React Native no2Styler - Setup Commands (Updated)

1. Create Project (Updated Command)

```bash
npx @react-native-community/cli@latest init no2Styler
cd no2Styler
```

Alternative Methods:

```bash
# Option A: Install CLI globally first
npm install -g @react-native-community/cli
npx @react-native-community/cli init no2Styler

# Option B: Use create-react-native-app
npx create-react-native-app no2Styler

# Option C: Use Expo (if preferred)
npx create-expo-app no2Styler --template blank-typescript
```

2. Install iOS Dependencies

```bash
cd ios
pod install
cd ..
```

3. Running on Single Platform

Android Only (Port 9081)

```bash
npx react-native start --port 9081
```

New Terminal:

```bash
npx react-native run-android --port 9081
```

iOS Only (Port 9082)

```bash
npx react-native start --port 9082
```

New Terminal:

```bash
npx react-native run-ios --port 9082
```

4. Running Both Platforms Simultaneously

Terminal 1 - Android Metro Server

```bash
npx react-native start --port 9081
```

Terminal 2 - Run Android App

```bash
npx react-native run-android --port 9081
```

Terminal 3 - iOS Metro Server

```bash
npx react-native start --port 9082
```

Terminal 4 - Run iOS App

```bash
npx react-native run-ios --port 9082
```

5. Alternative Method (Shared Metro)

```bash
# Terminal 1 - Start Metro on default port
npx react-native start

# Terminal 2 - Run Android
npx react-native run-android

# Terminal 3 - Run iOS
npx react-native run-ios
```

6. Specify iOS Simulator

```bash
# List available simulators
xcrun simctl list devices

# Run on specific simulator
npx react-native run-ios --simulator="iPhone 15" --port 9082
```

7. Troubleshooting Commands

Clear Cache and Restart

```bash
npx react-native start --reset-cache --port 9081
npx react-native start --reset-cache --port 9082
```

Kill All Metro Processes

```bash
pkill -f "metro"
```

Check Port Usage

```bash
lsof -i :9081
lsof -i :9082
```

Clean iOS Build

```bash
cd ios
rm -rf build/
pod deintegrate
pod install
cd ..
npx react-native run-ios --reset-cache --port 9082
```

Clean Android Build

```bash
cd android
./gradlew clean
cd ..
npx react-native run-android --reset-cache --port 9081
```

Force Kill Specific Port

```bash
# Kill process on port 9081
lsof -ti:9081 | xargs kill -9

# Kill process on port 9082
lsof -ti:9082 | xargs kill -9
```

8. Quick Start Sequence

Option A: Sequential Start

```bash
bash

# Step 1 — Create Project
npx @react-native-community/cli@latest init no2Styler
cd no2Styler

# Step 2 — Install iOS Dependencies
cd ios && pod install && cd ..

# Step 3 — Start Android (Terminal 1)
npx react-native start --port 9081

# Step 4 — Run Android (Terminal 2)
npx react-native run-android --port 9081

# Step 5 — Start iOS Metro (Terminal 3)
npx react-native start --port 9082

# Step 6 — Run iOS (Terminal 4)
npx react-native run-ios --port 9082
```

Option B: One-Line Setup

```bash
bash

# Create and setup (run this first)
npx @react-native-community/cli@latest init no2Styler && cd no2Styler && cd ios && pod
```

Then run in 4 separate terminals:

```bash
bash

# Terminal 1:
npx react-native start --port 9081

# Terminal 2:
npx react-native run-android --port 9081

# Terminal 3:
npx react-native start --port 9082

# Terminal 4:
npx react-native run-ios --port 9082
```

9. Port Configuration Summary

Platform	Metro Port	Command
Android	9081	`npx react-native run-android --port 9081`
iOS	9082	`npx react-native run-ios --port 9082`

10. Prerequisites Check

Before Starting

```bash
# Check Node.js version
node --version

# Check npm version
npm --version

# Check if Android emulator is running
adb devices

# Check available iOS simulators
xcrun simctl list devices

# Check if CocoaPods is installed
pod --version
```

Install Prerequisites (if needed)

```bash
# Install CocoaPods (if not installed)
brew install cocoapods

# Start Android emulator (replace with your AVD name)
emulator -avd Pixel_4_API_30

# Open iOS Simulator
open -a Simulator
```

11. Verification Commands

Check App is Running

```bash
# Check Android emulator
adb devices

# Check iOS simulator
xcrun simctl list devices | grep Booted
```

Check Metro Bundler Status

```bash
# Check Android Metro
curl http://localhost:9081/status

# Check iOS Metro
curl http://localhost:9082/status
```

Check if ports are free before starting

```bash
# Check if ports are available
lsof -i :9081
lsof -i :9082

# If ports are busy, kill processes
lsof -ti:9081 | xargs kill -9
lsof -ti:9082 | xargs kill -9
```

12. Common Error Solutions

If CLI installation fails

```bash
# Clear npm cache
npm cache clean --force

# Install with specific version
npx @react-native-community/cli@12.3.0 init no2Styler
```

If pod install fails

```bash
cd ios
pod repo update
pod install --repo-update
cd ..
```

If Android build fails

```bash
# Clean everything
cd android
./gradlew clean
rm -rf build/
rm -rf app/build/
cd ..
```

If Metro bundler won't start

```bash
# Reset everything
npx react-native start --reset-cache
rm -rf node_modules
rm package-lock.json
npm install
```

13. Final Checklist

✅ Before running commands:

☐ Node.js installed (v16+)
☐ Android Studio installed with emulator
☐ Xcode installed with simulator
☐ CocoaPods installed

✅ Project setup:

☐ Project created successfully
☐ iOS pods installed
☐ Both emulators/simulators running

✅ Running apps:

☐ Android Metro running on port 9081
☐ iOS Metro running on port 9082

- ☐ Android app launched successfully
- ☐ iOS app launched successfully