

|  |
| --- |
| **Instalación Desarrollo**  **Integración Verint**  **Claro** |

**CONFIDENCIAL**

**© Copyright Sixbell S.A.**

Versión 1.1

Agosto 2023

Índice de Contenidos

Tabla de contenido

[1. Introducción 4](#_Toc140137754)

[2. Instalación Frontend 5](#_Toc140137755)

[2.1. Requisitos 5](#_Toc140137756)

[2.2. Instalación 5](#_Toc140137757)

[3.-Instalación módulos 9](#_Toc140137758)

**Prefacio**

El objetivo de este documento es dar una definición precisa de los pasos a seguir que deberán ser implementados para desarrollar una aplicación que permita insertar la metadata en las tablas de Verint.

**Alcance**

El presente documento especifica los pasos a seguir para instalar el desarrollo para cumplir con el requerimiento.

**Resumen Ejecutivo**

En el presente documento se darán a conocer la instalación de la solución para desarrollar una aplicación que permita insertar la metadata en las tablas de Verint.

# Introducción

El objetivo de este documento es dar una definición precisa de los pasos a seguir que deberán ser implementados para el funcionamiento del integrador verint.

# Instalación Frontend

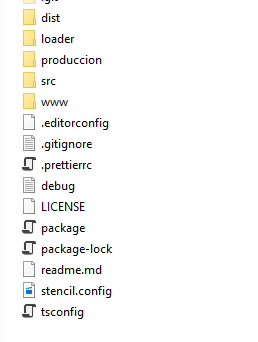
## Requisitos

Para instalar el portal en nodeJS será necesario contar con los siguiente.

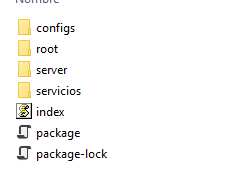
* Nodejs con express

## Instalación

* Desde GitHub <https://github.com/SixbellComponentes/ca_administrador_tmp.git> descargar el código fuente que tendrá la siguiente estructura



Luego copiar todo lo que se encuentra en el directorio de producción al servidor en donde se ejecutara el administrador TMP. A continuación, se explica el contenido de los directorios que componen al directorio producción

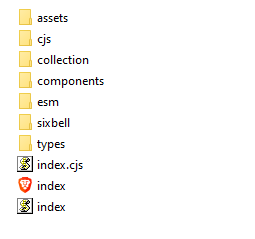


* configs: contiene el archivo de configuración **configProduccion.json** con la siguiente información

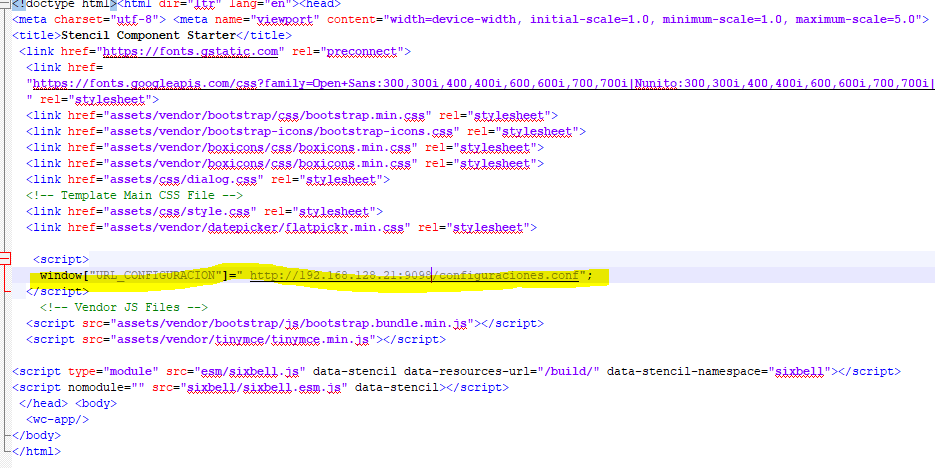


1. port: Puerto donde se ejecutará el servidor node
2. acceso: Información requerida por Genesis cloud
3. tmp: Datos necesarios para integrar el portal tmp en un iframe
4. servicio: Se debe configurar el id del servicio (Se recupera desde la tabla SERVICIO) correspondiente a Nicaragua que se asociara a las transferencias

* root: Contiene el contenido del frontend como componentes y paginas



En el **index.html** se debe configurar la url donde corre el aplicativo del frontend quien llama al servicio configuraciones.conf, que se encarga de recuperar los parámetros de configuración que necesita la aplicación para realizar los llamados a la api del backend. (url de los servicios, callback genesys cloud, url al logo del frontend)



* server: Contiene el archivo server.js que permite el manejo de los llamados post al servidor (no se modifica)
* servicios: Contiene lo siguiente:

1. Config.js: Recupera la información del archivo json configProduccion.json
2. LoadConfig.js: Mantiene los parámetros configurables con las url de los servicios del api backend. Parte del archivo de configuración se muestra a continuación:

* URL\_CALLBACK: Callback requerido por Genesys Cloud.
* LOGO: Ruta a la imagen del logo que se visualiza en el frontend
* URL\_SERVICES: Ruta a los servicios del backend
* PRODUCCION: Si el valor es **false** se utiliza para modo debug, **true** realiza el llamado a la apli de Genesys.
* index.js: Permite ejecutar el servidor nodeJS.

# 3.-Instalación módulos

* Una vez finalizado los pasos anteriores se debe ejecutar el comando **npm install** en el directorio donde se encuentra el archivo index.js. Este comando instalara todos los módulos requeridos por el administrador TMP.

Control de Versiones

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Versión** | **Fecha** | **Autor** | **Revisado por** | **Principales Cambios** |
| 1.1 | Julio 2023 |  |  | Versión Original |
|  |  |  |  |  |