

<b>DESCRIPTION D'UNE RÉALISATION PROFESSIONNELLE</b>		N° réalisation : 1
Nom, prénom : Spirine Vladimir		N° candidat : 2545863387
Épreuve ponctuelle <input checked="" type="checkbox"/>	Contrôle en cours de formation <input type="checkbox"/>	Date : 08/06/2026
Organisation support de la réalisation professionnelle SYADEM		
Intitulé de la réalisation professionnelle Outils d'analyse de tickets		
Période de réalisation : Décembre 2024 – Mars 2026 Lieu : Bordeaux		
Modalité : <input checked="" type="checkbox"/> Seul(e) <input type="checkbox"/> En équipe		
<b>Compétences travaillées</b> <ul style="list-style-type: none"> <li><input checked="" type="checkbox"/> Concevoir et développer une solution applicative</li> <li><input checked="" type="checkbox"/> Assurer la maintenance corrective ou évolutive d'une solution applicative</li> <li><input checked="" type="checkbox"/> Gérer les données</li> </ul>		
<b>Conditions de réalisation<sup>1</sup> (ressources fournies, résultats attendus)</b> <p><b>Ressources fournies :</b></p> <p>Exports de tickets issus de la plateforme Zendesk au format XML.</p> <p><b>Résultats attendus :</b></p> <p>Mettre en place un outil capable d'importer, nettoyer, structurer et analyser les tickets afin d'identifier les problèmes récurrents, faciliter leur exploitation et produire des résultats utiles sous forme de regroupements, résumés et statistiques.</p>		

<sup>1</sup> En référence aux *conditions de réalisation et ressources nécessaires* du bloc « Conception et développement d'applications » prévues dans le référentiel de certification du BTS SIO.

## **Description des ressources documentaires, matérielles et logicielles utilisées**

### **Ressources documentaires :**

Documentation officielle Ruby ; documentation Nokogiri (XML/HTML) ; documentation API Zendesk v2 ; documentation Ollama (API locale pour LLM) ; documentation Git ; échanges avec les utilisateurs métier.

### **Ressources matérielles :**

Ordinateur de développement dédié au projet ;

Environnement local de test et d'exécution ;

Poste configuré pour les traitements de données et l'exécution locale des modèles.

### **Ressources logicielles (versions utilisées) :**

- Ruby 3.3.10p183 ;
- Bundler 2.5.23 ;
- Nokogiri 1.19.1 ;
- HTTParty 0.24.2 ;
- dotenv 3.2.0 ;
- numo-narray 0.9.2.1 ;
- Rumale 2.1.0 (algorithmes de clustering, dont KMeans) ;
- Ollama (client 0.18.0) comme API locale pour piloter les LLM ;
- API Zendesk v2 ;
- Git 2.49.0.windows.1 ;
- Visual Studio Code 1.115.0.

## **Modalités d'accès aux productions et à leur documentation**

### **Code source :**

Dépôt GitHub du projet : [https://github.com/Farounaga/rb\\_tkts](https://github.com/Farounaga/rb_tkts)

### **Documentation du projet :**

Dossier documentaire partagé :

[https://drive.google.com/drive/folders/1ZjLn7umvwE\\_GNQ66TI6o1gSbzq24q19d?usp=drive\\_link](https://drive.google.com/drive/folders/1ZjLn7umvwE_GNQ66TI6o1gSbzq24q19d?usp=drive_link)

### **Documents disponibles :**

- Schéma de base de données
- Cahier des spécifications / cahier des charges
- Cahier de tests
- Documentation technique
- Documentation utilisateur
- Guides d'exploitation

**ANNEXE VII-1-B : Fiche descriptive de réalisation professionnelle  
(verso, éventuellement pages suivantes)**

**Épreuve E6 - Conception et développement d'applications (option SLAM)**

## **Descriptif de la réalisation professionnelle, y compris les productions réalisées et schémas explicatifs**

### **Contexte**

Cette réalisation s'inscrit dans le cadre de mon alternance au sein de **SYADEM**, éditeur de solutions logicielles dédiées à la vaccination et à la prévention. En tant que jeune développeur intégré à l'équipe de développement, cette mission m'a été confiée afin de me permettre de monter progressivement en compétences sur un besoin concret de l'entreprise.

La mission portait sur le traitement et l'analyse d'un volume important de tickets issus de l'activité support, un domaine nécessitant rigueur, fiabilité et autonomie.

### **Objectif de la réalisation**

Concevoir et développer un outil Ruby d'analyse de tickets permettant d'automatiser la chaîne de traitement, depuis l'import des données jusqu'à la restitution d'indicateurs utiles aux équipes support et produit.

### **Méthodologie de travail mise en œuvre**

- Recueil et clarification du besoin métier avec les utilisateurs ;
- Découpage en modules (import, parsing, nettoyage, vectorisation, clustering, similarité, reporting) ;
- Développement itératif avec tests réguliers sur des exports réels ;
- Versioning Git et validation progressive des résultats ;
- Ajustements des paramètres d'analyse selon les retours métier.

### **Architecture de la solution**

- Source de données : exports de tickets Zendesk (XML) ;
- Couche de traitement : scripts Ruby + Nokogiri pour l'extraction et la normalisation ;
- Couche IA locale : Ollama utilisé comme API locale pour piloter les LLM servant à la vectorisation et à la génération de résumés thématiques ;
- Couche analytique : clustering KMeans + calcul de similarité cosinus ;
- Couche de restitution : fichiers de synthèse et rapports exploitables par les équipes.

### **Modalités de déploiement**

La solution est exécutée en environnement local/interne, avec une configuration centralisée (variables d'environnement), des scripts versionnés et des sorties intermédiaires conservées pour audit et comparaison des exécutions.

### **Deploiement**

Exécution en environnement local/interne via scripts Ruby versionnés et configuration .env.

### **Securité**

Secrets/API keys hors dépôt Git, accès interne contrôlé, minimisation des données traitées.

### **Test**

Tests automatiques (rspec) et manuels, validations fonctionnelles sur exports réels, contrôle des clusters/similarités/résumés, validation métier.

### **Productions réalisées :**

- Développement du pipeline complet d'analyse des tickets ;
- Structuration et normalisation des données textuelles ;
- Mise en place des traitements IA/clusterisation ;
- Génération de livrables de synthèse (groupements, statistiques, éléments d'aide à la décision).

### **Flux de traitement (étapes) :**

- Import des exports de tickets Zendesk (XML)
- Extraction et parsing des données (Ruby + Nokogiri)
- Nettoyage et normalisation des tickets

- Vectorisation des contenus (embeddings via Ollama)
- Clustering KMeans et calcul de similarité cosinus
- Génération de résumés thématiques et reporting
- Restitution des résultats pour les équipes support et produit