

# Documentation de l'Application

## I. Objectif :

L'application web personnalisée offre une solution efficace pour organiser et gérer les tâches quotidiennes de manière optimale.

## II. Technologies Utilisées :

L'application est construite en utilisant la stack MERN (MongoDB, Express.js, React.js, Node.js), une combinaison de technologies modernes permettant de développer des applications web robustes et évolutives.

## III. Fonctionnalités :

### ➤ Création de Compte

- Description : Permet aux utilisateurs de créer un compte personnel.
- Instructions :

Accédez à la page d'inscription.

Remplissez le formulaire avec les informations requises.

### ➤ Espace de Travail Personnel

- Description : Chaque utilisateur a un espace de travail personnel pour gérer ses tâches.
- Instructions :

Connectez-vous avec vos identifiants.

Accédez à votre espace de travail personnalisé.

### ➤ Ajout, Modification et Suppression

- Description : Les utilisateurs peuvent ajouter, modifier ou supprimer des tâches.
- Instructions :

Ajoutez une nouvelle tâche avec la date d'aujourd'hui.

Modifiez une tâche existante en modifiant les informations.

Supprimez une tâche que vous n'avez plus besoin.

➤ **Changement de Statut**

- Description : Les utilisateurs peuvent marquer une tâche comme "En cours", "Done" ou "Not Done".
- Instructions :

Sélectionnez une tâche.

Choisissez le statut approprié dans les options disponibles.

➤ **Recherche**

- Description : Les utilisateurs peuvent effectuer une recherche de tâches par mots-clés ou par date.
- Instructions :

Utilisez la fonction de recherche.

Saisissez les mots-clés, la date ou date d'aujourd'hui.

➤ **Tri Alphabétique**

- Description : Les tâches sont organisées dans l'affichage selon l'ordre alphabétique.
- Instructions :

Visualisez vos tâches triées par ordre alphabétique.

➤ **Affichage des Tâches du Jour**

- Description : Les utilisateurs peuvent voir les tâches prévues pour la journée en cours.
- Instructions :

Accédez à la vue quotidienne pour voir les tâches du jour.

➤ **Déconnexion**

- Description : Permet aux utilisateurs de se déconnecter de leur compte.
- Instructions :

Trouvez l'option de déconnexion dans l'icône de profil.

Confirmez pour vous déconnecter en toute sécurité.

#### **IV. API :**

➤ **EndPoint liés aux tâches :**

- GET /AllTasks

Description : Récupère toutes les tâches de la base de données.

Réponse : Retourne un tableau de toutes les tâches.

The screenshot shows a REST client interface with a GET request to `http://localhost:5000/api/AllTasks`. The response is a JSON array of task objects. The status is 200 OK, time is 54 ms, and size is 1.08 KB.

Key	Value	Description
Key	Value	Description

```
1 [{"_id": "65c0915286169a0b0545d73f", "id_user": "65bfd5bc6a862904e90b1655", "Titre": "titre by ali", "Description": "description by ali", "Statut": "en cours", "DateEcheance": "2024-01-05T07:42:10.439Z", "__v": 0}, {"_id": "65c0916786169a0b0545d744", "id_user": "65bfd5bc6a862904e90b1655", "Titre": "second titre by ali", "Description": "another description by ali", "Statut": "en cours", "DateEcheance": "2024-02-05T07:42:31.649Z", "__v": 0}, {"_id": "65c0919586169a0b0545d74e", "id_user": "65bffe8e384bede9b810e590", "Titre": "title by mahmoud", "Description": "description by mahmoud", "Statut": "en cours", "DateEcheance": "2024-02-05T07:43:17.471Z", "__v": 0}, {"_id": "65c091b986169a0b0545d753", "id_user": "65bffe8e384bede9b810e590", "Titre": "Mahmoud second title", "Description": "Mahmoud Description", "Statut": "en cours", "DateEcheance": "2024-02-03T07:43:53.214Z", "__v": 0}]
```

- GET /GetById/:id

Description : Récupère une tâche spécifique par son ID.

Réponse : Retourne la tâche avec l'ID spécifié.

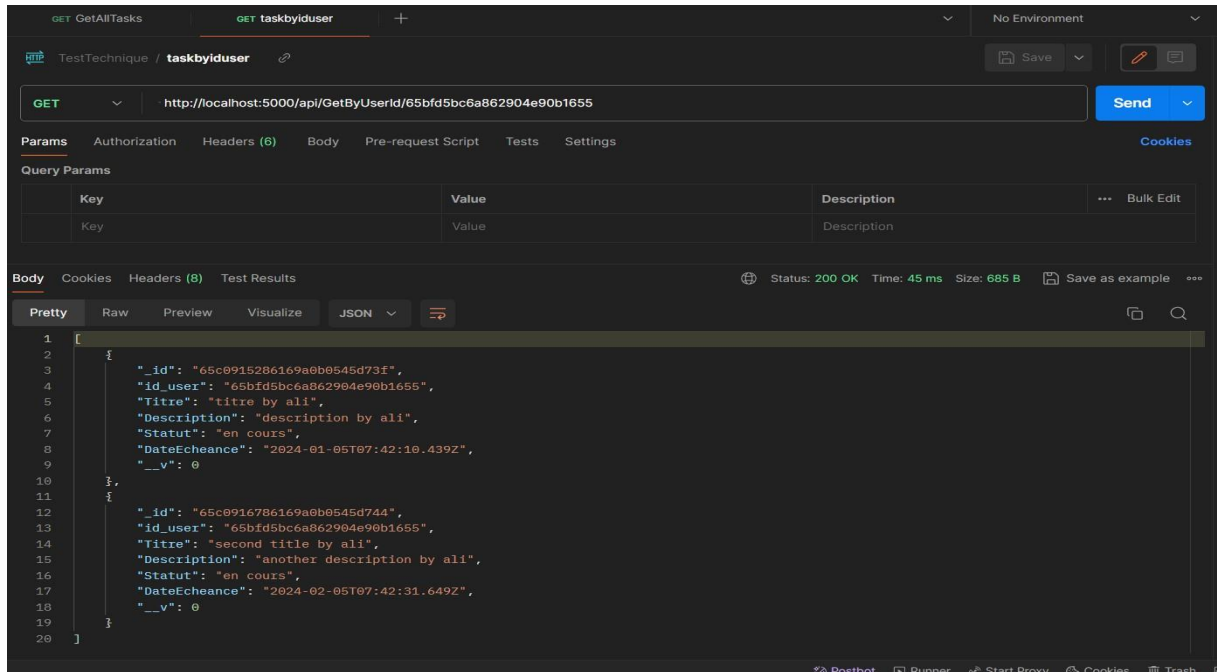
The screenshot shows a REST client interface with a GET request to `http://localhost:5000/api/GetById/65c0915286169a0b0545d73f`. The response is a single task object in JSON format. The status is 200 OK, time is 51 ms, and size is 467 B.

```
1 {
2   "_id": "65c0915286169a0b0545d73f",
3   "id_user": "65bfd5bc6a862904e90b1655",
4   "Titre": "titre by ali",
5   "Description": "description by ali",
6   "Statut": "en cours",
7   "DateEcheance": "2024-01-05T07:42:10.439Z",
8   "__v": 0
9 }
```

- GET /GetByUserId/:id\_user

Description : Récupère les tâches associées à un ID d'utilisateur spécifique.

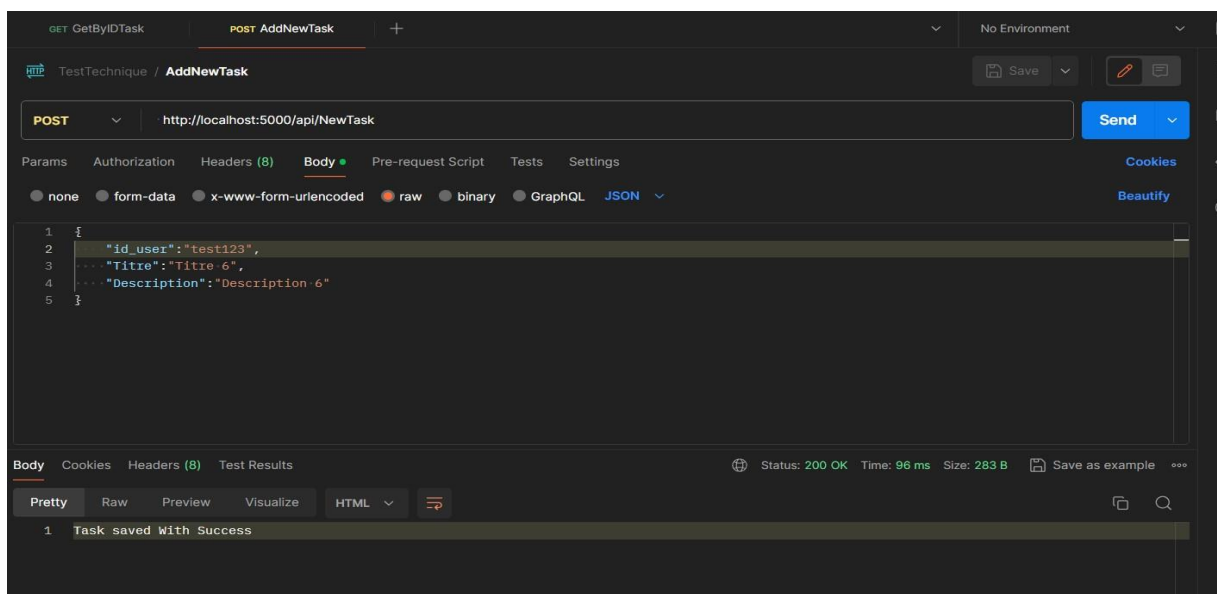
Réponse : Retourne un tableau de tâches associées à l'ID utilisateur spécifié.



- POST /NewTask

Description : Ajoute une nouvelle tâche à la base de données.

Réponse : En cas de succès, retourne un message de réussite ; sinon, retourne un message d'erreur.

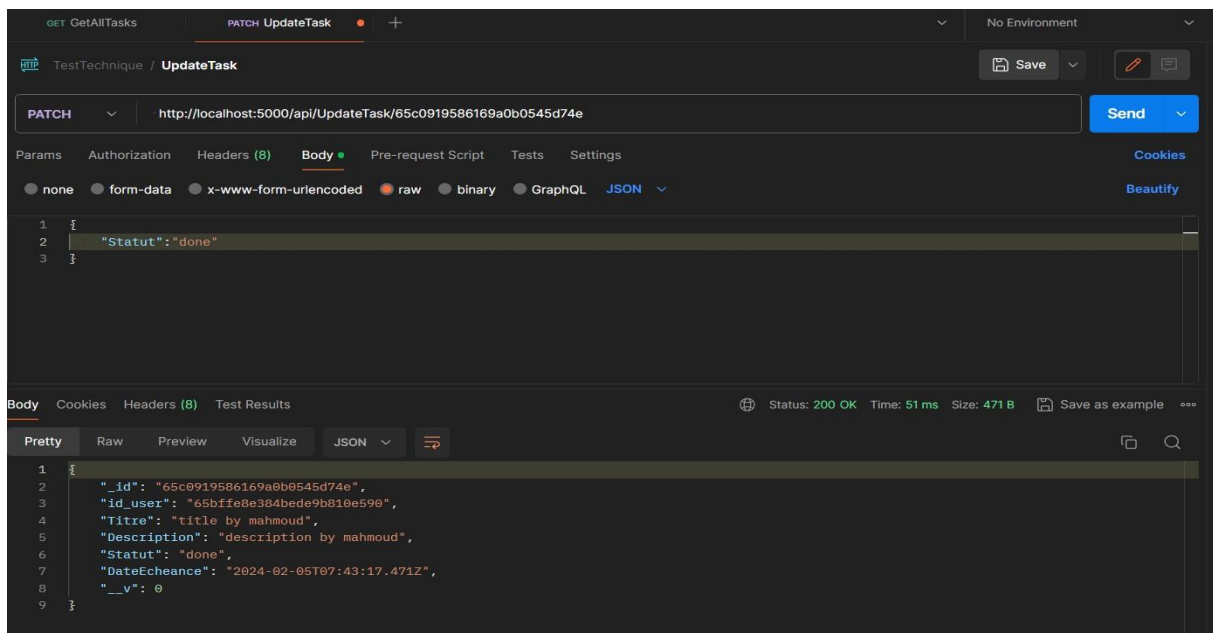


- PATCH /UpdateTask/:id

Description : Met à jour une tâche par son ID.

Corps de la requête : Accepte un objet JSON avec les champs à mettre à jour.

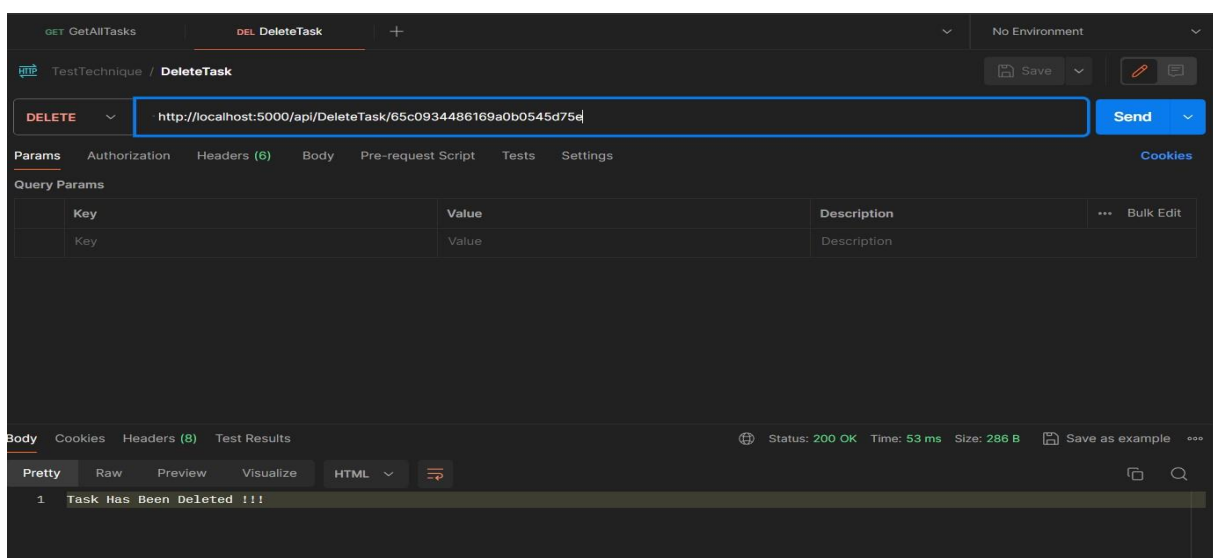
Réponse : Retourne la tâche mise à jour.



- DELETE /DeleteTask/:id

Description : Supprime une tâche par son ID.

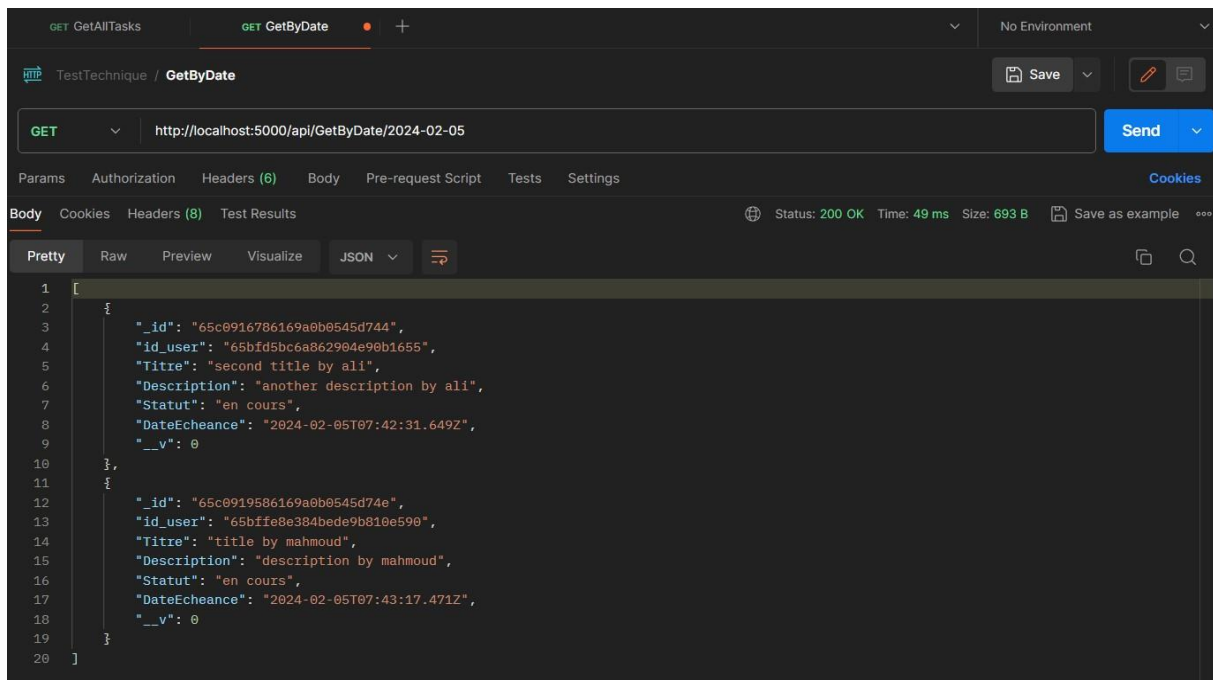
Réponse : Retourne un message de réussite si la suppression est réussie.



- GET /GetByDate/:date

Description : Récupère les tâches avec une date d'échéance ultérieure à la date fournie.

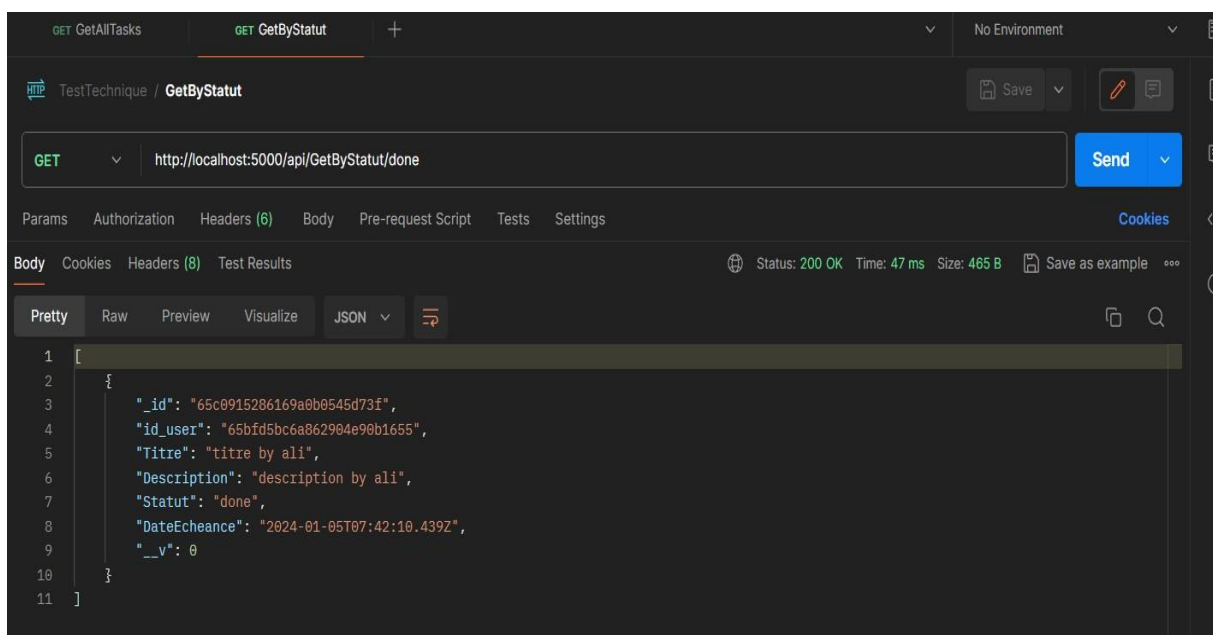
Réponse : Retourne un tableau de tâches.



- GET /GetByStatut/:statut

Description : Récupère les tâches par leur statut.

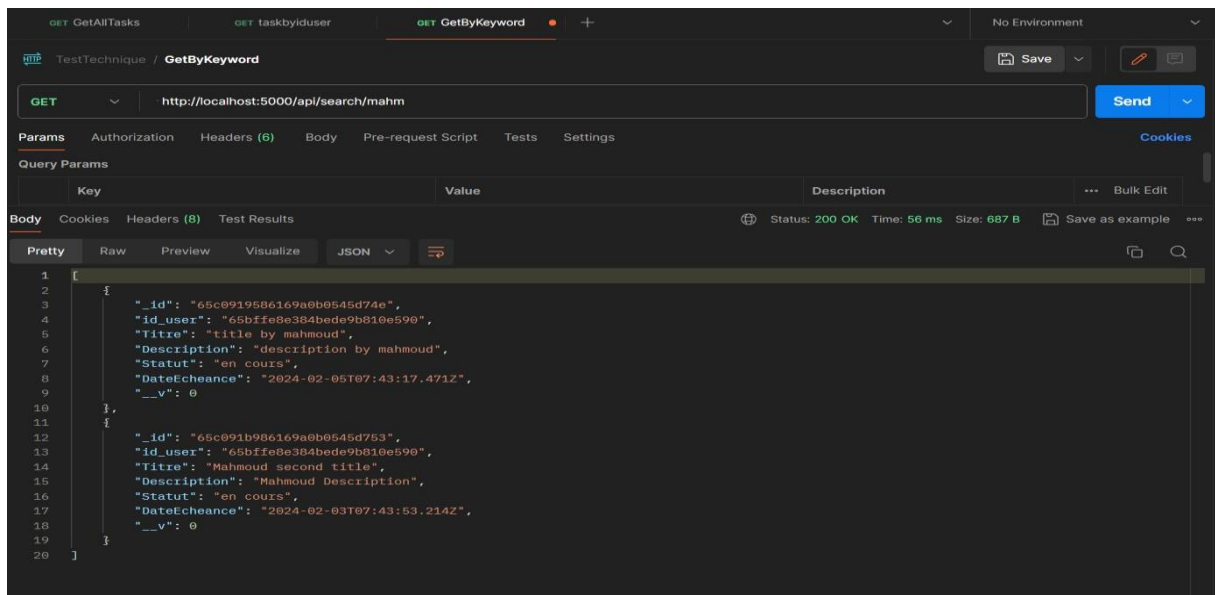
Réponse : Retourne un tableau de tâches avec le statut spécifié.



- GET /search/:keyword

Description : Recherche des tâches contenant un mot-clé spécifié dans le titre ou la description.

Réponse : Retourne un tableau de tâches correspondant au mot-clé.



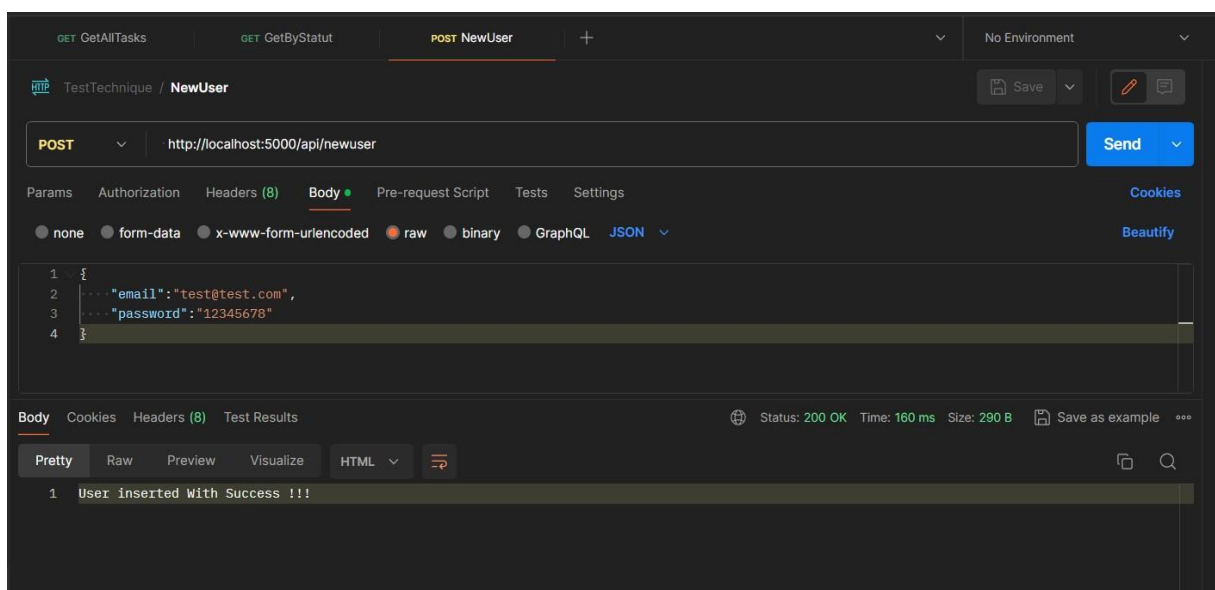
➤ EndPoint liés aux utilisateurs :

➤ POST /newuser

Description : Crée un nouvel utilisateur.

Corps de la requête : Nécessite un e-mail et un mot de passe.

Réponse : Retourne un message de réussite si l'utilisateur est créé ; sinon, retourne un message d'erreur.

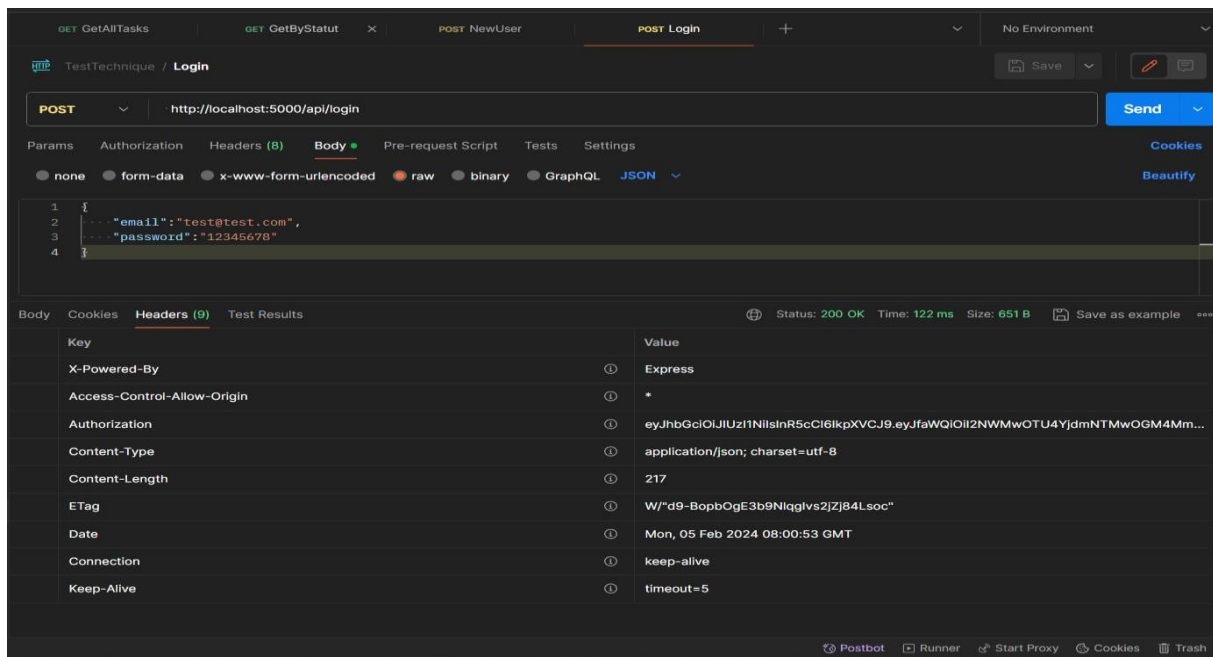


## ➤ POST /login

Description : Authentifie un utilisateur et génère un jeton JWT pour une autorisation ultérieure.

Corps de la requête : Nécessite un e-mail et un mot de passe pour l'authentification.

Réponse : Retourne un jeton JWT et l'ID de l'utilisateur si l'authentification est réussie ; sinon, retourne un message d'erreur.



## V. Interface Utilisateur :

### ➤ Page Login :



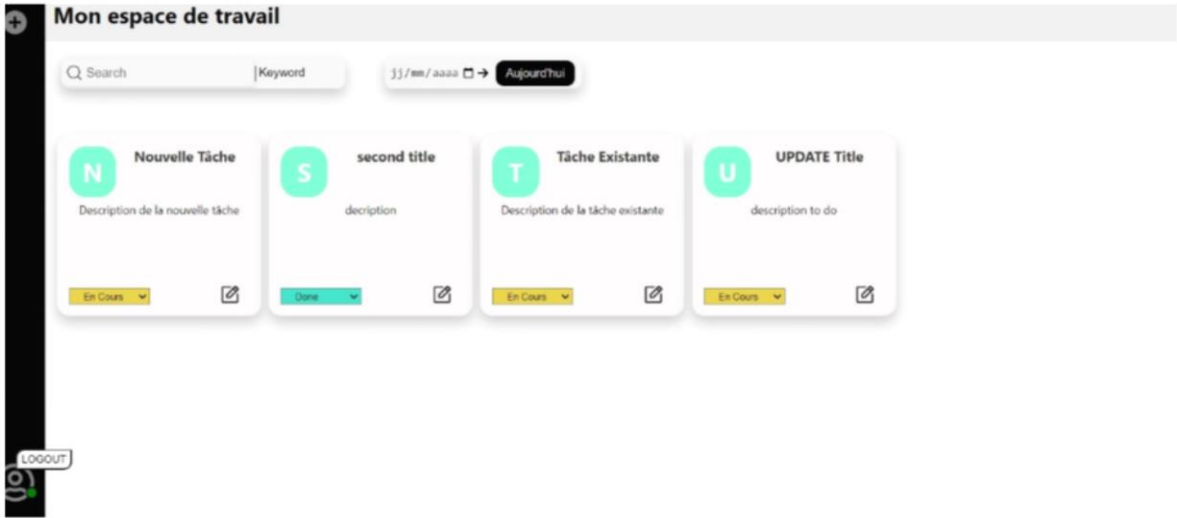


➤ Page Register :



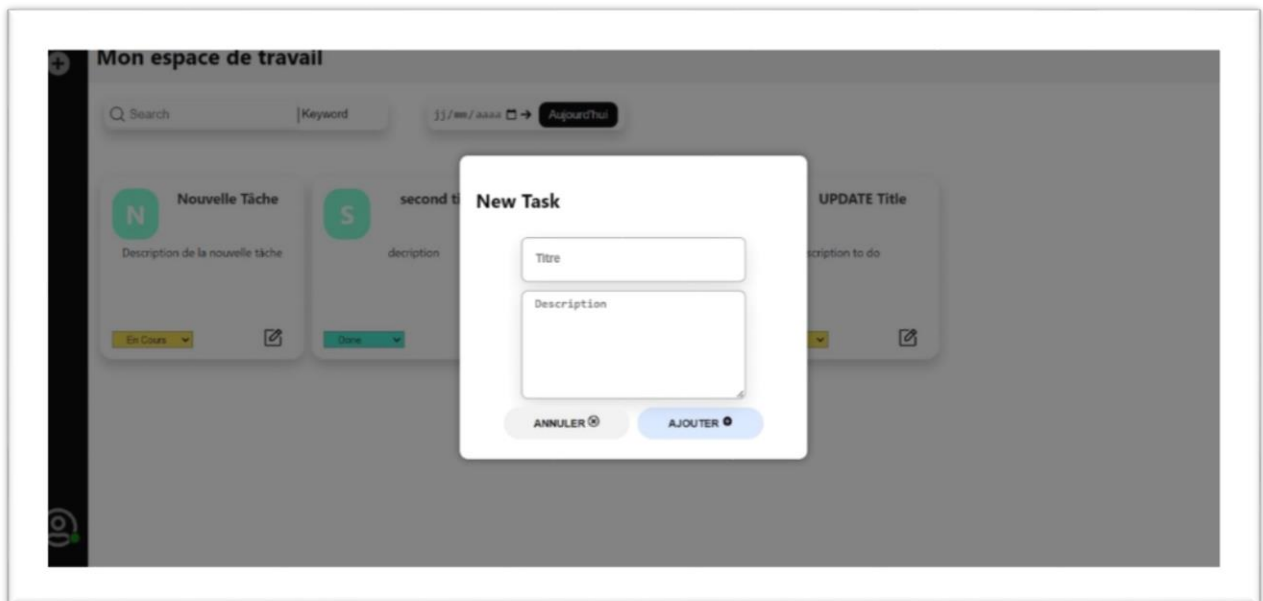
A dark-themed registration form titled "Register". It features two input fields for "Email" and "Password". Below the fields is a "REGISTER" button. At the bottom, there is a link "Already Have An Account" and a "SIGN IN" button.

➤ Page Espace De travail :



A dashboard titled "Mon espace de travail". It includes a search bar with "Search" and "Keyword" labels, a date selector showing "jj/mm/aaaa" and "Aujourd'hui", and a "LOGOUT" button in the bottom left. The main area displays four task cards: "Nouvelle Tâche", "second title", "Tâche Existante", and "UPDATE Title". Each card has a description, a status dropdown (e.g., "En Cours", "Done"), and an edit icon.

- Ajouter une tache :



- Modifier ou bien supprimer une tache :



## VI. Partie Base De Données :

User :

```
password: "$2b$10$m68NhKxcJN.PUTDvaeUthuoLj.Sjnlui/q2CB72pV3idV/z8y3rny"
__v: 0

_id: ObjectId('65bffe8e384bede9b810e598')
email: "mahmoud@gmail.com"
password: "$2b$10$3.IwX9P10u3SUKxR1jCdOd1EsJ0Zwq9/P9CLQ4jwwP5mf8jwRWva"
__v: 0

_id: ObjectId('65c0958b7f5388c82a6e5ae5')
email: "test@test.com"
password: "$2b$10$DwINLEfv2oZijnGWfhyqtuxPqSHfyWZYi/MQW6l0qzTRa9Q1jvIGC"
```

Les mots de passe sont hashés pour renforcer la sécurité

➤ Tache :

```
_id: ObjectId('65c0915286169a0b0545d73f')
id_user: "65bfd5bc6a862904e90b1655"
Titre: "titre by ali"
Description: "description by ali"
Statut: "en cours"
DateEcheance: 2024-01-05T07:42:10.439+00:00
__v: 0

_id: ObjectId('65c0916786169a0b0545d744')
id_user: "65bfd5bc6a862904e90b1655"
Titre: "second title by ali"
Description: "another description by ali"
Statut: "en cours"
DateEcheance: 2024-02-05T07:42:31.649+00:00
```

## VII. Partie Test :

➤ Exécution des Tests avec Jest :

```
PASS ./test.js
Task Routes
  Récupérez les tâches - Cas Limite
    ✓ devrait retourner un tableau vide lorsqu'il n'y a aucune tâche dans la base de données (448 ms)
  créer une nouvelle tâche - Cas Complexe
    ✓ devrait créer une nouvelle tâche et renvoyer un message de succès (190 ms)
    ✓ devrait gérer une demande invalide et renvoyer 400 avec un message d'erreur (12 ms)
    ✓ devrait gérer la création d'une tâche avec le même titre et renvoyer 400 avec un message d'erreur (98 ms)
Ajouter un Nouvel Utilisateur - Cas Limite
  ✓ devrait ajouter un nouvel utilisateur (181 ms)
Connexion d'Utilisateur - Cas Complexe
  ✓ devrait permettre la connexion d'un utilisateur avec des identifiants corrects (123 ms)
  ✓ devrait gérer un email invalide (45 ms)
  ✓ devrait gérer un mot de passe incorrect (112 ms)
  ✓ devrait gérer des identifiants manquants (46 ms)

Test Suites: 1 passed, 1 total
Tests: 9 passed, 9 total
Snapshots: 0 total
Time: 3.312 s, estimated 5 s
Ran all test suites matching /test.js/i.
Jest did not exit one second after the test run has completed.
```