

Documentazione API

Introduzione

Questa documentazione dettagliata descrive tutte le API disponibili nel sistema Trekkigram, un'applicazione social dedicata agli appassionati di trekking. Le API sono organizzate in base alle funzionalità principali del sistema.

Informazioni Generali

- **Base URL:** `http://trekkigram.com:8080`
- **Formato dati:** JSON
- **Autenticazione:** Session Token basato su cookie
- **CORS:** Abilitato per origine `http://trekkigram.com`
CORS - Access-Control-Allow-Origin è un meccanismo di sicurezza dei browser web che permette di controllare l'accesso a risorse provenienti da domini diversi; Nel caso di Trekkigram le API si trovano sulla porta 8080 mentre il frontend sulla porta 80, hanno quindi origini diverse e dobbiamo permettere richieste dal frontend

Autenticazione

Tutte le API (eccetto quelle di login e registrazione) richiedono un token di sessione valido che viene verificato attraverso la funzione `verifySessionToken()`. Il token viene fornito automaticamente nei cookie quando l'utente effettua il login.

Struttura delle risposte

Tutte le risposte API hanno il seguente formato:

```
{
  "success": true|false,
  "message": "Messaggio descrittivo",
  "data": { ... } // Dati specifici dell'endpoint (opzionale)
}
```

Autenticazione e Gestione Utenti (`/auth`)

1. Login

- **Endpoint:** `POST /auth/login`
- **Descrizione:** Autentica un utente e crea una sessione
- **Parametri:**

```
{
  "email": "utente@esempio.com",
  "password": "password"
}
```

- **Risposte:**

- `200 OK`:

```
{
  "success": true,
  "message": "Login effettuato con successo",
}
```

```

"data": {
  "success": true,
  "session_token": "token-di-sessione",
  "userinfo": {
    "UserID": 123,
    "Username": "nomeutente"
  }
}
}

```

- **401 Unauthorized** :

```

{
  "success": false,
  "message": "Credenziali non valide"
}

```

2. Registrazione

- **Endpoint:** **POST /auth/register**
- **Descrizione:** Registra un nuovo utente
- **Parametri (Form-Data):**
 - **email** : Email dell'utente
 - **name** : Nome
 - **surname** : Cognome
 - **username** : Nome utente
 - **password** : Password
 - **confirmPassword** : Conferma password
 - **privacy** : Impostazioni privacy ("1" per privato, "0" per pubblico)
 - **profileImage** : File immagine profilo (opzionale)
- **Risposte:**
 - **200 OK** :

```

{
  "success": true,
  "message": "Registrazione effettuata con successo",
  "data": {
    "success": true,
    "session_token": "token-di-sessione",
    "userinfo": {
      "UserID": 123,
      "Username": "nomeutente"
    }
  }
}

```

- **400 Bad Request** : Campi mancanti o password non corrispondenti
- **409 Conflict** : Email o username già in uso
- **500 Internal Server Error** : Errore durante la registrazione

3. Logout

- **Endpoint:** `DELETE /auth/logout`
- **Descrizione:** Termina la sessione dell'utente
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- `200 OK` :

```
{
  "success": true,
  "message": "Logout effettuato con successo"
}
```

- `401 Unauthorized` : Sessione non valida

4. Verifica Sessione

- **Endpoint:** `GET /auth/check-session`
- **Descrizione:** Verifica se una sessione è valida
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- `200 OK` (sessione valida):

```
{
  "success": true,
  "message": "Session valid",
  "data": {
    "authenticated": true,
    "userinfo": {
      "UserID": 123,
      "Username": "nomeutente"
    }
  }
}
```

- `200 OK` (sessione non valida):

```
{
  "success": false,
  "message": "Session invalid or expired",
  "data": {
    "authenticated": false
  }
}
```

Gestione Eventi (`/events`)

1. Creazione di un Evento

- **Endpoint:** `POST /events/create`
- **Descrizione:** Crea un nuovo evento di trekking
- **Autenticazione:** Token di sessione richiesto

- **Parametri** (Form-Data):

- `eventName` : Nome dell'evento
- `eventDescription` : Descrizione dell'evento (opzionale)
- `eventLocation` : Luogo dell'evento
- `eventDifficulty` : Difficoltà (facile, media, difficile)
- `eventDateTime` : Data e ora dell'evento (formato YYYY-MM-DD HH:MM:SS)
- `Latitude` : Latitudine del luogo (opzionale)
- `Longitude` : Longitudine del luogo (opzionale)
- `eventPhotos[]` : Array di file immagine per l'evento

- **Risposte:**

- `200 OK` :

```
{
  "success": true,
  "message": "Evento creato con successo"
}
```

- `400 Bad Request` : Dati mancanti
- `500 Internal Server Error` : Errore durante la creazione

2. Recupero dei Post

- **Endpoint:** `GET /events/posts`

- **Descrizione:** Recupera gli eventi da visualizzare nella home

- **Autenticazione:** Token di sessione richiesto

- **Parametri Query:**

- `page` : Numero pagina (default: 1)
- `perPage` : Post per pagina (default: 3)
- `tab` : Tipo di filtro ("perTe" o "followed", default: "perTe")
- `difficulty` : Filtro difficoltà (default: "all")
- `startDate` : Data inizio per filtraggio (opzionale)
- `endDate` : Data fine per filtraggio (opzionale)
- `region` : Filtro per località (default: "all")

- **Risposte:**

- `200 OK` :

```
{
  "success": true,
  "message": "Post caricati con successo",
  "data": {
    "post": [
      {
        "EventID": 123,
        "EventName": "Trekking sulle Dolomiti",
        "EventDescription": "Percorso panoramico...",
        "Location": "Dolomiti",

```

```

    "DateTime": "2025-05-10 09:00:00",
    "Difficulty": "media",
    "EventImages": "[\\"immagine1.jpg\\", \\"immagine2.jpg\\"]",
    "CreatorUsername": "nomeutente",
    "CreatorProfilePicture": "profile.jpg",
    "ParticipantsCount": 15
  }
],
"hasMore": true
}
}

```

3. Dettagli di un Evento

- **Endpoint:** `GET /events?eventId=ID`
- **Descrizione:** Recupera informazioni dettagliate su un evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri Query:**
 - `eventId` : ID dell'evento
- **Risposte:**
 - `200 OK` :

```

{
  "success": true,
  "message": "Informazioni evento recuperate con successo",
  "data": {
    "eventInfo": {
      "EventID": 123,
      "EventName": "Trekking sulle Dolomiti",
      "EventDescription": "Percorso panoramico...",
      "Location": "Dolomiti",
      "DateTime": "2025-05-10 09:00:00",
      "Difficulty": "media",
      "EventImages": "[\\"immagine1.jpg\\", \\"immagine2.jpg\\"]",
      "CreatorID": 456,
      "CreatorUsername": "nomeutente",
      "CreatorProfilePicture": "profile.jpg",
      "subscribed": "participate"
    }
  }
}

```

- `404 Not Found` : Evento non trovato

4. Aggiornamento di un Evento

- **Endpoint:** `PUT /events/update?eventId=ID`
- **Descrizione:** Modifica un evento esistente
- **Autenticazione:** Token di sessione richiesto
- **Parametri:**

```
{
  "EventName": "Nuovo nome evento",
  "EventDescription": "Nuova descrizione",
  "DateTime": "2025-05-15 10:00:00",
  "Difficulty": "difficile"
}
```

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Evento aggiornato con successo"
}
```

- **400 Bad Request** : ID mancante
- **403 Forbidden** : Non sei il creatore dell'evento
- **404 Not Found** : Evento non trovato

5. Eliminazione di un Evento

- **Endpoint:** **DELETE** /events/delete?eventId=ID

- **Descrizione:** Elimina un evento esistente

- **Autenticazione:** Token di sessione richiesto

- **Parametri Query:**

- **eventId** : ID dell'evento

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Evento eliminato con successo"
}
```

- **400 Bad Request** : ID mancante
- **403 Forbidden** : Non sei il creatore dell'evento
- **404 Not Found** : Evento non trovato

Gestione Partecipazioni (**/participations**)

1. Recupero Partecipanti di un Evento

- **Endpoint:** **GET** /participations?eventId=ID

- **Descrizione:** Ottiene la lista dei partecipanti ad un evento

- **Autenticazione:** Token di sessione richiesto

- **Parametri Query:**

- **eventId** : ID dell'evento

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Partecipanti recuperati con successo",
  "data": {
    "Participants": [
      {
        "UserID": 123,
        "Username": "nomeutente",
        "Name": "Nome",
        "Surname": "Cognome",
        "Image": "profile.jpg"
      }
    ]
  }
}
```

- **404 Not Found** : Evento non trovato

2. Aggiunta di una Partecipazione

- **Endpoint:** **POST** /participations
- **Descrizione:** Partecipa ad un evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri:**

```
{
  "eventId": 123
}
```

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Partecipazione aggiunta con successo"
}
```

- **400 Bad Request** : Già iscritto all'evento

3. Rimozione di una Partecipazione

- **Endpoint:** **DELETE** /participations
- **Descrizione:** Rimuove la partecipazione da un evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri:**

```
{
  "eventId": 123
}
```

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Partecipazione rimossa con successo"
}
```

- **400 Bad Request** : Non iscritto all'evento

4. Recupero Partecipazioni Attive

- **Endpoint:** **GET** /participations/active
- **Descrizione:** Ottiene tutti gli eventi a cui l'utente partecipa (non ancora passati)
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Partecipazioni attive recuperate con successo",
  "data": {
    "Events": [
      {
        "EventID": 123,
        "EventName": "Trekking sulle Dolomiti",
        "EventDescription": "Percorso panoramico...",
        "Location": "Dolomiti",
        "DateTime": "2025-05-10 09:00:00",
        "Difficulty": "media",
        "EventImages": "[\\"immagine1.jpg\\", \\"immagine2.jpg\\"]",
        "CreatorUsername": "nomeutente"
      }
    ]
  }
}
```

5. Invito di un Amico

- **Endpoint:** **POST** /participations/invite
- **Descrizione:** Invita un amico a partecipare ad un evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri:**

```
{
  "userId": 456,
  "eventId": 123
}
```

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
```



```
"message": "Invito inviato con successo"
}
```

- **400 Bad Request**: L'utente è già iscritto o già invitato

6. Recupero Inviti ad Eventi

- **Endpoint:** `GET /participations/request`
- **Descrizione:** Ottiene gli inviti ricevuti a partecipare ad eventi
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**
 - **200 OK**:

```
{
  "success": true,
  "message": "Inviti recuperati con successo",
  "data": {
    "Invitations": [
      {
        "ID": 123,
        "Sender": {
          "UserID": 456,
          "Username": "nomeutente"
        },
        "Event": {
          "EventID": 789,
          "EventName": "Trekking in montagna",
          "EventDate": "2025-05-15 10:00:00",
          "EventImage": "[\\"immagine.jpg\\"]"
        }
      }
    ]
  }
}
```

7. Accettazione di un Invito

- **Endpoint:** `PUT /participations/accept`
- **Descrizione:** Accetta un invito a partecipare ad un evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri:**

```
{
  "eventId": 123,
  "participationID": 456
}
```

- **Risposte:**
 - **200 OK**:

```
{
  "success": true,
```

```
"message": "Invito accettato con successo"
}
```

- **400 Bad Request** : Utente già iscritto o invito non valido

8. Rifiuto di un Invito

- **Endpoint:** **DELETE** /participations/reject
- **Descrizione:** Rifiuta un invito a partecipare ad un evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri:**

```
{
  "eventId": 123,
  "participationID": 456
}
```

- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Invito rifiutato con successo"
}
```

- **400 Bad Request** : Invito non valido

9. Stato Amici per un Evento

- **Endpoint:** **GET** /participations/friendList/status?eventId=ID
- **Descrizione:** Ottiene la lista degli amici con il loro stato di partecipazione all'evento
- **Autenticazione:** Token di sessione richiesto
- **Parametri Query:**
 - **eventId** : ID dell'evento
- **Risposte:**

- **200 OK** :

```
{
  "success": true,
  "message": "Lista di amici recuperata con successo",
  "data": {
    "FriendList": [
      {
        "UserID": 123,
        "Username": "nomeutente",
        "Name": "Nome",
        "Surname": "Cognome",
        "Image": "profile.jpg",
        "Status": "participate", // o "pending" o null
        "InvitedBy": 456 // opzionale
      }
    ]
  }
}
```

```
}  
}
```

- **400 Bad Request** : Evento non specificato
- **404 Not Found** : Evento non trovato

Gestione Amicizie (**/friendship**)

1. Invio Richiesta di Amicizia

- **Endpoint:** **POST /friendship/username**
- **Descrizione:** Invia una richiesta di amicizia
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- **200 OK** :

```
{  
  "success": true,  
  "message": "Richiesta di amicizia inviata con successo"  
}
```

- **400 Bad Request** : Richiesta già esistente o già amici

2. Accettazione Richiesta di Amicizia

- **Endpoint:** **PUT /friendship/username**
- **Descrizione:** Accetta una richiesta di amicizia
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- **200 OK** :

```
{  
  "success": true,  
  "message": "Richiesta di amicizia accettata con successo"  
}
```

- **400 Bad Request** : Già amici o richiesta non ricevuta

3. Rifiuto Richiesta di Amicizia

- **Endpoint:** **DELETE /friendship/username/reject**
- **Descrizione:** Rifiuta una richiesta di amicizia
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- **200 OK** :

```
{  
  "success": true,  
  "message": "Richiesta di amicizia rifiutata con successo"  
}
```

- **400 Bad Request** : Richiesta non esistente o non ricevuta

4. Cancellazione Richiesta di Amicizia

- **Endpoint:** `DELETE /friendship/username/cancel`
- **Descrizione:** Cancella una richiesta di amicizia inviata
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- `200 OK` :

```
{
  "success": true,
  "message": "Richiesta di amicizia cancellata con successo"
}
```

- `400 Bad Request` : Richiesta non esistente o non inviata

5. Rimozione Amicizia

- **Endpoint:** `DELETE /friendship/username`
- **Descrizione:** Rimuove un'amicizia esistente
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- `200 OK` :

```
{
  "success": true,
  "message": "Amicizia eliminata con successo"
}
```

- `400 Bad Request` : Amicizia non esistente

6. Recupero Richieste di Amicizia

- **Endpoint:** `GET /friendship/request`
- **Descrizione:** Ottiene tutte le richieste di amicizia ricevute
- **Autenticazione:** Token di sessione richiesto
- **Risposte:**

- `200 OK` :

```
{
  "success": true,
  "message": "Richieste di amicizia recuperate con successo",
  "data": {
    "FriendRequests": [
      {
        "ID": 123,
        "User": {
          "UserID": 456,
          "Username": "nomeutente",
          "Name": "Nome",
          "Surname": "Cognome",
          "ProfilePicture": "profile.jpg"
        }
      }
    ]
  }
}
```

```

    "FriendshipStatus": "pending"
  }
]
}
}

```

- **200 OK** (nessuna richiesta):

```

{
  "success": true,
  "message": "Nessuna richiesta di amicizia in sospeso"
}

```

Gestione Utenti (/user)

1. Ricerca Utenti

- **Endpoint:** `GET /user/search?username=term`
- **Descrizione:** Cerca utenti per nome utente
- **Autenticazione:** Token di sessione richiesto
- **Parametri Query:**
 - **username**: Termine di ricerca
- **Risposte:**

- **200 OK**:

```

{
  "success": true,
  "message": "Utenti trovati",
  "data": {
    "Users": [
      {
        "UserID": 123,
        "Username": "nomeutente",
        "Name": "Nome",
        "Surname": "Cognome",
        "Image": "profile.jpg"
      }
    ]
  }
}

```

- **404 Not Found**: Nessun utente trovato

2. Informazioni Utente

- **Endpoint:** `GET /user/username?tab=posts|partecipations`
- **Descrizione:** Ottiene informazioni sul profilo di un utente
- **Autenticazione:** Token di sessione richiesto
- **Parametri Query:**
 - **tab**: Tipo di contenuto da visualizzare ("posts" o "partecipations", default: "posts")
- **Risposte:**

o 200 OK :

```
{
  "success": true,
  "message": "Informazioni profilo utente recuperate con successo.",
  "data": {
    "UserInfo": {
      "UserID": 123,
      "Username": "nomeutente",
      "Email": "utente@esempio.com",
      "Name": "Nome",
      "Surname": "Cognome",
      "Privacy": "public",
      "ProfilePicture": "profile.jpg",
      "Description": "Descrizione profilo",
      "friendshipStatus": {
        "status": "friend",
        "direction": "received"
      },
    },
    "friendList": [
      {
        "UserID": 456,
        "Username": "altrouser",
        "Name": "Nome",
        "Surname": "Cognome",
        "Image": "profile2.jpg"
      }
    ],
    "userPostCount": 5,
    "posts": [
      {
        "EventID": 789,
        "EventName": "Trekking sulle Dolomiti",
        "EventDescription": "Percorso panoramico...",
        "Location": "Dolomiti",
        "DateTime": "2025-05-10 09:00:00",
        "Difficulty": "media",
        "EventImages": "[\\"immagine1.jpg\\", \\"immagine2.jpg\\"]",
        "ParticipantsCount": 15
      }
    ],
    "partecipations": [
      {
        "EventID": 987,
        "EventName": "Escursione in montagna",
        "EventDescription": "Percorso...",
        "Location": "Montagna",
        "DateTime": "2025-06-15 08:30:00",
        "Difficulty": "facile",
        "EventImages": "[\\"immagine3.jpg\\"]",
        "CreatorUsername": "creatore"
      }
    ]
  }
}
```

```
}  
}
```

- **404 Not Found** : Utente non trovato

3. Aggiornamento Profilo

- **Endpoint:** **POST /user/username/update**
- **Descrizione:** Aggiorna le informazioni del profilo utente
- **Autenticazione:** Token di sessione richiesto
- **Parametri** (Form-Data):
 - **description** : Descrizione profilo (opzionale)
 - **privacy** : Impostazioni privacy ("public" o "private", opzionale)
 - **removeImage** : Flag per rimuovere l'immagine profilo ("true" o "false", opzionale)
 - **newProfileImage** : Nuova immagine profilo (file, opzionale)

- **Risposte:**

- **200 OK** :

```
{  
  "success": true,  
  "message": "Informazioni profilo utente aggiornate con successo.",  
  "data": {  
    "description": "Nuova descrizione",  
    "privacy": "1",  
    "image": "nuovo_profile.jpg",  
    "removeImage": "false"  
  }  
}
```

- **403 Forbidden** : Non hai permesso di aggiornare questo profilo
- **404 Not Found** : Utente non trovato
- **500 Internal Server Error** : Errore durante l'aggiornamento

Gestione errori

Le API restituiscono i seguenti codici di stato HTTP:

- **200 OK** : Richiesta completata con successo
- **400 Bad Request** : Richiesta non valida o parametri mancanti
- **401 Unauthorized** : Autenticazione richiesta o sessione scaduta
- **403 Forbidden** : Accesso non autorizzato alla risorsa
- **404 Not Found** : Risorsa non trovata
- **405 Method Not Allowed** : Metodo HTTP non supportato per l'endpoint
- **409 Conflict** : Conflitto con lo stato attuale della risorsa
- **500 Internal Server Error** : Errore interno del server

Note sull'autenticazione

- Tutte le API (eccetto login e registrazione) richiedono un token di sessione valido

- Il token di sessione deve essere inviato nei cookie
 - Una sessione scaduta restituirà un errore 401 con un messaggio che indica di effettuare nuovamente il login
-