**System Requirements**

* **Processor:** AMD Ryzen 7 6800H with Radeon Graphics, 3.20 GHz
* **Installed RAM**: 16.0 GB (15.2 GB usable)
* **System Type:** 64-bit operating system, x64-based processor
* **Windows:** Windows 11 Home Single Language
  + **Version:** 22H2
  + **OS Build:** 22621.3007
* **Graphics:** Nvidia GeForce RTX 3060 Laptop GPU

**Installation Instructions**

**Node.js and npm**

**Downloading Node.js:**

* + Visit the [official Node.js website](https://nodejs.org/en) to download the Long Term Support (LTS) version. The LTS version is recommended for its stability and long-term support.
  + Choose the Windows Installer (.msi) option, selecting the 64-bit version to match the system requirements.

1. **Installing Node.js:**
   * Run the downloaded installer, following the on-screen instructions. The installer will set up both Node.js and npm on your system.
   * opt for the default installation settings to ensure all necessary components are installed, including npm.
2. **Verifying the Installation:**
   * To confirm that Node.js and npm are correctly installed, open a command prompt and enter:

***node -v***

This command displays the installed version of Node.js.

***npm -v***

This command displays the installed version of npm. Successful installation is indicated by the version numbers being displayed.

**Java Development Kit (JDK)**

**Downloading JDK 17:**

* + Access the [Oracle's JDK 17 archive page](https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html). JDK 17 is chosen for its compatibility with Android development tools.
  + Select the Windows x64 Installer (.msi file) for download, ensuring compatibility with the system's architecture.

1. **Installing JDK 17:**
   * Execute the downloaded MSI installer, following the prompts to install JDK on your system. The default installation options should suffice for most development needs.

**Android Studio and SDK**

**Installing Android Studio:**

* + Download Android Studio from the official download page.
  + Launch the downloaded installer, selecting options to install Android Studio along with the Android SDK, Android SDK Platform, and Android Virtual Device for emulating Android devices.

This build is using Android 14 and Build API tools 34.

**Configuration Steps**

**Setting Environment Variables**

Environment variables such as **ANDROID\_HOME** and **JAVA\_HOME** are crucial for development tools to locate the Android SDK and JDK installations on your system.

* **ANDROID\_HOME:** This variable points to the location of your Android SDK. The default path is usually **%LOCALAPPDATA%\Android\Sdk**.
* **JAVA\_HOME:** Set this variable to the path where JDK 17 is installed, facilitating tools to find the Java compiler and other utilities.

To set these environment variables:

1. Open the System Properties dialog by right-clicking on 'This PC' and selecting 'Properties' → 'Advanced system settings.'
2. Click on 'Environment Variables.'
3. Under 'System Variables,' click 'New' to add **JAVA\_HOME** and **ANDROID\_HOME** variables, providing the respective paths to JDK and Android SDK installations.
4. To modify the 'Path' variable, add the path to the **platform-tools** directory within the Android SDK installation path. This allows access to Android Debug Bridge (ADB) and other platform tools from the command line.

**Changing Gradle Version:**

1. Navigate to android > gradle > wrapper and open gradle-wrapper file.
2. Change the 8.3 in the value of distributionUrl to 8.5 or whatever version you want to.

**Creating and Running the React Native Project**

1. **Project Initialization:**
   * Open a command prompt and navigate to the directory where you want to create the "SugoiToDoListApp."
   * Initialize the project with React Native CLI using:

***npx react-native@latest init SugoiToDoListApp***

* + ***This*** command creates a new directory named "SugoiToDoListApp" with all the necessary React Native files and dependencies.

1. **Running the Project on a Physical Android Device:**
   * Connect your Android device to the computer via USB, ensuring USB debugging is enabled on the device.
   * Verify the device connection by running **adb devices**. Your device should be listed under "Connected devices."
   * Navigate to the project directory and start the application with:

***npm start***

In a new command prompt window, execute:

***npm run android***

* + This compiles the React Native app and installs it on the connected Android device.

**Troubleshooting**

**Common Issues and Solutions**

* **Android Development Environment Error:** If you encounter an error indicating that the Android development environment is not correctly set up, particularly after executing **npm run android**, check the following:
  + Verify that **JAVA\_HOME** is correctly set, pointing to the JDK installation path.
  + Ensure the Android SDK path is correctly specified in **ANDROID\_HOME** and that the SDK version matches the project's requirements.
* **Java Version Error:** For errors related to the Android Gradle plugin requiring a specific Java version, consult the following resources for guidance on configuring your development environment to use the correct Java version:
  + Stack Overflow discussion on Android Studio error regarding Java version: [Android Studio Error: Java Version](https://stackoverflow.com/questions/66980512/android-studio-error-android-gradle-plugin-requires-java-11-to-run-you-are-cur)

**Resources**

* Node.js Download: <https://nodejs.org/en>
* JDK 17 Archive Downloads: <https://www.oracle.com/java/technologies/javase/jdk17-archive-downloads.html>
* Running on Device - React Native Documentation: <https://reactnative.dev/docs/running-on-device>
* Android Studio Download: https://developer.android.com/studio