Setting up a React Native Environment

In this document, we will provide a step-by-step guide for setting up a React Native development environment on a Windows system, with a focus on developing for Android. The environment will include the installation of necessary tools and dependencies, project creation, and running the project on an Android device simulator, as well as additional troubleshooting guidelines and resources.

# System Requirements

Before proceeding, ensure that your system meets the following requirements:

* CPU: A multi-core processor with virtualization extensions (Intel VT-x or AMD-V).
* RAM: 8GB or more recommended.
* Windows version: Windows 10 or later (64-bit version).

# Installation and Configuration Instructions

**1. Node.js and JDK Installation**

1. Run Command Prompt as Administrator.
2. Install Node.js using Chocolatey, a package manager for Windows. It is recommended to use an LTS version of Node.js. Run the following command:

choco install -y nodejs-lts microsoft-openjdk11

1. Verify Node.js installation using the following command:

node -v

1. Ensure that you have Node 16 or newer.

**2. Android Studio Installation**

1. Download and install [Android Studio](https://developer.android.com/studio).
2. During the installation, make sure to select the following components:
   * Android SDK
   * Android SDK Platform
   * Android Virtual Device (AVD)
3. If you are not using Hyper-V, select "Performance (Intel ® HAXM)" or the equivalent for AMD or Hyper-V.

**3. Android SDK Configuration**

1. Open Android Studio.
2. Go to the SDK Manager, which can be found under "More Actions" or in "Settings" > "Languages & Frameworks" > "Android SDK."
3. In the "SDK Platforms" tab, check "Show Package Details."
4. Locate "Android 13 (Tiramisu)" and ensure the following items are checked:
   * Android SDK Platform 33
   * Intel x86 Atom\_64 System Image or Google APIs Intel x86 Atom System Image
5. Switch to the "SDK Tools" tab and check "Show Package Details" here as well.
6. In the "Android SDK Build-Tools" section, make sure version 33.0.0 is selected.
7. Click "Apply" to download and install the Android SDK and build tools.

**4. Environment Variable Configuration**

1. Open the Windows Control Panel.
2. Click on "User Accounts," then "User Accounts" again.
3. Click on "Change my environment variables."
4. Click "New" to create a new user variable named **ANDROID\_HOME** that points to the path of your Android SDK:

Note: Default location is: %LOCALAPPDATA%\Android\Sdk

1. Open a new Command Prompt to ensure the new environment variable is loaded.

**5. Add platform-tools to Path**

1. Open the Windows Control Panel.
2. Click on "User Accounts," then "User Accounts" again.
3. Select the "Path" variable and click "Edit."
4. Click "New" and add the path to platform-tools:

Note: Default location is: %LOCALAPPDATA%\Android\Sdk\platform-tools

# Project Creation

Now that the environment is set up, let's create a new React Native project.

1. Open a Command Prompt in the directory where you want to create the project.
2. Run the following command to create a new React Native project named "TodoListApp":

npx react-native init IncredibleTodoListApp

# Running the Project

To run the React Native project, follow these steps:

1. Start Metro, the JavaScript bundler, by running the following command inside your project folder:

npm start

1. In a new terminal inside your project folder, start the application:

npm run android

If everything is configured correctly, you should see your new app running in an Android emulator.

# Troubleshooting

If you encounter issues during setup or development, refer to the troubleshooting section in the official React Native documentation for solutions to common problems which can be found [here](https://reactnative.dev/docs/troubleshooting).

# Resources

* [React Native Official Documentation](https://reactnative.dev/docs/getting-started)
* [React Native CLI GitHub Repository](https://github.com/facebook/react-native)
* [Android Studio Installation Guide](https://developer.android.com/studio/install)
* [Node.js Downloads](https://nodejs.org/en/download/)

Congratulations! You have successfully set up a React Native environment, created a new project, and launched it on an Android emulator. You are now ready to start developing your mobile app.