

PROJET FINAL : LIGUE SPORTIVE D'AUVERGNE

Module : Développement API, Stack MERN et TypeScript

1. CONTEXTE ET OBJECTIFS

1.1. Contexte Métier

La Ligue Sportive d'Auvergne souhaite moderniser ses services en proposant une application web permettant la location de matériel sportif à ses adhérents. Cette plateforme doit permettre aux utilisateurs de réserver du matériel et aux administrateurs de gérer le catalogue et les utilisateurs.

1.2. Objectifs Pédagogiques

Ce projet de fin de module a pour but de valider l'acquisition des compétences suivantes :

- Développement d'une application web complète (Fullstack) en partant de zéro.
- Conception et implémentation d'une base de données NoSQL (MongoDB).
- Développement d'une API RESTful avec Node.js et Express.
- Utilisation du langage TypeScript et de la Programmation Orientée Objet (POO).
- Développement d'une interface utilisateur dynamique avec React JS.
- Travail en équipe et gestion de projet.

2. SPÉCIFICATIONS FONCTIONNELLES

L'application doit comporter trois volets distincts : l'accès public, l'espace adhérent et l'espace administrateur.

2.1. Accès Public et Authentification

- **Page d'inscription** : Formulaire permettant à un visiteur de créer un compte pour devenir adhérent.
- **Page de connexion** : Formulaire permettant l'authentification des adhérents et des administrateurs pour accéder aux fonctionnalités protégées.

2.2. Espace Adhérent

Une fois connecté, l'adhérent doit avoir accès aux fonctionnalités suivantes :

- **Catalogue Produits** : Listing des produits disponibles à la location.
 - Les produits doivent être filtrables ou organisés par thématique sportive (ex: Football, Natation).
 - Exemples de produits Football : Ballon, chasuble, maillot arbitre.
 - Exemples de produits Natation : Bonnet, ballon water-polo, lunettes.
- **Panier de réservation** :
 - Possibilité d'ajouter des articles au panier.
 - Page récapitulative du panier permettant de valider la réservation.
- **Gestion des stocks** : La validation d'un panier doit impacter les quantités disponibles en base de données (décrémenter le stock).

2.3. Espace Administrateur

L'administrateur dispose des mêmes droits que l'adhérent, avec des privilèges supplémentaires :

- **Gestion des Adhérents (CRUD)** : Interface permettant de Créer, Lire, Mettre à jour et Supprimer des comptes utilisateurs.
- **Gestion des Produits (CRUD)** : Interface permettant d'ajouter de nouveaux produits, de modifier les informations existantes (nom, stock, catégorie) et de supprimer des articles.

3. SPÉCIFICATIONS TECHNIQUES

Le respect de la stack technique MERN et de l'usage de TypeScript est impératif.

3.1. Backend (Serveur et API)

- **Technologie** : Node.js avec le framework Express.
- **Langage** : TypeScript.
- **Architecture** : Le code doit être structuré en utilisant la Programmation Orientée Objet (Classes, Contrôleurs, Modèles).
- **Architecture API** : Architecture RESTful standard.

3.2. Base de Données

- **Technologie** : MongoDB.
- **Hébergement** : MongoDB Atlas (Cluster Cloud).
- **Modélisation** : Définition de schémas stricts pour les collections (Users, Products, Orders).
- **Opérations** : Implémentation complète des opérations CRUD (Create, Read, Update, Delete).

3.3. Frontend (Interface Client)

- **Technologie** : React JS.
- **Langage** : JavaScript ou TypeScript (au choix, cohérence requise).

- **Navigation** : Utilisation de react-router-dom pour la gestion des routes (SPA).
- **Persistance locale** : Utilisation du LocalStorage pour gérer la persistance des données de session (token, panier en cours).
- **Composants** : Utilisation de composants fonctionnels et des Hooks React.

4. MODALITÉS D'ÉVALUATION ET LIVRABLES (JOUR 5)

L'évaluation porte à la fois sur la qualité technique du produit, la qualité du code et la restitution professionnelle.

4.1. Rapport Écrit (Dossier Projet)

Un document d'analyse synthétisant les deux jours de projet. Il doit contenir :

- **Analyse du besoin** : Compréhension du sujet.
- **Déroulement de la production** : Étapes clés du développement.
- **Organisation** : Tableau de répartition des tâches au sein de l'équipe.
- **Documentation technique** : Explication des choix d'architecture, diagrammes (optionnel mais valorisé), et justification des solutions techniques face aux difficultés rencontrées.

4.2. Soutenance Orale

Présentation devant le jury avec support visuel (PowerPoint, Canva, PDF).

- **Durée** : À définir (généralement 15-20 minutes).
- **Contenu de la présentation** :
 1. Présentation du contexte et de l'équipe (rôle de chacun).
 2. Démonstration fonctionnelle de l'application (Démocast ou Live).
 3. Explication technique (Architecture, points de complexité résolus).
 4. Conclusion (Bilan du projet, axes d'amélioration).