

## **GUÍA DE DESPLIEGUE**

### **ESTUDIANTES:**

Andrés Julián Álzate Díaz  
Cristian Alejandro Vasquez Zuluaga

### **ASESOR:**

Jorge Eliecer Giraldo Plaza

**PROGRAMA DE INGENIERÍA INFORMÁTICA  
ÁREA DE PROGRAMAS INFORMÁTICOS Y TELECOMUNICACIONES (APIT)  
FACULTAD DE INGENIERÍAS**

**POLITÉCNICO COLOMBIANO  
JAIME ISAZA CADAVID**

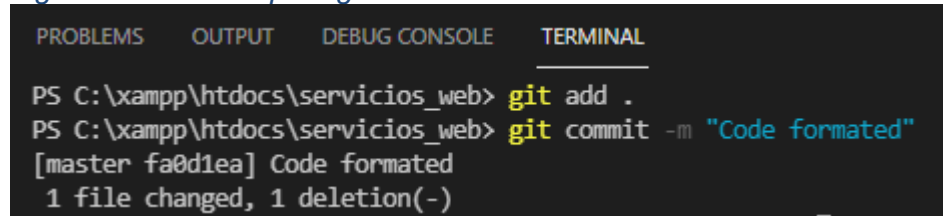
Medellín, febrero 7 del 2022

## Guía de despliegue

A continuación, se presenta el paso a paso del despliegue tanto del proveedor de servicios, como del portal web utilizando Heroku con git. En ambos casos se repiten los mismos pasos ya que ambos están desplegados en Heroku lo único que cambia es el acceso remoto que para los servicios sería heroku y para el portal sería portal-heroku

Primero se deben guardar los cambios que se hicieron en el proyecto en un commit como se muestra en la Figura 1.

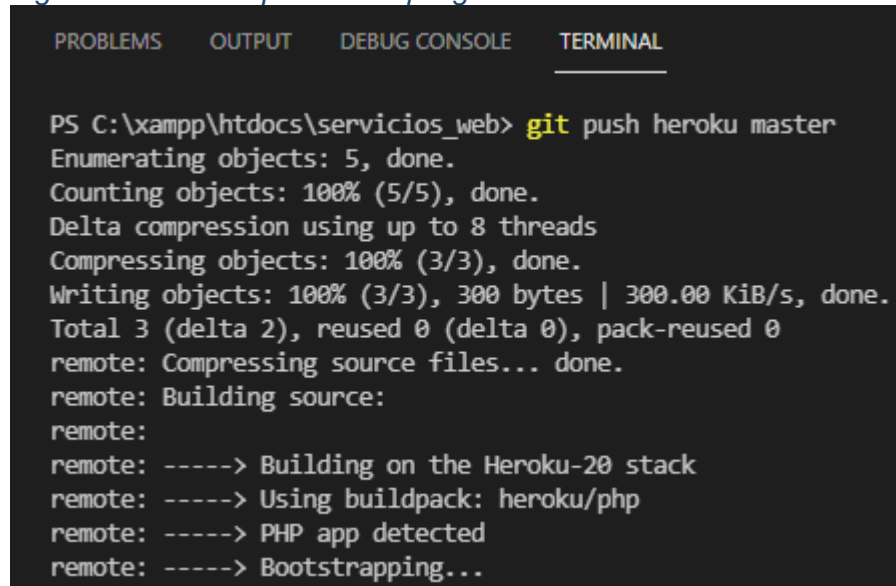
*Figura 1 comandos para guardar en un commit*

A screenshot of a terminal window with a dark background. At the top, there are tabs labeled 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', and 'TERMINAL', with 'TERMINAL' being the active tab. The terminal shows the following commands and output:

```
PS C:\xampp\htdocs\servicios_web> git add .
PS C:\xampp\htdocs\servicios_web> git commit -m "Code formatted"
[master fa0d1ea] Code formatted
1 file changed, 1 deletion(-)
```

Para hacer el despliegue de la aplicación en Heroku, se utiliza el comando git push para enviar el código desde la rama principal en el repositorio local al control remoto **heroku** o para el portal web **portal-heroku**.

*Figura 2 Comando para el despliegue de los servicios*

A screenshot of a terminal window with a dark background. At the top, there are tabs labeled 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', and 'TERMINAL', with 'TERMINAL' being the active tab. The terminal shows the output of the command 'git push heroku master':

```
PS C:\xampp\htdocs\servicios_web> git push heroku master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 300 bytes | 300.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
remote:
remote: -----> Building on the Heroku-20 stack
remote: -----> Using buildpack: heroku/php
remote: -----> PHP app detected
remote: -----> Bootstrapping...
```

*Figura 3 Comando para el despliegue del portal web*

```
PS C:\Users\57301\Documents\projects\react\soap-client> git push portal-heroku master
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 430 bytes | 215.00 KiB/s, done.
Total 5 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Compressing source files... done.
remote: Building source:
```

Después de finalizado el despliegue, se mostrará por consola un mensaje de éxito y el dominio de nuestra aplicación.

*Figura 4 Mensaje de éxito y dominio de la aplicación*

```
remote: Verifying deploy... done.
To https://git.heroku.com/frozen-river-98217.git
 2854792..fa0d1ea master -> master
PS C:\xampp\htdocs\servicios_web> █
```