

**<Sistema de Inventario Ágil y Práctico>**

Manual de Instalación

Versión: 0100

**HOJA DE CONTROL**

| **Organismo** | S.I.A.P | | |
| --- | --- | --- | --- |
| **Proyecto** | Sistema de Inventario Ágil y Practico | | |
| **Entregable** | Manual de Instalación | | |
| **Autor** | **laura suarez** | | |
| **Aprobado por** |  | **Fecha Aprobación** | DD/MM/AAAA |
|  |  | **Nº Total de Páginas** |  |

REGISTRO DE CAMBIOS

| **Versión** | **Causa del Cambio** | **Responsable del Cambio** | **Fecha del Cambio** |
| --- | --- | --- | --- |
| 001 | Versión inicial | laura suarez | 11/06/2024 |
|  |  |  |  |
|  |  |  |  |

CONTROL DE DISTRIBUCIÓN

| **Nombre y Apellidos** |
| --- |
| Laura suarez |
| Daniel arrieta |
|  |
|  |
|  |

**ÍNDICE**

[1.1 Objeto 3](#_heading=h.3znysh7)

[1.2 Alcance 4](#_heading=h.2et92p0)

[2 DESCRIPCIÓN DEL SISTEMA 5](#_heading=h.1t3h5sf)

[2.1 Antecedentes y descripción funcional del sistema 5](#_heading=h.4d34og8)

[2.2 Componentes fundamentales 5](#_heading=h.2s8eyo1)

[2.3 Relación con otros sistemas 5](#_heading=h.17dp8vu)

[3 RECURSOS HARDWARE 7](#_heading=h.3rdcrjn)

[3.1 Servidores 7](#_heading=h.26in1rg)

[3.2 Estaciones cliente 7](#_heading=h.lnxbz9)

[3.3 Conectividad 7](#_heading=h.35nkun2)

[3.4 Restricciones 8](#_heading=h.1ksv4uv)

[4 RECURSOS SOFTWARE 9](#_heading=h.44sinio)

[4.1 Matriz de certificación 9](#_heading=h.2jxsxqh)

[8.2 Restricciones técnicas del sistema 9](#_heading=h.z337ya)

[8.3 Requisitos de otros sistema 9](#_heading=h.3j2qqm3)

[9 INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE 12](#_heading=h.1y810tw)

[10 CONFIGURACIÓN DEL SISTEMA 13](#_heading=h.4i7ojhp)

[10.1 Configuración de otros sistemas 13](#_heading=h.1ci93xb)

[11 COMPILACIÓN DEL SISTEMA 14](#_heading=h.3whwml4)

[12 INSTALACIÓN DEL SISTEMA 16](#_heading=h.2bn6wsx)

[12.1 Requisitos previos 16](#_heading=h.qsh70q)

[12.2 Procedimiento de instalación 16](#_heading=h.3as4poj)

[13 VERIFICACIÓN DEL PROCESO DE INSTALACIÓN 18](#_heading=h.1pxezwc)

[14 MARCHA ATRÁS DE LA INSTALACIÓN Y CONFIGURACIÓN 19](#_heading=h.49x2ik5)

[14.1 Requisitos previos 19](#_heading=h.2p2csry)

[14.2 Marcha atrás del sistema 19](#_heading=h.147n2zr)

[14.3 Marcha atrás del software base 20](#_heading=h.3o7alnk)

[14.4 Resumen de tareas de configuración 20](#_heading=h.ihv636)

[15 GLOSARIO 22](#_heading=h.32hioqz)

[16 BIBLIOGRAFÍA Y REFERENCIAS 23](#_heading=h.1hmsyys)

## Objeto

Proporciona instrucciones detalladas y paso a paso para su correcta implementación, dirigido a administradores de sistemas y personal técnico. Su propósito es asegurar una instalación eficiente y sin errores, garantizando la compatibilidad del entorno, la estabilidad del software y la disponibilidad de soporte técnico.

## Alcance

El manual de instalación del software está dirigido a todos los departamentos involucrados en su implementación, configuración y mantenimiento. El Departamento de TI desarrolla, actualiza e implementa el manual, además de verificar el funcionamiento del software. El Equipo de Desarrollo de Software contribuye con detalles técnicos y soporte durante la instalación. La Auditoría Interna valida y registra el manual, garantizando su conformidad con las políticas internas. Los Usuarios Finales, como el personal de operaciones y logística, configuran el software según sus necesidades y confirman su funcionalidad, proporcionando retroalimentación. Este enfoque colaborativo asegura una instalación eficiente y efectiva del sistema de inventario.

# DESCRIPCIÓN DEL SISTEMA

## Antecedentes y descripción funcional del sistema

La empresa supermercado Santa Martha es un negocio de tipo autoservicio que cuenta con varias sedes distribuidas en Bogotá D.C; Las ganancias se derivan de la venta de sus productos y llevan a cabo el manejo de sus inventarios en la zona de la bodega de una forma empírica o tradicional, sin ningún tipo de modelo a seguir o sistema de apoyo.

Este problema dentro de la bodega fue descubierto tras haber realizado un análisis en la empresa partiendo de la encuesta diseñada para el jefe, se suministró una breve información, donde se evidencia que la ausencia del sistema de gestión de inventario provoca falencias en el control, planeación y organización de los pedidos dificultando la optimización y la distribución de los productos a las diferentes sedes y por ende la insatisfacción de las mismas generando inconformidad, disminución en las ventas y pérdida económica a la empresa, ya sea por extravío de productos, por robo o se dañan con el tiempo.

## Componentes fundamentales

| **Módulo** | **Descripción** |
| --- | --- |
| Módulo de Registro de Productos: | Este módulo permite ingresar y almacenar información detallada sobre los productos en el inventario, incluyendo nombre, descripción, fecha de caducidad, precio y otras características relevantes. |
| Módulo de Gestión de Inventarios: | Aquí se lleva a cabo el seguimiento y control de los niveles de inventario, alertando sobre productos con existencias bajas o próximos a vencer, y optimizando la reposición de productos para evitar faltantes o excedentes. |
| Módulo de Movimientos de Inventario | Este módulo registra todas las entradas y salidas de productos |
| Módulo de Administración de Productos: | Aquí se lleva a cabo la gestión de productos, permitiendo agregar, eliminar y actualizar productos de manera sencilla, manteniendo la base de datos de productos actualizada y precisa. |

## Relación con otros sistemas

| **Sistema** | **Relación** |
| --- | --- |
| Sistema de Ventas: | El sistema de ventas proporciona información sobre las transacciones de productos, como ventas realizadas y productos vendidos, que se utiliza para actualizar automáticamente los niveles de inventario y poder generar facturas al momento de la salida de productos. |
| Sistema de Compras y Proveedores | El sistema de gestión de inventario se relaciona con el sistema de compras y proveedores para gestionar la reposición de inventario. La información sobre pedidos de compra y datos de proveedores se utiliza para realizar el aprovisionamiento de productos de manera oportuna y eficiente. |

# RECURSOS HARDWARE

## Servidores

| **Servidor 1** | | |
| --- | --- | --- |
| **Dato** | **Valor mínimo** | **Valor recomendado** |
| Procesador | Intel core I3 4a generación | Intel core I5 10a generación |
| Memoria RAM | 4 GB | 8 GB |
| Tamaño Almacenamiento | 120 GB de almacenamiento interno SSD | 120 GB de almacenamiento interno SSD |
| Otros |  |  |

## Estaciones cliente

| **Dato** | **Valor mínimo** | **Valor recomendado** |
| --- | --- | --- |
| Procesador | Intel core I5 3a generación | intel core I5 11a generación |
| Memoria RAM | 4 GB | 8 GB |
| Tamaño Almacenamiento | 120 GB de almacenamiento interno Hdd | 150 GB de almacenamiento interno SSD |
| Otros |  |  |

## Conectividad

| **Dato** | **Valor mínimo** | **Valor recomendado** |
| --- | --- | --- |
| Tarjeta de Red | Ethernet 20 Mbps | Ethernet 50 Mbps |
| Tipo de Red | Wifi 4 | Red alambrica |
| Otros |  |  |

## Restricciones

| **Restricción** | **Detalle** |
| --- | --- |
| Capacidad de Almacenamiento | La capacidad de almacenamiento del servidor de archivos puede ser limitada, lo que podría afectar la capacidad de almacenar grandes volúmenes de archivos o datos. |
| Integración con Otros Sistemas | El servidor de gestión de productos puede tener limitaciones en la integración con otros sistemas, especialmente si requiere interoperabilidad con sistemas externos que tienen requisitos específicos de compatibilidad o comunicación. |
| Recursos Limitados | Puede enfrentar limitaciones en el rendimiento si no cuenta con suficientes recursos de CPU, RAM y almacenamiento, lo que podría afectar la velocidad de respuesta y la capacidad de procesamiento de solicitudes, especialmente en operaciones de búsqueda y actualización de grandes volúmenes de datos. |

# RECURSOS SOFTWARE

## Matriz de certificación

Este apartado detalla la compatibilidad de la versión del sistema objeto de la instalación con diversas versiones de software base y componentes externos.

1. **Servidor de Aplicaciones:**
   1. Se requiere el uso de MySQL como servidor de aplicaciones.
2. **Servidor de Base de Datos:**
   1. La versión mínima requerida para el servidor de base de datos es MySQL 8.0.
3. **LDAP (Protocolo Ligero de Acceso a Directorios):**
   1. No se requiere la integración con LDAP para el funcionamiento del sistema.

Esta información asegura que el sistema objeto de la instalación pueda operar correctamente con las versiones específicas de los componentes mencionados, garantizando así su compatibilidad y funcionamiento adecuado.

## Restricciones técnicas del sistema

| **Elemento** | **Descripción** |
| --- | --- |
| Sistema operativo | windows 10 en adelante |
| Servidor de aplicaciones | Apache tomcat 9 en adelante y Mysql |
| Servidor de base de datos | Mysql 8.0 y azure |
| Compilador | visual studio code |
| JVM | java 11 en adelante |
| Otros |  |

## Requisitos de otros sistema

| **Sistema de ventas** | |
| --- | --- |
| Descripción | Sistema utilizado para gestionar la salida y entrada de productos. |
| Funcionalidad utilizada | Registro de ventas, generación de informes sobre el movimiento de los productos, gestión de inventario. |
| Requisito formal 1 | El administrador del sistema proporcionará a cada empleado un nombre de usuario y una contraseña únicos. La asignación de estas credenciales dependerá del rol que desempeñe el empleado en la organización y determinará su acceso a las diferentes partes del sistema. |
| Requisito técnico 1 | Todos los usuarios deben utilizar el software específico de ventas proporcionado por la organización. Este software debe estar instalado en cada dispositivo desde el cual se acceda al sistema de ventas. |
| Requisito de comunicación 1 | Los dispositivos utilizados por los empleados para acceder al sistema de ventas deben estar conectados a la red interna de la organización para garantizar la seguridad de los datos y la integridad del sistema. |
| Otros |  |

| **Sistema de compra y proveedores** | |
| --- | --- |
| Descripción | Se utiliza para gestionar la reposición de inventario. La información sobre pedidos de compra y datos de proveedores se utiliza para realizar el aprovisionamiento de productos de manera oportuna y eficiente. |
| Funcionalidad utilizada | Registro de proveedores, gestión de órdenes de compra, gestión de inventario de productos. |
| Requisito formal 1 | Los proveedores deben ser registrados en el sistema por el administrador. La información requerida incluye datos de contacto, detalles de productos ofrecidos y términos comerciales. |
| Requisito técnico 1 | Los empleados encargados de realizar compras deben utilizar el sistema interno de gestión de compras proporcionado por la empresa. Este software estará disponible para todos los usuarios autorizados |
| Requisito de comunicación 1 | El sistema debe permitir la comunicación interna entre los departamentos relevantes, como compras, almacén, para coordinar la generación de órdenes de compra, recibir y registrar las entregas de los productos, y gestionar la facturación y pagos correspondientes. |
| Otros |  |

# INSTALACIÓN Y CONFIGURACIÓN DEL SOFTWARE BASE

| **Xampp** | | |
| --- | --- | --- |
| Descripción | XAMPP es una distribución de Apache fácil de instalar que contiene MariaDB (MySQL), PHP y Perl. Es una solución completamente gratuita y de código abierto para el desarrollo de aplicaciones web en local. | |
| Localización | https://www.apachefriends.org/es/index.html | |
| Procedimiento de instalación | | |
| Paso 1 | Descargar el instalador de XAMPP desde el enlace proporcionado. | |
| Paso 2 | Ejecutar el instalador y seguir las instrucciones del asistente de instalación. | |
| Paso 3 | Seleccionar los componentes que se desean instalar (Apache, MySQL, PHP, etc.). | |
| Paso 4 | Completar la instalación y abrir el Panel de Control de XAMPP. | |
| Procedimiento de configuración | | |
| Paso 1 | Iniciar los servicios de Apache y MySQL desde el Panel de Control de XAMPP. | |
| Paso 2 | Configurar la seguridad de MySQL utilizando phpMyAdmin para establecer la contraseña de root y crear usuarios adicionales con permisos específicos en caso de ser necesario. | |
| Paso 3 | Reiniciar los servicios de Apache y MySQL para aplicar los cambios. | |
| Parámetros a configurar | | |
| Parámetro 1 | MySQL | username |
| Parámetro 2 | MySQL | password |

# CONFIGURACIÓN DEL SISTEMA

## Configuración de otros sistemas

| **Visual Code** | | |
| --- | --- | --- |
| Descripción | Visual Studio Code (VS Code) es un editor de código fuente desarrollado por Microsoft. Es multiplataforma y soporta extensiones para diferentes lenguajes de programación, incluyendo soporte para debugging, control de versiones, y herramientas de desarrollo integradas. | |
| Localización | https://code.visualstudio.com/download | |
| Procedimiento de instalación | | |
| Paso 1 | Descargar el instalador | |
| Paso 2 | Ejecutar el instalador y seguir las instrucciones del asistente de instalación. Aceptar los términos y condiciones y seleccionar las opciones de instalación preferidas. | |
| Paso 3 | Elegir el directorio de instalación. | |
| Paso 4 | Completar la instalación. | |
| Procedimiento de configuración | | |
| Paso 1 | Instalar extensiones necesarias desde la sección de extensiones (icono de cuadrado en la barra lateral izquierda). | |
| Paso 2 | Configurar la integración con sistemas de control de versiones como Git. hacer clic en el icono de Git en la barra lateral izquierda y seguir las instrucciones para configurar un nuevo repositorio. | |
| Parámetros a configurar | | |
|  |  |  |

# 

| **Node.js** | | |
| --- | --- | --- |
| Descripción | Node.js es un entorno de ejecución de JavaScript basado en el motor V8 de Chrome que permite la ejecución de código JavaScript en el lado del servidor. Es ampliamente utilizado para construir aplicaciones de red y servidores eficientes, de alta concurrencia, y de baja latencia. | |
| Localización | https://nodejs.org/en/download/prebuilt-installer | |
| Procedimiento de instalación | | |
| Paso 1 | Descargar el instalador | |
| Paso 2 | Ejecutar el instalador y seguir las instrucciones del asistente de instalación. Aceptar los términos y condiciones y seleccionar las opciones de instalación preferidas. | |
| Paso 3 | Elegir las opciones de instalación preferidas, incluyendo la adición de Node.js al PATH del sistema para facilitar su uso desde la línea de comandos. | |
| Paso 4 | Completar la instalación y verificar que Node.js se haya instalado correctamente ejecutando node -v y npm -v en una terminal o línea de comandos para ver las versiones de Node.js y npm, respectivamente. | |
| Procedimiento de configuración | | |
| Paso 1 | Instalar paquetes de desarrollo necesarios para el proyecto usando npm, por ejemplo, ejecutando npm install <nombre-del-paquete>. | |
| Parámetros a configurar | | |
|  |  |  |

# 

| **Angular** | | |
| --- | --- | --- |
| Descripción | Angular es un framework de desarrollo web de código abierto creado por Google para construir aplicaciones dinámicas de una sola página (SPA) con un enfoque en la escalabilidad y la mantenibilidad. Utiliza TypeScript como lenguaje principal y proporciona una arquitectura modular basada en componentes, inyección de dependencias, y un robusto sistema de plantillas. | |
| Localización | https://nodejs.org/en/download/prebuilt-installer | |
| Procedimiento de instalación | | |
| Paso 1 | Asegúrate de tener Node.js instalado en tu sistema, ya que Angular CLI depende de Node.js y npm. Verifica la instalación ejecutando node -v y npm -v en la terminal. | |
| Paso 2 | Instalar Angular CLI globalmente usando npm. Ejecuta el siguiente comando en la terminal o línea de comandos  npm install -g @angular/cli | |
| Paso 3 | Verifica la instalación correcta de Angular CLI ejecutando:  ng version | |
| Paso 4 | Crea un nuevo proyecto Angular ejecutando:  ng new nombre-del-proyecto | |
| Procedimiento de configuración | | |
| Paso 1 | Navega al directorio del proyecto creado:  cd nombre-del-proyecto | |
| Paso 2 | Ejecuta el servidor de desarrollo para iniciar la aplicación en tu navegador:  ng serve | |
| Parámetros a configurar | | |
|  |  |  |

# 

# COMPILACIÓN DEL SISTEMA

| **Requisitos de compilación** | |
| --- | --- |
| **Requisito** | **Descripción** |
| **Ubicación Fuentes** | https://github.com/DanielArrieta04/SIAP |
| **Configuración 1** | Abrir el proyecto desde Visual Studio Code después de clonar el repositorio. |
| **Configuración 2** | Configurar las extensiones necesarias para el desarrollo, como Git para control de versiones |
| **Configuración 3** | Abrir el Panel de Control de XAMPP y asegurarse de que Apache y MySQL estén iniciados. |
| **Repositorio 1** | https://github.com/DanielArrieta04/SIAP |
| **Dependencias** | Framework express  npm install  mysql2  body-parser  cors  jsonwebtoken  sweetalert2  nodemon -g  concurrently  rxjs  pm2 |
| **Producto final** | Archivo WAR (Web Application Archive) generado después de la compilación del proyecto SIAP. |

| **Procedimiento de compilación** | |
| --- | --- |
| **Paso** | **Descripción** |
| **1** | Clonar el repositorio SIAP desde Git utilizando Visual Studio Code.  **Descripción:** Descargar una copia del repositorio SIAP desde el servidor remoto a la máquina local. |
| **2** | Compilar el proyecto SIAP en Visual Studio Code.  **Descripción:** Configurar Apache y MySQL en XAMPP para soportar el desarrollo y pruebas locales del SIAP. |
| **3** | Compilar el proyecto SIAP en Visual Studio Code.  **Descripción:** Ejecutar los comandos de compilación necesarios para transformar los archivos fuente en el producto final |

# INSTALACIÓN DEL SISTEMA

Para la instalación del sistema, asegúrese de tener Visual Studio Code, Git, XAMPP, Node.js y Angular. Instale Visual Studio Code y Git, configure las credenciales de Git, y use Git en VS Code para clonar el repositorio del proyecto. Instale XAMPP y asegúrese de que Apache y MySQL funcionen correctamente. Descargue e instale Node.js, y luego Angular CLI globalmente usando npm. Verifique las instalaciones con los comandos adecuados. Navegue al directorio del proyecto clonado, instale las dependencias necesarias, y configure Angular, incluyendo ajustes en angular.json y variables de entorno. Finalmente, documente todos los pasos y configuraciones realizados para asegurar la correcta funcionalidad del sistema.

## Requisitos previos

**Requisitos Previos:**

**Verificar Requisitos del Sistema:** Asegúrese de que el sistema operativo cumpla con los requisitos mínimos para Visual Studio Code, Git, XAMPP, Node.js, y Angular.

* Instalar Visual Studio Code, Git , XAMPP,Node.js y Angular en el sistema operativo si aún no están instalados.
* Configurar Git con las credenciales adecuadas para acceder a los repositorios del proyecto.
* Asegurarse de que XAMPP esté configurado correctamente y los servicios de Apache y MySQL estén funcionando.
* Verifique la instalación correcta ejecutando node -v y npm -v en la terminal.
* verifique la instalación ejecutando ng versión en la terminal.

## Procedimiento de instalación

| **Procedimiento de instalación** | |
| --- | --- |
| **Paso 1** | |
| **Tipo** | Instalación |
| **Componente** | XAMPP |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Descargar el instalador de XAMPP desde el sitio oficial. Ejecutar el instalador y seguir las instrucciones del asistente para completar la instalación. Incluir la configuración de los servicios de Apache y MySQL durante la instalación. |
| **Paso 2** | |
| **Tipo** | Configuración |
| **Componente** | XAMPP, Apache, MySQL |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Iniciar XAMPP y asegurarse de que los servicios de Apache y MySQL estén funcionando correctamente. Configurar los puertos y ajustes adicionales según las necesidades del proyecto. |

| **Paso 3** | |
| --- | --- |
| **Tipo** | Instalación |
| **Componente** | Visual Studio Code |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Descargar el instalador de Visual Studio Code desde el sitio oficial. Ejecutar el instalador y seguir las instrucciones del asistente para completar la instalación. |
| **Paso 4** | |
| **Tipo** | Configuración |
| **Componente** | Visual Studio Code |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Configurar las extensiones necesarias desde la sección de extensiones de Visual Studio Code, selecciona la extensión oficial "Git" ofrecida por Microsoft.  Haz clic en "Install" (Instalar) para instalar la extensión.   * Una vez instalada la extensión, es posible que necesites reiniciar Visual Studio Code para activarla completamente. * Abre un proyecto o directorio que esté gestionado por Git, o inicializa un nuevo repositorio Git en tu proyecto existente si no lo has hecho aún (puedes hacerlo desde la terminal de Visual Studio Code o desde una terminal externa utilizando comandos de Git). |

| **Paso 5** | |
| --- | --- |
| **Tipo** | Instalación |
| **Componente** | Node.js |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Descargar el instalador de Node.js. Ejecutar el instalador y seguir las instrucciones del asistente para completar la instalación, asegurándose de que Node.js y npm se añadan al PATH del sistema. Verificar la instalación correcta ejecutando node -v y npm -v en la terminal para comprobar las versiones instaladas. |
| **Paso 6** | |
| **Tipo** | Configuración |
| **Componente** | Node.js |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Navega al directorio de tu proyecto utilizando la terminal, y utiliza npm para instalar las dependencias del proyecto ejecutando npm install. Esto asegura que todas las dependencias listadas en el archivo package.json estén instaladas y listas para usar en el desarrollo de tu aplicación Node.js. |

| **Paso 7** | |
| --- | --- |
| **Tipo** | Instalación |
| **Componente** | Angular CLI |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Descargar e instalar Angular CLI globalmente utilizando npm. Para ello, abre una terminal y ejecuta el comando npm install -g @angular/cli. Este proceso instalará Angular CLI en tu sistema y lo añadirá al PATH, permitiendo que puedas usar el comando ng desde cualquier directorio. Verifica la instalación correcta ejecutando ng version en la terminal para confirmar la versión instalada. |
| **Paso 8** | |
| **Tipo** | Configuración |
| **Componente** | Angular CLI |
| **Permisos** | Usuario con permisos de administración |
| **Descripción** | Abre Visual Studio Code y navega al directorio de tu proyecto Angular. Usa ng serve para iniciar la aplicación en modo desarrollo, accesible en http://localhost:4200. Asegúrate de que las configuraciones de Angular como el enrutamiento y los módulos estén correctamente definidas. Puedes gestionar y ajustar la configuración de tu aplicación desde archivos como angular.json, y personalizar variables de entorno según las necesidades del desarrollo. |

# VERIFICACIÓN DEL PROCESO DE INSTALACIÓN

**Acceso al Sistema:**

* Configurar adecuadamente los permisos de acceso para Apache y MySQL en XAMPP.
* Verificar la correcta instalación y acceso de Node.js y Angular CLI con los permisos necesarios.

**Funcionalidad Básica:**

* Realizar pruebas exhaustivas para confirmar la navegación fluida entre páginas y la correcta visualización de datos en aplicaciones Angular.
* Verificar que las operaciones CRUD funcionen sin errores en aplicaciones Node.js, y que las aplicaciones web alojadas en Apache dentro de XAMPP respondan como se espera.

**Documentación y Registro:**

* Documentar detalladamente todas las pruebas realizadas, incluyendo resultados obtenidos y problemas identificados durante las pruebas post-despliegue.
* Mantener un registro actualizado de todas las configuraciones específicas de XAMPP, Apache, MySQL, Node.js, Angular y Visual Studio Code para futuras referencias y mantenimiento del sistema.

# MARCHA ATRÁS DE LA INSTALACIÓN Y CONFIGURACIÓN

## Requisitos previos

**Copia de Seguridad:**

* Realizar una copia de seguridad completa de todos los datos críticos del sistema, incluyendo bases de datos, archivos de configuración y cualquier otro dato relevante que pueda ser afectado durante el proceso de marcha atrás.

**Documentación Detallada:**

* Contar con documentación detallada de todos los pasos realizados durante la instalación y configuración inicial. Esto incluye registros de configuración de software, ajustes de red, permisos de usuario, entre otros.

**Herramientas de Desinstalación:**

* En caso de ser necesario identificar y tener a mano las herramientas y utilidades necesarias para desinstalar o revertir los componentes del sistema instalados.

## Marcha atrás del sistema

| **Procedimiento de marcha atrás** | |
| --- | --- |
| **Paso 1** | |
| **Tipo** | Desinstalación de software |
| **Componente** | Visual Studio Code |
| **Permisos** | Acceso de administrador |
| **Descripción** | Eliminar el programa, así como limpiar cualquier configuración residual que pueda haber quedado en el sistema. |
| **Paso 2** | |
| **Tipo** | Desinstalación de software |
| **Componente** | XAMPP (Apache y MySQL) |
| **Permisos** | Acceso de administrador |
| **Descripción** | Asegurarse de detener todos los servicios relacionados antes de proceder con la desinstalación para evitar conflictos. Esto incluye eliminar los servicios de Apache y MySQL y restaurar cualquier configuración previa al estado original. |
| **Paso 3** | |
| **Tipo** | Desinstalación de software |
| **Componente** | Node.js |
| **Permisos** | Acceso de administrador |
| **Descripción** | Para desinstalar Node.js correctamente, asegúrate de detener todos los procesos relacionados y luego utiliza el desinstalador provisto por Node.js para tu sistema operativo. Elimina cualquier instalación adicional y limpia las referencias de Node.js y npm del PATH del sistema para evitar conflictos. Finalmente, elimina archivos de configuración como .npmrc y módulos globales instalados previamente con npm si es necesario. |
| **Paso 4** | |
| **Tipo** | Desinstalación de software |
| **Componente** | Angular CLI |
| **Permisos** | Acceso de administrador |
| **Descripción** | Para desinstalar Angular CLI correctamente, primero detén todos los procesos relacionados y luego utiliza npm para eliminar la instalación global ejecutando npm uninstall -g @angular/cli en la terminal. Asegúrate de verificar y eliminar cualquier otra instalación local de Angular CLI y limpiar cualquier referencia en el PATH del sistema para evitar conflictos futuros. Elimina archivos de configuración como angular.json si es necesario. Estos pasos aseguran una desinstalación efectiva y un entorno limpio para futuros desarrollos con Angular. |

## Marcha atrás del software base

#### Visual Studio Code:

1. **Desinstalación de Visual Studio Code:**
   * Utiliza el desinstalador de Visual Studio Code proporcionado por el sistema operativo (como el Panel de Control en Windows o la aplicación de desinstalación en macOS).
   * Sigue las instrucciones del desinstalador para eliminar completamente el software del sistema.
   * Asegúrate de eliminar cualquier configuración adicional durante el proceso.
2. **Desinstalación Node.js:**
   * Ejecuta el desinstalador y sigue los pasos proporcionados para asegurar una desinstalación completa.
   * Verifica y elimina cualquier otra instalación de Node.js que puedas tener, especialmente si utilizaste gestores de versiones como nvm (Node Version Manager)
   * Revisa y elimina archivos de configuración residuales, como archivos .npmrc o directorios de módulos globales, para limpiar completamente el sistema de configuraciones relacionadas con Node.js.
3. **Desinstalación Angular CLI:**
   * Abre una terminal y ejecuta el comando npm uninstall -g @angular/cli para eliminar la instalación global de Angular CLI.
   * Verifica y elimina cualquier instalación local de Angular CLI en tus proyectos. Esto puede implicar eliminar directorios como node\_modules en tus proyectos Angular.
   * Revisa y elimina cualquier referencia de Angular CLI del PATH del sistema para evitar conflictos. Puedes hacerlo revisando las variables de entorno en la configuración del sistema.
   * Elimina archivos de configuración como angular.json y archivos de cache relacionados con Angular CLI si deseas realizar una limpieza adicional del entorno de desarrollo.

#### XAMPP (Apache y MySQL):

1. **Detención de Servicios:**
   * Inicia XAMPP y detén los servicios de Apache y MySQL desde el Panel de Control de XAMPP.
   * Asegúrate de que todos los servicios estén detenidos antes de proceder.
2. **Desinstalación de XAMPP:**
   * Navega al directorio de instalación de XAMPP en tu sistema operativo (por ejemplo, C:\xampp en Windows).
   * Ejecuta el desinstalador de XAMPP que se encuentra en este directorio.
   * Sigue las instrucciones del desinstalador para eliminar completamente Apache y MySQL del sistema.
   * Verifica que todos los componentes y configuraciones relacionadas hayan sido eliminados correctamente.

## Resumen de tareas de configuración

**Configuración del Entorno de Desarrollo:**

* Configuración de Visual Studio Code para integración con Git y otras herramientas necesarias.

**Configuración de Git:**

* Configuración de nombre de usuario y correo electrónico en Git.
* Clonación del repositorio del proyecto desde Git para comenzar el desarrollo.

**Configuración de XAMPP:**

* Configuración de Apache y MySQL en XAMPP para el entorno local de desarrollo.
* Asegurar que los puertos y configuraciones de red sean adecuados para el proyecto.

**Configuración de Node.js:**

* Configuración de Node.js para el entorno local de desarrollo, asegurando que todos los aspectos necesarios estén correctamente ajustados para el proyecto específico. Esto incluye configurar las variables de entorno relevantes, y asegurarse de que las dependencias y módulos necesarios estén correctamente instalados y configurados.

**Configuración de Angular CLI:**

* Ajusta la configuración de Angular para el desarrollo local, asegurando que Angular CLI esté correctamente configurado para gestionar proyectos. Esto incluye la configuración del archivo angular.json para definir la estructura del proyecto, rutas y opciones de compilación. Verifica que los puertos del servidor de desarrollo estén configurados adecuadamente y que cualquier configuración de red necesaria esté correctamente establecida para garantizar el funcionamiento fluido del proyecto Angular en el entorno local.

# GLOSARIO

| **Término** | **Descripción** |
| --- | --- |
|  |  |
|  |  |
|  |  |

# BIBLIOGRAFÍA Y REFERENCIAS

| **Referencia** | **Título** | **Código** |
| --- | --- | --- |
| Ref. 1 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |