

DEPARTEMENT MATHÉMATIQUES ET INFORMATIQUE

Filière Ingénieur :

« Ingénierie Informatique : Big Data et Cloud Computing »
II-BDCC I

Rapport du Projet de Fin de Module



Application de Blogging basée sur Node.js et
Express : une plateforme web pour les blogs



Réaliser par :

Nouhayla MOUAKKAL

Encadré par :

Pr Abdelaziz DAAIF

Sommaire :

Introduction :	3
Présentation du projet :	4
Cahier de Charge :	5
1-Contexte générale du projet :	5
2- Contraint fonctionnelles :	5
3-Relations entre les entités :	6
Diagramme de Classe :	6
Backend :	7
Article :	10
Catégorie :	11
Commentaire.....	12
Utilisateur	13
Authentification.....	14
Frontend :	16
Conclusion :	25

Introduction :

Avec la montée en puissance du développement web et l'expansion continue des besoins, Node.js s'est établi comme un pilier incontestable pour la création d'applications cross-plateformes. Que ce soit pour le développement d'applications mobiles avec des Framework tels que Ionic ou la conception d'applications pour ordinateurs portables grâce à Electron, Node.js offre une polyvalence incomparable. En outre, il est largement utilisé dans le domaine du développement de serveurs web et la mise en place d'API REST. Face à cette demande croissante, les développeurs se tournent vers des Framework tels qu'Express.js, qui leur permettent de construire rapidement des bases solides pour leurs applications tout en favorisant une collaboration harmonieuse et le respect de normes communes. Ainsi, ces Framework offrent une solution efficace pour accélérer le processus de développement, répondre aux exigences croissantes et offrir des résultats de qualité supérieure.

Présentation du projet :

❖ Objectif :

L'objectif de ce projet est de créer une plateforme de blogs permettant aux utilisateurs d'afficher et de gérer des articles liés à différentes catégories. Les blogs seront accessibles au public, offrant ainsi une plateforme de partage d'informations et d'interactions.

❖ Description :

Ce projet vise à fournir une solution complète pour les blogs, offrant des fonctionnalités spécifiques en fonction du rôle de chaque utilisateur. Les utilisateurs peuvent être des administrateurs, des utilisateurs normaux ou des blogueurs.

En tant qu'administrateur, l'utilisateur aura le pouvoir de gérer les autres utilisateurs, de gérer les articles existants ainsi que les catégories. Cette fonctionnalité permet de maintenir l'ordre et de garantir que le contenu du blog reste pertinent et de haute qualité.

Les utilisateurs normaux auront la possibilité de consulter les blogs, de lire les articles et d'ajouter des commentaires. Cela favorise l'engagement de la communauté en permettant aux utilisateurs de donner leur avis, de poser des questions ou de partager des idées.

Les blogueurs, quant à eux, auront des privilèges supplémentaires. Ils pourront créer leurs propres articles et les gérer, mais uniquement pour les articles qu'ils ont créés. Cela permet aux blogueurs de partager leurs connaissances et leurs expériences spécifiques dans leur domaine d'expertise.

En résumé, ce projet vise à offrir une plateforme de blogs complète et interactive, permettant aux utilisateurs d'accéder à une multitude de sujets et de catégories, de partager leurs propres articles et de participer à des discussions enrichissantes au sein de la communauté des blogs.

Cahier de Charge :

1-Contexte générale du projet :

Nous avons l'intention de créer un blog dont les parties backend et frontend seront développées et testées de manière indépendante. Le backend sera conçu sous la forme d'une API JSON. Voici les technologies que nous prévoyons d'utiliser pour ce projet :

Du côté **backend** :

Le Framework Node.js Express pour le serveur web.

- L'ORM Prisma et une base de données MySQL/MariaDB pour la persistance des données.

Du côté **frontend** :

- HTML/CSS/JavaScript.

Vous pouvez utiliser la bibliothèque JavaScript jQuery et le Framework CSS Bootstrap.

2- Contraint fonctionnelles :

L'application Blog doit respecter les contraintes fonctionnelles suivantes, en gérant quatre entités :

1. Utilisateur :

- Nom
- Email
- Mot de passe
- Rôle (ADMIN || AUTHOR)

3. Catégorie :

- Nom

2. Article :

- Titre
- Contenu
- Image
- Date de création
- Date de mise à jour
- Statut de publication sous forme booléenne

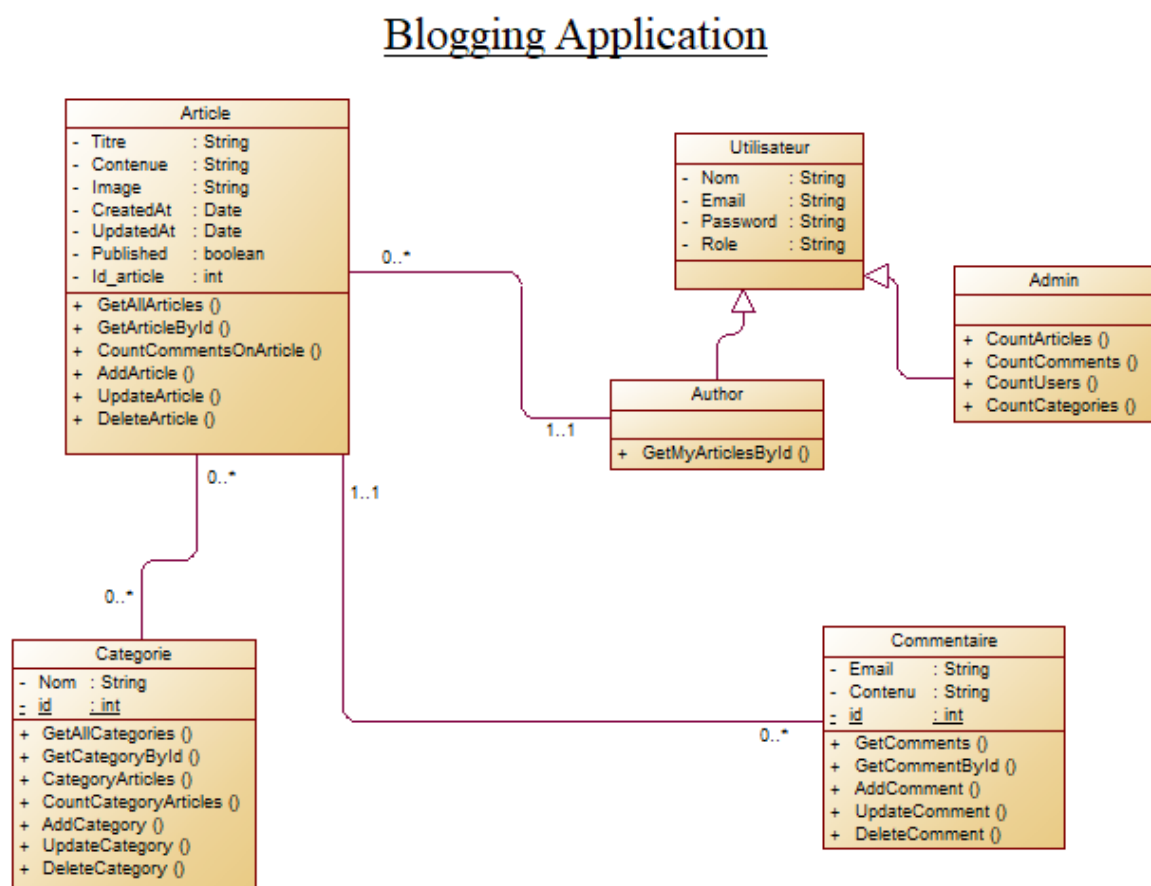
4. Commentaire :

- Email
- Contenu

3-Relations entre les entités :

- Un **article** est associé à **un et un seul utilisateur** (Cet utilisateur devrait avoir le rôle AUTHOR)
- Un **utilisateur** (ayant le rôle AUTHOR) peut écrire **zéro ou plusieurs articles**
- Un **article** est associé à **zéro ou plusieurs catégories**
- Une **catégorie** est associée à **zéro ou plusieurs articles**
- Un **commentaire** est associé à **exactement un article**
- Un **article** est associé à **zéro ou plusieurs commentaires**.

Diagramme de Classe :

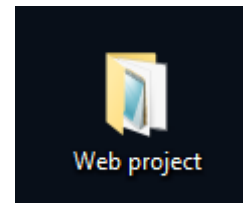


Réalisation du projet :

Backend :

1- J'ai créé le répertoire de notre projet :

```
C:\Users\lenovo g50-80\Desktop
λ mkdir "Web project"
```



2- J'ai saisi les commandes nécessaires pour commencer :

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ express --no-view
```

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ npm i -g nodemon
```

⇒ Pour

installer le répertoire node modules :

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ npm i
```

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ dir node_modules\
Volume in drive C has no label.
Volume Serial Number is AC86-6E11

Directory of C:\Users\lenovo g50-80\Desktop\Web project\node_modules

05/18/2023 12:54 PM <DIR>      .
05/18/2023 12:54 PM <DIR>      ..
05/16/2023 12:32 PM <DIR>      .bin
05/10/2023 04:27 PM <DIR>      .cache
05/18/2023 11:31 AM      42,099 .package-lock.json
05/10/2023 04:36 PM <DIR>      .prisma
05/16/2023 01:06 PM <DIR>      @faker-js
05/10/2023 04:35 PM <DIR>      @prisma
05/09/2023 10:30 PM <DIR>      accepts
05/16/2023 12:32 PM <DIR>      acorn
05/16/2023 12:32 PM <DIR>      acorn-globals
05/16/2023 12:32 PM <DIR>      align-text
05/16/2023 12:32 PM <DIR>      amdefine
05/09/2023 10:30 PM <DIR>      array-flatten
05/16/2023 12:32 PM <DIR>      asap
05/16/2023 12:32 PM <DIR>      basic-auth
05/16/2023 12:32 PM <DIR>      bignumber.js
05/16/2023 12:32 PM <DIR>      body-parser
05/16/2023 12:32 PM <DIR>      bytes
05/16/2023 12:32 PM <DIR>      camelcase
05/16/2023 12:32 PM <DIR>      center-align
05/16/2023 12:32 PM <DIR>      character-parser
05/16/2023 12:32 PM <DIR>      clean-css
```

→Création des trois fichiers

```
C:\Users\lenovo g50-80\Desktop\Web project\routes (master)
λ for %i in (articles.js commentaires.js categories.js) do echo. > %i

echo. 1>articles.js

echo. 1>commentaires.js

echo. 1>categories.js
```

3- Prisma ORM :

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ npm i -D prisma
```

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ npm i @prisma/client
```

```
C:\Users\lenovo g50-80\Desktop\Web project (master)
λ npx prisma init
```

4- Env

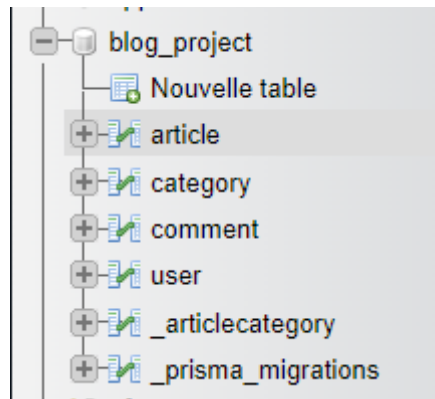
```
DATABASE_URL="mysql://root:@localhost:3306/blog_project"
```

5- L'ajout des modèles reflétant les entités et les relations entre elles et la migration :

```
1 generator client {
2   provider = "prisma-client-js"
3 }
4
5 datasource db {
6   provider = "mysql"
7   url      = env("DATABASE_URL")
8 }
9
10 > model User { ...
18   }
19
20 > enum Role { ...
23   }
24
25 > model Article { ...
37   }
38
39 > model Category { ...
43   }
44
45 > model Comment { ...
53   }
54
```

6- On exécute la commande : **npx prisma migrate dev --name init**

⇒ Donc notre base de données a été créer :



7- Dans le fichier “**web project\routes\seeds\seed.js**” on exécute un code utilisant la bibliothèque Faker qui va nous permettre de créer :

- 10 utilisateurs ayant le rôle “AUTHOR”
- 1 utilisateur ayant le rôle “ADMIN”
- 10 catégories
- 100 articles appartenant à (de 1 à 4 catégories aléatoires) et écrit par l’un des 10 utilisateurs (AUTHOR)
- Pour chaque article, créer de 0 à 20 commentaires)

Voilà un exemple de l’exécution :

		id	title	contenu	image	createdAt	updatedAt	published	userId
<input type="checkbox"/>	Editor	76	Expedita nisi architecto laudantium dolore itaque.	Velit amet voluptates. Eligendi assumenda recusand...	https://loremflickr.com/640/480?lock=7503189343469...	2021-06-10 07:24:55.780	2023-05-18 12:05:55.935	1	56
<input type="checkbox"/>	Editor	77	Eius dolorum a asperiores temporibus saepe blandit...	Eius tempora commodi repellat sapiente eaque. Repu...	https://picsum.photos/seed/ogXIT54L/640/480	2022-11-24 03:01:28.300	2023-05-18 12:05:55.901	1	56
<input type="checkbox"/>	Editor	78	Corrupti autem blanditis asperiores adipisci maxi...	Dignissimos provident dolores vitae impedit in. Of...	https://picsum.photos/seed/UKaXgLIh/640/480	2021-12-29 14:16:37.504	2023-05-18 12:05:55.288	1	62
<input type="checkbox"/>	Editor	79	Quam possimus sed perspiciatis culpa.	Quibusdam sunt omnis error quidem enim autem. Iure...	https://loremflickr.com/640/480?lock=8849412332919...	2020-01-21 15:26:46.559	2023-05-18 12:05:57.232	1	56
<input type="checkbox"/>	Editor	80	Voluptates commodi ducimus saepe voluptas inoidunt...	Qui animi laborum. Quibusdam modi beatae corrupti...	https://picsum.photos/seed/evdvwI00M/640/480	2022-11-08 20:48:14.532	2023-05-18 12:05:58.302	0	62
<input type="checkbox"/>	Editor	81	Veritatis vel cumque nisi sequi iste ut.	Eos tempora architecto debitis. Rem possimus ut na...	https://picsum.photos/seed/ftpLA/640/480	2020-05-10 16:43:32.886	2023-05-18 12:05:59.136	0	61
<input type="checkbox"/>	Editor	82	Temporibus dignissimos accusamus soluta nihil.	Similique quasi fugiat. Mollitia enim exoritation...	https://loremflickr.com/640/480?lock=7032671874517...	2021-03-18 08:43:21.060	2023-05-18 12:06:00.764	0	65
<input type="checkbox"/>	Editor	83	Nulla atque dolore rem beatae commodi quasi distin...	Placeat velit similique ab facilis. Velit voluptat...	https://loremflickr.com/640/480?lock=8615154190860...	2023-05-11 19:00:34.246	2023-05-18 12:06:01.568	0	59
<input type="checkbox"/>	Editor	84	Odio magni fuga impedit ullam odio sint.	Eligendi mollitia totam sint officia necessitatibu...	https://picsum.photos/seed/dfe4q/640/480	2022-10-31 20:03:38.831	2023-05-18 12:06:02.908	0	62
<input type="checkbox"/>	Editor	85	Sed corporis inoidunt cum praesentium velit dolore...	Distinctio tempore at aliquid non sit ab assumenda...	https://loremflickr.com/640/480?lock=1023849733872...	2022-06-18 06:50:30.400	2023-05-18 12:06:03.810	1	64
<input type="checkbox"/>	Editor	86	Facilis soluta repellendus aut adipisci.	Quia ab possimus explicabo quibusdam quos cum id m...	https://picsum.photos/seed/9OONGgog/640/480	2021-06-22 10:54:04.780	2023-05-18 12:06:05.233	1	64
<input type="checkbox"/>	Editor	87	Recusandae quam et placeat nulla.	Vitae illo nisi. Facere quaerat nam occaecati fuga...	https://picsum.photos/seed/9rBT/640/480	2022-02-17 07:45:24.399	2023-05-18 12:06:05.842	0	58
<input type="checkbox"/>	Editor	88	Voluptatum tenetur reprehenderit beatae ut molliti...	Officia similique laboriosam a suscipit omnis at m...	https://loremflickr.com/640/480?lock=4212073414262...	2021-02-05 12:18:05.478	2023-05-18 12:06:06.283	1	59
<input type="checkbox"/>	Editor	89	Assumenda placeat tempore at.	Delentis dolores nisi explicabo a inoidunt quam un...	https://picsum.photos/seed/ASZer9K/640/480	2021-10-22 03:50:38.142	2023-05-18 12:06:07.900	0	64
<input type="checkbox"/>	Editor	90	Officia quod aliquid dolorumque modi delectus qui.	Quos rerum quis harum. Laudantium id ab eveniet ma...	https://loremflickr.com/640/480?lock=2671124428177...	2020-01-27 22:01:01.488	2023-05-18 12:06:08.516	1	63
<input type="checkbox"/>	Editor	91	Animi quisquam eius quidem illo ducimus inventore...	Iusto tenetur error odit eligendi magnam magni asp...	https://loremflickr.com/640/480?lock=8616766372358...	2021-05-02 04:41:52.347	2023-05-18 12:06:08.555	1	63
<input type="checkbox"/>	Editor	92	Quam minima distinctio et earum velit corporis.	Nesciunt cupiditate dolores. Inoidunt quidem repel...	https://picsum.photos/seed/LuQQL4Pn/640/480	2020-07-22 12:02:43.017	2023-05-18 12:06:09.738	1	63
<input type="checkbox"/>	Editor	93	Suscipit quaerat perspiciatis iste est minima laud...	A sunt quibusdam dicta placeat. Amet tenetur enim...	https://picsum.photos/seed/zydq6C9K/640/480	2021-05-18 10:22:14.730	2023-05-18 12:06:10.592	1	65
<input type="checkbox"/>	Editor	94	Error ab vitae animi eveniet quos non provident ma...	Vitae corporis dolorumque error quod dolorumque tot...	https://loremflickr.com/640/480?lock=4822383015002...	2021-04-14 11:31:26.774	2023-05-18 12:06:11.643	1	64
<input type="checkbox"/>	Editor	95	Placeat quam nulla repellendus accusantium quia pa...	Illo distinctio impedit quisquam accusamus amet qu...	https://loremflickr.com/640/480?lock=8245108444538...	2021-09-30 16:52:26.278	2023-05-18 12:06:11.787	0	65

Article :

GET

```

Send Request
GET http://localhost:3000/articles?take=2&skip=0 HTTP/1.1

###
Send Request

{
  "id": 1,
  "title": "Nulla persiciatis fugit doloribus necessitatibus facilis ducimus provident.",
  "contenu": "Assumenda possimus excepturi pariatur animi sed adipisci ad ullam. Delectus aperiam impedit sequi unde eaque voluptatem. Sapiente accusantium excepturi veritatis.",
  "image": "https://loremflickr.com/640/480?lock=5628235823972352",
  "createdAt": "2020-09-13T11:23:08.669Z",
  "updatedAt": "2023-05-18T12:05:01.929Z",
  "published": false,
  "userId": 56
},
{
  "id": 2,
  "title": "Minima atque ut quasi quis dignissimos quae quisquam.",
  "contenu": "Quam atque maiores perferendis veniam illum a. Dolorum libero mollitia quo sequi porro rem. Eveniet cumque esse aperiam eveniet.",
  "image": "https://picsum.photos/seed/KEvZ4h3/640/480",
  "createdAt": "2021-01-24T09:28:23.487Z",
  "updatedAt": "2023-05-18T12:05:02.308Z",
  "published": false,
  "userId": 58
}
}

###
Send Request
GET http://localhost:3000/articles/103

{
  "id": 103,
  "title": "Article mis à jour",
  "contenu": "Nouveau contenu de l'article",
  "image": "https://loremflickr.com/640/480?lock=1314580215103488",
  "createdAt": "2020-03-26T17:37:33.386Z",
  "updatedAt": "2020-03-26T17:37:33.386Z",
  "published": true,
  "userId": 56
}

```

POST

```

Send Request
POST http://localhost:3000/articles

Content-Type: application/json

{
  "title": "Article Test",
  "contenu": "Ceci est un test",
  "image": "https://picsum.photos/500/300.jpg?",
  "createdAt": "2023-05-14T00:00:00.000Z",
  "updatedAt": "2023-05-14T00:00:00.000Z",
  "published": true,
  "userId": 57
}

{
  "id": 103,
  "title": "Article Test",
  "contenu": "Ceci est un test",
  "image": "https://picsum.photos/500/300.jpg?",
  "createdAt": "2023-05-14T00:00:00.000Z",
  "updatedAt": "2023-05-14T00:00:00.000Z",
  "published": true,
  "userId": 57
}

```

PATCH

```

Send Request
PATCH http://localhost:3000/articles/103

Content-Type: application/json

{
  "title": "Article mis à jour",
  "contenu": "Nouveau contenu de l'article",
  "image": "https://loremflickr.com/640/480?lock=1314580215103488",
  "createdAt": "2020-03-26T17:37:33.386Z",
  "updatedAt": "2023-05-18T12:05:05.023Z",
  "published": true,
  "userId": 56
}

{
  "id": 103,
  "title": "Article mis à jour",
  "contenu": "Nouveau contenu de l'article",
  "image": "https://loremflickr.com/640/480?lock=1314580215103488",
  "createdAt": "2020-03-26T17:37:33.386Z",
  "updatedAt": "2020-03-26T17:37:33.386Z",
  "published": true,
  "userId": 56
}

```

DELETE

```

Send Request
DELETE http://localhost:3000/articles/6

{
  "message": "Article with id 6 deleted."
}

```

Catégorie :

GET

```

Send Request
GET http://localhost:3000/categories?take=5&skip=0

9 ∨ [
10 ∨ {
11   "id": 41,
12   "nom": "dolores"
13 },
14 ∨ {
15   "id": 42,
16   "nom": "iste"
17 },
18 ∨ {
19   "id": 43,
20   "nom": "id"
21 },
22 ∨ {
23   "id": 44,
24   "nom": "consequatur"
25 },
26 ∨ {
27   "id": 45,
28   "nom": "ut"
29 }
30 ]

```

```

8   ### Get category by id
Send Request
9   GET http://localhost:3000/categories/49
10
11   {
12     "id": 49,
13     "nom": "odio"
14   }

```

POST

```

Send Request
POST http://localhost:3000/categories
Content-Type: application/json

{
  "nom": "New category added in db"
}

9 ∨ {
10   "id": 57,
11   "nom": "New category added in db"
12 }

```

PATCH

```

Send Request
PATCH http://localhost:3000/categories/57
Content-Type: application/json

{
  "nom": "An updated category"
}

9 ∨ {
10   "id": 57,
11   "nom": "An updated category"
12 }

```

DELETE

```

### Delete category
Send Request
DELETE http://localhost:3000/categories/57

8
9 ∨ {
10   "message": "Categorie with id 57 deleted."
11 }

```

Commentaire

GET

```

Send Request
GET http://localhost:3000/commentaires?take=2&skip=5

### Get commentaires by id

9 √ [
10 √ {
11   "id": 6,
12   "email": "Moises_Kuhic@gmail.com",
13   "contenu": "Quam quasi facere corrupti numquam eos. Corrupti aut laud
antium quisquam reiciendis similique necessitatibus animi assumenda a. Ve
ro voluptatum debitis quibusdam suscipit doloribus officia do",
14   "articleId": 2,
15   "userId": 62
16 },
17 √ {
18   "id": 7,
19   "email": "Lizeth.Stamm73@gmail.com",
20   "contenu": "Error minus dolorem doloremque voluptatem libero necessit
atibus tempore ratione. Atque pariatur sed architecto laudantium incidun
t. Officiis quasi vero commodi fuga a maxime neque consequuntur",
21   "articleId": 3,
22   "userId": 59
23 }
24 ]

```

```

### Get commentaire by id
Send Request
GET http://localhost:3000/commentaires/1

9 √ {
10   "id": 1,
11   "email": "Fannie.Buckridge74@hotmail.com",
12   "contenu": "At eos voluptas deserunt. Beatae minus porro consequatur ul
lam repellendus fugit fugit. Iste minima voluptas nemo dicta optio nam.",
13   "articleId": 1,
14   "userId": 61
15 }

```

POST

```

### Create new commentaire !!
Send Request
POST http://localhost:3000/commentaires/
Content-Type: application/json

{
  "email": "testComment@gmail.com",
  "contenu": "This is my testing comment",
  "articleId": 2,
  "userId": 59
}

7 Connection: close
8
9 √ {
10   "id": 1014,
11   "email": "testComment@gmail.com",
12   "contenu": "This is my testing comment",
13   "articleId": 2,
14   "userId": 59
15 }

```

PATCH

```

Send Request
PATCH http://localhost:3000/commentaires/1014
Content-Type: application/json

{
  "contenu": "update on my comment",
  "articleId": 2,
  "userId": 59
}

8
9 √ {
10   "id": 1014,
11   "email": "testComment@gmail.com",
12   "contenu": "update on my comment",
13   "articleId": 2,
14   "userId": 59
15 }

```

DELETE

```

### Delete commentaire
Send Request
DELETE http://localhost:3000/commentaires/1014

8
9 √ {
10   "message": "Commentaire with id 1014 deleted."
11 }

```

GET

POST

PATCH

DELETE

```
####
Send Request
DELETE http://localhost:3000/users/69

8
9  ✓ {
10    "message": "User with id 69 deleted."
11  }
```

Authentification

Login :

Dans mon fichier que j'ai l'appellé Checkin.js :

```
require('dotenv').config(); 2.9k (gzipped: 1.4k)
const jwt = require("jsonwebtoken"); 125.8k (gzipped: 41.2k)
app.post("/login", async (req, res) => {
  const { email, password } = req.body;
  const user = await prisma.user.findUnique({ where: { email: email } });
  const secretKey = process.env.SECRET_KEY;

  if (user && user.password === password) {
    // Informations d'identification valides
    const token = jwt.sign({ email, role: "AUTHOR" }, secretKey);
    const role = user.role;
    const id = user.id;

    res.json({ token, role, id });
  } else {
    // Informations d'identification invalides
    res.status(401).json({ message: "Email ou mot de passe incorrect" });
  }
});
```

On va rechercher l'utilisateur ayant l'adresse e-mail saisie, car elle est unique. Ensuite, On va vérifier le mot de passe et le rôle de l'utilisateur en utilisant le code JavaScript suivant :

```
document.getElementById("login_form").addEventListener("submit", function(event) {
  event.preventDefault(); // Empêche la soumission du formulaire
  // Récupération des valeurs des champs du formulaire
  const email_lf = document.querySelector("input[name='email']").value;
  const password_lf = document.querySelector("input[name='password']").value;
  fetch("http://localhost:3000/login", {
    method: "POST",
    headers: {
      "Content-Type": "application/json"
    },
    body: JSON.stringify({
      email: email_lf,
      password: password_lf
    })
  })
  .then(response => response.json())
  .then(data => {
    if (data.token) {
      const token = data.token;
      const role = data.role;
      const getUserId = data.id;
      if (role === "AUTHOR") { ... }
      else if (role === "ADMIN") { ... }
    }
    else {
      alert("Email ou mot de passe incorrect");
    }
  })
  .catch(error => {
    console.error("Erreur lors de la connexion :", error);
  });
});
```

Sign up :

Voici comment un nouvel utilisateur peut s'inscrire :

```

56 //sign up
57 $(document).ready(function() {
58     $("#signup_form").submit(function(event) {
59         event.preventDefault();
60         var nom = $("#nom_signup").val();
61         var email = $("#email_signup").val();
62         var password = $("#password_signup").val();
63         // Création de l'objet de données JSON à envoyer
64         var utilisateur = {
65             nom: nom,
66             email: email,
67             password: password,
68             role: "AUTHOR"
69         };
70         // Envoi de la requête POST avec les données JSON
71         $.ajax({
72             url: "http://localhost:3000/users",
73             type: "POST",
74             contentType: "application/json",
75             data: JSON.stringify(utilisateur),
76             success: function(response) {
77                 alert("Vous êtes inscrit avec succès:", response);
78             },
79             error: function(xhr, status, error) {
80                 console.error("Erreur lors de l'ajout de l'utilisateur :", error);
81             }
82         });
83     });
84 });

```

Navigation sans rafraîchissement de la page :

→ Concernant la navigation sur les articles sans rafraîchissement de la page, nous allons simplement ajouter un nombre à la variable "take". Par exemple, si nous avons pris (take=10, skip=0), nous allons augmenter "take" de 10 dans une fonction pour afficher d'autres articles à partir de notre base de données.

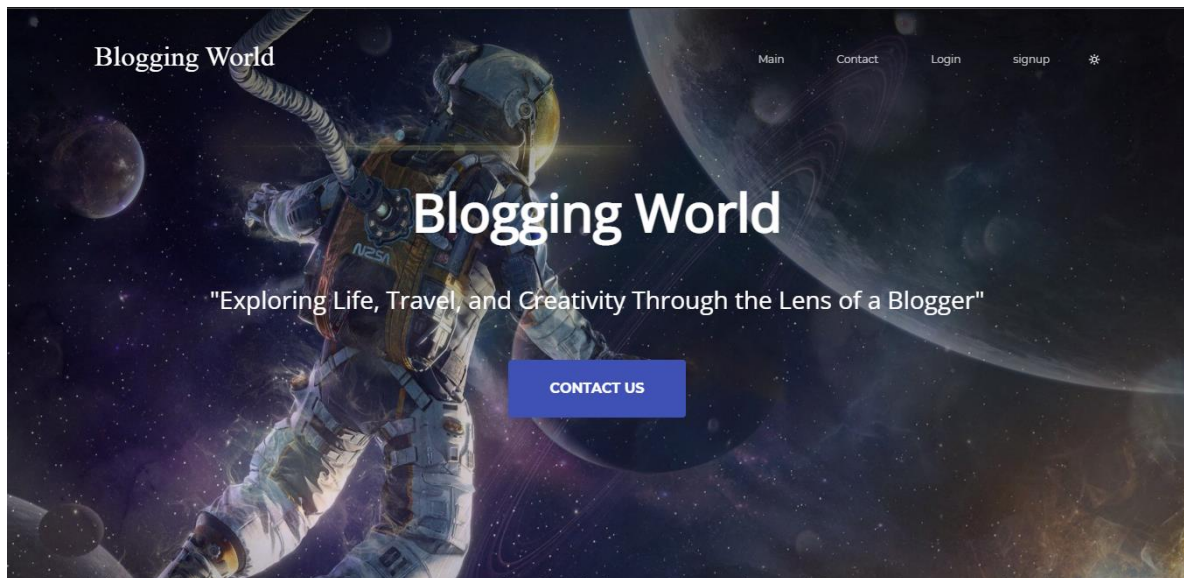
```

176 // Fonction pour effectuer et mettre à jour le contenu de la page
177 function loadArticles(take, skip) {
178     $.ajax({
179         url: 'http://localhost:3000/articles/',
180         data: { take: take, skip: skip },
181         dataType: 'json',
182         success: function (articles) {
183             if (skip === 0) $('#articles-list').empty();
184
185             // Ajouter les nouveaux articles
186             articles.forEach(function (article) {
187                 var listItem = $('<li>').text(article.title);
188                 $('#articles-list').append(listItem);
189             });
190
191             // Ajouter un bouton "Charger plus" si le nombre d'articles chargés est inférieur à take
192             if (articles.length < take) $('#load-more-button').hide();
193             else $('#load-more-button').show();
194         },
195         error: function (xhr, status, error) {
196             console.error(error);
197         }
198     });
199     // Gérer le clic sur le bouton "Charger plus"
200     $('#load-more-button').on('click', function () {
201         // récupérer les valeurs actuelles de take et skip
202         var currentTake = parseInt($(this).data('take'));
203         var skip = parseInt($(this).data('skip'));
204
205         var newTake = currentTake + 10;
206         $(this).data('take', newTake); // Mettre à jour la valeur de take dans les attributs de données du bouton
207         loadArticles(newTake, skip);
208     });

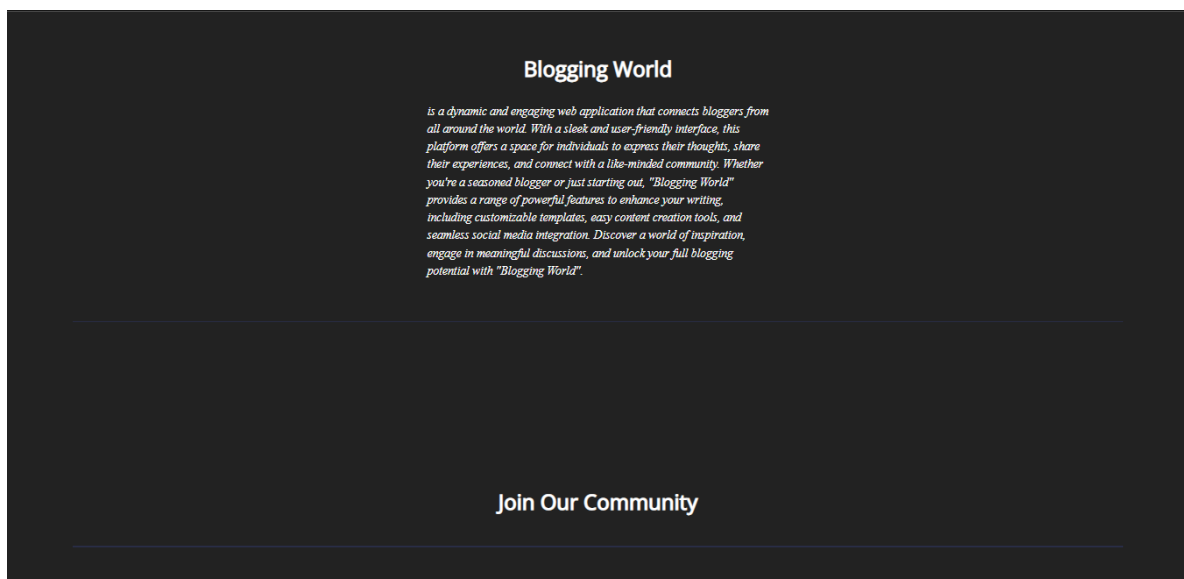
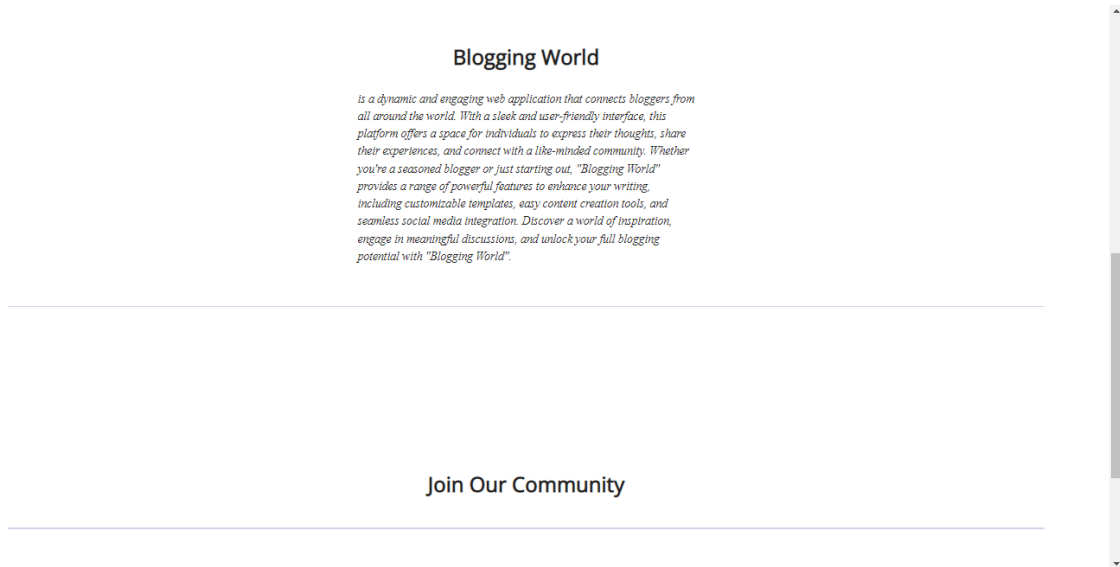
```


Frontend :

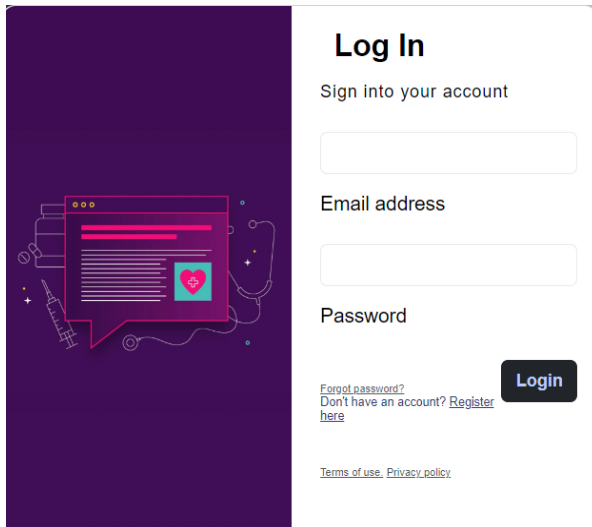
➔ La page d'accueil contient des arrière-plans qui changent avec des animations, et on peut changer le mode de la page en mode sombre ou clair.



→ Il y a au-dessous une petite présentation sur mon site web (la vue avec les deux modes)



→ Les deux interfaces d'authentification



Log In

Sign into your account

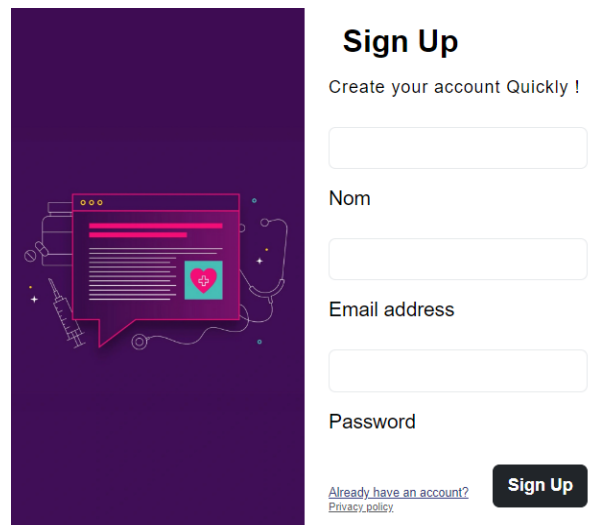
Email address

Password

[Forgot password?](#)
Don't have an account? [Register here](#)

[Terms of use](#) [Privacy policy](#)

Login



Sign Up

Create your account Quickly !

Nom

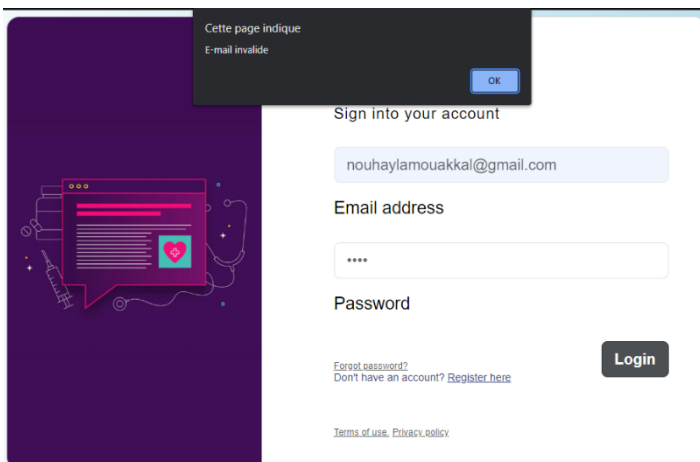
Email address

Password

[Already have an account?](#)
[Privacy policy](#)

Sign Up

→ On vérifie d'abord si l'email existe déjà dans notre base de données. Si c'est le cas, nous vérifions également le mot de passe. Si le mot de passe est également correct, nous vérifions le rôle de cet utilisateur afin de le rediriger vers son profil.



Cette page indique
E-mail invalide

Log In

Sign into your account

nouhaylamouakkal@gmail.com

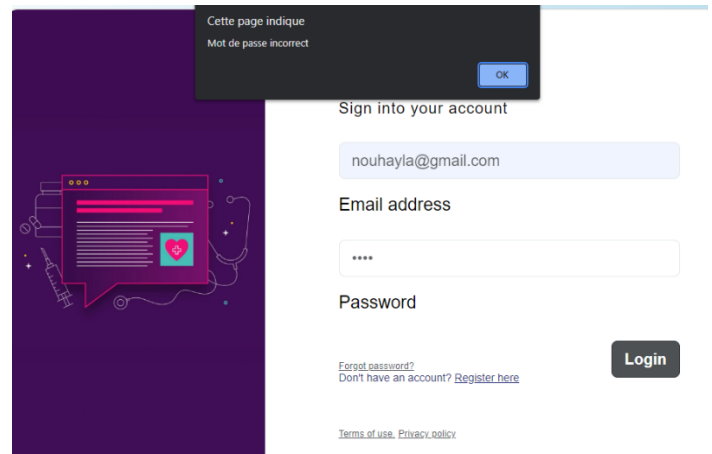
Email address

Password

[Forgot password?](#)
Don't have an account? [Register here](#)

[Terms of use](#) [Privacy policy](#)

Login



Cette page indique
Mot de passe incorrect

Log In

Sign into your account

nouhayla@gmail.com

Email address

Password

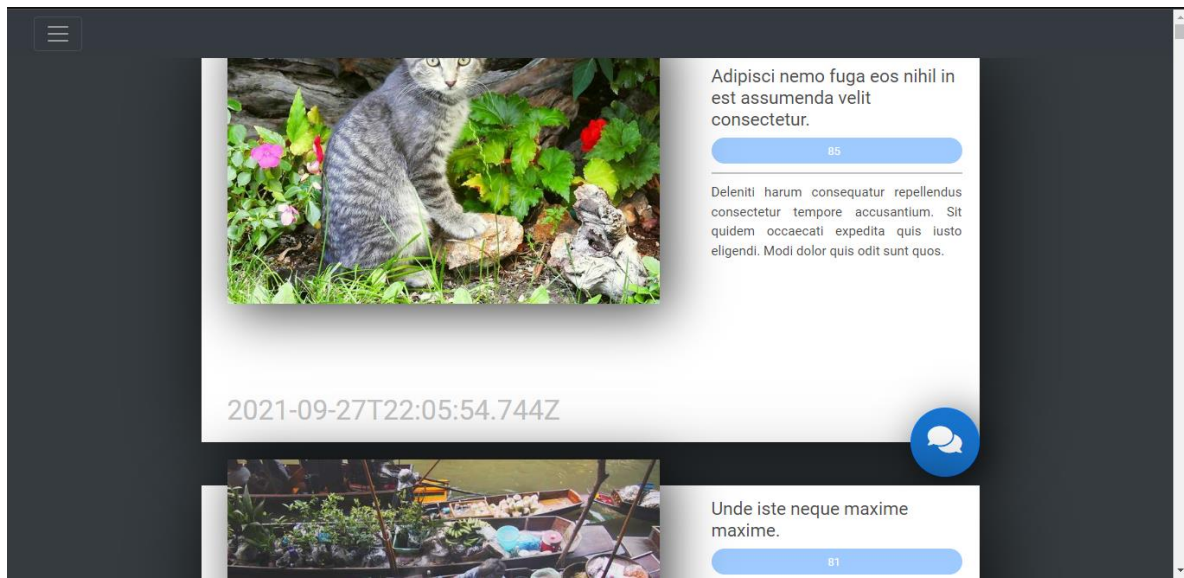
[Forgot password?](#)
Don't have an account? [Register here](#)

[Terms of use](#) [Privacy policy](#)

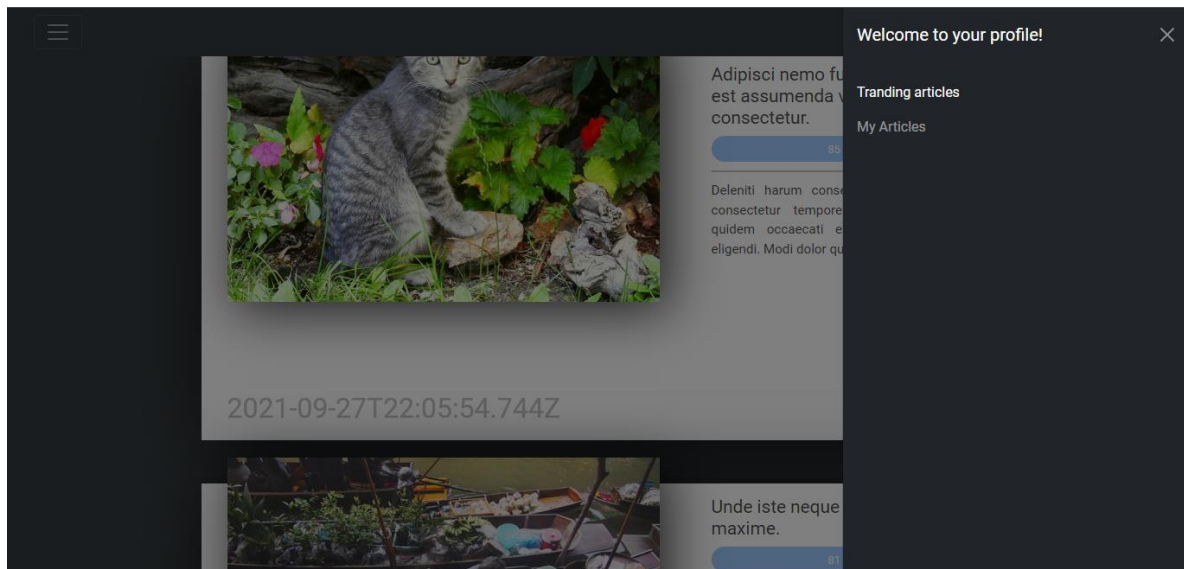
Login

Interface Auteur :

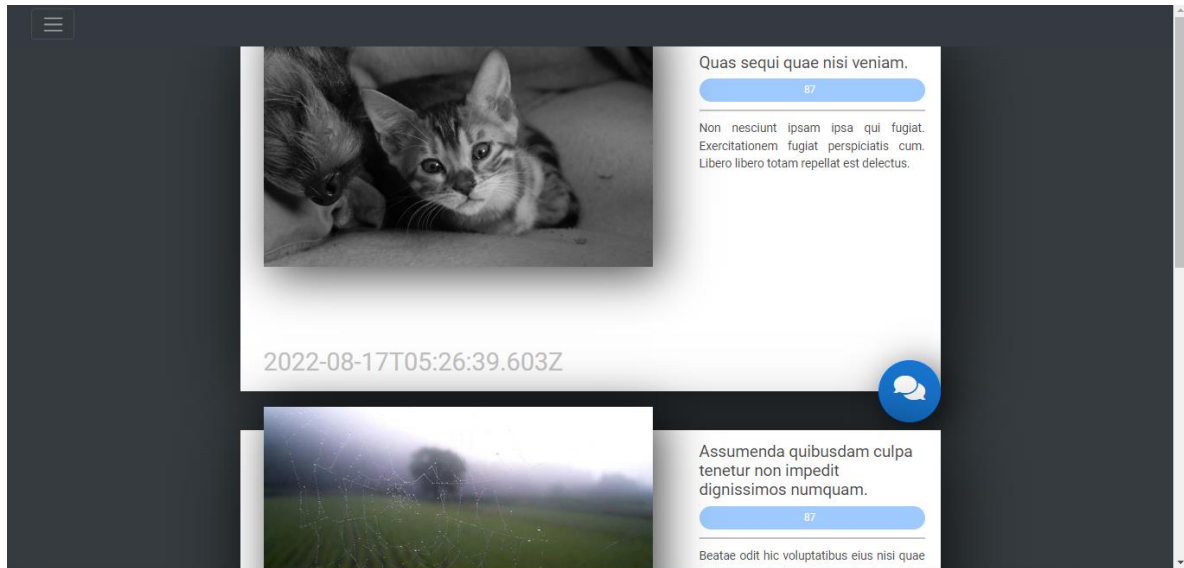
➔ Cette page affichera tous les articles présents dans la base de données. Vous pourrez y lire différents articles de différentes catégories ainsi que les commentaires associés à chaque article.



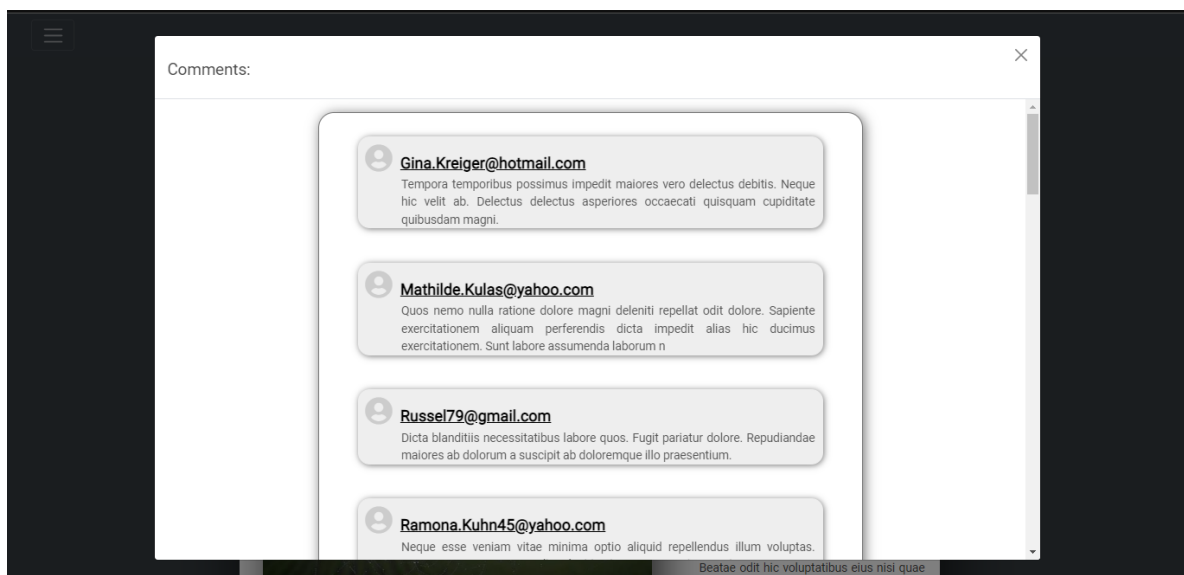
➔ Nav bar qui donne l'utilisateur de voir tous les articles ou bien ses propres articles



→Voila un exemple des articles de notre utilisateur

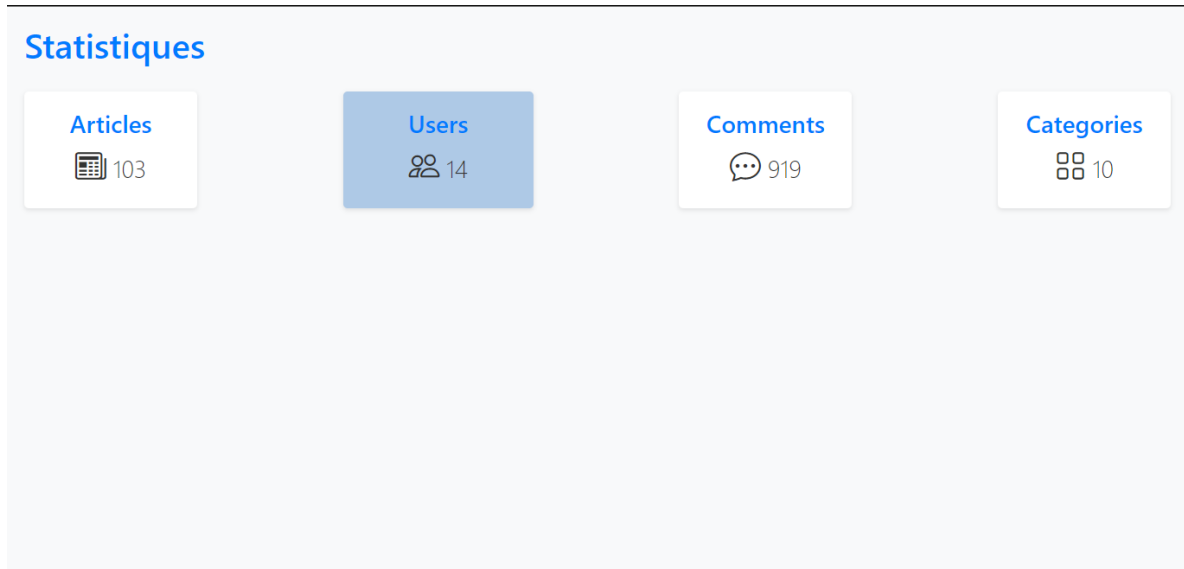


→ Lorsque vous cliquez sur l'icône des commentaires, vous pouvez les parcourir et les visualiser.

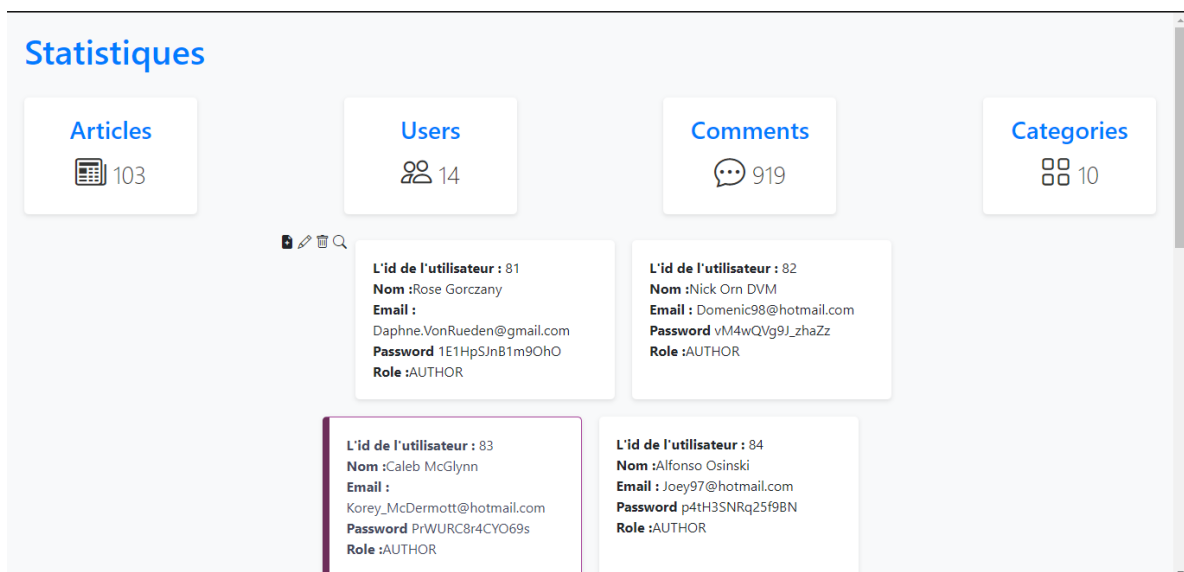


Interface Admin :

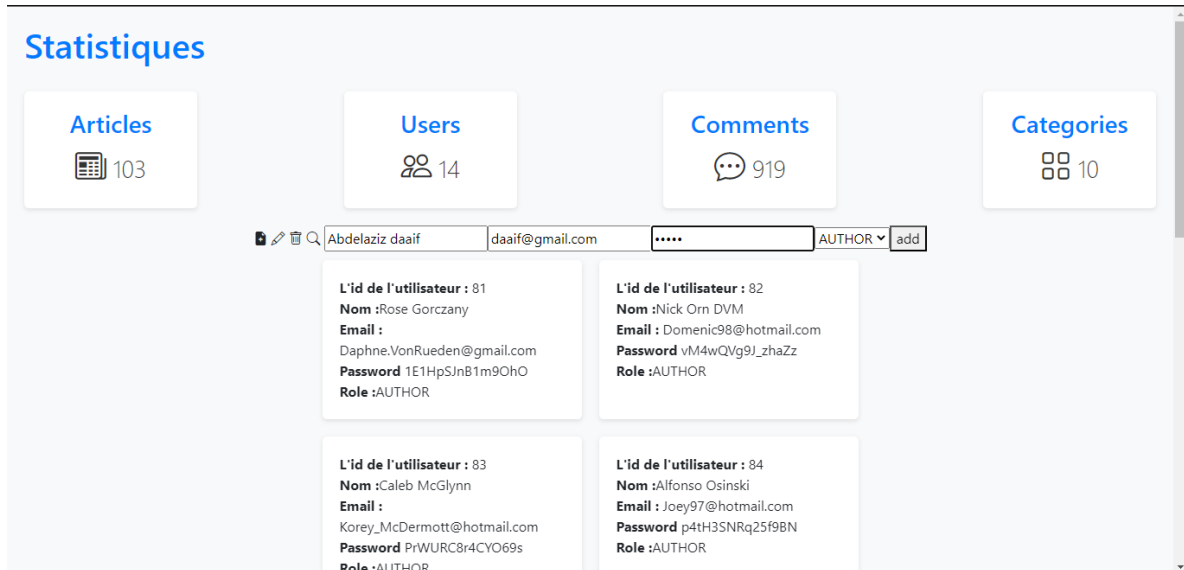
→ Admin peut voir le nombre total de chaque entité :



→ Et pour chaque entité il peut faire (ajout , modification , suppression , recherche)



→ Exemple d'ajout d'un utilisateur



Statistiques

- Articles: 103
- Users: 14
- Comments: 919
- Categories: 10

Form fields:

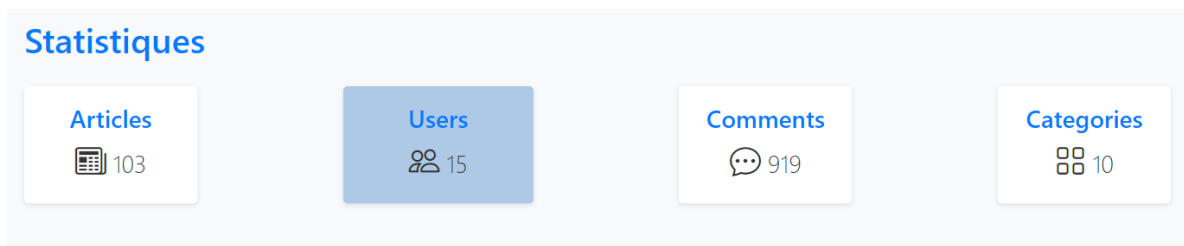
L'id de l'utilisateur : 81
Nom : Rose Gorczany
Email : Daphne.VonRueden@gmail.com
Password : 1E1HpSjnB1m9OhO
Role : AUTHOR

L'id de l'utilisateur : 82
Nom : Nick Orn DVM
Email : Domenic98@hotmail.com
Password : vM4wQVg9J_zhaZz
Role : AUTHOR

L'id de l'utilisateur : 83
Nom : Caleb McGlynn
Email : Korey_McDermott@hotmail.com
Password : PrWURC8r4CYO69s
Role : AUTHOR

L'id de l'utilisateur : 84
Nom : Alfonso Osinski
Email : Joey97@hotmail.com
Password : p4tH3SNRq25f9BN
Role : AUTHOR

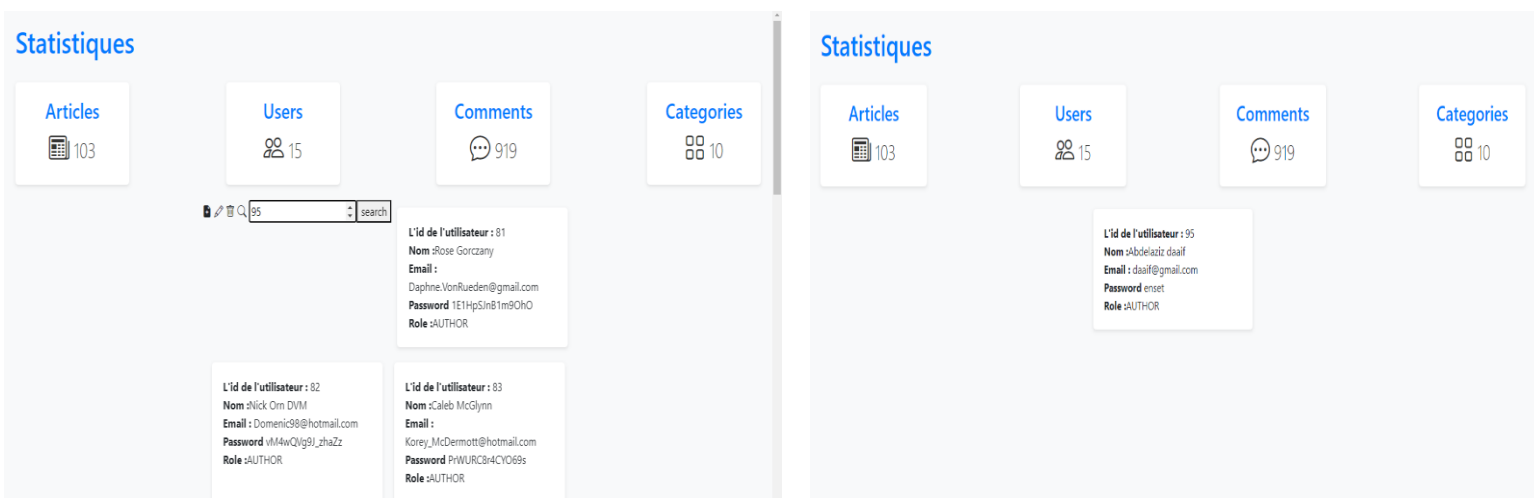
→ On peut voir que l'utilisateur a été ajouté et que le nombre total d'utilisateurs a été mis à jour.



Statistiques

- Articles: 103
- Users: 15
- Comments: 919
- Categories: 10

→ Exemple de recherche de notre utilisateur qu'on a ajouté



Statistiques

- Articles: 103
- Users: 15
- Comments: 919
- Categories: 10

Search:

L'id de l'utilisateur : 81
Nom : Rose Gorczany
Email : Daphne.VonRueden@gmail.com
Password : 1E1HpSjnB1m9OhO
Role : AUTHOR

L'id de l'utilisateur : 82
Nom : Nick Orn DVM
Email : Domenic98@hotmail.com
Password : vM4wQVg9J_zhaZz
Role : AUTHOR

L'id de l'utilisateur : 83
Nom : Caleb McGlynn
Email : Korey_McDermott@hotmail.com
Password : PrWURC8r4CYO69s
Role : AUTHOR

L'id de l'utilisateur : 95
Nom : Abdelaziz daaif
Email : daaif@gmail.com
Password : enset
Role : AUTHOR

→ Exemple de modification

Statistiques

Articles
103

Users
15

Comments
919

Categories
10

95

prof abdelaziz daaif

daaif@gmail.com

.....

AUTHOR

add

L'id de l'utilisateur : 81

Nom : Rose Gorczany

Email : Daphne.VonRueden@gmail.com

Password : 1E1HpSJnB1m9OhO

Role : AUTHOR

L'id de l'utilisateur : 82

Nom : Nick Orn DVM

Email : Domenic98@hotmail.com

Password : vM4wQVg9J_zhaZz

Role : AUTHOR

L'id de l'utilisateur : 83

Nom : Caleb McGlynn

Email : Korey_McDermott@hotmail.com

Password : PdMILBC8rACVd69e

Role : AUTHOR

L'id de l'utilisateur : 84

Nom : Alfonso Osinski

Email : Joey97@hotmail.com

Password : p4tH3SNRq25f9BN

Role : AUTHOR

→ Voilà le résultat

Statistiques

Articles
103

Users
15

Comments
919

Categories
10

L'id de l'utilisateur : 95

Nom : prof abdelaziz daaif


Email : daaif@gmail.com


Password : ii-bdcccglid


Role : AUTHOR


→ Finalement, la suppression d'un utilisateur



Statistiques

Articles
 103

Users
 15

Comments
 919

Categories
 10


L'id de l'utilisateur : 81
Nom : Rose Gorczany
Email : Daphne.VonRueden@gmail.com
Password : 1E1HpSJnB1m9OhO
Role : AUTHOR


L'id de l'utilisateur : 82
Nom : Nick Orn DVM
Email : Domenic98@hotmail.com
Password : vM4wQVg9J_zhaZz
Role : AUTHOR


L'id de l'utilisateur : 83
Nom : Caleb McGlynn
Email : Korey_McDermott@hotmail.com
Password : PrWURC8r4CYO69s
Role : AUTHOR


→ La suppression a été effectuée avec succès et le nombre total d'utilisateurs a été modifié.

Statistiques

Articles
 103

Users
 14

Comments
 919

Categories
 10

Conclusion :

Ce projet m'a permis de mettre en valeur mes atouts et connaissances tout en découvrant une approche de conception et d'analyse. J'ai respecté les délais pour les documents à rendre et j'ai acquis une nouvelle expérience dans le monde de l'informatique, approfondissant mes connaissances en développement web et découvrant de nouvelles technologies pour résoudre certains problèmes. Mon projet a pleinement répondu aux exigences du cahier des charges. En outre, il s'agit d'une première étape pour de futurs développements. À l'avenir, cette application pourrait être améliorée et enrichie en ajoutant des fonctionnalités avancées. Bien que le côté backend ait été respecté, il existe des interfaces côté frontend qui pourraient être ajoutées et améliorées, mais en raison de contraintes de temps, je n'ai pas pu tester tous les scénarios sur différentes interfaces.