Name: Jhomer Clemente  
  
**Lab Assignment 1: Spinning Up React Native App**

**1. System requirements: Document specifications of the system being used for React Native development. This includes the CPU, RAM size, and Windows version.**

Processor: 11th Gen Intel(R) Core(TM) i7-11370H @ 3.30GHz 3.30 GHz RAM: 4 GB or higher.

Ram: 16.0GB.

Version: Windows 11 Home, Version 22H2, OS build 22621.3007.

**2. Installation instructions: Include step-by-step instructions for installing the necessary tools and dependencies required for the framework, such as Node.js, and the React Native CLI.**

* Visit the Official React Native Website:
  + Go to https://reactnative.dev/ to access the official React Native website.
* Navigate to Development Guides:
  + Click on "Development" and locate the "Guides" section under the "Development" category.
* Access Environment Setup Guide:
  + Navigate to "Environment setup," locate the "Setting Up the Development Environment" section and access the guide within the "Environment Setup" section for instructions on setting up the development environment.
* Select React Native CLI Quickstart:
  + Select the "React Native CLI Quickstart" tab to access comprehensive setup instructions.
* Specify Development and Target OS:
  + Select "Windows" as the Development OS and choose "Android" as the Target OS.
* Verify Node.js Installation:
  + Ensure Node.js is installed on your system, preferably using the Long Term Support (LTS) version.
* Download and Configure Android Studio:
  + Download Android Studio from the official website.
  + Run the setup with default options.
  + Finish the setup and choose not to import any existing configurations.
* Privacy Settings in Android Studio:
  + Decide whether to share usage statistics with Google during the Android Studio setup.
* Standard Android Studio Configuration:
  + Click "Next," keep the "Standard" configuration selected, and proceed with the Android Studio setup.
* Accept License Agreements:
  + Accept all license agreements to finalize the Android Studio installation.
* Access Additional Options in SDK Manager:
  + On the create project screen, choose "More Options" and then select "SDK Manager."
* Ensure Android API (34) in SDK Platforms:
  + In the SDK Platforms section, verify the existence of Android API (34). Make sure to select "Android SDK Platform" by clicking on "Show Package Details."
* Verify Necessary Tools in SDK Tools:
  + In the SDK Tools section, ensure that Android SDK Build Tools 34, Android Emulator, and Intel Accelerator are installed.
* Configure Environment Variables - Set Up ANDROID\_HOME:
  + Right-click on the Windows Start menu and go to "Settings."
  + In the search bar, type "environment variables."
  + Choose "Edit environment variables for your account."
  + Click "New" and set up a variable named ANDROID\_HOME, using the SDK's installation location as the value.
* Update Path Variable - Include platform-tools:
  + In the same environment variables window, select "Path" and click "Edit."
  + Add the path to the platform-tools directory under the SDK to the existing list.

**3. Configuration steps: Detail any necessary configuration steps required to set up the framework, such as setting environment variables or configuring project settings.**

* Progress through the installation wizard by clicking "Next" and maintaining the "Standard" configuration before clicking "Next" again.
* Agree to all license agreements and complete the installation process.
* While creating a project, access additional options by selecting "More Options" and navigating to the "SDK Manager."
* In the SDK Platforms section, confirm the presence of Android API (34) and ensure that "Android SDK Platform" is selected by clicking "Show Package Details."
* Under SDK Tools, ensure that Android SDK Build Tools 34, Android Emulator, and Intel Accelerator are installed. Confirm the selections by clicking "OK."
* Open the Windows Start menu, right-click, and select "Settings."
* Type "environment variables" in the search bar and edit environment variables for your account by selecting "Edit."
* Create a new variable named ANDROID\_HOME by clicking "New" and paste the SDK's installation location as the value. To find it, go to Android Studio, click "More Options," "SDK Manager," and locate the information at the top.
* Select the "Path" variable and click "Edit," then add the path to the platform-tools directory within the SDK to the existing list and click "OK."

**4. Project creation: Outline the steps involved in creating a new project using the framework, including any necessary setup or configuration.**

* Open File Explorer and Navigate to Project Folder:
  + Open the file explorer and go to the folder where you want to create the new React Native project.
* Open Command Prompt (CMD):
  + In the search bar, type "cmd" and press enter to open the command prompt.
* Initialize React Native Project:
  + In the command line, copy and paste the following: npx react-native@latest init
  + Hit space and type the desired project name, for example, "IncredibleToDoListApp," then press enter.
* Confirm Project Initialization:
  + Type "y" and press enter to confirm the project initialization.
* Update npm:
  + After project initialization is complete, run the following command to update npm: npm i -g npm
* Open Project in Visual Studio Code:
  + Open Visual Studio Code, click on "File," then choose "Open Folder." Select the folder where the project was created and click "Select."
* Change App File to JavaScript:
  + To convert the App file to JavaScript, right-click on "App.tsx," choose "Rename," rename it to "App.jsx," and eliminate all lines with errors.
* Create Emulator Device:
  + Click "Create Device" or use the "+" sign.
  + Choose the device type and select a device model Pixel 7
  + Click the download button next to "R," and once the download is complete, click "Finish." Now, choose "R" and click "Next."
* Configure Device Settings:
  + Select "Portrait Mode" for the device orientation and click "Finish."
* Start the Emulator:
  + Click the play button on the created emulator to initiate the Android emulator.

**5. Running the project: Detail how to run the project in an Android Device Simulator.**

* Open Android Studio and navigate to your project by clicking "Open," selecting the folder where your project is located, and choosing the "android" folder.
* Build the project by clicking on "Build" and "Make Project." Once the build is complete, run the app by clicking "Run" and "Run app."
* If you encounter an "Unable to Load Script" error, resolve it by going back to the command line in the project directory. Type and enter the command npm run start and then press 'r' to reload the app.
* Within the project directory in the command line, start the project by typing and entering the command npm run start. Press 'a' to launch the app.

**6. Troubleshooting: Include information on how to troubleshoot common issues that may arise during setup or development, such as debugging and error messages. This may need to be updated in the future.**

* Encountered an issue with the npm run start command, resulting in an "Unable to find script" error. Resolved by checking the package.json file for script configuration and ensuring the correct script name was used.
* Ran into a problem with the application not rendering properly on the device. Discovered it was caused by a missing dependency, and the issue was resolved by installing the required library using npm install.

**7. Resources: Provide links to additional resources and documentation that can be used for reference and further learning, such as official documentation, tutorials, and Stack Overflow answers.**

https://github.com/