



Files

Search

Search files

public

> images

favicon.ico

placeholder.svg

robots.txt

src

> assets

pets...

> component...

> contexts

AuthC...

Them...

> hooks

> integrati...

> lib

> pages

> types

App.css

App.tsx

index.css

main.tsx

```
src/pages/Index.tsx           public/placeholder.svg        
```

1 import React, { createContext, useContext, useEffect, };
2 import { User, Session } from '@supabase/supabase-js';
3 import { supabase } from '@/integrations/supabase/client';
4 import { Profile } from '@/types';
5
6 interface AuthContextType {
7 user: User | null;
8 session: Session | null;
9 profile: Profile | null;
10 loading: boolean;
11 signUp: (email: string, password: string, username: string) => Promise<void>;
12 signIn: (email: string, password: string) => Promise<void>;
13 signOut: () => Promise<void>;
14 refreshProfile: () => Promise<void>;
15 }
16
17 const AuthContext = createContext<AuthContextType | undefined>();
18
19 export function AuthProvider({ children }: { children: React.ReactNode }) {
20 const [user, setUser] = useState<User | null>(null);
21 const [session, setSession] = useState<Session | null>(null);
22 const [profile, setProfile] = useState<Profile | null>(null);
23 const [loading, setLoading] = useState(true);
24
25 const fetchProfile = async (userId: string) => {
26 const { data } = await supabase
27 .from('profiles')
28 .select('*')
29 .eq('user_id', userId)
30 .single();
31
32 if (data) {
33 setProfile(data as Profile);
34 }
35 };
36
37 const refreshProfile = async () => {
38 if (user) {
39 await fetchProfile(user.id);
40 }
41 };
42
43 return {
44 user,
45 session,
46 profile,
47 loading,
48 refreshProfile,
49 AuthContext,
50 }
51 }
52
53 const App = () => {
54 const { user, session, profile, loading, refreshProfile } = useContext(AuthContext);
55
56 return (

```
vite-env.... 41 };  
42  
✓ supabase 43 useEffect(() => {  
44  
45 })
```