ShareCookBook

# Introduction

## Description rapide

Ce projet vise à créer un site internet, plutôt un réseau social dont le but est de créer des recettes de cuisine, afin de les partager avec les autres utilisateurs etc.

## Plan d’action

En premier lieu, il est important de faire le cahier des charges, ensuite un plan d’action des tâches à faire sous forme de diagramme de Gantt, et pour finir, lancer l’exécution des différentes tâches afin de créer ce que contient le cahier des charges

# Cahier des charges

## Introduction

### 2.1.1 Idée initiale

C’est lorsque je me suis rendu compte que ma grand-mère rédigeait tout le temps ses recettes à la main pour nous les envoyer quand on en avait besoin que me vint cette idée : Créer une application dans laquelle toute la famille, amis, et autres pourraient écrire leurs recettes, afin de pouvoir se les partager entre nous et échanger plus facilement sur ce que l'on doit faire et autres.

### 2.1.2 Public cible

Le public cible serai plutôt les personnes âgées, ou les passionnés de cuisine, mais pas nécessairement une personne lambda qui cherche simplement une recette rapide pour le soir, ce serai plutôt une application de passionnés qui discutent, créé, et cuisinent les recettes qu'ils se partagent, entre les recettes de grand-mère, les créations et expérimentations de chacun et les recettes très connues, revisitées par chaque personne voulant rajouter sa petite touche

### 2.1.3 Objectifs

Créer un endroit de partage de passions pour tous les utilisateurs, mais surtout être a l'écoute des utilisateurs qui ont très souvent de très bonnes idées sur ce qu'il serai bon pour l'application en termes de fonctionnalités

### 2.1.4 Vision du produit

Deux utilisations seront possibles, premièrement les personnes aimant simplement la bonne cuisine, qui aimeraient partager leurs recettes à leurs amis / famille, et deuxièmement les personnes aimant expérimenter, car l'application leur permettra de trouver d'autres personnes partageant la même passion, cela leur permettra de s'entre aider ou de trouver de l'inspiration

## Description du projet

### 2.2.1 Fonctionnalités clés

Les fonctionnalités les plus importantes et à créer en priorité dans le cas d’une première version sont :

* Créer et publier des recettes
* Avoir un feed
* Pouvoir commenter et liker des recettes
* Pouvoir effectuer une recherche

Cependant, une liste sera faite après l’exécution de celle-ci afin de planifier les fonctionnalités à implémenter une fois que l’application avec les fonctionnalités de base sera ouverte au grand public

### 2.2.2 Plateformes cibles

Pour le moment, il est prévu qu'une application soit disponible sur mobile, étant donné que la majorité des utilisateurs vont se servir de l'application avec leur téléphone, mais il est prévu d'avoir une application web accessible via n'importe quel appareil. Il faudra alors adapter l’UI pour desktop plus tard

### 2.2.3 Monétisation

Mettre en place un système de publicités, obligatoirement soit en rapport avec des magasins tels que Grand frais, Fresh et autre, soit en rapport avec des ustensiles et autres éléments de cuisine

### 2.2.4 Evolutivité

Une fois l'application sortie, les futures évolutions se baseront massivement sur l'avis des utilisateurs, plus une fonctionnalité ou évolution sera demandée, plus elle sera en haut dans la liste des priorités. Bien sûr, l'équipe de développement et autres peuvent et doivent choisir les plus pertinentes.

## Exigences fonctionnelles

### 2.3.1 Gestion des utilisateurs

**Création de compte :**

Lors de la création de compte, l'utilisateur sera amené à choisir un user Name, qui devra être unique, le code devra donc vérifier en base si cet user Name existe déjà ou non, ainsi que pour l’adresse mail. L’utilisateur sera aussi invité à entrer un Nom, qui ne sera pas obligatoirement unique, ainsi qu’un mot de passe, avec la confirmation du mot de passe.

**Connexion/Déconnexion :**

L’utilisateur pourra alors se connecter grâce à son user Name, et son mot de passe, ainsi qu’une checkbox lui proposant de rester connecté. Il aura aussi la possibilité de se déconnecter depuis la page paramètres.

**Gestion de profil :**

L’utilisateur aura accès à une page profil, ou sera affiché son user Name, son Nom et son mot de passe dans une section, et les recettes qu’il a publié dans une autre section

### 2.3.2 Feed et recherche

**Feed :**

Une page dédiée au "Feed" sera disponible, et sera surtout la page d'accueil. Sur cette page, l'utilisateur pourra trouver les dernières recettes postées les plus pertinentes. Lorsque la fonctionnalité de signets sera disponible, un bouton sur la page d'accueil (Feed) sera disponible pour que l'utilisateur puisse afficher par défaut les recettes qu'il aura enregistrées, au lieu de voir les nouveautés.

**Recherche :**

Sur la page d'accueil (Feed) ou sur une page séparée (l'option sera disponible à deux endroits) l'utilisateur pourra effectuer une recherche par mots clés, en tapant le titre de la recette ou l'intitulé du plat (exemple, taper crêpes donnera à l'utilisateur les recettes de crêpes bretonnes, crêpes classiques, crêpes salées, etc.)

### 2.3.3 Publications

L'utilisateur aura accès à une section "nouvelle recette" qui pourra lui servir à écrire une nouvelle recette, la page sera composée d’un système d’onglet, servant à afficher à la fois la page de création, mais aussi une page de prévisualisation de la recette, dans la pasge de création, il y aura une section informations ou l’utilisateur pourra entrer le titre, le temps de préparation ainsi que le nombre de personnes pour lequel est prévue la recette, en dessous se trouvera une section ingrédients ou l’utilisateur pourra ajouter les ingrédients avec une seule et même commande, les ingrédients seront affichés en dessous, et enfin une dernière section comportant un champ vide supportant du markdown, et bien sur un bouton de validation, et un pour réinitialiser la recette.

# 3. Etude de marché/concurrence

## 3.1 Marché

En ce qui concerne le marché, mes observations, faites sur une centaine de comptes instagram, ainsi qu’une trentaine de comptes twitter, sont que ces comptes postent au moins une fois tous les deux jours une recette de cuisine. Certains posts en réponse mentionnent l’impraticité de ces réseaux.

## 3.2 Concurrence

### 3.2.1 Applications de recettes

Il existe beaucoup d'applications de recettes, parmi lesquelles nous pouvons citer les applications des magasins de type Carrefour et autres qui proposent des applications contenant des recettes à faire avec les produits qui sont disponibles dans leurs magasins

Il existe aussi des applications de partage de recettes mais celles-ci ne sont pas du tout communautaires, les recettes proposées dedans sont uniquement écrites par l'éditeur et ne permettent donc pas une utilisation personnelle, c'est seulement un gros livre de recettes numériques :

* 750g
* Marmiton
* SuperCook
* Jow

La dernière catégorie d'applications destinées à des recettes de cuisine sont les applications servant afin de suivre un régime particulier, l'application proposera à l'utilisateur de choisir son régime alimentaire et cette application va ensuite donner toutes les recettes adaptées aux besoins de l'utilisateur

L'application que je créé étant uniquement communautaire (bien que l'éditeur ait la possibilité de mettre en avant certaines recettes les plus pertinentes), l'utilisation sera donc suffisamment différente de l'utilisation des applications cités au-dessus pour qu'elles ne soient pas concurrentes directe de la nôtre

### 3.2.2 Réseaux sociaux

**Twitter :**

Pour twitter, c'est pour l'instant le concurrent coté réseau social le plus connu pour les recettes, cependant il y a beaucoup d'inconvénients :

Premièrement, l'application n'est pas axée sur le partage de recettes, ce qui fait que bien souvent sur les posts de personnes telles que [Jigmé](https://twitter.com/ClichesDeJigme) il y a beaucoup d'utilisateurs qui crachent leur haine, ce qui rend l'application très peu agréable a utliser

Deuxièmement, twitter n'était pas fait pour, ça rend le partage de recettes compliqué, soit par des posts de vidéo, soit par de l'écrit mais cela n'est pas pratique déjà à cause du nombre de caractères limité par post, mais aussi par le partage de photos ou la mise en forme du texte.

Nous pouvons aussi citer le fait que si les ingrédients sont au début de la page, il faut remonter puis redescendre à chaque fois, alors que dans notre application il suffira d'appuyer sur un bouton pour faire apparaitre une fenêtre à gauche contenant les ingrédients et leur quantité, idem pour le calcul des quantités, impossible sur les réseaux sociaux de faire ça de manière personnalisée

**Threads :**

Threads a bien sur tous les inconvénients de twitter étant donné que c'est quasiment un copier-coller, cependant il y a un défaut que je trouve moins c'est la haine.

Threads a été envahi de personnes bienveillantes pour la plupart, ce qui fait que notre post de recettes sera moins souvent confronté à de la haine gratuite

**Instagram :**

Instagram est encore pire que les deux autres.

Premièrement les posts sont soit des photos soit des vidéo, impossible donc de poster la recette écrite sans faire de montage photo qui vont forcer les utilisateurs a swiper à chaque fois qu'ils arrivent au bout de la photo, mais cela limite aussi la longueur de la recette.

Et mettre le contenu de la recette en description de la photo c'est tout autant une mauvaise idée, bien sur le texte, comme pour les deux autres, ne sera pas formaté ni rien, mais en plus la section sera limitée en caractère mais sur Instagram, impossible de rajouter un autre post en dessous pour compléter.

# 4. Réponse au cahier des charges

## 4.1 UI et UX

Toutes les maquettes seront faites grâce à figma, afin de ne pas perdre de temps lors de la création de l'application côté code, étant donné que tout le design sera déjà créé

### 4.1.1 UI

### **Style graphique :**

Pour cette application, il faudra un style graphique plutôt simpliste et très lisse avec quelques légers dégradés de couleurs pour mettre un peu de texture / profondeur, des boutons plutôt classiques rectangulaires avec des arrondis sur les bords, sans que les côtés soient des demi-cercles pour autant.

Il est inutile de mettre des effets de "Hover" étant donné que l'application sera pour le moment disponible uniquement sur mobile. Les effets de hover de autres mouvements seront implémentés dans la version web, accessible sur Desktop.

Selon moi, il est aussi inutile de mettre des effets de mouvements sur les boutons ou autres, il faut rester dans une interface lisse sans trop d'effets qui pourraient peut-être un peu trop renforcer le côté application / futuriste, le but étant de rester plus proche d'un cahier de notes connecté à internet qu'a une application dans un téléphone.

### **Couleurs :**

En ce qui concerne les couleurs, j'opterai plutôt pour des couleurs style vert pâle et blanc cassé, pour amener un côté naturel bien que l'interface soit dans un style plutôt sobre / moderne / simpliste.

Les couleurs pourraient reprendre l'idée des couleurs de l'application "Seek" (application servant à scanner et déterminer l'espèce d'animaux, insectes et plantes à travers le monde (ayant surtout pour but de localiser des espèces à travers le monde)), les couleurs seraient celles-ci :



Le violet sera utilisé pour ajouter du contraste, ou différencier deux boutons plutôt que d’utiliser du rouge

### 4.1.2 UX

### **Bottom bar :**

Tout se passera dans une bottom bar, qui comportera 5 boutons, un central pour accéder à la création d'une nouvelle recette, un tout à gauche qui sera le Feed, la page d'accueil, un bouton entre les deux qui sera le bouton de recherche, un bouton tout à droite permettant d'accéder aux paramètres, ainsi qu'un bouton entre les paramètres et la création d'un nouveau post, un bouton permettant d'accéder à son profil

### **Liste des recettes :**

Dans le feed ou dans la page de recherche, les recettes seront sous forme de liste, chaque élément de la liste sera dans une boite divisée en trois parties, la première sur fond violet comportera le nombre de likes, le temps de préparation et le nombre de personnes, et les deux autres parties seront une seule et même partie avec le titre de la recette

### **Recette :**

Lors de l'ouverture d'une recette, la page de la recette sera composée :

D'un header contenant le titre de la recette, ainsi que juste en dessous, le temps, nombre de personnes et de likes dans trois sections différentes

Du corps de la page, ou sera affiché le texte de la recette, grâce à un interpréteur, transformant le Markdown en HTML,

Puis tout en bas de page, un Footer contenant, si on le divise en 5, deux parties afficher l’username de la personne ayant posté la recette, un bouton de like, un bouton permettant d’afficher et de poster des commentaires, et un bouton, pour l’instant non fonctionnel, servant à enregistrer la recette dans son profil

## 4.2 Monétisation

Travaillant actuellement chez Prosol, exploitant des magasins grand frais et Fresh, j’ai entamé des discussions sur un possible financement de leur part contre des publicités pour les magasins dans l’application, pour le moment l’équipe chargée du marketing et des publicitées est en train d’étudier la question avant de me dire si c’est possible ou pas, et discuter des objectifs de trafic et du financement.

## 4.3 Architecture

### 4.3.1 Frontend

Pour le frontend, le large choix m'a fait me poser beaucoup de questions, entre les Framework tels que react, angular, vue... et les langages tels que PHP offrent un choix varié.

Mon choix s'est alors tourné vers les Framework.

Premièrement ayant pour but de pouvoir utiliser les packages Node dont j'ai plus l'habitude de me servir que ceux de composer de PHP.

Deuxièmement car je pouvais alors me servir de React, Framework que j'ai le plus l'habitude d'utiliser, bien que Angular était un choix pertinent, J'ai donc fait le choix de React, avec TSX

En ce qui concerne les packages installés, j'ai fait le choix d'utiliser les icônes de Bootstrap, que j'ai utilisé dans le package "react-icons" qui intègre les icônes directement dans des composants, "react-markdown" me permettant d'avoir un composant prenant en paramètre une variable contenant un string, qui le transforme en markdown afin d'afficher les recettes correctement, les packages classiques tels que react-dom, react-router-dom, etc...

### 4.3.2 Backend

Le choix pour le backend a été beaucoup plus rapide, étant donné que je n'avais encore jamais fait de backend, mais aussi que le backend n'avais pas besoin de faire beaucoup de traitement, mais seulement servir pour renvoyer les données de la base, mon choix s'est porté sur un backend en node, écrit en js.

J'ai utilisé bien sur des packages node tels que express pour la connexion à la base de données, cors afin de cloisonner l'accès (aux requêtes, et donc à la base de données) uniquement à l'adresse ip du vps, ainsi que Winston pour me créer un fichier de logs afin de savoir ce qui s'est passé en cas de bugs et autres problèmes remontés par les utilisateurs (même si le back ne considère pas ça comme des erreurs).

### 4.3.3 Base de données

Le choix de la base de données était initialement porté sur MySQL, car c'était le seul SGBD que j'avais déjà utilisé. Cependant, après avoir utilisé PostgreSQL dans le cadre professionnel, et ma base de données n'étant pas encore très imposante, j'ai rapidement changé de SGBD afin de passer sur PostgreSQL, dans le but de m'en servir plus facilement dans le cadre professionnel, mais aussi dans le but de me servir plus que de celui-ci dans le reste des projets que j'entreprendrai (tout comme le backend en js, qui après utilisation, correspond totalement à ce dont j'ai besoin)

### 4.3.4 Hébergement

Pour l'Hébergement, j'ai choisi Hostinger, qui va me permettre de gérer à la fois le nom de domaine et le VPS, mais qui va aussi me permettre de créer un sous domaine dédié à mon application, j'avais le choix avec beaucoup d'autres solutions mais Hostinger m'a semblé être un choix judicieux car un ami m'a déjà beaucoup parlé de cette solution qu'il utilise personnellement alors je savais qu'en cas de soucis je pourrais avoir de l'aide, même si je n'en ai pas eu besoin

# 5. Développement

## 5.1 Frontend

Le frontend se compose, excepté pour les écrans de connexion et de création de compte :

D'un header dont le contenu diffère en fonction de l'écran, qui accueille notamment le titre de l'écran sur lequel nous nous trouvons, et un bouton retour si nécessaire

Du corps de l'écran, qui dépend a 100% de l'écran sélectionné mais qui respecte quand même le style de base de l'application

Une bottom bar contenant 5 boutons, servant à naviguer entre les écrans les plus importants tels que :

* L'écran d'accueil : contenant la prévisualisation des dernières publications postées de la plus récente a la plus ancienne
* L'écran de recherche : contenant une barre de recherche, avec en dessous les résultats, ou un message si le résultat de la recherche est vide
* L'écran de création : composé de deux onglets, le premier sert à entrer toutes les informations relatives à la nouvelle recette que l'utilisateur désire poster, et le deuxième onglet servant à prévisualiser sa recette, reprenant exactement le même style que l'écran affiché lors du clic sur une prévisualisation de recette
* L'écran de compte : affiche en premier lieu les informations relatives au compte sur lequel nous sommes connectés, tels que le nom d'utilisateur, l'adresse mail ainsi que le nom complet, et en second lieu la prévisualisation des recettes que l'utilisateur a posté dans l'ordre de la plus récente a la plus ancienne
* L'écran de paramètres : ne servant pour l'instant qu'à avoir accès à un bouton "Déconnexion"

La bottom bar grise légèrement le bouton correspondant à l'écran sur lequel nous sommes, excepté pour deux écrans, les seuls deux écrans qui ne sont pas accessibles depuis la bottom bar :

* L'écran de recette : affiché après le clic sur une prévisualisation de recette, l'écran est composé d'un bloc d'informations au-dessus, contenant les informations : titre, temps de préparation, nombre de personnes ainsi que le nombre de likes, suivi d'un espace ou la recette est renseignée, d'abord une liste d'ingrédients, ensuite le corps de la recette, et juste au-dessus de la bottom bar, un autre bloc contenant le nom d'utilisateur de la personne ayant posté la recette, et trois boutons d'action : liker, commenter et enregistrer (malheureusement le bouton enregistrer ne fonctionne pas encore)
* L'écran de commentaires : accessible depuis l'écran de recette, il affiche tous les commentaires les uns à la suite des autres, le plus récent en bas, à la manière d'un chat, les commentaires contiennent le nom d'utilisateur de la personne ayant commenté, ainsi que le contenu du commentaire

## 5.2 Backend

Le backend se compose de plusieurs fichiers :

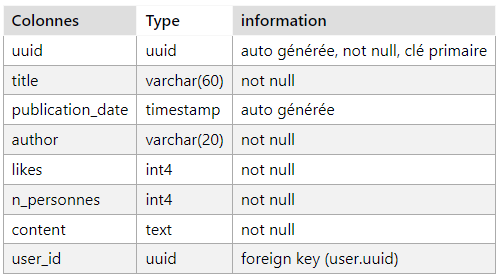
* index.js, contenant tous les chemins d'appel, et faisant la liaison avec les deux fichiers de méthodes
* PublicationQuerries.js, contenant toutes les requêtes relatives aux publications tels que les likes, le post, etc.
* UserQuerries.js, contenant toutes les requêtes relatives aux utilisateurs tels que connexion, création de compte, accès aux informations
* Logger.js, contenant les méthodes pour mettre des informations et erreurs dans le fichier de log, avec la date, le nom de la méthode, et si elle a réussi, ou si elle a levé une erreur avec le contenu de l'erreur dans ce cas
* DateUtils.js : servant simplement à récupérer la date actuelle ainsi que traiter la donnée reçue pour lui donner ce format : dd-MM-yyyy HH:mm:ss

## 5.3 Base de données

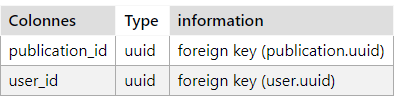
La base de données comporte actuellement quatre tables :

* publications
* users
* comments
* publication\_likes

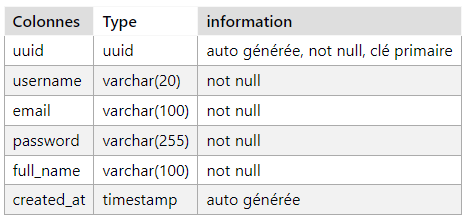
Publications :



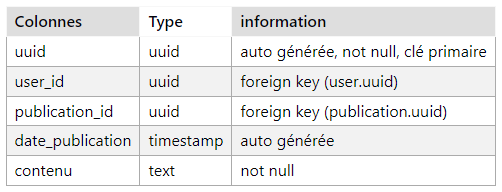
Publication\_likes :



Accounts :



commentaires :



## 5.4 Hébergement

Une fois mon compte Hostinger créé et l'achat d'un vps effectué (j'ai pris le moins cher, ayant pas encore besoin de beaucoup de ressources, le premier plan me suffisait amplement), j'ai alors choisi le système d'exploitation, Ubuntu 23.04 64bit sans interface graphique, étant donné que mon vps n'a pas beaucoup de ressources je me suis dit que ça ne servait à rien d'avoir une interface.

Alors, j'ai ajouté les fichiers de mon application, backend et frontend, dans le dossier root afin de lancer les deux. J'ai utilisé screens afin de gérer les deux applications en même temps.

En ce qui concerne le backend, je l'ai simplement lancé comme je le faisait sur ma machine, en faisant node index.js, puis je l'ai laissé tourner dans mon screen.

Pour le frontend, j'ai build mon application, et envoyé les fichiers du build dans le dossier /var/www/html.

En ce qui concerne le nom de domaine, j'ai acheté ebasson.fr depuis l'interface de Hostinger, avant de créer un sous domaine pointant sur l'adresse ip du vps que je venais d'installer

Après avoir lié mon nom de domaine a mon vps, j'ai suivi un tutoriel pour installer et utiliser correctement Nginx afin qu'il redirige tout trafic http du port 80 vers les fichiers de mon application, dans le dossier /var/www/html

J'ai ensuite automatisé grâce à une fonction .bashrc afin de récupérer automatiquement le dernier commit, et refaire toute la séquence afin de mettre à jour l'application, ce qui me permet de faire des mises à jour et correctifs très rapidement après les retours des premiers utilisateurs

En ce qui concerne la base de données, j'ai installé PostgreSQL sur la machine et configuré correctement le fichier ph\_hba.conf pour autoriser mon adresse ip personnelle à accéder aux bases, afin de me connecter avec dbeaver, rendant les évolutions sur la base de données aussi simple que la mise à jour de l'application de production