

# API BileMo - Documentation Utilisateur

## Introduction

L'API **BileMo** permet de gérer des données liées à des **clients** et d'accéder à des **téléphones mobiles** (non inclus dans cette portion de l'API, mais prévu par la description générale). L'authentification se fait via **JWT (JSON Web Token)**.

---

## Authentification

### POST /api/login\_check

Permet de récupérer un **token JWT** en fournissant des identifiants valides.

#### Requête :

```
{
  "username": "email@exemple.com",
  "password": "motdepasse"
}
```

#### Réponse :

```
{
  "token": "eyJ0eXAiOiJKV1QiLCJh..."
}
```

Ce token devra ensuite être utilisé dans les en-têtes des requêtes suivantes :

Authorization: Bearer {token}

# Clients

## GET /api/customers

Récupère une **liste paginée** de tous les clients.

- **Paramètres de requête :**
  - **page** (optionnel, entier) : numéro de la page à récupérer.
  - **limit** (optionnel, entier) : nombre de clients par page.

**Réponse :**

```
[
  {
    "id": 1,
    "name": "Jean Dupont",
    "email": "jean@exemple.com",
    ...
  },
  ...
]
```

## POST /api/customers

Crée un **nouveau client**.

**Requête :**

```
{
  "name": "Marie Curie",
  "email": "marie@exemple.com",
  "password": "motdepasse"
}
```

**Réponse :**

- Code HTTP : **201 Created**

```
{
  "id": 5,
  "name": "Marie Curie",
  "email": "marie@exemple.com"
}
```

## GET /api/customers/{id}

Récupère les **informations d'un client spécifique**.

- **Paramètre d'URL :**
  - **id** (obligatoire) : ID numérique du client.

**Réponse :**

```
{
  "id": 3,
  "name": "Albert Einstein",
  "email": "albert@exemple.com"
}
```

## DELETE /api/customers/{id}

Supprime un **client existant**.

- **Paramètre d'URL :**
  - **id** (obligatoire) : ID numérique du client.
- **Réponse :**
  - Code HTTP : **204 No Content**

# Produits

## GET /api/products

Récupère la liste paginée des produits.

**Paramètres :**

- **page** (*int, optionnel*) : Numéro de la page à récupérer.
- **limit** (*int, optionnel*) : Nombre d'éléments par page.

**Réponse :**

```
[
  {
    "id": 1,
    "name": "Iphone 36",
    "manufacturer": "Apple",
    ...
  },
  ...
]
```

]

## **POST /api/products**

**Crée un nouveau produit.**

**Corps de la requête :**

```
{
  "name": "Galaxy S21",
  "manufacturer": "Samsung",
  ...
}
```

**Réponse :**

- Code HTTP : **201 Created**

```
{
  "id": 2
  "name": "Galaxy S21",
  "manufacturer": "Samsung",
  ...
}
```

## GET /api/products/{id}

Récupère les détails d'un produit spécifique.

Paramètre d'URL :

- `id` (*string, obligatoire*) : ID du produit.

Réponse :

```
{
  "id": 1,
  "name": "Iphone 36",
  "manufacturer": "Apple",
  ...
}
```

## PUT /api/products/{id}

Modifie un produit existant.

Paramètre d'URL :

- `id` (*string, obligatoire*) : ID du produit.

Corps de la requête :

```
{
  "name": "Iphone 36",
  "manufacturer": "Apple",
  ...
}
```

Réponse :

- Code HTTP : **200 OK**

## **DELETE /api/products/{id}**

Supprime un produit.

Paramètre d'URL :

- **id** (*string, obligatoire*) : ID du produit.

Réponse :

- Code HTTP : **204 No Content**

## **POST /api/products/{id}/addCustomers**

Associe un ou plusieurs clients à un produit.

Paramètre d'URL :

- **id** (*string, obligatoire*) : ID du produit.

Corps de la requête :

```
{
  "customer": [1, 2, 3, ...]
}
```

Réponse :

- Code HTTP : **204 No Content**

# Utilisateurs

## GET /api/users

Récupère la liste paginée des utilisateurs.

Paramètres de requête :

- **page** (*int, optionnel*) : Numéro de la page à récupérer
- **limit** (*int, optionnel*) : Nombre d'utilisateurs par page

Réponse :

```
[
  {
    "id": 1,
    "email": "j.dupond@example.com",
    "firstname": "Jean",
    "lastname": "Dupond"
    ...
  },
  ...
]
```

## POST /api/users

Crée un nouvel utilisateur.

Corps de la requête :

```
{
  "email": "j.dupond@example.com",
  "firstname": "Jean",
  "lastname": "Dupond",
  "password": "1234"
  ...
}
```

**Réponse :**

- Code HTTP : 201 Created

```
{
  "id": 2,
  "email": "j.dupond@example.com",
  "firstname": "Jean",
  "lastname": "Dupond"
  ...
}
```

**GET /api/users/{id}**

Récupère les informations d'un utilisateur.

**Paramètre d'URL :**

- **id** (*string, obligatoire*) : ID de l'utilisateur

**Réponse :**

```
{
  "id": 1,
  "email": "j.dupond@example.com",
  "firstname": "Jean",
  "lastname": "Dupond"
  ...
}
```



## PUT /api/users/{id}

Modifie un utilisateur.

Paramètre d'URL :

- `id` (*string, obligatoire*) : ID de l'utilisateur

Corps de la requête :

```
{  
  "email": "j.dupond@example.com",  
  "firstname": "Jean",  
  "lastname": "Dupond"  
  ...  
}
```

Réponse :

- Code HTTP : `200 OK`

## DELETE /api/users/{id}

Supprime un utilisateur.

Paramètre d'URL :

- `id` (*string, obligatoire*) : ID de l'utilisateur

Réponse :

- Code HTTP : `204 No Content`

## PUT /api/users/{id}/resetPassword

Réinitialise le mot de passe d'un utilisateur.

Corps de la requête :

```
{
  "password": "1234"
}
```

Réponse :

- Code HTTP : 200 OK

## PUT /api/users/{id}/updateRoles

Met à jour les rôles d'un utilisateur.

Corps de la requête :

```
{
  "roles": ["ROLE_ADMIN"]
}
```

Réponse :

- Code HTTP : 200 OK

## PUT /api/users/{id}/updateCustomer

Modifie le client associé à un utilisateur.

Corps de la requête :

```
{
  "customer": 2
}
```

✅ Réponse :

- Code HTTP : 200 OK