RNTester

The RNTester showcases React Native views and modules.

Running this app

Before running the app, make sure you ran:

```
git clone https://github.com/facebook/react-native.git
cd react-native
yarn install
```

Running on iOS

If you are testing non-fabric component, modify [the fabric_enabled flag in RNTester's Podfile](#).

```
fabric_enabled = false
```

Also, if you previously built RNTester with fabric enabled, you might need to clean up the build files and Pods.

```
# Clean the generated files and folders to clean install RNTester
cd packages/rn-tester
yarn clean-ios
```

If you are still having a problem after doing the clean up (which can happen if you have built RNTester with older React Native versions where files were generated inside the react-native folder.), the best way might be to clean-install react-native (e.g. remove node_modules and yarn install).

Both macOS and Xcode are required.

1. `cd packages/rn-tester`
2. Install [Bundler](#): `gem install bundler`. We use bundler to install the right version of [CocoaPods](#) locally.
3. Install Bundler and CocoaPods dependencies: `bundle install && bundle exec pod install` or `yarn setup-ios-jsc`. In order to use Hermes engine instead of JSC, run: `USE_HERMES=1 bundle exec pod install` or `yarn setup-ios-hermes` instead.
4. Open the generated `RNTesterPods.xcworkspace`. This is not checked in, as it is generated by CocoaPods. Do not open `RNTesterPods.xcodeproj` directly.

Note for M1 users

If you own a Mac M1 laptop, you need to run some different commands to install and run cocoapods.

- `sudo arch -x86_64 gem install ffi`: this installs the `ffi` package to load dynamically-linked libraries.
- `arch -x86_64 pod install`: this run `pod install` with the right architecture.

Running on Android

You'll need to have all the [prerequisites](#) (SDK, NDK) for Building React Native installed.

Start an Android emulator.

```
cd packages/rn-tester
# In order to use Hermes engine, run `yarn install-android-hermes` instead.
yarn install-android-jsc
yarn start
```

Note: Building for the first time can take a while.

Open the RNTester app in your emulator. If you want to use a physical device, run `adb devices`, then `adb -s <device name> reverse tcp:8081 tcp:8081`. See [Running on Device](#) for additional instructions on using a physical device.

Running with Buck

Follow the same setup as running with gradle.

Install Buck from [here](#).

Run the following commands from the react-native folder:

```
./gradlew :ReactAndroid:packageReactNdkLibsForBuck
buck fetch rntester
buck install -r rntester
./scripts/packager.sh
```

Note: The native libs are still built using gradle. Full build with buck is coming soon(tm).

Building from source

Building the app on both iOS and Android means building the React Native framework from source. This way you're running the latest native and JS code the way you see it in your clone of the github repo.

This is different from apps created using `react-native init` which have a dependency on a specific version of React Native JS and native code, declared in a `package.json` file (and `build.gradle` for Android apps).