Pemrograman Berbasis Framework

“Firebase ReactJS”

Disusun Sebagai:

MATA KULIAH: PEMROGRAMAN BERBASIS FRAMEWORK

Oleh:
AGUS SUBAKTIAR
1741720199
TI – 3B / 05

PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG 2020

1. Index.js

index.js pertemuan11-firebase • src | style.css | firebase.config.js | listArtikel.json pertemuan10-glo

pertemuan11-firebase > src > index.js > App

```
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 "./style.css";
7   import * as firebase from "firebase";
8   import firebaseConfig from "./firebase.config";
9
10  import * as serviceWorker from './serviceWorker';
11
12  firebase.initializeApp(firebaseConfig);
13
14  export const AuthContext = React.createContext(null);
15
16  function App(){
17      const [isLoggedIn, setLoggedIn] = useState(false);
18
19      return(
20          <AuthContext.Provider value={{ isLoggedIn, setLoggedIn}}>
21              Is Logged in? {JSON.stringify(isLoggedIn)}
22              <div className="App">
23                  <Router>
24                      <Header />
25
26                          <Switch>
27                              {routes.map(route => (
28                                  <Route
29                                      key={route.path}
30                                      path={route.path}
31                                      exact={route.exact}
32                                      component={route.main}
33                                      />
34                              ))}
35                          </Switch>
36                      </Router>
37                  </div>
38              </AuthContext.Provider>
39      );
40  }
41
42  const rootElement = document.getElementById("root");
43  ReactDOM.render(<App />, document.getElementById('root'));
44
45  // If you want your app to work offline and load faster, you can change
46  // unregister() to register() below. Note this comes with some pitfalls.
47  // Learn more about service workers: https://bit.ly/CRA-PWA
48  serviceWorker.unregister();
```

2. Login.js

JS Login.js × JS Join.js JS Header.js JS routes.js JS index.js pertemuan11-firebase • src

pertemuan11-firebase > src > JS Login.js > default

```
1   import React, { useState, useContext } from "react";
2   import { AuthContext} from "./index";
3   import * as firebase from "firebase";
4   import { withRouter } from 'react-router-dom'
5
6   const Login = ({history}) => {
7       const [email, setEmail] = useState("");
8       const [password, setPassword] = useState("");
9       const [error, setErrors] = useState("");
10
11      const Auth = useContext(AuthContext);
12
13      const handleForm = e => {
14          e.preventDefault();
15          firebase.auth().signInWithEmailAndPassword(email, password).then(res => {
16              if (res.user) Auth.setLoggedIn(true);
17          })
18          .catch(e => {
19              setErrors(e.message);
20          });
21      };
22
23      const signInWithGoogle = () => {
24          const provider = new firebase.auth.GoogleAuthProvider();
25          firebase
26          .auth()
27          .setPersistence(firebase.auth.Auth.Persistence.SESSION)
28          .then(() => {
29            firebase
30            .auth()
31            .signInWithPopup(provider)
32            .then(result => {
33              console.log(result)
34              history.push('/reports')
```

JS Login.js × JS Join.js JS Header.js JS routes.js JS index.js pertemuan11-firebase • src # style.css JS

pertemuan11-firebase > src > JS Login.js > default

```
35              Auth.setLoggedIn(true)
36            })
37            .catch(e => setErrors(e.message))
38          })
39
40        }
41
42      return (
43          <div>
44              <h1>Login</h1>
45              <form onSubmit={e => handleForm(e)}>
46                  <input
47                      value={email}
48                      onChange={e => setEmail(e.target.value)}
49                      name="email"
50                      type="email"
51                      placeholder="email"
52                      />
53                   <input
54                      onChange={e => setPassword(e.target.value)}
55                      name="password"
56                      value={password}
57                      type="password"
58                      placeholder="password"
59                      />
60                  <hr />
61                  <button onClick={() => signInWithGoogle()} className="googleBtn" type="button">
62                      <img
63                          src="https://upload.wikimedia.org/wikipedia/commons/5/53/Google_%22G%22_Logo.sv
64                          alt="logo"
65                          />
66                          Login With Google
67                  </button>
68
69                   <button type="submit">Login</button>
70                   <span>{error}</span>
71              </form>
72          </div>
73      );
74  };
75
76  export default Login;
```

3. Join.js

pertemuan11-firebase > src > JS Join.js > default

```
import React, { useState, useContext } from "react";
import { AuthContext} from "./index";
import * as firebase from "firebase";
import { withRouter } from 'react-router-dom';

const Join = ({history}) => {
    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 handleGoogleLogin = () => {
        const provider = new firebase.auth.GoogleAuthProvider();

        firebase
        .auth()
        .setPersistence(firebase.auth.Auth.Persistence.SESSION)
          .then(() => {
            firebase
            .auth()
            .signInWithPopup(provider)
            .then(result => {
              console.log(result)
              history.push('/reports')
```

pertemuan11-firebase > src > JS Join.js > default

```
              Auth.setLoggedIn(true)
            })
            .catch(e => setErrors(e.message))
          })

    }
  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 onClick={() => handleGoogleLogin()} className="googleBtn" type="button">
              <img
                  src="https://upload.wikimedia.org/wikipedia/commons/5/53/Google_%22G%22_Logo.svg"
                  alt="logo"
                  />
                  Login With Google
          </button>
          <button type="submit">Login</button>
          <span>{error}</span>
```

```
            <span>{error}</span>
        </form>
    </div>
);
};

export default Join;
```

4. Header.js

pertemuan11-firebase > src > JS Header.js > [@] default

```
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;
```

5. Routes.js

pertemuan11-firebase > src > JS routes.js > [@] default

```
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;
```

6. Firebase.config.js

pertemuan11-firebase > src > JS firebase.config.js > default

```
export default {
    apiKey: "AIzaSyCq-hVeNqtbJA6--ubV9pMyH5NyEXXRgM0",
    authDomain: "agussubaktiar-pbf-3076a.firebaseapp.com",
    databaseURL: "https://agussubaktiar-pbf-3076a.firebaseio.com",
    projectId: "agussubaktiar-pbf-3076a",
    storageBucket: "agussubaktiar-pbf-3076a.appspot.com",
    messagingSenderId: "19447465681",
    appId: "1:19447465681:web:9ab58e0ca927893dbea7f0",
    measurementId: "G-QNNQRKC4FP"
  };
```

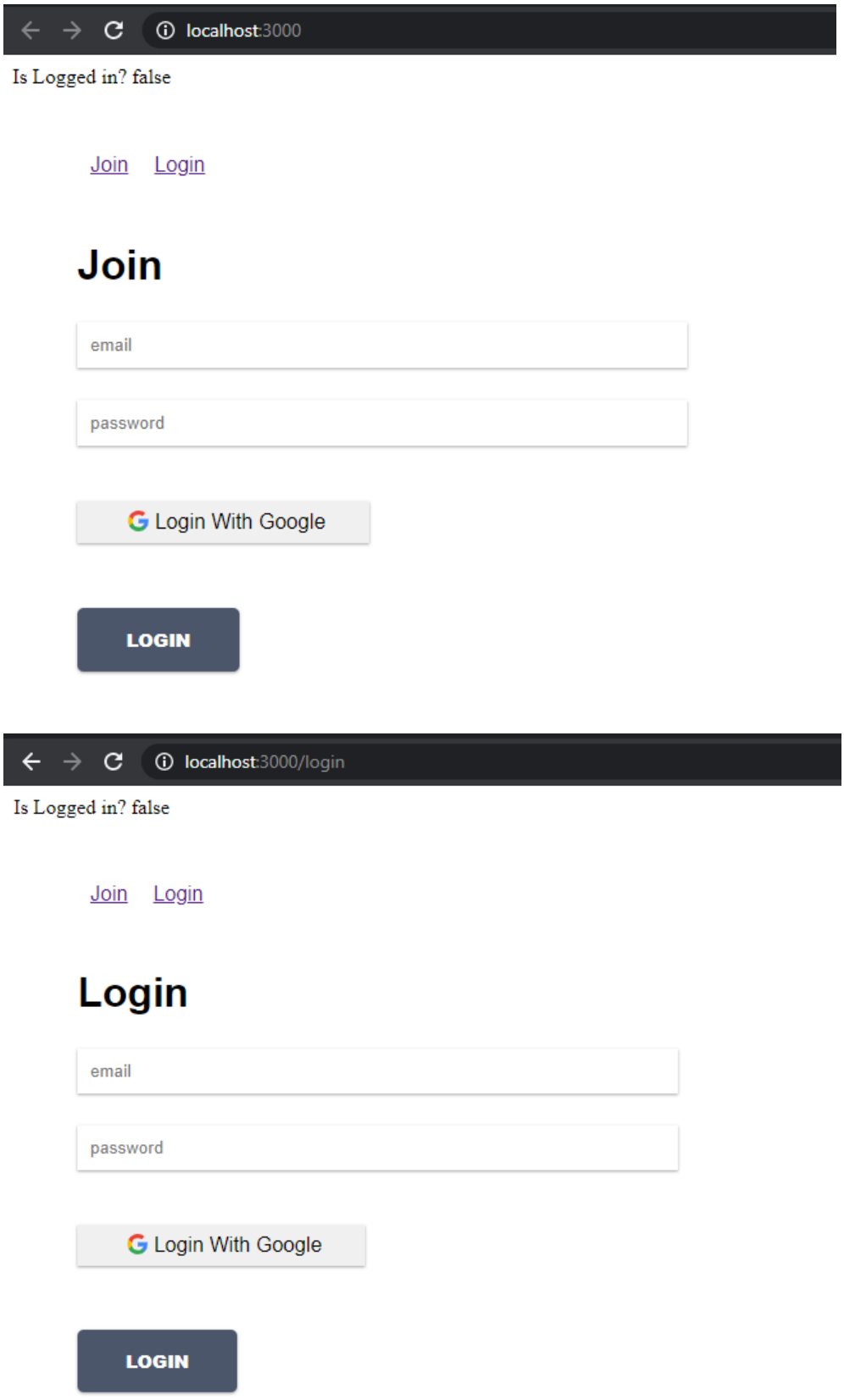
7. Style.css

pertemuan11-firebase > src > # style.css > .nav li

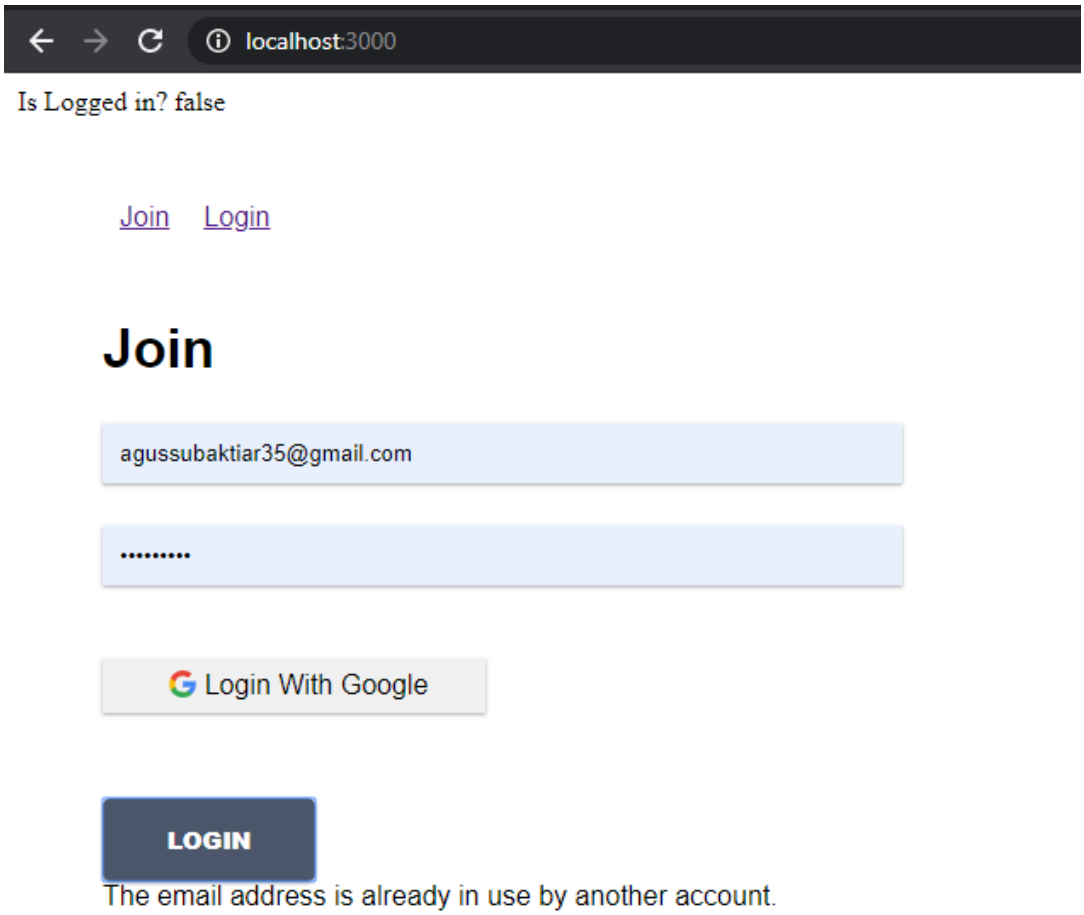
```
.App {
    font-family: sans-serif;
    padding: 50px;
}

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;
}

```

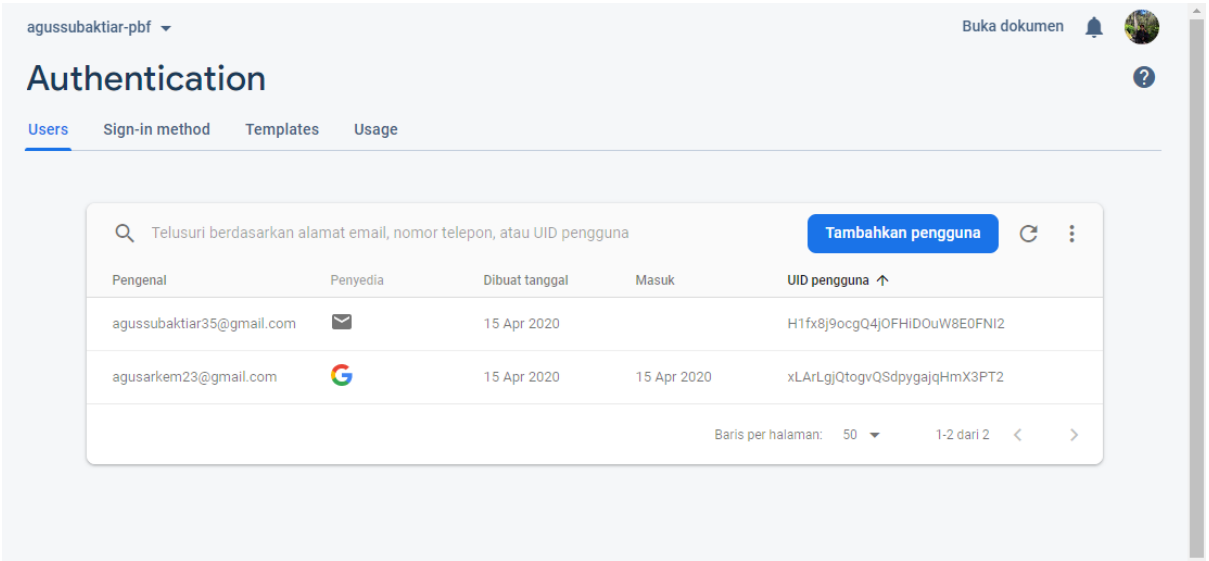
```css
button[type="submit"] {
    background-color: #4c566a;
    color: white;
    font-size: 14px;
    font-weight: 900;
    border-radius: 5px;
    text-transform: uppercase;
    padding: 15px;
}

.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;
    padding: 0;
}


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

8. Output

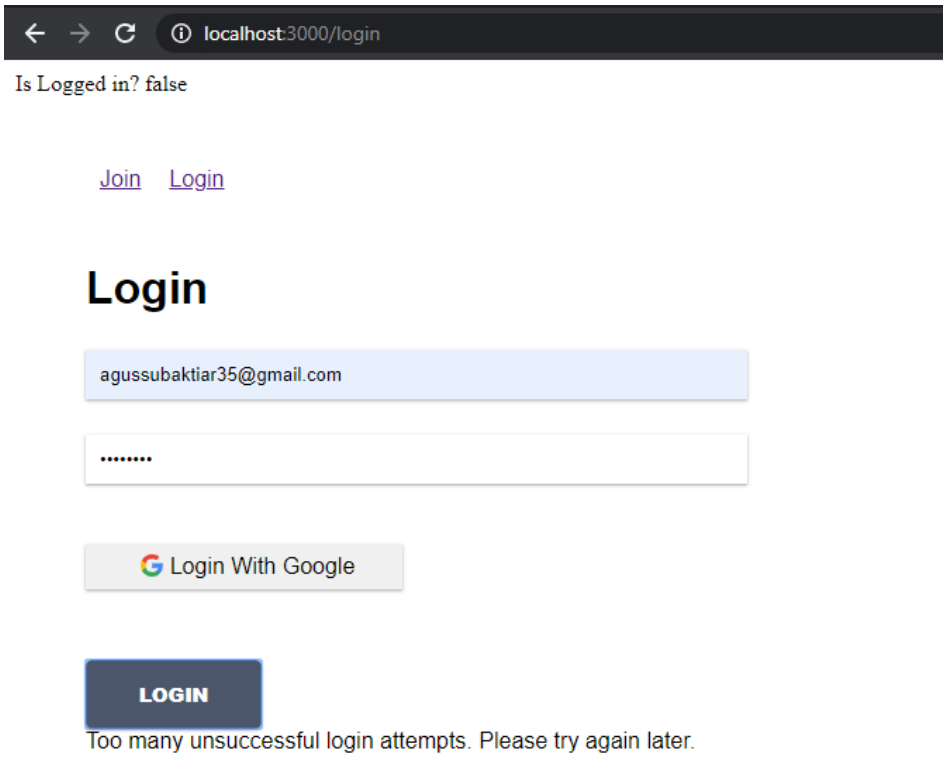

localhost:3000
Is Logged in? false
Join
Login
Join
email
password
Login With Google
LOGIN
localhost:3000/login
Is Logged in? false
Join
Login
Login
email
password
Login With Google
LOGIN

9. Setelah Membuat Firebase, Menyambungkannya, Berhasil
    a. Join Class



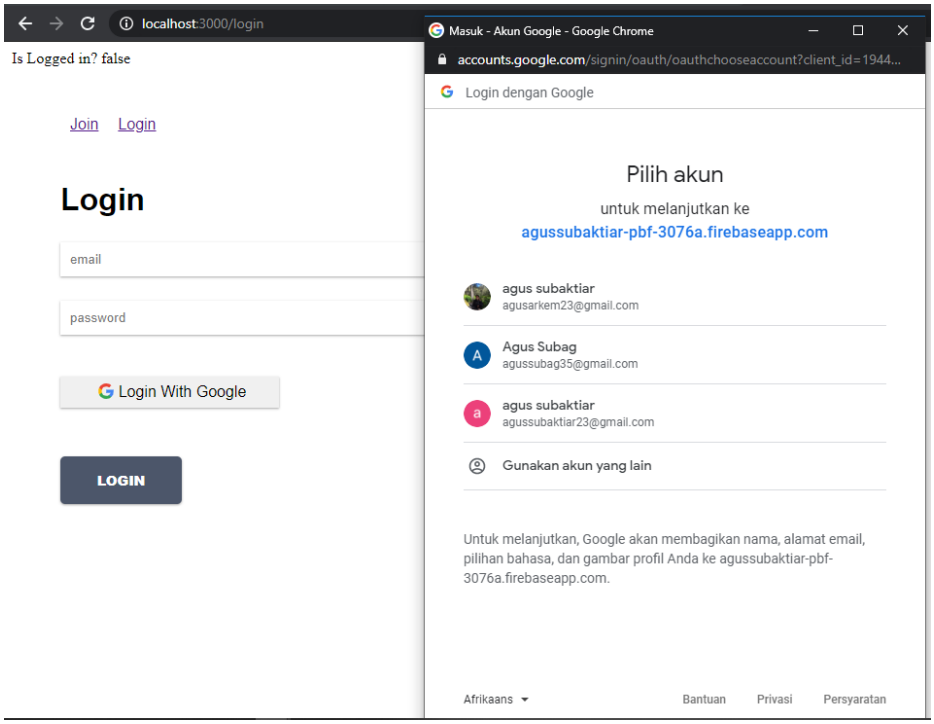


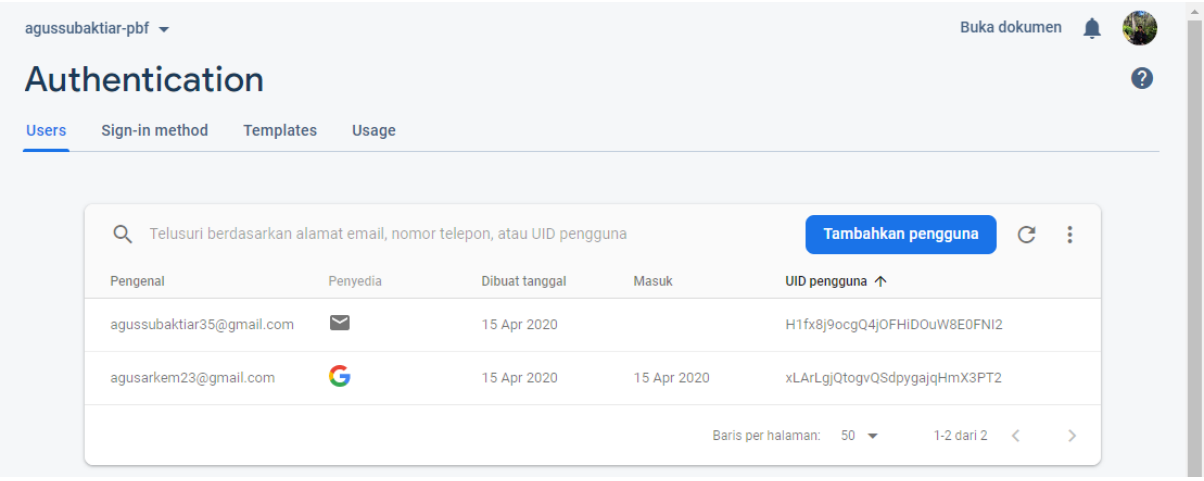
b. Login class



10. Login with Google

Ketika menekan tombol Login With Google maka akan muncul action Pop-Up memilih email untuk mendaftar





Ketika berhasil mendaftar maka akan akun google yang sudah di pilih akan masuk di firebase yang sudah kita buat dan sambungkan.