

**LAPORAN PRAKTIKUM  
PEMROGRAMAN WEB & MOBILE I**



**NAMA : EVAN ALPHARIO IMANUEL**  
**NIM : 193030503059**  
**KELAS : A**  
**MODUL : VI ( React Native )**

**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PALANGKA RAYA**  
**2021**

# BAB I

## LANDASAN TEORI

### 1.1 Landasan Teori

#### 1.1.1 React Native

React Native adalah *framework mobile app development* yang memungkinkan pengembangan aplikasi secara multi-*platform* yaitu Android dan iOS. Sederhananya, React Native adalah *framework* yang memungkinkan *developer* membuat aplikasi untuk berbagai *platform* dengan menggunakan basis kode yang sama.

Penggabungan antara *native mobile app* dan React juga bisa dilakukan dengan mudah. Jadi, pengembang bisa membuat aplikasi Android dan iOS dengan lebih cepat.

React Native sendiri pertama kali dirilis pada tahun 2015 oleh Facebook dan menjadi bagian dari proyek *open source* mereka.

Kemudian, hanya dalam beberapa tahun saja, React Native telah menjadi andalan bagi banyak *developer* untuk mengembangkan aplikasi seluler.

Beberapa contoh aplikasi seluler yang menggunakan *framework* React Native adalah Instagram, Facebook, Pinterest, hingga Skype.

Ada beberapa alasan lain mengapa *React Native* begitu populer, antara lain:

- *Developer* hanya perlu membuat satu kode untuk mengembangkan aplikasi berbasis Android atau iOS. Jadi, *framework* ini benar-benar menghemat waktu.

- React Native adalah *framework* yang dibangun berdasarkan React, yaitu *library* dari JavaScript yang sudah populer.
- *Framework* ini mendorong *frontend developer* yang sebelumnya hanya dapat bekerja dengan teknologi berbasis web saja. Kini, mereka bisa dengan mudah untuk mengembangkan *mobile platform*.

Selain beberapa alasan di atas, *React Native* juga memiliki keunggulan lain yaitu tidak akan *me-render webviews* dalam kodenya. Jadi program akan dijalankan sesuai dengan tampilan dan komponen dari *native* yang awalnya digunakan.

### 1.1.2 Cara Kerja React Native

Setelah mengetahui pengertian dan beberapa alasan mengapa React Native begitu populer. Kini, coba ketahui dulu bagaimana kerjanya.

*React Native* adalah *framework* yang ditulis dengan campuran bahasa JavaScript dan JXL, sebuah kode *markup* khusus yang menyerupai XML.

*Framework* ini memiliki kemampuan untuk berkomunikasi dengan kedua ranah sekaligus, yaitu *threads* yang berbasis JavaScript dan *threads* dari *native app*.

Jadi, bagaimana caranya untuk berkomunikasi? Rupanya *React Native* menggunakan apa yang disebut dengan “*bridge*” atau jembatan.

JavaScript dan *threads native* memang ditulis dengan dua bahasa pemrograman yang berbeda. Namun, fitur *bridging* dari React Native tetap memungkinkan untuk komunikasi dua arah.

Itu artinya saat pengembang sudah memiliki aplikasi Android atau iOS, mereka masih tetap bisa menggunakan komponennya saat menggunakan React Native.

### **1.1.3 Kelebihan React Native**

#### **1. Digunakan oleh Perusahaan-Perusahaan Besar**

Meskipun tergolong *framework* baru, React Native tercatat telah digunakan oleh sejumlah perusahaan besar dalam membuat aplikasi. Sebut saja aplikasi Facebook, Instagram, Airbnb, dan Baidu. Nyatanya, React Native memang dikembangkan oleh tim Facebook untuk membuat aplikasi *mobile*.

#### **2. Kode yang digunakan lebih simpel**

React Native lebih mudah dipelajari daripada ‘Native’. Hal ini dikarenakan kode yang digunakan lebih simpel. Untuk Anda yang terbiasa dengan Javascript, menggunakan HTML, dan sejenisnya tentu akan merasa sangat familiar ketika dihadapkan dengan React Native.

Selain kode yang digunakan lebih simpel, Anda tidak perlu lagi menguasai bahasa pemrograman lainnya demi bisa membuat aplikasi yang bisa berjalan di dua sistem operasi sekaligus.

#### **3. Cocok untuk startup**

Untuk perusahaan yang baru merintis, atau biasa disebut dengan *start up*, dan ingin mengembangkan aplikasi *mobile*, React Native bisa menjadi pilihan *framework* yang tepat. Kenapa?

Karena perusahaan hanya perlu merekrut satu orang *developer* yang menguasai React Native dan bisa menghasilkan aplikasi yang bisa berjalan di dua sistem operasi sekaligus. Sayangnya, masih jarang *developer* yang menguasai *framework* ini.

#### **4. Lebih hemat**

Karena bisa merekrut satu orang untuk dua tugas sekaligus, perusahaan akan hemat dalam menggaji karyawan. Anggaran yang seharusnya digunakan untuk menggaji dua *developer* bisa dialihkan untuk kebutuhan lainnya yang tak kalah penting.

Misalnya seperti untuk konten, marketing, atau produk activation yang tentu bakal membutuhkan banyak sumber daya.

### **1.1.4 Kekurangan React Native**

#### **1. JavaScript yang Berkembang Sangat Cepat**

JavaScript menjadi kelemahan dari React Native karena javascript berkembang begitu cepat sehingga efeknya menimbulkan beberapa hal tidak bisa dilakukan sesuai harapan.

#### **2. Framework**

Sama seperti JavaScript, framework React Native juga menjadi kelemahan tersendiri karena perubahan yang begitu cepat terjadi.

#### **3. Platform**

Sebagaimana kita ketahui, untuk bisa mengembangkan aplikasi mobile berbasis Ios maka developer juga harus menggunakan perangkat MAC OS karena sistem operasi ini eksklusif hanya tersedia di Mac, maka setidaknya Anda harus menyediakan Macbook.

#### **4. Performance**

Manusia memang tidak akan pernah puas soal performa, sehingga saat ini masih ada perdebatan terkait kinerja React Native secara umum di kalangan developer. Nah, bagi Anda yang memiliki pendapat kurang tentu hal ini menjadi poin negatif soal kekurangan React Native.

#### **1.1.5 Android Studio**

Android Studio adalah Integrated Development Environment (IDE) resmi untuk pengembangan aplikasi Android, yang didasarkan pada IntelliJ IDEA. Selain sebagai editor kode dan fitur developer IntelliJ yang andal, Android Studio menawarkan banyak fitur yang meningkatkan produktivitas Anda dalam membuat aplikasi Android, seperti:

- Sistem build berbasis Gradle yang fleksibel
- Emulator yang cepat dan kaya fitur
- Lingkungan terpadu tempat Anda bisa mengembangkan aplikasi untuk semua perangkat Android
- Terapkan Perubahan untuk melakukan push pada perubahan kode dan resource ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi
- Template kode dan integrasi GitHub untuk membantu Anda membuat fitur aplikasi umum dan mengimpor kode sampel
- Framework dan alat pengujian yang lengkap

- Alat lint untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya
- Dukungan C++ dan NDK
- Dukungan bawaan untuk Google Cloud Platform, yang memudahkan integrasi Google Cloud Messaging dan App Engine

## BAB II

### PEMBAHASAN

Pada praktikum modul 2 ini kita diminta untuk membuat program aplikasi android atau mobile yang dimana dalam program tersebut terdapat program search menggunakan flatlist react-native.

Dari program aplikasi yang terdapat dari modul sebelumnya kita dapat menggunakan program tersebut dan melakukan perubahan pada file App.js, kemudian menambahkan file baru yang bernama nama.json.

Sebelum masuk ke file app.js dan nama.json kita dapat menginstal ekstensi pada folder yang kita gunakan sebelumnya dengan menggunakan CMD ,berikut adalah perintahnya :

```
npm install native-base
```

Setelah selesai menginstal native base maka kita dapat melakukan pengeditan pada app.js yang dimana kita dapat merubah seluruh isinya menjadi berikut :

```
import React, {Component} from 'react';
import{
  Container,
  Header,
  Content,
  Left,
  Right,
  Body,
  Icon,
  Text,
  ListItem,
  Item,
  Input,
} from 'native-base';
let helperArray = require('./nama.json');
export default class App extends Component {
  constructor(props){
    super(props);
    this.state = {
      allUsers: helperArray,
      usersFiltered:helperArray,
    };
  }
}
```



```

searchUser(textToSearch) {
  this.setState({
    usersFiltered: this.state.allUsers.filter(i =>
      i.name.toLowerCase().includes(textToSearch.toLowerCase()),
    ),
  });
}

render() {
  return (
    <Container>
      <Header searchBar rounded>
        <Item>
          <Icon name="search"/>
          <Input
            placeholder="Search User"
            onChangeText={text => {
              this.searchUser(text);
            }}
          />
        </Item>
      </Header>
      <Content>
        {this.state.usersFiltered.map((item,index)=>(
          <ListItem>
            <Left></Left>
            <Body>
              <Text>{item.name}</Text>
              <Text>{item.address}</Text>
            </Body>
            <Right />
          </ListItem>
        ))}
      </Content>
    </Container>
  );
}
}

```

Pada bagian pertama baris program terdapat fungsi untuk melakukan pengimportan file yang dimana di import dari 'from native-base'; yang dimana dilanjutkan dengan perintah dibawahnya yang dimana let helpArray= require ('./nama.json'); yang dimana dari program tersebut terdapat pemanggilan dari import yang dimana yang akan ditampilkan dalam program menggunakan file bernama nama.json.

Yang dimana file dari name.json tersebut berisikan data berikut :

**nama.json**

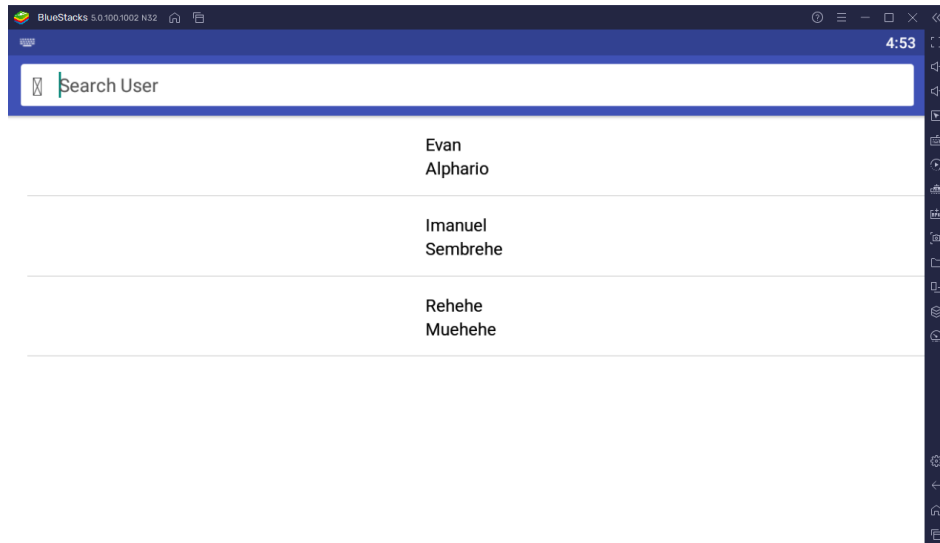
```
[
{
  "name": "Evan",
  "address": "Alphario"
},
{
  "name": "Imanuel",
  "address": "Sembrehe"
},
{
  "name": "Rehehe",
  "address": "Muehehe"
},
]
```

Dari program diatas merupakan nama dan alamat yang akan ditampilkan pada saat program dijalankan.

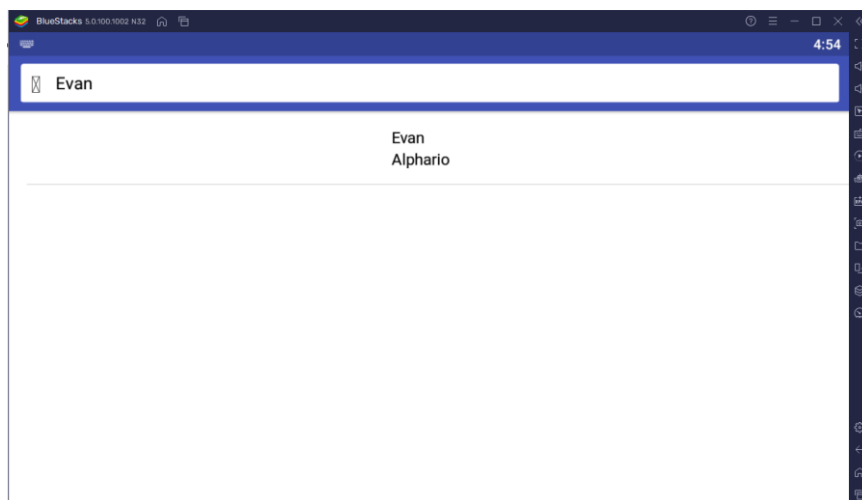
Selanjutnya pada bagian baris di bawahnya terdapat perintah untuk melakukan search dan menampilkan nya didalam aplikasi, yang mana fungsi tersebut akan digunakan untuk mencari nama berdasarkan kata yang diinputkan pada kolom pencarian. Fungsi searchUser ini menggunakan filter, yang dimana akan melakukan penyaringan atau seleksi sesuai dari kata yang diinputkan. Lalu terdapat toLowerCase() dimana digunakan untuk merubah nilai string ke nilai string yang terdiri dari huruf kecil semua.

Kemudian pada bagian yang terakhir adalah merupakan fungsi terakhir yaitu fungsi terakhir yang dimana dimana fungsi ini berisi perintah-perintah untuk menampilkan data-data dalam bentuk list. Yang Dimana data-data tersebut merupakan data yang terdapat pada variabel helperArray.

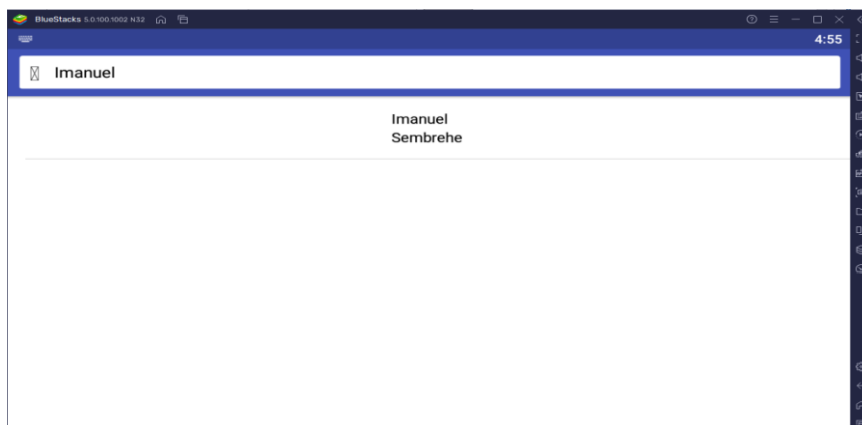
Setelah selesai dengan program diatas program dapat dijalankan yang dimana berikut adalah tampilan dari aplikasi nya :



**Gambar 2.1** Tampilan Aplikasi Search user Modul6



**Gambar 2.2** Tampilan Aplikasi menggunakan fungsi Search



**Gambar 2.3** Tampilan Aplikasi menggunakan fungsi Search

### **BAB III**

### **KESIMPULAN**

Dari praktikum kali ini dapat disimpulkan bahwa dengan adanya search flatlist kita dapat membuat suatu aplikasi yang memerlukan fungsi dari sebuah pencarian yang akan sangat membantu pengguna aplikasi tersebut sehingga membuat aplikasi lebih efisien.

## DAFTAR PUSTAKA

Javatpoint. (n.d.). *React Native FlatList*.

<https://www.javatpoint.com/react-native-flatlist>

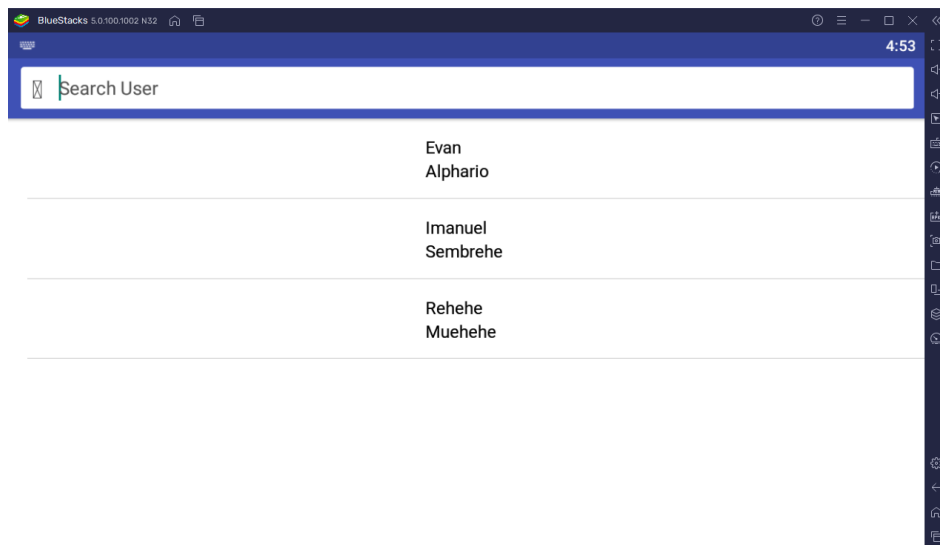
React-native. (2020). *Basic React Native Part 25 - React Native Flatlist*.

<https://cloudborneo.com/basic-react-native-part-25-react-native-flatlist/>

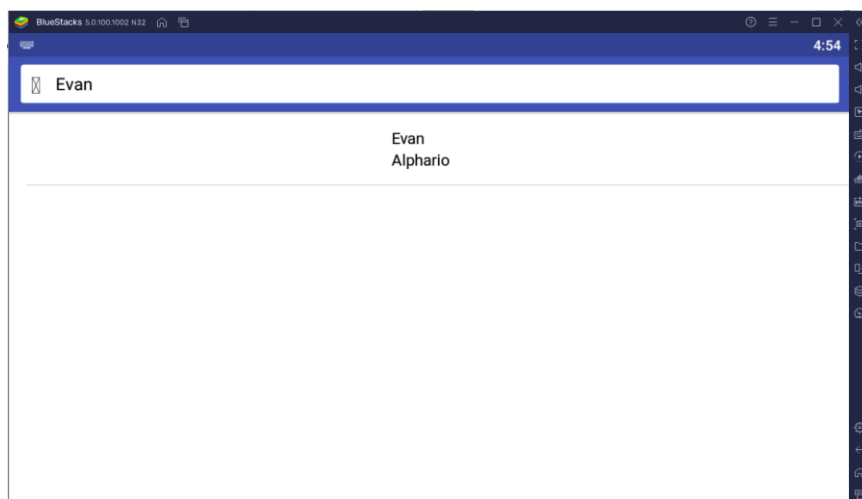
Ismi, T. (2021). *React Native: Ketahui Pengertian serta Kelebihan dan Kekurangannya*.

<https://glints.com/id/lowongan/react-native-adalah/#.YHOqeT8RXDc>

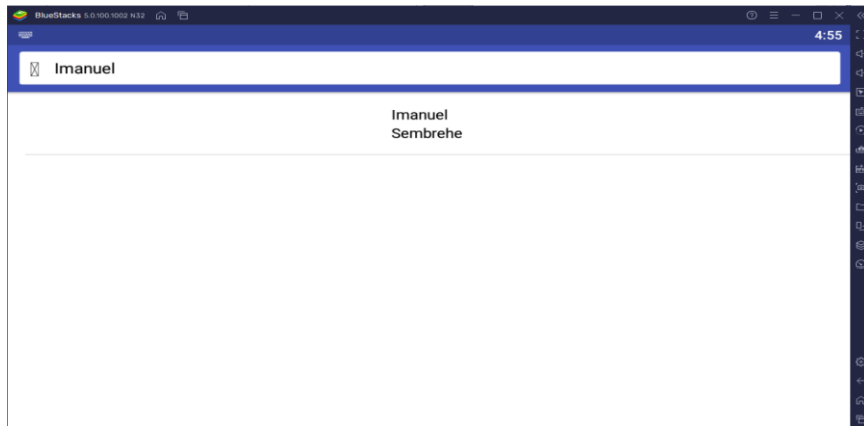
## LAMPIRAN



**Gambar 2.1** Tampilan Aplikasi Search user Modul6



**Gambar 2.2** Tampilan Aplikasi menggunakan fungsi Search



**Gambar 2.3** Tampilan Aplikasi menggunakan fungsi Search