

API Routes

api/user/auth

Para saber si un usuario está autenticado. GET

Request

```
{}
```

Response

- Si está autenticado devuelve:

```
{
  data: Iauth
}
```
- De lo contrario:

```
{
  error: "not auth"
}
```

api/user/register

Registra un usuario en la base de datos. POST

Request

```
{
  username: string;
  name: string;
  email: string;
  password: string;
  avatar_url: string;
}
```

api/user/login

Verifica las credenciales del usuario en la base de datos. POST

Request

```
{
  email: string; //Note: can be an email or an username
  password: string;
}
```

Response

- Con éxito

```
{
  info: "success validation.",
  user: Iauth
  success: "whatever"
}
```
- Mal nombre de usuario o contraseña

```
{
  info: "password or username are not valid.",
```

```

        failure: "whatever"
    }
    • De lo contrario:
    {
        error: "internal error."
    }

```

api/formulas/add

Agrega una fórmula creada por el usuario. **POST**

Request

```

{
    formula: {
        title: string;
        latexCode: string;
        tags: string[];
        category: string | null;
        images: Iimage[];
    }
}

```

api/formulas/search

Retorna todas las formulas si no se le envia nada, funcionará como filtro.

POST

Request

```

{
    data: string; // can be formula title, id, added date, tag name,
category name, o id.
}

```

Nota: El usuario está implícito.

Response

- Con éxito

```

{
    formulas: Iformulas[]
}

```
- Fracaso

```

{
    error: ""
}

```

Si no se encuentra ninguna fórmula, se envía [].

api/formulas/edit

Edita la formula en la base de datos. **POST**

Request

```
{
    formula: Iformula //solo se llama si hay al menos un cambio en algun
    atributo, y envia todos los atributos.
}
```

Response

- Con éxito

```
{
    Info: "some success message"
}
```
- Fracaso

```
{
    error: "some error msg"
}
```

api/formulas/delete/

Elimina una formula. POST

Request

```
{
    id_formula: string; // formula id
}
```

Response

- Con éxito

```
{
    Info: "some info message"
}
```
- Fracaso

```
{
    error: "some error message"
}
```

api/user/search/

Retorna información relevante para construir la página del perfil del usuario. POST

Request

```
{
    username: string;
}
```

Response

- Con éxito

```
{
    data: {
        user: Iauth
        formulas: IprofileFormulas[]
    }
}
```

- Fracaso


```
{
  Error: "some message error"
}
```

api/tags

Retorna todas las tags disponibles para el usuario. GET

Response

- Con éxito


```
{
  tags: string[]
}
```
- Fracaso


```
{
  Error: "some message error"
}
```

api/categories

Retorna todas las categorias disponibles para el usuario. GET

Response

- Con éxito


```
{
  categories: string[]
}
```
- Fracaso


```
{
  Error: "some message error"
}
```

api/tags/add

Crea una etiqueta a nivel de bases de datos para el usuario. POST

daRequest

```
{
  tag: string; // nombre de la etiqueta
}
```

Response

- Con éxito


```
{
  success: "some success msg"
}
```
- Fracaso


```
{
  error: "some error msg"
}
```

Types

```
interface IUser {
  idUser: number;
  username: string;
  name: string;
  email: string;
  avatarUrl: string;
  registeredAt: string;
}

IprofileFormulas {
  totalFormulas: number;
  mainFormulas: Iformula[];
}
```

```
interface Iauth {
  username: string;
  avatarUrl: string;
  email: string;
}
```

```
interface Iformula {
  idFormula: number;
  addedAt: string;
  tags: string[];
  category: string;
  title: string;
  latexCode: string;
  images: Iimage[];
  isDeleted: boolean;
}
```

```
interface Iimage {
  idImage: number;
  date: string;
  url: string;
  title: string;
}
```