



# Documentazione dell'API SmartLinx

## Indice

### 1. Introduzione

---

### 2. Accesso all'API

---

### 3. Gestione dell'autenticazione

---

### 4. Codici di errore generici

---

### 5. Metodi dell'API

#### 1. Aggiunta di un nuovo utente

---

#### 2. Ottenere un utente tramite email

---

#### 3. Aggiornamento di un utente esistente

---

#### 4. Eliminazione di un utente

---

#### 5. Aggiunta di una nuova casa

---

#### 6. Ottenere le case di un utente

---

#### 7. Aggiornamento di una casa esistente



---

8. Eliminazione di una casa

---

9. Impostare l'hub di una casa

---

10. Aggiunta di un nuovo membro della famiglia

---

11. Ottenere i membri della famiglia di una casa

---

12. Rimozione di un membro della famiglia

---

13. Aggiunta di una nuova stanza

---

14. Ottenere le stanze di una casa

---

15. Aggiornamento di una stanza esistente

---

16. Eliminazione di una stanza

---

17. Aggiunta di un nuovo dispositivo

---

18. Ottenere i dispositivi di una stanza

---

19. Aggiornamento di un dispositivo esistente

---

20. Eliminazione di un dispositivo

---

21. Associazione di un dispositivo MQTT all'hub



---

22. Ottenere lo stato di un dispositivo MQTT

---

23. Impostare lo stato di un dispositivo MQTT

---

24. Disassociazione di un dispositivo MQTT dall'hub

---

25. Aggiunta di un dispositivo ai preferiti

---

26. Ottenere i dispositivi preferiti

---

27. Rimozione di un dispositivo dai preferiti

---

28. Ottenere le informazioni dell'utente

---

## 1 - Introduzione

L'API REST fornisce servizi per gestire il backend del servizio SmartLinx. Questa API consente di interagire con gli utenti, le case, le stanze, i dispositivi e gli hub associati all'ecosistema SmartLinx.

## 2 - Accesso all'API

L'API SmartLinx è accessibile tramite richieste HTTP inviate all'indirizzo **smartlinx.it:5500**.

Tutte le richieste devono includere il token di autenticazione e l'ID utente (UID) negli header o nei parametri della richiesta.



## 3 - Gestione dell'autenticazione

L'autenticazione avviene utilizzando un token JWT (JSON Web Token) generato da Firebase Authentication. Il token deve essere incluso in tutte le richieste come parametro di query con il nome **token**, insieme all'UID dell'utente come parametro di query con il nome **uid**.

## 4 - Codici di errore generici

- **400 Bad Request:** indica che la richiesta inviata al server è errata
- **401 Unauthorized:** Indica che la richiesta non è autorizzata perché il token di autenticazione è mancante, non valido o scaduto.
- **404 Not Found:** Questo codice di errore viene restituito quando il server non trova la risorsa richiesta. Può essere causato da un URI errato o da una risorsa che non esiste sul server.
- **405 Method Not Allowed:** Indica che il metodo HTTP utilizzato nella richiesta non è supportato per la risorsa richiesta. Ad esempio, potrebbe essere restituito se si tenta di eseguire una richiesta POST su un endpoint che supporta solo metodi GET.
- **500 Internal Server Error:** Indica un errore interno del server durante l'esecuzione della richiesta inviata.

## 5 - Metodi dell'API

### 5.1. Aggiunta di un nuovo utente

- **Path:** /smartlinx/user
- **Metodo:** POST
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**



- **token:** Token di autenticazione JWT.
- **uid:** UID dell'utente.
- **Body:** Oggetto JSON rappresentante il nuovo utente.
- **Risposta:**
  - **201 Created** se l'utente è stato creato con successo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.2. Ottenere un utente tramite email

- **Path:** /smartlinx/user
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un oggetto JSON rappresentante l'utente.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.

## 5.3. Aggiornamento di un utente esistente

- **Path:** /smartlinx/user
- **Metodo:** PUT
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**



- **token:** Token di autenticazione JWT.
- **uid:** UID dell'utente.
- **Body:** Oggetto JSON contenente i dati aggiornati dell'utente.
- **Risposta:**
  - **200 OK** se l'aggiornamento è avvenuto con successo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.4. Eliminazione di un utente

- **Path:** /smartlinx/user
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se l'eliminazione è avvenuta con successo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.5. Aggiunta di una nuova casa

- **Path:** /smartlinx/home
- **Metodo:** POST
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.



- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Body:** Oggetto JSON rappresentante la nuova casa.
- **Risposta:**
  - **201 Created** se la casa è stata creata con successo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.6. Ottenere le case di un utente

- **Path:** /smartlinx/home
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un array JSON contenente le case associate all'utente.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.

## 5.7. Aggiornamento di una casa esistente

- **Path:** /smartlinx/home
- **Metodo:** PUT
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.



- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Body:** Oggetto JSON contenente i dati aggiornati della casa.
- **Risposta:**
  - **200 OK** se l'aggiornamento è avvenuto con successo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.8. Eliminazione di una casa

- **Path:** /smartlinx/home
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **homeId:** ID della casa da eliminare.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se l'eliminazione è avvenuta con successo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.9. Impostare l'hub di una casa

- **Path:** /smartlinx/home/hub





- **Metodo:** PUT
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **homelid:** ID della casa.
  - **hubid:** ID dell'hub da associare alla casa.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se l'hub è stato impostato con successo per la casa.
  - **404 Not Found** se l'hub specificato non è stato trovato.
  - **500 Internal Server Error** se si verifica un errore durante l'operazione di aggiornamento nel database o durante la connessione al broker MQTT.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.10. Aggiunta di un nuovo membro della famiglia

- **Path:** /smartlinx/familyMember
- **Metodo:** POST
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **mail:** Indirizzo email del nuovo membro della famiglia.
  - **homelid:** ID della casa.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **201 Created** se il membro della famiglia è stato aggiunto con successo.



- **401 Unauthorized** se l'utente non ha i diritti per aggiungere membri alla casa.
- **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.11. Ottenere i membri della famiglia di una casa

- **Path:** /smartlinx/familyMember
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **homeId:** ID della casa.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un array JSON contenente i membri della famiglia associati alla casa.
- **Codici di Errore:**
  - **401 Unauthorized** se l'utente non ha i diritti per visualizzare i membri della famiglia della casa.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.

## 5.12. Rimozione di un membro della famiglia

- **Path:** /smartlinx/familyMember
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **mail:** Indirizzo email del membro della famiglia da rimuovere.



- **homelid:** ID della casa.
- **token:** Token di autenticazione JWT.
- **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se il membro della famiglia è stato rimosso con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per rimuovere membri dalla casa.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.13. Aggiunta di una nuova stanza

- **Path:** /smartlinx/room
- **Metodo:** POST
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **homelid:** ID della casa.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Body:** Oggetto JSON rappresentante la nuova stanza.
- **Risposta:**
  - **201 Created** se la stanza è stata creata con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per aggiungere stanze alla casa.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.



## 5.14. Ottenere le stanze di una casa

- **Path:** /smartlinx/room
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **homeld:** ID della casa.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un array JSON contenente le stanze associate alla casa.
- **Codici di Errore:**
  - **401 Unauthorized** se l'utente non ha i diritti per visualizzare le stanze della casa.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.

## 5.15. Aggiornamento di una stanza esistente

- **Path:** /smartlinx/room
- **Metodo:** PUT
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Body:** Oggetto JSON contenente i dati aggiornati della stanza.
- **Risposta:**
  - **200 OK** se l'aggiornamento è avvenuto con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per aggiornare la stanza.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**



- **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.16. Eliminazione di una stanza

- **Path:** /smartlinx/room
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **roomId:** ID della stanza da eliminare.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se la stanza è stata eliminata con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per eliminare la stanza.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.17. Aggiunta di un nuovo dispositivo

- **Path:** /smartlinx/device
- **Metodo:** POST
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Body:** Oggetto JSON rappresentante il nuovo dispositivo.
- **Risposta:**



- **201 Created** se il dispositivo è stato creato con successo.
- **401 Unauthorized** se l'utente non ha i diritti per aggiungere dispositivi alla stanza.
- **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.18. Ottenere i dispositivi di una stanza

- **Path:** /smartlinx/device
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **roomId:** ID della stanza.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un array JSON contenente i dispositivi associati alla stanza.
- **Codici di Errore:**
  - **401 Unauthorized** se l'utente non ha i diritti per visualizzare i dispositivi della stanza.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.

## 5.19. Aggiornamento di un dispositivo esistente

- **Path:** /smartlinx/device
- **Metodo:** PUT
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.



- **uid:** UID dell'utente.
- **Body:** Oggetto JSON contenente i dati aggiornati del dispositivo.
- **Risposta:**
  - **200 OK** se l'aggiornamento è avvenuto con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per aggiornare il dispositivo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.20. Eliminazione di un dispositivo

- **Path:** /smartlinx/device
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **deviceId:** ID del dispositivo da eliminare.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se il dispositivo è stato eliminato con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per eliminare il dispositivo.
  - **500 Internal Server Error** se si verifica un errore nel database SQL.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.21. Associazione di un dispositivo MQTT all'hub

- **Path:** /smartlinx/device/mqtt



- **Metodo:** POST
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **hubId:** ID dell'hub MQTT.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se il dispositivo è stato associato all'hub con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per associare dispositivi all'hub.
  - **500 Internal Server Error** se si verifica un errore durante l'associazione.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.22. Ottenere lo stato di un dispositivo MQTT

- **Path:** /smartlinx/device/mqtt
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **hubId:** ID dell'hub MQTT.
  - **ieee\_address:** Indirizzo IEEE del dispositivo MQTT.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** con lo stato corrente del dispositivo MQTT.
  - **401 Unauthorized** se l'utente non ha i diritti per ottenere lo stato del dispositivo MQTT.





- **500 Internal Server Error** se si verifica un errore durante la richiesta dello stato.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

### 5.23. Impostare lo stato di un dispositivo MQTT

- **Path:** /smartlinx/device/mqtt
- **Metodo:** PUT
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Body:** Oggetto JSON contenente l'ID dell'hub MQTT, l'indirizzo IEEE del dispositivo e il comando da inviare.
- **Risposta:**
  - **200 OK** se il comando è stato inviato con successo al dispositivo MQTT.
  - **401 Unauthorized** se l'utente non ha i diritti per impostare lo stato del dispositivo MQTT.
  - **500 Internal Server Error** se si verifica un errore durante l'invio del comando.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

### 5.24. Disassociazione di un dispositivo MQTT dall'hub

- **Path:** /smartlinx/device/mqtt
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**



- **hubId**: ID dell'hub MQTT.
- **ieee\_address**: Indirizzo IEEE del dispositivo MQTT.
- **token**: Token di autenticazione JWT.
- **uid**: UID dell'utente.
- **Risposta**:
  - **200 OK** se il dispositivo è stato disassociato dall'hub con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per disassociare dispositivi dall'hub.
  - **500 Internal Server Error** se si verifica un errore durante la disassociazione.
- **Codici di Errore**:
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.

## 5.25. Aggiunta di un dispositivo ai preferiti

- **Path**: /smartlinx/favourite
- **Metodo**: POST
- **Gestione Autenticazione**: Richiede un token di autenticazione valido.
- **Parametri Query**:
  - **deviceId**: ID del dispositivo da aggiungere ai preferiti.
  - **token**: Token di autenticazione JWT.
  - **uid**: UID dell'utente.
- **Risposta**:
  - **201 Created** se il dispositivo è stato aggiunto ai preferiti con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per aggiungere dispositivi ai preferiti.
  - **500 Internal Server Error** se si verifica un errore durante l'aggiunta.
- **Codici di Errore**:
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.



## 5.26. Ottenere i dispositivi preferiti

- **Path:** /smartlinx/favourite
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un array JSON contenente gli ID dei dispositivi preferiti.
- **Codici di Errore:**
  - **401 Unauthorized** se l'utente non ha i diritti per visualizzare i dispositivi preferiti.
  - **500 Internal Server Error** se si verifica un errore nel recupero dei dati.

## 5.27. Rimozione di un dispositivo dai preferiti

- **Path:** /smartlinx/favourite
- **Metodo:** DELETE
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **deviceId:** ID del dispositivo da rimuovere dai preferiti.
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:**
  - **200 OK** se il dispositivo è stato rimosso dai preferiti con successo.
  - **401 Unauthorized** se l'utente non ha i diritti per rimuovere dispositivi dai preferiti.
  - **500 Internal Server Error** se si verifica un errore durante la rimozione.
- **Codici di Errore:**
  - **401 Unauthorized** se il token di autenticazione è mancante o non valido.



## 5.28. Ottenere le informazioni dell'utente

- **Path:** /smartlinx/user/info
- **Metodo:** GET
- **Gestione Autenticazione:** Richiede un token di autenticazione valido.
- **Parametri Query:**
  - **token:** Token di autenticazione JWT.
  - **uid:** UID dell'utente.
- **Risposta:** Restituisce un oggetto JSON contenente le informazioni dell'utente, come ad esempio nome, cognome, e-mail, etc.
- **Codici di Errore:**
  - **401 Unauthorized** se l'utente non ha i diritti per accedere alle proprie informazioni.
  - **500 Internal Server Error** se si verifica un errore nel recupero delle informazioni dell'utente.