

Examen final, GitHub, DockerHub y Render

Estefanía Manjón González

Parte1:

Subir Archivo a Github

- Logeo en mi cuenta de GitHub
- Creo el repositorio

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*

Owner *

 EManjonG ▾

/

Repository name *

ExamenFinal

✔ ExamenFinal is available.

Great repository names are short and memorable. Need inspiration? How about [fluffy-dollop](#) ?

Description (optional)

☒

 **Public**

Anyone on the internet can see this repository. You choose who can commit.

☐

 **Private**

You choose who can see and commit to this repository.

Initialize this repository with:

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: **None** ▾

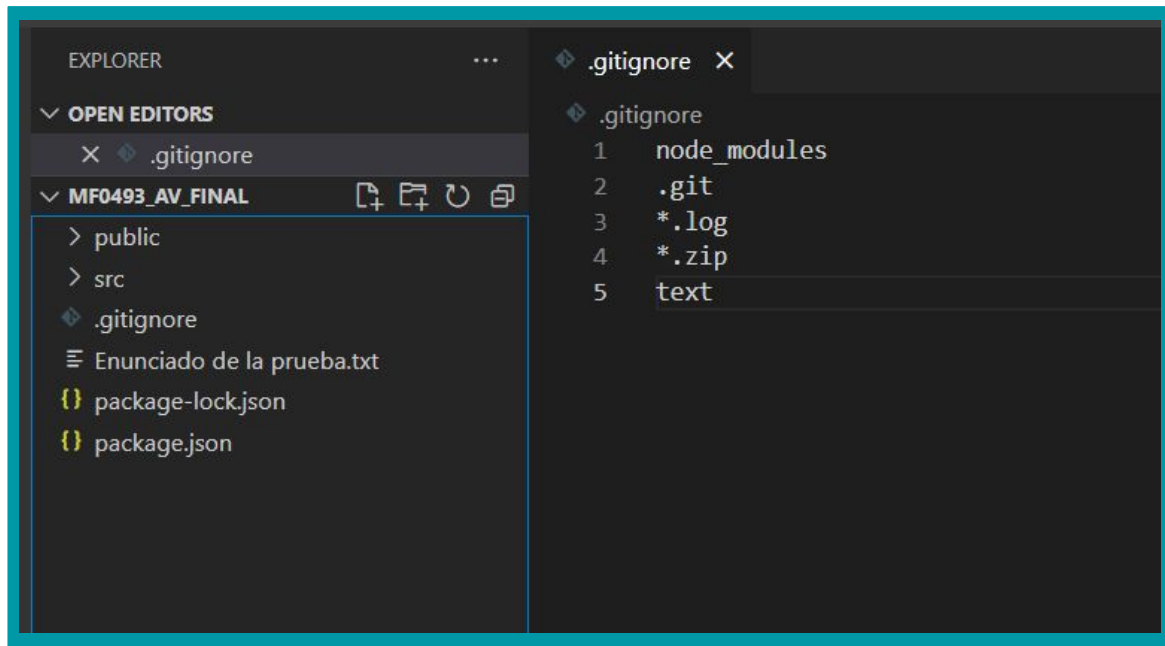
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

Parte1:

Subir Archivo a Github

- Creamos el .gitignore
- Añadimos los archivos que queremos que no se suban a git



Parte1:

Subir Archivo a Github

- Abrimos terminal bash y añadimos los comandos
- git init
- git add .
- git commit -m "el mensaje que sea"
- git branch -M main. Para cambiar master por main
- git remote add origin y ruta del repositorio
- git push -u origin main <- Lo subimos

```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final
● $ git init
Initialized empty Git repository in C:/Users/Aidet/Downloads/mf0493_av_final/.git/

Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (master)
● $ git add .
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'public/favicon.svg', LF will be replaced by CRLF the next time Git touches it
```

```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (master)
● $ git commit -m "Primer commit"
[master (root-commit) 205c9d6] Primer commit
12 files changed, 1523 insertions(+)
create mode 100644 .gitignore
create mode 100644 Enunciado de la prueba.txt
```

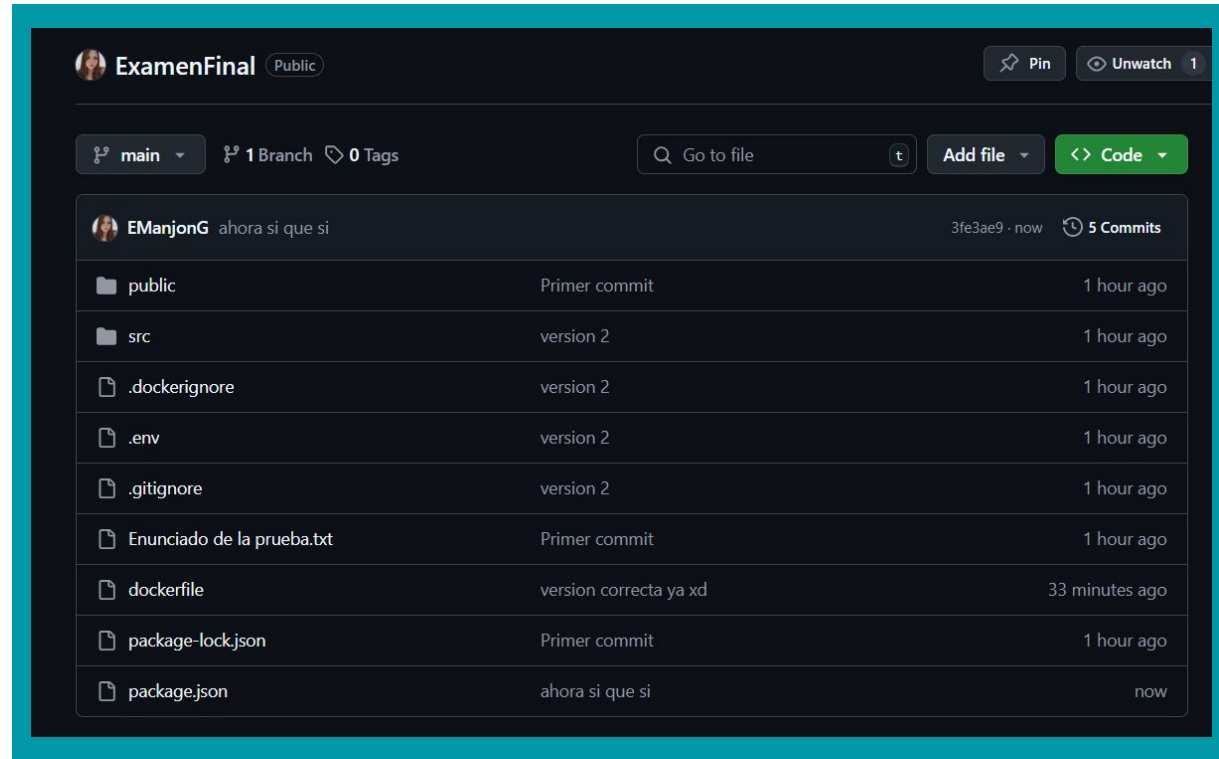
```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (master)
$ git branch -M main
● git remote add origin https://github.com/EManjonG/ExamenFinal.git
git push -u origin main
Enumerating objects: 16, done.
Counting objects: 100% (16/16), done.
```

Parte1:

Subir Archivo a Github

- Captura de pantalla con el repositorio final. El .text no se ha ignorado porque había olvidado poner “.text” en el .gitignore.

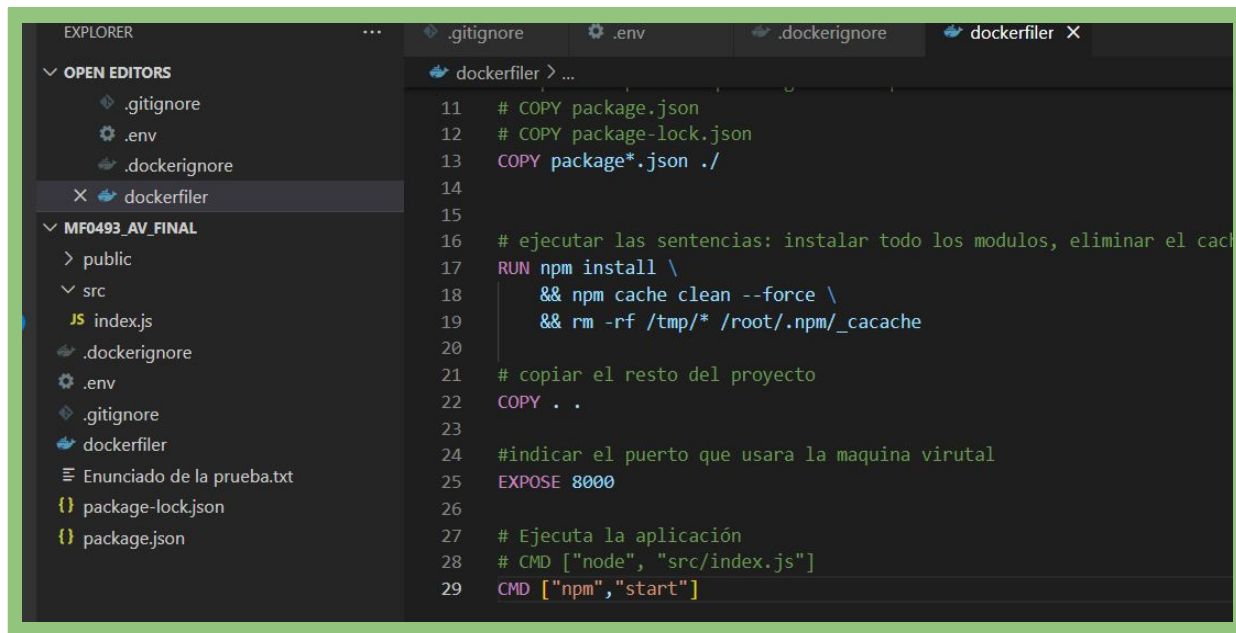
<https://github.com/EManjonG/ExamenFinal>



Parte2:

Docker y DockerHub

- Iniciamos DockerDesktop
- Creamos .env donde decimos el puerto
- .dockerignore (igual que .gitignore)
- y dockerfile que contiene la configuración
- Cambiamos el `--watch` del `package.json`



The screenshot shows a code editor with a file explorer on the left and a code editor on the right. The file explorer shows a project structure with files like `.gitignore`, `.env`, `.dockerignore`, `dockerfiler`, `Enunciado de la prueba.txt`, `package-lock.json`, and `package.json`. The code editor shows the content of the `dockerfiler` file, which is a Dockerfile. The Dockerfile contains the following instructions:

```
11 # COPY package.json
12 # COPY package-lock.json
13 COPY package*.json ./
14
15
16 # ejecutar las sentencias: instalar todo los modulos, eliminar el cach
17 RUN npm install \
18     && npm cache clean --force \
19     && rm -rf /tmp/* /root/.npm/_cacache
20
21 # copiar el resto del proyecto
22 COPY . .
23
24 #indicar el puerto que usara la maquina virtual
25 EXPOSE 8000
26
27 # Ejecuta la aplicación
28 # CMD ["node", "src/index.js"]
29 CMD ["npm", "start"]
```

Aquí al hacer la imagen me di cuenta que me había equivocado de nombre en el documento “dockerfile” lo llame “dockerfiler”. Está corregido pero en las capturas no.

Parte2:

Docker y DockerHub

- Cambiamos el `--watch` del `package.json`
- Y ejecutamos `npm init` para instalar dependencias con mi configuración y rellenamos con el nombre descripción, version...

```
package.json > {} scripts > [⌘] dev
1  {
2    "name": "exemple_node_docker",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "start": "node src/index.js",
8      "dev": "node --watch src/index.js"
9    },
10   "keywords": [],
11   "author": "",
12   "license": "ISC",
13   "dependencies": {
14     "express": "^4.18.2",
15     "morgan": "^1.10.0"
16   }
17 }
18
```

```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (main)
$ npm init
(node:14192) ExperimentalWarning: Support for loading ES Module in requi
(Use `node --trace-warnings ...` to show where the warning was created)
This utility will walk you through creating a package.json file.
```

Parte2:

Docker y DockerHub

- construimos la imagen con `docker build -t examenfinal_estefania_manjon:v1 .`
- Con docker images podemos ver las imagenes creadas
- iniciamos con run

```
• $ docker build -t examenfinal_estefania_manjon:v1 .  
[+] Building 5.9s (11/11) FINISHED  
=> [internal] load build definition from dockerfile  
=> => transferring dockerfile: 690B  
=> [internal] load metadata for docker.io/library/node:22  
=> [auth] library/node:pull token for registry-1.docker.io
```

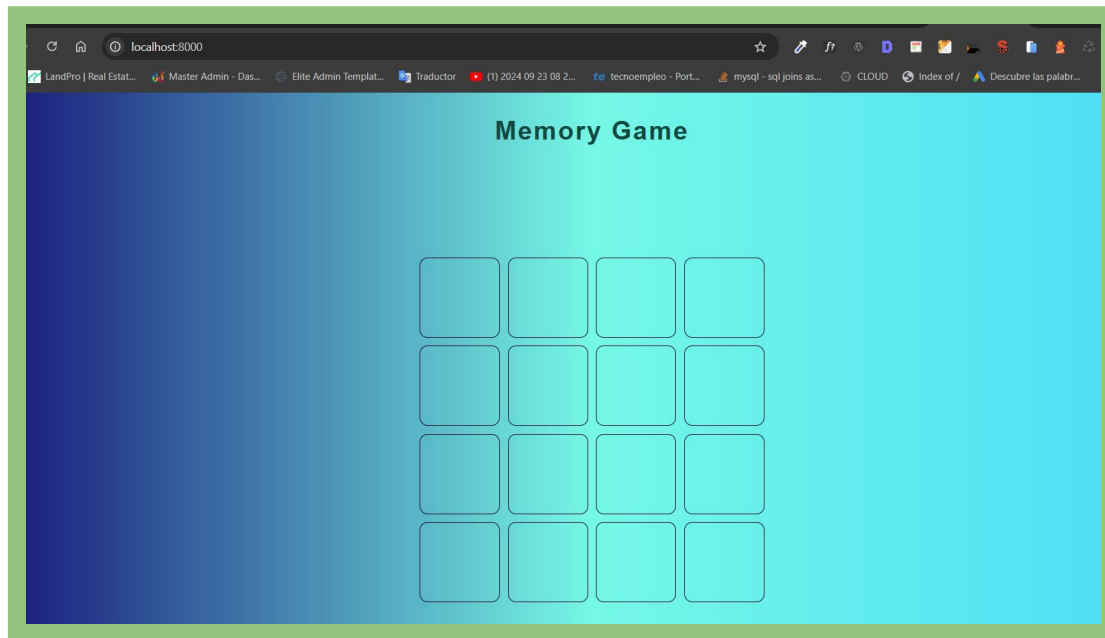
```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (main)  
• $ docker images  
REPOSITORY              TAG       IMAGE ID       CREATED        SIZE  
examenfinal_estefania_manjon  v1       b8c8ce6ac613   30 seconds ago 1.61GB  
emanjon/pruebadocker1      latest    aa0e77cac86e   25 hours ago  1.62GB
```

```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (main)  
$ docker run -it -p 8000:3000 examenfinal_estefania_manjon:v1  
  
> estefania@1.0.0 start  
> node src/index.js  
  
Servidor web arrancat a http://localhost:3000
```

Parte2:

Docker y DockerHub

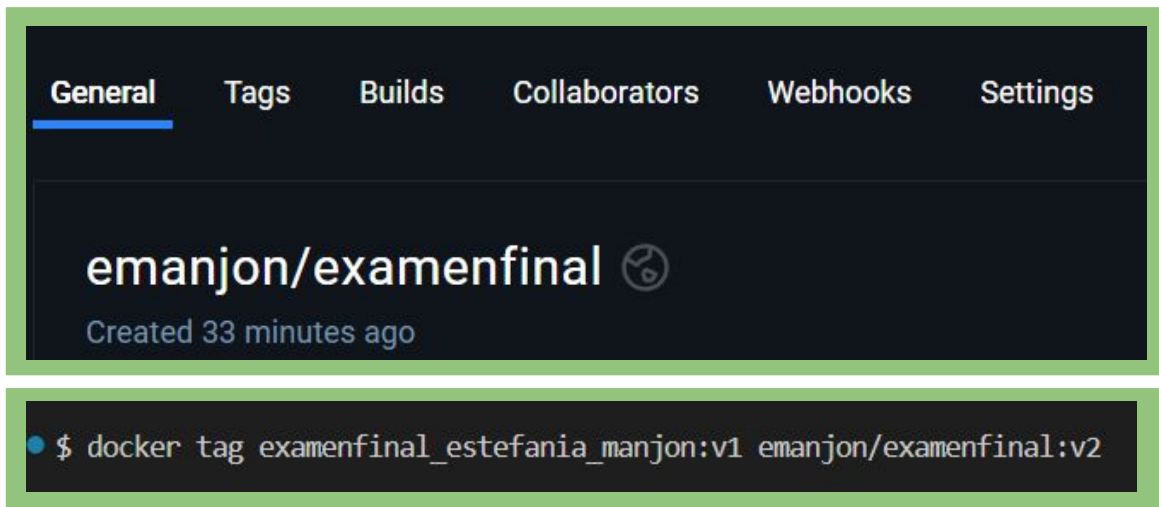
- Ya lo tenemos en local <http://localhost:8000/>



Parte2:

Docker y DockerHub

- Ahora crearemos el repositorio en dockerhub
- Creó una versión para el repositorio
- Nos autentificamos en docker



```
• $ docker tag examenfinal_estefania_manjon:v1 emanjon/examenfinal:v2
```

```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av$ do$ docker login -u "emanjon" -p "[REDACTED]"
WARNING! Using --password via the CLI is insecure. Use --password-stdin.
Login Succeeded
```

Parte2:

Docker y DockerHub

- y hacemos push para subirlo
- <https://hub.docker.com/repository/docker/emanjon/examenfinal/genera>
!

```
Aidet@Aidet-portatil MINGW64 ~/Downloads/mf0493_av_final (main)
$ docker push emanjon/examenfinal:v2
The push refers to repository [docker.io/emanjon/examenfinal]
4b9a7226bd1: Pushed
Let's see a little bit of the Docker Hub interface
```

emanjon/examenfinal

Last pushed 1 minute ago

This repository does not have a description   INCOMPLETE

This repository does not have a category   INCOMPLETE

Tags

This repository contains 1 tag(s).

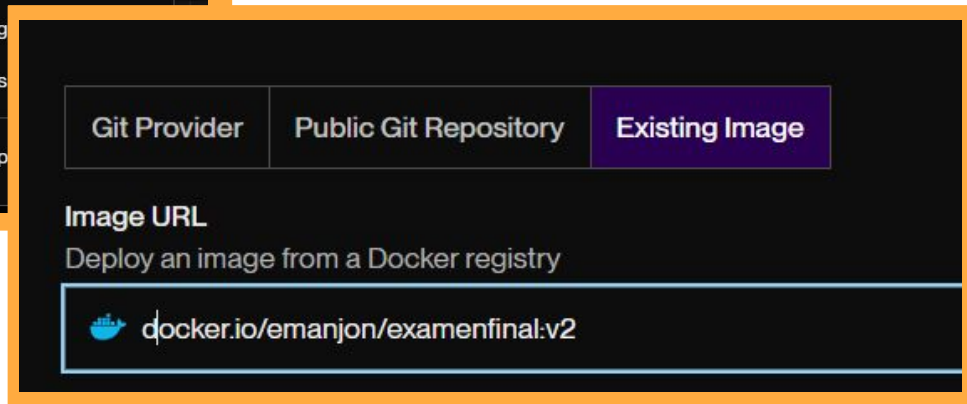
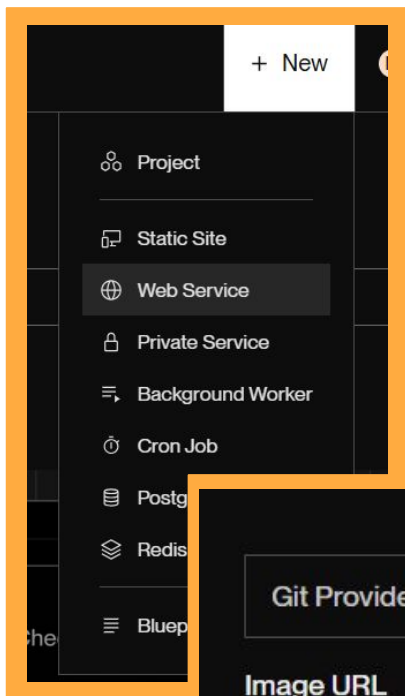
Tag	OS	Type	Pulled	Pushed
 v2		Image	a few seconds ago	a few seconds ago

[See all](#)

Parte3:

Render

- Entramos en render.io y le damos a > new > web service
- Existing image > y la url de docker hub



Parte3: Render

Region

Your services in the same [region](#) can communicate over a [private network](#). You currently have services running in



Frankfurt (EU Central)

1 existing service

- Region - la mas cercana
- plan: free
- y Deploy web service

November 20, 2024 at 12:49 PM

✓ Live

Deploy for 82e52a7 v2

All logs ▾

Q Search

Live tail ▾

GMT+1

↑

↶ ↷

```
Nov 20 12:49:29 PM ⓘ ==> Starting service...
Nov 20 12:49:39 PM ⓘ
Nov 20 12:49:39 PM ⓘ > estefania@1.0.0 start
Nov 20 12:49:39 PM ⓘ > node src/index.js
Nov 20 12:49:39 PM ⓘ
Nov 20 12:49:40 PM ⓘ Servidor web arrancat a http://localhost:10000
Nov 20 12:49:40 PM ⓘ HEAD / 200 2.340 ms - 514
Nov 20 12:49:50 PM ⓘ ==> Your service is live 🎉
Nov 20 12:49:51 PM ⓘ GET / 200 3.836 ms - 514
```

<https://examenfinal-v2.onrender.com/>