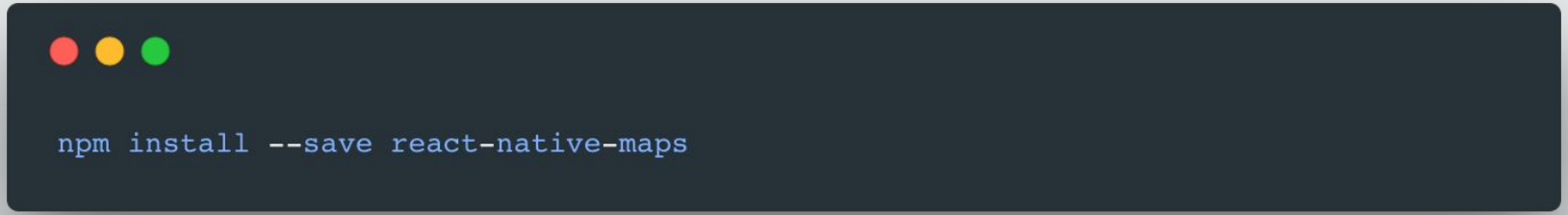
1. Dentro da pasta **app/pages** crie um arquivo chamado **Gps.js**

No terminal rode:



No terminal rode:

```
react-native link react-native-maps
```

Dentro de android/app/src/main/AndroidManifest.xml, altere:

```
<application>
...
  <meta-data
    android:name="com.google.android.geo.API_KEY"
    android:value="AIzaSyBkxPY1I6vMblwi6QaiwKFgyVtvyMid1gQ"
  />
</application>
```

Dentro de android/app/src/main/AndroidManifest.xml, altere:

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
```

Dentro de android/app/build.gradle, substitua compile project(':react-native-maps'):, por:

```
compile(project(':react-native-maps')){
    exclude group: 'com.google.android.gms'
}
compile ("com.google.android.gms:play-services-base:10.0.1") {
  force = true;
}
compile ("com.google.android.gms:play-services-maps:10.0.1") {
  force = true;
}
compile ("com.google.android.gms:play-services-gcm:10.0.1") {
  force = true;
}
```

1. Instale o google play services dentro do genymotion

Dentro de app/pages/Gps.js

```
import React, { Component } from 'react';
import {
  StyleSheet,
  Text,
  View,
  Button
} from 'react-native';
import MapView from 'react-native-maps';
```

Dentro de app/pages/Gps.js

```
export default class Gps extends Component {

  static navigationOptions = ({ navigation }) => ({
    drawerLabel: 'Gps',
    title: 'GPS - MAPA',
    headerLeft:
      <View style={{ marginLeft: 10 }}>
        <Button title="menu" onPress={() => navigation.navigate('DrawerToggle')} />
      </View>
  });

  constructor() {
    super();
    this.state = {
      region: {}
    };
  }

}
```

Dentro de app/pages/Gps.js, método render:

```
render() {
  if (Object.keys(this.state.region).length) {
    return (
      <View style={styles.container}>
        <MapView
          region={this.state.region}
          style={styles.map}
        >
          <MapView.Marker
            coordinate={{ ...this.state.region }}
            title={'Minha localização'}
            description={'estou na ufmg'}
          />
        </MapView>
      </View>
    );
  }

  return (
    <View>
      <Text>carregando localização...</Text>
    </View>
  );
}
```

Dentro de app/pages/Gps.js, crie o método :

```
componentDidMount() {
    navigator.geolocation.getCurrentPosition(
      (region) => {
        this.setState({
          region: {
            latitude: region.coords.latitude,
            longitude: region.coords.longitude,
            latitudeDelta: 0.0922,
            longitudeDelta: 0.0421
          }
        });
      },
      (error) => console.log(error),
      { enableHighAccuracy: true, timeout: 10000, maximumAge: 1000 }
    );
}
```

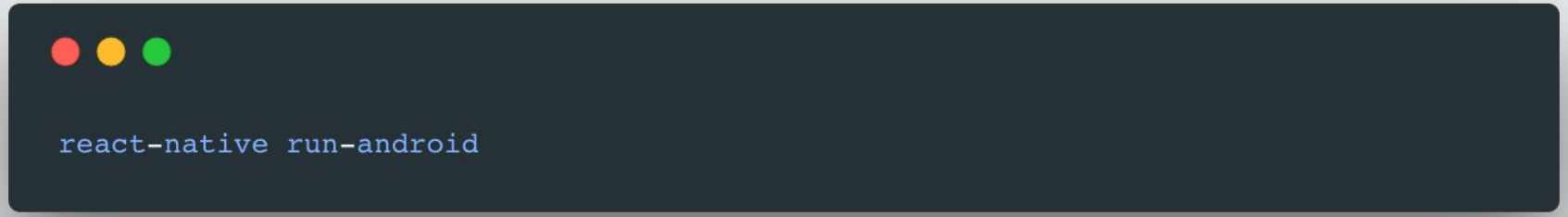
Dentro de app/pages/Gps.js

```
const styles = StyleSheet.create({
  container: {
    flex: 1
  },
  map: {
    ...StyleSheet.absoluteFillObject
  }
});
```

Dentro de app/pages/index.js, altere:

```
...

import Gps from './pages/Gps';

const Drawer = DrawerNavigator({
  ...

  Gps: { screen: Gps }
}, {
  contentComponent: props => <ScrollView><DrawerItems {...props} /></ScrollView>,
});

...
```

No terminal rode:



Veja o resultado!

1. Vamos escutar as mudanças do gps

Dentro de app/pages/Gps.js, adicione dentro do componentDidMount:

```
this.watchId = navigator.geolocation.watchPosition(
     (region) => {
       this.setState({
         region: {
           latitude: region.coords.latitude,
           longitude: region.coords.longitude,
           latitudeDelta: 0.0922,
           longitudeDelta: 0.0421
         }
       });
     },
     (error) => console.log(error),
     { enableHighAccuracy: true, timeout: 10000, maximumAge: 1000 }
   );
```

Dentro de app/pages/Gps.js, adicione o método:

```
componentWillUnmount() {
  navigator.geolocation.clearWatch(this.watchId);
}
```

Veja o resultado!