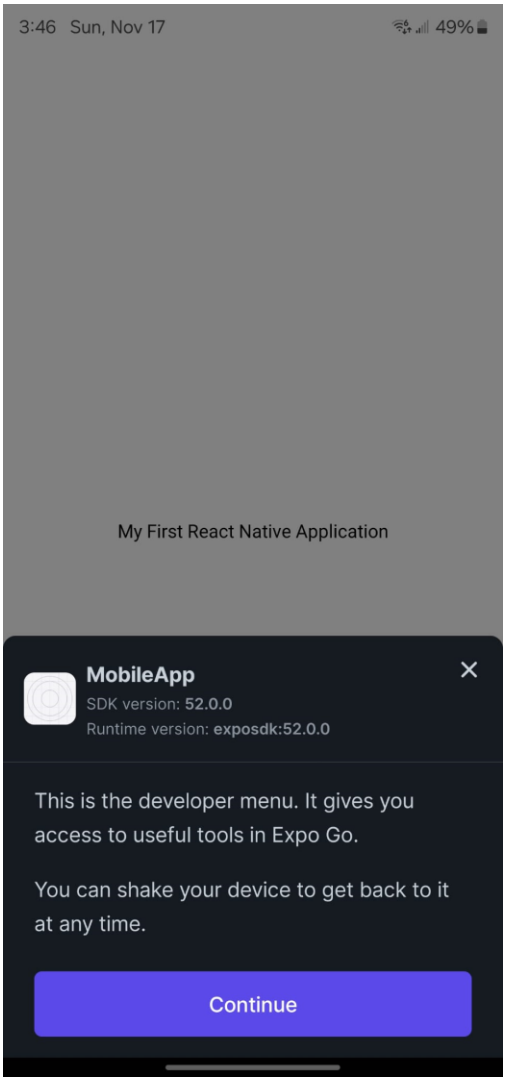
Lab 3

Sivalakshmi Chaitanya Koppuravuri Venkata 001286953

Github link: https://github.com/SivalakshmiChaitanya/Lab_3-Task_1

Task 1 (40 points):

(1)



After making changes in the App.js file the changes are visible in Physical device but not in emulator that’s the difference I noticed. And the address is saved in the physical device so that we can run it form anywhere.

(2)

Setting up Android Studio:

- First, Installed Android Studio in the local machine.
- Next downloaded the required SDK tools for the application.
- Later Updated the environment variables by placing the path the system variables.
- Setting the Metro bundle and later run the application.

Problems I faced while setting up the Android studio is while building the app due some path issues. Later updated the Path in the environment variables and rebuilding the app solved the issue.

(3)

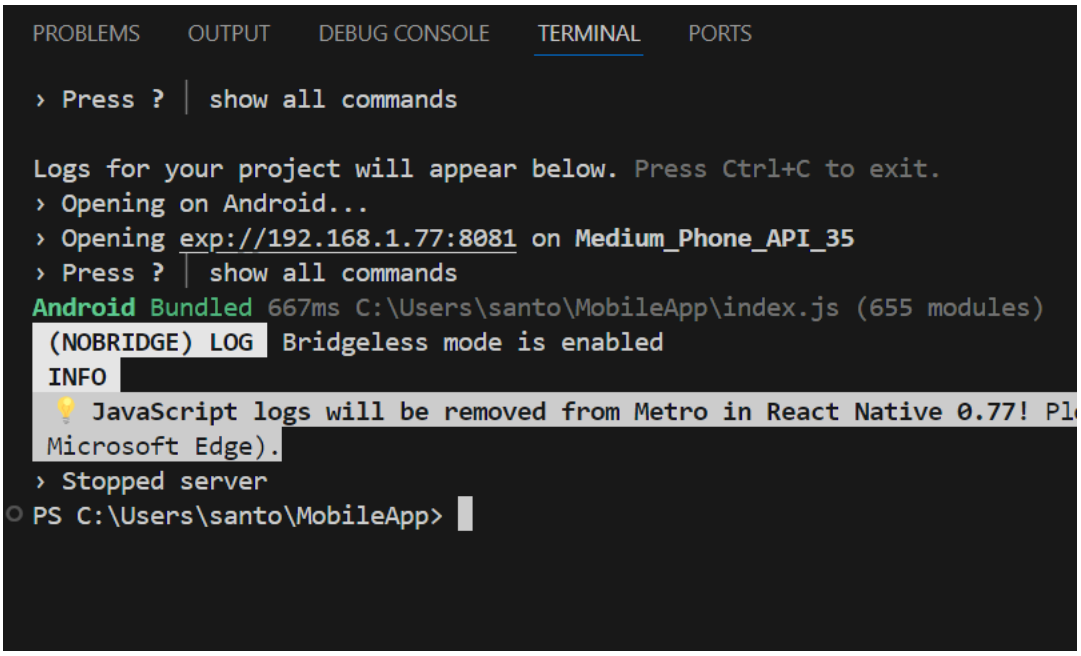
First, we setup the Expo which we use to connect our react app to physical device.

Use this command to install expo client - **npm install -g expo-cli**

Then we create a new expo application - **expo init app-name**

Then we navigate to the app folder and run the application by using **npm start** command

After running the app we get a link like this :



Later we install Expo go app in Android or iOS and insert the link then we can see the application running in the physical device.

(4)

Performance:

Emulators can be slower due to the overhead of simulating a mobile environment on a desktop. Performance may vary depending on your computer's hardware. Physical devices usually provide smoother performance since they reflect real-world usage with actual hardware, offering a true representation of how your app will behave.

Hardware Features:

In Emulators, limited access to certain hardware features like camera quality, GPS accuracy, fingerprint sensors, NFC, and other device-specific components. Some features may be simulated but are not always accurate. But in Physical device, it has complete access to all hardware features, making it ideal for testing functionalities like biometric authentication, camera, NFC, and other sensors. This allows for comprehensive testing of hardware-dependent features.

Startup Time:

Emulators are faster to launch. Startup speed in physical device depends in the data transfer speed.

Advantages using Emulator:

- Easily accessible within your development environment. No need to connect an external device, making it quick for initial testing.
- You can simulate a wide range of device types, screen sizes, and OS versions, which helps test different scenarios and layouts.
- Avoids using the battery and resources of a physical device, which can be handy for long development sessions.

Disadvantages using Emulator:

- Emulators can be slower, especially on machines without hardware acceleration (e.g., Intel HAXM for Android). This may affect the responsiveness of apps, particularly with animations.
- Emulators have limitations in simulating hardware features like camera quality, sensors
- Running an emulator can be resource-heavy, consuming a lot of CPU and RAM, which can slow down your development machine.

Advantages using Physical Device:

- Provides an accurate representation of how the app will behave on a real device, including performance, responsiveness, and user experience.
- Generally faster and more responsive than emulators, especially for graphically intensive or animation-heavy apps.
- Allows testing in various real-world network scenarios (e.g., switching from Wi-Fi to cellular, simulating poor network conditions).

Disadvantages using Physical Device:

- Testing on a single physical device does not account for the wide variety of screen sizes, resolutions, and hardware configurations in the market.
- Requires setting up USB debugging, managing device permissions, and dealing with potential connection issues
- Prolonged use for testing may drain the battery and wear out the physical device faster, especially if testing intensive features or running continuous builds.

(5)

I got some issues while setting up the environment because the required dependencies are not installed properly. Later I reinstalled the dependencies and updated the PATH in system variables and the environment was created successfully.