

# Proyecto de Versionamiento

Este documento es el manual de usuario de instalación del aplicativo para manejo de versiones. Tener en cuenta que, para muchos pasos, es necesario el acompañamiento del desarrollador.

## Requirements

- Descargar el ejecutable con el proyecto.

```
http://adacsc.co:1443/svn/repository/ADA/SICOF/Estructuras%20cargas%20datos/Proyectos_luis/versionador-install.exe
```

- Tener instalado el tortoiseSVN.

- Tener instalado vscode (preferencialmente) o algún editor de código.

## Installation

1. Descargar el ejecutable del repositorio.
2. Dar click en el ejecutable descargado y elegir una ruta para la instalación de preferencia dejar en la que aparece en el instalador.
3. Donde se guardó el aplicativo agregar la ruta a las variables de entorno del sistema, para que sea reconocido el aplicativo
4. Luego de tener el código del repositorio en la carpeta hooks, abrir la carpeta hooks en vscode o el editor elegido y ejecutar los siguientes comandos.

```
CONTROL + BLOCK MAYUS + Ñ -- Abre la terminal
```

Ejecutar el siguiente comando en la consola.

```
npm install oracledb@4.2.0 -- instala la ultima version compatible con oracle 11g  
npm i -- Instala las dependencias necesarias
```

Luego

```
cd ventana -- Cambiar de carpeta
```

y

```
npm install oracledb@4.2.0 -- instala la última versión compatible con oracle 11g
```

## `npm i -- Instala las dependencias necesarias`

5. Esto generará una carpeta llamada `node_modules` y dos archivos `package` en cada carpeta (hooks y ventana), pero es algo normal.

6. Después de esto, se deben realizar varias verificaciones para asegurar que este funcionando bien, ejecutar los siguientes pasos:

- Ítem de lista desordenada Abrir el cmd y ejecutar el siguiente comando.

```
netstat -ano | findstr 3000 -- No debe devolver nada, de no ser así, hablar con el desarrollador.
```

- Ítem de lista desordenada Volver a vscode y a la terminal y ejecutar el siguiente comando.

```
cd ventana -- Si no estas en la carpeta ventana
```

- Ítem de lista desordenada Luego:

```
node backend.js -- Debe devolver un aviso de que se esta ejecutando en el puerto 3000
```

- Ítem de lista desordenada Y volver abrir el cmd y ejecutar el siguiente comando:

```
netstat -ano | findstr 3000 -- Debe devolver algo, de no ser así, hablar con el desarrollador.
```

En caso de que todo funcione bien.

7. Luego de esto, se deben configurar los hooks apuntando al repositorio en el que se desea implementar la funcionalidad, se preparo un video para esto, así que aquí comparto el [enlace](<https://drive.google.com/file/d/1IErreVaCkDYavLUN9cMUXLgdkDAj1hle/view?usp=drivesdk>).

8. Por último, se debe validar recuerda que se debe validar que tanto el client x64 de oracle como node js estén agregados en las variables e entorno y tener instalada la línea de comandos de tortoiseSvn (obligatorio).

9. Se debe setear la ruta del repositorio también en el código ya que este extrae la información directamente del svnserver, este se debe setear en el archivo `take_commit` y el archivo `backend.js`, tener en cuenta que también se deben setear las rutas en el post y pre commit, dependiendo de como organices la carpeta, crear la carpeta repositorio en la ruta C:, recomendable.

Notas: Se debe modificar en el código la ruta del svn donde se quiere hacer commit para que pueda traer los datos ya las conexiones vienen seteado en el código para la base de datos deben tener conexión a internet y a la VPN para la conexión a la base de datos.

El repositorio se debe setear en el archivo `backend.js` y `take_commit.js` la pueden buscar con control f y la constante se llama `repoURL`.

```
JS take_commit.js X
nmit.js > dbConfig > connectString
const repoUrl = 'http://adacsc.co:1443/svn/repository/ADA/SICOF/Estructuras%20cargas%20datos/';
```

```
JS backend.js X
ventana > JS backend.js > app.post('/insertData') callback > version
1 const repoUrl = 'http://adacsc.co:1443/svn/repository/ADA/SICOF/Estructuras%20cargas%20datos/';
```

## USE

Respecto al funcionamiento del aplicativo se debe hacer lo siguiente.

Se debe al igual que siempre dar clic derecho y dar en commit.

Al igual que siempre poner notas al commit indicando cosas importantes sobre este commit

Luego le debes dar en el botón de commit y esperar unos segundos que suba el programa, luego va a salir un formulario de la siguiente manera.

The image shows a web form titled "Formulario de Versionamiento" (Versioning Form) set against a dark background with a forest scene. The form contains the following fields and a button:

- Version:** A text input field with the placeholder text "Ingresa la version".
- Descripción:** A larger text input field with the placeholder text "Ingresa la descripcion".
- S\_funcional:** A text input field with the placeholder text "Ingresa la solucion funcional".
- S\_tecnica:** A text input field with the placeholder text "Ingresa la solucion tecnica".
- Notas:** A larger text input field with the placeholder text "notas commit".
- Guardar:** A blue rounded button at the bottom of the form.

En este se deben llenar todos los campos.

- Campo versión: Se debe ingresar la versión a desplegar la que va a salir a producción.
- Campo Descripción: información acerca del cambio o que se esta liberando.

- Campo S\_FUNCIONAL: Información sobre la solución que se dio al problema presentado en palabras mas sencillas.

- Campo S\_TECNICA: Información sobre la solución que se le dio al problema se puede incluir la información técnica.

Notas: Alguna observación o tema a tener en cuenta sobre la liberación de esta versión.

Luego de dar clic en guardar, debe salir un aviso de datos guardados, cierre la ventana y espero pocos segundos a que se termine de realizar el commit y ese sería el procedimiento.

From:

<http://wiki.adacsc.co/> - **Wiki**

Permanent link:

[http://wiki.adacsc.co/doku.php?id=ada:tips:sicoferp:general:versionamiento\\_svn](http://wiki.adacsc.co/doku.php?id=ada:tips:sicoferp:general:versionamiento_svn)



Last update: **2024/01/29 12:57**