**Contenido**

[Control de cambios](#_8l1840k19k6o) 4

[1. Arquitectura](#_h9doy5tr97ap) 5

[2. Preparación de la actualización](#_3znysh7) 6

[2.1 Conectarse al servidor](#_2et92p0) 6

[2.2 Realizar un respaldo “backup” de la base de datos edca](#_tyjcwt) 6

[2.3 Detener los servicios de los módulos “Dashboard” y “Captura”](#_3dy6vkm) 6

[2.4 Realizar “backup” de código de los módulos “Dashboard” y “Captura”](#_1t3h5sf) 7

[2.5 Descargar la actualización del código](#_4d34og8) 8

[3. Ejecución de la actualización](#_2s8eyo1) 9

[3.1 Actualización de la estructura de la base de datos](#_wozrbvgqt1dn) 9

[3.2 Actualización de tablas en esquema “dashboard”](#_3rdcrjn) 11

[4. Configurar las variables de usuario y password “development”](#_26in1rg) 12

[5. Instalación de ORM, para migración de tablas del módulo de Infraestructura abierta](#_a3bh08qhoqkn) 13

[5.1 Migración de las tablas para el del módulo de infraestructura abierta](#_ivvkmlr1g2ap) 13

[6. Configuración de variables de conexión a base de datos](#_z337ya) 15

[6.1 Configuración de variable de conexión “mongo”](#_ymztrxrkd9i5) 18

[6.2 Asignación de variables para conexión a la API de PNT](#_3j2qqm3) 18

[6.3 Asignación de variables MAPTOKEN](#_1y810tw) 20

[6.4 Actualización de dashboard](#_2xcytpi) 22

[7. Iniciar los servicios de los módulos](#_44sinio) 24

[8. Acceder a los módulos](#_3whwml4) 25

[8.1 Módulo de captura](#_2bn6wsx) 25

[8.2 Módulo de visualización (dashboard)](#_qsh70q) 26

[Anexos.](#_3c520ci80j5t) 28

[Anexo 1. Errores comunes.](#_qatxi9pcyck4) 28

[Anexo 2. Configuraciones Generales para más de una instancia en el mismo servidor.](#_4r0lqgwfpo83) 30

[Anexo 3. Re-publicación de procesos de contratación.](#_lhglxcq52cfy) 34

# 

*Fecha de actualización: 14-Mayo-2021*

# Control de cambios

|  |  |  |
| --- | --- | --- |
| **Fecha** | **Versión** | **Resumen del cambio** |
| 23/12/2020 | 2.1.0 | Integración del módulo de Infraestructura Abierta y la funcionalidad de interconexión con la PNT. |
| 16/02/2021 | 2.1.2 | Integración de la sección “Datos Abiertos”. |
| 14/05/2021 | 2.1.4 | Integración de la descarga del formato CSV en la sección “Datos Abiertos”.  En el módulo de Sistema de captura, se habilita un formulario para registro del periodo (año) a publicar y un formulario para la edición de la sección “Política de publicación” con el usuario administrador y en formato html.  En la gráfica “Información general de contratos y proveedores” se actualiza el orden de los filtros y la visualización de las opciones dentro del botón Filtrar. |

# 

# 1. Arquitectura

La herramienta necesita los siguientes requerimientos para su funcionamiento:

Hardware

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

Software:

|  |  |
| --- | --- |
| Sistema Operativo | CentOS 7 |
| Base de datos relacional | PostgreSQL 10 |
| Base de datos de documentos | MongoDB v3 |
| Lenguaje de programación | Nodejs v10 |
| Servidor web | NGINX |

**NOTA:**

1. Se requiere que el servidor tenga salida a internet, esto permitirá la descarga de dependencias que existen para el funcionamiento de las gráficas del visualizador.
2. El servidor requiere de la configuración de dos puertos para permitir el acceso a los módulos de captura y dashboard. Actualmente, usa el puerto *3000* para el *sistema de captura* y el 4000 para el *dashboard,* sin embargo,estos puertos son parametrizables con la finalidad de poder tener más de una instancia en el mismo servidor.

Para configurar más de una instancia en el servidor, revisar el **Anexo 2**, al final de este documento.

1. Se necesita tener instalado el software: git y pm2. Para mayor referencia se pueden consultar las siguientes ligas: <https://desktop.github.com/>, <https://git-lfs.github.com/> y https://www.npmjs.com/package/pm2.
2. En PostgreSQL, se debe de configurar los métodos de conexión **local** e **IPv4** como **trust,** en el archivo llamado **pg\_hba.conf**

# 2. Preparación de la actualización

Para realizar la actualización del código, se deben de seguir los siguientes pasos:

## 2.1 Conectarse al servidor

**Paso 1.** Ingresar al servidor con el usuario ***root.***

## 2.2 Realizar un respaldo “backup” de la base de datos edca

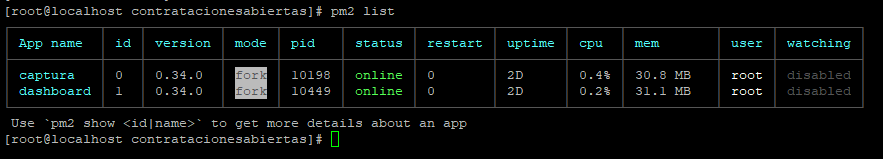
**Paso 1.** Ubicarse en la carpeta donde se desee guardar el archivo sql de respaldo. Ejecutar el siguiente comando para exportar la base de datos completa (estructura y datos) edca.

|  |
| --- |
| *pg\_dump -U postgres edca > respaldo.sql* |

## 2.3 Detener los servicios de los módulos “Dashboard” y “Captura”

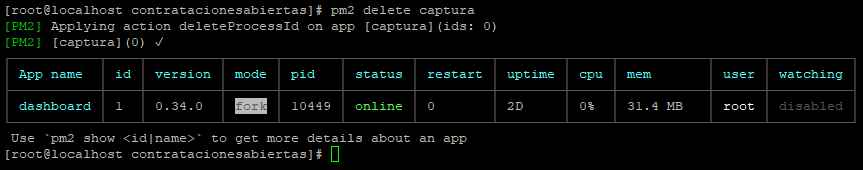
**Paso 1.** Ejecutar el siguiente comando para listar los servicios que se encuentran iniciados:

|  |
| --- |
| *pm2 list* |

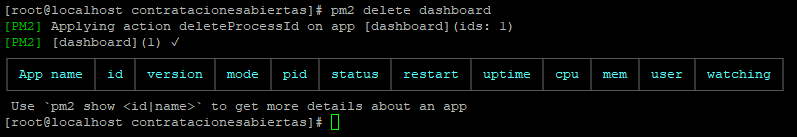


**Paso 2.** Borrar los servicios iniciados ejecutando un comando por servicio iniciado:

|  |
| --- |
| *pm2 delete captura* |



|  |
| --- |
| *pm2 delete dashboard* |



## 2.4 Realizar “backup” de código de los módulos “Dashboard” y “Captura”

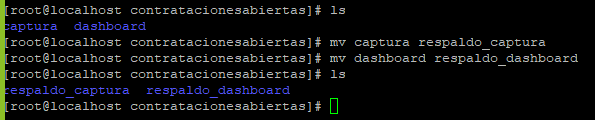
**Paso 1.** Posicionarnos en la ruta en donde se encuentra instalado el código del aplicativo, ejecutando el siguiente comando:

|  |
| --- |
| */var/www/html/contratacionesabiertas* |

**Paso 2.** Ejecutar las siguientes instrucciones:

|  |
| --- |
| *mv captura <Nombre del folder backup captura>*  *mv dashboard <Nombre del folder backup dashboard>* |

Como se muestra en el siguiente ejemplo:

****

## 2.5 Descargar la actualización del código

**Paso 1.** Clonar el repositorio GitHub con los archivos del código actualizado, ejecutando el siguiente comando:

|  |
| --- |
| *git clone https://github.com/datosabiertosmx/contrataciones-abiertas-infraestructura.git* |

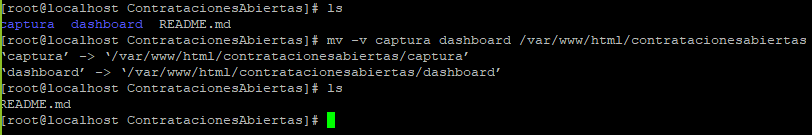
**Paso 2.** Una vez descargado el código, acceder a la ruta que contiene los directorios *captura* y *dashboard.*

|  |
| --- |
| *cd contrataciones-abiertas-infraestructura/contratacionesabiertas/* |



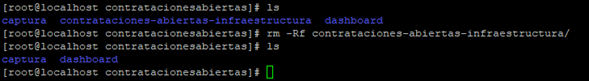
**Paso 3.** Mover los directorios *captura* y *dashboard* a la carpeta raíz, ejecutando el siguiente comando:

|  |
| --- |
| *mv -v captura dashboard /var/www/html/contratacionesabiertas* |



**Paso 4.** Posicionarse en la carpeta donde se clonó el repositorio para eliminarlo, ejecutando los siguientes comandos:

|  |
| --- |
| *cd ../..*  *rm -rf contrataciones-abiertas-infraestructura/* |



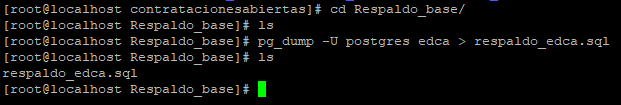
# 3. Ejecución de la actualización

## 3.1 Actualización de la estructura de la base de datos

**Para conservar los datos registrados por los usuarios, es necesario realizar los siguientes pasos:**

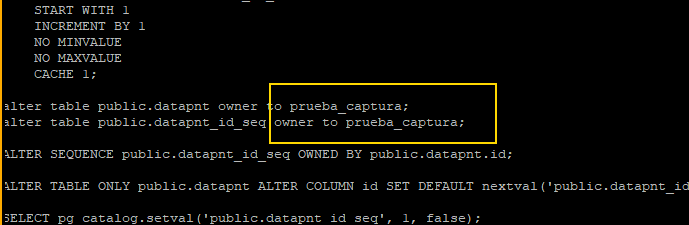
**Paso 1. Respaldar la base de datos actual.** Ubicarse en la carpeta donde se almacenará el respaldo y ejecutar el comando:

|  |
| --- |
| *pg\_dump -U postgres edca > respaldo\_edca.sql* |



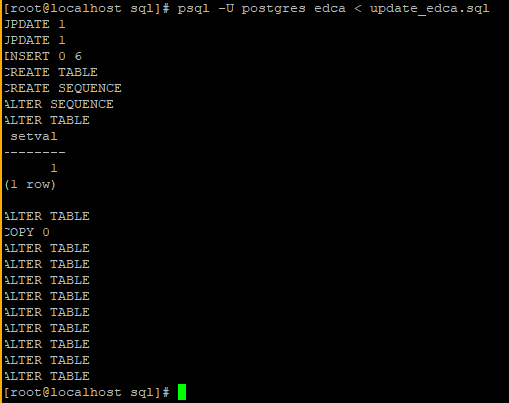
**Paso 2.** Ingresar a la carpeta /captura/sql/ y editar el script “**update\_edca.sql**”. Actualizar el usuario “**prueba\_captura**” por el usuario creado para el esquema public.

*> vi update\_edca.sql*



**Paso 3.** Ejecutar el script “**update\_edca.sql**” con el comando:

*> psql -U postgres edca < update\_edca.sql*

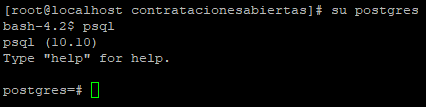


La ejecución de este script **no debe de mostrar errores**.

## 3.2 Actualización de tablas en esquema “dashboard”

**Paso 1.** Ingresar al gestor de bases de datos postgresql, ejecutando el comando:

|  |
| --- |
| *> su postgres*  *> psql* |

****

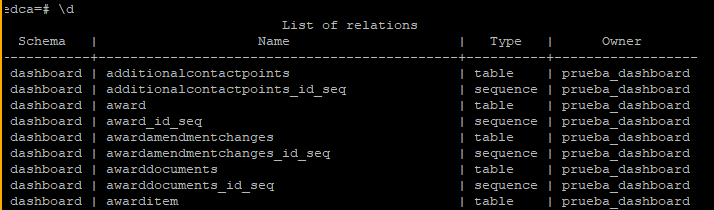
**Paso 2.** Seleccionar el esquema **dashboard**, ejecutando el comando:

|  |
| --- |
| *> set search\_path to dashboard;* |

****

**Paso 3.** Verificar que se visualice el esquema **dashboard**, ejecutando el comando:

|  |
| --- |
| *> \d* |

****

**Paso 4.** Ejecutar las siguientes sentencias de modificación a las tablas en el esquema dashboard:

*ALTER TABLE budgetbreakdown ADD COLUMN origin varchar;*

*ALTER TABLE budgetbreakdown ADD COLUMN fund\_type varchar;*

*ALTER TABLE roles RENAME COLUMN clarificationmeetingattendee TO attendee;*

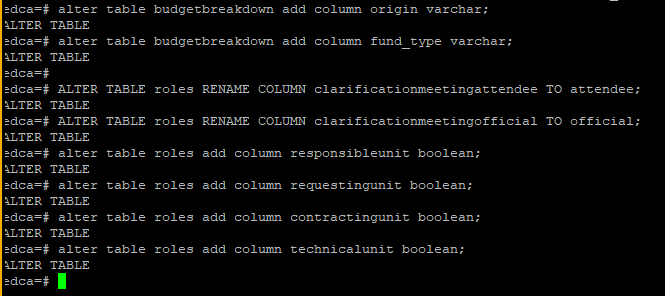
*ALTER TABLE roles RENAME COLUMN clarificationmeetingofficial TO official;*

*ALTER TABLE roles ADD COLUMN responsibleunit boolean;*

*ALTER TABLE roles ADD COLUMN requestingunit boolean;*

*ALTER TABLE roles ADD COLUMN contractingunit boolean;*

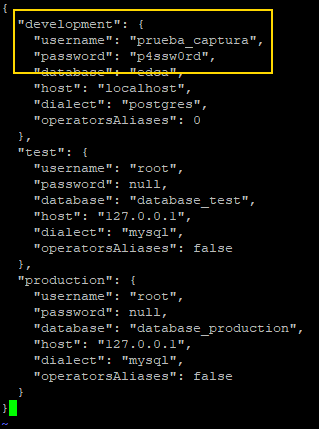
*ALTER TABLE roles ADD COLUMN technicalunit boolean;*



Ejemplo de la ejecución, dicha ejecución de estas sentencias **no debe de marcar ningún error.**

# 4. Configurar las variables de usuario y password “*development*”

**Paso 1**. Ingresar a la carpeta **/captura/config/** y editar el archivo con el usuario correspondiente al esquema public y guardar los cambios.

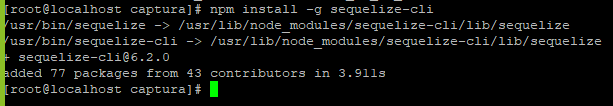


**NOTA:** Si el servidor se configuró previamente con el ORM, ir al paso 5.1, en caso de no tener esta configuración ejecutar los pasos del numeral 5.

# 5. Instalación de ORM, para migración de tablas del módulo de Infraestructura abierta

**Paso 1**. Ingresar a la carpeta **/captura/** y ejecutar los comandos:

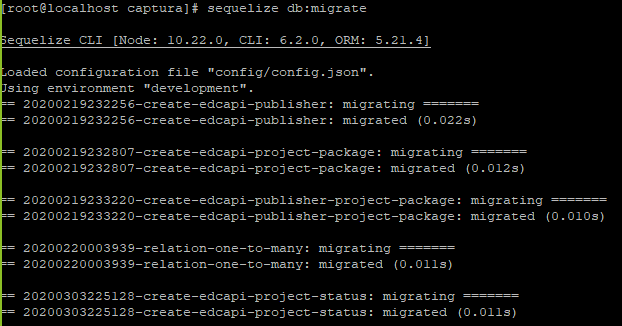
*> npm install -g sequelize-cli*



## 5.1 Migración de las tablas para el del módulo de infraestructura abierta

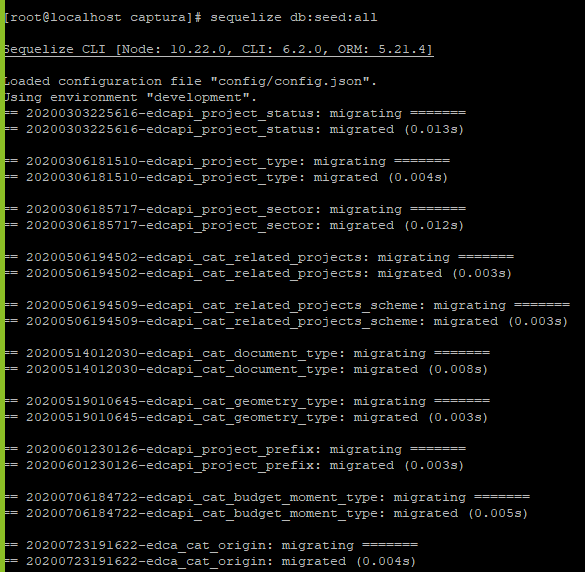
**Paso 1.** Ingresar a la carpeta **/captura/** y ejecutar el siguiente comando:

*> sequelize db:migrate*



**Paso 2.** Dentro de la misma carpeta **/captura/** ejecutar el siguiente comando:

*> sequelize db:seed:all*



La ejecución de estos comandos no debe de mostrar ningún error.

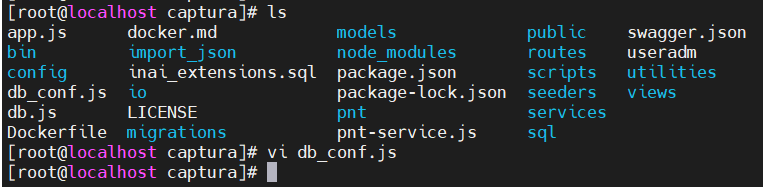
**NOTA:** En caso de que se desee deshacer la migración de las tablas, se debe de ejecutar el siguiente comando:

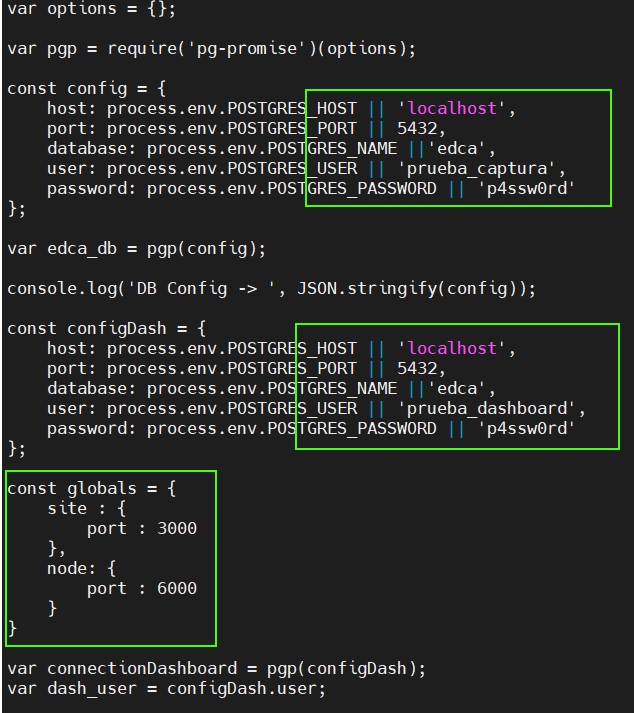
*> sequelize db:migrate:undo:all*

# 6. Configuración de variables de conexión a base de datos

**Paso 1.** Dentro de la carpeta **/captura/** editar el archivo “**db\_conf.js**”, actualizar los datos de acuerdo a su configuración: HOST, PORT, DATABASE, USER y PASSWORD, para la conexión a la base de datos, los usuarios y contraseñas corresponden a los creados para los esquemas public y dashboard respectivamente; y configurar las “**const globals**” con los puertos designados para la instalación, ejecutando los comandos siguientes:

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



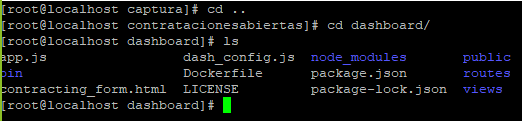


Actualizar, y guardar los cambios con el siguiente comando:

*> :wq*

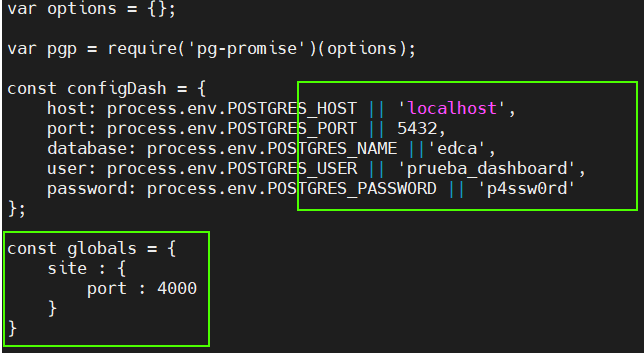
**Paso 2.** Salir de la carpeta **/carpeta/** e ingresar a la carpeta **/dashboard/**, ejecutar los siguientes comandos:

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



**Paso 3.** Editar en el archivo llamado “**dash\_conf.js**” los datos de acuerdo a su configuración propia, HOST, PORT, DATABASE, USER y PASSWORD, para la conexión a la base de datos, el usuario y contraseña corresponde al creado para el schema dashboard; y configurar las “**const globals**” con los puertos designados para la instalación, ejecutar el comando siguiente:

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



Actualizar, y guardar los cambios con el comando:

> *:wq*

## 6.1 Configuración de variable de conexión “mongo”

Es necesario validar que la conexión a base de datos sea correcta.

**Paso 1.** Ingresar a la carpeta “**captura**” en la siguiente ruta:

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

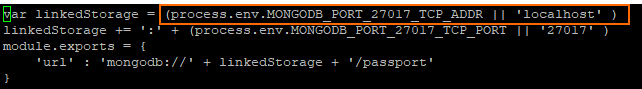
**Paso 2.** Abrir el archivo **db.js**



La variable **linkedStorage** debe estar de la siguiente manera:

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

Ejemplo:



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

Es necesario validar que la conexión a la API de PNT sea correcta, esto se realiza en los archivos:

* **process-pnt.js**
* **api-pnt.js**

**Paso 1.** Ingresar a la siguiente ruta:

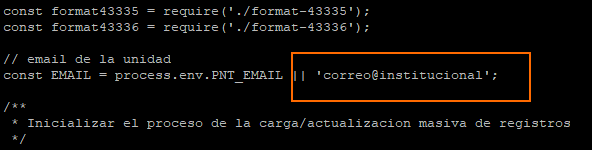
|  |
| --- |
| cd /var/www/html/contratacionesabiertas/captura/pnt/ |

**Paso 2.** Abrir el archivo process-pnt.js

|  |
| --- |
| *vi process-pnt.js* |

La variable EMAIL se debe de configurar con el email de la unidad, dejarlo como se ve en la siguiente línea:

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



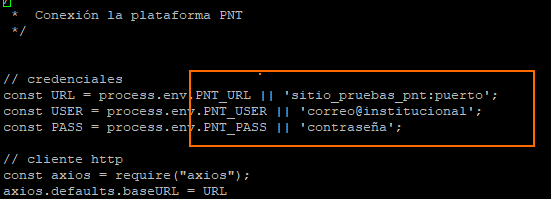
Guardar los cambios y salir del archivo.

**Paso 3.** Abrir el archivo api-pnt.js

|  |
| --- |
| *vi api-pnt.js* |

En las siguientes variables se configuran las credenciales correspondientes de acuerdo a sus datos propios de configuración:

|  |
| --- |
| **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 funcionalidad de interconexión de la herramienta Contrataciones Abiertas contempla el envío de información de los formatos de la fracción XXVIII, del artículo 70, de la Ley General de Transparencia y Acceso a la Información Pública.

El correo electrónico, URL de PNT y contraseña son accesos que deberán gestionarse con el órgano garante local.

Es importante que, previo al envío de información a la PNT productiva, se considere un tiempo, como parte del proceso de implementación, para llevar a cabo pruebas de envío de información de Contrataciones Abiertas al sistema pre-productivo de la PNT, que existe para dicho propósito. De esta manera, se podrá validar la configuración de los formatos, el correcto envío de los datos, y en su caso, podrán realizarse ajustes a la configuración de los formatos para cumplir con las particularidades que existan como parte de la normativa local en la materia.

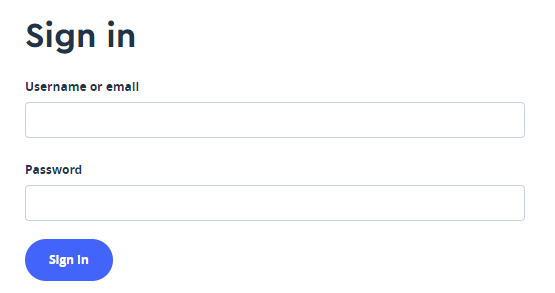
## 6.3 Asignación de variables MAPTOKEN

Para el servicio de Mapas es necesario configurar lo siguiente:

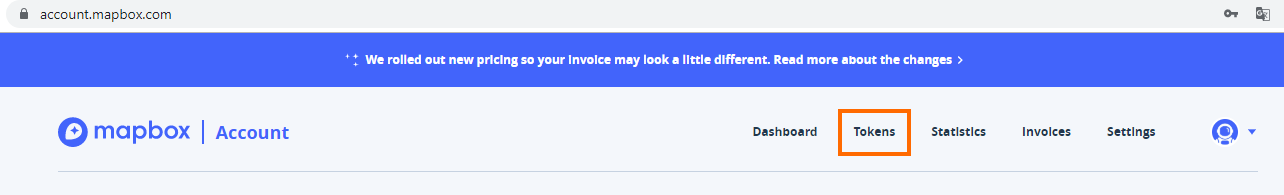
**Paso 1.** Para obtener el token es necesario ingresar a la siguiente ruta:

[*https://www.mapbox.com/signup/*](https://www.mapbox.com/signup/)

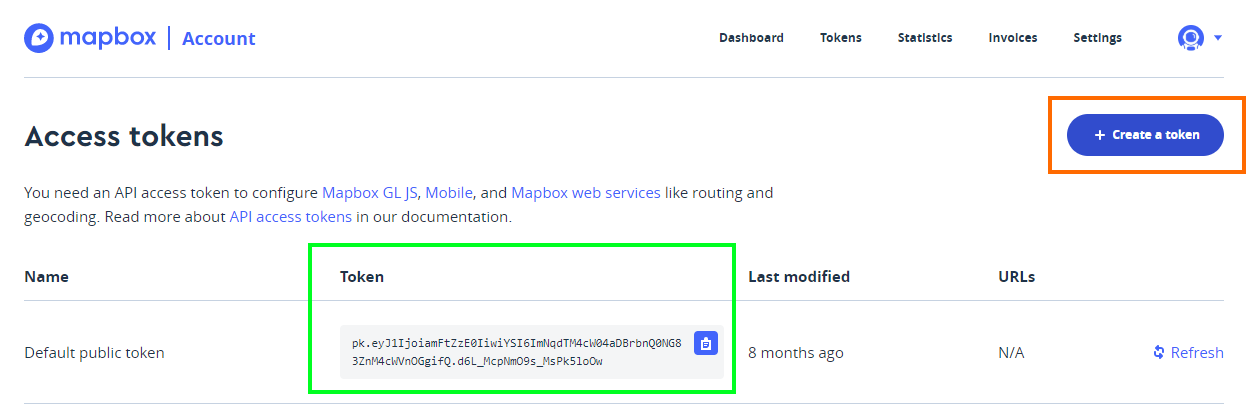
Donde hay que ingresar con las credenciales registradas previamente.



**Paso 2.** Una vez que se ingresa a mapbox, se da clic en la opción “**Tokens**”:



Se puede utilizar el valor del token que se encuentra por default o crear uno, asociando un nombre:



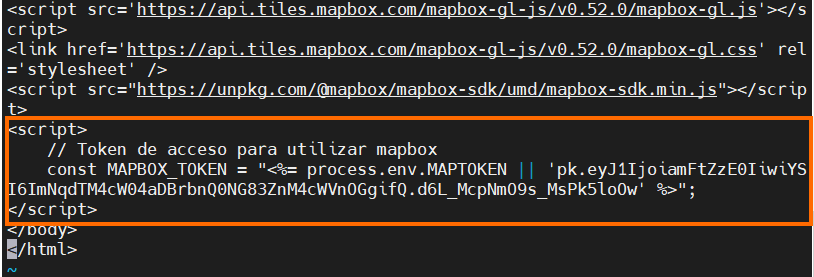
**Paso 3.** Una vez que se obtenga el token, ingresar a la siguiente ruta:

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

**Paso 4.** Abrir el archivo main.js

|  |
| --- |
| vi main.ejs |

Y buscar en el archivo el siguiente valor: MAPBOX\_TOKEN y actualizar el valor del token, que se encuentra entre comillas simples ' '



**Paso 5.** Guardar cambios y salir.

## 6.4 Actualización de dashboard

Se deben ejecutar los siguientes pasos para la actualización de código del módulo dashboard - visualizador.

**Paso 1.** Ingresar a la siguiente ruta:

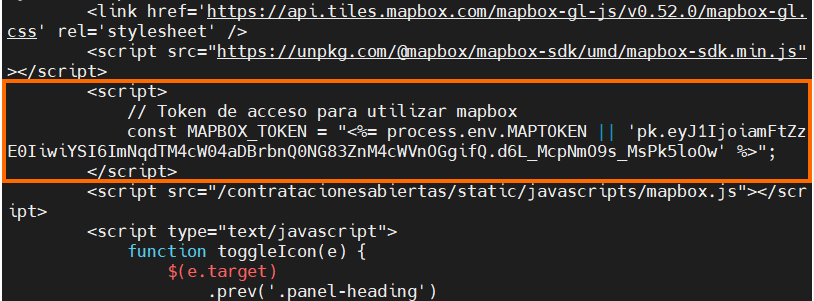
|  |
| --- |
| cd /var/www/html/contratacionesabiertas/dashboard/views |

**Paso 2.** Abrir el archivo contract.ejs ejecutando el siguiente comando:

|  |
| --- |
| *vi contract.js* |

****

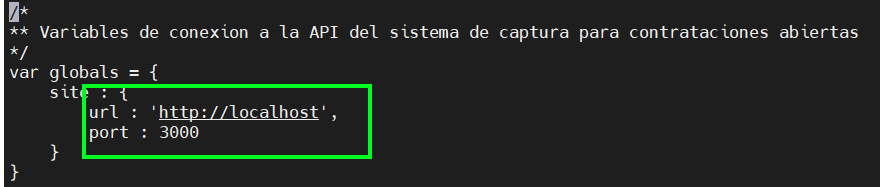
**Paso 3.** Actualizar el valor **MAPBOX\_TOKEN** por el token correspondiente, el cual se encuentra entre comillas simples ' '.



Guardar cambios y salir.

**Paso 4.** Ingresar a la carpeta /contratacionesabiertas/dashboard/public/javascript/, editar el archivo “**common.js**” para asignar la IP correspondiente del servidor y el puerto asignado para el módulo Sistema de captura.

|  |
| --- |
| *vi common.js* |



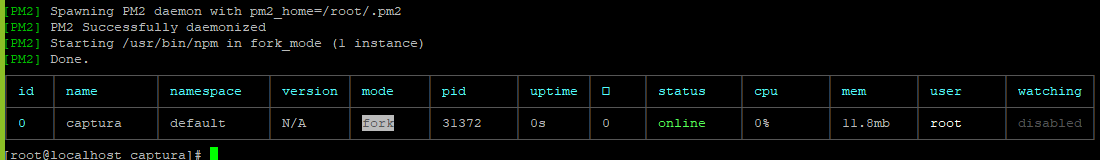
# 7. Iniciar los servicios de los módulos

**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* |

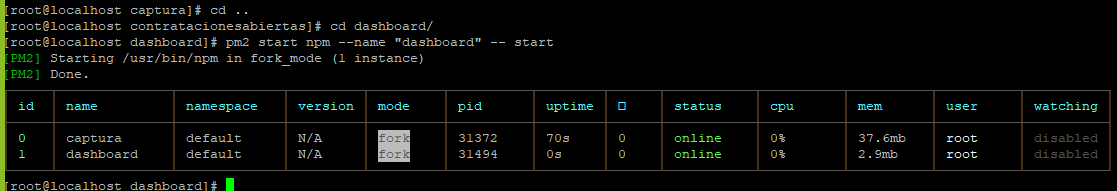


**Paso 3.** Salir de la carpeta captura e ingresar a la carpeta dashboard, ejecutando los siguientes comandos:

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

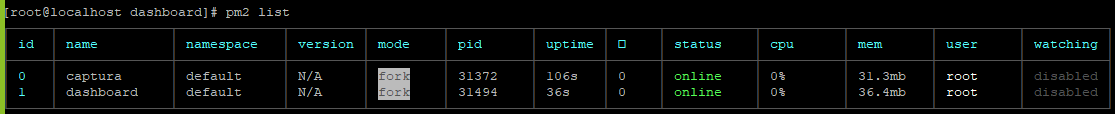
**Paso 4.** Iniciar el servicio del módulo “**Dashboard - visualizador**”, ejecutando el siguiente comando:

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



**Paso 5.** Ejecutar el siguiente comando para verificar que el estado de los servicios estén correctos.

|  |
| --- |
| *pm2 list* |



# 8. Acceder a los módulos

Una vez iniciado los servicios de los dos módulos, para ingresar a cada uno de ellos, ejecutamos los siguientes pasos.

## 8.1 Módulo de captura

**Paso 1.** Ingresar a la siguiente dirección:

|  |
| --- |
| *http://ip\_del\_servidor:3000* |



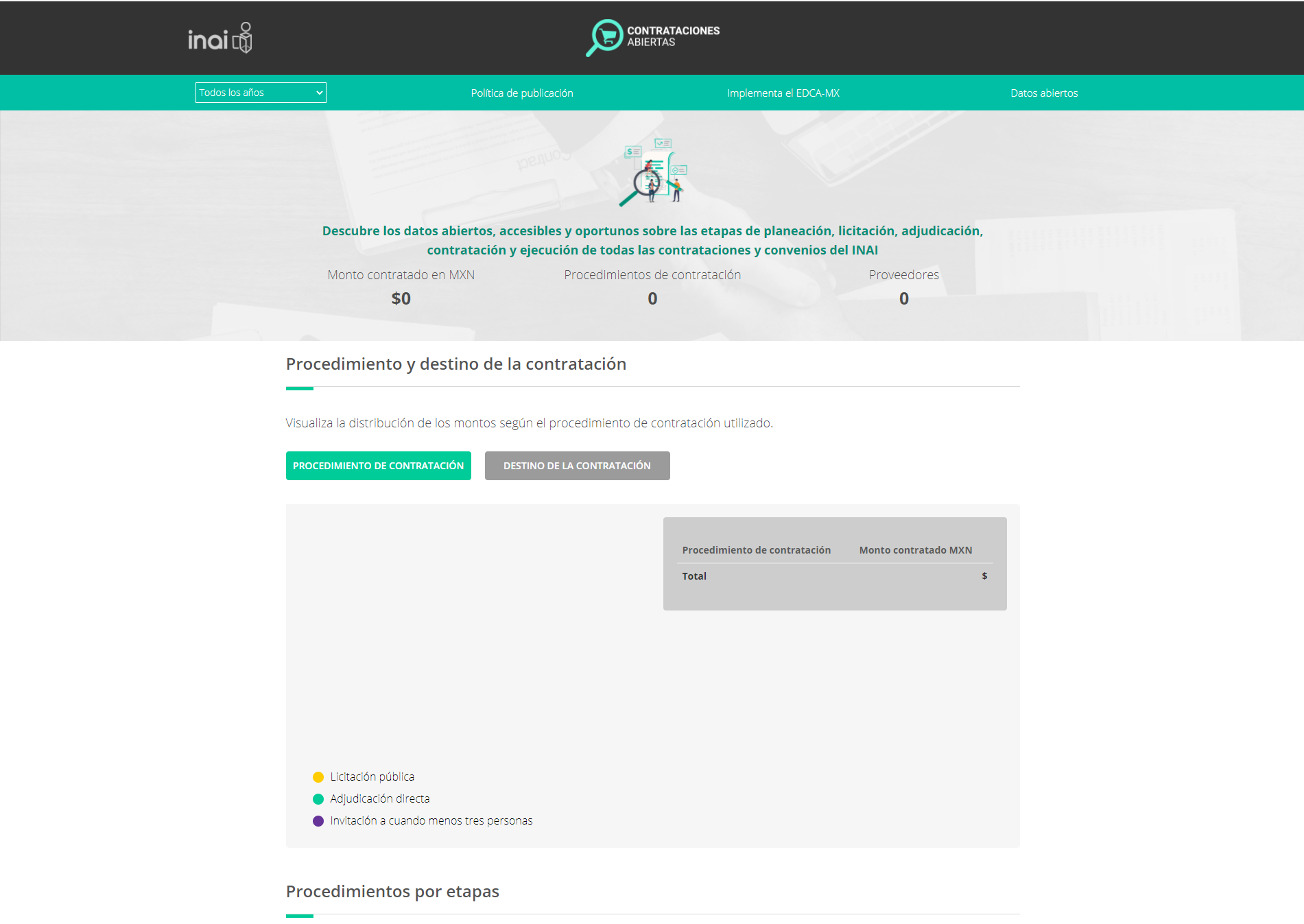
**Paso 2.** Ingresar el usuario y contraseña, creados desde la primera instalación.



## 8.2 Módulo de visualización (dashboard)

**Paso 1.** Ingresar a la siguiente dirección:

|  |
| --- |
| *http://ip\_del\_servidor:4000/contratacionesabiertas/contratos* |



Con esto se finaliza la actualización del código.

## 

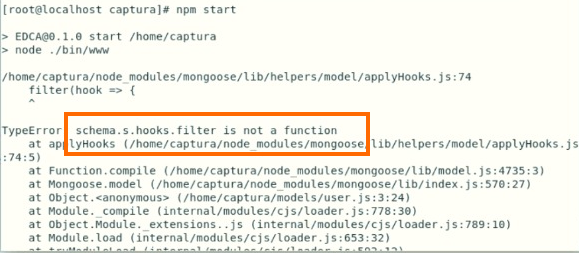
## 

# Anexos.

# Anexo 1. Errores comunes.

1. Error al iniciar el servicio del módulo Sistema de captura. Si al abrir en el navegador el módulo y no se visualiza la herramienta, se debe de ejecutar el comando npm start para verificar si el error que muestra es el siguiente:

schema.s.hooks.filter is not a function



Solución. Dentro de la carpeta /captura/ ejecutar el comando para borrar la carpeta “node\_modules”

rm -Rf node\_modules

y ejecutar el comando

npm install

Una vez ejecutado el comando, iniciamos el servicio con el comando npm start y realizamos la prueba de visualización del módulo en el navegador.

Si el módulo Sistema de captura ya se visualiza correctamente, detener el servicio con “ctrl + c” y se inicia el servicio con el comando

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

1. Error en la creación de una nueva contratación.

Causa. Omisión de la ejecución de las sentencias del paso “*5.1 Migración de las tablas para el del módulo de infraestructura”*



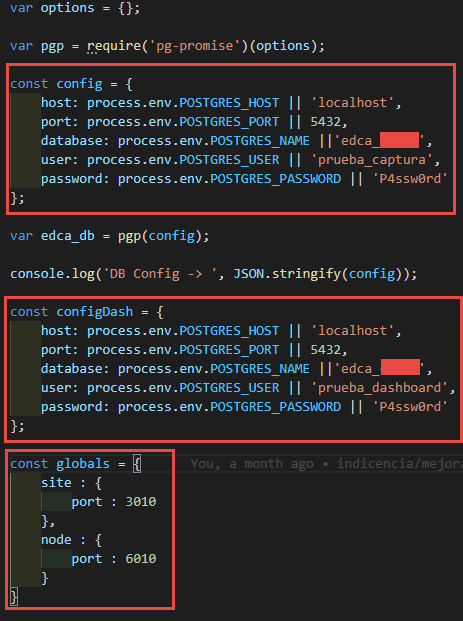
Solución. Ejecutar dentro de la carpeta captura, las sentencias:

*> sequelize db:migrate*

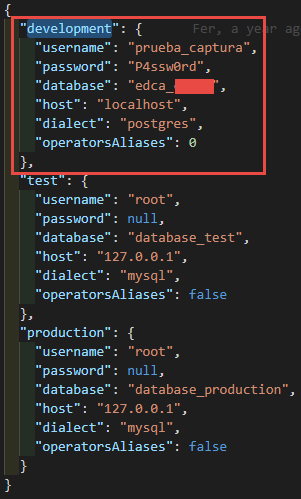
*> sequelize db:seed:all*

# Anexo 2. Configuraciones Generales para más de una instancia en el mismo servidor.

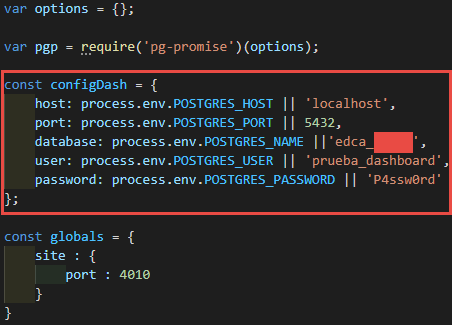
1. Configuración de cadenas de conexión para el “*sistema de captura*”:
   1. Modificar el archivo “*db\_conf.js*” de la ruta “*/captura/db\_conf.js*” en las variables de conexión “*host*”, “*database”, “user” y “password”* en “const *config”* y “const *configDash”*.  
        
      Modificar “const globals” con las variables correspondientes a los puertos.



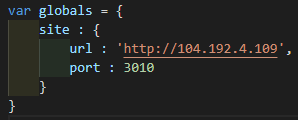
* 1. La configuración de los puertos por default es el número 3000 y 6000, estos se deberán cambiar si se desea tener otra instancia de la aplicación. Se puede ocupar el número de puerto 3001 hasta el 3999 y del 6001 hasta el 6999 respectivamente para cada instancia.
  2. Entrar al archivo “*config.json*” de la ruta “*/captura/config/config.json*” en el objeto “*development*” modificar las variables de conexión “*host*”, “*database”, “username” y “password”*.



1. Configuración de cadenas de conexión para el sistema “*dashboard*”:  
   1. Entrar en el archivo “*dash\_config.js*” de la ruta “*/dashboard/dash\_config.js*” y modificar las variables de conexión “*host*”, “*database”, “user”, “password”* en “const *configDash”*.
   2. Modificar “const globals” con las variables correspondientes al puerto.



* 1. La configuración del puerto es por default el número 4000, esto se deberán cambiar si se desea tener otra instancia de la aplicación. Se puede ocupar el número del puerto desde el 4001 hasta el 4999 uno por instancia.
  2. Entrar en el archivo “*common.js*” de la ruta “*/dashboard/javascript/common.js*” modificar la variable del puerto y de la ip correspondiente a la instancia instalada previamente en el archivo “*db\_conf.js*” de la ruta “*/captura/db\_conf.js*”.



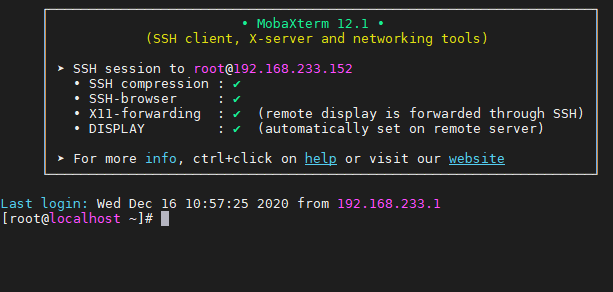
**NOTA:** Es necesario tener las reglas de firewall pertinentes en el servidor para el uso de estos puertos.

**NOTA 2:** Es necesario para que funcionen correctamente las múltiples instancias especificar una base de datos única para cada instancia, como por ejemplo “*edca\_gen*”.

# Anexo 3. Re-publicación de procesos de contratación.

Este punto es en el caso, que se requiera re-publicar todos los proceso de contratación de la herramienta.

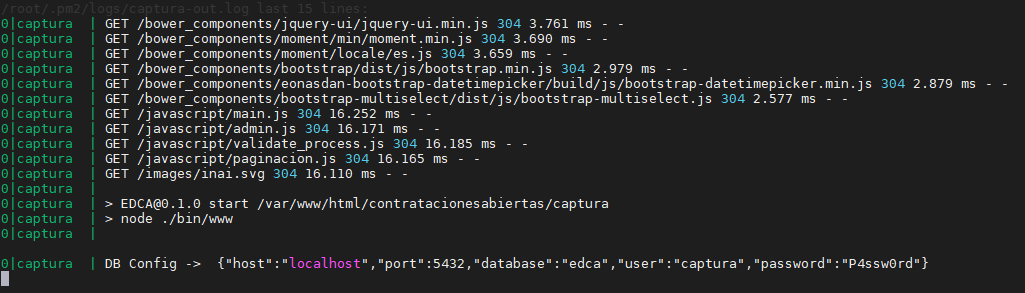
**Paso 1.** Abrir una terminal en el servidor de linux donde se encuentra instalada la aplicación.



**Paso 2**. Ingresar a la carpeta de “captura” donde se encuentra la carpeta de contrataciones abiertas y ejecutar el comando *pm2 logs captura*:



Ejemplo:



**Paso 3.** Ingresar al módulo de captura con las credenciales de administrador.

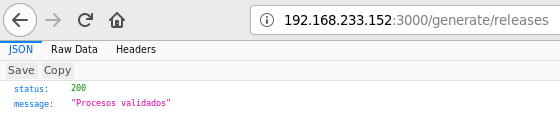


**Paso 4.** En una nueva ventana ingresar la siguiente dirección y darle enter:

|  |
| --- |
| *http://ip\_del\_servidor:3000/generate/releases* |

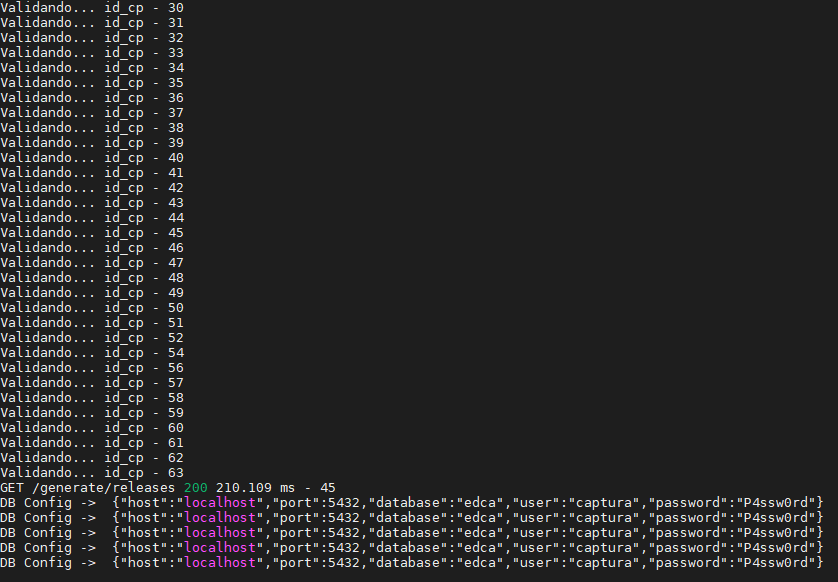
**NOTA:** Sustituir “*ip\_del\_servidor*” por la ip local del servidor donde se encuentra instalada la aplicación.

Ejemplo:

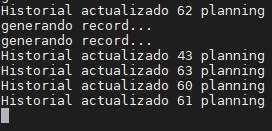


Este proceso es un poco tardado, lo que genera es un cambio en todos los procesos de contratación y valida cada proceso de contratación, debes de revisar la consola para que te muestre el avance del proceso de contratación.

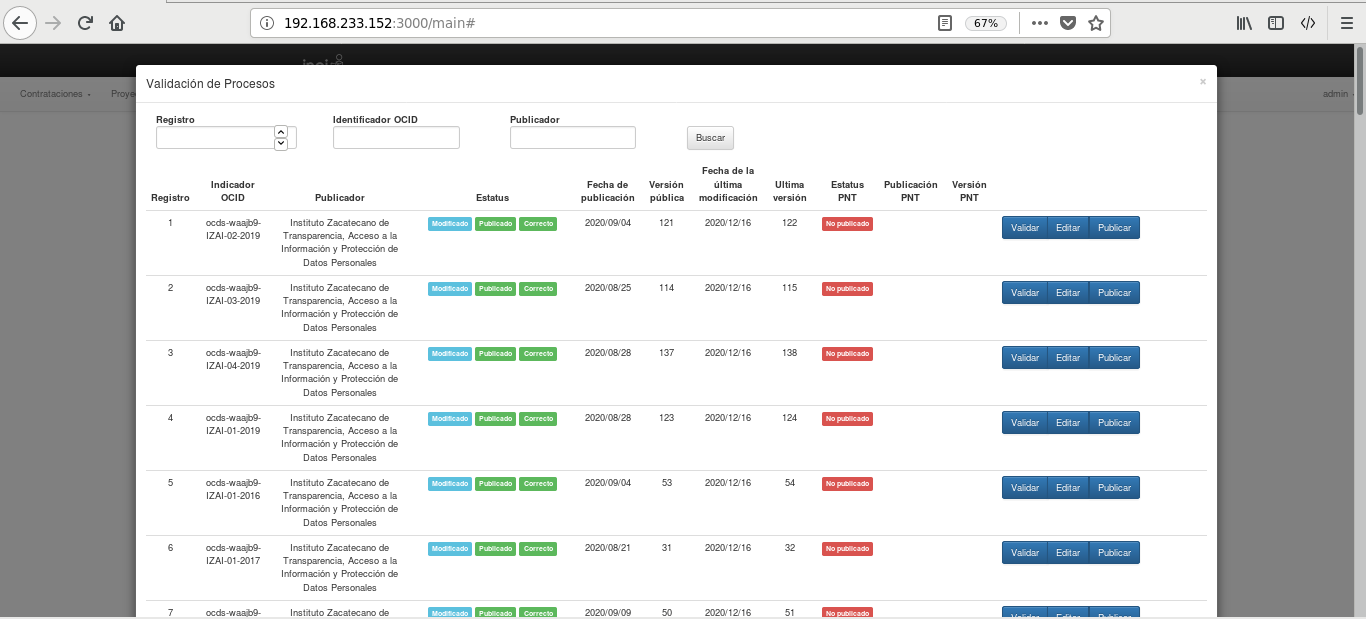
Al iniciar el proceso, en la terminal te muestra una lista de todos los procesos de contratación para que puedas saber cuando es que se termina dicho proceso.



En este ejemplo, el último proceso de contratación es el número 63, por lo tanto el proceso culmina cuando se ejecuta el proceso de la actualización del historial del último proceso de contratación.



Terminando este proceso cerramos la ventana de ejecución y actualizamos la lista de las contrataciones “Validación de Procesos” y verificamos que el estatus de todas las contrataciones se haya actualizado a “Modificado”.



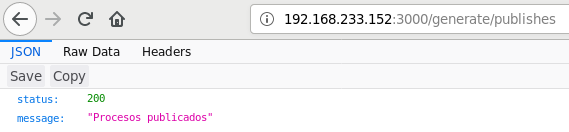
**Importante**: Es necesario mencionar que durante este proceso no se puede utilizar la herramienta, es un proceso tardado y se recomienda que se ejecute de manera nocturna.

**Paso 2.** Una vez terminado el proceso de actualización de las contrataciones se procede a publicarlas, en una ventana nueva se debe de ingresar a la siguiente dirección y dar enter:

|  |
| --- |
| *http://ip\_del\_servidor:3000/generate/publishes* |

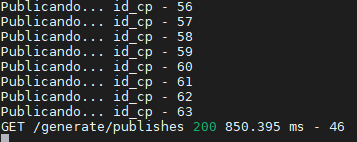
**NOTA:** Sustituir “*ip\_del\_servidor*” por la ip local del servidor donde se encuentra instalada la aplicación.

Ejemplo:

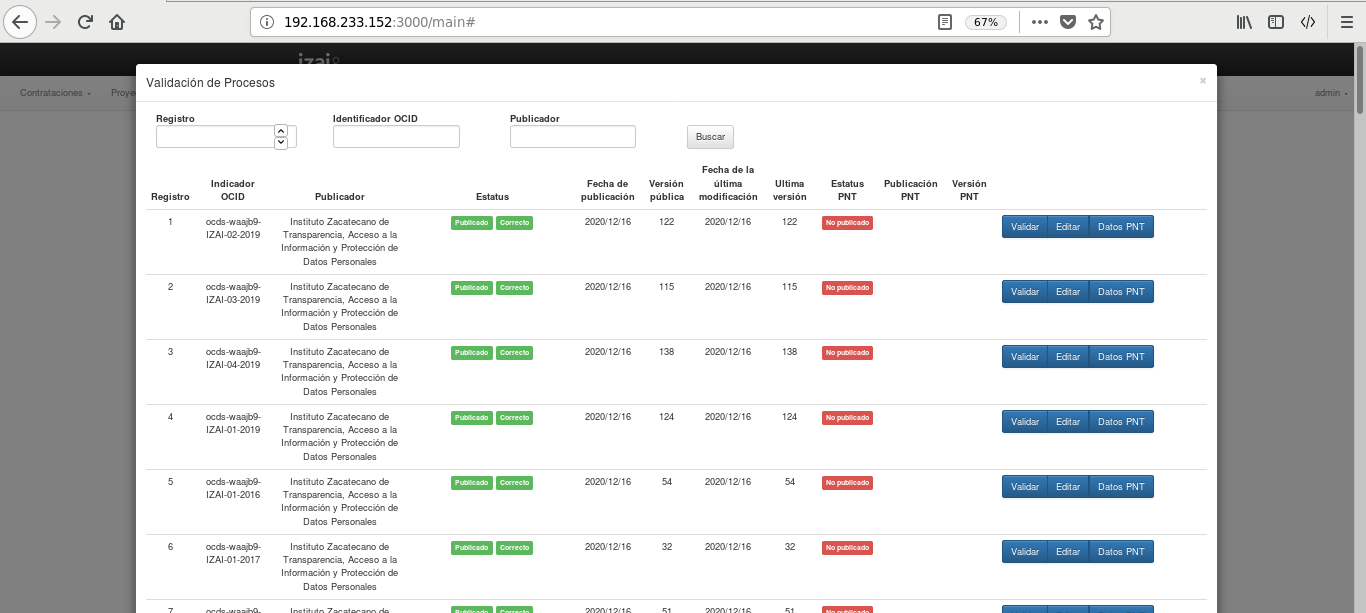


Este paso publicará cada proceso de contratación.

De igual manera se deberá monitorear su ejecución a través de la terminal, indicando así el último proceso de contratación publicado.

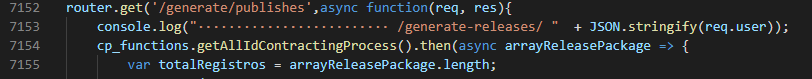


Terminando este proceso cerrar la ventana y se debe de actualizar la lista de las contrataciones “Validación de Procesos” y verificamos que el estatus de todas las contrataciones se haya actualizado a “Publicado”.



**Paso 3.** Comentar los métodos “***/generate/releases***” y “***/generate/publishes***” del archivo “***index.js***” de la carpeta “***routes***” en la ruta “***contratacionesabiertas/captura/routes***”, se debe comentar desde la línea 7119 a la línea 7185 y posteriormente guardar las modificaciones al archivo:





**NOTA:** Para comentar se debe de anteponer a las líneas del método los caracteres “//”

**Paso 4.** Reiniciar el servicio del módulo de captura con los siguientes comandos, dentro de la carpeta captura:

|  |
| --- |
| *pm2 delete captura* |

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