

# Configuración del Entorno de Desarrollo

---

## Requisitos del Sistema

---

### Windows

- Node.js 18 o superior
- npm o yarn
- Git
- Android Studio
- JDK 11 o superior

## Configuración de Android

---

### 1. Instalar Android Studio

1. Descargar desde <https://developer.android.com/studio>
2. Instalar Android Studio con las opciones por defecto
3. Abrir Android Studio y seguir el Setup Wizard
4. Instalar Android SDK, Android SDK Platform y Android Virtual Device

### 2. Configurar Variables de Entorno

#### Método 1: PowerShell (Temporal)

```
$env:ANDROID_HOME = "C:\Users\[TU_USUARIO]\AppData\Local\Android\Sdk"  
$env:Path += ";$env:ANDROID_HOME\platform-tools;$env:ANDROID_HOME\tools;$env:AND
```



#### Método 2: Variables de Sistema (Permanente)

1. Abrir "Variables de entorno del sistema"
2. Crear nueva variable de sistema:
  - Nombre: ANDROID\_HOME
  - Valor: C:\Users\[TU\_USUARIO]\AppData\Local\Android\Sdk
3. Editar variable Path y agregar:
  - %ANDROID\_HOME%\platform-tools
  - %ANDROID\_HOME%\tools

- %ANDROID\_HOME%\tools\bin

### 3. Verificar Configuración

```
npx react-native doctor
```

### 4. Problemas Comunes

#### ADB no reconocido

- Verificar que `ANDROID_HOME` esté configurado correctamente
- Verificar que `platform-tools` esté en el PATH
- Reiniciar PowerShell/CMD después de configurar variables

#### SDK no encontrado

- Verificar ruta del SDK en Android Studio: File > Settings > System Settings > Android SDK
- La ruta común es: `C:\Users\[USUARIO]\AppData\Local\Android\Sdk`

## Configuración de iOS (solo macOS)

---

### 1. Instalar Xcode

- Descargar desde Mac App Store
- Instalar Command Line Tools: `xcode-select --install`

### 2. Instalar CocoaPods

```
sudo gem install cocoapods
```

### 3. Instalar dependencias iOS

```
cd ios && pod install
```

## Configuración del Proyecto


---

### 1. Instalar dependencias


```
npm install
```

## 2. Configuración específica

### React Native Vector Icons

-  Ya configurado en `android/app/build.gradle`
- Para iOS: verificar fonts en `ios/[AppName]/Info.plist`

### React Native Vision Camera

-  Permisos configurados en `AndroidManifest.xml`
- Para iOS: configurar permisos en `Info.plist`

## 3. Ejecutar el proyecto

### Android

```
npm run android
# o
npx react-native run-android
```

### iOS (solo macOS)

```
npm run ios
# o
npx react-native run-ios
```

## 4. Iniciar Metro Bundler

```
npm start
# o
npx react-native start
```

## Solución de Problemas

---

### Metro Bundler

```
# Limpiar caché
npx react-native start --reset-cache
```

*# O usar las tareas de VS Code configuradas*

## Build de Android

```
# Limpiar build
cd android && ./gradlew clean
cd .. && npx react-native run-android
```

## Logs y Debug

```
# Logs de Android
npx react-native log-android
```

```
# Logs de iOS
npx react-native log-ios
```

## Extensiones Recomendadas de VS Code

---

- React Native Tools
- ES7+ React/Redux/React-Native snippets
- Prettier - Code formatter
- ESLint
- TypeScript Hero
- Auto Rename Tag
- Bracket Pair Colorizer
- GitLens

## Comandos Útiles

---

```
# Verificar entorno
npx react-native doctor
```

```
# Info del entorno
npx react-native info
```

```
# Listar dispositivos Android
adb devices
```

```
# Listar simuladores iOS (macOS)
xcrun simctl list devices
```

```
# Reinstalar node_modules
rm -rf node_modules && npm install

# Reset completo del proyecto
npm run clean-install
```

## Scripts Personalizados del package.json

---

```
# Iniciar Metro Bundler
npm start
```

```
# Ejecutar en Android
npm run android
```

```
# Ejecutar en iOS
npm run ios
```

```
# Ejecutar tests
npm test
```

```
# Linting
npm run lint
```