## **Cahier des Charges : Application de Gestion d'Inventaire du Mydil**

### **1. Introduction et Contexte**

### **1.1. Présentation du Projet**

Le Mydil de l'EPSI est un espace crucial d'innovation et de création, fonctionnant comme un véritable FabLab mis à la disposition de l'ensemble du corps étudiant. Cet environnement est richement doté d'un parc de matériel technologique varié, allant des équipements informatiques de pointe (ordinateurs haute performance, casques de Réalité Virtuelle) aux outils spécialisés pour la fabrication et l'électronique (imprimantes 3D, kits Arduino/Raspberry Pi, fers à souder, multiprises spécifiques, etc.).

Actuellement, la gestion de cet inventaire repose sur des méthodes archaïques et manuelles (feuilles de calcul, cahiers de prêt papier, communication orale). Ce mode de fonctionnement a atteint ses limites, créant des difficultés majeures pour le suivi précis de l'état du matériel, la gestion des prêts et des retours, et la planification des opérations de maintenance. Il en résulte une perte de temps significative pour les administrateurs et une frustration pour les étudiants qui ne peuvent pas connaître la disponibilité d'un équipement en temps réel.

Le projet vise à résoudre ces inefficacités en développant une application centralisée, accessible à la fois via une interface web et une application mobile. L'ambition est de fournir une solution simple et moderne qui transformera la gestion d'inventaire du Mydil, passant d'un processus manuel et dispersé à un système optimisé, transparent et numérique.

### **1.2. Objectifs du Projet**

L'application doit remplir plusieurs objectifs stratégiques qui permettront d'optimiser l'utilisation des ressources du Mydil :

* **Centraliser l'Information :** L'objectif fondamental est de créer une **Base de Données Unique et fiable** regroupant l'intégralité du matériel. Chaque article, qu'il s'agisse de "l'Imprimante 3D Prusa i3 n°1" ou du "Casque VR Oculus Quest 2 n°3", doit avoir une fiche d'identité numérique complète, accessible à tous les acteurs autorisés.
* **Simplifier la Réservation et l'Accès :** Il s'agit d'offrir aux étudiants un moyen facile et immédiat de consulter la disponibilité du matériel et de le **réserver en ligne**. Un étudiant doit pouvoir vérifier sur son téléphone si un équipement est libre pour le créneau souhaité avant de se rendre au FabLab.
* **Optimiser le Suivi en Temps Réel :** Le système doit assurer un **suivi dynamique et précis** des flux. Dès qu'un administrateur valide un prêt (*Check-out*), le statut du matériel doit instantanément passer à "En Cours d'Emprunt", permettant une traçabilité sans faille des biens. De même, les retards de retour doivent générer des alertes immédiates.
* **Faciliter la Maintenance Proactive :** L'outil doit évoluer vers une logique de maintenance préventive. Il permettra de **planifier et de tracer** toutes les interventions (exemples : nettoyage hebdomadaire des buses d'imprimantes 3D, mise à jour trimestrielle du firmware des casques VR). En cas d'incident, l'historique des réparations sera directement attaché à la fiche de l'équipement.
* **Responsabiliser les Utilisateurs :** L'application doit mettre en place un **historique clair et inaltérable** des emprunts pour chaque étudiant. Cette traçabilité vise à responsabiliser les utilisateurs vis-à-vis du matériel emprunté, le respect des délais et le signalement des problèmes.

### **1.3. Cible Utilisateurs**

Le succès de l'application dépend de son adéquation avec les besoins spécifiques des deux grandes catégories d'utilisateurs au sein de l'EPSI :

* **Les Étudiants de l'EPSI :** Ils constituent la cible principale de l'application. Leur rôle est double : consulter la disponibilité du catalogue et effectuer des demandes de réservation et d'emprunt. Pour eux, l'application doit être avant tout simple, rapide, intuitive et accessible via leurs identifiants habituels (SSO).
* **Les Administrateurs du Mydil :** Ce groupe inclut le personnel technique, les enseignants responsables du FabLab et toute personne désignée par l'EPSI. Leur rôle est celui de la gestion totale : ils administrent l'inventaire (ajout/modification de matériel), valident ou refusent les demandes d'emprunt soumises par les étudiants, supervisent le *check-out* et le *check-in*, et assurent la gestion de la maintenance. Pour eux, l'outil doit être puissant, fournir une vue synthétique (Dashboard) de l'état du parc et permettre des actions rapides via des outils comme la lecture de QR Code.

## **2. Périmètre Fonctionnel**

Le périmètre fonctionnel de l'application est délimité par deux ensembles majeurs de fonctionnalités, conçus spécifiquement pour répondre aux besoins distincts des étudiants et des administrateurs du Mydil.

### **2.1. Fonctionnalités pour les Étudiants**

L'objectif principal du volet Étudiant est de simplifier l'accès à l'inventaire et la procédure d'emprunt, tout en garantissant une expérience utilisateur fluide et sécurisée.

#### **2.1.1. Authentification Unique (SSO)**

L'accès à l'application doit être instantané et s'intégrer parfaitement à l'environnement numérique de l'EPSI.

**Description détaillée :** La connexion s'effectuera par le mécanisme de l'**Authentification Unique (Single Sign-On ou SSO)**. Cela signifie qu'il n'y aura pas de formulaire d'inscription ou de connexion spécifique à l'application. Les étudiants utiliseront leurs identifiants de domaine habituels (ceux utilisés pour l'intranet, les mails, etc.).

**Processus de Connexion Simplifié :**

|  |  |  |
| --- | --- | --- |
| Étape | Action de l'Utilisateur | Résultat dans l'Application |
| 1. Accès | L'étudiant arrive sur la page d'accueil de l'application. | Affichage d'un bouton unique : "Se connecter avec les identifiants EPSI". |
| 2. Redirection | Clic sur le bouton de connexion. | Redirection sécurisée vers le portail d'authentification de l'école (Fournisseur d'Identité - IdP). |
| 3. Authentification | Saisie des identifiants (Login/Mot de passe EPSI). | Validation par le système central de l'EPSI. |
| 4. Accès & Sync. | Redirection automatique vers l'application Mydil. | L'application est ouverte et le profil de l'étudiant (Nom, Prénom, Classe) est automatiquement synchronisé et mis à jour. |

#### **2.1.2. Consultation de l'Inventaire**

L'inventaire doit fonctionner comme un véritable **catalogue en ligne interactif**, permettant une recherche rapide et une information précise sur la disponibilité.

**Catalogue et Fiches Produits :** Chaque équipement sera présenté sous forme de fiche détaillée incluant :

* Une photo claire de l'article.
* Une description textuelle.
* Les caractéristiques techniques principales (Exemple : Résolution du casque VR, taille du plateau de l'imprimante 3D).
* Le statut actuel de l'unité ("Disponible", "Réservée", "En cours d'Emprunt", "En Maintenance").

**Recherche Avancée et Filtrage :** L'étudiant doit pouvoir naviguer facilement dans l'inventaire en utilisant :

* La **Recherche textuelle** par nom ou mot-clé (Exemple : "Arduino", "Casque").
* Des **Filtres par Catégorie** (Informatique, Impression 3D, Électronique, Outils).
* Des filtres par **Statut** (Uniquement afficher ce qui est "Disponible").

**Vue de Disponibilité (Calendrier et Vue Tabulaire) :** La gestion de la disponibilité est cruciale. L'application proposera deux niveaux de visualisation :

1. **Vue Détaillée (Équipement Individuel) :** Pour un article spécifique (Exemple : *Casque VR 1*), un **calendrier visuel** affichera jour par jour les créneaux déjà réservés et ceux encore disponibles.
2. **Vue Synthétique (Catégorie de Matériel) :** Pour une catégorie comprenant plusieurs unités (Exemple : Casques VR), une **vue tabulaire (ou planning)** affichera l'ensemble des unités sur une période donnée (semaine type), permettant à l'étudiant de choisir rapidement une unité disponible.

|  |  |  |  |
| --- | --- | --- | --- |
| Type de Vue | Cible | Format | Objectif |
| Calendrier | Unité spécifique (Ex: PC Portable n°4) | Agenda graphique | Identifier les créneaux exacts (date et heure) sur une période longue. |
| Vue Tabulaire | Catégorie multiple (Ex: Souris filaires) | Tableau ou planning | Voir d'un coup d'œil combien d'unités sont disponibles sur les 7 prochains jours. |

##### **2.1.3. Gestion des Réservations**

* **Système de réservation simple** : L'étudiant peut sélectionner un équipement et choisir un créneau horaire (date et heure de début et de fin).
* **Panier de réservation** : Possibilité de réserver plusieurs articles en une seule fois.
* **Notifications** : Envoi d'emails ou de notifications push pour la confirmation, le rappel de début et le rappel de fin de prêt.
* **Historique des emprunts** : L'étudiant peut consulter son historique personnel d'emprunts.
* **Processus de Demande d'Emprunt** :
  1. **Sélection et Créneau** : L'étudiant sélectionne l'équipement et définit la période d'emprunt souhaitée.
  2. **Soumission de la Demande** : Un clic sur "**Réserver**" soumet la demande à l'administrateur. La demande passe au statut "**En attente de validation**".
  3. **Validation par l'Administrateur** : L'emprunt ne devient effectif qu'après validation par un administrateur du Mydil.
  4. **Confirmation/Refus** : L'étudiant reçoit une notification (email/push) l'informant de la validation ou du refus de sa demande, avec un motif clair en cas de refus.
  5. **Statut de la Réservation** : L'historique personnel de l'étudiant reflète le statut actuel (En attente, Validée, Refusée, En cours, Terminée).

### **2.2. Fonctionnalités pour les Administrateurs**

Le volet Administrateur est l'interface de pilotage, conçue pour être complète et optimiser la gestion quotidienne de l'inventaire.

#### **2.2.1. Gestion Complète de l'Inventaire**

**Tableau de Bord (Dashboard) :** La page d'accueil de l'administrateur sera un centre de commande visuel présentant les indicateurs clés :

* Nombre total d'équipements / Nombre disponible.
* Statistiques de maintenance (Équipements en panne, Interventions prévues).
* Alerte rapide sur les **demandes d'emprunt en attente** de validation et les **retards de retour**.

**Interface CRUD (Create, Read, Update, Delete) :** Interface complète pour la gestion du cycle de vie des équipements. L'administrateur peut :

* **Ajouter** un nouvel équipement (avec photo, caractéristiques, localisation).
* **Modifier** les informations ou le statut d'un équipement existant.
* **Supprimer** un équipement (avec historique archivé).

**Gestion par QR Code :** Pour fluidifier les actions de terrain, chaque équipement se verra attribuer un QR Code unique.

* L'administrateur pourra scanner le QR Code avec l'application mobile pour :
  + Accéder directement à la fiche détaillée de l'équipement.
  + Déclencher immédiatement les actions de **Check-out** (prêt) ou **Check-in** (retour).

#### **2.2.2. Gestion des Prêts et Retours**

Cette interface centralise l'ensemble du flux des emprunts.

**Gestion du Flux de Demandes (Validation) :** Un tableau de bord spécifique affichera, par ordre de priorité, toutes les demandes soumises par les étudiants.

|  |  |  |
| --- | --- | --- |
| Action | Description | Impact |
| Valider | Confirmer la demande d'emprunt. Possibilité d'ajustement du créneau (après accord étudiant) ou d'ajout d'un commentaire interne. | La réservation passe au statut "Validée". Une notification est envoyée à l'étudiant. |
| Refuser | Rejeter la demande. Obligation de saisir un Motif de Refus clair. | La réservation passe au statut "Refusée". L'équipement redevient disponible. |
| Check-out | Validation physique du prêt au moment de la remise de l'équipement (via scan QR Code). | Le statut passe à "En Cours". Déclenchement du chronomètre de prêt. |
| Check-in | Validation physique du retour de l'équipement (via scan QR Code). | Le statut passe à "Terminée". L'équipement redevient "Disponible". |

**Suivi des Retards :** Le système doit générer des **alertes automatiques** dès qu'un équipement n'est pas rendu à l'heure convenue. Un tableau de bord affichera clairement les emprunts en retard, permettant à l'administrateur de prendre les mesures nécessaires (envoi de rappel, contact de l'étudiant).

#### **2.2.3. Suivi de la Maintenance**

L'application agit comme un système de GMAO (Gestion de la Maintenance Assistée par Ordinateur) léger pour le Mydil.

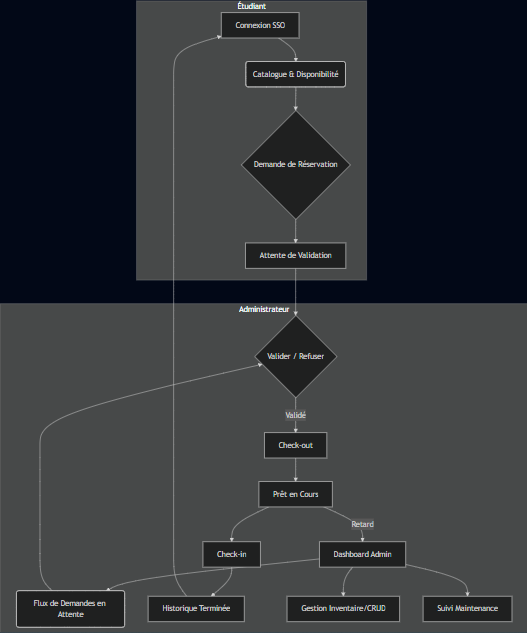
* **Déclaration d'Incident :** Un administrateur peut rapidement déclarer un incident sur une fiche équipement (Exemple : "Le ventilateur de l'Imprimante 3D n°2 fait un bruit suspect"). Le statut de l'équipement passe immédiatement à **"En Maintenance"**.
* **Planification et Historique :** Interface dédiée pour planifier les interventions (préventives ou correctives). L'historique des opérations (date, nature de l'intervention, coût éventuel) est conservé et attaché à l'équipement pour une analyse de la durée de vie.

#### **2.2.4. Gestion des Utilisateurs et des Rôles**

Le système doit maintenir à jour la liste des utilisateurs de l'EPSI et gérer leurs droits d'accès.

* **Synchronisation :** La gestion des utilisateurs est basée sur la synchronisation avec l'annuaire de l'école (via LDAP, SAML, ou protocole SSO).
* **Attribution des Rôles :** Une interface simple permettra aux administrateurs de haut niveau d'attribuer et de modifier les rôles :
  + **Étudiant :** Accès limité à la consultation et à la réservation.
  + **Administrateur :** Accès complet aux fonctionnalités de gestion (CRUD, Validation des prêts, Maintenance).
  + *(Potentiellement un rôle intermédiaire :)* **Administrateur Junior :** Accès à la validation des prêts uniquement, sans pouvoir modifier l'inventaire.

**Schéma des Flux Principaux :**



## **3. Spécifications Techniques et Contraintes**

Cette section définit l'environnement technologique requis pour le développement, la sécurité, et le déploiement de l'Application de Gestion d'Inventaire du Mydil. Ces choix techniques sont fondamentaux pour garantir la performance, l'évolutivité et la bonne intégration dans l'infrastructure de l'EPSI.

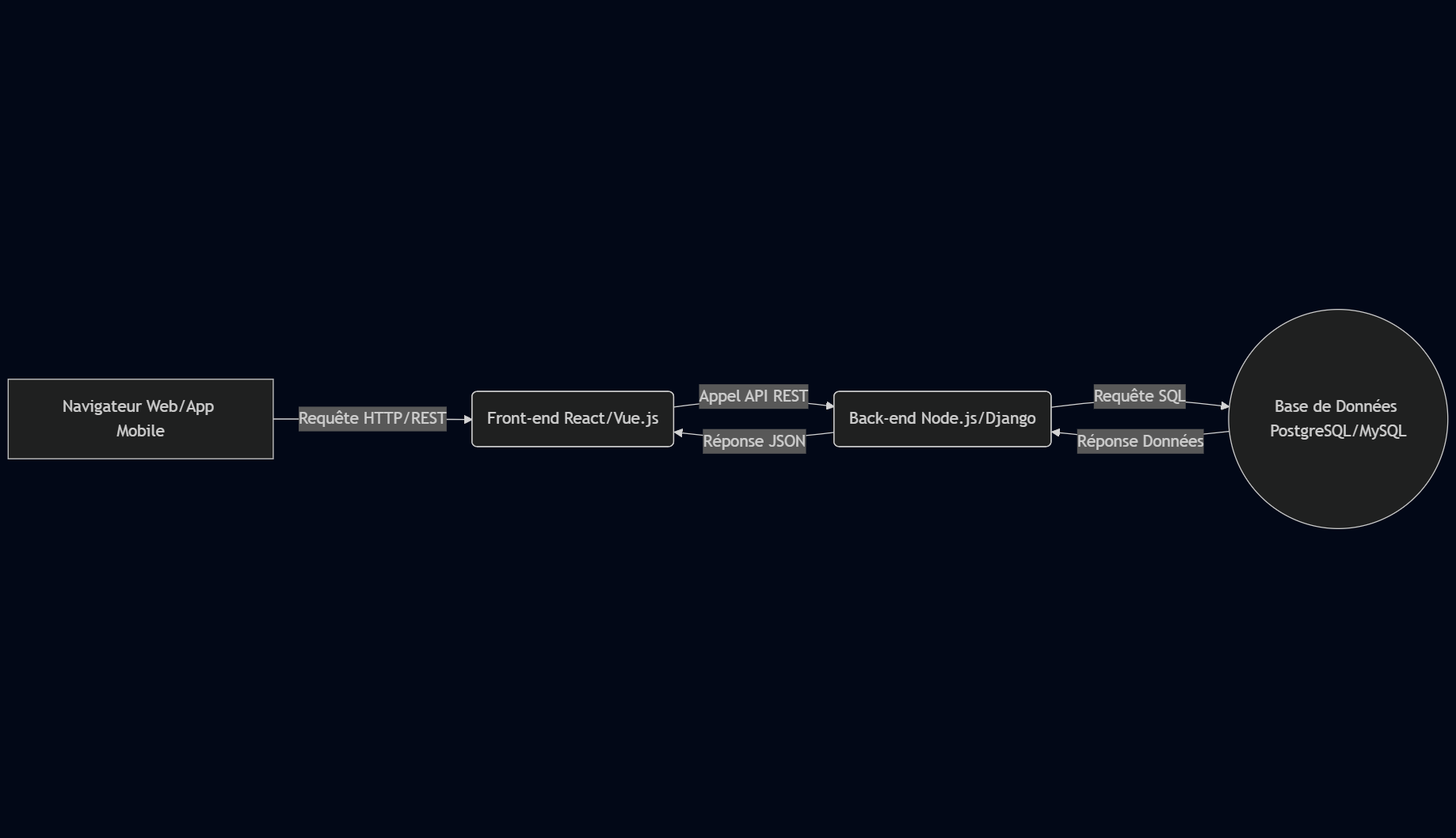
### **3.1. Architecture Technique de l'Application**

Le projet adoptera une architecture moderne, de type **client-serveur**, séparant clairement la logique de présentation (Front-end) des processus métiers et de la gestion des données (Back-end et Base de Données).

**Composition de l'Architecture :**

|  |  |  |  |
| --- | --- | --- | --- |
| Composante | Rôle | Recommandations Techniques (Exemples) | Contraintes |
| Front-end (Client) | Interface utilisateur (Web et Mobile). Doit assurer l'affichage rapide du catalogue et des calendriers. | Web : React (pour sa robustesse et sa communauté) ou Vue.js (pour sa légèreté et sa courbe d'apprentissage rapide). | Conception obligatoire en Responsive Design pour une utilisation optimale sur mobile (lecture QR Code, consultation en déplacement). |
| Back-end (API) | Contient la logique métier, la gestion des utilisateurs, la validation des demandes et la communication avec la base de données. | Node.js (avec Express ou NestJS) pour la rapidité de développement, ou un framework basé sur Python (Django) / PHP (Symfony). | Doit exposer une API RESTful claire et documentée pour permettre l'échange de données structurées entre le front-end et le back-end. |
| Base de Données | Stockage persistant et structuré de toutes les informations (Inventaire, Utilisateurs, Historique des prêts, Maintenance). | **Base de Données Relationnelle (SGBDR) :** PostgreSQL ou MySQL, privilégiées pour leur fiabilité dans la gestion des relations complexes (Équipement ↔ Prêt ↔ Utilisateur). | |  | | --- | | Performance cruciale pour les requêtes sur la disponibilité (calendriers). Nécessité d'une modélisation rigoureuse. |   Exporter vers Sheets |

**Schéma de l'Architecture (Vue Simplifiée) :**



### **3.2. Sécurité et Authentification**

La sécurité est une priorité absolue, notamment en raison de l'intégration avec le système d'information central de l'EPSI (SSO) et du traitement de données personnelles des étudiants.

#### **3.2.1. Authentification SSO (Single Sign-On)**

L'application doit s'intégrer de manière transparente avec le fournisseur d'identité (IdP) de l'EPSI.

* **Protocole d'Intégration :** L'intégration devra utiliser un protocole standard et sécurisé. Les protocoles à privilégier sont **SAML 2.0** ou **OpenID Connect (OIDC)**, qui garantissent un échange sécurisé des identités après la connexion de l'utilisateur. Une vérification technique préalable avec le service informatique de l'EPSI est impérative pour déterminer le protocole exact (ex: si l'EPSI utilise un annuaire LDAP ou Active Directory, l'intégration pourrait passer par une passerelle).
* **Flux d'Information :** L'authentification réussie doit permettre de récupérer automatiquement les attributs nécessaires à la création du profil utilisateur dans l'application (Nom, Prénom, Classe, Identifiant unique EPSI).

#### **3.2.2. Gestion des Droits et des Autorisations**

La sécurité doit être gérée par un système de contrôle d'accès basé sur les rôles (Role-Based Access Control - RBAC)

|  |  |  |
| --- | --- | --- |
| Rôle | Périmètre d’accès | Exemples d'Actions Autorisées |
| Étudiant | Périmètre Utilisateur | Consulter l'inventaire, Réserver du matériel, Consulter son historique. |
| Administrateur | Périmètre Gestionnaire | Toutes les actions de l'étudiant, + CRUD inventaire, Validation des prêts, Gestion de la Maintenance. |

Chaque appel à l'API (Back-end) devra systématiquement vérifier les droits de l'utilisateur (via un *token* JWT ou équivalent) avant d'exécuter l'action demandée.

#### **3.2.3. Protection des Données et Conformité (RGPD)**

L'application manipulera des données à caractère personnel (nom, prénom, classe, historique d'emprunt).

* **Sécurité des Données :** La base de données doit être sécurisée contre les accès non autorisés. Toutes les communications entre le client et le serveur devront être chiffrées via **HTTPS/TLS**.
* **Respect du RGPD :**
  + **Principe de Minimisation :** Seules les informations strictement nécessaires au fonctionnement de l'application seront stockées.
  + **Conservation :** Définition d'une politique de conservation des historiques d'emprunt et d'une procédure d'anonymisation ou de suppression des données des étudiants ayant quitté l'école.

## **4. Hébergement de l'Application**

Le choix de l'hébergement impacte directement les coûts, la maintenance et la disponibilité de l'application.

Le projet doit être conçu pour être agnostique en matière d'infrastructure, en utilisant la **conteneurisation** (Docker) pour garantir une portabilité maximale entre les différents environnements.

|  |  |  |  |
| --- | --- | --- | --- |
| Option d'Hébergement | Description Détaillée | Avantages pour le Projet | Inconvénients et Contraintes |
| Option 1 : Hébergement Local (On-Premise) | Déploiement sur un serveur physique ou virtuel géré par le Service Informatique de l'EPSI. | Contrôle total sur les données et l'infrastructure. Pas de coûts récurrents de location (après l'investissement initial). Idéal si les connexions internes au SSO et LDAP sont facilitées par l'hébergement local. | Coût initial d'acquisition/mobilisation du serveur. Le Service Informatique de l'EPSI est responsable de la maintenance, des mises à jour de l'OS et de la haute disponibilité. |
| Option 2 : Hébergement Cloud (VPS ou PaaS) | Utilisation d'un Serveur Privé Virtuel (VPS) chez un fournisseur (Exemples : Scaleway, OVHcloud, Azure, AWS). | Flexibilité et Scalabilité (montée en charge facilitée). Maintenance de l'infrastructure de base déléguée au fournisseur. Haute disponibilité assurée par l'opérateur. | Coûts récurrents (loyer mensuel). Nécessite une configuration sécurisée des réseaux et des firewalls pour garantir l'accès au SSO interne de l'école |

Recommandation Technique Finale :

Quelle que soit l'option retenue par l'EPSI, il est fortement recommandé d'utiliser Docker et Docker Compose pour conteneuriser le Front-end, le Back-end et la Base de Données. Cela garantira que le déploiement sur un serveur local ou un VPS Cloud est reproductible et standardisé, simplifiant la phase de mise en production.

### **5. Livrables Attendus**

* Dossier de conception technique.
* Code source complet et commenté.
* Application fonctionnelle déployée et accessible.
* Documentation utilisateur et technique.