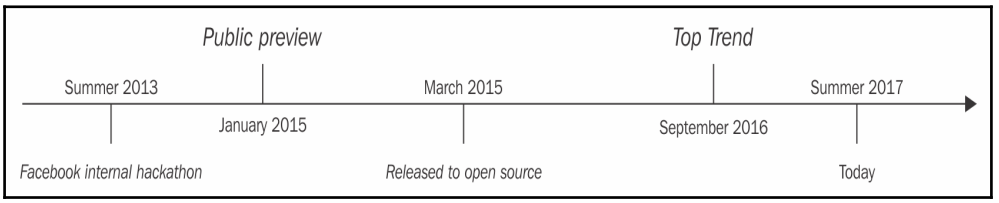
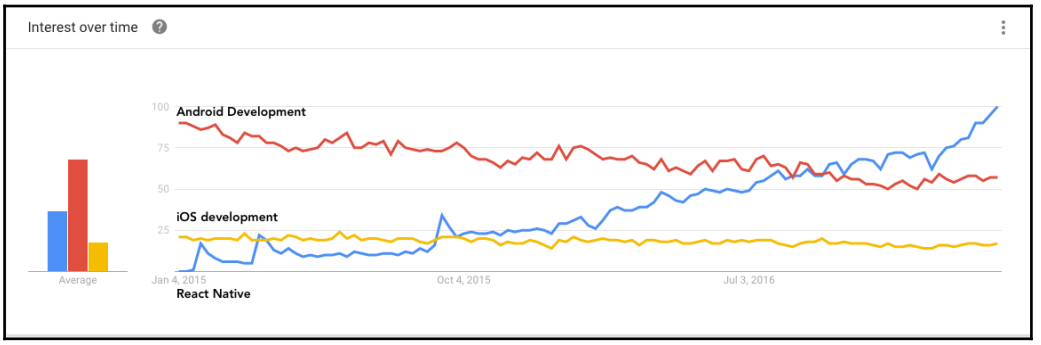
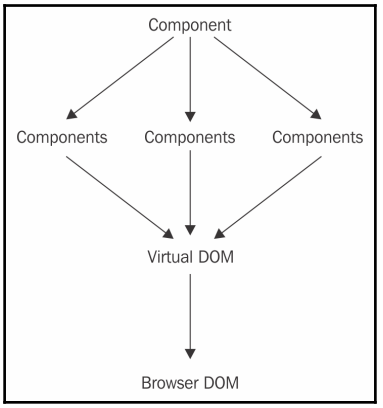
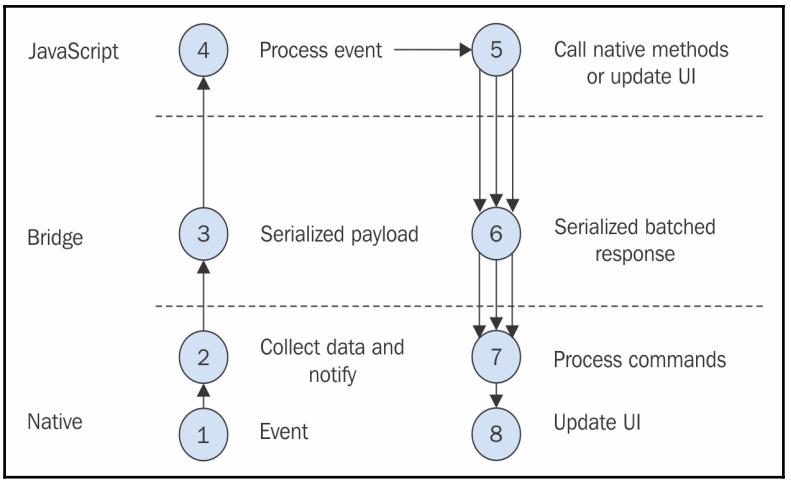
Chapter 1: Understanding Why React Native is the Future of Mobile Apps

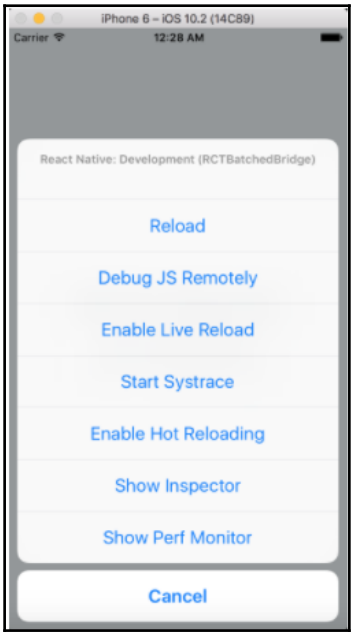


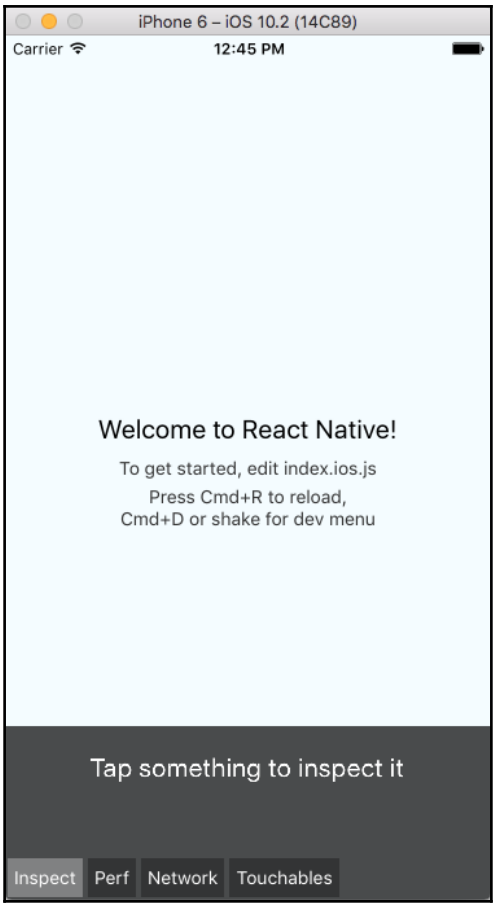


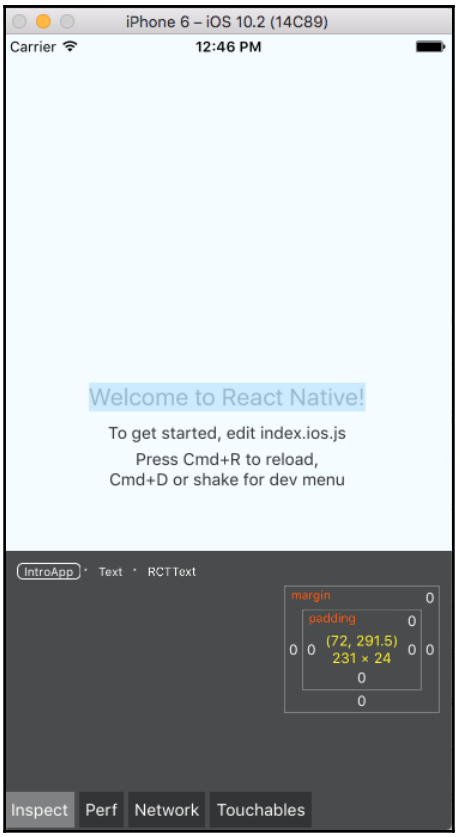
$$UI = f(state)$$



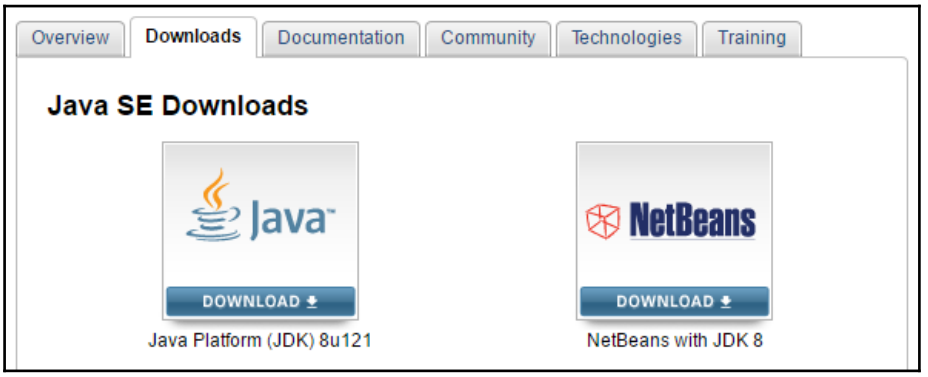

JavaScript
4
Process event
5
Call native methods or update UI
Bridge
3
Serialized payload
6
Serialized batched response
2
Collect data and notify
7
Process commands
Native
1
Event
8
Update UI


iPhone 6 – iOS 10.2 (14C89)
Carrier
12:28 AM
React Native: Development (RCTBatchedBridge)
Reload
Debug JS Remotely
Enable Live Reload
Start Systrace
Enable Hot Reloading
Show Inspector
Show Perf Monitor
Cancel


iPhone 6 – iOS 10.2 (14C89)
Carrier
12:45 PM
Welcome to React Native!
To get started, edit index.ios.js
Press Cmd+R to reload,
Cmd+D or shake for dev menu
Tap something to inspect it
Inspect
Perf
Network
Touchables


iPhone 6 – iOS 10.2 (14C89)
Carrier
12:46 PM
Welcome to React Native!
To get started, edit index.ios.js
Press Cmd+R to reload,
Cmd+D or shake for dev menu
IntroApp
Text
RCTText
margin
0
padding
0
0
0
(72, 291.5)
231 × 24
0
0
0
0
0
Inspect
Perf
Network
Touchables

Chapter 2: Working with React Native

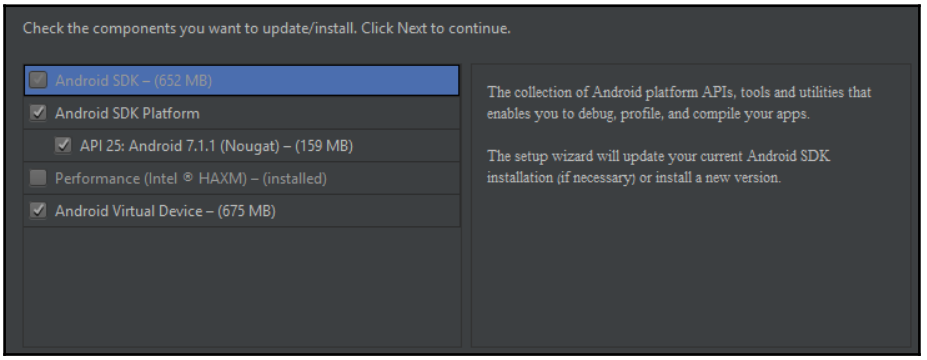


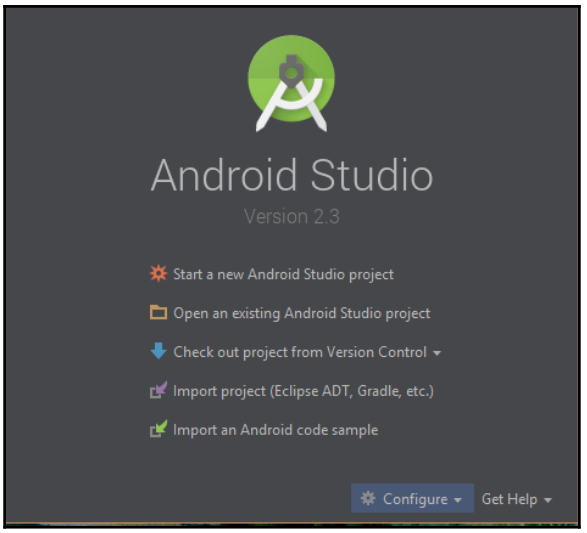
Java SE Development Kit 8u121

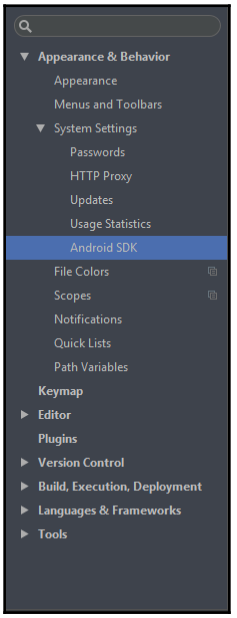
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

○ Accept License Agreement ◉ Decline License Agreement

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.86 MB	jdk-8u121-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.83 MB	jdk-8u121-linux-arm64-vfp-hflt.tar.gz
Linux x86	162.41 MB	jdk-8u121-linux-i586.rpm
Linux x86	177.13 MB	jdk-8u121-linux-i586.tar.gz
Linux x64	159.96 MB	jdk-8u121-linux-x64.rpm
Linux x64	174.76 MB	jdk-8u121-linux-x64.tar.gz
Mac OS X	223.21 MB	jdk-8u121-macosx-x64.dmg
Solaris SPARC 64-bit	139.64 MB	jdk-8u121-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.07 MB	jdk-8u121-solaris-sparcv9.tar.gz
Solaris x64	140.42 MB	jdk-8u121-solaris-x64.tar.Z
Solaris x64	96.9 MB	jdk-8u121-solaris-x64.tar.gz
Windows x86	189.36 MB	jdk-8u121-windows-i586.exe
Windows x64	195.51 MB	jdk-8u121-windows-x64.exe




Android Studio
Version 2.3
Start a new Android Studio project
Open an existing Android Studio project
Check out project from Version Control
Import project (Eclipse ADT, Gradle, etc.)
Import an Android code sample
Configure
Get Help

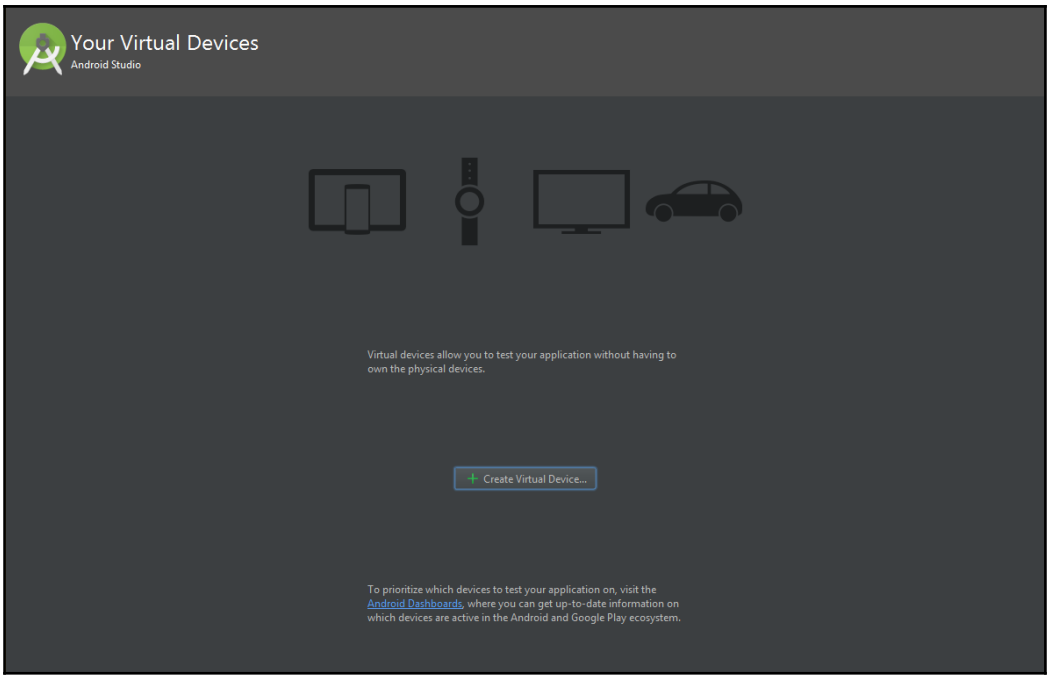

Appearance & Behavior
Appearance
Menus and Toolbars
System Settings
Passwords
HTTP Proxy
Updates
Usage Statistics
Android SDK
File Colors
Scopes
Notifications
Quick Lists
Path Variables
Keymap
Editor
Plugins
Version Control
Build, Execution, Deployment
Languages & Frameworks
Tools

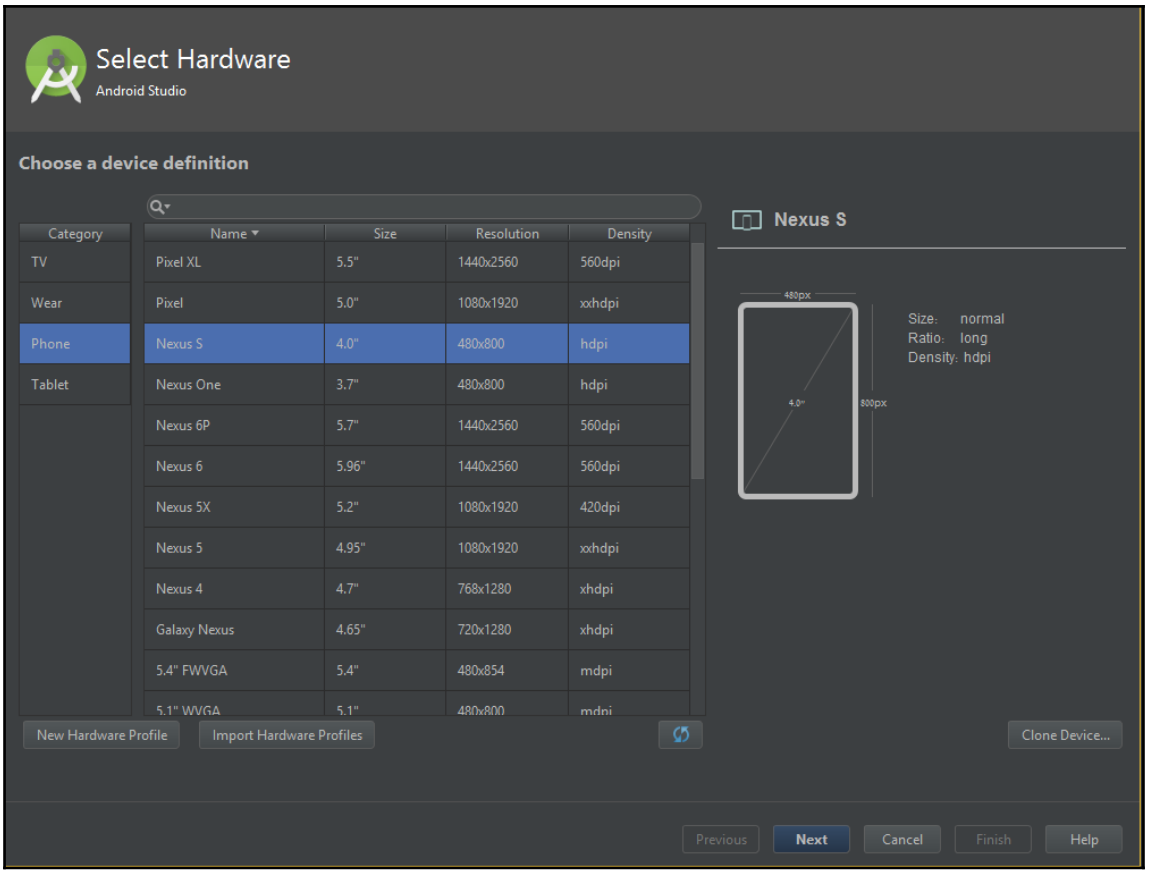
SDK Platforms | SDK Tools | SDK Update Sites

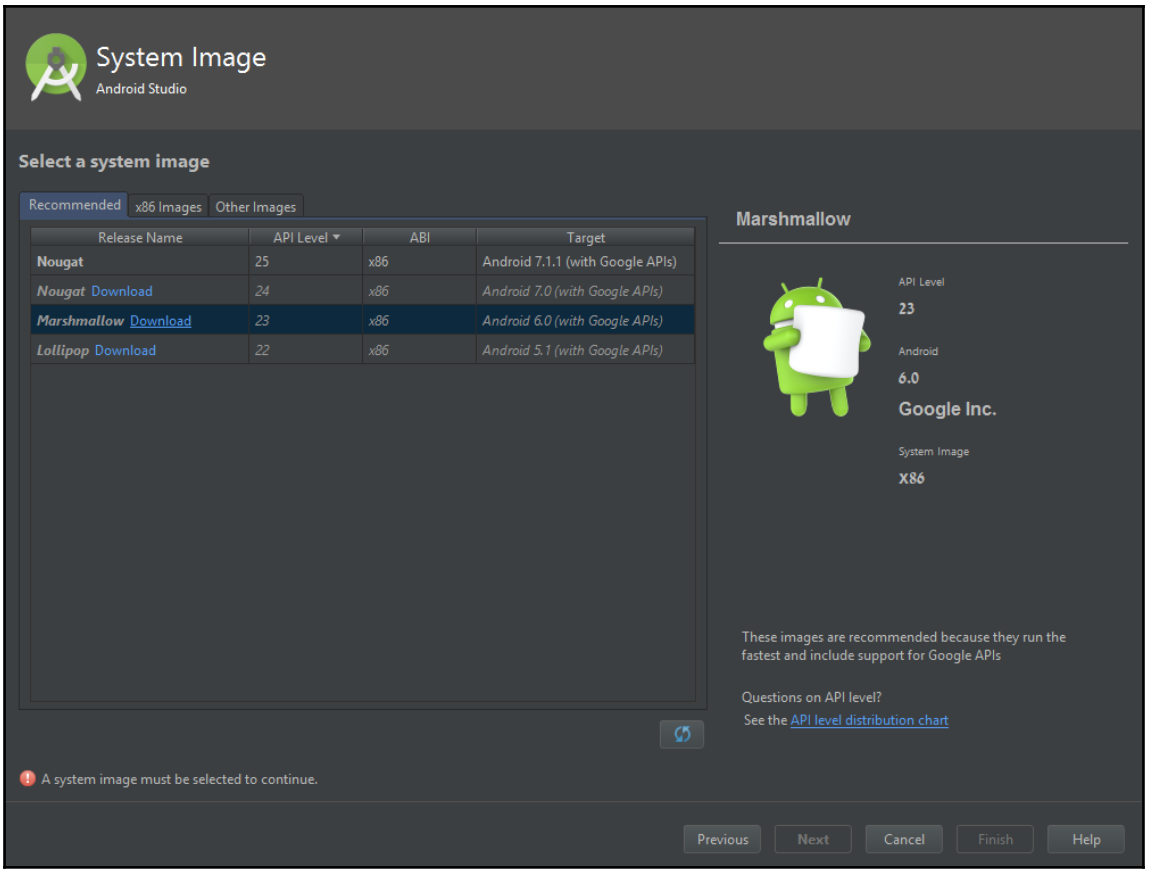
Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

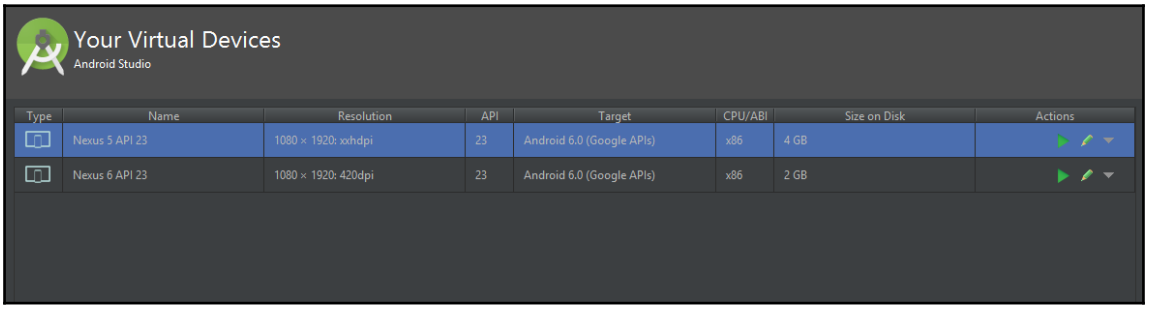
Name	API Level	Revision	Status
Android 7.1.1 (Nougat)			
Android 7.0 (Nougat)			
Android 6.0 (Marshmallow)			
Google APIs	23	1	Installed
Android SDK Platform 23	23	3	Installed
Sources for Android 23	23	1	Installed
Android TV ARM EABI v7a System Image	23	3	Not installed
Android TV Intel x86 Atom System Image	23	9	Not installed
Android Wear ARM EABI v7a System Image	23	6	Not installed
Android Wear Intel x86 Atom System Image	23	6	Not installed
ARM EABI v7a System Image	23	6	Not installed
Intel x86 Atom System Image	23	9	Not installed
Intel x86 Atom_64 System Image	23	9	Installed
Google APIs ARM EABI v7a System Image	23	20	Not installed
Google APIs Intel x86 Atom System Image	23	20	Not installed
Google APIs Intel x86 Atom_64 System Image	23	20	Installed
Android 5.1 (Lollipop)			
Google APIs	22	1	Not installed
Android SDK Platform 22	22	2	Not installed
Sources for Android 22	22	1	Not installed
Android TV ARM EABI v7a System Image	22	1	Not installed
Android TV Intel x86 Atom System Image	22	3	Not installed
ARM EABI v7a System Image	22	2	Not installed

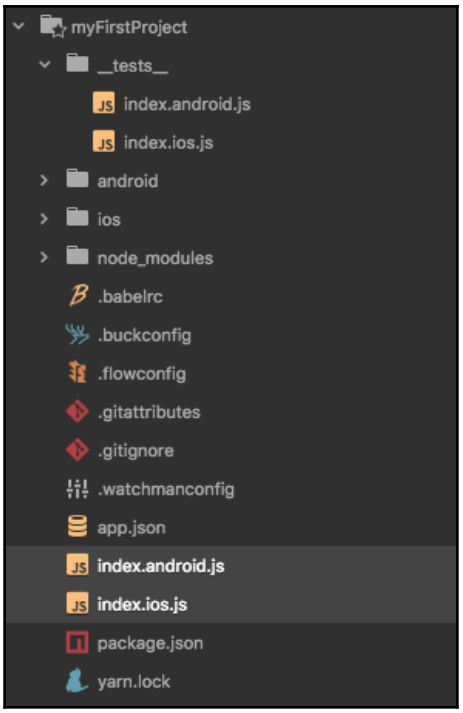
Show Package Details

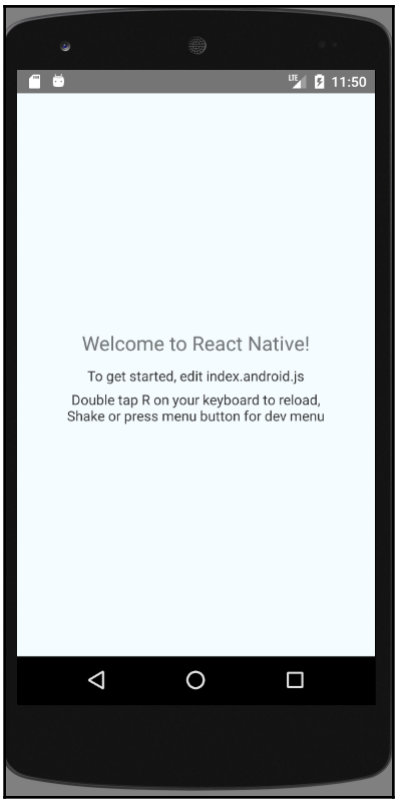

Your Virtual Devices
Android Studio
Virtual devices allow you to test your application without having to own the physical devices.
Create Virtual Device...
To prioritize which devices to test your application on, visit the Android Dashboards, where you can get up-to-date information on which devices are active in the Android and Google Play ecosystem.

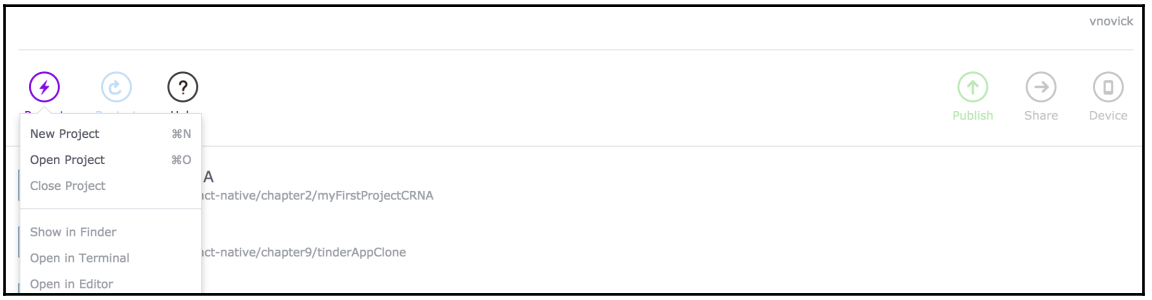

Select Hardware
Android Studio
Choose a device definition
Category
Name
Size
Resolution
Density
TV
Wear
Phone
Tablet
Pixel XL 5.5" 1440x2560 560dpi
Pixel 5.0" 1080x1920 xxhdpi
Nexus S 4.0" 480x800 hdpi
Nexus One 3.7" 480x800 hdpi
Nexus 6P 5.7" 1440x2560 560dpi
Nexus 6 5.96" 1440x2560 560dpi
Nexus 5X 5.2" 1080x1920 420dpi
Nexus 5 4.95" 1080x1920 xxhdpi
Nexus 4 4.7" 768x1280 xhdpi
Galaxy Nexus 4.65" 720x1280 xhdpi
5.4" FWVGA 5.4" 480x854 mdpi
5.1" WVGA 5.1" 480x800 mdpi
Nexus S
480px
800px
4.0"
Size: normal
Ratio: long
Density: hdpi
New Hardware Profile
Import Hardware Profiles
Clone Device...
Previous
Next
Cancel
Finish
Help

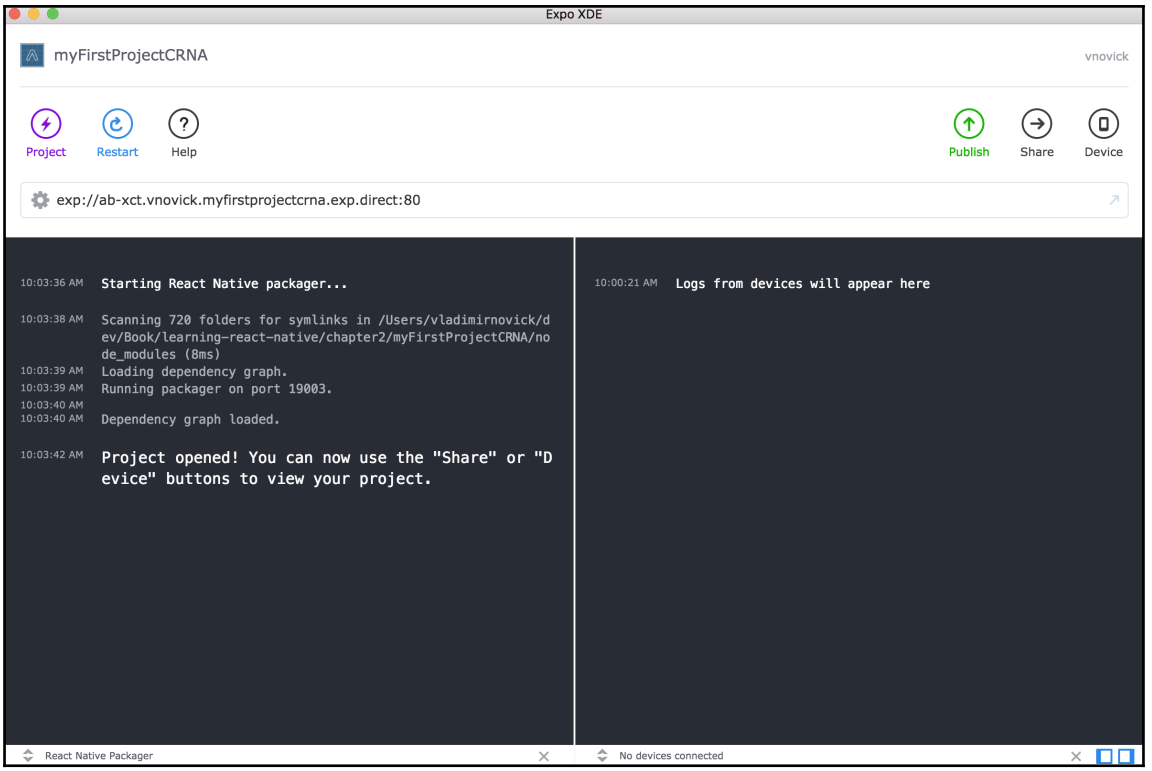

System Image
Android Studio
Select a system image
Recommended
x86 Images
Other Images
Release Name
API Level
ABI
Target
Nougat
25
x86
Android 7.1.1 (with Google APIs)
Nougat Download
24
x86
Android 7.0 (with Google APIs)
Marshmallow Download
23
x86
Android 6.0 (with Google APIs)
Lollipop Download
22
x86
Android 5.1 (with Google APIs)
Marshmallow
API Level
23
Android
6.0
Google Inc.
System Image
X86
These images are recommended because they run the fastest and include support for Google APIs
Questions on API level?
See the API level distribution chart
A system image must be selected to continue.
Previous
Next
Cancel
Finish
Help

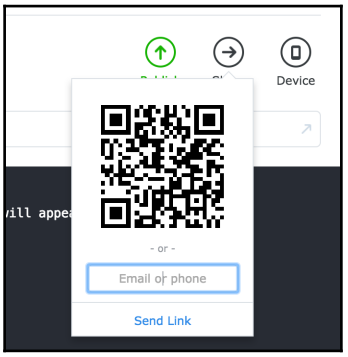

Your Virtual Devices
Android Studio
Type
Name
Resolution
API
Target
CPU/ABI
Size on Disk
Actions
Nexus 5 API 23
1080 × 1920: xxhdpi
23
Android 6.0 (Google APIs)
x86
4 GB
Nexus 6 API 23
1080 × 1920: 420dpi
23
Android 6.0 (Google APIs)
x86
2 GB

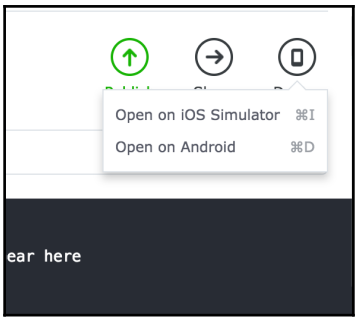

myFirstProject
__tests__
index.android.js
index.ios.js
android
ios
node_modules
.babelrc
.buckconfig
.flowconfig
.gitattributes
.gitignore
.watchmanconfig
app.json
index.android.js
index.ios.js
package.json
yarn.lock


11:50
Welcome to React Native!
To get started, edit index.android.js
Double tap R on your keyboard to reload,
Shake or press menu button for dev menu

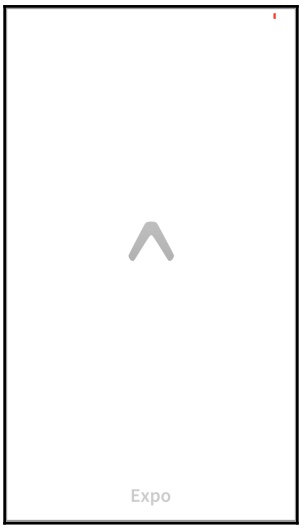

vnovick
Publish
Share
Device
New Project ⌘N
Open Project ⌘O
Close Project
Show in Finder
Open in Terminal
Open in Editor
ct-native/chapter2/myFirstProjectCRNA
ct-native/chapter9/tinderAppClone

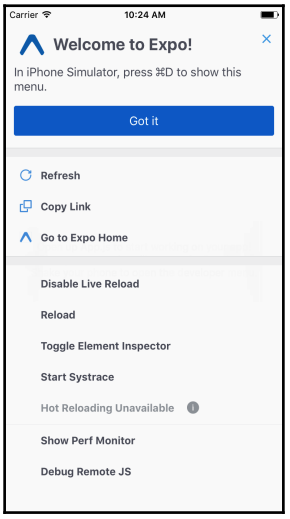

Expo XDE
myFirstProjectCRNA
vnovick
Project
Restart
Help
Publish
Share
Device
exp://ab-xct.vnovick.myfirstprojectcrna.exp.direct:80
10:03:36 AM Starting React Native packager...
10:03:38 AM Scanning 720 folders for symlinks in /Users/vladimirnovick/dev/Book/learning-react-native/chapter2/myFirstProjectCRNA/node_modules (8ms)
10:03:39 AM Loading dependency graph.
10:03:39 AM Running packager on port 19003.
10:03:40 AM
10:03:40 AM Dependency graph loaded.
10:03:42 AM Project opened! You can now use the "Share" or "Device" buttons to view your project.
10:00:21 AM Logs from devices will appear here
React Native Packager
No devices connected

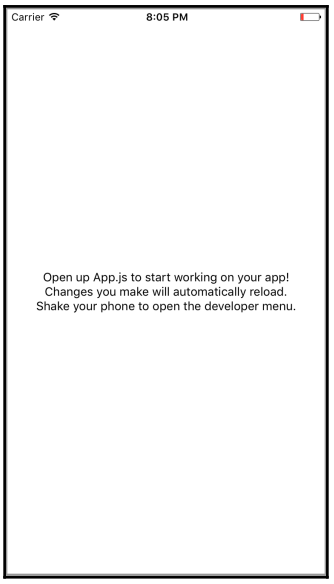

Device
- or -
Email or phone
Send Link


Open on iOS Simulator ⌘I
Open on Android ⌘D
ear here

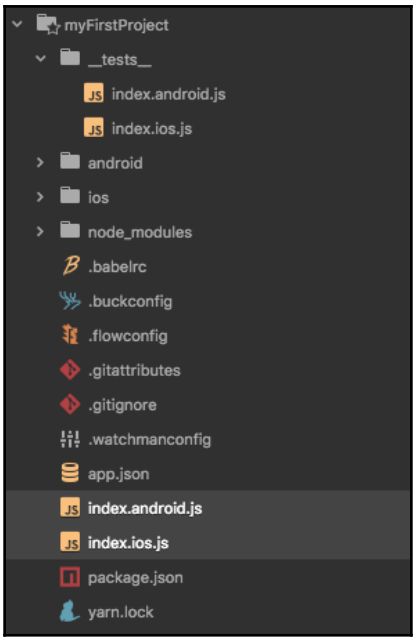

~/dev/Book/learning-react-native/chapter2/myFirstProjectCRNA(master) » npm run ios
> myFirstProjectCRNA@0.1.0 ios /Users/vladimirnovick/dev/Book/learning-react-native/chapter2/myFirstProjectCRNA
> react-native-scripts ios
10:09:41: Starting packager...
10:10:00: Starting simulator...
10:10:36: Packager started!
To view your app with live reloading, point the Expo app to this QR code.
You'll find the QR scanner on the Projects tab of the app.
Or enter this address in the Expo app's search bar:
exp://10.0.0.8:19000
Your phone will need to be on the same local network as this computer.
For links to install the Expo app, please visit https://expo.io.
Logs from serving your app will appear here. Press Ctrl+C at any time to stop.
If you restart the simulator or change the simulated hardware, you may need to restart this process.
Carrier
10:10 AM
Friday
18
Calendar
Photos
Maps
Wallet
Reminders
News
Health
Settings
Open in "Expo"?
Cancel
Open
Safari
Messages

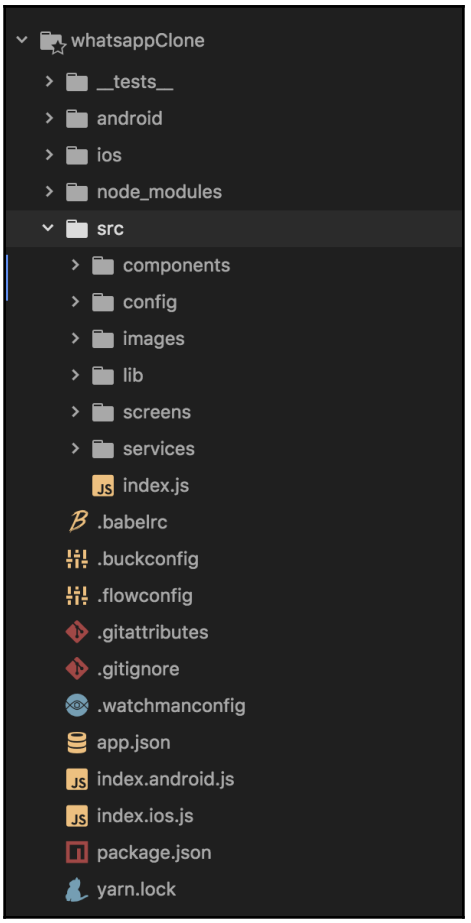

Expo


Carrier
10:24 AM
Welcome to Expo!
In iPhone Simulator, press ⌘D to show this menu.
Got it
Refresh
Copy Link
Go to Expo Home
Disable Live Reload
Reload
Toggle Element Inspector
Start Systrace
Hot Reloading Unavailable
Show Perf Monitor
Debug Remote JS

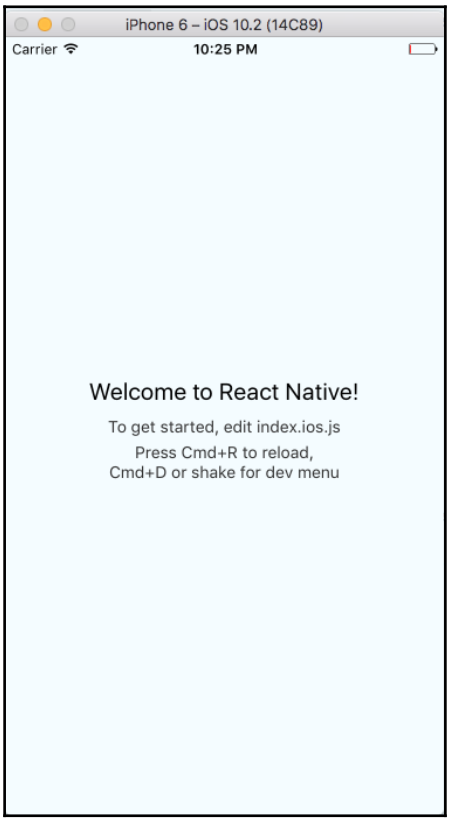

Carrier
8:05 PM
Open up App.js to start working on your app!
Changes you make will automatically reload.
Shake your phone to open the developer menu.

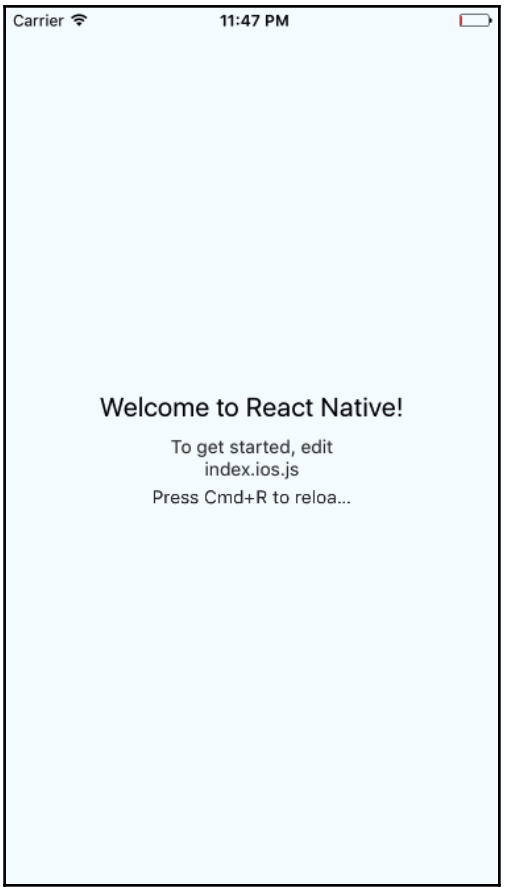

Simulator
File
Edit
Hardware
About Simulator
Services
Reset Content and Settings...
Hide Simulator ⌘H
Hide Others ⌥⌘H
Show All
Quit Simulator ⌘Q

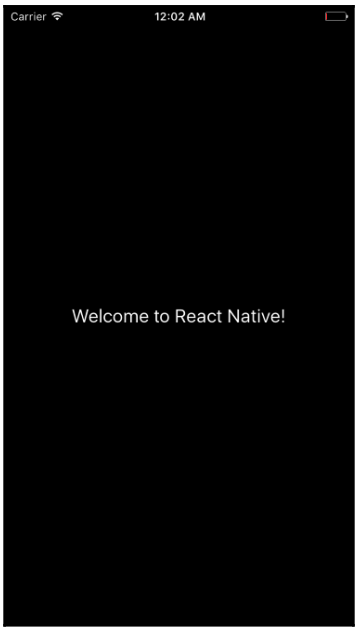

myFirstProject
__tests__
index.android.js
index.ios.js
android
ios
node_modules
.babelrc
.buckconfig
.flowconfig
.gitattributes
.gitignore
.watchmanconfig
app.json
index.android.js
index.ios.js
package.json
yarn.lock

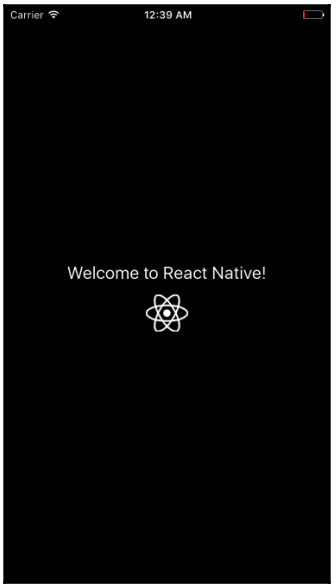

whatsappClone
__tests__
android
ios
node_modules
src
components
config
images
lib
screens
services
index.js
.babelrc
.buckconfig
.flowconfig
.gitattributes
.gitignore
.watchmanconfig
app.json
index.android.js
index.ios.js
package.json
yarn.lock

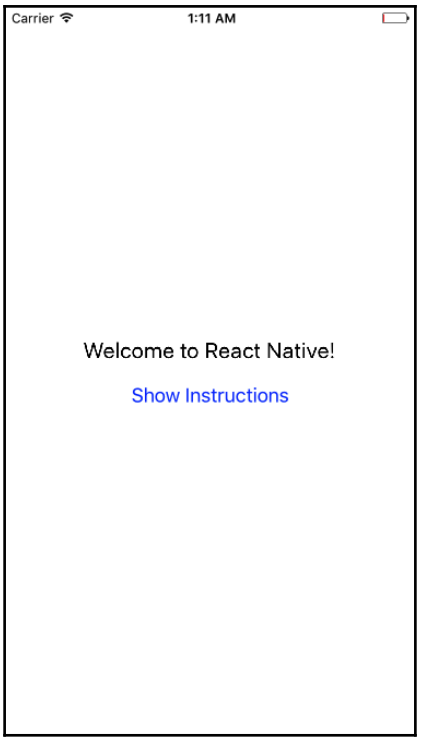
Chapter 3:
Getting Familiar with React Native Components

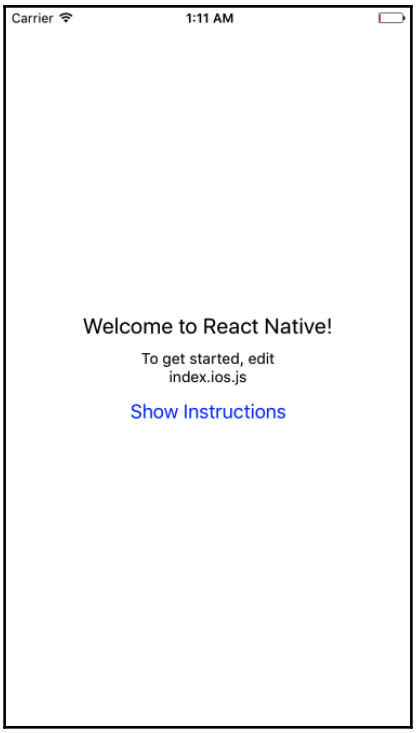


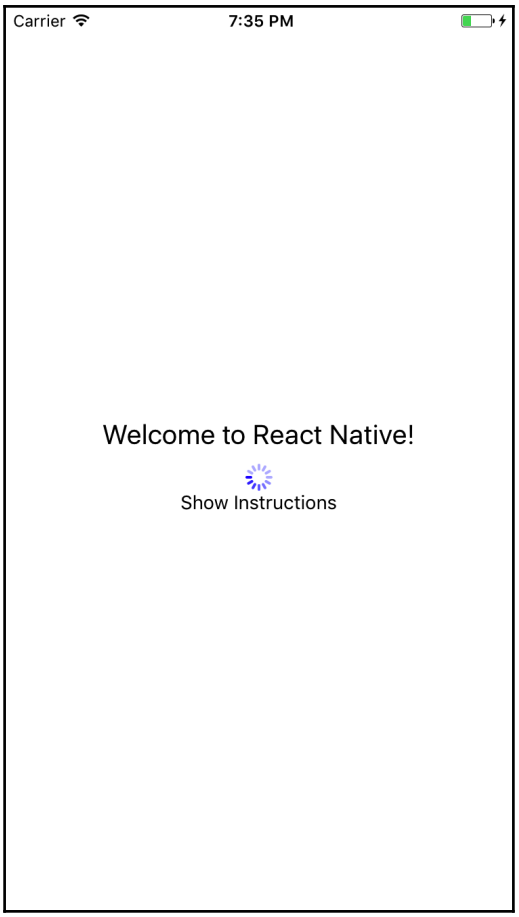

Carrier
11:47 PM
Welcome to React Native!
To get started, edit
index.ios.js
Press Cmd+R to reloa...


Carrier
12:02 AM
Welcome to React Native!

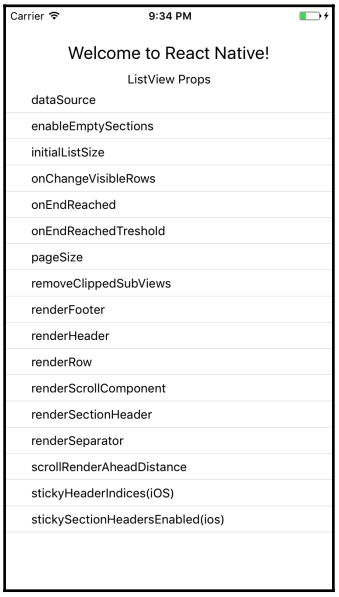

Carrier
12:39 AM
Welcome to React Native!

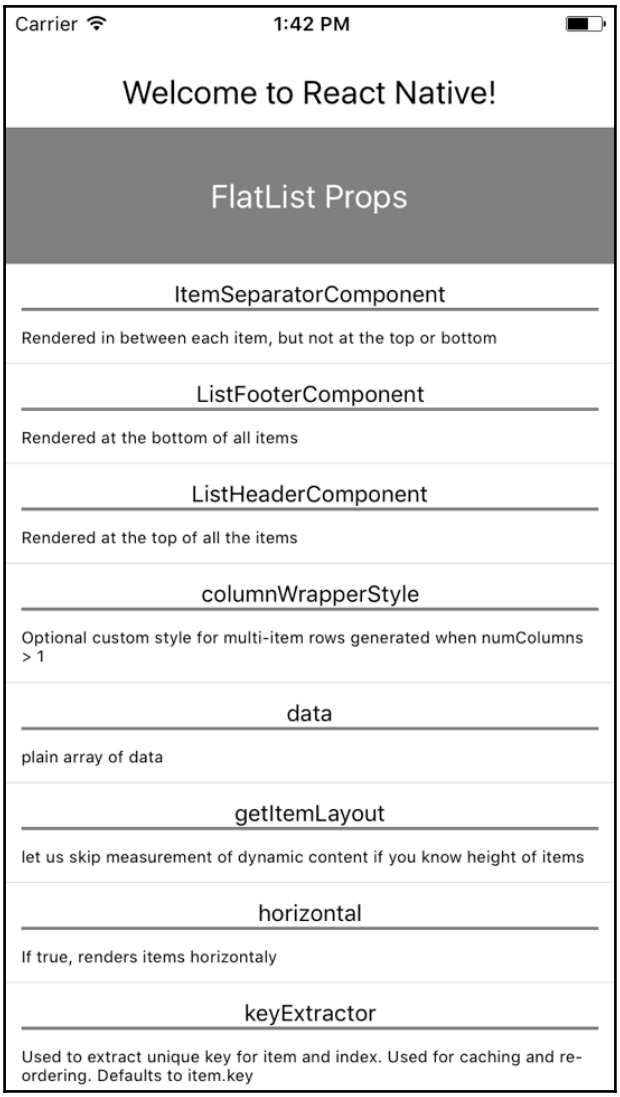

Carrier
1:11 AM
Welcome to React Native!
Show Instructions

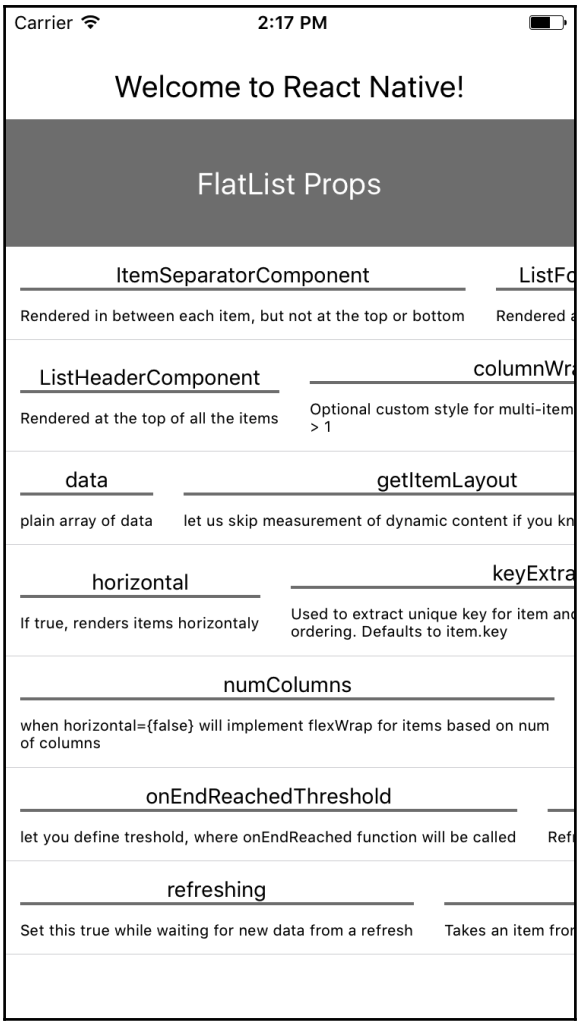

Carrier
1:11 AM
Welcome to React Native!
To get started, edit
index.ios.js
Show Instructions

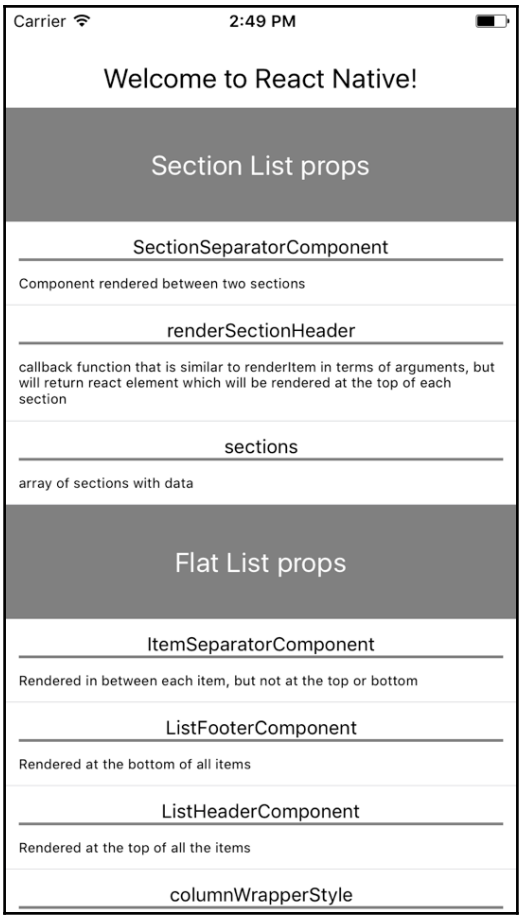

Carrier
7:35 PM
Welcome to React Native!
Show Instructions


Carrier
7:35 PM
Welcome to React Native!
To get started, edit
index.ios.js

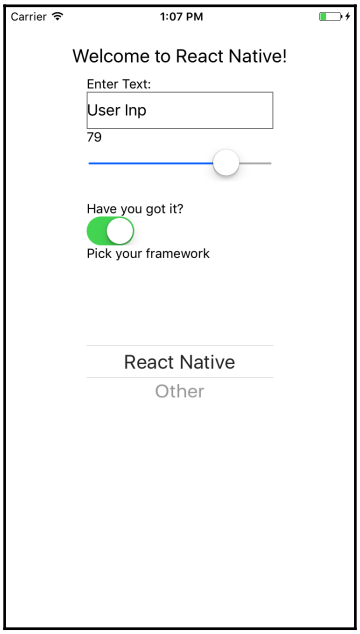

Carrier
9:34 PM
Welcome to React Native!
ListView Props
dataSource
enableEmptySections
initialListSize
onChangeVisibleRows
onEndReached
onEndReachedTreshold
pageSize
removeClippedSubViews
renderFooter
renderHeader
renderRow
renderScrollComponent
renderSectionHeader
renderSeparator
scrollRenderAheadDistance
stickyHeaderIndices(iOS)
stickySectionHeadersEnabled(ios)


Carrier
1:42 PM
Welcome to React Native!
FlatList Props
ItemSeparatorComponent
Rendered in between each item, but not at the top or bottom
ListFooterComponent
Rendered at the bottom of all items
ListHeaderComponent
Rendered at the top of all the items
columnWrapperStyle
Optional custom style for multi-item rows generated when numColumns > 1
data
plain array of data
getItemLayout
let us skip measurement of dynamic content if you know height of items
horizontal
If true, renders items horizontaly
keyExtractor
Used to extract unique key for item and index. Used for caching and re-ordering. Defaults to item.key

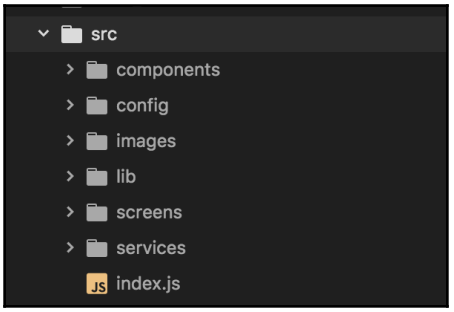

Carrier
2:17 PM
Welcome to React Native!
FlatList Props
ItemSeparatorComponent
Rendered in between each item, but not at the top or bottom
ListFo
Rendered a
ListHeaderComponent
Rendered at the top of all the items
columnWra
Optional custom style for multi-item
> 1
data
plain array of data
getItemLayout
let us skip measurement of dynamic content if you kn
horizontal
If true, renders items horizontaly
keyExtra
Used to extract unique key for item and
ordering. Defaults to item.key
numColumns
when horizontal={false} will implement flexWrap for items based on num
of columns
onEndReachedThreshold
let you define treshold, where onEndReached function will be called
Refr
refreshing
Set this true while waiting for new data from a refresh
Takes an item fron

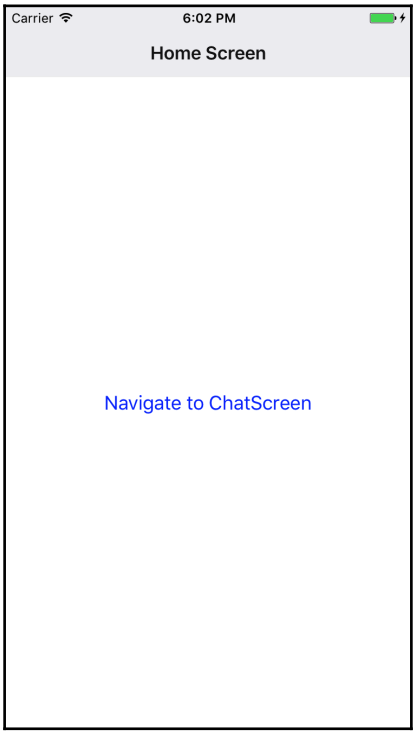

Carrier
2:49 PM
Welcome to React Native!
Section List props
SectionSeparatorComponent
Component rendered between two sections
renderSectionHeader
callback function that is similar to renderItem in terms of arguments, but will return react element which will be rendered at the top of each section
sections
array of sections with data
Flat List props
ItemSeparatorComponent
Rendered in between each item, but not at the top or bottom
ListFooterComponent
Rendered at the bottom of all items
ListHeaderComponent
Rendered at the top of all the items
columnWrapperStyle

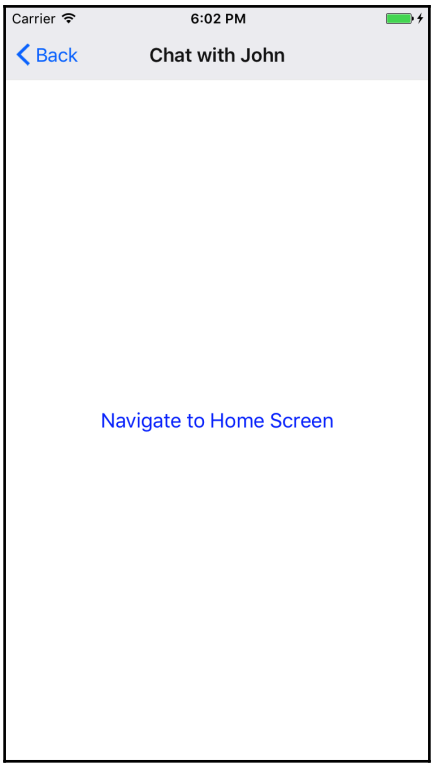

Carrier
1:35 PM
Welcome to React Native!
React Native 0.42
GitHub
React
Docs
Help
Showcase
Blog
WebView
WebView renders web content in a native view.
import React, { Component } from 'react';
import { WebView } from 'react-native';
class MyWeb extends Component {
render() {
return (
<WebView
source={{uri:
'https://github.com/facebook/react-
native'}}
style={{marginTop: 20}}
/>
);
}
}
You can use this component to navigate back and forth in the web view's history and configure various properties for the web content.


Carrier
1:07 PM
Welcome to React Native!
Enter Text:
User Inp
79
Have you got it?
Pick your framework
React Native
Other

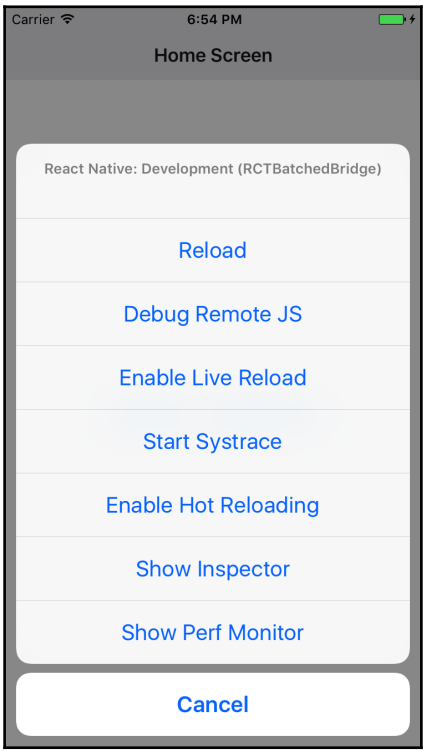

First
Second

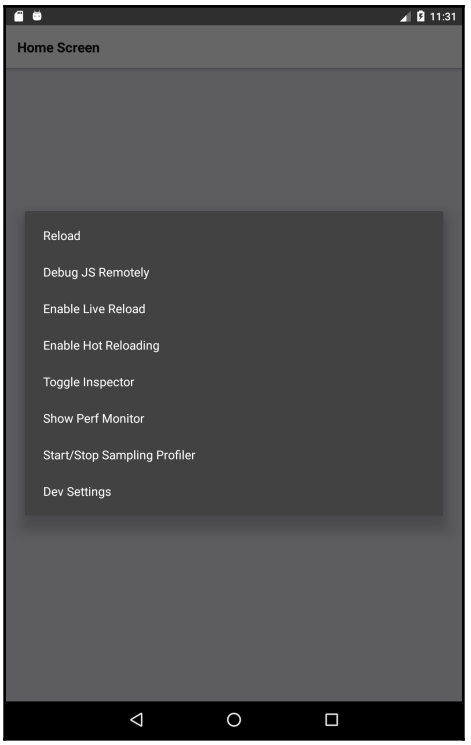

src
components
config
images
lib
screens
services
index.js

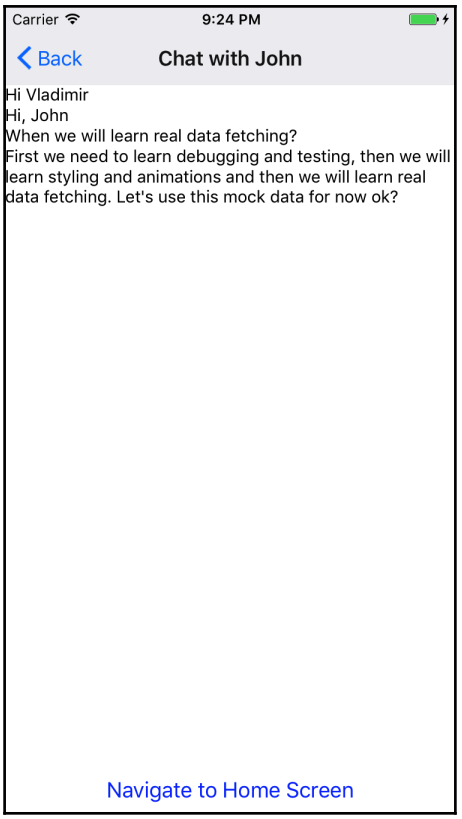

Carrier
6:02 PM
Home Screen
Navigate to ChatScreen

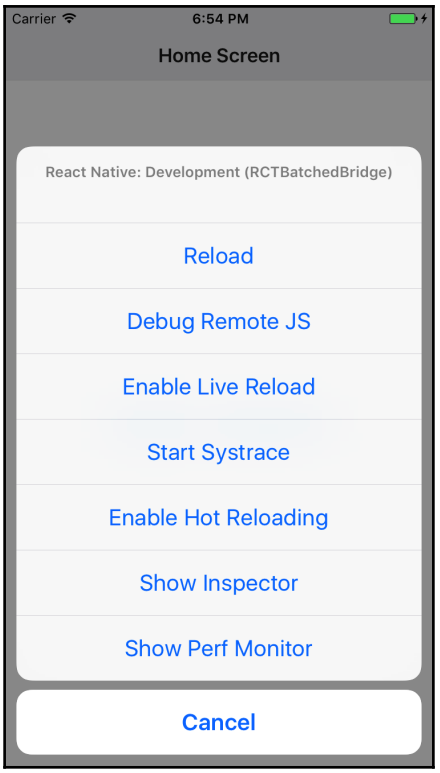

Carrier
6:02 PM
Back
Chat with John
Navigate to Home Screen

Chapter 4:
Debugging and Testing React Native




11:31
Home Screen
Reload
Debug JS Remotely
Enable Live Reload
Enable Hot Reloading
Toggle Inspector
Show Perf Monitor
Start/Stop Sampling Profiler
Dev Settings


Carrier
9:24 PM
Back
Chat with John
Hi Vladimir
Hi, John
When we will learn real data fetching?
First we need to learn debugging and testing, then we will learn styling and animations and then we will learn real data fetching. Let's use this mock data for now ok?
Navigate to Home Screen

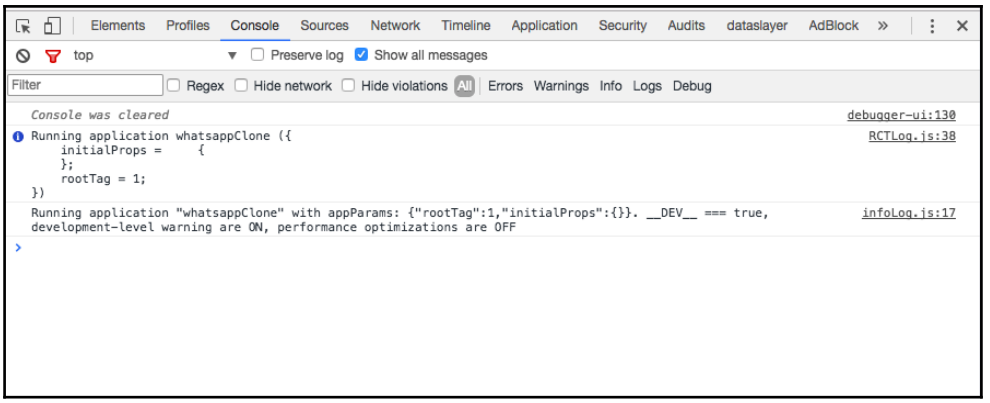


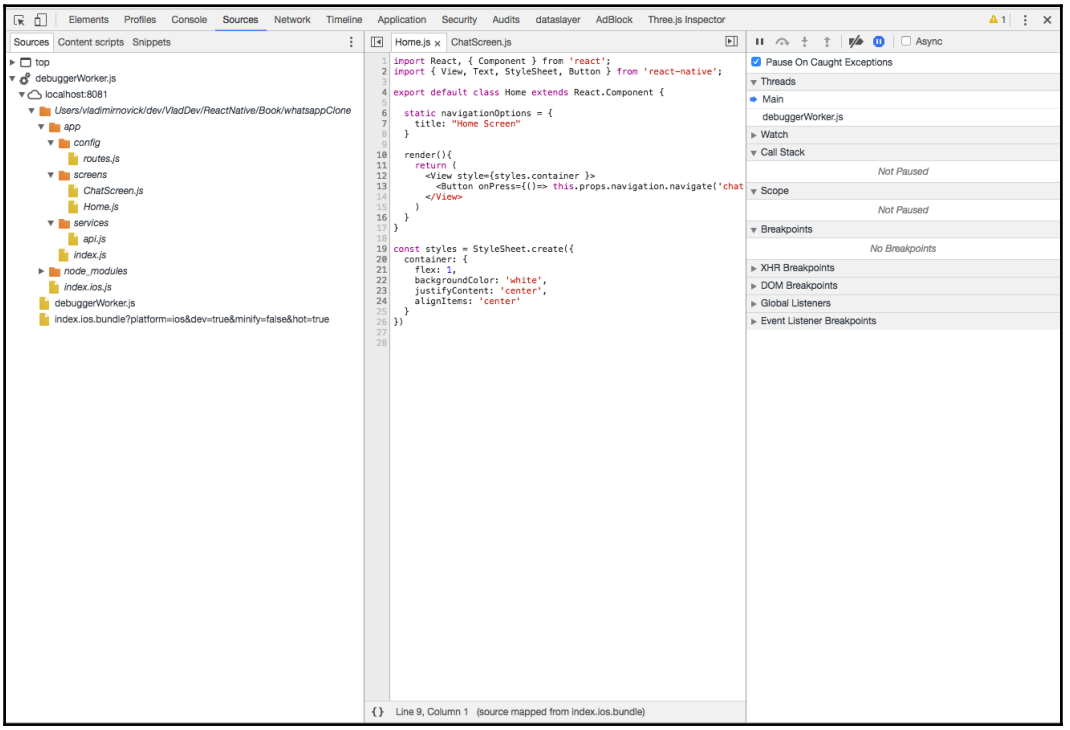
Dark background

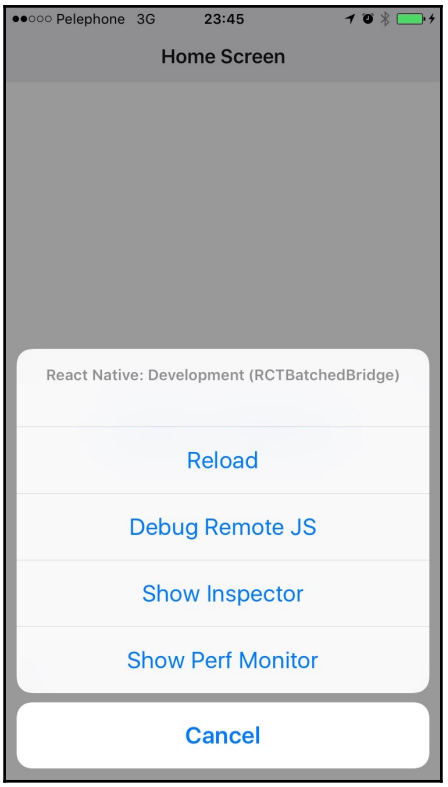
React Native JS code runs inside this Chrome tab.

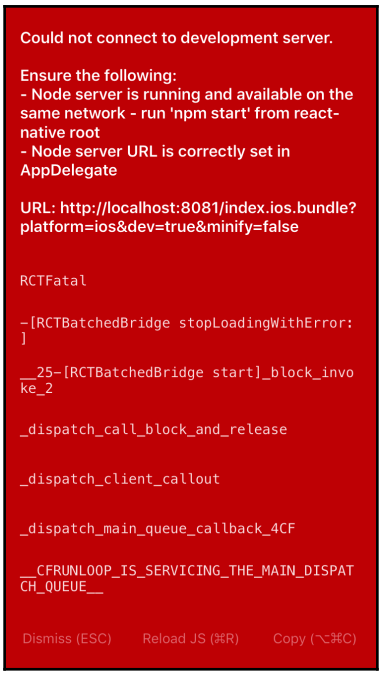
Press ⌘⌥J to open Developer Tools. Enable Pause On Caught Exceptions for a better debugging experience.

Status: Debugger session #13679 active.

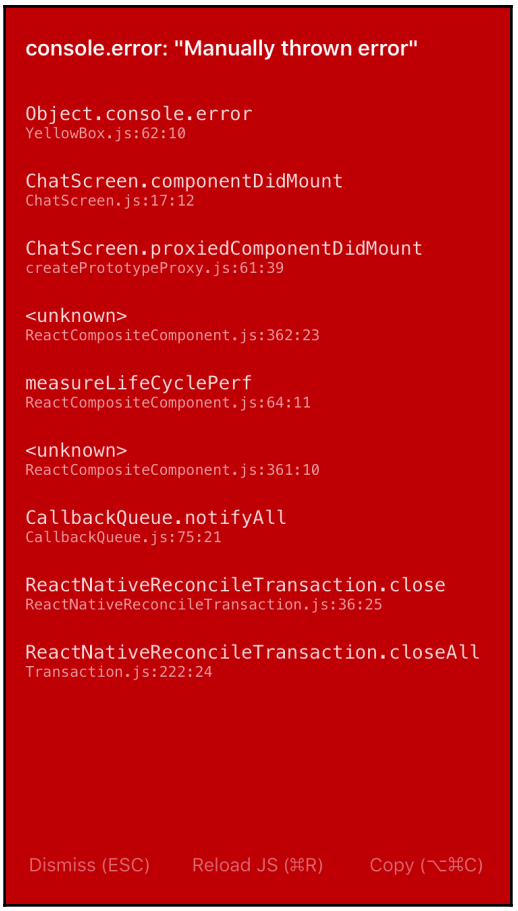

Elements
Profiles
Console
Sources
Network
Timeline
Application
Security
Audits
dataslayer
AdBlock
top
Preserve log
Show all messages
Filter
Regex
Hide network
Hide violations
All
Errors
Warnings
Info
Logs
Debug
Console was cleared
debugger-ui:130
Running application whatsappClone ({
initialProps = {
};
rootTag = 1;
})
RCTLog.js:38
Running application "whatsappClone" with appParams: {"rootTag":1,"initialProps":{}}. __DEV__ === true, development-level warning are ON, performance optimizations are OFF
infoLog.js:17

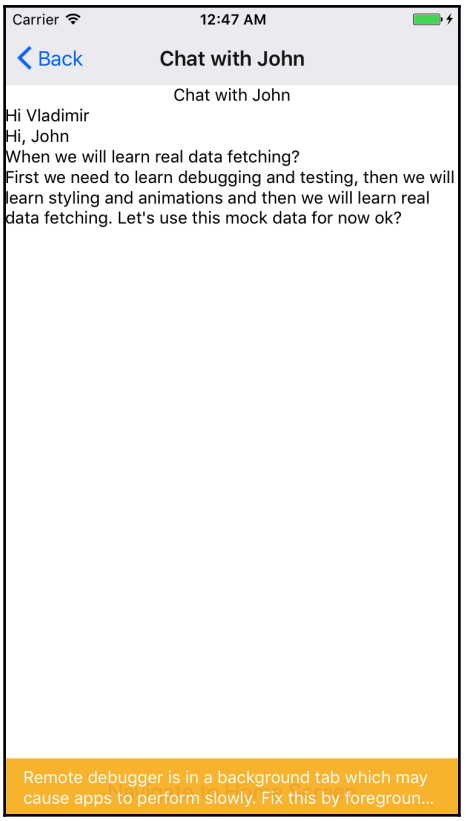

Elements
Profiles
Console
Sources
Network
Timeline
Application
Security
Audits
dataslayer
AdBlock
Three.js Inspector
Sources
Content scripts
Snippets
top
debuggerWorker.js
localhost:8081
Users/vladimirnovick/dev/VladDev/ReactNative/Book/whatsappClone
app
config
routes.js
screens
ChatScreen.js
Home.js
services
api.js
index.js
node_modules
index.ios.js
debuggerWorker.js
index.ios.bundle?platform=ios&dev=true&minify=false&hot=true
Home.js
ChatScreen.js
import React, { Component } from 'react';
import { View, Text, StyleSheet, Button } from 'react-native';
export default class Home extends React.Component {
static navigationOptions = {
title: "Home Screen"
}
render(){
return (
<View style={styles.container }>
<Button onPress={()=> this.props.navigation.navigate('chat
</View>
)
}
}
const styles = StyleSheet.create({
container: {
flex: 1,
backgroundColor: 'white',
justifyContent: 'center',
alignItems: 'center'
}
})
{}
Line 9, Column 1
(source mapped from index.ios.bundle)
Async
Pause On Caught Exceptions
Threads
Main
debuggerWorker.js
Watch
Call Stack
Not Paused
Scope
Not Paused
Breakpoints
No Breakpoints
XHR Breakpoints
DOM Breakpoints
Global Listeners
Event Listener Breakpoints

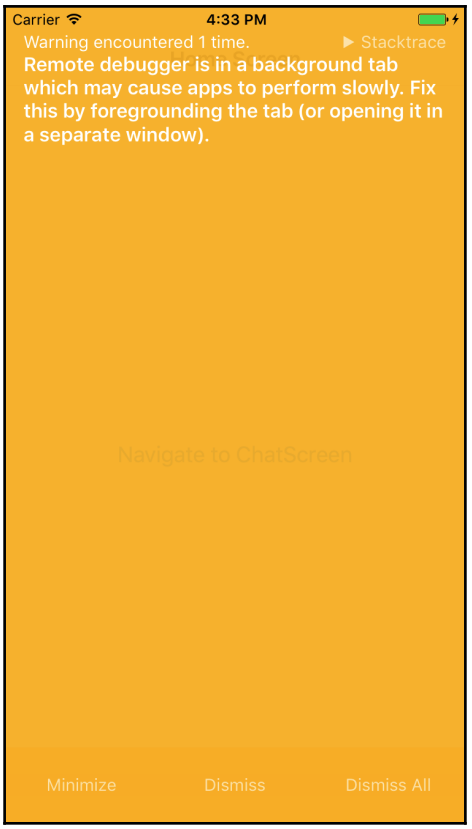

●●○○○ Pelephone 3G
23:45
Home Screen
React Native: Development (RCTBatchedBridge)
Reload
Debug Remote JS
Show Inspector
Show Perf Monitor
Cancel

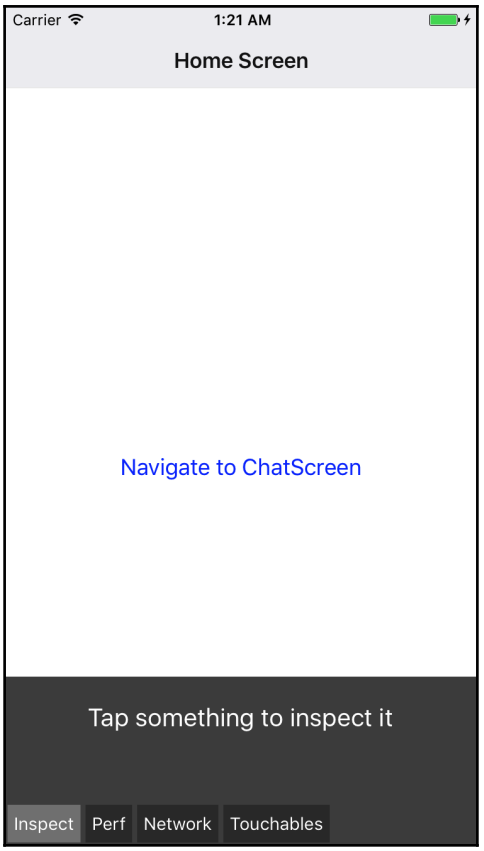

Could not connect to development server.
Ensure the following:
- Node server is running and available on the same network - run 'npm start' from react-native root
- Node server URL is correctly set in AppDelegate
URL: http://localhost:8081/index.ios.bundle?platform=ios&dev=true&minify=false
RCTFatal
-[RCTBatchedBridge stopLoadingWithError:]
__25-[RCTBatchedBridge start]_block_invoke_2
_dispatch_call_block_and_release
_dispatch_client_callout
_dispatch_main_queue_callback_4CF
__CFRUNLOOP_IS_SERVICING_THE_MAIN_DISPATCH_QUEUE__
Dismiss (ESC)
Reload JS (⌘R)
Copy (⌥⌘C)

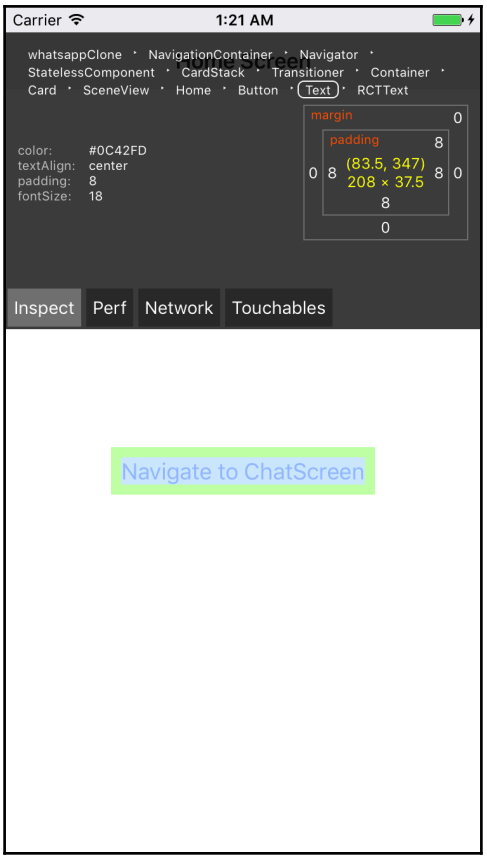

Unable to resolve module `react-navigation` from `/Users/vladimirnovick/dev/VladDev/ReactNative/Book/whatsappClone/app/index.js`: Module does not exist in the module map or in these directories:
/Users/vladimirnovick/dev/VladDev/ReactNative/Book/whatsappClone/node_modules
This might be related to https://github.com/facebook/react-native/issues/4968
To resolve try the following:
1. Clear watchman watches: `watchman watch-del-all`.
2. Delete the `node_modules` folder: `rm -rf node_modules && npm install`.
3. Reset packager cache: `rm -fr $TMPDIR/react-*` or `npm start --reset-cache`.
RCTFatal
-[RCTBatchedBridge stopLoadingWithError:]
__25-[RCTBatchedBridge start]_block_invoke_2
_dispatch_call_block_and_release
Dismiss (ESC)
Reload JS (⌘R)
Copy (⌥⌘C)

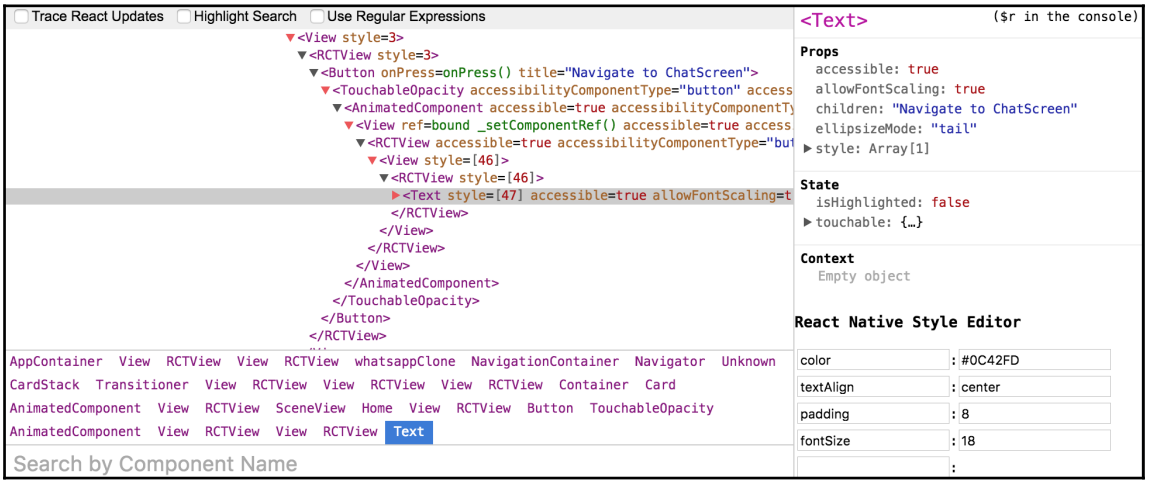

console.error: "Manually thrown error"
Object.console.error
YellowBox.js:62:10
ChatScreen.componentDidMount
ChatScreen.js:17:12
ChatScreen.proxiedComponentDidMount
createPrototypeProxy.js:61:39
<unknown>
ReactCompositeComponent.js:362:23
measureLifeCyclePerf
ReactCompositeComponent.js:64:11
<unknown>
ReactCompositeComponent.js:361:10
CallbackQueue.notifyAll
CallbackQueue.js:75:21
ReactNativeReconcileTransaction.close
ReactNativeReconcileTransaction.js:36:25
ReactNativeReconcileTransaction.closeAll
Transaction.js:222:24
Dismiss (ESC)
Reload JS (⌘R)
Copy (⌥⌘C)

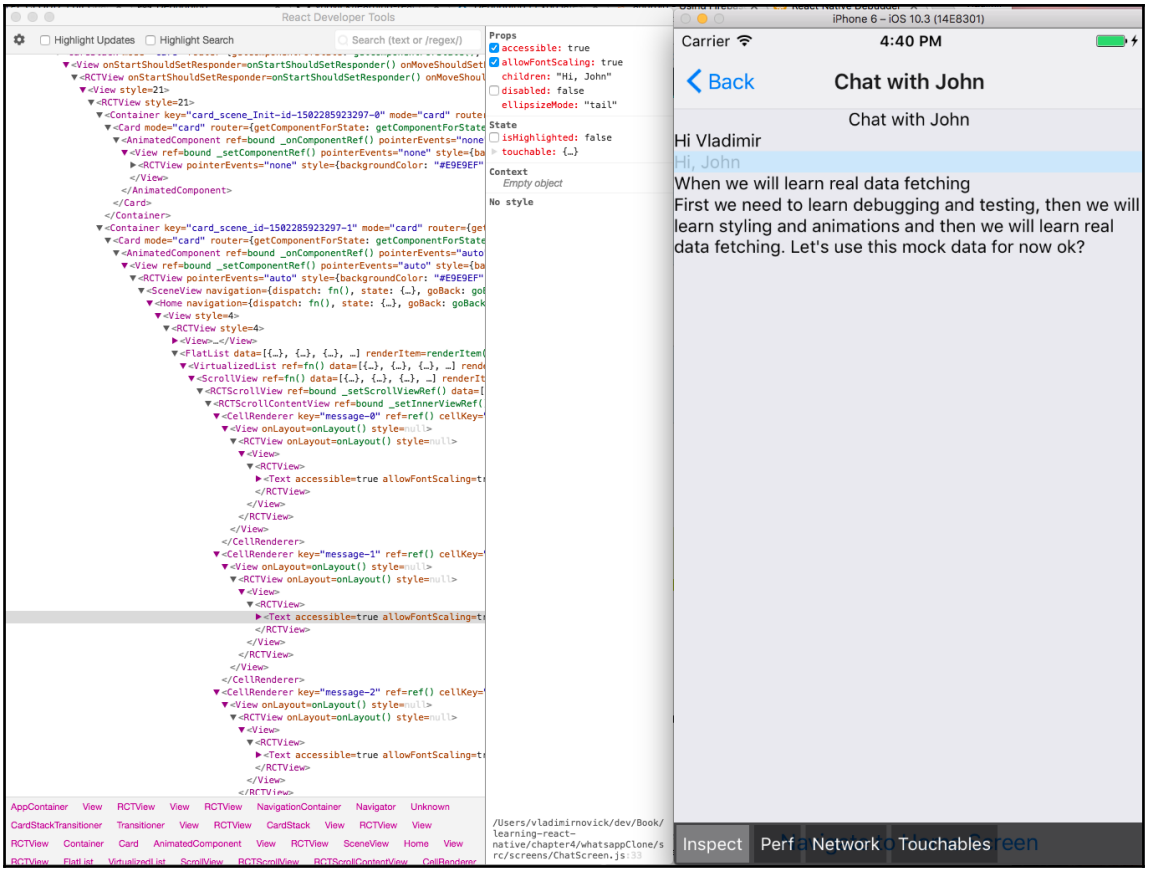

Carrier
12:47 AM
Back
Chat with John
Chat with John
Hi Vladimir
Hi, John
When we will learn real data fetching?
First we need to learn debugging and testing, then we will learn styling and animations and then we will learn real data fetching. Let's use this mock data for now ok?
Remote debugger is in a background tab which may cause apps to perform slowly. Fix this by foregroun...

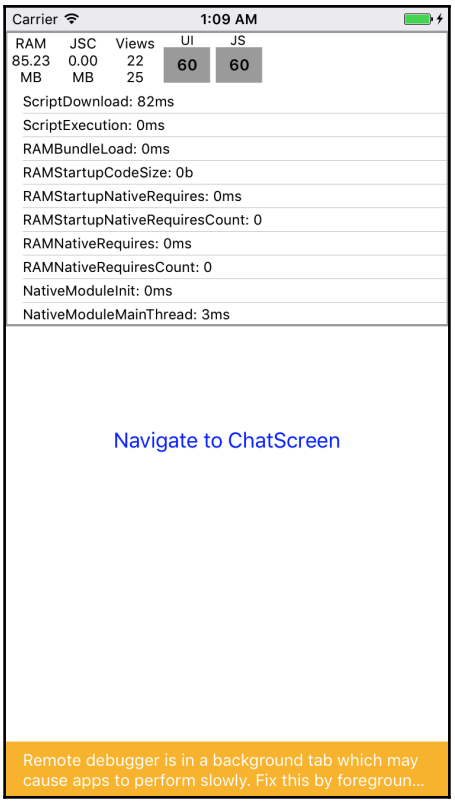

Carrier
4:33 PM
Warning encountered 1 time.
Stacktrace
Remote debugger is in a background tab which may cause apps to perform slowly. Fix this by foregrounding the tab (or opening it in a separate window).
Navigate to ChatScreen
Minimize
Dismiss
Dismiss All


Carrier
1:21 AM
Home Screen
Navigate to ChatScreen
Tap something to inspect it
Inspect
Perf
Network
Touchables

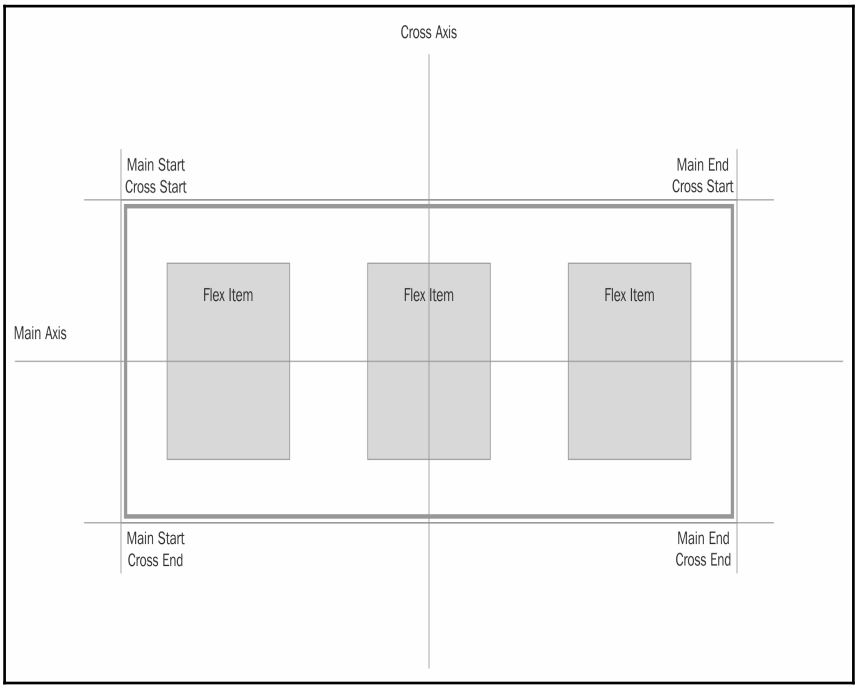

Carrier
1:21 AM
whatsappClone • NavigationContainer • Navigator •
StatelessComponent • CardStack • Transitioner • Container •
Card • SceneView • Home • Button • Text • RCTText
color: #0C42FD
textAlign: center
padding: 8
fontSize: 18
margin
padding
(83.5, 347)
208 × 37.5
Inspect
Perf
Network
Touchables
Navigate to ChatScreen

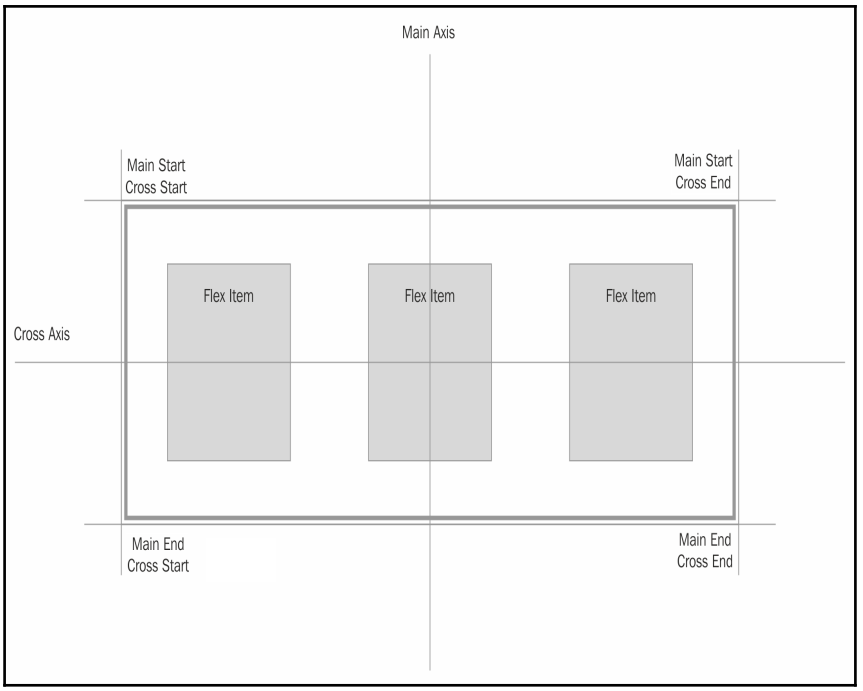

Trace React Updates
Highlight Search
Use Regular Expressions
<View style=3>
<RCTView style=3>
<Button onPress=onPress() title="Navigate to ChatScreen">
<TouchableOpacity accessibilityComponentType="button"
<AnimatedComponent accessible=true
<View ref=bound _setComponentRef() accessible=true
<RCTView accessible=true
<View style=[46]>
<RCTView style=[46]>
<Text style=[47] accessible=true allowFontScaling=t
AppContainer View RCTView View RCTView whatsappClone NavigationContainer Navigator Unknown
CardStack Transitioner View RCTView View RCTView View RCTView Container Card
AnimatedComponent View RCTView SceneView Home View RCTView Button TouchableOpacity
AnimatedComponent View RCTView View RCTView Text
Search by Component Name
<Text>
($r in the console)
Props
accessible: true
allowFontScaling: true
children: "Navigate to ChatScreen"
ellipsizeMode: "tail"
style: Array[1]
State
isHighlighted: false
touchable: {…}
Context
Empty object
React Native Style Editor
color : #0C42FD
textAlign : center
padding : 8
fontSize : 18

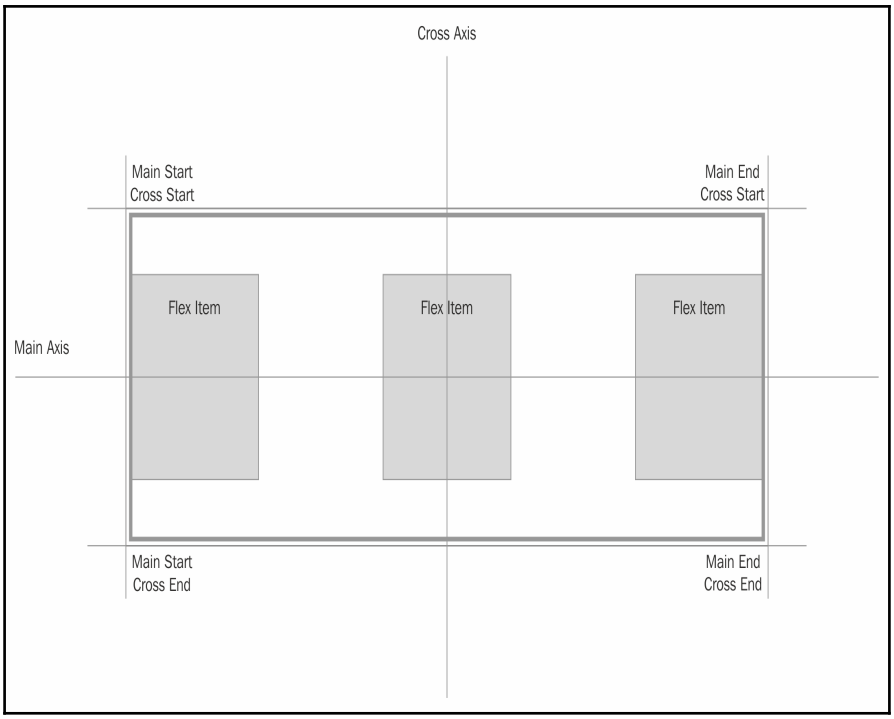

React Developer Tools
Highlight Updates
Highlight Search
Search (text or /regex/)
Props
accessible: true
allowFontScaling: true
children: "Hi, John"
disabled: false
ellipsizeMode: "tail"
State
isHighlighted: false
touchable: {…}
Context
Empty object
No style
iPhone 6 – iOS 10.3 (14E8301)
Carrier
4:40 PM
Back
Chat with John
Chat with John
Hi Vladimir
Hi, John
When we will learn real data fetching
First we need to learn debugging and testing, then we will learn styling and animations and then we will learn real data fetching. Let's use this mock data for now ok?
Inspect
Perf
Network
Touchables

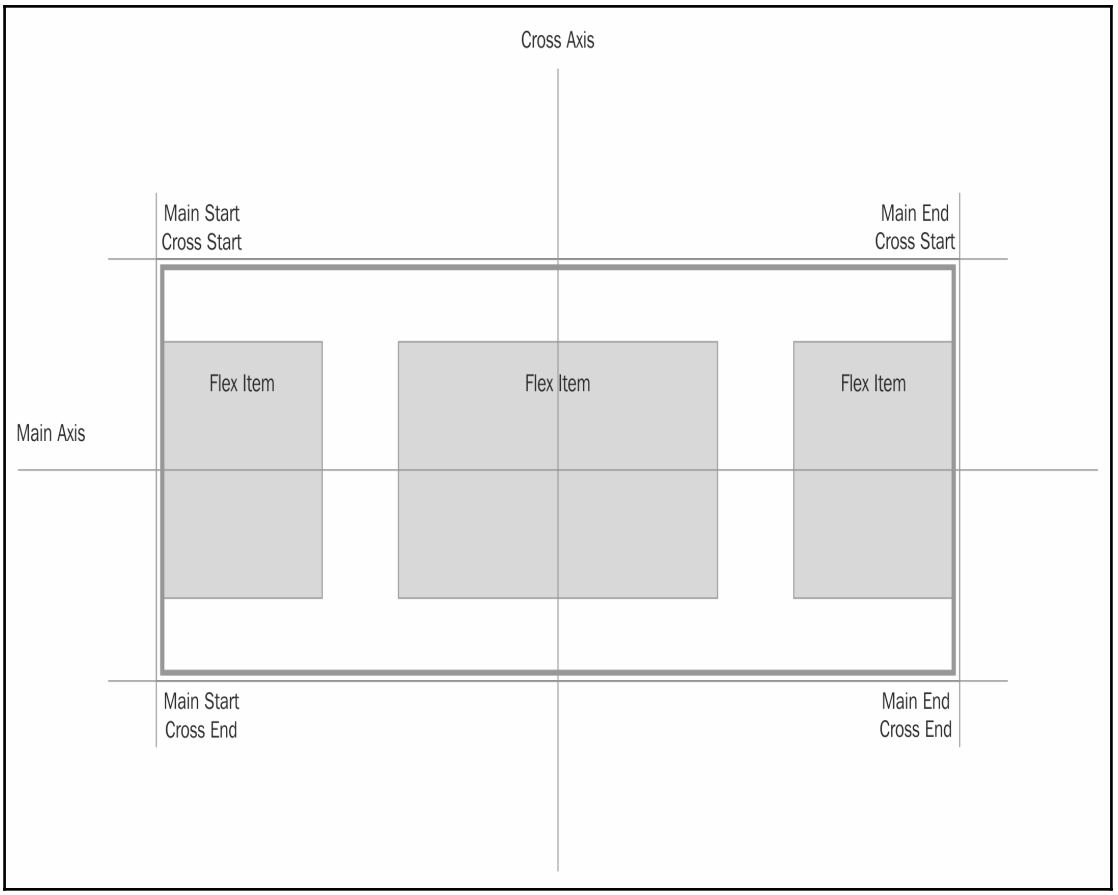

Carrier
1:09 AM
RAM
85.23
MB
JSC
0.00
MB
Views
22
25
UI
60
JS
60
ScriptDownload: 82ms
ScriptExecution: 0ms
RAMBundleLoad: 0ms
RAMStartupCodeSize: 0b
RAMStartupNativeRequires: 0ms
RAMStartupNativeRequiresCount: 0
RAMNativeRequires: 0ms
RAMNativeRequiresCount: 0
NativeModuleInit: 0ms
NativeModuleMainThread: 3ms
Navigate to ChatScreen
Remote debugger is in a background tab which may cause apps to perform slowly. Fix this by foregroun...

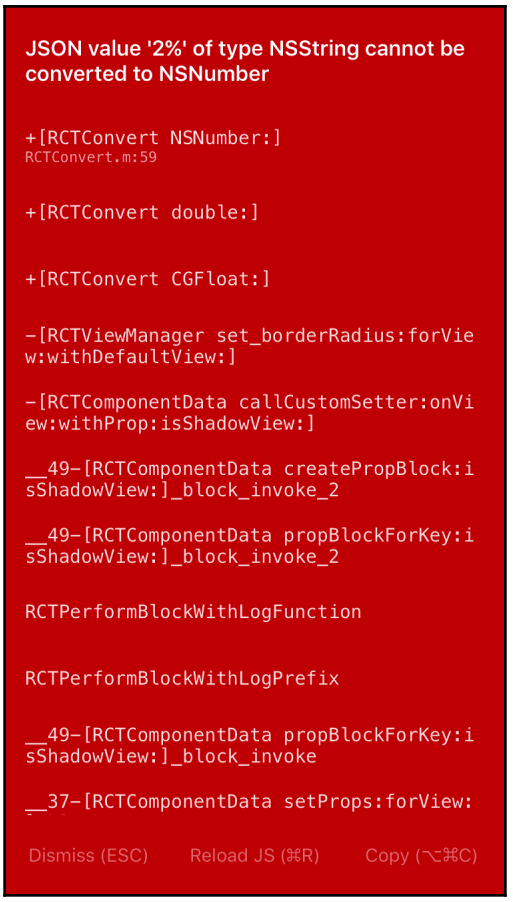
Chapter 5:
Bringing the Power of Flexbox to the Native World




Main Axis
Main Start
Cross Start
Main Start
Cross End
Flex Item
Flex Item
Flex Item
Cross Axis
Main End
Cross Start
Main End
Cross End


Cross Axis
Main Start
Cross Start
Main End
Cross Start
Flex Item
Flex Item
Flex Item
Main Axis
Main Start
Cross End
Main End
Cross End


Cross Axis
Main Start
Cross Start
Main End
Cross Start
Flex Item
Flex Item
Flex Item
Main Axis
Main Start
Cross End
Main End
Cross End


JSON value '2%' of type NSString cannot be converted to NSNumber
+[RCTConvert NSNumber:]
RCTConvert.m:59
+[RCTConvert double:]
+[RCTConvert CGFloat:]
-[RCTViewManager set_borderRadius:forView:withDefaultView:]
-[RCTComponentData callCustomSetter:onView:withProp:isShadowView:]
__49-[RCTComponentData createPropBlock:isShadowView:]_block_invoke_2
__49-[RCTComponentData propBlockForKey:isShadowView:]_block_invoke_2
RCTPerformBlockWithLogFunction
RCTPerformBlockWithLogPrefix
__49-[RCTComponentData propBlockForKey:isShadowView:]_block_invoke
__37-[RCTComponentData setProps:forView:
Dismiss (ESC)
Reload JS (⌘R)
Copy (⌥⌘C)


Carrier
4:38 PM
Back
Chat with John

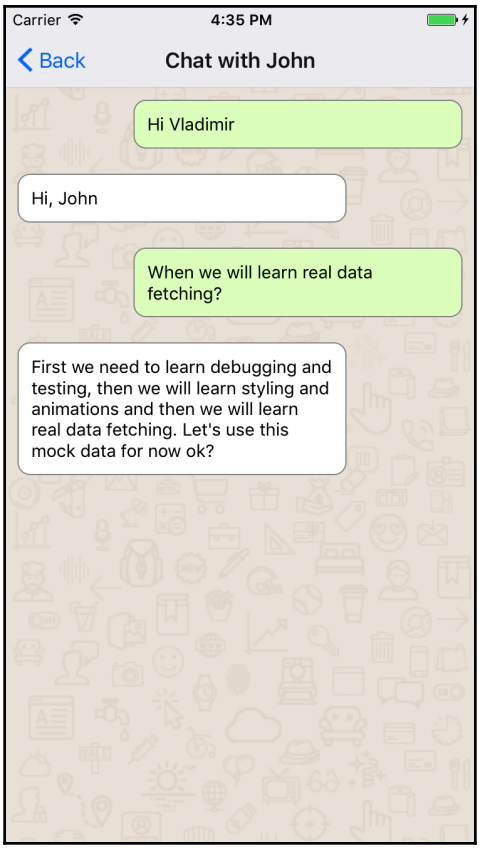

Carrier
4:35 PM
Back
Chat with John
Hi Vladimir
Hi, John
When we will learn real data fetching?
First we need to learn debugging and testing, then we will learn styling and animations and then we will learn real data fetching. Let's use this mock data for now ok?

Table of Contents

Index