

React Native Development

Project Albums

Walkthrough

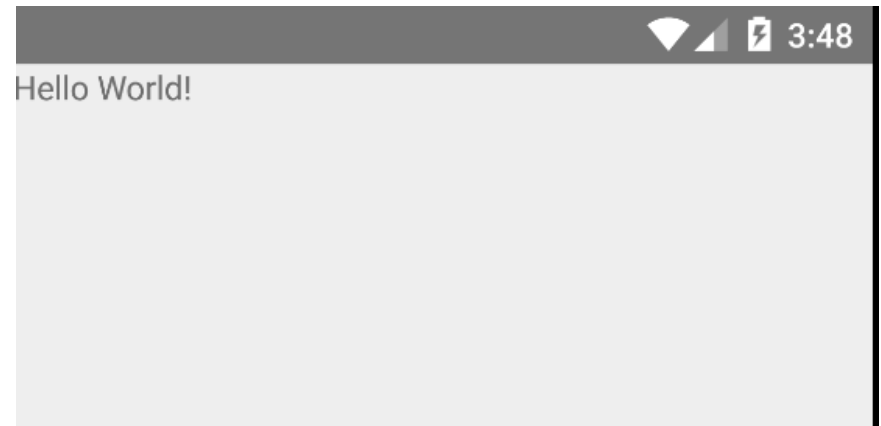
Step 1

- Buka **App.js**, lalu hapus semua yang ada didalamnya. Dan kemudian isi dengan code ini:

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

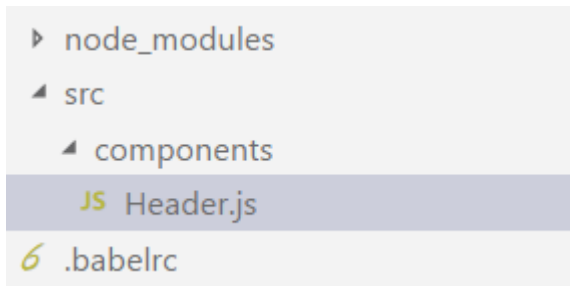
const App = () => {
  return (
    <View>
      <Text>Hello World!</Text>
    </View>
  );
};

export default App;
```



Step 2

- Buat folder src, lalu buat folder components didalam folder src, kemudian buat file js (Header.js)



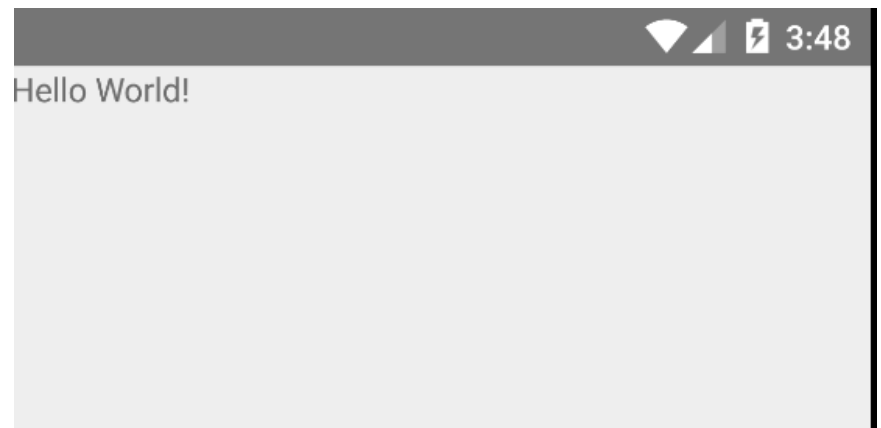
Step 3

- Isi Header.js dengan code ini :

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

const Header = () => {
  return (
    <View>
      <Text>Header</Text>
    </View>
  );
};

export default Header;
```



Step 4

- Import dan pasang component Header ke App.js

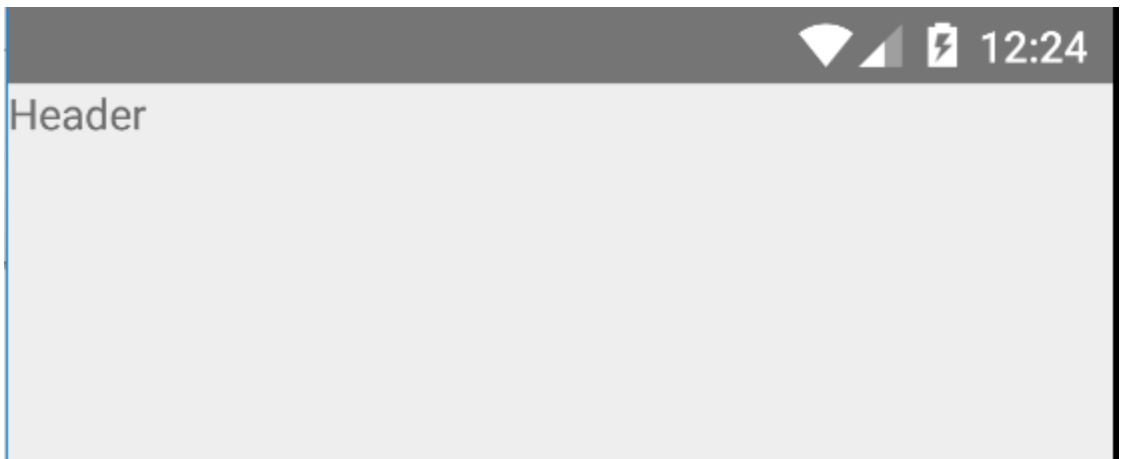
```
import React from 'react';
```

```
import { View } from 'react-native';
```

```
import Header from './src/components/Header';
```

```
const App = () => {  
  return (  
    <View>  
      <Header />  
    </View>  
  );  
};
```

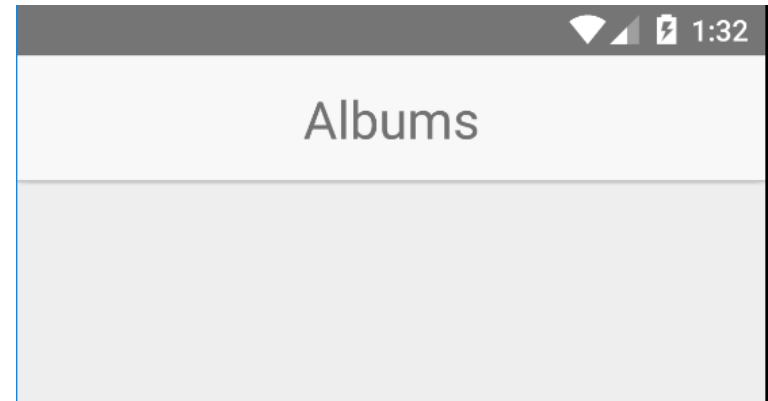
```
export default App;
```



Step 5

- Kasih Style ke Headernya biar terlihat bagus

```
const styles = {  
  viewStyle: {  
    backgroundColor: '#F8F8F8',  
    justifyContent: 'center',  
    alignItems: 'center',  
    height: 60,  
    shadowColor: '#000',  
    shadowOffset: { width: 0, height: 2 },  
    shadowOpacity: 0.2,  
    elevation: 2,  
    position: 'relative'  
  },  
  textStyle: {  
    fontSize: 25  
  }  
};
```



```
const Header = () => {  
  const { textStyle, viewStyle } = styles;  
  return (  
    <View style={viewStyle}>  
      <Text style={textStyle}>Albums</Text>  
    </View>  
  );  
};
```

Step 6

- Buat Header Component lebih reusable

```
const Header = (props) => {  
  const { textStyle, viewStyle } = styles;  
  return (  
    <View style={viewStyle}>  
      <Text style={textStyle}>{ props.headerText }</Text>  
    </View>  
  );  
};
```

Dan File App.js di edit juga

```
const App = () => {  
  return (  
    <View>  
      <Header headerText={'Albums'} />  
    </View>  
  );  
};
```

Step 7

- Buat component baru AlbumList.js di folder components, dan tulis:

```
import React, { Component } from 'react';
```

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

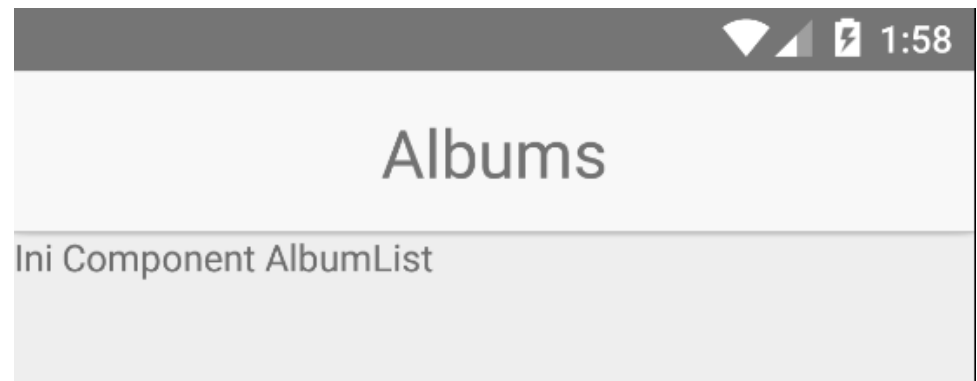
```
class AlbumList extends Component {  
  render() {  
    return (  
      <View>  
        <Text>Ini Component AlbumList</Text>  
      </View>  
    );  
  }  
}
```

```
export default AlbumList;
```


Step 8

- Kemudian pasang component tersebut ke App.js

```
const App = () => {  
  return (  
    <View>  
      <Header headerText={'Albums'} />  
      <AlbumList />  
    </View>  
  );  
};
```



Step 9

- Tambahkan Lifecycle Method `componentWillMount()` kedalam component AlbumList di atas function `render()`
- Kemudian install Axios untuk melakukan http Request dengan mengetik (`npm install --save axios`)
- Kemudian import library tersebut ke AlbumList.js untuk dipakai `import axios from 'axios';`
- Kemudian buat component level state, taruh diatas `componentWillMount()`

```
state = {  
  albums: []  
};
```
- Kemudian tulis code ini didalam function `componentWillMount()`

```
axios.get('https://rallycoding.herokuapp.com/api/music_albums')  
  .then(response => this.setState({ albums: response.data }));
```

Step 10

- Kemudian buat function `renderAlbums()` diatas function `render()`

```
renderAlbums() {  
    return this.state.albums.map(album =>  
        <Text key={album.title}>{album.title}</Text>  
    );  
}
```

- Kemudian panggil function tersebut di dalam function `render()`

```
render() {  
    return (  
        <View>  
            {this.renderAlbums()}  
        </View>  
    );  
}
```

Step 11

- Kemudian buat component baru AlbumDetail.js di folder yg sama

```
import React from 'react';
```

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

```
const AlbumDetail = (props) => {  
  return (  
    <View>  
      <Text>{props.album.title}</Text>  
    </View>  
  );  
};
```

```
export default AlbumDetail;
```

Step 12

- Kemudian import dan pakai component `AlbumDetail` di `AlbumList`
`import AlbumDetail from './AlbumDetail';`

Lalu ubah function `renderAlbums`:

```
renderAlbums() {  
    return this.state.albums.map(album =>  
        <AlbumDetail key={album.title} album={album}/>  
    );  
}
```

Step 13

- Buat file component baru Card.js difolder yang sama:

```
import React from 'react';
```

```
import { View } from 'react-native';
```

```
const Card = (props) => {  
  return (  
    <View>  
      {props.children}  
    </View>  
  );  
};
```

```
export default Card;
```

Step 14

- Baru kasih style ke card component tersebut

```
const styles = {  
  containerStyle: {  
    borderWidth: 1,  
    borderRadius: 2,  
    borderColor: '#ddd',  
    borderBottomWidth: 0,  
    shadowColor: '#000',  
    shadowOffset: { width: 0, height: 2 },  
    shadowOpacity: 0.1,  
    shadowRadius: 2,  
    elevation: 1,  
    marginLeft: 5,  
    marginRight: 5,  
    marginTop: 10  
  }  
};
```

```
const Card = (props) => {  
  return (  
    <View style={styles.containerStyle}>  
      {props.children}  
    </View>  
  );  
};
```

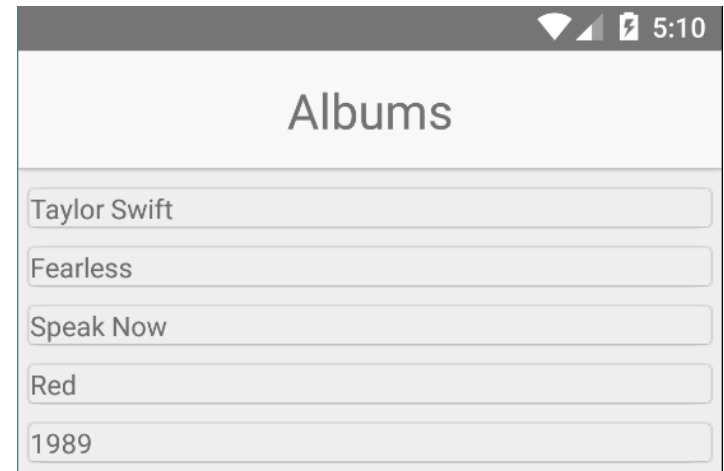
Step 15

- Baru import Card component tersebut ke AlbumDetail

```
import Card from './Card';
```

- Ganti Tag View ke Card:

```
const AlbumDetail = (props) => {  
  return (  
    <Card>  
      <Text>{props.album.title}</Text>  
    </Card>  
  );  
};
```



Step 16

- Buat File Component CardSection difolder yang sama:

```
import React from 'react';  
import { View } from 'react-native';
```

```
const CardSection = (props) => {  
  return (  
    <View>  
      {props.children}  
    </View>  
  );  
};
```

```
export default CardSection;
```

Step 17

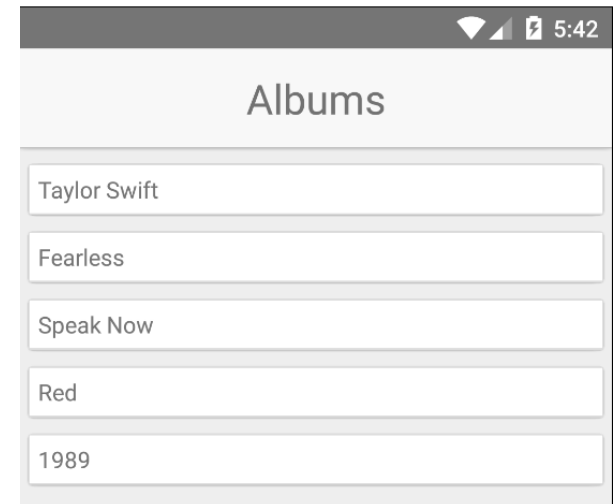
- Tambahkan Style untuk CardSection tersebut mirip seperti component Card

```
const styles = {  
  containerStyle: {  
    borderBottomWidth: 1,  
    padding: 5,  
    backgroundColor: '#fff',  
    justifyContent: 'flex-start',  
    flexDirection: 'row',  
    borderColor: '#ddd',  
    position: 'relative'  
  }  
};
```

Step 18

- Import CardSection di AlbumDetail kemudian pasang

```
const AlbumDetail = (props) => {  
  return (  
    <Card>  
      <CardSection>  
        <Text>{props.album.title}</Text>  
      </CardSection>  
    </Card>  
  );  
};
```



Step 19

- Kemudian destructure props kirimannya di [AlbumDetail](#) seperti ini

```
import CardSection from './CardSection';

const AlbumDetail = ({album}) => {
  const {
    title,
    artist,
    thumbnail_image,
    image,
    url
  } = album;
  return (
```

Step 20

- Buat title Albumnya seperti ini:

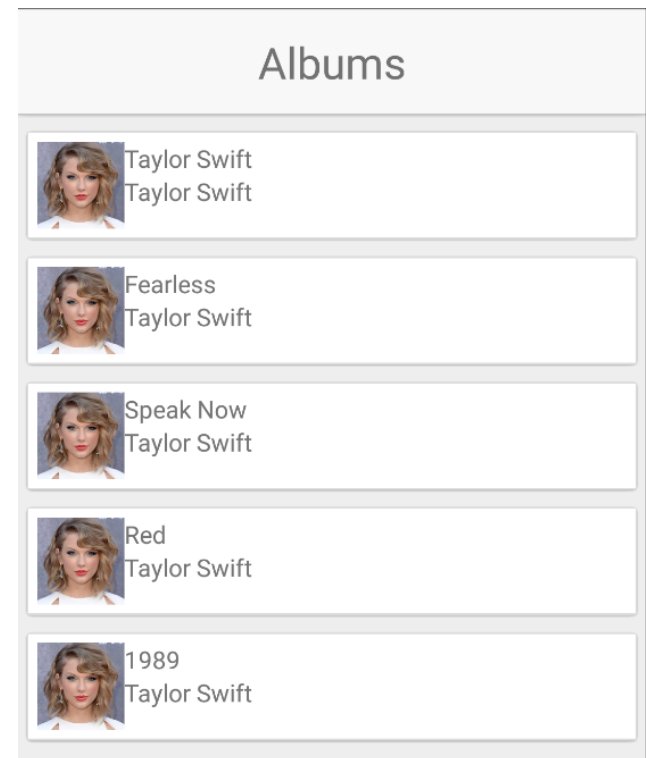
```
} = album;  
return (  
  <Card>  
    <CardSection>  
      <View>  
        <Image  
          source={{ uri: thumbnail_image }}  
        />  
      </View>  
      <View>  
        <Text>{title}</Text>  
        <Text>{artist}</Text>  
      </View>  
    </CardSection>  
  </Card>  
)
```

Albums
Taylor Swift Taylor Swift
Fearless Taylor Swift
Speak Now Taylor Swift
Red Taylor Swift
1989 Taylor Swift

Step 21

- Kemudian kasih style ke image thumbnailnya biar terlihat:

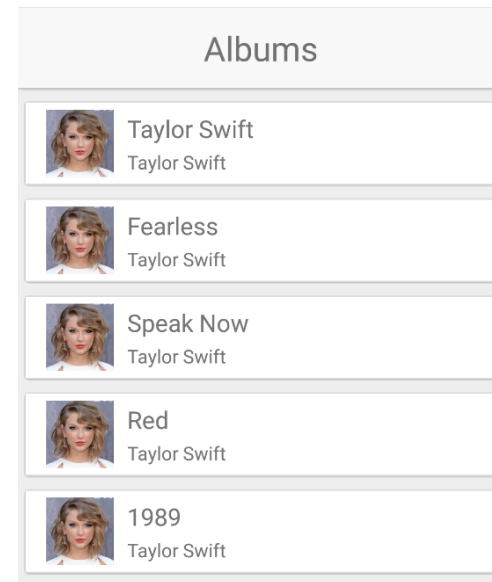
```
const styles = {  
  thumbnailStyle: {  
    height: 50,  
    width: 50  
  }  
};  
  
<CardSection>  
  <View>  
    <Image  
      style={styles.thumbnailStyle}  
      source={{ uri: thumbnail_image }}  
    />  
  </View>
```



Step 22

- Tambahkan 3 style ini ke `const Styles`:

```
headerContentStyle: {
  justifyContent: 'space-around'
},
headerTextStyle: {
  fontSize: 18
},
thumbnailContainerStyle: {
  justifyContent: 'center',
  alignItems: 'center',
  marginLeft: 10,
  marginRight: 10
},
```



```
<View style={styles.thumbnailContainerStyle}>
  <Image
    style={styles.thumbnailStyle}
    source={{ uri: thumbnail_image }}
  />
</View>
<View style={styles.headerContentStyle}>
  <Text style={styles.headerTextStyle}>{title}</Text>
  <Text>{artist}</Text>
</View>
```

Step 23

- Kemudian destructure stylesnya jadi:

```
const {  
  headerContentStyle,  
  headerTextStyle,  
  thumbnailStyle,  
  thumbnailContainerStyle,  
} = styles;
```

```
<View style={thumbnailContainerStyle}>  
  <Image  
    style={thumbnailStyle}  
    source={{ uri: thumbnail_image }}  
  />  
</View>  
<View style={headerContentStyle}>  
  <Text style={headerTextStyle}>{title}</Text>  
  <Text>{artist}</Text>  
</View>
```


Step 24

- Tambahkan CardSection baru dibawah CardSection Title

```
<CardSection>  
  <Image  
    style={imageStyle}  
    source={{ uri: image }}  
  />  
</CardSection>
```

- Kemudian tambahkan imageStyle ke const styles, dan panggil ditempat destructurenya

```
imageStyle: {  
  height: 300,  
  flex:1  
}
```



Step 25

- Ubah View tag di AlbumList menjadi ScrollView supaya bisa ngescroll kebawah(jangan lupa import ScrollView terlebih dahulu):

```
<ScrollView>
```

```
    {this.renderAlbums()}
```

```
</ScrollView>
```

Step 26

- Kemudian buat file Component Button.js di folder yg sama:

```
import React from 'react';
```

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

```
const Button = () => {  
  return (  
    <TouchableOpacity>  
      <Text>  
        Click Me!  
      </Text>  
    </TouchableOpacity>  
  );  
};
```

```
export default Button;
```

Step 27

- Kemudian import ke AlbumDetail dan pasang ditaruh dibawah CardSection Image:

```
<CardSection>
  <Image
    style={imageStyle}
    source={{ uri: image }}
  />
</CardSection>
<CardSection>
  <Button />
</CardSection>
```

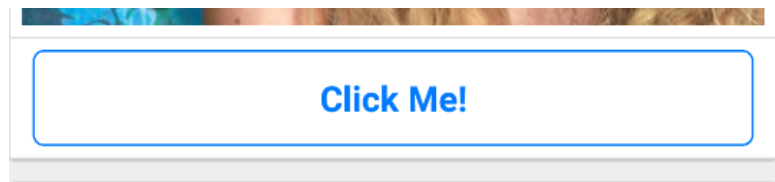


Step 28

- Kemudian kasih style ke Buttonnya dan Textnya:

```
const styles = {  
  buttonStyle: {  
    flex: 1,  
    alignSelf: 'stretch',  
    backgroundColor: '#fff',  
    borderRadius: 5,  
    borderWidth: 1,  
    borderColor: '#007aff',  
    marginLeft: 5,  
    marginRight: 5  
  },  
  textStyle: {  
    alignSelf: 'center',  
    color: '#007aff',  
    fontSize: 16,  
    fontWeight: '600',  
    paddingTop: 10,  
    paddingBottom: 10  
  }  
};
```

```
const Button = () => {  
  const { buttonStyle, textStyle } = styles;  
  
  return (  
    <TouchableOpacity style={buttonStyle}>  
      <Text style={textStyle}>  
        Click Me!  
      </Text>  
    </TouchableOpacity>  
  );  
};
```



Step 29

- Kemudian buat Component Button jadi lebih reuseable dan bisa menerima user input(onPress):

```
const Button = ({ onPress, children }) => {  
  const { buttonStyle, textStyle } = styles;  
  
  return (  
    <TouchableOpacity onPress={onPress} style={buttonStyle}>  
      <Text style={textStyle}>  
        {children}  
      </Text>  
    </TouchableOpacity>  
  );  
};
```

Step 30

- Kemudian kirim children props sama onPress props dari AlbumDetail ke Button :

```
<CardSection>
  <Button onPress={() => Linking.openURL(url)} >
    Buy Now
  </Button>
</CardSection>
```

- Linking didapatkan dari import dari ReactNative :

```
import { Text, View, Image, Linking } from 'react-native';
```

Finish!!!

- Apps pertama kita akhirnya selesai yeay!!!

