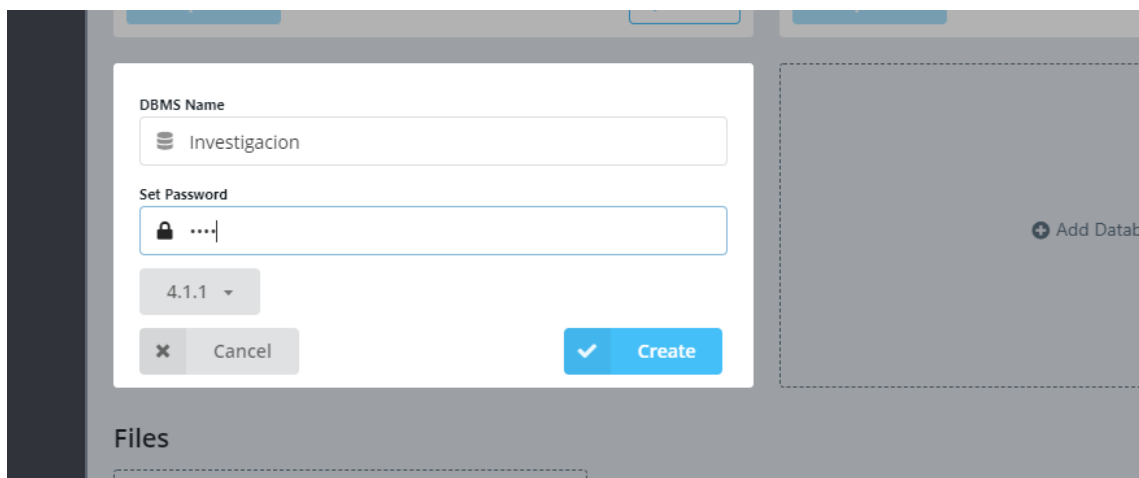


Instalación de la Aplicación InJour

Para poder ejecutar la aplicación una vez descargada y extraída la carpeta con el código, se deben seguir los siguientes pasos:

1. Crear la base de datos en Neo4J Desktop. Habrá que crear una base de datos local, a la cual se le puede dar el nombre que quieras, con la contraseña; “1234”. La versión de la base de datos debe ser la 4.1.1.



Una vez creada, se inicializa y se abre el browser. Dentro de la línea de comando que aparece en el browser, se pegará el código que aparece en el archivo de texto “crearBBDD.txt” y se ejecutará, creando así la base de datos.



Recordar que la base de datos tiene que mantenerse activa para que la aplicación funcione correctamente.

2. Instalar y ejecutar el backend. Desde una terminal o consola de comandos te ubicas con el comando '*cd*' en la carpeta backend del código descargado. Una vez ahí ejecutas el comando '*npm install*' y esperas a que termine el proceso de instalación de las dependencias.

```
PS C:\Users\USUARIO\Desktop\SiBi-InJour-main> cd .\backend\  
PS C:\Users\USUARIO\Desktop\SiBi-InJour-main\backend> npm install  
npm WARN backend@1.0.0 No description  
npm WARN backend@1.0.0 No repository field.  
  
added 60 packages from 54 contributors and audited 60 packages in 3.697s  
found 0 vulnerabilities  
  
PS C:\Users\USUARIO\Desktop\SiBi-InJour-main\backend> |
```

Cuando haya terminado el proceso ejecutas el comando '*nodemon app.js*' para poder así ejecutar el backend. Recordar no cerrar la terminal ya que dejara de ejecutarse el backend y no funcionara la aplicación.

```
PS C:\Users\USUARIO\Desktop\SiBi-InJour-main\backend> nodemon app.js  
[nodemon] 2.0.6  
[nodemon] to restart at any time, enter `rs`  
[nodemon] watching path(s): *.*  
[nodemon] watching extensions: js,mjs,json  
[nodemon] starting `node app.js`  
Estoy escuchando en el siguiente puerto 3000!  
|
```

3. Instalacion y ejecución del frontend. Al igual que en el backend abrimos una nueva terminal y con el comando '*cd*' nos ubicamos en la carpeta frontend del código descargado. Una vez ahí ejecutamos el comando '*npm install*' para instalar las dependencias.

```

PS C:\Users\USUARIO\Desktop\Sibi-InJour-main> cd .\frontend\
PS C:\Users\USUARIO\Desktop\Sibi-InJour-main\frontend> npm install

> yorkie@2.0.0 install C:\Users\USUARIO\Desktop\Sibi-InJour-main\frontend\node_modules\yorkie
> node bin/install.js

setting up Git hooks
can't find .git directory, skipping Git hooks installation

> core-js@3.8.1 postinstall C:\Users\USUARIO\Desktop\Sibi-InJour-main\frontend\node_modules\core-js
> node -e "try{require('./postinstall')}catch(e){}"

Thank you for using core-js ( https://github.com/zloirock/core-js ) for polyfilling JavaScript standard library!

The project needs your help! Please consider supporting of core-js on Open Collective or Patreon:
> https://opencollective.com/core-js
> https://www.patreon.com/zloirock

Also, the author of core-js ( https://github.com/zloirock ) is looking for a good job -)

> ejs@2.7.4 postinstall C:\Users\USUARIO\Desktop\Sibi-InJour-main\frontend\node_modules\ejs
> node ./postinstall.js

Thank you for installing EJS: built with the Jake JavaScript build tool (https://jakejs.com/)

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\webpack-dev-server\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"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.13 (node_modules\webpack-chokidar2\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"})
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@2.1.3 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@2.1.3: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})

added 1321 packages from 969 contributors and audited 1324 packages in 36.31s
found 0 vulnerabilities

PS C:\Users\USUARIO\Desktop\Sibi-InJour-main\frontend>

```

Cuando este proceso haya terminado se ejecuta el comando `'npm run serve'` para ejecutar el frontend. Cuando se procese la ejecución nos aparecerán dos links, y pulsando en cualquiera de ellos nos abrirá el frontend en el navegador y podremos comenzar a usar la aplicación y explorar su funcionalidad.

```

DONE Compiled successfully in 27058ms

App running at:
- Local: http://localhost:8080/
- Network: http://192.168.1.37:8080/

Note that the development build is not optimized.
To create a production build, run npm run build.

```

Todos estos pasos se encuentran explicados y ejecutados en el video: “instalacionAplicacionInJour.mp4” de la carpeta videos.