

# Search4

## API Documentation

### USERS

- **Return all** currently active users on Search4:

HTTP Method: GET

URL: {Base API URL}/users

Example: http://localhost:8080/Search4/api/v1/users

Returns: JSON List of users

- **Return one** user from Search4 based on **id** number:

HTTP Method: GET

URL: {Base API URL}/users/{id}

Example: http://localhost:8080/Search4/api/v1/users/1

Returns: JSON user

- Add a **new user** to Search4:

HTTP Method: POST

URL: {Base API URL}/users

Example: http://localhost:8080/Search4/api/v1/users

HEADER INFO: Content-Type = application/json

JSON BODY:

```
{
  "firstName" : "John",
  "lastName" : "Doe",
  "email" : "johndoe@alive.com",
  "password" : "password4Me"
}
```

Limitations:

<b>firstName</b>	must be less than or equal to 255 characters
<b>lastName</b>	<b>not required</b>
<b>email</b>	must be a new (unique) email adress and less than or equal to 255 characters
<b>password</b>	must be 4 or more characters

Returns: JSON user

- **Edit/update** an existing **user** registered to Search4:

this API can only update a user's first name, last name and/or email address and requires valid login information (old email and correct password)

HTTP Method: PUT

URL: {Base API URL}/users/{id}

Example: http://localhost:8080/Search4/api/v1/users/1

HEADER INFO: Content-Type = application/json

JSON BODY:

```
{
  "email" : "johndoe@alive.com",
  "password" : "password4Me",
  "firstName" : "Johnny",
  "lastName" : "Doeson",
  "updatedEmail" : "johnnydoeson@gmail.com"
}
```

Limitations:

<b>email</b>	must be the old email address
<b>password</b>	must be the correct password
<b>firstName</b>	<b>not required</b> - must be less than or equal to 255 characters.
<b>lastName</b>	<b>not required</b>
<b>updatedEmail</b>	<b>not required</b> - must be must be a new (unique) email adress and less than or equal to 255 characters

Returns: Transaction Information

- **Delete one** user from Search4 based on **id** number:

HTTP Method: DELETE

URL: {Base API URL}/users/{id}

Example: http://localhost:8080/Search4/api/v1/users/1

Returns: Transaction Information

- **Return all providers** saved for one user from Search4 based on **user id** number:

HTTP Method: GET

URL: {Base API URL}/users/{id}/providers

Example: http://localhost:8080/Search4/api/v1/users/1/providers

Returns: JSON List of providers

- **Return all subscriptions** saved for one user from Search4 based on **user id** number:

HTTP Method: GET

URL: {Base API URL}/users/{id}/subscriptions

Example: http://localhost:8080/Search4/api/v1/users/1/subscriptions

Returns: JSON List of subscriptions

## Movies

- **Return one** movie from Search4 based on **id** number:

HTTP Method: GET

URL: {Base API URL}/movies/{id}

Example: http://localhost:8080/Search4/api/v1/movies/1

Returns: JSON movie

- **Return all** movies on Search4:

HTTP Method: GET

URL: {Base API URL}/getallmovies

Example: http://localhost:8080/Search4/api/v1/getallmovies

Returns: JSON List of movies

- **Return** list of movies from Search4 based on search **value** in an **ascending title** order

HTTP Method: GET

URL: {Base API URL}/movies/title/asc/{value}

Example: http://localhost:8080/Search4/api/v1/movies/title/asc/cowboys

Returns: JSON list of movies

- **Return** list of movies from Search4 based on search **value** in a **descending title** order

HTTP Method: GET

URL: {Base API URL}/movies/title/desc/{value}

Example: http://localhost:8080/Search4/api/v1/movies/title/desc/cowboys

Returns: JSON list of movies

- **Return** list of movies from Search4 based on search **value** in an **ascending date** order

HTTP Method: GET

URL: {Base API URL}/movies/date/asc/{value}

Example: http://localhost:8080/Search4/api/v1/movies/date/asc/cowboys

Returns: JSON list of movies

- **Return** list of movies from Search4 based on search **value** in a **descending date** order

HTTP Method: GET

URL: {Base API URL}/movies/date/desc/{value}

Example: http://localhost:8080/Search4/api/v1/movies/date/desc/cowboys

Returns: JSON list of movies

## **SUBSCRIPTIONS**

- **Return a Subscription** using Subscription ID

HTTP Method: GET

URL: {Base API URL}/subscriptions/{id}

Example: http://localhost:8080/Search4/api/v1/subscriptions/3

Returns: JSON Subscription

- **Return all Subscriptions** for a User using user ID

HTTP Method: GET

URL: {Base API URL}/subscriptions/user/{id}

Example: http://localhost:8080/Search4/api/v1/subscriptions/user/1

Returns: JSON List of Subscriptions

- **Return all Subscriptions** related to a Movie using movie ID

HTTP Method: GET

URL: {Base API URL}/subscriptions/movie/{id}

Example: http://localhost:8080/Search4/api/v1/subscriptions/movie/

Returns: JSON List of Users

- **Return all Subscriptions**

HTTP Method: GET

URL: {Base API URL}/subscriptions/

Example: http://localhost:8080/Search4/api/v1/subscriptions/

Returns: JSON List of all Subscriptions

- **Add a Subscription**

HTTP Method: POST

URL: {Base API URL}/subscriptions/

Example: http://localhost:8080/Search4/api/v1/subscriptions

HEADER INFO: Content-Type = application/json

JSON BODY: {  
    "movieId" : "453",  
    "userId" : "12"  
}

Returns: JSON Subscription

- **Delete a Subscription**

HTTP Method: DELETE

URL: {Base API URL}/subscriptions/{id}

Example: http://localhost:8080/Search4/api/v1/subscriptions/1

Returns: Transaction Information

## Providers

- **Return all** providers from Search4:

HTTP Method: GET

URL: {Base API URL}/providers

Example: <http://localhost:8080/Search4/api/v1/providers>

Returns: JSON list of providers

- **Return one** provider by provider id from Search4:

HTTP Method: GET

URL: {Base API URL}/providers/providerid/{id}

Example: <http://localhost:8080/Search4/api/v1/providers/providerid/1>

Returns: JSON provider

- **Return all** providers for one user by user id from Search4:

HTTP Method: GET

URL: {Base API URL}/providers/userid/{id}

Example: <http://localhost:8080/Search4/api/v1/providers/userid/1>

Returns: JSON list of providers

- **Add a new provider** with userid to Search4:

HTTP Method: POST

URL: {Base API URL}/providers/{id}

Example: <http://localhost:8080/Search4/api/v1/providers/1>

HEADER INFO: Content-Type = application/json

JSON BODY:

```
{
  "name": "providerName"
}
```

Returns: JSON provider

- **Delete a provider** with provider id from Search4:

HTTP Method: DELETE

URL: {Base API URL}/providers/{id}

Example: <http://localhost:8080/Search4/api/v1/providers/1>

HEADER INFO: Content-Type = application/json

JSON BODY:

```
{  
  "name": "providerName"  
}
```

Returns: Transaction information