

Guía de Instalación y Desarrollo - App de Asistencia

Esta guía explica paso a paso cómo configurar el proyecto en un ordenador nuevo para desarrollo y modificaciones.

Requisitos Previos

1. Software Necesario

Node.js y npm

- Versión requerida - Node.js 18 o superior
- Descarga - <https://nodejs.org/>
- Verificar instalación

Bash:

node --version

```
PS C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app> node --version
v22.16.0
```

npm --version

```
PS C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app> npm --version
10.9.2
```

Git

- Descarga: <https://git-scm.com/>
- Verificar instalación:

Bash

git --version

```
PS C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app> git --version
git version 2.49.0.windows.1
```

Editor de Código (Recomendado)

- Visual Studio Code: <https://code.visualstudio.com/>
- Extensiones recomendadas:

- ES7+ React/Redux/React-Native snippets
- TypeScript Importer
- Expo Tools
- React Native Tools

2. Herramientas de Desarrollo Móvil

Para Android

- Android Studio: <https://developer.android.com/studio>
- Configurar emulador Android o habilitar depuración USB en dispositivo físico

Para iOS (Solo en macOS)

- Xcode: Desde Mac App Store
- iOS Simulator: Incluido en Xcode

Instalación del Proyecto

Paso 1: Clonar el Repositorio

```
bash
git clone git@github.com:Adr07/asistencia-app.git
```

```
cd asistencia-app
```

Nota: Entrar en la carpeta del proyecto desde la terminal

Paso 2: Instalar Dependencias

- bash
- npm install
- Resto de dependencias listadas en el archivo requirements.txt

Paso 3: Instalar Expo CLI

```
bash
npm install -g @expo/cli
```

expo --version

```
PS C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app> expo --version
WARNING: The legacy expo-cli does not support Node +17. Migrate to the new local Expo CLI: https://blog.expo.dev/the-new-expo-cli-f4250d8e3421.
6.3.12
PS C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app> []
```

Configuración del Proyecto

1. Configurar Conexión con Odoo

Editar el archivo `components/AttendanceKiosk/otros/config.tsx`:

```
export const RPC_URL = 'http://localhost:8069/jsonrpc';
export const DB = 'nombre-de-tu-base-de-datos';
```

```
// URL del servidor RPC de producción
export const RPC_URL = 'https://registro.sinerkia-dev.com/jsonrpc';

// Nombre de la base de datos en Odoo
export const DB = 'registro';
```

Opcional: Solo en caso de cambiar de servidor de odoo. Puede haber problemas de conexión y compatibilidad revisar antes de cambiar las funciones internas de odoo

2. Verificar Permisos de Ubicación

Archivo `app.json`:

```
{
  "plugins": [
    "expo-router",
    [
      "expo-location",
      {
        "locationAlwaysAndWhenInUsePermission": "Esta aplicación necesita acceso a la ubicación para verificar que estés en el lugar de trabajo al registrar tu asistencia.",
        "locationAlwaysPermission": "Esta aplicación necesita acceso a la ubicación para verificar que estés en el lugar de trabajo al registrar tu asistencia.",
        "locationWhenInUsePermission": "Esta aplicación necesita acceso a la ubicación para verificar que estés en el lugar de trabajo al registrar tu asistencia."
      }
    ],
  ],
}
```

3. Variables de Entorno (Opcional)

```
// URL del servidor RPC de producción
import { Platform } from 'react-native';

// URL del servidor RPC de desarrollo (opcional, para pruebas locales)
export const DEV_RPC_URL = Platform.select({
  ios: 'http://localhost:8069/jsonrpc',
  android: 'http://10.0.2.2:8069/jsonrpc',
  web: 'http://localhost:8069/jsonrpc',
  default: 'http://localhost:8069/jsonrpc', // Valor por defecto para otros
});
```

Nota: Importar e instalar paquete Platform e implementar el código. Permite implementar una url distinta para cada plataforma .El modo web necesita un proxy para poder probar la app en modo desarrollo o usar axios directos

Ejecutar el Proyecto

Modo Desarrollo

Npm run start

```
PS C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app> npm run start

> asistencia-app@1.0.0 start
> expo start

Starting project at C:\Users\User\Documents\Proyectos\App-React\Asistencia\asistencia-app
Starting Metro Bundler



> Metro waiting on exp+asistencia-app://expo-development-client/?url=http%3A%2F%2F192.168.1.136%3A8081
> Scan the QR code above to open the project in a development build. Learn more: https://expo.fyi/start

> Web is waiting on http://localhost:8081

> Using development build
> Press s | switch to Expo Go

> Press a | open Android
> Press w | open web

> Press j | open debugger
> Press r | reload app
> Press m | toggle menu
> shift+m | more tools
> Press o | open project code in your editor

> Press ? | show all commands

Logs for your project will appear below. Press Ctrl+C to exit.
```

Opciones de Ejecución

1. En Navegador Web

- Presiona `w` o abre <http://localhost:8081>

2. En Emulador Android

Necesario:

- Ejecutar emulador
 - Andorid Studio
 - Android IOS Emulator de Visual Studio Code (plugin)

Nota: En el caso del plugin seguir los pasos de instalación según la documentación del plugin

- Presiona `a` o escanea el QR con Expo Go

3. En iOS Simulator

- Presiona `i` o escanea el QR

4. En Dispositivo Físico

- Instala Expo Go y escanea el QR

Desarrollo y Modificaciones

Estructura del Proyecto

(asistencia-app/...)

Archivos Clave

- useAttendanceMain.ts, handleCheck.ts
- index.tsx, attendanceStepsTS/
- odooApi.ts, config.tsx
- useLocation.ts, [useLocationPermission.ts](#)

Build y Distribución

Desarrollo (EAS Updates)

Bash:

```
npm install -g eas-cli  
eas login  
eas update --branch development
```

Producción

Bash:

```
eas build --platform android  
eas build --platform ios  
eas build --platform all
```

Recursos Adicionales

- Expo Docs: <https://docs.expo.dev/>
- React Native: <https://reactnative.dev/>
- TypeScript: <https://www.typescriptlang.org/>

Contacto y Soporte

- Revisar logs en consola
- Verificar configuración Odoo
- Revisar permisos
- Consultar documentación