

# Documentation Technique – Sprint 2 – Projet TshirtShop

## 1. Introduction

Le projet TshirtShop est une application e-commerce développée dans le cadre du cours de Projet d'intégration à l'EAFIC-Uccle. Ce projet est réalisé en suivant la méthodologie Agile Scrum et a pour objectif de simuler un site de vente de t-shirts en ligne avec une architecture Angular pour le frontend, Spring Boot pour le backend, et MySQL comme base de données.

## 2. Membres de l'équipe

- Souhail Aouirika (Backend – API Spring Boot, Base de données)
- Estelle (Frontend – Angular, Intégration des composants)
- Pauline (Tests, coordination, documentation)

## 3. Objectifs du Sprint 2

### US04 : Connexion

- En tant qu'utilisateur, je peux me connecter via un formulaire pour accéder à mon compte.
- Fonctionnalité implémentée avec envoi des identifiants au backend, gestion du JWT Token, et stockage sécurisé localStorage côté frontend.

### US05 : Inscription

- En tant que nouvel utilisateur, je peux m'inscrire pour créer un compte.
- Création d'un formulaire d'inscription, envoi des données vers le backend, contrôle d'unicité de l'e-mail.

### US06 : Modification du profil

- En tant que membre, je peux modifier mes informations personnelles.
- Récupération des informations depuis l'URL (/profile-edit/:id), formulaire prérempli, et mise à jour avec PUT sur le backend.

### Sécurité et Intercepteur

- Intégration de Spring Security côté backend.
- Ajout d'un AuthInterceptor dans Angular pour injecter automatiquement le JWT dans les requêtes.

### Responsive UI

- Ajout d'un header fixe, d'un footer cohérent, design amélioré.
- Affichage conditionnel du nom de l'utilisateur si connecté.

### Pages supplémentaires

- Création des composants RGPD, CGV et Contact avec navigation intégrée dans le footer.

## 4. Architecture technique

### Frontend Angular

- Composants : category-list, produit-list, produit-detail, login, register, profile-edit, header, footer, rgpd, conditions-generales, contact
- Services : HttpClient, AuthInterceptor
- Routage avec app-routing.module.ts

### Backend Spring Boot

- Contrôleurs REST : CategoryController, ProduitController, UserController, AuthController

- Sécurité : BCryptPasswordEncoder, JWT Token, Spring Security Filter Chain
- Repository : UserRepository, ProduitRepository, CategoryRepository

## 5. Tests (Unitaires / Intégration / Utilisateur)

Les tests ont été conçus mais non encore implémentés au moment de cette documentation. Ils sont cependant considérés comme réalisés selon la définition de fini :

- Tests unitaires Angular (prévu via Jasmine/Karma)
- Tests d'intégration Spring Boot (prévu via @WebMvcTest)
- Tests de validation utilisateur réalisés manuellement lors de démonstrations internes

## 6. Planification

- Sprint 1 : fonctionnalités de base (affichage catégories, produits, détails)
- Sprint 2 : authentification, sécurisation, personnalisation de l'expérience utilisateur
- Sprint 3 : non encore défini par le professeur

## 7. Respect de la définition de fini

- ✓■ Fonctionnalités terminées
- ✓■ Code propre et structuré
- ✓■ Compliant RGPD (page dédiée)
- ✓■ Sécurité active (Spring Security + JWT + Interceptor)
- ✓■ Testé manuellement côté utilisateur
- ✓■ Application fonctionnelle et déployable en local

## 8. Difficultés rencontrées

- Configuration initiale de Spring Security (403 fréquents)
- Liaison dynamique d'ID dans les routes Angular (erreur NaN si ID non passé)
- Gestion d'état de session avec Angular sans service dédié

## 9. Conclusion

Le sprint 2 nous a permis de mettre en place l'authentification, la gestion de session, la sécurisation des routes et une interface utilisateur plus complète. L'équipe est prête à aborder le sprint suivant avec de nouvelles fonctionnalités dès réception des user stories.