Cloud deployment on Heroku

Name: Ignacio Solórzano

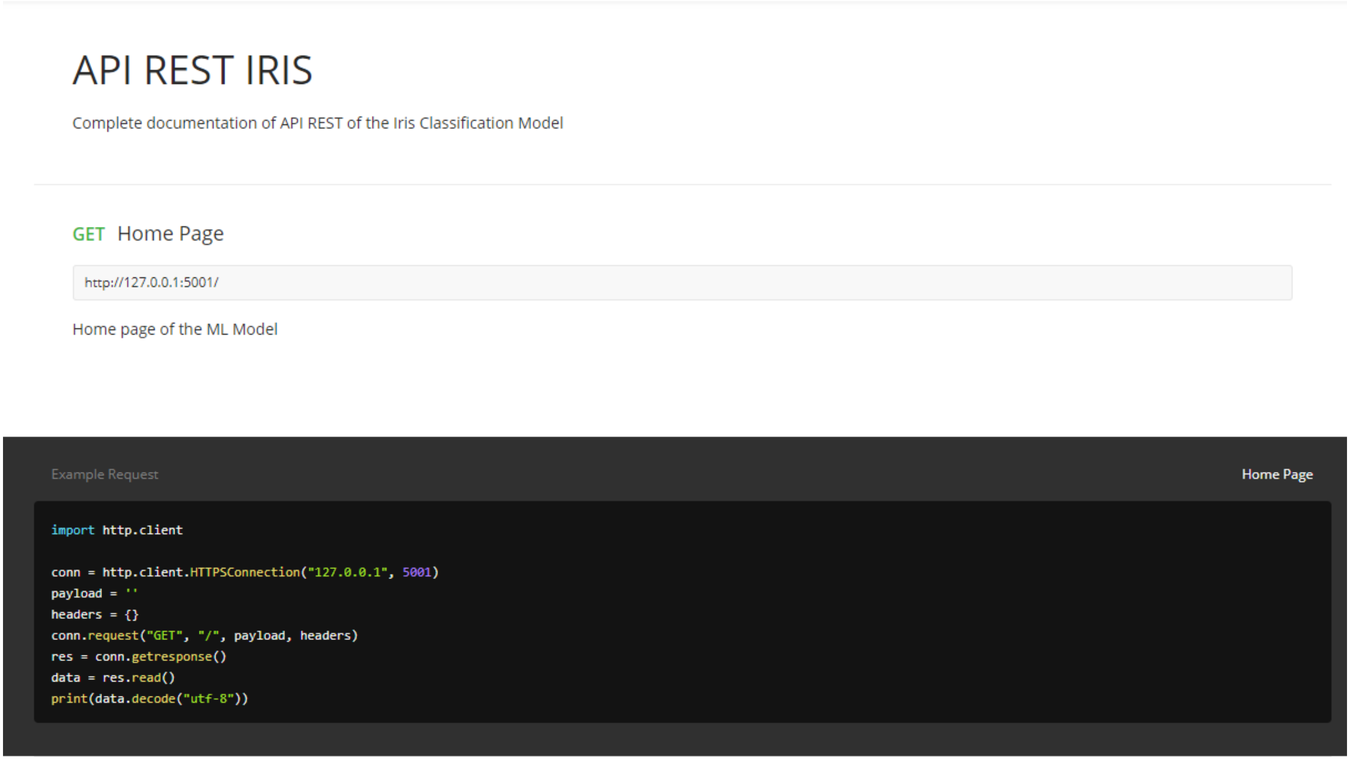
Batch code: LISP01

Submitted date: 26/03/2021

Submitted to: Data Glacier

1- API Deployment

Postman Deployment Documentation



API REST IRIS

Complete documentation of API REST of the Iris Classification Model

GET Home Page

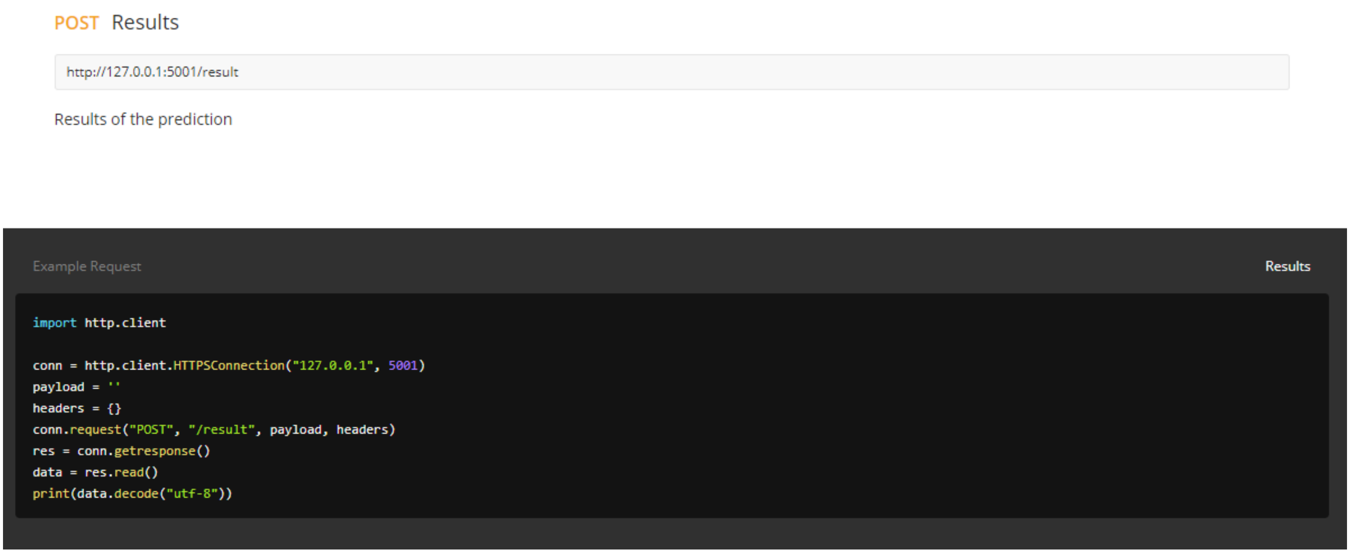
http://127.0.0.1:5001/

Home page of the ML Model

Example Request

Home Page

```
import http.client

conn = http.client.HTTPSConnection("127.0.0.1", 5001)
payload = ''
headers = {}
conn.request("GET", "/", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```



POST Results

http://127.0.0.1:5001/result

Results of the prediction

Example Request

Results

```
import http.client

conn = http.client.HTTPSConnection("127.0.0.1", 5001)
payload = ''
headers = {}
conn.request("POST", "/result", payload, headers)
res = conn.getresponse()
data = res.read()
print(data.decode("utf-8"))
```

You can check the documentation in this path: https://documenter.getpostman.com/view/15163582/TzCL99C8

Repository

This project has the next structure:

1-. Idea (folder)

- . gitignore
- Flask.iml

2- model (folder)

- model.pkl
- model.py

3- static (folder)

--css (folder)

---demo2.css

4-templates (folder)

--home

--result

--template

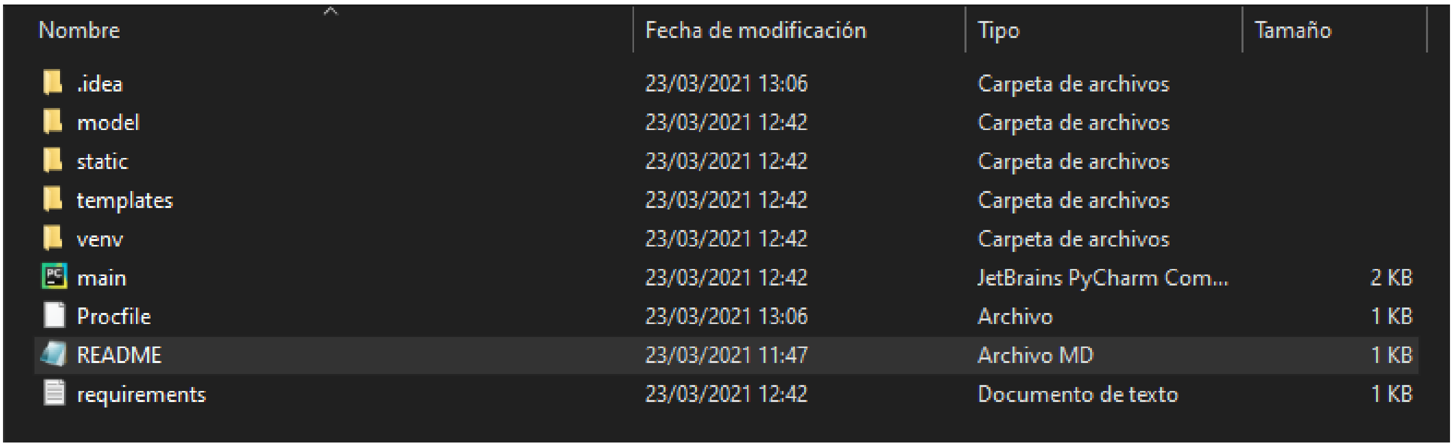
5-venv (folder)

7- **main.py**

8- Procfile

9- README.txt

10-requirements.txt



Nombre	Fecha de modificación	Tipo	Tamaño
.idea	23/03/2021 13:06	Carpeta de archivos	
model	23/03/2021 12:42	Carpeta de archivos	
static	23/03/2021 12:42	Carpeta de archivos	
templates	23/03/2021 12:42	Carpeta de archivos	
venv	23/03/2021 12:42	Carpeta de archivos	
main	23/03/2021 12:42	JetBrains PyCharm Com...	2 KB
Procfile	23/03/2021 13:06	Archivo	1 KB
README	23/03/2021 11:47	Archivo MD	1 KB
requirements	23/03/2021 12:42	Documento de texto	1 KB

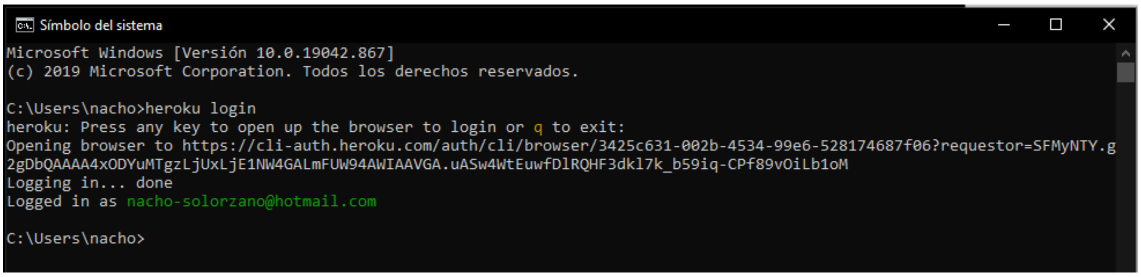
1- Getting Ready for Deployment

Following the line of my last work “ML-Deployment-on-Flask”, I keep with the same repository (so the same files) and deploy this model on any open-source cloud.

I choose Heroku platform to deploy my model. So, the first step was to install Heroku and signing in.

Ones we have created the account we are ready to log in and start creating our webapp

```
heroku login
```



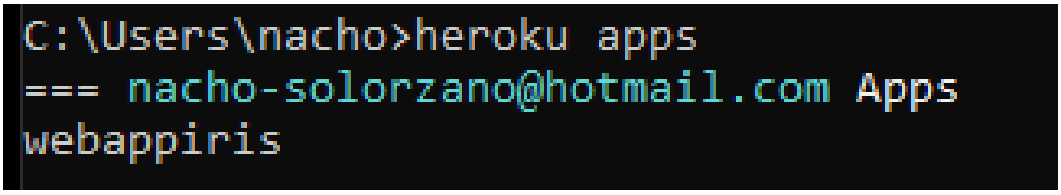
```
Microsoft Windows [Versión 10.0.19042.867]
(c) 2019 Microsoft Corporation. Todos los derechos reservados.

C:\Users\nacho>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/3425c631-002b-4534-99e6-528174687f06?requestor=SFMyNTY.g
2gDbQAAAA4xODYuMTgzLjUxLjE1NW4GALmFUW94AWIAAVGA.uASw4WtEuwfDlRQHF3dkl7k_b59iq-CPf89vOiLb1oM
Logging in... done
Logged in as nacho-solorzano@hotmail.com

C:\Users\nacho>
```

2- Creating Web App

The first step is to create the Web app. We can do it directly from the command prompt with the next piece of code:

```
heroku apps: name_of_webapp
```

The name of the Web app is “webappiris”

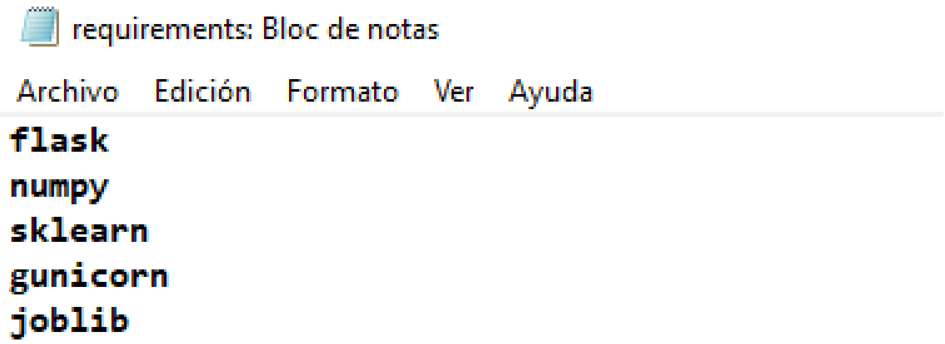


```
C:\Users\nacho>heroku apps
=== nacho-solorzano@hotmail.com Apps
webappiris
```

As I have already created the web app, I can access to the list of every applications I have created with the command: `heroku apps`

3- File requirements.txt

This is the first point of entry into the program. It will install all the necessary dependencies to run your Code. ```requirements.txt``` will tell heroku that this project will require all these libraries to run the application correctly.

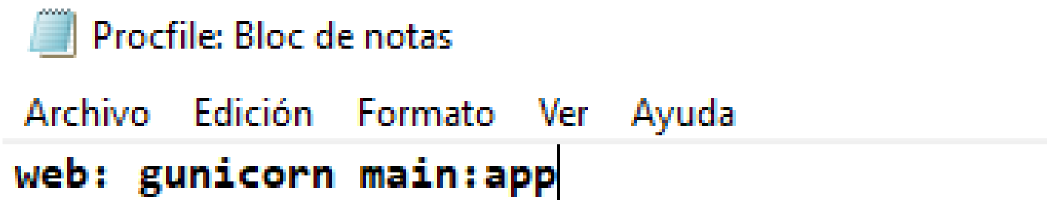


```
flask
numpy
sklearn
gunicorn
joblib
```

4- File Procfile

Heroku requires Procfile to be present in the root directory of your application. It will tell Heroku how to run the application. This file must be a simple file with no extension

The part to the left of the colon is the type of process and the part to the right is the command to run to start that process. In this, we can say on which port the code should be deployed and it can start and stop these processes.



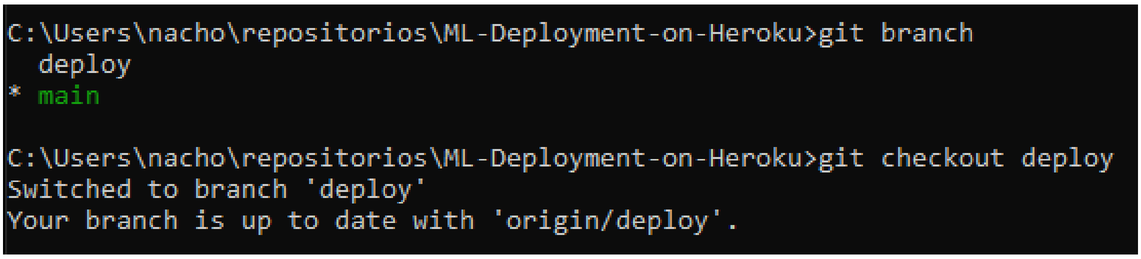
```
web: gunicorn main:app
```

This file tells heroku that we want to use the web process with the Gunicorn command and the name of the application.

5- Deployment

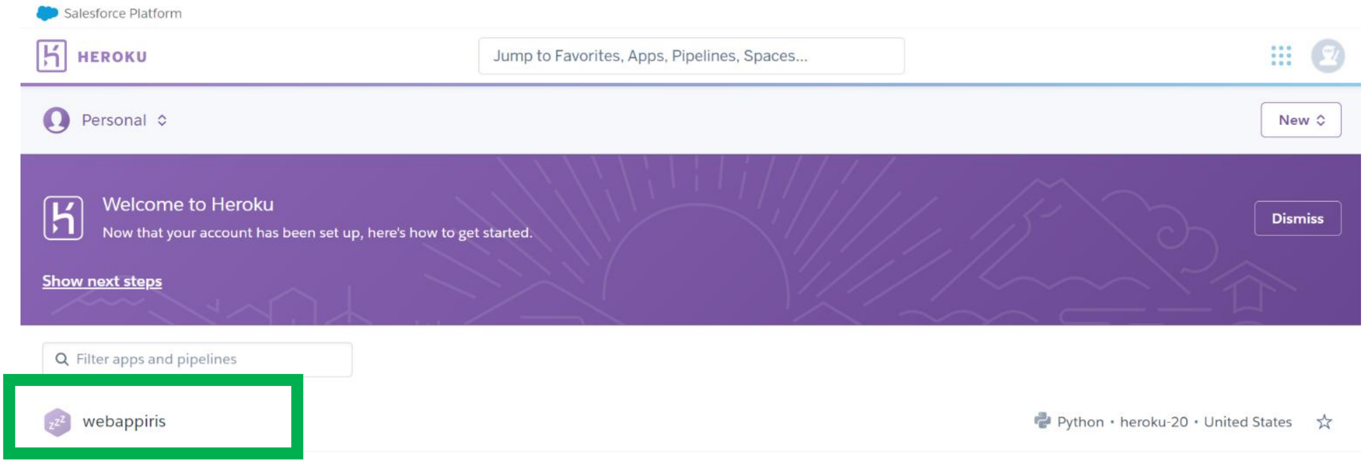
Before we deploy the web app, we must be sure that Procfile and requirements.txt are in the main root of our repository.

Then we will create the repository on GitHub and create a brunch called "deploy" to connect the automatic deploy from GitHub. This means, every modification in the deploy brunch will modify the web app automatically.



```
C:\Users\nacho\repositorios\ML-Deployment-on-Heroku>git branch
  deploy
* main

C:\Users\nacho\repositorios\ML-Deployment-on-Heroku>git checkout deploy
Switched to branch 'deploy'
Your branch is up to date with 'origin/deploy'.
```

Once we have created the deploy branch, we must set up the automatic deployment from GitHub to Heroku. We have to go to our Heroku dashboard https://dashboard.heroku.com/apps

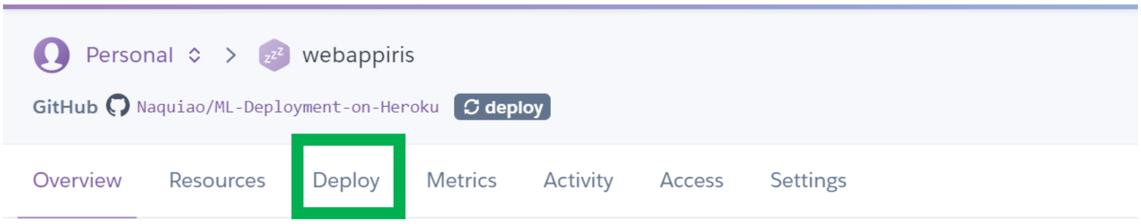

Salesforce Platform
HEROKU
Jump to Favorites, Apps, Pipelines, Spaces...
Personal
New
Dismiss
Show next steps
Filter apps and pipelines

Welcome to Heroku

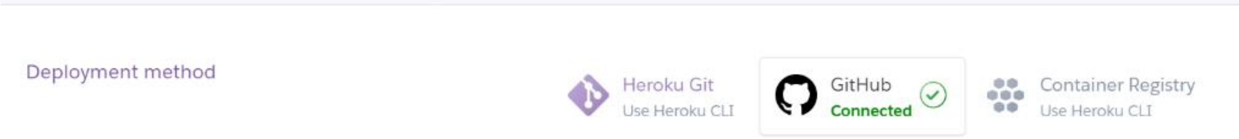
Now that your account has been set up, here's how to get started.

webappiris

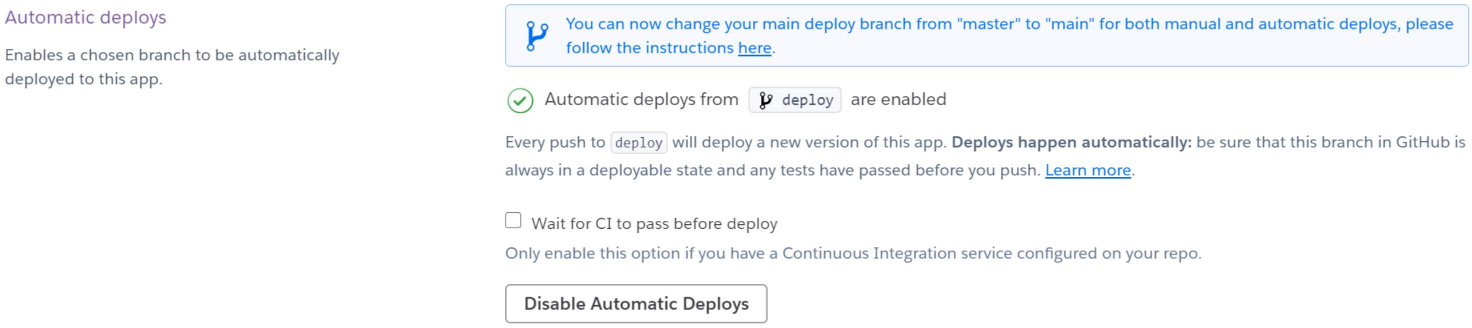
Python • heroku-20 • United States



Then select the Deployment method as GitHub and connect the repository from GitHub



Once we have connected the repository, select the branch deploy and enable the automatic deploy.



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.

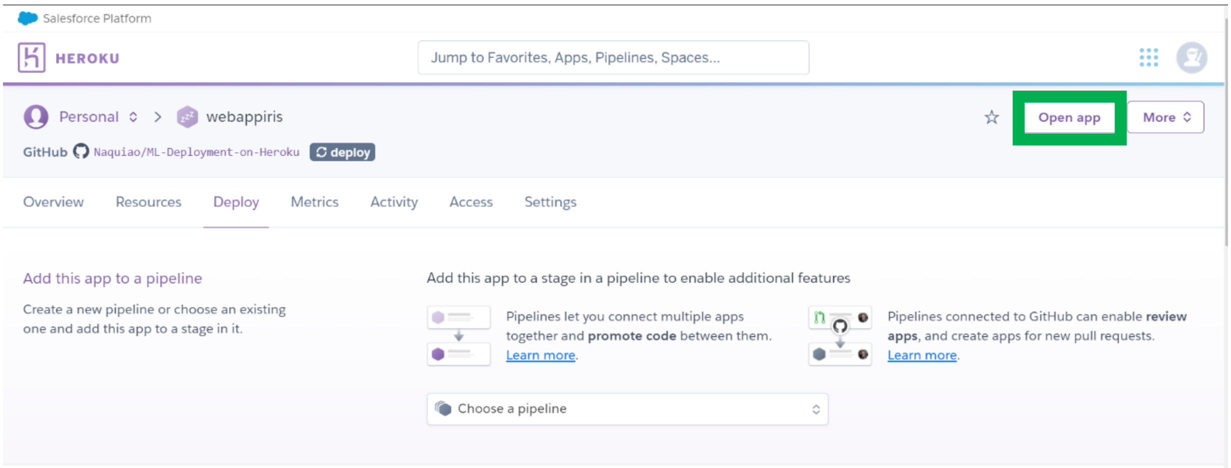
✓ Automatic deploys from `deploy` are enabled

Every push to `deploy` will deploy a new version of this app. **Deploys happen automatically:** be sure that this branch in GitHub is always in a deployable state and any tests have passed before you push. Learn more.

□ Wait for CI to pass before deploy

Only enable this option if you have a Continuous Integration service configured on your repo.

6 – Open app



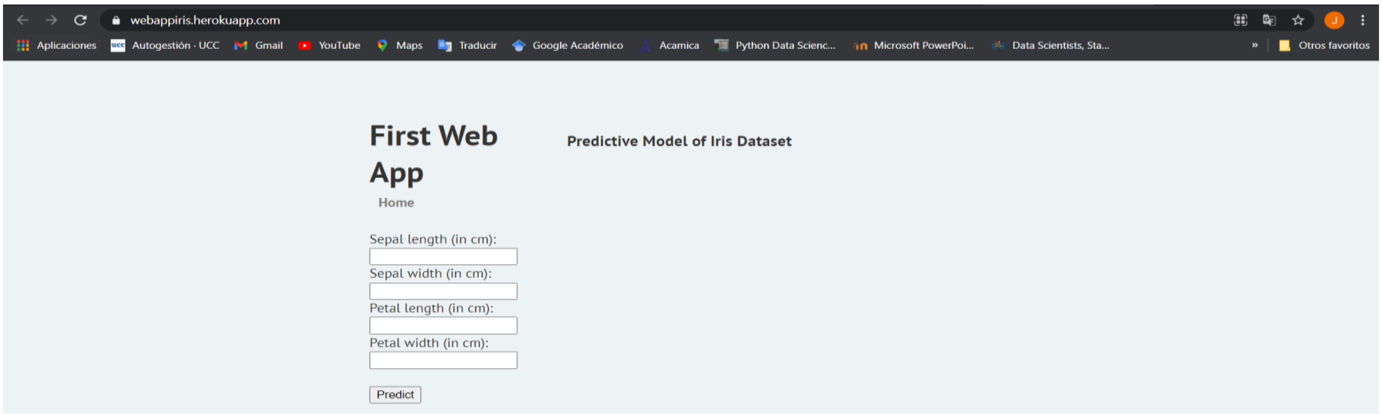
Add this app to a pipeline

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.



First Web App

Home

Predictive Model of Iris Dataset

Sepal length (in cm):

Sepal width (in cm):

Petal length (in cm):

Petal width (in cm):

You can check the webapp: https://webappiris.herokuapp.com/

Finally, every time we make changes in the repository of our branch “deploy” and then get those changes push, our web app will be automatically modified.