

Pràctica 3. Comprovació

Gela Rafael
02/04/2025
DAW2
M6

Gestió d'usuaris

Registre de l'usuari

1. Des de Postman fes la petició per registrar un usuari.

The screenshot displays the Postman interface for a REST client. At the top, a request is defined as a POST to `10.0.70.40:8080/users`. Below this, the specific endpoint `10.0.70.40:8080/users/register` is selected. The request method is set to **POST**. The body is configured as **raw** JSON. The JSON payload is as follows:

```
1 {
2   "name" : "Rafa",
3   "surname": "Grdz",
4   "email": "prova.grdz@gmail.com",
5   "password": "1234"
6 }
```

The bottom section of the interface shows the response. The status is **201 Created**. The response body is also in JSON format, containing a success message:

```
1 {
2   "message": "Usuari registrat correctament"
3 }
```

MongoDB Compass - react/userdb.users

Connections Edit View Collection Help

Compass

My Queries

CONNECTIONS (1)

Search connections

- react
 - admin
 - config
 - local
 - userdb
 - notes
 - users

react > userdb > users

Documents 3 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

ADD DATA EXPORT DATA UPDATE DELETE

```

_id: ObjectId('67bf6715bd4ec06693a02854')
__v: 0

```

```

_id: ObjectId('67e43c53a63854bb35eff0d2')
name: "rafall11"
surname: "rafall11"
email: "rafall1@gmail.com"
password: "$2b$10$mAVSQBB/sKyTjoPsBzc65.q83NWIwWezWtkT8gtzd6YvEcmjicVkc"
createDate: 2025-03-26T17:41:39.925+00:00
updateDate: 2025-03-26T17:41:39.925+00:00
__v: 0

```

```

_id: ObjectId('67ed54964580f9982120d49a')
name: "Rafa"
surname: "Grdz"
email: "prova.grdz@gmail.com"
password: "$2b$10$QuPN3sDXTab8G0/8ATbkZuHM4D6h0KEZ/ENUHvIfZsbqAr0T/5hui"
createDate: 2025-04-02T15:15:34.028+00:00
updateDate: 2025-04-02T15:15:34.028+00:00
__v: 0

```

2. Gestió d'errors:

Fes una petició amb tots els camps d'usuari que no compleixin les normes de validació:

POST 10.0.70.40:8080/users, +

HTTP 10.0.70.40:8080/users/register

POST 10.0.70.40:8080/users/register

Params Authorization Headers (9) Body ● Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▾

```
1 {
2   "name" : "",
3   "surname": "",
4   "email": "prova1.grdz@gmail",
5   "password": ""
6 }
```


Body Cookies (2) Headers (10) Test Results ↻

{ } JSON ▾ ▶ Preview 🔍 Visualize ▾

```
4   "message": "Name is required",
5   "properties": {
6     "message": "Name is required",
7     "type": "required",
8     "path": "name",
9     "value": ""
10  },
11   "kind": "required",
12   "path": "name",
13   "value": ""
14 },
15 "surname": {
16   "name": "ValidatorError",
17   "message": "Surname is required",
18   "properties": {
19     "message": "Surname is required",
20     "type": "required",
21     "path": "surname",
22     "value": ""
23   },
24   "kind": "required",
25   "path": "surname",
26   "value": ""
27 },
28 "password": {
29   "name": "ValidatorError",
30   "message": "Password is required",
31   "properties": {
```

Login d'usuari

1. Login correcte


10.0.70.40:8080/users/login

POST
10.0.70.40:8080/users/login

Params
Authorization
Headers (9)
Body
Scripts
Settings

☐ none
 ☐ form-data
 ☐ x-www-form-urlencoded
 ☒ raw
 ☐ binary
 ☐ GraphQL
 JSON

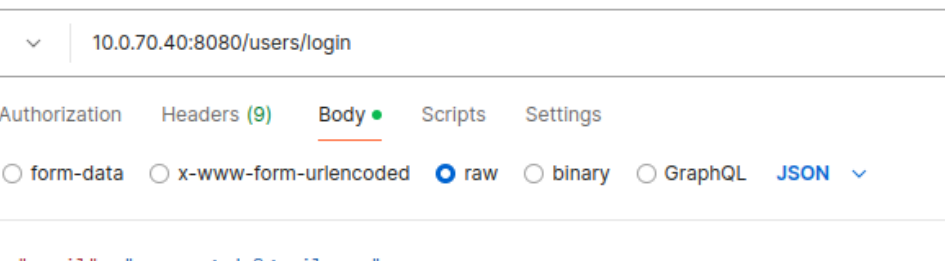
```

1  {
2    "email": "prova.grdz@gmail.com",
3    "password": "1234"
4  }

```

Body
Cookies (1)
Headers (11)
Test Results

| Name | Value | Domain | Path | Expires | HttpOnly |
|------|------------------------------------|------------|------|-------------------------------|----------|
| jwt | eyJhbGciOiJIUzI1NiIsInR5cCI6Ikp... | 10.0.70.40 | / | Mon, 28 Jul 2025 07:21:36 GMT | true |



The screenshot displays a web browser window with the address bar showing `10.0.70.40:8080/users/login`. The browser's developer tools are open, showing the **Network** tab. A single request is listed, which is a **POST** to `10.0.70.40:8080/users/login`. The request's **Body** tab is selected, showing a JSON payload: `{ "email": "prova.grdz@gmail.com", "password": "1234" }`. The response is a **200 OK** with a **Content-Type** of `application/json`. The response's **Body** tab is selected, showing a JSON payload: `{ "message": "Usuari autenticat" }`. The browser's status bar at the bottom shows `200 OK`.

HTTP 10.0.70.40:8080/users/login

POST 10.0.70.40:8080/users/login

Params Authorization Headers (9) Body Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "email": "prova.grdz@gmail.com",
3   "password": "1234"
4 }
```

Body Cookies (1) Headers (11) Test Results

{ } JSON Preview Visualize

```
1 {
2   "message": "Usuari autenticat"
3 }
```

2. Error de login

The screenshot shows a REST client interface with the following details:

- URL:** 10.0.70.40:8080/users/login
- Method:** POST
- Body Type:** raw (selected)
- Request Body (JSON):**

```
1  {  
2    "email": "prova.grdz@gmail.com",  
3    "password": "123"  
4  }
```
- Response Body (JSON):**

```
1  {  
2    "error": "Contrasenya incorrecta"  
3  }
```

The response body indicates an incorrect password error.

POST 10.0.70.40:8080/users, +



10.0.70.40:8080/users/login

POST



10.0.70.40:8080/users/login

Params Authorization Headers (8) **Body** Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** ▾

```
1 {  
2   "email": "prova.grdz@gmail.com",  
3   "password": "123"  
4 }
```

Body **Cookies** Headers (10) Test Results ↻




No cookies received from the server

All your cookies and their associated domains will appear here.

Gestió de notes

1. Crear nota


 New Collection / 10.0.70.40:8080/notes/create



POST ▾ | 10.0.70.40:8080/notes/create

Params Authorization Headers (9) **Body ●** Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON ▾**

```
1  {
2    "title" : "Nota1",
3    "text" : "prueba nota 1",
4    "status" : "Archived",
5    "author" : "67ed54964580f9982120d49a"
6  }
```


Body Cookies (1) Headers (10) Test Results 

{ } JSON ▾  Preview  Visualize ▾

```
1  {
2    "message": "Nota creada correctament"
3  }
```


2. Error quan es crea la nota

POST 10.0.70.40:8080/notes • POST 10.0.70.40:8080/notes • +

 New Collection / 10.0.70.40:8080/notes/create


POST ▾ 10.0.70.40:8080/notes/create

Params Authorization Headers (9) **Body ●** Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON ▾**

```
1 {
2   "title" : "",
3   "text" : "prueba nota 1",
4   "status" : "Archied",
5   "author" : "67ed54964580f9982120d49a"
6 }
```

Body Cookies (1) Headers (10) Test Results | ↺

{ } JSON ▾ ▶ Preview  Visualize ▾

```
1 {
2   "title": {
3     "name": "ValidatorError",
4     "message": "Title is required",
5     "properties": {
6       "message": "Title is required",
7       "type": "required",
8       "path": "title",
9       "value": ""
10    },
11    "kind": "required",
12    "path": "title",
13    "value": ""
14  },
15  "status": {
16    "name": "ValidatorError",
17    "message": "Only Published, Draft or Archived permitted.",
18    "properties": {
19      "message": "Only Published, Draft or Archived permitted.",
20      "type": "enum",
21      "enumValues": [
22        "Published",
23        "Draft",
24        "Archived"
25      ],
26      "path": "status",
27      "value": "Archied"
```

3. Actualitzar nota

Ho tenia malament i ho he arreglat de la següent manera:

Al fer `Note.findById...` etc, en comptes de "text" posaba "body".

```
const updateNote = async (req, res) => {
  try {
    const { id, title, text, status } = req.body;

    if (!id) {
      return res.status(404).json({ error: "ID de nota no afegida." });
    }

    const updateNote = await Note.findByIdAndUpdate(
      id,
      { title, text, status }
    );

    if (!updateNote) {
      return res.status(404).json({ error: "Nota no trobada." });
    }

    res.status(200).json({
      message: "Nota actualitzada correctament.",
    });
  } catch (error) {
    res.status(400).json({ error: "error al actualitzar" });
  }
}
```

POST 10.0.70.40:8080/notes • PUT 10.0.70.40:8080/notes/c +

HTTP New Collection / 10.0.70.40:8080/notes/create

PUT 10.0.70.40:8080/notes/update

Params Authorization Headers (9) Body • Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL

```
1 {
2   "title" : "Nota1",
3   "text" : "prueba actualizar nota 1",
4   "status" : "Archived",
5   "id" : "67ed578986be83f9f18894e3"
6 }
```

Body Cookies (1) Headers (10) Test Results | ↻

{ } JSON ▾ ▶ Preview 🔗 Visualize ▾

```
1 {
2   "message": "Nota actualitzada correctament."
3 }
```

CONNECTIONS (1) ✕ + ...

Search connections

- ▼ react
 - ▶ admin
 - ▶ config
 - ▶ local
 - ▼ userdb
 - notes ...
 - users

Documents 1 Aggregations Schema Indexes 1 Validation

🕒 ▼ Type a query: { field: 'value' } or [Generate query](#) ⚡

[+ ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#)

```
_id: ObjectId('67ed578986be83f9f18894e3')
title: "Nota1"
text: "prueba actualizar nota 1"
status: "Archived"
author: ObjectId('67ed54964580f9982120d49a')
createDate: 2025-04-02T15:28:09.337+00:00
updateDate: 2025-04-02T15:28:09.337+00:00
__v: 0
```

4. Error quan s'actualitza la nota

POST 10.0.70.40:8080/notes. ● PUT 10.0.70.40:8080/notes/. ● +

 New Collection / 10.0.70.40:8080/notes/create


PUT ▾ 10.0.70.40:8080/notes/update

Params Authorization Headers (9) Body ● Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ G

```
1  {
2    "title" : "Nota1",
3    "text" : "prueba actualitzar nota 1",
4    "status" : "ffff",
5    "id" : "67ed578986be83f9f18894e3111"
6  }
```

Body Cookies (1) Headers (10) Test Results | ⌚

{ } JSON ▾ ▶ Preview  Visualize ▾

```
1  {
2    "error": "error al actualitzar"
3  }
```

5. Actualitzar nota no autoritzat per no existir la cookie

POST 10.0.70.40:8080/notes • PUT 10.0.70.40:8080/notes/ • +

 New Collection / 10.0.70.40:8080/notes/create


PUT 10.0.70.40:8080/notes/update

Params Authorization Headers (8) Body • Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▾

```
1  {
2    "title" : "Nota1",
3    "text" : "prueba actualizar nota 1",
4    "status" : "ffff",
5    "id" : "67ed578986be83f9f18894e3111"
6  }
```

Body Cookies Headers (10) Test Results | ↺

{ } JSON ▾ ▶ Preview  Visualize ▾

```
1  {
2    "error": {
3      "name": "JsonWebTokenError",
4      "message": "jwt must be provided"
5    }
6  }
```

POST 10.0.70.40:8080/notes • PUT 10.0.70.40:8080/notes/ • +

 New Collection / 10.0.70.40:8080/notes/create

PUT 10.0.70.40:8080/notes/update

Params Authorization Headers (8) **Body** • Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON ▾

```
1 {  
2   "title" : "Nota1",  
3   "text" : "prueba actualitzar nota 1",  
4   "status" : "ffff",  
5   "id" : "67ed578986be83f9f18894e3111"  
6 }
```

Body **Cookies** Headers (10) Test Results | ⌚



No cookies received from the server

All your cookies and their associated domains will appear here.

