

# 1. INTRODUCTION

## 1.1 Contexte

Avec l'évolution du numérique, de nombreux services du quotidien tendent à se digitaliser. Cependant, la recherche d'artisans qualifiés (plombiers, serruriers, peintres, électriciens, etc.) reste souvent informelle, basée sur le bouche-à-oreille ou des contacts peu fiables.

Ce manque de structuration entraîne :

- des difficultés pour les clients à trouver rapidement un artisan compétent ;
  - une visibilité limitée pour les artisans locaux ;
  - une absence de suivi et de traçabilité des interventions.
- 

## 1.2 Objectif du projet

L'objectif de ce projet est de concevoir et développer une **plateforme numérique de mise en relation entre artisans et clients**, permettant :

- aux clients de trouver rapidement un artisan adapté à leur besoin ;
  - aux artisans de proposer leurs services et gérer leurs demandes ;
  - à la plateforme d'assurer un cadre structuré, sécurisé et fiable.
- 

# 2. PÉRIMÈTRE DU PROJET

Le projet couvre :

- le développement d'un **backend RESTful** avec Spring Boot ;
- le développement d'un **frontend web** avec React ;
- la gestion des utilisateurs, des services et des demandes.

Le projet **n'inclut pas** :

- le paiement en ligne réel (paiement simulé si nécessaire) ;
  - la géolocalisation GPS en temps réel ;
  - les applications mobiles natives.
- 

# 3. ACTEURS DU SYSTÈME

### **3.1 Client**

- Personne recherchant un service artisanal.

### **3.2 Artisan**

- Prestataire proposant des services (plomberie, serrurerie, etc.).

### **3.3 Administrateur**

- Responsable de la supervision et de la gestion de la plateforme.
- 

## **4. FONCTIONNALITÉS DU SYSTÈME**

### **4.1 Fonctionnalités Client**

- Création de compte et authentification ;
  - Consultation des catégories de services ;
  - Création d'une demande de service ;
  - Diagnostic guidé du besoin (questionnaire) ;
  - Suivi de l'état de la demande ;
  - Notation de l'artisan après intervention.
  - Publication avant-apres
  - v
- 

### **4.2 Fonctionnalités Artisan**

- Inscription et authentification ;
  - Gestion du profil professionnel ;
  - Ajout et modification des services proposés ;
  - Réception et traitement des demandes ;
  - Acceptation ou refus d'une demande ;
  - Consultation de l'historique des interventions.
- 

### **4.3 Fonctionnalités Administrateur**

- Gestion des utilisateurs (clients et artisans) ;
- Validation ou suspension des comptes artisans ;
- Gestion des catégories de services ;

- Consultation des statistiques générales ;
  - Modération des avis.
- 

## 5. FONCTIONNALITÉ SPÉCIALE (DIFFÉRENCIANTE)

### Diagnostic de besoin guidé

Le système propose un **assistant de diagnostic** sous forme de questionnaire simple (oui/non), permettant :

- d'identifier automatiquement le type de service requis ;
  - de réduire les erreurs de choix par le client ;
  - d'améliorer la pertinence des demandes envoyées aux artisans.
- 

## 6. CONTRAINTES TECHNIQUES

### 6.1 Backend

- Langage : Java
  - Framework : Spring Boot
  - Architecture : API REST
  - Sécurité : Spring Security + JWT
  - Base de données : MySQL ou PostgreSQL
  - ORM : JPA / Hibernate
- 

### 6.2 Frontend

- Framework : React
  - Langage : TypeScript
  - Navigation : React Router
  - Communication API : Axios
- 

### 6.3 Outils

- Gestion de version : Git / GitHub
- Tests API : Postman
- Documentation API : Swagger / OpenAPI

---

## 7. EXIGENCES NON FONCTIONNELLES

- Sécurité des données utilisateurs ;
  - Performance acceptable pour plusieurs utilisateurs simultanés ;
  - Interface simple et intuitive ;
  - Code structuré et maintenable ;
  - Respect des bonnes pratiques de développement.
- 

## 8. LIVRABLES ATTENDUS

- Cahier des charges validé ;
  - Diagrammes UML (cas d'utilisation, classes) ;
  - Code source backend (Spring Boot) ;
  - Code source frontend (React) ;
  - Documentation technique de l'API ;
  - Rapport final du projet.
- 

## 9. PLANNING PRÉVISIONNEL (indicatif)

Phase	Description	Durée
Analyse & conception	Cahier des charges, UML	1 semaine
Backend	API, sécurité, base de données	2 semaines
Frontend	Interfaces, routes, API	1 semaine
Tests & validation	Tests fonctionnels	1 semaine

---

## 10. CONCLUSION

Ce projet vise à proposer une solution numérique moderne et efficace pour faciliter la mise en relation entre artisans et clients, tout en mettant en œuvre des compétences techniques solides en développement backend et frontend.