

Deployment on Heroku

Name: Anticipating Diabetes With AI

Report date: 9-Jul-2021

Internship Batch: LISUM01

Report intake by: Yousef Elbayoumi

Report intake reviewer: Data Glacier team

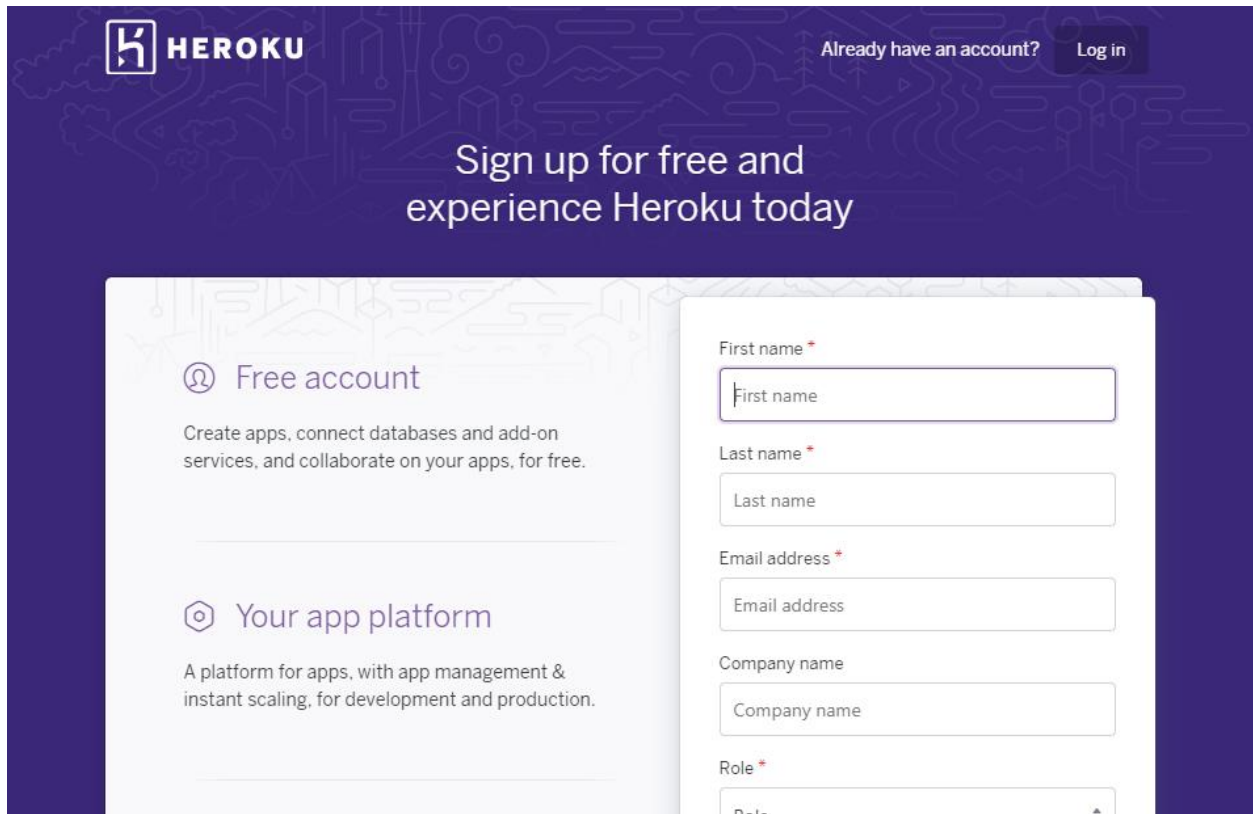
Report location: https://github.com/yousefhosam-b/DataGlacier_Internship/tree/main/Task5

Tabular data details:

Total number of observations	768 rows
Total number of files	1 file
Total number of features	9 columns
Base format of the file	csv file
Size of the data	24 KB

Starting with Heroku:

Create an account:

The image shows the Heroku sign-up page. At the top left is the Heroku logo. To its right, there's a link "Already have an account?" and a "Log in" button. The main heading says "Sign up for free and experience Heroku today". Below this, there are two columns. The left column has a "Free account" section with a description: "Create apps, connect databases and add-on services, and collaborate on your apps, for free." Below that is a "Your app platform" section with a description: "A platform for apps, with app management & instant scaling, for development and production." The right column contains a sign-up form with fields for "First name", "Last name", "Email address", "Company name", and "Role". Each field has a corresponding input box with placeholder text. The "Role" field has a dropdown arrow on the right.


Create an app:


Create New App

App name

app-name

Choose a region

 United States

 Add to pipeline...

Create app

The App should look like this:



Downloading Heroku:

Download from Heroku's website:

Introduction

Set up

Prepare the app

Deploy the app

View logs

Define a Procfile

Scale the app

Declare app dependencies

Run the app locally

Push local changes

Provision add-ons


Start a console

Define config vars

Provision a database

Next steps


Set up

 The Heroku CLI requires Git, the popular version control system. If you don't already have Git installed, complete the following before proceeding:

- [Git installation](#)
- [First-time Git setup](#)

In this step you'll install the Heroku Command Line Interface (CLI). You use the CLI to manage and scale your applications, provision add-ons, view your application logs, and run your application locally.


Download and run the installer for your platform:

 **macOS**

Download the installer

Also available via Homebrew:

```
$ brew install heroku/brew/heroku
```

 **Windows**

Download the appropriate installer for your Windows installation:

64-bit installer

32-bit installer

Submit Feedback

Checking if Heroku was downloaded successfully:

As the commands can be shown, the installation completed successfully.

```
Microsoft Windows [Version 10.0.19041.1083]
(c) Microsoft Corporation. All rights reserved.

C:\Users\sefaa>heroku
CLI to interact with Heroku

VERSION
  heroku/7.56.0 win32-x64 node-v12.21.0

USAGE
  $ heroku [COMMAND]

COMMANDS
  access      manage user access to apps
  addons      tools and services for developing, extending, and operating your app
  apps        manage apps on Heroku
  auth        check 2fa status
  authorizations OAuth authorizations
  autocomplete display autocomplete installation instructions
  buildpacks  scripts used to compile apps
  certs       a topic for the ssl plugin
  ci          run an application test suite on Heroku
  clients     OAuth clients on the platform
  config       environment variables of apps
  container   Use containers to build and deploy Heroku apps
  domains     custom domains for apps
  drains       forward logs to syslog or HTTPS
  features    add/remove app features
  git         manage local git repository for app
  help        display help for heroku
```

Preparing the Required Files:

Preparing **Requirements.txt** Text Document:

The text document was created manually, but I used “pip freeze > requirements.txt” command.

Preparing **Procfile** File:

The file was created manually.



Preparing **The Model and Python** File:

I used the same model and python file from Flask task, therefore I didn't have to do anything here.

Uploading the Files to GitHub:

The screenshot shows the GitHub interface for the repository 'yousefhosam-b / Deployment_on_Heroku'. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation bar, there are buttons for 'main' branch, '1 branch', and '0 tags', along with 'Go to file', 'Add file', and 'Code' buttons. The main content area displays a commit history table with the following entries:

Commit Hash	Commit Message	Time
ad3586b	Update requirements.txt	20 hours ago
	Added Procfile File	20 hours ago
	Create README.md	20 hours ago
	Code for Deployment	20 hours ago
	Update requirements.txt	20 hours ago

Connecting Heroku with GitHub Repository:

The screenshot shows the Heroku dashboard with three main options: 'Heroku Git Use Heroku CLI', 'GitHub Connect to GitHub', and 'Container Registry Use Heroku CLI'. The 'GitHub Connect to GitHub' button is highlighted. Below this, there is a search bar with the text 'Search for a repository to connect to'. The search bar contains the text 'yousefhosam-b' and 'repo-name', with a 'Search' button. Below the search bar, there is a link: 'Missing a GitHub organization? [Ensure Heroku Dashboard has team access.](#)'

It should look like this after connecting:

The screenshot shows the Heroku dashboard after connecting to GitHub. The 'GitHub Connect to GitHub' button is now 'Connected' with a green checkmark. Below this, there is a section titled 'Connected to' with the text 'yousefhosam-b/Deployment_on_Heroku by yousefhosam-b'. To the right of this text is a 'Disconnect...' button. Below this section, there is a link: 'Releases in the [activity feed](#) link to GitHub to view commit diffs'.

Deploying the Model:

Click “Deploy Branch”

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

Deploying:

Choose a branch to deploy

main

Deploy Branch

Receive code from GitHub



Build main ad3586be



```
-----> No change in requirements detected, installing from cache
-----> Using cached install of python-3.9.6
-----> Installing pip 20.2.4, setuptools 47.1.1 and wheel 0.36.2
-----> Installing SQLite3
-----> Installing requirements with pip
-----> Discovering process types
      Procfile declares types -> web
-----> Compressing...
```

☒ Autoscroll with output

[View build log](#)

Release phase

Deploy to Heroku

It should look like this when it finishes:

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 ad3586be



Release phase



Deploy to Heroku



Your app was successfully deployed.

 View

Deployment Result:


















I had similar same result as the deployment on Flask because I used same model.

The screenshot shows a web browser window with the URL `diabetes-data-glacier.herokuapp.com`. The page title is "Yousef Elbayoumi Description about the Data". The main heading is "Anticipating Diabetes With AI" in red. Below it, a subtitle reads "This Machine Learning model will help you finding if you have Diabetes or not". A prompt "Enter the values for anticipating Diabetes" is followed by a list of input fields: Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, DiabetesPedigreeFunction, and Age. A blue button labeled "Show the Result" is highlighted with a red box. Below the button, the placeholder text `{{ prediction_text }}` is visible. Social media icons for GitHub, LinkedIn, Instagram, and Twitter are at the bottom.

Adding some Config Vars:

Config Vars

Hide Config Vars

Age	50	 
BloodPressure	72	 
BMI	33.6	 
DiabetesPedigreeFunction	0.627	 
Glucose	148	 
Insulin	0	 
Pregnancies	6	 
SkinThickness	35	 
KEY	VALUE	

Review:

Can everything be that perfect? Without any errors? Surely no, I wasn't able to deploy the model for couple of reasons, I will share them in the following figures.

The screenshot displays the Heroku deployment history for the user `yousefxelbayomi@gmail.com`. The history consists of eight entries:

- Deployed** (ad3586be): Yesterday at 11:23 PM · v3 · [Roll back to here](#)
- Build succeeded**: Yesterday at 11:22 PM · [View build log](#)
- Build failed**: Yesterday at 11:20 PM · [View build log](#)
- Build failed**: Yesterday at 10:56 PM · [View build log](#)
- Build failed**: Yesterday at 10:44 PM · [View build log](#)
- Build failed**: Yesterday at 10:35 PM · [View build log](#)
- Build failed**: Yesterday at 10:33 PM · [View build log](#)
- Enable Logplex**: Jul 7 at 9:59 PM · v2 · [Roll back to here](#)

GitHub Repository:

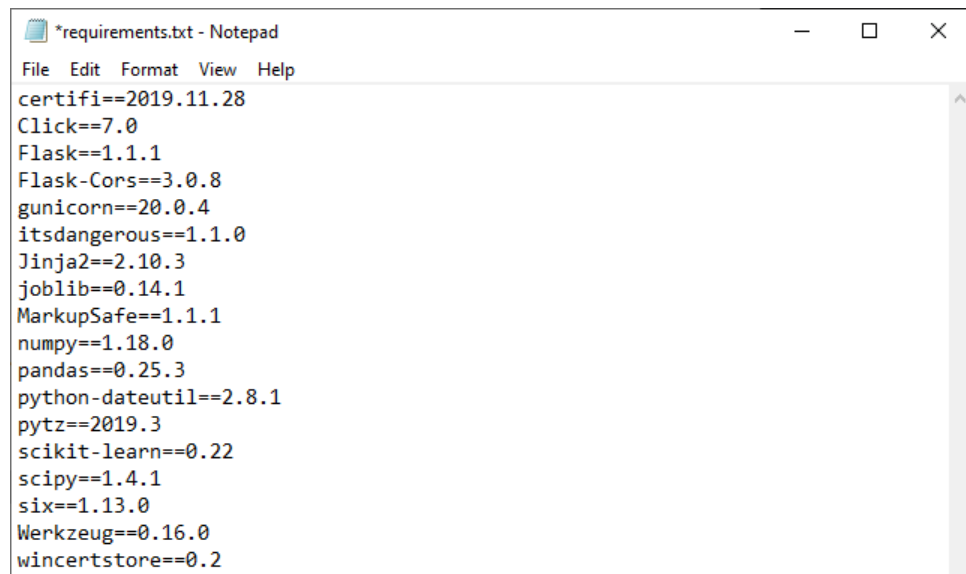
In the first deploy, I created Task5 folder in my DataGlacier_Internship repository and pushed all required files into it, you can check it below. However, that gave error, and I solved it by creating new repository which is “Deployment_on_Heroku” and pushed all files into it directly without creating inner folders, I shared figure of it in previous figures you can check it.

The screenshot shows the GitHub repository `yousefhosam-b / DataGlacier_Internship`. The commit history for the `Task5` folder is displayed, showing a single commit by `yousefhosam-b` titled "Added Required Files for Deployment on Heroku" (commit hash `8bda441`, 23 hours ago).

File	Commit Message	Time
..	Added Required Files for Deployment on Heroku	23 hours ago
Procfile	Added Required Files for Deployment on Heroku	23 hours ago
app.py	Added Required Files for Deployment on Heroku	23 hours ago
requirements.txt	Added Required Files for Deployment on Heroku	23 hours ago

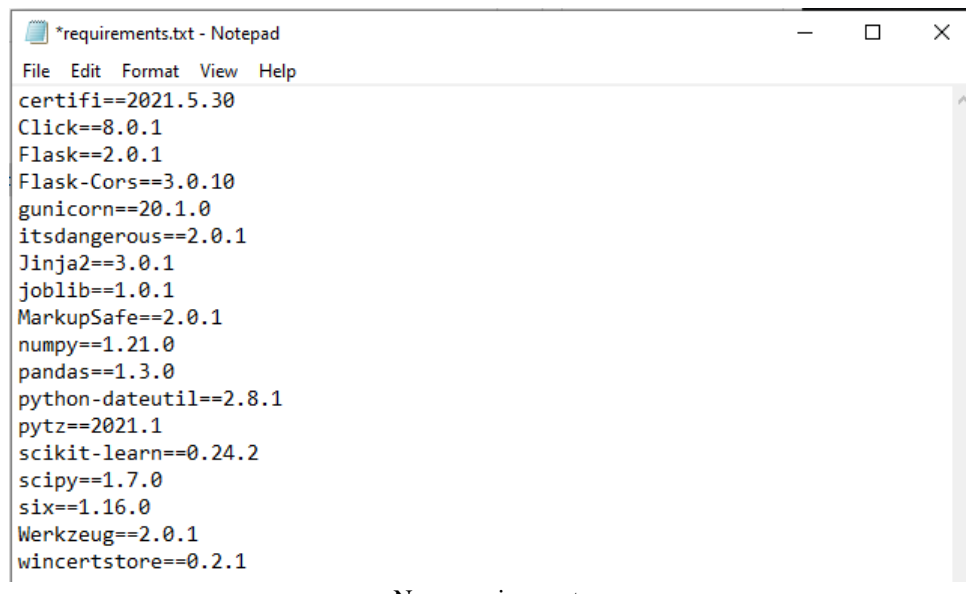
Requirements Text Document:

I used these versions which gave me an error, after searching I learned about the latest versions and used them to solve the problem, down below the first figure is the old requirement.txt and the second figure is the new one, you can check the difference in the versions.

A screenshot of a Notepad window titled '*requirements.txt - Notepad'. The window contains a list of package versions for a requirements file. The packages and their versions are: certifi==2019.11.28, Click==7.0, Flask==1.1.1, Flask-Cors==3.0.8, gunicorn==20.0.4, itsdangerous==1.1.0, Jinja2==2.10.3, joblib==0.14.1, MarkupSafe==1.1.1, numpy==1.18.0, pandas==0.25.3, python-dateutil==2.8.1, pytz==2019.3, scikit-learn==0.22, scipy==1.4.1, six==1.13.0, Werkzeug==0.16.0, and wincertstore==0.2.

```
*requirements.txt - Notepad
File Edit Format View Help
certifi==2019.11.28
Click==7.0
Flask==1.1.1
Flask-Cors==3.0.8
gunicorn==20.0.4
itsdangerous==1.1.0
Jinja2==2.10.3
joblib==0.14.1
MarkupSafe==1.1.1
numpy==1.18.0
pandas==0.25.3
python-dateutil==2.8.1
pytz==2019.3
scikit-learn==0.22
scipy==1.4.1
six==1.13.0
Werkzeug==0.16.0
wincertstore==0.2
```

Old requirements

A screenshot of a Notepad window titled '*requirements.txt - Notepad'. The window contains a list of updated package versions for a requirements file. The packages and their versions are: certifi==2021.5.30, Click==8.0.1, Flask==2.0.1, Flask-Cors==3.0.10, gunicorn==20.1.0, itsdangerous==2.0.1, Jinja2==3.0.1, joblib==1.0.1, MarkupSafe==2.0.1, numpy==1.21.0, pandas==1.3.0, python-dateutil==2.8.1, pytz==2021.1, scikit-learn==0.24.2, scipy==1.7.0, six==1.16.0, Werkzeug==2.0.1, and wincertstore==0.2.1.

```
*requirements.txt - Notepad
File Edit Format View Help
certifi==2021.5.30
Click==8.0.1
Flask==2.0.1
Flask-Cors==3.0.10
gunicorn==20.1.0
itsdangerous==2.0.1
Jinja2==3.0.1
joblib==1.0.1
MarkupSafe==2.0.1
numpy==1.21.0
pandas==1.3.0
python-dateutil==2.8.1
pytz==2021.1
scikit-learn==0.24.2
scipy==1.7.0
six==1.16.0
Werkzeug==2.0.1
wincertstore==0.2.1
```

New requirements

One more thing:

In the previous two figures, last requirement in both of them is “wincertstore”, and 0.2.1 is the latest version of it. However, this version gave error while deploying the model, and when I used the previous version which is 0.2 the deployment worked.

This error wasn’t expected to be honest but at the end I managed to deploy the model.

That was all from me, thanks for reading.