React Native Development Report

Priyal Dharmendra Patel

November 2024

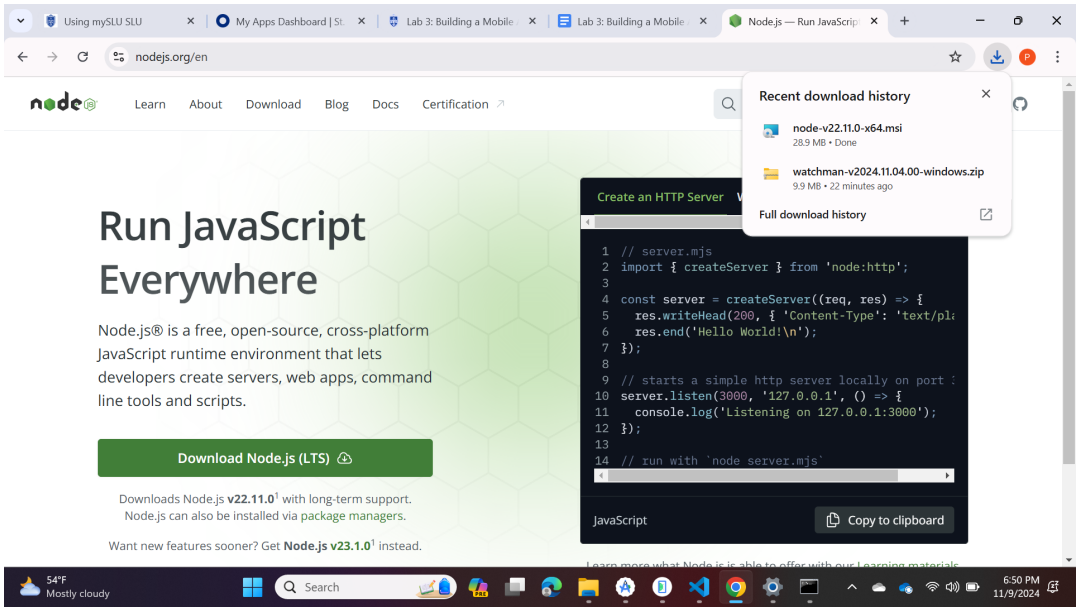
Github:

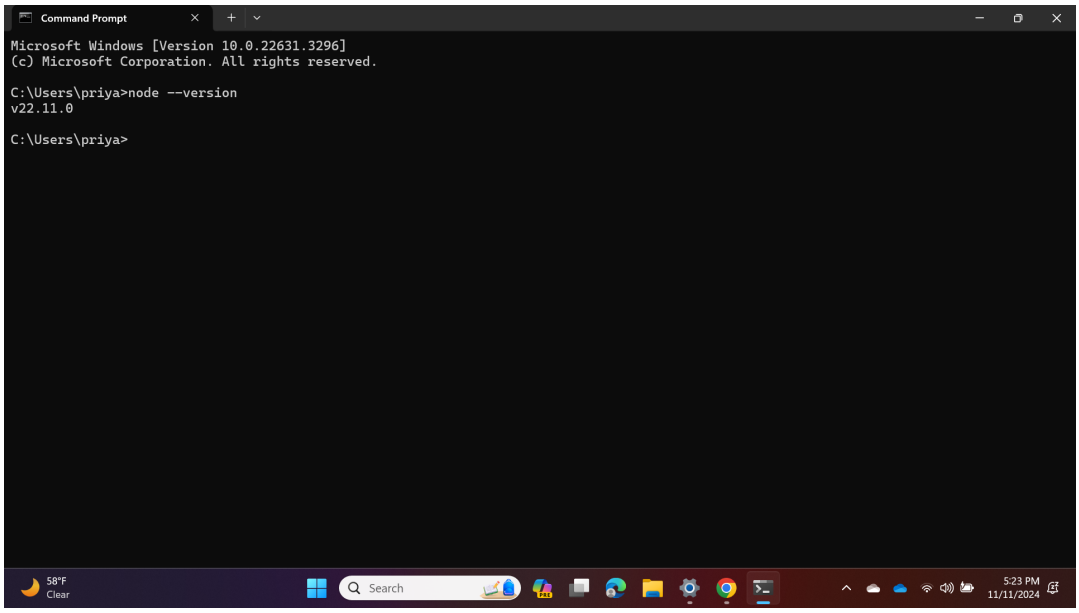
https://github.com/mepripri/SimpleTodoApp.

Task 1: Set Up the Development Environment

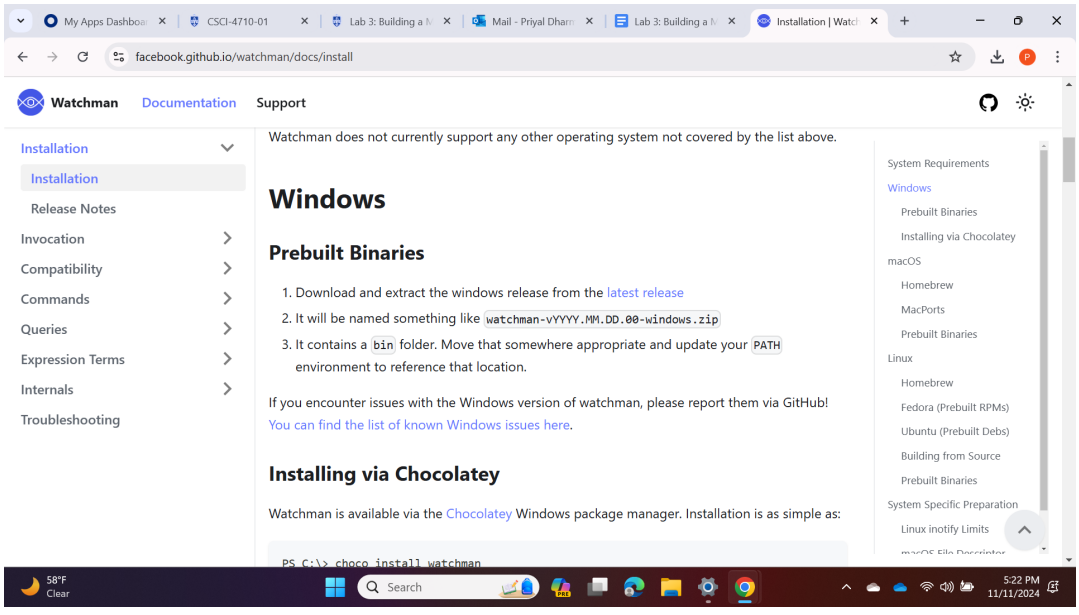
Step 1: Install Node.js and Watchman

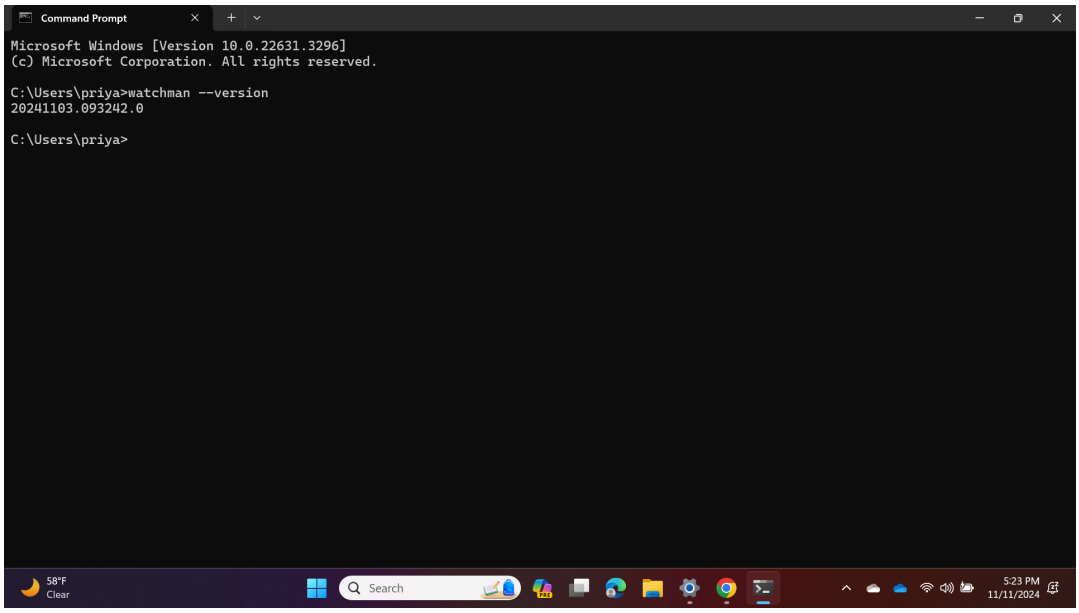
- **Install Node.js:**





- **Install Watchman (Optional for macOS/Linux):**



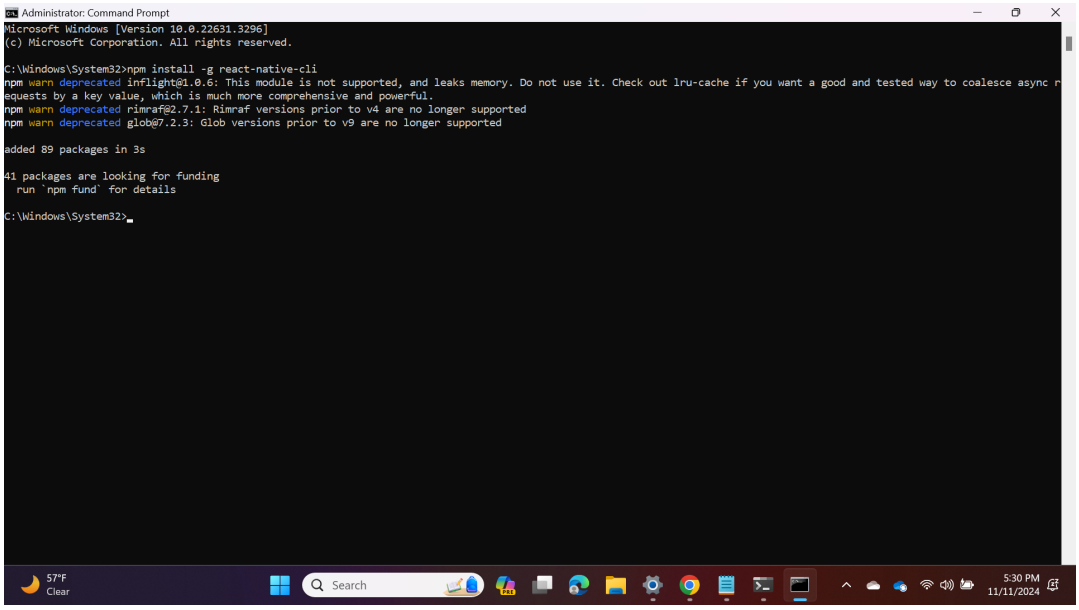


Step 2: Install React Native CLI

The React Native CLI allows you to create and manage React Native projects.

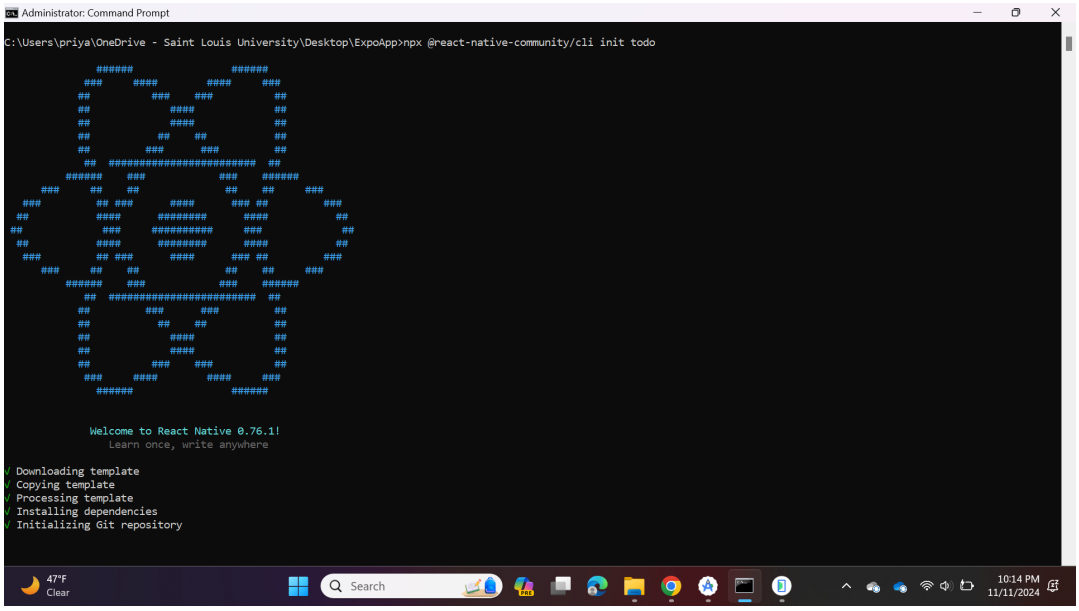
- Open your terminal and run:

```
npm install -g react-native-cli
```



- If the global package is deprecated, use the local version with:

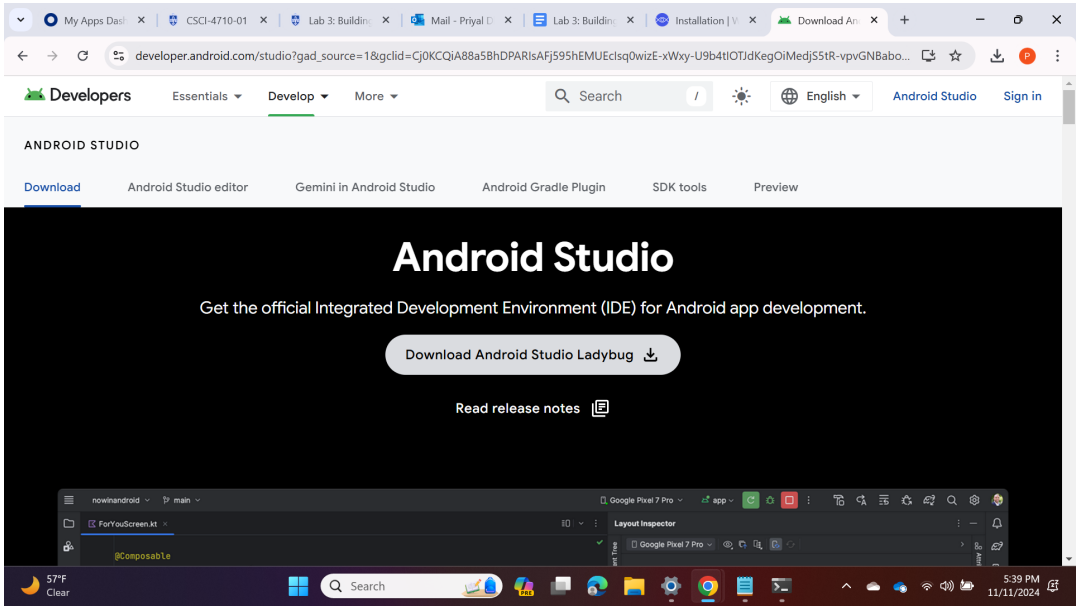
  ```
  npx react-native init YourProjectName
  ```



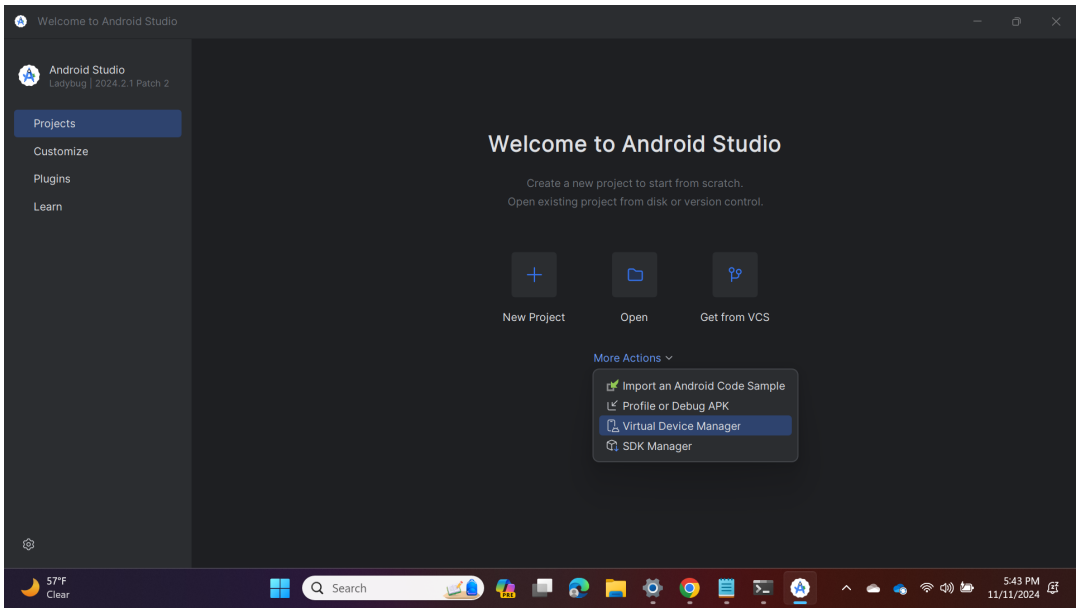
Step 3: Set Up Android Studio (or Xcode for iOS)

For Android Users:

1. Download and install Android Studio from https://developer.android.com/studio.

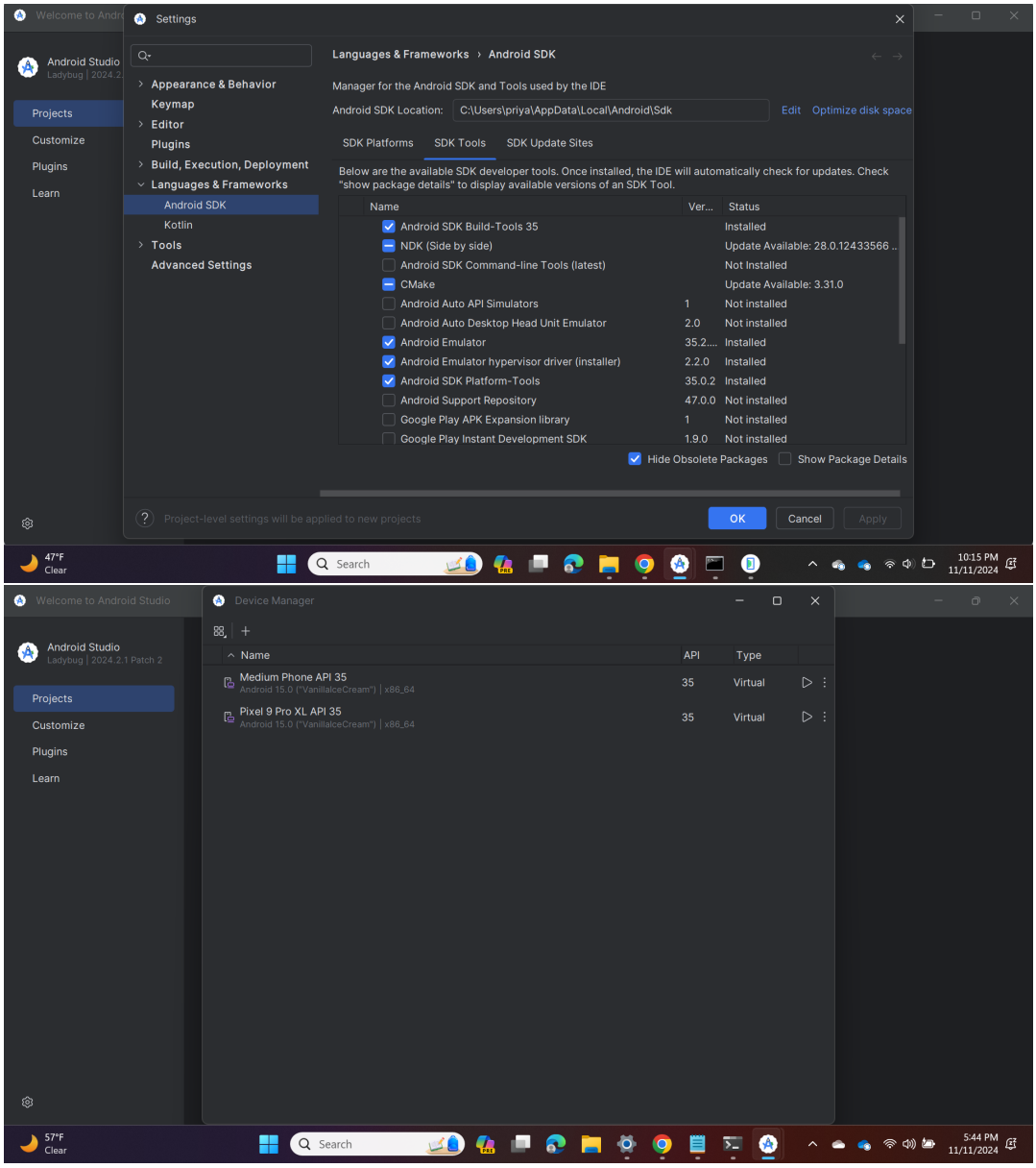


1. Configure Android Studio:
   Preferences ¿ Appearance & Behavior ¿ System Settings ¿ Android SDK



1. Enable the following recommended SDK tools:
   - Android SDK Build-Tools (latest version)

- Android SDK Platform-Tools
- Android Emulator
- Google Play Services (if required)

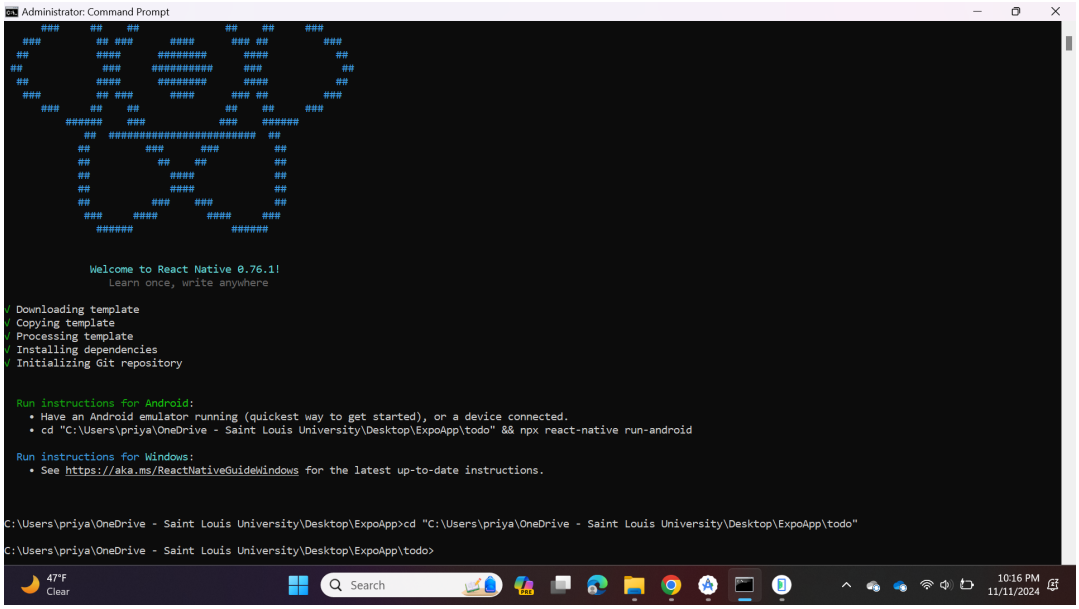


Step 4: Create a New React Native Project

- Run the following commands:

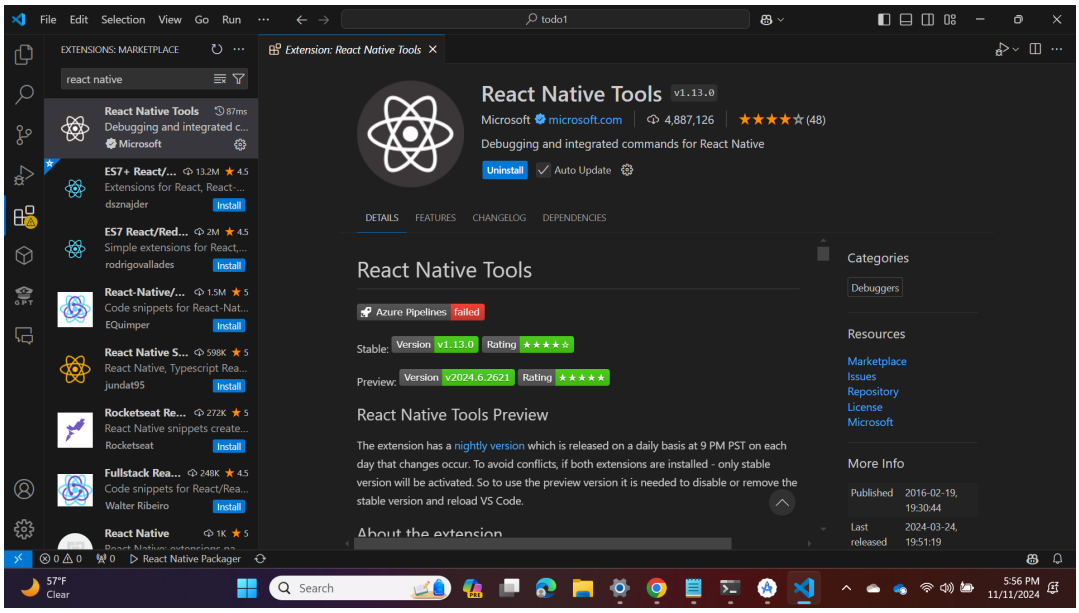
```
npx react-native init YourProjectName
cd YourProjectName
```

- This creates a new React Native project.

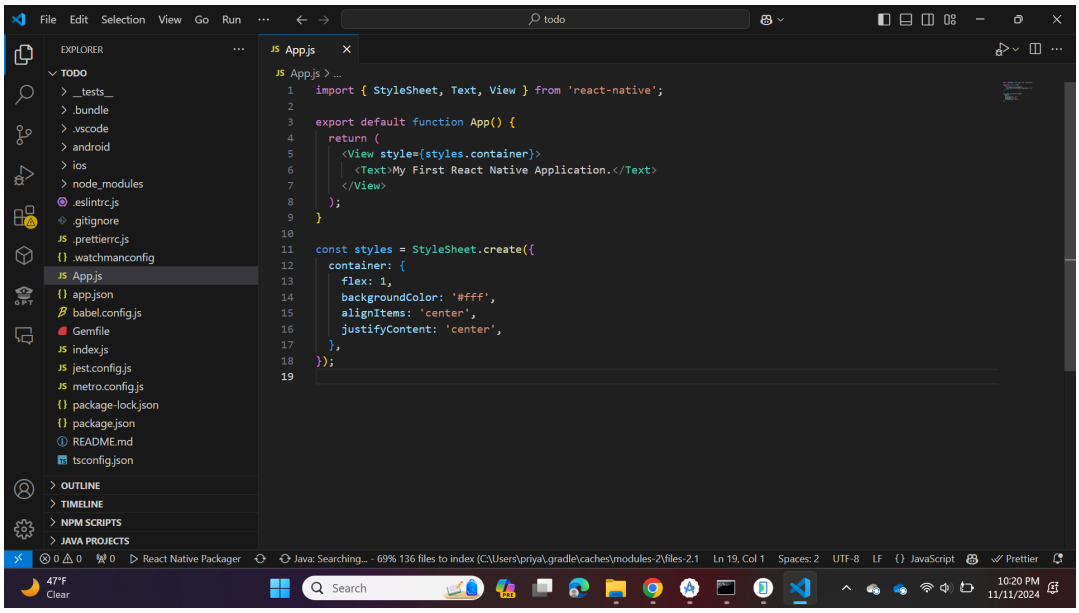


Step 5: Open the Project in Visual Studio Code

- Launch Visual Studio Code.
- Open the project folder via **File ¿ Open Folder**.
- Install the React Native Tools extension for enhanced development experience.



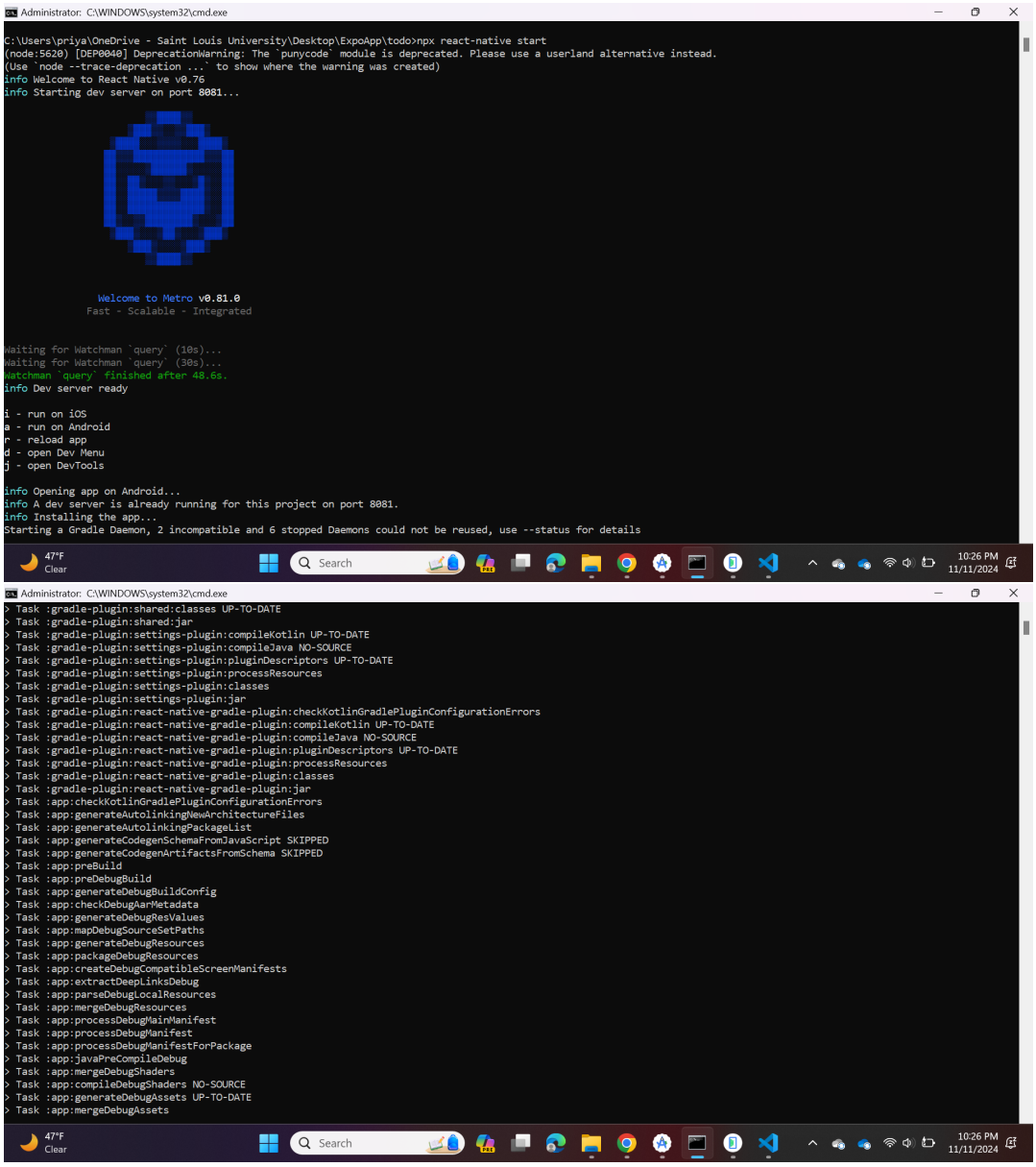
- Modify `App.js` to display *"My First React Native Application"*.

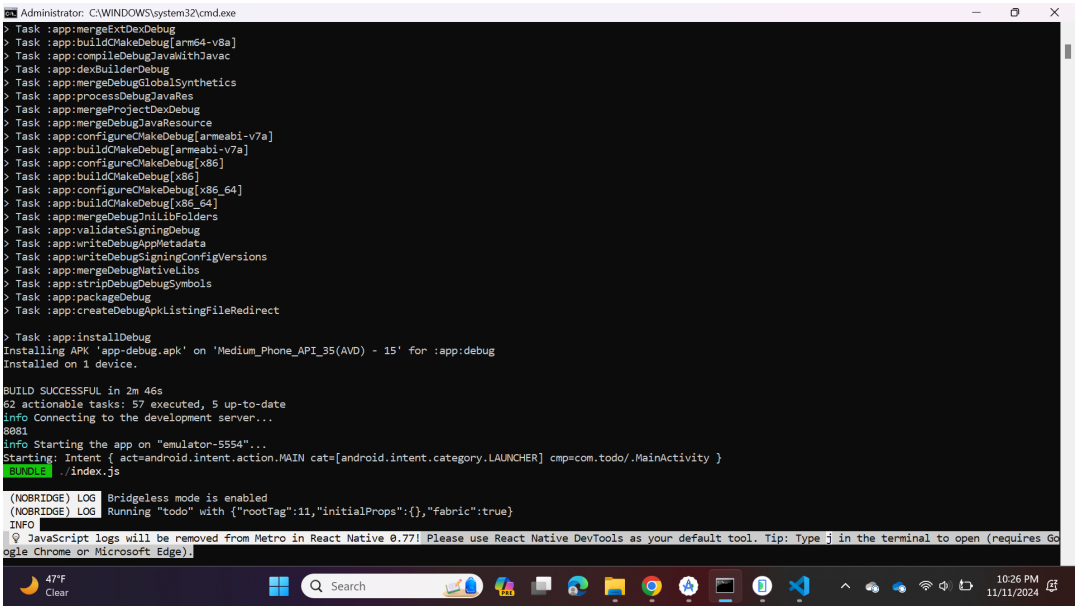


Step 6: Start the Metro Bundler

- Run the following command in the terminal:

```
npx react-native start
```


Administrator: C:\WINDOWS\system32\cmd.exe
C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoApp\todo>npx react-native start
(node:5620) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
info Welcome to React Native v0.76
info Starting dev server on port 8081...
Welcome to Metro v0.81.0
Fast - Scalable - Integrated
Waiting for Watchman `query` (10s)...
Waiting for Watchman `query` (30s)...
Watchman `query` finished after 48.6s.
info Dev server ready
i - run on iOS
a - run on Android
r - reload app
d - open Dev Menu
j - open DevTools
info Opening app on Android...
info A dev server is already running for this project on port 8081.
info Installing the app...
Starting a Gradle Daemon, 2 incompatible and 6 stopped Daemons could not be reused, use --status for details
47°F
Clear
Search
10:26 PM
11/11/2024
Administrator: C:\WINDOWS\system32\cmd.exe
> Task :gradle-plugin:shared:classes UP-TO-DATE
> Task :gradle-plugin:shared:jar
> Task :gradle-plugin:settings-plugin:compileKotlin UP-TO-DATE
> Task :gradle-plugin:settings-plugin:compileJava NO-SOURCE
> Task :gradle-plugin:settings-plugin:pluginDescriptors UP-TO-DATE
> Task :gradle-plugin:settings-plugin:processResources
> Task :gradle-plugin:settings-plugin:classes
> Task :gradle-plugin:settings-plugin:jar
> Task :gradle-plugin:react-native-gradle-plugin:checkKotlinGradlePluginConfigurationErrors
> Task :gradle-plugin:react-native-gradle-plugin:compileKotlin UP-TO-DATE
> Task :gradle-plugin:react-native-gradle-plugin:compileJava NO-SOURCE
> Task :gradle-plugin:react-native-gradle-plugin:pluginDescriptors UP-TO-DATE
> Task :gradle-plugin:react-native-gradle-plugin:processResources
> Task :gradle-plugin:react-native-gradle-plugin:classes
> Task :gradle-plugin:react-native-gradle-plugin:jar
> Task :app:checkKotlinGradlePluginConfigurationErrors
> Task :app:generateAutolinkingNewArchitectureFiles
> Task :app:generateAutolinkingPackageList
> Task :app:generateCodegenSchemaFromJavaScript SKIPPED
> Task :app:generateCodegenArtifactsFromSchema SKIPPED
> Task :app:preBuild
> Task :app:preDebugBuild
> Task :app:generateDebugBuildConfig
> Task :app:checkDebugAarMetadata
> Task :app:generateDebugResValues
> Task :app:mapDebugSourceSetPaths
> Task :app:generateDebugResources
> Task :app:packageDebugResources
> Task :app:createDebugCompatibleScreenManifests
> Task :app:extractDeepLinksDebug
> Task :app:parseDebugLocalResources
> Task :app:mergeDebugResources
> Task :app:processDebugMainManifest
> Task :app:processDebugManifest
> Task :app:processDebugManifestForPackage
> Task :app:javaPreCompileDebug
> Task :app:mergeDebugShaders
> Task :app:compileDebugShaders NO-SOURCE
> Task :app:generateDebugAssets UP-TO-DATE
> Task :app:mergeDebugAssets
47°F
Clear
Search
10:26 PM
11/11/2024

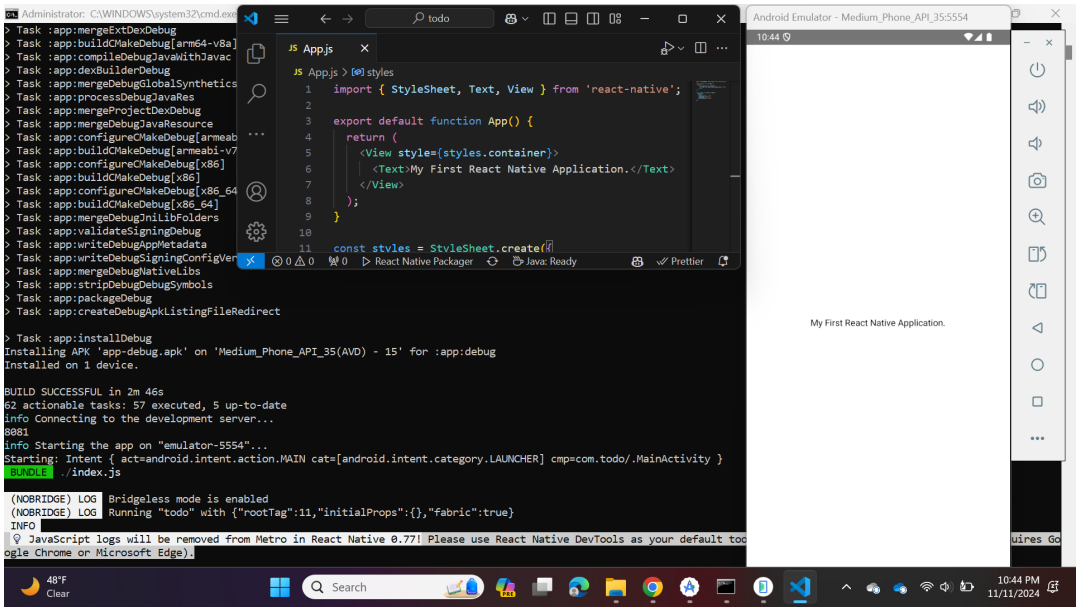


- This starts the Metro Bundler, which monitors your files and serves JavaScript code.

Step 7: Run Your App on an Emulator or Device

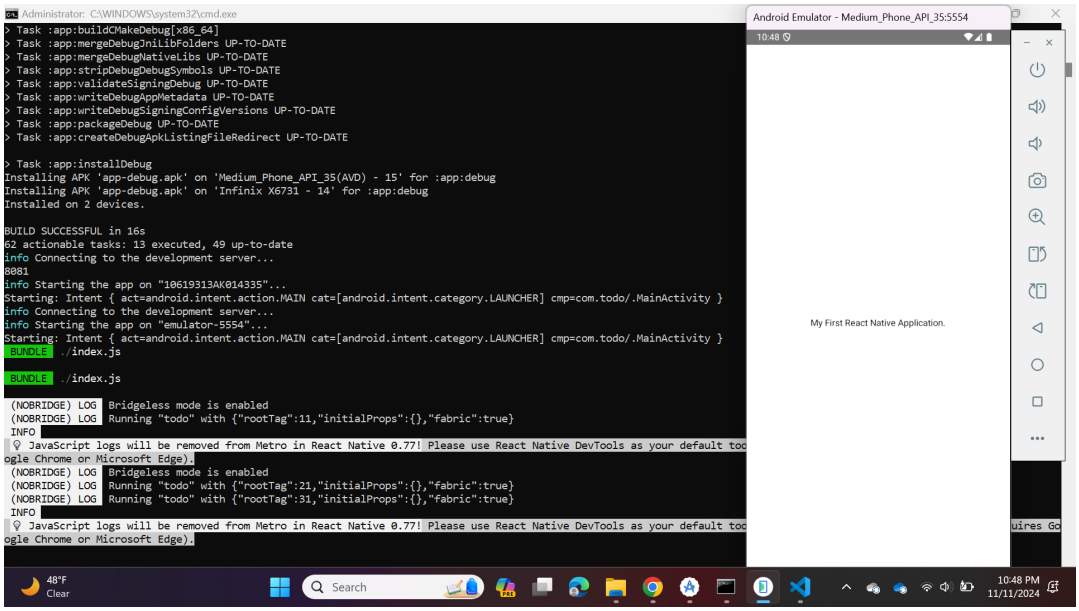
For Android:

1. Ensure an emulator is running via Android Studio's AVD Manager, or connect a physical device with USB debugging enabled.

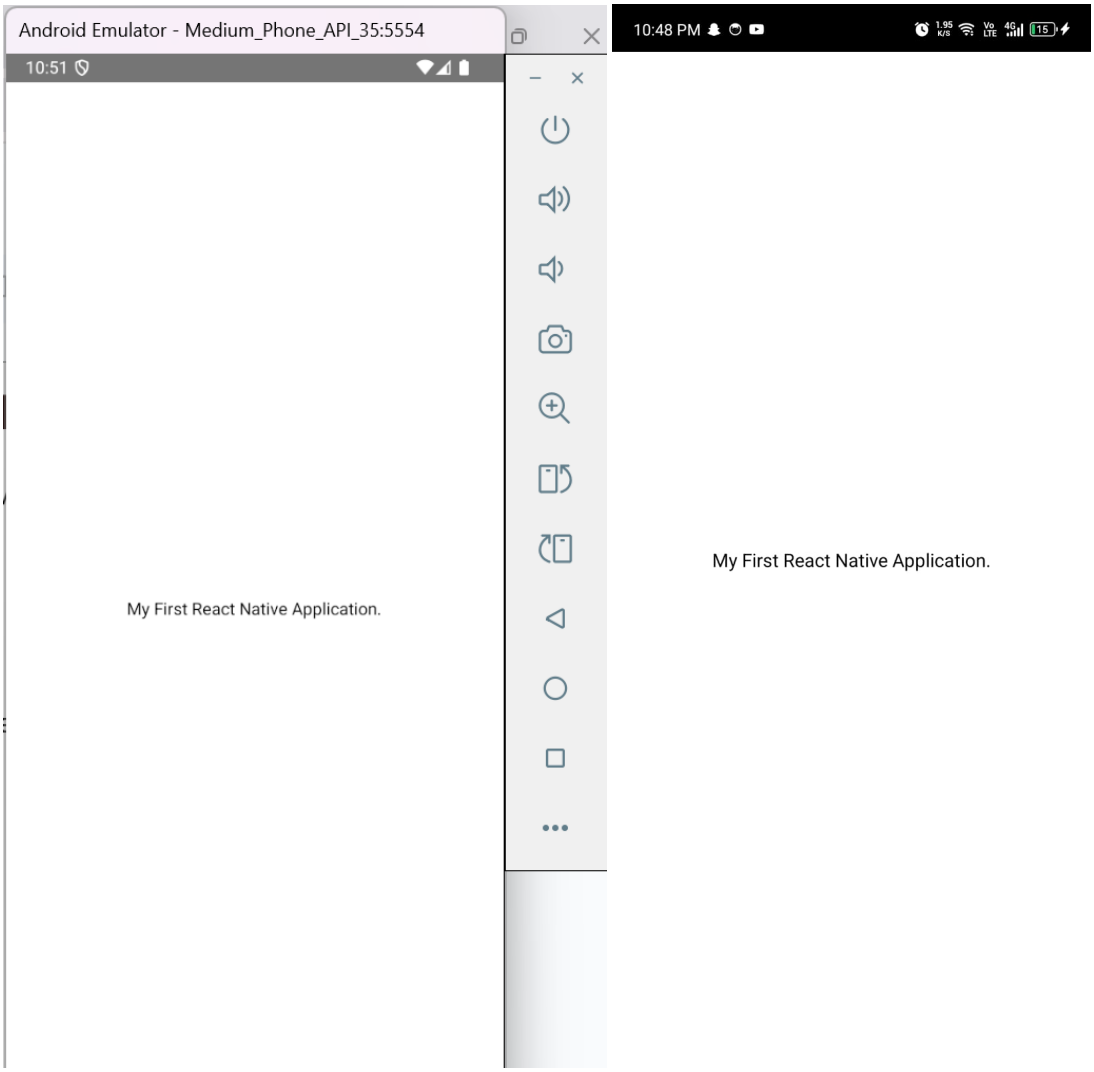


1. Run:

```
npx react-native run-android
```



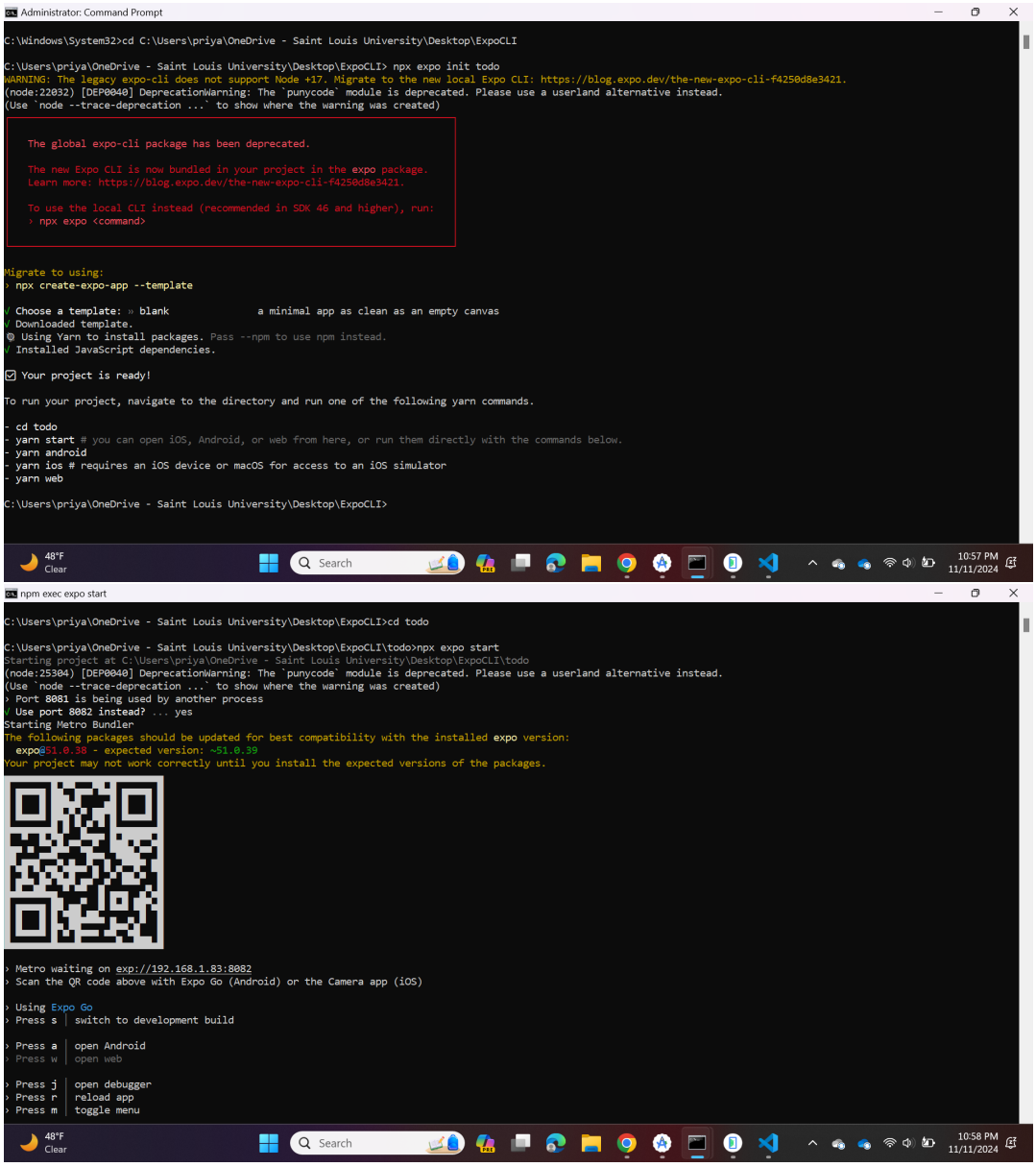
1. This installs and launches the app on the selected device.

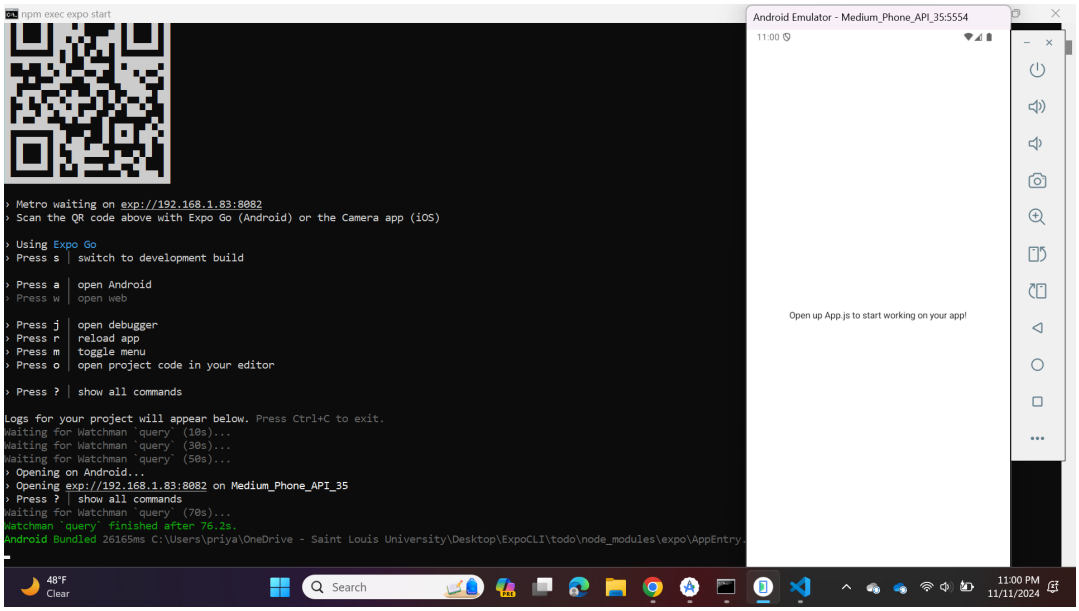


Step 8: Run Your App on a Mobile Device Using Expo

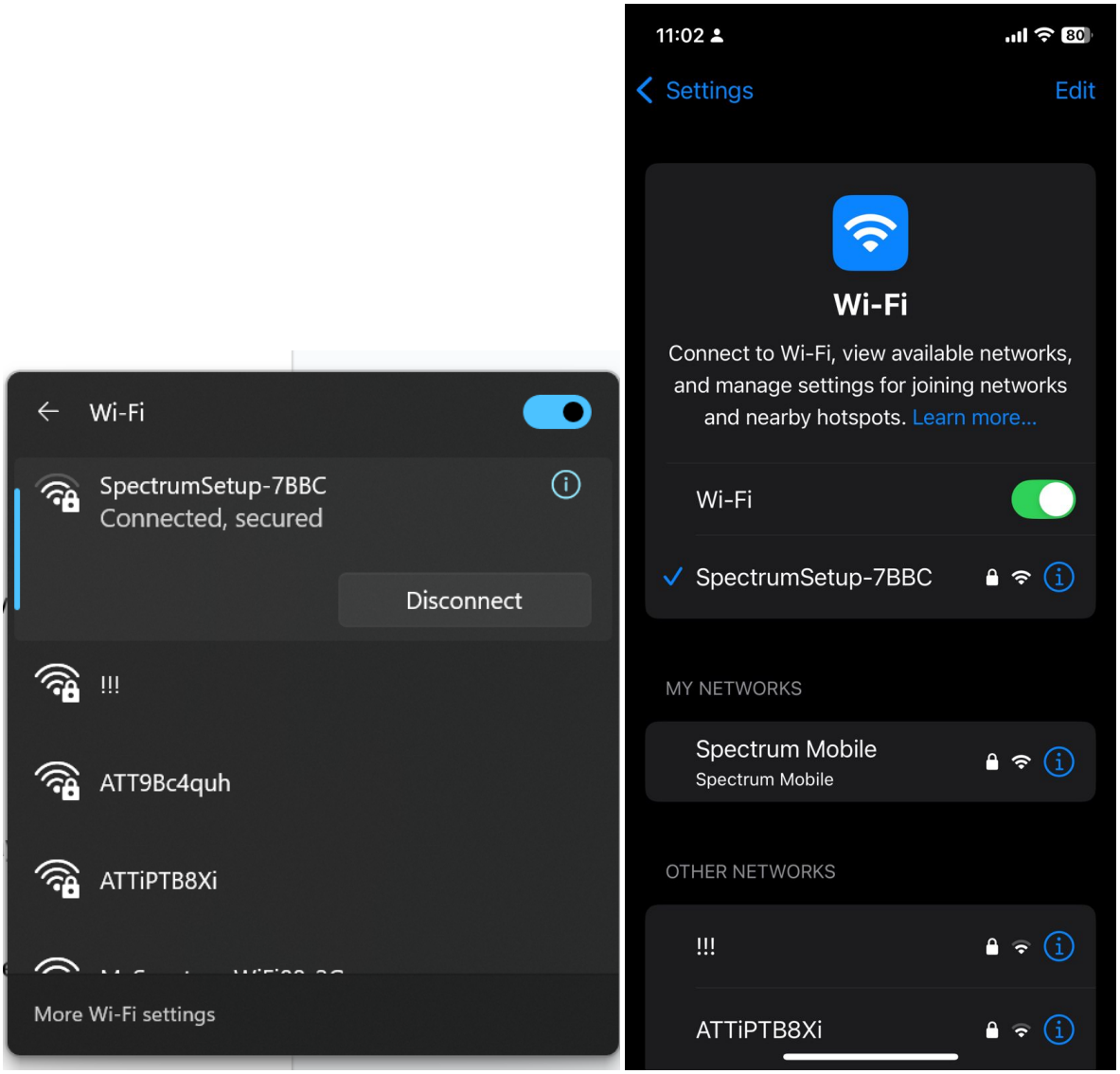
1. Install Expo CLI and create a new project:

```
npm install -g expo-cli
npx expo init YourProjectName
cd YourProjectName
npx expo start
```

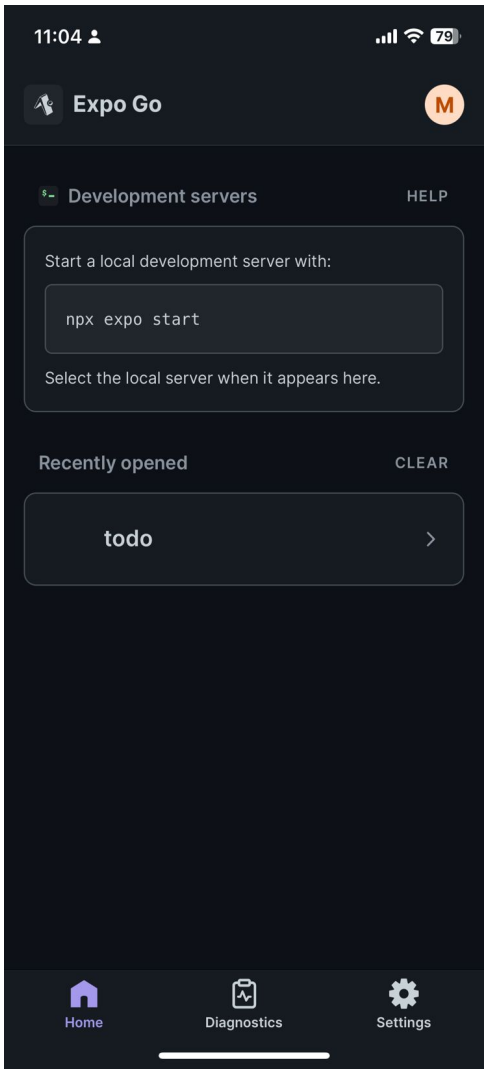

Administrator: Command Prompt
C:\Windows\System32>cd C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI
C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI> npx expo init todo
WARNING: The legacy expo-cli does not support Node +17. Migrate to the new local Expo CLI: https://blog.expo.dev/the-new-expo-cli-f4250d8e3421.
(node:22032) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
The global expo-cli package has been deprecated.
The new Expo CLI is now bundled in your project in the expo package.
Learn more: https://blog.expo.dev/the-new-expo-cli-f4250d8e3421.
To use the local CLI instead (recommended in SDK 46 and higher), run:
› npx expo <command>
Migrate to using:
› npx create-expo-app --template
√ Choose a template: » blank a minimal app as clean as an empty canvas
√ Downloaded template.
Using Yarn to install packages. Pass --npm to use npm instead.
√ Installed JavaScript dependencies.
Your project is ready!
To run your project, navigate to the directory and run one of the following yarn commands.
- cd todo
- yarn start # you can open iOS, Android, or web from here, or run them directly with the commands below.
- yarn android
- yarn ios # requires an iOS device or macOS for access to an iOS simulator
- yarn web
C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI>
48°F Clear
Search
10:57 PM 11/11/2024
npm exec expo start
C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI>cd todo
C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI\todo>npx expo start
Starting project at C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI\todo
(node:25304) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
› Port 8081 is being used by another process
√ Use port 8082 instead? ... yes
Starting Metro Bundler
The following packages should be updated for best compatibility with the installed expo version:
expo@51.0.38 - expected version: ~51.0.39
Your project may not work correctly until you install the expected versions of the packages.
› Metro waiting on exp://192.168.1.83:8082
› Scan the QR code above with Expo Go (Android) or the Camera app (iOS)
› Using Expo Go
› Press s | switch to development build
› Press a | open Android
› Press w | open web
› Press j | open debugger
› Press r | reload app
› Press m | toggle menu
48°F Clear
Search
10:58 PM 11/11/2024



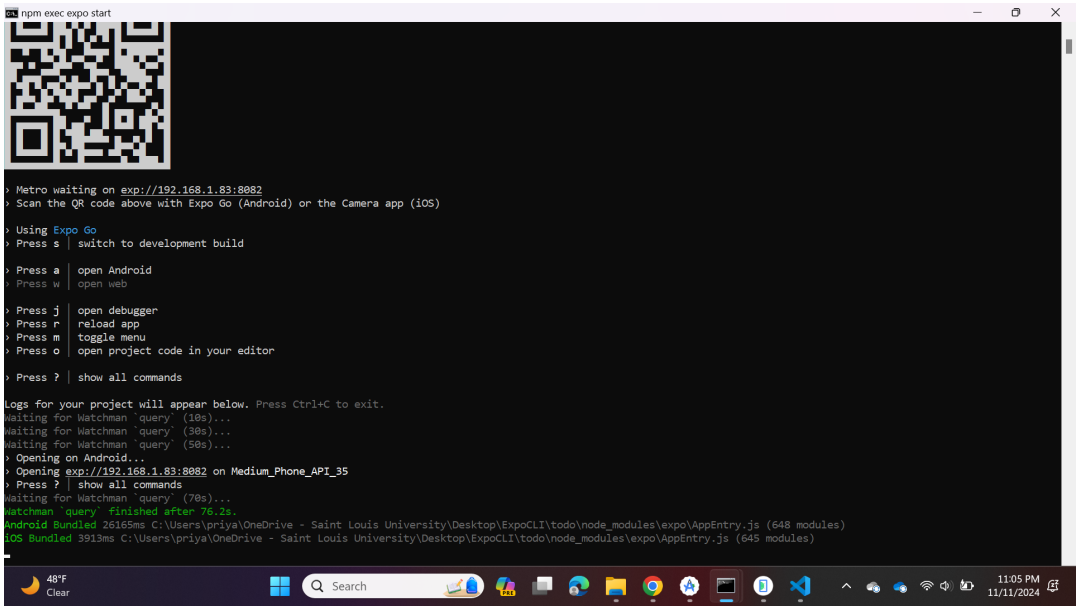
1. Connect your mobile device to the same Wi-Fi network.

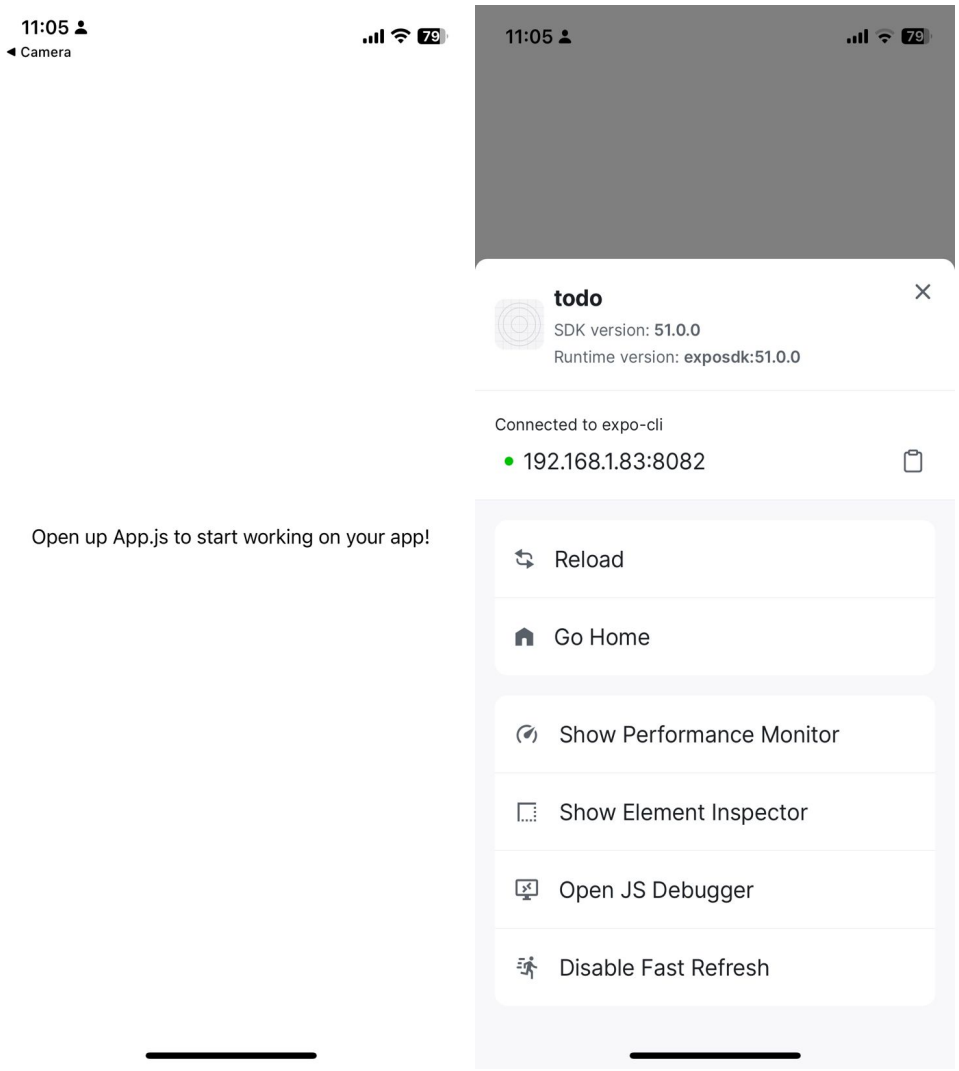


1. Install the Expo Go app from the App Store or Google Play Store.



1. Scan the QR code provided in the Expo developer tools to launch the app.

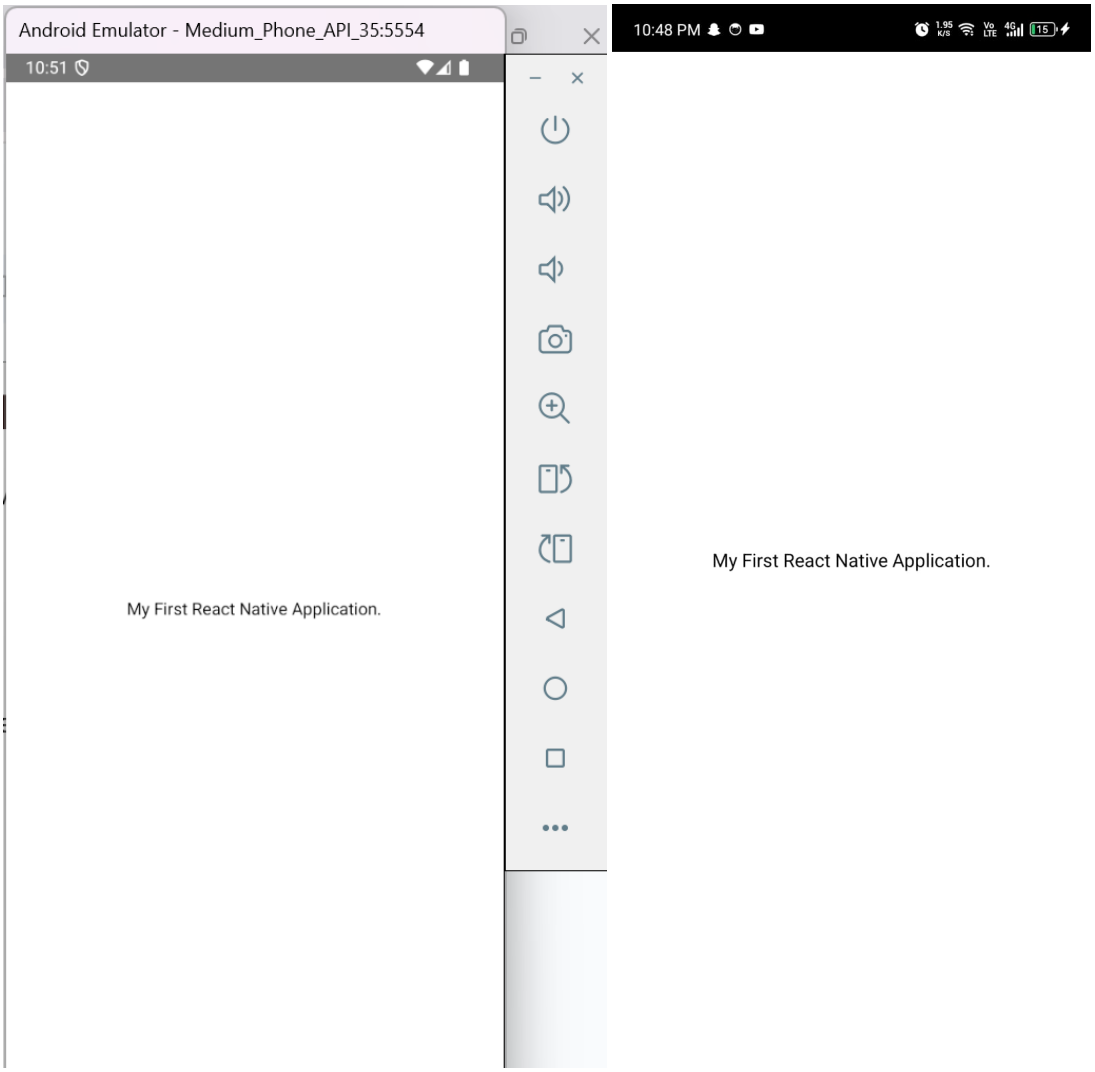

npm exec expo start
› Metro waiting on exp://192.168.1.83:8082
› Scan the QR code above with Expo Go (Android) or the Camera app (iOS)
› Using Expo Go
› Press s | switch to development build
› Press a | open Android
› Press w | open web
› Press j | open debugger
› Press r | reload app
› Press m | toggle menu
› Press o | open project code in your editor
› Press ? | show all commands
Logs for your project will appear below. Press Ctrl+C to exit.
Waiting for Watchman `query` (10s)...
Waiting for Watchman `query` (30s)...
Waiting for Watchman `query` (50s)...
› Opening on Android...
› Opening exp://192.168.1.83:8082 on Medium_Phone_API_35
› Press ? | show all commands
Waiting for Watchman `query` (70s)...
Watchman `query` finished after 76.2s.
Android Bundled 26165ms C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI\todo\node_modules\expo\AppEntry.js (648 modules)
iOS Bundled 3913ms C:\Users\priya\OneDrive - Saint Louis University\Desktop\ExpoCLI\todo\node_modules\expo\AppEntry.js (645 modules)
48°F
Clear
Search
11:05 PM
11/11/2024



What to Submit for Task 1 (Total 40 Points)

1. Screenshots of Your App (5 Points)

- Attach screenshots of your app running on an emulator and on a physical Android or iOS device.



- Describe any differences you observed between running the app on an emulator versus a physical device.

  When running the app on an emulator, it felt slower compared to a physical device, which was much faster. The graphics on the emulator sometimes lagged or looked slightly different, while the physical device displayed the UI more accurately. Also, using the emulator meant relying on a mouse and keyboard for gestures, but the physical device allowed for real touch interactions, making it more intuitive.

2. Setting Up an Emulator (10 Points)

- Explain the steps you followed to set up an emulator in Android Studio or Xcode.
  - I opened Android Studio and went to the AVD Manager.
  - I created a virtual device by selecting a phone model (e.g., Pixel 4).
  - Then, I downloaded the Android version I needed and set the emulator's settings like RAM and storage.
  - After that, I clicked the play button to launch the emulator.
- Discuss any challenges you faced during the setup and how you overcame them.
  - The emulator was really slow.
  - I also had trouble with a specific Android version.

3. Running the App on a Physical Device Using Expo (10 Points)

- Describe how you connected your physical device to run the app using Expo.
  - I downloaded the Expo Go app on my phone.
  - I made sure my phone and computer were on the same Wi-Fi.
  - I started the app on my computer using expo start and scanned the QR code with my phone's Expo Go app.
- Include any troubleshooting steps if you encounter issues.
  - Once, my phone wasn't on the same Wi-Fi as my computer, so I connected them to the same network, and it worked fine.
  - Sometimes, the QR code wouldn't work, so I copied the link from the terminal and opened it manually in the Expo Go app.

4. Comparison of Emulator vs. Physical Device (10 Points)

- Compare and contrast using an emulator versus a physical device for React Native development.
  - Emulator:
    * Slower and sometimes lags.
    * Simulates touch with a mouse, which isn't natural.
    * Easy to set up and test on different virtual devices.

  - Physical Device:
    * Faster and shows real-world performance.
    * Lets you test real touch gestures.
    * Takes more effort to set up but gives accurate results.

- Discuss the advantages and disadvantages of each option.
  - Advantages:
    * Easy to set up and use.
    * Runs faster and shows exactly how users will experience the app.
    * Can test on lots of "fake" devices with different screen sizes.
  - Disadvantages:
    * Slower and sometimes laggy.
    * Takes more effort to set up.
    * Harder to test on different types of devices unless you have several phones.

5. Troubleshooting a Common Error (5 Points)

- Identify a common error you encountered when starting your React Native app. Note that it is very unlikely that everyone will get the same error here.

  I faced dependencies error where my jdk wasn't compatible with the running react-native version. I had to run npx react-native doctor to find the loose dependencies.

- Explain the cause of the error and the steps you took to resolve it.

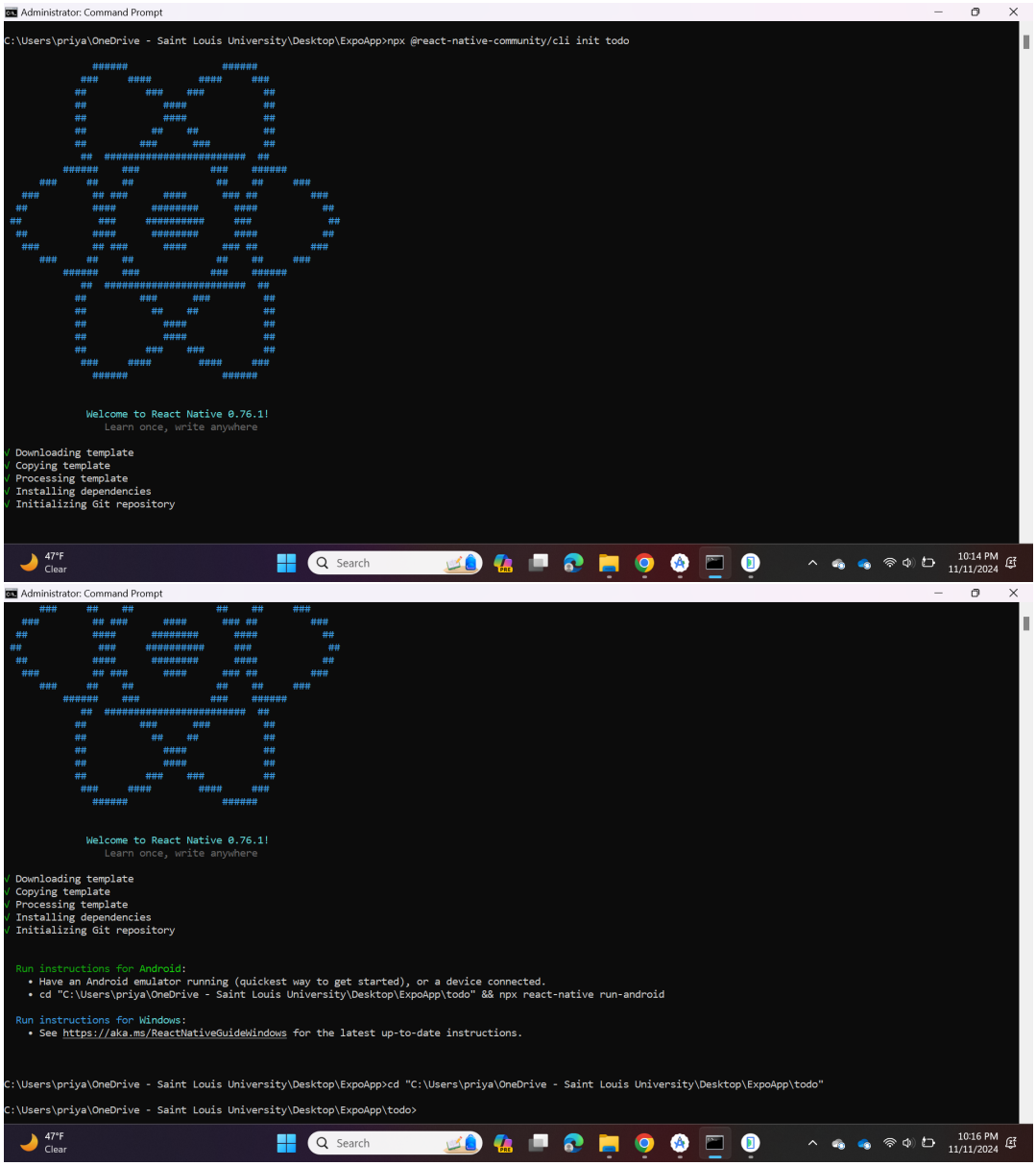
  This caused due to incompatible jdk version. To solve this I download jdk-17 restarted the computer and ran the run-android command again to start the app.

Task 2: Building a Simple To-Do List App

Step 1: Set Up the Project

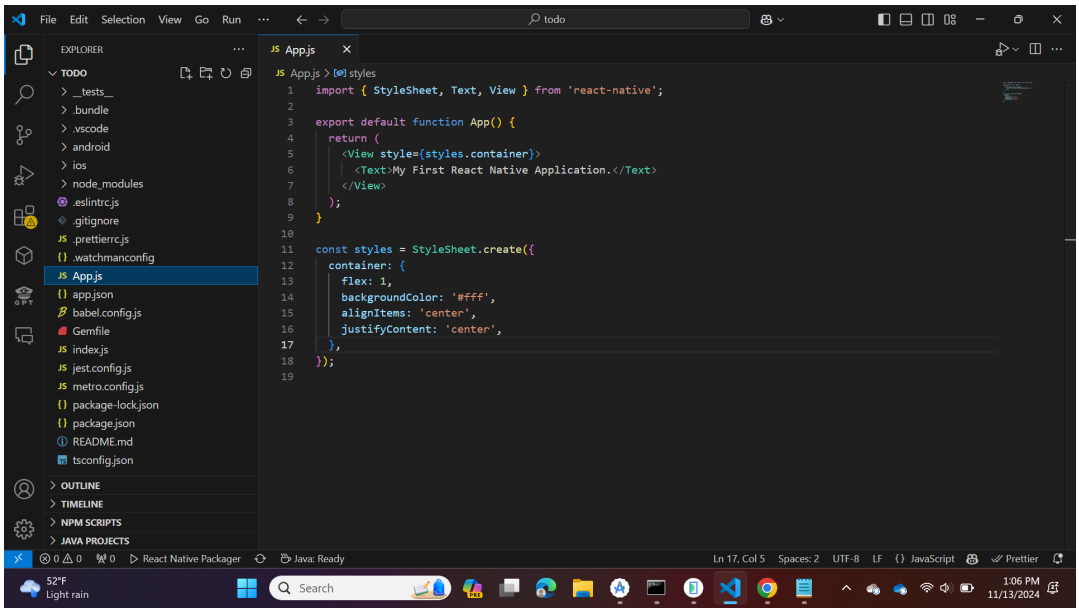
- Create a new project:

```
npx react-native init SimpleTodoApp
cd SimpleTodoApp
```



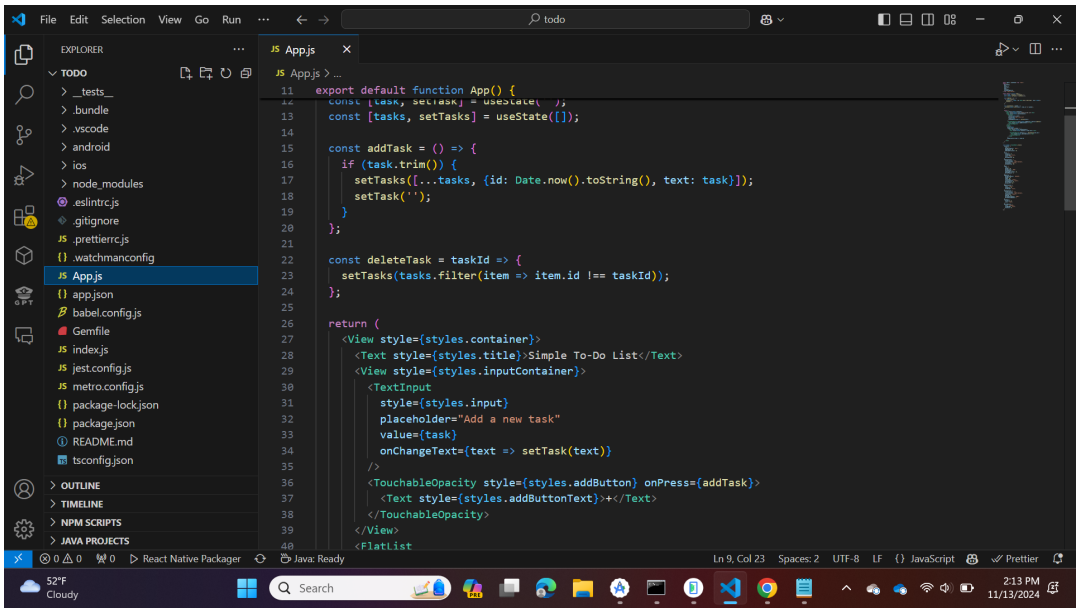
- Open the project in VS Code:

  ```
  code .
  ```



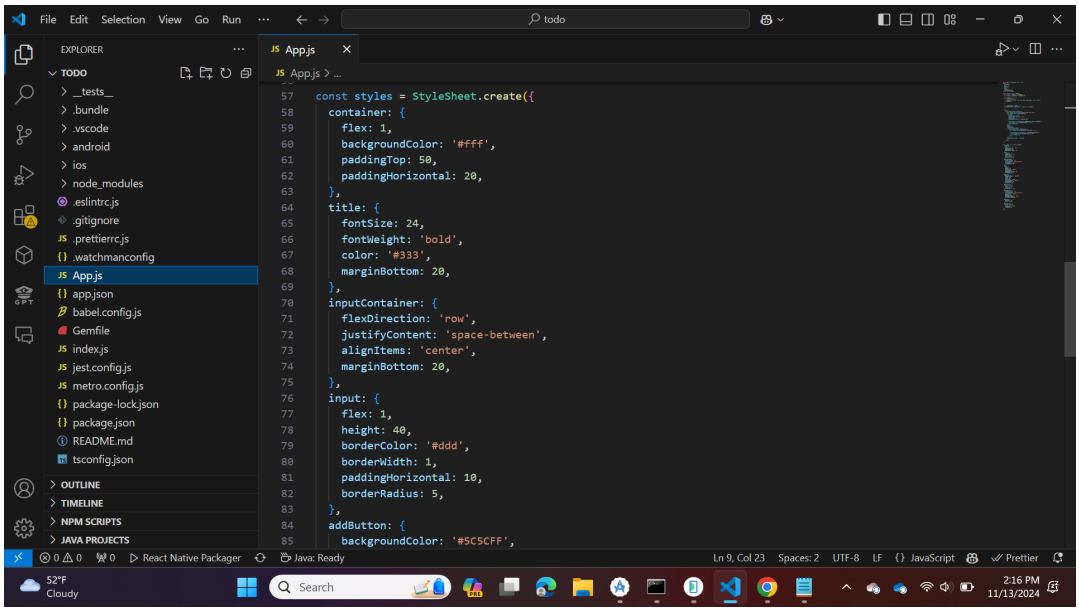
Step 2: Create the Basic To-Do List Structure

Replace the content of `App.js` with a basic structure for the To-Do list.



Step 3: Add Styles for the To-Do List

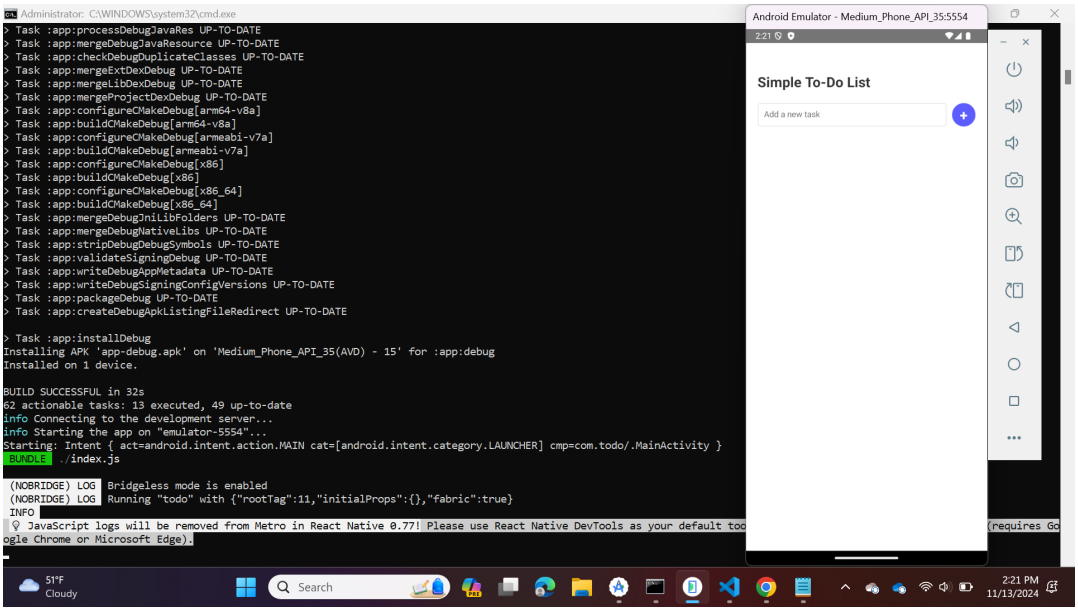
Add personalized styles to the bottom of `App.js` to improve the UI.



Step 4: Running the App

- Run the app with:

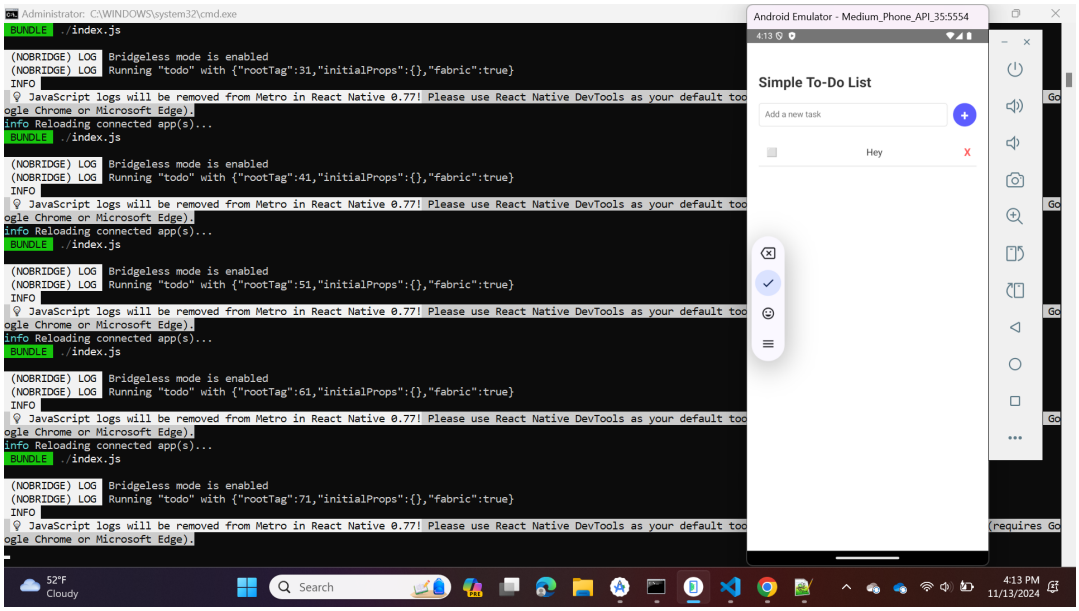
```
npx react-native run-android
```



- This compiles and launches the app on your selected platform.

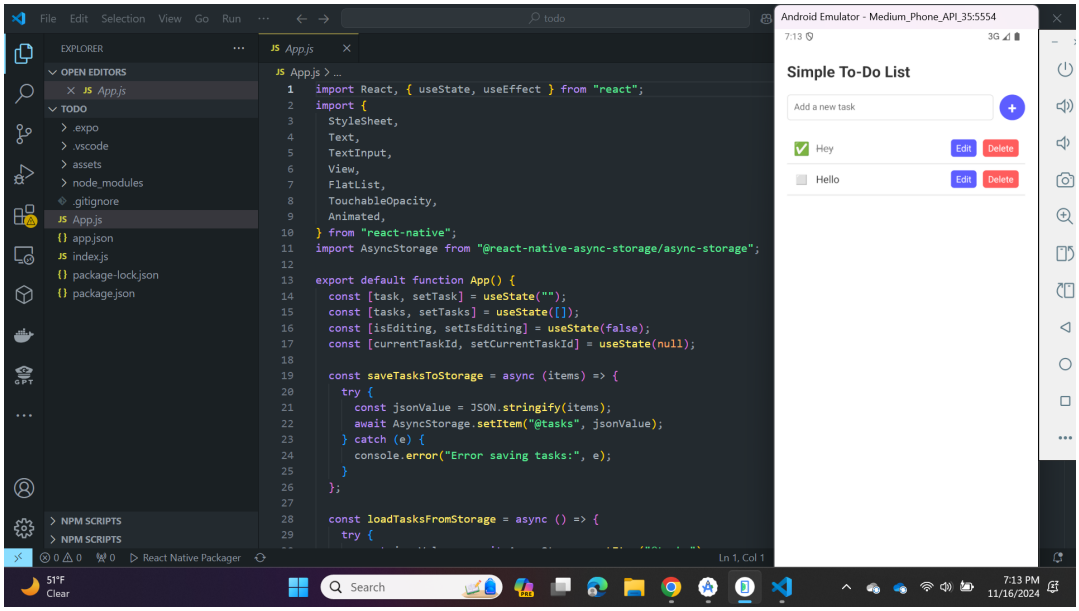
Features to Implement

1. **Mark Tasks as Complete (15 Points):**
   - Add a toggle function to mark tasks as completed.
   - Style completed tasks with strikethrough text or color changes.



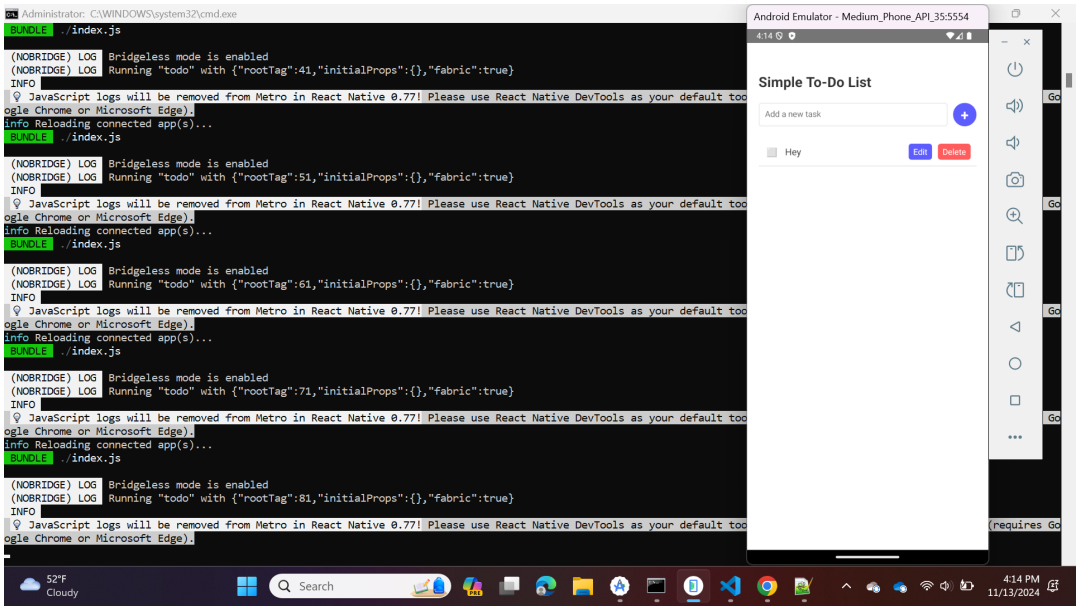
1. **Persist Data Using AsyncStorage (15 Points):**

   - Use AsyncStorage to save and retrieve tasks, ensuring persistence after app closure.

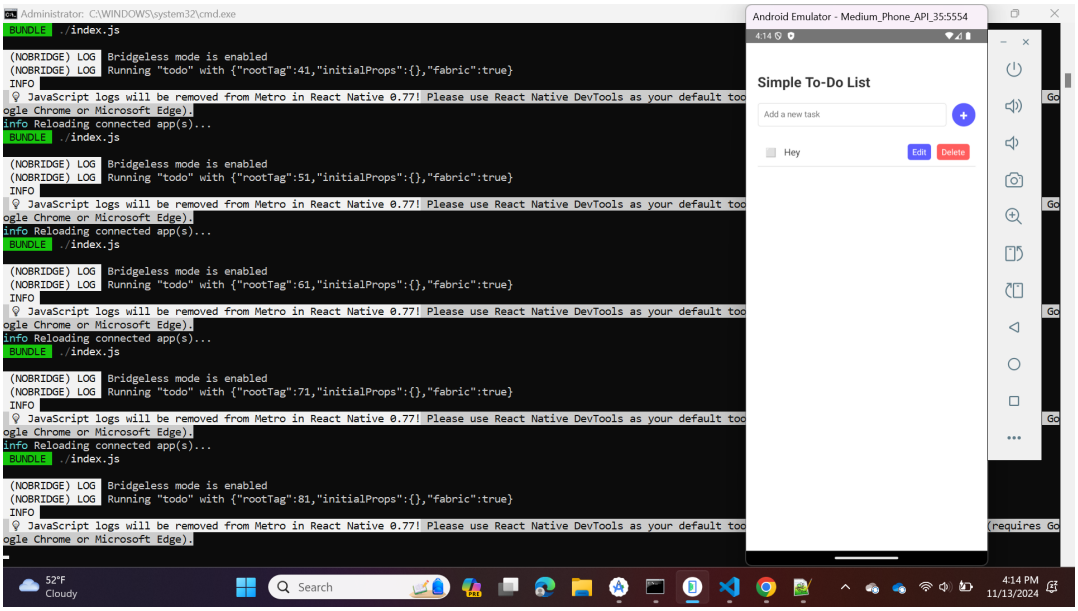


1. **Edit Tasks (10 Points):**

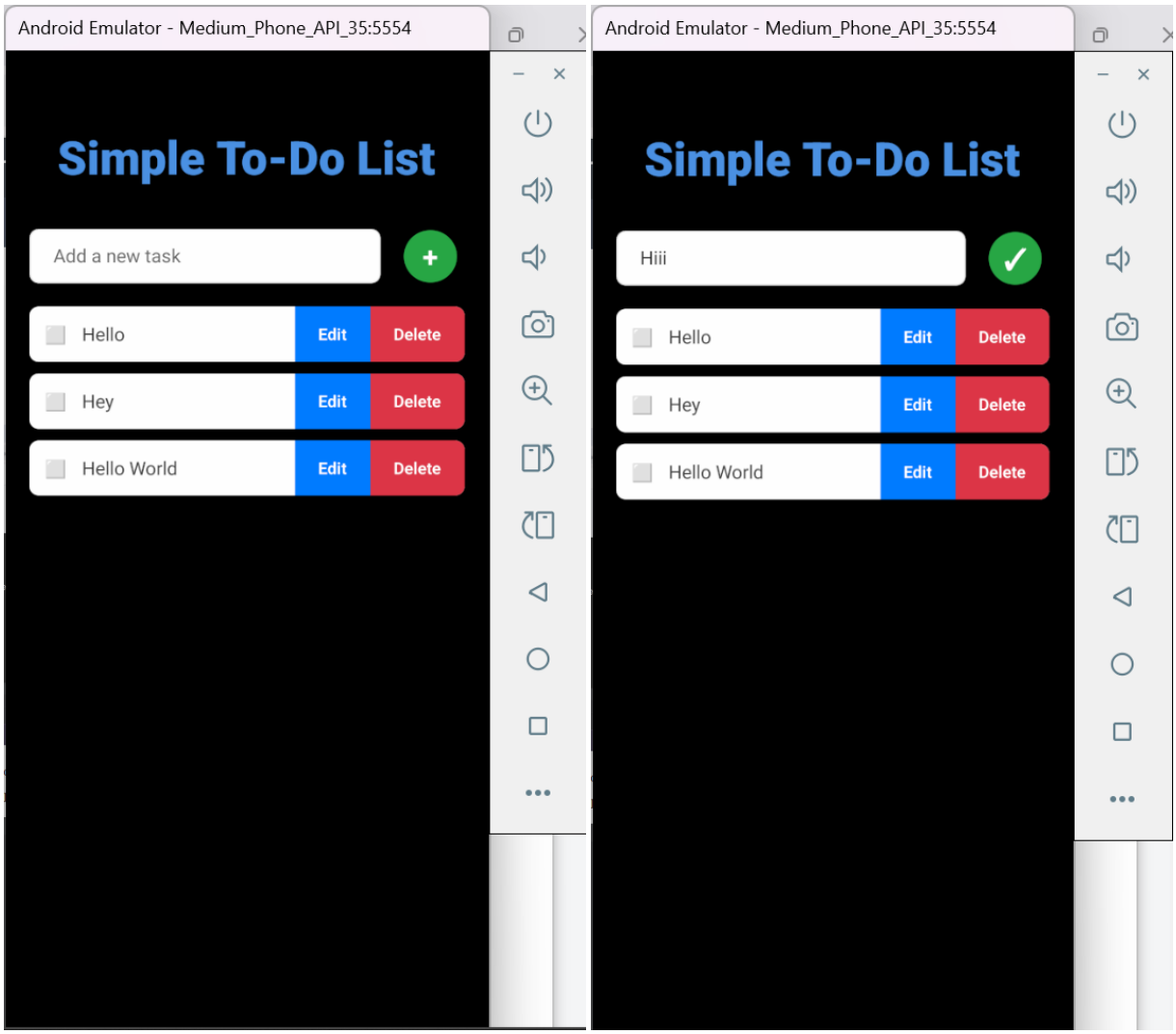
- Allow users to tap on tasks to edit their content.
- Implement a function to update tasks in the state array.

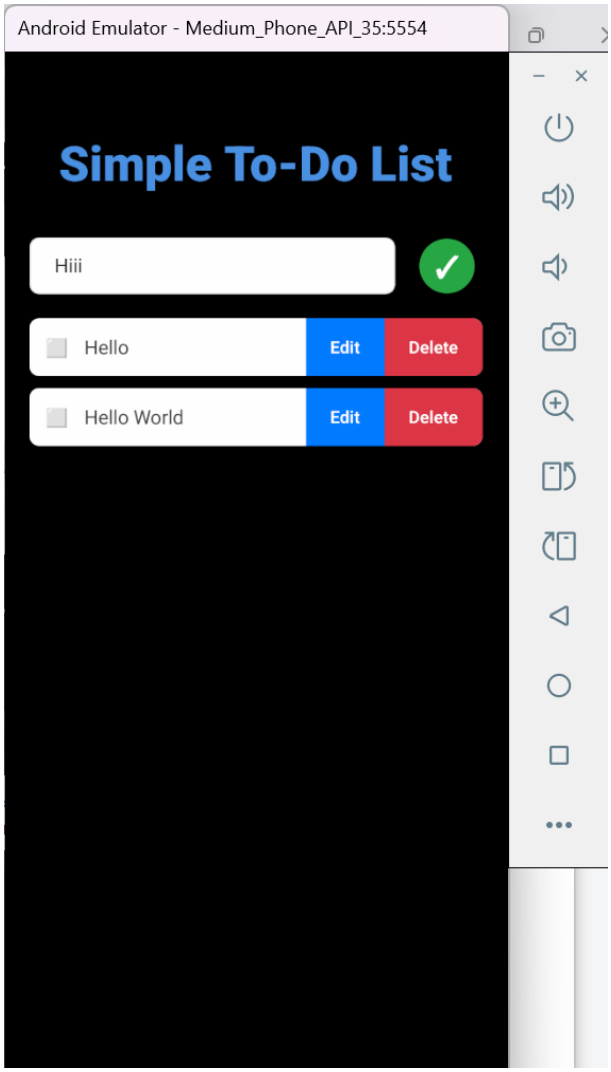


1. **Add Animations (10 Points):**
   - Use the Animated API for visual effects when adding or deleting tasks.
   - Describe how animations enhance the user experience.



- Changed UI Styling:


Android Emulator - Medium_Phone_API_35:5554
Simple To-Do List
Add a new task
Hello
Edit
Delete
Hey
Edit
Delete
Hello World
Edit
Delete
Android Emulator - Medium_Phone_API_35:5554
Simple To-Do List
Hiii
Hello
Edit
Delete
Hey
Edit
Delete
Hello World
Edit
Delete


Android Emulator - Medium_Phone_API_35:5554
Simple To-Do List
Hiii
Hello
Edit
Delete
Hello World
Edit
Delete