MANUAL DE INSTALACIÓN DE LA HERRAMIENTA

**CONTRATACIONES**

**ABIERTAS**

Contenido

[1. Arquitectura. 3](#_Toc24112967)

[2. Creación y configuración de la base de datos. 3](#_Toc24112968)

[3. Descargar el archivo del código. 5](#_Toc24112969)

[4. Configuración de variables de conexión a la base de datos 7](#_Toc24112970)

[5. Ejecutar los script para crear la estructura en la base edca. 10](#_Toc24112971)

[6. Compilación y ejecución en el módulo “Sistema de captura”. 14](#_Toc24112972)

[6.1. Actualización de versión de Bower. 16](#_Toc24112975)

[6.2. Validación de conexión a base de datos. 17](#_Toc24112976)

[6.3. Asignación de variables para conexión a la API de PNT. 18](#_Toc24112977)

[6.4 Asignación de variables MAPTOKEN. 19](#_Toc24112983)

[7. Compilación y ejecución en el módulo “Dashboard - visualizador”. 22](#_Toc24112984)

[8. Crear usuario del aplicativo. 24](#_Toc24112985)

[9. Iniciar los servicios de los módulos 27](#_Toc24112986)

[10. Acceder a los módulos. 28](#_Toc24112987)

[10.1. Módulo Sistema de captura 28](#_Toc24112992)

[10.2. Módulo de visualización (dashboard). 29](#_Toc24112993)

# Arquitectura.

El servidor requiere de los siguientes requerimientos de hardware:

|  |  |
| --- | --- |
| Memoria RAM. | 4 GB o superior. |
| Velocidad del procesador. | Intel Core Duo 2.0 GHz o superior. |
| Almacenamiento. | 250 GB o superior. |

El servidor requiere de los siguientes requerimientos de software:

|  |  |
| --- | --- |
| Sistema Operativo | RedHat 7 |
| Base de datos relacional | PostgreSQL 10 |
| Base de datos de documentos | MongoDB v3 |
| Lenguaje de programación | Nodejs v9 |
| Servidor web | NGINX |

**Nota:**

**1.-** Adicionalmente se requiere que el servidor tenga salida a internet, esto por la descarga de dependencias que existen.

**2.**- El servidor debe permitir el acceso a travéz de los puertos 3000 y 4000.

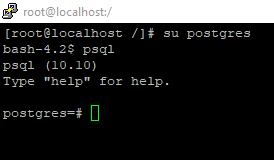
**3.**- Se necesita tener instalado git.

# Creación y configuración de la base de datos.

Paso 1. Conectarse al servidor con el usuario root y abrir una terminal.

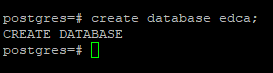
Paso 2. ingresar a postgres para crear la base de datos, ejecutando los siguientes comandos:

|  |
| --- |
| su postgres  psql |



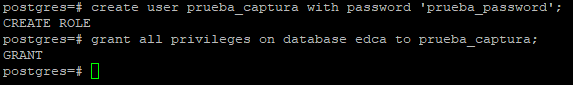
Paso 3. Una vez dentro de postgres, crear la base de datos llamándola “*edca”* utilizando el siguiente comando:

|  |
| --- |
| create database edca; |



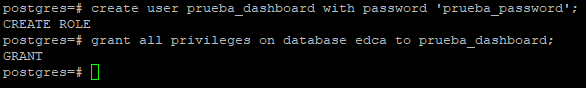
Paso 4. Crear el usuario que se le asignará al schema public de la base y otorgarle todos los permisos sobre la base. Se deben ejecutar los siguientes comandos:

|  |
| --- |
| create user *prueba\_captura* with password 'prueba\_password';  grant all privileges on database edca to *prueba\_captura*; |



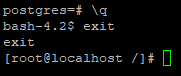
Paso 5. Crear el usuario que se le asignará al schema dashboard de la base y otorgarle todos los permisos sobre la base. Se deben ejecutar los siguientes comandos:

|  |
| --- |
| create user *prueba\_dashboard* with password 'prueba\_password';  grant all privileges on database edca to *prueba\_dashboard*; |



Paso 6. Salir de postgres ejecutando los comandos:

|  |
| --- |
| \q  exit |

****

# Descargar el archivo del código.

Paso 1. Crear la carpeta en donde se realizará la instalación del código ejecutando el siguiente comando:

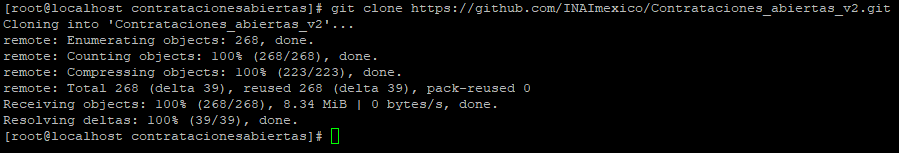
|  |
| --- |
| mkdir /var/www/html/contratacionesabiertas |



Paso 2. Ingresar a la carpeta recién creada y ejecutar el siguiente comando para clonar el repositorio con los archivos de instalación:

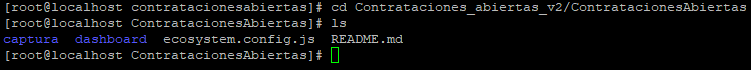
|  |
| --- |
| cd /var/www/html/contratacionesabiertas  git clone –single-branch –branch INAI https://github.com/INAImexico/Contrataciones\_abiertas\_v2.git |





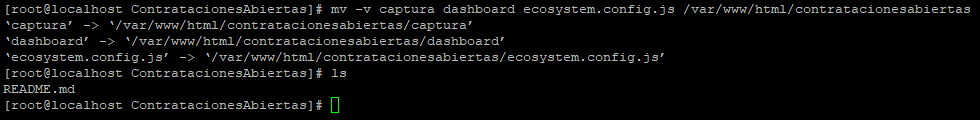
Paso 3. Acceder a la ruta que contiene los directorios *captura* y *dashboard,* ejecutando el siguiente comando:

|  |
| --- |
| cd Contrataciones\_abiertas\_v2/ContratacionesAbiertas |



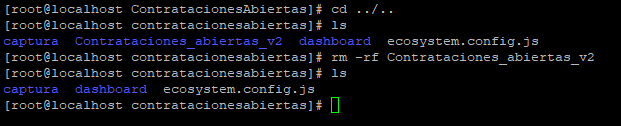
Paso 4. Mover los directorios *captura* y *dashboard* y el archivo *ecosystem.config.js* a la carpeta raíz, ejecutando el siguiente comando:

|  |
| --- |
| mv -v captura dashboard ecosystem.config.js /var/www/html/contratacionesabiertas |



Paso 5. Posicionarse en la carpeta donde se clonó el repositorio para eliminarlo, ejecutando los siguientes comandos.

|  |
| --- |
| cd ../..  rm -rf Contrataciones\_abiertas\_v2 |

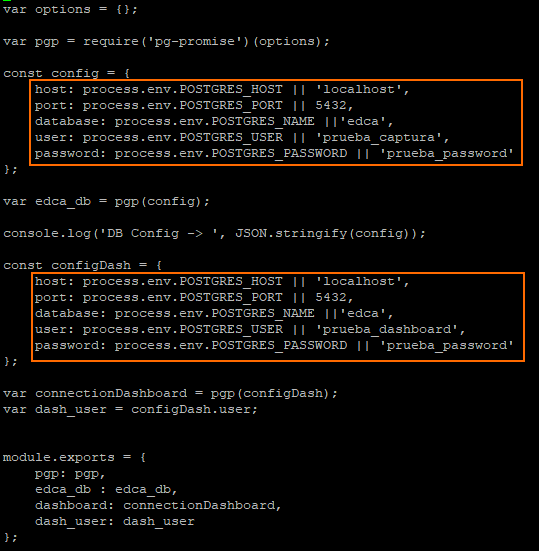
**

# Configuración de variables de conexión a la base de datos

Paso 1. Ingresar a la carpeta /captura/ y editar en el archivo llamado “db\_conf.js” los datos de acuerdo a su configuración propia, HOST, PORT, USER y PASSWORD, para la conexión a la base de datos, los usuarios y contraseñas corresponden a los creados para los schemas public y dashboard respectivamente; ejecutando los comandos siguientes:

|  |
| --- |
| cd captura  vi db\_conf.js |

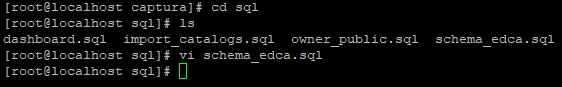
Actualizar, y guardar los cambios

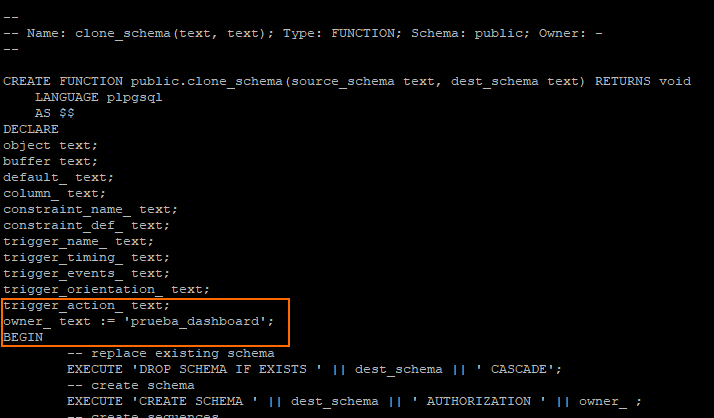


Paso 2. Ingresar a la carpeta /captura/sql/ y editar en cada uno de los archivos “schema\_edca.sql”, “owner\_public.sql” y “dashboard.sql” el nombre del usuario correspondiente de conexión creado para los respectivos schemas, ejecutando los comandos:

|  |
| --- |
| cd captura  cd sql |

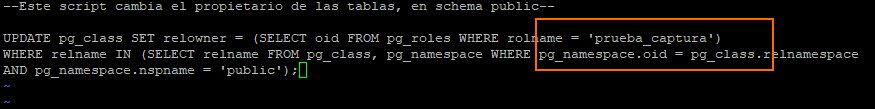
|  |
| --- |
| vi schema\_edca.js |





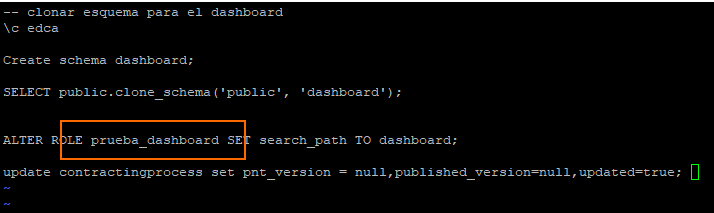
|  |
| --- |
| vi owner\_public.sql |





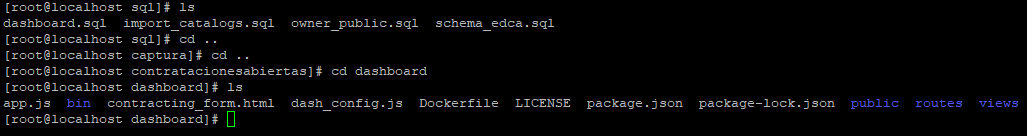
|  |
| --- |
| vi dashboard.sql |





Paso 3. Salir de la carpeta /carpeta/sql/ e ingresar a la carpeta /dashboard/, ejecutar los siguientes comandos:

|  |
| --- |
| cd ..  cd ..  cd dashboard |

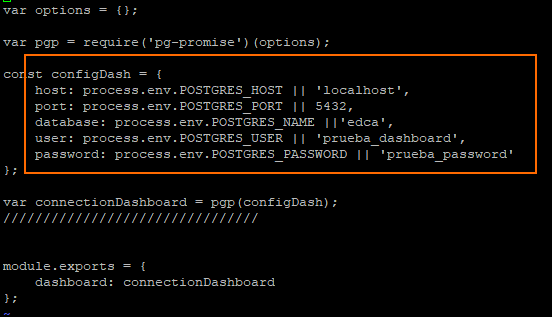


Paso 4. Editar en el archivo llamado “dash\_conf.js” los datos de acuerdo a su configuración propia, HOST, PORT, USER y PASSWORD, para la conexión a la base de datos, el usuario y contraseña corresponde al creado para el schema dashboard; ejecutar el comando siguiente:

|  |
| --- |
| vi dash\_conf.js |



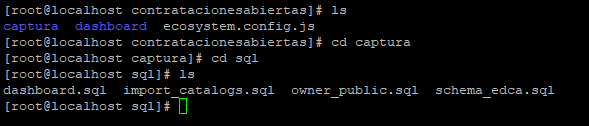
Actualizar, y guardar los cambios



# Ejecutar los script para crear la estructura en la base edca.

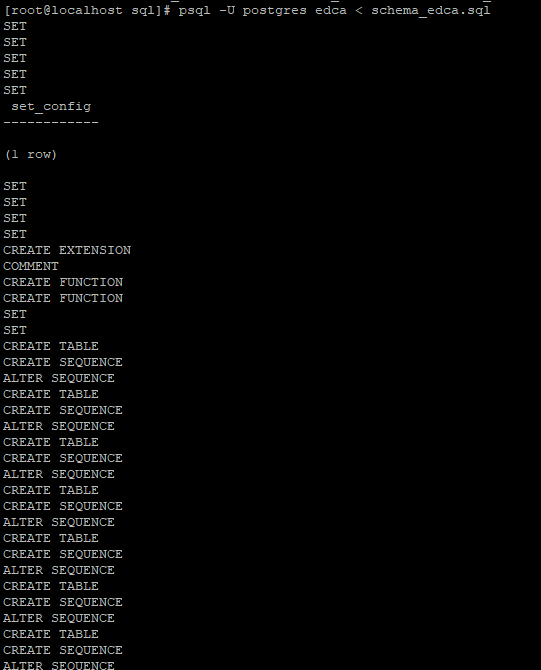
Paso 1. Ingresar a la carpeta /captura/sql/ y ejecutar el siguiente comando para importar el script que contiene la estructura de la base de datos:

|  |
| --- |
| cd captura  cd sql |

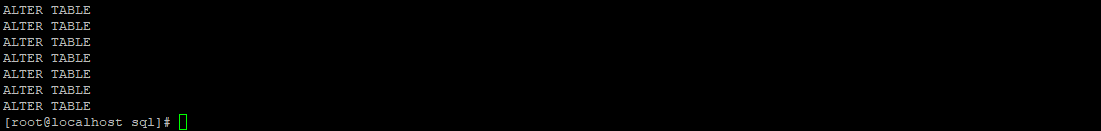


Paso 2. Ejecutar el script schema\_edca.sqlque contiene la estructura de la base de datos (tablas sin catálogos), ejecutar el siguiente comando:

|  |
| --- |
| *psql -U postgres edca < schema\_edca.sql* |



La ejecución de este script no debe de mostrar error.

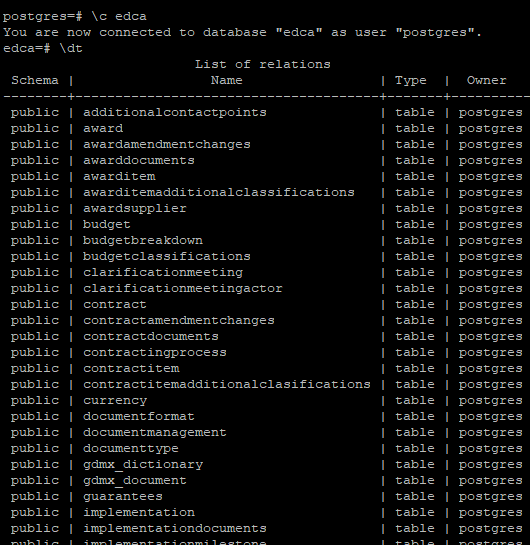


Las tablas creadas con la ejecución del script, por default pertenecen al schema “Public” con el propietario “postgres”. Para el funcionamiento correcto del módulo “Sistema de captura” de la herramienta, se debe de actualizar el propietario de las tablas con el usuario creado para dicho schema, es decir: prueba\_captura.

Ejemplo.

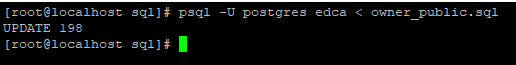
Consulta de la base edca:

* Schema: Public
* Owner: postgres



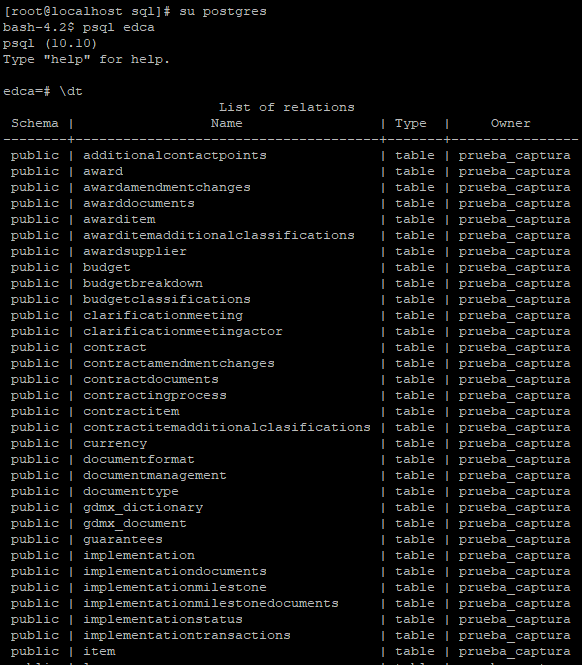
Paso 3. Ejecutar el script owner\_public.sqlque actualiza el propietario de las tablas del schema public, en la base de datos edca, con el siguiente comando:

|  |
| --- |
| *psql -U postgres edca < owner\_public.sql* |



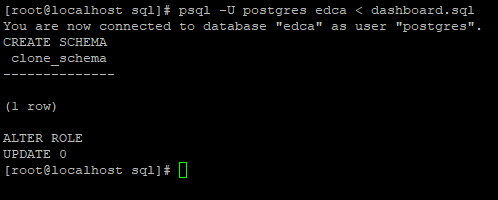
Se actualizó el propietario del schema public.

Ahora al consultar la base edca, el propietario es prueba\_captura:



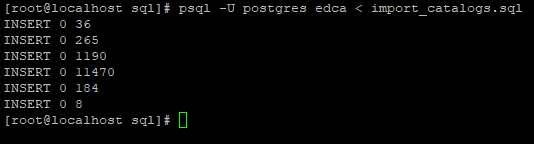
Paso 4. Ejecutar el script dashboard.sql,este script crea el shema dashboard y le asigna los permisos del usuario prueba\_dashboard. Ejecutar el siguiente comando:

|  |
| --- |
| *psql -U postgres edca < dashboard.sql* |



Paso 5. Ejecutar el script import\_catalogs.sqlque inserta los catálogos utilizados en el sistema de captura (moneda, lenguaje, ítems, etc), ejecutar el siguiente comando:

|  |
| --- |
| *psql -U postgres edca < import\_catalogs.sql* |



# Compilación y ejecución en el módulo “Sistema de captura”.

Paso 1. Realizar la compilación y ejecución del código; la ruta donde debe de realizarse, es la siguiente:

|  |
| --- |
| /var/www/html/contratacionesabiertas/captura |

Paso 2. Ejecutar el siguiente comando:

|  |
| --- |
| npm install |

Al ser ejecutado, el resultado no debe de marcar ningún error.

Paso 3. Ingresar a la siguiente ruta:

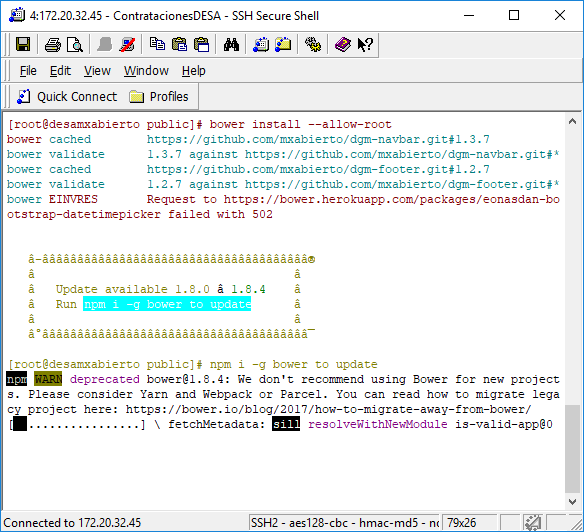
|  |
| --- |
| /var/www/html/contratacionesabiertas/captura/public |

Paso 4. Ejecutar el siguiente comando para la instalación de dependencias del sistema por medio de bower:

|  |
| --- |
| bower install --allow-root |

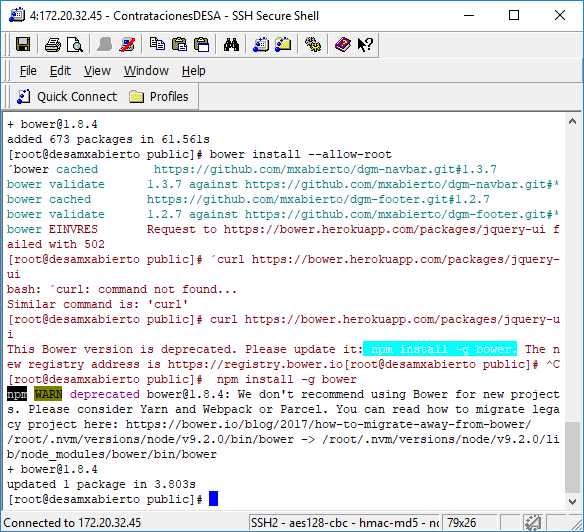
Paso 5. Ejecutar el siguiente comando para la instalación de bower:

|  |
| --- |
| npm install -g bower |





# Actualización de versión de Bower.



**Nota:** En el caso de que aparezcan alguna pantalla similar a la anterior, quiere decir que nuestra versión de bower esta desactualizada, por lo que es necesario actualizar, lo cual se realiza de la siguiente manera:

Paso 1. Desinstalar el bower de /usr/bin, ejecutar el siguiente comando:

|  |
| --- |
| npm uninstall bower -g --prefix=/usr |

Paso 2. Ejecutar nuevamente el siguiente comando:

|  |
| --- |
| bower install --allow-root |

Paso 3. Ejecutar nuevamente el siguiente comando:

|  |
| --- |
| npm install -g bower |

Paso 4. Ejecutar el siguiente comando para conocer el repositorio raíz de npm:

|  |
| --- |
| npm list -g --depth=0 |

Este comando arrojara una salida como la siguiente:

**/home/inai/.npm-data/lib**

**├── bower@1.8.4**

**├── npm@6.5.0**

**└── pm2@3.2.2**

Paso 5. Sustituir la palabra lib del final, por bin

**/home/inai/.npm-data/bin**

Paso 6. Agregar esa ruta al PATH con el siguiente comando:

|  |
| --- |
| export PATH=$PATH: /home/inai/.npm-data/bin |

Es importante mencionar que la ruta que se menciona en el paso anterior es de ejemplo, se tiene que dejar la que arroje de salida el comando npm list -g –depth=0

# Validación de conexión a base de datos.

Paso 1. Validar que la conexión a base de datos sea correcta, ingresar a la ruta:

|  |
| --- |
| /var/www/html/contratacionesabiertas/captura/ |

Paso 2. Abrir el archivo llamado:

|  |
| --- |
| db.js |

Paso 3. La siguiente variable, debe estar configurada como se ve en la siguiente línea:

**var linkedStorage = (process.env.MONGODB\_PORT\_27017\_TCP\_ADDR || 'localhost' )**

Otra opción es setear la variable de ambiente "POSTGRES\_HOST", utilizando el comando siguiente:

|  |
| --- |
| export POSTGRES\_HOST=localhost |

# Asignación de variables para conexión a la API de PNT.

Es necesario validar que la conexión a la API de PNT sea correcta.

Paso 1. Ingresar a la carpeta “pnt”, ubicada en la ruta siguiente:

|  |
| --- |
| /var/www/html/contratacionesabiertas/captura/pnt/ |

Paso 2. Abrir el archivo llamado:

|  |
| --- |
| process-pnt.js |

Paso 3. En la variable EMAIL se configura el email de la unidad correspondiente y debe de configurarse como se ve en la siguiente línea:

**const EMAIL = process.env.PNT\_EMAIL || 'correo@institucional';**

Paso 4. Abrir el archivo llamado:

|  |
| --- |
| api-pnt.js |

Paso 5. En las siguientes variables se configuran las credenciales y debe de configurarse como se ve en las siguientes líneas:

**const** **URL = process.env.PNT\_URL || 'sitio\_pruebas\_pnt:puerto';**

**const** **USER = process.env.PNT\_USER || 'correo@institucional';**

**const PASS = process.env.PNT\_PASS || 'contraseña';**

**Nota:** La conexión está configurada para enviar información del artículo 70, fracción 28 para los formatos A y B de la ley general de transparencia. El correo electrónico, URL de PNT y contraseña son accesos que deberán gestionarse con su órgano garante local.



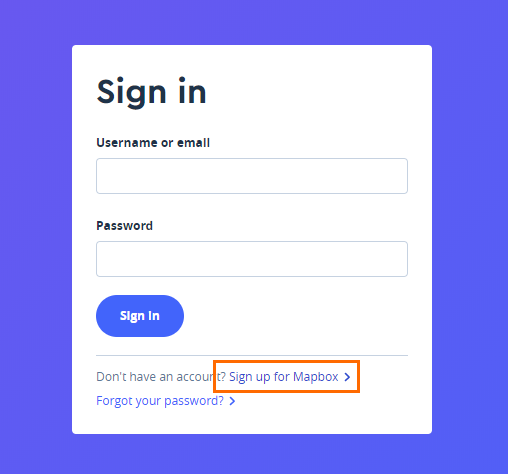
# Asignación de variables MAPTOKEN.

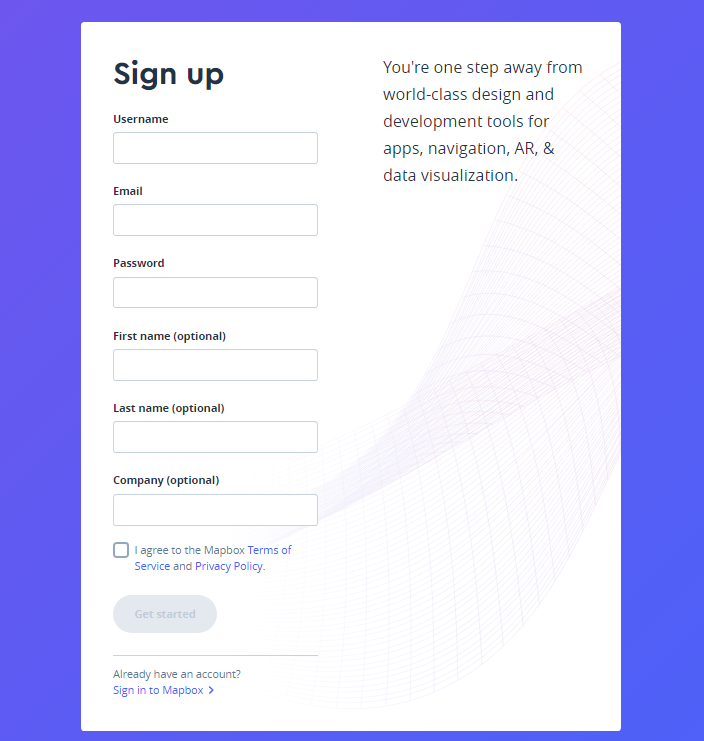
Para el servicio de Mapas es necesario configurar lo siguiente:

Paso 1. Para obtener el token es necesario ingresar a la siguiente URL:

<https://www.mapbox.com/signup/>

Donde hay que registrarse.

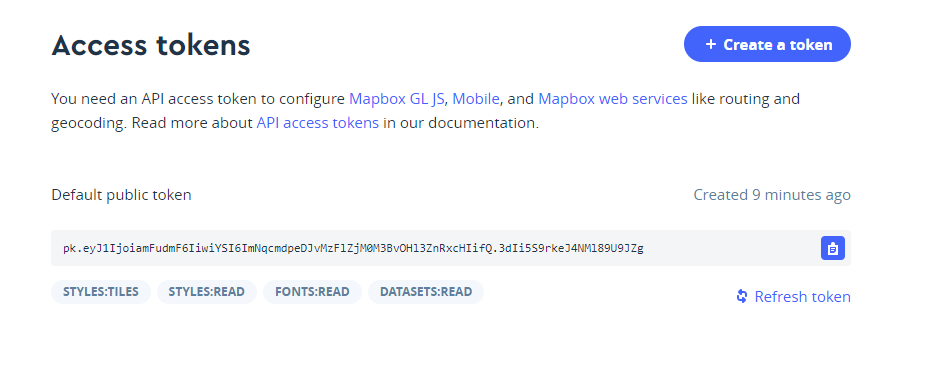




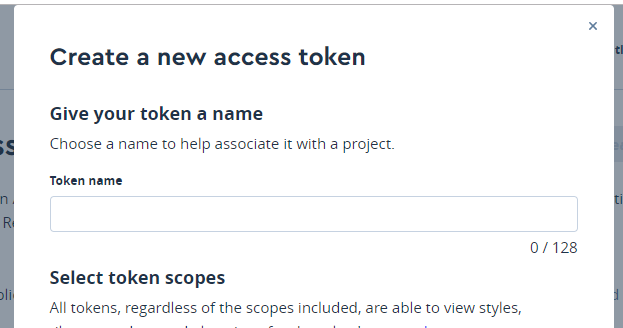
Paso 2. Una vez realizado el registro, ingresar a la página y le damos clic en la opción:



Y veremos la siguiente pantalla:

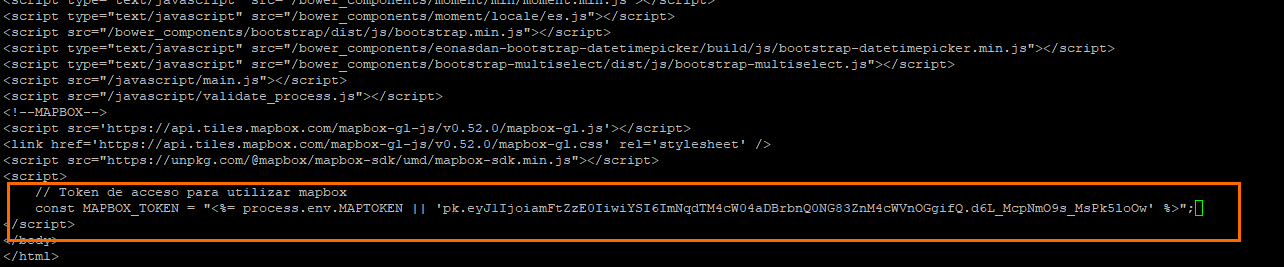


Paso 3. Copiar el valor del token que se encuentra ahí o crear uno asociándole un nombre



Paso 4. Ingresar a la carpeta /captura/views/ y abrir el archivo “main.ejs” para editar el valor MAPTOKEN, utilizando los siguientes comandos:

|  |
| --- |
| cd captura  cd views  vi main.ejs |



Sustituir con el valor del token

|  |  |
| --- | --- |
| Variable | Valor |
| MAPTOKEN | Token de Mapbox |

Guardar el cambio y salir.

# Compilación y ejecución en el módulo “Dashboard - visualizador”.

Paso 1. Ingresar a la siguiente ruta:

|  |
| --- |
| /var/www/html/contratacionesabiertas/dashboard |

Paso 2. Ejecutar el siguiente comando:

|  |
| --- |
| npm install |

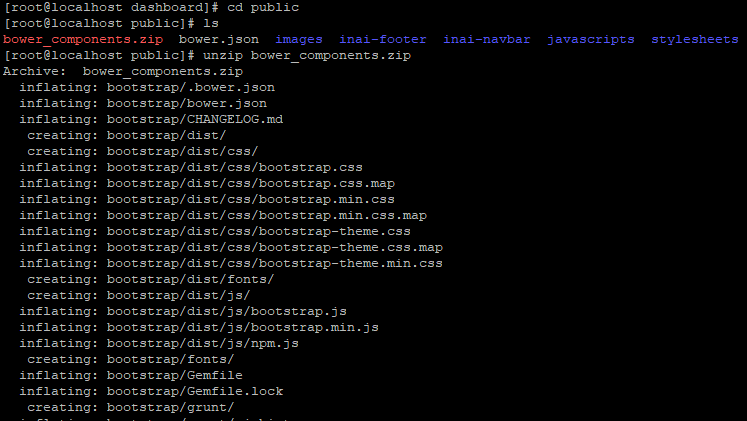
Al ser ejecutado, el resultado no debe de marcar ningún error.

Paso 3. Ingresar a la siguiente ruta:

|  |
| --- |
| /var/www/html/contratacionesabiertas/dashboard/public |

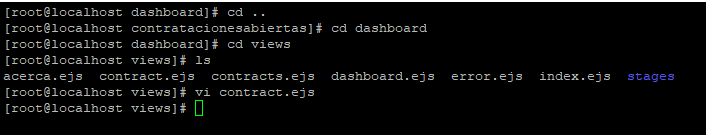
Paso 4. Ejecutar el siguiente comando para descromprimir las dependencias del sistema por medio de bower:

|  |
| --- |
| unzip bower\_components.zip |



Paso 5. Ingresar a la carpeta /dashboard/views/ y abrir el archivo “contract.ejs” para editar el valor MAPTOKEN, utilizando los siguientes comandos:

|  |
| --- |
| cd dashboard  cd views  vi contract.ejs |

****

Sustituir con el valor del token

|  |  |
| --- | --- |
| Variable | Valor |
| MAPTOKEN | Token de Mapbox |



Guardar el cambio y salir.

Paso 6. Abrir el archivo “contracts.ejs” para editar el valor CAPTURE\_URL, utilizando el siguiente comando:

|  |
| --- |
| vi contracts.ejs |

****

Sustituir con la IP del servidor del módulo de captura, debe tener el puerto 3000.

|  |  |
| --- | --- |
| Variable | Valor |
| CAPTURE\_URL | ip\_servidor:3000 |

Guardar el cambio y salir.

# Crear usuario del aplicativo.

Paso 1. Para crear el usuario del aplicativo, ir a la carpeta “captura”, ubicada en la siguiente ruta: /var/www/html/contratacionesabiertas/ y ejecutar el script para la creación del usuario administrador del sistema de captura.

|  |
| --- |
| ./useradm add *username* |

Donde “username” se actualiza por su usuario administrador que deseen.

**Nota:** El archivo *useradm* debe tener permisos de ejecución, en caso de no tenerlos, ejecutar el siguiente comando dentro de la carpeta captura:

|  |
| --- |
| chmod u+x useradm |

**Nota:** Los campos deben cumplir las siguientes características:

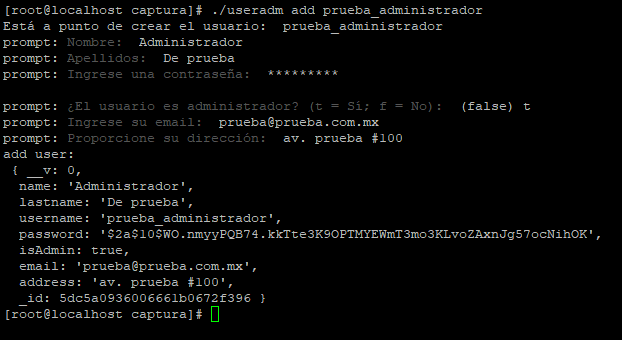
* Nombre: solo se permiten caracteres alfabéticos. Los nombres para los usuarios no se pueden repetir.
* Apellido: solo se permiten caracteres alfabéticos.
* Contraseña: se permiten caracteres alfanuméricos y espacios.
* ¿El usuario es un administrador?: este campo permite elegir el tipo de usuario que se

pretende crear. Los tipos de usuario son: administrador y usuario del sistema.  
1) El usuario administrador tiene acceso a funcionalidades de creación de usuarios desde la herramienta, puede capturar información, eliminar usuarios

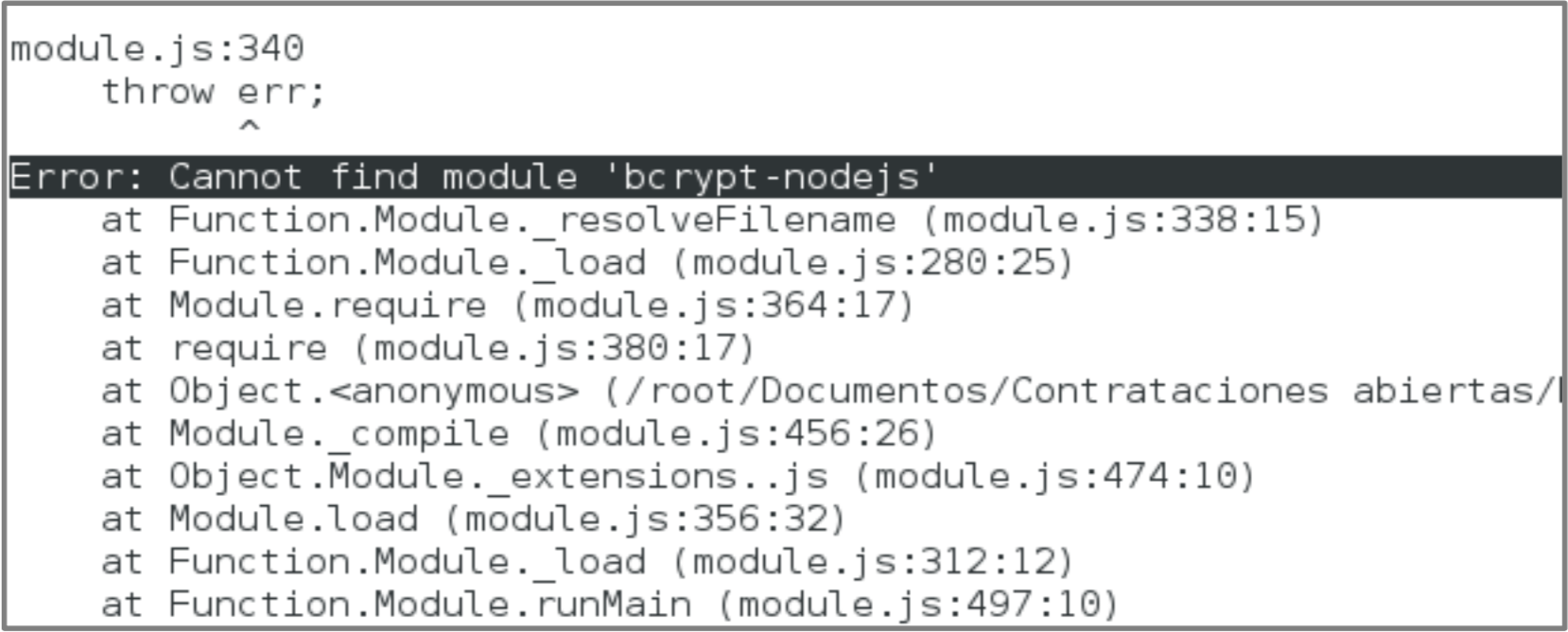
y asignar permisos sobre procesos de contratación.

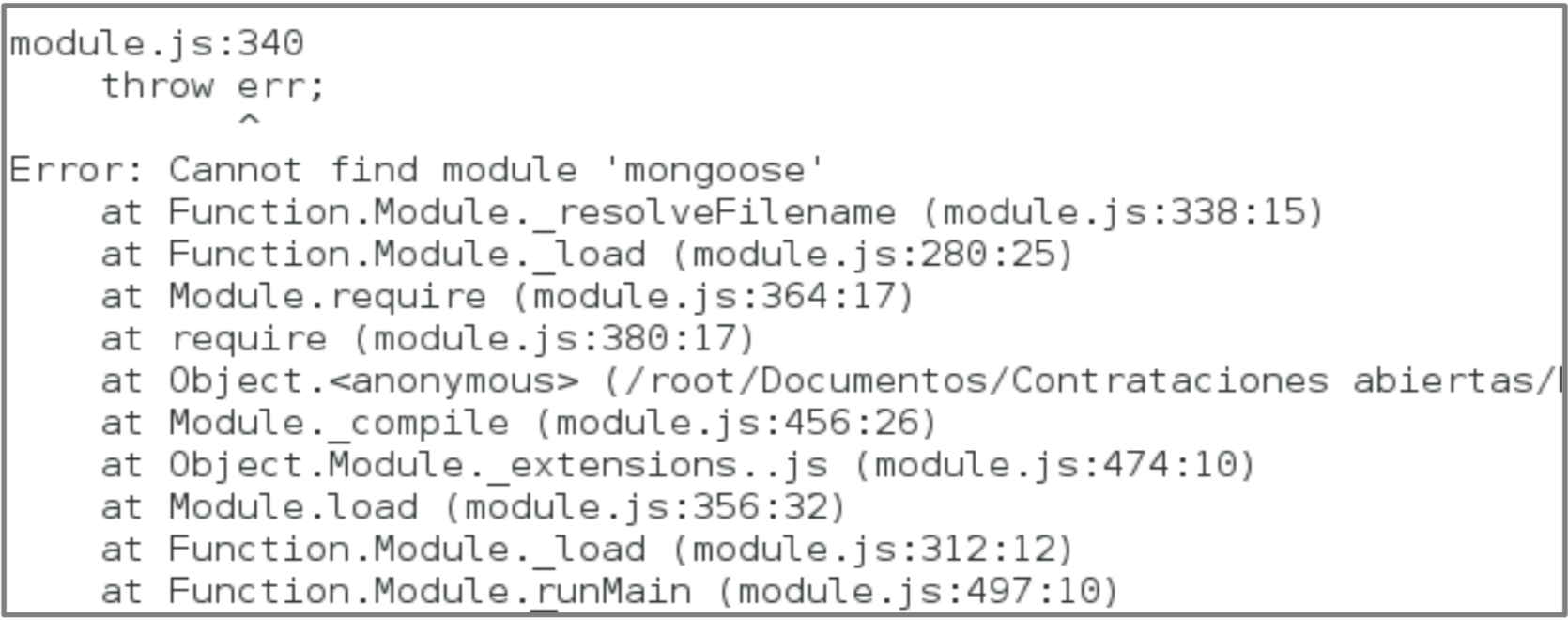
2) El usuario del sistema solo tiene acceso para capturar información.

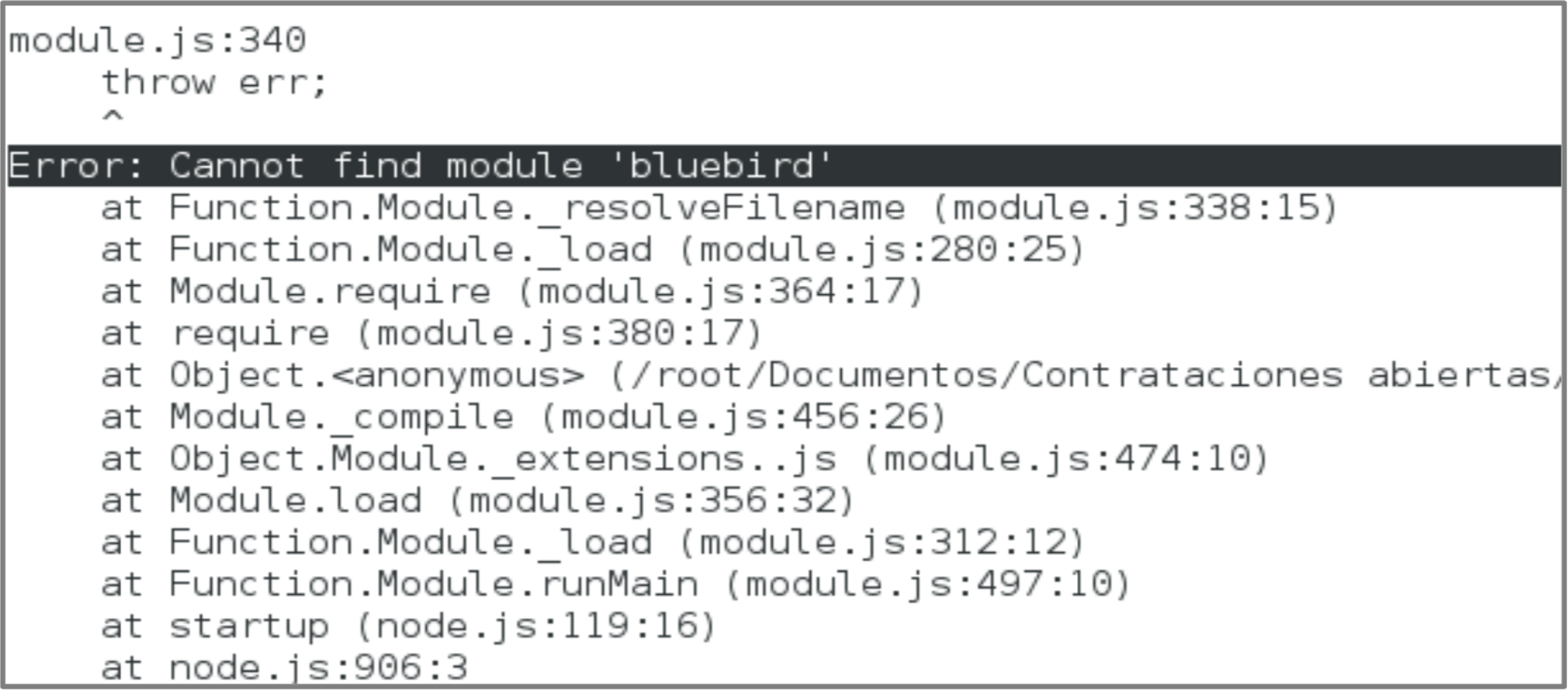
* Correo: debe presentar la estructura de un correo válido como la que se muestra a continuación; user@example.com
* Dirección: se permite ingresar la dirección completa incluyendo calle, número, municipio, etc.



**Importante:** Si al crear un usuario salen los siguientes errores:







Se deben instalar los siguientes módulos:

**Bycrypt-nodejs**

|  |
| --- |
| npm install bycrypt-nodejs |

**Bluebird**

|  |
| --- |
| npm install bluebird |

**Mongoose**

|  |
| --- |
| npm install mongoose |

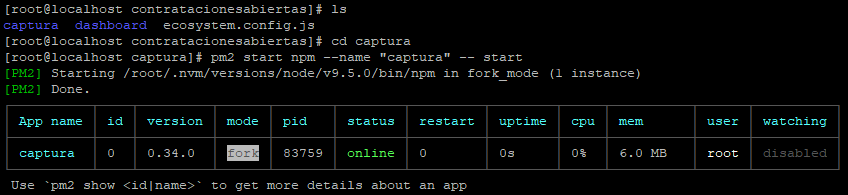
# Iniciar los servicios de los módulos

Paso 1. Ingresar a la carpeta captura, utilizando el siguiente comando:

|  |
| --- |
| cd captura |

Paso 2. Iniciar el servicio del módulo “Sistema de captura”, ejecutando el siguiente comando:

|  |
| --- |
| pm2 start npm --name "captura" -- start |

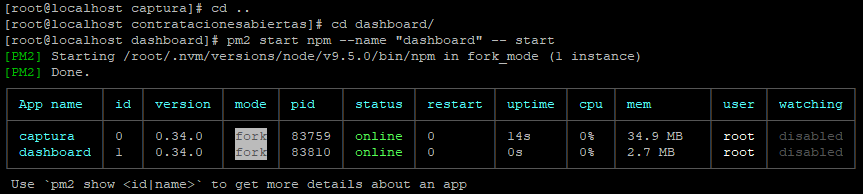


Paso 3. Salir de la carpeta captura e ingresar a la carpeta dashboard, ejecutando los siguientes comandos:

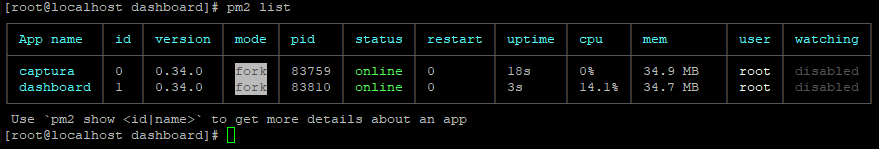
|  |
| --- |
| cd ..  cd dashboard |

Paso 4. Iniciar el servicio del módulo “Dashboard - visualizador”, ejecutando el siguiente comando:

|  |
| --- |
| pm2 start npm --name "dashboard" -- start |



Paso 5. Ejecutar el comando pm2 list para ver el estado de los servicios y que estén correctos.



# Acceder a los módulos.



# Módulo Sistema de captura

Ingresar a la siguiente dirección:

|  |
| --- |
| http://url\_del\_servidor:3000 |



Ingresar con el usuario y contraseña creados en el paso 5



# Módulo de visualización (dashboard).

Ingresar a la siguiente dirección:

|  |
| --- |
| http://url\_del\_servidor:4000/contratacionesabiertas/contratos |



