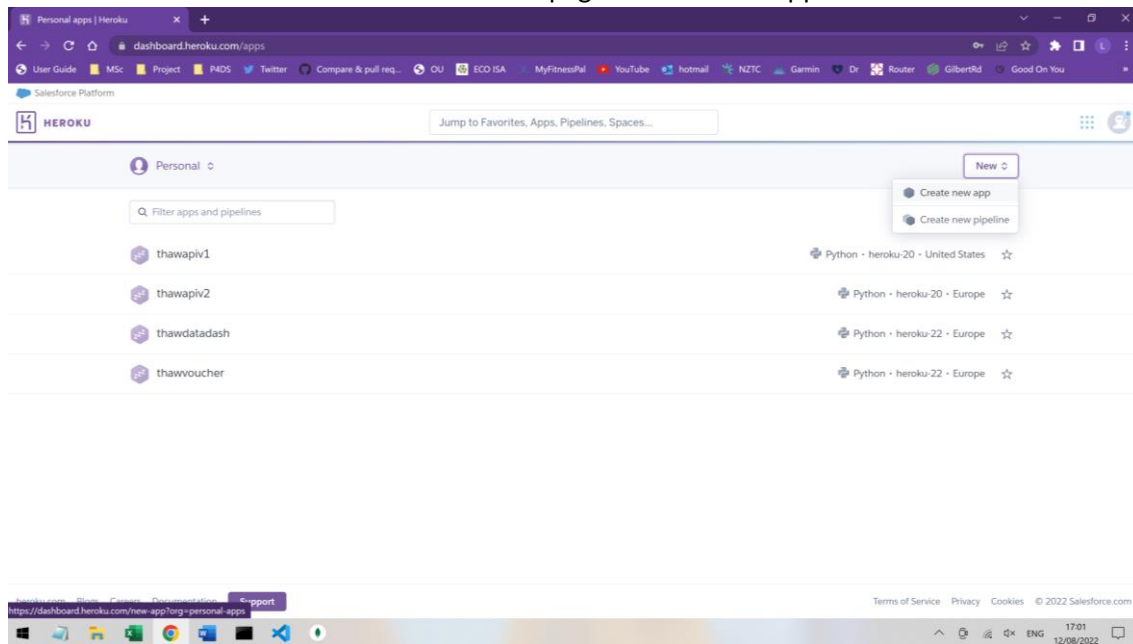# Hosting and Deploying the API on Heroku
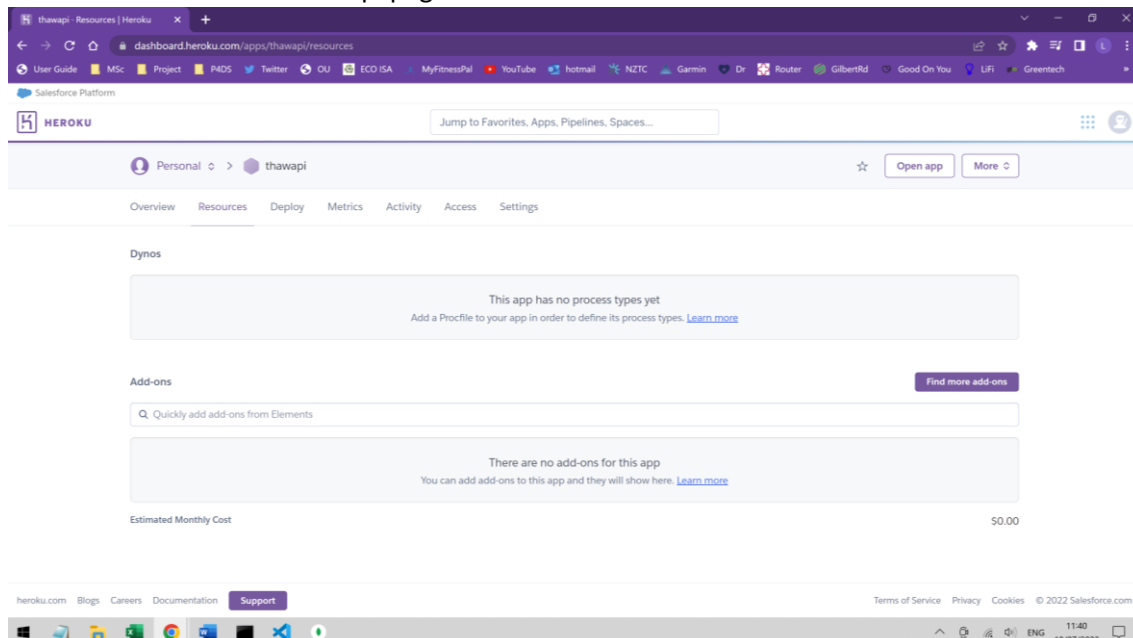
## 1. Via the Website

First attempt for deploying the Flask app used used the same method for creating app as used for the Streamlit app.  For deploying Streamlit follow these instructions.
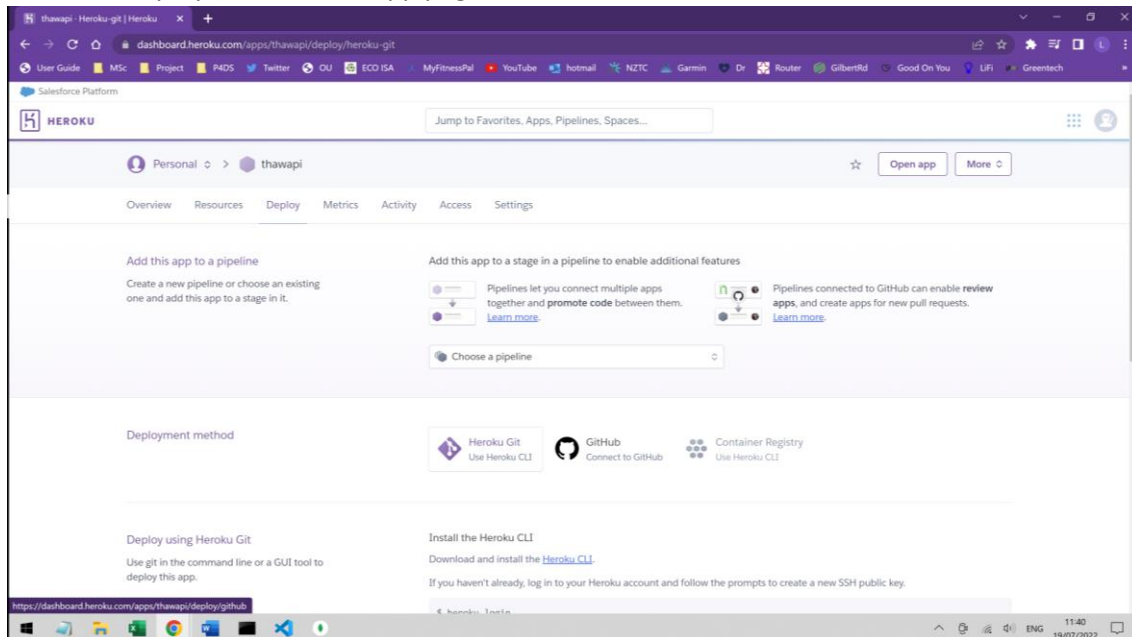
Create a Heroku account and from the home page create a new app:
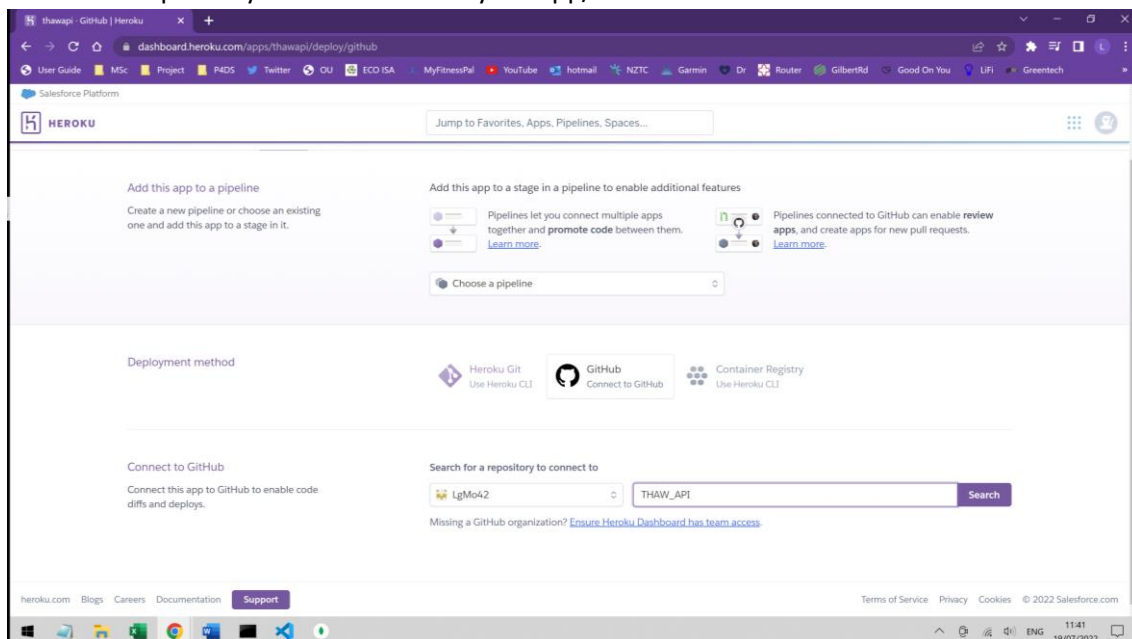
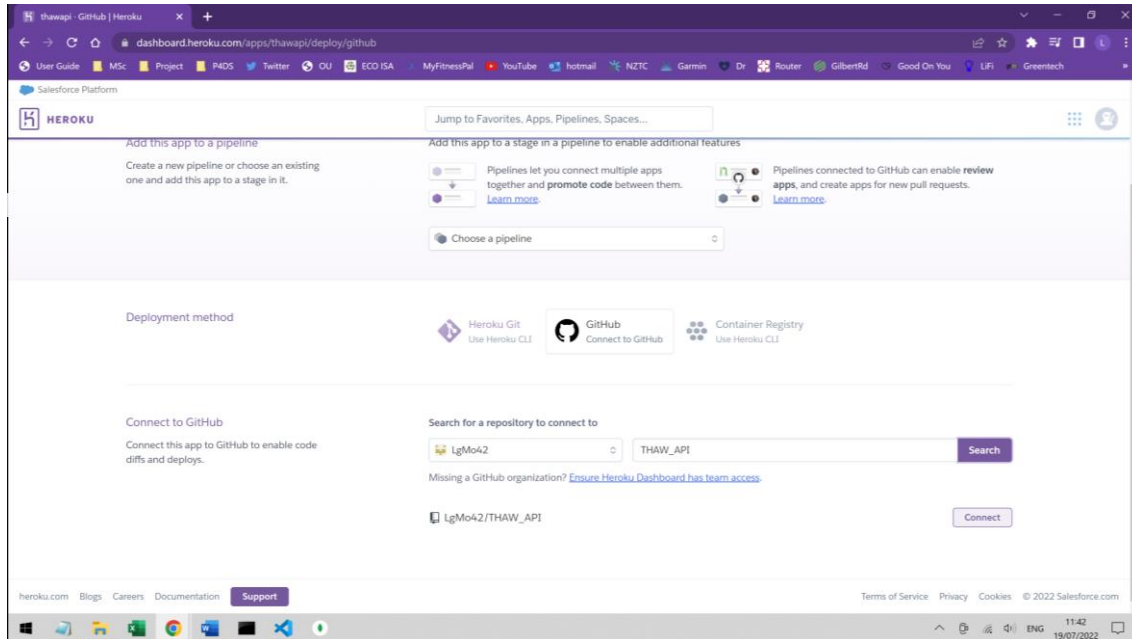

Here is shown the new thawapi page:

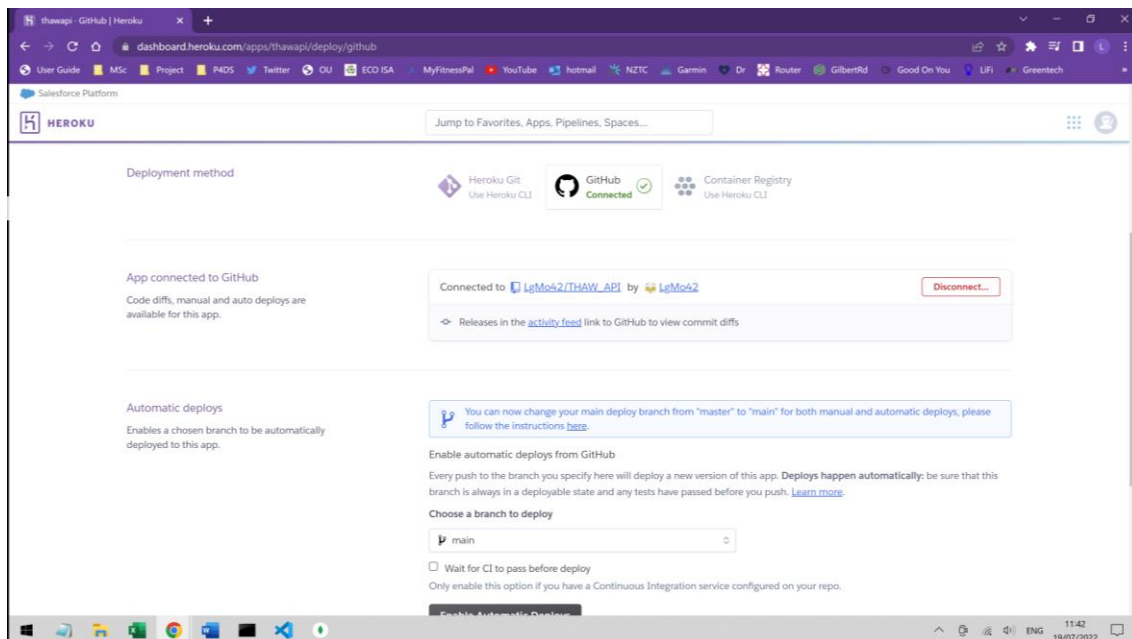From the Deploy menu on the app page connected to GitHub:



Search for repository with the code for your app/API:

Connect to the repository:



Connected!

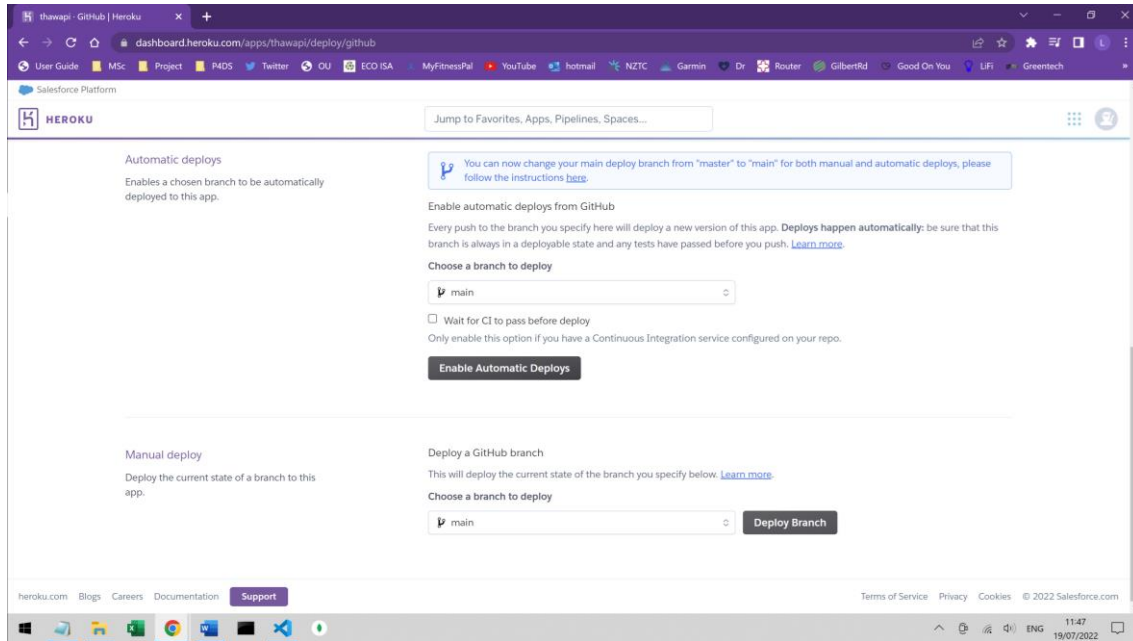From Setting page reveal config vars:

Enter the config var details for connection to MongoDB Atlas THAW database:
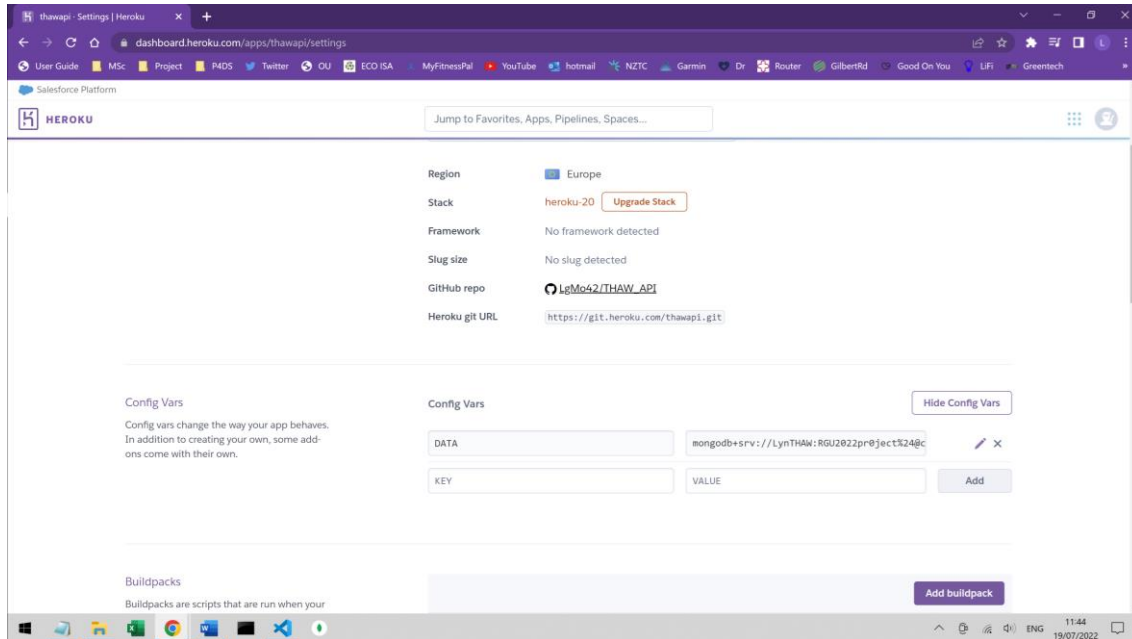


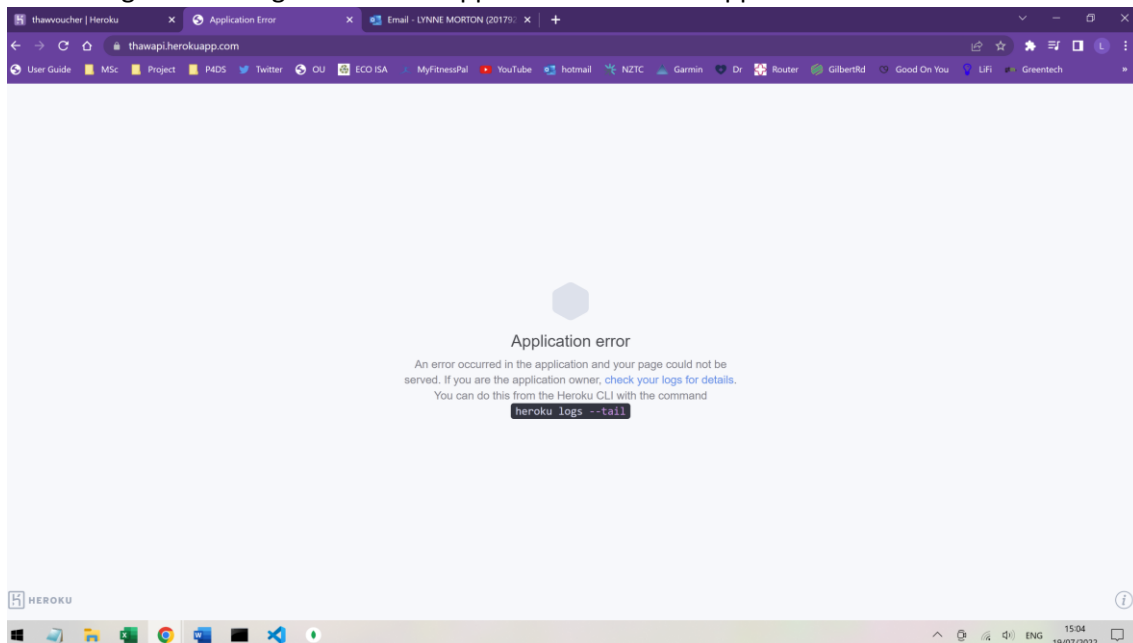If you click on edit you can see the full link:
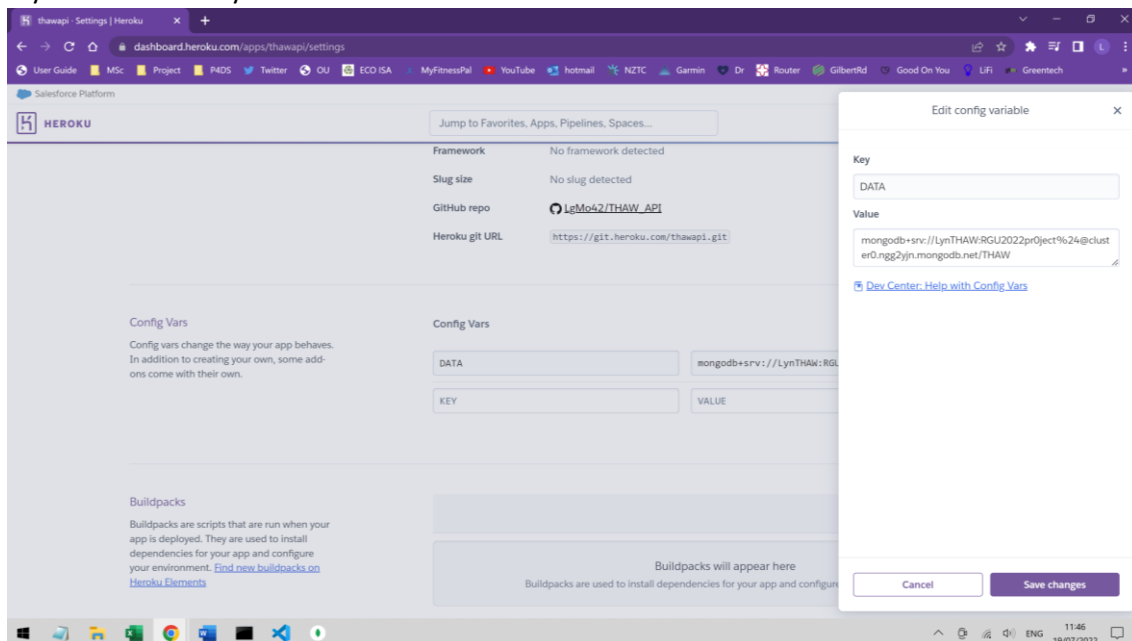
On the Deploy page click deploy branch:



Here branch is main and here are no other options. When developing the app in another branch can be deployed from to test it before merging the new branch with the main and overwriting a working version.

Something went wrong for the Flask app. For the streamlit app the above works.



## 2. Via Heroku Cli

A search online suggested flask apps need to be uploaded using the Heroku Cli. Once downloaded and installed the command prompt was used to create and deploy the flask app successfully. This was done by following the tutorial on https://www.youtube.com/watch?v=SiCAIRc0pEI
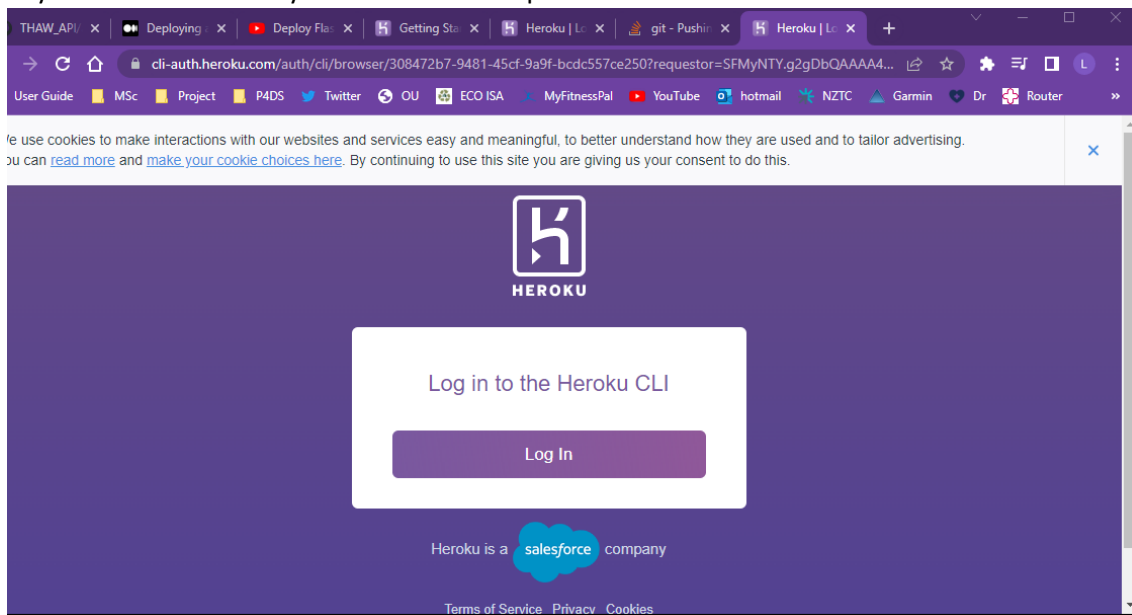
Open command prompt from folder that app is in or navigate to it via cd command.

Type: heroku login



This will open a page in the browser. If logged in already you will see the screen below. If not logged in you will be asked for your username and password.



After logging in create the Heroku app by typing: heroku create
(To set app name as thawapiv2 and region as eu type: heroku create thawapiv2 --region=eu  )

To check Heroku app has been created type: git remote -v
This will show the Heroku apps http has been created and the repository it is linked to.

To deploy the code form the main branch type: git push heroku main
This will push the code  from the main branch of the repository listed to the heroku app just created.
(To deploy from a branch type: git push heroku branchname:main)

Once deployed there is a verify message and the heroku link will be active.

If you push the code without changes then the app will not redeploy and a message saying all is up to date will be output instead.

To remove the Heroku app type: git remote rm heroku

To access an existing app type: heroku git:remote -a appname

Success!