

## Instructivo configuración de repositorio local para pruebas automatizadas en Cypress

Pre instalaciones necesarias:

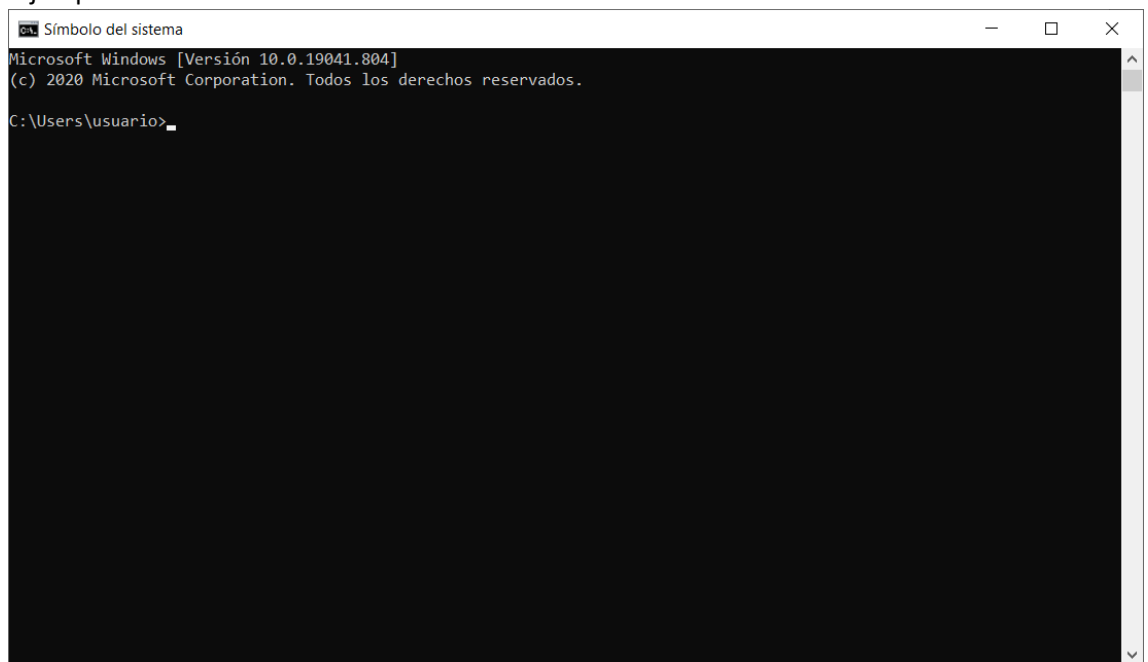
- Nodejs : <https://nodejs.org/es/download/>
- Git: <https://git-scm.com/downloads>

Instalar ambas en sus últimas versiones estables en caso de no tenerlas.

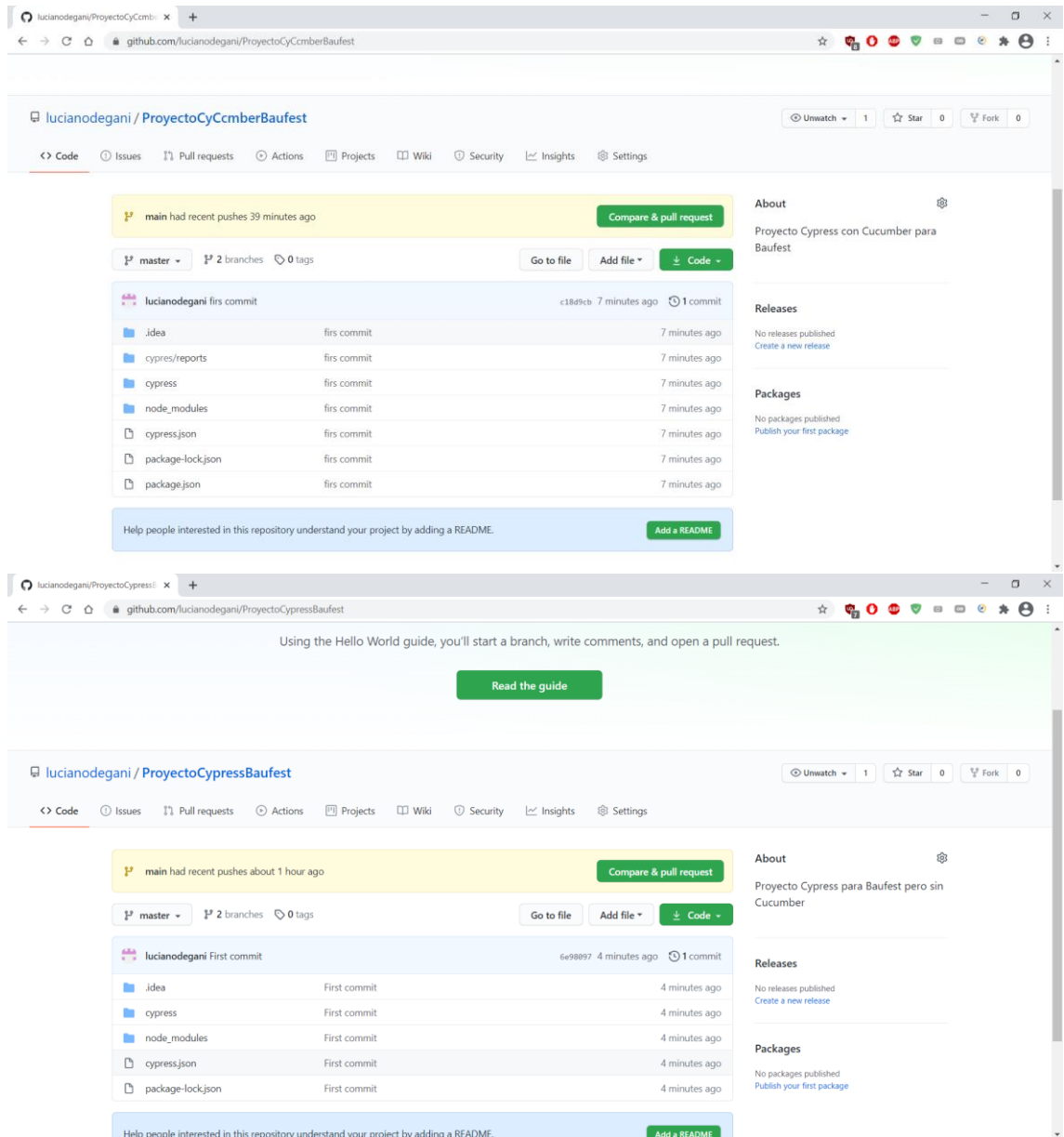
- 1) Elegir y crear una carpeta en nuestro sistema de archivos donde queramos descargar nuestro repositorio local. Abrimos cmd, la consola de windows y nos posicionamos en la ruta de la carpeta nueva que creamos.

Abrir CMD la consola de comando

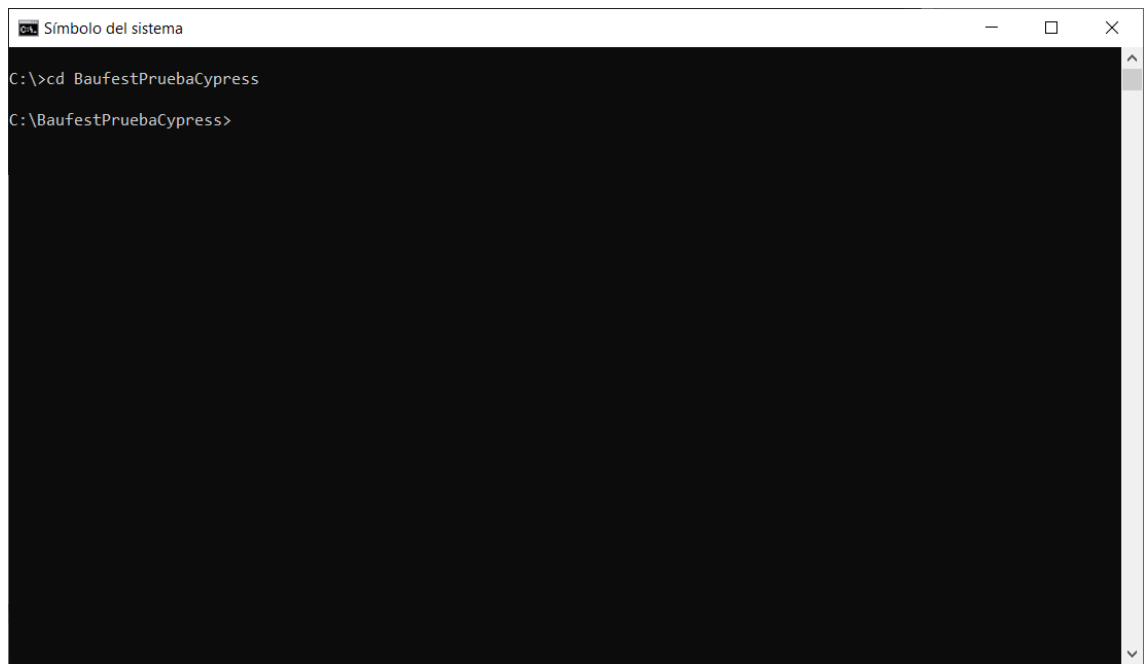
Ejemplo:



- 2) <https://github.com/lucianodegani/ProyectoCyCmberBaufest>  
es la versión con la implementación de cucumber y es la recomendada para ejecutar  
<https://github.com/lucianodegani/ProyectoCypressBaufest>  
es la versión sin cucumber  
en ambos están implementados los dos ejercicios requeridos de front end y servicios



- 3) Elegir y crear una carpeta en nuestro sistema de archivos donde queramos descargar nuestro repositorio local.
- 4) Abrimos cmd o una consola de windows y nos posicionamos en la ruta de la carpeta nueva que creamos. Ejemplo: C:\BaufestPruebaCypress

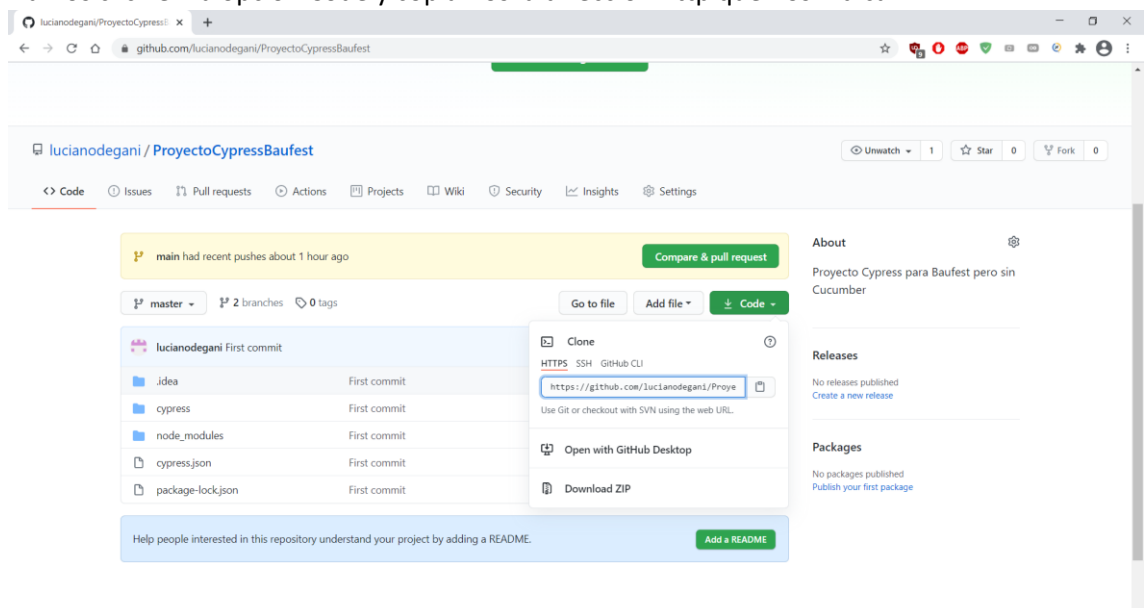


```
Símbolo del sistema

C:\>cd BaufestPruebaCypress

C:\BaufestPruebaCypress>
```

- 5) Abrimos <https://github.com/lucianodegani/ProyectoCyCmberBaufest>  
Y elegimos el repositorio que se quiere clonar en code  
Damos click en la opción Code y copiamos la dirección http que nos indica.



En este caso es <https://github.com/lucianodegani/ProyectoCypressBaufest.git>

En la consola abierta previamente ponemos lo siguiente: git clone seguido de la url que copiamos. Ej: git clone

<https://github.com/lucianodegani/ProyectoCyCmberBaufest.git>

Esto hará que se comience a clonar el repositorio indicado en la carpeta que hemos creado.

- 6) Una vez finalizado el paso 5, en la misma consola ponemos: `npm install` Esto hará que se instalen todos los paquetes y dependencias del repositorio
- 7) En caso de que no se instalen todas las dependencias instalarlas por separado aunque con el `npm install` seria suficiente.

```
npm install cypress --save-dev
npm install --save-dev cucumber
npm install --save-dev cypress-cucumber-preprocessor
npm install --save-dev mochawesome
npm install --save-dev mochawesome-report-generator
```

```
C:\BaufestPruebaCypress> git clone https://github.com/lucianodegani/ProyectoCyCmberBaufest.git
Cloning into 'ProyectoCyCmberBaufest'...
remote: Enumerating objects: 24698, done.
remote: Counting objects: 100% (24698/24698), done.
remote: Compressing objects: 100% (16152/16152), done.
remote: Total 24698 (delta 6827), reused 24695 (delta 6827), pack-reused 0
Receiving objects: 100% (24698/24698), 29.94 MiB | 4.40 MiB/s, done.
Resolving deltas: 100% (6827/6827), done.
Updating files: 100% (25748/25748), done.

C:\BaufestPruebaCypress> ProyectoCyCmberBaufest
"ProyectoCyCmberBaufest" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.

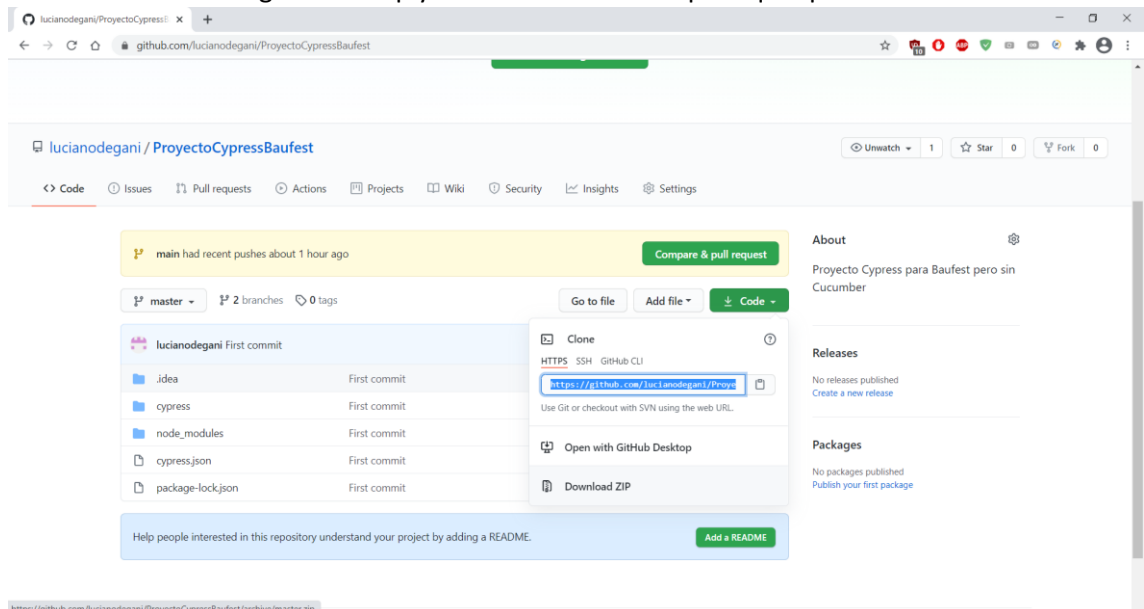
C:\BaufestPruebaCypress> cd ProyectoCyCmberBaufest

C:\BaufestPruebaCypress\ProyectoCyCmberBaufest> npm install
npm WARN mochawesome@6.2.2 requires a peer of mocha@>=7 but none is installed. You must install peer dependencies yourself.
npm WARN react-markdown@5.0.3 requires a peer of @types/react@>=16 but none is installed. You must install peer dependencies yourself.
npm WARN cypress-cucumber-baufest@1.0.0 license should be a valid SPDX license expression
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

audited 860 packages in 3.787s

56 packages are looking for funding
```

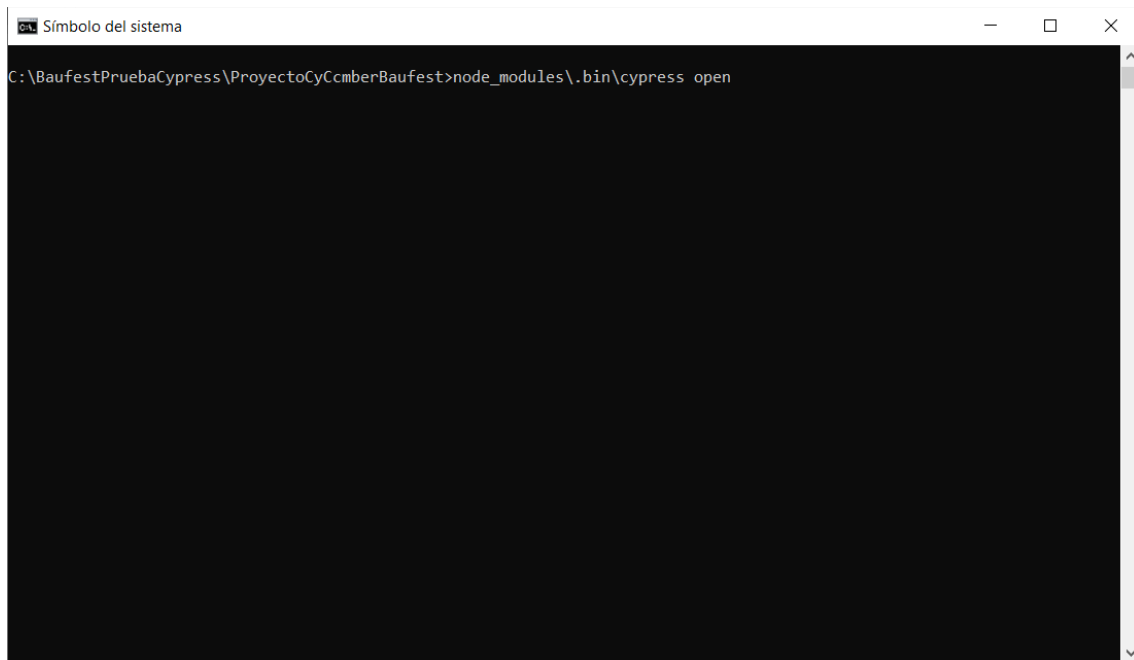
- 8) Otra forma es descargar como zip y lo ubicamos en la carpeta que queremos



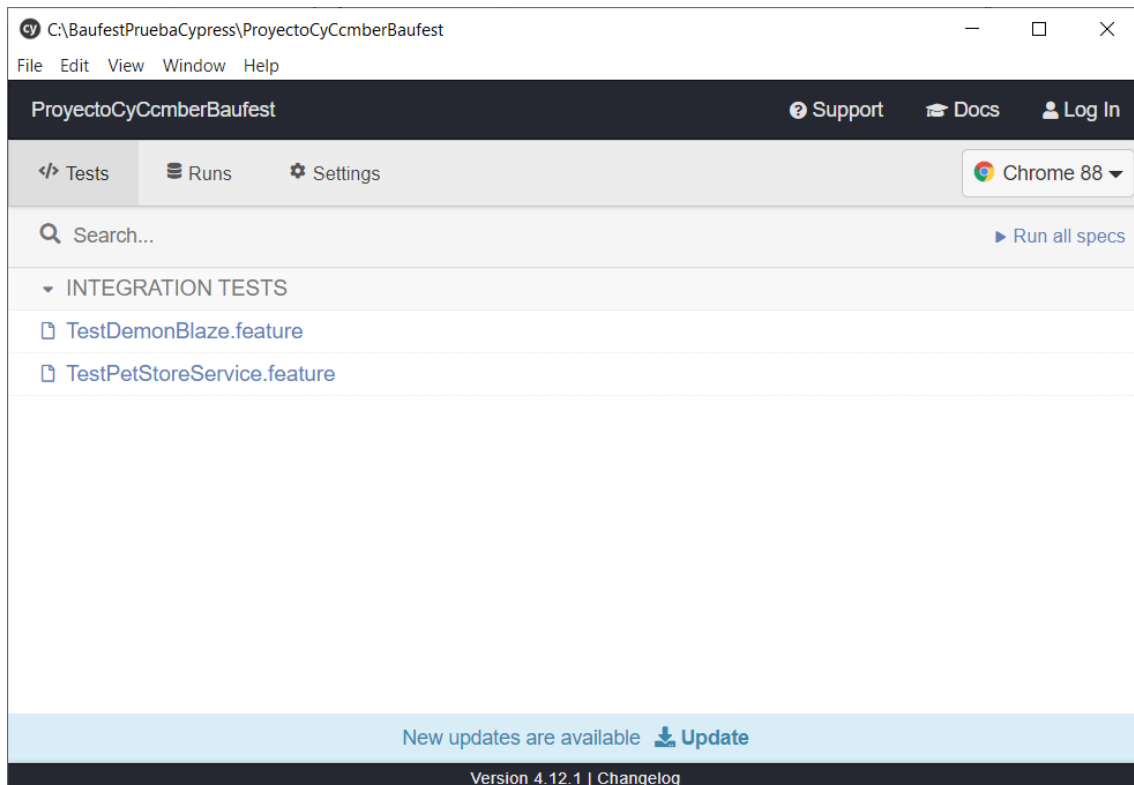
- 9) Una vez finalizado los pasos de instalación de dependencias se puede ejecutar de dos maneras los test
- 10) Desde la consola de comando en el directorio de nuestro proyecto `node_modules\.bin\cypress open`

para iniciar las pruebas ponemos en la consola:

```
node_modules\.bin\cypress open
```

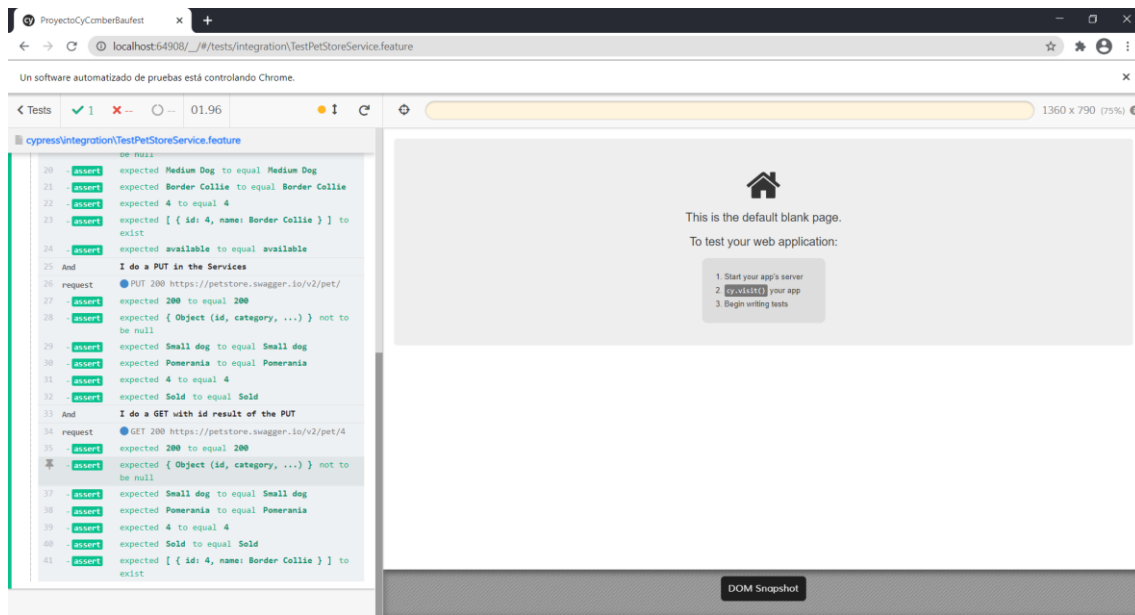


Esto abrirá una ventana como la que vemos a continuación y ya en este punto podremos seleccionar el archivo que necesitamos probar para que Cypress inicie la prueba.



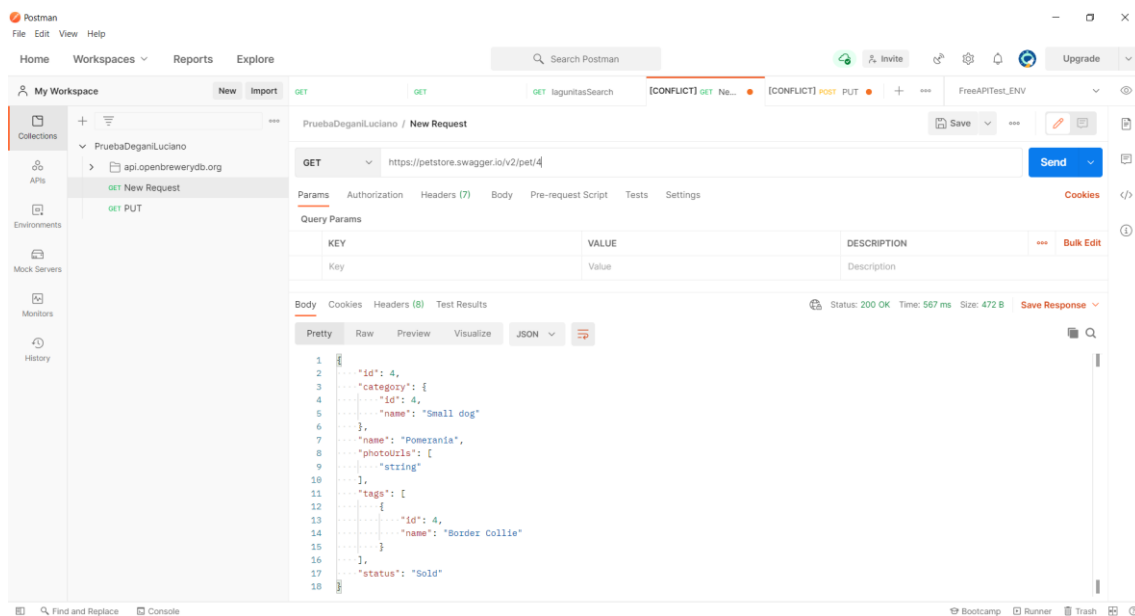
El ejercicio uno de la pagina de Demon Blaze

## Y el ejercicio 2 del web service de Pet Store



Si vemos en el servicio en Postman

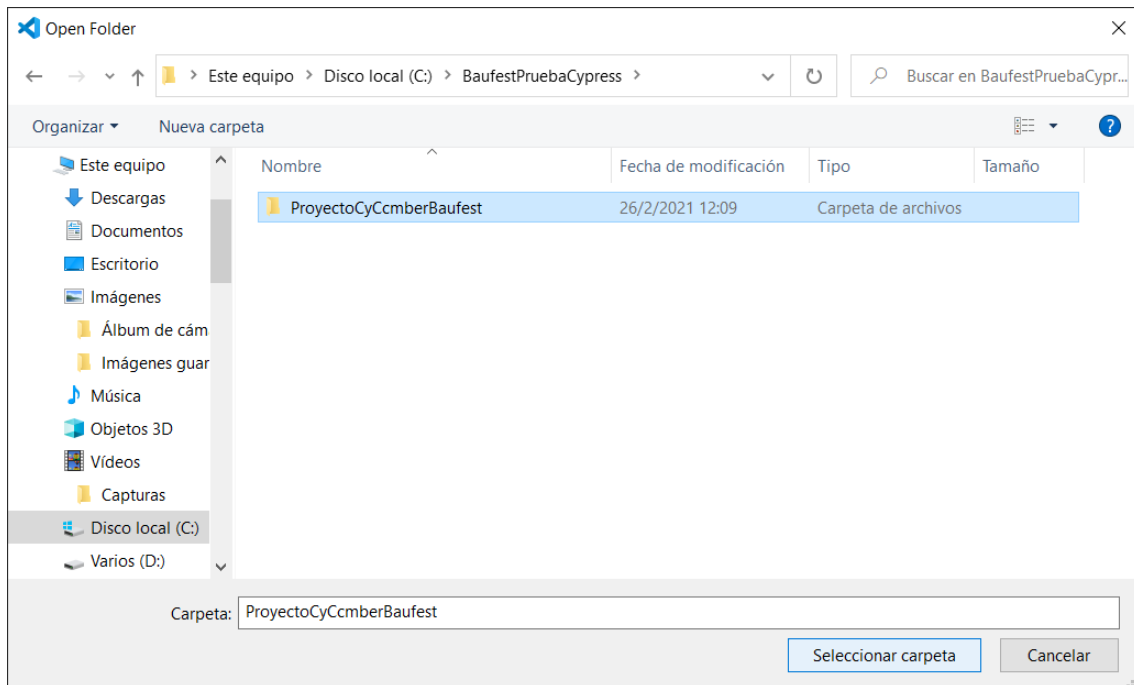
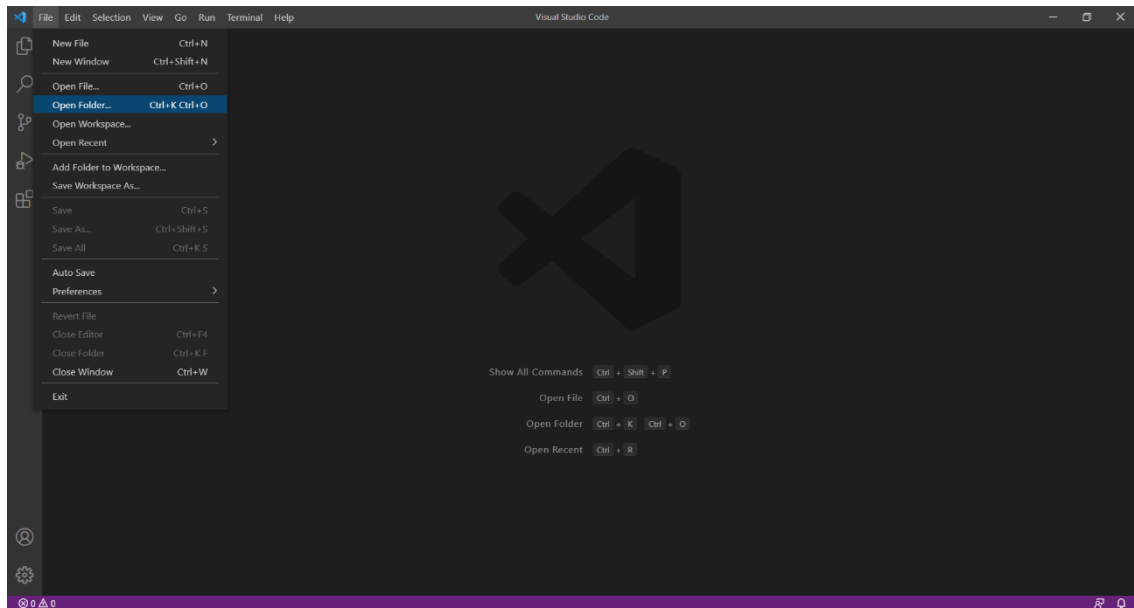
Veremos la última inclusión de la mascota en el servicio



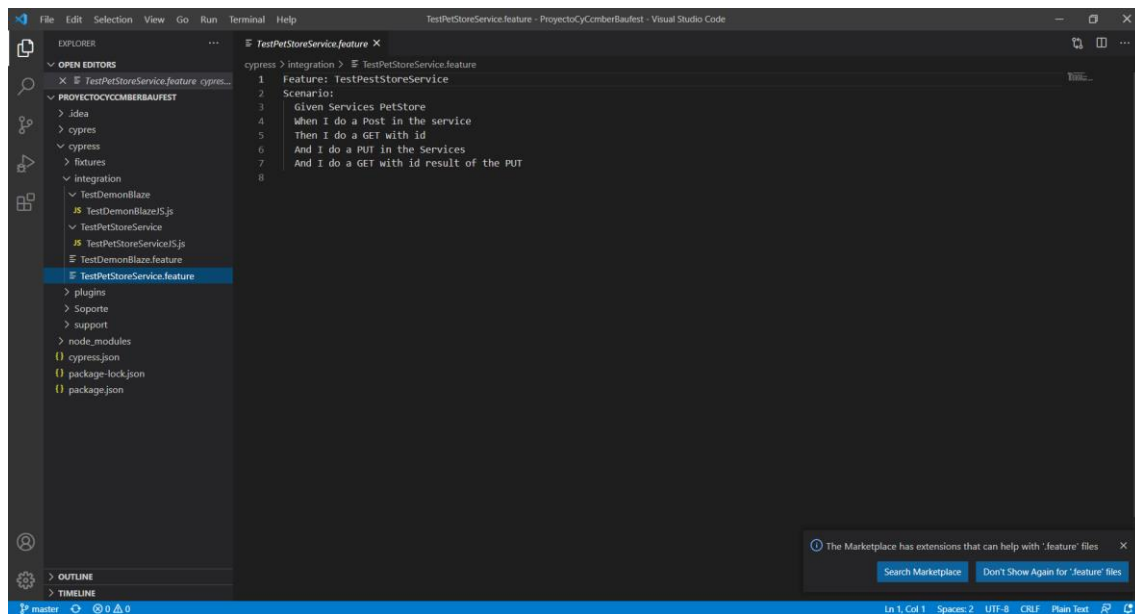
Para ver el código se puede ver a través de Visual Studio Code

<https://code.visualstudio.com/download>

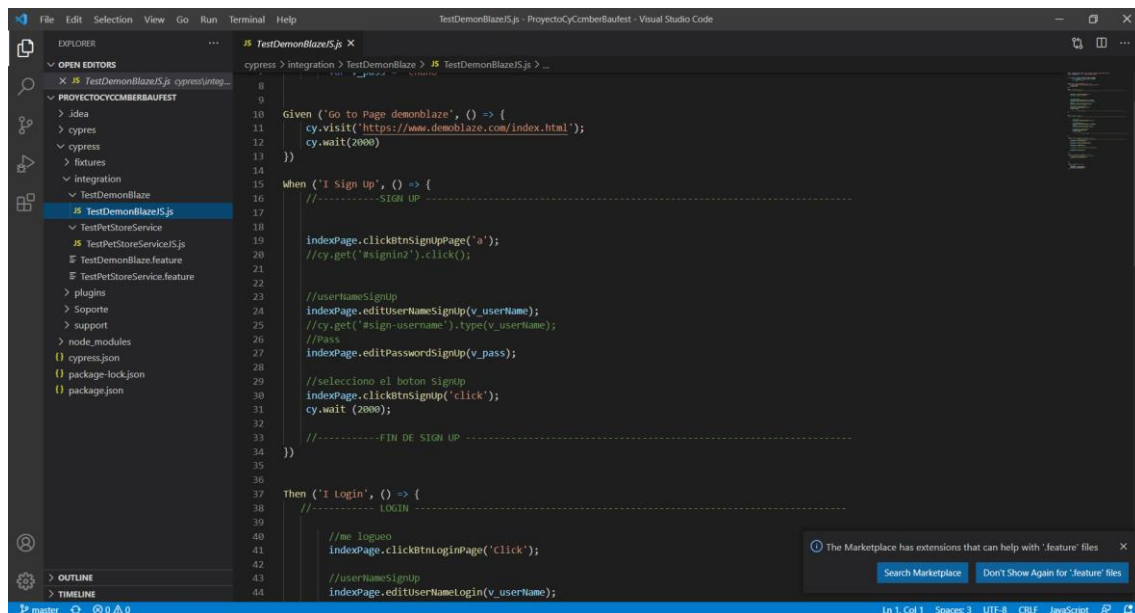
una vez instalado



Vemos acá implementado cucumber y los archivos feature



## Y los test



La otra forma de ejecución que a su vez guarda los reportes de mocha es la ejecución individual de los test



```
Símbolo del sistema
C:\BaufestPruebaCypress\ProyectoCyCmberBaufest>node_modules\.bin\cypress open

GET / 200 11.003 ms - -
GET /_cypress/runner/cypress_runner.css 200 3.991 ms - -
GET /_cypress/runner/cypress_runner.js 200 3.330 ms - -
GET /_cypress/static/favicon.ico 200 1.461 ms - -
GET /_cypress/iframe/integration/TestPetStoreService.feature 200 2.742 ms - 986
GET /_cypress/runner/fonts/fa-solid-900.woff2 200 1.066 ms - 76120
GET /_cypress/tests?p=cypress%5Csupport%5Cindex.js 200 1564.974 ms - -
GET /_cypress/tests?p=cypress%5Cintegration%5CTestPetStoreService.feature 200 3070.548 ms - -
HEAD / - ms - -
HEAD / - ms - -
HEAD / - ms - -

C:\BaufestPruebaCypress\ProyectoCyCmberBaufest>node_modules\.bin\cypress run --spec cypress\integration\TestDemonBlaze.feature --browser chrome
```

node\_modules\.bin\cypress run --spec cypress\integration\TestDemonBlaze.feature --browser chrome

node\_modules\.bin\cypress run --spec cypress\integration\TestPetStoreService.feature --browser chrome

```
Símbolo del sistema
C:\BaufestPruebaCypress\ProyectoCyCmberBaufest>node_modules\.bin\cypress run --spec cypress\integration\TestPetStoreService.feature --browser chrome

(Run Starting)

Cypress: 4.12.1
Browser: Chrome 88
Specs: 1 found (TestPetStoreService.feature)
Searched: cypress\integration\TestPetStoreService.feature

Running: TestPetStoreService.feature (1 of 1)

TestPetStoreService
✓ (1000ms)

1 passing (2s)

[mochawesome] Report JSON saved to C:\BaufestPruebaCypress\ProyectoCyCmberBaufest\cypress\reports\mochawesome.json
[mochawesome] Report HTML saved to C:\BaufestPruebaCypress\ProyectoCyCmberBaufest\cypress\reports\mochawesome.html

(Results)

Tests: 1
Passing: 1
Failing: 0
Pending: 0
Skipped: 0
Screenshots: 0
Video: false
Duration: 1 second
Spec. Ran: TestPetStoreService.feature

(Run Finished)
```