

# Projet NFA018 : Système de Gestion de Parking (Car Parking Management System)

## I- INTRODUCTION

### I.1- But du document

Ce document sert de guide pour la réalisation du projet de développement d'un **Système de Gestion de Parking (Car Parking Management System - CPMS)**. Il définit les objectifs, les exigences, les contraintes, l'organisation, la planification, les ressources nécessaires, ainsi que les livrables attendus. Ce document est destiné à l'équipe projet, aux parties prenantes et aux évaluateurs.

### I.2- Historique

Avec l'urbanisation croissante et l'augmentation du nombre de véhicules, la gestion des parkings est devenue un défi majeur. Les systèmes traditionnels, souvent manuels, sont inefficaces, coûteux et sujets à des erreurs. Ce projet vise à moderniser la gestion des parkings en proposant une solution automatisée, intuitive et scalable.

### I.3- Références

- Normes de développement logiciel : **ISO/IEC 12207**.
- Méthodologies de gestion de projet : **PMBOK, PRINCE2**.
- Études de marché sur les systèmes de gestion de parking existants.
- Documentation technique sur les technologies utilisées (Java, Python, SQL, etc.).

### I.4- Définition et acronymes

- **CPMS** : Car Parking Management System.
- **WBS** : Work Breakdown Structure (Structure de découpage du travail).
- **RACI** : Responsible, Accountable, Consulted, Informed (Matrice des responsabilités).
- **PERT** : Program Evaluation and Review Technique (Technique d'évaluation et d'examen de programmes).
- **UI/UX** : User Interface / User Experience (Interface utilisateur / Expérience utilisateur).

## II- PRESENTATION DU PROJET

### II.1- Objectifs du projet

- **Automatiser la gestion des places de parking** : Suivi en temps réel des places disponibles.
- **Faciliter la réservation en ligne** : Interface utilisateur pour réserver une place à l'avance.
- **Optimiser l'utilisation des parkings** : Réduction des temps d'attente et des congestions.
- **Générer des rapports statistiques** : Données sur l'occupation, les revenus, etc.
- **Améliorer l'expérience utilisateur** : Interface intuitive et accessible.

### II.2- Périmètres et non-périmètre

- **Périmètre** :
  - Gestion des places de parking (disponibilité, réservation).
  - Interface utilisateur pour les clients et les administrateurs.
  - Suivi des véhicules entrant et sortant.
  - Génération de rapports (occupation, revenus, etc.).
- **Non-périmètre** :
  - Gestion des paiements en ligne (optionnel pour une phase ultérieure).
  - Maintenance des véhicules (hors scope).
  - Gestion des amendes ou des infractions.

### II.3- Exigences prioritaires

1. **Fonctionnelles** :
  - a. Réservation de places en temps réel.
  - b. Gestion des utilisateurs (clients et administrateurs).
  - c. Suivi des véhicules (entrées/sorties).
  - d. Génération de rapports (occupation, revenus, etc.).
2. **Non-fonctionnelles** :
  - a. Interface utilisateur intuitive (UI/UX).
  - b. Sécurité des données (authentification, chiffrement).
  - c. Scalabilité (capacité à gérer plusieurs parkings).
  - d. Performance (temps de réponse < 2 secondes).

### II.4- Contraintes

- **Budget** : Limité à 8 000 €.
- **Délai** : Date limite de livraison fixée au **19 février 2021**.
- **Ressources humaines** : Équipe de 4 personnes (1 chef de projet, 2 développeurs, 1 designer).

- **Ressources matérielles** : Serveurs, ordinateurs, licences logicielles.

### III- ORGANISATION DU PROJET

#### III-1- Matrice des rôles du projet

Rôle	Responsabilités
<b>Chef de projet</b>	Supervision globale, gestion des délais et budget.
<b>Développeurs</b>	Conception, codage et tests du système.
<b>Designer</b>	Conception de l'interface utilisateur (UI/UX).
<b>Testeurs</b>	Validation des fonctionnalités et reporting bugs.

#### III-2- Besoins en formation

- Formation sur les outils de développement : **Java, Python, SQL**.
- Formation sur les outils de gestion de projet : **Microsoft Project, Trello**.
- Formation sur les bonnes pratiques de sécurité des données.

#### III-3- Ressources et environnement matériels

- **Serveurs** : Hébergement du système (cloud ou local).
- **Outils de développement** : IDE (Visual Studio Code), bases de données (MySQL).
- **Matériel informatique** : Ordinateurs portables, accès internet haut débit.
- **Logiciels** : Licences pour les outils de développement et de design (Adobe XD, Figma).
- **Outils de communication** : Utilisation de plateformes comme Slack pour la communication d'équipe et Zoom pour les réunions à distance.

### IV- PILOTAGE DE PROJET

#### IV-1- Estimation des charges et durée

- **Charge estimée** : 200 heures réparties sur 6 semaines.

- **Durée : Du 1er janvier 2021 au 19 février 2021.**

## **IV-2- Planification**

### **IV-2-1- WBS (Work Breakdown Structure)**

1. **Conception** (20 heures)
  - a. Analyse des besoins.
  - b. Conception de l'architecture technique.
  - c. Conception de l'interface utilisateur (UI/UX).
2. **Développement** (120 heures)
  - a. Développement du backend (gestion des places, réservations).
  - b. Développement du frontend (interface utilisateur).
  - c. Intégration des modules.
3. **Tests** (40 heures)
  - a. Tests unitaires.
  - b. Tests d'intégration.
  - c. Tests de performance.
4. **Déploiement** (20 heures)
  - a. Installation sur serveur.
  - b. Formation des utilisateurs.
  - c. Documentation technique et utilisateur.

### **IV-2-2- Diagramme de GANTT**

- **Semaine 1-2** : Conception (1er janvier - 15 janvier).
- **Semaine 3-4** : Développement (16 janvier - 5 février).
- **Semaine 5** : Tests (6 février - 12 février).
- **Semaine 6** : Déploiement et documentation (13 février - 19 février).

### **IV-2-3- Jalons**

- **Jalon 1** : Fin de la conception (15 janvier).
- **Jalon 2** : Fin du développement (5 février).
- **Jalon 3** : Fin des tests (12 février).
- **Jalon 4** : Livraison du projet (19 février).

## **IV-3- Matrice de responsabilité : Modèle RACI**

Tâche	Responsable	Accountable	Consulted	Informed
-------	-------------	-------------	-----------	----------

Conception	Développeur	Chef de projet	Designer	Testeurs
Développement	Développeur	Chef de projet	-	Testeurs
Tests	Testeurs	Chef de projet	Développeur	-
Déploiement	Chef de projet	Développeur	-	Tous

#### IV-4- Réunion de suivi

- **Fréquence** : Hebdomadaire.
- **Objectifs** :
  - Faire le point sur l'avancement des tâches.
  - Identifier les risques et proposer des solutions.
  - Ajuster la planification si nécessaire.

#### IV-5- Gestion des risques

Risque	Impact	Probabilité	Solution proposée
Retard dans le développement	Élevé	Moyenne	Planification avec marges de sécurité.
Problèmes techniques	Élevé	Faible	Tests réguliers et documentation détaillée.
Dépassement du budget	Moyen	Faible	Suivi rigoureux des dépenses.

#### IV-6- Tableau récapitulatif des livrables

Livable	Date prévue
Document de conception	15 janvier
Version beta	5 février
Version finale	19 février

#### ***V- BUDGET***

<b>Poste</b>	<b>Coût (€)</b>
Ressources humaines	5 000
Matériel	2 000
Logiciels	1 000
<b>Total</b>	<b>8 000</b>

#### ***CONCLUSION***

Ce projet de système de gestion de parking répond à un besoin concret d'automatisation et d'optimisation des parkings. Grâce à une planification rigoureuse, une gestion efficace des ressources et une communication claire entre les membres de l'équipe, nous sommes confiants dans la réussite de ce projet.