

Configure React Native Project under proxy

1. Configure Proxy variable. It will be used for all other configuration where we'll be using proxy.

```
export http_proxy=http://macproxy.cognizant.com:6050
```

2. Install CocoaPods (Pods are used to download ios frameworks and libraries)

```
sudo gem install --http-proxy $http_proxy cocoapods
```

3. Setup proxy for git and npm

```
git config --global http.proxy $http_proxy  
git config --global https.proxy $http_proxy
```

```
npm config set proxy $http_proxy  
npm config set https-proxy $http_proxy
```

4. Install brew (brew will be used to install other components, i.e. watchman, yarn, ect)

```
/usr/bin/ruby -e "$(curl -x http://macproxy.cognizant.com:6050 -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

5. Set proxy for brew

```
ALL_PROXY=http://macproxy.cognizant.com:6050 brew upgrade
```

6. Install Watchman

```
http_proxy=http://macproxy.cognizant.com:6050  
https_proxy=http://macproxy.cognizant.com:6050 brew install --HEAD Watchman
```

7. Install Yarn

```
http_proxy=http://macproxy.cognizant.com:6050  
https_proxy=http://macproxy.cognizant.com:6050 brew install yarn
```

8. Apply write access to Node_Modules global folder

```
sudo chown -R $(whoami) /usr/local/lib/node_modules/
```

9. Install React Native Cli and Yo

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

10. Configure git to use https instead of git http scheme

```
git config --global url.https://github.com/.insteadOf git://github.com/
```

11. Clone caribou repository to your preferred directory

```
git clone https://github.com/DiscoverFinancial/caribou.git
```

12. Move to directory where the repository is cloned and execute yarn to install required node modules

```
Yarn install
```

13. Move to iOS directory to install the pods. Make sure to clean Pods directory before executing Pod command.

```
Pod install
```

14. Open project workspace in xCode to verify if configuration is successful.
15. Execute the build from xCode, alternatively it can also be executed using following command.

```
react-native run-ios
```

16. On few machines, if after execution it shows error as bundle not available, if such error is encountered, use following command to generate the required bundle

```
react-native bundle --dev false --assets-dest ./ios --entry-file  
index.ios.js --platform ios --bundle-output ios/main.jsbundle
```

17. If above error is encountered and package is not running, add following code in AppDelegate.swift file under application didFinishLaunchingWithOptions.
Under #if DEBUG

```
if (jsCodeLocation == nil) {  
    jsCodeLocation = URL.init(string:  
"http://127.0.0.1:8081/index.bundle?platform=ios&dev=true")  
}
```