

## Relazione progetto ReactReminder Nabil Boudlal

ReactReminder è una web app realizzata in React e Next js, lo scopo principale di questa applicazione è quello di permettere ad un utente di inserire dei reminder all'interno della sua area personale e poterli: modificare, cancellare o contrassegnare come completati, l'applicazione si ispira quindi alle Todo App presenti nei più famosi store.

### Panoramica Generale del Funzionamento

- **Autenticazione:** Gli utenti possono registrarsi e accedere utilizzando Firebase Authentication. L'autenticazione è richiesta per accedere alle funzionalità principali dell'app.
- **Gestione dei Compiti (Reminders):** Gli utenti possono aggiungere nuovi reminders nella loro area personale. Possono anche modificarli, eliminarli e contrassegnarli come completati.
- **Design Reattivo:** L'app presenta un design moderno e reattivo grazie all'utilizzo di Tailwind CSS, garantendo un'esperienza utente uniforme su dispositivi desktop e mobili.

### Librerie Utilizzate

React Next App è stato sviluppato utilizzando diverse librerie chiave per garantire funzionalità avanzate e una user experience di alta qualità:

- **React:** La libreria principale per la creazione di interfacce utente reattive e component-based.
- **Next.js:** Un framework per React che semplifica il rendering del lato server, il routing e l'ottimizzazione delle prestazioni.
- **Firebase:** Una piattaforma di sviluppo mobile e web basata su cloud che è stata utilizzata per l'autenticazione degli utenti e la gestione del database.
- **Tailwind CSS:** Un framework CSS utilizzato per la creazione di una UI moderna e reattiva con facilità.

### Gestione dell'Autenticazione

ReactReminder utilizza Firebase per gestire l'autenticazione degli utenti. Firebase semplifica notevolmente l'implementazione dell'autenticazione, consentendo agli utenti di registrarsi, accedere e reimpostare la password.

Vengono utilizzati i metodi Firebase, come `createUserWithEmailAndPassword()`, `signInWithEmailAndPassword()` e `signOut()`, per implementare l'autenticazione nel codice dell'app.

Proteggere le rotte dell'app in base all'autenticazione dell'utente, garantendo che solo gli utenti autenticati possano accedere alle funzionalità riservate.

## Gestione del Database

Firebase è stato utilizzato anche per la gestione del database dell'applicazione. In particolare, è stato utilizzato Firebase Firestore, un database NoSQL basato su documenti. Firestore è altamente scalabile e consente di archiviare dati strutturati in collezioni e documenti. In ReactReminder i reminder degli utenti vengono archiviati in Firestore.

Le operazioni CRUD (Create, Read, Update, Delete) per i compiti sono state implementate utilizzando le API Firestore, tra cui *addDoc*, *updateDoc*, e *deleteDoc*. Gli utenti possono aggiungere nuovi compiti, modificarli, eliminarli e contrassegnarli come completati in tempo reale. La funzionalità di aggiornamento in tempo reale è stata resa possibile grazie all'utilizzo di *onSnapshot*, che consente all'app di reagire automaticamente ai cambiamenti nel database senza dover ricaricare la pagina.