

# Model Deployment on Heroku

Name: Robin Masawi

Batch Code: LISP01

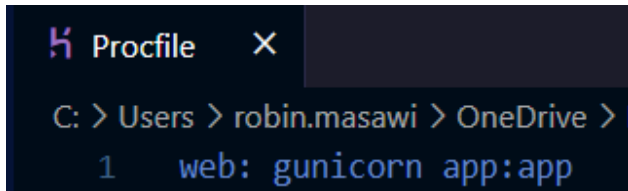
Submission Date: 24-March-2021

Submitted To: Data Glacier

1. Used the same model that we had when we did the Flask deployment.  
So the same app.py file was used.

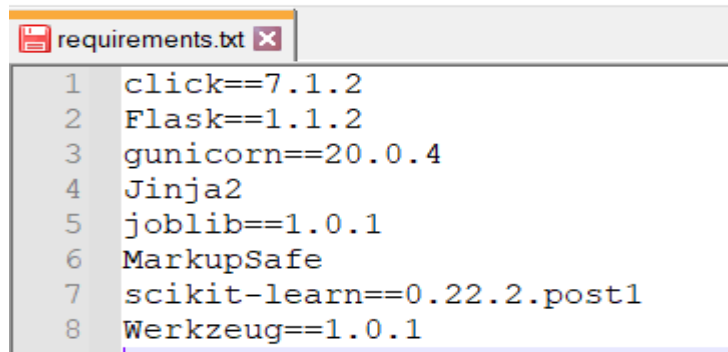
```
app.py x
app.py > ...
1  import numpy as np
2  from flask import Flask, request, jsonify, render_template
3  import pickle
4
5  app = Flask(__name__)
6  model = pickle.load(open('model.pkl', 'rb'))
7
8  @app.route('/')
9  def home():
10
11      return render_template('index.html')
12
13
14  @app.route('/predict', methods=['POST'])
15  def predict():
16
17      ...
18
19      For rendering results on HTML GUI
20
21      ...
22
23      float_features = [float(x) for x in request.form.values()]
24      final_features = [np.array(float_features)]
25      prediction = model.predict(final_features)
26
27      output = prediction[0]
28
29      return render_template('index.html', prediction_text='Diabetes progression one year after baseline is {}'.format(output))
30
31
32
33  @app.route('/predict_api', methods=['POST'])
34  def predict_api():
35
36      ...
37
38      For direct API calls through request
39
40      ...
41
42      data = request.get_json(force=True)
43      prediction = model.predict([np.array(list(data.values()))])
44
45      output = prediction[0]
46
47      return jsonify(output)
48
49
50
51  if __name__ == "__main__":
52
53      app.run(debug=True)
```

2. Create file Procfile (no file extension) which has “web: gunicorn app: app”.  
The web implies that this is a web app and gunicorn is the server on which our app will run.  
The first app represents the file name from where the web app should start. The second app represents the name of the app.



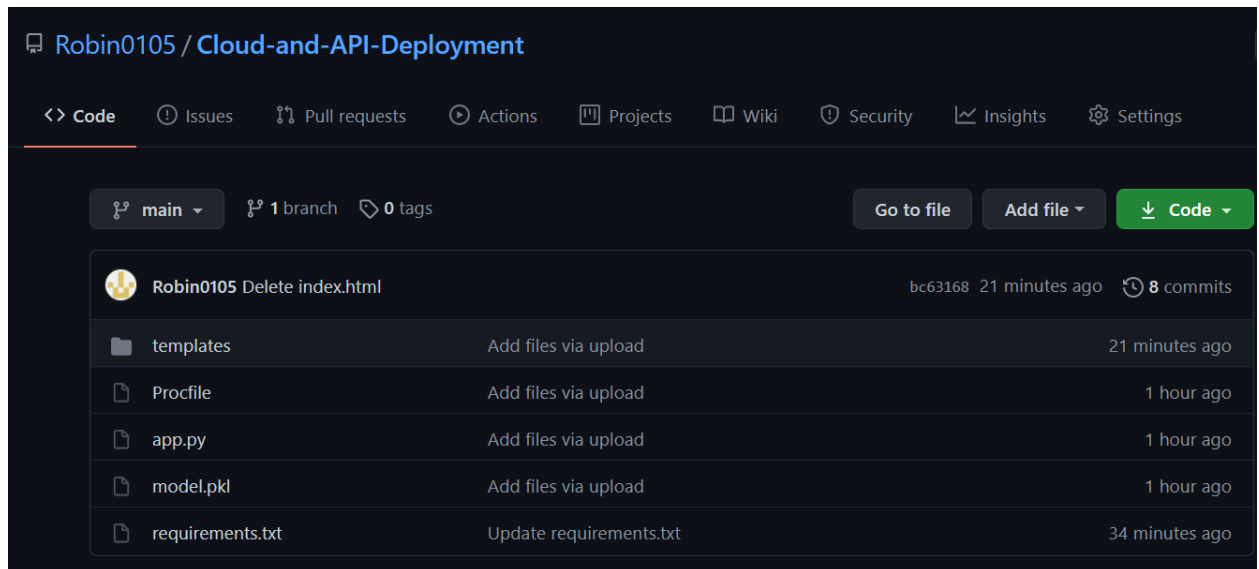
```
Procfile X
C: > Users > robin.masawi > OneDrive > I
1 web: gunicorn app:app
```

3. Create a requirements.txt file where we will add all the libraries that our web app will use.



```
requirements.txt X
1 click==7.1.2
2 Flask==1.1.2
3 gunicorn==20.0.4
4 Jinja2
5 joblib==1.0.1
6 MarkupSafe
7 scikit-learn==0.22.2.post1
8 Werkzeug==1.0.1
```

4. A new repository was created on GitHub and all the required files were uploaded into the repository.



5. Create an account on Heroku.  
Once the account is created, click on new and then create a new app.

App name

diabetesprogression

diabetesprogression is available

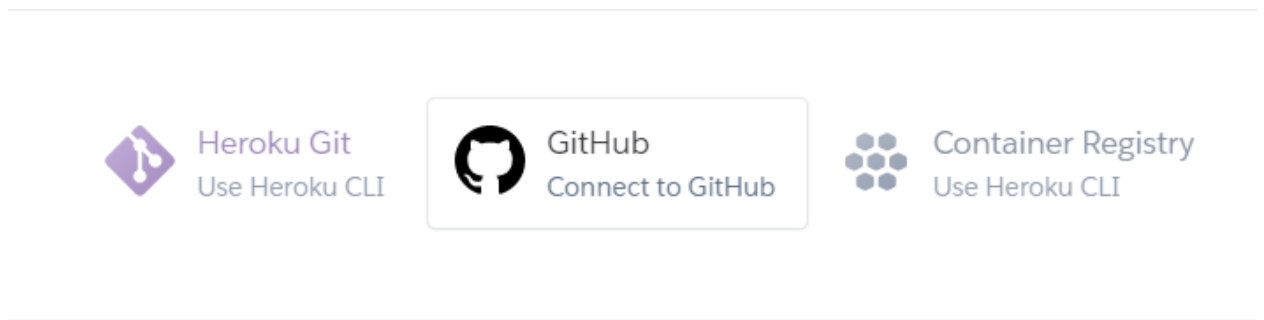
Choose a region

United States

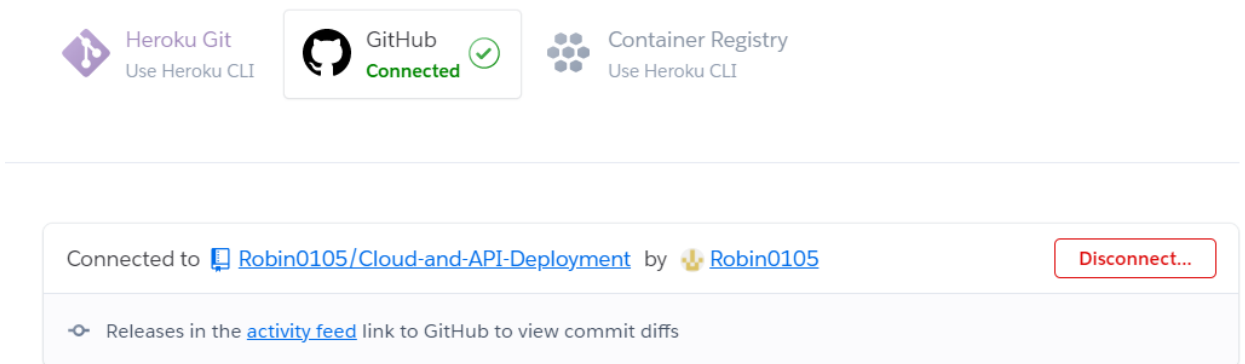
Add to pipeline...

Create app

- Once app has been created there will be various deployment options under Deploy, choose GitHub.  
A prompt window will pop up to authorize the connection between Heroku and GitHub, authorize.



- Click on Search and connect to the repository created on GitHub.





8. Scroll downwards and click on Deploy Branch.

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 bc631682



Release phase



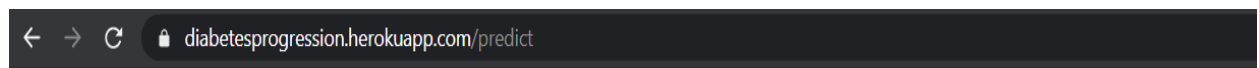
Deploy to Heroku



Your app was successfully deployed.

 View

9. Click on Open App to visit the Web Application.



## Quantitative Measure of Diabetes Progression One Year After Baseline

Age	Sex	BMI	BP	Predict measure of diabetes progression
-----	-----	-----	----	---

Diabetes progression one year after baseline is [68.0333446]