

Mobile Application Development Using React Native

M. Isham Azmansyah Faizal – 13514014
School of Electrical Engineering and Informatics
Institut Teknologi Bandung
Bandung, Indonesia
i@adimaja.com

Abstract—React Native is a framework for building mobile apps using React and JavaScript. It is created to help in building cross platform mobile applications, which are usually manpower-intensive and time-consuming. To try it, the author created a simple mobile app using React Native. The resulting app is finished in a very short time and with significantly less code than an equivalent native application.

Keywords—react native; react; redux; javascript; mobile; app.

I. INTRODUCTION

As computers become smaller and more portable, mobile applications are getting increasingly important. More companies decide to create mobile applications to suit their customers or users.

However, there are two major mobile application platforms; iOS and Android. Developing an app for both platforms is time-consuming and expensive. Both mobile platforms require massive time investments to learn and master the development tools and environments specific to each platform. As a result, developing an app for both platforms usually require a doubling of manpower.

React Native helps developers by providing its own platform for creating mobile apps. An app created using React Native can be run on both platforms, with minor modifications depending on the native features used.

In addition to this, React Native also makes it easy to write UIs in a declarative way, which greatly simplifies mobile application development. The declarative UI uses React, which has been used by web developers to create large, scalable, and high performance apps.

II. LITERATURE STUDY

A. React Native

React Native is a framework for creating mobile applications using React and JavaScript. This is done by using a special native app (customized for each platform) that can be controlled by JavaScript code. In this way, the app can have the performance of native applications, unlike other JavaScript-to-Android frameworks.

B. React

React is a library for creating declarative UIs. With React, the developers don't have to worry about changes to the data; they can simply assume that they're creating the entire UI from scratch. React will compute the differences between the old and

new UI and perform only the necessary changes to update the UI. This keeps React-based applications fast.

By itself, React doesn't have a way to render the UI, i.e. put the UI to a screen. It uses other modules, such as React DOM to draw to a web page, or React Native to draw it in a smartphone.

C. Redux

Redux is a library for managing application state. It uses a single object, called a *store*, to manage the entire state of an application. This state can include anything from data models such as to-do items to more low-level things like the item that is currently hovered over.

By storing the state in a single location, it can be more easily used and manipulated. Things like saving the entire app state or implementing undo and redo functions becomes simple.

The only way to modify this state is by dispatching objects called *actions* that describe what happened in the app, e.g. the user clicking a button. This ensures that data only flows in a single direction. This has several beneficial implications which makes application development easier.

III. METHODS AND TOOLS

In order to evaluate these tools and libraries, the author will create a simple mobile application using the previously mentioned tools and libraries.

To create the application the author uses the React Native framework, the Redux library, various extension libraries for Redux, and the Moment.js library.

While developing the application, the author uses the guides and resources available in the libraries' web sites.

IV. RESULTS

The author created an application to track a person's expenses. The application is simple, but it has several essential features that most real-world applications use such as data persistence, screen size flexibility, and screen rotation. The application's interface can be seen in Figure 1.

The application was developed in a short period of time (about 2 hours). This is the author's first React Native application; a more experienced developer would have finished it in significantly less time.

The application's source code is also very short. An examination using cloc (a tool for counting source code) shows that the application contains less than two hundred lines of code

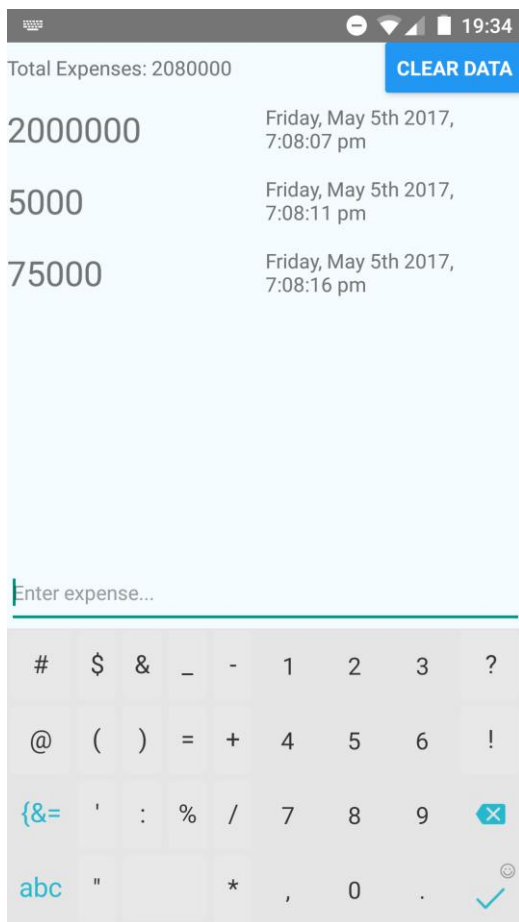


Figure 1 Screenshot of the expense tracking application

(not counting automatically generated code). By comparison, a single screen in the author's previous Android application using Java consists of four hundred lines.

By using React Native, the application doesn't have to handle the complex application lifecycle present in Android. A native Android app would have to implement various methods so that it would work correctly under various conditions, such as a turned off screen, the user temporarily going to the home screen, an incoming phone call, and many others. A React Native app handles them all automatically.

By using Redux as its state container, the application can easily store the user's data using React Native's own built in

storage system and a small persistence library. The data persistence configuration in this application uses less than five lines of code. In comparison, a native Android application would have to define a schema for the SQLite database and configure the SQLite interface, which can easily reach hundreds of lines of Java code.

The source code for this application can be found at <https://github.com/ishamf/expenses-tracker>.

V. CONCLUSION

Using React Native and various libraries that have been created for React, a developer can create great mobile apps with little effort.

In addition, an application created using React Native can be used on both Android and iOS, which significantly reduces the work needed to build cross-platform mobile applications.

REFERENCES

- [1] Faizal, I. A. (2017, May 5). *Source Code for Expenses Tracker*. Retrieved from GitHub: <https://github.com/ishamf/expenses-tracker>
- [2] React Developers. (2017, May 5). Retrieved from React Web Site: <https://facebook.github.io/react/>
- [3] React Native Developers. (2017, May 5). Retrieved from React Native Web Site: <https://facebook.github.io/react-native/>
- [4] Redux Developers. (2017, May 5). Retrieved from Redux Web Site: <http://redux.js.org/>

DECLARATION

I declare that this paper is written by myself, and is not an adaptation, translation, or plagiarism of another person's work.

Bandung, May 5, 2017

M. Isham Azmansyah Faizal

13514014