# Catchr API Endpoints

**Note: All endpoints are prefixed with “http://localhost:4000**

Table 1: "Users" API Endpoints

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **GET** | /api/users/:user\_id |  | username | Gets a list of all users | No | No | None Required |
|  | /api/users |  | email | Finds a user by user ID |  |  | None Required |
|  |  |  | sortBy |  |  |  |  |
|  |  |  | sortOrder |  |  |  |  |
| **POST** | /api/users/register | username |  | Creates a new user account | No | No | None Required |
|  |  | email |  |  |  |  |  |
|  |  | password |  | Posts a login attempt, returns an encoded JSON web token with user ID and admin status encoded if successful |  |  |  |
|  | /api/users/login | username OR email |  | No | No | None Required |
|  |  | password |  | Updates a users personal details. Most provide a correct password to update any fields.  Updates a users personal details. Most provide a correct password to update any fields. |  |  |  |
| **PUT** | /api/users/ | new\_username |  | No | No | User's own information or admin |
|  |  | new\_email |  |  |  |  |
|  |  | admin |  |  |  |  |
|  |  | password |  |  |  |  |
|  |  | user\_id |  | Updates a user's password. Must confirm old password is correct to change to new password.  Updates a user's password. Must confirm old password is correct to change to new password. |  |  |  |
|  | /api/users/password | oldpassword |  | No | No | User's own password or admin |
|  |  | newpassword |  |  |  |  |
|  |  | user\_id |  | Deletes a user account |  |  |  |

Table 2: "Cards" API Endpoints

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **GET** | /api/cards/ |  | card\_name, | Gets all cards with basic details, with optional filtering / sorting | Yes | No | None Required |
|  |  |  | set\_id, |  |  |  |  |
|  |  |  | category\_id, |  |  |  |  |
|  |  |  | type\_id, |  |  |  |  |
|  |  |  | sortBy, |  |  |  |  |
|  |  |  | page |  |  |  |  |
|  |  |  | limit |  |  |  |  |
|  | /api/cards/:card\_id |  |  | Gets a single card by ID | No | No | None Required |

Table 3: "Collections" API endpoints

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **GET** | /api/collections |  | sortBy | Gets a list of all collections on the site with username | No | No | None Required |
|  |  |  | sortOrder |  |  |  |
|  |  |  | page |  |  |  |
|  |  |  | limit |  |  |  |
|  | /api/collections/user/:user\_id |  |  | Gets a single collection by user ID with cards and stats | No | No | None Required |
|  | /api/collections/:collection\_id |  |  | Gets a single collection by collection ID with cards and stats | No | No | None Required |
| **POST** | /api/collections/card | card\_id |  | Adds a card to a collection | No | No | User's own collection or admin |
|  |  | collection\_id |  |  |  |  |
| **DELETE** | /api/collections/card | card\_id |  | Removes a card from a collection | No | No | User's own collection or admin |
|  |  | collection\_id |  |  |  |  |
|  | /api/collections/:collection\_id |  |  | Deletes a collection | No | No | User's own collection or admin |

Table 4: "Comments" API Endpoints

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **GET** | /api/comments/collection/:collection:id |  |  | Gets all comments on a single collection | No | No | None Required |
|  | /api/comments/user/:user\_id |  |  | Gets all comments from a single user | No | No | None Required |
|  | /api/comments |  |  | Gets all comments on the website | No | No | None Required |
| **POST** | /api/comments | collection\_id |  | Posts a new comment on a collection | No | No | User only |
|  |  | comment\_body |  |  |  |  |  |
|  |  | user\_id |  |  |  |  |  |
| **DELETE** | /api/comment/:comment\_id |  |  | Deletes a comment | No | No | User's own comment or admin |

Table 5: "Likes" API Endpoints

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **GET** | /api/likes/singlelike |  | user\_id | Gets a single instance of a like on a collection | No | No | None required |
|  |  |  | collection\_id |  |  |  |  |
|  | /api/likes/ |  |  | Gets all likes on the website | Yes | No | None required |
|  | /api/likes/user/:user\_id |  |  | Gets all likes from a single user | No | No | None required |
| **POST** | /api/likes | collection\_id |  | Posts a new like on a collection | No | No | User only |
|  |  | user\_id |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **DELETE** | '/api/likes |  | collection\_id | Removes a like from a collection | No | No | None Required |
|  |  |  | user\_id |  |  |  |  |

Table 6: "Others" API Endpoints

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **HTTP Verb** | **Endpoint + route params** | **Body** | **Query Params** | **Description** | **Cacheable** | **Paginated** | **Permissions** |
| **GET** | /api/others/sets |  |  | gets all card sets (expansions) | Yes | No | None Required |
|  | /api/others/rarities |  |  | gets all card "rarity" options with an image | Yes | No | None Required |
|  | /api/others/types |  |  | Gets all card "types" | Yes | No | None Required |
| **POST** | /api/others/rarities | rarity\_icon |  |  | No | No | None Required |
|  |  | rarity\_description |  |  |  |  |  |