FONDAZIONE
ITSINCOM
ACADEMY
Tecnologie dell'informazione
e della comunicazione

CORSO: **BACKEND SYSTEM INTEGRATOR & FRONTEND DEVELOPER PER L'INDUSTRIA 4.0**

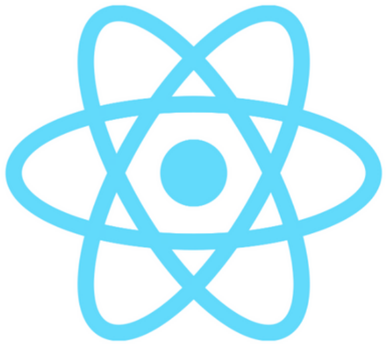
ANNO FORMATIVO: **2022/2023**

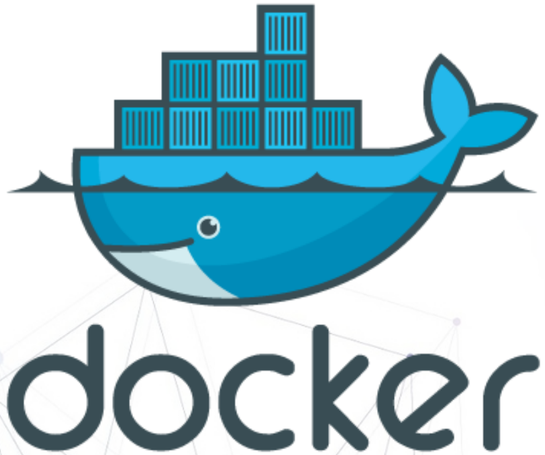
PROJECT WORK: **Laboratorio FullStack**
GRUPPO N: 1
COMPONENTI: Sabri Abdelmounaim - Ares Boscaro - Andrea Genovese

INDICE

Tecnologie Utilizzate

TECNOLOGIE UTILIZZATE





TECNOLOGIE UTILIZZATE

NextJS

NextjS è un framework JavaScript per applicazioni web front-end.
Le sue caratteristiche principali comprendono il rendering server-side e il supporto per applicazioni React.
Abbiamo optato per NextJS come framework front-end in quanto offre un ambiente di sviluppo agevole e permette la realizzazione di applicazioni web reattive e performanti.

Tailwind

Tailwind CSS è un framework CSS che offre un approccio utility-first.
È stato scelto e implementato per garantire un design rapido ed efficiente dell'applicazione, semplificando e velocizzando la creazione di pagine e interfacce.

TECNOLOGIE UTILIZZATE

Docker

Docker è una tecnologia che consente di creare un ambiente di esecuzione isolato per l'applicazione, garantendo la gestione coerente delle dipendenze e semplificando il processo di distribuzione su vari ambienti.

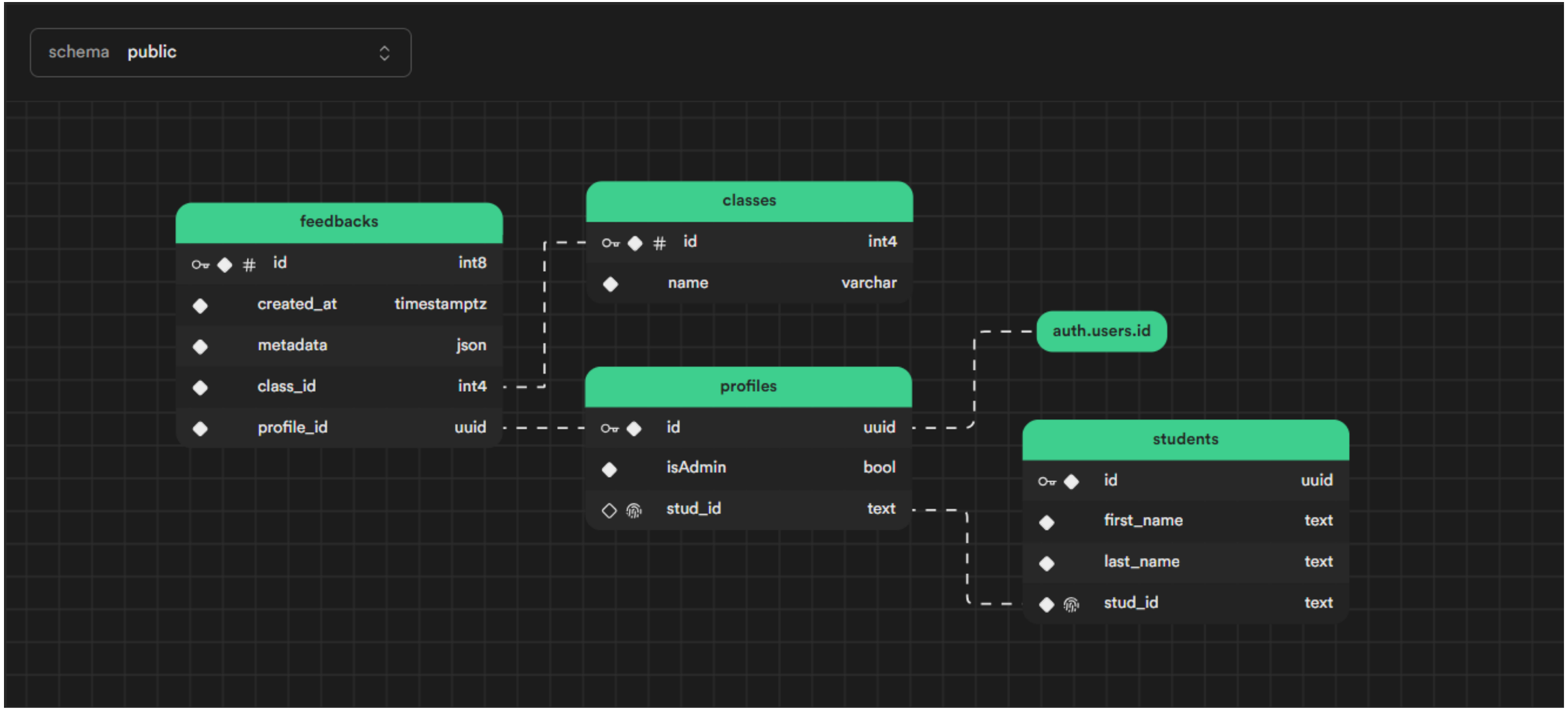
Supabase

Supabase è un servizio di database open-source basato su PostgreSQL che fornisce funzionalità di autenticazione, autorizzazione e archiviazione dei dati in tempo reale.
Abbiamo scelto Supabase per la sua facilità d’uso e sicurezza, nonché per la sua integrazione agevole con applicazioni web e la possibilità di essere facilmente raggiungibile anche da un container.

FONDAZIONE ITSINCOM ACADEMY
Tecnologie dell'informazione e della comunicazione

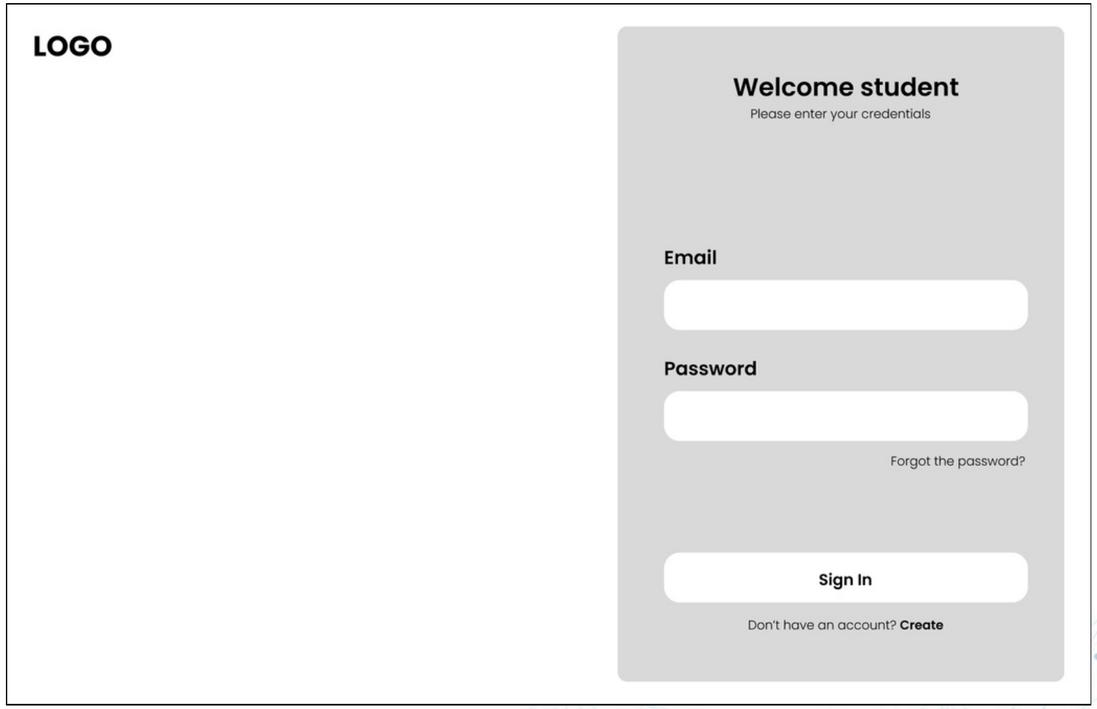
Progettazione del Database

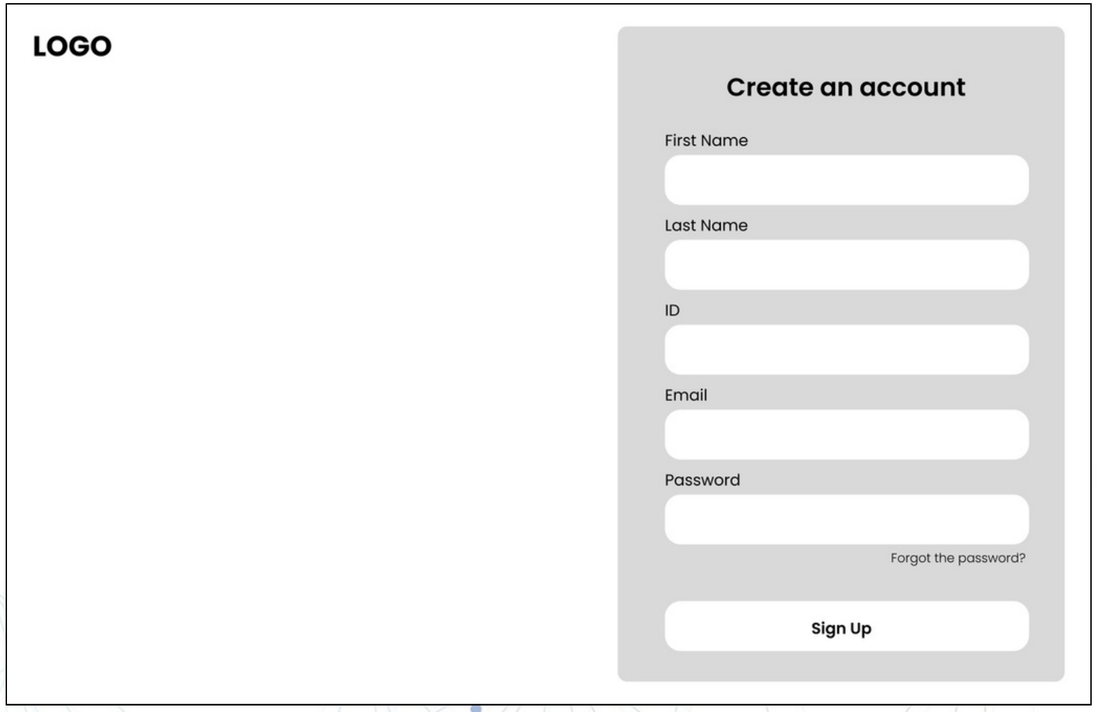
Progettazione del Database



Sviluppo interfaccia Web

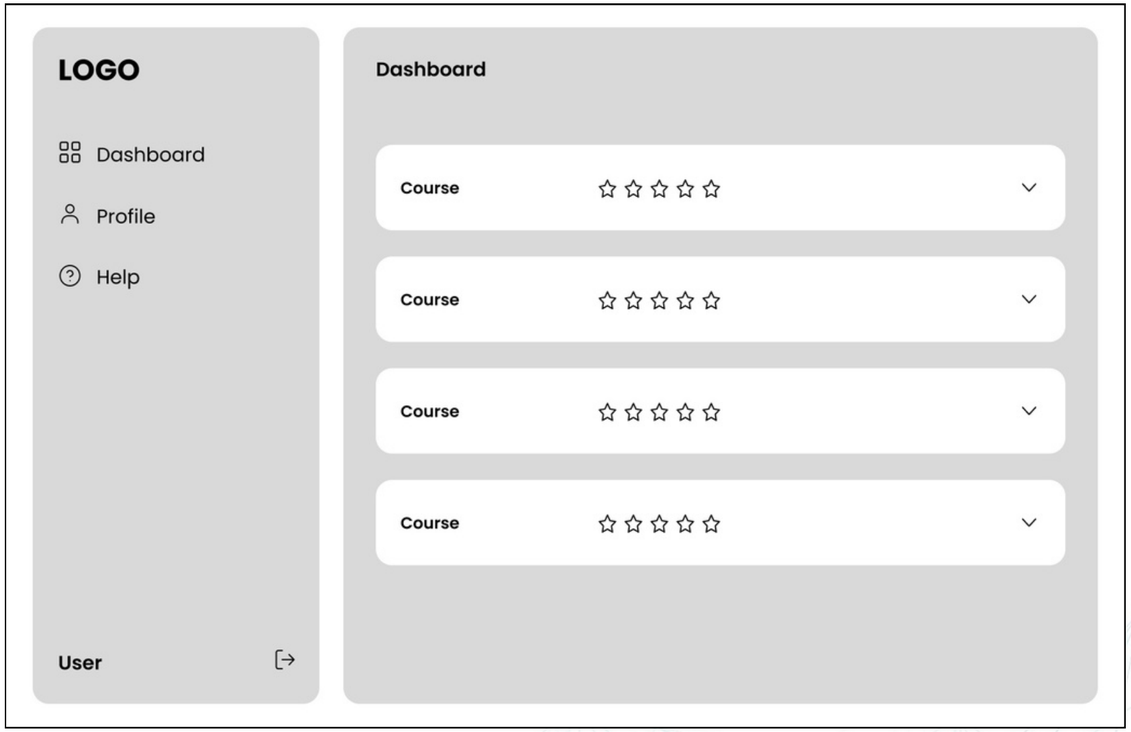
AUTH



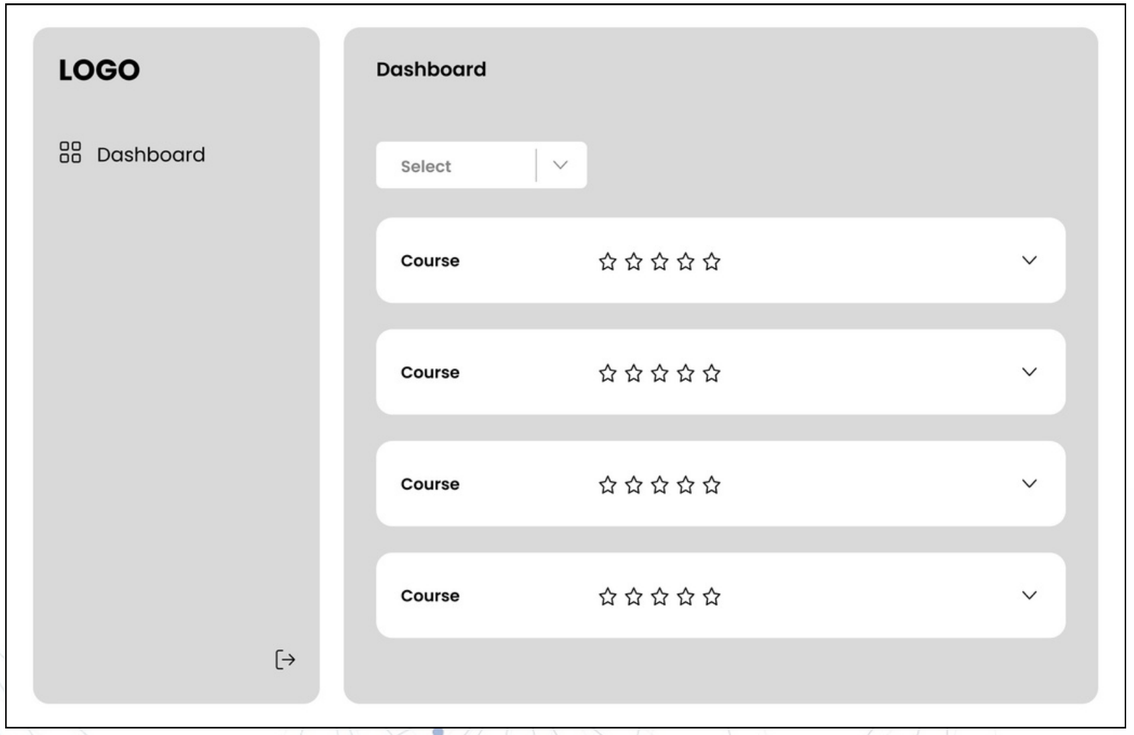


Sviluppo interfaccia Web

DASHBOARDS



Students



Admin

Implementazione Logiche

Implementazione Logiche

```
69    useEffect(() => {
70      handleUser();
71    }, []);
72
73    useEffect(() => {
74      if (User) {
75        getProfile();
76      }
77    }, [User]);
78
79    const router = useRouter();
80
81    useEffect(() => {
82      if (Profile) {
83        if (Profile.isAdmin) {
84          router.push("/admin");
85        } else {
86          router.push("/dashboard");
87        }
88      }
89    }, [Profile]);
```

```
useEffect(() => {
  if (!Profile) redirect("/");
}, [Profile]);
```

Snippets della logica principale di routing dell'app;
Viene inserita in un provider che wrappa tutta l'app e una piccola parte in un provider che wrappa esclusivamente le dashboards.

- La funzione handleUser() viene eseguita alla prima renderizzazione dell'app - in caso ci sia una sessione aperta recupera l'user loggato, altrimenti lo recupera appena si logga.
- Grazie allo useEffect che ascolta lo state User, appena c'è un utente loggato si recupera il profilo (che contiene il suo id, ruolo (admin o studente) - in caso sia studente anche nome, cognome, matricola e i feedbacks che ha rilasciato ai diversi corsi.
- Infine, nello useEffect che ascolta lo state Profile nel provider che wrappa tutta l'app quando arriva il profilo dal db, in base al suo ruolo lo redirecta nella dashboard apposita - nel provider che wrappa le dashboards nel caso non ci sia un profilo (quindi nessun userloggato) redirecta l'user alla pagina di Sign In (poichè le dashboards sono protected routes).

Implementazione Logiche

FONDAZIONE ITSINCOM ACADEMY
Tecnologie dell'informazione e della comunicazione

```
14  const handleUser = async () => {
15    const {
16      data: { user },
17    } = await supabaseClient.auth.getUser();
18
19    if (user) setUser(user);
20  };
21
22  const getProfile = async () => {
23    const { data: profileData } = await supabaseClient
24      .from("profiles")
25      .select()
26      .eq("id", User.id)
27      .single();
28
29    if (profileData && !profileData.isAdmin) {
30      const { data: studData } = await supabaseClient
31        .from("students")
32        .select()
33        .eq("stud_id", profileData.stud_id)
34        .single();
35
36      if (studData) {
37        setProfile({
38          id: profileData.id,
39          isAdmin: profileData.isAdmin,      You, 2 days ago •
40          first_name: studData.first_name,
41          last_name: studData.last_name,
42          stud_id: studData.stud_id,
43        });
44        getFeedbacks();
45      }
46    } else if (profileData && profileData.isAdmin) {
47      setProfile({
48        id: profileData.id,
49        isAdmin: profileData.isAdmin,
50      });
51    }
52  };
```

```
54    const getFeedbacks = async () => {
55      const { data } = await supabaseClient          You,
56        .from("feedbacks")
57        .select("id, metadata, class_id")
58        .eq("profile_id", User.id);
59
60      if (data)
61        setProfile((prev) => {
62          return {
63            ...prev,
64            feedbacks: data,
65          };
66        });
67    };
68
```

Grazie

Per l'attenzione!!!