Logotipo

Descripción generada automáticamente

**TABLA DE CONTENIDOS**

[**1**](#_heading=h.30j0zll) **Introducción 3**

[**2**](#_heading=h.1fob9te) **Alcance 3**

[**3**](#_heading=h.2et92p0) **Requisitos del sistema 3**

[**4**](#_heading=h.3dy6vkm) **Pasos de instalación 4**

[**5**](#_heading=h.tyjcwt) **Conclusión 6**

1. **Introducción**

Este manual está diseñado para guiar a los desarrolladores en la instalación, configuración y ejecución del sistema **Medical AID**, utilizando las herramientas y tecnologías mencionadas. El objetivo es proporcionar una guía clara y detallada para preparar el entorno de desarrollo en Windows 10/11.

1. **Alcance**

El sistema **Medical AID** es una solución que combina una API backend desarrollada con Node.js y Express, un frontend creado con React, y una base de datos gestionada con PostgreSQL. Este manual incluye todos los pasos necesarios para instalar y configurar las herramientas requeridas para que el sistema funcione tanto en entornos de desarrollo como en producción.

1. **Requisitos del sistema**

* Sistema Operativo: Windows 10 o 11 (64 bits).
* Hardware Recomendado:
* Procesador: Intel Core i5 o superior.
* Memoria RAM: 8 GB o superior.
* Espacio en disco: 20 GB disponibles.
* Software Requerido: Navegador moderno (Google Chrome, Microsoft Edge).

1. **Instalación y Configuración del Entorno de Desarrollo**

**5.1 Instalación de Node.js y npm**

Para instalar Node.js v20.18.0 y npm, dirígete al siguiente enlace: [instaladores de nodejs](https://nodejs.org/en/blog/release/v20.18.0),  y descarga el instalador que corresponda a tu sistema operativo, luego ejecutalo e instala el software.

Si prefieres usar el administrador de paquetes, consulta la [guía oficial de Node.js](https://nodejs.org/en/download/package-manager/) para las instrucciones de instalación. Recuerda que la versión de Node.js debe ser la v20.18.0.

**5.2 Instalación de PIP y Python**

1. **Descarga e instalación de Python:**

* Visita la página oficial de descarga de Python.
* Descarga la versión más reciente compatible con tu sistema operativo.
* Durante la instalación, asegúrate de seleccionar la opción 'Add Python to PATH' en la pantalla de configuración inicial.

1. **Verificación de la instalación:**

* Abre una terminal o línea de comandos.
* Ejecuta los siguientes comandos para verificar que Python y PIP se instalaron correctamente:
  + python --version (También puede ser: py –version o python3 –version, dependiendo de tu sistema operativo).



* + Pip --version

Texto

Descripción generada automáticamente

Si ambos comandos devuelven una versión válida, la instalación fue exitosa.

1. **Actualización de PIP:**

* Para asegurarte de que PIP está actualizado, ejecuta: TU\_RUTA\_A\_PYTHON\python.exe -m pip install –upgrade

Texto

Descripción generada automáticamente

1. **Resolución de problemas comunes:**

* Si encuentras problemas al ejecutar los comandos anteriores, verifica que Python esté correctamente añadido al PATH:
  + Ve a la configuración avanzada del sistema operativo.
  + Asegúrate de que la ruta al ejecutable de Python (por ejemplo, C:\Python39\) esté incluida en la variable de entorno PATH.

**5.3 Clonación de repositorio**

1. **Abrir una terminal o línea de comandos:**

* En Windows, puedes utilizar **PowerShell** o la línea de comandos (cmd).
* Alternativamente, usa una terminal integrada en tu editor de código, como Visual Studio Code.

1. **Clonar el repositorio del proyecto:**

* Ejecuta el siguiente comando para clonar el repositorio en tu máquina local:
* git clone https://github.com/Mumoide/Medical-AID.git

1. **Navegar al directorio del proyecto clonado:**

* Usa el comando cd para acceder al directorio del proyecto:
  + cd TU\_RUTA\Medical-AID\Codigo.

Esto te ubicará en la raíz del proyecto, desde donde podrás realizar configuraciones o instalar dependencias.

**5.4 Instalación de Herramientas y Dependencias**

**Librerías de Python**

1. **Navegar a la carpeta del script de predicciones:**

* Abre una terminal o línea de comandos.
* Usa el siguiente comando para acceder a la carpeta que contiene el modelo de predicciones:
  + - cd TU\_RUTA\Medical-AID\Código\Endpoint\_modelo

1. **Instalar dependencias:**

* Ejecuta el siguiente comando para instalar las librerías necesarias listadas en el archivo requirements.txt:
  + - pip install -r requirements.txt

1. Asegúrate de que pip esté correctamente configurado en tu entorno.

**Librerías de JavaScript**

1. **Instalar dependencias del frontend:**

* Navega al directorio del frontend con el siguiente comando:
  + cd TU\_RUTA\Medical-AID\Codigo\frontend
* Instala las dependencias ejecutando:
  + npm install

Esto descargará e instalará todas las librerías necesarias para el frontend desde package.json.

1. **Instalar dependencias del backend:**

* Navega al directorio del backend:
  + cd TU\_RUTA\Medical-AID\Codigo\backend
* Instala las dependencias ejecutando:
  + npm install

1. **Verificación de la instalación:**

* Para confirmar que las dependencias se instalaron correctamente, puedes ejecutar los siguientes comandos:
  + Para el frontend:
    - npm start

1. Esto debe iniciar el servidor de desarrollo en tu navegador.
   * Para el backend:
     + npm run dev
2. Esto debe iniciar el servidor en http://localhost:3001.

**5.5 Configuración de la base de datos PostgreSQL**

Instala [PostgreSQL](https://www.postgresql.org/download/) en tu sistema, durante la instalación, configura un usuario con contraseña. Luego debes instalar PGAdmin, el cual se encuentra como instalación opcional, por lo que no olvides seleccionarlo dentro de los componentes de instalación.

**Creación y conexión de la base de datos**

1. Abre PGAdmin e inicia sesión.
2. Crea una nueva base de datos llamada Medical\_AID.
3. Crea el modelo de base de datos desde el archivo SQL que se encontrará en la carpeta: “***TU\_RUTA\Medical-AID\Codigo\Backend\database\schema.sql****”.*
4. Configura las credenciales en las variables de entorno (ver siguiente sección).

**Inserción de datos estáticos**

1. Ejecutar archivos: “***importacion\_tabla\_enfermedades***” e “***importacion\_tabla\_sintomas.py***” ubicados en *“****TU\_RUTA\Medical-AID\Codigo\Importacion\_data\_tablas\_estaticas****”.*
2. Ejecutar scripts SQL ubicados en *“****TU\_RUTA\Medical-AID\Codigo\Backend\database\fixes****”*

**5.6 Configuración de variables de entorno**

El archivo. env almacena las variables de entorno del backend de manera segura y accesible. Sigue estos pasos para crear y configurar este archivo:

1. **Ubicación del archivo. env:**
   * Asegúrate de estar en el directorio raíz del backend del proyecto:
     + cd TU\_RUTA\Medical-AID\Codigo\Backend\backend
2. **Crear el archivo. env:**
   * Usa cualquier editor de texto o terminal para crear un archivo llamado “.env” en este directorio. Si estás en la terminal, puedes usar:
     + touch .env
3. **Configurar las variables de entorno:**
   * Abre el archivo. env en tu editor de texto y copia las siguientes variables:
     + PG\_USER="TU\_USUARIO"
     + PG\_HOST=localhost
     + PG\_DATABASE=Medical\_AID
     + PG\_DATABASE\_TEST=test\_Medical\_AID
     + PG\_PASSWORD="TU\_CONTRASEÑA"
     + PG\_PORT=5432
     + SECRET\_KEY="CLAVE\_SECRETA\_PARA\_TOKEN"
     + EMAIL\_USER="CASILLA@PARAENVIAR.CORREOS"
     + EMAIL\_PASS="CONTRASEÑA\_DE\_CASILLA"
4. **Detalles importantes:**
   * **PG\_USER**: Reemplázalo con el nombre de usuario de tu base de datos PostgreSQL.
   * **PG\_HOST**: Si usas una base de datos en un servidor remoto, cambia localhost por la dirección del servidor.
   * **PG\_DATABASE**: El nombre de la base de datos principal que utiliza el proyecto.
   * **PG\_DATABASE\_TEST**: La base de datos destinada para pruebas.
   * **PG\_PASSWORD**: Reemplázalo con la contraseña de tu usuario PostgreSQL.
   * **SECRET\_KEY**: Una clave secreta que se utiliza para firmar tokens de autenticación. Usa una cadena larga y única.
   * **EMAIL\_USER y EMAIL\_PASS**: Credenciales de la cuenta de correo que se usará para enviar emails desde la aplicación.
5. **Asegurar la confidencialidad del archivo .env:**
   * Asegúrate de que el archivo .env no sea incluido en tu repositorio. Agrega la siguiente línea al archivo .gitignore si aún no está:
     + .env
6. **Levantamiento de la Aplicación**

Este punto describe cómo iniciar los servicios del frontend, backend y el endpoint del modelo, y cómo validar que el sistema se haya desplegado correctamente.

**6.1 Levantamiento del Frontend**

1. Navega al directorio del frontend desde la terminal:

* cd TU\_RUTA\Medical-AID\Codigo\Backend\frontend

1. Inicia el servidor de desarrollo con el siguiente comando:

* npm start

**Esto levantará la interfaz del sistema en el puerto predeterminado.**

**6.2 Levantamiento del Backend**

1. Abre una nueva terminal y navega al directorio del backend:

* cd TU\_RUTA\Medical-AID\Codigo\Backend\backend

1. Inicia el servidor del backend en modo desarrollo con el siguiente comando:

* npm run dev

Esto levantará la API en el puerto configurado en las variables de entorno (predeterminado: 3001).

**6.3 Ejecución del Script de Python**

1. Abre otra terminal y navega al directorio del modelo de predicciones:
   * cd TU\_RUTA\Medical-AID\Codigo\Backend\Endpoint\_modelo
2. Ejecuta las pruebas para validar el correcto funcionamiento del clasificador de síntomas:
   * pytest test\_symptom\_classifier\_proba.py

Este comando ejecutará las pruebas unitarias y asegurará que el modelo funcione como se espera.

**6.4 Validación del Correcto Levantamiento**

1. **Frontend:**
   * Abre un navegador web y verifica que el frontend esté corriendo en:
     1. http://localhost:3000
   * Asegúrate de que la interfaz carga correctamente y que no haya errores en la consola del navegador.
2. **Backend:**
   * Verifica que el backend esté corriendo, accediendo a:
   * http://localhost:3001
   * Puedes usar herramientas como Postman o cURL para probar los endpoints de la API y asegurarte de que responden correctamente.
3. **Interacción entre Frontend y Backend:**
   * Realiza pruebas desde la interfaz del frontend para confirmar que las solicitudes hacia la API funcionan como se espera.

****

**Conclusión**

El sistema Medical AID combina múltiples tecnologías para ofrecer una solución robusta. Siguiendo este manual, cualquier desarrollador podrá configurar y ejecutar el entorno de desarrollo o producción sin complicaciones. Si se siguen los pasos correctamente, se garantiza un entorno estable y funcional.

**Contacto con el equipo de soporte**

Este espacio está reservado para la resolución de problemas comunes que puedan surgir durante el despliegue. En caso de encontrar problemas no cubiertos en esta sección, por favor, contacta al equipo de soporte técnico:

* [ma.brard@duocuc.cl](mailto:ma.brard@duocuc.cl)
* ra.estevez@duouc.cl
* ma.inostrozam@duocuc.cl