

Deploy de aplicación de prueba a Heroku

1) Como prerequisites, es necesario crearse una cuenta en **GitHub** y **Heroku**.

Posteriormente a eso, hay que descargar e instalar:

Git: <https://git-scm.com/downloads>

Heroku CLI: <https://devcenter.heroku.com/articles/heroku-cli>

2) Loguearse en Heroku y en GitHub.

3) Clonar el repositorio de prueba: <https://github.com/heroku/python-getting-started> desde el terminal utilizando:

git clone <https://github.com/heroku/python-getting-started.git>

4) Es posible que en esta instancia nos pida loguearnos con **GitHub** en el navegador, si es así, hay que hacerlo.

5) Ingresar a la carpeta del repositorio clonada en nuestra computadora con:

cd python-getting-started

6) Estando en esa carpeta dentro del terminal, crear la aplicación en Heroku y pushear los archivos del repositorio local al servidor. Es posible que en este momento nos pida loguearnos con Heroku en el navegador.

heroku create

git push heroku main

7) Al finalizar nos indicará el link de nuestra aplicación en el servidor de Heroku, podemos copiar el enlace y pegarlo en el navegador o ingresar directamente con **heroku open**

Pushear modificaciones de la aplicación al servidor de Heroku

1) Para replicar en el servidor posibles cambios que se hagan sobre los archivos hay que estar posicionado con el terminal en la carpeta **python-getting-started**.

2) Desde allí, agregar todos los cambios a la staging area: **git add** .

Hacer un commit de los cambios: **git commit -m "Comentario del commit"**

Y volver a pushear los cambios al servidor: **git push heroku main**

Consideraciones

No pasar la aplicación a producción antes de modificar las siguientes opciones en settings.py

(settings.py)

SECURITY WARNING: Esta clave no debe ser revelada

SECRET_KEY = "Utilizar una secret key que se cargue como variable de entorno o desde otro archivo"

SECURITY WARNING: Cambiar a False cuando se pase la aplicación a producción

DEBUG = True

Para más información consultar: <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Deployment>

Base de Datos - PostgreSQL

Este ejemplo ya incluye una base de datos que se configura automáticamente en Heroku con el único plan gratuito que provee la plataforma. Para corroborar que está instalada podemos ejecutar `heroku addons`

```
Add-on      Plan      Price  State
heroku-postgresql (postgresql-transparent-04858) hobby-dev free   created
└─ as DATABASE

The table above shows add-ons and the attachments to the current app (still-falls-82900) or other apps.
```

Dentro de nuestra cuenta en Heroku podemos ver las credenciales de la base de datos:

Resources

Heroku Postgres

Attached as DATABASE

Hobby Dev

Free

Datastores

postgresql-transparent-04858

SERVICE

heroku-postgresql

PLAN

hobby-dev

BILLING APP

still-falls-82900

Overview

Durability

Settings

Dataclips

HEALTH

Available

PRIMARY

Yes

VERSION

13.5

CREATED

3 minutes ago

MAINTENANCE

Unsupported

ROLLBACK

Unsupported

UTILIZATION

0 of 20

0 of 10,000

7.9 MB

0

CONNECTIONS

ROWS

DATA SIZE

TABLES

ADMINISTRATION

Database Credentials

Get credentials for manual connections to this database.

View Credentials...

Reset Database

Reset the database to its originally-provisioned state, deleting all data inside it.

Reset Database...

Destroy Database

Destroys the database and all of the data inside it.

Destroy Database...