**Cahier des charges**

1. Contextualisation
2. Problématique de communication
3. Stratégie de communication
4. Stratégie créative
5. Stratégie des moyens
6. Structure

**I / Contextualisation**

**Concept**

D’après une étude de l’INSEE, 4% des ménages ne possèdent pas de machine à laver. Ces personnes sont principalement des étudiants et des jeunes actifs. En effet, ils n’ont pas le budget, manquent de place, ou ne meublent pas leur logement de façon permanente.

L’environnement prend de plus en plus de place dans notre quotidien. Une partie croissante de la population souhaite consommer mieux, être minimaliste, ne plus acheter d’équipements superflus. La durée d’un lave linge est de 7 en moyenne. 8 français sur 10 pratiquent ou ont l'intention de pratiquer la consommation collaborative : cette nouvelle façon de consommer est en pleine expansion.

Les laveries sont généralement la seule solution pour ces personnes. Cependant utiliser une comporte des problèmes qui sont l’attente pendant le cycle du lavage et le manque d’hygiène. De plus, il n’y a pas forcément de laverie dans un lieu proche de son domicile, et cela peut s’avérer difficile au quotidien, s’il faut utiliser la voiture, le bus, le métro etc....

Wash@Home est une application qui permet aux détenteurs de machines à laver de proposer un service de nettoyage du linge sale. Avec Wash@Home nous “uberisons” le nettoyage des vêtements. Wash@Home propose de mutualiser une machine à laver. Cela permettrait d’économiser de l’espace, de l’énergie, de l’eau, des déchets et du temps et de l’argent.

En se connectant sur l’application, l’utilisateur souhaitant laver son linge peut rechercher une machine à laver (un Washer) disponible près de chez elle/lui. Il peut réserver un créneau pour déposer son linge au washer qui lui rendra propre à une date ultérieure. Le washer peut proposer ce service contre une rémunération via l’application. Chaque Washer définit le montant qu’il veut appliquer pour ce service. Chaque partie peut déposer un avis et s’entre noter.

**L’équipe :**

#### **Nico :** Product Owner & dev front

#### **Magali :** Scrum Master & dev back

#### **Guilhem :** Lead dev back

#### **Swann :** Lead dev front & git master

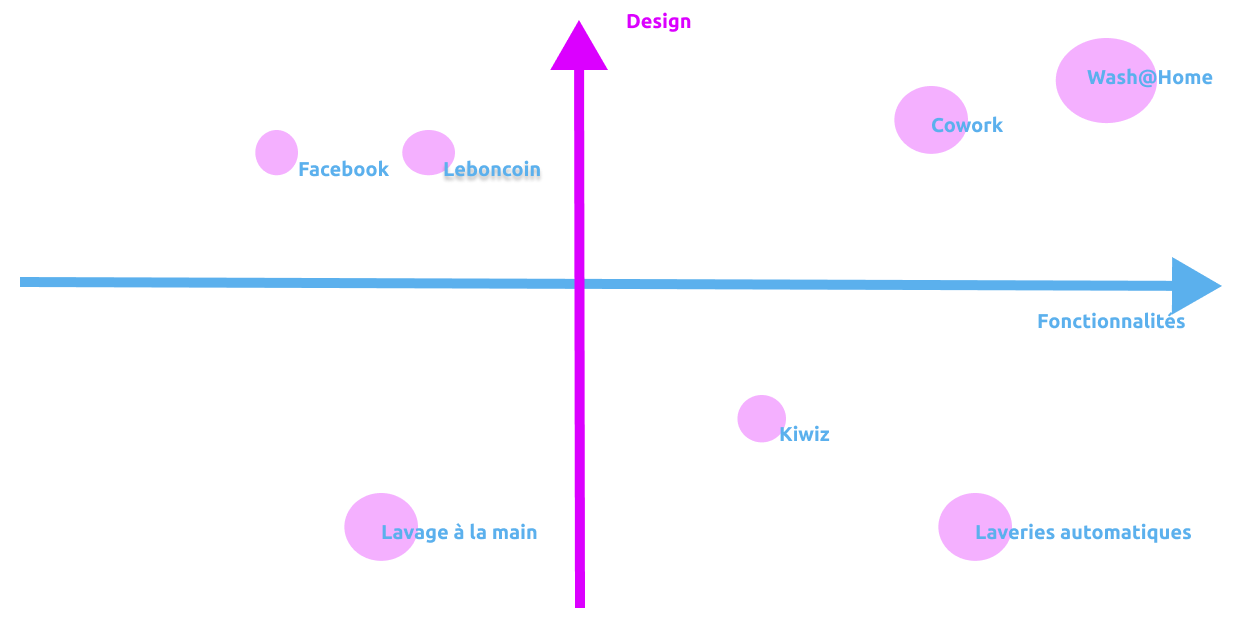
**Développement complémentaires**

* Affichage des meilleurs machines du moment sur la page principale
* Service de chat pour prendre contact via l’application entre Washer et Wearer
* Service de paiement en ligne
* Service de porte-monnaie en ligne
* Envoi des facture par email en pdf
* Visionner les commandes passées
* Ajouter des favoris
* Points fidélités (un baril de lessive offert pour le meilleur washer en partenariat avec ...)

**Présentation de la concurrence**

* Les laveries automatiques: assez chères, pas toujours simples d’accès, parfois sales, il faut amener sa lessive et attendre sur place, pas ouverte 24h24
* [Cowash :](https://www.cowash.fr/)  Pressing à domicile ( tarifs très élevés) de professionnels à particuliers, uniquement dans 68 villes en France, joli design, fonctionnalités : choix par ville,
* [Leboncoin :](https://www.leboncoin.fr/) plus une plateforme de vente que d’échanges de services
* La machine du voisin (plateforme fermée)
* [kiwiz :](https://www.kiwiiz.fr/location-particuliers/electromenager/lave-linge/) location de machine à laver entre particuliers, tarifs élevés, site UI/UX peu adaptée, peu de fonctionnalités, peu d’offres par secteur
* [facebook :](https://www.facebook.com/) groupes communautaires sur les réseaux sociaux, compliqué de trouver ce qu’on veut

**Mapping de la concurrence**



**Analyse SWOT**

| **Strength/Force** | **Weakness/Faiblesse** |
| --- | --- |
| * Design graphique * Simplicité d’utilisation UX/ UI * Prix attractif * Rapide * Local et écolo-friendly * Mobile first * Scalable * consommation collaborative à la mode | * Nombre limité d’utilisateurs au début donc peu de washers donc peu d’offres * Hébergement en nombre d’utilisateurs et poids de la photo de la machine * Prix de l’hébergeur * Prix du service de paiement cb |
| **Opportunities/Opportunités** | **Threats/Menaces** |
| * Pas de concurrent direct * un marché existant en recherche de ce service * un marché potentiellement en développement : 8 français sur 10 ont l’intention d’utiliser la consommation collaborative * facile à mettre en oeuvre | * Passage en direct une fois mis en contact * dégager des marges suffisantes pour faire vivre l’entreprise * sécurisation des données rgpd et paiements * médiation sur les avis * remboursement si service non fournis * responsabilité du linge (le washer) en cas de litige (pas nous) |

**II / Problématiques de communication**

**Synthèse**

Nos analyses ont permis de démontrer que les concurrents principaux ne proposent pas d'offres permettant à la fois de laver son linge au poids dans un rayon proche de chez soi, de particulier à particulier, à un prix abordable.

Comment créer une application web pour mettre en relation des personnes souhaitant laver leur linge et des personnes possédant une machine à laver afin d’implémenter la consommation collaborative sur le secteur du linge?

**III / Stratégie de communication**

**Positionnement**

Key words : attractif, simple, accessible, mobile responsive, dynamique

**Objectifs**

**Cognitif :** Développer la notoriété du site auprès des personnes souhaitant laver leur linge. En parler aux étudiants dans les facs à leur inscription / campagne d’emailing et posts dans les groupes communautaires des villes sur les réseaux sociaux.

**Affectif :** Simple d’utilisation : design épuré, approche UX, inspiré des acteurs majeurs du marché. Rapide, choix de react pour l’affichage dynamique des données. Economie collaborative en vogue.

**Cognatif :** Le porte-monnaie en ligne offert pour le premier mois.

**Cibles**

**Les cibles principales :**

**Les Bringers:** Les utilisateurs de machines à laver :

* les étudiants
* les étudiants étrangers/internationaux
* ceux qui ne peuvent/souhaitent pas investir dans l’achat d’une machine à laver
* ceux dont la machine à laver est tombée en panne
* ceux qui n’ont pas l’espace ou de branchements pour mettre une machine chez eux
* ceux qui ne souhaitent pas aller dans une laverie automatique et attendre 2h que la machine soit finie.
* les minimalistes
* ceux qui souhaitent être écolo et préfèrent louer plutôt que d’acheter ( consommation collaborative)
* les voyageurs qui n’ont pas de machine à laver dans leur location

**Les Washers :** fournisseurs de machines à laver

* N’importe quelle personne qui souhaite un complément de revenu et possède une machine à laver
* ceux qui souhaitent rencontrer des voisins
* des personnes à la retraite

**Le coeur de cible :**

* les étudiants, les jeunes actifs, les consommateurs collaboratifs qui souhaitent laver leur linge et ne possèdent pas de machine.

**Cible secondaires :**

* Toute personne possédant un lave linge et souhaitant un complément de revenu

**IV / Stratégie créative**

**Promesses, preuves, tons et contraintes**

**Notre promesse :** Wash@Home vous permet de trouver facilement un particulier louant les services de sa machine à laver dans votre secteur géographique.

**Preuve :** La plateforme permet en quelques clics d’accéder à des offres de particuliers dans votre secteur géographiques

**Ton :** Le cœur de cible étant les étudiants et jeunes actifs, nous avons déterminé qu’il était préférable d’adopter un ton smart-casual, dynamique et cordial.

**Contrainte:** La conservation des données, le respect de la législation (RGPD), les données sécurisées pour le paiement, le choix du serveur pour la scalabilité et le grand nombre d’offres nécessaires au maillage du territoire français.

**V / Stratégie des moyens**

**Les fonctionnalités du projet (spécifications fonctionnelles) MVP**

**Version 1 :** Le site contient les fonctionnalités les plus fondamentales :

* chercher des machines selon un code postal
* filtrer les résultats
* consulter le détail d’une machine
* sélectionner un horaire
* sélectionner des options : séchage, repassage, pliage, café, transport
* s’inscrire via un formulaire
* se connecter
* réserver une machine sur un horaire avec des options
* envoi de notification à la réservation pour validation par le washer
* confirmation par mail
* Avoir accès aux coordonnées du washer pour le contacter et lui apporter le linge
* ajouter / supprimer / modifier une machine à laver, ses horaires de disponibilité et ses options
* consulter et gérer les reservation côté bringer et washer
* formulaire rgpd / composant pop-up accepter les cookies

**Version 2 :** Le site contient les fonctionnalités annexes suivantes :

* geolocalisation du user
* rechercher des machine en entrant une adresse
* ajouter la notion de distances entre le bringer et les machines
* autocompletion dans les champs de recherche des machines
* visualiser les machines sur une carte à la recherche et sur le détail
* visualiser la machine réservée pour guider le bringer
* ajouter des commentaires et une note en fin de process
* possibilité pour le washer de consulter et répondre aux commentaires/notes
* signaler un commentaire ou un problème
* permettre à un admin de faire de la modération de commentaires

**Version 3 :** Le site contient les fonctionnalités avancées :

* possibilité de chat entre bringers et washers
* possibilité de payer sur l’application
* possibilité de garder les session et cookies
* dark mode

**La liste des technologies utilisées pour le projet (spécifications techniques)**

**Front**

* HTML
* SCSS pour le style
* Javascript / ReactJS : pour les vues, appointment-picker pour le date picker
* Redux : pour la gestion des states et des actions
* APIs : google map, stripe, paypal
* socket.io pour pour le chat
* openCage

**Back**

* NodeJS pour le serveur
* SQL pour la base de donnée
* PostGres
* Sqitch
* ORM
* JSdoc
* JWToken : Token d’authentifications
* API Rest

**MCD**

propose, 11 MACHINE , 0N UTILISATEUR

UTILISATEUR : pseudo, nom, prénom, téléphone , mail, mot de passe, avatar

effectue, 11 RESERVATION , 0N UTILISATEUR

:

MACHINE : capacité, nom, description, code postale, adresse ,ville, latitude , longitude, prix, Picture

correspond, 11 RESERVATION , 0N MACHINE

RESERVATION: dispo, température, prix total

possede, 11 RESERVATION ,0N STATUS

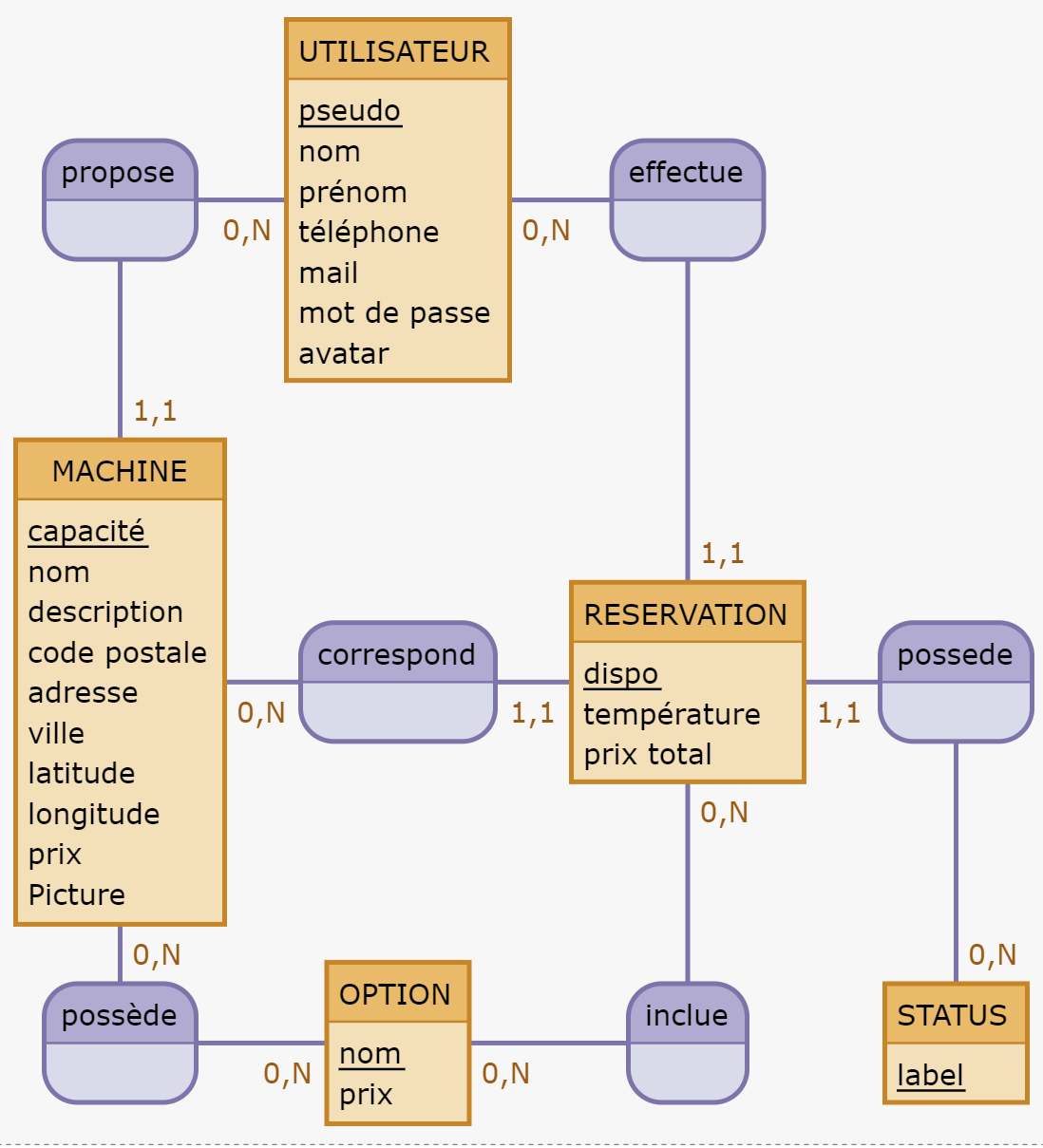
possède, 0N OPTION, 0N MACHINE

OPTION : nom, prix

inclue, 0N OPTION ,0N RESERVATION

STATUS : label

:



**MLD**

BOOKING ( id, température,dispo, #bringer\_id,#washer\_id, #machine\_id, #status\_id)

USER( id, pseudo, firstname, lastname, phone\_number, mail, password, avatar)

MACHINE( id, capacity, name, description, zip, address, city, latitude, longitude price,photo, #user\_id)

OPTION(id, name, price)

STATUS (id,label)

INCLUDE(#reservation\_id, #option\_id)

ANCILLARY (#option\_id, #machine\_id)

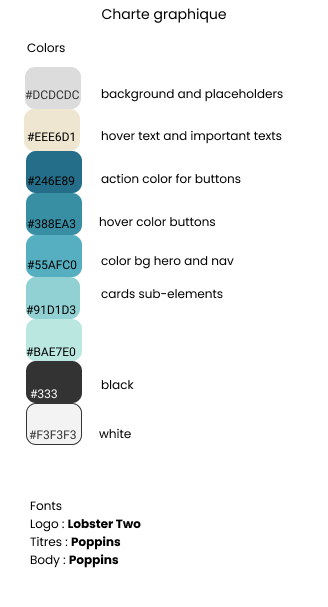
**DICTIONNAIRE DES DONNÉES**

**Lien vers le dico des données :** <https://docs.google.com/spreadsheets/d/1_0qWjRNgg8MgrNe3SXgevlaFDPqHP8QrPTaNsaPKRq4/edit#gid=0>

**VI / Structure**

**Identité visuelle**

#### 

****

**Wireframe :**

<https://whimsical.com/wireframe-KZdtJarxaCpTnN9MfEAAhn>

<https://www.figma.com/file/SH5XgnFPNRFqDAVkLyUiSD/wireframe-washAtHome?node-id=0%3A1>

**Les navigateurs compatibles**

Tous les navigateurs classiques (Chrome, Edge, Firefox...) en mobile first.

**L'arborescence de l'application (le chemin de l'utilisateur)**

**Créer un compte**

* l’utilisateur choisit de s’inscrire ou de se connecter via la barre de navigation ou le menu en mode mobile.
* Il arrive sur un formulaire où il remplit : **table user :** nom, prénom, @ email, @ postale, code postal, nom de la ville, n° téléphone, avatar.
* sur un dernier volet le nouvel utilisateur a le choix d’ajouter une machine ou non
* s’il clique sur ajouter une machine il poursuit l’enregistrement sur un nouveau volet
* sinon le formulaire User est validé => pop up well done !

**Chercher une machine**

* L’utilisateur arrive sur la page d'accueil ou se trouve une barre de recherche
* Il tape le nom d’une ville dans la barre et cliquez sur rechercher
* Il arrive sur une liste de machines avec description simplifiée et classée par : prix, nom de ville, code postale, adresse postale distance (près de l’utilisateur)
* il clique sur l’une d’entre elles et tombe sur la description détaillée avec les horaires disponibles et les options
* Une fois l’horaire et les options choisis, il soumet le formulaire
* Une nouvelle page s’affiche pour confirmer l’envoi du formulaire et indique qu’un email sera envoyé dès le retour du détenteur de la machine
* il est redirigé sur la page d'accueil après 5s

**Soumettre une machine**

* L’utilisateur arrive sur la page d'accueil ou se trouve une barre de recherche
* Il clique sur “Soumettre une nouvelle machine” ou sur un gros +
* Il se retrouve sur une page de formulaire ou rentrer toutes les informations relative à ses services (adresse, horaires de disponibilité, tarifs par options, caractéristiques de la machine)
* il soumet le formulaire et se retrouve sur une page de confirmation
* il est redirigé sur la page d'accueil après 5s

**La liste des routes prévues**

* route POST soumission sign up (/signup)
* route POST sign in (/signin)
* route GET machine (/:machine)
* route POST formulaire de choix de machine et d’options (/:machine/:option)
* route GET formulaire de création de machine (/:machine)
* route POST soumission de formulaire de création de machine (/:machine)
* route DELETE suppression de machine (/:machine)
* route PUT modification de machine (/:machine)

Tableau des routes :

| **Méthode** | **URL** | **Description** |
| --- | --- | --- |
| GET | /searchall | Chercher toutes les machines |
| GET | /search/:postalcode | Rechercher des machines par code postal |
| GET | /machine/:id | Requête des détails d’une machine avec ses options et disponibilité |
| POST | /machine | Création d’une machine et/ou option et/ou disponibilité |
| PATCH | /machine | Modification d’une machine et/ou disponibilité et/ou options |
| DELETE | /machine/:id/:token | Suppression d’une machine avec ses disponibilités et ses options |
| POST | /signup | Envoi du formulaire de création de compte |
| PATCH | /account | Modification du compte |
| DELETE | /account/:token | Suppression du compte avec le token jwt |
| POST | /login | Envoi du formulaire de connexion |
| GET | /autologin | Attends un token en header, envoi les informations user |
| POST | /reservation | Création d’une réservation |
| PATCH | /reservation | Modification de l'état d’une réservation |
| DELETE | /reservation | Suppression d’une réservation |
| POST | /availability | Soumission d’une disponibilite |
| DELETE | /availability | Suppression d’une disponibilte |

**Route front :** juste url, renvoie les pages

**Route back :** plein de trucs dès que c’est un get, post, patch, delete

#### 