

TUGAS INDIVIDU/PERORANGAN

Firestore React

Disusun sebagai

MATA KULIAH : PEMROGRAMAN BERBASIS FRAMEWORK



Oleh:

Arifanny Ramadhan Sukma

1741720217 / TI-3A

04

PROGRAM STUDI D-IV TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2020

Praktikum

Dalam praktikum ini kita membutuhkan firebase sebagai backend dan reactjs sebagai frontend, oleh karena itu pertama kita lakukan set up pada firebase.

1.1 Langkah Praktikum

1. Buatlah project react pada directory yang baru.
2. Setelah itu pada folder **src** buatlah file dengan nama *index.js* ; *Login.js* ; *Join.js* ; *Header.js* ; *routes.js* ; *styles.css* (file lain yang ada di folder src bisa dihapus saja).
3. Karena pada praktikum ini kita menggunakan react router dom, maka lakukan *npm install react-router-dom* dahulu.
4. Jika sudah, isilah code seperti dibawah ini sesuai dengan nama file masing-masing.

index.js

```
1  import React, { useState } from "react";
2  import ReactDOM from "react-dom";
3  import { Switch, BrowserRouter as Router, Route } from "react-router-dom";
4  import routes from "./routes.js";
5  import Header from "./Header";
6  import "./styles.css";
7
8  export const AuthContext = React.createContext(null);
9
10 function App() {
11   const [isLoggedIn, setLoggedIn] = useState(false);
12
13   return (
14     <AuthContext.Provider value={{ isLoggedIn, setLoggedIn }}>
15       Is logged in? {JSON.stringify(isLoggedIn)}
16       <div className="App">
17         <Router>
18           <Header />
19
20           <Switch>
21             {routes.map(route => (
22               <Route
23                 key={route.path}
24                 path={route.path}
25                 exact={route.exact}
26                 component={route.main}
27               />
28             ))}
29
30           </Switch>
31         </Router>
32       </div>
33     </AuthContext.Provider>
34   );
35 }
36
37 const rootElement = document.getElementById("root");
38 ReactDOM.render(<App />, rootElement);
```

Login.js

```
1 import React, { useState, useContext } from "react";
2 import { AuthContext } from "../index";
3
4 const Login = () => {
5   const [email, setEmail] = useState("");
6   const [password, setPassword] = useState("");
7   const [error, setErrors] = useState("");
8
9   const Auth = useContext(AuthContext);
10  const handleForm = e => {
11    e.preventDefault();
12    console.log(Auth);
13    Auth.setLoggedIn(true);
14  };
15
16  return (
17    <div>
18      <h1>Login</h1>
19      <form onSubmit={e => handleForm(e)}>
20        <input
21          value={email}
22          onChange={e => setEmail(e.target.value)}
23          name="email"
24          type="email"
25          placeholder="email"
26          />
27
28        <input
29          onChange={e => setPassword(e.target.value)}
30          name="password"
31          value={password}
32          type="password"
33          placeholder="password"
34          />
35        <hr />
36        <button class="googleBtn" type="button">
37          
41          Login With Google
42        </button>
43        <button type="submit">Login</button>
44        <span>{error}</span>
45      </form>
46    </div>
47  );
48 };
49 export default Login;
50
```

Join.js

```
1 import React, { useState, useContext } from "react";
2 import { AuthContext } from "../index";
3
4 const Join = () => {
5   const [email, setEmail] = useState("");
6   const [password, setPassword] = useState("");
7   const [error, setErrors] = useState("");
8
9   const Auth = useContext(AuthContext);
10  const handleForm = e => {
11    e.preventDefault();
12    console.log(Auth);
13    Auth.setLoggedIn(true);
14  };
15
16  return (
17    <div>
18      <h1>Join</h1>
19      <form onSubmit={e => handleForm(e)}>
20        <input
21          value={email}
22          onChange={e => setEmail(e.target.value)}
23          name="email"
24          type="email"
25          placeholder="email"
26          />
27      </form>
28    </div>
29  );
30 };
31 export default Join;
```

```

27     <input
28       onChange={e => setPassword(e.target.value)}
29       name="password"
30       value={password}
31       type="password"
32       placeholder="password"
33     />
34     <hr />
35     <button className="googleBtn" type="button">
36       
40       Join With Google
41     </button>
42
43     <button type="submit">Login</button>
44
45     <span>{error}</span>
46   </form>
47 </div>
48 );
49 };
50
51 export default Join;
52

```

Header.js

```

1  import React from "react";
2  import routes from "./routes";
3  import { Link } from "react-router-dom";
4
5  const Header = () => (
6    <ul className="nav">
7      {routes.map((route, i) => (
8        <li key={i}>
9          <Link to={route.path}>{route.name}</Link>
10        </li>
11      ))}
12    </ul>
13  );
14
15  export default Header;
16

```

routes.js

```

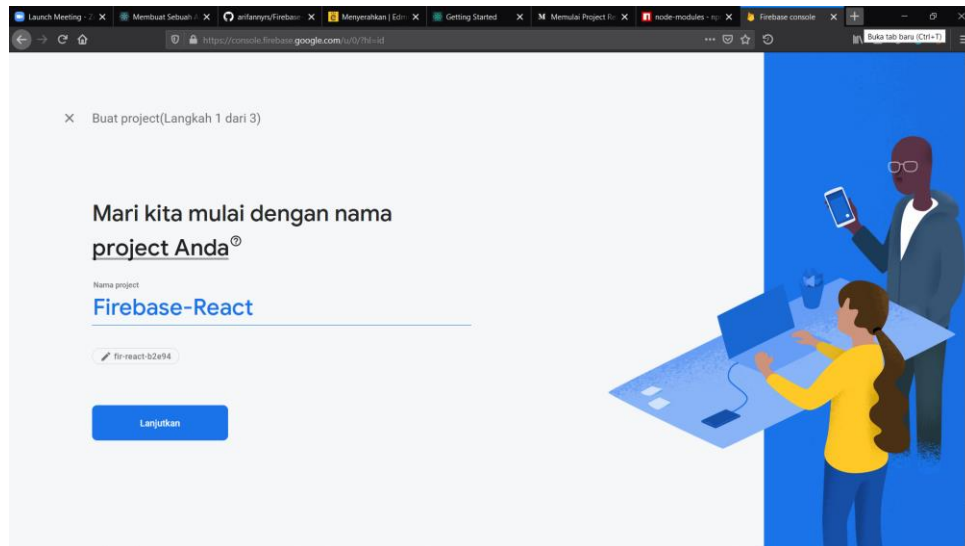
1  import React from "react";
2  import Login from "./Login";
3  import Join from "./Join";
4
5  const routes = [
6    { name: "Join", path: "/", exact: true, main: () => <Join /> },
7    { name: "Login", path: "/login", exact: true, main: () => <Login /> }
8  ];
9
10 export default routes;
11

```

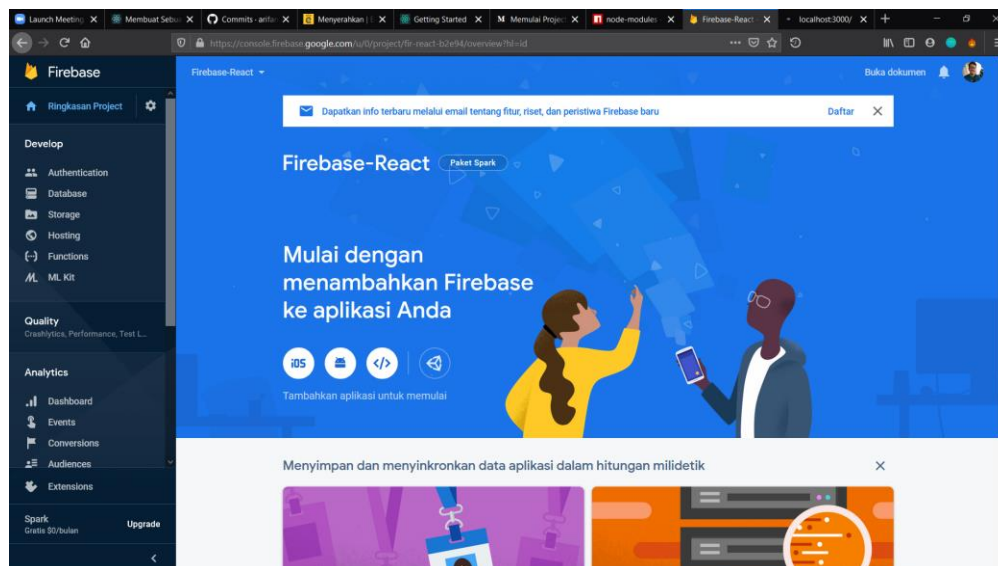
style.css

```
1  .App {
2    font-family: sans-serif;
3    padding: 50px;
4  }
5
6  h1 {
7    margin: 0 0 25px 0;
8    text-align: left;
9  }
10
11 form {
12   display: flex;
13   flex-direction: column;
14 }
15
16 input {
17   max-width: 450px;
18   width: auto;
19   border: none;
20   margin: 0 0 25px 0;
21   box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
22   transition: all 0.3s cubic-bezier(0.25, 0.8, 0.25, 1);
23   padding: 10px;
24 }
25
26 button {
27   width: 125px;
28   box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
29   border: none;
30   padding: 7px;
31   text-align: center;
32   cursor: pointer;
33 }
34
35 button[type="submit"] {
36   background-color: #4c566a;
37   color: white;
38   font-size: 14px;
39   font-weight: 900;
40   border-radius: 5px;
41   text-transform: uppercase;
42   padding: 15px;
43 }
44
45 .googleBtn {
46   width: 225px;
47   margin: 0 0 50px 0;
48   display: flex;
49   justify-content: center;
50   font-size: 16px;
51 }
52
53 .googleBtn img {
54   width: 16px;
55   height: 16px;
56   padding: 0;
57   margin: 0 5px;
58   vertical-align: middle;
59 }
60
61 .nav {
62   display: flex;
63   text-decoration: none;
64   list-style-type: none;
65   justify-content: flex-start;
66   margin: 0 0 50px 0;
67   padding: 0;
68 }
69
70 .nav li {
71   margin: 0 10px;
72 }
73
```

5. Kemudian buka <https://console.firebase.google.com/> setelah itu buat project baru dengan nama project yang kalian inginkan.



6. Pada cmd lakukan perintah ***npm install --save firebase*** di directory project react untuk menambahkan dependency firebase kedalam react kita.
7. Setelah itu lakukan setting pada project firebase yang kalian buat tadi agar react kalian dapat terhubung dengan firebase. Pada setting get started pilih yang untuk web.



8. Lalu setelah itu anda akan mendapatkan script konfigurasi yang akan menjadi penghubung dengan project react anda. Salin script tersebut lalu tempel pada file baru di dalam folder **src** yang kita beri nama **firebase.config.js** (gambar dibawah adalah contoh, sesuaikan script yang anda dapat dari firebase).

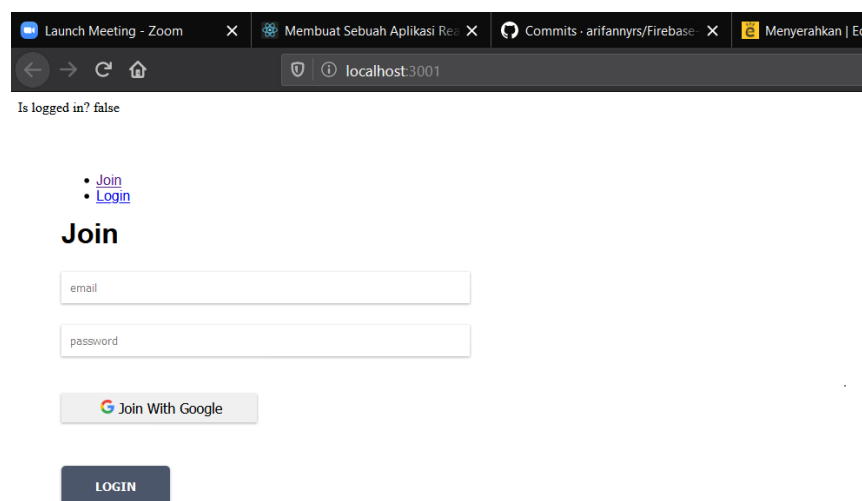
firebase.config.js

```
export default {
  apiKey: "AIzaSyAQX4Bgu_-7qwsNWuKXcSiisCuhu5-DVUc",
  authDomain: "fir-react-b2e94.firebaseio.com",
  databaseURL: "https://fir-react-b2e94.firebaseio.com",
  projectId: "fir-react-b2e94",
  storageBucket: "fir-react-b2e94.appspot.com",
  messagingSenderId: "479170393890",
  appId: "1:479170393890:web:28d1f76dbc76ca27a9977b",
  measurementId: "G-X5B8QNLJFC"
}
```

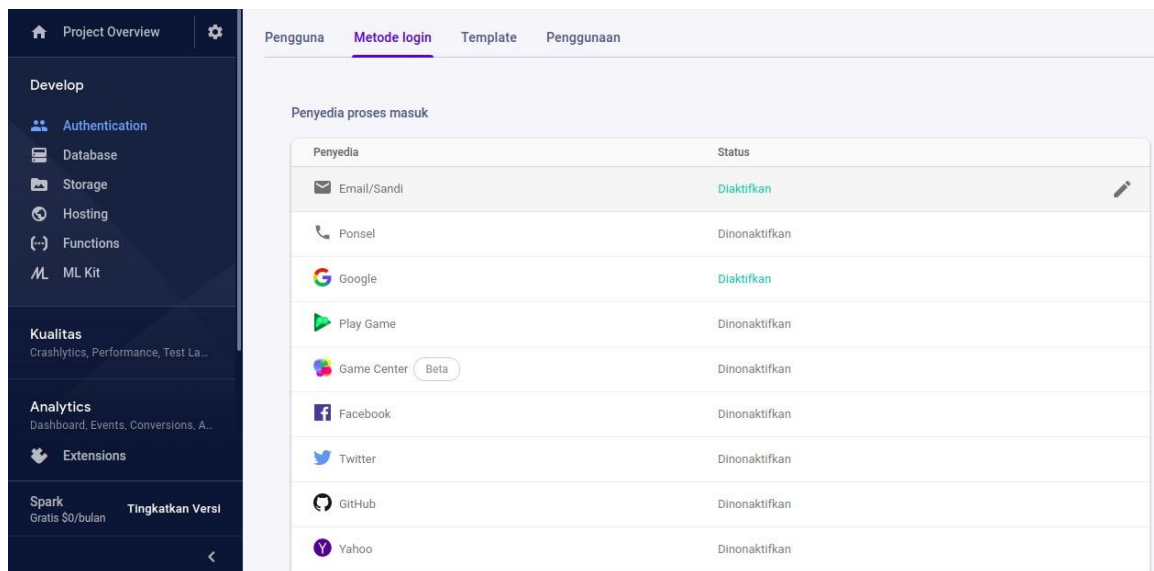
9. Lalu lakukan inialisasi firebase config dengan menambahkan code pada **index.js** seperti dibawah ini.

```
1 import React, { useState } from "react";
2 import ReactDOM from "react-dom";
3 import { Switch, BrowserRouter as Router, Route } from "react-router-dom";
4 import routes from "./routes.js";
5 import Header from "./Header";
6 import "./styles.css";
7 import * as firebase from "firebase";
8 import firebaseConfig from "./firebase.config";
9
10 firebase.initializeApp(firebaseConfig);
11
12 export const AuthContext = React.createContext(null);
```

10. Jika sudah berhasil maka coba jalankan, jika tidak terdapat error maka project anda telah terhubung dengan firebase.



11. Selanjutnya aktifkan terlebih dahulu authentication email dan google pada firebase console.



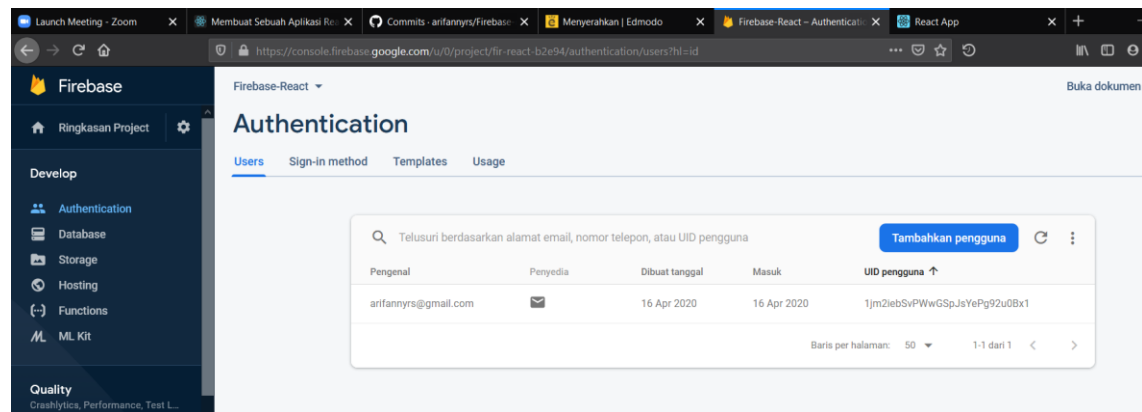
12. Jika sudah, lakukan edit code pada **Join.js** untuk membuat email dan password untuk user. Pertama import dahulu objek firebase agar bisa digunakan pada **Join.js** dengan code berikut.

```
1 import React, { useState, useContext } from "react";
2 import { AuthContext } from "./index";
3 import * as firebase from "firebase";
4
```

Kemudian cari method auth dan lakukan edit code seperti dibawah ini.

```
9
10 const Auth = useContext(AuthContext);
11 const handleForm = e => {
12   e.preventDefault();
13   firebase
14     .auth()
15     .createUserWithEmailAndPassword(email, password)
16     .then(res => {
17       if (res.user) Auth.setLoggedIn(true);
18     })
19     .catch(e => {
20       setErrors(e.message);
21     });
22 };
23
24 return (
```

13. Silahkan coba pada aplikasi dengan klik join, lalu isikan email dan password. Coba cek pada firebase maka user telah terbuat seperti dibawah ini.



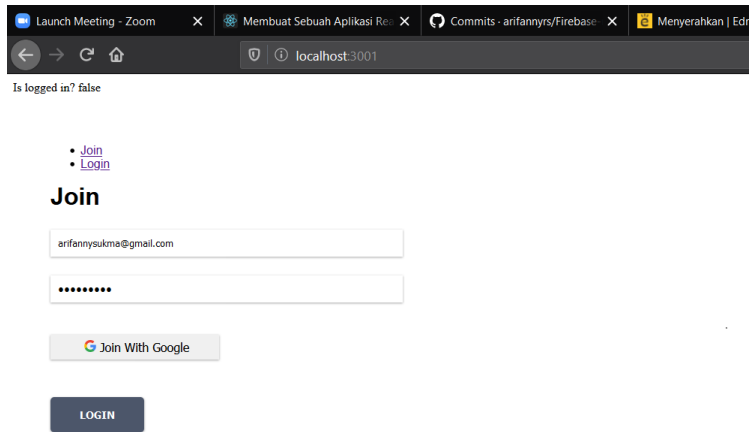
14. Langkah selanjutnya adalah bagaimana caranya agar kita dapat sign in/login dengan akun yang telah dibuat, yaitu dengan cara melakukan edit code pada **Login.js**. Pertama import dahulu objek firebase agar bisa digunakan pada **Login.js** dengan code berikut.

```
1 import React, { useState, useContext } from "react";
2 import { AuthContext } from "../index";
3 import * as firebase from "firebase";
4
```

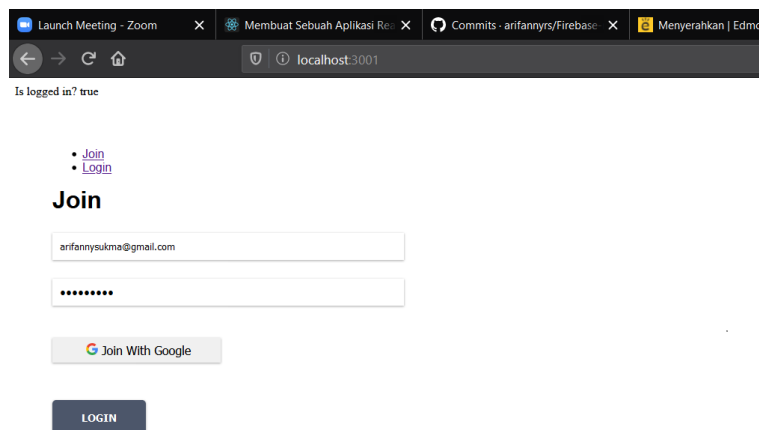
Kemudian cari method auth dan lakukan edit code seperti dibawah ini.

```
9
10 const Auth = useContext(AuthContext);
11 const handleForm = e => {
12   e.preventDefault();
13   firebase
14     .auth()
15     .signInWithEmailAndPassword(email, password)
16     .then(res => {
17       if (res.user) Auth.setLoggedIn(true);
18     })
19     .catch(e => {
20       setErrors(e.message);
21     });
22 };
23
24 return (
```

15. Silahkan coba pada aplikasi dengan cara klik login kemudian masukkan email dan password yang telah didaftarkan, kemudian perhatikan keterangan login pada pojok kiri atas dengan perubahan kondisi dari **false** menjadi **true** yang menandakan anda telah berhasil login.



Menjadi



1.2 Soal Latihan Praktikum

1. Buatlah laporan dengan menyertakan screenshot hasil dan code yang telah dibuat.
2. Pada button join with google belum terdapat aksi, silahkan tambahkan function agar dapat mendaftar dengan cara klik button tersebut lalu memilih akun gmail yang akan dipakai.

