Lab 3 Report

JEET PATEL

18 NOV, 2024

GitHub Repository

https://github.com/mrpateljeet/LAB3_ASS_WEB

Task 1 (Total 40 Points)

- A) Screenshots of Your App (5 Points)
- B) Differences Between Running the App on an Emulator versus a Physical Device

When running the app on an emulator and a physical device, you may notice the following differences:

- **Performance:** The physical device often runs the app more smoothly, with faster load times and fewer lags, especially with animations. Emulators may be slower, particularly on machines with limited resources.
- User Interaction: The physical device allows real touch gestures, like swipes and pinches, whereas the emulator mimics touch with mouse clicks, which isn't as precise.
- Hardware-Specific Features: Certain features, like the camera or sensors, may not work as expected on an emulator but function correctly on a physical device.
- Screen Resolution and Display: Physical devices show the actual display quality, including color and brightness. Emulators approximate this, but physical devices may reveal UI issues related to screen size and aspect ratio more accurately.
- 2) Setting Up an Emulator (10 Points)
- A) Steps to Set Up an Emulator in Android Studio or Xcode
 - 1. **Install Xcode:** Download Xcode from the Mac App Store and open it once installed, following any initial setup instructions.

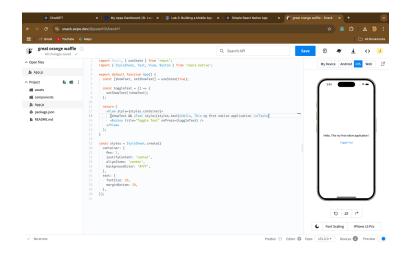


Figure 1: Hello Page First Native App

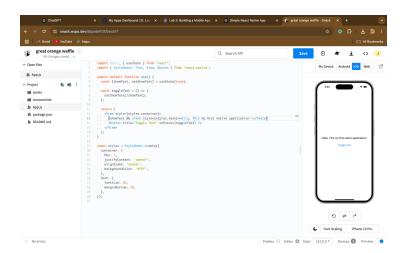


Figure 2: Hello Page First Native App

```
PATATIONAL PROPERTY OF THE PATATION OF THE PAT
```

Figure 3: Installation Part(Homebrew)

- 2. **Set Command Line Tools:** Go to Xcode, Preferences, Locations tab and select the version of Xcode you installed under Command Line Tools.
- 3. **Open the Simulator:** Open the Simulator from Xcode by navigating to Xcode, Open Developer Tool, Simulator.
- 4. **Set Up a New iOS Simulated Device:** Go to Window, Devices, and Simulators, click the Simulators tab, press the + button to add a new simulator, and click Create.
- 5. **Start the Simulator:** Select the newly created simulator in the Devices and Simulators window and click Start to launch it.

B) Challenges Faced and Solutions

- SDK Version Compatibility: Selecting a stable or recommended SDK version resolved compatibility issues.
- Slow Performance: Increasing RAM allocation for the emulator in AVD settings improved performance.
- **Device-Specific Issues:** Switching to another device model or Android version resolved unique bugs or crashes.

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

A) Connecting the Physical Device

Installed the Expo CLI using the command npm install -g expo-cli. Expo Developer Tools opened in the browser, displaying a QR code. Installed the

2:56 **◆** Code Scanner



Hello, This my first native application!

Toggle Text

4

Figure 4: connection successfully

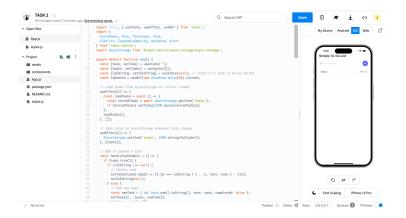


Figure 5: User can add task by clicking "+" Button

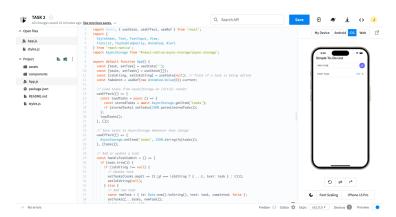


Figure 6: User can Edit task by clicking on "Edit"

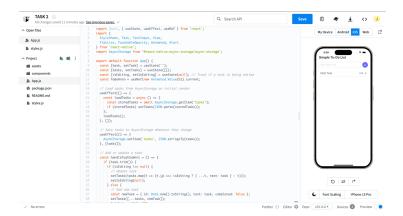


Figure 7: User can successfully edit/Update task at same time

```
And the service and the service service and the service price of the control of the service service and the service and the service service and the service service and the service and the service se
```

Figure 8: By generating this QR code connect Phone via Expo Go

Expo Go App from the Apple Store, scanned the QR code with the device, and loaded the app.

B) Troubleshooting Steps

Advantages and Disadvantages

- Advantages of Emulator: Quick testing on various versions and screen sizes; simulated environment without physical devices.
- Advantages of Physical Device: Accurate representation of real-world usage; access to hardware-dependent features.

Aspect	Emulator	Physical Device
Performance	Slower on lower-end ma-	Faster, showing real-world
	chines	app performance
Touch Accu-	Simulated with mouse clicks	Natural touch gestures
racy		
Hardware Ac-	Limited access (e.g., cam-	Full access to all hardware
cess	era, GPS)	features
Screen Quality	Approximate colors and res-	True screen quality and
	olution	color display
Testing Scope	Easy to test on multiple de-	Limited to a specific device
	vices	

Table 1: Comparison of Emulator vs. Physical Device

```
Autorizacia Photogramiani

The Califordia nee directorias will be created:

April 2014 Photogramiani

The Califordia nee directorias will be created:

April 2014 Photogramiani

April 2014 Photogramian
```

Figure 9: Installation

- Disadvantages of Emulator: Slower performance, limited hardware access, inaccurate display.
- Disadvantages of Physical Device: Limited to single device testing, network dependency.

5) Troubleshooting a Common Error (5 Points)

A common error encountered: "Unable to load script" due to Metro Bundler connection issues. Resolved by ensuring the device and computer are on the same network and clearing the cache.

Task 2: Building a Simple To-Do List App (60 Points)

Explanation of Features Implemented

- Add New Tasks: The handleTaskSubmit function adds new tasks if the input is not empty.
- Update Existing Tasks: Users can edit tasks, changing the input field and button state.
- Delete Tasks: Tasks can be deleted with a confirmation alert.

- Scrollable Task List: Implemented with FlatList for scrollability.
- User-Friendly Interface: Clear buttons, and animations for task management.

Explanation of Code

- State Management: useState and useEffect manage state and persist tasks.
- Adding a Task: addTask function creates new tasks with unique IDs.
- Deleting a Task: deleteTask function removes tasks from the state.
- Toggling Completion: toggleTaskCompletion applies strikethrough styles to completed tasks.
- Rendering the List: Used FlatList to display tasks with deletion and completion toggling capabilities.