

# Parking intelligent

## Problématique

Aujourd'hui, les villes sont prises avec la question insoluble du stationnement : les zones les plus attractives suscitent une demande toujours croissante de stationnement, mais l'abondance de cases nuit à la qualité des milieux de vie.

À Paris par exemple, les conducteurs prennent en moyenne vingt minutes pour se garer, et une voiture sur cinq en circulation cherche une place – d'après Sébastien Capelle, expert Smart Parking et Mobilité Territoriale chez Orange Business Services.

De ce fait, les enjeux de diminution du stress, de gain de temps, mais aussi de décongestion des villes et de réduction de la pollution sont donc très importants.

Si on prend en compte le facteur « humain », connaître quasi instantanément le nombre et la localisation des places de stationnement libres, et se dire qu'on va gagner du temps est une source de stress en moins : En ville, où la vie citadine est déjà assez stressante, pouvoir diminuer ces facteurs de tension est une véritable aubaine.

Désormais, la tendance est au smart parking : un parking plus intelligent car il repense l'espace existant pour l'optimiser, ainsi, nul besoin de construire des parkings onéreux et polluants.

## Proposition de valeur

On se propose de modéliser un parking intelligent qui permet la gestion d'un parking en termes de places disponibles. Tout utilisateur de véhicule peut choisir à l'avance une place – la plus proche possible pour gagner du temps - dans le parking.

Le parking est équipé à l'entrée par des capteurs qui permettent d'ouvrir la barrière à l'entrée et à la sortie des véhicules ainsi de faire le paiement sans contact avec un smartphone.

Toute place dans le parking est équipée de capteurs qui détectent la présence d'un véhicule, et une application mobile regroupe toutes les informations du parking ( le nombre de voitures garées, la position des places vides, le taux d'occupation...)

# BUSINESS MODEL CANVAS

## Partenaires

Les fournisseurs de services cloud.

Les fournisseurs de matériels électroniques.

## Activités Clés

Conception de câblage et architecture.

Développement de l'application mobile.

Service clientèle.

## Ressources Clés

Développement mobile.

Ingénierie IOT.

Matériels IOT et service cloud.

Infrastructure réseau.

## Proposition de Valeur

Élargissement du segment des clients grâce à la bonne organisation des parking.

Une application sécurisée et facile à utiliser.

Possibilité de réserver une place dans un parking.

Une solution moins chère et plus efficace par rapport à l'existant.

## Relation Client

Mise à jour de l'application et un service après-vente.

Assistance intelligente 24/24 et 7/7.

Application personnalisée.

Documentation gratuite pour s'adapter avec l'application.

## Canaux de distribution

Contact direct (Bouches à oreilles).

Réseaux Sociaux.

## Segments Clients

Les parking privés.

Les parking publiques dans les stades, les foires ...

Les hypermarchés.

## Structures de Coûts

Achat du matériel nécessaire.  
Maintenance de la solution.  
Service cloud.

## Sources de Revenus

Ventes de de solution.

Frais de mise à jour et de personnalisation.

## Cartes et capteurs utilisés

- Capteur Ultrason HC-SR04 pour détecter la présence d'un véhicule dans une place. Ce capteur ultrason est relativement low cost par rapport les autres capteurs (infrarouge ou laser par exemple), il fonctionne avec une tension d'alimentation de 5 volts, dispose d'un angle de mesure de 15° environ et permet de faire des mesures de distance entre 2 centimètres et 4 mètres avec une précision de 3 mm.
- Capteur ultrason pour détecter l'entrée et la sortie des voitures.
- Module PN532 NFC pour le paiement mobile sans contact. Presque tous les téléphones haut de gamme du marché prennent en charge NFC.
- Un servomoteur pour contrôler la position de la barrière.
- Un afficheur LCD à l'entrée du parking, pour indiquer le nombre de places vides.
- Carte Arduino UNO : pas chère, langage de programmation simple et répond aux fonctionnalités exigées de notre projet.

## Choix technologique

La partie backend :

- MongoDB pour la gestion de base de données NoSQL. MongoDB est flexible et facile à gérer.
- Mosquitto : MQTT broker, Mosquitto est un courtier très léger. Il prend en charge TLS et il existe des plugins pour l'autorisation à l'aide d'une base de données.

La partie Middleware :

- serveur Node.js : Node.js brille dans les applications Web en temps réel utilisant la technologie push sur les Websockets avec des connexions bidirectionnelles.

Mobile device : Flutter, il dispose d'un riche ensemble de bibliothèques et plusieurs benchmarks affirment qu'il est plus performant que React Native.

## Livrables

Cahier de charges conceptuelle : Ce document présente de manière détaillée et structurée les spécifications, les services à rendre, les contraintes de cette solution aussi que la conception architecturale et la conception détaillée ( diagrammes UML).

Le code source de la solution : L'ensemble des instructions et des fichiers dans un répertoire en Github contenant le code de l'application mobile développée.

Manuel Utilisateur : Un document contenant des instructions bien claires et lisibles pour la mise en marche de la solution et des interfaces développées.

## Les grandes lignes du projet

- Conception et architecture des objets connectés
- Préparation de l'environnement et génération des certificats
- Préparer la partie hardware et MQTT
- Schématisation de la base de données et connexion avec le middleware server
- Liaison hardware et software
- Préparer les interfaces
- Sécurité et localisation
- Préparation des livrables

## Schéma de conception

