



23 MAI 2021

# COOK, EAT AND CONNECT

Belorgey Rafaele (32004862)

Cunin Julie (32004511) ,

Herbin Aurélien (31805648)

Jacob--Guizon Xavier (31617773)

---

M1 SCIENCES COGNITIVES  
Technologies du web pour  
l'innovation





# Accès au projet

[https://github.com/blondinet/Technologie\\_innovante\\_projet](https://github.com/blondinet/Technologie_innovante_projet)

Le projet final est contenu dans le dossier:  
projet\_reuni





# Sommaire

- 1) Organisation
- 2) Définition de l'application
- 3) Définition des priorités (MosCow)
- 4) Choix de l'identité de l'application
- 5) Maquettes
- 6) Fonctionnalités implémentées dans notre prototype
- 7) Spécifications techniques
- 8) Conclusion





# Organisation

Pour concevoir cette application, nous avons décidé d'adopter une méthode agile avec des itérations et réunions chaque semaine depuis le lancement du projet.

Notre trello est disponible à l'adresse suivante :

<https://trello.com/invite/b/Lmw50D1s/98730dee0f924342525781b677e508ec/projet-technologie-innovante>



# DÉFINITION DE L'APPLICATION

Notre application a pour objectif de proposer des recettes basées sur des données sur la santé physique d'un individu afin de lui proposer une liste de plats adaptée à ses besoins énergétiques.

Une autre fonctionnalité de notre application permet de proposer des plats à l'utilisateur à partir d'une liste d'aliments en sa possession. Il sera également possible d'enregistrer une liste d'aliments à acheter.

L'application pourra aussi mémoriser les choix de l'utilisateur pour lui proposer des plats qui s'adaptent au mieux à ses préférences. Celles-ci pourront également être régulées par le biais d'un filtre de préférences. Ce filtre permet également d'anticiper les possibles spécificités alimentaires : régime végétarien, allergie...

Une option "frigo connecté" pourra également être proposée par la suite afin de pouvoir automatiquement récupérer la liste des aliments présents dans celui-ci.



# DÉFINITION DES PRIORITÉS

Nous avons utilisé la méthode Moscow afin de prioriser les besoins lors du développement de l'application :

## 1. **Must**, doit être fait :

- Page de compte dans laquelle on peut créer un compte, se connecter ou voir ses informations.
- Profil avec :
  - Sexe
  - Âge
  - Taille
  - Poids
- Demander les ingrédients que possède l'utilisateur chez lui.
- Statistiques récoltées grâce à une montre connectée :
  - Exercice physique : le nombre de calorie dépensée par jour
- Une page de recette
- Une connexion à une base de données reliée à l'application
- Une connexion à la montre connectée par une API Rest (ou autre)



## 2. **Should**, devrait être fait dans la mesure du possible :

- Page sur la gestion des données personnelles (acceptation des conditions générales)
- Lors de la présentation de la liste des plats :
  - Système de "J'essaye ça"
  - Système de "Plus tard..."
  - Accès aux détails d'un plat (avec fenêtre modale qui montre les ingrédients) :
    - Temps de préparation
    - Ingrédients
    - Ustensiles
    - Apports nutritifs :
      - Calories
      - Protéines
      - Lipides
      - Glucides
      - Vitamines
      - Etc.
- Proposer un système de "like/dislike" pour les plats qui ont été choisis auparavant pour :
  - Identifier les préférences gustatives ("Like")
  - Journal de ses préférences modifiables : repas choisis ("Like" et "Dislike") ; une préférence peut être modifiée ou juste supprimée.
  - Journal alimentaire hebdomadaire
- Sélection d'un mode de vie (compte) : régime végétarien, sans gluten, allergie, etc.
- Proposer les recettes par rapport à des ingrédients que la personne possédera.



**3. Could**, pourrait être fait dans la mesure où cela n'a pas d'impact sur les autres tâches :

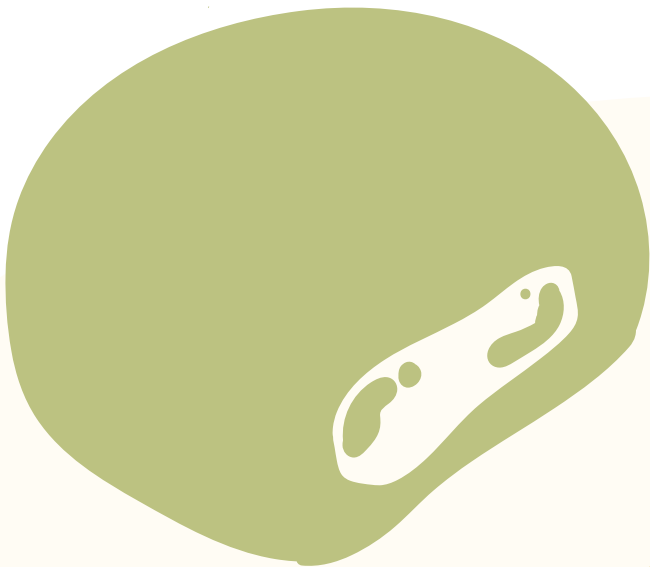
- L'utilisateur remplit une liste de courses à acheter. Avec les "Plus tard...", l'appli va proposer des ingrédients à l'utilisateur pendant la rédaction de sa liste de courses : il pourra accepter ou refuser.
- Proposer une version anglophone en plus de la francophonie de base
- Proposer une version jour/nuit de l'interface
- Rendre l'application plus accessible aux daltoniens (par le biais de jeux/profils de couleurs différents pour l'entièreté de l'application selon le type de daltonisme, le profil apparent modifiable de la même manière que l'on modifierait une langue, voir : <https://www.alsacreations.com/astuce/lire/811-couleurs-accessibilite-contraste.html>)
- Rendre l'application plus accessible aux dyslexiques (par le biais d'une police appropriée) et malvoyants, voir : <https://www.reseau-canope.fr/agence-des-usages/les-sites-web-plus-accessibles-aux-malvoyants-et-aux-dyslexiques.html>
- Objectif (objectif poids : défaut / normal / perdre / gagner)
  - Une page de formulaire
  - Un chemin de complétion
  - Une confirmation (une page de résumé à l'utilisateur)



# DÉFINITION DES PRIORITÉS

4. **Won't**, ne sera pas fait cette fois, mais sera fait plus tard :

- Relier à un frigo connecté pour les aliments au frais, remplaçant la nécessité que ces aliments en question soient entrés à la main par l'utilisateur-même
- Scan des produits permettant à l'utilisateur de n'avoir pas à taper le nom des produits



# CHOIX DE L'IDENTITÉ DE L'APPLICATION

Nous avons choisi de réaliser l'application dans les tons roses, gris et bleus car ce sont des couleurs dynamiques ponctuées d'une pointe de délicatesse qui permettent d'avoir un contraste suffisant.

Nous avons également choisi d'appeler notre application Cook Eat and Connect car c'est un nom qui n'existe pas actuellement et qui rappelle les trois piliers de notre application : la cuisine, manger des plats savoureux et connectés à notre santé.

De plus, le rythme ternaire nous a semblé important tout comme le fait de choisir un nom anglophone dans l'idée d'ouvrir l'application à l'international.

Concernant notre logo, nous nous sommes rapidement dirigés vers l'idée de vouloir représenter à la fois le repas (avec la fourchette), les fruits et légumes (avec la fraise), la santé (avec le cœur que l'on peut voir à travers la fraise) et les données (avec le signal traversant la fraise)

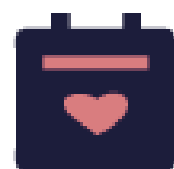
---

## NOS LOGOS ET VISUELS

L'ensemble des visuels/icônes a été réalisé grâce à Illustrator

*Cook,  
Eat & Connect*


---



# MAQUETTES

L'ensemble des maquettes a été réalisé grâce à Figma

Cook,  
Eat & Connect




**Bienvenue**

☐ Se souvenir de moi

[Mot de passe oublié ?](#)

ou

Cook,  
Eat & Connect



**Mot de passe oublié ?**

Communiquez-nous votre adresse email liée à votre compte et nous vous enverrons un mail avec un lien magique vous permettant de réinitialiser votre mot de passe.

ou

[Annuler](#)


Cook,  
Eat & Connect



**C'est bien vous !**

Merci d'avoir suivi le lien magique. Vous pouvez à présent renseigner un nouveau mot de passe !

Cook,  
Eat & Connect



**Inscription**

Devenez membre de Cook, Eat and Connect et profitez de recettes saines sur mesure !

En cliquant sur **Je deviens membre !** vous confirmez avoir lu et approuvé les [conditions d'utilisation](#).

Cook,  
Eat & Connect




**Mon profil**

▼

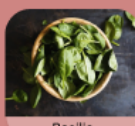
ou

[Renseigner ultérieurement](#)


Cook,  
Eat & Connect



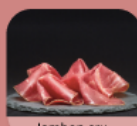
**Ma liste**




Basilic




Pâtes




Jambon cru



Choux de Bruxelles



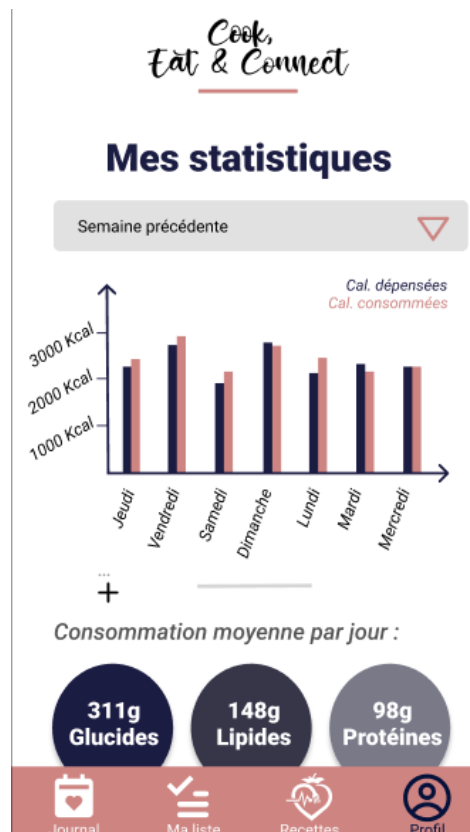
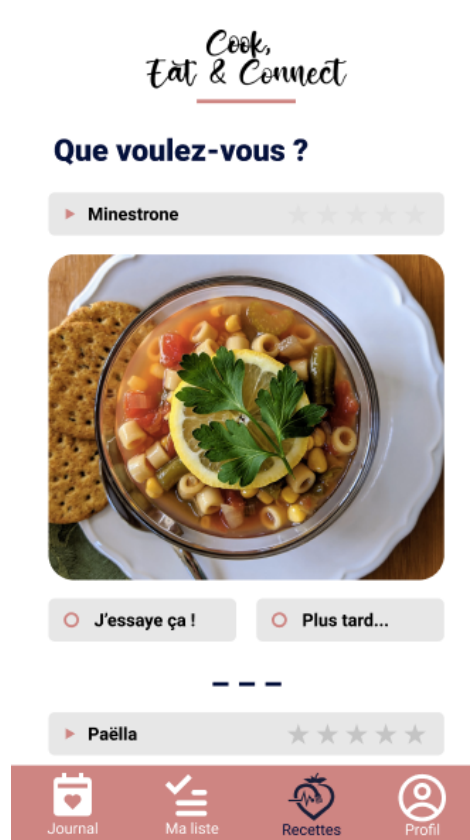
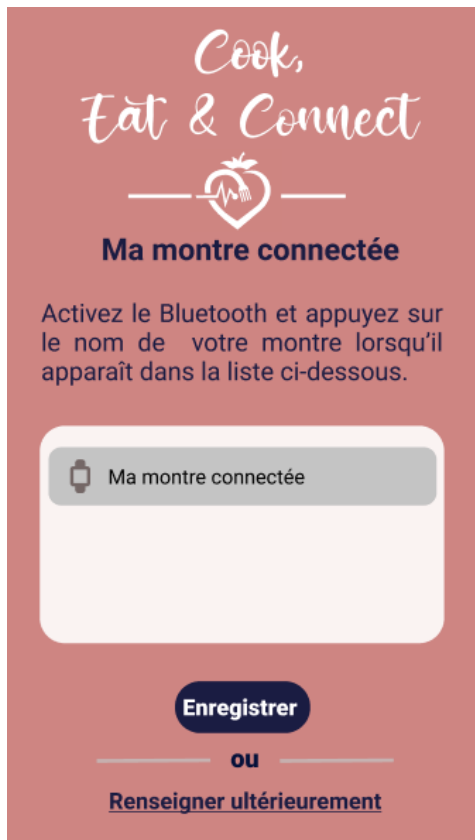
Piments



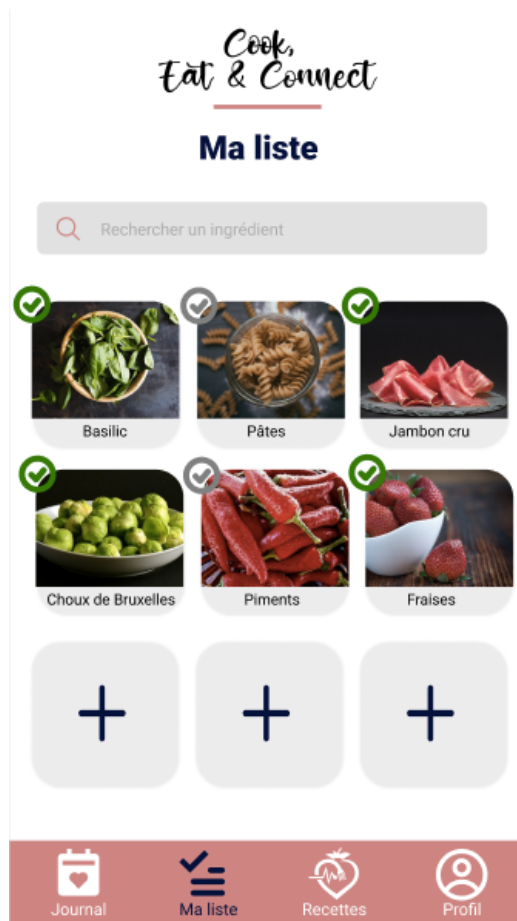
ou

[Renseigner ultérieurement](#)

# MAQUETTES



# MAQUETTES



# FONCTIONNALITÉS IMPLÉMENTÉES DANS NOTRE PROTOTYPE

## PAGES "SIGN IN" (CONNEXION) ET RÉCUPÉRATION MOT DE PASSE

## PAGE "SIGN UP" (INSCRIPTION GÉNÉRALE + PROFIL+ LISTE+MONTRE )

Après avoir renseigné un email et son mot de passe l'utilisateur arrive directement sur une page où il peut renseigner son profil ( date de naissance, sexe, taille, régime spécifique...) mais il peut décider de le renseigner ultérieurement. La page où il peut entrer la liste d'aliments qu'il possède est ensuite affichée mais il peut également le renseigner ultérieurement. Enfin, l'utilisateur arrive sur la page où il peut détecter sa montre en Bluetooth et arrive ensuite au coeur de l'application où des recettes lui seront recommandées.

## PAGE PROFIL

Sur la page profil, l'utilisateur peut accéder à ses statistiques, ses objectifs (non implémenté), ses informations (non implémenté), et aux paramètres.

## PAGE PARAMÈTRES

La page paramètres est prête au niveau HTML et CSS, et dans le futur il sera simplement question de lui associer du code JavaScript pour mettre en action les interrupteurs à bascule (toggle switch) permettant de basculer hypothétiquement d'un mode dit "jour" à un mode dit "nuit" pour l'un, et d'une version francophone de l'application à une version anglophone de celle-ci pour l'autre, ainsi que les divers boutons dirigeants pour trois d'entre eux vers du simple texte et pour le restant vers un formulaire de contact.

## PAGE "HISTORIQUE DES CALORIES/ STATISTIQUES

L'utilisateur peut voir l'historique des calories dépensées ou consommées sur la page profil dans mes statistiques. Il peut y découvrir ses calories dépensées et consommées dans les 7 jours précédents, les 4 semaines précédentes), ou encore sur les 12 mois précédents.



# FONCTIONNALITÉS IMPLÉMENTÉES DANS NOTRE PROTOTYPE

## PAGE LISTE D'ALIMENTS

L'utilisateur peut entrer la liste d'aliments qu'il possède afin de pouvoir voir par la suite des recettes adaptées à ce qu'il possède

## PAGE RECETTES

La page recette propose une liste de plats, suivant d'abord les besoins diététiques de l'utilisateur, puis ses préférences hypothétiques. La page présente un plat, ainsi que 2 boutons : ils font remplacer ce plat par un autre de la liste ; le premier bouton va en plus enregistrer le plat dans la section "Journal". Dans tous les cas, quand la liste des plats a été parcourue en entier, la page boucle vers le début (en refaisant une nouvelle liste).

Pour former cette liste, le programme passe par 2 phases : une première phase où il sélectionne les plats conseillés à l'utilisateur pour ses besoins énergétiques et une seconde phase où il trie les plats dans l'ordre décroissant vers ce que l'utilisateur apprécierait le plus. La première phase va alors calculer les besoins énergétiques de la personne à l'aide de son métabolisme de base : il se calcule suivant sa taille, son poids et son sexe ; le métabolisme indique le nombre de Kcal qu'une personne a besoin d'avoir par jour et comme il ne mange que 2 fois par jour, cette valeur est divisée par 2, puis additionnée au nombre de Kcal dépensé (donné issu d'une montre connectée, déduite par son nombre de pas). Suivant cette valeur, le programme parcourt la liste des plats dans sa base de données : chaque plat se présente par une liste d'ingrédients ; chaque ingrédient indique le nombre de protéines, glucides et lipides qu'il contient et ensemble, ils permettent de déduire le nombre de Kcal que cet ingrédient possède et avec l'ensemble de ses ingrédients, le programme déduit le nombre de Kcal que contient un plat.

# FONCTIONNALITÉS IMPLÉMENTÉES DANS NOTRE PROTOTYPE

## PAGE RECETTES (SUITE)

Une fois la sélection des plats qui rempliront au mieux les besoins énergétiques de l'utilisateur, les plats sont triés en fonction de leur préférence hypothétique : quand l'utilisateur donne son appréciation sur un plat, cette dernière va en fait caractériser l'appréciation des ingrédients de ce plat ; pour deviner ou retrouver l'appréciation d'un plat, il suffit de faire la moyenne des préférences des ingrédients qui constitue ce dernier. La valeur de préférence va aussi dépendre de la durée de préparation d'un plat : plus elle est courte, meilleure la préférence doit être.

Ainsi, le programme n'a plus qu'à afficher le premier plat de cette liste, puis avec les 2 boutons parcourir cette liste.

Pour finir, chaque plat possède un bouton (par encore implémenté) : il permet d'afficher par l'intermédiaire d'une fenêtre modale, des données sur ce plat comme ses ingrédients, comment la réaliser, etc.







# SPÉCIFICATIONS TECHNIQUES

## STATISTIQUES

Les statistiques sont illustrées à l'aide de la librairie chart.js, permettant ici de créer des graphiques en courbes. Les 3 différents graphiques (semaine, mois, année) sont affichés alternativement selon quelle option est sélectionnée dans une liste déroulante. Le code JavaScript va utiliser la date du jour établie selon le fuseau horaire dans lequel est située la personne pour déterminer quels sont les 7 jours précédents, les 4 semaines précédentes, et les 12 mois précédents, ceux-ci se mettant à jour à chaque changement de date.

## BASE DE DONNÉES

La Base de données utilisées est Postgres, en l'état, elle ne comprend rien et a juste été installée. Avec plus de temps, les tables auraient compris des données utilisateurs et de paramétrage, ainsi que des ingrédients et des identifiants pour relier plats, ingrédients et calories à une API

## UTILISATION D'UNE API REST

L'utilisation d'une API connectée à notre application aurait pu être implémentée avec plus de temps. Nous aurions procédé par l'ajout d'un fichier texte avec des informations JSON. Ce fichier aurait compris des informations que nous avons déjà définies et qui sont inspirées d'API réelle comme les data sets présents à ces adresses :

[https://www.data.gouv.fr/fr/datasets/table-de-composition-nutritionnelle-des-aliments-ciqual/#\\_](https://www.data.gouv.fr/fr/datasets/table-de-composition-nutritionnelle-des-aliments-ciqual/#_)

<https://www.kaggle.com/vaishnavivenkatesan/food-and-their-calories>

Ce fichier aurait été lu par react et utilisé pour simuler une API et pour fournir en temps réel les interfaces en données ainsi que réaliser des traitements de données afin de comprendre les calories gagnées et perdues de l'utilisateur.



# CONCLUSION

En conclusion, malgré le fait que notre groupe soit composé majoritairement de novices n'ayant peu ou jamais programmé dans ces langages, nous avons réussi à nous entraider et à ce que l'ensemble du groupe participe au code HTML, CSS et React.

L'organisation a été facilitée grâce à un Trello et les maquettes ont été réalisées suite à une définition précise du projet. Ce projet nous a également permis d'expérimenter une gestion de projet agile, de découvrir le fonctionnement d'une API et d'une base de données.

Ainsi, malgré le fait que le projet ne soit pas complètement abouti, en raison d'un manque important de temps (et parfois de compétences), nous avons réussi à réaliser la majorité de nos objectifs pour ce prototype.

ETUDE DU FONCTIONNEMENT  
D'UNE API ET D'UNE BASE DE  
DONNÉES (NON IMPLÉMENTÉ PAR  
MANQUE DE TEMPS)



GESTION AGILE DU PROJET



DÉCOUVERTE DE REACT



APPRENTISSAGE DU CSS ET HTML

