



RAPPORT DE PROJET

Semestre 1 - Conception d'application

Durée : 3 mois

HealthyCore

(Application mobile de sport et de nutrition)

Réalisé par :

- EL MOURABIT Badre
- ROBLES Nicolas
- CATTAROSSO DARTIGUELONGUE Thomas

Master 1 Informatique Parcours Full Stack Web Mobile
2024-2025

Sommaire

Introduction.....	3
I. Contexte et analyse de marché.....	4
A. Objectifs de l'application.....	4
B. Cibles de l'application.....	4
C. Analyse de la concurrence.....	5
II. Besoins non fonctionnels.....	5
A. Ergonomie et expérience utilisateur.....	5
B. Facilité d'utilisation et facteurs humains.....	6
C. Exigences de sécurité et conformité RGPD.....	7
III. Besoins fonctionnels et conception.....	8
A. Diagramme des cas d'utilisation.....	8
B. Scénarios des cas d'utilisation de la version 1.....	9
Cas d'utilisation 1 : S'inscrire.....	9
Cas d'utilisation 2 : Se connecter.....	9
Cas d'utilisation 3 : Suivre sa nutrition.....	10
Cas d'utilisation 4 : Suivre un programme de sport.....	10
Cas d'utilisation 5 : Découvrir des recettes.....	10
Cas d'utilisation 6 : Découvrir des programmes de sport.....	11
Cas d'utilisation 7 : Compléter ses objectifs quotidiens.....	11
Cas d'utilisation 8 : Suivre son évolution.....	11
Cas d'utilisation 9 : Administrer les comptes utilisateurs.....	12
Cas d'utilisation 10 : CRUD des programmes / séances de sport.....	12
C. Maquettes UX/UI.....	12
D. Schéma relationnel.....	13
IV. Contraintes et vision techniques.....	14
A. Contraintes techniques.....	14
Compatibilité multi-plateforme.....	14
Gestion des fonctionnalités matérielles.....	14
Optimisation de la taille de l'application.....	15
Simplicité dans la gestion des interfaces utilisateurs.....	15
Gestion de requêtes simultanées.....	15
Intégration d'une API externe.....	15
B. Choix des technologies.....	16
Front-end : React Native et Expo.....	16
Back-end : Node.js, Express et PostgreSQL.....	16
Conclusion et planification.....	18
Webographie.....	20
Annexes.....	24
Annexe 1 : Questionnaire d'onboarding.....	24
Annexe 2 : Calculs techniques et nutritionnels.....	25

Introduction

Dans le cadre du premier semestre de Master 1, nous devions concevoir un projet de développement d'application. Nous avons choisi de nous diriger vers le domaine du sport. Le marché des applications mobiles de nutrition et de sport est un énorme pan de l'industrie mobile. En effet, il existe une grande quantité d'applications dans ce secteur. Malgré cette abondance, seules quelques-unes parviennent à intégrer efficacement sport et nutrition dans une seule et même application. Cela conduit souvent les débutants à jongler entre plusieurs applications pour reprendre leur santé en main [1], ce qui peut rapidement entraîner une perte de motivation.

C'est de cette observation qu'est née l'idée de **HealthyCore** : créer une application qui rassemble sport et nutrition de manière cohérente et pratique, offrant ainsi une solution complète pour accompagner les utilisateurs dans leurs objectifs de santé.

Le projet HealthyCore est porté par une équipe de trois personnes : Badre, Nicolas et Thomas. Tout au long de ce premier semestre, nous avons mis en commun nos connaissances et compétences pour concevoir et définir les bases de ce que nous développerons au second semestre. En nous inspirant de nos expériences personnelles en tant qu'utilisateurs d'applications mobiles, nous avons identifié les caractéristiques essentielles de notre application et les éléments qui la distinguent des autres sur le marché.

Nous avons structuré ce rapport en quatre grandes parties distinctes. Dans un premier temps, nous parlerons du contexte dans lequel notre projet s'inscrit en identifiant les objectifs de l'application, ses cibles, ainsi que les forces et faiblesses des applications concurrentes. Ensuite, nous parlerons des besoins non fonctionnels liés à l'application, notamment en matière d'ergonomie, d'accessibilité et de conformité au RGPD. Nous entrerons ensuite dans la conception pure et dure en présentant les cas d'utilisation de notre application, le schéma relationnel qui en découle, suivi d'une maquette fonctionnelle réalisée sur Figma. Nous finirons par aborder les technologies que nous allons utiliser pour le projet, en présentant les principales contraintes rencontrées.

I. Contexte et analyse de marché

A. Objectifs de l'application

HealthyCore est une application mobile de suivi nutritionnel et sportif. Elle est destinée à accompagner les utilisateurs dans leurs objectifs de santé, qu'il s'agisse de perdre du poids, de maintenir leur forme ou d'augmenter leur masse musculaire. HealthyCore propose une personnalisation avancée de l'expérience utilisateur via un suivi calorique, des recettes et des programmes sportifs personnalisés. L'application se démarque aussi par l'implémentation d'un système de gamification pour pousser les utilisateurs à atteindre leurs objectifs.

B. Cibles de l'application

L'application HealthyCore s'adresse à une variété de profils d'utilisateurs, chacun ayant des besoins spécifiques en matière de nutrition et de sport. Afin de répondre au mieux à ces attentes, nous avons identifié plusieurs groupes de personnes susceptibles d'être intéressés par notre solution, en fonction de leurs besoins personnels.

Cible	Besoins
Personnes cherchant à améliorer leur condition physique.	Plans nutritionnels et sportifs personnalisés pour des objectifs comme la perte de poids, la prise de masse musculaire, ou le maintien d'un poids sain.
Sportifs de tous niveaux.	Programmes d'entraînement adaptés (cardio, musculation, HIIT) avec un suivi de progression (poids, séances effectuées) pour ajuster les routines selon l'évolution de la forme physique et des objectifs.
Personnes suivant des régimes spécifiques.	Planification de repas et recettes personnalisées pour les régimes particuliers (végétarien, végétalien, sans gluten), garantissant un bon équilibre nutritionnel
Utilisateurs motivés par la gamification.	Système de points, badges et défis pour renforcer la motivation, encourager l'engagement et suivre les accomplissements.

Chaque catégorie d'utilisateur a des attentes particulières en termes de programmes nutritionnels et sportifs, et l'application est conçue pour répondre à ces besoins de manière personnalisée.

C. Analyse de la concurrence

Dans le domaine des applications de nutrition et de sport, la majorité des solutions se concentrent que dans une seule thématique : soit le sport, soit la nutrition. Cependant, certaines applications allient ces deux aspects. Parmi celles-ci, nous en avons analysé trois, en détaillant leurs points forts et leurs points faibles.

Application	Points positifs	Points Négatifs
8FIT	<ul style="list-style-type: none">- Propose des recettes et des entraînements adaptés aux débutants ;- Interface simple et navigation facile.	<ul style="list-style-type: none">- Suivi nutritionnel pas assez détaillé. (pas d'informations sur les macronutriments) ;- Programmes limités aux débutants.
MyFitnessPal	<ul style="list-style-type: none">- Large base de données alimentaires ;- Intégration avec d'autres applications comme Apple Health, Fitbit.	<ul style="list-style-type: none">- Interface complexe ;- 18 questions d'onboarding [6] ;- Manque de gamification.
TrainSweatEat	<ul style="list-style-type: none">- Programmes d'entraînement variés (cardio, muscu, HIIT¹) ;- Recettes pratiques et conseils nutritionnels.	<ul style="list-style-type: none">- Peu de contenus gratuits ;- Absence de système de points pour motiver les utilisateurs.

Notre application va donc essayer de combler les défauts des concurrents en offrant des recettes et des programmes adaptés à l'utilisateur grâce à un questionnaire d'onboarding ([cf. Annexe 1](#)) simple mais efficace. Nous incluons également une gamification pour motiver les utilisateurs à atteindre leurs objectifs

II. Besoins non fonctionnels

A. Ergonomie et expérience utilisateur

- **Interface intuitive** : L'application doit être simple et agréable à utiliser, avec une navigation fluide et intuitive. En suivant les guidelines de Material 3 Design pour la navigation [9], celle-ci doit être cohérente et inclure une barre fixe en bas de l'écran, composée d'icônes universelles facilement compréhensibles [9a]. Par exemple : une icône de maison pour l'accueil, des icônes de couverts et d'haltères pour les sections sportives et nutritionnelles, ainsi qu'une icône de profil pour les paramètres utilisateur. De plus, on utilise un indicateur actif sur l'icône de la page actuellement affichée dans l'application. Cela se traduit par l'affichage d'une forme ovale [9b] en fond de

¹ HIIT : High-Intensity Interval Training

l'icône. Ensuite, l'ajout d'un bouton retour systématique [10a] en haut à gauche de l'écran permet à l'utilisateur de pouvoir revenir sur ses pas si il le souhaite.

- **Feedback visuel** : Nous devons faire en sorte que l'utilisateur ait un retour des actions qu'il effectue. Cela passe premièrement par la présence d'indicateurs de progression [11] dans le questionnaire d'onboarding. Dans ce cas précis du questionnaire, cela permet de dire à l'utilisateur à quelle question il se situe et combien de questions il lui reste à répondre. De plus, nous allons utiliser des cercles de progression [12a] dans le suivi nutritionnel, afin de montrer à l'utilisateur sa consommation des macronutriments dans la journée. Si la consommation dépasse le cercle (excède 100% des quantités recommandés) alors le cercle changera de couleur. Les cercles de progression nous ont paru plus adaptés que des barres de progressions classiques [12b], car nous devons avoir un élément pour chaque macronutriment, et le fait d'empiler des barres de progressions nous semblait moins adapté visuellement.
- **Accessibilité** : Afin que l'application soit utilisable pour le plus de personnes possible, nous devons prendre en compte les principes d'accessibilité [13] (contraste de couleurs, textes lisibles, navigation simplifiée). Tout d'abord, une structure hiérarchique typographique des titres et des différents éléments (icône, texte, image) [10b] permet de faciliter l'accès aux informations les plus importantes en premier. De plus, afin de garantir une bonne visibilité du texte, un niveau de contraste suffisant devra être utilisé et respecter le critère 1.4.3 du référentiel du w3.org [14]. Enfin, la couleur des boutons cliquables permet de les différencier des autres non-cliquables, facilitant donc la navigation.

B. Facilité d'utilisation et facteurs humains

- **Courbe d'apprentissage courte** : L'interface et le processus d'inscription doivent être conçus de manière à être compréhensibles même pour un utilisateur peu expérimenté. Cela passe tout d'abord par le questionnaire d'inscription. Celui-ci a été conçu de sorte à ce qu'il puisse être rempli rapidement avec une saisie des données guidée étape par étape : il possède 7 questions [15a], des choix pré-remplis qu'il suffit juste de sélectionner [15b]. Ensuite, le fonctionnement de l'interface est facile à retenir car on y utilise des icônes universelles (comme la maison pour l'accueil etc...), des cercles de progressions [12a] pour le suivi nutritionnel ainsi que des cards pour l'exploration des programmes de sport et des recettes.
- **Aide intégrée** : Afin d'aider l'utilisateur en cas de problème ou de recherche d'une fonctionnalité, l'application proposera une section FAQ² pour répondre au besoin de l'utilisateur sur les problèmes les plus courants. La faq devra être structuré selon les différentes principales catégories [16a] (nutrition, sport, suivi etc...) et devra apporter des réponses claires et précises sur les différents problèmes [16b].

² FAQ : Frequently asked question

- **Gamification** : De nombreuses études [7a] ont démontré l'impact positif de la gamification sur les changements de comportements alimentaires. Une étude approfondie menée auprès de 220 utilisateurs d'application nutritionnelles [7b] a analysé les préférences des utilisateurs concernant différents éléments de gamification et les a classés dans un tableau :

Table 2. Counting analysis of BWS

GE	Best	Worst	Std. mean	SD	Ranking
Goals	583	43	0.409	0.513	1
Performance Graph	592	106	0.368	0.627	2
Progress Bar	530	100	0.326	0.609	3
Rewards	380	169	0.160	0.625	4
Levels	355	208	0.111	0.644	5
Points	218	234	-0.012	0.585	6
Leaderboards	246	563	-0.240	0.745	7
Badges	183	556	-0.283	0.693	8
Social Interaction	83	579	-0.376	0.600	9
Narratives	10	742	-0.464	0.668	10

Ce tableau montre que les objectifs ("Goals"), les graphiques de performance ("Performance Graph") et les barres de progression ("Progress Bar") sont les éléments les plus appréciés par les utilisateurs, avec des scores standardisés respectifs de 0.409, 0.368 et 0.326. Sur la base de ces résultats, notre application intégrera principalement les objectifs quotidiens et les graphiques de performance, complétés par un système de badges (avec un score de -0.283) qui reste un élément classique de gamification.

Pour renforcer davantage l'engagement des utilisateurs, nous ajouterons également un système de "streaks" (séries). Selon un article de Plotline [8], les streaks sont particulièrement efficaces pour favoriser un engagement régulier et former des habitudes durables, car ils motivent les utilisateurs à revenir quotidiennement.

C. Exigences de sécurité et conformité RGPD

- **Protection des données personnelles** : L'application devra dans un premier temps obtenir un consentement explicite [17a,20] pour le traitement de données de santé. Les données liées aux utilisateurs doivent impérativement être chiffrées dans la base de données. Les mots de passe seront chiffrés [18] en base de données grâce à la bibliothèque bcrypt. L'application traite d'autres données sensibles liées aux informations de santé des utilisateurs [19]. Ces données, qui servent à la recommandation au sein de l'application et à l'affichage des progrès, seront elles aussi chiffrées.
- **Authentification et contrôle d'accès** : Un système de rôles distincts (administrateur et utilisateur) est implémenté pour assurer une gestion granulaire des accès [22]. Les administrateurs disposent uniquement des droits de gestion des comptes et de modification du contenu de l'application, sans accès aux données de santé des utilisateurs. Chaque utilisateur n'a accès qu'à ses propres données, garantissant ainsi la confidentialité des informations personnelles.

- **Minimisation et gestion des données** : En accord avec le principe de minimisation du RGPD, seules les données strictement nécessaires [17b] au fonctionnement de l'application sont collectées et stockées [21]. De plus, un système de traçabilité est mis en place via l'horodatage des créations et modifications de compte, permettant un suivi précis des évolutions des données.

III. Besoins fonctionnels et conception

A. Diagramme des cas d'utilisation

Nous avons organisé les différents cas d'utilisation en fonction des deux principaux types d'utilisateurs de l'application : l'utilisateur et l'administrateur. L'utilisateur est la personne qui utilise l'application pour suivre son alimentation et son activité physique. L'administrateur, quant à lui, dispose de droits spécifiques pour gérer les contenus de l'application (programmes de sport et recettes). Les fonctionnalités prévues ont été réparties en trois versions distinctes.

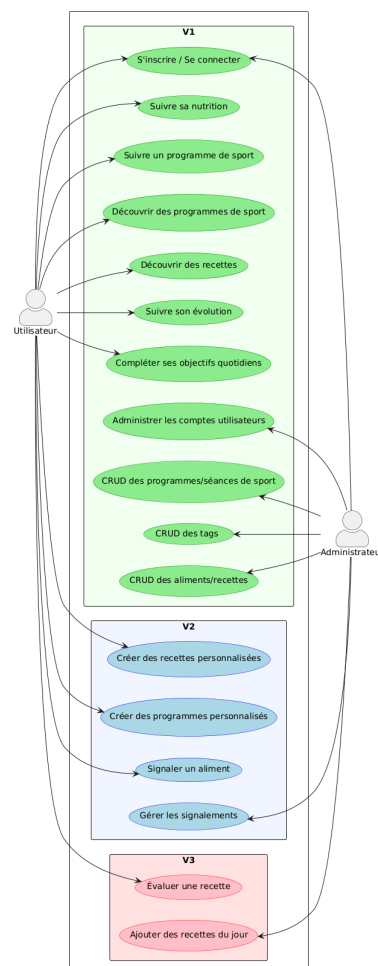


Figure 1 : Diagramme des cas d'utilisation

Le diagramme ci-dessus présente l'ensemble des cas d'utilisation possibles au sein de l'application.

B. Scénarios des cas d'utilisation de la version 1

Cas d'utilisation 1 : S'inscrire

Étape	Utilisateur	Système HealthyCore
Accès à la page d'inscription	Ouvre l'application HealthyCore et sélectionne "Créer un compte".	Affiche la page d'inscription avec les champs requis.
Saisie des informations personnelles	Remplit les champs (Nom, prénom, e-mail, mot de passe).	Vérifie les données saisies.
	Accepte les conditions d'utilisation et clique sur "Continuer".	Affiche un message d'erreur si nécessaire (champs obligatoires non remplis ou données invalides).
Questionnaire de personnalisation	Remplit les sections :	Présente un questionnaire structuré pour collecter les informations.
	- Informations personnelles (sexe, poids, taille).	Enregistre les réponses validées par l'utilisateur.
	- Niveau d'activité.	
	- Objectifs et préférences nutritionnelles/sportives.	
	Valide chaque section.	
Accès à l'application	Attends la création du compte.	Crée un profil utilisateur personnalisé.
	Accède au dashboard d'accueil.	Redirige l'utilisateur vers le dashboard d'accueil.
Variantes - Gestion des erreurs	Corrige les données invalides (par ex. mot de passe faible ou objectif irréaliste) si un message d'erreur s'affiche.	Affiche des messages d'erreur adaptés selon les problèmes détectés (données invalides, objectif incohérent, etc.).

Cas d'utilisation 2 : Se connecter

Étape	Utilisateur	Système HealthyCore
Accès à la connexion	Ouvre l'application et sélectionne "Se connecter".	Affiche le formulaire de connexion demandant l'email et le mot de passe.
Saisie des identifiants	Renseigne l'email et le mot de passe.	Attend la soumission des informations.
Validation des identifiants	Clique sur "Connexion".	Vérifie l'email et le mot de passe dans la base de données.
Authentification réussie	Accède au dashboard d'accueil.	Si les informations sont valides, redirige l'utilisateur vers son tableau de bord.
Authentification échouée	Consulte le message d'erreur si les informations sont incorrectes.	Affiche un message d'erreur si l'email ou le mot de passe est invalide.
Variante - Mot de passe oublié	Clique sur "Mot de passe oublié".	Affiche un champ pour entrer l'email.
	Renseigne son email et clique sur "Envoyer".	Envoie un email avec un lien de réinitialisation.
Réinitialisation du mot de passe	Ouvre l'email, suit le lien et crée un nouveau mot de passe.	Vérifie l'email et permet la création d'un nouveau mot de passe.
Nouvelle tentative de connexion	Retourne à la connexion et entre le nouveau mot de passe.	Vérifie à nouveau les identifiants et redirige vers le dashboard si valides.

Cas d'utilisation 3 : Suivre sa nutrition

Étape	Utilisateur	Système HealthyCore	API Externe (OpenFoodFacts)
Accès au suivi nutritionnel	Ouvre la section "Suivi nutritionnel" depuis le dashboard.	Affiche l'interface de suivi nutritionnel avec les options disponibles.	
Choix d'une action	Sélectionne soit "Ajouter un aliment" soit "Ajouter une recette".	Propose deux options : Rechercher un aliment / Scanner un code-barres OU Sélectionner une recette.	
Ajout d'un aliment	Recherche un aliment dans la base ou scanne un code-barres.	Interroge l'API OpenFoodFacts si un code-barres est scanné.	Retourne les informations nutritionnelles de l'aliment.
Ajout d'une recette	Recherche une recette ou sélectionne une parmi les suggestions.	Affiche les recettes disponibles avec leurs valeurs nutritionnelles.	
Saisie de la quantité	Indique la quantité consommée pour l'aliment ou la recette choisie.	Prend en compte la quantité et calcule les valeurs nutritionnelles associées.	
Mise à jour des données		Met à jour les calories et la répartition des macronutriments dans le journal nutritionnel.	
Affichage du suivi	Consulte les nouvelles données nutritionnelles et leur impact sur ses objectifs.	Affiche un tableau récapitulatif des apports journaliers + une représentation graphique des macronutriments.	
Postcondition	L'aliment ou la recette est enregistré dans le journal nutritionnel.	Met à jour et affiche les données du suivi nutritionnel.	

Cas d'utilisation 4 : Suivre un programme de sport

Étape	Utilisateur	Système HealthyCore
Accès au suivi sportif	Accède à la section "Suivi sportif" depuis le tableau de bord.	Affiche l'interface du programme sportif en cours, organisé par semaine.
Sélection d'une séance	Choisit une séance hebdomadaire à consulter.	Affiche la liste des exercices de la séance sélectionnée.
Affichage des détails	Consulte les exercices et leurs informations associées.	Affiche pour chaque exercice : description, GIF explicatif, durée, répétitions et séries.
Exécution de la séance	Effectue les exercices selon les indications.	-
Validation de la séance	Coche la séance comme "effectuée" une fois terminée.	Met à jour la progression de l'utilisateur dans le programme sportif.
Affichage de la progression	Consulte son avancement dans le programme.	Met à jour et affiche les statistiques de progression (séances complétées/restantes).
Postcondition	La séance est enregistrée comme complétée dans l'application.	Les données de progression de l'utilisateur sont mises à jour et accessibles dans son suivi.

Cas d'utilisation 5 : Découvrir des recettes

Étape	Utilisateur	Système HealthyCore
Accès à l'exploration nutritionnelle	Sélectionne l'option "Explorer nutrition" depuis le menu principal.	Affiche la page "Explorer nutrition" avec les catégories et suggestions de recettes.
Choix d'une catégorie ou recherche	Sélectionne une catégorie de recettes ou effectue une recherche spécifique.	Filtre et affiche les résultats de recettes correspondantes depuis la base de données.
Affichage des résultats	Consulte la liste des recettes proposées.	Présente une liste de recettes avec images, titres et brève description.
Sélection d'une recette	Clique sur une recette pour voir les détails.	Affiche les détails : ingrédients, étapes, temps de préparation, et valeurs nutritionnelles.
Postcondition	Consulte la recette et décide de l'ajouter à son plan nutritionnel.	Permet l'ajout de la recette au plan nutritionnel de l'utilisateur.

Cas d'utilisation 6 : Découvrir des programmes de sport

Étape	Utilisateur	Système HealthyCore
Accès à l'exploration sportive	Sélectionne l'option "Explorer sportif" depuis le menu principal.	Affiche la page "Explorer sportif" avec des catégories de programmes sportifs disponibles.
Choix d'une catégorie	Sélectionne une catégorie de programmes sportifs qui l'intéresse.	Affiche la liste des programmes associés avec détails (durée, intensité, etc.).
Sélection d'un programme	Choisit un programme pour voir les détails.	Affiche le programme avec : planning hebdomadaire, exercices détaillés (GIF, explications), équipements requis.
Ajout du programme	Ajoute le programme à son suivi sportif.	Enregistre le programme sélectionné dans la section "Suivi sportif" de l'utilisateur.
Postcondition	Le programme est accessible pour une utilisation ultérieure.	L'utilisateur peut commencer à suivre les séances hebdomadaires via "Suivi sportif".

Cas d'utilisation 7 : Compléter ses objectifs quotidiens

Étape	Utilisateur	Système HealthyCore
Accès aux objectifs quotidiens	Ouvre l'application et accède à la liste des objectifs sur le dashboard.	Affiche la liste des objectifs du jour (sport, nutrition, autres tâches à accomplir).
Réalisation des tâches	Effectue les actions requises (ex : réalise une séance de sport, ajoute un aliment).	Met à jour l'état des objectifs complétés en temps réel.
Suivi de progression	Consulte la progression des objectifs quotidiens et voit les tâches restantes.	Affiche une mise à jour des objectifs réalisés/restants et ajuste l'interface en conséquence.
Validation de la journée	Termine toutes les tâches du jour.	Si toutes les tâches sont accomplies, déclenche une streak et l'enregistre.
Attribution de badges	Consulte les badges obtenus.	Débloque et affiche les badges en fonction des streaks et des actions effectuées.
Postcondition	A complété ses objectifs quotidiens et peut voir sa progression sur plusieurs jours.	L'application enregistre la streak et met à jour les données de suivi de l'utilisateur.
Variante - Objectif partiel	N'accomplit pas toutes les tâches de la journée.	La streak n'est pas créée. L'utilisateur voit les tâches non complétées.
Variante - Oubli de tâche	Oublie une tâche ou tarde à la compléter.	Peut recevoir une notification de rappel pour inciter à finaliser ses objectifs du jour.

Cas d'utilisation 8 : Suivre son évolution

Étape	Utilisateur	Système HealthyCore
Accès à l'évolution	Accède à la page "Voir mon évolution" via la page de profil.	Affiche une vue d'ensemble des progrès avec graphiques et statistiques détaillées.
Consultation des progrès	Visualise les statistiques de progression (nombre de séances effectuées, évolution du poids).	Génère et affiche des graphiques interactifs basés sur les données de l'utilisateur.
Sélection d'une période	Choisit une période d'analyse pour voir l'évolution (ex : semaine, mois, année).	Met à jour l'affichage des données et des graphiques selon la période sélectionnée.
Mise à jour du poids	Met à jour son poids en saisissant une nouvelle valeur.	Enregistre le nouveau poids et ajuste les graphiques d'évolution en conséquence.
Consultation des badges	Consulte les badges obtenus liés à ses progrès.	Affiche les badges débloqués et les critères pour obtenir les prochains.
Postcondition	Dispose d'une vue claire sur son évolution sportive et nutritionnelle.	Stocke et met à jour les données de suivi et affiche les progrès de l'utilisateur.

Cas d'utilisation 9 : Administrer les comptes utilisateurs

Étape	Administrateur	Système HealthyCore
Accès à la gestion des comptes	Accède à l'interface d'administration et sélectionne la section "Comptes utilisateurs".	Affiche la liste des utilisateurs avec leurs informations de base.
Recherche ou filtrage	Filtre ou recherche un utilisateur spécifique.	Affiche les résultats correspondant aux critères de recherche.
Consultation du profil utilisateur	Sélectionne un utilisateur pour voir ses informations détaillées.	Affiche le profil de l'utilisateur avec ses données de suivi nutritionnel, sportif et progrès.
Suppression d'un compte	Clique sur "Supprimer" pour un utilisateur spécifique.	Demande une confirmation avant la suppression pour éviter les erreurs.
Confirmation et exécution	Confirme la suppression du compte.	Supprime définitivement le compte de la base de données et met à jour la liste.
Consultation des statistiques	Consulte les statistiques générales des utilisateurs (nombre total, tendances, etc.).	Génère et affiche des statistiques mises à jour.
Postcondition	Les modifications (suppression) sont appliquées et visibles immédiatement.	Met à jour la base de données et reflète les changements dans l'interface administrateur.

Cas d'utilisation 10 : CRUD des programmes / séances de sport

Étape	Administrateur de l'application	Système HealthyCore
Accès à la gestion des programmes/séances	Accède à l'interface d'administration et sélectionne la gestion des programmes/séances de sport.	Affiche la liste des programmes et séances de sport existants.
Création d'un programme	Renseigne les informations du programme (nom, catégorie, durée, intensité, séances...).	Enregistre le nouveau programme dans la base de données et le rend disponible.
Consultation d'un programme	Sélectionne un programme pour voir ses détails.	Affiche les informations complètes du programme sélectionné.
Modification d'un programme	Modifie les informations du programme (ex : durée, intensité, contenu des séances).	Met à jour les informations et applique les modifications immédiatement.
Suppression d'un programme	Supprime un programme existant.	Demande confirmation avant suppression et met à jour la base de données.
Création d'une séance de sport	Renseigne les détails de la séance (exercices, répétitions, durée, intensité...).	Enregistre la nouvelle séance et l'intègre au programme concerné.
Consultation d'une séance	Sélectionne une séance pour voir ses détails.	Affiche les informations détaillées de la séance de sport.
Modification d'une séance	Modifie les informations d'une séance existante.	Met à jour la séance de sport et ajuste le programme correspondant.
Suppression d'une séance	Supprime une séance existante.	Demande confirmation avant suppression et met à jour la base de données.
Gestion des catégories	Ajoute, modifie ou supprime des catégories de programmes et de recettes.	Applique les modifications immédiatement et met à jour les listes correspondantes.
Postcondition	Les programmes, séances ou catégories ont été créés, modifiés ou supprimés.	Applique les modifications en temps réel et les rend accessibles aux utilisateurs.

C. Maquettes UX/UI

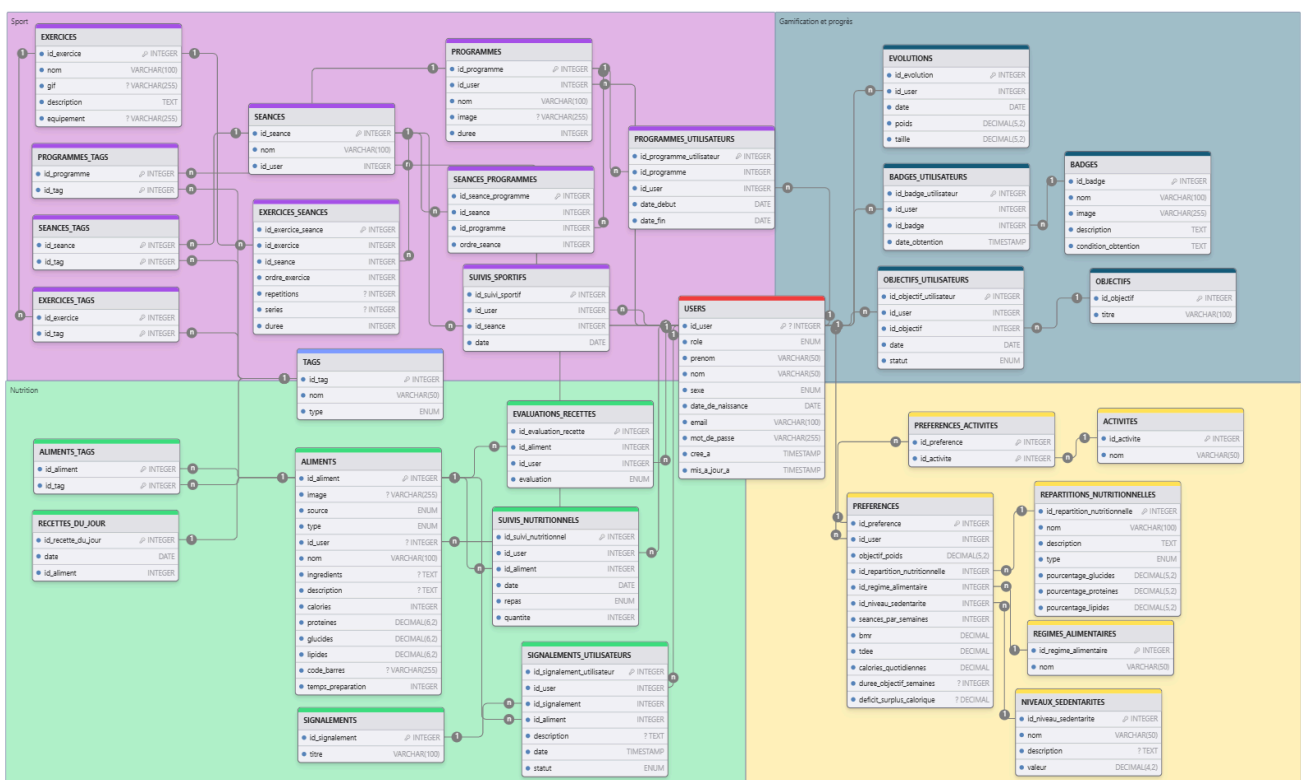
Afin d'avoir une idée précise de ce que nous voulons développer, nous avons conçu une maquette grâce avec Figma en suivant les principes détaillés dans la partie [II. A. Ergonomie et expérience utilisateur](#).

La maquette fonctionnelle se trouve à l'adresse suivante :

<https://www.figma.com/proto/3jKPeCGXuYUHsS4b8DQ8P8/HealthyCore-UI?node-id=3-103&node-type=canvas&t=TFwbHAuL5hP7I7Ys-1&scaling=scale-down&content-scaling=fixed&page-id=0%3A1&starting-point-node-id=3%3A103>

D. Schéma relationnel

Afin de répondre aux différents cas d'utilisation citées plus haut, nous avons conçu un modèle qui représente l'architecture des données nécessaires pour notre application HealthyCore. Le schéma a pour but de structurer et de visualiser les entités, leurs relations et les contraintes logiques qui régissent l'organisation des informations.



REPARTITIONS_NUTRITIONNELLES, REGIMES_ALIMENTAIRES et NIVEAUX_SEDENTARITES.

L'intégrité des données est assurée par plusieurs mécanismes. Les valeurs prédéfinies sont gérées via des ENUM (rôles utilisateur, sexe, types de repas, statuts des signalements et objectifs, sources d'aliments), tandis que les relations plusieurs-à-plusieurs sont gérées par des tables de jonction (ALIMENTS_TAGS, PREFERENCES_ACTIVITES, etc.) conformément à la première forme normale (1NF). Un système de tags flexible permet la catégorisation transversale des contenus (aliments, programmes, séances).

La table ALIMENTS peut contenir des données administrateur, des données d'API ou des contenus générés par les utilisateurs. Cette flexibilité est complétée par des mécanismes de modération (SIGNALEMENTS) et d'engagement communautaire (EVALUATIONS_RECETTES).

L'ensemble est sécurisé par un système complet de contraintes d'intégrité : clés primaires sur toutes les tables, clés étrangères, contraintes CHECK pour les valeurs numériques, et contraintes UNIQUE pour éviter les duplications indésirables.

IV. Contraintes et vision techniques

A. Contraintes techniques

Compatibilité multi-plateforme

L'application HealthyCore doit être compatible à la fois sur les plateformes iOS et Android pour toucher un large public. Cela nécessite une expérience utilisateur cohérente sur les deux systèmes d'exploitation, tout en garantissant une performance optimale et une gestion fluide des ressources. Lorsqu'il s'agit de solutions cross-platform, plusieurs noms ressortent. Tout d'abord React Native, développé par Facebook, permet d'utiliser Javascript pour développer des applications natives sur iOS et Android à la fois avec une seule et même codebase [23]. Ensuite, Flutter, développé par Google, permet d'utiliser le langage Dart (similaire au java) pour également développer des applications natives sur iOS et Android à la fois avec une seule codebase [24].

Gestion des fonctionnalités matérielles

L'application devra intégrer plusieurs fonctionnalités matérielles sur les appareils mobiles, telles que la caméra et les notifications push. En effet, pour pouvoir scanner des produits, l'application doit pouvoir accéder facilement à la caméra. Cette fonctionnalité doit être accessible sans nécessiter une configuration manuelle complexe de modules natifs, afin de simplifier le processus de développement et de garantir une intégration fluide. React native, avec sa librairie

react-native-vision-camera [25] permet d'accéder à la caméra facilement. Avec flutter, et son package qr_code_scanner [26] on peut utiliser la caméra en effectuant des configurations sur les fichiers natifs.

Optimisation de la taille de l'application

L'application devra être suffisamment légère pour ne pas surcharger les appareils des utilisateurs [27]. Il est donc essentiel que la taille de l'application soit optimisée, sans inclure de fonctionnalités superflues ou d'éléments qui augmenteraient inutilement le poids du build final. Les applications sont en moyenne plus lourdes lorsqu'elles sont faites en Flutter [28a] qu'en React native.

Simplicité dans la gestion des interfaces utilisateurs

Les interfaces utilisateurs doivent être simples et fonctionnelles. Bien que l'application nécessite des transitions et animations simples, il n'est pas nécessaire d'intégrer des animations fluides ou complexes. L'objectif est de créer une interface claire et rapide, qui favorise l'expérience utilisateur plutôt que d'ajouter des éléments visuels trop sophistiqués. En termes d'UI, Flutter permet d'avoir le contrôle de A à Z sur la customisation des composants [28b]. React Native permet de récupérer des composants déjà existants pour les modifier à notre guise [28c].

Gestion de requêtes simultanées

L'application devra être capable de gérer un grand nombre de requêtes simultanées. De nombreux utilisateurs suivront leur programme nutritionnel et sportif en temps réel, avec des interactions fréquentes pour ajouter des repas, suivre leurs progrès, ou gérer leurs séances d'entraînement. Il est donc crucial que le système soit capable de traiter efficacement ces interactions en parallèle, sans blocage. Node.js, un framework javascript, possède une architecture asynchrone et non-bloquante en plus d'un 'event loop' [29] qui permet de gérer efficacement de nombreuses requêtes simultanées. Django, une librairie python, permet également de gérer des requêtes grâce aux 'QuerySets' [30] qui exécute les requêtes que lorsque c'est nécessaire et la méthode 'iterator()' qui permet de traiter efficacement de grandes quantités de données.

Intégration d'une API externe

L'application doit se connecter à une API externe OpenFoodFacts [5] pour obtenir des informations nutritionnelles détaillées sur les produits alimentaires. Cette API open-source gratuite possède une base de données riche et une documentation claire [31].

B. Choix des technologies

Front-end : React Native et Expo

- Compatibilité Multi-plateforme :

Nous allons utiliser le framework React Native car il est une solution idéale pour le développement d'une application multi-plateforme. Il permet de maintenir une seule base de code pour les deux systèmes d'exploitation majeurs (iOS et Android), ce qui réduit le temps de développement et assure une expérience utilisateur cohérente sur les deux plateformes. L'utilisation d'Expo avec React Native facilite encore ce processus en fournissant des outils pour accéder directement aux fonctionnalités matérielles comme la caméra [32], les notifications push et les capteurs, sans avoir besoin de configurations natives complexes.

- Optimisation de la taille de l'application :

De plus React Native, comparé à d'autres frameworks comme Flutter, est plus léger en termes de taille d'application. Flutter inclut son propre framework graphique dans chaque build, ce qui peut augmenter la taille de l'application. En revanche, React Native permet un build plus optimisé et plus léger, [28a] ce qui est essentiel pour ne pas alourdir les appareils des utilisateurs.

- Simplicité dans la gestion des interfaces utilisateurs :

React Native utilise une syntaxe de type CSS avec Flexbox pour la création des interfaces utilisateur [33], ce qui rend le code plus simple et plus lisible par rapport à des technologies comme Flutter qui utilisent des widgets [34]. Cette approche permet de créer des interfaces plus simples et plus fonctionnelles, idéales pour une application où la rapidité et la clarté sont primordiales.

Back-end : Node.js, Express et PostgreSQL

- Gestion des requêtes simultanées :

Node.js est particulièrement adapté pour gérer un grand nombre de requêtes simultanées grâce à son moteur d'exécution non-bloquant et asynchrone [29]. Dans une application comme HealthyCore, où plusieurs utilisateurs interagissent en temps réel, Node.js permet de traiter efficacement de nombreuses requêtes sans ralentir l'application. Il est capable de gérer de multiples actions simultanées, comme l'ajout de repas ou le suivi des séances d'entraînement, en arrière-plan sans bloquer l'interface utilisateur.

- Express pour le back-end :

Express, un framework minimaliste pour Node.js, est conçu pour créer des API RESTful simples et efficaces. Il facilite la gestion des interactions avec la base de données [35].

- PostgreSQL pour la gestion des données structurées :

PostgreSQL a été sélectionné pour gérer les données relationnelles complexes de l'application. Ce choix s'inscrit dans une approche basée sur la stack PERN (PostgreSQL, Express, React Native, Node.js) [36], qui permet un développement unifié en JavaScript, tant côté backend que frontend.

Conclusion et planification

Bilan du premier semestre

Le premier semestre a été particulièrement exigeant. Rassembler toutes les idées et les organiser de manière cohérente a demandé beaucoup de réflexion et de travail d'équipe. Il n'a pas toujours été facile de faire le tri parmi les nombreuses propositions, de déterminer quelles fonctionnalités étaient essentielles, et surtout de planifier précisément les besoins en termes de ressources et de temps.

L'équipe a su collaborer efficacement et mener à bien cette phase de conception.

Nous sommes globalement satisfaits du résultat obtenu pour la phase de conception. Nous avons établi une base solide pour le projet et les premières maquettes, ainsi que les diagrammes fonctionnels, reflètent bien notre vision. Bien sûr, nous savons qu'il y a toujours des points à améliorer, mais nous sommes confiants dans la direction que nous avons prise et sommes fiers du travail accompli jusqu'à présent.

Objectifs pour le second semestre

Le semestre 2 sera consacré au développement de l'application. Afin de mettre en avant nos qualités, nous avons décidé de nous répartir le travail de la sorte :

Badre	Développement du front-end de l'application et intégration des maquettes conçues.
Nicolas et Thomas	Création du back-end, gestion de la base de données, et conception des programmes et des recettes.

Planification et calendrier prévisionnel

Voici la planification du développement pour le semestre 2, découpée par phases :

Semaine 1 (5 jours) : 27/01 - 31/01

Objectifs : Lancement du projet, création de la base de données, développement des premières pages, et mise en place du back-end.

- Base de données (2 jours)
 - Analyse des besoins, création du schéma initial (1 jour, Nicolas & Thomas).
 - Mise en place de la base de données dans l'environnement de développement (1 jour, Thomas).
- Front-end (2 jours)
 - Création du projet front-end et intégration des premières maquettes (1,5 jours, Badre).
 - Connexion front-end avec l'API CRUD aliments (0,5 jour, Badre).
- Back-end (1 jour)
 - Développement des API pour CRUD aliments et recettes (1 jour, Nicolas).

Semaine 4-5 (10 jours) : 17/02 - 01/03

Objectifs : Finalisation de la version 1 et lancement de la version 2.

- Finalisation de la version 1 (3 jours)
 - Finalisation des fonctionnalités CRUD et gestion des utilisateurs (2 jours, Nicolas & Thomas).
 - Corrections UI/UX des maquettes intégrées (1 jour, Badre).
- Développement de la version 2 (7 jours)
 - Développement des fonctionnalités "Suivre sa nutrition" et "Suivre son évolution" (2 jours, Nicolas).
 - Connexion des données utilisateur et affichage dynamique sur le front-end (2 jours, Badre).
 - Développement des programmes personnalisés (2 jours, Thomas).
 - Tests initiaux pour version 2 (1 jour, tous).

Semaine 9-10 (10 jours) : 24/03 - 04/04

Objectifs : Amélioration et finalisation de la version 2, lancement de la version 3.

- Amélioration et finalisation de la version 2 (5 jours)
 - Intégration des fonctionnalités "Signaler un aliment" et "Créer des recettes personnalisées" (3 jours, Badre).
 - Développement des API pour la gestion des signalements et des programmes personnalisés (2 jours, Nicolas & Thomas).
- Développement de la version 3 (5 jours)
 - Développement des fonctionnalités "Ajouter des recettes du jour" et "Évaluer une recette" (3 jours, Nicolas & Thomas).
 - Intégration front-end des nouvelles fonctionnalités (2 jours, Badre).

Semaine 14 (5 jours) : 21/04 - 26/04

Objectifs : Finalisation de la version 3.

- Tests finaux et corrections (3 jours)
 - Vérifications globales des fonctionnalités front-end et back-end (3 jours, tous).
- Documentation et déploiement (2 jours)
 - Documentation technique et utilisateur (1 jour, Nicolas & Thomas).
 - Préparation et exécution du déploiement (1 jour, Nicolas & Thomas).

Webographie

[1] - Rizzo, Nicholas. "100+ Fitness App Statistics 2021/2022 [Research Review]." Athletic Shoe Reviews, 18 Oct. 2023, runrepeat.com/fitness-app-statistics. "71% of users abandon their apps by month 3."

[2] - Mifflin, M D, et al. "A New Predictive Equation for Resting Energy Expenditure in Healthy Individuals." *The American Journal of Clinical Nutrition*, vol. 51, no. 2, 1 Feb. 1990, pp. 241–247, <https://doi.org/10.1093/ajcn/51.2.241>.

[3] - Team Acko. "TDEE Calculator: Calculate Total Daily Energy Expenditure." ACKO, 2019, www.acko.com/calculators/tdee-calculator.

[4] - "Max Healthcare.", www.maxhealthcare.in/calculator/tdee.

[5] - "Open Food Facts." Openfoodfacts.org, 2024, fr.openfoodfacts.org/data.

[6] - Khalkho, Arpana. "Emphasizing Value Proposition to Improve User Onboarding Experience | by Arpana Khalkho | in Bootcamp - Freedium." Freedium.cfd, 6/29/23, freedium.cfd/medium.com/design-bootcamp/emphasizing-value-proposition-to-improve-user-onboarding-experience-455dc66387d0. Accessed 19 Jan. 2025. "The current onboarding flow of MyFitnessPal has 18 steps [...]."

[7] - Berger, Michelle, and Carolin Jung. "Gamification in Nutrition Apps -Users' Gamification Element Preferences: A Best-Worst-Scaling Approach." ResearchGate, 1/8/21, www.researchgate.net/publication/348336387_Gamification_in_Nutrition_Apps_-_Users. Accessed 19 Jan. 2025.

[7a] - Page 1337, 2.2. Gamification in the Context of Nutrition.

[7b] - Page 1340, Table 2. Counting analysis of BWS.

[8] - Vairavarajan, Ananya. "Streaks for Gamification in Mobile Apps." Plotline.so, 9 Dec. 2024, www.plotline.so/blog/streaks-for-gamification-in-mobile-apps. Accessed 19 Jan. 2025. "Streaks motivate users to return daily by offering tangible or visual rewards."

[9] - Google. "Navigation Bar – Material Design 3." Material Design, m3.material.io/components/navigation-bar/guidelines. Accessed 19 Jan. 2025.

[9a] - "The nav bar is positioned at the bottom of screens for convenient access. Each destination is represented by an icon and optional text label."

[9b] - "The active indicator is a background shape communicating which destination of the navigation bar is currently being displayed."

[10] - Costa, Rebeka. "Navigation Design: Almost Everything You Need to Know." Www.justinmind.com, 12 Mar. 2020, www.justinmind.com/blog/navigation-design-almost-everything-you-need-to-know/. Accessed 19 Jan. 2025.

[10a] - "When you think of navigation, the back button on your web browser might spring to mind. It's a great feature, [...]."

[10b] - "Define the order and placement of navigation elements. Main sections should be easy to find and access, while sub-sections should be logically grouped."

[11] - Google. "Progress Indicators – Material Design 3." Material Design, 2025, m3.material.io/components/progress-indicators/accessibility. Accessed 19 Jan. 2025. "The active indicator, which displays progress, provides visual contrast of at least 3:1 against most background colors."

[12] - Experience Architecture. "Progress Circle - Compose Design." Compose Design - Copyright 2024 Cegid, 17 July 2024, composedesign.ila.cegid.com/development/components/progress-circle/.

[12a] - "Progress Circles are an effective way of showing progress in terms of ongoing activity or remaining time."

[12b] - "Progress bars, with their linear format, offer a distinct starting and ending point, making them well-suited for tasks with definite completion points."

[13] - "Principes d'accessibilité." Web Accessibility Initiative (WAI), 2019, www.w3.org/WAI/fundamentals/accessibility-principles/fr. Accessed 19 Jan. 2025.

[14] - Web Content Accessibility Guidelines (WCAG) 2.2." W3.org, 20 July 2023, www.w3.org/TR/WCAG22/#contrast-minimum. Accessed 19 Jan. 2025.

"The visual presentation of text and images of text has a contrast ratio of at least 4.5:1."

[15] - "8 Astuces à Connaître Pour Une Page de Connexion / Inscription Parfaite Sur Mon Application." Beapp.fr, 19 Oct. 2024, www.beapp.fr/blog/page-login-application-mobile. Accessed 19 Jan. 2025.

[15a] - "3. Demandez peu d'informations et n'ajoutez pas de distraction"

[15b] - "7. Facilitez le remplissage des champs."

[16] - Gadant, Mélanie. "5 Exemples de FAQ Web Réussies (+ Conseils Pour Faire La Vôtres)." Hubspot, 17 Sept. 2021, blog.hubspot.fr/service/faq-exemple. Accessed 19 Jan. 2025.

[16a] - "Bien structurer sa FAQ"

[16b] - "Apporter la bonne réponse."

[17] - Commission européenne. "Code de Conduite Sur La Protection de La Vie Privée Sur Les Applications de Santé Mobiles." 28 Oct. 2024, digital-strategy.ec.europa.eu/fr/policies/privacy-mobile-health-apps.

[17a] - "Il faut obtenir un consentement explicite pour le traitement des données de santé."

[17b] - "Seules les données strictement nécessaires à la fonctionnalité de l'application peuvent être traitées."

[18] - "Sécurité : Chiffrement, Hachage, Signature." Cnil.fr, 14 Mar. 2024, www.cnil.fr/fr/securite-chiffrement-hachage-signature. Accessed 19 Jan. 2025. "Utiliser un

algorithme reconnu et sûr, par exemple, les algorithmes suivants : [...] bcrypt, scrypt, Argon2 ou PBKDF2 pour stocker les mots de passe”;

[19] - “Applications Mobiles En Santé et Protection Des Données Personnelles : Les Questions à Se Poser.” Cnil.fr, 17 Aug. 2018, www.cnil.fr/fr/applications-mobiles-en-sante-et-protection-des-donnees-personnelles-les-questions-se-poser. Accessed 19 Jan. 2025. “Du fait de leur sensibilité, ces données font l’objet d’un régime juridique de protection renforcée.”

[20] - CHAPITRE I - Dispositions Générales.” Cnil.fr, 23 May 2016, www.cnil.fr/fr/reglement-europeen-protection-donnees/chapitre1#Article4. Accessed 19 Jan. 2025. “ “consentement” de la personne concernée, toute manifestation de volonté, libre, spécifique, éclairée et univoque par laquelle la personne concernée accepte, par une déclaration ou par un acte positif clair, que des données à caractère personnel la concernant fassent l’objet d’un traitement”;

[21] - Data Legal Drive. “Logiciel Conformité RGPD, Logiciel DPO, Logiciel RGPD.”, 27 May 2024, datalegaldrive.com/rgpd-le-role-de-la-securite-dans-la-conformite/. Accessed 19 Jan. 2025. “Seules les données strictement nécessaires à des fins définies doivent être collectées et traitées.”

[22] - “PRACTICE GUIDE GDPR SECURITY of PERSONAL DATA.” CNIL, 2024. https://www.cnil.fr/sites/cnil/files/2024-03/cnil_guide_securite_personnelle_ven_0.pdf “Define authorisation profiles in systems by separating tasks and areas of responsibility, in order to restrict users’ access to only the data strictly necessary for fulfilling their responsibilities.”

[23] - Walia, Raghav. “Reusing a Single Codebase for Creating Multiple React Native Applications and Ways to Maintain the Same.” Geekyants.com, <https://clutch.co/profile/geekyants#reviews>, 9 July 2024, geekyants.com/blog/reusing-a-single-codebase-for-creating-multiple-react-native-applications-and-ways-to-maintain-the-same. Accessed 19 Jan. 2025.

[24] - Rabiaza, Narijaona. “Introduction Au Développement d’Applications Flutter - Bocasay.” Bocasay, 7 June 2024, www.bocasay.com/fr/introduction-developpement-applications-flutter/. Accessed 19 Jan. 2025. “[...] Sa base de code unique, [...]”

[25] - react-native-vision-camera, github.com/mrousavy/react-native-vision-camera

[26] - qr_code_scanner, https://pub.dev/documentation/qr_code_scanner/latest/

[27] - “Réduire La Taille de Votre Application.” Android Developers, 2024, developer.android.com/topic/performance/reduce-apk-size?hl=fr. Accessed 19 Jan. 2025. “Les utilisateurs évitent souvent de télécharger des applications qui paraissent trop volumineuses [...]”

[28] - Shah, Hardik. "Flutter vs. React Native: Which Framework Suits Your Project?" Prismetric, 18 Oct. 2023, www.prismetric.com/flutter-vs-react-native/. Accessed 19 Jan. 2025.

[28a] - "Flutter apps are larger than the equivalent React Native apps. A default Hello World Flutter app can be around 7 MB, while for React Native, it could be around 5 MB."

[28b] - "Flutter provides the maximum flexibility and customization for UI design."

[28c] - "React Native provides native components that adhere to platform norms."

[29] - Roquilly, Lucas. "NodeJS : Guide Complet Pour Tout Comprendre Du Javascript Serveur." WeLoveDevs.com, 27 Apr. 2020, welovedevs.com/fr/articles/nodejs/. "Une boucle d'événements, appelée aussi event loop NodeJS, permettant d'exécuter plusieurs opérations simultanées de façon asynchrone et non bloquante en tirant profit des multiples fils d'exécution (multithreading) des noyaux des processeurs modernes."

[30] - Making Queries | Django Documentation." Django Project, 2020, docs.djangoproject.com/en/5.1/topics/db/queries/. "QuerySets are lazy – the act of creating a QuerySet doesn't involve any database activity. "

[31] - "Présentation de Open Food Facts API et de Son Utilisation." All API, 2024, all-api.fr/api/detail/open-food-facts-api. Accessed 19 Jan. 2025.

[32] - "Camera." Expo Documentation, docs.expo.dev/versions/latest/sdk/camera/. Accessed 19 Jan. 2025.

[33] - "Layout with Flexbox", React Native, <https://reactnative.dev/docs/flexbox>

[34] - "Widgets catalog", Flutter, <https://docs.flutter.dev/ui/widgets>

[35] - Sharma, Anubhav. "What Is Express JS in Node JS | Simplilearn." Simplilearn.com, 11 July 2024, www.simplilearn.com/tutorials/nodejs-tutorial/what-is-express-js. Accessed 19 Jan. 2025. "It saves a lot of coding time almost by half and still makes web and mobile applications are efficient."

[36] - Alves, Rita. "Get Started with the PERN Stack: An Introduction and Implementation Guide." Medium, 2 Mar. 2023, medium.com/@ritapalves/get-started-with-the-pern-stack-an-introduction-and-implementation-guide-e33c55d09994. Accessed 19 Jan. 2025.

Annexes

Annexe 1 : Questionnaire d'onboarding

Informations personnelles

1. Quel est votre sexe ?
 - ☐ Homme
 - ☐ Femme
 - ☐ Préfère ne pas répondre
2. Quel est votre poids (en kg) et taille (en cm) ?
3. Quel est votre niveau d'activité quotidienne ?
 - ☐ Très faible (travail de bureau)
 - ☐ Faible (travail légèrement actif)
 - ☐ Modéré (activités quotidiennes impliquant une activité physique régulière)
 - ☐ Élevé (travail physique intense)
 - ☐ Super actif (mode de vie très exigeant physiquement)

Objectifs

4. Quel est votre objectif de poids (en kg) ?
5. Choisissez votre répartition nutritionnelle :
 - ☐ Pour la perte de poids :
 - Cardio (60% glucides, 20% protéines, 20% graisses)
 - Athlète (55% glucides, 20% protéines, 25% graisses)
 - Durable (55% glucides, 15% protéines, 30% graisses)
 - ☐ Pour la prise de masse :
 - Cardio (60% glucides, 20% protéines, 20% graisses)
 - Athlète (55% glucides, 20% protéines, 25% graisses)
 - Se muscler (50% glucides, 25% protéines, 25% graisses)
 - ☐ Pour le maintien : tous les plans sont disponibles

Personnalisation nutritionnelle

6. Suivez-vous un régime alimentaire spécifique ?
 - ☐ Aucun
 - ☐ Végétarien
 - ☐ Végétalien
 - ☐ Sans gluten

Personnalisation sportive

7. Quel type d'activité physique préférez-vous ? (plusieurs choix possibles)
 - ☐ Cardio modéré (marche, vélo...)
 - ☐ HIIT
 - ☐ Musculation
 - ☐ Yoga/Pilates
8. Combien de séances par semaine souhaitez-vous faire ?
 - ☐ 1 à 2
 - ☐ 3 à 4
 - ☐ 5 à 6
 - ☐ 7+

Annexe 2 : Calculs techniques et nutritionnels

Calcul de l'apport quotidien calorique :

1. Calcul du Basal Metabolic Rate (BMR) grâce à l'équation de Muffin-St Jeor [2]

Le BMR représente le nombre de calories qu'une personne brûle au repos, sans tenir compte des activités physiques quotidiennes. Ce calcul dépend de plusieurs facteurs : le sexe, le poids, la taille et l'âge.

Pour les hommes :

$$BMR = (10 * \text{poids en kg}) + (6.25 * \text{taille en cm}) - (5 * \text{âge en années}) + 5$$

Pour les femmes :

$$BMR = (10 * \text{poids en kg}) + (6.25 * \text{taille en cm}) - (5 * \text{âge en années}) - 161$$

Bien que l'équation de Mifflin-St Jeor soit considérée comme la plus moderne et largement utilisée dans les domaines de la nutrition et du fitness, elle présente des marges d'erreur de l'ordre de 5 à 10 %. Cela signifie que le BMR calculé peut légèrement sous-estimer ou surestimer les besoins énergétiques réels d'un individu. Ces erreurs sont normales car la formule ne prend pas en compte des facteurs plus précis comme la composition corporelle, les niveaux hormonaux ou les variations métaboliques individuelles.

2. Calcul du Total Daily Energy Expenditure (TDEE)

Le TDEE représente le nombre total de calories qu'une personne brûle en une journée, en tenant compte de son niveau d'activité physique, y compris les activités quotidiennes et l'exercice formel [3]. Pour calculer le TDEE, on multiplie le BMR par un facteur d'activité qui reflète le niveau d'activité générale de l'individu [4].

Niveau d'activité	Facteur d'activité
Très faible (travail de bureau, peu ou pas d'exercice)	1.2
Faible (travail légèrement actif, exercice léger ou de courte durée)	1.375
Modéré (activités quotidiennes impliquant une activité physique régulière, comme marcher, se tenir debout, ou porter des objets légers)	1.55
Élevé (travail physique intense, nécessitant un effort physique important, comme dans la construction ou des tâches manuelles)	1.725
Super actif (mode de vie très exigeant physiquement, avec des mouvements constants et des tâches très actives)	1.9

Le calcul du TDEE s'effectue ainsi : $TDEE = BMR * Facteur\ d'activit $

3. Calcul des apports caloriques en fonction des objectifs

Une fois que les informations de base sont collect es et que le TDEE a  t  calcul , l'application peut d terminer le nombre de calories n cessaires pour atteindre les objectifs sp cifiques de l'utilisateur, que ce soit la perte de poids, la prise de masse musculaire ou le maintien du poids.

Perte de poids :

1. Calcul des calories n cessaires   la perte de poids :
 $Calories\ n cessaires = TDEE \times 0.9$
2. Calcul du d ficit calorique total pour atteindre l'objectif de perte de poids :
 $D ficit\ calorique\ total = Objectif\ de\ perte\ de\ poids \times 7700\ kcal/kg$
3. Calcul du d ficit calorique quotidien :
 $D ficit\ calorique\ quotidien = TDEE - Calories\ n cessaires$
4. Calcul du nombre de jours n cessaires pour atteindre l'objectif :
 $Nombre\ de\ jours = \frac{D ficit\ calorique\ total}{D ficit\ calorique\ quotidien}$
5. Calcul du nombre de semaines n cessaires pour atteindre l'objectif :
 $Nombre\ de\ semaines = \frac{Nombre\ de\ jours}{7}$

Perte de poids :

1. Calcul des calories n cessaires   la prise de poids :
 $Calories\ n cessaires = TDEE \times 1.1$
2. Calcul du surplus calorique total pour atteindre l'objectif de prise de poids :
 $Surplus\ calorique\ total = Objectif\ de\ prise\ de\ poids \times 7700\ kcal/kg$
3. Calcul du surplus calorique quotidien :
 $Surplus\ calorique\ quotidien = Calories\ n cessaires - TDEE$
4. Calcul du nombre de jours n cessaires pour atteindre l'objectif :
 $Nombre\ de\ jours = \frac{Surplus\ calorique\ total}{Surplus\ calorique\ quotidien}$
5. Calcul du nombre de semaines n cessaires pour atteindre l'objectif :
 $Nombre\ de\ semaines = \frac{Nombre\ de\ jours}{7}$

Maintien du poids :

1. Calcul des calories n cessaires pour maintenir le poids :
 $Calories\ n cessaires = TDEE$

4. Plans nutritionnels et r partition

Apr s avoir calcul  les apports caloriques, on propose   l'utilisateur de choisir entre diff rentes r partition nutritionnelle.

Chaque macronutriment apporte un certain nombre de calories par gramme :

- Glucides : 4 kcal par gramme
- Prot ines : 4 kcal par gramme
- Lipides (graisses) : 9 kcal par gramme

Pour la perte de poids :

- Cardio (60 % glucides, 20 % protéines, 20 % graisses) :
Ce plan est conçu pour ceux qui pratiquent régulièrement des activités cardio comme la course, le vélo ou la natation. L'accent est mis sur un apport plus élevé en glucides pour fournir de l'énergie rapide nécessaire à l'endurance, tout en maintenant un apport adéquat en protéines pour préserver la masse musculaire.
- Athlète (55 % glucides, 20 % protéines, 25 % graisses) :
Adapté aux athlètes qui souhaitent optimiser leurs performances sportives tout en perdant du poids. La combinaison de glucides pour l'énergie, de protéines pour la réparation musculaire, et de graisses saines pour la récupération permet une approche équilibrée.
- Durable (55 % glucides, 15 % protéines, 30 % graisses) :
Ce plan favorise une approche progressive et durable de la perte de poids. Il est idéal pour ceux qui souhaitent maigrir sans sacrifier l'équilibre nutritionnel. Les protéines sont légèrement réduites, tandis que l'apport en graisses reste relativement élevé pour favoriser la satiété et un bon métabolisme lipidique.

Pour la prise de poids :

- Cardio (60 % glucides, 20 % protéines, 20 % graisses) :
Ce plan est destiné aux sportifs qui combinent cardio et musculation légère. L'apport élevé en glucides soutient les performances pendant les séances d'entraînement cardio, tandis que les protéines permettent une légère augmentation musculaire.
- Athlète (55 % glucides, 20 % protéines, 25 % graisses) :
Ce programme a été conçu pour les athlètes qui souhaitent développer leur masse musculaire tout en maintenant un niveau de performance élevé. Un apport équilibré entre glucides, protéines et graisses soutient à la fois la récupération musculaire et les besoins énergétiques élevés.
- Se muscler (50 % glucides, 25 % protéines, 25 % graisses) :
Ce plan est idéal pour les adeptes de musculation cherchant à augmenter leur masse musculaire. L'apport en protéines est ici maximisé (25 %) pour favoriser la synthèse musculaire, tandis que l'apport équilibré en glucides et graisses garantit l'énergie nécessaire pour les entraînements intensifs.