



# Documentation

# Cahier des charges

# BTS SIO SLAM

Mettre à disposition des utilisateurs un service informatique

*Par Damien Monchaty, Avril 2023*

Page 1 sur 13

## Sommaire :

Réaliser les tests d'intégration et d'acceptation d'un service.....	2
Déployer un service .....	10
Accompagner les utilisateurs dans la mise en place d'un service.....	12

# Réaliser les tests d'intégration et acceptation d'un service

Les tests d'intégration est une étape essentielle du développement et des tests de logiciel.

Ils vérifient simplement que les différentes parties de votre programme, que nous avons testé individuellement via des tests unitaires, fonctionnent bien une fois *intégrées* ensemble.

L'idée ici c'est de **simuler un comportement en production**, de bout en bout.

**Contexte :** Notre exemple sera le développement de tests d'intégrations d'une application de gestion de recette de cuisine.

**url :** <https://mycellar-api.azurewebsites.net/swagger/index.html>

Nous allons nous intéresser aux fonctionnalités suivantes :

- ▶ Identification
- ▶ Récupération de la liste d'article

Plusieurs options sont possibles pour réaliser des tests d'intégrations :

- ▶ Utiliser un outil tierce comme **soapUI, Postman**
- ▶ Ecrire un test d'intégration via l'utilisation d'une librairie **C#** comme **Cunit, MsTest**

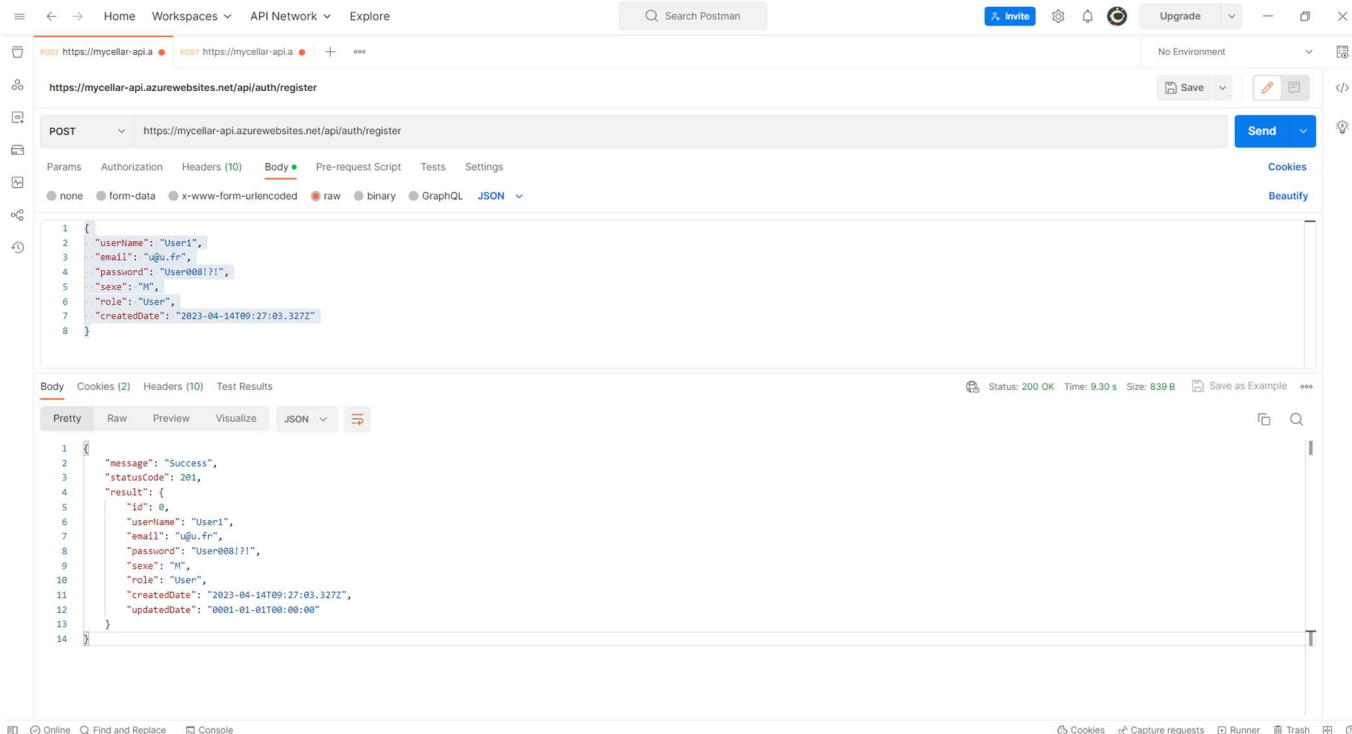
J'ai privilégié l'approche **Postman** pour ce contexte.

## Scénario de test :

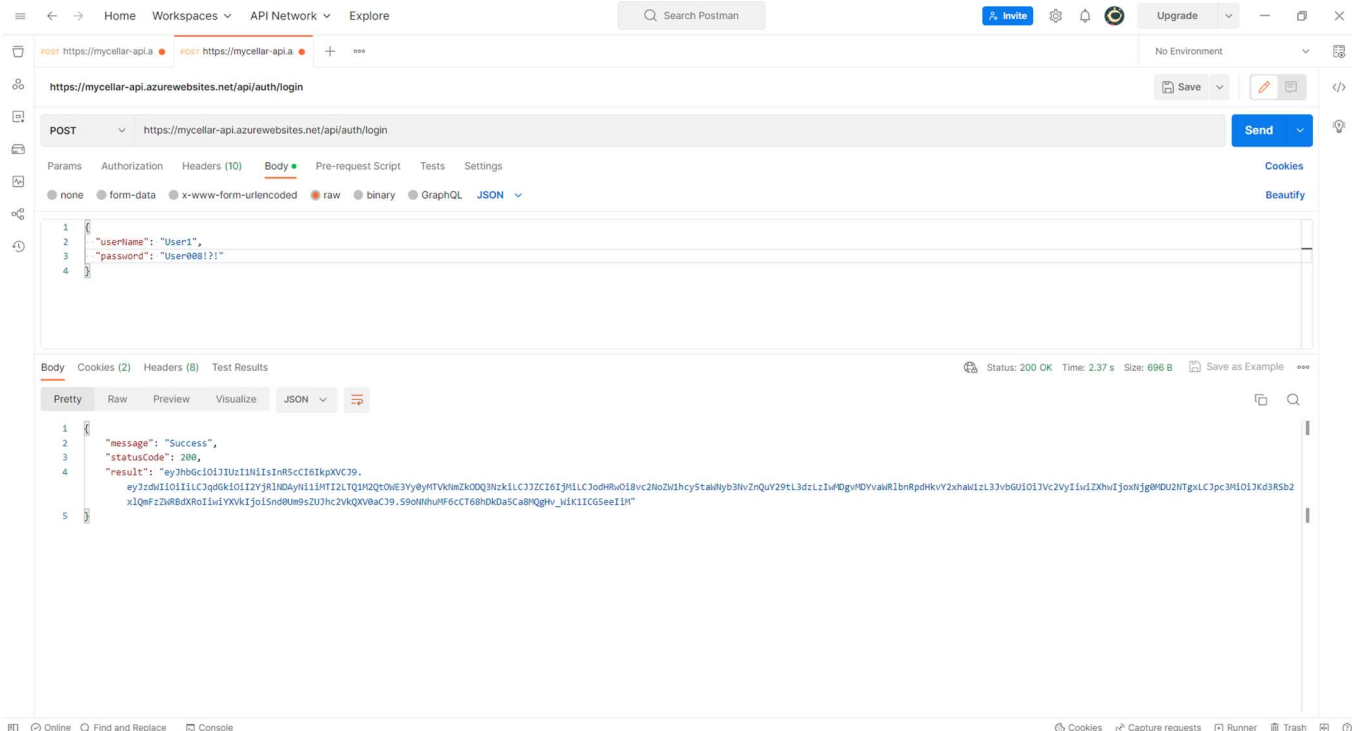
- ▶ 1. En tant qu'utilisateur ayant un role **USER**, je peux m'enregistrer..
- ▶ 2. En tant qu'utilisateur ayant un role **USER**, je peux me connecter.
- ▶ 3. En tant qu'utilisateur ayant un role **USER**, je peux afficher une liste de produits
- ▶ 4. En tant qu'utilisateur ayant un role **USER**, je peux afficher les informations d'un produit
- ▶ 5. En tant qu'utilisateur ayant un role **USER**, je ne peux pas créer de nouveaux produits
- ▶ 6. En tant qu'utilisateur ayant un role **USER**, je ne peux modifier les informations d'un produit.
- ▶ 7. En tant qu'utilisateur ayant un role **USER**, je ne peux pas supprimer un produit

## Etales :

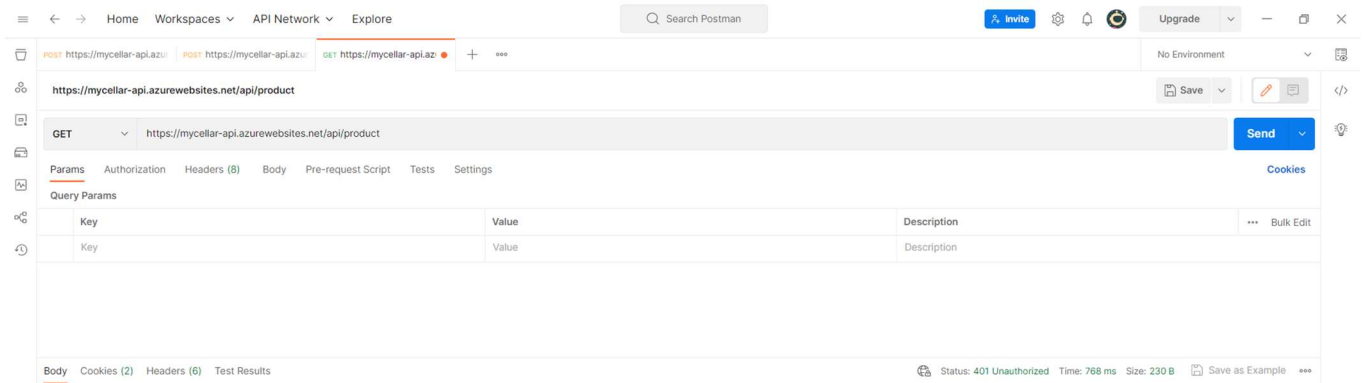
- 1. Faire une requête **POST** sur le chemin « **/auth/register** » pour **créer** un utilisateur.



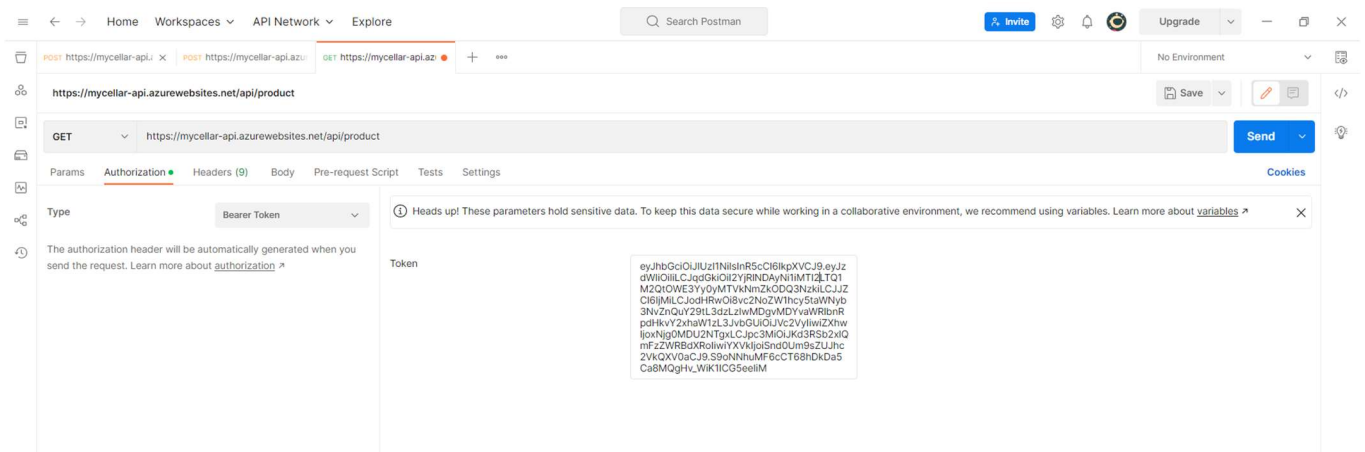
- 2. Faire une requête **POST** sur le chemin « **/auth/login** » pour **se connecter** avec les **informations de l'utilisateur** créé à l'étape précédente.



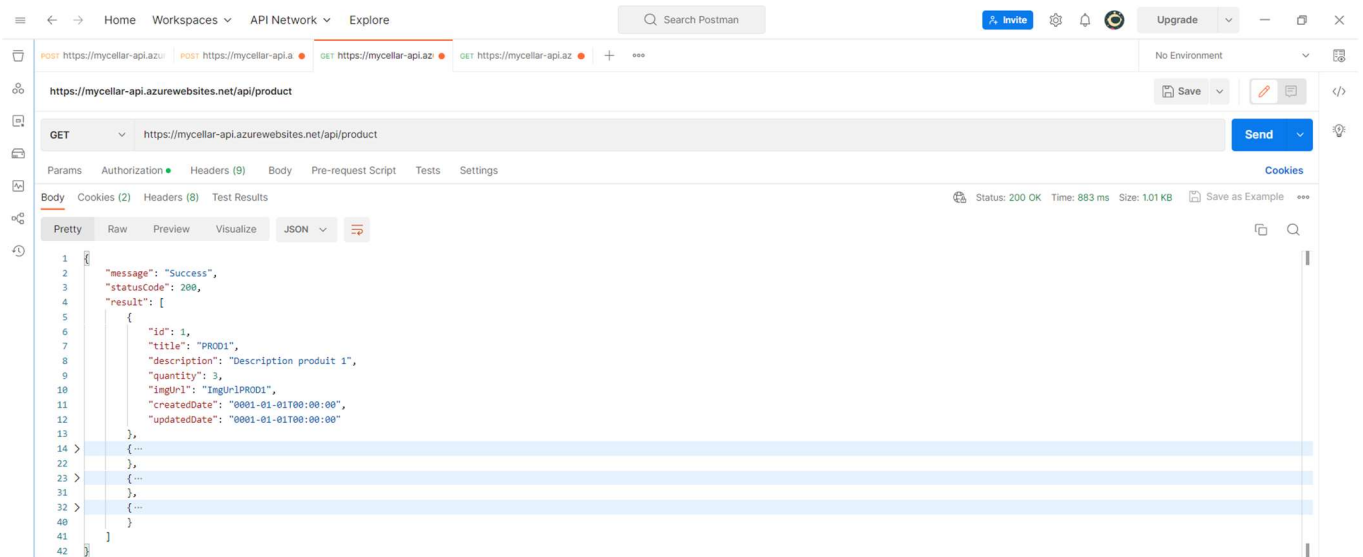
- **Ouvrir** un nouvel onglet de requête, **preciser** le chemin « **/api/product** », puis **faire** un requête **GET**, un **status code 401 Unauthorized** devrait être retournée dans la réponse.



- Dans l'onglet « **Authorization** », **sélectionnez** dans la barre déroulante « **Bearer Token** ».
- **Copier** le **JWT TOKEN** retourné dans la réponse de la requête « **/auth/register** » et **placer-le** à l'endroit désigné.



### ► 3. Faire de nouveau la même requête, et la liste de produits.



- Création d'une collection « **MyCellar-API** » dans l'outil **Postman**.
- Enregistrer cette requête dans la collection « **MyCellar-API** ».

**SAVE REQUEST**

Request name

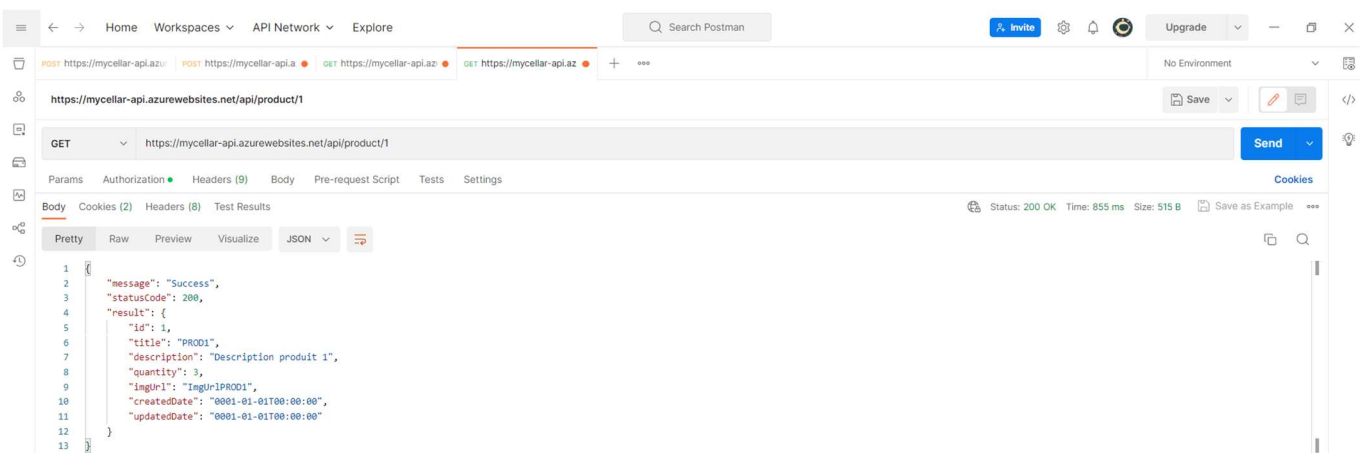
`https://mycellar-api.azurewebsites.net/api/product/1`

[Add description](#)

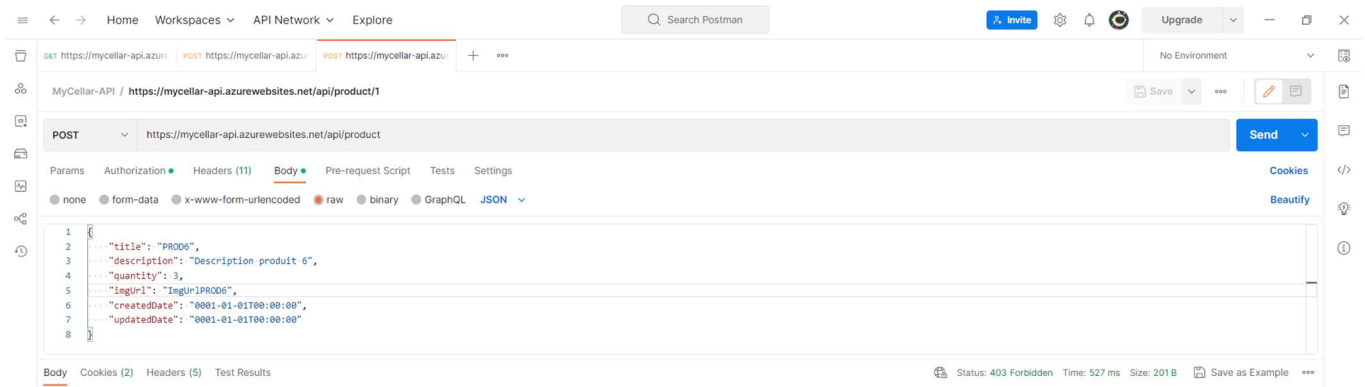
Save to My Workspace / **MyCellar-API**

`GET https://mycellar-api.azurewebsites.net/api/product`

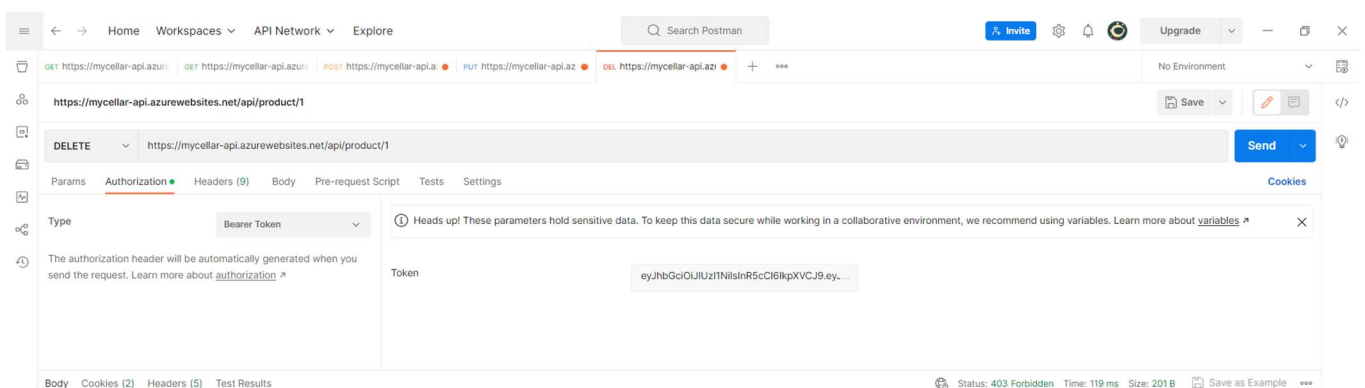
- 4. Ouvrir un nouvel onglet de requête, préciser le chemin « **/api/product/1** », puis faire un requête **GET**
- Enregistrer cette requête dans la collection « **MyCellar-API** ».



- ▶ 5. Ouvrir un nouvel onglet de requête, préciser le chemin « **/api/product** », puis faire un requête **POST**
- ▶ Préciser un corps (body) à la requête « **/api/product** »
- ▶ Un **status code 403 Forbidden** devrait être retournée dans la réponse car un utilisateur ayant un role « User » ne peut ni créer, ni modifier, ni supprimer un produit
- ▶ Enregistrer cette requête dans la collection « **MyCellar-API** ».

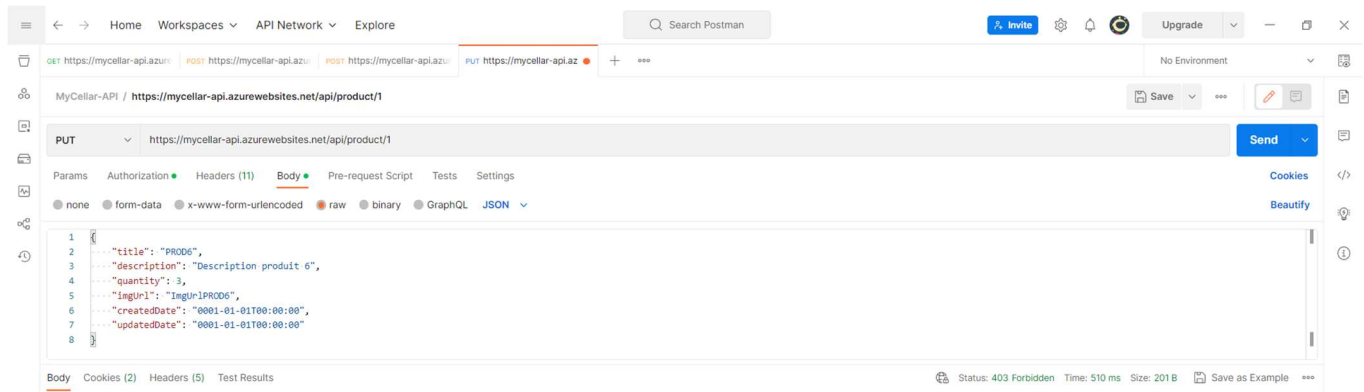


- ▶ 6. Ouvrir un nouvel onglet de requête, préciser le chemin « **/api/product/1** », puis faire un requête **PUT**.
- ▶ Préciser un corps (body) à la requête « **/api/product/1** ».
- ▶ Un **status code 403 Forbidden** devrait être retournée dans la réponse car un utilisateur ayant un role « User » ne peut ni créer, ni modifier, ni supprimer un produit
- ▶ Enregistrer cette requête dans la collection « **MyCellar-API** ».

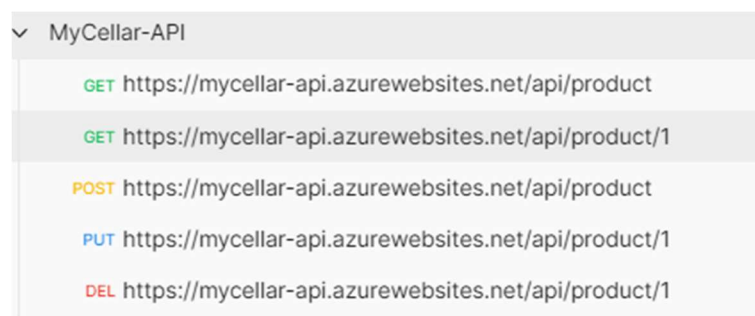




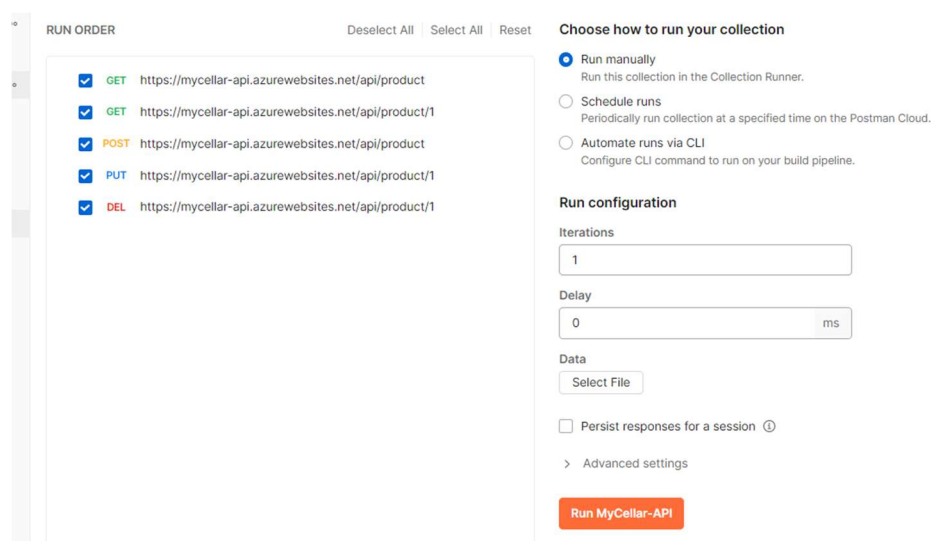
- ▶ 7. Ouvrir un nouvel onglet de requête, préciser le chemin « **/api/product/1** », puis faire une requête **DELETE**.
- ▶ Un **status code 403 Forbidden** devrait être retournée dans la réponse car un utilisateur ayant un rôle « User » ne peut ni créer, ni modifier, ni supprimer un produit
- ▶ Enregistrer cette requête dans la collection « **MyCellar-API** ».



- ▶ Notre collection POSTMAN



- ▶ Execution de la collection





► Résultat

MyCellar-API - Run results

Run Again

Automate Run ▾

+ New Run

📄 Export Results

👤 Run on Today, 13:05:21 · [View all runs](#)

Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time
Runner	none	1	1s 595ms	5	173 ms

All Tests Passed (5) Failed (0) Skipped (0)

[View Summary](#)

Iteration 1

1

<b>GET</b> https://mycellar-api.azurewebsites.net/api/product https://mycellar-api.azurewebsites.net/api/product	200 OK 281 ms 1.031 KB
PASS Status code is 200	
<b>GET</b> https://mycellar-api.azurewebsites.net/api/product/1 https://mycellar-api.azurewebsites.net/api/product/1	200 OK 271 ms 515 B
PASS Status code is 200	
<b>POST</b> https://mycellar-api.azurewebsites.net/api/product https://mycellar-api.azurewebsites.net/api/product	403 Forbidden 105 ms 201 B
PASS Status code is 203	
<b>PUT</b> https://mycellar-api.azurewebsites.net/api/product/1 https://mycellar-api.azurewebsites.net/api/product/1	403 Forbidden 105 ms 201 B
PASS Status code is 203	
<b>DELETE</b> https://mycellar-api.azurewebsites.net/api/product/1 https://mycellar-api.azurewebsites.net/api/product/1	403 Forbidden 105 ms 201 B
PASS Status code is 203	

# Déployer un service

Pour le déploiement, j'ai opté pour une approche **Cloud Computing** car cela me permet de stocker mes applications facilement via Internet.

Cela m'a permis de réduire les coûts de déploiement et de pouvoir travailler depuis n'importe où.

**Contexte :** Notre exemple sera le déploiement d'une application de gestion de ressources humaines / ticket.

Les environnements seront les suivants :

- ▶ L'environnement de **développement est notre machine locale**,
- ▶ L'environnement de **test est sur notre serveur**
- ▶ L'environnement de **production sera un hébergement de type Cloud Microsoft Azure**.

## Etapes :

- ▶ Se connecter à son compte **Microsoft Azure**

The screenshot shows the Microsoft Azure portal interface. At the top, there's a search bar and a user profile. Below, the 'Services Azure' section displays various service icons like App Services, Elasticsearch, and Azure Cosmos DB. The 'Ressources' section shows a table of recent resources. The 'Naviguer' section at the bottom provides quick links to subscriptions, resource groups, and the dashboard.

Nom	Type	Dernier affichage
MyCellar-API	App Service	il y a 3 heures
MyCellar	Groupe de ressources	il y a 3 heures
Abonnement Azure 1	Abonnement	il y a 5 heures
mysql-instanceserver	Serveur flexible Azure Database pour MySQL	il y a 6 heures

## ► Créer un service de type **App Services**

Microsoft Azure | Rechercher dans les ressources, services et documents (G+/)

Accueil > App Services

Répertoire par défaut (damienmonchaty@hotmail.onmicrosoft.com)

+ Créer | ⚙ Gérer la vue | ↻ Actualiser | ⬇ Exporter au format CSV | 🔗 Ouvrir une requête | 🏷 Attribuer des étiquettes | ▶ Démarrer | ⏪ Redémarrer | □ Arrêter | 🗑 Supprimer

Filtrer un champ... | Abonnement égal à tout | Groupe de ressources égal à tout | Emplacement égal à tout | Ajouter un filtre

Affichage de 1 à 1 sur 1 enregistrements.

<input type="checkbox"/> Nom ↑↓	Statut ↑↓	Emplacement ↑↓	Niveau tarifaire ↑↓	Plan App Service ↑↓	Abonnement ↑↓	Type d'a... ↑↓
<input type="checkbox"/> MyCellar-API	En cours d'ex...	USA Est	Gratuit	ASP-MyCellar-8964	Abonnement Azure 1	Application w...

< Précédent | Page 1 sur 1 | Suivant > | Envoyer des commentaires

- Ecrire les informations nécessaires dans l'onglet « **De base** »
- Ecrire les informations nécessaires dans l'onglet « **Déploiement** »
  - Mise en place du déploiement continu via **GitHub Actions**
- Appuyer sur le bouton « **Créer** » dans l'onglet « **Vérifier + créer** »

### Déploiement :

- Projet portfolio  
url : <https://damienmonchaty-btssio.com>
- Projet HRManagement  
url : <https://hrmanagementweb.azurewebsites.net>
- Projet MyCellar  
url : <https://mycellar-api.azurewebsites.net>
- Projet FormulaNews  
url : <https://formulanews.azurewebsites.net/>
- Serveur **Mysql** sur **Microsoft Azure**  
url : <https://mysql-instanceserver.mysql.database.azure.com>  
username : sql\_root  
password : password001!

# Accompagner les utilisateurs dans la mise en place d'un service

Pour certains projets, des manuels d'utilisation ont été créés pour offrir la **meilleure expérience client possible**.

**Contexte :** Conception de manuel utilisateur pour bonne usage du service déployé

**Par exemple :**

► **Projet : FormulaNews**

## Manuel D'utilisation

FormulaNews

Xamarin Forms – SqlLite

---

### Etape 1 : Déplacement du dossier dans votre espace de travail

- Dézippez le projet et **déplacez-le** dans votre **espace de travail**.
- Puis **ouvrez-le** dans votre **Visual Studio 2022**.

<https://visualstudio.microsoft.com/fr/vs/community/>

---

### Etape 2 : Mise en place de la librairie .NET MAUI

- Dans **Visual Studio 2022**, veuillez **cliquer** sur le bouton « Outils ».
- **Cliquer** sur le bouton « Outils et Fonctionnalités ».
- **Sélectionner** la charge « .NET Multi-platform App UI Development ».
- **Cliquer** sur le bouton « Installer » dans le coin inférieur droit.

---

### Etape 3 : Configuration d'un appareil - Android (Emulateur)

- **Activer le débogage** sur émulateur.
- **Cliquer** sur la flèche déroulante à côté du bouton « Déboguer ».
- **Choisir** « Créer un émulateur Android ».
- Dans l'écran de création de l'émulateur Android, utiliser les valeurs **par défaut**, puis **cliquer** sur le bouton « Créer ».
- La fenêtre de **gestionnaire d'appareils** devrait apparaître, **cliquer** sur le bouton « Démarrer ».
- **Visual Studio 2022** doit maintenant **afficher** le nouvel émulateur sur le bouton « Déboguer »

---

### Etape 3bis : Configuration d'un appareil – Android (Physique)

- **Activer le débogage** sur l'appareil.
- **Brancher** le téléphone via un **port USB**
- **Accéder** à l'écran « Paramètres ».
- **Sélectionner** « À propos du téléphone ».
- **Appuyer 7 fois** sur « Build number » jusqu'à ce que vous soyez maintenant **développeur** ! est visible.
- **Vérifier** que le **débogage USB** est activé sur l'appareil.

---

### Etape 4 : Démarrage de l'application

- Dans **Visual Studio 2022**, Charger le projet **FOneMobileApp.Android** en tant que **projet de démarrage**.
- **Cliquer** sur la flèche déroulante à côté du bouton « Déboguer » pour **sélectionner** un appareil.

► **Projet : HrManagement**

# Manuel D'utilisation

HRManagement .net6.0 - Mysql - ElasticSearch

---

## Etape 1 : Déplacement du dossier dans votre espace de travail

---

- Dézippez le projet et déplacez-le dans votre espace de travail.
- Puis ouvrez-le dans votre Visual Studio 2022.  
<https://visualstudio.microsoft.com/fr/vs/community/>

## Etape 2 : Installation de la base de données MySql

---

- Télécharger le serveur Mysql à l'adresse suivante :  
<https://dev.mysql.com/downloads/workbench/>
- Configurer votre serveur Mysql.
- Ouvrir une instance de Server dans MySQL Workbench
- Dans la section « File », cliquer sur le bouton « Open SQL Script ... »
- Rendez-vous à l'endroit où vous avez dézipper le projet et importer le script HRManDb.
- Dans la section « Query », cliquer sur le bouton « Execute (All or Selection)»

## Etape 3 : Mise en place d'ElasticSearch

---

- Télécharger le serveur ElasticSearch à l'adresse suivante :  
<https://www.elastic.co/fr/downloads/elasticsearch>
- Dézipper le dossier « ElasticSearch ».
- Se déplacer dans le dossier « bin » à la racine du dossier.
- Démarrez le serveur ElasticSearch en double-cliquant le fichier « elasticsearch.bat ».

## Etape 4 : Modification de la chaine de connexion à Mysql

---

- Rendez-vous dans le fichier appSettings.json.
- A la ligne ..., remplacer hrmandb par le nom sélectionné durant l'étape 2.3.
- Puis modifier les Username et Pass admin1@admin.fr word par celles configurés durant l'étape 2.2.

## Etape 6 : Démarrage de l'application

---

- Dans Visual Studio 2022, veuillez exécuter l'application via le serveur IIS.
- Utiliser ces informations pour vous connecter à l'application :
- En tant que Manager :  
Email : [admin1@admin.fr](mailto:admin1@admin.fr)  
Mot de Passe : [Man012345!](#)
- En tant qu'Employé :  
Email : [emp2@emp.fr](mailto:emp2@emp.fr)  
Mot de Passe : [Emp012345!](#)