



Despliegue de proyecto en Heroku

A continuación, se desplegará un ambiente de producción de un proyecto sencillo en Heroku.

Pasos:

1. Crear un proyecto en REACT, de la forma:

```
npx create-react-app proyecto
```

2. Crear una cuenta en Heroku

<https://www.heroku.com>

3. Las formas de crear un proyecto en Heroku son:
 - Desde consola (CMD)
 - Desde la web de Heroku
 - Desde un repo en github

Crear una app desde consola y con la CLI de Heroku

Una vez creado el proyecto, lo abrimos desde vscode y modificamos cualquier línea de código para establecer cambios en el proyecto. En este ejercicio, modifiquemos el mensaje que nos muestra nuestro proyecto simple hecho con REACT en el archivo "App.js", de la forma:



Código original:

```
1 import logo from './logo.svg';
2 import './App.css';
3
4 function App() {
5   return (
6     <div className="App">
7       <header className="App-header">
8         <img src={logo} className="App-logo" alt="logo" />
9         <p>
10           Edit <code>src/App.js</code> and save to reload.
11         </p>
12         <a
13           className="App-link"
14           href="https://reactjs.org"
15           target="_blank"
16           rel="noopener noreferrer"
17         >
18           Learn React
19         </a>
20       </header>
21     </div>
22   );
23 }
24
25 export default App;
```

Ahora, modificamos el mensaje que muestra en la app (línea 10 y eliminamos la etiqueta completa <a>), de la forma:

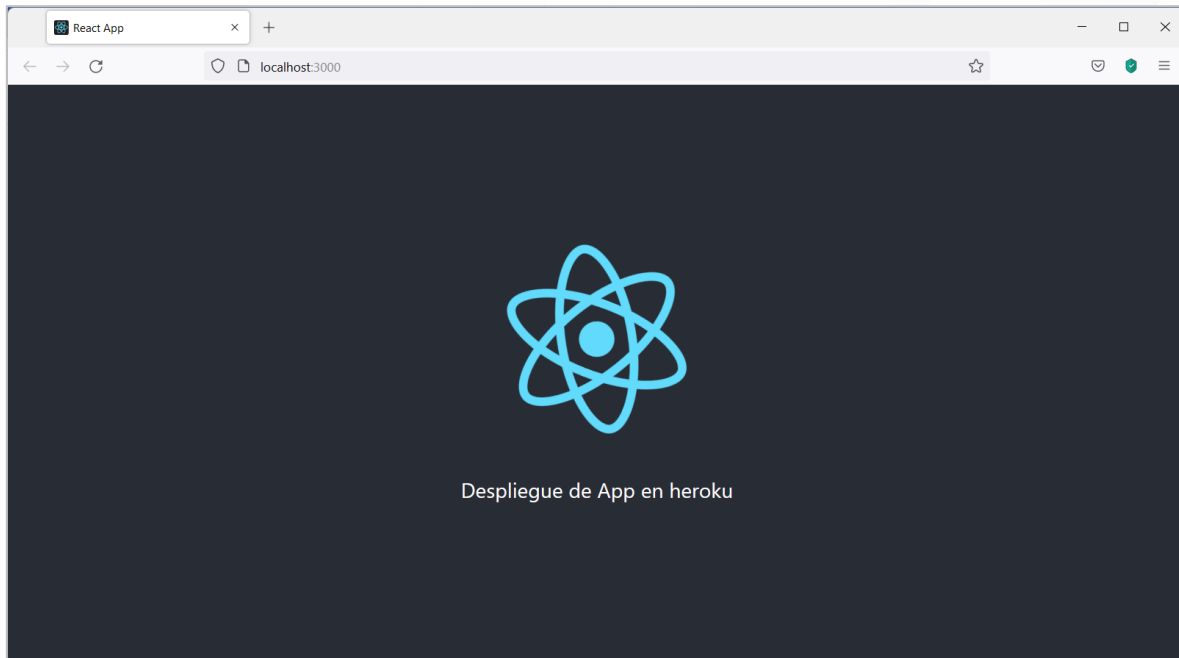
```
1 import logo from './logo.svg';
2 import './App.css';
3
4 function App() {
5   return (
6     <div className="App">
7       <header className="App-header">
8         <img src={logo} className="App-logo" alt="logo" />
9         <p>Despliegue de App en heroku</p>
10       </header>
11     </div>
12   );
13 }
14
15 export default App;
```

Ahora, abrimos la app creada ejecutando el siguiente comando desde la consola de vscode, de la forma:

```
npm start
```



Observemos los cambios hechos en nuestra app.



Ahora procedemos a instalar la CLI (Command Line Interface) de Heroku. Esta CLI se puede descargar desde la siguiente página web:

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

Una vez instalada la CLI procedemos a verificar que esta correctamente instalada. Para ello, ejecutamos el siguiente comando desde la consola de vscode, de la forma:

```
heroku -v
```

Iniciar nuestro proceso de despliegue

Para iniciar nuestro proceso de despliegue, ejecutamos los siguientes comandos desde la consola de vscode, de la forma:

Autenticarse en Heroku

```
heroku login
```

Ejemplo

```
D:\proyecto> heroku login
```



Seguidamente, ejecutamos el siguiente comando de consola para verificar el estado de actualizaciones sin agregar a nuestro repo.

```
git status
```

Ejemplo

```
D:\proyecto> git status
```

Esto nos permitirá observar que tenemos un cambio hecho en el archivo App.js. Seguidamente, agregamos nuestros cambios para que sean subidos a nuestro repo, de la forma:

```
git add .
```

Ejemplo

```
D:\proyecto> git add .
```

Una vez agregados nuestros cambios, procedemos a hacer un commit de la forma:

```
git commit -m "mensaje"
```

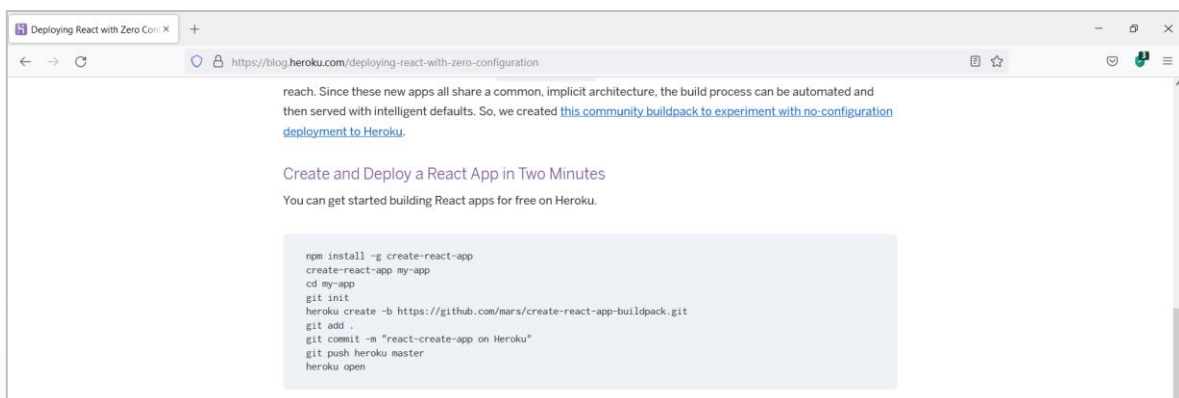
Ejemplo

```
D:\proyecto> git commit -m "Mi primer commit de ejemplo"
```

Ahora, nos vamos al siguiente enlace de Heroku donde se muestra un despliegue de ejemplo que sugiere Heroku:

<https://blog.heroku.com/deploying-react-with-zero-configuration>

En este enlace conseguiremos las instrucciones básicas de despliegue tal y como se muestran en la siguiente imagen:





Una vez identificado la lista de comandos de ejemplo, seleccionamos la línea #5 y la pegamos en nuestra consola. Adicional a este evento, vamos a colocarle al final un nombre a nuestro proyecto (despliegueproyecto), de esta forma, Heroku no va a crear un nombre al azar; seremos nosotros quien le demos un nombre a nuestro despliegue

```
heroku create -b https://github.com/mars/create-react-app-buildpack.git despliegueproyecto
```

Ejemplo

```
D:\proyecto> heroku create -b https://github.com/mars/create-react-app-buildpack.git despliegueproyecto
```

Conocer la ruta de nuestro proyecto en Heroku

Para conocer cual es la ruta de nuestro proyecto, ejecutemos el siguiente comando de consola en vscode, de la forma:

```
git remote -v
```

Ejemplo

```
D:\proyecto> git remote -v
```

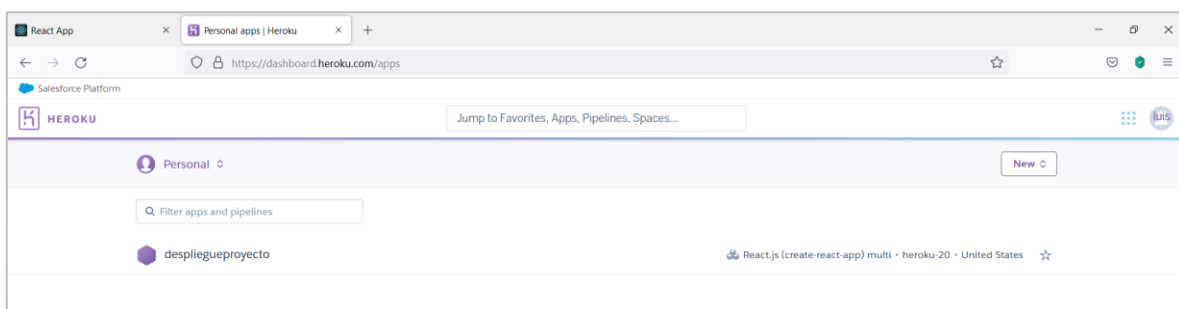
Una vez revisado nuestras direcciones en Heroku, ejecutemos el siguiente comando de consola en vscode para hacer un push a Heroku, de la forma:

```
git push heroku master
```

Ejemplo

```
D:\proyecto> git push heroku master
```

A continuación, en la página de Heroku veremos nuestro proyecto montado





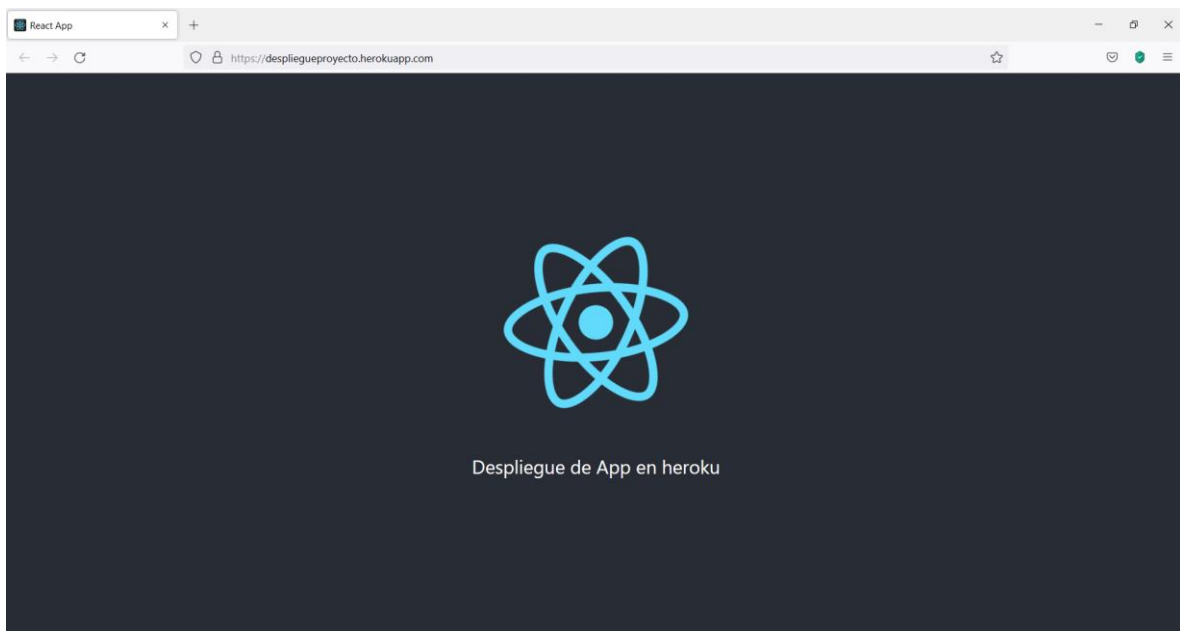
Abrir nuestro Proyecto en Heroku

Finalmente, para abrir nuestro proyecto en Heroku ejecutamos el siguiente comando de consola, de la forma:

```
Heroku open
```

Ejemplo

```
D:\proyecto> Heroku open
```



Remover nuestro Proyecto de Heroku

Para remover nuestro proyecto de Heroku por cualquier motivo, ejecutamos el siguiente comando de consola, de la forma:

```
git remote remove Heroku
```

Ejemplo

```
D:\proyecto> git remote remove Heroku
```



Actualizar cambios en Heroku luego de una actualización de código

Para actualizar nuestro cambios en Heroku luego de una actualización de código, ejecutar el siguiente comando de consola, de la forma:

```
git push heroku master
```

Ejemplo

```
D:\proyecto> git push heroku master
```

Visitar vídeo

<https://www.youtube.com/watch?v=gItqzShex5M>