

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université de Jijel
Faculté des Sciences exactes et de l'informatique
Département d'informatique



– Module –
Environnements et Programmation Dédiés

Master 1 : IA

Enseignant du module : Dr. Hemza FICEL

Contact: hemza.ficel@univ-jijel.dz

TP 3

Travail demandé

Développer une application web modulaire reposant sur deux services RESTful communiquant entre eux, afin de répondre à **un besoin métier spécifique choisi par l'étudiant** (exemples : gestion RH, éducation, tourisme, commerce, santé, etc.).

1. Une vitrine web (interface utilisateur) développée avec Spring Boot + Thymeleaf.
2. Un service RESTful pour les opérations CRUD sur une base de données locale.
3. Un service IA développé dans un autre langage (ex. Python Flask, Node.js, PHP...) qui consomme une API externe et renvoie des données enrichies.

Vitrine web

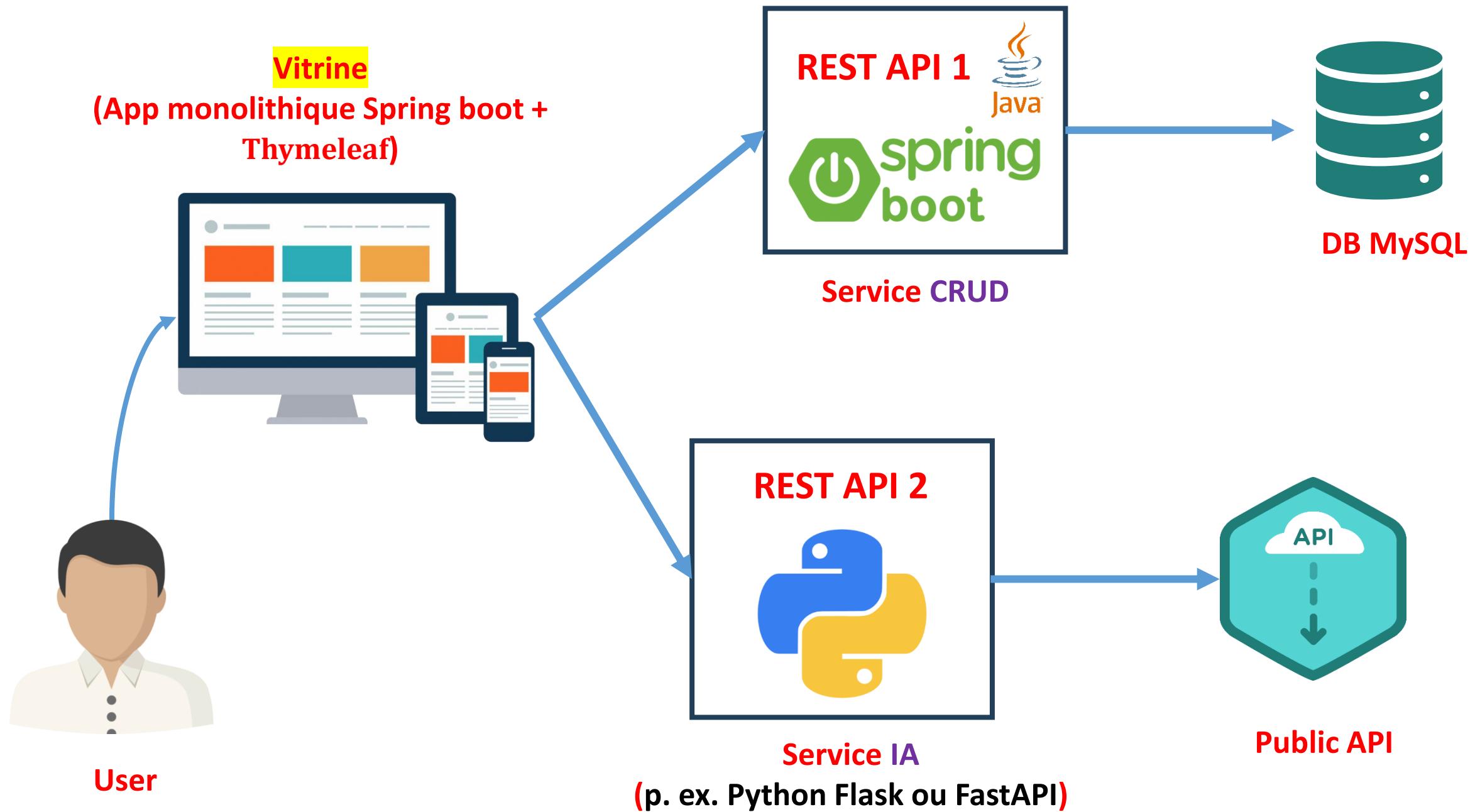
1. Développée sous la forme d'une application monolithique avec **Spring Boot et Thymeleaf**.
2. Sert **d'interface utilisateur principale pour interagir avec les services REST**.
3. Permet l'affichage et la gestion des données provenant du service REST principal (service de gestion de données).
4. Permet l'envoi de requêtes vers le service IA pour générer ou enrichir des informations (texte, résumé, prédiction, etc.).

Service RESTful de gestion de données

1. Développé avec Spring Boot.
2. Gère la base de données locale (mySQL) à l'aide de Spring Data JPA.
3. Expose des endpoints RESTful pour effectuer les opérations CRUD sur les entités métier (ex. : employés, produits, étudiants, clients, etc.).
4. Respecte les principes RESTful (stateless, JSON, verbes HTTP standards, ...).

Service RESTful d'IA

1. Expose une API REST Développé dans un autre langage que Java (par exemple : Python avec Flask ou FastAPI, Node.js, PHP...).
2. Consomme une API externe accessible en ligne (ex. : API d'un modèle LLM, API météo, API de recommandation, API de traduction, etc.).
3. Retourne un résultat enrichi ou généré automatiquement (ex. texte, prédition, résumé, conseil, contenu).
4. Respecte les principes RESTful (stateless, JSON, verbes HTTP standards, ...).



Options avancées

Mettre en place un mécanisme d'authentification et d'autorisation stateless pour sécuriser la communication entre les services de l'application.

1. Technologies à utiliser : serveur [Keycloak](#) + JWT + Spring Security + Keycloak Adapter.
2. L'utilisateur saisit ses identifiants (nom d'utilisateur, mot de passe) dans la vitrine web.
3. Ces informations sont envoyées au serveur Keycloak, qui agit comme un serveur d'identité (Identity Provider).
4. Si les identifiants sont valides :Keycloak génère un token JWT signé contenant les informations de l'utilisateur.
5. Le client joint ce token dans chaque requête HTTP vers les services REST.
6. Les services REST (Spring Boot) vérifient la validité du token avant d'exécuter l'opération demandée.

Free LLM API resources

Free API resources

<https://github.com/cheahjs/free-llm-api-resources>

<https://github.com/public-apis/public-apis>

<https://www.postman.com/cs-demo/public-rest-apis/collection/tfzpjqfc/public-rest-apis>

<https://apivault.dev>

Exemples d'idées de logique métier

Application Touristique intelligente

1. La vitrine permet à l'utilisateur de créer son profil voyageur.
2. Le service Spring boot permet la gestion des données (utilisateurs, destinations, hôtels, activités, ...).
3. Le service IA consomme une API externe (ex. Api LLM + OpenWeather, Google Maps, ...) pour recommander des destinations selon la météo, la saison, et les tendances, ...
4. Les recommandations reviennent sous forme de JSON à afficher sur la vitrine.

Boutique e-commerce avec recommandations IA

1. La vitrine permet à l'utilisateur de parcourir les produits, de consulter son panier et d'obtenir des suggestions.
2. Le service Spring Boot gère les données (produits, catégories, utilisateurs, commandes, avis).
3. Le service IA consomme une API externe (ex. Api LLM + une API de tendances de marché) pour recommander des produits similaires selon le comportement d'achat.
4. Les recommandations reviennent sous forme de JSON à afficher sur la vitrine.