

# Creating a Mobile App: Android Studio vs React Native

By : *Taufic Leonardo Sutejo* / 13514022

*Program Studi Informatika*

*Sekolah Teknik Elektro dan Informatika Institut Teknologi Bandung*

*Jalan Ganesha 10 Bandung 40132, Indonesia*

[13514022@std.stei.itb.ac.id](mailto:13514022@std.stei.itb.ac.id) / [tauficleonardo@gmail.com](mailto:tauficleonardo@gmail.com)

**Abstract**—React Native is a framework that provide building mobile native application such as Android and IOS application using Javascript. It has the same architecture with web development. By comparing React Native and Android Studio, we have the knowledge of what case they are good at. To prove which are the best for what condition, the author use React Native advantages and disadvantages point as the comparison topics. Based on the exploration, React Native is the best in creating both platforms quick, simple UI and less using Native API.

**Keywords**—*React Native, framework, Javascript, Android Studio*

## I. INTRODUCTION

Nowadays, in society, technology is used in daily activities. One of the technology we are talking about is smartphone. Smartphone now not only provide telephone services, network services but also applications that keep company with our daily life. For example, social media, education, music, games, etc. They are all created by human to satisfy and help them while they do not have the technology that support it. What the author means is the technology for simple example calculator, we do not need to have real calculator accompany us every time when we need to count something. We just need to install the calculator app in our smartphone and we can use it anytime, anywhere we like to.

So, how does the human create the mobile application? The one who create mobile application is what they call themselves mobile developer. They use tools or framework to help them to create software. Tools to create mobile application for example Android Studio, Visual Studio, Xcode, etc. Framework to create mobile application for example, PhoneGap, React Native, Ionic, Accelerator Titanium, Mobile Angular UI, etc.

One of the most popular framework for creating mobile application is React Native. Before this framework exists, people still think how to create mobile application using the same method as web application without learning extra programming language such as java for Android and Swift or Objective-C for IOS. Moreover, they also think how to create mobile application in two platforms using one code only. Two years ago, React Native gave the solution to developer to develop one application once and deploy it on both IOS and Android.

Yes, it's true that not only React Native can provide what has been mentioned before. There are other frameworks can provide it such as Xamarin, Secha Touch, Accelerator Titanium, etc. But in this paper, the writer will discuss more about how the advantages and disadvantages by using React Native that provide two platforms using one code compare to Android Studio.

This paper is organized into five section. The first section is Introduction. The second section will explain basic theory of React Native and Android Studio and pros and cons about React Native. The third section will more about the exploration creating a single application. The forth section will describe the comparison and discuss the result. Finally, in last section we give conclusion.

## II. LITERATURE STUDY

### A. *React Native*

React Native is an open-sourced framework to develop mobile application for IOS and coming soon for Android. It's created by Facebook. Still, it's under development and possible to have bugs or not supported/implemented, React Native has the similar implementation to web development. At this point, web developers can now easily to create mobile application without learning more programming language because they have already known about HTML, CSS, Javascript. Moreover, you just need one code to develop both Android and IOS at the same time.

React Native applications are written in JSX (Javascript and XML) and itself will render the code for both Android and IOS. By using text editor, chrome debugger, testing tools, and any other tools are enough to write React Native code. The API depends on the Operating System. Of course, the user interface viewed in your mobile screen is not web component but it's real mobile component. React Native defines its user interface by breaking down into several components. Besides Javascript and XML, as mobile developer still need to know about HTML, CSS, and Object Oriented Programming while developing a mobile application.

Using React Native to develop mobile application, there are advantages and disadvantages. Below are the briefs explanation about React Native advantages.

### 1. CSS-like styling

Styling user interface of the mobile application is the same as styling in web application. Layout in React Native uses Flexbox provide ways to position each component in the mobile screen.

### 2. Reusability

React Native gives the possibility to reuse the UI code if it's necessary. It means whenever there is common UI code, it can be shared together with the same logic.

### 3. Learn Once, Write Everywhere

By learning Javascript once, we can write in any platform especially React Native support Android and IOS platform. We do not need to learn Java for Android and XCode for IOS.

### 4. Declarative API

The UI become more predictable and easier to understand because we do not need to keep track all the changes when the application has different states.

### 5. Hot Reloading

Usually when the mobile developers change the code, they need to recompile and install the app and it takes too long, but React Native is different. By refreshing the emulator or the mobile, it will be refreshed the changes just like refreshing web page.

Below are the briefs explanation about React Native disadvantages.

#### 1. Type Error

Javascript does not have type in the variable and parameters to function.

#### 2. Create for IOS in the first place.

It was created for IOS in the first place and Android a bit late supported. The documentation of android still improving.



Figure 1 React Native Logo

Source: <http://www.appcoda.com/react-native-introduction/>

### B. Android Studio

Android Studio is an IDE official from google to develop application for Android platform. The application can run not only in mobile but also in desktop and other technology that run on Android. The other technology for example TV, Wearable Technology, Glass, etc. Android Studio provide good user interface while developing application and there are templates gave new developers

first experience developing Android application. The tool also provides the continuous integration allows the developers to test their code and immediately reports if there are problems or errors occur. Android Studio provide solution in Java or C++.

In this paper, the author does not mention advantages and disadvantages of Android Studio because it's more focus in React Native

### III. EXPLORING REACT NATIVE

The Author creates simple calculator application. At the very beginning, the author designs the calculator. The calculator has two layers. One is for display layer, other is for the button layer. The way of dividing the layer using flexbox that have been mentioned before in second section CSS-like styling. Next, create one component of the button then the component iterates as the view below.



Figure 2 Simple calculator

At last, the author gives the button touch handler. The button handler is created for changing the text display and doing the arithmetic solve. The button handler updating the text display using State. State usually need to be initialized in the constructor and then call the setState when the button is touched.

```
constructor(props) {
  super(props);
  this.state = {
    inputValue: 0,
    prevInputValue: 0,
    selectedSymbol: null
  }
}
```

Figure 3 State initialize in the constructor

```
this.setState({
  inputValue: inputValue
})
```

Figure 4 setState when the button is touched

#### IV. RESULT

The author has performed some exploration and now we can see how the comparison between React Native and Android Studio from the advantages and disadvantages of React Native. First is about the CSS-like styling, React Native way of styling have the same way with web development. On the other hand, Android Studio using XML for styling or drag and drop component.

```
var styles = StyleSheet.create({
  rootContainer: {
    flex: 1
  },
});
```

Figure 5 CSS styling

Second is about Reusability, when you have defined the component, the component can be used several times. Moreover, the component can be iterated or manipulated. But Android Studio to define 16 Button components, you need to create 16 Button UI.

```
for (var i = 0; i < inputButtons.length; i++) {
  let row = inputButtons[i];

  let inputRow = [];
  for (var j = 0; j < row.length; j++) {
    let input = row[j];
    inputRow.push(
      <InputButton value={input}
        highlight={this.state.selectedSymbol === input}
        onPress={this._onButtonPressed.bind(this, input)}
        key={i + "-" + j} />
    );
  }
}
```

Figure 6 Iteration the input component

Third is Learn Once, Write Everywhere, you only need to learn Javascript for React Native. Meanwhile, Android Studio you need to learn Java and you can only create for Android not for both Android and IOS.

Forth is Declarative API, you just need to change the component source code and automatically do to others with the same use. Different from Android Studio, you need to change one by one.

Fifth is Hot Reloading, React Native you just need to double press R and it will refresh the screen view, but Android Studio is different. It will build the project from the beginning and install it to the emulator or to the device each time you compile the project. Meanwhile, React Native just need to compile the project once.

```
BUILD SUCCESSFUL
Total time: 22.39 secs
```

Figure 7 time spend building the project for React Native

The disadvantages of the React Native cannot be proved due to uncertainty about them. So, the author will skip this part.

#### V. CONCLUSION

By knowing the advantages and disadvantages, the writer does not want to choose whether React Native, others framework or cross-platform development and Native development tools could help better when developing an application. It all depends on what kind of situation. For example, React Native let you to reuse the component with the same functionality easily than Android Studio. If you would like to get an application in both IOS and android quick, have the basic of React and Javascript, do not need extra native API that's not provided by React Native and any sophisticated UI. You may choose React Native otherwise, you must go with the Native development tools such as Android Studio or XCode.

#### ACKNOWLEDGMENT

Taufic Leonardo Sutejo as the author of this paper would like to say thanks to his parents that always support him, thanks to his both sisters that always give advice, thanks to his friends that always there when he needs the most. I want to thank to Mr. Rinaldi Munir, Mrs. Ayu Purwarianti, and Mrs. Dessy Puji Lestari as the lectures of IF3280 – “Socio Informatics and Professionalism”. Because of them I can finish this paper.

#### REFERENCES

- [1] Nakamurakare. (2017). *App Development Using React Native vs Android Studio/IOS*, Retrieved May 2, 2017, from <https://www.packtpub.com/books/content/app-development-using-react-native-vs-androidios>
- [2] Narayan. (2017). *React Native vs Native IOS/Android Studio*, Retrieved May 3, 2017, from <https://www.coursereport.com/blog/so-you-want-to-build-a-mobile-app-react-native-vs-native-mobile>
- [3] Eisenman. (2017). *Learning React Native*, Retrieved May 2, 2017, from <https://www.safaribooksonline.com/library/view/learning-react-native/9781491929049/ch01.html>
- [4] Hughes. *An Introduction to React Native*, Retrieved May 4, 2017, from <http://authenticff.com/journal/intro-to-react-native>
- [5] David. (2015). *Learn More about the Android Studio IDE from Google*, Retrieved May 4, 2017 from <http://searchsoftwarequality.techtarget.com/feature/Learn-more-about-the-Android-Studio-IDE-from-Google>

## STATEMENT

With this I declare that my paper I wrote by myself, not taken or translate from others paper and it's not plagiarism.

Bandung , 5<sup>th</sup> Mei 2017

A handwritten signature in black ink, appearing to be 'Taufic Leonardo Sutejo', written in a cursive style.

Taufic Leonardo Sutejo