

TUGAS PERORANGAN/INDIVIDU

LAPORAN TUGAS MINGGU 12

Disusun sebagai

MATA KULIAH : PEMROGRAMAN BERBASIS FRAMEWORK

Oleh :

Izmir Ta'sis Khasna Lugazi 1741720180

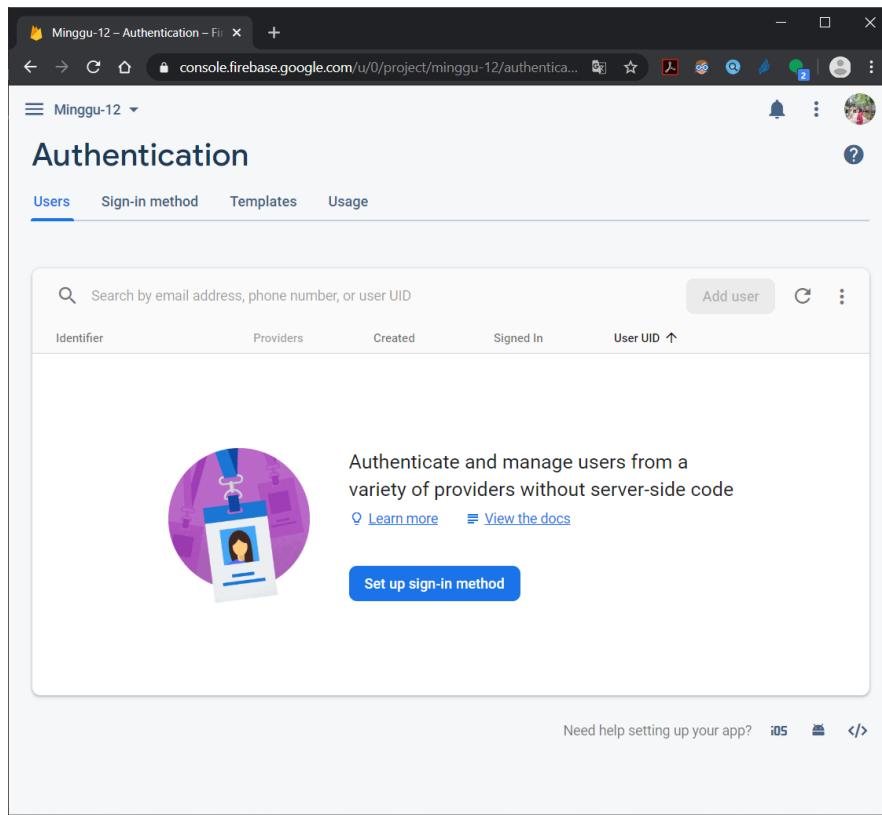
Kelas :

3B



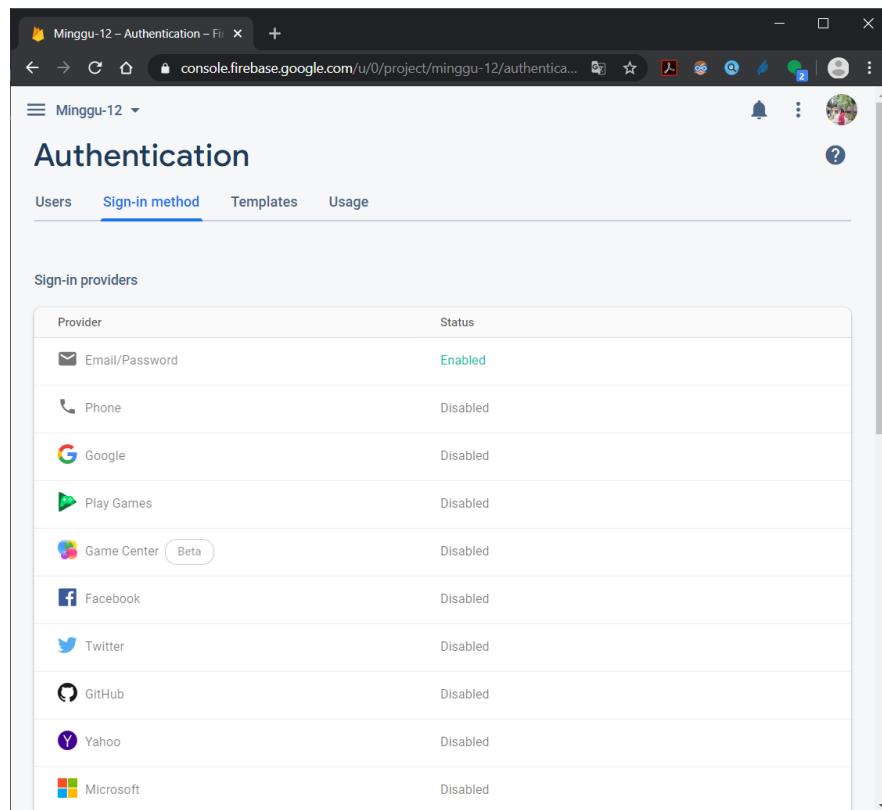
**PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2020**

Setting firebase console terlebih dahulu



The screenshot shows the Firebase Authentication console interface. At the top, there's a navigation bar with tabs for 'Users', 'Sign-in method', 'Templates', and 'Usage'. Below the navigation bar is a search bar and a table header with columns for 'Identifier', 'Providers', 'Created', 'Signed In', and 'User UID'. A large central area features a purple circular icon with a user profile and a 'Set up sign-in method' button. Text below the icon reads: 'Authenticate and manage users from a variety of providers without server-side code'. It includes links for 'Learn more' and 'View the docs'. At the bottom right, there's a help section for iOS and Android development.

Pastikan Email/Password enabled



The screenshot shows the Firebase Authentication console interface with the 'Sign-in method' tab selected. The page title is 'Authentication'. Below the title, there's a section for 'Sign-in providers' with a table. The table has two columns: 'Provider' and 'Status'. The data is as follows:

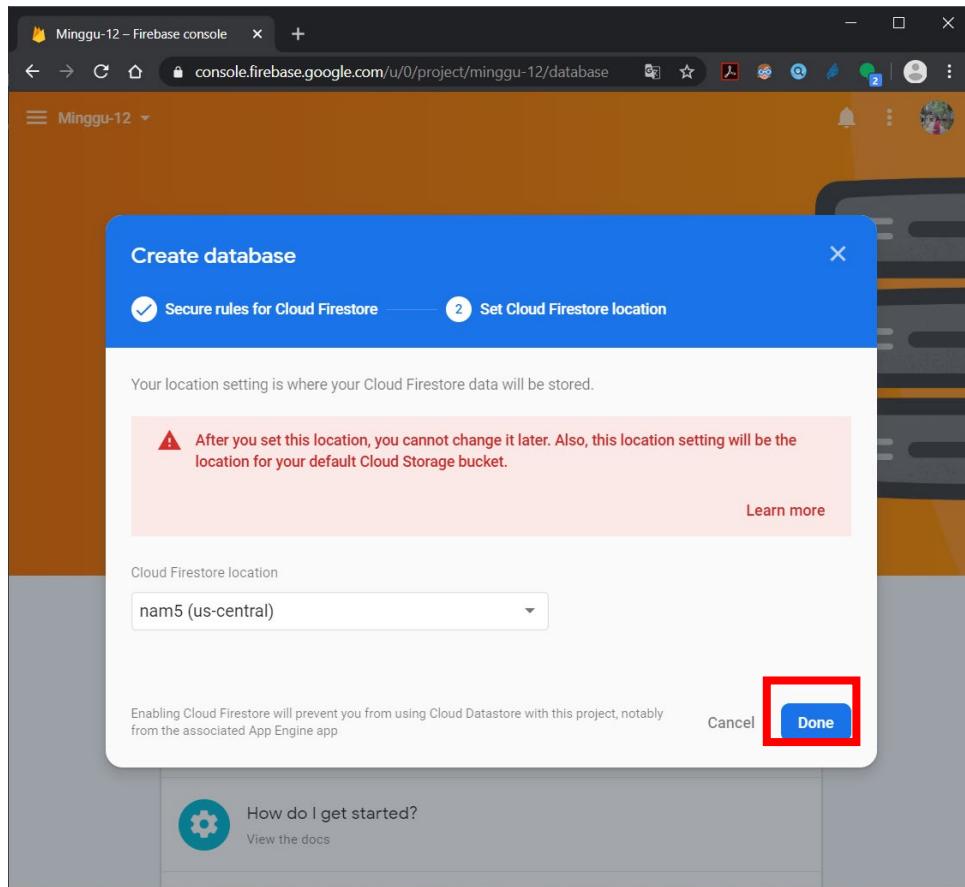
Provider	Status
Email/Password	Enabled
Phone	Disabled
Google	Disabled
Play Games	Disabled
Game Center	Beta
Facebook	Disabled
Twitter	Disabled
GitHub	Disabled
Yahoo	Disabled
Microsoft	Disabled

Create database dan ikuti sesuai Langkah dibawah

The screenshot shows the 'Create database' dialog box from the Firebase console. It has two main steps: 1. Secure rules for Cloud Firestore and 2. Set Cloud Firestore location. Step 1 is active. A note says: 'After you define your data structure, you will need to write rules to secure your data.' Below it are two options: 'Start in production mode' and 'Start in test mode'. The 'Start in test mode' option is selected and highlighted with a red box. To its right is a code snippet for security rules:

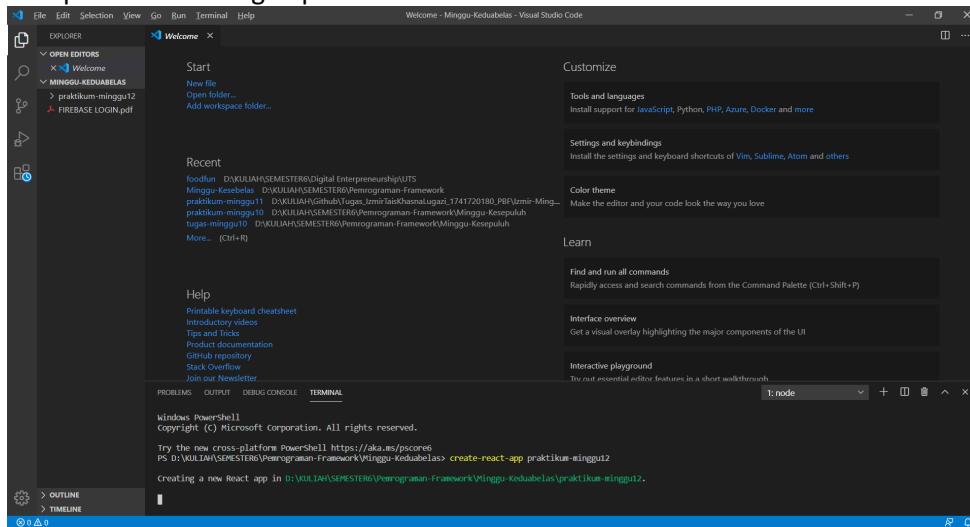
```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read, write: if
        request.time < timestamp.date(2020, 5, 23);
    }
  }
}
```

A warning message below the code says: 'Anyone with your database reference will be able to read or write to your database for 30 days'. At the bottom left of the dialog, there's a note: 'Enabling Cloud Firestore will prevent you from using Cloud Datastore with this project, notably from the associated App Engine app'. At the bottom right are 'Cancel' and 'Next' buttons, with the 'Next' button highlighted with a red box.



Setting react app dan firebase

1. Buat aplikasi baru dengan perintah



2. Tuliskan “npm start”

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following text:

```
Compiled successfully!
You can now view praktikum-minggu12 in the browser.
http://localhost:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
```

3. Ikuti sesuai Langkah di bawah dengan menuliskan “npm install -g firebase-tools”

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following text:

```
http://localhost:3000
Note that the development build is not optimized.
To create a production build, use npm run build.
Terminate batch job (Y/N)? y
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\praktikum-minggu12> npm install -g firebase-tools
[1/1] ⠄ rollbackFailedOptional: verb npm-
```

4. Selanjutnya tuliskan “firebase login”, pastikan sukses

The screenshot shows the Visual Studio Code interface with the terminal tab active. The terminal window displays the following text:

```
+ firebase-tools@8.1.1
added 531 packages from 353 contributors in 224.77ms
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\praktikum-minggu12> firebase login
1 Firebase optionally collects CLI usage and error reporting information to help improve our products. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to identify you.
? Allow Firebase to collect CLI usage and error reporting information? Yes
? To change your data collection preference at any time, run 'firebase logout' and log in again.
visit this URL on this device to log in:
```

5. Tahap selanjutnya ketikkan “firebase init”

The screenshot shows the Visual Studio Code interface with the terminal tab active. The command 'firebase init' has been run, and the output shows:

```
+ Success! Logged in as izmirtaiz@gmail.com
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Minggu-KeduaBelas\praktikum-minggu1> firebase init
```

Below the terminal, there is a digital art representation of the word "FIREBASE" made of small squares.

Pilih Firestore dan Hosting

The screenshot shows the Visual Studio Code interface with the terminal tab active. A message asks if the user is ready to proceed:

```
? Are you ready to proceed? [y/n]
? Most features require you to want to set up for this folder? Press Space to select features, then Enter to confirm your choices.
```

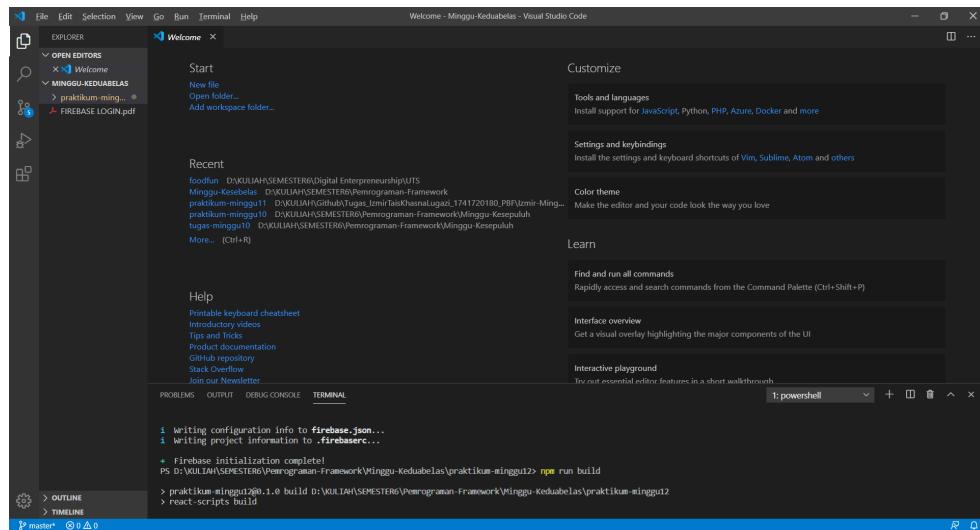
Below the terminal, there is a digital art representation of the word "FIREBASE" made of small squares.

Pastikan sukses

The screenshot shows the Visual Studio Code interface with the terminal tab active. A message confirms the setup was successful:

```
? What file should be used for Firestore indexes? firestore.indexes.json
== Hosting Setup
Your public directory is the folder (relative to your project directory) that have a build process for your assets, use your build's output directory.
? What do you want to use as your public directory? public
? Configure as a single-page app (rewrite all urls to /index.html)? (y/n) y
```

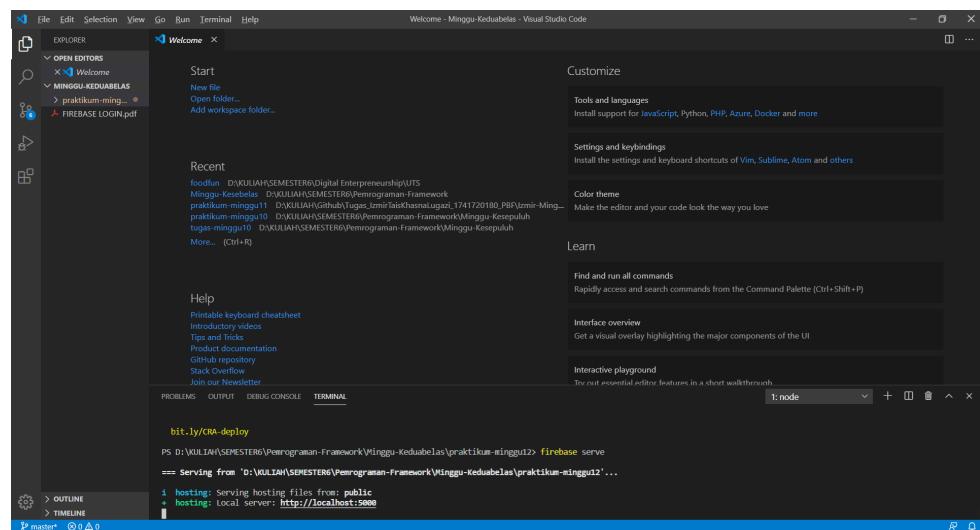
6. Coba dengan ketik “npm run build”



The screenshot shows the Visual Studio Code interface with the terminal tab active. The command `npm run build` was entered, and the output shows the build process for a React application named `praktikum-minggu12`. The terminal output is as follows:

```
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12> npm run build
> react-scripts build
> praktikum-minggu12@0.1.0 build D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12
+-- react-scripts@4.0.3
`-- react-scripts-build@1.0.0
```

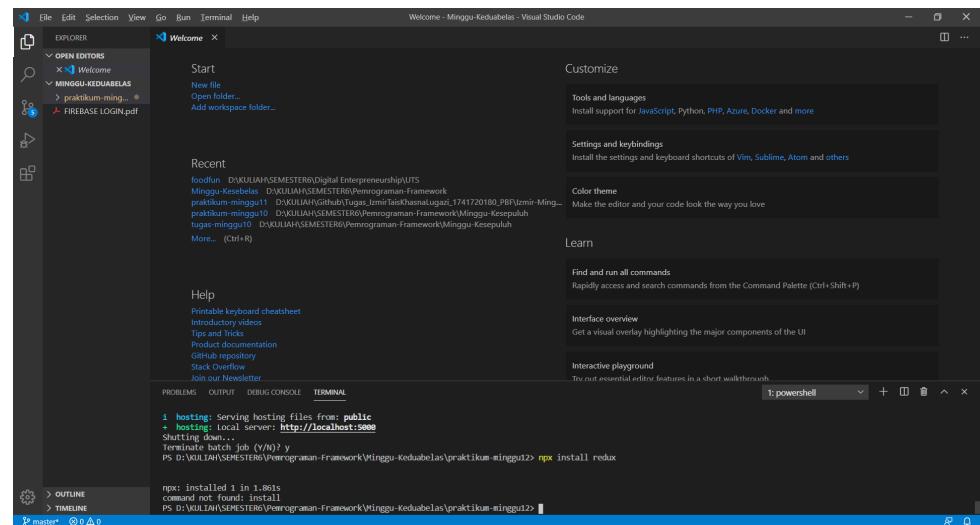
7. Lanjutkan step dengan mengetik “firebase serve”



The screenshot shows the Visual Studio Code interface with the terminal tab active. The command `firebase serve` was entered, and the output shows the hosting configuration and the local server URL (`http://localhost:3008`). The terminal output is as follows:

```
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12> firebase serve
== Serving from 'D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12'...
+-- hosting: Serving hosting files from: public
+-- hosting: Local server: http://localhost:3008
```

8. Selanjutnya install redux



The screenshot shows the Visual Studio Code interface with the terminal tab active. The command `npx install redux` was entered, and the output shows the installation of the `redux` package. The terminal output is as follows:

```
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12> npx install redux
npm: installed 1 in 1.895s
Command not found: install
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12>
```

9. Install react-redux dan pastikan berhasil

```
+ hosting: local server: http://localhost:5000
Shutting down...
Terminate batch job (Y/n)? y
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Minggu-Kedabelas\praktikum-minggu12> npm install redux

rex: installed 1 in 1s
command not found: instal
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Minggu-Kedabelas\praktikum-minggu12> npm install react-redux
[1/4] ⠄ [██████████] | LoadExtraneous: 511 resolvewithnewmodule hostis non-react-statics@3.3.2 checking installable status
```

10. Install redux-thunk

```
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.12: wanted {"os": "darwin", "arch": "any"} (current: {"os": "win32", "arch": "x64"})
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.12 (node modules\jest-haste-map\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.12: wanted {"os": "darwin", "arch": "any"} (current: {"os": "win32", "arch": "x64"})
+ redux-thunk@2.0
added 2 packages from 3 contributors and audited 931414 packages in 17.166s
found 0 vulnerabilities
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Minggu-Kedabelas\praktikum-minggu12> npm install redux-thunk
[1/4] ⠄ [██████████] | LoadExtraneous: 511 resolvewithnewmodule redux-thunk@2.3.0 checking installable status
```

11. Install react-redux-dom

```
+ redux-thunk@2.3.0
added 1 package from 1 contributor and audited 931415 packages in 14.995s
found 0 vulnerabilities
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Minggu-Kedabelas\praktikum-minggu12> npm install react-router-dom
npm WARN react-router@7.0.0 requires a peer of redux@>0.0.0 || ^3.0.0-0 but none is installed. You must install peer dependencies yourself.
npm WARN react-script@0.4.1 requires a peer of typescript@>3.2.1 but none is installed. You must install peer dependencies yourself.
npm WARN react-sass-loader@4.0.0 requires a peer of node-sass@>4.0.0 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of sass@>1.3.0 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of fiber@> 3.1.0 but none is installed. You must install peer dependencies yourself.
```

12. Install firebase

The screenshot shows the Visual Studio Code interface with the terminal tab active. The command `npm install firebase` was run, and the output shows several dependency installation errors:

```
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12> npm install firebase
> protobufjs@6.9.0 postinstall D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12\node_modules\protobufjs\scripts\postinstall
npm WARN react-reduxbus@2.0.0 requires a peer of redux@>2.0.0 || ^3.0.0 || ^4.0.0-0 but none is installed. You must install peer dependencies yourself.
npm WARN react-select@3.4.1 requires a peer of typescript@>3.2.1 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of node-sass@>4.0.0 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of sass@>1.3.0 but none is installed. You must install peer dependencies yourself.
npm WARN sass-loader@8.0.2 requires a peer of fibers@> 3.1.0 but none is installed. You must install peer dependencies yourself.
```

13. Install UI library

The screenshot shows the Visual Studio Code interface with the terminal tab active. The command `npm install @material-ui/core` was run, and the output shows the package being added and a warning about a deprecated package:

```
PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12> npm install @material-ui/core
+ firebase@7.14.1
added 54 packages from 17 contributors and audited 931651 packages in 198.108s
found 0 vulnerabilities

PS D:\KULIAH\SEMESTER6\Pemrograman-Framework\Winggu-KeduaBelas\praktikum-minggu12> npm install @material-ui/core
npm WARN deprecated popper.js@1.16.1: You can find the new Popper v2 at @popperjs/core, this package is dedicated to the legacy v1
[...]
npm WARN deprecated [REDACTED] / loadixtraneous: [REDACTED] resolveWithNewModule css-vendor@0.8 checking installable status
```

14. Paste kan file public/index.html

The screenshot shows the Visual Studio Code interface with the file `public/index.html` open in the editor. The file contains the standard Create React App index HTML template, including the DOCTYPE declaration, head, and body sections with various meta tags and links.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta name="description" content="Web site created using create-react-app" />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
  </head>
  <!--
    manifest.json provides metadata used when your web app is installed on a user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
  -->
  <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
  <!--
    Notice the use of %PUBLIC_URL% in the tags above.
    It will be replaced with the URL of the public folder during the build.
    Only files inside the public folder can be referenced from the HTML.
  -->
  <link rel="stylesheet" href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap" />
  <title>React App</title>
</head>
<body>
  <noscript>You need to enable JavaScript to run this app.</noscript>
</body>
```

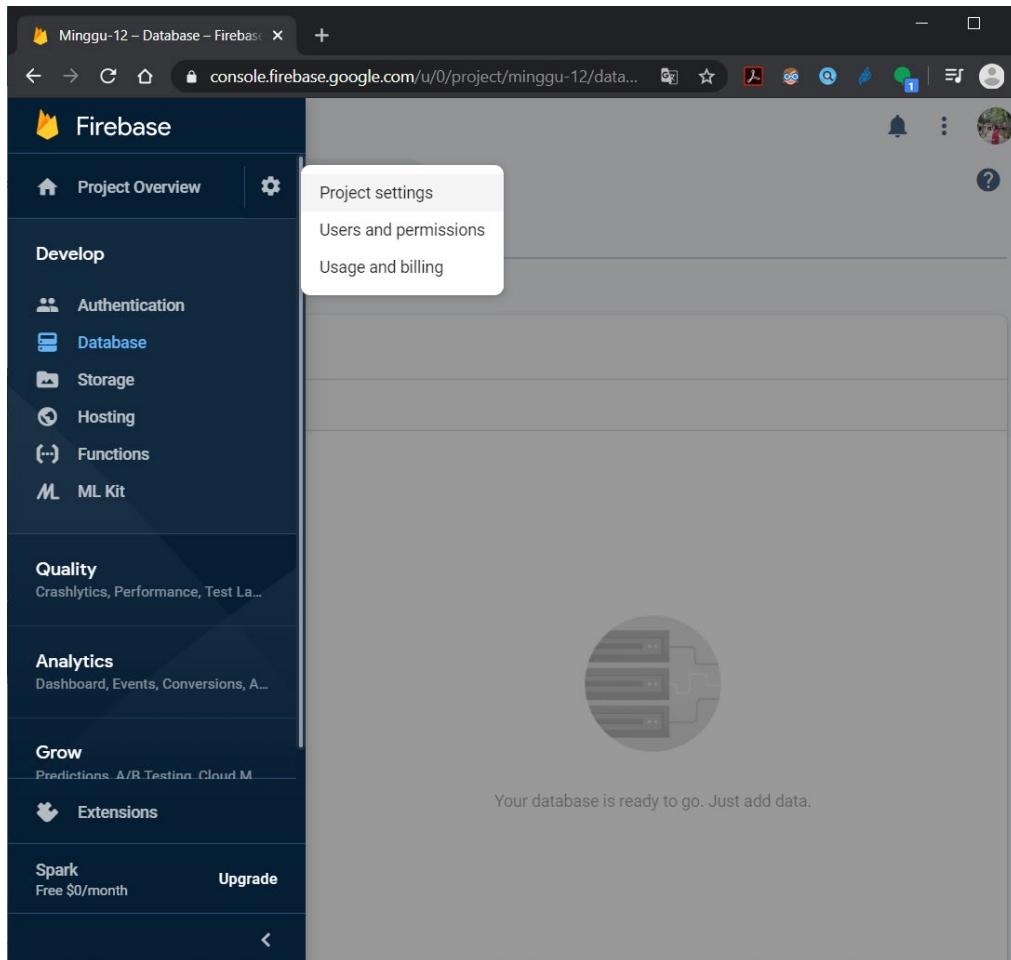
15. Install material icon

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer:** Shows the project structure with files like `index.html`, `favicon.ico`, `logos182.png`, `logos512.png`, `manifest.json`, `robots.txt`, `src`, `firebase`, `gitignore`, `firebase.json`, `firestore.rules`, `package-lock.json`, `package.json`, and `README.md`.
- Editor:** The `index.html` file is open, displaying code that includes imports for Material UI CSS from `https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&display=swap`.
- Terminal:** The terminal shows the command `npm install @material-ui/icons` being run.

Setting Firebase Config

1. Pilih project setting



Minggu-12 – Settings – Firebase

console.firebaseio.google.com/u/0/project/minggu-12/setting

Web API Key: AlzaSyAiRs_ov91O8lpgROOp_YowyDZeBjq-xqQ

Public settings

These settings control instances of your project shown to the public

Public-facing name: project-208265740640

Support email: Not configured

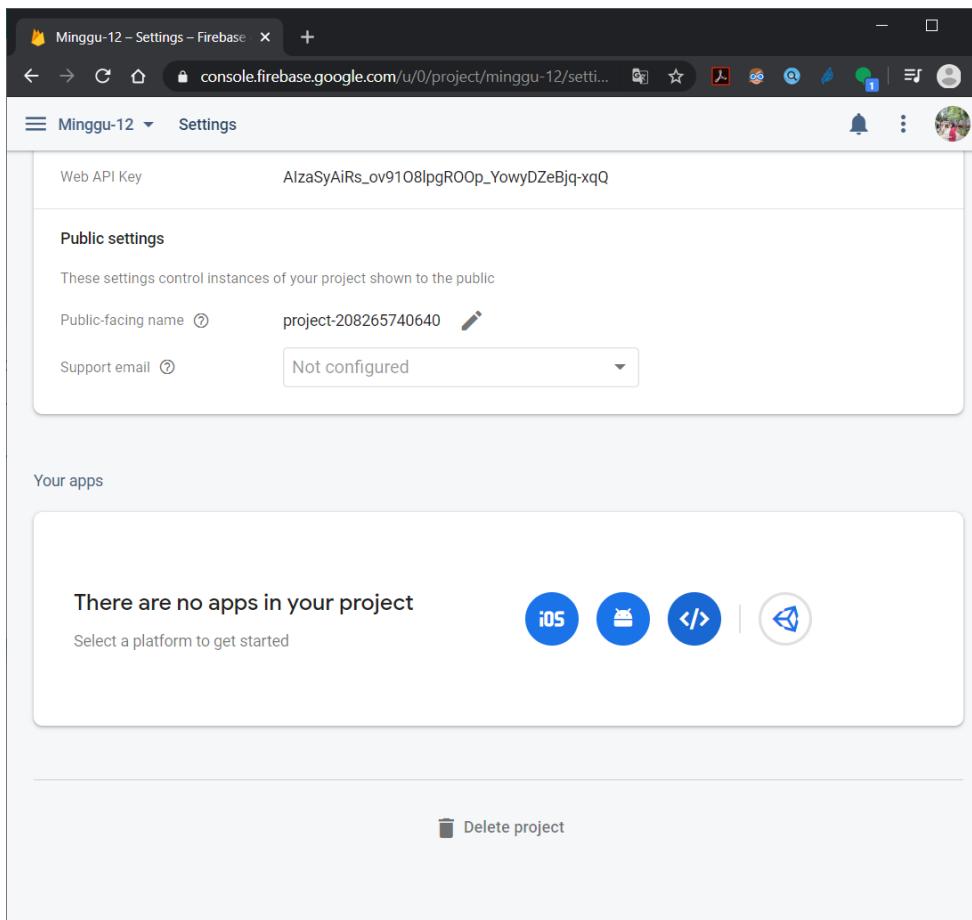
Your apps

There are no apps in your project

Select a platform to get started

iOS Android Web Cloud Functions

Delete project



Minggu-12 – Settings – Firebase

console.firebaseio.google.com/u/0/project/minggu-12/setting

Go to doc

x Add Firebase to your web app

1 Register app

App nickname: firebase-login-app

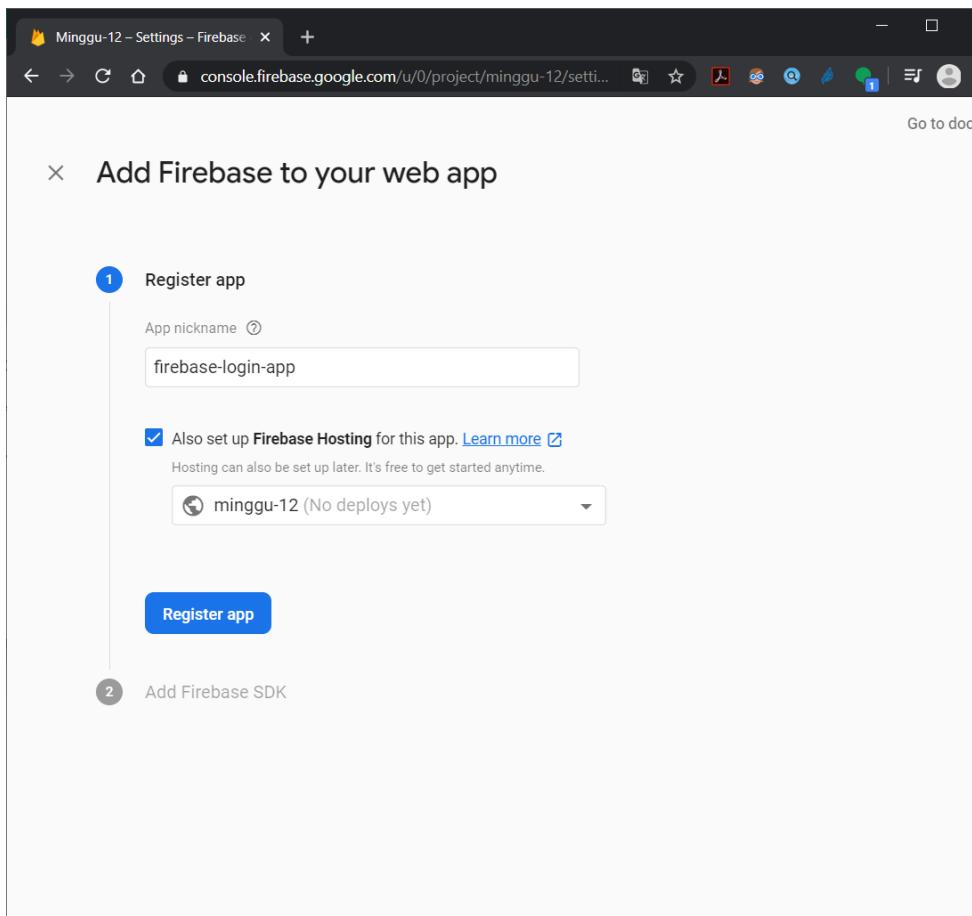
Also set up **Firebase Hosting** for this app. [Learn more](#)

Hosting can also be set up later. It's free to get started anytime.

minggu-12 (No deploys yet)

Register app

2 Add Firebase SDK



2. Pilih Config

The screenshot shows the Firebase console interface. On the left, there's a sidebar with 'Minggu-12' and 'Settings'. The main area is titled 'Web apps' and shows a list with one item: 'firebase-login-app'. To the right of the list, there are fields for 'App nickname' (set to 'firebase-login-app'), 'App ID' (set to '1:208265740640:web:2576b536d198cc29e21e05'), and 'Linked Firebase Hosting site' (set to 'minggu-12'). Below these, there's a section for 'Firebase SDK snippet' with three radio button options: 'Automatic', 'CDN', and 'Config' (which is selected). A code block displays the configuration script:

```
const firebaseConfig = {
  apiKey: "AIzaSyAiRs_ov9108lpgR0Op_YowyDZeBjq-xQQ",
  authDomain: "minggu-12.firebaseio.com",
  databaseURL: "https://minggu-12.firebaseio.com",
  projectId: "minggu-12",
  storageBucket: "minggu-12.appspot.com",
  messagingSenderId: "208265740640",
  appId: "1:208265740640:web:2576b536d198cc29e21e05",
  measurementId: "G-8T5RFDF81H"
};
```

3. Edit firebase/firebase.js

The screenshot shows a Visual Studio Code window with a dark theme. The left sidebar shows a project structure with files like 'index.html', 'firebase.js', 'App.css', etc. The main editor tab is 'firebase.js' with the following content:

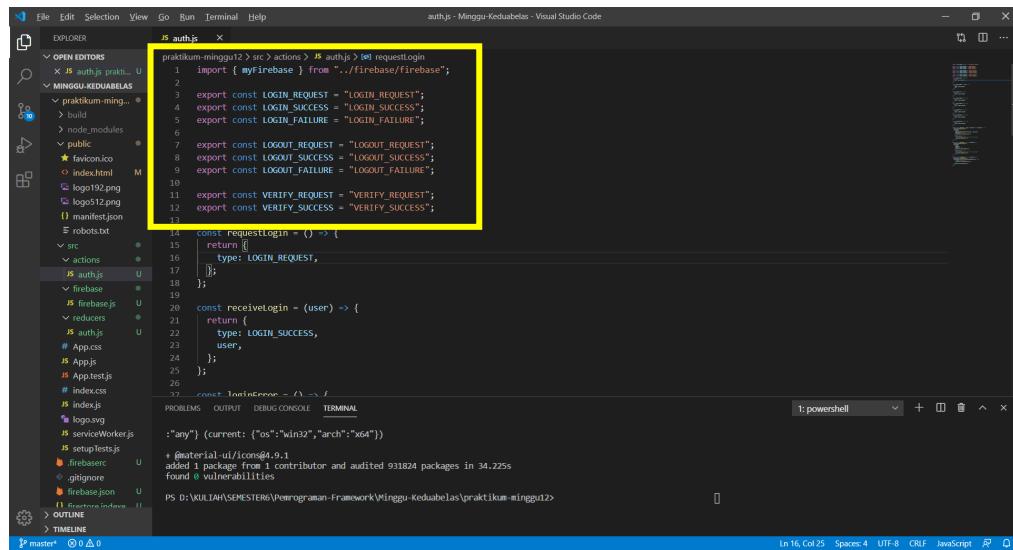
```
const firebaseConfig = {
  apiKey: "AIzaSyAiRs_ov9108lpgR0Op_YowyDZeBjq-xQQ",
  authDomain: "minggu-12.firebaseio.com",
  databaseURL: "https://minggu-12.firebaseio.com",
  projectId: "minggu-12",
  storageBucket: "minggu-12.appspot.com",
  messagingSenderId: "208265740640",
  appId: "1:208265740640:web:2576b536d198cc29e21e05",
  measurementId: "G-8T5RFDF81H"
};

export const myFirebase = firebase.initializeApp(firebaseConfig);
const baseDB = myFirebase.firestore();
export const db = baseDB;
```

The terminal at the bottom shows some npm audit results:

```
PS D:\VULIAH\SEMESTER6\Pemrograman Framework\Praktikum-KeduaBelas\praktikum-minggu2> npm audit
npm audit v1.1.0
added 1 package from 1 contributor and audited 931824 packages in 34.22s
found 0 vulnerabilities
```

4. Edit actions/auth.js



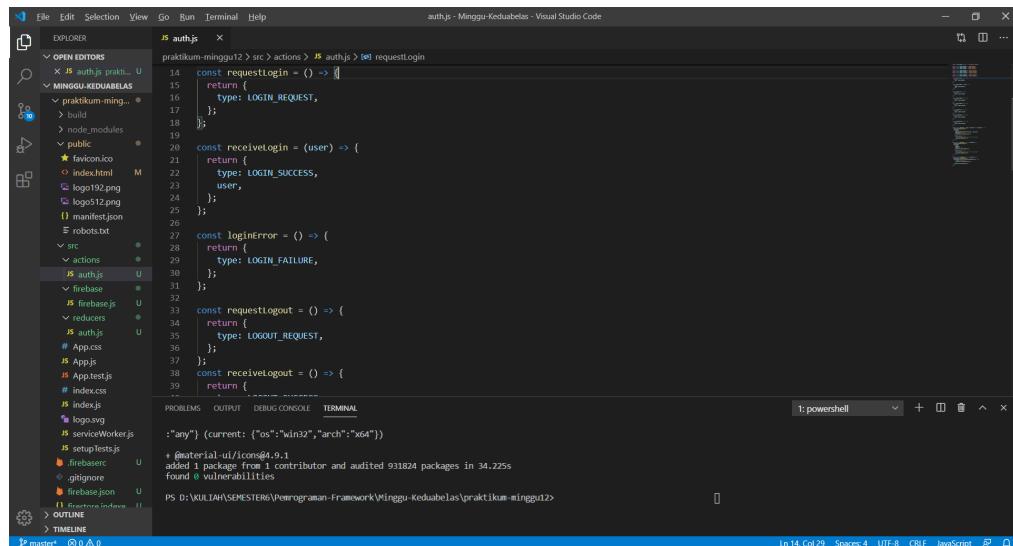
```
praktikum-mingut2> src> actions > JS auth.js > [o] requestLogin
import { myFirebase } from './firebase/firebase';

export const LOGIN_REQUEST = "LOGIN_REQUEST";
export const LOGIN_SUCCESS = "LOGIN_SUCCESS";
export const LOGIN_FAILURE = "LOGIN_FAILURE";

export const LOGOUT_REQUEST = "LOGOUT_REQUEST";
export const LOGOUT_SUCCESS = "LOGOUT_SUCCESS";
export const LOGOUT_FAILURE = "LOGOUT_FAILURE";

export const VERIFY_REQUEST = "VERIFY_REQUEST";
export const VERIFY_SUCCESS = "VERIFY_SUCCESS";
```

5. User requesting to login masih dalam actions/auth.js



```
praktikum-mingut2> src> actions > JS auth.js > [o] requestLogin
const requestLogin = () => {
  return {
    type: LOGIN_REQUEST,
  };
};

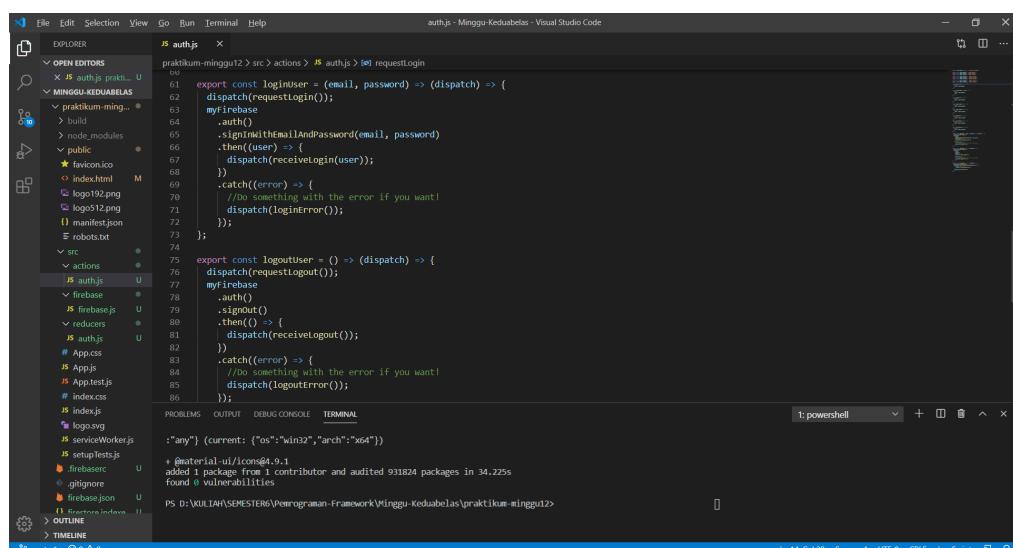
const receiveLogin = (user) => {
  return {
    type: LOGIN_SUCCESS,
    user,
  };
};

const loginError = () => {
  return {
    type: LOGIN_FAILURE,
  };
};

const requestLogout = () => {
  return {
    type: LOGOUT_REQUEST,
  };
};

const receiveLogout = () => {
  return {
    type: LOGOUT_SUCCESS,
  };
};
```

6. Edit pada bagian loginUser()



```
praktikum-mingut2> src> actions > JS auth.js > [o] requestLogin
export const loginUser = (email, password) => (dispatch) => {
  dispatch(requestLogin());
  myFirebase
    .auth()
    .signInWithEmailAndPassword(email, password)
    .then((user) => {
      dispatch(receiveLogin(user));
    })
    .catch((error) => {
      //Do something with the error if you want!
      dispatch(loginError());
    });
};

export const logoutUser = () => (dispatch) => {
  dispatch(requestLogout());
  myFirebase
    .auth()
    .signOut()
    .then(() => {
      dispatch(receiveLogout());
    })
    .catch((error) => {
      //Do something with the error if you want!
      dispatch(logoutError());
    });
};
```

7. Buka file auth.js pada folder reducers

```

praktikum-minggu12 > src > reducers > JS auth.js > default
1 import {
2   LOGIN_REQUEST,
3   LOGIN_SUCCESS,
4   LOGIN_FAILURE,
5   LOGOUT_REQUEST,
6   LOGOUT_SUCCESS,
7   LOGOUT_FAILURE,
8   VERIFY_REQUEST,
9   VERIFY_SUCCESS,
10 } from './actions/auth';
11
12 export default (
13   state = {
14     isLoggingIn: false,
15     isLoggingOut: false,
16     isVerifying: false,
17     loginError: false,
18     logoutError: false,
19     isAuthenticated: false,
20     user: {}
21   },
22   action
23 ) => {
24   switch (action.type) {
25     case LOGIN_REQUEST:
26       return {
27       ...state,
28       isLoggingIn: true,
29       loginError: false,
30     };
31 }

```

```

praktikum-minggu12 > src > reducers > JS auth.js > default
31
32   case LOGIN_SUCCESS:
33     return [
34       ...state,
35       isLoggingIn: false,
36       isAuthenticated: true,
37       user: action.user,
38     ];
39
40   case LOGIN_FAILURE:
41     return [
42       ...state,
43       isLoggingIn: false,
44       isAuthenticated: false,
45       loginError: true,
46     ];
47
48   case LOGOUT_REQUEST:
49     return [
50       ...state,
51       isLoggingOut: true,
52       logoutError: false,
53     ];
54
55   case LOGOUT_SUCCESS:
56     return [
57       ...state,
58       isLoggingOut: false,
59       isAuthenticated: false,
60       user: {}
61     ];
62
63   case LOGOUT_FAILURE:
64     return [
65       ...state,
66       isLoggingOut: false,
67       logoutError: true,
68     ];
69
70   case VERIFY_REQUEST:
71     return [
72       ...state,
73       isVerifying: true,
74       verifyingError: false,
75     ];
76
77   case VERIFY_SUCCESS:
78     return [
79       ...state,
80       isVerifying: false,
81     ];
82   default:
83     return state;
84   }
85 }

```

```

praktikum-minggu12 > src > reducers > JS auth.js > default
85
86
87   case LOGIN_REQUEST:
88     return [
89       ...state,
90       isLoggingIn: true,
91       loginError: false,
92     ];
93
94   case LOGIN_SUCCESS:
95     return [
96       ...state,
97       isLoggingIn: false,
98       isAuthenticated: true,
99       user: action.user,
100     ];
101
102   case LOGIN_FAILURE:
103     return [
104       ...state,
105       isLoggingIn: false,
106       loginError: true,
107     ];
108
109   case LOGOUT_REQUEST:
110     return [
111       ...state,
112       isLoggingOut: true,
113       logoutError: false,
114     ];
115
116   case LOGOUT_SUCCESS:
117     return [
118       ...state,
119       isLoggingOut: false,
120       isAuthenticated: false,
121       user: {}
122     ];
123
124   case LOGOUT_FAILURE:
125     return [
126       ...state,
127       isLoggingOut: false,
128       logoutError: true,
129     ];
130
131   case VERIFY_REQUEST:
132     return [
133       ...state,
134       isVerifying: true,
135       verifyingError: false,
136     ];
137
138   case VERIFY_SUCCESS:
139     return [
140       ...state,
141       isVerifying: false,
142     ];
143   default:
144     return state;
145   }
146 }

```

8. Edit reducers/index.js

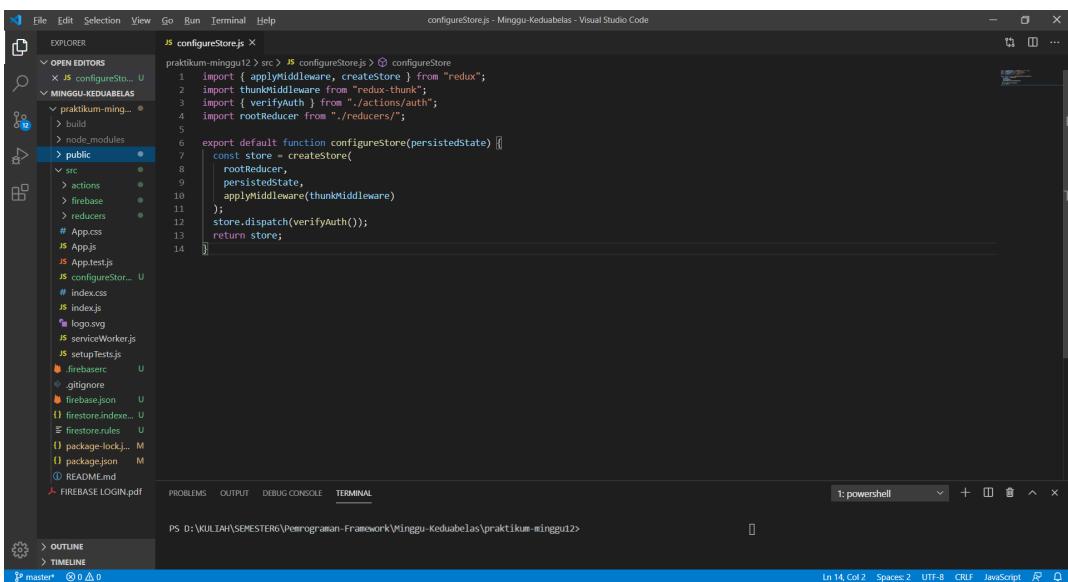
```

praktikum-minggu12 > src > reducers > JS index.js > default
1 import { combineReducers } from "redux";
2 import auth from "./auth";
3
4 export default combineReducers({ auth });

```

9. Setting redux plumbing

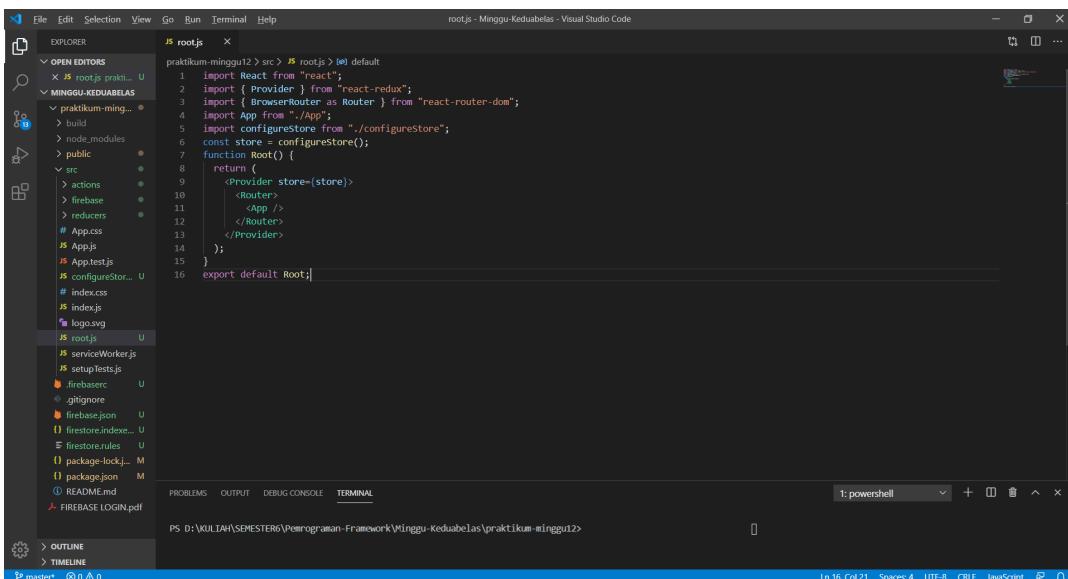
10. Pada folder src buatlah file configureStore.js



The screenshot shows the Visual Studio Code interface with the file `configureStore.js` open in the editor. The code defines a function `configureStore` that creates a store using `createStore` from `redux`. It applies middleware like `applyMiddleware`, `createStore`, `thunkMiddleware`, and `verifyAuth`. It also includes a `rootReducer` and dispatches `verifyAuth`.

```
praktikum-minggu12 > src > JS configureStore.js > configureStore
1 import { applyMiddleware, createStore } from "redux";
2 import thunkMiddleware from "redux-thunk";
3 import { verifyAuth } from "./actions/auth";
4 import rootReducer from "./reducers";
5
6 export default function configureStore(persistedState) {
7   const store = createStore(
8     rootReducer,
9     persistedState,
10    applyMiddleware(thunkMiddleware)
11  );
12  store.dispatch(verifyAuth());
13  return store;
14}
```

11. Pada folder src buatlah file root.js



The screenshot shows the Visual Studio Code interface with the file `root.js` open in the editor. The code defines a component `Root` that wraps an `App` component with a `Provider` component from `react-redux`, passing the store as a prop.

```
praktikum-minggu12 > src > JS root.js > default
1 import React from "react";
2 import { Provider } from "react-redux";
3 import { BrowserRouter as Router } from "react-router-dom";
4 import App from "./App";
5 import configureStore from "./configureStore";
6 const store = configureStore();
7 function Root() {
8   return (
9     <Provider store={store}>
10       <Router>
11         <App />
12       </Router>
13     </Provider>
14   );
15 }
16 export default Root;
```

12. Pada folder src buatlah file index.js

```

File Edit Selection View Go Run Terminal Help
indexjs - praktikum-minggu12 - Visual Studio Code
OPEN EDITORS
  JS index.js
  src > JS index.js
    1 import React from 'react';
    2 import ReactDOM from 'react-dom';
    3 import './index.css';
    4 import App from './App';
    5 import * as serviceWorker from './servicesWorker';
    6
    7 ReactDOM.render(
    8   <React.StrictMode>
    9     <App />
   10   </React.StrictMode>;
   11   document.getElementById('root')
   12 );
   13
   14 // If you want your app to work offline and load faster, you can change
   15 // unregister() to register() below. Note this comes with some pitfalls.
   16 // Learn more about service workers: https://bit.ly/CRA-PWA
   17 serviceWorker.unregister();
   18

```

master 0 0 0 0 Initializing ES/TS language features In 18, Col 1 Spaces: 2 UTF-8 LF JavaScript

13. Buatlah coding App.js

```

File Edit Selection View Go Run Terminal Help
App.js - praktikum-minggu12 - Visual Studio Code
OPEN EDITORS
  JS App.js
  src > JS App.js
    1 import React from "react";
    2 import { Route, Switch } from "react-router-dom";
    3 import { connect } from "react-redux";
    4 import ProtectedRoute from "./components/ProtectedRoute";
    5 import Home from "./components/Home";
    6 import Login from "./components/Login";
    7
    8 function App(props) {
    9   const { isAuthenticated, isVerifying } = props;
   10   return (
   11     <Switch>
   12       <ProtectedRoute
   13         exact
   14         path="/"
   15         component={Home}
   16         isAuthenticated={isAuthenticated}
   17         isVerifying={isVerifying}
   18       />
   19       <Route path="/login" component={Login} />
   20     </Switch>
   21   );
   22 }
   23
   24 function mapStateToProps(state) {
   25   return {
   26     isAuthenticated: state.auth.isAuthenticated,
   27     isVerifying: state.auth.isVerifying,
   28   };
   29 }
   30
   31 export default connect(mapStateToProps)(App);

```

master 0 0 0 0 In 19, Col 48 Spaces: 2 UTF-8 LF JavaScript

14. Buatlah file dan simpan components/Login.js

```

File Edit Selection View Go Run Terminal Help
Login.js - praktikum-minggu12 - Visual Studio Code
OPEN EDITORS
  JS Login.js
  src > components > JS Login.js > (e) default
    1 import React, { Component } from "react";
    2 import { connect } from "react-redux";
    3 import { Redirect } from "react-router-dom";
    4 import { loginUser } from "./actions";
    5 import { withStyles } from "@material-ui/core/styles";
    6 import Avatar from "@material-ui/core/Button/index";
    7 import Button from "@material-ui/core/Button";
    8 import TextField from "@material-ui/core/TextField";
    9 import LockOutlinedIcon from "@material-ui/icons/LockOutlined";
   10 import Typography from "@material-ui/core/Typography";
   11 import Paper from "@material-ui/core/Paper";
   12 import Container from "@material-ui/core/Container";
   13
   14 const styles = () => ({
   15   "@global": {
   16     body: {
   17       backgroundColor: "#fff",
   18     },
   19   },
   20   paper: {
   21     margin: 100,
   22     display: "flex",
   23     padding: 20,
   24     flexDirection: "column",
   25     alignItems: "center",
   26   },
   27   avatar: {
   28     margin: 10,
   29     marginLeft: "auto",
   30     marginRight: "auto",
   31     background: "#fff000",
   32   },
   33   form: {
   34     margin: 10,
   35   },
   36   errorText: {
   37     color: "#f50057",
   38   }
   39 });

```

master 0 0 0 0 In 19, Col 48 Spaces: 2 UTF-8 LF JavaScript

EXPLORER

OPEN EDITORS

- XS Login.js src...
- U PRAKTIKUM-MINGGU12
- > build
- > node_modules
- > public
- > src
 - > actions
 - > components
 - JS Login.js U
 - > firebase
 - JS firebase.js
 - > reducers
 - # App.css
 - JS App.js M
 - JS App.test.js
 - JS configureStore.js
 - # index.css
 - JS index.js
 - logo.svg
 - JS root.js
 - serviceWorker.js
 - JS setupTests.js
 - firebaserc
 - .gitignore
 - firebase.json
 - firestore.indexes.json
 - firestore.rules
 - package-lock.json
 - package.json
 - REACT_APP_md
- > OUTLINE
- > TIMELINE
- > FROM SCRATCH

JS Login.js

```

src > components > JS Login.js > (9) default
  37   color: "#f5a957";
  38   marginBottom: 5;
  39   textAlign: "center";
  40   },
  41 });
  42
  43 class Login extends Component {
  44   state = { email: "", password: "" };
  45   handleEmailChange = ({ target }) => {
  46     this.setState({ email: target.value });
  47   };
  48   handlePasswordChange = ({ target }) => {
  49     this.setState({ password: target.value });
  50   };
  51   handleSubmit = () => {
  52     const { dispatch } = this.props;
  53     const { email, password } = this.state;
  54     dispatch(loginUser(email, password));
  55   }
  56   render() {
  57     const { classes, loginError, isAuthenticated } = this.props;
  58     if (isAuthenticated) {
  59       return <Redirect to="/" />;
  60     } else {
  61       return (
  62         <Container component="main" maxWidth="xs">
  63           <Paper className={classes.paper}>
  64             <Avatar className={classes.avatar}>
  65               <LockOutlinedIcon />
  66             </Avatar>
  67             <Typography component="h1" variant="h5">
  68               Sign In
  69             </Typography>
  70             <TextField
  71               variant="outlined"
  72               margin="normal"
  73               fullWidth
  74               id="email"
  75               label="Email Address"
  76               name="email"
  77               onChange={this.handleEmailChange}
  78             />
  79             <TextField
  80               variant="outlined"
  81               margin="normal"
  82               fullWidthname="password"
  83               label="Password"
  84               type="password"
  85               id="password"
  86               onChange={this.handlePasswordChange}
  87             />
  88             {loginError && (
  89               <Typography component="p" className={classes.errorText}>
  90                 Incorrect email or password.
  91               </Typography>
  92             )}
  93             <Button
  94               type="button"
  95               fullWidth
  96               variant="contained"
  97               color="primary"
  98               className={classes.submit}
  99               onClick={this.handleSubmit}
  100             >
  101               Sign In
  102             </Button>
  103           </Paper>
  104         </Container>
  105       );
  106     }
  107   }
  108 }

```

EXPLORER

OPEN EDITORS

- XS Login.js src...
- U PRAKTIKUM-MINGGU12
- > build
- > node_modules
- > public
- > src
 - > actions
 - > components
 - JS Login.js U
 - > firebase
 - JS firebase.js
 - > reducers
 - # App.css
 - JS App.js M
 - JS App.test.js
 - JS configureStore.js
 - # index.css
 - JS index.js
 - logo.svg
 - JS root.js
 - serviceWorker.js
 - JS setupTests.js
 - firebaserc
 - .gitignore
 - firebase.json
 - firestore.indexes.json
 - firestore.rules
 - package-lock.json
 - package.json
 - REACT_APP_md
- > OUTLINE
- > TIMELINE

JS Login.js

```

src > components > JS Login.js > (9) default
  73   fullWidth
  74   id="email"
  75   label="Email Address"
  76   name="email"
  77   onChange={this.handleEmailChange}
  78   />
  79   <TextField
  80     variant="outlined"
  81     margin="normal"
  82     fullWidthname="password"
  83     label="Password"
  84     type="password"
  85     id="password"
  86     onChange={this.handlePasswordChange}
  87   />
  88   {loginError && (
  89     <Typography component="p" className={classes.errorText}>
  90       Incorrect email or password.
  91     </Typography>
  92   )}
  93   <Button
  94     type="button"
  95     fullWidth
  96     variant="contained"
  97     color="primary"
  98     className={classes.submit}
  99     onClick={this.handleSubmit}
  100   >
  101     Sign In
  102   </Button>
  103 </Paper>
  104 </Container>
  105 );
  106 }
  107 }
  108

```

EXPLORER

OPEN EDITORS

- XIS Login.js src...
- U PRAKTIKUM-MINGGU12
- > build
- > node_modules
- > public
- > src
 - > actions
 - > components
 - JS Login.js U
 - JS Home.js U
 - > firebase
 - JS firebase.js
 - > reducers
 - # App.css
 - JS App.js M
 - JS App.test.js
 - JS configureStore.js
 - # index.css
 - JS index.js
 - logo.svg
 - JS root.js
 - serviceWorker.js
 - JS setupTests.js
 - firebaserc
 - .gitignore
 - firebase.json
- > OUTLINE
- > TIMELINE

JS Login.js

```

src > components > JS Login.js > (9) default
  109
  110   function mapStateToProps(state) {
  111     return {
  112       isloggingIn: state.auth.isLoggingIn,
  113       loginError: state.auth.loginError,
  114       isAuthenticated: state.auth.isAuthenticated,
  115     };
  116   }
  117
  118   export default withStyles(styles)(connect(mapStateToProps)(Login));

```

15. Buka component/Home.js

EXPLORER

OPEN EDITORS

- XIS Login.js src...
- U PRAKTIKUM-MINGGU12
- > build
- > node_modules
- > public
- > src
 - > actions
 - > components
 - JS Login.js U
 - JS Home.js U
 - > firebase
 - JS firebase.js
 - > reducers
 - # App.css
 - JS App.js M
 - JS App.test.js
 - JS configureStore.js
 - # index.css
 - JS index.js
 - logo.svg
 - JS root.js
 - serviceWorker.js
 - JS setupTests.js
 - firebaserc
 - .gitignore
 - firebase.json
- > OUTLINE

JS Home.js

```

src > components > JS Home.js > (9) default
  1   import React, { Component } from "react";
  2   import { connect } from "react-redux";
  3   import { logoutUser } from "./actions/auth";
  4
  5   class Home extends Component {
  6     handleLogout = () => {
  7       const { dispatch } = this.props;
  8       dispatch(logoutUser());
  9     }
 10
 11     render() {
 12       const { isloggingOut, logoutError } = this.props;
 13       return (
 14         <div>
 15           <h1>This is your app's protected area.</h1>
 16           <p>Any routes here will also be protected.</p>
 17           <button onClick={this.handleLogout}>Logout</button>
 18           {isloggingOut && <p>Logging Out...</p>}
 19           {logoutError && <p>Error logging out</p>}
 20         </div>
 21       )
 22     }
 23
 24     function mapStateToProps(state) {
 25       return {
 26         isloggingOut: state.auth.isLoggingOut,
 27         logoutError: state.auth.logoutError,
 28       };
 29     }
 30
 31     export default connect(mapStateToProps)(Home);

```

16. Running start



17. Masuk ke firebase console kemudian add user

Identifier	Providers	Created	Signed In	User UID
izmirtais28@gmail.com	✉	Apr 23, 20...		YXARlxkGUcbzkqGikWUCSb...

18. Pilih npm start kemudian masukkan username dan password



19. Kemudian masuk ke dalam halaman beranda

