



Documentation API

Introduction [↗](#)

Bienvenue dans la documentation de l'API Omnimatch. Notre API fournit des fonctionnalités avancées pour requêter entre notre base de donnée et l'application Web.

Accès à l'API [↗](#)

Notre Api requête la base de donnée de l'application web. Il faut donc avoir accès à la base de donnée pour utiliser l'API.

La configuration d'un **.env** est **obligatoire pour le fonctionnement de l'API**.

Pour se faire il faut créer un fichier .env et copier coller le code ci dessous:

```
1 DB_HOST=
2 DB_PORT=
3 DB_NAME=
4 DB_USER=
5 DB_PASSWORD=
6 MY_SECRET_KEY=
7 MAIL_KEY=
```

Remplissez le code en fonction de vos paramètres.

Exemple de requêtes [↗](#)

```
1 router.get('/', (req, res) => {
2   db.query('SELECT * FROM town', (err, results) => {
3     if (err) {
4       console.error('Erreur lors de la récupération des données : ' + err.message);
5       res.status(500).json({ err: 'Erreur serveur' });
6       return;
7     }
8
9     res.status(200).json(results);
10  });
11 });
12
13 router.get('/:input', (req, res) => {
14   const input = req.params.input;
15   // Ici, vous intégreriez la logique pour rechercher la ville dans votre base de données
16   // Pour l'exemple, je vais juste renvoyer le mot "input"
17   res.json({ message: `Recherche effectuée pour : ${input}` });
18 });
```

Paramètre de requêtes [↗](#)

Les paramètres des requêtes sont à mettre dans le fichier index.js.

Il est fortement conseiller de faire une arborescence comme ci-dessous pour éviter tous problèmes !

RacinesDuDossier\NomduProjet\api

RacinesDuDossier\NomduProjet\front

```

1 origin: '*',
2   methods: 'GET,PUT,POST,DELETE',
3   allowedHeaders: 'Content-Type, Authorization',

```

🌐 Réponses 🔗

Email de l'utilisateur reçu : { club_mail: "", password: "" }

Utilisateur trouvé : RowDataPacket {

```

  id_user: 58,
  id_club: 57,
  club_name: 'Magny-Football',
  club_mail: 'Magny.football@orange.fr',
  password_hash: '$2a$10$6JXWyo9I5akkOu/Lxv38oufzfzqffegzegztrgdfgegegegeggegDrftXz6WyoYYYwtXCKqoxPf5t2DHnaq'
}

```

payload : RowDataPacket {

```

  id_user: 58,
  id_club: 57,
  club_name: 'Magny-Football',
  club_mail: 'Magny.football@orange.fr',
  password_hash: '$2a$10$6JXTGEZGGHZRTHBEERBWyo9I5akkOu/Lxv38ouDrftXz6WyoYYYwtXCKqoxPf5t2DHnaq'
}

```

Token généré :

```

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6OTEsImklX2NsdWliOiJk3LCJ1c2VybmFtZSI6Ik1hZ2Z5LUZvb3RiYWxsliwiOiJWFpbAGCI6ImEBEBEBZEBEVEVVAEBAEVAEFXZXXXZGBoaWwyODExNkNbnWFpbC5jb20iLCJpYXQiOiJlE3MDUzMzIzMzYslmV4cCI6MTcwAGEGANTMzNTkzG.O7lpJdTT65HT-GAGAGAhfEY7calwhos7hsLpCXDIDkhiGEaTo

```

Token décodé : {

```

  header: { alg: 'HS256', typ: 'JWT' },
  payload: {
    id: 58,
    id_club: 57,
    username: 'Magny-Football',
    mail: 'Magny.football@orange.fr',
    iat: 170533233FE6,
    exp: 170533593FE6
  },
}

```

🔧 Gestion des erreurs 🔗

Lors de la connexion à la base de donnée, des erreurs apparaîtront si vous n'avez pas tous les prérequis (voir [🎯 Plan de projet](#)).

Voici un exemple de gestion d'erreurs :

```

1 db.connect((err) => {
2   if (err) {
3     console.error('Erreur de connexion à la base de données : ' + err.message);
4   } else {
5     console.log('Connexion à la base de données établie');
6   }
7 });

```

Ici un exemple de class Controller pour gérer les erreurs :

```
1 class ClubController {
2   static addClub(req, res) {
3     console.log("on passe dans le controller");
4     try {
5       model.createClubs(db, req, (err, results) => {
6         if (err) {
7           res.status(500).json({ err: "Erreur serveur" });
8         } else {
9           res.status(200).json({ message: "Le club a bien été ajouté !" });
10          const mail_key = process.env.MAIL_KEY;
11          const transporter = nodemailer.createTransport({
12            service: "gmail",
13            auth: {
14              user: "Magny.football@orange.fr",
15              pass: mail_key,
16            },
17          });

```

Authentification et sécurité

Pour que l'API fonctionne, il faut avoir accès à un compte qui est fournis par nos soins.

Une fois le compte fournis, vous pourrez effectuer vos requêtes une fois le Token(jeton) générer.

Lors du lancement de l'application un message json sera afficher dans le terminal et génèrera un token(jeton) qui vous sera assigner.