



# Guía para Nuevos Desarrolladores

## (/docs/guia-desarrolladores.md)

Este documento proporciona la información esencial para que cualquier nuevo desarrollador pueda configurar su entorno, ejecutar el proyecto y comprender las tecnologías clave.

Markdown

# 🛠️ Guía para Nuevos Desarrolladores - AppLimpieza

### ## 1. Introducción

Bienvenido/a al equipo de desarrollo de **AppLimpieza**, el Sistema de Gestión de Limpieza de la EPET N°20. Este documento es su punto de partida para configurar el entorno de desarrollo y comenzar a contribuir.

### ## 2. Requisitos Previos

Asegúrese de tener instaladas las siguientes herramientas en su sistema:

Icono	Herramienta	Versión Mínima	Propósito
●	<b>Node.js</b>	18+	Entorno de ejecución de JavaScript y servidor local.
📦	<b>npm</b>	9+	Gestor de paquetes (instalación y manejo de dependencias).
🌳	<b>Git</b>	Más reciente	Control de versiones y clonación del repositorio.
💻	<b>Editor de Código</b>	VS Code (Recomendado)	Para desarrollo, edición y depuración del código fuente.

## ## 3. Configuración del Entorno Local

Siga estos pasos para obtener una copia local del proyecto y prepararla para la ejecución.

### ### 3.1. Clonación del Repositorio

Abra su terminal (o Git Bash) y ejecute los siguientes comandos:

```
``bash
# Reemplazar por la URL real del repositorio
git clone [URL_DEL_REPOSITORIO_AQUÍ]
cd AppLimpieza
```

## 3.2. Instalación de Dependencias

El proyecto utiliza **npm** para la gestión de paquetes.

```
Bash
npm install
```

## 3.3. Configuración de Variables de Entorno (FIREBASE)

Para conectar la aplicación con la base de datos y la autenticación de Firebase, es **obligatorio** crear un archivo de configuración.


1. **Cree un archivo llamado `.env`** en la raíz del proyecto.
2. **Copie y pegue** la siguiente plantilla de configuración (sustituyendo los valores por sus credenciales reales de Firebase):

Fragmento de código

```
# Firebase Configuration
REACT_APP_FIREBASE_API_KEY=tu_api_key_aqui
REACT_APP_FIREBASE_AUTH_DOMAIN=tu-proyecto.firebaseio.com
REACT_APP_FIREBASE_PROJECT_ID=tu-proyecto-id
REACT_APP_FIREBASE_STORAGE_BUCKET=tu-proyecto.appspot.com
REACT_APP_FIREBASE_MESSAGING_SENDER_ID=123456789
REACT_APP_FIREBASE_APP_ID=1:123456789:web:abcdef123456

# Google OAuth
REACT_APP_GOOGLE_CLIENT_ID=tu-client-id.apps.googleusercontent.com

# Environment
REACT_APP_ENV=development
```

 **Alerta de Seguridad:** Verifique que el archivo `.gitignore` incluya la línea `.env` para evitar subir credenciales sensibles.

## 4. Scripts de Ejecución (Apéndice F)

Utilice los siguientes comandos para trabajar con el proyecto:

Comando	Descripción
<b>npm start</b>	Inicia el servidor de desarrollo en <a href="http://localhost:3000">http://localhost:3000</a> . Soporta <i>Hot Reload</i> .
<b>npm run build</b>	Compila la aplicación para producción, creando una carpeta <code>/build</code> optimizada y lista para el <i>deploy</i> .
<b>npm test</b>	Ejecuta la suite de pruebas automatizadas.

## 5. Estructura y Tecnologías Clave

### 5.1. Estructura de Carpetas

La lógica principal de la aplicación reside en la carpeta **src/** (Ver </docs/estructura.md> para el mapa completo).

Directorio	Propósito
<b>src/views/</b>	Contiene los componentes de <b>página completa</b> (ej: <a href="#">VistaAdmin.jsx</a> ).
<b>src/componentes/</b>	Componentes de <b>UI reutilizables</b> (ej: <a href="#">Navbar</a> , <a href="#">ThemeToggle</a> ).
<b>src/contexts/</b>	Lógica de <b>estado global</b> (ej: <a href="#">ThemeContext.js</a> ).
<b>src/firebase/</b>	Inicialización y configuración del SDK de Firebase.

### 5.2. Dependencias Relevantes (Apéndice E)

Paquete	Rol

<b>react</b>	Framework principal de la UI.
<b>react-router-dom</b>	Manejo de navegación entre <i>/admin</i> y <i>/usuario</i> .
<b>firebase</b>	<b>Backend as a Service</b> (Autenticación y Base de Datos).
<b>@google-cloud/local-auth</b>	Soporte para el login de Google.

## 6. Estándares y Convenciones

- **Convención de Nombres:** Utilizar **PascalCase** para los componentes de React (*MiComponente.jsx*).
- **Estilos:** Los estilos se manejan mediante CSS Modules (*.css* anidados en *componentes/estilos/*).
- **Roles:** La lógica de permisos debe implementarse en la capa de *Routing* (*App.js*) y reforzarse en los componentes.