CSE118 Summer 2023

Assignment 8

In this assignment you will develop a simple React Native App based on components developed in Kotlin for Assignment 2 and Swift for Assignment 5 to enhance your understanding of:

- JavaScript
- React and JSX
- React Native Navigation
- Automated testing using Jest and React Native Testing Library

This assignment is worth 10% of your final grade.

Late submissions will not be graded.

Configuration

As for Assignment 7.

Setup

Download the starter code archive from Canvas and expand into an empty folder. I recommend creating a folder for the class and individual folders beneath that for each assignment.

- In Android Studio, start the Pixel 5 API 32 AVD
- In a terminal (command prompt on Windows) navigate to the installation folder and run the following command which will take a while to complete:

```
$ npm install
```

- Then run the following command to start the react-native development server:

```
$ npx react-native start
```

- Now open another terminal (command prompt on Windows) navigate to the installation folder again and enter the following command:

```
$ npx react-native run-android
```

  The app should appear in the Android Simulator, and you can start constructing your solution by adding JSX components in the `src` folder.

- To run the tests and check code coverage:

```
$ npm test
```

- To run check code quality:

```
$ npm run lint
```


Requirements

Basic:

As for Assignment 5. Study the Basic tests for this assignment to see what the accessibility labels for the navigation back buttons should be.

Advanced:

As for Assignment 5. Take care to honor the required accessibility labels and think carefully about how you will test messages just a few seconds, minutes, and hours old.

Stretch:

Your App should exhibit no linter errors and 100% class, method, line, and branch coverage when the provided Basic tests and your Advanced tests (if any) are executed.

What steps should you take to tackle this?

Use you understanding of what worked well and what didn't in Assignment 5 as the basis of the steps you will take here. Certainly, get the basic requirement out of the way quickly and submit to Canvas before starting work on the Advanced requirement.

How much code will you need to write?

A model solution that satisfies all requirements has approximately 350 lines of code, including tests for the Advanced requirement.

Grading scheme

The following aspects will be assessed:

1. (100%) **Does it work?**

    a. Basic (40%)
    b. Advanced (40%)
    c. Stretch (20%)

2. (-100%) **Did you give credit where credit is due?**

    a. Your submission is found to contain code segments copied from on-line resources or created by code generation tools and you failed to give clear and unambiguous credit to the original author(s) in your source code You will also be subject to the university academic misconduct procedure as stated in the class academic integrity policy. (-100%).

    b. Your submission is determined to be a copy of a past or present student's submission. (-100%)

    c. Your submission is found to contain code segments copied from on-line resources that you did give a clear an unambiguous credit to in your source code, but the copied code constitutes too significant a percentage of your submission:

        - < 25% copied code No deduction
        - 25% to 50% copied code (-50%)
        - > 50% copied code (-100%)

What to submit

On the console (PowerShell on Windows), navigate to the folder you extracted the starter code into and run the appropriate command to create the submission archive:

Windows:

```
$ Compress-Archive -Path App.js,src,__tests__ -DestinationPath Assignment8.Submission.zip
```

Linux:

```
$ zip -r Assignment8.Submission.zip App.js src __tests__
```

Mac:

```
$ zip -x "*.DS_Store" -r Assignment8.Submission.zip App.js src __tests__
```

**** UPLOAD** `Assignment8.Submssion.zip` **TO THE CANVAS ASSIGNMENT AND SUBMIT ****

