

# API Mongoose

## TD : API MongoDB avec Mongoose

### Objectifs

- Créer une API RESTful avec Express.js
- Configurer une base de données MongoDB avec Mongoose
- Définir un modèle Mongoose pour un profil utilisateur
- Implémenter des routes CRUD pour gérer les profils

### Étapes

#### 1. Initialisation du projet

- Créez un nouveau projet Node.js
- Installez les dépendances nécessaires : `express` , `mongoose` , `dotenv` ou tout autre module utile
- Implementez un serveur mongodb docker

```
version: '3.8'
services:
  mongo:
    image: mongo:latest
    container_name: mongo_container
    ports:
      - "27017:27017"
    volumes:
      - mongo_data:/data/db
    environment:
      MONGO_INITDB_ROOT_USERNAME: root
      MONGO_INITDB_ROOT_PASSWORD: example
volumes:
  mongo_data:
```

## 2. Configuration de l'environnement

- Créez un fichier `.env` pour stocker les variables d'environnement (URI MongoDB)
- Configurez le fichier `mongodb.js` pour se connecter à MongoDB

## 3. Définition du modèle Mongoose

- Créez un fichier `api/profiles/model.js`
- Définissez le schéma Mongoose pour un profil utilisateur avec les champs suivants :
  - `name` : String, requis
  - `email` : String, requis, unique
  - `experience` : Array d'objets (`_id`, titre, entreprise, dates, description)
  - `skills` : Array de Strings
  - `information` : Objet (bio, localisation, site web)

## 4. Création des routes

- Créez un fichier `api/profiles/index.js`
- Implémentez les routes CRUD pour gérer les profils :
  - `GET /profiles` : Récupérer tous les profils
  - `GET /profiles/:id` : Récupérer un profil par ID
  - `POST /profiles` : Créer un nouveau profil (uniquement nom et email)
  - `PUT /profiles/:id` : Mettre à jour un profil par ID (uniquement nom et email)
  - `DELETE /profiles/:id` : Supprimer un profil par ID (soft-delete)
  - `POST /profiles/:id/experience` : Ajouter une expérience à un profil
  - `DELETE /profiles/:id/experience/:exp` : Supprimer une expérience d'un profil par ID d'expérience
  - `POST /profiles/:id/skills` : Ajouter une compétence à un profil
  - `DELETE /profiles/:id/skills/:skill` : Supprimer une compétence d'un profil
  - `PUT /profiles/:id/information` : Mettre à jour les informations d'un profil

## 5. Implémentation des méthodes du contrôleur

- Créez un fichier `api/profiles/controller.js`
- Implémentez les méthodes pour chaque route définie dans `api/profiles/index.js`

## 6. Intégration des routes dans l'application

- Créez un fichier `routes.js` pour centraliser les routes
- Mettez à jour le fichier `index.js` pour utiliser les routes définies

## 7. Bonus : Recherche et filtres

- Implémentez des fonctionnalités de recherche et de filtres pour les profils (par compétences, localisation, etc.) dans la route `GET /profiles`

## Ressources

- Express.js : <https://expressjs.com/>
- Mongoose : <https://mongoosejs.com/>
- MongoDB : <https://www.mongodb.com/>
- Postman : <https://www.postman.com/>