

LAPORAN PRAKTIKUM DAN TUGAS
PEEMEROGGRAMMAN BERBASIS FRAMEWORK



Nama : Alenovan Wiradhika Putra
Kelas : TI-3C D-IV T.INFORMATIKA
NIM : 1741720065

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

Karna praktium dan tugas masih berhubungan stepnya maka dari itu laporan saya jadikan satu

Index.js

```
import React, { useState } from 'react';
import ReactDOM from 'react-dom';
import { Switch, BrowserRouter as Router, Route } from "react-router-dom";
import routes from "./routes";
import Header from "./Header";
import './style.css';
import * as firebase from "firebase";
import firebaseConfig from "./firebase.config";

firebase.initializeApp(firebaseConfig);

export const AuthContext = React.createContext(null);

function App(){
  const [isLoggedIn, setLoggedIn] = useState(false);

  return (
    <AuthContext.Provider value={{ isLoggedIn, setLoggedIn }}>
      Is Logged in? {JSON.stringify(isLoggedIn)}
      <div className="App">
        <Router>
```

```

        <Header/>
        <Switch>
          {routes.map(route => (
            <Route
              key = {route.path}
              path = {route.path}
              exact = {route.exact}
              component = {route.main}
            />
          ))}
        </Switch>
      </Router>
    </div>
  </AuthContext.Provider>
)
}

const rootElement = document.getElementById("root");
ReactDOM.render(<App />, rootElement);

```

Login.js

```

import React, { useState, useContext } from "react";
import { AuthContext } from "../index";
import * as firebase from "firebase";

const Login = () => {
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [error, setErrors] = useState("");
  const Auth = useContext(AuthContext);
  const handleForm = (e) => {
    e.preventDefault();
    firebase
      .auth()
      .signInWithEmailAndPassword(email, password)
      .then((res) => {
        if (res.user) Auth.setLoggedIn(true);
        console.log(Auth);
      })
      .catch((e) => {
        setErrors(e.message);
      });
  };
};

```

```

    });
  };

  return (
    <div>
      <h1>Login</h1>
      <form onSubmit={e => handleForm(e)}>
        <input
          value={email}
          onChange={e => setEmail(e.target.value)}
          name="email"
          type="email"
          placeholder="email"
        />

        <input
          onChange={e => setPassword(e.target.value)}
          name="password"
          value={password}
          type="password"
          placeholder="password"
        />
        <hr />
        <button className="googleBtn" type="button">
          
          Login With Google
        </button>
        <button type="submit">Login</button>
        <span>{error}</span>
      </form>
    </div>
  );
};

export default Login;

```

Join.js

```
import React, { useState, useContext } from "react";
```

```

import { AuthContext } from "../index";
import * as firebase from "firebase";

const Join = () => {
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [error, setErrors] = useState("");
  const Auth = useContext(AuthContext);
  const handleForm = (e) => {
    e.preventDefault();
    firebase
      .auth()
      .createUserWithEmailAndPassword(email, password)
      .then((res) => {
        if (res.user) Auth.setLoggedIn(true);
      })
      .catch((e) => {
        setErrors(e.message);
      });
  };
};

const googleJoin = () => {
  const provider = new firebase.auth.GoogleAuthProvider();
  firebase
    .auth()
    .signInWithPopup(provider)
    .then((res) => {
      console.log(res)
      // Auth.setLoggedIn(true);
    });
};

return (
  <div>
    <h1>Join</h1>
    <form onSubmit={(e) => handleForm(e)}>
      <input
        value={email}
        onChange={(e) => setEmail(e.target.value)}
        name="email"
        type="email"
        placeholder="email"
      />
      <input
        onChange={(e) => setPassword(e.target.value)}

```

```

        name="password"
        value={password}
        type="password"
        placeholder="password"
      />
      <hr />
      <button
        className="googleBtn"
        type="button"
        onClick={() => googleJoin()}
      >
        
        Join With Google
      </button>
      <button type="submit">Login</button>
      <span>{error}</span>
    </form>
  </div>
);
};

export default Join;

```

Header.js

```

import React from "react";
import routes from "../routes";
import {Link} from "react-router-dom";

const Header = () => (
  <ul className="Nav">
    {routes.map((route, i) => (
      <li key={i}>
        <Link to={route.path}>{route.name}</Link>
      </li>
    ))}
  </ul>
);

```

```
export default Header;
```

routes.js

```
import React from "react";
import Login from "./Login";
import Join from "./Join";

const routes = [
  { name: "Join", path: "/", exact: true, main: () => <Join /> },
  { name: "Login", path: "/Login", exact: true, main: () => <Login /> }
];

export default routes;
```

style.css

```
.App {
  font-family: sans-serif;
  padding: 50px;
}

h1 {
  margin: 0 0 25px 0;
  text-align: left;
}

form {
  display: flex;
  flex-direction: column;
}

input {
  max-width: 450px;
  width: auto;
  border: none;
  margin: 0 0 25px 0;
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
  transition: all 0.3s cubic-bezier(0.25, 0.8, 0.25, 1);
  padding: 10px;
}

button{
```

```
width: 125px;
box-shadow: 0 1px 3px rgba(0, 0, 0, 0.12), 0 1px 2px rgba(0, 0, 0, 0.24);
border: none;
padding: 7px;
text-align: center;
cursor: pointer;
}

button[type="submit"]{
  background-color: #4c566a;
  color: white;
  font-size: 14px;
  font-weight: 900;
  border-radius: 5px;
  text-transform: uppercase;
  padding: 15px;
}

.googleBtn{
  width: 225px;
  margin: 0 0 50px 0;
  display: flex;
  justify-content: center;
  font-size: 16px;
}

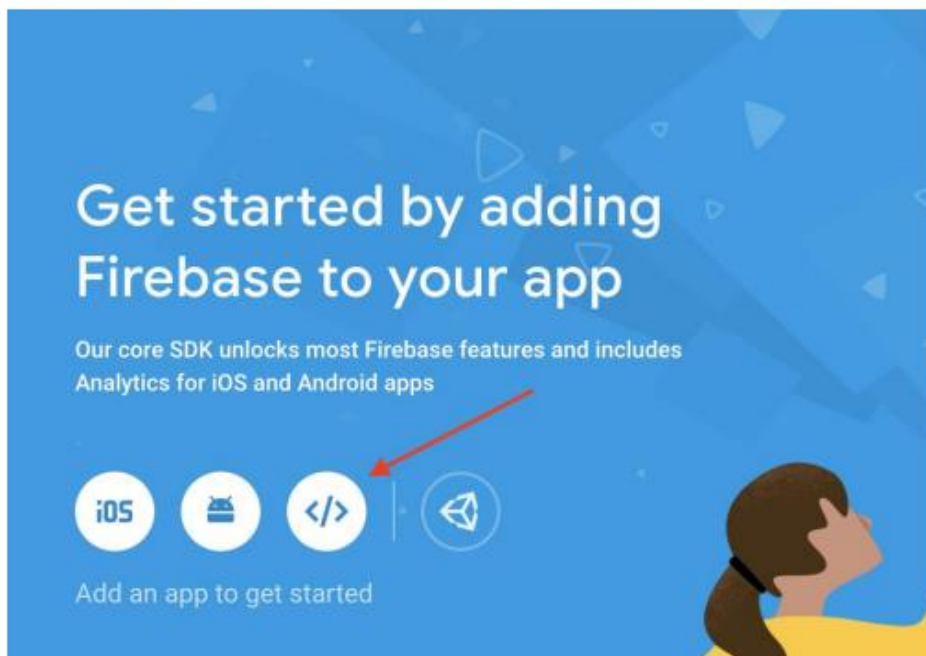
.googleBtn img {
  width: 16px;
  height: 16px;
  padding: 0;
  margin: 0 5px;
  vertical-align: middle;
}

.nav{
  display: flex;
  text-decoration: none;
  list-style-type: none;
  justify-content: flex-start;
  margin: 0 0 50px 0;
  padding: 0;
}

.nav li{
  margin: 0 10px;
```


}

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.

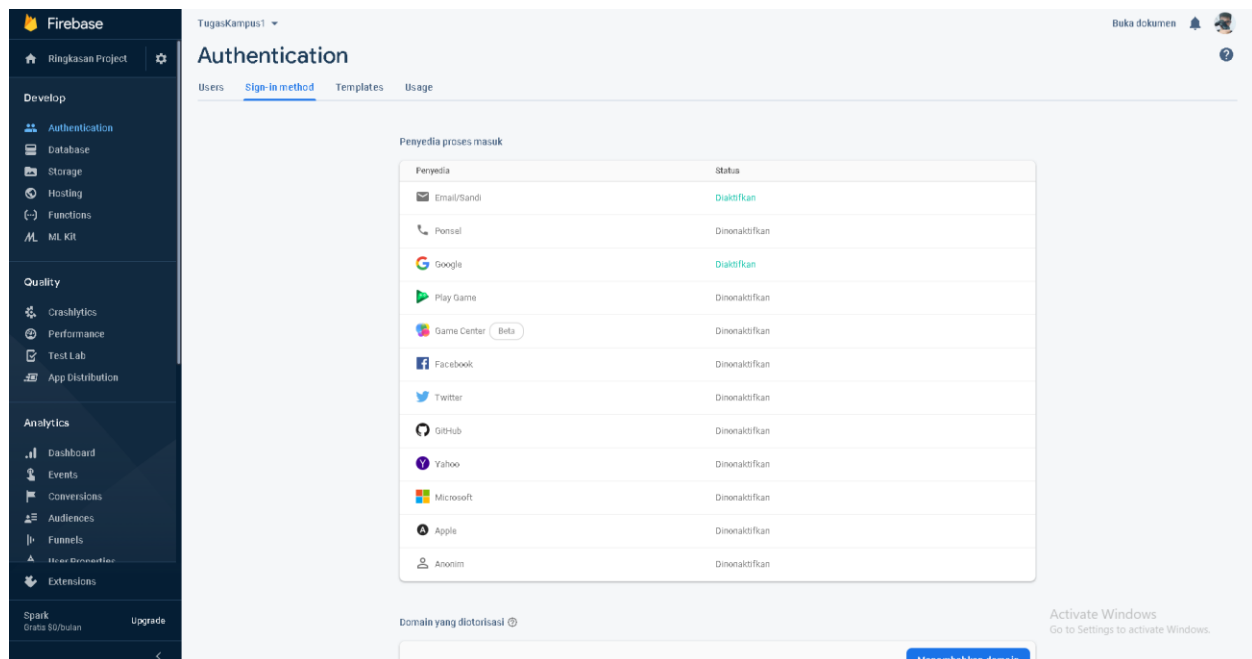


8. Lalu setelah itu anda akan mendapatkan script konfigurasi yang akan menjadi penghubung dengan project react anda. Salin script tersebut lalu tempel pada file baru di dalam folder **src** yang kita beri nama **firebase.config.js** (gambar dibawah adalah contoh, sesuaikan script yang anda dapat dari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

```
/* <script src="https://www.gstatic.com/firebasejs/7.14.1/firebase-app.js"></script>
<script src="https://www.gstatic.com/firebasejs/7.14.1/firebase-analytics.js"></script>

export default {
  apiKey: "AIzaSyA...",
  authDomain: "tugaskampus1-16cd5.firebaseio.com",
  databaseURL: "https://tugaskampus1-16cd5.firebaseio.com",
  projectId: "tugaskampus1-16cd5",
  storageBucket: "tugaskampus1-16cd5.appspot.com",
  messagingSenderId: "860486048306",
  appId: "1:860486048306:web:...",
  measurementId: "G-..."
};
// firebase.initializeApp(firebaseConfig);
// firebase.analytics();
```

Aleho van



Join menggunakan Google

Is Logged in? false



- [Join](#)
- [Login](#)

Join

 Join With Google

LOGIN

Data tesimpan di firebase

Telusuri berdasarkan alamat email, nomor telepon, atau UID pengguna					Tambahkan pengguna		
Pengenal	Penyedia	Dibuat tanggal	Masuk	UID pengguna ↑			
rewqui21@gmail.com		17 Apr 2020	17 Apr 2020	AYeusrpEkzTG9xlalnVSXUgF7Yo1			
alputra354@gmail.com		17 Apr 2020	17 Apr 2020	sjU2xS25iqMEqfvVdxhMdWhXJtp2			
Baris per halaman: 50					1-2 dari 2	<	>

Kemudian login


Is Logged in? true

- [Join](#)
- [Login](#)

Login

alputra354@gmail.com

.....

 Login With Google

LOGIN

```
Elements Network Console Sources
top
DevTools failed to parse SourceMap: chrome-ext...
[HMR] Waiting for update signal from WDS...
Download the React DevTools for a better devel...
It looks like you're using the development bui...
When deploying Firebase apps to production, it...
the individual SDK components you intend to us...
For the module builds, these are available in ...
(replace <PACKAGE> with the name of a componen...
CommonJS Modules:
const firebase = require('firebase/app');
require('firebase/<PACKAGE>');
ES Modules:
import firebase from 'firebase/app';
import 'firebase/<PACKAGE>';
Typescript:
import * as firebase from 'firebase/app';
import 'firebase/<PACKAGE>';
./src/firebase.config.js
Line 1:1: Block is redundant no-lone-block:
A cookie associated with a cross-site resource...
cross-site requests if they are set with `Same...
estatus.com/feature/5088147346030592 and https...
DevTools failed to parse SourceMap: chrome-ext...
{isLoggedIn: false, setLoggedIn: f}
```

Status berubah menjadi true

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.

Penjelasan Tugas Nomor 2

firebase.config.js

untuk data di firebase .js berisi **api key dan settign lainnya** di karenakan ini bersifat private di file yang saya kumpulkan sya **remove** di bagian api key

Menambah fungsi join dengan google, dimana nanti akan keluar popup list email , dan nanti akan bisa mengambil data dan di simpan di firebase

```
{/* <script src="https://www.gstatic.com/firebasejs/7.14.1/firebase-
app.js"></script>

<script src="https://www.gstatic.com/firebasejs/7.14.1/firebase-
analytics.js"></script> */}

export default {
  apiKey: "-----",
  authDomain: "tugaskampus1-16cd5.firebaseio.com",
  databaseURL: "https://tugaskampus1-16cd5.firebaseio.com",
  projectId: "tugaskampus1-16cd5",
  storageBucket: "tugaskampus1-16cd5.appspot.com",
  messagingSenderId: "869186910796",
  appId: "1:869186910796:web:011c5b534c1894498f92e9",
  measurementId: "G-3FN27WPPEZ"
};
//   firebase.initializeApp(firebaseConfig);
//   firebase.analytics();
```

1.Ubah Script Join.js dan tambahkan

```
const googleJoin = () => {
  const provider = new firebase.auth.GoogleAuthProvider();
  firebase
    .auth()
    .signInWithPopup(provider)
    .then((res) => {
      console.log(res)
      // Auth.setLoggedIn(true);
    });
};
```

```

<button
  className="googleBtn"
  type="button"
  onClick={() => googleJoin()}
>
  
  Join With Google
</button>

```

Yang nanti script lengkapnya akan menjadi seperti ini

```

import React, { useState, useContext } from "react";
import { AuthContext } from "../index";
import * as firebase from "firebase";

const Join = () => {
  const [email, setEmail] = useState("");
  const [password, setPassword] = useState("");
  const [error, setErrors] = useState("");
  const Auth = useContext(AuthContext);
  const handleForm = (e) => {
    e.preventDefault();
    firebase
      .auth()
      .createUserWithEmailAndPassword(email, password)
      .then((res) => {
        if (res.user) Auth.setLoggedIn(true);
      })
      .catch((e) => {
        setErrors(e.message);
      });
  };
};

const googleJoin = () => {
  const provider = new firebase.auth.GoogleAuthProvider();
  firebase

```

```

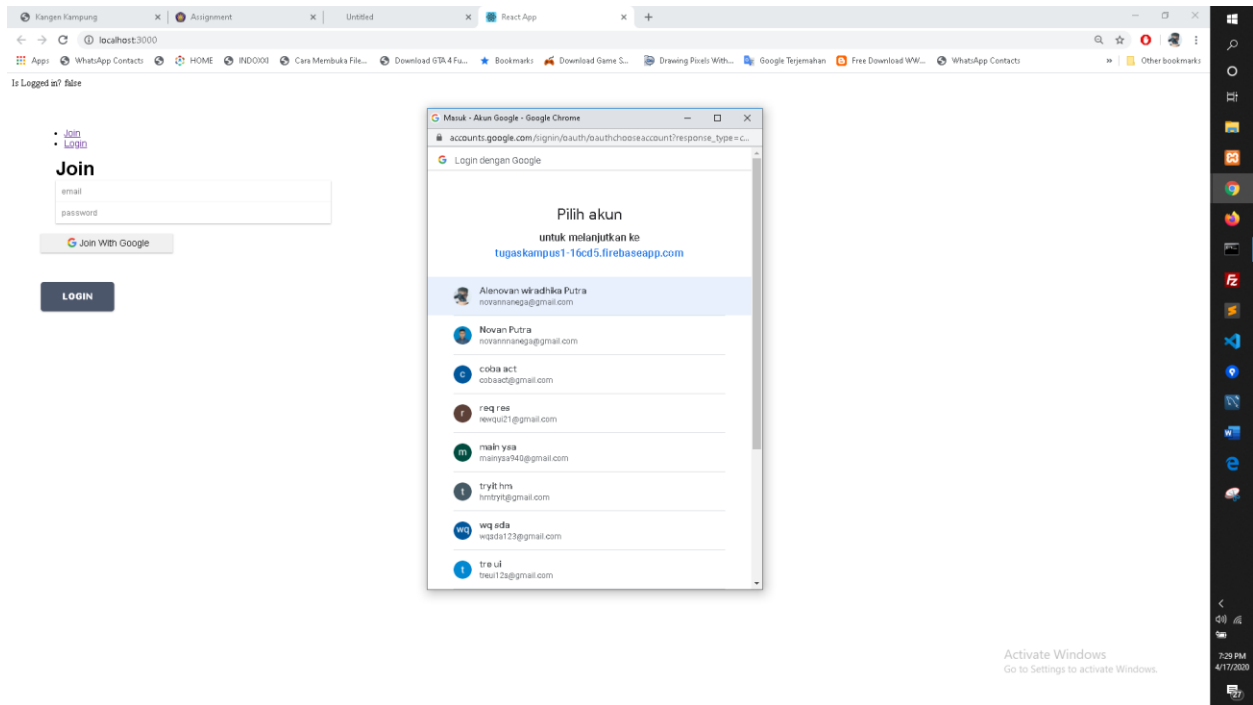
        .auth()
        .signInWithPopup(provider)
        .then((res) => {
            console.log(res)
            Auth.setLoggedIn(true);
        });
    });
};

return (
    <div>
        <h1>Join</h1>
        <form onSubmit={(e) => handleForm(e)}>
            <input
                value={email}
                onChange={(e) => setEmail(e.target.value)}
                name="email"
                type="email"
                placeholder="email"
            />
            <input
                onChange={(e) => setPassword(e.target.value)}
                name="password"
                value={password}
                type="password"
                placeholder="password"
            />
            <hr />
            <button
                className="googleBtn"
                type="button"
                onClick={() => googleJoin()}
            >
                
                Join With Google
            </button>
            <button type="submit">Login</button>
            <span>{error}</span>
        </form>
    </div>
);

```

```
};  
  
export default Join;
```

hasil dari penambahan script di atas , akan beraksi ketika di click join dengan google



Setelah mendaftar dan sukses status login akan langsung terisi karna fungsi ini

```
Auth.setLoggedIn(true);
```

Kemudia Kita Cek Di bagian firebase apakah masuk data tersebut

Telusuri berdasarkan alamat email, nomor telepon, atau UID pengguna					Tambahkan pengguna	↺	⋮
Pengenal	Penyedia	Dibuat tanggal	Masuk	UID pengguna ↑			
rewqui21@gmail.com		17 Apr 2020	17 Apr 2020	AYeusrpEkzTG9xlalnVSXUgF7Yo1			
hmtryit@gmail.com		17 Apr 2020	17 Apr 2020	h2UflKiz7gSNAyq7sNRlVcwq89j2			
alputra854@gmail.com		17 Apr 2020	17 Apr 2020	sjU2xS25iqMEqfvVdxhMdWhXJtp2			
Baris per halaman: 50					1-3 dari 3	←	→

Data telah tersimpan di firebase dan bisa di gunakan