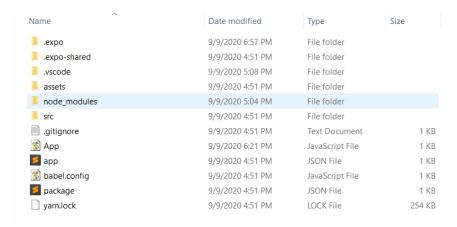
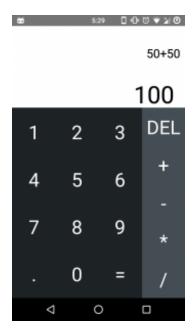


Lab 4: JavaScript for Mobile Device Programming

1. ให้นักศึกษาทำการสร้าง New Project ใหม่ ชื่อ ProjectCal ใน Folder Mobile\<รหัสนักศึกษา>\ProjectCal แบบ Blank Project โดยใช้คำสั่ง



2. ให้นักศึกษาทำการสร้างโปรแกรมเครื่องคิดเลขบนอุปกรณ์เคลื่อนที่ ดังแสดงตามรูปข้างล่างนี้



บันทึกผลการทดลอง:



```
Js App.js
           ×
                Js Keyboard.js
                                 Js Button.js
Js App.js > 😭 App > 🔑 fontSize
       import React, { useState } from "react";
       import { StyleSheet, Text, View } from "react-native";
       import Keyboard from "./src/Keyboard";
       export default function App() {
         const [text, setText] = useState("");
         const [result, setResult] = useState("");
         return (
           <View style={styles.container}>
             <View style={{ flex: 2 }}>
               <Text style={styles.text}>{text}</Text>
 12
               <Text style={{\left\{\left\}...styles.text, fontSize: 40\right\}}>{result}</Text>
              </View>
             <Keyboard setText={setText} text={text} setResult={setResult} />
           </View>
         );
       const styles = StyleSheet.create({
         container: {
           flex: 1,
           backgroundColor: "#fff",
           flexDirection: "column",
           marginTop: 25,
         },
         text:{
           textAlign: "right",
           fontSize: 30,
           margin: 10
       });
```



```
JS Keyboard.js X JS Button.js
Js App.js
src > 15 Keyboard.js > [6] Keyboard
       import React from "react";
       import { StyleSheet, Text, View } from "react-native";
       import Button from "./Button";
       const Keyboard = ({ setText, text, setResult }) => {
        const setInfix = (e, text1) => {
           setText(text+text1)
         const deleteText = (e) => {
         if (text !== "") {
            text = text.slice(0, text.length - 1);
             setText(text);
         };
         const calculate = (e) => {
          try {
            const sum = eval(text);
           setResult(sum);
           } catch (e) {
           if (e instanceof SyntaxError) {
              setResult("Error")
           }
         };
         const setDot = (e, text1) =>{
           let isSet = false
           text.split("").forEach((value, index, arr) =>{
             if (value === "."){
               isSet = false
             else if (isNaN(value)){
              isSet = true
               if (index === arr.length - 1){
                isSet = false
             }
           })
           if (isSet){
             setText(text+text1)
```



```
    Keyboard.js 
    ■ Button.js

Js App.js
src 🗦 ూ Keyboard.js 🗦 💋 Keyboard
          <View style={styles.container}>
            <View style={styles.col1}>
              <Button text1={"1"} setInfix={setInfix} />
             <Button text1={"4"} setInfix={setInfix} />
             <Button text1={"7"} setInfix={setInfix} />
              <Button text1={"."} setInfix={setDot} />
            </View>
            <View style={styles.col1}>
             <Button text1={"2"} setInfix={setInfix} />
              <Button text1={"5"} setInfix={setInfix} />
             Button text1={"8"} setInfix={setInfix} /
              <View style={styles.col1}>
             <Button text1={"3"} setInfix={setInfix} />
             <Button text1={"6"} setInfix={setInfix} />
             <Button text1={"9"} setInfix={setInfix} />
             <Button text1={"="} setInfix={calculate} />
            <View style={styles.col2}>
             <Button text1={"Del"} setInfix={deleteText} />
             <Button text1={"+"} setInfix={setInfix} />
              <Button text1={"-"} setInfix={setInfix} />
              <Button text1={"*"} setInfix={setInfix} />
              <Button text1={"/"} setInfix={setInfix} />
      const styles = StyleSheet.create({
        container: {
         flex: 3,
          flexDirection: "row",
         flex: 1,
          flexDirection: "column",
          backgroundColor: "#444",
          flex: 1,
          flexDirection: "column",
          backgroundColor: "#333",
      export default Keyboard;
```



```
Js Keyboard.js
                                Js Button.js X
Js App.js
src > Js Button.js > [@] Button
       import React from "react";
       import { StyleSheet, Text, View, TouchableOpacity } from "react-native";
  4 v const Button = ({ text1, setInfix }) => {
         return (
           <TouchableOpacity
             style={styles.container}
             onPress={(e) => setInfix(e, text1)}
             <Text style={{ fontSize: 32, color: "white" }}>{text1}</Text>
          </TouchableOpacity>
  11
         );
       };
       const styles = StyleSheet.create({
         container: {
           flex: 1,
  17
          alignItems: "center",
         justifyContent: "center",
        },
       });
       export default Button;
```