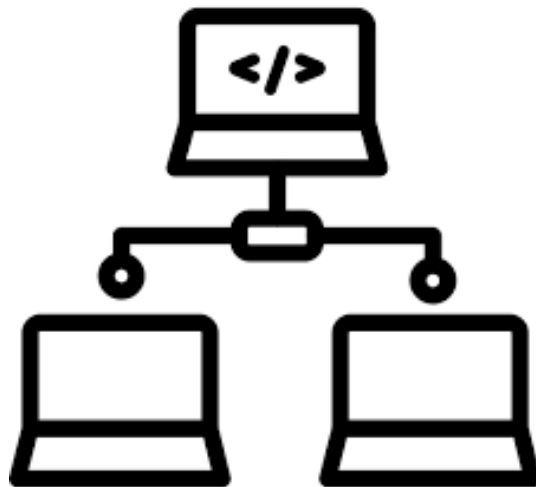


Manual de Despliegue



Sport Planet – TFG DAW 23-24

Manuel Jesús Terrón Girón

ÍNDICE

DESPLIEGUE	3
1. ANGULAR	3
2. EXPRESS	4

DESPLIEGUE

1. ANGULAR

- En angular lo primero que hice fue instalar lo siguiente

`npm install -g angular-cli-ghpages`

- Mediante este comando conseguí instalar lo necesario para desplegar mi frontend en github Pages, posterior a eso añadi este comando para comprimir todas mis páginas de angular en un directorio llamado “dist”, que se usará para ejecutar mi página en github Pages

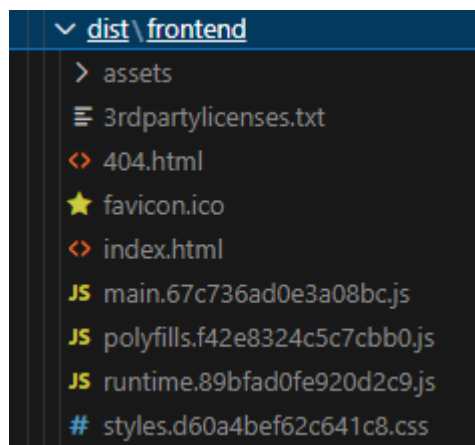
```
PS C:\Users\ManuF\OneDrive\Documentos\GitHub\Proyecto\Proyecto_Sport_Planet\src\frontend> ng build
✓ Browser application bundle generation complete.
✓ Copying assets complete.
* Generating index html...2 rules skipped due to selector errors:
  .form-floating~label -> Did not expect successive traversals.
  .form-floating~label -> Did not expect successive traversals.
✓ Index html generation complete.

Initial Chunk Files | Names | Raw Size | Estimated Transfer Size
styles.d60a4bef62c641c8.css | styles | 219.08 kB | 21.80 kB
polyfills.f42e8324c5c7cbb0.js | polyfills | 33.02 kB | 10.64 kB
runtime.89bfad0fe920d2c9.js | runtime | 894 bytes | 518 bytes
```

- Luego de eso ya estaría mi aplicación comprimida y lista para ser lanzada, solo que tendría que lanzar el siguiente comando para que automaticamente se creara una branch nueva llamada “gh-pages” en mi proyecto con el contenido de dist

```
PS C:\Users\ManuF\OneDrive\Documentos\GitHub\Proyecto\Proyecto_Sport_Planet\src\frontend> npx ngh --dir=dist/frontend --no-silent
📶 Uploading via git, please wait...
🚀 Successfully published via angular-cli-ghpages! Have a nice day!
PS C:\Users\ManuF\OneDrive\Documentos\GitHub\Proyecto\Proyecto_Sport_Planet\src\frontend> █
```

- Dejo una imagen de como se veia dist:



- Luego de eso solo tendria que habilitarlo en mi repositorio de github y decirle a github pages el branch que tiene que usar

Build and deployment

Source

Deploy from a branch ▾

Branch

Your GitHub Pages site is currently being built from the gh-pages branch. [Learn more about configuring the publishing source for your site.](#)

gh-pages ▾

/ (root) ▾

Save

Learn how to [add a Jekyll theme](#) to your site.

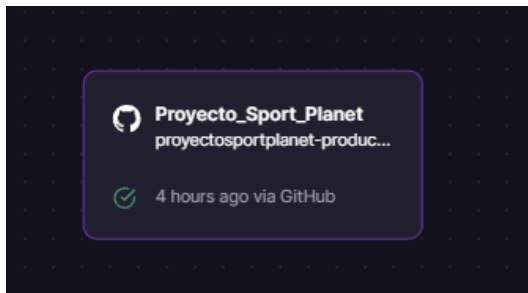
Your site was last deployed to the [github-pages](#) environment by the [pages build and deployment](#) workflow. [Learn more about deploying to GitHub Pages using custom workflows](#)

- Ya posterior a eso solo tendría que comprobar que fue lanzado (también hubo un problema con los enlaces, si añades un enlace así **/prueba** no lo pillaba por lo que tuve que cambiarlos todos y ponerles un punto delante **./prueba** o sin la barra **prueba**, algo muy difícil de localizar al inicio)



2. EXPRESS

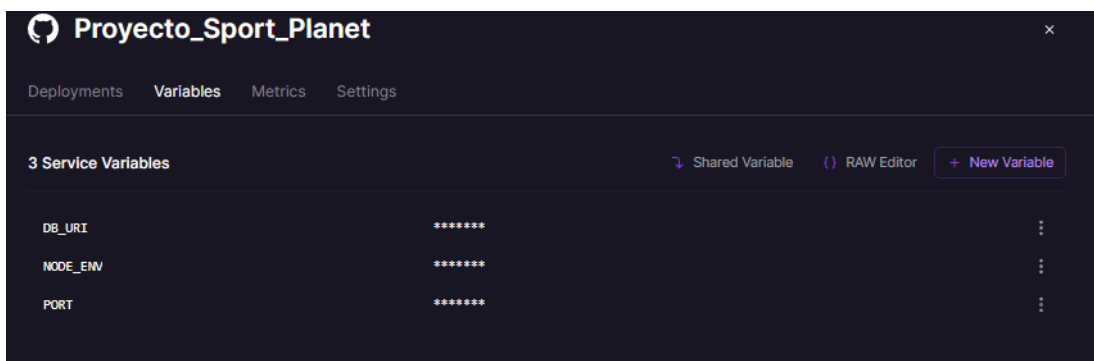
Para express use railway, al principio era algo difícil de entender pero a la que lees algo de documentación o investigas un poco acabas dando con la tecla, me registre con github y hice la importación de mi repositorio



Luego de eso entrar en settings y añadirle la ruta que tiene que seguir para entrar en la aplicación de express:

A dark-themed settings panel. The 'Root Directory' section has a description and a text input field containing '/src/backend/'. The 'Railway Config File Path' section has a description and a text input field containing 'railway.toml'.

luego de eso es importante pasarle las variables de entorno, en el apartado de variables



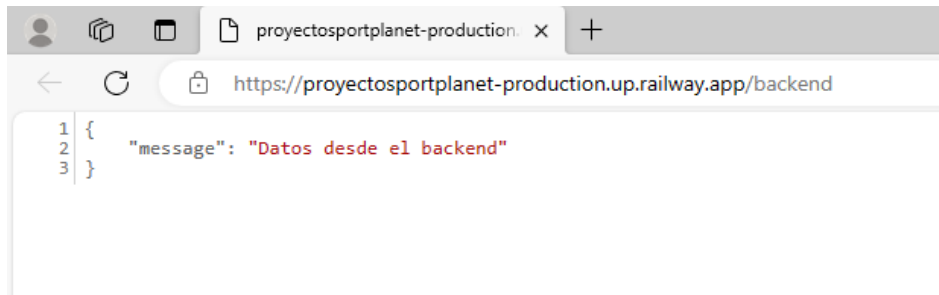
y posterior a eso railway se iniciará automáticamente cuando haya un push a github o cuando lo digamos si instalamos:

```
npm install -g railway
```

y posteriormente añadimos el siguiente comando

```
railway init
```

y nuestro backend estaría totalmente listo



Terrón Girón Manuel Jesús 2ºDAW