

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN WEB DAN MOBILE I**



**NAMA : Muhammad Fauzan**

**NIM : 193020503024**

**KELAS : A**

**MODUL : VI (Flatlist)**

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

## **BAB I**

### **TUJUAN DAN LANDASAN TEORI**

#### **1.1. Tujuan**

**1.1.1.** Mahasiswa mampu melakukan instalasi react native

#### **1.2. Landasan Teori**

##### **1.2.1. React Native**

React Native adalah framework yang digunakan untuk membuat mobile app di dua sistem operasi sekaligus, yaitu Android dan iOS. Untuk itulah, React Native disebut juga dengan cross-platform network karena Anda bisa membuat satu aplikasi yang bisa digunakan di berbagai platform, yaitu Android dan iOS.

Dalam membuat sebuah aplikasi, Anda perlu belajar bahasa pemrograman yang spesifik atau khusus (disebut juga dengan ‘Native’) untuk platform tersebut. Hal ini dikarenakan setiap sistem operasi memiliki bahasa pemrograman yang berbeda-beda.

Untuk membuat aplikasi di Android, Anda butuh belajar bahasa pemrograman Java. Sedangkan untuk membuat aplikasi di iOS, Anda perlu belajar menggunakan bahasa pemrograman Objective C atau Swift. Namun dengan menggunakan React Native sebagai framework, Anda tidak perlu lagi melakukan hal spesifik tersebut.

Alih-alih menggunakan Java atau Objective C maupun Swift, Anda hanya butuh belajar bahasa pemrograman Javascript untuk bisa membuat aplikasi dengan menggunakan platform ini. Javascript sendiri biasa digunakan oleh para developer dalam membuat website.

Jadi, dengan menggunakan platform ini, Anda hanya perlu membuat satu aplikasi. Anda tidak perlu lagi membuat banyak versi aplikasi yang compatible dengan sistem operasi yang ada. (Andy, 2020)

### **1.2.2. 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

### **1.2.3. Flatlist**

React Native Flatlist merupakan komponen-komponen sederhana untuk menampilkan list daftar data namun banyak digunakan, contoh flatlist yang akan membantu anda untuk memahami bagaimana menggunakannya.

FlatList menampilkan data terstruktur yang serupa dalam daftar yang dapat di-scroll. Ini berfungsi dengan baik untuk daftar data yang

besar di mana jumlah item daftar dapat berubah seiring waktu. FlatList hanya menampilkan elemen render yang saat ini ditampilkan di layar, tidak semua elemen daftar sekaligus.

Komponen FlatList membutuhkan dua props yang diperlukan: data dan renderItem. Data adalah sumber elemen untuk daftar, dan renderItem mengambil satu item dari sumber dan mengembalikan komponen yang diformat untuk dirender.

## BAB II

### PEMBAHASAN

#### Tugas Praktikum

1. Membuat Flatlist

#### Pembahasan

Pada praktikum ini praktikan diminta untuk membuat sebuah program aplikasi android, yang dimana didalam program tersebut dapat melakukan search menggunakan flatlist react-native. Pertama, buka cmd terlebih dahulu lalu buat projectnya dengan menggunakan perintah cmd.

```
React-native init MuhammadFauzan
```

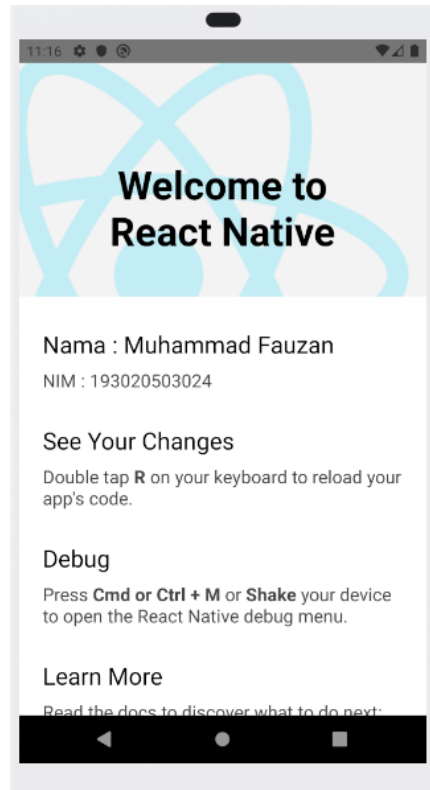
Tunggu syntax instalasi program diatas selesai. Setelah selesai, maka akan ada foilder baru dengan nama MuhammadFauzan lalu cd in ke dalam folder tersebut dengan syntax *“cd MuhammadFauzan”* maka kita sudah berada didalam folder project dengan nama MuhamamdFauzan. Setelah itu, install native base dengan syntax berikut.

```
Npm install native-base
```

Tunggu setelah syntax diatas selesai lalu jalan kan syntax seperti dibawah ini.

```
React-native run-android
```

Syntax diatas digunakan untuk menjalankan suatu system android dengan menggunakan emulator sebagai debugging atau device android sebagai debugging. Setelah syntax diatas berhasil dijalankan maka program android akan berjalan seperti gambar dibawah ini.



**Gambar 2.1.** React Native Phone

Setelah itu buka folder project yang telah dibuat dengan menggunakan code editor untuk membuat flatlist pada app.js menjadi seperti yang ada dibawah ini.

```
1  /**
2   * Sample React Native App
3   * https://github.com/facebook/react-native
4   *
5   * @format
6   * @flow strict-local
7   */
8
9  import React, {Component} from 'react';
10 import {
11   Container,
12   Header,
13   Content,
14   Left,
15   Right,
16   Body,
17   Icon,
18   Text,
19   ListItem,
20   Item,
```

**Gambar 2.2.** App.js part1

```

21 Input,
22 } from 'native-base';
23 let helperArray = require('./nama.json');
24 export default class App extends Component {
25   constructor(props){
26     super(props);
27     this.state = {
28       allUsers: helperArray,
29       usersFiltered:helperArray,
30     };
31   }
32
33   searchUser(textToSearch) {
34     this.setState({
35       usersFiltered: this.state.allUsers.filter(i =>
36         i.username.toLowerCase().includes(textToSearch.toLowerCase()),
37       ),
38     });
39   }
40
41 /**
42  * Sample React Native App
43  * https://github.com/facebook/react-native
44  *
45  * @format
46  * @flow strict-local
47  */
48
49 import React, {Component} from 'react';
50 import{
51   Container,
52   Header,
53   Content,
54   Left,
55   Right,
56   Body,
57   Icon,
58   Text,
59   ListItem,
60   Item,
61   Input,
62 } from 'native-base';
63 let helperArray = require('./nama.json');
64 export default class App extends Component {
65   constructor(props){
66     super(props);
67     this.state = {
68       allUsers: helperArray,
69       usersFiltered:helperArray,
70     };
71   }
72
73   searchUser(textToSearch) {
74     this.setState({
75       usersFiltered: this.state.allUsers.filter(i =>
76         i.username.toLowerCase().includes(textToSearch.toLowerCase()),
77       ),
78     });
79   }
80

```

**Gambar 2.3.** App.js part2

Dapat dilihat pada gambar diatas terdapat bagian import yang digunakan untuk dapat menggunakan komponen - komponen yang terdapat pada library react-native dan juga library native-base.

Lalu terdapat bagian deklarasi variabel yang dimana variabel tersebut membutuhkan sebuah file untuk dapat dihubungkan, Yaitu pada syntax *require*. Syntax tersebut digunakan untuk menghubungkannya dengan file bernama nama.json yang merupakan file yang berisikan nama – nama yang akan dibuat menjadi program flatlist.

Terdapat fungsi searching atau pencarian yang terdapat pada file app.js ini yang berfungsi mencari berdasarkan username yang tersedia dalam nama.json .

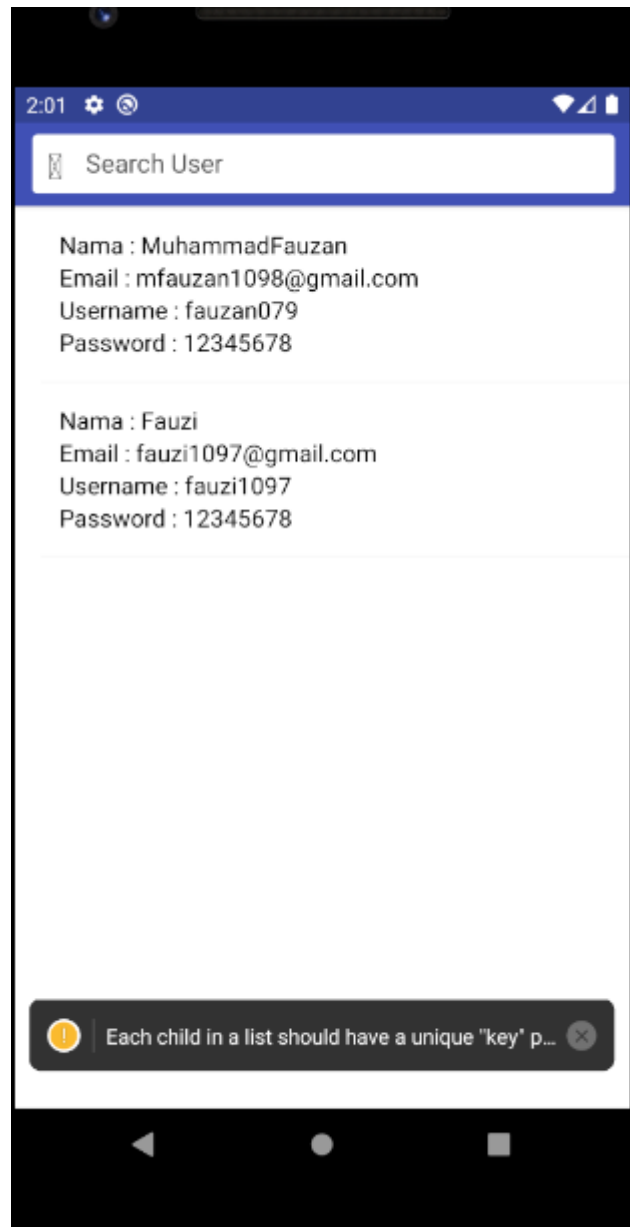
Dan yang terakhir terdapat bagian fungsi render yang mempunyai fungsi untuk menampilkan data – data yang ada pada file nama.json menjadi berbentuk seperti list. Dan untuk data yang dapat dicari adalah data yang terdapat pada nama.json, berikut ini adalah isi dari file nama.json

```
1  ✓ [
2  ✓   {
3      "name": "Muhammad Fauzan",
4      "email": "mfauzan1098@gmail.com",
5      "username": "fauzan079",
6      "password": "12345678"
7   },
8
9  ✓   [
10      "name": "Fauzi",
11      "email": "fauzi1097@gmail.com",
12      "username": "fauzi1097",
13      "password": "12345678"
14   ]
15 ]
```

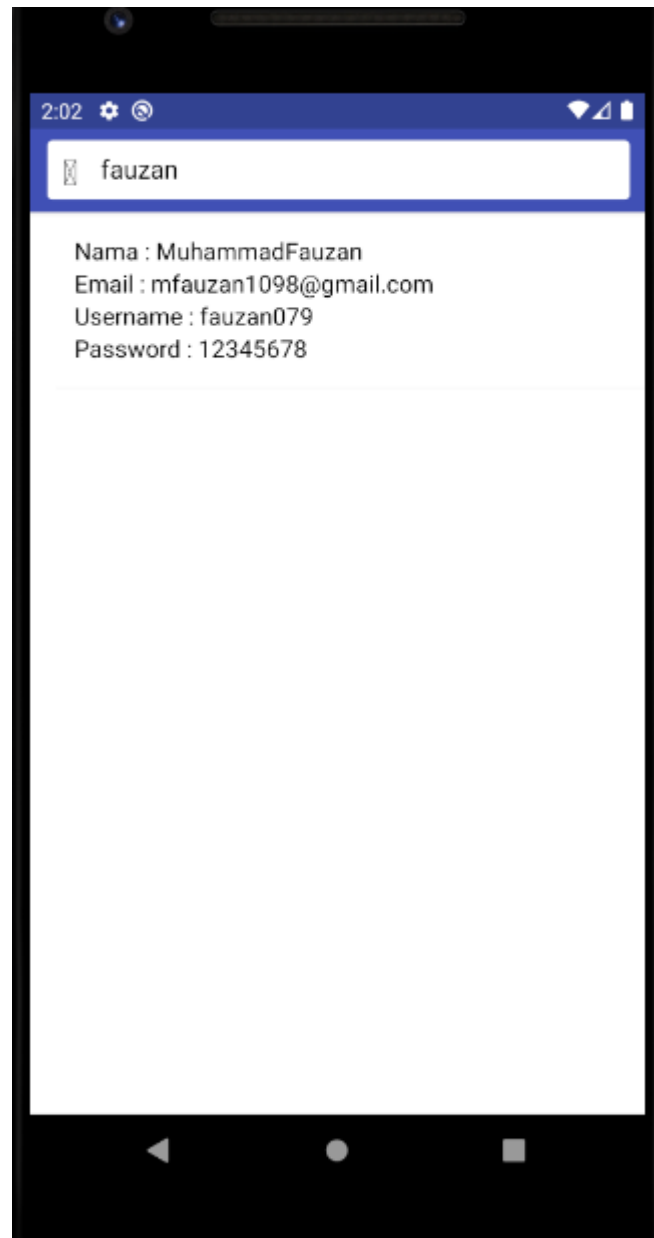
**Gambar 2.4.** Nama.json

Setelah mempunyai data pada nama.json berikut nya adalah melakukan pencarian dalam aplikasi yang telah dibuat berikut ini adalah tampilan aplikasi yang dibuat dan yang sudah menggunakan fungsi search.





**Gambar 2.5.** Flatlist



**Gambar 2.6.** Search Flatlist

### **BAB III**

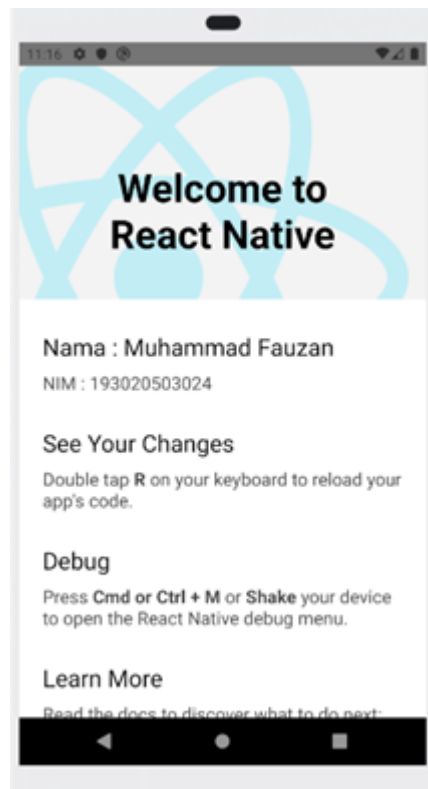
#### **KESIMPULAN**

Dari praktikum ini dapat disimpulkan bahwa, React native merupakan sebuah framework cross-platform yang memiliki basis bahasa pemrograman JavaScript. React native memungkinkan developer membuat mobile apps yang bisa dijalankan di dalam dua sistem operasi mobile yaitu android dan iOS. Dan search flatlist pada react native, sangat membantu dalam membuat sebuah program yang dimana program tersebut memerlukan sebuah fitur pencarian di dalamnya sehingga pengguna menjadi lebih nyaman dalam menggunakan program yang di dalamnya terdapat fitur search.

## DAFTAR PUSTAKA

- Andy. (2020, Maret 8). *Apa itu React Native dan Kenapa Cocok Digunakan untuk Perusahaan Anda?* - Qwords. <https://qwords.com/blog/apa-itu-react-native/>
- Rasupe. (2019). *Cara install react native di windows 10* – Rasupe. <https://rasupe.com/cara-install-react-native-di-windows-10/>

## LAMPIRAN



Gambar 2.1. React Native Phone

```
1  /**
2   * Sample React Native App
3   * https://github.com/facebook/react-native
4   *
5   * @format
6   * @flow strict-local
7   */
8
9  import React, {Component} from 'react';
10 import {
11   Container,
12   Header,
13   Content,
14   Left,
15   Right,
16   Body,
17   Icon,
18   Text,
19   ListItem,
20   Item,
```

Gambar 2.2. App.js part1

```

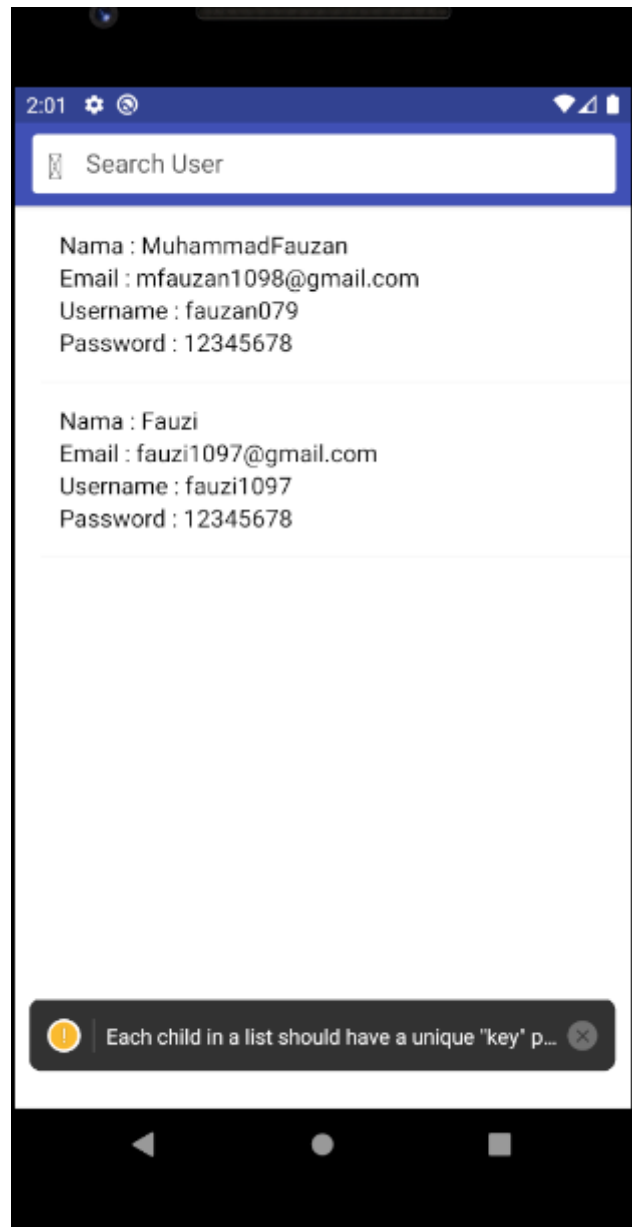
21   Input,
22   } from 'native-base';
23   let helperArray = require('./nama.json');
24   export default class App extends Component {
25     constructor(props){
26       super(props);
27       this.state = {
28         allUsers: helperArray,
29         usersFiltered:helperArray,
30       };
31     }
32
33     searchUser(textToSearch) {
34       this.setState({
35         usersFiltered: this.state.allUsers.filter(i =>
36           i.username.toLowerCase().includes(textToSearch.toLowerCase()),
37         ),
38       });
39     }
40
41 /**
42  * Sample React Native App
43  * https://github.com/facebook/react-native
44  *
45  * @format
46  * @flow strict-local
47  */
48
49 import React, {Component} from 'react';
50 import {
51   Container,
52   Header,
53   Content,
54   Left,
55   Right,
56   Body,
57   Icon,
58   Text,
59   ListItem,
60   Item,
61   Input,
62 } from 'native-base';
63 let helperArray = require('./nama.json');
64 export default class App extends Component {
65   constructor(props){
66     super(props);
67     this.state = {
68       allUsers: helperArray,
69       usersFiltered:helperArray,
70     };
71   }
72
73   searchUser(textToSearch) {
74     this.setState({
75       usersFiltered: this.state.allUsers.filter(i =>
76         i.username.toLowerCase().includes(textToSearch.toLowerCase()),
77       ),
78     });
79   }
80

```

Gambar 2.3. App.js part2

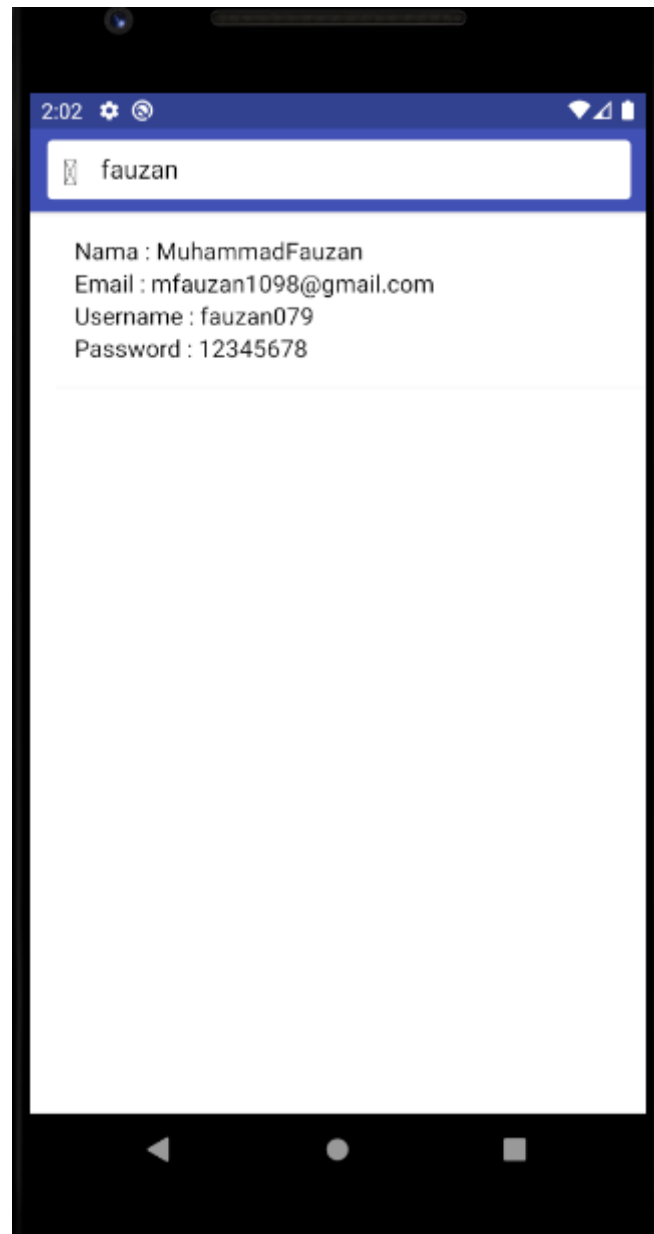
```
1  ✓ [
2  ✓   {
3      "name": "MuhammadFauzan",
4      "email": "mfauzan1098@gmail.com",
5      "username": "fauzan079",
6      "password": "12345678"
7   },
8
9  ✓   {
10      "name": "Fauzi",
11      "email": "fauzi1097@gmail.com",
12      "username": "fauzi1097",
13      "password": "12345678"
14   }
15 ]
```

**Gambar 2.4.** Nama.json



**Gambar 2.5.** Flatlist





**Gambar 2.6.** Search Flatlist