

örslag på nya endpoints5

# API-dokumentation för doSTHLMs REST-API

Simon Olofsson

# Contents

Endpoints

4

## Endpoints

### /login

#### POST

Loggar in en användare. Förväntar sig en HTTP-body med följande utseende:

```
{
  emailAddress: ANVÄNDARENS EMAILADDRESS,
  password: ANVÄNDARENS LÖSENORD
}
```

Returnerar en autentiseringstoken som kan komma åt antingen via en cookie som sätts eller själva response-bodyn, som har formatet:

```
{
  authToken: AUTENTISERINGSTOKEN
}
```

Förutsatt att vi lagrar svaret i en variabel kallad *response* kan alltså denna token komma åt som *response.authToken*.

### /facebook/login

Loggar in en användare via Facebook.

#### Headerfält:

FACEBOOK-AUTH-TOKEN – en valid autentiseringstoken från Facebooks Graph API

#### Returvärden:

Vänlista – en lista innehållandes de av den angivna användarens Facebookvänner som också är registrerade användare av appen.

### /swipingsessions

#### GET

Hämtar alla swiping-sessioner där de angivna användarna deltagit.

#### Parametrar:

- emailAddresses – en array i json-format innehållandes en serie emailadresser. Matchningen kommer att göras på alla emailadresser som skickas, med andra ord kommer bara swiping-sessioner där alla användare kopplade till emailadresserna deltagit att hämtas. Matchningen är också exakt i det att den inte heller kommer att göras på de swiping-sessioner där några andra användare, utöver dessa, deltagit.

**Exempel:** `/swipingsessions?emailAddresses=["user1@example.com","user2@example.com"]`

## POST

Skapar en ny swipingsession med alla angivna användare som deltagare. Returnerar den nya swipingsessionens id samt en lista med aktiviteter.

### Parametrar:

- emailAddresses – en array i json-format innehållandes en serie emailadresser.

### Returvärden:

- swipingSessionId – Den nya swipingsessionens id, används för att unikt identifiera den vid uppdateringar.
- activities – En lista på aktiviteter som användarna kan swipe:a på.

**Exempel:** `/swipingsessions?emailAddresses=["user1@example.com","user2@example.com"]`

## PUT

Uppdaterar en swipingsession med valda aktiviteter för den angivna användaren.

### Parametrar:

- swipingSessionId – id:t för den swipingsession som ska uppdateras. Fås vid skapandet av swipingsessionen eller genom en GET-request till denna endpoint.
- email – den emailadress associerad med den användare som just valt aktiviteter.
- activities – en array i json-format innehållandes namnet på de aktiviteter som valts.

**Exempel:** `/swipingsessions?swipingSessionId=1&email=user1@example.com&activities=["Moderna Museet","Medelhavsmuseet"]`

## Förslag på nya endpoints

### /accommodation

Hämtar hyresobjekt. Främst tänkt för hyresgästen men kan troligtvis användas för hyresvärderna för att hämta det objekt som denne hyr ut.

Parametrar:

- count
- offset
- price – bör stödja olika operatorer som =, <, >, <= och >=
- size – bör stödja olika operatorer som =, <, >, <= och >=
- Olika parametrar baserat på den öppna datan.

Returnerar en lista med hyresobjekt.

### **/tenants**

För hyresvärden. Hämtar de intressenter som swipe:at ja på hyresobjektet.

Parametrar:

- renterId
- count
- offset

Returnerar en lista på potentiella hyresgäster.

### **/matches**

För hyresgästen. Hämta de objekt vars hyresvärd swipe:at ja på hyresgästen i fråga.

Parametrar:

- tentantId
- count
- offset

Returnerar en lista med hyresobjekt, troligtvis bör också information om hyresvärden följa med. Inte nödvändigtvis för att hyresgästen ska kunna se den, men för att frontenden ska kunna koppla ihop hyresgäst och hyresvärd för chatt.