

## CHAPTER 4

### ANALYSIS AND DESIGN

#### 4.1 Analysis

The purpose of this project is to determine and analyze the method of processing a large data in React Native. The data is gained from Banjar & Seririt market traders in Bali.

The methods we will use here are ScrollView and FlatList which are React Native methods used to process data.

##### 1. ScrollView

ScrollView is a method used to process data that instantly renders all components resulting in decreased performance if the data is loaded on a large scale.

How to use ScrollView:

```
import { Text, ScrollView } from 'react-native';
```

```
<ScrollView>  
  
  { data.map((o) => (  
    <Text>{ o.namapedagang }</Text>  
  )  
}  
  
</ScrollView>
```

## 2. FlatList

FlatList is a method for processing data on a large scale. FlatList renders items lazily, when they are about to appear, and removes items that scroll way off-screen to save memory & optimize the performance. FlatList can also be customized so that we can set, for example, control the initial amount of data on first render.

How to use FlatList:

```
import { Text, View, FlatList } from 'react-native';  
  
<FlatList  
  data={DATA}  
  initialNumToRender={10}  
  renderItem={({ item }) => (  
    <Text>{ item.title }<Text />  
  )};  
  keyExtractor={item => item.id}  
>
```

## 3. Array.Map()

The map() method creates a new array & sometimes it used to loop a data. Array.Map() is comes from default Javascript array method. To use Array.Map() it must have a set of data.

How to use map():

```
<ScrollView>  
  
  { data.map((o) => (  
  
    <Text> {o.namapedagang} </Text>  
  
  )  
  
  }  
  
</ScrollView>
```

#### 4. Lodash

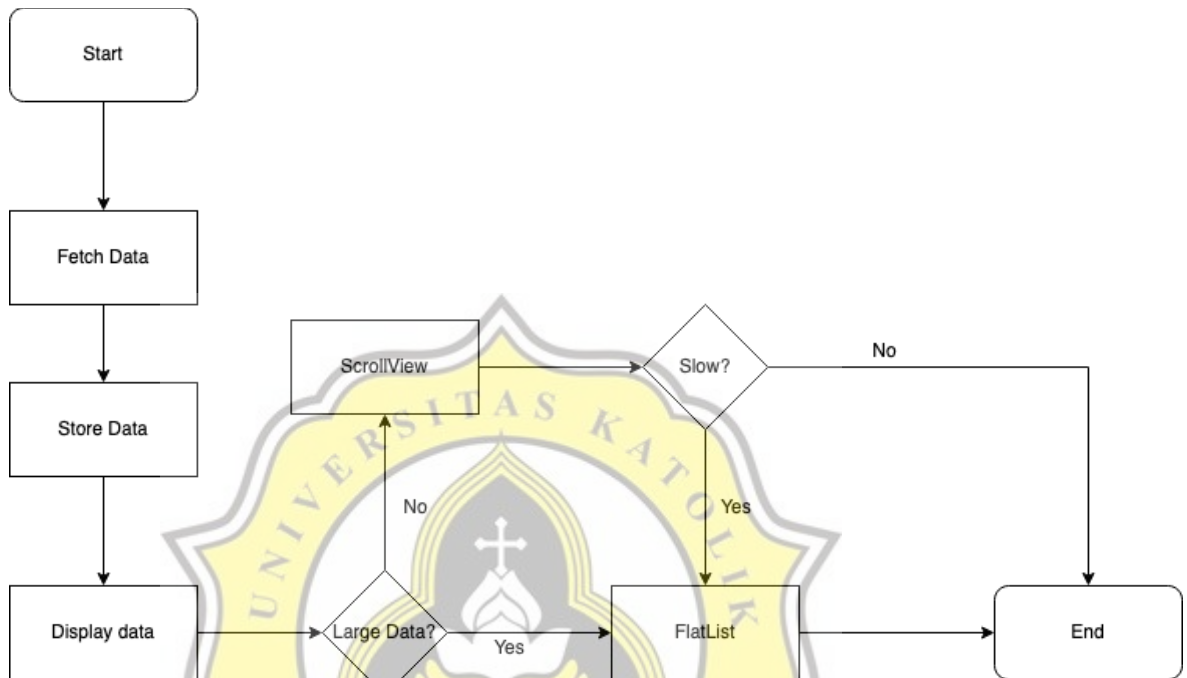
Lodash is Javascript popular utility library that provides utility functions, that make coding in Javascript more clean and easier. In here, will be used map() from Lodash. Lodash is also easy to use because all of its output will be in the form of an array so that it is easy to process for various purposes.

How to use Lodash:

Import \_ from 'lodash'

```
<ScrollView>  
  
  { _.map(data, (o) => (  
  
    <Text>{o.namapedagang}</Text>  
  
  ))  
  
  }  
  
</ScrollView>
```

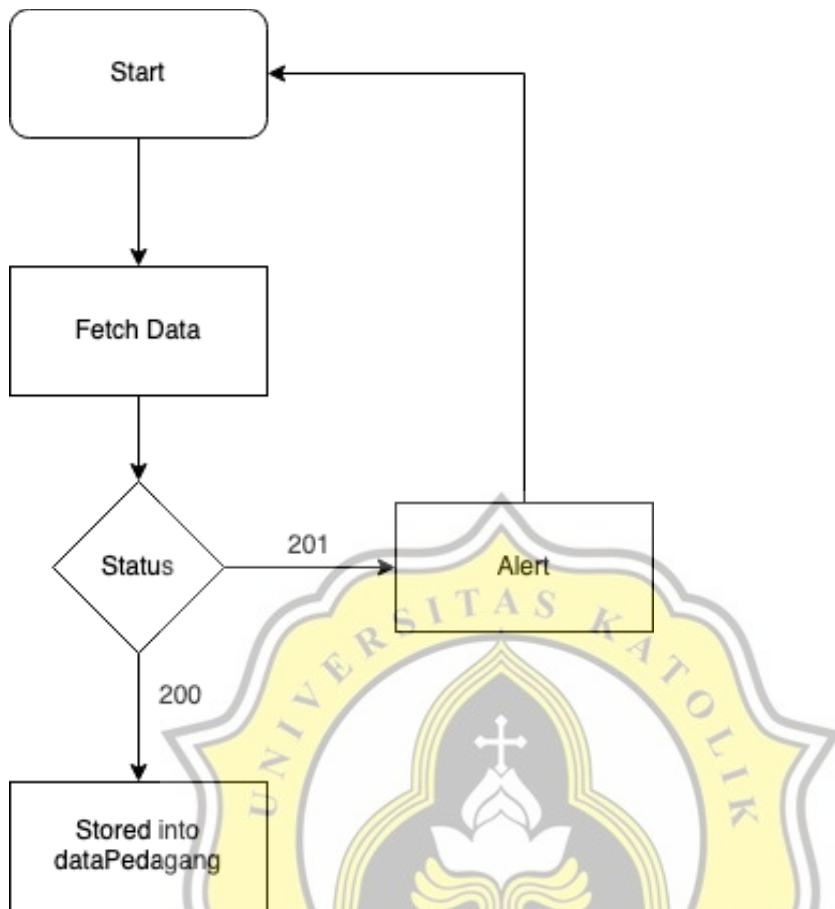
## 4.2 Design



**Figure 4.1.** Flowchart Process

The work process of this project can be seen on the Flowchart above, the first step is fetch the data from API Backend, if the response status is JSON 201 then an error will occur so that the data cannot be retrieved, and if the response status is JSON 200 then the data is successfully obtained. After the data is successfully obtained, the data will be store to local state in React Native and display the data. We can determine which method to use to process the data, if it's a large data we should use FlatList for better performance & save memory. And if it's not a large data, we can go with ScrollView.

ScrollView have several options for processing, namely with Lodash and Array.Map() generally have the same function.



**Figure 4.2.** FlowChart Fetch Data

This is a process flowchart for storing Banjar market trader data on React Native Via API. When the user enters the search page, it will automatically run a function to fetch data. When JSON responds to status from server 201, 204, or 500, an Alert will appear containing information if an error occurs with the status code earlier. If the JSON 200 status response is a sign that the data has been successfully obtained and can be stored in a variable to accommodate the data.