



FirebaseConnect

Comprehensive API

Documentation

1. Introduction

2. Authentication

[Authentication Endpoints](#)

[Authentication Methods](#)

3. Firestore

[Firestore Endpoints](#)

[Firestore Methods](#)

4. Realtime Database

[Realtime Database Endpoints](#)

[Realtime Database Methods](#)

5. Storage

[Storage Endpoints](#)

[Storage Methods](#)

6. Conclusion

Welcome to the comprehensive documentation for FirebaseConnect, an advanced API designed to streamline integration with Firebase

services. This document aims to provide an in-depth understanding of FirebaseConnect's endpoints, methods, and functionalities for seamless interaction with Firebase Authentication, Cloud Firestore, Firebase Realtime Database, and Firebase Cloud Storage.

1. Introduction

FirebaseConnect is an advanced API that simplifies integration with Firebase services, providing developers with powerful tools to manage user authentication, data storage, and real-time communication. This comprehensive documentation aims to guide new users through FirebaseConnect's extensive features and functionalities.

2. Authentication

FirebaseConnect's Authentication module offers a robust set of endpoints and methods for managing user authentication with Firebase Authentication.

Authentication Endpoints

- `/auth/signup` : Creates a new user account.
- `/auth/login` : Logs in with an existing user account.
- `/auth/logout` : Logs out the currently authenticated user.
- `/auth/reset-password` : Resets the password for a user account.
- `/auth/verify-email` : Sends a verification email to the user's email address.

Authentication Methods

- `signup(userData)` : Creates a new user account with the provided user data.
- `login(email, password)` : Logs in with the specified email and password.

- `logout()` : Logs out the currently authenticated user.
- `resetPassword(email)` : Resets the password for the specified email address.
- `verifyEmail()` : Sends a verification email to the user's email address.

3. Firestore

FirebaseConnect's Firestore module provides seamless integration with Cloud Firestore, offering intuitive endpoints and methods for efficient data management.

Firestore Endpoints

- `/firestore/create` : Creates a new document in the Firestore collection.
- `/firestore/read` : Reads a document from the Firestore collection.
- `/firestore/update` : Updates a document in the Firestore collection.
- `/firestore/delete` : Deletes a document from the Firestore collection.

Firestore Methods

- `create(data)` : Creates a new document in the Firestore collection with the provided data.
- `read()` : Reads a document from the Firestore collection.
- `update(data)` : Updates the document in the Firestore collection with the provided data.
- `delete()` : Deletes the document from the Firestore collection.

4. Realtime Database

FirebaseConnect's Realtime Database module simplifies real-time data synchronization, offering intuitive endpoints and methods for seamless communication.

Realtime Database Endpoints

- `/realtime/create` : Creates a new item in the Realtime Database.
- `/realtime/read` : Reads items from the Realtime Database.
- `/realtime/update` : Updates an item in the Realtime Database.
- `/realtime/delete` : Deletes an item from the Realtime Database.

Realtime Database Methods

- `create(data)` : Creates a new item in the Realtime Database with the provided data.
- `read()` : Reads items from the Realtime Database.
- `update(id, newData)` : Updates the specified item in the Realtime Database with the new data.
- `delete(id)` : Deletes the specified item from the Realtime Database.

5. Storage

FirebaseConnect's Storage module offers seamless file storage with Firebase Cloud Storage, providing intuitive endpoints and methods for efficient file management.

Storage Endpoints

- `/storage/upload` : Uploads a file to Firebase Storage.
- `/storage/download` : Downloads a file from Firebase Storage.
- `/storage/delete` : Deletes a file from Firebase Storage.

Storage Methods

- `uploadFile(file, path)` : Uploads the specified file to Firebase Storage at the given path.
- `downloadFile(path)` : Downloads the file from Firebase Storage at the specified path.

- `deleteFile(path)` : Deletes the file from Firebase Storage at the specified path.

6. Conclusion

FirebaseConnect provides developers with advanced tools and functionalities to seamlessly integrate with Firebase services, enabling the development of robust and scalable applications with ease.