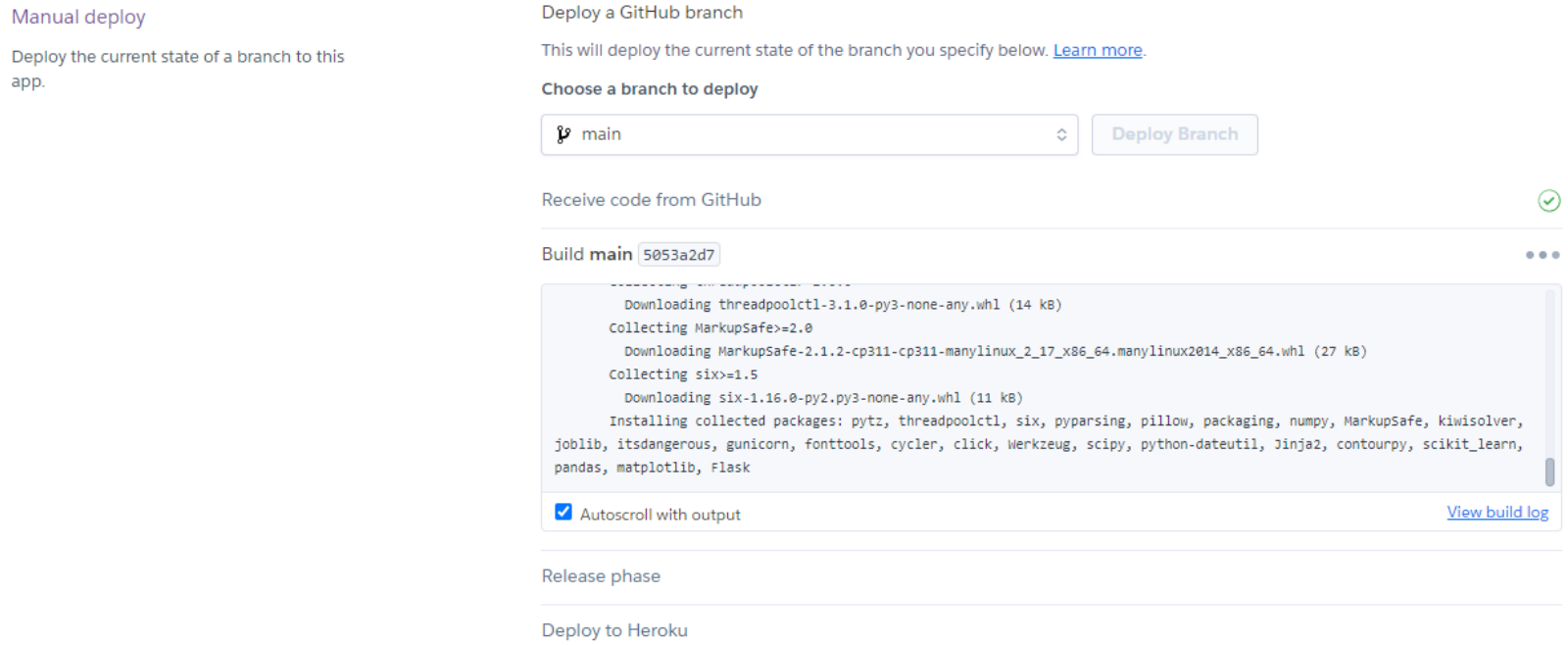


After a successful connection is established, the next step is to deploy the app using the Deploy Branch button as shown above.

Data Glacier

Your Deep Learning Partner

Deployment

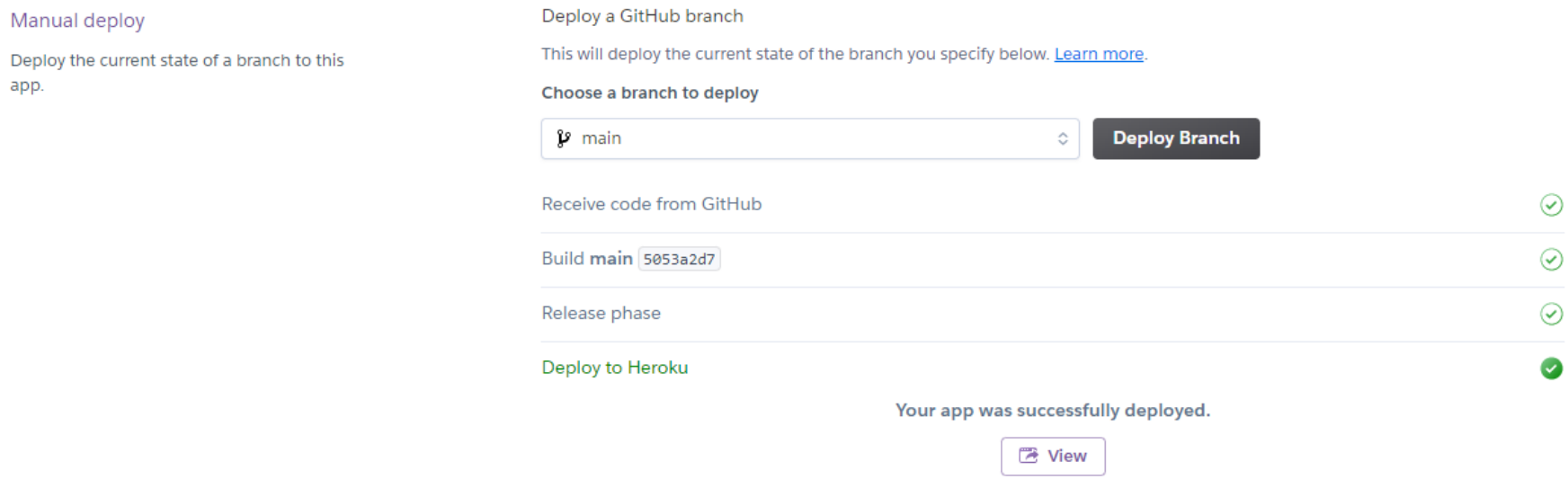


This shows the process of app deployment with building logs scrolling down in the deploy window portal.

Data Glacier

Your Deep Learning Partner

Deployment



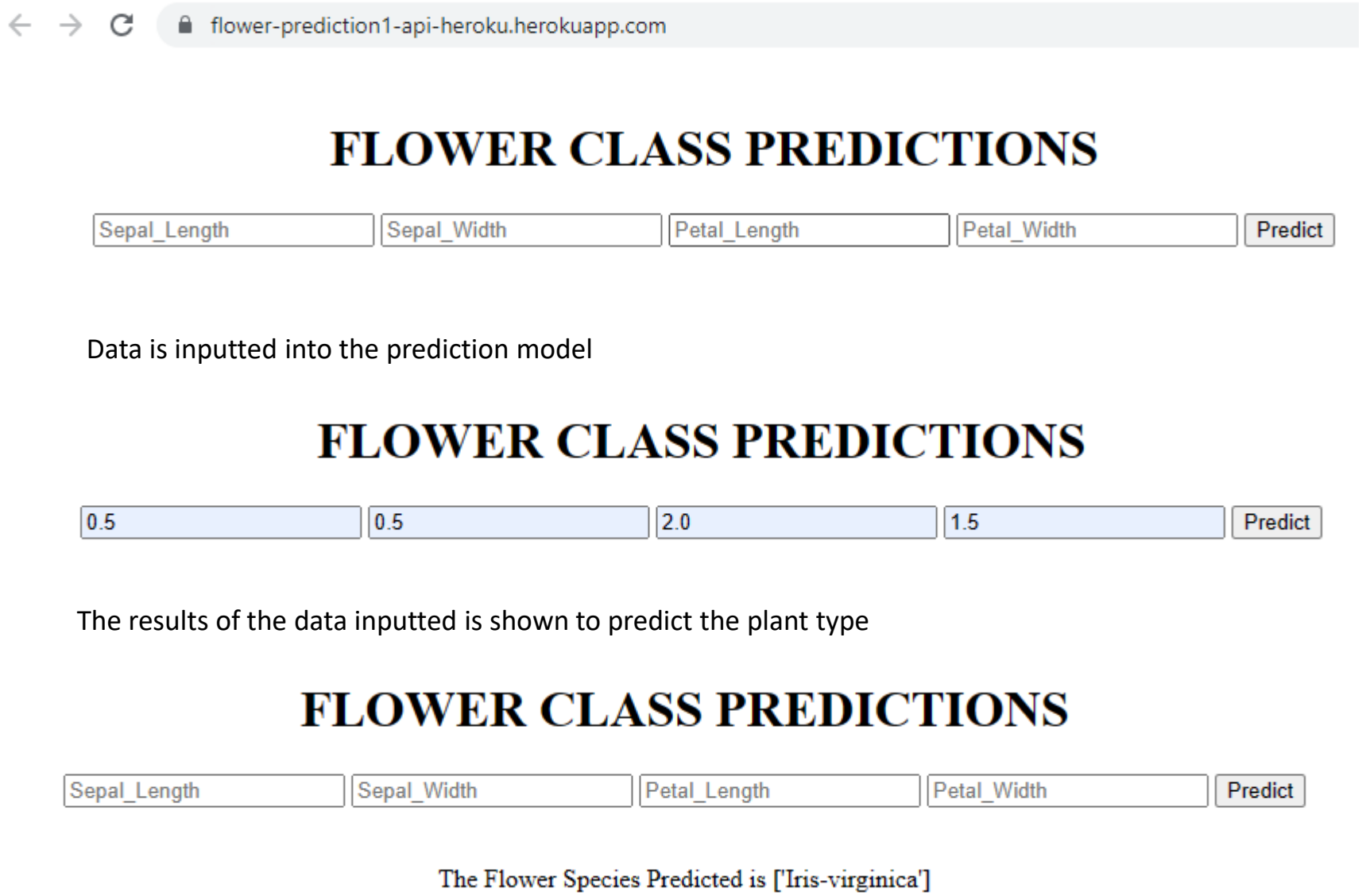
The app successfully gets deployed and generates a view tab where the app could be automatically viewed on a web browser.

Data Glacier

Your Deep Learning Partner

Deployment

The ML Model Deployment API



Data is inputted into the prediction model

The results of the data inputted is shown to predict the plant type

Data Glacier

Your Deep Learning Partner

Conclusion

Finally we have successfully developed a model app that could be run on the web browser using Heroku, once we input the values and click on predict button, the screen will show the class type of the iris plant based on the length and width of sepals and petals.

The machine learning model could be viewed on any web browser by clicking on the view button which takes you straight unto the web browser displaying the created app. In conclusion, we have successfully dealt with how machine learning models are built, how to connect them onto a web interface application, and deploying it using Heroku which is user friendly where one can create a server and deploy it in no time. It is also a very excellent choice for small or medium size deployment where most of the operations like auto-scaling, logging, monitoring, health check and configurations are all backed by a AWS a major player for providing cloud based solutions.

Your Deep Learning Partner

Thank You