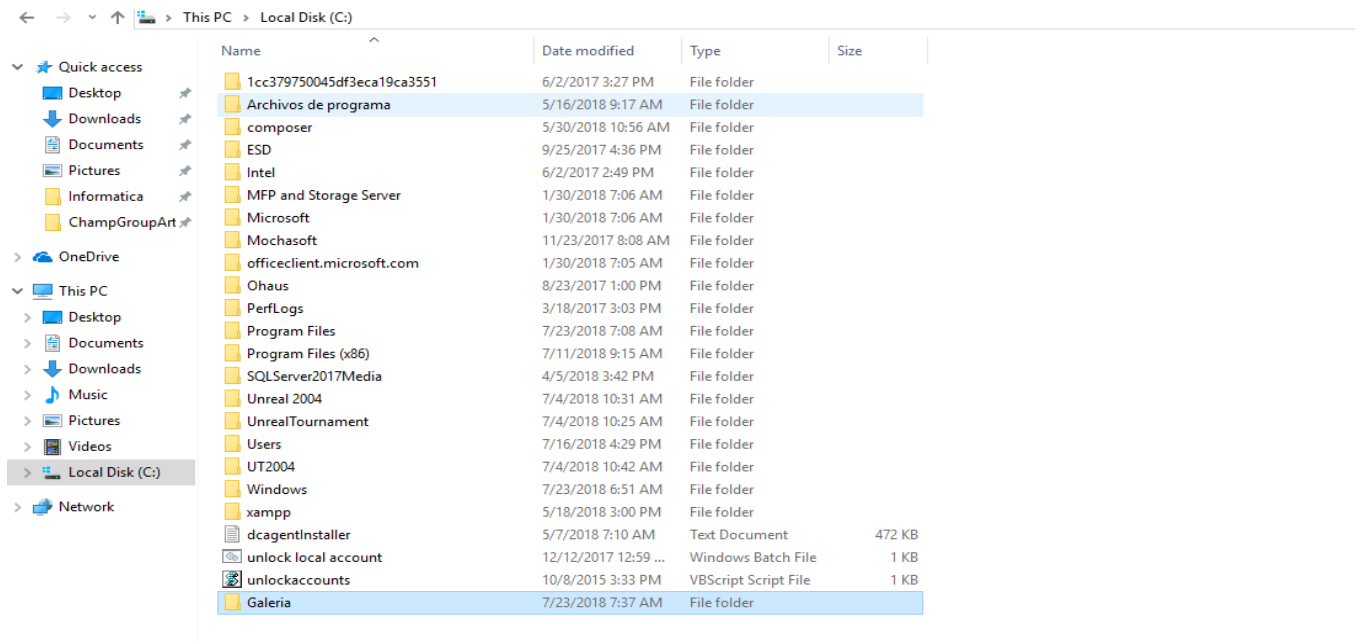
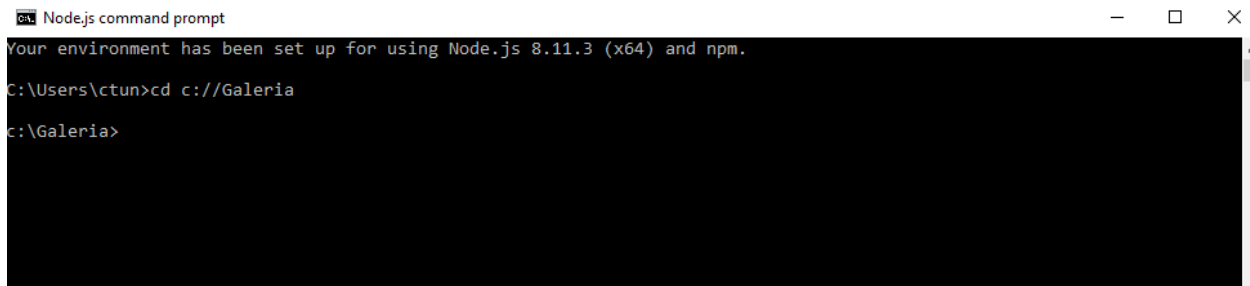


Manual para realizar una
galería de imágenes obtenida
de una base de datos.

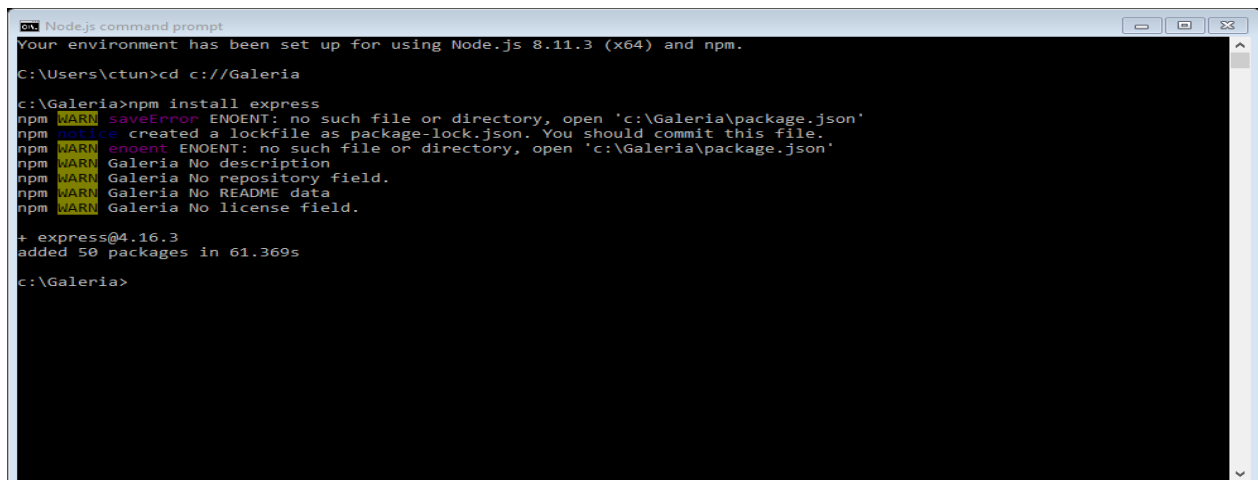
1-crearemos una carpeta llamada Galería



2-Abrimos la línea de comandos node.js y escribimos lo siguiente:



3-instalaremos Express con el siguiente comando:



4-- Creamos la siguiente instrucción, esto con el objetivo de crear un proyecto con Express

```
Node.js command prompt
c:\Galeria>npm install express-generator -g
C:\Users\ctun\AppData\Roaming\npm\express -> C:\Users\ctun\AppData\Roaming\npm\node_modules\express-generator\bin\express-cli.js
+ express-generator@4.16.0
added 10 packages in 6.055s

c:\Galeria>
```

5-Con el siguiente comando creamos la estructura de la galería

```
Node.js command prompt
error: unknown option '--view-ejs'

c:\Galeria>express --view=ejs
destination is not empty, continue? [y/N] y

  create : public\
  create : public\javascripts\
  create : public\images\
  create : public\stylesheets\
  create : public\stylesheets\style.css
  create : routes\
  create : routes\index.js
  create : routes\users.js
  create : views\
  create : views\error.ejs
  create : views\index.ejs
  create : app.js
  create : package.json
  create : bin\
  create : bin\www

install dependencies:
  > npm install

run the app:
  > SET DEBUG=galeria:* & npm start

c:\Galeria>
```

6-Corremos la aplicación con el siguiente comando

```
Node.js command prompt
c:\Galeria>npm start

> galeria@0.0.0 start c:\Galeria
> node ./bin/www

module.js:549
  throw err;
  ^
Error: Cannot find module 'cookie-parser'
    at Function.Module._resolveFilename (module.js:547:15)
    at Function.Module._load (module.js:474:25)
    at Module.require (module.js:596:17)
    at require (internal/module.js:11:18)
    at Object.<anonymous> (c:\Galeria\app.js:4:20)
    at Module._compile (module.js:652:30)
    at Object.Module._extensions..js (module.js:663:10)
    at Module.load (module.js:565:32)
    at tryModuleLoad (module.js:505:12)
    at Function.Module._load (module.js:497:3)
npm ERR! code ELIFECYCLE
npm ERR! errno 1
npm ERR! galeria@0.0.0 start: `node ./bin/www`
npm ERR! Exit status 1
```

7-se puede observar que al momento de correr la aplicación surgieron errores esto requiere de una dependencia para que funcione correctamente así que se escribirá el siguiente comando.

```
Node.js command prompt
c:\Galeria>npm install http-errors

npm update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$USER C:\Users\ctun\.config

+ http-errors@1.6.3
updated 1 package in 1.935s

c:\Galeria>
```

8-corremos de nuevo la aplicación en caso de haber error se ejecutará un comando que se mostrará a continuación:

```
Node.js command prompt

c:\Galeria>npm start

> galeria@0.0.0 start c:\Galeria
> node ./bin/www

module.js:549
  throw err;
  ^

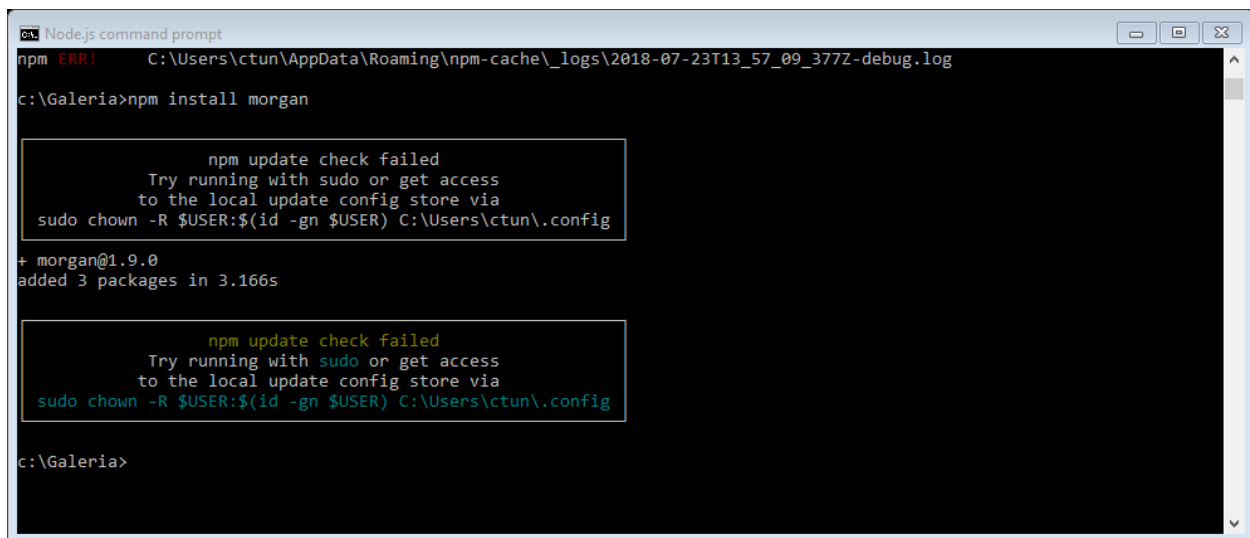
Error: Cannot find module 'cookie-parser'
    at Function.Module._resolveFilename (module.js:547:15)
    at Function.Module._load (module.js:474:25)
    at Module.require (module.js:596:17)
    at require (internal/module.js:11:18)
    at Object.<anonymous> (c:\Galeria\app.js:4:20)
    at Module._compile (module.js:652:30)
    at Object.Module._extensions..js (module.js:663:10)
    at Module.load (module.js:565:32)
    at tryModuleLoad (module.js:505:12)
    at Function.Module._load (module.js:497:3)
npm ERR! code ELIFECYCLE
npm ERR! errno 1
npm ERR! galeria@0.0.0 start: `node ./bin/www`
```

```
Node.js command prompt
C:\Users\ctun\AppData\Roaming\npm-cache\_logs\2018-07-23T13_31_28_622Z-debug.log

c:\Galeria>npm install cookie-parser
+ cookie-parser@1.4.3
added 1 package in 8.367s

c:\Galeria>
```

9- Realizaremos el mismo procedimiento, pero con la dependencia Morgan



```
Node.js command prompt
npm ERR! C:\Users\ctun\AppData\Roaming\npm-cache\_logs\2018-07-23T13_57_09_377Z-debug.log

c:\Galeria>npm install morgan

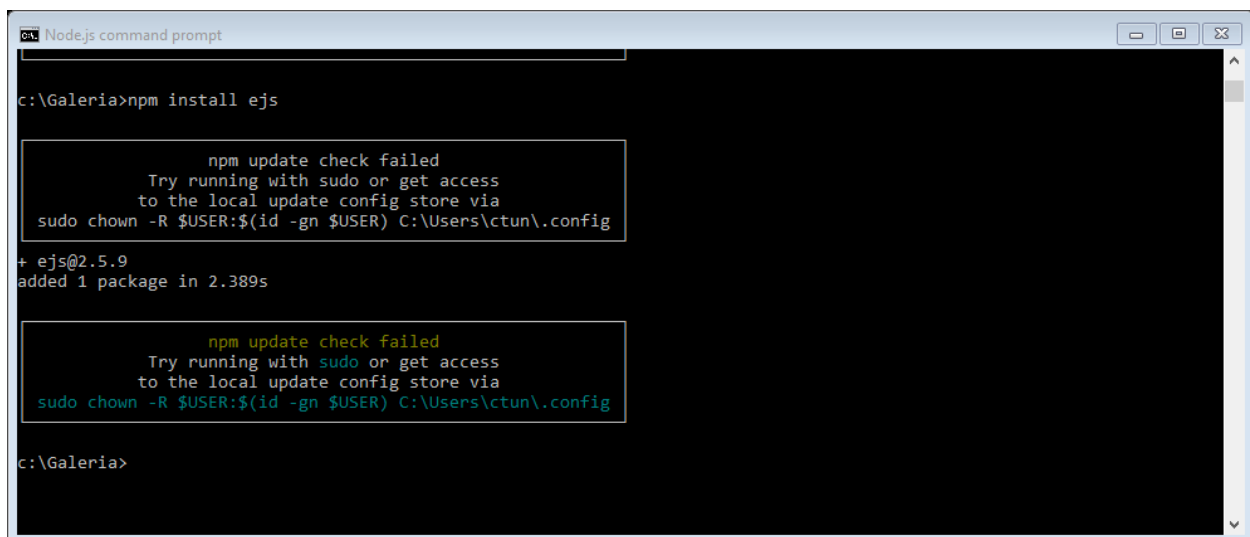
npm update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$(id -gn $USER) C:\Users\ctun\.config

+ morgan@1.9.0
added 3 packages in 3.166s

npm update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$(id -gn $USER) C:\Users\ctun\.config

c:\Galeria>
```

10-Como último punto instalaremos el motor de plantillas ejs para trabajar con las views



```
Node.js command prompt

c:\Galeria>npm install ejs

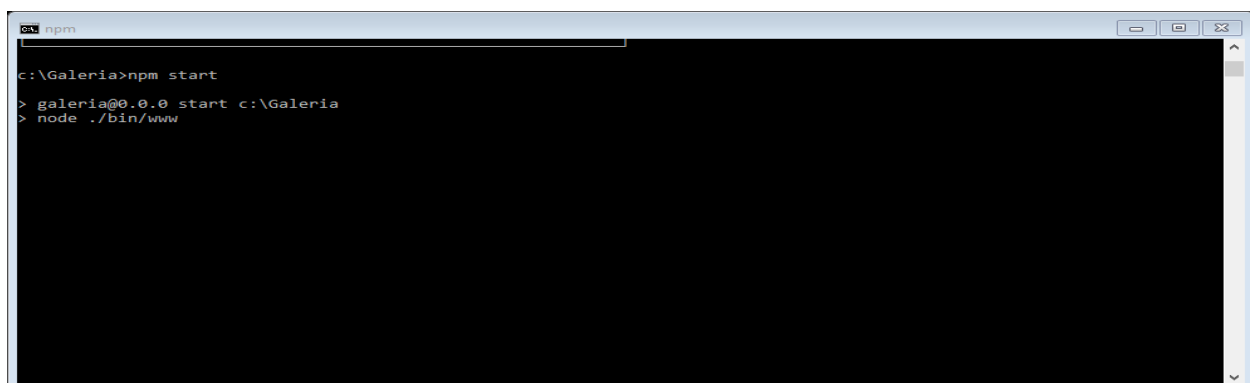
npm update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$(id -gn $USER) C:\Users\ctun\.config

+ ejs@2.5.9
added 1 package in 2.389s

npm update check failed
Try running with sudo or get access
to the local update config store via
sudo chown -R $USER:$(id -gn $USER) C:\Users\ctun\.config

c:\Galeria>
```

11-Una vez terminado todo el procedimiento de creación del proyecto corremos de nuevo la aplicación:

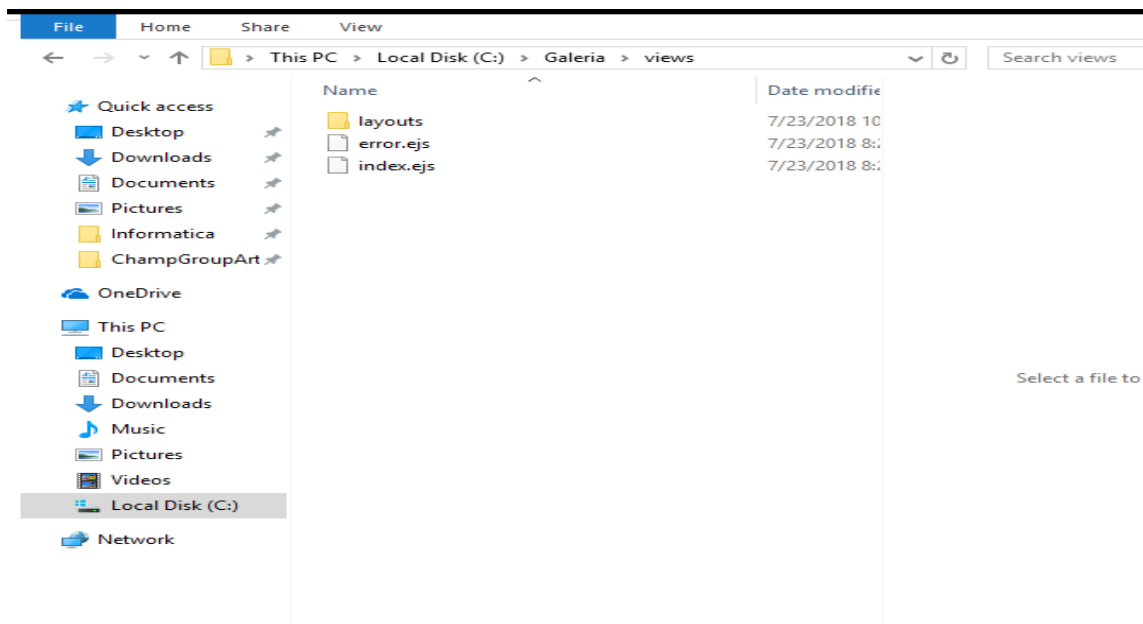


```
npm

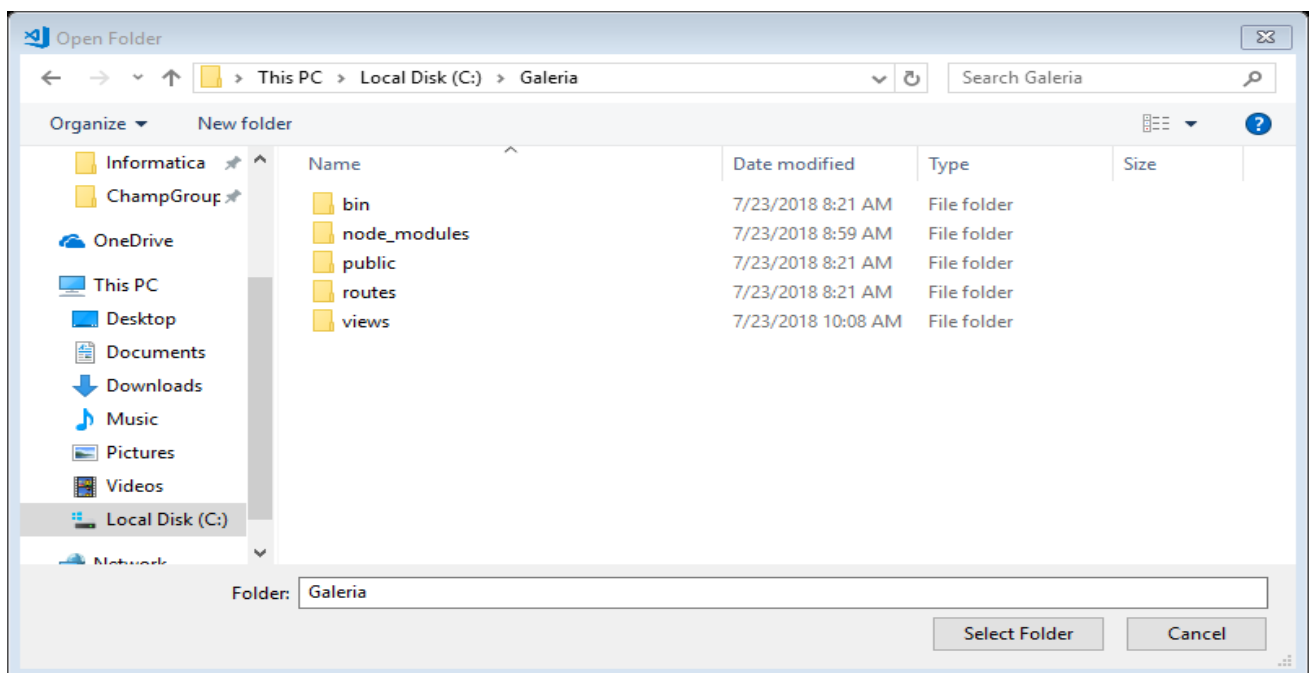
c:\Galeria>npm start

> galeria@0.0.0 start c:\Galeria
> node ./bin/www
```

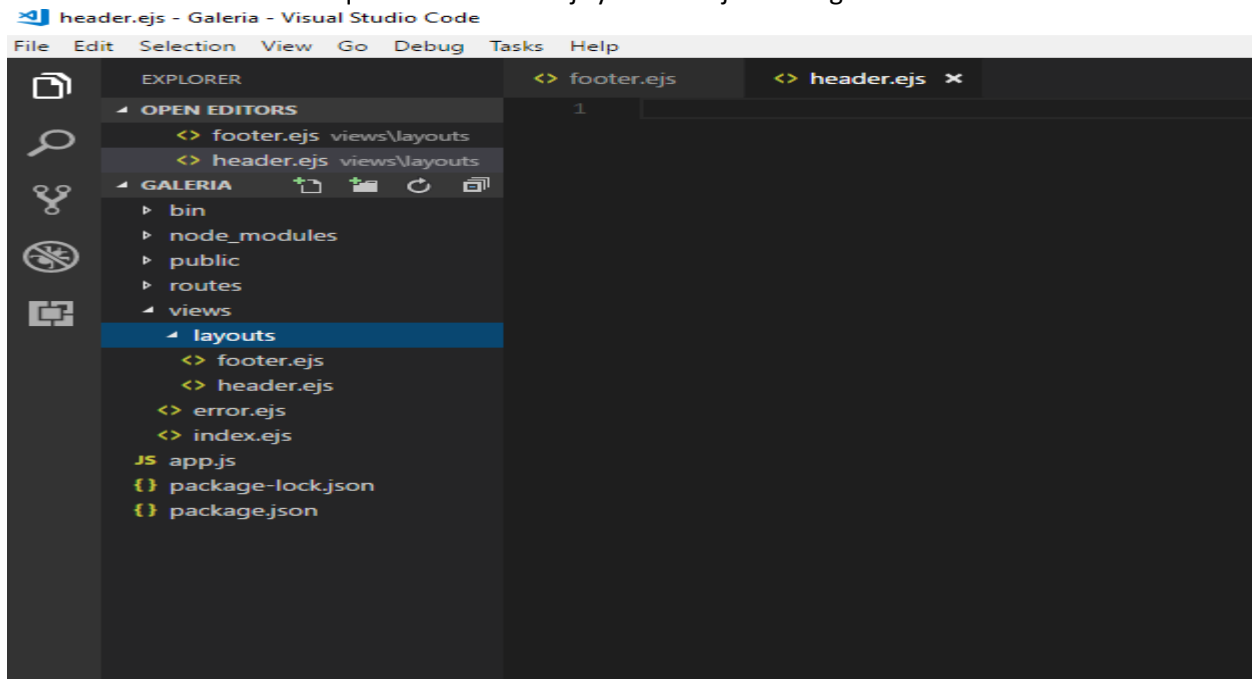
12-Para crear el proyecto vamos a irnos a la siguiente ruta y crearemos una carpeta llamada layouts:



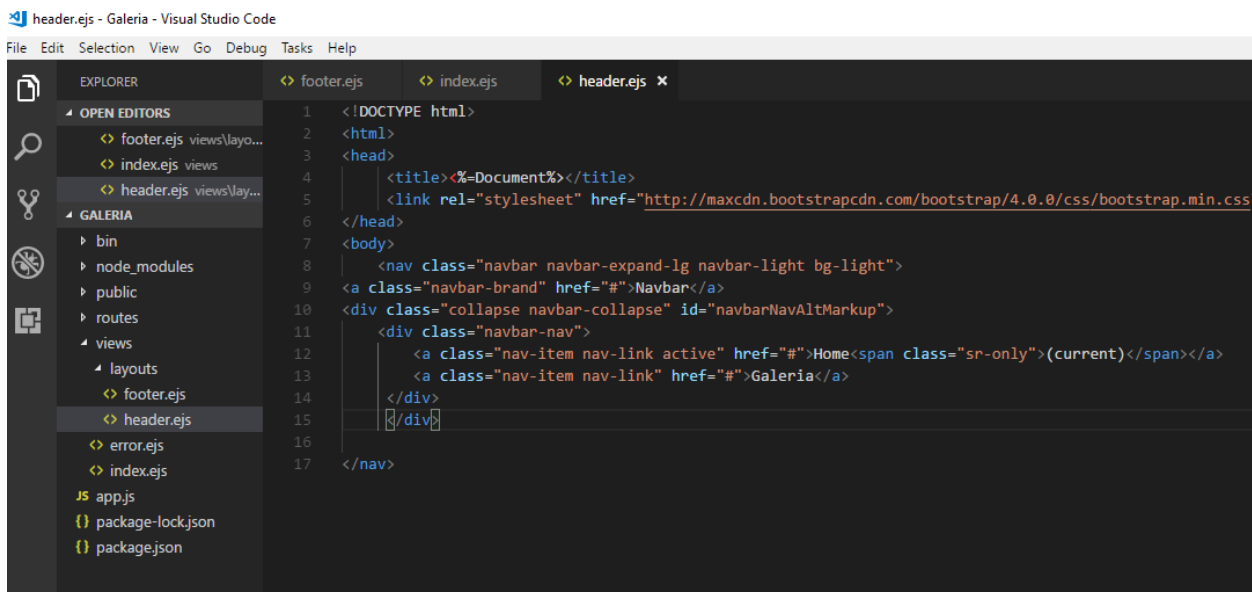
13.- Nos iremos a visual studio Code y abriremos la carpeta del proyecto Galeria



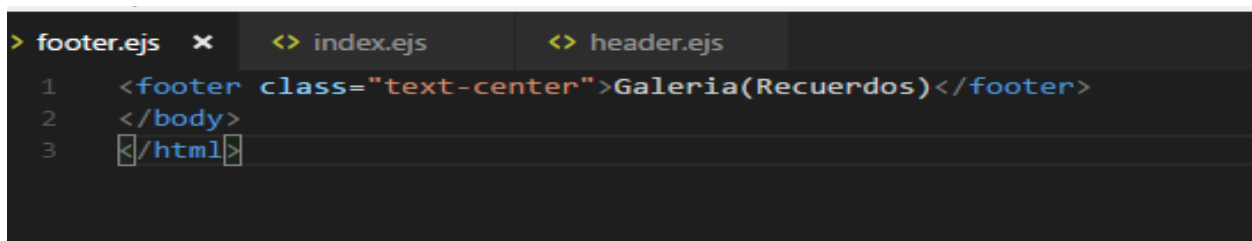
14-Crearemos dos archivos que serán: Footer.ejs y header.ejs de la siguiente manera



15-Adentro de header escribiremos el siguiente código :



16-A continuación, escribiremos ahora en el footer el siguiente código:



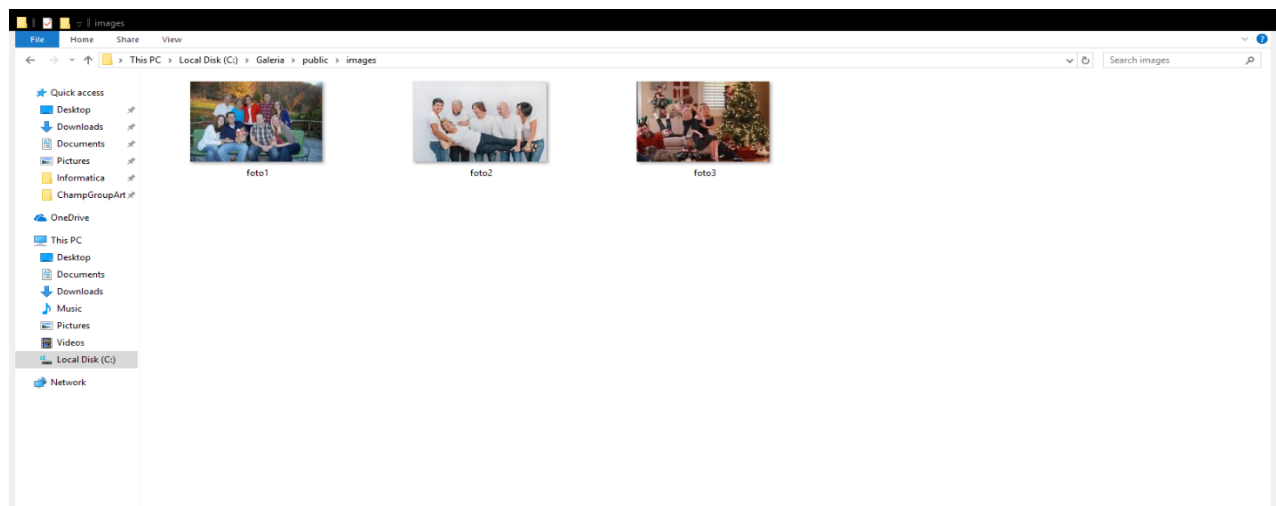
17-- Vamos a crear la base de datos en MySQL que le llamaremos Galeria con una tabla llamada imágenes con 2 campos

The screenshot shows the phpMyAdmin interface. On the left, there's a sidebar with a tree view of databases and tables. The main area is titled 'Server: 127.0.0.1 » Database: Galeria » Table: imagenes'. Below the title bar, there are tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', and 'Operation'. The 'Structure' tab is active, showing the table 'imagenes' with 1 column(s). The table structure is defined as follows:

Name	Type	Length/Values	Default	Collation	Attributes
id	INT		None		
imagen	VARCHAR	1000	None		

Below the table structure, there are fields for 'Table comments:', 'Collation:', and 'Storage Engine:'. The 'Storage Engine' is set to 'InnoDB'.

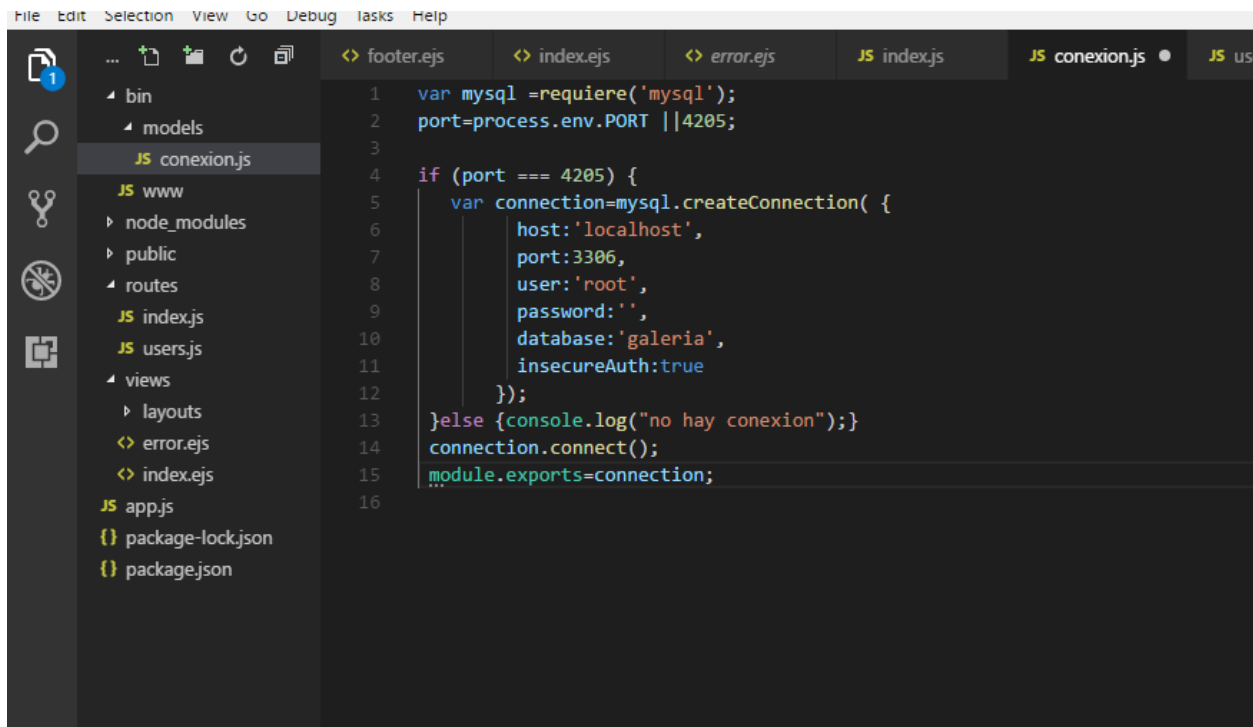
18- A continuación, crearemos 3 imágenes que estarán relacionadas con 3 registros en la tabla productos se llamarán foto1.jpg, foto2.jpg y foto3.jpg (más adelante se podrán adjuntar estas imágenes utilizando un formulario para agregar registros)



19-- Vamos a insertar 3 registros en nuestra base de datos llenando la información de cada foto

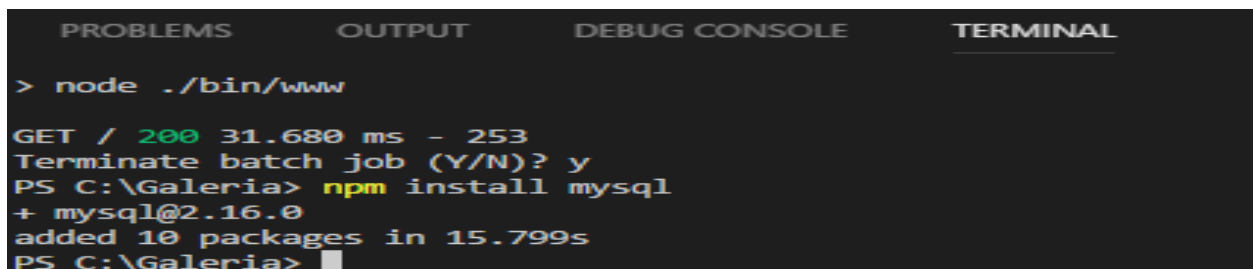
The screenshot shows the phpMyAdmin SQL interface. At the top, a green status bar indicates 'Showing rows 0 - 5 (6 total, Query took 0.0028 seconds.)'. Below this, the SQL query is displayed: `SELECT * FROM `imagenes``. At the bottom, there are controls for 'Show all', 'Number of rows: 25', and 'Filter rows: Search this table'. Below the query area, there are links for '+ Options' and 'imagen'. The 'imagen' link is highlighted, and below it, three image files are listed: 'foto1.jpg', 'foto2.jpg', and 'foto3.jpg'.

20-Crearemos la conexión a la base de datos, como se muestra en la siguiente imagen:



```
1 var mysql =require('mysql');
2 port=process.env.PORT ||4205;
3
4 if (port === 4205) {
5     var connection=mysql.createConnection( {
6         host:'localhost',
7         port:3306,
8         user:'root',
9         password:'',
10        database:'galeria',
11        insecureAuth:true
12    });
13 }else {console.log("no hay conexion");}
14 connection.connect();
15 module.exports=connection;
16
```

21-conectamos la base de datos Mysql y por ultimo corremos npm start para ver las imágenes en la página.



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
> node ./bin/www
GET / 200 31.680 ms - 253
Terminate batch job (Y/N)? y
PS C:\Galeria> npm install mysql
+ mysql@2.16.0
added 10 packages in 15.799s
PS C:\Galeria>
```

22-Ejecutamos el programa y obtenemos la galería

