

---

**Spécialité : Génie logiciel et informatique décisionnelle**

---

## **Gestion des résultats des matchs de football**

---

**Elaboré par :  
Mohamed Tijani Daoud  
Amina Turki**

**Validé par : Mr Jemal Ahmed**

*Année Universitaire : 2023/2024*

## 1 Introduction

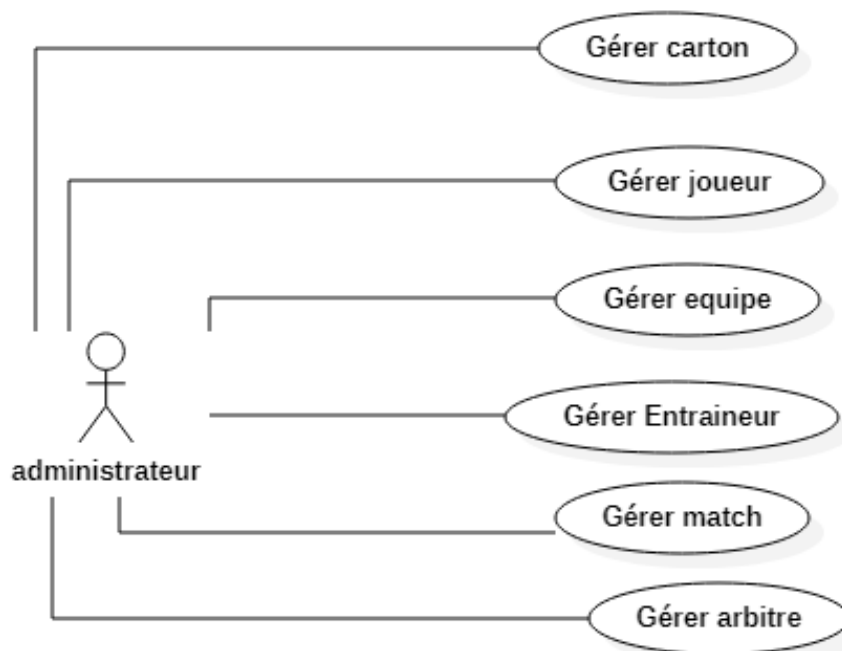
GestChamp est une application de gestion des résultats de matchs de football de la ligue 1 tunisienne, développée en utilisant JEE et Spring Cloud selon une architecture micro-service.

Les données sont persistées dans 3 dépôts de base MySQL. Pour faciliter le test et l'interaction avec les micro-services, Postman a été choisi comme technologie de front-end,

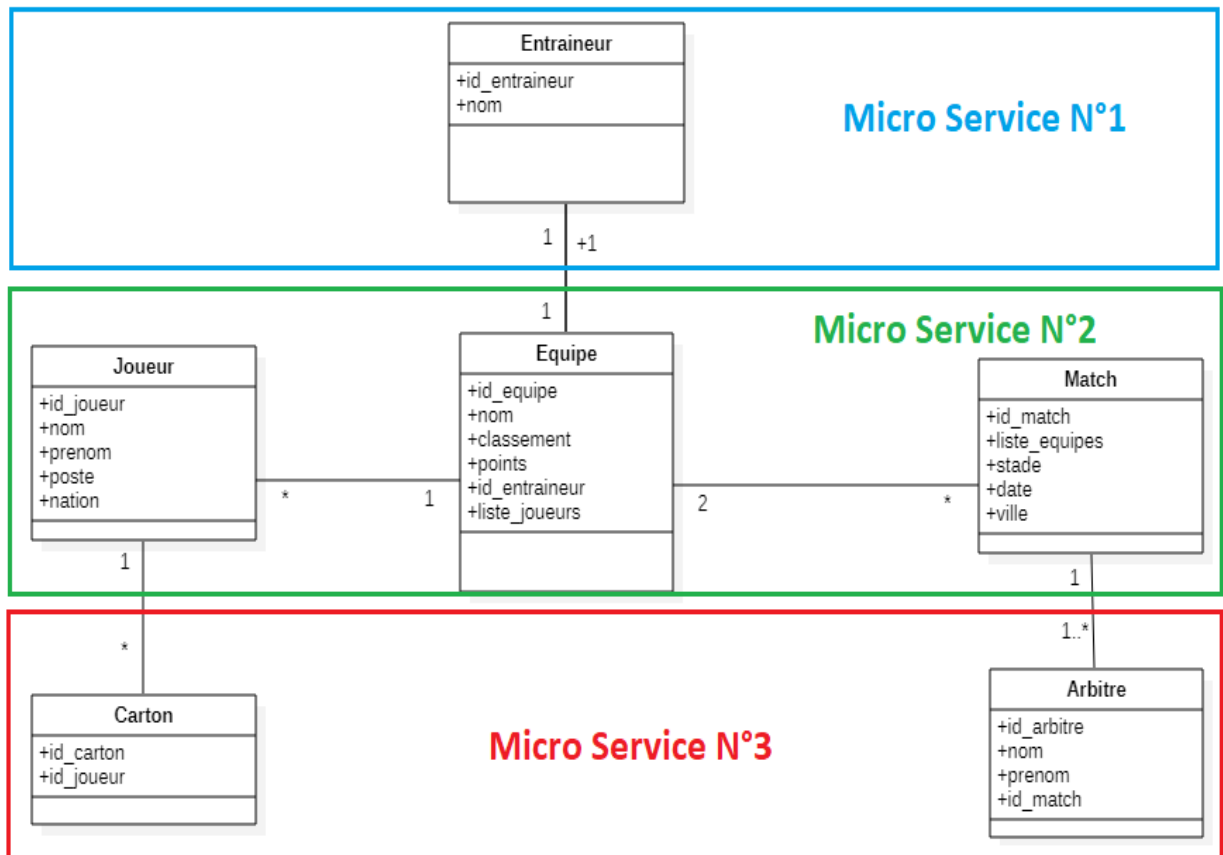
**Objectif du projet :** Créer une application web permettant aux utilisateurs de suivre les résultats des matchs de football. L'application permettra aux utilisateurs de :

- Voir les résultats des matchs d'une journée
- Voir le détail d'un match
- Voir le classement des équipes
- Voir la composition des équipes

## 2 Diagramme de cas d'utilisation



### 3 Diagramme de classe



### 4 Réalisation

#### ❖ Ajouter entraîneur :

Pour ajouter un entraîneur, l'administrateur peut utiliser l'URL suivante :

<http://localhost:5052/api/entraineur/create>. Il doit inclure les champs "nom", "prénom" et "âge" avec les informations appropriées lors de la requête.

```
{
  "nom": "montassar",
  "prenom": "Talbi",
  "age": 24
}
```

POST `http://localhost:5052/api/entraîneur/create` Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

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

```
1 {
2   "nom": "amina",
3   "prenom": "turki",
4   "age": 26
5 }
```

Body Cookies Headers (5) Test Results Status: 200 OK Time: 61 ms Size: 212 B Save Response

Pretty Raw Preview Visualize JSON ⌵ 🔍

```
1 {
2   "id": 8,
3   "nom": "amina",
4   "prenom": "turki",
5   "age": 26
6 }
```

### ❖ Modifier entraîneur :

L'administrateur peut mettre à jour un entraîneur en utilisant l'URL spécifiée

<http://localhost:5052/api/entraîneur/update/{id}>. En effet, cette URL permet la mise à jour d'un entraîneur dans le système en utilisant l'identifiant spécifique de l'entraîneur à modifier.

```
{
  "nom": "montassar",
  "prenom": "Talbi",
  "age": 17
}
```

PUT `http://localhost:5052/api/entraîneur/update/8` Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

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

```
1 {
2   "nom": "amina",
3   "prenom": "turki",
4   "age": 25
5 }
```

Body Cookies Headers (5) Test Results Status: 200 OK Time: 43 ms Size: 212 B Save Response

Pretty Raw Preview Visualize JSON ⌵ 🔍

```
1 {
2   "id": 8,
3   "nom": "amina",
4   "prenom": "turki",
5   "age": 25
6 }
```

## ❖ Afficher les coordonnées d'un entraîneur :

Pour afficher les coordonnées d'un entraîneur, l'administrateur peut utiliser l'URL suivante : <http://localhost:5052/api/entraîneur/getById/{id}>. En utilisant cette URL, les informations spécifiques de l'entraîneur avec l'identifiant 9 seront récupérées et affichées.

GET <http://localhost:5052/api/entraîneur/getById/8> Send

Params Authorization Headers (6) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL Text

1

Body Cookies Headers (5) Test Results Status: 200 OK Time: 20 ms Size: 212 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 8,
3   "nom": "amina",
4   "prenom": "turki",
5   "age": 25
6 }
```

## ❖ Supprimer un entraîneur :

Pour supprimer un entraîneur, l'administrateur peut utiliser l'URL suivante : <http://localhost:5052/api/entraîneur/deleteById/{id}>. En utilisant cette URL, l'entraîneur ayant l'identifiant 9 sera supprimé du système.

DELETE <http://localhost:5052/api/entraîneur/deleteById/8> Send

Params Authorization Headers (6) **Body** Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (5) Test Results Status: 200 OK Time: 497 ms Size: 197 B Save Response

Pretty Raw Preview Visualize Text

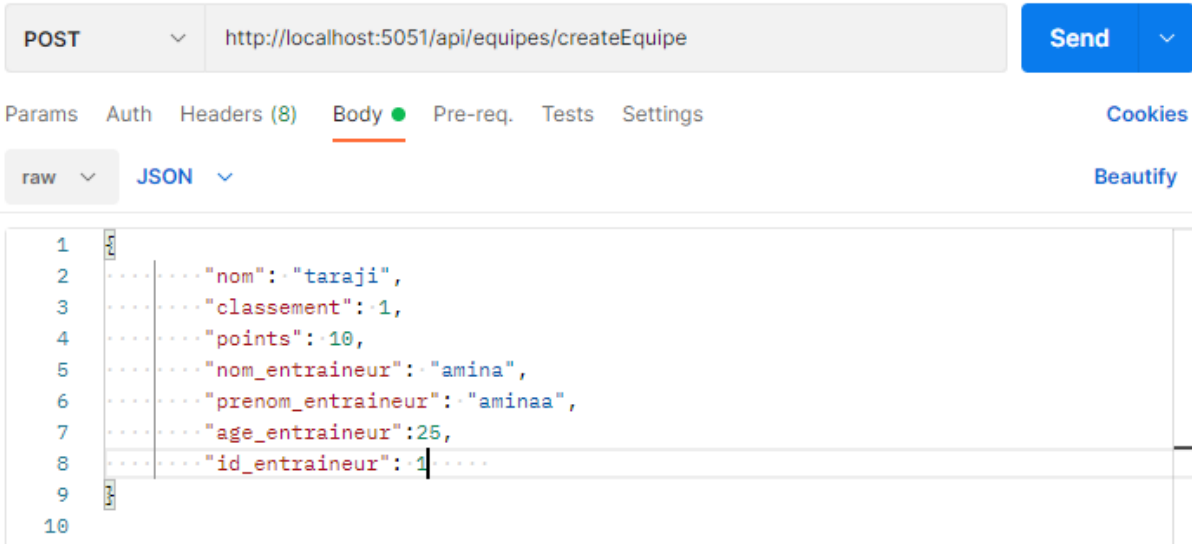
```
1 Entraîneur supprimé avec succès
```

## ❖ Ajouter équipe :

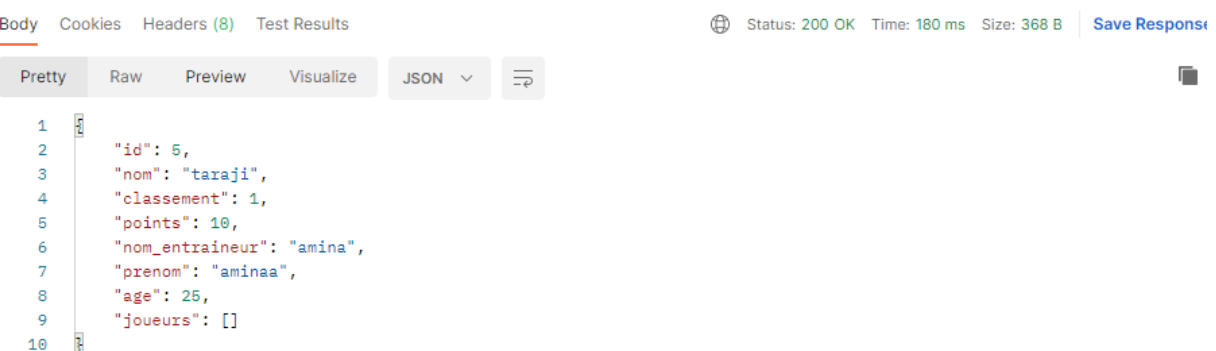
Pour ajouter une équipe, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/equipes/createEquipe>.

En incluant les champs nécessaires tels que "nom de l'équipe", "classement" et d'autres informations appropriées dans la requête, une équipe avec son entraîneur seront ajoutés dans les deux micro services du système.

```
{
  "nom": "est",
  "classement": 1,
  "points": 7,
  "nom_entraineur": "amina",
  "prenom_entraineur": "aminaa",
  "age_entraineur": 25,
  "id_entraineur": 1
}
```



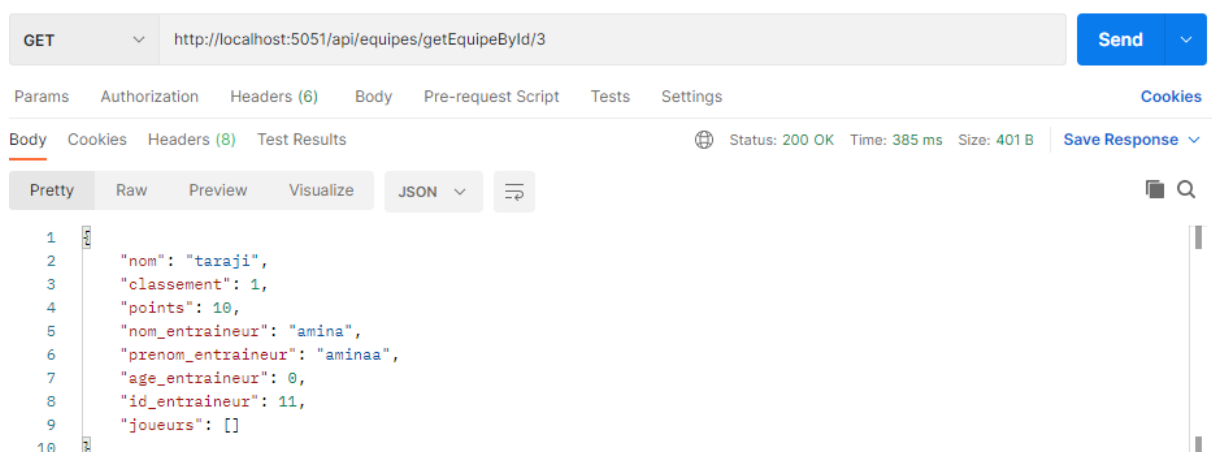
## Résultat



```
{
  "id": 5,
  "nom": "taraji",
  "classement": 1,
  "points": 10,
  "nom_entraineur": "amina",
  "prenom": "aminaa",
  "age": 25,
  "joueurs": []
}
```

## ❖ Afficher les coordonnées d'une équipe :

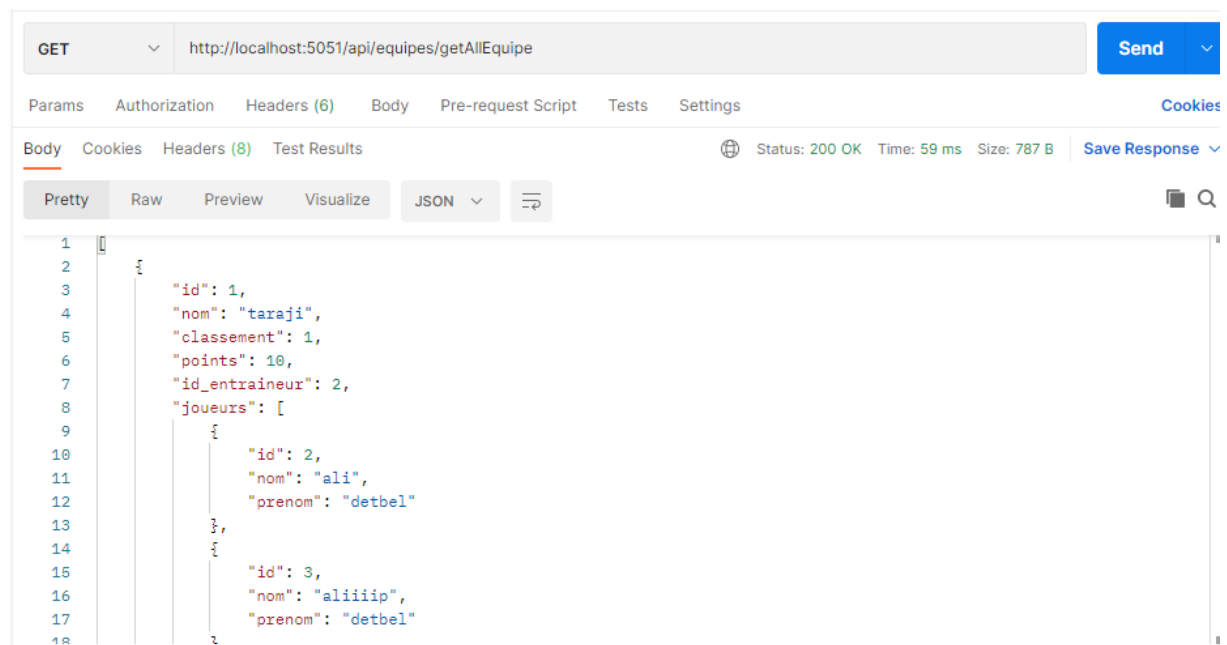
Pour afficher les coordonnées d'une équipe, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/equipes/getEquipeById/{id}>. Cette URL permet de récupérer les informations spécifiques d'une équipe avec l'entraîneur correspondant avec l'identifiant 3 dans le système.



```
GET http://localhost:5051/api/equipes/getEquipeById/3
Status: 200 OK Time: 385 ms Size: 401 B
Body: {"nom": "taraji", "classement": 1, "points": 10, "nom_entraineur": "amina", "prenom_entraineur": "aminaa", "age_entraineur": 0, "id_entraineur": 11, "joueurs": []}
```

## ❖ Afficher la liste d'une équipe :

Pour afficher les coordonnées d'un entraîneur, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/equipes/getAllEquipe>. Cette URL permet de récupérer les informations des équipes.

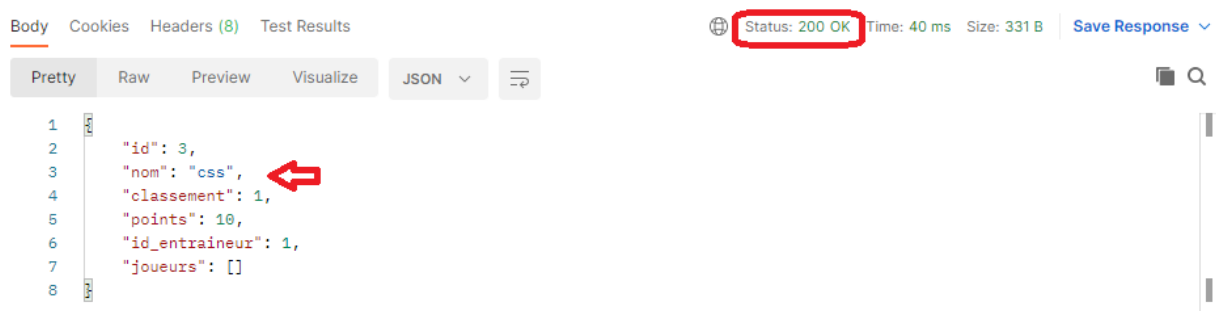
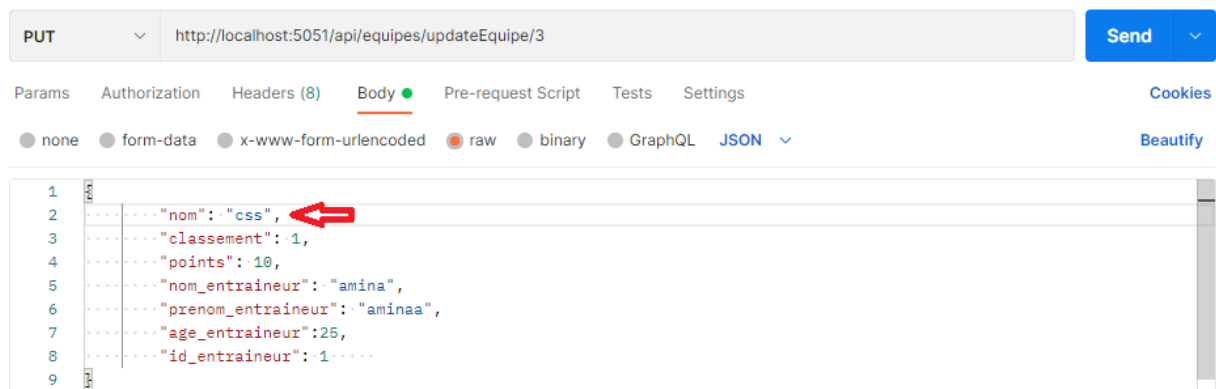


```
GET http://localhost:5051/api/equipes/getAllEquipe
Status: 200 OK Time: 59 ms Size: 787 B
Body: [{"id": 1, "nom": "taraji", "classement": 1, "points": 10, "id_entraineur": 2, "joueurs": [{"id": 2, "nom": "ali", "prenom": "detbel"}]}, {"id": 3, "nom": "aliiip", "prenom": "detbel"}]
```

## ❖ Modifier les coordonnées d'une équipe :

L'administrateur peut mettre à jour les coordonnées d'une équipe en utilisant l'URL spécifiée <http://localhost:5051/api/equipes/updateEquipe/{id}>. En effet, cette URL permet la mise à jour d'une équipe dans le système.

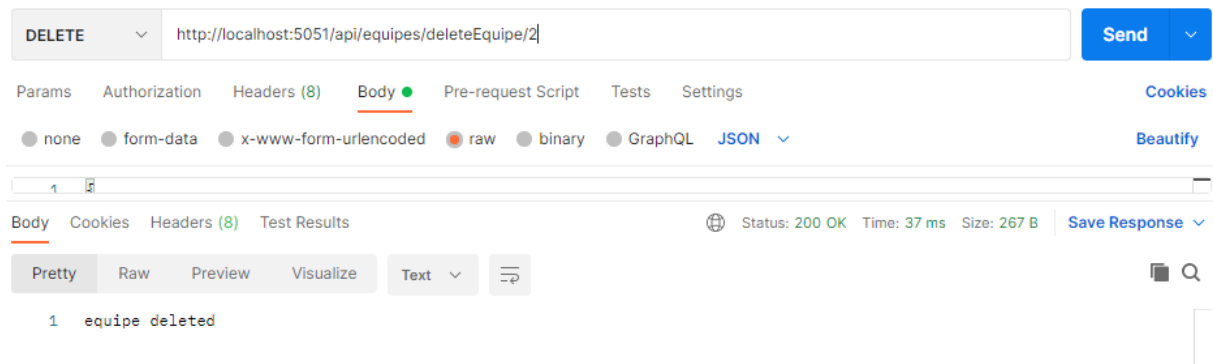
```
{
  "nom": "Est",
  "classement": 7,
  "points": 10,
  "nom_entraineur": "amina",
  "prenom_entraineur": "aminaa",
  "age_entraineur": 25,
  "id_entraineur": 1
}
```



## ❖ Supprimer équipe

Pour supprimer une équipe, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/equipes/deleteEquipe/{id}>. En utilisant cette URL, l'équipe identifiée par le numéro 2 sera supprimée du système.



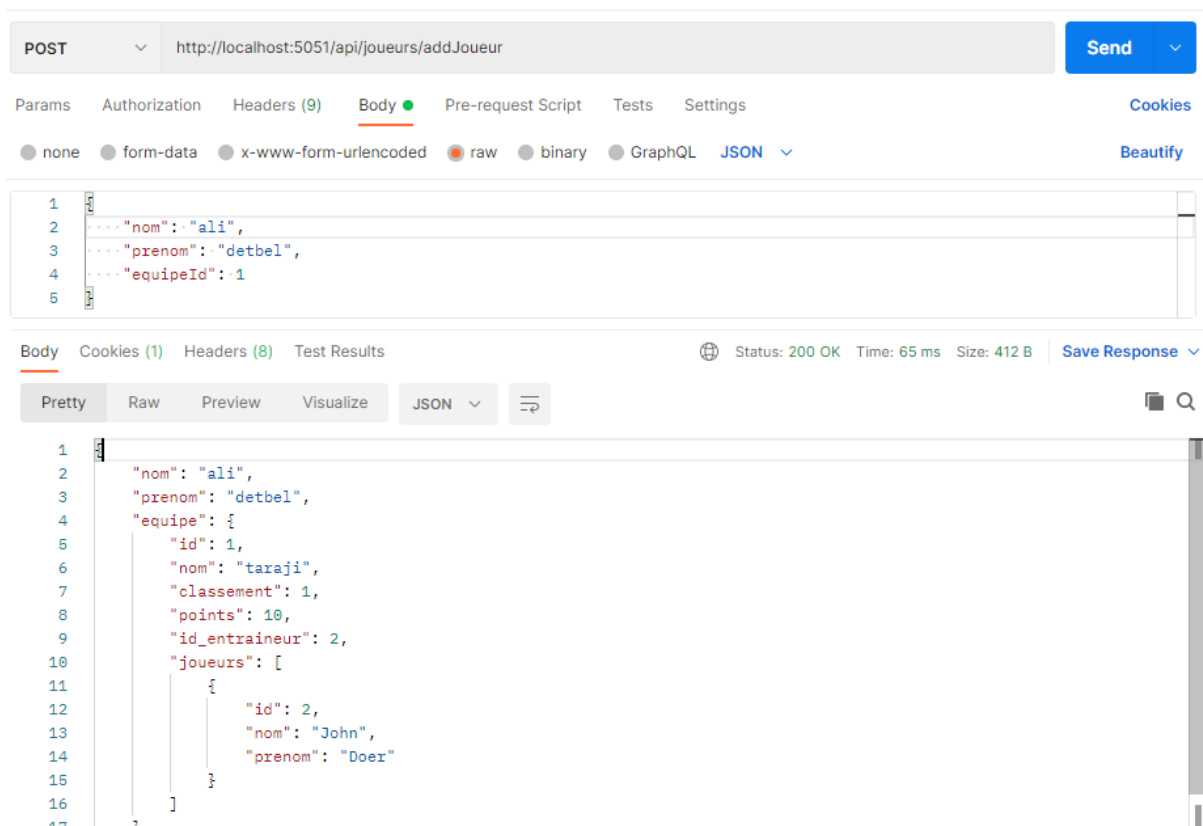








## ❖ Ajouter joueur :

Pour ajouter un joueur, l'utilisateur peut utiliser l'URL suivante :

<http://localhost:5051/api/joueurs/addJoueur>. En incluant les champs nécessaires tels que "nom", "prénom" et d'autres informations appropriées dans la requête, le joueur sera ajouté au système.

```
{
  "nom": "Ali",
  "prenom": "Maaloul",
  "equipeId": 7
}
```



<div>← T →</div>					id	nom	prenom	id_equipe		
<input type="checkbox"/>		Éditer		Copier		Supprimer	2	John	Doer	1
<input type="checkbox"/>		Éditer		Copier		Supprimer	3	ali	detbel	1

### ❖ Afficher les coordonnées d'un joueur :

Pour afficher les coordonnées d'un entraîneur, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/joueurs/getJoueurById/{id}>. Cette URL permet de récupérer les informations spécifiques du joueur avec l'identifiant 3 dans le système.

The screenshot shows a REST client interface with a GET request to `http://localhost:5051/api/joueurs/getJoueurById/3`. The response status is 200 OK. The response body is displayed in JSON format:

```

{
  "id": 3,
  "nom": "ali",
  "prenom": "detbel"
}

```

### ❖ Modifier les coordonnées d'un joueur :

L'administrateur peut mettre à jour un joueur en utilisant l'URL spécifiée <http://localhost:5051/api/joueurs/updateJoueur/{id}>. En effet, cette URL permet la mise à jour d'un entraîneur dans le système en utilisant l'identifiant spécifique de l'entraîneur à modifier.

```

{
  "nom": "Ali",
  "prenom": "Maaloul",
  "equipeId": 7
}

```

PUT http://localhost:5051/api/joueurs/updateJoueur/3 Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```

1  {
2    "nom": "aliiii",
3    "prenom": "detbel",
4    "equipeId": 1
5  }

```

Body Cookies (1) Headers (8) Test Results Status: 200 OK Time: 23 ms Size: 294 B Save Response

Pretty Raw Preview Visualize JSON

```

1  {
2    "id": 3,
3    "nom": "aliiii",
4    "prenom": "detbel"
5  }

```

## ❖ Supprimer joueur

Pour supprimer un joueur, l'administrateur peut utiliser l'URL suivante :

<http://localhost:5051/api/joueurs/deleteJoueur/{id}>. En utilisant cette URL, le joueur identifié par le numéro 3 sera supprimé du système.

DELETE http://localhost:5051/api/joueurs/deleteJoueur/3 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (8) Test Results Status: 200 OK Time: 49 ms Size: 287 B Save Response

Pretty Raw Preview Visualize Text

```

1  Joueur with ID 1 has been deleted.

```

## ❖ Ajouter match :

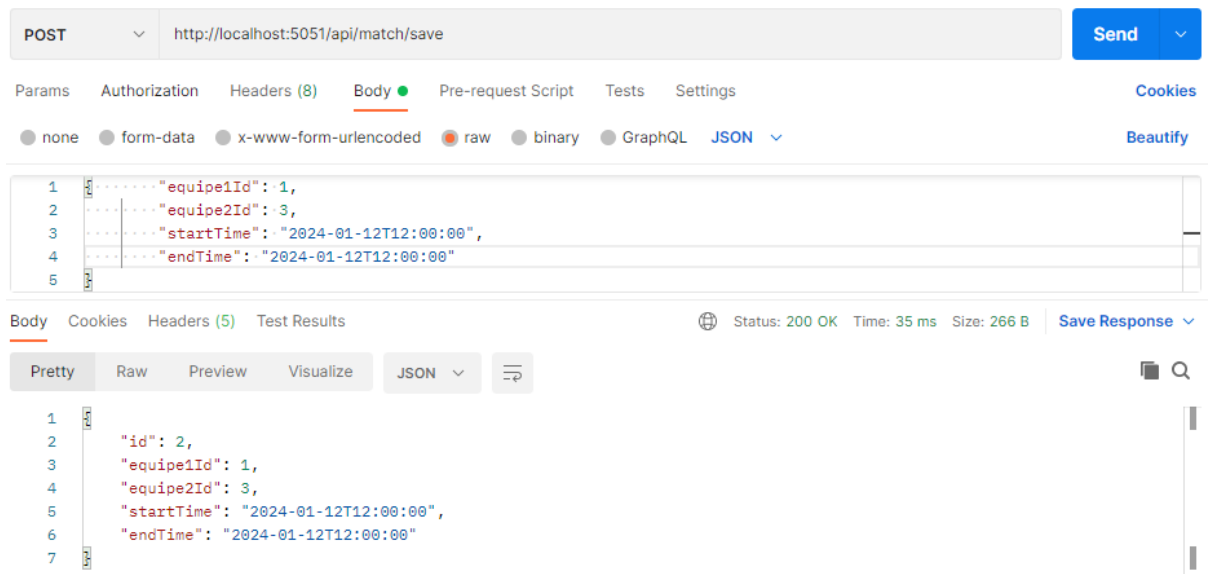
Pour ajouter un match, l'utilisateur peut utiliser l'URL suivante :

<http://localhost:5051/api/match/save>. En incluant les champs nécessaires tels que equipe1, equipe2 et d'autres informations appropriées dans la requête, le joueur sera ajouté au système.

```

{
  "equipe1Id": 1,
  "equipe2Id": 2,
  "startTime": "2024-01-12T12:00:00",
  "endTime": "2024-01-12T12:00:00"
}

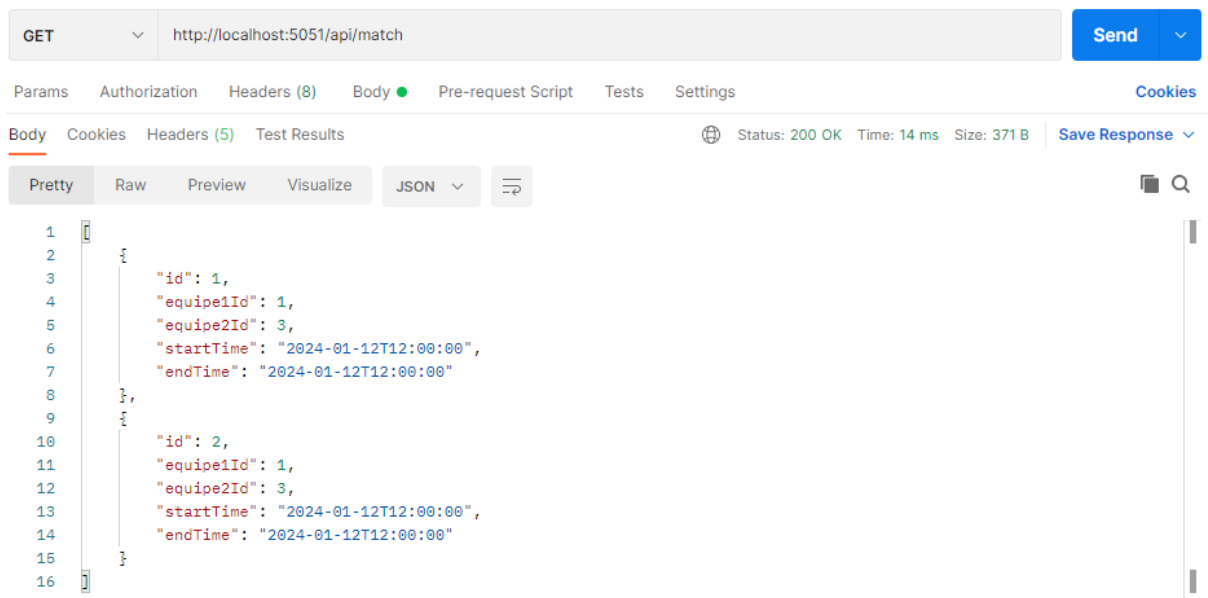
```



### ❖ Afficher la liste des matchs:

Pour afficher les coordonnées des matchs, l'administrateur peut utiliser l'URL suivante :

<http://localhost:5051/api/match>. Cette URL permet de récupérer les informations des matchs.



### ❖ Modifier les coordonner d'un match :

L'administrateur peut mettre à jour un match en utilisant l'URL spécifiée

<http://localhost:5051/api/match/update/{id}>. En effet, cette URL permet la mise à jour d'un match dans le système en utilisant l'identifiant spécifique du match à modifier.

```

{
  "equipe1Id": 1,
  "equipe2Id": 2,
  "startTime": "2024-01-12T12:00:00",
  "endTime": "2024-01-12T19:00:00"
}

```

PUT <http://localhost:5051/api/match/update/2> Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 {
2   "equipe1Id": 1,
3   "equipe2Id": 3,
4   "startTime": "2024-01-12T12:00:00",
5   "endTime": "2024-01-15T12:00:00"
6 }
```

Body Cookies Headers (5) Test Results Status: 200 OK Time: 19 ms Size: 266 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 3,
3   "equipe1Id": 1,
4   "equipe2Id": 3,
5   "startTime": "2024-01-12T12:00:00",
6   "endTime": "2024-01-15T12:00:00"
7 }
```

### ❖ Supprimer match

Pour supprimer un match, l'administrateur peut utiliser l'URL suivante :

<http://localhost:5051/api/match/delete/{id}>. En utilisant cette URL, le match identifiée par le numéro 2 sera supprimé du système.

DELETE <http://localhost:5051/api/match/delete/2> Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
1 {
2   "equipe1Id": 1,
3   "equipe2Id": 3,
4   "startTime": "2024-01-12T12:00:00",
5   "endTime": "2024-01-15T12:00:00"
6 }
```

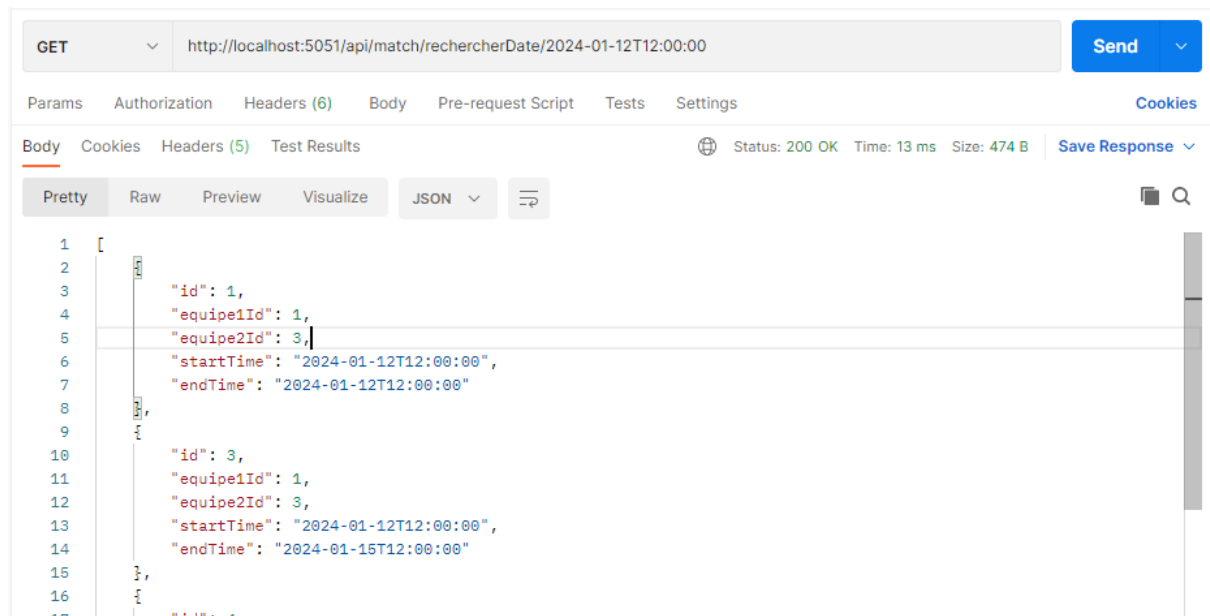
Body Cookies Headers (5) Test Results Status: 200 OK Time: 22 ms Size: 168 B Save Response

Pretty Raw Preview Visualize JSON

```
1 true
```

### ❖ Recherche de match par date :

Pour afficher les coordonnées des matchs demandés par date, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/match/rechercherDate/2024-01-12T12:00:00>. Cette URL permet de récupérer les informations des matchs programmés à la date spécifiée.

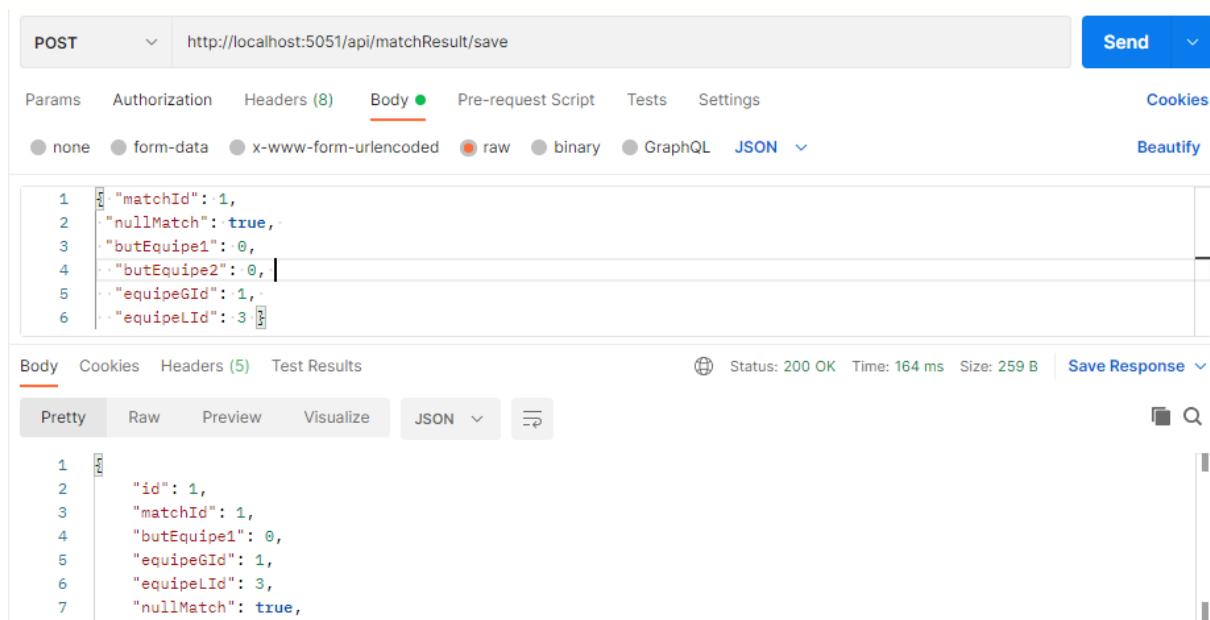


### ❖ Ajouter résultat du match :

Pour ajouter un match, l'utilisateur peut utiliser l'URL suivante :

<http://localhost:5051/api/matchResult/save>. En incluant les champs nécessaires tels que equipe1, equipe2 et d'autres informations appropriées dans la requête, le joueur sera ajouté au système.

```
{ "matchId": 1, "nullMatch": true, "butEquipe1": 0, "butEquipe2": 0,
  "equipeGId": 1, "equipeLId": 3 }
```



### ❖ Afficher les coordonnées d'un résultat :

Pour afficher les coordonnées d'un résultat, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/matchResult/byMatch/{id}>. Cette URL permet de récupérer les informations spécifiques d'un résultat avec l'entraîneur correspondant avec l'identifiant 3 dans le système.

GET <http://localhost:5051/api/matchResult/byMatch/1> Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "matchId": 1,
3   "nullMatch": true,
4   "butEquipe1": 0,
5   "butEquipe2": 0,
6   "equipeGId": 1,
7   "equipeLId": 3
8 }
```

Body Cookies Headers (5) Test Results Status: 200 OK Time: 37 ms Size: 259 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "matchId": 1,
4   "butEquipe1": 0,
5   "equipeGId": 1,
6   "equipeLId": 3,
7   "nullMatch": true,
8 }
```

### ❖ Modifier les coordonnées d'un résultat :

L'administrateur peut mettre à jour un résultat en utilisant l'URL spécifiée <http://localhost:5051/api/matchResult/update/{id}>. En effet, cette URL permet la mise à jour d'un résultat dans le système en utilisant l'identifiant spécifique du résultat à modifier.

PUT <http://localhost:5051/api/matchResult/update/1> Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "matchId": 1,
3   "nullMatch": true,
4   "butEquipe1": 0,
5   "butEquipe2": 0,
6   "equipeGId": 1,
7   "equipeLId": 4
8 }
```

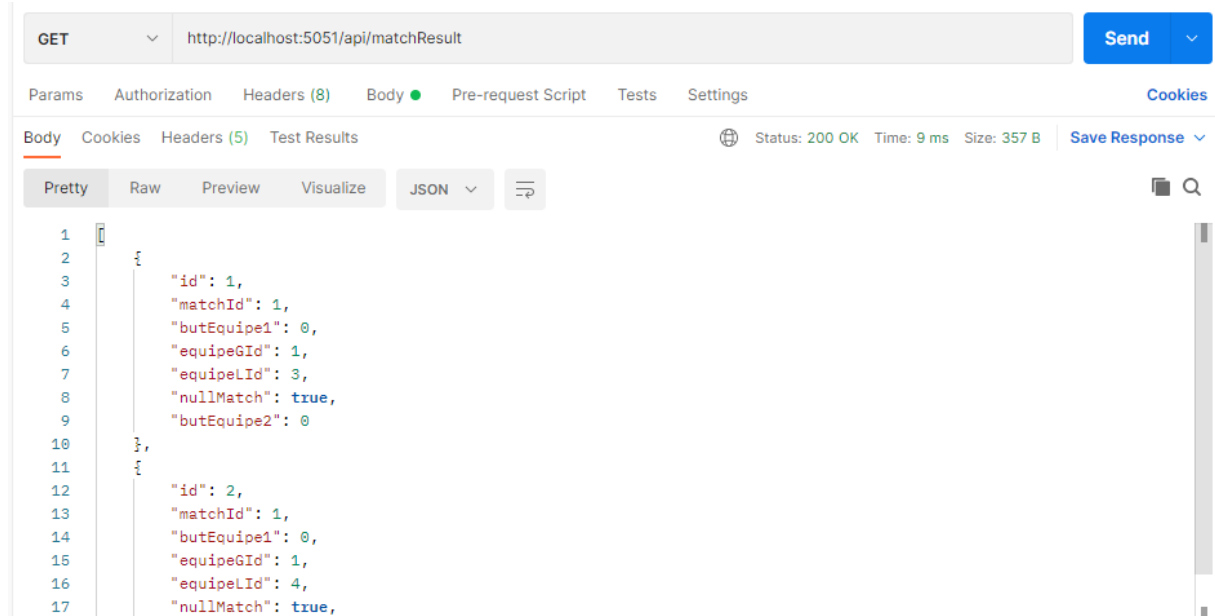
Body Cookies Headers (5) Test Results Status: 200 OK Time: 120 ms Size: 259 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 2,
3   "matchId": 1,
4   "butEquipe1": 0,
5   "equipeGId": 1,
6   "equipeLId": 4,
7   "nullMatch": true,
8 }
```

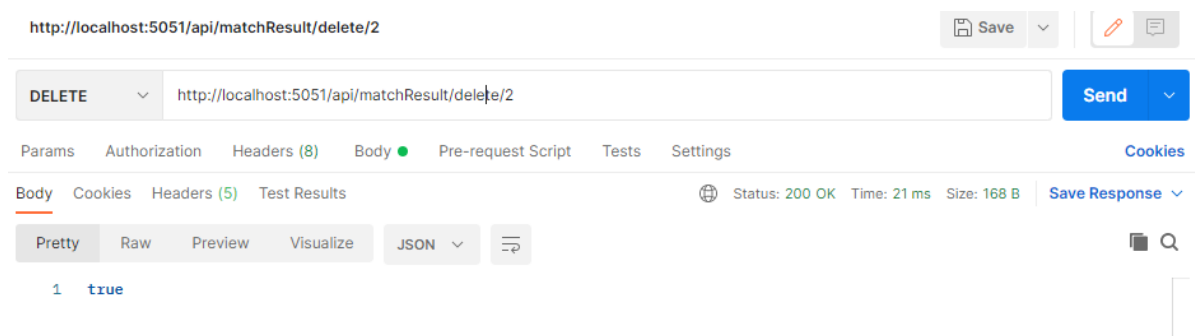
### ❖ Afficher la liste des résultats:

Pour afficher les coordonnées des résultats, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/matchResult>. Cette URL permet de récupérer les informations des résultats.



### ❖ Supprimer résultat

Pour supprimer un résultat, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/matchResult/delete/{id}>. En utilisant cette URL, le résultat identifié par le numéro 2 sera supprimé du système.





## ❖ Afficher la liste de classement des équipes:

Pour afficher la liste de classement des équipes par ordre croissant, l'administrateur peut utiliser l'URL suivante : <http://localhost:5051/api/equipes/order> .

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

- Method:** GET
- URL:** http://localhost:5051/api/equipes/order
- Status:** 200 OK
- Time:** 362 ms
- Size:** 784 B
- Response Format:** JSON

The JSON response body is as follows:

```
1 {
2   {
3     "id": 1,
4     "nom": "taraji",
5     "classement": 1,
6     "points": 11,
7     "id_entraineur": 2,
8     "joueurs": [
9       {
10        "id": 3,
11        "nom": "aliiiip",
12        "prenom": "detbel"
13      },
14      {
15        "id": 4,
16        "nom": "ali",
17        "prenom": "detbel"
18      }
19    ]
20  }
21 }
```