

# 1. Qu'est-ce qu'un test d'intégration ?

Les **tests d'intégration** sont une étape du processus de test logiciel qui consiste à vérifier que **plusieurs composants d'une application fonctionnent correctement ensemble**.

## En détail :

- Après avoir validé chaque module séparément avec des **tests unitaires**, on passe aux **tests d'intégration** pour tester **l'interaction entre ces modules**.
- Le but est de détecter les **problèmes d'interfaçage** : mauvaise communication entre services, erreurs de format de données, bugs de logique, etc.

## Exemple simple :

Imaginons une application de gestion d'utilisateurs :

1. Le module A gère la création d'un utilisateur.
2. Le module B gère la connexion.
3. Le module C gère la récupération du profil.

Un **test d'intégration** pourrait vérifier que :

- Après avoir créé un utilisateur avec A, on peut bien se connecter avec B,
- et que la récupération du profil avec C renvoie les bonnes informations.

## 2. Types de tests d'intégration

- **Big Bang** : on intègre tout d'un coup et on teste.
- **Incremental** : on ajoute les modules un par un :
  - *Top-down* : on commence par les modules les plus hauts (interfaces)
  - *Bottom-up* : on commence par la base (données, services internes)

## 3. Démonstration avec Postman

Prenons un exemple concret d'API REST d'un service d'utilisateurs.

### Exemple d'API :

**Base URL :** `https://api.exemple.com`

Action	Méthode	Endpoint
Créer un utilisateur	POST	/users
Se connecter	POST	/auth/login
Obtenir le profil	GET	/users/me

# Étape 1 : Créer un utilisateur

Dans Postman :

1. Choisis la méthode **POST**
2. URL : `https://api.exemple.com/users`
3. Onglet **Body** → **raw** → **JSON**

```
{  
  "name": "Alice",  
  "email": "alice@test.com",  
  "password": "123456"  
}
```

4. Clique sur **Send**

Tu devrais recevoir une réponse comme :

```
{  
  "id": 1,  
  "name": "Alice",  
  "email": "alice@test.com"  
}
```

## Étape 2 : Se connecter

Toujours dans Postman :

1. Méthode **POST**

2. URL : `https://api.exemple.com/auth/login`

3. Body :

```
{  
  "email": "alice@test.com",  
  "password": "123456"  
}
```

#### 4. Cliquez sur **Send**

Réponse attendue :

```
{  
  "token": "eyJhbGciOiJIUzI1..."  
}
```

Copie le **token JWT** reçu.



## Étape 3 : Récupérer le profil (test d'intégration)

1. Méthode **GET**

2. URL : `https://api.exemple.com/users/me`

3. Onglet **Headers**

- Clé : `Authorization`
- Valeur : `Bearer eyJhbGciOiJIUzI1...` (le token reçu précédemment)

## 4. Cliquez sur **Send**

Réponse attendue :

```
{  
  "id": 1,  
  "name": "Alice",  
  "email": "alice@test.com"  
}
```

## Interprétation :

- Si les trois requêtes fonctionnent correctement **dans cet enchaînement**, alors **l'intégration entre les modules "création", "authentification" et "profil" est réussie.**

## Bonus : Automatiser le test dans Postman

Tu peux automatiser le scénario avec **Postman Tests** :

Dans l'onglet **Tests** après chaque requête, tu peux écrire du JavaScript, par exemple :

```
pm.test("Statut 200 OK", function () {  
    pm.response.to.have.status(200);  
});  
  
pm.test("Le token est présent", function () {  
    const jsonData = pm.response.json();  
    pm.expect(jsonData).to.have.property("token");  
});
```