**React Native Development Environment Setup**

**System Requirements**

**CPU**: A multi-core Intel i7 processor is advisable, ideally equipped with virtualization capabilities.

**RAM**: The RAM of the system is 16 GB.

**Storage**: An SSD with 1 TB is used for this system.

**Operating System**: Windows 11 is the current operating system.

**Installation Instructions**

To set up your system for React Native development, execute the following steps meticulously to install all required tools and dependencies:

1. **Node.js Installation**:

- Acquire the most recent Long-Term Support (LTS) version of Node.js directly from [Node.js Website](https://nodejs.org/).

- Activate the installer and comply with the instructions to finalize the Node.js installation.

2. **Android Studio Setup**:

- Install Android Studio, ensuring the inclusion of Android SDK, Android SDK Platform, and Android Virtual Device.

- Dismiss project import prompts in the installation wizard.

- Opt-in or out of data sharing with Google Analytics based on your preference.

- Once Android Studio is open, navigate to "More Actions," access the "SDK Manager," and under "SDK Platforms," select "Show Package Details" located in the bottom-right corner.

- Choose "Android SDK Platform 33" and the appropriate system image (Intel x86 Atom\_64 System Image or Google APIs Intel x86 Atom System Image).

- Verify the selection of "Android SDK Build Tools" in the "SDK Tools" section.

- Add the "Android Home" environment variable to your computer's system variables.

- To configure this, access the Windows Control Panel, navigate to "User Accounts," and modify the environment variables accordingly.

- Confirm the addition of the environment variable via the Command Prompt using the command `Get-ChildItem -Path Env:\`.

- Append the platforms' tools path to the system's environment variables through the Control Panel.

**Initiating a New React Native Project**

To create a new React Native application, follow these uncomplicated steps:

1. Establish a dedicated directory for your project files.

2. Launch the command prompt or terminal within the newly created project directory.

3. Execute the command `npx react-native@latest init AwesomeProject` to instantiate a fresh project named `AwesomeProject`.

**Running the Project**

To run your project in an Android Device Simulator, follow these steps:

1. **Start the Android Device Simulator:**

- Open Android Studio.

- Click on `AVD Manager`.

- Start your preferred emulator.

- Download the R image for the device and then click download device.

2**. Run the React Native Project:**

- In the command prompt or terminal, navigate to your project directory.

- Run `react-native run-android` to build your app and run it on the emulator.

**Troubleshooting**

Common issues can arise during setup or development. Here are a few tips:

1. One of the errors I faced during the installation process was that I didn’t get to install the Intel x86 Atom\_64 System Image which was missing so I had to go to the SDK manager inside more actions and then install it.

2. The other issue I faced was that I couldn't find the location of the Android Studio SDK Manager and where it was located so the location is at the top of the SDK Manager in the more actions and then u can add the environment variable.

3. The Android Home variable was not added on the first time, I had to go and add it again and that’s when it was added to the system.

**Resources**

For further learning and troubleshooting, refer to the following resources:

- React Native Official Documentation( [Setting up the development environment · React Native](https://reactnative.dev/docs/environment-setup?guide=native))