Spiegazione codice del progetto (javascript)

Progetto 1 – M306

Samuele Ambrosetti

2024

Sommario

[Introduzione 2](#_Toc180612345)

[Configurazione firebase 2](#_Toc180612346)

[1 – Importazione dei Moduli 2](#_Toc180612347)

[2 – Configurazione di Firebase 2](#_Toc180612348)

[3 – Inizializzazione di Firebase 2](#_Toc180612349)

[Auth.js 3](#_Toc180612350)

[1 – Funzioni di Autenticazione 3](#_Toc180612351)

[1.1 – Login con Google 3](#_Toc180612352)

[1.2 – Login con Email e Password 3](#_Toc180612353)

[1.3 – Registrazione di Nuovi Utenti 4](#_Toc180612354)

[Index.js 5](#_Toc180612355)

[1 – Verificare lo stato di autenticazione 5](#_Toc180612356)

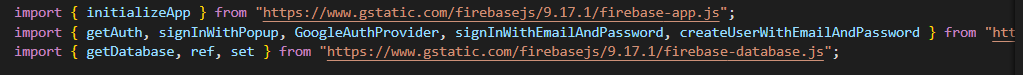
[2 – Funzione di logout 5](#_Toc180612357)

# Introduzione

Questo documento serve a spiegare come funziona il codice di firebase generalmente. Verrà spiegato come funziona l’attivazione di firebase nel progetto e approfondito il funzionamento dei metodi.

# Configurazione firebase

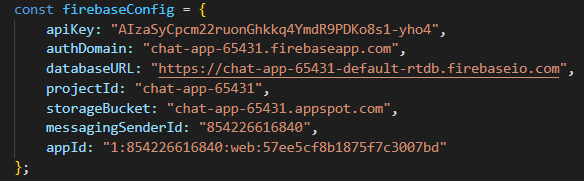
## 1 – Importazione dei Moduli



**Descrizione dei Moduli Importati:**

* **firebase-app.js**: Questo modulo consente di inizializzare l'app Firebase.
* **firebase-auth.js**: Include funzioni per gestire l'autenticazione degli utenti, sia tramite Google che tramite email e password.
* **firebase-database.js**: Fornisce funzioni per interagire con il database Firebase, consentendo di salvare e recuperare dati.

## 2 – Configurazione di Firebase

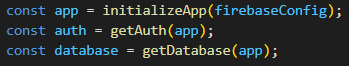


**Descrizione dei Parametri:**

* **apiKey**: Chiave API per autorizzare le richieste.
* **authDomain**: Dominio utilizzato per l'autenticazione.
* **databaseURL**: URL del database Firebase.
* **projectId**: ID del progetto Firebase.
* **storageBucket**: Bucket di archiviazione di Firebase.
* **messagingSenderId**: ID del mittente per i messaggi.
* **appId**: ID dell'applicazione Firebase.

## 3 – Inizializzazione di Firebase

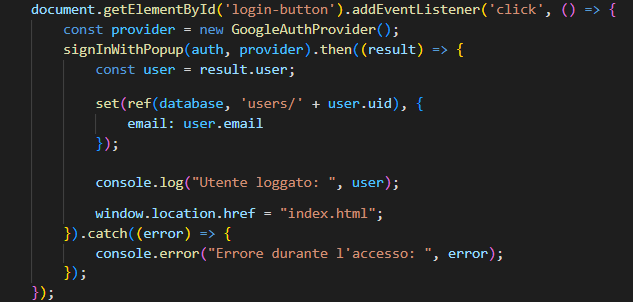
Vengono creati tre oggetti che saranno utilizzati per l’interazione con firebase, in particolare “app” rappresenta l’applicazione di firebase avviata, “auth” rappresenta l’utente autenticato mentre “database” rappresenta il realtime database di firebase, usato per inviare o richiedere dati.



# Auth.js

## 1 – Funzioni di Autenticazione

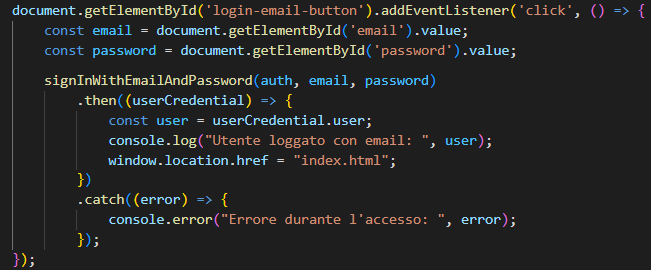
### 1.1 – Login con Google



**Descrizione:**

* **addEventListener**: Aggiunge un evento al pulsante di login.
* **GoogleAuthProvider**: Crea un provider (oggetto) per gestire il login tramite Google.
* **signInWithPopup**: Avvia il processo di login tramite una finestra esterna.
* **set**: Salva l'email dell'utente nel database di Firebase.
* **window.location.href**: Reindirizza l'utente alla pagina principale.

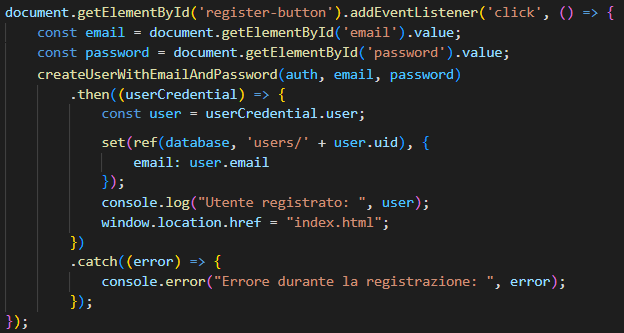
### 1.2 – Login con Email e Password



**Descrizione:**

* **signInWithEmailAndPassword**: Autentica l'utente utilizzando email e password.
* **console.log**: Visualizza informazioni sull'utente loggato.
* **window.location.href**: Reindirizza l'utente alla pagina principale.

### 1.3 – Registrazione di Nuovi Utenti

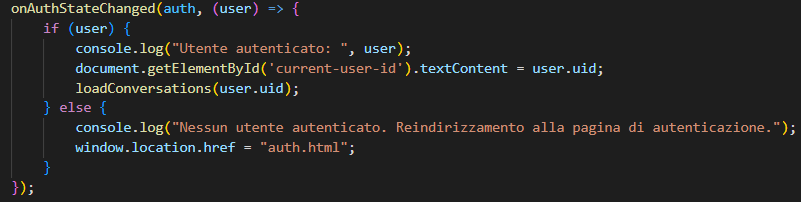


**Descrizione:**

* **createUserWithEmailAndPassword**: Metodo di firebase per creare un utente con email e password
* **set**: Salva l'email dell'utente nel database di Firebase.
* **console.log**: Visualizza informazioni sull'utente registrato.
* **window.location.href**: Reindirizza l'utente alla pagina principale.

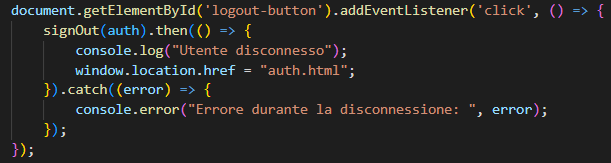
# Index.js

## 1 – Verificare lo stato di autenticazione



Questa funzione controlla se un utente è autenticato. Se sì, carica le conversazioni; altrimenti, reindirizza l'utente alla pagina di autenticazione.

## 2 – Funzione di logout



Questo metodo serve a disconnettere l’utente dalla sessione corrente, in seguito reindirizzandolo alla pagina di login.

**Descrizione:**

* **signOut:** metodo di firebase per invalidare l’autenticazione corrente.
* **window.location.href**: Reindirizza l'utente alla pagina di autenticazione.