

Projet Node Js

Struys Wivin

Package utilisé :

```
"bcryptjs": "^2.4.3",
"body-parser": "^1.20.0",
"dotenv": "^16.0.1",
"express": "^4.18.1",
"express-fileupload": "^1.4.0",
"express-validator": "^6.14.2",
"jsonwebtoken": "^8.5.1",
"nodemon": "^2.0.19",
"pg": "^8.7.3",
"pg-hstore": "^2.3.4",
"sequelize": "^6.21.3",
"swagger-jsdoc": "^6.2.1",
"swagger-ui-express": "^4.5.0"
```

Authentication :

Route register → Enregistrement

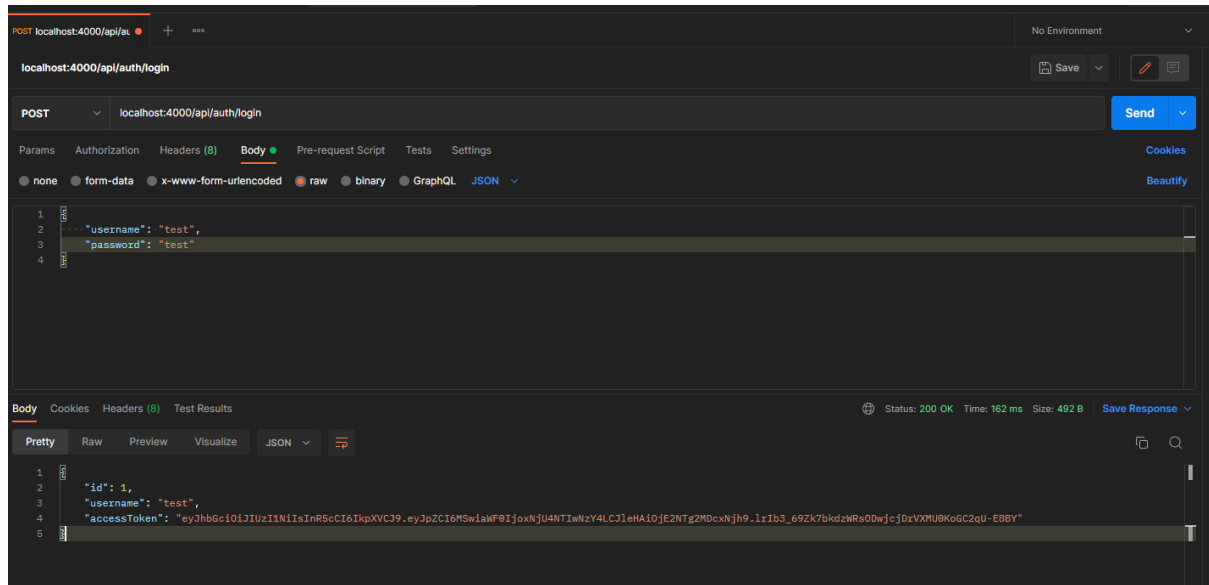
The screenshot displays a REST client interface with the following details:

- Method:** POST
- URL:** localhost:4000/api/auth/register
- Body (JSON):**

```
{  "firstname": "test",  "lastname": "test",  "birthdate": "19/02/1993",  "username": "test",  "email": "test@gmail.com",  "password": "test"}
```
- Status:** 200 OK
- Time:** 103 ms
- Size:** 350 B
- Response (JSON):**

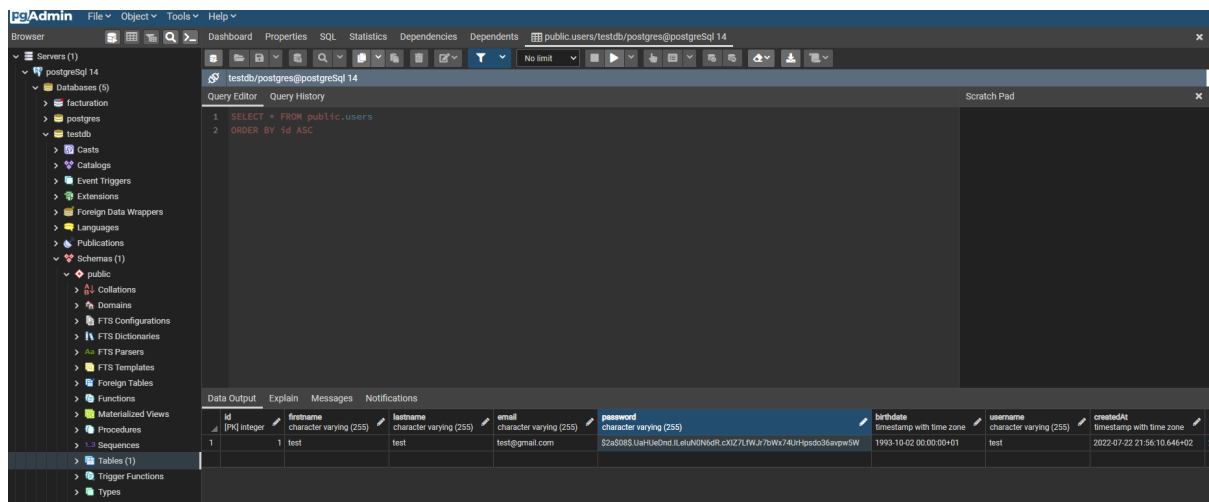
```
{  "message": "User created successfully"}
```

Route Login → Connexion

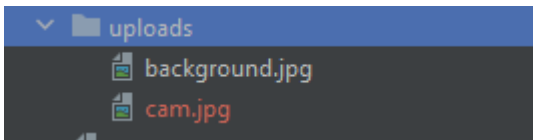
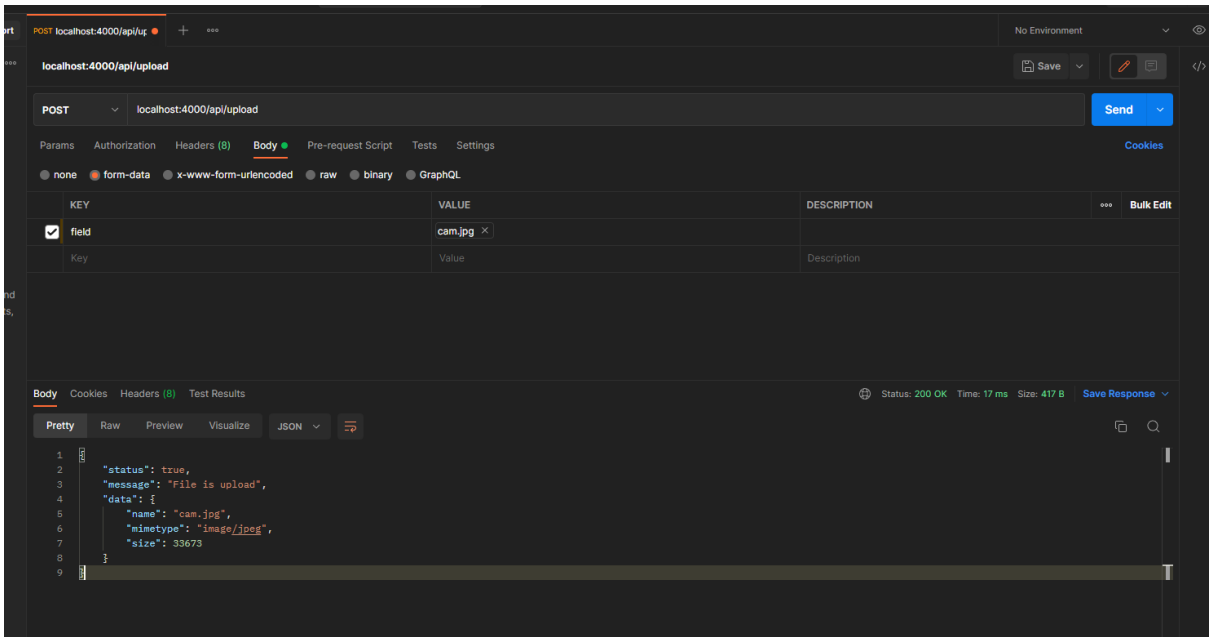


Connexion Base de donnée:

Enregistrement dans la base de donnée

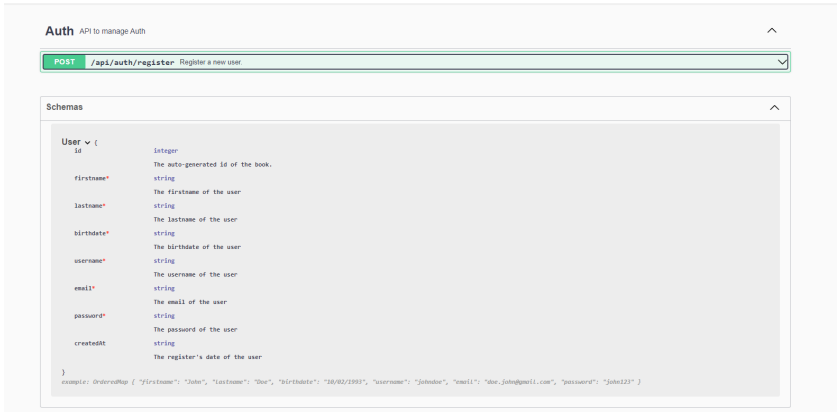


Upload de fichier :



Swagger :

Documentation → Schema



Documentation → Exemple Route register

POST

/api/auth/register

Register a new user

Parameters

Try it out

No parameters

Request body

required

application/json

Example Value

Schema

```
{
  "firstname": "John",
  "lastname": "Doe",
  "birthdate": "18/02/1993",
  "username": "johndoe",
  "email": "doe.johnd@gmail.com",
  "password": "john123"
}
```

Responses

Code	Description	Links
201	Register a new User	No links

Media type

application/json

Controls accept header

Example Value

Schema

```
{
  "firstname": "John",
  "lastname": "Doe",
  "birthdate": "18/02/1993",
  "username": "johndoe",
  "email": "doe.johnd@gmail.com",
  "password": "john123"
}
```