

```

import { initializeApp } from "firebase/app";
import { getFirestore } from "firebase/firestore/lite";
import {
  getAuth,
  signInWithEmailAndPassword,
  createUserWithEmailAndPassword,
  signOut,
  signInWithPopup,
  GoogleAuthProvider,
  FacebookAuthProvider,
} from "firebase/auth";
import "firebase/compat/auth";
import "firebase/compat/firestore";
import { getMessaging, getToken, onMessage } from "firebase/messaging";

import firebaseConfig, { REACT_APP_VAPID_KEY } from "configs/FirebaseConfig";
import { FIREBASE_TOKEN } from "constants/AuthConstant";

const firebaseApp = initializeApp(firebaseConfig);

const db = getFirestore(firebaseApp);
const auth = getAuth(firebaseApp);
const currentUser = auth.currentUser;
const googleAuthProvider = new GoogleAuthProvider();
const facebookAuthProvider = new FacebookAuthProvider();

const messaging = getMessaging(firebaseApp);

// export const onMessageListener = () =>
//   new Promise((resolve) => {
//     onMessage(messaging, (payload) => {
//       resolve(payload);
//     });
//   });

export {
  db,
  auth,
  messaging,
  currentUser,
  googleAuthProvider,
  GoogleAuthProvider,
  facebookAuthProvider,
  FacebookAuthProvider,
  signInWithEmailAndPassword,
  signOut,
  signInWithPopup,
  createUserWithEmailAndPassword,
};

export const generateToken = async () => {
  try {
    if (Notification.permission === "granted") {
      const currentToken = await getToken(messaging, {
        vapidKey: REACT_APP_VAPID_KEY,
      });

      if (currentToken) {
        localStorage.setItem(FIREBASE_TOKEN, currentToken);
      } else {
        console.log(
          "No registration token available. Request permission to generate one."
        );
      }
    } else {
      alert(
        "Bạn đã từ chối quyền truy cập thông báo. bạn sẽ không thể nhận được thông báo từ hệ thống ERP!"
      );
    }
  } catch (err) {
    console.log("An error occurred while retrieving token. ", err);
  }
}

```

```
}  
};
```