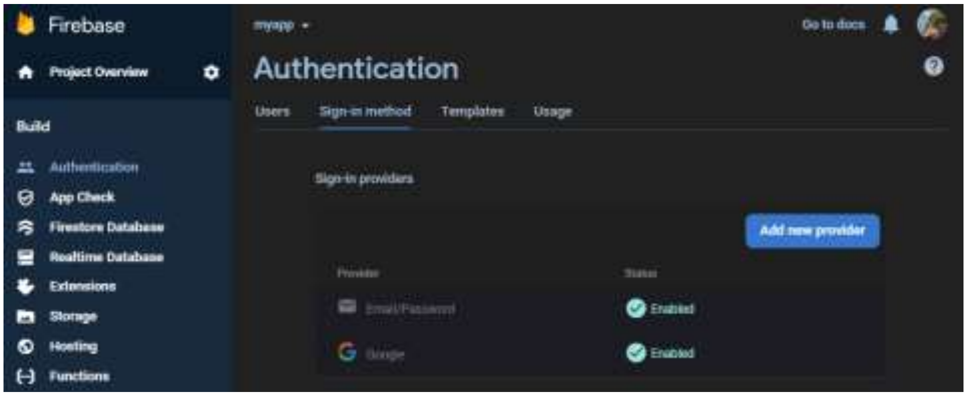
JOBSHEET 6

Omar Al-Maktary

TI-3H 1941720237

EXPERIMENT

1- Making sure authentication methods are enabled:

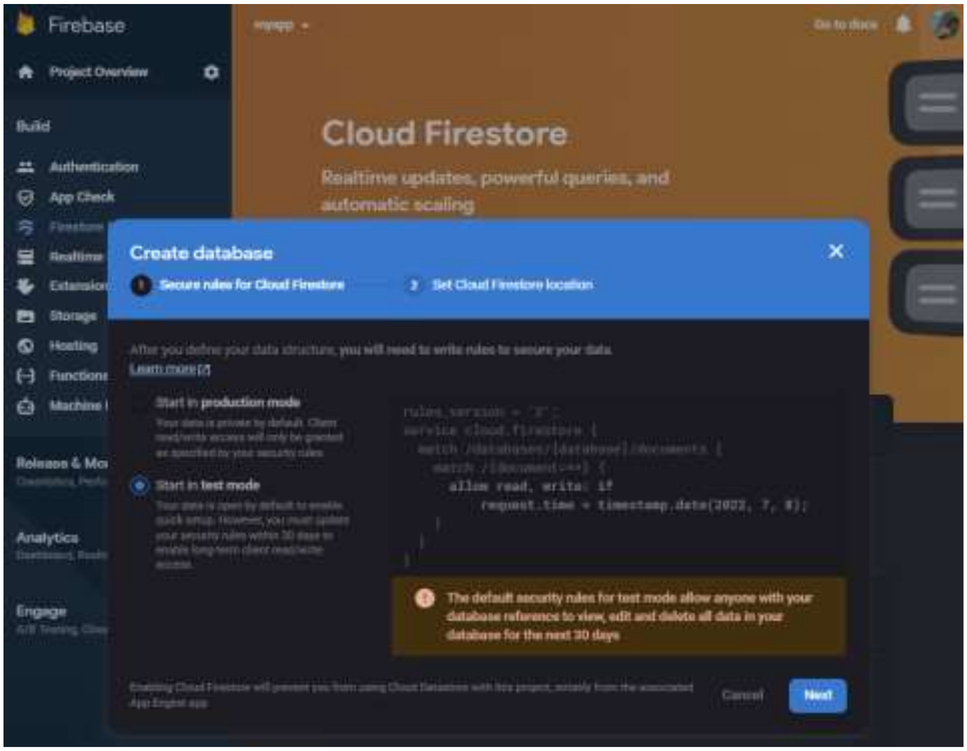


Authentication

Sign-in providers

Provider	Status
Email/Password	Enabled
Google	Enabled

2- Creating Database in test mode using Firestore Database:



Create database

1 Secure rules for Cloud Firestore 2 Set Cloud Firestore location

After you define your data structure, you will need to write rules to secure your data.
Learn more

○ Start in production mode
Your data is private by default. Client read/write access will only be granted as specified by your security rules.

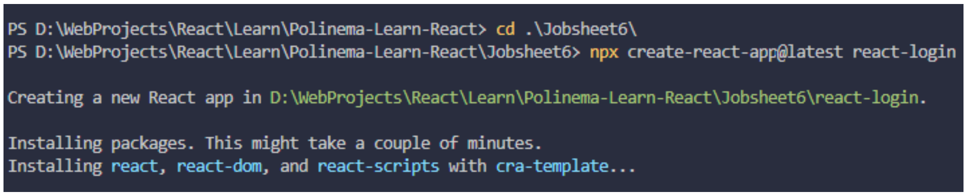
● Start in test mode
Your data is open by default to enable quick setup. However, you must update your security rules within 30 days to enable long-term client read/write access.

```
rules_version = '2';
service cloud.firestore {
  match /databases/{database}/documents {
    match /{document=**} {
      allow read, write: if
          request.time < timestamp.date(2022, 7, 8);
    }
  }
}
```

The default security rules for test mode allow anyone with your database reference to view, edit and delete all data in your database for the next 30 days

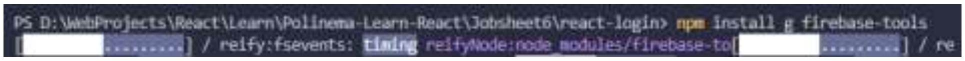
Enabling Cloud Firestore will prevent you from using Cloud Datastore with this project, notably from the associated App Engine app.

3- Creating react project with the name react login:



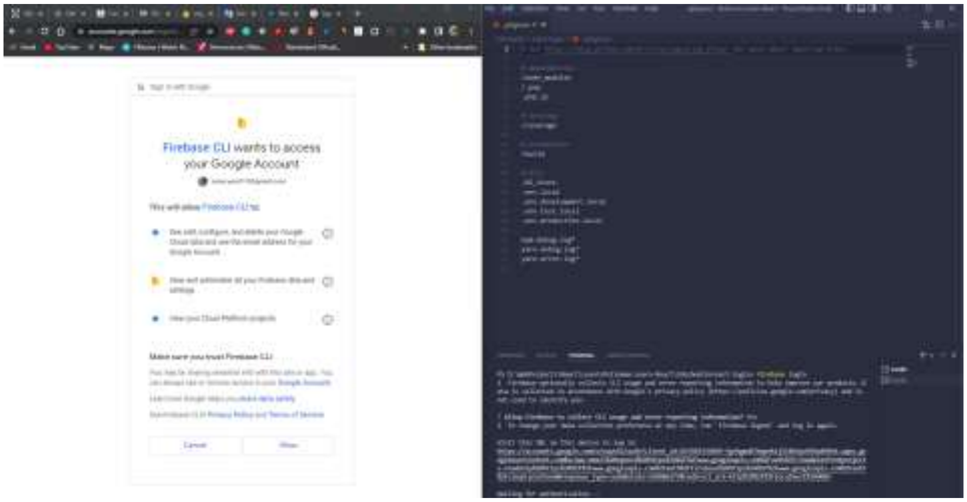
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React> cd .\Jobsheet6\
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6> npx create-react-app@latest react-login

Creating a new React app in D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...
```

4- Install firebase tools for the react project:



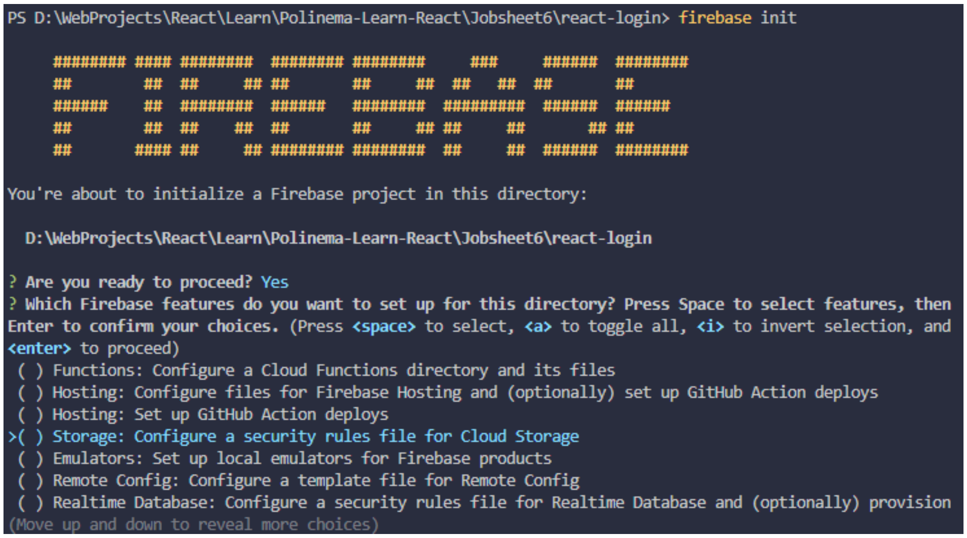
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install g firebase-tools
[          .........] / reify:fsevents: timing reifyNode:node_modules/firebase-to[          .........] / re
```

5- Login into firebase CLI using firebase in the terminal:



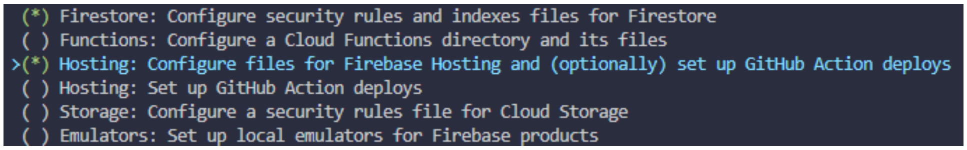
Firebase CLI wants to access your Google Account

6- Initialize firebase in the project:



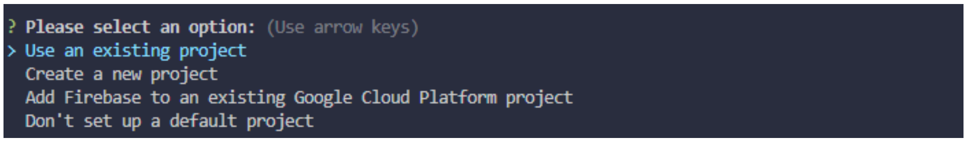
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> firebase init

     ######## #### ########  ######## ########     ###     ######  ########
     ##        ##  ##     ## ##       ##     ##  ##   ##  ##       ##
     ######    ##  ########  ######   ########  #########  ######  ######
     ##        ##  ##    ##  ##       ##     ## ##     ##       ## ##
     ##       #### ##     ## ######## ########  ##     ##  ######  ########

You're about to initialize a Firebase project in this directory:

  D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login

? Are you ready to proceed? Yes
? Which Firebase features do you want to set up for this directory? Press Space to select features, then
Enter to confirm your choices. (Press <space> to select, <a> to toggle all, <i> to invert selection, and
<enter> to proceed)
 ( ) Functions: Configure a Cloud Functions directory and its files
 ( ) Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
 ( ) Hosting: Set up GitHub Action deploys
>( ) Storage: Configure a security rules file for Cloud Storage
 ( ) Emulators: Set up local emulators for Firebase products
 ( ) Remote Config: Configure a template file for Remote Config
 ( ) Realtime Database: Configure a security rules file for Realtime Database and (optionally) provision
(Move up and down to reveal more choices)
```

7- Select Firestore and hosting features:



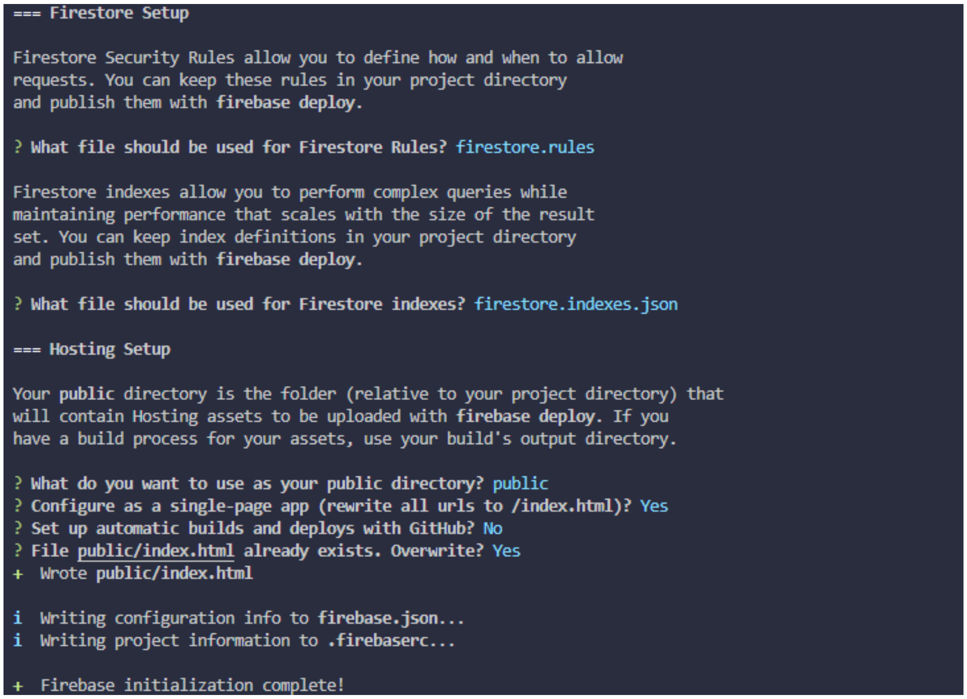
```
 (*) Firestore: Configure security rules and indexes files for Firestore
 ( ) Functions: Configure a Cloud Functions directory and its files
>(*) Hosting: Configure files for Firebase Hosting and (optionally) set up GitHub Action deploys
 ( ) Hosting: Set up GitHub Action deploys
 ( ) Storage: Configure a security rules file for Cloud Storage
 ( ) Emulators: Set up local emulators for Firebase products
```

8- Choosing a firebase project to be the default project for the react project using firebase CLI:



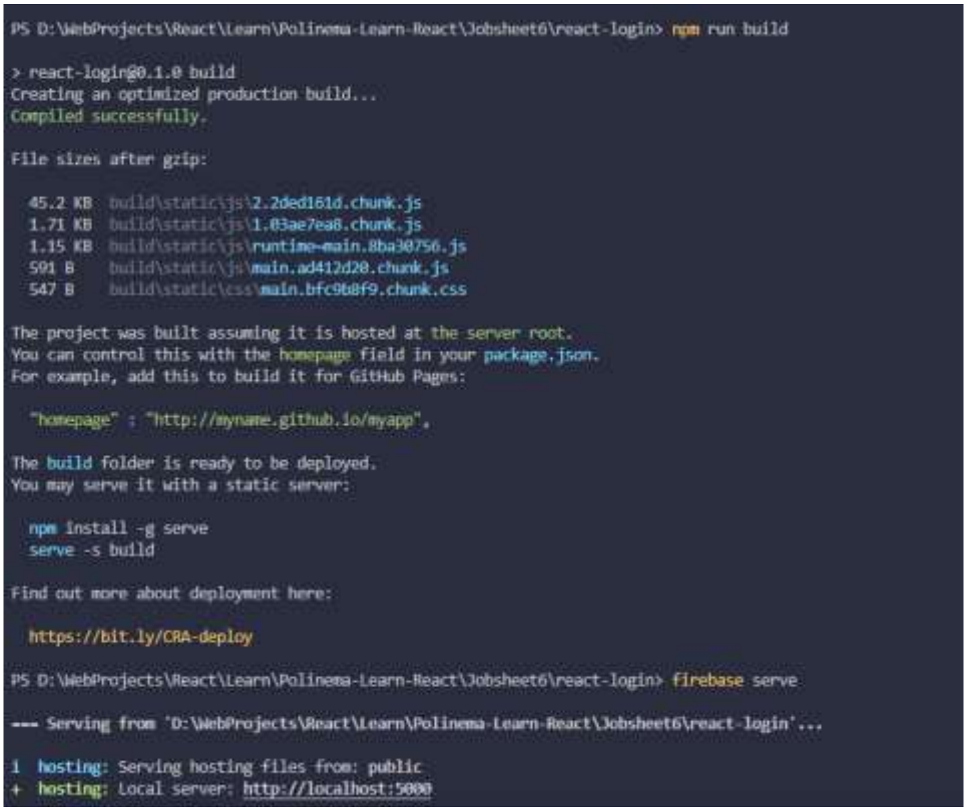
```
? Please select an option: (Use arrow keys)
> Use an existing project
  Create a new project
  Add Firebase to an existing Google Cloud Platform project
  Don't set up a default project
```

9- Firestore setup:



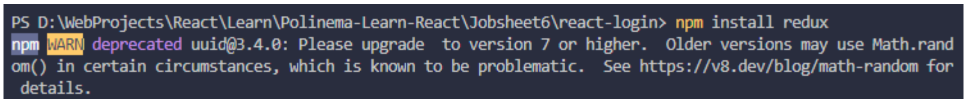
```
=== Firestore Setup

Firestore Security Rules allow you to define how and when to allow
requests. You can keep these rules in your project directory
and publish them with firebase deploy.

? What file should be used for Firestore Rules? firestore.rules

Firestore indexes allow you to perform complex queries while
maintaining performance that scales with the size of the result
set. You can keep index definitions in your project directory
and publish them with firebase deploy.

? 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
will contain Hosting assets to be uploaded with firebase deploy. If you
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)? Yes
? Set up automatic builds and deploys with GitHub? No
? File public/index.html already exists. Overwrite? Yes
+  Wrote public/index.html

i  Writing configuration info to firebase.json...
i  Writing project information to .firebaserc...

+  Firebase initialization complete!
```

10- Build the react project and serve the firebase one

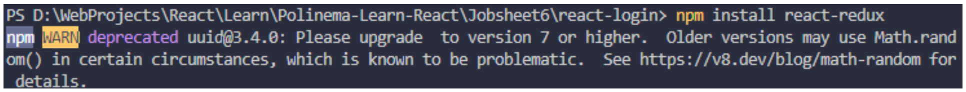


```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm run build

> react-login@0.1.0 build
Creating an optimized production build...
Compiled successfully.

File sizes after gzip:

  45.2 KB  build\static\js\2.2ded161d.chunk.js
  1.71 KB  build\static\js\1.03ae7ea8.chunk.js
  1.15 KB  build\static\js\runtime-main.8ba30756.js
  591 B    build\static\js\main.ad412d20.chunk.js
  547 B    build\static\css\main.bfc9b8f9.chunk.css

The project was built assuming it is hosted at the server root.
You can control this with the homepage field in your package.json.
For example, add this to build it for GitHub Pages:

  "homepage" : "http://myname.github.io/myapp",

The build folder is ready to be deployed.
You may serve it with a static server:

  npm install -g serve
  serve -s build

Find out more about deployment here:

  https://bit.ly/CRA-deploy

PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> firebase serve

=== Serving from 'D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login'...

i  hosting: Serving hosting files from: public
+  hosting: Local server: http://localhost:5000
```

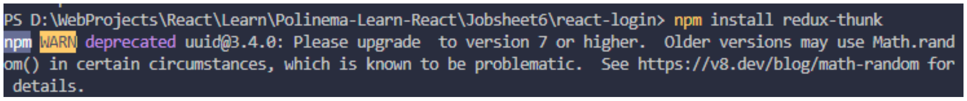
11- Redux Installation:



```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install redux
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older versions may use Math.rand
om() in certain circumstances, which is known to be problematic.  See https://v8.dev/blog/math-random for
 details.
```

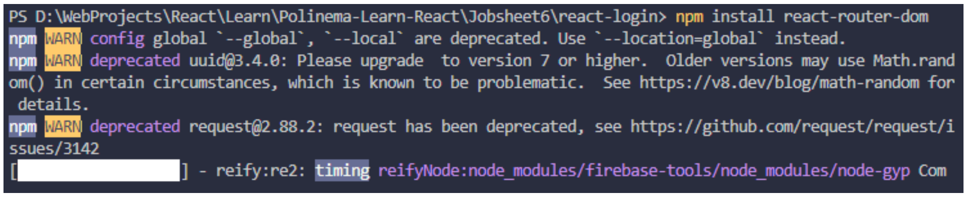


```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install react-redux
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older versions may use Math.rand
om() in certain circumstances, which is known to be problematic.  See https://v8.dev/blog/math-random for
 details.
```



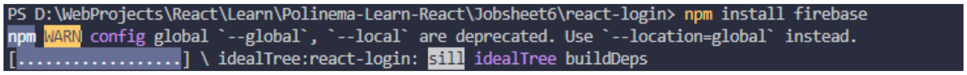
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install redux-thunk
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older versions may use Math.rand
om() in certain circumstances, which is known to be problematic.  See https://v8.dev/blog/math-random for
 details.
```

Install react route dom



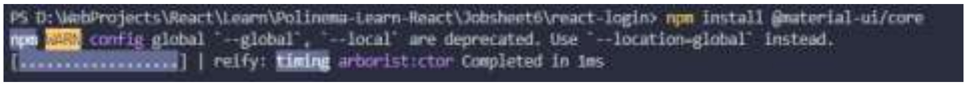
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install react-router-dom
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.
npm WARN deprecated uuid@3.4.0: Please upgrade  to version 7 or higher.  Older versions may use Math.rand
om() in certain circumstances, which is known to be problematic.  See https://v8.dev/blog/math-random for
 details.
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/i
ssues/3142
[                  ] - reify:re2: timing reifyNode:node_modules/firebase-tools/node_modules/node-gyp Com
```

12- Install Firebase



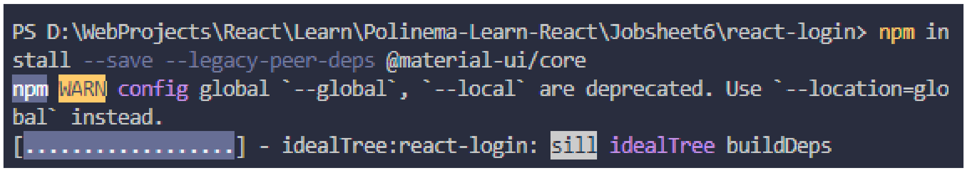
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install firebase
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.
[..................] \ idealTree:react-login: sill idealTree buildDeps
```

13- Install UI library material UI



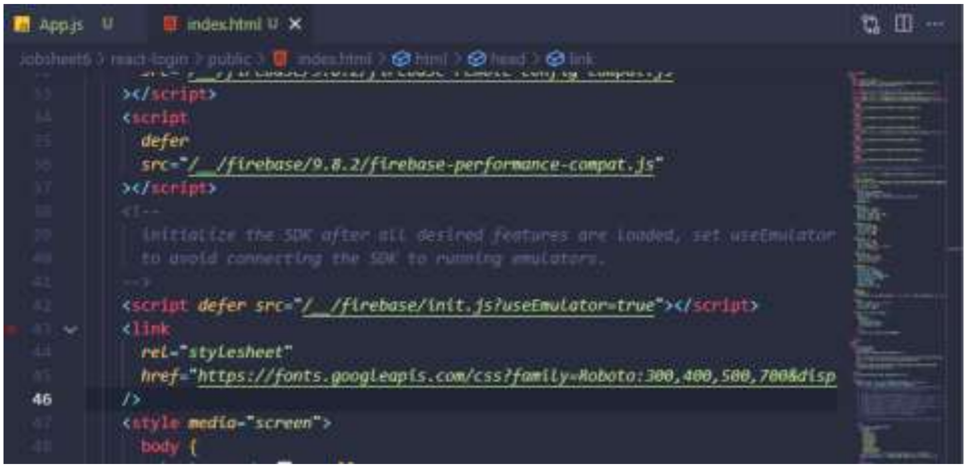
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install @material-ui/core
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.
[..................] | reify: timing arborist:ctor Completed in 1ms
```

14- Let material UI works with the current version of react



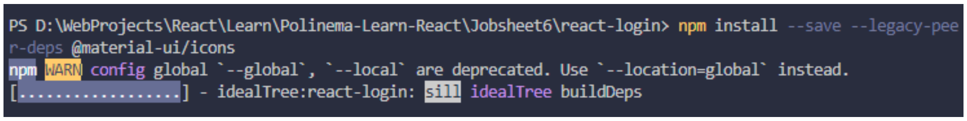
```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm in
stall --save --legacy-peer-deps @material-ui/core
npm WARN config global `--global`, `--local` are deprecated. Use `--location=glo
bal` instead.
[..................] - idealTree:react-login: sill idealTree buildDeps
```

15- Add font link to index.html in public directory of the react project


App.js U
index.html U
Jobsheet6 > react-login > public > index.html > html > head > link

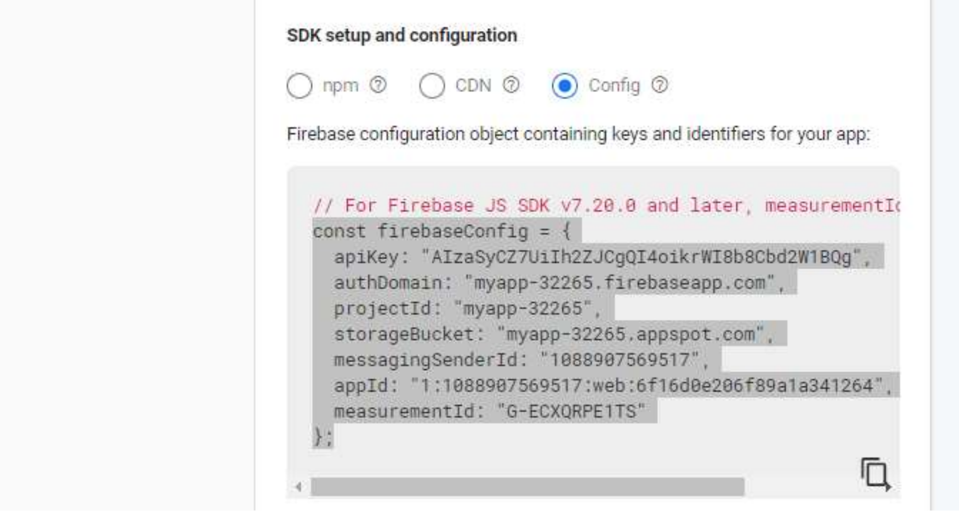
```
    ></script>
    <script
      defer
      src="/__/firebase/9.8.2/firebase-performance-compat.js"
    ></script>
    <!--
      initialize the SDK after all desired features are loaded, set useEmulator
      to avoid connecting the SDK to running emulators.
    -->
    <script defer src="/__/firebase/init.js?useEmulator=true"></script>
    <link
      rel="stylesheet"
      href="https://fonts.googleapis.com/css?family=Roboto:300,400,500,700&disp
    />
    <style media="screen">
      body {
```

16- Install material icons



```
PS D:\WebProjects\React\Learn\Polinema-Learn-React\Jobsheet6\react-login> npm install --save --legacy-pee
r-deps @material-ui/icons
npm WARN config global `--global`, `--local` are deprecated. Use `--location=global` instead.
[..................] - idealTree:react-login: sill idealTree buildDeps
```

17- Setting Firebase Config

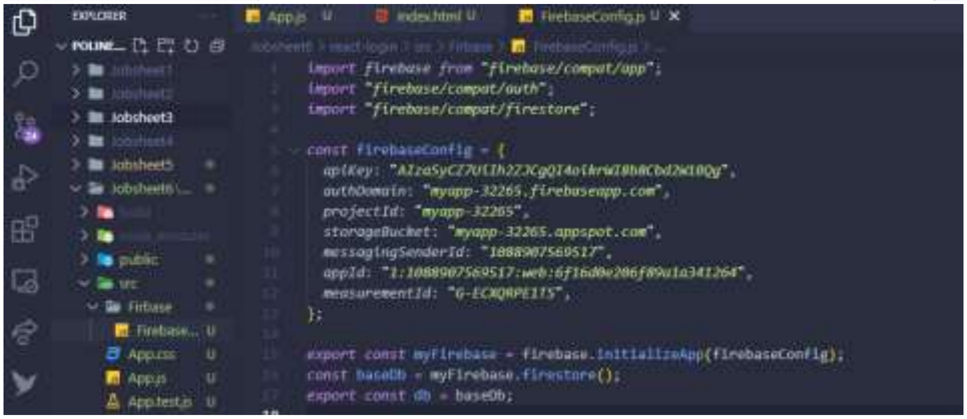


SDK setup and configuration

○ npm ⓘ ○ CDN ⓘ ● Config ⓘ

Firebase configuration object containing keys and identifiers for your app:

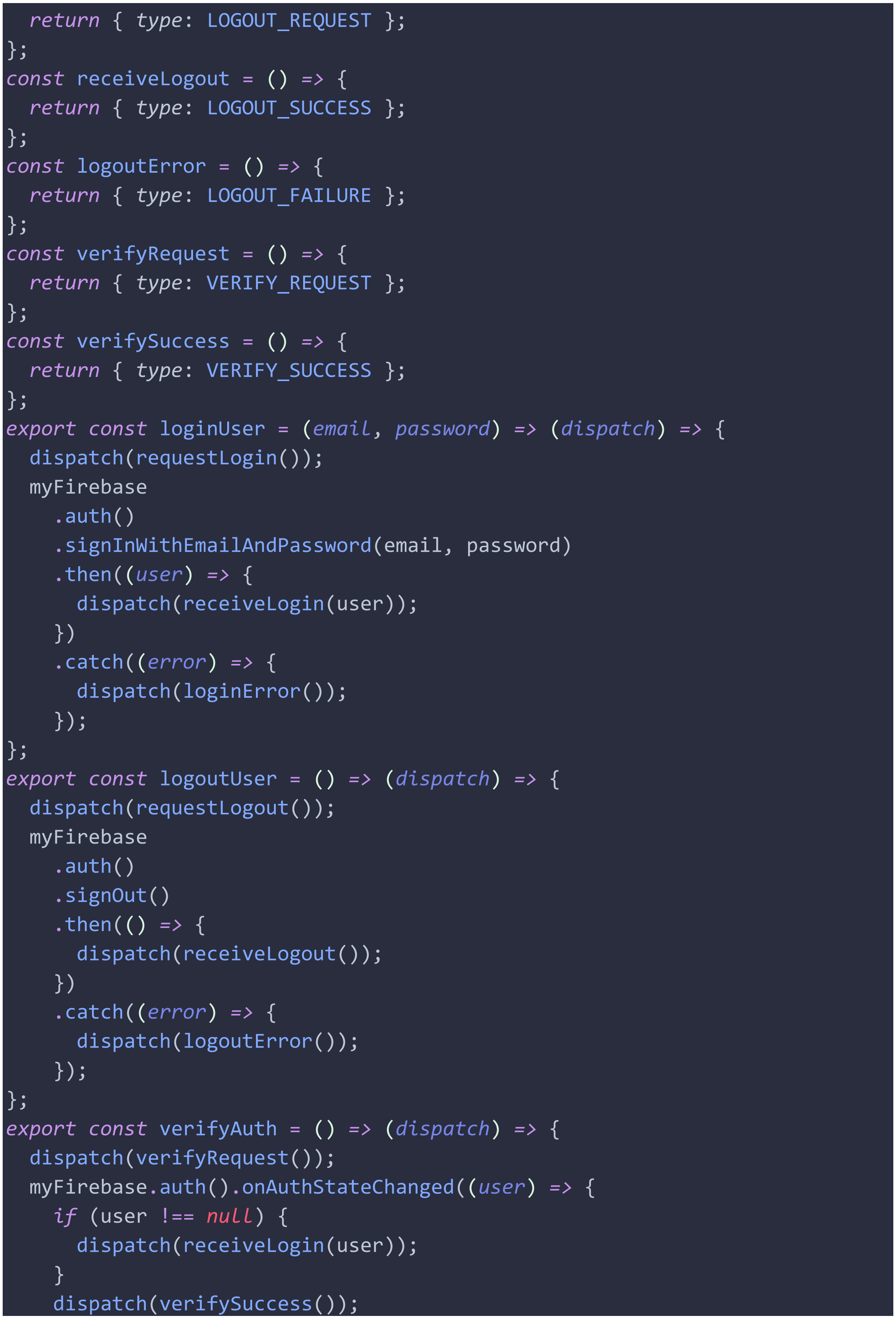
```
// For Firebase JS SDK v7.20.0 and later, measurementIc
const firebaseConfig = {
  apiKey: "AIzaSyCZ7UiIh2ZJCgQI4oikrWI8b8Cbd2W1BQg",
  authDomain: "myapp-32265.firebaseapp.com",
  projectId: "myapp-32265",
  storageBucket: "myapp-32265.appspot.com",
  messagingSenderId: "1088907569517",
  appId: "1:1088907569517:web:6f16d0e206f89a1a341264",
  measurementId: "G-ECXQRPE1TS"
};
```



```
import firebase from "firebase/compat/app";
import "firebase/compat/auth";
import "firebase/compat/firestore";

const firebaseConfig = {
  apiKey: "AIzaSyCZ7UiIh2ZJCgQI4oikrWI8b8Cbd2W1BQg",
  authDomain: "myapp-32265.firebaseapp.com",
  projectId: "myapp-32265",
  storageBucket: "myapp-32265.appspot.com",
  messagingSenderId: "1088907569517",
  appId: "1:1088907569517:web:6f16d0e206f89a1a341264",
  measurementId: "G-ECXQRPE1TS",
};

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

18- Auth.js file in the actions folder in the react project:

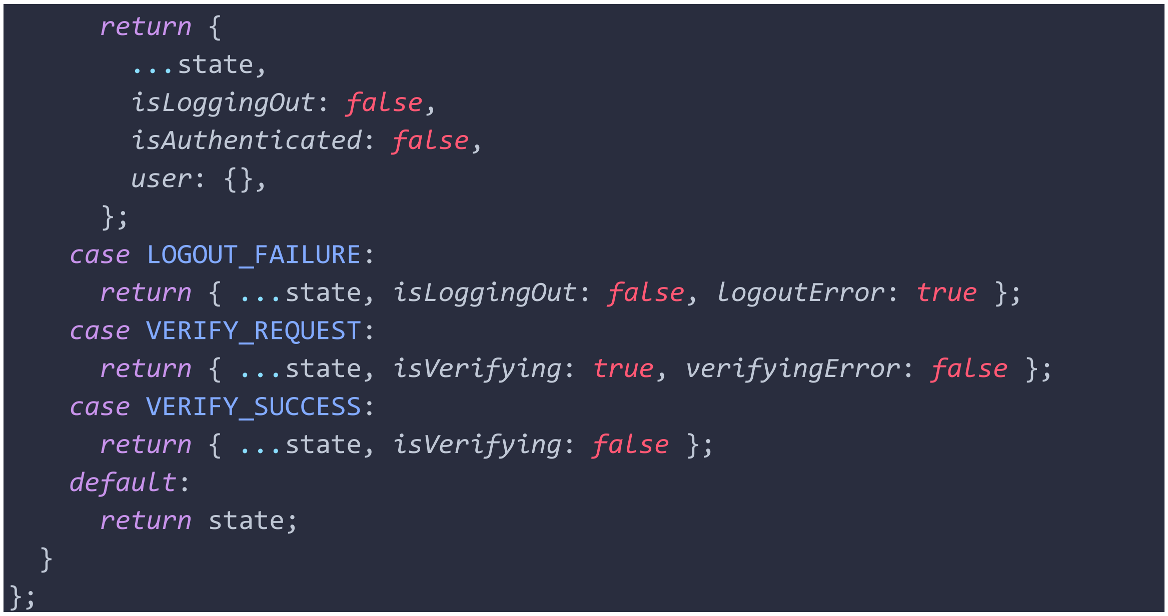
```
import { myFirebase } from "../Firbase/FirebaseConfig";
export const LOGIN_REQUEST = "LOGIN_REQUEST";
export const LOGIN_SUCCESS = "LOGIN_SUCCESS";
export const LOGIN_FAILURE = "LOGIN_FAILURE";
export const LOGOUT_REQUEST = "LOGOUT_REQUEST";
export const LOGOUT_SUCCESS = "LOGOUT_SUCCESS";
export const LOGOUT_FAILURE = "LOGOUT_FAILURE";
export const VERIFY_REQUEST = "VERIFY_REQUEST";
export const VERIFY_SUCCESS = "VERIFY_SUCCESS";
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 };
};
const logoutError = () => {
  return { type: LOGOUT_FAILURE };
};
const verifyRequest = () => {
  return { type: VERIFY_REQUEST };
};
const verifySuccess = () => {
  return { type: VERIFY_SUCCESS };
};
export const loginUser = (email, password) => (dispatch) => {
  dispatch(requestLogin());
  myFirebase
    .auth()
    .signInWithEmailAndPassword(email, password)
    .then((user) => {
      dispatch(receiveLogin(user));
    })
    .catch((error) => {
      dispatch(loginError());
    });
};
export const logoutUser = () => (dispatch) => {
  dispatch(requestLogout());
  myFirebase
    .auth()
    .signOut()
    .then(() => {
      dispatch(receiveLogout());
    })
    .catch((error) => {
      dispatch(logoutError());
    });
};
export const verifyAuth = () => (dispatch) => {
  dispatch(verifyRequest());
  myFirebase.auth().onAuthStateChanged((user) => {
    if (user !== null) {
      dispatch(receiveLogin(user));
    }
    dispatch(verifySuccess());
  });
};
```

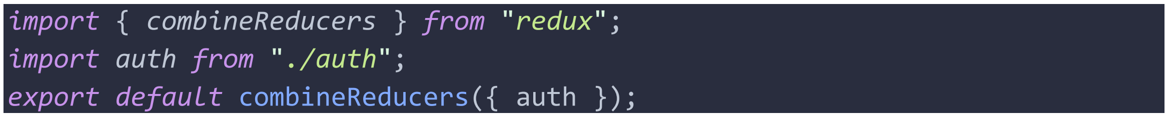
19- Setting up a reducer:

```
import {
  LOGIN_REQUEST,
  LOGIN_SUCCESS,
  LOGIN_FAILURE,
  LOGOUT_REQUEST,
  LOGOUT_SUCCESS,
  LOGOUT_FAILURE,
  VERIFY_REQUEST,
  VERIFY_SUCCESS,
} from "../actions/auth";
export default (
  state = {
    isLoggingIn: false,
    isLoggingOut: false,
    isVerifying: false,
    loginError: false,
    logoutError: false,
    isAuthenticated: false,
    user: {},
  },
  action
) => {
  switch (action.type) {
    case LOGIN_REQUEST:
      return { ...state, isLoggingIn: true, loginError: false };
    case LOGIN_SUCCESS:
      return {
        ...state,
        isLoggingIn: false,
        isAuthenticated: true,
        user: action.user,
      };
    case LOGIN_FAILURE:
      return {
        ...state,
        isLoggingIn: false,
        isAuthenticated: false,
        loginError: true,
      };
    case LOGOUT_REQUEST:
      return { ...state, isLoggingOut: true, logoutError: false };
    case LOGOUT_SUCCESS:
```



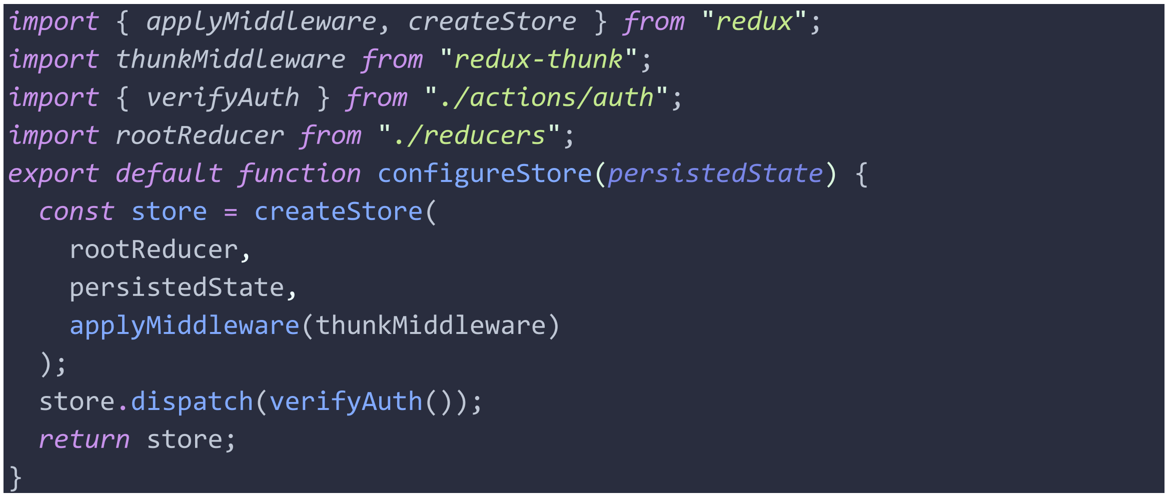
```
      return {
        ...state,
        isLoggingOut: false,
        isAuthenticated: false,
        user: {},
      };
    case LOGOUT_FAILURE:
      return { ...state, isLoggingOut: false, logoutError: true };
    case VERIFY_REQUEST:
      return { ...state, isVerifying: true, verifyingError: false };
    case VERIFY_SUCCESS:
      return { ...state, isVerifying: false };
    default:
      return state;
  }
};
```

20- Index file in the reducer folder:



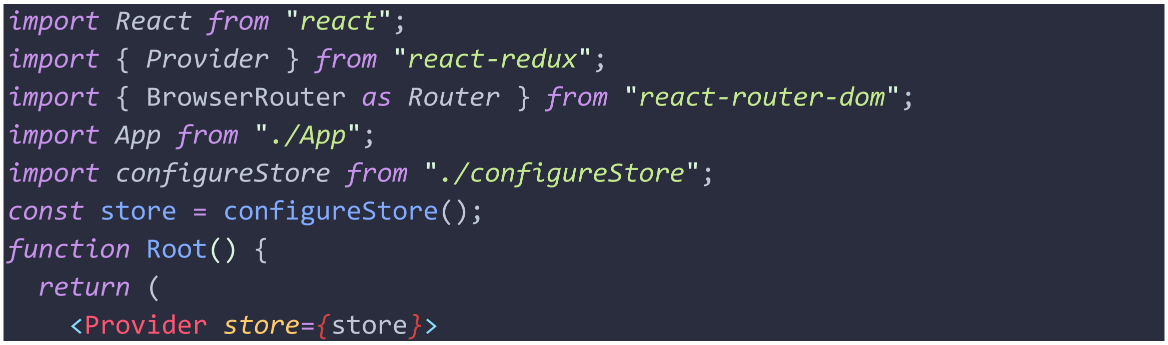
```
import { combineReducers } from "redux";
import auth from "./auth";
export default combineReducers({ auth });
```

21- configureStore.js file in the scr folder:



```
import { applyMiddleware, createStore } from "redux";
import thunkMiddleware from "redux-thunk";
import { verifyAuth } from "./actions/auth";
import rootReducer from "./reducers";
export default function configureStore(persistedState) {
  const store = createStore(
    rootReducer,
    persistedState,
    applyMiddleware(thunkMiddleware)
  );
  store.dispatch(verifyAuth());
  return store;
}
```

22- root.js file in the scr folder:



```
import React from "react";
import { Provider } from "react-redux";
import { BrowserRouter as Router } from "react-router-dom";
import App from "./App";
import configureStore from "./configureStore";
const store = configureStore();
function Root() {
  return (
    <Provider store={store}>
```

```
      <Router>
        <App />
      </Router>
    </Provider>
  );
}
export default Root;
```

23- index.js file in the scr folder:

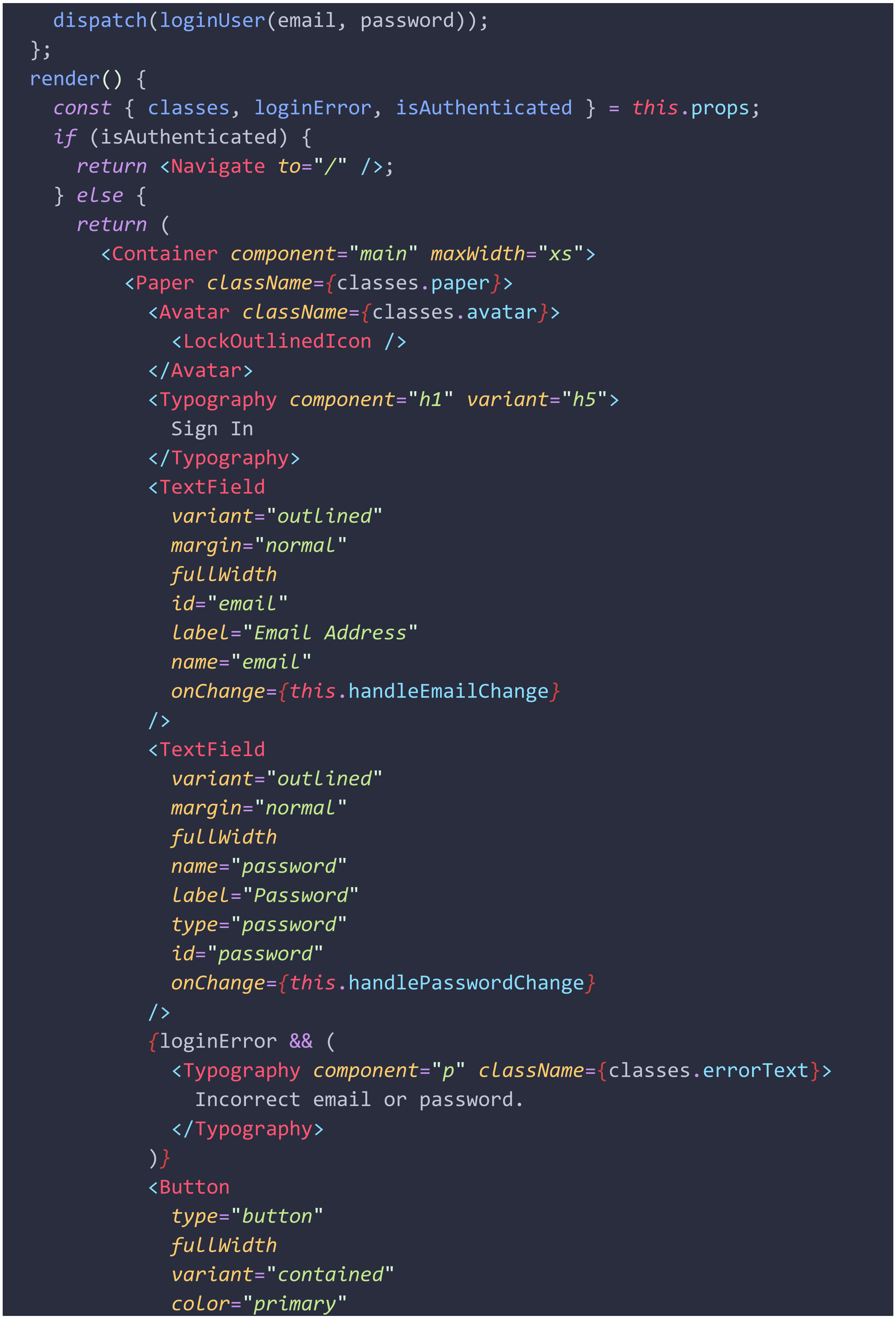
```
import React from "react";
import ReactDOM from "react-dom";
import Root from "./root";
import * as serviceWorker from "./serviceWorker";
ReactDOM.render(<Root />, document.getElementById("root"));
// If you want your app to work offline and load faster, you can change
// unregister() to register() below. Note this comes with some pitfalls.
// Learn more about service workers: https://bit.ly/CRA-PWA
serviceWorker.unregister();
```

24- App.js file in the scr folder:

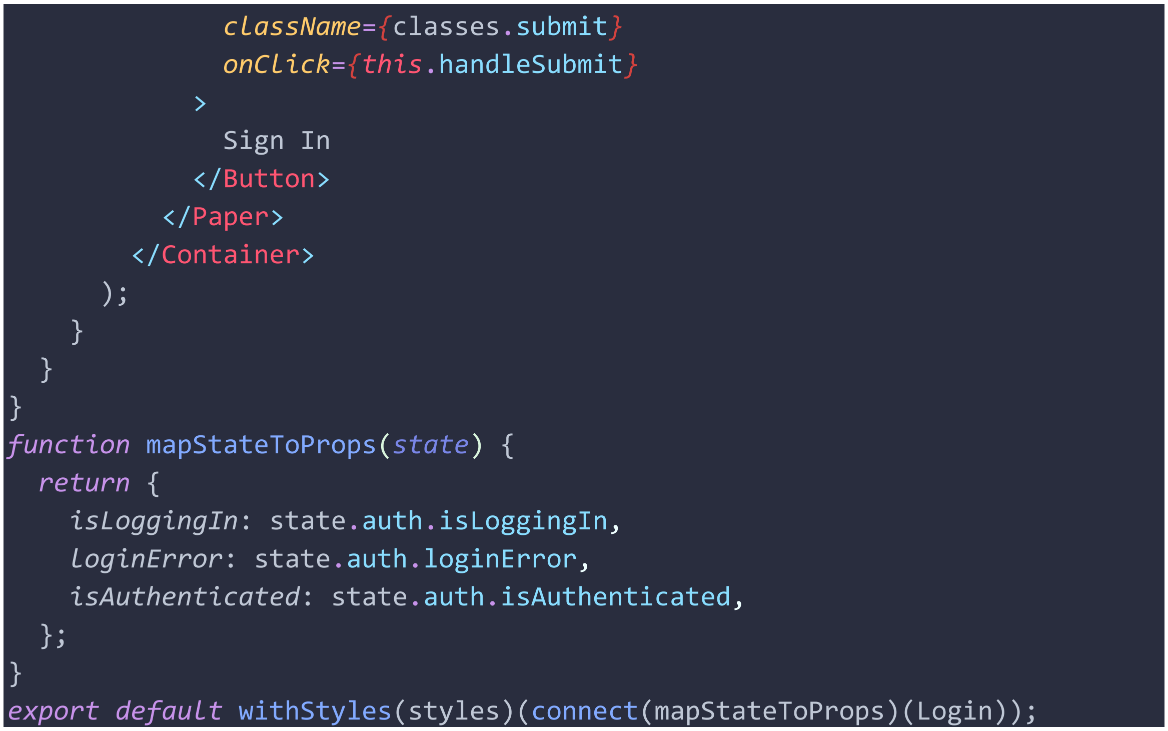
```
import React from "react";
import { Route, Routes } from "react-router-dom";
import { connect } from "react-redux";
import ProtectedRoute from "./components/ProtectedRoute";
import Home from "./components/Home";
import Login from "./components/Login";
function App(props) {
  const { isAuthenticated, isVerifying } = props;
  return (
    <Routes>
      <ProtectedRoute
        exact
        path="/"
        component={Home}
        isAuthenticated={isAuthenticated}
        isVerifying={isVerifying}
      />
      <Route path="/login" component={Login} />
    </Routes>
  );
}
function mapStateToProps(state) {
  return {
    isAuthenticated: state.auth.isAuthenticated,
    isVerifying: state.auth.isVerifying,
  };
}
export default connect(mapStateToProps)(App);
```

25- Login Page

```
import React, { Component } from "react";
import { connect } from "react-redux";
import { Navigate } from "react-router-dom";
import { loginUser } from "../actions/auth";
import { withStyles } from "@material-ui/styles";
import Avatar from "@material-ui/core/Avatar";
import Button from "@material-ui/core/Button";
import TextField from "@material-ui/core/TextField";
import LockOutlinedIcon from "@material-ui/icons/LockOutlined";
import Typography from "@material-ui/core/Typography";
import Paper from "@material-ui/core/Paper";
import Container from "@material-ui/core/Container";
const styles = () => ({
  "@global": { body: { backgroundColor: "#fff" } },
  paper: {
    marginTop: 100,
    display: "flex",
    padding: 20,
    flexDirection: "column",
    alignItems: "center",
  },
  avatar: {
    marginLeft: "auto",
    marginRight: "auto",
    marginBottom: "15px",
    backgroundColor: "#f50057",
  },
  form: { marginTop: 1 },
  errorText: { color: "#f50057", marginBottom: 5, textAlign: "center" },
});
class Login extends Component {
  state = { email: "", password: "" };
  handleEmailChange = ({ target }) => {
    this.setState({ email: target.value });
  };
  handlePasswordChange = ({ target }) => {
    this.setState({ password: target.value });
  };
  handleSubmit = () => {
    const { dispatch } = this.props;
    const { email, password } = this.state;
```

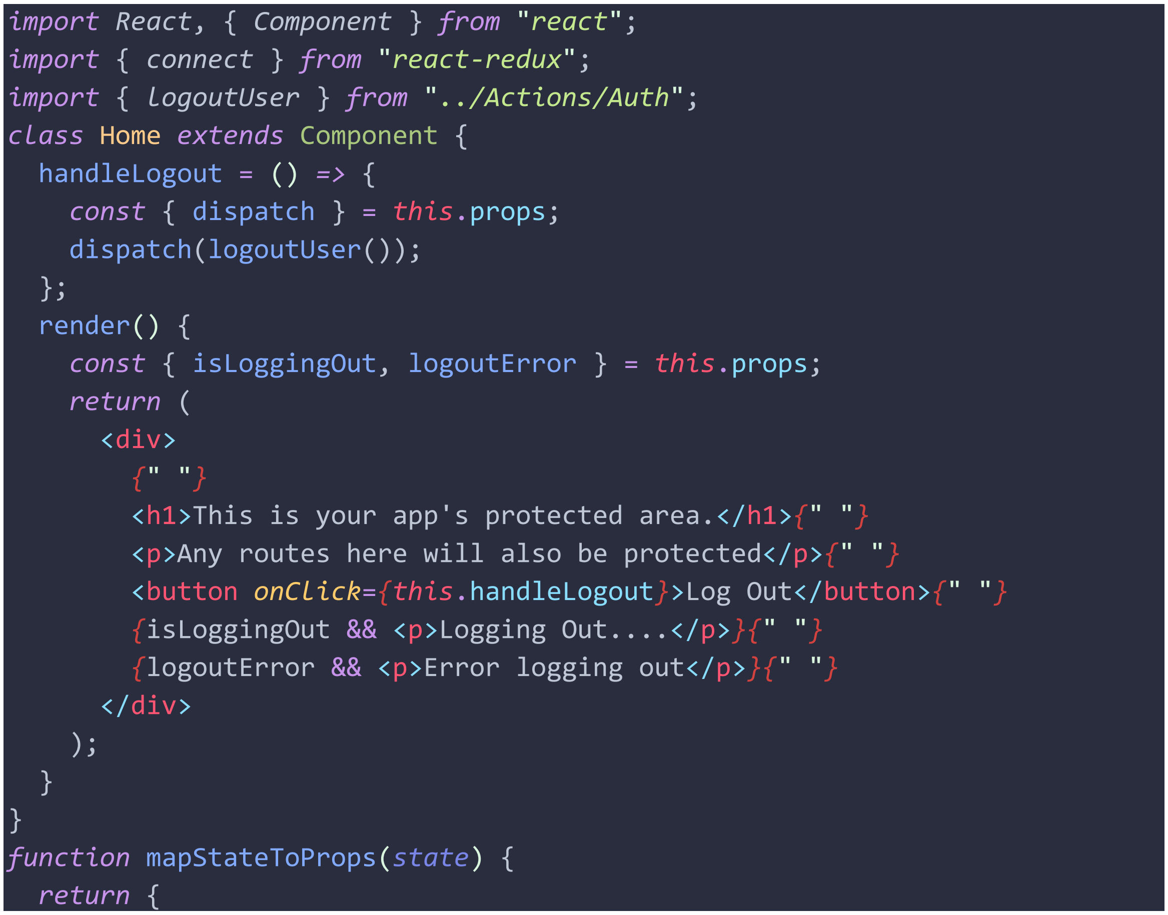


```
    dispatch(loginUser(email, password));
  };
  render() {
    const { classes, loginError, isAuthenticated } = this.props;
    if (isAuthenticated) {
      return <Navigate to="/" />;
    } else {
      return (
        <Container component="main" maxWidth="xs">
          <Paper className={classes.paper}>
            <Avatar className={classes.avatar}>
              <LockOutlinedIcon />
            </Avatar>
            <Typography component="h1" variant="h5">
              Sign In
            </Typography>
            <TextField
              variant="outlined"
              margin="normal"
              fullWidth
              id="email"
              label="Email Address"
              name="email"
              onChange={this.handleEmailChange}
            />
            <TextField
              variant="outlined"
              margin="normal"
              fullWidth
              name="password"
              label="Password"
              type="password"
              id="password"
              onChange={this.handlePasswordChange}
            />
            {loginError && (
              <Typography component="p" className={classes.errorText}>
                Incorrect email or password.
              </Typography>
            )}
            <Button
              type="button"
              fullWidth
              variant="contained"
              color="primary"
```



```
              className={classes.submit}
              onClick={this.handleSubmit}
            >
              Sign In
            </Button>
          </Paper>
        </Container>
      );
    }
  }
}
function mapStateToProps(state) {
  return {
    isLoggingIn: state.auth.isLoggingIn,
    loginError: state.auth.loginError,
    isAuthenticated: state.auth.isAuthenticated,
  };
}
export default withStyles(styles)(connect(mapStateToProps)(Login));
```

26- Home Page

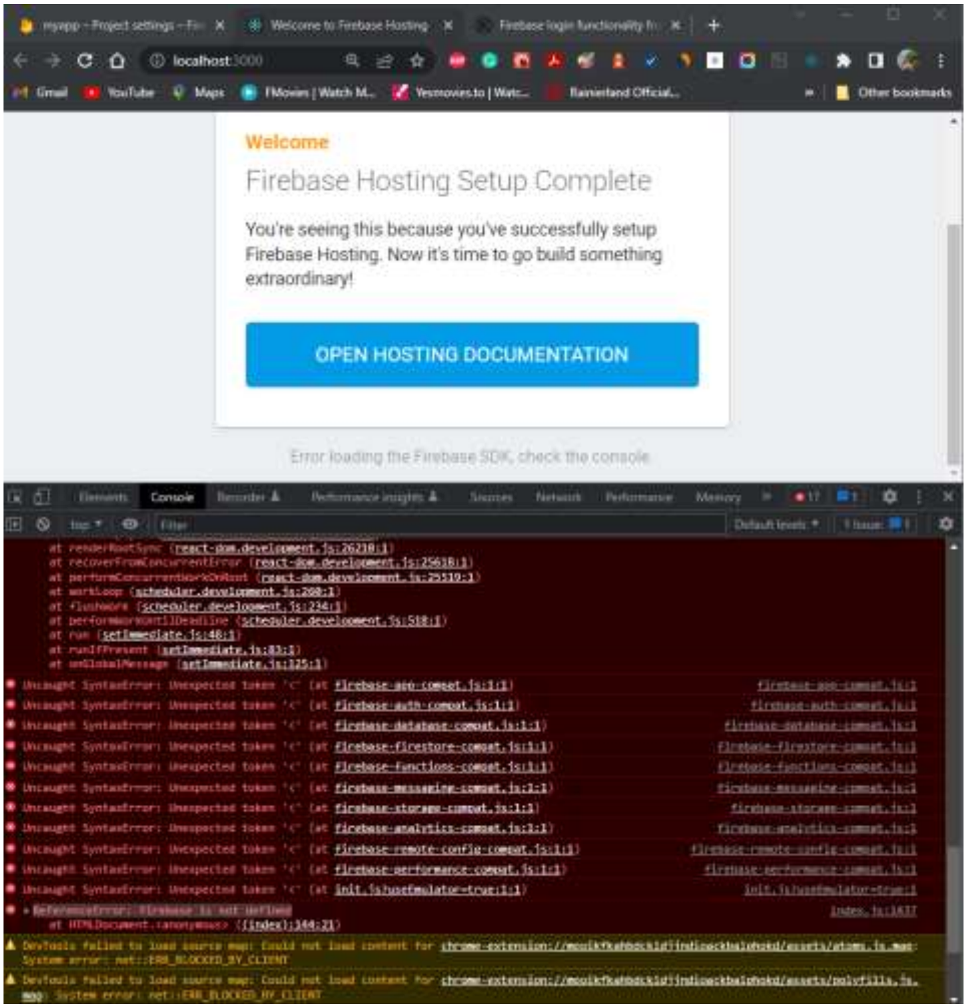


```
import React, { Component } from "react";
import { connect } from "react-redux";
import { logoutUser } from "../Actions/Auth";
class Home extends Component {
  handleLogout = () => {
    const { dispatch } = this.props;
    dispatch(logoutUser());
  };
  render() {
    const { isLoggingOut, logoutError } = this.props;
    return (
      <div>
        {" "}
        <h1>This is your app's protected area.</h1>{" "}
        <p>Any routes here will also be protected</p>{" "}
        <button onClick={this.handleLogout}>Log Out</button>{" "}
        {isLoggingOut && <p>Logging Out....</p>}{" "}
        {logoutError && <p>Error logging out</p>}{" "}
      </div>
    );
  }
}
function mapStateToProps(state) {
  return {
    isLoggingOut: state.auth.isLoggingOut,
    logoutError: state.auth.logoutError,
  };
}
export default connect(mapStateToProps)(Home);
```

27- ProtectedRoutes.js file

```
import React from "react";
import { Route, Navigate } from "react-router-dom";
const ProtectedRoute = ({
  component: Component,
  isAuthenticated,
  isVerifying,
  ...rest
}) => (
  <Route
    {...rest}
    render={(props) =>
      isVerifying ? (
        <div />
      ) : isAuthenticated ? (
        <Component {...props} />
      ) : (
        <Navigate
          to={{ pathname: "/login", state: { from: props.location } }}
        />
      )
    }
  />
);
export default ProtectedRoute;
```

28- Final results are error because the firebase object is not defined and the protected route can’t a child of the component Route



Welcome

Firebase Hosting Setup Complete

You're seeing this because you've successfully setup Firebase Hosting. Now it's time to go build something extraordinary!

Error loading the Firebase SDK, check the console.

```
    at renderRootSync (react-dom.development.js:26210:1)
    at recoverFromConcurrentError (react-dom.development.js:25618:1)
    at performConcurrentWorkOnRoot (react-dom.development.js:25519:1)
    at workLoop (scheduler.development.js:266:1)
    at flushWork (scheduler.development.js:234:1)
    at performWorkUntilDeadline (scheduler.development.js:518:1)
    at run (setImmediate.js:40:1)
    at runIfPresent (setImmediate.js:83:1)
    at onGlobalMessage (setImmediate.js:125:1)
● Uncaught SyntaxError: Unexpected token '<' (at firebase-app-compat.js:1:1)                firebase-app-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-auth-compat.js:1:1)              firebase-auth-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-database-compat.js:1:1)      firebase-database-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-firestore-compat.js:1:1)    firebase-firestore-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-functions-compat.js:1:1)    firebase-functions-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-messaging-compat.js:1:1)    firebase-messaging-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-storage-compat.js:1:1)        firebase-storage-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-analytics-compat.js:1:1)    firebase-analytics-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-remote-config-compat.js:1:1) firebase-remote-config-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at firebase-performance-compat.js:1:1) firebase-performance-compat.js:1
● Uncaught SyntaxError: Unexpected token '<' (at init.js?useEmulator=true:1:1)        init.js?useEmulator=true:1
● ▸ReferenceError: firebase is not defined                                                       index.js:1437
    at HTMLDocument.<anonymous> ((index):144:21)
▲ DevTools failed to load source map: Could not load content for chrome-extension://mooikfkahbdckldjjndioackbalphokd/assets/atoms.js.map:
  System error: net::ERR_BLOCKED_BY_CLIENT
▲ DevTools failed to load source map: Could not load content for chrome-extension://mooikfkahbdckldjjndioackbalphokd/assets/polyfills.js.
  map: System error: net::ERR_BLOCKED_BY_CLIENT
```