

Grau en Enginyeria Informàtica de Gestió i Sistemes d'Informació

APLICACIÓ DE CREACIÓ DE DIETES PERSONALITZADES

Memòria

FRANCESC XAVIER HOMAR MASACHS
TUTORA: CATALINA JUAN NADAL

CURS 2022-2023

Agraïments

Vull agrair a la meva família per estar sempre creient en mi recolzant-me en totes les meves decisions.

Abstract

Due to the remarkable increase in healthy lifestyles and the overwhelming amount of information on the internet, it is sought to present a simple and practical point of view to achieve a balanced diet. This thesis focuses on the design and development of an affordable and easy-to-use personalized diet creation application for all audiences.

Resum

A causa de l'increment notable dels estils de vida saludables i l'angoixosa quantitat d'informació disponible a internet, es busca presentar un punt de vista simple i pràctic per aconseguir una dieta equilibrada. El present treball de final de grau es focalitza en el disseny i desenvolupament d'una aplicació de creació de dietes personalitzades, assequible i fàcil d'utilitzar per a tots els públics.

Resumen

Debido al incremento notable de los estilos de vida saludables y la angustiosa cantidad de información disponible en internet, se busca presentar un punto de vista simple y práctico para conseguir una dieta equilibrada. El presente trabajo de fin de grado se focaliza en el diseño y desarrollo de una aplicación de creación de dietas personalizadas, asequible y fácil de utilizar para todos los públicos.

Índex

Índex de figures.....	III
Índex de taules	V
Glossari de termes	VII
1. Introducció.....	1
2. Marc teòric.....	3
2.1. Context.....	3
2.2. Antecedents.....	3
2.2.1. Tipus de consums energètics	4
2.2.2. Càlcul de requeriments energètics	5
2.2.3. Guies dietètiques	8
2.2.4. Necessitats d'informació tecnològiques.....	12
3. Anàlisi de referents	15
3.1. Aplicación móvil para la generación y preparación de menús saludables personalizados.....	15
3.2. Nootric	16
3.3. Fitia	17
3.4. Lifesum	19
3.5. Anàlisi i conclusions dels antecedents.....	20
4. Objectius i abast.....	23
5. Metodologia.....	25
6. Definició de requeriments funcionals i tecnològics	29
7. Desenvolupament.....	31
7.1. FrontEnd	31
7.2. BackEnd.....	43
7.2.1. Deserialització dels objectes d'API externs	43
7.2.2. Processament de creació de dieta.....	44
7.2.3. Connexió amb la base de dades	45
7.2.4. Processament de dades d'usuari	45
7.2.5. Endpoints.....	47
8. Anàlisi de resultats	51

9. Conclusions.....	53
10. Possibles ampliacions	55
11. Bibliografia.....	57

Índex de figures

Fig. 2.3.1.1. Consum energètic humà.....	5
Fig. 2.3.2.1. Relació Kcal diàries amb l'edat.....	7
Fig. 2.3.3.1. Piràmide de l'alimentació saludable.....	9
Fig. 2.3.3.2. Distribució calòrica d'una dieta de 2300 Kcal.....	12
Fig. 3.2.1 Captures de l'aplicació Nootric.....	17
Fig. 3.3.1 Captures de l'aplicació Fitia.....	18
Fig. 3.4.1 Captures de l'aplicació Lifesum.....	20
Fig. 7.1.1. Pantalla de Log In.....	31
Fig. 7-1.2. Pantalles de creació de compte.....	32
Fig. 7.1.3. Menú principal.....	32
Fig. 7.1.4. Pantalla de perfil d'usuari.....	33
Fig. 7.1.5. Pantalla de visualització de dieta.....	33
Fig. 7.1.6 Pantalla de creació de dieta.....	34
Fig. 7.1.7. Pantalla home de l'aplicació.....	35
Fig. 7.1.8. Pantalla Log in de l'aplicació.....	36
Fig. 7.1.9 Pantalla de Sign in de l'aplicació.....	37
Fig. 7.1.10. Pantalla Dashboard de l'aplicació.....	38
Fig. 7.1.11. Pantalla Perfil i edició de Perfil de l'aplicació.....	39
Fig. 7.1.12. Pantalla de visualització de Dietes de l'aplicació.....	40
Fig. 7.1.13. Pantalla de visualització de receptes de l'aplicació.....	41
Fig. 7.1.14. Pantalla de qüestionari de generació de dietes.....	42
Fig. 7.2.1.1. Esquema de classes de deserialització d'API.....	44
Fig. 7.2.4.1 Taula de BBDD de User.....	47

Índex de taules

Taula 7.2.4.1 Activitat diària en relació calòrica	46
--	----

Glossari de termes

TFG Treball Final de Grau

API *Application Programming Interface* / Interfície de programació d'aplicacions

VSCode Visual Studio Code

IMC Índex de massa corporal

Kcal Quilocalories

BBDD Base de dades

DTO *Data transfer object* / Objecte de transferència de dades

1. Introducció

L'objecte del projecte és el de dissenyar i crear un aplicatiu que sigui capaç de generar dietes personalitzades basades en les característiques i objectius de l'usuari, sigui disminuir, mantenir o augmentar el pes corporal. Aquestes dietes són generades setmanalment, tenint en compte les preferències de l'usuari com ara els aliments que vol evitar, el nombre de menjars diaris i les seves al·lèrgies.

L'aplicació té com a finalitat proporcionar funcionalitats avançades que superin les d'altres aplicacions disponibles en el mercat, tot reunint-les en una sola plataforma robusta i completa, amb l'objectiu de captar l'atenció dels usuaris.

La motivació darrere del projecte és abordar el problema de l'abundància d'informació contradictòria disponible a Internet i la dificultat que representa aprendre a menjar de manera saludable.

2. Marc teòric

A continuació es procedeix a estudiar i analitzar el context, els antecedents i les necessitats d'informació del projecte i els seus punts principals sobre els quals es basa per poder aclarir quin procediment es porta a terme.

2.1. Context

En els últims anys hi ha hagut una gran revolta en el món fitness i la gent ha començat a fer esport en massa. Amb aquest fet també ha augmentat la idea generalitzada de mantenir una dieta equilibrada, sigui per perdre o per guanyar pes. El problema presentat per tots aquests fets és la incapacitat de crear o planificar una dieta que tingui en compte totes les coses que necessita una persona per tenir una dieta saludable. Tot això tenint en compte que només un percentatge petit de la població ha estudiat dietètica i la informació a internet sobre com alimentar-se és molt extensa, confusa i de vegades contradictòria.

Amb aquest fet van sortir diferents aplicacions al mercat que arreglaven el problema però només parcialment. Algunes tenen característiques que altres mancaven i aquestes disposaven d'algunes que les primeres no tenien.

2.2. Antecedents

A continuació es presenten uns TFG dels quals aquest està fortament inspirat, posteriorment es mostren aplicacions de mercat que són similars a la proposada, amb els seus avantatges i desavantatges.

Aquest projecte parteix de diversos antecedents, però en concret un fet l'any 2019 per Óscar Jesús Socas González de la universitat Universidad de La Laguna titulat "Aplicación móvil para la generación y preparación de menús saludables personalizados" [1].

En aquest projecte es planteja i s'executa la idea de crear una aplicació Android capaç de crear dietes setmanals automatitzades a partir de la condició física de l'usuari. I es poden distingir certs avantatges i desavantatges entorn del producte.

Altres treballs destacables són "Nutrain: aplicación web de nutrición, entrenamiento y otros aspectos de la salud" per Daniel Suárez Cáceres de la universitat Universidad de Las Palmas

de Gran Canaria [2] que desenvolupa una aplicació que tracta el punt de la nutrició i les dietes esmentat abans i a més afegeix altres blocs com: entrenament, psicologia i fisioteràpia i "Dielthy: Una web para dietes especiales" de Javier Alaguero Martín de la universitat Universidad de Valladolid [3] que se centra en la creació d'una pàgina web amb el tema principal de la nutrició de gent amb intoleràncies.

S'ha escollit tres aplicacions del mercat d'aplicacions mòbils per a fer una anàlisi en profunditat d'elles i extreure els punts positius i negatius que presenten, amb la finalitat d'arribar quins són els punts més forts i tenir-los en compte en el moment d'analitzar els de l'aplicació.

Aquestes tres aplicacions cercades són Nootric, Fitia i Lifesum, totes tres presenten més punts forts que febles, però les mancances que proporcionen es creu que són prescindibles, per tant, es busca crear l'aplicació afegint funcions que no figuren en els antecedents.

A posterior (3. Anàlisi de referents) es fa una anàlisi en detall tant dels TFGs d'altres universitats com de les tres aplicacions de mercat mencionades prèviament. Necessitats d'informació

En aquest apartat es donen a conèixer els aspectes teòrics base entorn de la dietètica de la qual gira entorn el projecte.

2.2.1. Tipus de consums energètics

- GET: El consum energètic total és la quantitat absoluta que gasta el cos en un rang de 24 hores.
- GMB: El consum metabòlic basal es defineix com la quantitat d'energia que es consumeix en situació de repòs i dejuni. El GMB es pot extrapolar a 24 hores i es parla d'ell com gest energètic basal (GEB), suposant un 60-70% del GET.
- GEAF: El consum energètic per activitat física és la variable que es pot controlar de forma més fàcil per modificar el GET, ja que suposa un 10-30% del GET. En esportistes d'elit pot suposar fins a un 40%, pel que si deixen de fer exercici poden patir canvis dràstics a la seva composició corporal.

- ETA: El consum per efecte termogènic dels aliments fa referència a les calories produïdes en forma de calor durant la ingesta i metabolització dels aliments. Suposa un 10% de l'energia total consumida amb els aliments.
- GER: El consum energètic en repòs es refereix al consum en 24 hores de repòs, però incloent el consum derivat de la ingesta d'aliment. A la pràctica s'utilitzen indistintament GEB i GER. El principal determinant del GER és la quantitat de massa magra corporal, altres factors són l'edat, el sexe, l'estat nutricional, els condicionaments genètics, la funció tiroide, etc.



Fig. 2.3.1.1. Consum energètic humà

2.2.2. Càlcul de requeriments energètics

Calorimetria directa. Observació de la producció de la calor de l'organisme mitjançant la introducció de l'individu a una càmera i indica la quantitat total d'energia generada a partir de l'oxidació dels substrats.

Calorimetria indirecta. Mesura l'energia total produïda per l'organisme estant relacionada amb la síntesi d'ATP amb l'oxidació de substrats i, per tant, amb el consum d'oxigen. Es basa en el càlcul del consum d'oxigen i la producció de CO_2 que reflecteix el consum de nutrients a nivell cel·lular.

Fórmules predictives: Es basen en calorimetria indirecta i són bastant exactes amb persones sanes.

- Harris-Benedict. Calcula el consum energètic basal, però tal com es descriu en l'estudi que es va fer al crear-la el que realment calcula és el consum energètic en repòs.

$$\begin{aligned} \text{Homes: GER} &= 66 + (13,7 \times \text{pes en kg}) + (5 \times \text{alçada en cm}) \\ &\quad - (6,7 \times \text{edat en anys}) \end{aligned} \tag{2.1}$$

$$\begin{aligned} \text{Dones: GER} &= 65 + (9,6 \times \text{pes en kg}) + (1,8 \times \text{alçada en cm}) \\ &\quad - (4,7 \times \text{edat en anys}) \end{aligned} \tag{2.2}$$

- Mifflin-St. Jeor. Serveix per mesurar el consum energètic real. Té major utilització amb pacients greus. Té més en compte el pes i la talla i no tant l'edat.

$$\begin{aligned} \text{Homes: GER} &= 5 + (10 \times \text{pes en kg}) + (6,25 \times \text{alçada en cm}) \\ &\quad - (5 \times \text{edat en anys}) \end{aligned} \tag{2.3}$$

$$\begin{aligned} \text{Dones: GER} &= -161 + (10 \times \text{pes en kg}) + (6,25 \times \text{alçada en cm}) \\ &\quad - (5 \times \text{edat en anys}) \end{aligned} \tag{2.4}$$

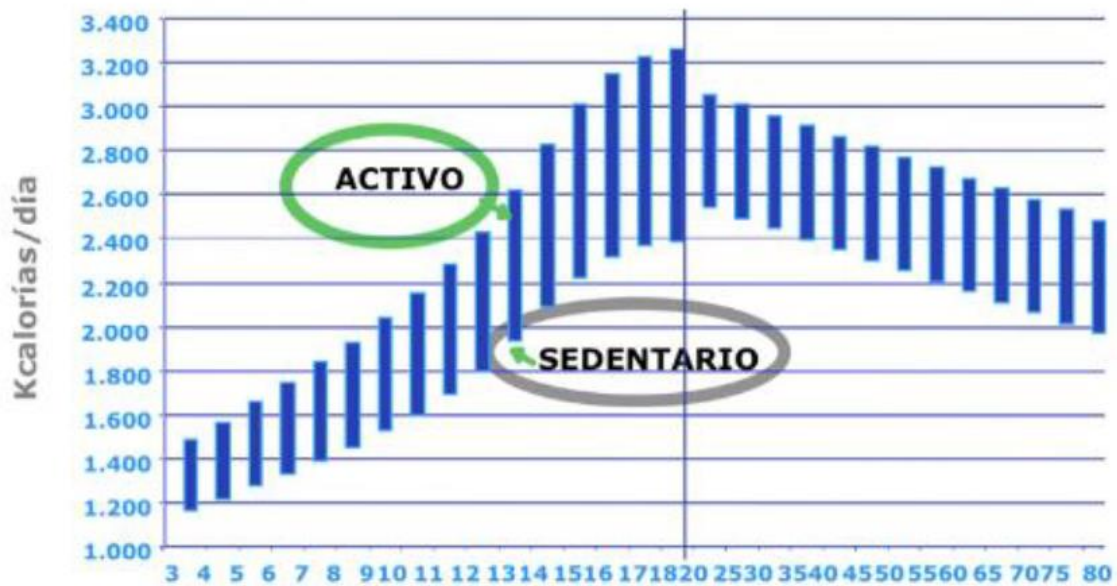


Fig. 2.3.2.1. Relació Kcal diàries amb l'edat

Gràfica que representa com varien els requeriments energètics d'un home de pes mitjà en funció de la seva activitat física i la seva edat. Es veu una disminució significativa del metabolisme a partir dels 25 anys. Per això és molt important ajustar les necessitats metabòliques segons aquests paràmetres.

L'índex de massa corporal (IMC) és una mesura estandarditzada de la relació entre la seva massa i l'alçada d'una persona. Es calcula dividint el pes (en kg) pel quadrat de la seva alçada (en metres).

$$IMC = \frac{\text{pes en kg}}{\text{alçada en m}^2}$$

(2.5)

El resultat es pot interpretar com un índex de pes adequat, sobrepès o obesitat. El rang d'IMC considerat saludable varia segons el gènere i l'edat. En general, un IMC entre 18,5 i 24,9 es considera saludable per a adults, mentre que un IMC per sobre de 25 es considera sobrepès i un IMC per sobre de 30 es considera obès.

2.2.3. Guies dietètiques

Les guies dietètiques expressen les recomanacions nutricionals de manera pràctica i concreta, emfatitzant en aliments i grups d'aliments més que en nutrients. És imprescindible traslladar aquesta informació a material educatiu orientat al consumidor i com a punt de referència per als professionals involucrats en l'educació nutricional i en l'elaboració de dietes per a diferents grups de població. Els objectius de les guies dietètiques són ser fàcils d'entendre per a la població general. Cada guia està acompanyada d'un document de consens i es tradueix gràficament en una piràmide d'alimentació saludable. Objectius nutricionals per a la població espanyola:

- Carbohidrats: 45-65% del valor calòric total.
- Greixos: 20-35% del valor calòric total.
- Proteïnes: 10-35% del valor calòric total.

Es pot apreciar com els percentatges tenen valors compresos entre un rang de números, això es deu al fet que depenent de la complexitat de la persona i altres factors com el gènere i l'edat pot requerir més o menys percentatge. També hi ha variació degut al factor que es vol tractar en aquesta investigació i és el fet de dietes per perdre o guanyar pes.



Fig. 2.3.3.1. Piràmide de l'alimentació saludable [5]

Aquesta piràmide nutricional proporcionada per la Generalitat de Catalunya representa i posa en consideració el que s'hauria de seguir per tenir una dieta equilibrada. La piràmide està dividida en diferents seccions i fora d'ella es fan consideracions que també són necessàries, però que no tracten amb els aliments directament.

Consideracions externes:

Activitat física diària. L'objectiu que es planteja és de 60 minuts d'exercici diaris amb una intensitat moderada, o el seu equivalent de 10.000 passes al dia, preferiblement amb alguns trams o períodes puntuals de major intensitat.

Útils i tècniques culinàries saludables. Les tècniques utilitzades a la preparació i conservació dels aliments poden afectar a la seva qualitat nutricional, organolèptica (sabor, textura, olor o color) i contribuir a la possible transferència de substàncies indesitjables. Com a norma general, la cuina al vapor, les preparacions segures en cru o les menys elaborades proporcionen un millor valor nutricional i inclòs organolèptic.

Ingesta adequada d'aigua. Incorporar la ingesta d'aigua i altres líquids o aliments fins a ingerir al voltant de dos litres diaris adaptats a situacions de temperatura i/o edat.

Consum diari:

Cereals de gra sencer i productes derivats, junt amb altres aliments rics en hidrats de carboni complexos, són la base d'una alimentació adequada. La recomanació se centra a prioritzar els cereals i derivats integrals o elaborats amb farines de gra sencer. L'aportació energètica s'ha de calcular en referència a l'activitat física.

Les fruites en general, són un grup alimentari d'especial importància pel manteniment i la promoció de la salut. S'ha d'incloure tres o més racions o peces de fruita variada al dia, en el seu adequat estat de maduració, amb rentatge previ i també pelat si no es tracta de productes amb cultiu orgànic.

Les verdures i hortalisses de temporada són una font de nutrients de gran importància per afavorir un millor estat de salut. És recomanable incorporar com a mínim dues racions de verdures i hortalisses cada dia. A poder ser una de les racions en format cru amb varietats de diferent color.

L'oli d'oliva és la millor referència grassa per l'acompanyament en molts aliments, tant en preparacions o processos culinaris com pel seu consum en cru. És recomanable escollir l'oli d'oliva verge extra d'extracció en fred a partir de qualsevol de les múltiples varietats d'oliva.

Els fruits secs i les llavors són d'interès en el marc de la dieta mediterrània. Prioritzar el consum de varietats locals en el format natural, o poc manipulades sense sal o sucres afegits.

Es recomana el consum de lactis de bona qualitat entre dues i quatre vegades al dia prioritzant les preparacions baixes en grasses i sense sucres afegits.

Setmanals:

Els ous són una excel·lent font de nutrients. Es recomana una freqüència de consum i format culinari de caràcter individualitzat, prioritzant les varietats ecològiques o les camperes, no només per la seva possible diferenciació nutricional o organolèptica, sinó també per la contribució a humanitzar en major part la cadena productiva alimentària.

Les carns blanques de qualitat són una bona opció com a font de proteïna i d'altres nutrients d'interès, prioritzant les preparacions amb poca matèria grassa afegida i l'acompanyament habitual amb guarnició de verdures o amanida.

Els llegums en totes les seves varietats són una font important d'energia i nutrients, referents de la cuina tradicional i mediterrània. Es recomana el consum de dues o tres racions a la setmana amb les tècniques culinàries que millorin la seva digestibilitat i valor nutricional.

Els peixos i mariscs són una bona font de proteïnes, minerals i vitamines amb un perfil gras ric en àcids grassos poliinsaturats. S'ha de prioritzar les captures sostenibles de temporada utilitzant les mides, quantitats i preparacions culinàries que assegurin la innocuïtat del producte. La recomanació es situa en el consum de peix com a mínim dues o tres vegades per setmana.

Ocasional:

Carns vermelles i processades, haurien de moderar-se en el marc de l'alimentació saludable, escollint productes d'excel·lent qualitat, procediments culinaris sense contacte directe amb el foc i, sempre, amb l'acompanyament d'una guarnició d'hortalisses fresques. S'ha demostrat que poden produir càncer si es consumeixen en excés.

Les grasses d'untar de tota mena haurien de ser de consum moderat i ocasional, amb preferència cap a la mantega sense sal afegida.

La brioixeria industrial, pastisseria, laminadures, begudes ensucrades, gelats i confiteria haurien de consumir-se opcionalment i ocasionalment de forma moderada.

Consideracions complementàries:

Moderar el consum de sucre i de productes ensucrats per no sobrepassar el 10% de la ingesta energètica diària a partir de sucres afegits.

Moderar la ingesta de sal i de productes amb elevada aportació de sal estructural o afegida per no superar la quantitat de sis grams de sal total al dia.

Es moderarà el consum de begudes alcohòliques, de forma responsable i de baixa graduació i bona qualitat. Màxim dues copes de vi en homes i una en dones.

A vegades i sota una supervisió mèdica, farmacèutica, dietista nutricional, infermera, matrona o fisioterapeuta, si existeixen necessitats especials pot ser d'utilitat el consum habitual o temporal de suplementos o aliments específics

Alimentació sostenible, mantenir la sostenibilitat de tots els processos de la cadena alimentària, reduir malbaratament d'aliments, reutilitzar sobres amb seguretat i reciclar.

Ingerir productes de temporada i proximitat en consonància amb la dieta tradicional i en consonància amb factors crono biològics.



Fig. 2.3.3.2. Distribució calòrica d'una dieta de 2300 Kcal

No és obligatori fer 5 menjars. Encara que si la persona ha d'ingerir més calories, és recomanable distribuir-les en més ingestes. Per tant, s'adapta el nombre d'ingestes en referència al nombre de calories que s'ha de consumir.

2.2.4. Necessitats d'informació tecnològiques

En aquest apartat es presenta totes les eines requerides per al correcte desenvolupament de l'aplicació plantejada. Només s'entra en detall de les funcionalitats de l'eina i la seva aportació al projecte, en el capítol de metodologia s'explica perquè aquestes han estat escollides sobre unes altres i es fan les comparacions pertinents.

Edamam [4] és una API que proporciona accés a una base de dades de nutrició i informació de receptes. Aquesta API ofereix una varietat de funcions, com ara la capacitat de cercar receptes en funció d'ingredients específics, obtenir informació nutricional detallada sobre una recepta o ingredient, i obtenir recomanacions de receptes en funció de les preferències de dieta o restriccions alimentàries.

Per la part del FrontEnd s'utilitza Visual Studio Code (VSCode) [8] que és un *Integrated Development Environment* (IDE) de codi obert desenvolupat per Microsoft. És una eina d'alt

rendiment que ofereix una gran quantitat d'eines de desenvolupament per a molts llenguatges de programació, tals com JavaScript, TypeScript, Python, C++ i d'altres. Addicionalment, VSCode també ofereix una àmplia gamma de característiques afegides, com ara la capacitat de depurar codi, control de versions, integració amb altres eines de desenvolupament, i moltes altres. A més, VSCode és molt personalitzable i se li pot afegir moltes extensions disponibles a la seva botiga d'addons.

Juntament amb VSCode s'utilitza Ionic React que és un framework de desenvolupament d'aplicacions mòbils que utilitza React, una biblioteca de JavaScript d'interfície d'usuari, per crear aplicacions d'alt rendiment per a dispositius mòbils. Ionic React té una gran quantitat de components d'interfícies d'usuari, com ara botons, fitxers d'estil, icones, etc. que es poden personalitzar aconseguint així aplicacions mòbils atractives i intuïtives. També ofereix una eina de navegació integrada que facilita la navegació entre les diferents pantalles de l'aplicació.

Ionic React [7] és un framework en constant desenvolupament i evolució, amb data de la redacció del document l'última actualització va ser l'1 de febrer del 2023, a més d'això hi ha molta ajuda a la web i la seva documentació oficial és molt extensa i detallada amb ajudes per a principiants.

Ajuntant aquests dos conceptes últims conceptes tenim el millor entorn de treball per programar amb Ionic React a causa de la:

- Gran comunitat de VSCode: Ja que és una plataforma oberta i disposa d'una gran comunitat d'usuaris, significa que hi ha moltes extensions disponibles per ajudar a millorar la programació amb Ionic React, tals com "Ionic Visual Code", "Ionic 4 Snippets" o "Prettier".
- Funcionalitat: VSCode presenta una gran quantitat de funcions integrades com ara l'autocompleció de codi, la depuració, la navegació de codi, etc.
- Integració amb altres eines: VSCode es pot integrar amb moltes eines de treball diferents com Git, npm, etc. Això facilita la gestió del codi i la seva integració amb altres sistemes.

- Renderització ràpida: VSCode té un rendiment excel·lent i es carrega ràpidament, el que és important per a les aplicacions Ionic React, ja que aquests aplicatius poden ser grans i complexos.
- Interfície d'usuari neta: VSCode té una interfície d'usuari neta i fàcil d'utilitzar, el que significa que és més fàcil de navegar i utilitzar per a tots els usuaris.

Per la part del BackEnd es proposa l'ús d'IntelliJ IDEA [10] que és un *Integrated Development Environment* enfocat especialment a la programació de BackEnd amb Java. Ofereix moltes opcions útils per a la programació, com la depuració de codi, la gestió de versions i la integració amb altres eines de desenvolupament. Presenta integracions de base que no pot arribar a proporcionar VSCode com visualitzadors de BBDD que ajuden a facilitar programar-ne una. Junt amb una interfície d'usuari personalitzable i intuïtiva, i detecta i soluciona problemes de codificació amb eficàcia.

Spring [11] és un framework de codi obert per a aplicacions Java. Proporciona una gran quantitat d'eines i funcions útils per a la creació d'aplicacions web, mòbils, d'escriptori i cloud. Les seves característiques principals són:

- Injecció de dependències: permet a l'aplicació obtenir les funcions necessàries sense haver de gestionar manualment les dependències.
- Gestió de transaccions: simplifica la gestió de transaccions en aplicacions basades en base de dades.
- Seguretat: proporciona mecanismes de seguretat integrats per a la protecció de les aplicacions i les dades.
- MVC: presenta una arquitectura Model-View-Controller per separar la lògica del negoci i la presentació.

3. Anàlisi de referents

Prèviament s'ha mencionat els diferents referents en els que es basa aquest treball, a continuació es fa un anàlisi de les qualitats que es presenten a favor i en contra d'aquests models.

A continuació es parla de les aplicacions que es poden trobar al mercat per a generar menús dietètics i es comenta les seves funcionalitats. Totes les aplicacions que són avaluades són analitzades d'acord amb la secció gratuïta disponible.

3.1. Aplicación móvil para la generación y preparación de menús saludables personalizados

Aquesta aplicació creada per Óscar Jesús Socas González de la universitat *Universidad de La Laguna* creada al 2019 [1] és el major referent que presenta aquest projecte degut a la versemblança entre els dos treballs. No s'ha pogut accedir a l'aplicació però la recerca s'ha fet a través de la memòria del propi TFG.

Avantatges:

- Genera un menú personalitzat.
- Proporciona una llista de la compra amb els aliments requerits per el menú generat.
- Proporciona un seguiment dels aliments que hi ha a la despensa.
- Posseeix una interfície d'usuari simple i fàcil d'entendre.
- Mostra les macros diàries.
- Mostra la informació nutricional de cada producte .

Desavantatges:

- No genera plans d'exercici personalitzats.
- No proporciona un historial de pesatge de l'usuari.
- No disposa de la capacitat de posar productes en whitelists/blacklist per la dieta.

- No es poden modificar els productes de la dieta una vegada generada.
- No es poden escollir els productes que es volen canviar a la dieta.

3.2. Nootric

Aquesta aplicació creada i llençada per Nootric presumeix de produir dietes personalitzades addicionalment de generar plans d'exercici, tot fet per nutricionistes. Disposa d'aplicació per a iOS i Android, però no té una implementació en web.

Avantatges:

- Genera un menú personalitzat.
- Proporciona un historial de pesatge de l'usuari.
- Posseeix una interfície d'usuari simple i fàcil d'entendre.
- Permet focalitzar la dieta en referència si es vol perdre pes, guanyar múscul, etc.
- Es poden modificar els productes de la dieta una vegada generada.
- Proporciona els aliments per cuinar el plat.
- Proporciona els passos que s'han de seguir per cuinar el plat.
- Genera plans d'exercici personalitzats.

Desavantatges:

- No permet crear whitelists/blacklists per a la creació de dietes.
- No mostra la informació nutricional de cada producte.
- L'aliment que es modifica a la dieta no pot ser escollit per l'usuari, sempre canvia a un aliment aleatori.
- No permet controlar al·lèrgens.

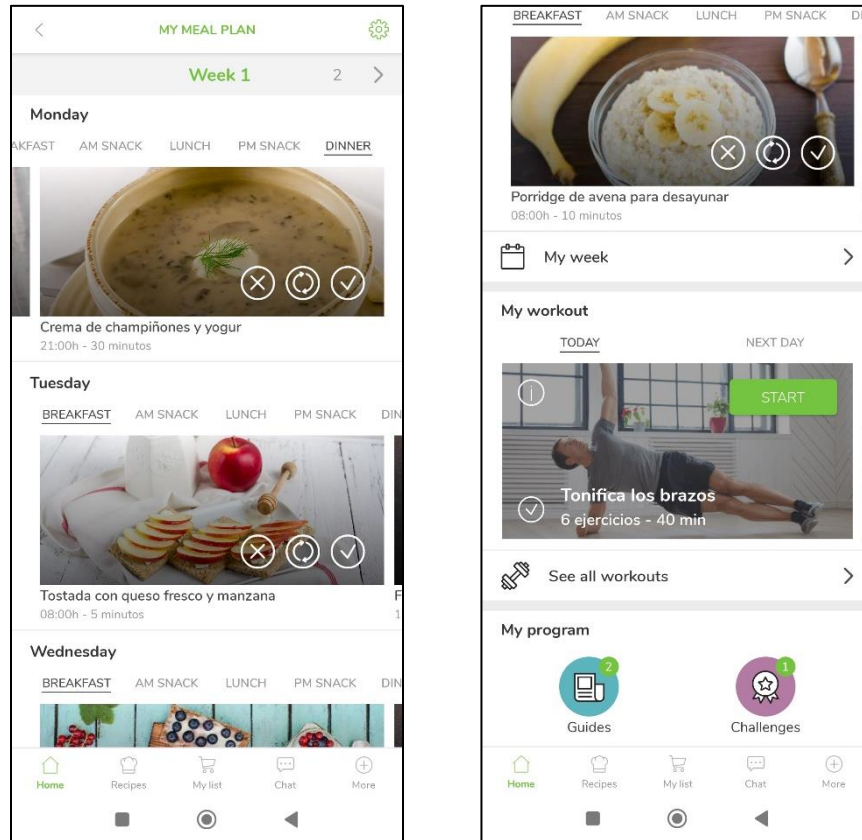


Fig. 3.2.1 Captures de l'aplicació Nootric

3.3. Fitia

Fitia és una aplicació mòbil de creació de dietes personalitzades feta per la companyia del mateix nom. Presumeix de tenir una gran base de dades amb més de 400.000 productes verificats. Disposa d'aplicacions tant a iOS com a Android, però no està disponible per web.

Avantatges:

- Genera un menú personalitzat.
- Té la capacitat de posar productes en whitelists/blacklist per la dieta.
- Proporciona un historial de pesatge de l'usuari.
- Permet focalitzar la dieta en referència si es vol perdre pes, guanyar múscul, etc.
- Posseeix una interfície d'usuari simple i fàcil d'entendre.

- Permet la possibilitat d'afegir ingredients externs a l'aplicació.
- Mostra les macros diàries.
- Mostra la informació nutricional de cada producte.
- Es poden modificar els productes de la dieta una vegada generada.
- Es poden escollir els productes que es volen canviar a la dieta.

Desavantatges:

- No proporciona receptes, només proporciona els aliments base.
- No genera plans d'exercici personalitzats.
- Moltes de les característiques que es presenten com a primordials de l'aplicació estan darrere d'una subscripció.

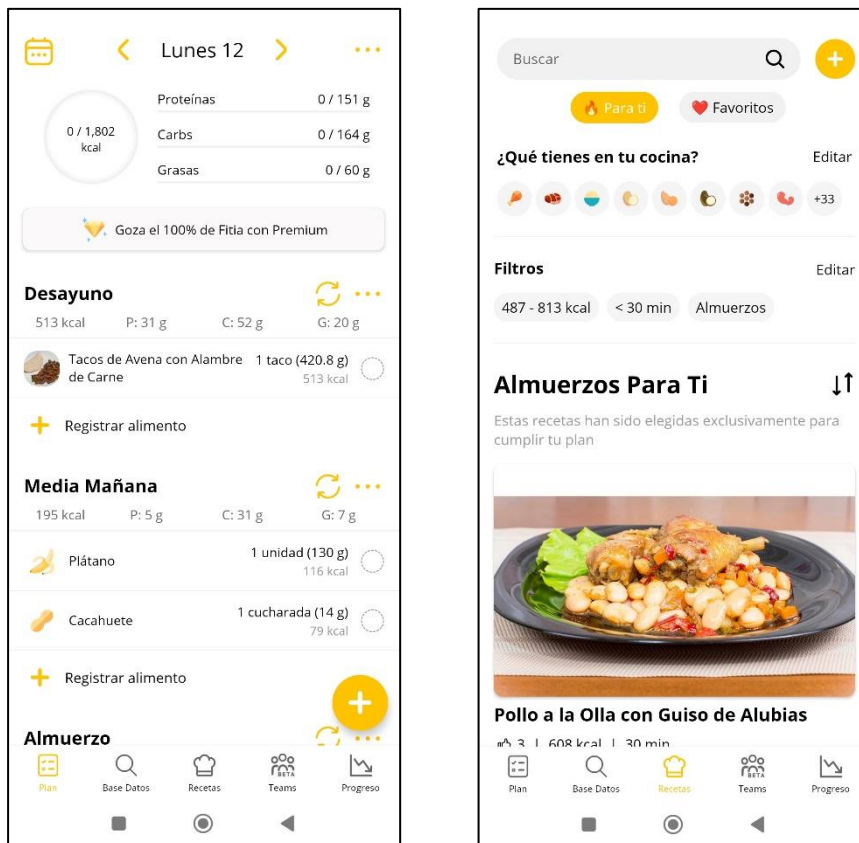


Fig. 3.3.1 Captures de l'aplicació Fitia

3.4. Lifesum

Lifesum és una aplicació de control de dieta creada per l'empresa del mateix nom i que permet, a la vegada, planificar rutines d'esport. Presenta aplicacions per a iOS i Android, però no per a web.

Avantatges:

- Permet focalitzar la dieta en referència si es vol perdre pes, guanyar múscul, etc.
- Mostra les macros diàries.
- Proporciona un historial de pesatge de l'usuari.
- Mostra la informació nutricional de cada producte.

Desavantatges:

- No genera un menú personalitzat.
- Només permet comptabilitzar les macros, els altres nutrients estan darrere d'una subscripció.
- No hi ha cap dieta que sigui gratuïta.

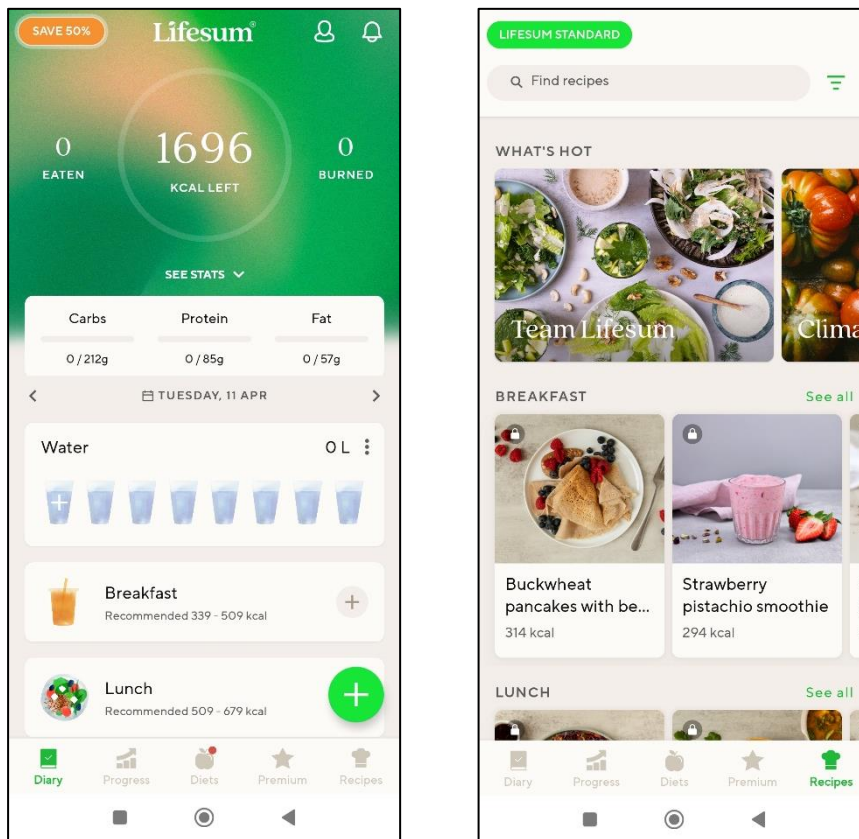


Fig. 3.4.1 Captures de l'aplicació Lifesum

3.5. Anàlisi i conclusions dels antecedents

Dels antecedents esmentats, es poden identificar diversos punts forts que es tenen en compte en l'anàlisi de les diferents opcions. En primer lloc, es busca crear una aplicació que sigui capaç de generar una dieta personalitzada que s'adapti a les necessitats específiques de cada usuari en el moment del registre. Això implica tenir en compte factors com l'objectiu de pèrdua de pes o de guany de múscul, així com altres preferències o restriccions alimentàries tals com possibilitat de fer dietes vegetarianes o sense gluten.

A més, una altra prioritat és proporcionar una interfície d'usuari agradable i fàcil d'entendre i utilitzar. Es posa un èmfasi especial en la usabilitat de l'aplicació, de manera que sigui intuïtiva i accessible per a tots els usuaris, independentment del seu nivell de coneixement tècnic o experiència prèvia amb aplicacions similars.

Per tal de millorar encara més l'experiència d'usuari, es decideix afegir una pantalla on es mostrin les calories i macros a ingerir cada dia. Aquesta funcionalitat proporcionarà

informació valuosa i permetrà als usuaris seguir el seu progrés i fer un seguiment de la seva alimentació d'una manera més precisa.

Adicionalment, es té en compte la possibilitat de connectar l'aplicació amb un correu electrònic, amb l'objectiu de facilitar la usabilitat i accessibilitat de l'aplicació. Això es connecta amb la seva següent característica, una clau del projecte, la funcionalitat multiplataforma. Es busca una aplicació que sigui accessible des de diferents dispositius i sistemes operatius, com iOS, Android i web, per garantir que els usuaris puguin accedir a l'aplicació des de la plataforma de la seva elecció i tenir una experiència coherent i completa independentment del dispositiu que utilitzin.

En resum, l'objectiu és crear una aplicació que ofereixi una dieta personalitzada, una interfície d'usuari amigable, la visualització de calories i macros diàries, la integració amb correu electrònic i la compatibilitat multiplataforma. Aquests punts forts reflecteixen les necessitats i preferències dels usuaris i contribuiran a proporcionar una experiència positiva i satisfactòria per a tots els usuaris de l'aplicació.

Els punts forts que es poden mostrar de l'aplicació que es crea són:

- Genera un menú personalitzat.
- Permet focalitzar la dieta en referència si es vol perdre pes, guanyar múscul, etc.
- Multiplataforma.
- Mostra les macros diàries.
- Mostra la informació nutricional de cada producte.
- Proporciona els aliments per cuinar el plat.
- Posseeix una interfície d'usuari simple i fàcil d'entendre.
- Permet controlar al·lèrgens.
- Es poden modificar els productes de la dieta una vegada generada.

Per altra banda, les funcionalitats que mostren alguns antecedents presentats i que no formen part de l'aplicació:

- Proporciona un historial de pesatge de l'usuari.
- Proporciona els passos que s'han de seguir per cuinar el plat.
- No genera plans d'exercici personalitzats.

S'agafen elements que es veuen en comú en alguns antecedents a més a més de la multiplataforma que no està present en cap. Aconseguint una proposta de valor nova pel client no presentada per cap altra.

4. Objectius i abast

L'objectiu principal del treball és programar una aplicació automàtica i personalitzada de creació de dietes que tingui unes interfícies fàcils d'entendre per a l'usuari, seguint principis bàsics d'experiència d'usuari, usabilitat i accessibilitat, sense la necessitat de tenir coneixements exhaustius dins del món de la dietètica. Es vol crear una aplicació molt intuïtiva i que només requereixi la introducció de dades bàsiques com el pes, l'objectiu de pes final, l'edat, l'alçada i el sexe. Amb aquestes dades, l'aplicació proporciona una dieta personalitzada exclusivament per a l'usuari.

Adicionalment, les dietes generades poden ser modificades en qualsevol moment per l'usuari, intercanviant menjars segons les preferències personals. Garantint així una solució nutricional completament personalitzada per a cada usuari.

Per a aquells usuaris que tenen restriccions alimentàries o preferències específiques, l'aplicació proporciona la possibilitat d'afegir blacklists o whitelists d'aliments, menjars o tipus d'aliments en específic. Aconseguint així que l'usuari pugui seleccionar els aliments que desitja incloure o excloure de la seva dieta.

L'aplicació es focalitza principalment en els usuaris que vulguin baixar o pujar de pes i que no tinguin coneixements de dietètica, no importa l'edat ni el sexe.

Objectius enfocats al client:

- Crear una aplicació multiplataforma per android, ios i web.
- Desenvolupar interfícies intuïtives i “*user friendly*”.
- Disposar de whitelists/blacklists de productes.
- Disposar d'informació nutricional de cada aliment.
- Connectar amb la base de dades pròpia amb la informació necessària dels usuaris.
- Connectar amb la base de dades externa de l'API Edamam i no crear una de pròpia.
- Calcular les macros diàries de l'usuari.

Objectius enfocats a l'usuari:

- Iniciar sessió a l'aplicació amb un correu electrònic.
- Registrar les seves dades a l'aplicació, tals com el pes, l'edat i l'exercici que fan.
- Escollir si la dieta serà per baixar de pes, guanyar múscul o mantenir-se.
- Triar quin tipus de productes no apareixeran a la dieta.
- Seleccionar quins productes han d'aparèixer obligatòriament a la dieta.
- Actualitzar les dades registrades a l'aplicació.
- Veure les macros ingerides i les restants.
- Veure les calories ingerides i les restants.

Abast:

- La BBDD serà responsable d'emmagatzemar les dades dels usuaris, però no de les receptes.
- L'aplicació no pot ser responsable d'emmagatzemar i crear els aliments, ja que es fa a partir de l'API Edamam.
- L'usuari no pot ser capaç de crear/afegir aliments que no estiguin ja a la base de dades.
- L'aplicació no pot generar passos a seguir per cuinar un aliment.

5. Metodologia

La recerca d'informació teòrica sobre dietètica està basada a través d'internet utilitzant Google Academic i documentació del grau universitari de medicina de la Universidad de Cantabria. La informació obtinguda a través del grau de medicina proporciona un punt de vista anatòmic i real de l'usuari amb el qual es tracta a més a més de proporcionar informació actual i elaborada per professionals qualificats en l'àmbit. La combinació d'aquestes dues fonts garanteix una recerca exhaustiva i ben informada sobre dietètica.

Per part de cercar informació sobre el programari i llenguatge de programació de fonts fiables s'ha optat per mirar tant a la pàgina web oficial com a la documentació online.

Haver escollit pàgines tant objectives presenta un punt de vista imparcial de primeres i l'elecció i anàlisi tant de programari per treballar com de llenguatge de programació no està subjecte a cap opinió que es pogués tenir abans. La forma d'anàlisi que s'ha seguit ha estat una en la que es busca la possibilitat més gran d'exportació de l'aplicació i una gran quantitat de documentació online.

La metodologia de procés que es planteja per la creació de l'aplicació és agile què és un enfocament iteratiu i incremental per a la gestió de projectes que busca una entrega ràpida i continua de valor al client. La metodologia agile valora la flexibilitat i la capacitat d'adaptació davant les incerteses i els canvis, i busca aconseguir la màxima satisfacció del client a través de l'entrega contínua de valor.

Especificant el tipus de metodologia agile que s'utilitza s'arriba a l'scrumban, una combinació de les metodologies scrum i kanban les dues metodologies agile. Al fer això es millora la gestió del flux de treball obtenint-ne una de més eficaç. Scrumban adopta la iteració i la incertesa d'scrum i la visualització i la flexibilitat de kanban.

Amb aquesta metodologia escollida es busca crear un prototip usable el més abans possible i anar-lo millorant amb el pas de les setmanes, millorant o canviant les coses que poden anar malament o no funcionen tal com estaven previstes en el plantejament del projecte. Evitant així el no saber si l'aplicació funcionarà bé amb totes les seves característiques o si serà

intuïtiva d'usar fins a la seva etapa final com passaria amb altres metodologies de procés com cascade.

Juntament amb la metodologia scrumban per organitzar el projecte en general, es presenta la metodologia per al desenvolupament de programari: el Test Driven Development (TDD). Seguint les idees proposades pel llibre Clean Code [6]. Aquesta metodologia permet crear el codi de forma ràpida i eficaç i executar tests automatitzats abans de codificar la funcionalitat real, garantint que l'aplicació compleixi les especificacions i necessitats. La metodologia TDD es basa en una sèrie de passos repetitius: escriure una prova que falli, escriure codi per passar la prova i refactoritzar si és necessari. S'ha escollit pel fet que permet tenir una visualització més allunyada del producte i evitar així la visió de túnel entorn de la funcionalitat final, amb el TDD es pot arribar a millorar i perfeccionar cada part del codi per separat evitant així mal funcionaments innecessaris.

Per a l'organització de l'arquitectura del codi s'utilitza l'arquitectura per capes que proporciona un enfocament per dissenyar i construir sistemes informàtics complexos, que es divideix en diferents capes, cadascuna amb una responsabilitat específica, evitant així el principi de responsabilitat única. Això permet una millor sostenibilitat, escalabilitat i reutilització del codi. En comparació a altres enfocaments, com ara l'arquitectura monolítica o l'arquitectura basada en microserveis, l'arquitectura per capes ofereix una major flexibilitat i escalabilitat a mesura que el sistema es fa més complex. També permet dividir en tasques específiques cada etapa ajudant a garantir una millor qualitat del codi, una millor capacitat per arreglar errors i reforçant l'enfocament de la metodologia escollida, l'scrumban.

Entorn de la tria d'alternatives pel llenguatge de programació i el framework no es disposa de massa alternatives que permetin crear una aplicació en el mateix entorn de producció i exportar-la a diferents sistemes operatius.

Les dues opcions principals que tenen la capacitat d'exportar a diferents sistemes operatius i a més a més disposen de molta ajuda a internet per facilitar la corba d'aprenentatge amb els frameworks són Ionic React i Xamarin.

Xamarin [9] està basat en el llenguatge C# i utilitza el framework .NET per a la creació d'aplicacions. A més Xamarin també ofereix accés complet a les API natives de cada

plataforma, permetent així crear aplicacions amb una aparença i funcionalitat totalment nativa.

Per altra banda, Ionic React s'apropa més a les tecnologies web i s'integra amb altres eines de desenvolupament, tals com Node.js i React, facilitant així el desenvolupament per persones amb formació en desenvolupament web. En comptar amb les característiques mencionades prèviament la corba d'aprenentatge d'Ionic React és més vertical, però proporciona moltes més possibilitats a l'hora de crear aplicacions híbrides.

Per poder disposar de les dades de l'usuari en qualsevol moment des de qualsevol sistema operatiu es disposarà dels serveis Azure VM i Azure SQL, que respectivament són un espai per tenir el BackEnd funcionant en tot moment i un allotjament de base de dades. S'han escollit aquests dos serveis a causa del fet de ser d'Azure i proporcionar una senzilla i bona integració.

S'han descartat altres serveis d'allotjament de BackEnd com Heroku o Back4App perquè no són completament gratuïts i no disposar d'un servei de BBDD al qual estan vinculats.

A meitat del desenvolupament es pren la decisió de canviar la host de la BBDD a Firebase Firestore a causa aquest proveïdor també proporciona un servei d'autorització de comptes anomenat Firebase Authentication i permet una molt bona integració entre ells.

6. Definició de requeriments funcionals i tecnològics

Els requeriments funcionals engloben tota mena de funcionalitats que rep el client en el moment d'utilitzar l'aplicació:

- Fer inici de sessió amb un correu.
- Presentar un formulari d'introducció de pes, edat, i l'exercici que fa l'usuari la primera vegada que inicia l'aplicació.
- Tenir un apartat on l'usuari pot actualitzar les dades personals.
- Calcular el consum energètic basal de l'usuari seguint la fórmula de Harris-Benedict.
- Omplir un formulari cada vegada que es vulgui crear una dieta.
- Fer una estimació de la dieta segons les calories límit de l'usuari i els percentatges de cada menjar (veure taula 2.3.3.2.) per no haver de fer crides a l'API excessivament grans.
- Cridar a l'API amb el filtre d'aliments proposat per l'usuari.
- Generar una dieta amb els aliments rebuts de la base de dades i mostrar-la a l'usuari.
- Rebre la dieta i visualitzar-la a la pantalla de dietes.
- Poder canviar un aliment que no es vulgui a la dieta i rebre un que dietèticament sigui similar.
- Processar les calories de l'aliment que es vol canviar i fer una crida a l'API Edamam amb un de similar.
- Mostrar les macros ingerides i les restants a la pantalla principal.
- Presentar una opció per saber si s'ha consumit algun aliment de la dieta.
- Pressionar l'opció de l'aliment actualitza les calories ingerides i les macros diàries.
- Pressionar l'opció de l'aliment per segona vegada es desfà el moviment anterior.

Per altra banda, es tenen els requeriments tecnològics que constitueixen les necessitats que té l'aplicació:

- Connectar amb l'API Edamam cada vegada que es vulguin aliments.
- Connectar amb la base de dades d'usuaris cada vegada que es vulgui iniciar sessió o actualitzar dades.
- Funcionar en entorns multiplataforma.
- Disposar de connexió a internet.
- Connectar amb el BackEnd allotjat a Azure a través d'API cada vegada que es vulgui crear/modificar una dieta.
- Omplir la interfície amb les dades del BackEnd.

7. Desenvolupament

En aquest apartat es presenta la forma en la qual s'enfoca el disseny de la producció del projecte.

7.1. FrontEnd

Per la part d'organització d'interfície visual d'usuari es crea un mock up a la plataforma online Figma. Es busca una aplicació amb una navegació compacta i senzilla, sense moltes pantalles per les que navegar.

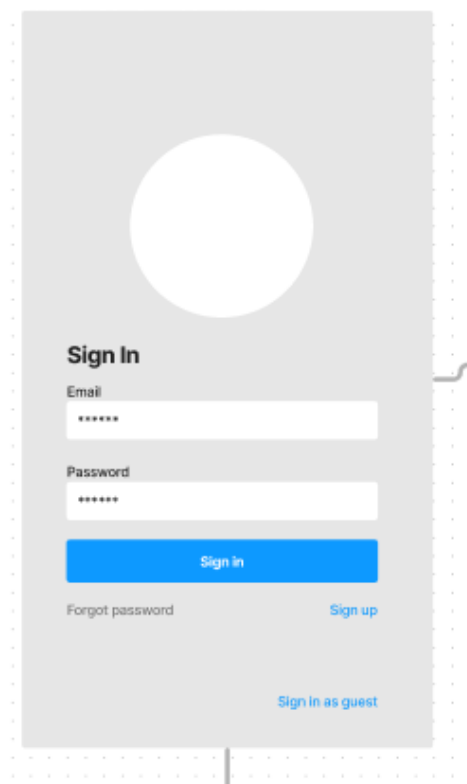


Fig. 7.1.1. Pantalla de Log In [15]

Per a la de Log In que tots els usuaris veuen la primera vegada que s'iniciï l'aplicació s'opta per una de registre que permeti guardar les seves dades a la BBDD. Les dades són guardades i poden ser accedides en qualsevol dispositiu, sigui mòbil o web. També existeix l'opció d'entrar sense registrar-se o com a invitat, aquesta acció no permet guardar les dades de l'usuari, per tant, totes les funcions que disposa l'usuari queden limitades a aquell dispositiu.

The image shows two mobile application screens for account creation. The left screen features a vertical list of input fields: 'First name', 'Last name', 'Email', 'Password', 'Re enter password', and 'Age'. The right screen features a vertical list of input fields: 'Height', 'Weight', 'Level of activity', 'Genre', 'Birth body genre', and 'Desired weight'. Both screens have an orange header bar with a back arrow on the left and a menu icon on the right. A curved arrow at the top indicates a transition from the left screen to the right screen.

Fig. 7-1.2. Pantalles de creació de compte [15]

Per la pantalla de registre només es demanen les dades necessàries per a poder crear les dietes basades en el tipus d'usuari que estigui usant l'aplicació.

The image shows a mobile application screen with a main menu. The screen has an orange header bar with a back arrow on the left and a menu icon on the right. Below the header, there are three orange buttons stacked vertically: 'Profile', 'Check diet', and 'Generate new diet'. A grey arrow points from the left towards the buttons.

Fig. 7.1.3. Menú principal [15]

A la pantalla d'inici una vegada iniciada la sessió es mostren tres botons que reenvien a diferents seccions de l'aplicació, essent aquestes el perfil de l'usuari, la dieta de la qual disposa l'usuari i el formulari per generar una nova dieta.

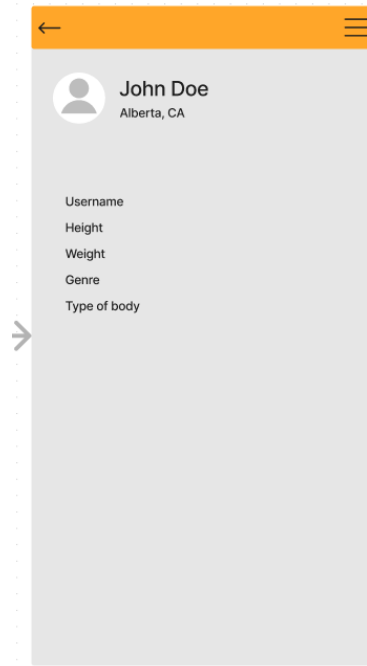


Fig. 7.1.4. Pantalla de perfil d'usuari [15]

A la secció del perfil es mostren les dades proporcionades prèviament per l'usuari amb l'opció de canviar-les.

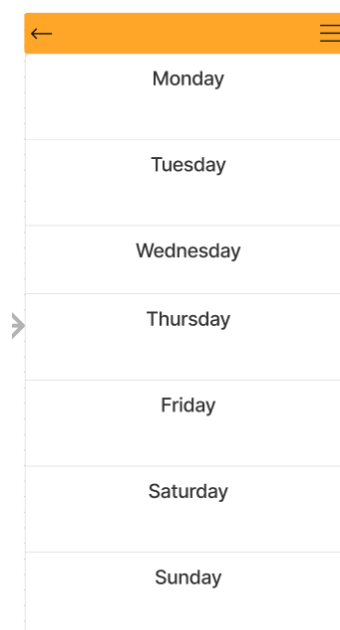


Fig. 7.1.5. Pantalla de visualització de dieta [15]

A la pantalla de vista de dieta es mostren tots els menjars assignats de la setmana ajuntats en una llista i separats pel tipus d'àpat corresponent.

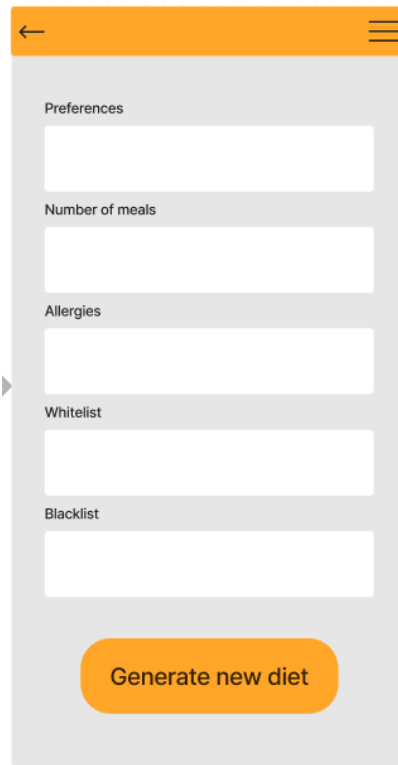


Fig. 7.1.6 Pantalla de creació de dieta [15]

A la pantalla del qüestionari de nova dieta es pregunten aspectes essencials a l'hora de crear una de nova, aquests es reparteixen en:

- Preferències. Quin tipus de dieta es vol obtenir, per exemple: alta en proteïna, vegetariana, etc.
- Número d'àpats. El nombre de menjars que es vol tenir durant en dia, aquest valor pot oscil·lar entre tres i cinc.
- Al·lèrgies. Les al·lèrgies que pateix l'usuari.
- Whitelist. Llista de productes que han d'estar a la dieta.
- Blacklist. Llista de productes que no han d'estar a la dieta.

Una vegada fet el mock up a la plataforma Figma es comença amb la preparació del FrontEnd al framework Ionic React, al final s'opta per crear una pantalla anterior a la del Log in que permeti escollir si es vol fer Log in o Sign up. Es manté una estètica simple que sigui fàcil d'entendre, no massa intrusiva i que es mantingui per tot el projecte. Es crea un estil per a mode fosc i un per a mode clar de pantalla, perquè l'usuari esculli el que prefereixi.

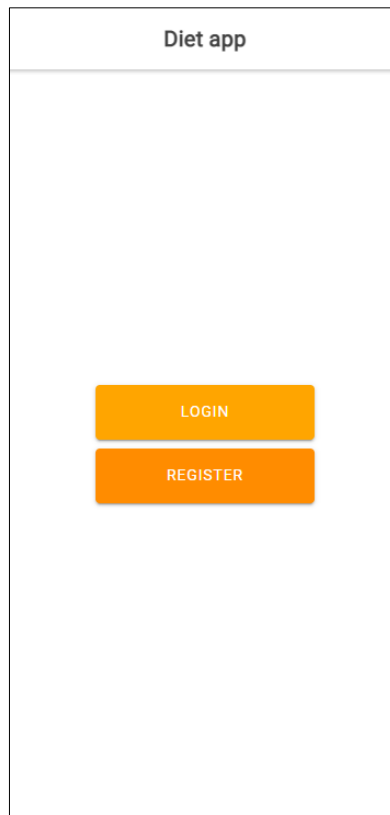
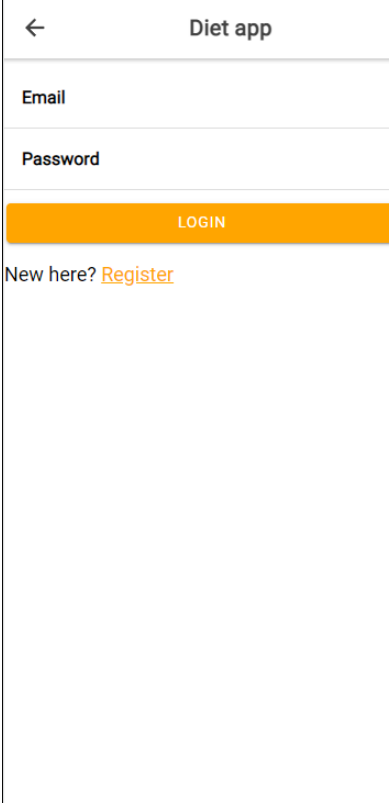


Fig. 7.1.7. Pantalla home de l'aplicació

Per a la pantalla de Log in del projecte es decideix mantenir els camps bàsics presentats anteriorment. Addicionalment, s'afegeix una drecera per poder anar directament a la pantalla de Sign up. També es pot apreciar que la temàtica de colors es manté al llarg de tot el projecte.



← Diet app

Email

Password

LOGIN

New here? [Register](#)

Fig. 7.1.8. Pantalla Log in de l'aplicació

Per la pantalla de Sign up es mantenen tots els camps presentats amb anterioritat, però s'organitzen en una sola pantalla que es pot desplaçar amunt i avall. Addicionalment, també s'afegeix una drecera per poder anar directament a la pantalla de Log in.

S'afegeixen validacions al FrontEnd sobre els camps que s'han d'omplir en el formulari per no haver de fer crides innecessàries al BackEnd.

Diet app	
First name	
Last name	
Email	
Age	
Password	
Confirm password	
Height	
Weight	
Level of activity	▼
Genre	▼
Birth body genre	▼
Desired weight	

Diet app	
Email	
Age	
Password	
Confirm password	
Height	
Weight	
Level of activity	▼
Genre	▼
Birth body genre	▼
Desired weight	
REGISTER	
Already have an account? Login	

Fig. 7.1.9 Pantalla de Sign in de l'aplicació

Els errors que proporciona Firebase Authentication també es mostren en pantalla però clarificant la informació del missatge per l'usuari.

Tota mena de validació errònia es mostra amb un toast o notificació flotant de color gris per avisar a l'usuari de l'error. Quan l'error provocat ve per part de Firebase es mostra de color vermell, aquesta opció només pot aparèixer quan el compte de correu està en ús, la contrasenya no compleix les condicions de seguretat o la connexió amb Firebase ha donat un error.

Finalment, quan el compte s'ha creat correctament es mostra una notificació de color verd i l'usuari és redirigit a la pantalla home de l'aplicació.

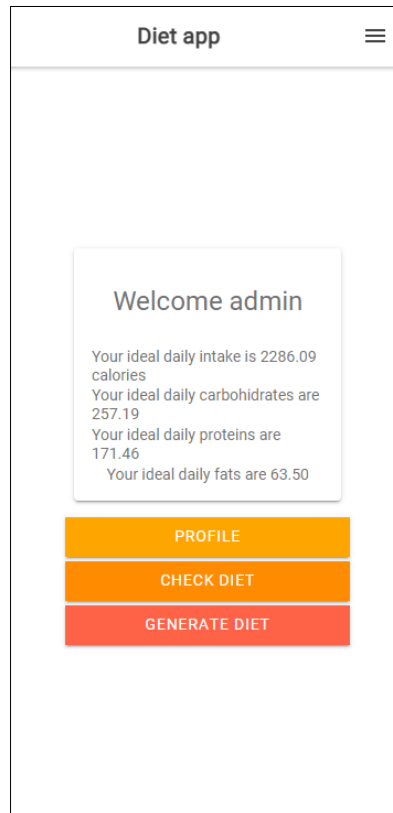


Fig. 7.1.10. Pantalla Dashboard de l'aplicació

Adicionalment, als tres botons principals que permeten navegar per l'aplicació s'afegeix un missatge de benvinguda personalitzat amb el nom de cada usuari i les seves calories, carbohidrats, proteïnes i grasses diàries a consumir.

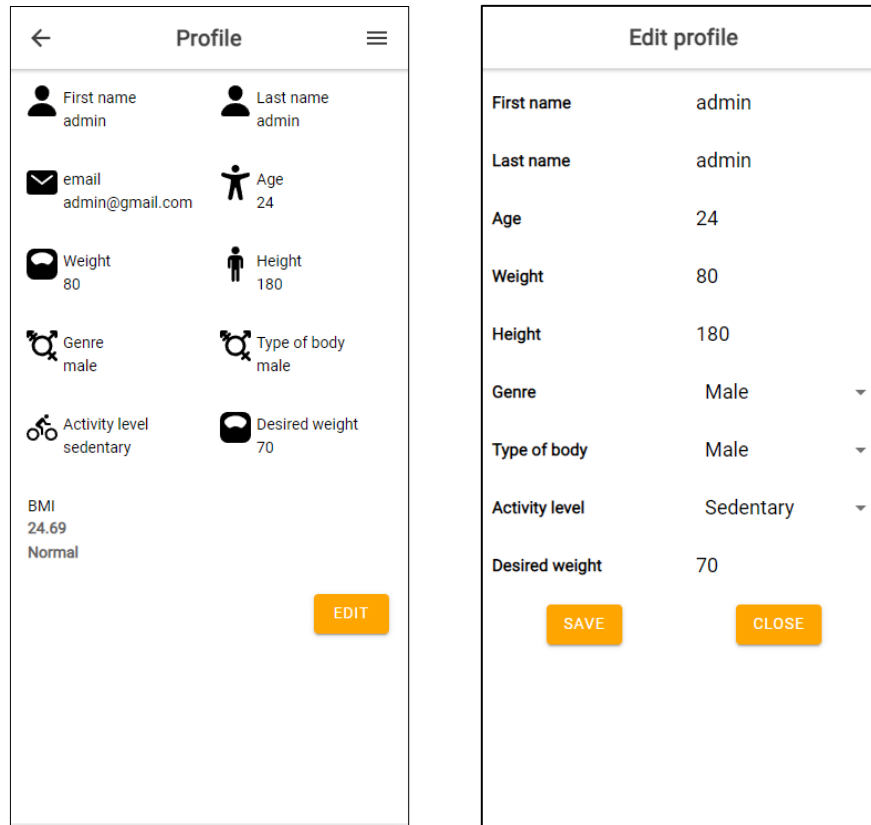


Fig. 7.1.11. Pantalla Perfil i edició de Perfil de l'aplicació

En aquesta pantalla es mostren tots els camps que l'usuari ha introduït en el formulari de registre i addicionalment el camp BMI/IMC o índex de massa corporal, autogenerat per l'aplicació. S'afegeix una icona al costat de cada camp que representant el seu significat. S'opta per eliminar la fotografia del perfil de l'usuari per limitacions de la BBDD. Per altra banda, s'agrega un botó d'edició per poder modificar qualsevol dels camps mostrats tret del BMI/IMC.

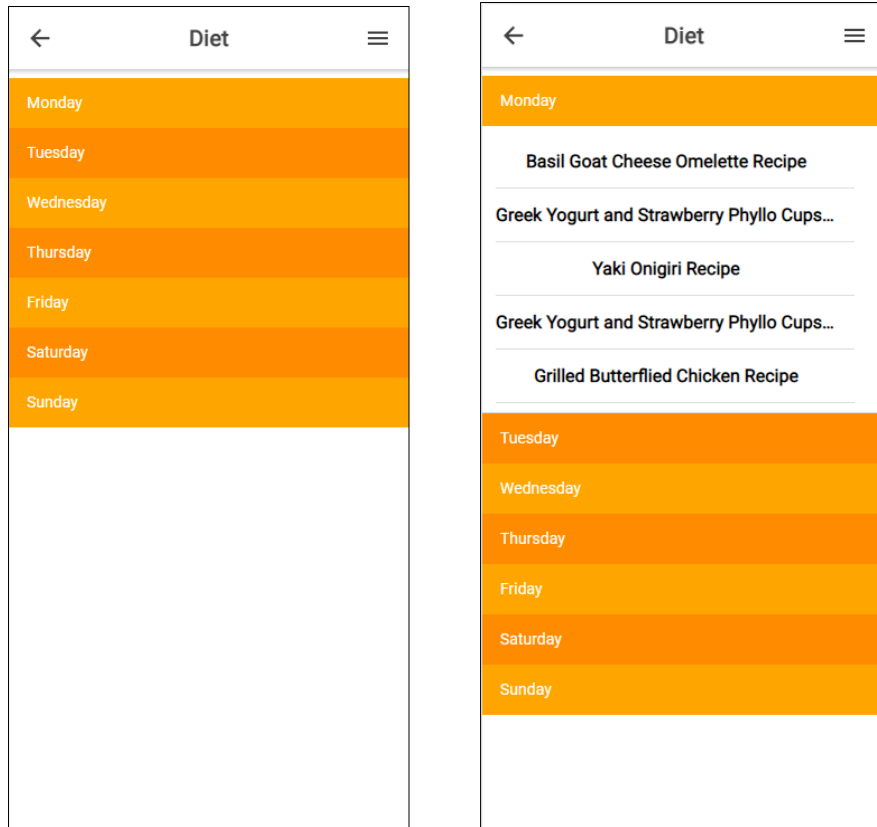


Fig. 7.1.12. Pantalla de visualització de Dietes de l'aplicació

Com es presenta anteriorment la pantalla de visualització de dietes mostra una llista de botons, un per cada dia de la setmana, que en clicar desplega la dieta diària per al dia seleccionat.

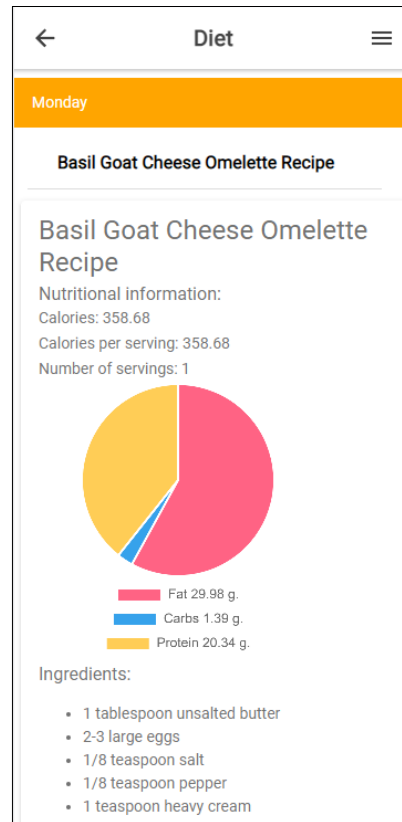


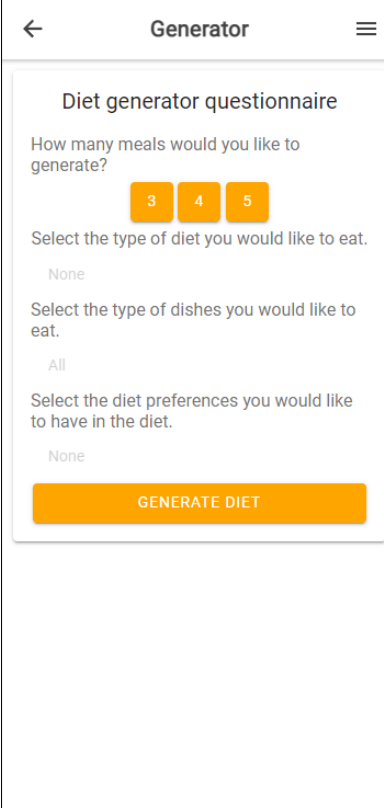
Fig. 7.1.13. Pantalla de visualització de receptes de l'aplicació

Una vegada es clica a la recepta que es vol visualitzar aquesta torna a desplegar un espai on es mostra la informació nutricional bàsica del plat. Es mostra el nom en gran a dalt, seguit de les calories totals del plat, les calories que proporciona cada ració ingerida i el número de porcions que conté la recepta.

Sota aquesta informació es renderitza un gràfic circular que mostra les macros del plat, representades pels colors:

- Vermell: Greixos
- Blau: Carbohidrats
- Groc: Proteïna

A baix de tot de l'espai es mostra els ingredients necessaris per a dur a terme la recepta.



The screenshot shows a mobile application interface titled "Generator". At the top, there is a back arrow on the left and a hamburger menu icon on the right. The main content area is titled "Diet generator questionnaire" and contains three questions:

- Question 1: "How many meals would you like to generate?" with three orange buttons labeled "3", "4", and "5".
- Question 2: "Select the type of diet you would like to eat." with a "None" option below it.
- Question 3: "Select the type of dishes you would like to eat." with an "All" option below it.

Below the questions is a fourth question: "Select the diet preferences you would like to have in the diet." with a "None" option below it. At the bottom of the questionnaire is a large orange button labeled "GENERATE DIET".

Fig. 7.1.14. Pantalla de qüestionari de generació de dietes

Aquesta pantalla ha sofert canvis en els textos que es presenten anteriorment, però els actuals tenen la mateixa funcionalitat. El qüestionari presenta les preguntes necessàries per generar una dieta setmanal:

- La primera pregunta clarifica el número de menjars per dia.
- La segona pregunta clarifica el tipus de dieta que es vol seguir, ja sigui, per exemple, alta en proteïnes, equilibrada, alta en fibra, etc. Es poden seleccionar més d'un tipus de dieta, encara que això pot donar a incompatibilitats a l'hora de generar la dieta.
- La tercera pregunta clarifica el tipus de plats que es mostraran a la dieta, hi ha l'opció de tenir-los tots barrejats o de seleccionar un d'específic. Per exemple: Mediterrània, xinesa, japonesa, italiana, etc.
- L'última pregunta clarifica el tipus de preferències que es volen a la dieta, en altres paraules són especificacions dels plats, per exemple: Sense gluten, sense peix, vegetarià, vega, etc.

7.2. BackEnd

Per a la realització del BackEnd s'ha optat per crear una API en la qual el FrontEnd ha de fer les crides necessàries per obtenir la informació de retorn de les dietes. Per a l'organització de l'API s'ha optat per una arquitectura per capes en la que la responsabilitat més gran recau en els controladors que fan la majoria dels càlculs, amb ells les crides a l'API Edamam. Posteriorment, aquesta informació és retornada als Rest Services que envien la informació al FrontEnd.

7.2.1. Deserialització dels objectes d'API externs

Per a obtenir la informació i emmagatzemar-la en memòria de forma pràctica i eficient s'opta per la deserialització i implantació a objectes. Per a aconseguir-ho s'han creat 7 classes per acomodar els camps que són necessaris per a la creació de dietes, els altres camps restants són descartats. Les classes seleccionades per ordre jeràrquic són:

- **RecipeList.** És un objecte que només conté una llista que engloba a altres objectes, ja que el sistema jeràrquic de l'emmagatzematge de dades ens ve guiat pel retorn de l'API.
- **RecipeWrap.** Són els objectes que conformen la llista de l'objecte superior, només conté un objecte interior anomenat **Recipe**.
- **Recipe.** És l'objecte base de tota la jerarquia, la que conté la majoria d'informació i amb la que es fan la majoria d'operacions en el controlador. Conté la informació bàsica de les calories de la recepta, les calories d'una ració, el número de racions que proporciona, un link a una imatge. Addicionalment, conté llistes amb altres objectes inferiors que guarden informació nutricional de la recepta, com nutrients, etiquetes amb informació per al·lèrgics.

Les classes restants serveixen simplement per acabar d'acomodar la informació de forma organitzada respectant la rebuda de l'API.

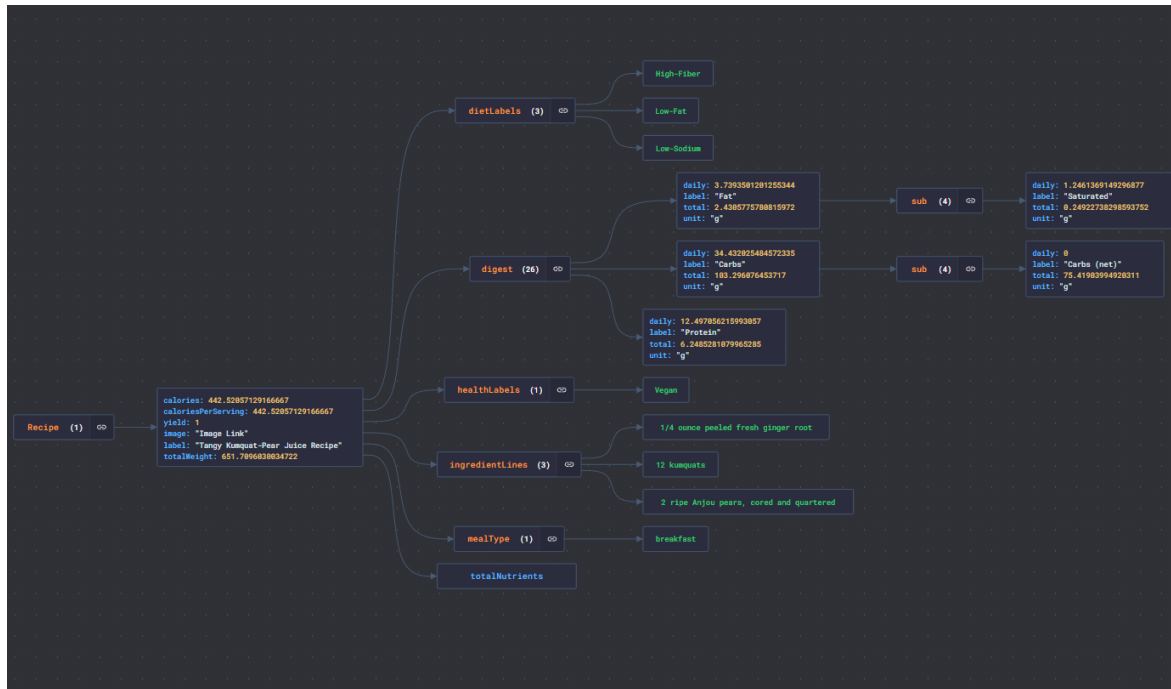


Fig. 7.2.1.1. Esquema de classes de deserialització d'API [16]

7.2.2. Processament de creació de dieta

La crida de les receptes a l'api es fa en el cas específic que l'usuari ompli el formulari de generació de dieta des del FrontEnd. El fet d'omplir el formulari iniciar una crida a la Rest API del BackEnd de tipus POST, on l'objecte que s'envia és "Diet Request", un objecte de tipus DTO que emmagatzema la informació necessària per a la creació d'una dieta setmanal personalitzada, contenint els camps:

- Número de menjars diaris: El número d'àpats que es vol ingerir cada dia, aquest número només pot variar entre tres i cinc.
- Tipus de dieta: El tipus de plats que apareixen a la dieta que es genera, entren en aquest grup: alts en proteïnes, alts en fibra, equilibrats, etc. La informació s'emmagatzema en un array i no hi ha límit tret de les opcions permeses.
- Tipus de plat: L'origen o al tipus de la dieta dels plats que s'obtenen per fer la dieta. Només es pot seleccionar un tipus.

- Preferències: Especificacions que es volen o no a la dieta, per exemple: sense gluten, vegà, vegetarià, sense peix, etc. Aquest camp, també, consisteix en un array que permet més d'una preferència escollida per l'usuari.

El procés farà un número de crides a l'API introduint els filtres mencionats anteriorment. El número de crides es repetirà les mateixes vegades que hagi escullit l'usuari, limitat entre tres i cinc. Al rebre la informació de les crides a l'api i emmagatzemar-la correctament en memòria es comença a omplir una llista que serà el retorn definitiu cap al FrontEnd. Aquesta llista disposa dels set dies de la setmana i a l'interior de cada dia hi ha una segona llista on es concentren els àpats que es retornaran. Es poden omplir cinc àpats amb només quatre crides d'API com que es pren com a similar l'esmorzar i l'àpat de mig matí, requerint així simplement una crida que retorni el mateix tipus de receptes.

7.2.3. Connexió amb la base de dades

En una primera instància s'utilitza Azure per tenir el hosting de la BBDD, després de veure les limitacions que proporciona la plataforma respecte a l'emmagatzematge d'objectes i les relacions que presenten entre ells s'escull l'opció de canviar el host de la BBDD a Firebase Firestore, essent una base de dades NoSQL amb una capacitat molt superior a la d'Azure. Addicionalment, a aquesta característica de la BBDD, Firebase també presenta un aplicatiu per controlar i emmagatzemar comptes anomenat Firestore Authentication. S'ha canviat de servei pel fet de poder relacionar aquests productes primer creant un usuari de Auth i emmagatzemant-lo dins de Firebase Authentication, per posteriorment crear aquell mateix usuari a la BBDD de Firebase Firestore i tenir els dos relacionats per la mateixa Id única.

7.2.4. Processament de dades d'usuari

Quan l'usuari omple el formulari de registre de l'aplicació s'envia una petició al servei de Firebase Authentication, que verifica que l'email no estigui en ús per un altre compte i que la contrasenya sigui adequada. Si es compleixen totes les condicions un nou registre és creat a la base de dades d'usuaris d'Authentication i retorna un id únic que identifica a l'usuari.

Una vegada en possessió d'aquest id es comença la creació de l'usuari a la BBDD de Firestore Firebase [14]. Abans de poder emmagatzemar-les es fan uns càlculs ràpids i necessaris que l'usuari no ha proporcionat, però que s'emmagatzemaran a la base de dades de totes formes,

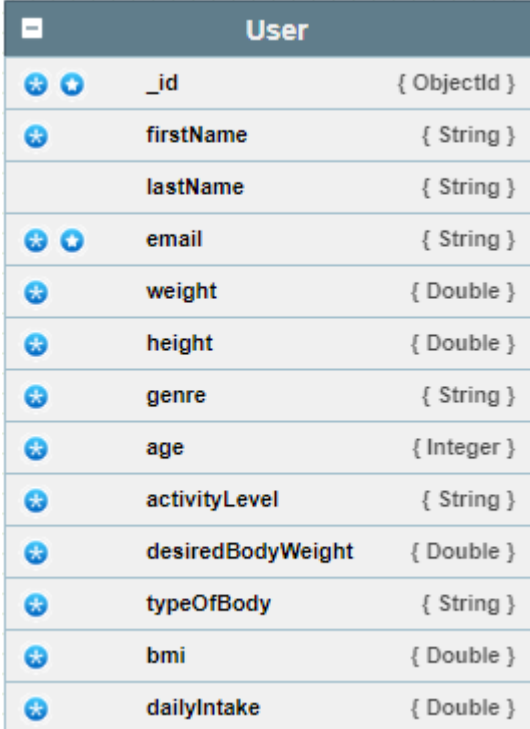
aquests camps autocalculats són l'índex de massa corporal i la ingesta diària de calories, les dues calculades a partir de les dades proporcionades per l'usuari prèviament. El consum de calories diàries es calcula utilitzant la fórmula de Harris-Benedict (2.1), (2.2) observant si és de sexe masculí o femení. Posteriorment a aquests càlculs es multiplica per un nombre fix dependent de l'activitat diària que fa l'usuari.

Activitat	Multiplicador
Sedentari	x1.2
Lleugerament actiu	x1.375
Moderadament actiu	x1.55
Molt actiu	x1.725
Extremadament actiu	x1.9

Taula 7.2.4.1 Activitat diària en relació calòrica

Posteriorment a aquests càlculs s'afegeixen o es resten tres-centes calories del total dependent de si la persona vol perdre o guanyar pes. Se sap que vol fer la persona observant el valor que posa el camp del pes desitjat en comparació amb el pes actual.

Finalitzats els càlculs es guarden les dades essencials per a fer funcionar el BackEnd i poder fer les crides per part del FrontEnd. La taula allotjada a Firebase és la següent:



User		
*	* _id	{ Objectid }
*	firstName	{ String }
	lastName	{ String }
*	* email	{ String }
*	weight	{ Double }
*	height	{ Double }
*	genre	{ String }
*	age	{ Integer }
*	activityLevel	{ String }
*	desiredBodyWeight	{ Double }
*	typeOfBody	{ String }
*	bmi	{ Double }
*	dailyIntake	{ Double }

Fig. 7.2.4.1 Taula de BBDD de User [13]

Es pot apreciar que la majoria dels camps són obligatoris a excepció del "Last name" com que no és informació completament necessària per fer les funcions mencionades anteriorment. S'escull de clau primària un camp id que s'autogenera quan es crea un usuari nou i es proporciona com a segona clau primària el camp "email" com que ha sigut marcat com a únic.

Quan es vol recuperar un usuari s'agafa la referència que ens retorna el login de l'usuari a Firebase Authentication, que és el mateix id que l'allotjat a la BBDD, al fer la query amb aquesta informació es retornen les dades al FrontEnd per poder-les mostrar a l'usuari.

7.2.5. Endpoints

L'API Rest del BackEnd està compresa en dues funcionalitats separades. Es poden apreciar dos API Rest, una per la gestió d'usuaris i una altra per la de les dietes.

Per a la gestió d'usuaris s'ha creat una API Rest basada en les tècniques CRUD (Create, Read, Update, Delete) de base de dades per poder tenir un control acurat dels usuaris. Seguint aquest concepte de gestió d'usuaris l'API Rest ens genera quatre endpoints:

- `/create`. Petició POST que inicialitza el flux de creació d'usuari a partir d'un objecte `user` proporcionat pel FronEnd. Addicionalment, a tots els camps proporcionats per l'objecte es generen dos camps més en el controlador del BackEnd, essent aquests el BMI i les calories diàries de l'usuari.
- `/get`. Petició GET que proporcionada una Id d'usuari retorna un objecte `user` complet.
- `/update`. Petició PATCH que proporcionat un objecte `user`, actualitza l'usuari actual, agafant el mateix Id de dins de l'objecte i recalculant els camps BMI i calories diàries, per si hi ha hagut canvis en els camps dels quals depenen.
- `/delete`. Petició DELETE que proporcionada una Id esborra de la BBDD un usuari i les seves receptes.

Els endpoints de l'API Rest de dietes són només dos:

- `/generate`. Petició POST que inicia el flux de creació de dieta, accepta un objecte "Diet Request" i un Id d'usuari i a partir de la informació continguda en aquest crea una dieta que posteriorment és emmagatzemada a la BBDD. Les dades són emmagatzemades com a variable de l'usuari del qual s'ha proporcionat la Id.
- `/get`. Petició GET que inicia el flux de recollida de dades de dieta, accepta un Id d'usuari del qual es recollida la dieta. Retorna un JSON amb totes les dades de la dieta continguda a la BBDD.

S'estableix una configuració a l'API per limitar el nombre d'endpoints per on es pot cridar i el tipus de crides que es poden arribar a fer. La configuració aplica als Rest Services de les entitats *user* i *diet*.

Es configura el Rest Service de *user* a la direcció interna de `"/user/**"` i es permeten les crides de tipus GET, POST, DELETE i PATCH. Per al Rest Service, el de *diet*, es configura a la direcció interna `"/diet/**"` i es limiten les crides als tipus GET i POST.

Aquesta configuració de la API permet establir una estructura clara i segura per gestionar les crides als endpoints. Amb aquesta limitació de tipus de crides i la definició de rutes específiques per a cada Rest Service, es controla i restringeix l'accés a les operacions i dades de l'API segons les necessitats de la crida.

8. Anàlisi de resultats

En aquest capítol es presenta una descripció exhaustiva de tot el que s'aconsegueix durant el desenvolupament del projecte i es relaciona amb punts presentats amb anterioritat per respaldar les afirmacions.

S'assoleix l'objectiu de crear una interfície d'usuari usable i senzilla, tant d'utilitzar com d'entendre, tal com s'ha identificat a través de les entrevistes fetes a diferents possibles usuaris de diferents edats i gèneres. Es té en compte les necessitats i les preferències dels usuaris per proporcionar una experiència satisfactòria i fàcil d'usar en la interacció amb l'aplicatiu i s'ha anat millorant en les diferents iteracions que es duen a terme.

No obstant això, s'ha hagut de dedicar més temps del planificat a assolir la tasca a causa de la inexperiència i manca de coneixement amb el framework utilitzat. Aquesta carència ha suposat un aprenentatge continu i una comprensió progressiva del framework, la seva estructura i les seves funcionalitats. Això ha alentit el procés de desenvolupament i ha requerit un esforç addicional per superar els obstacles associats.

S'enfronten dificultats específiques a l'hora de plasmar el prototip a l'aplicatiu real, a causa de les limitacions presents a l'API usada pel projecte. Hi ha problemes amb l'addició de whitelists a la creació de dietes a causa de com l'API Edamam demana ser cridada. Per poder fer la crida l'usuari hauria d'escriure manualment les paraules de la whitelist i posteriorment a fer iteracions i analitzar les reaccions de possibles usuaris es descarta per no tenir massa claredat implícita dels aliments que es poden escriure i els que no. Addicionalment, a aquest punt, no es té amb seguretat que l'usuari escrigui bé la paraula o l'aliment, fet que fa retornar un resultat buit de l'API.

A més, la funció de filtrar que proporciona l'API amb els paràmetres tipus de dieta i paràmetres, proporcionats en el FrontEnd, junt amb el filtre col·locat automàticament pel BackEnd de limitació de calories per poder formar la dieta correctament. A vegades no retorna suficients receptes per poder formar una dieta completa durant tota la setmana. Aquest problema es relaciona amb el fet de filtrar excessivament els resultats limitant molt les opcions de retorn, addicionalment del requisit de set receptes per poder formar una dieta setmanal correctament.

A més d'això, es destaca que no és possible implementar la funcionalitat de canviar una recepta a causa de les restriccions imposades per l'API Edamam. Aquesta limitació suposa una dificultat addicional en el desenvolupament del projecte.

En resum, durant el desenvolupament del projecte s'enfronten desviacions en l'experiència del framework utilitzat, limitacions de l'API en l'addició de whitelists, problemes en el filtratge de receptes i la restricció per canviar una recepta. Tot i això, es prenen mesures per superar aquestes dificultats i es posa l'èmfasi en la creació d'una interfície d'usuari usable i en l'adaptació de l'aplicació a les necessitats dels usuaris, aportant una experiència satisfactòria en l'ús de l'aplicació.

9. Conclusions

Les conclusions que s'extreuen de l'anàlisi de la feina feta en aquest projecte es concreten posteriorment en els següents punts.

En primer lloc, cal destacar la selecció adequada d'objectius finals pel projecte, com que s'aconsegueix crear un aplicatiu únic que proporciona funcionalitats de diferents aplicacions, però que és el primer a tenir-les totes juntes en un sol programa. Un dels aspectes destacats és el valor de la multiplataforma. A l'estar programat en un entorn híbrid es pot exportar i visualitzar en diferents medis, omplint un buit existent en els antecedents estudiats. Aquesta capacitat de ser multiplataforma representa una aportació professional significativa i permet arribar a un ampli públic d'usuaris amb diferents dispositius. Això pot contribuir a l'èxit i la utilitat de l'aplicació en conjunt.

D'altra banda, es destaca la selecció inadequada d'API com a un dels principals aprenentatges, a causa de la seva falta d'opcions a l'hora d'escollir receptes amb els filtres activats i especificats. Addicionalment, es descobreix que l'API té limitacions significatives. S'observa que està limitada a un nombre màxim de deu crides per minut, fet que no permet l'escalabilitat a llarg termini, i que no era gratuïta per a un ús amb un volum major de crides. Això suposa un repte per al projecte, ja que el pressupost assignat no permetia contractar els serveis de l'API per satisfer les necessitats dels usuaris. Aquesta restricció requereix ajustar l'enfocament i buscar alternatives per aconseguir els resultats de manera més eficient. Acció que s'ha obtingut optimitzant les accions de cerca aplicades al BackEnd, limitant les crides a les receptes que s'ajustin a les calories personalitzades per a cada usuari, en lloc de fer crides a l'engròs. Els problemes experimentats a causa d'aquesta decisió han posat de manifest la importància de realitzar una investigació exhaustiva i una valoració adequada abans de prendre decisions rellevants en el desenvolupament d'un projecte.

Addicionalment, la planificació inicial es veu allargada per la manca d'experiència sobre el framework Ionic React. La inexperiència en l'ús del framework genera dificultats i una corba d'aprenentatge més pronunciada. És necessari invertir temps addicional en l'estudi i l'assimilació dels conceptes, funcionalitats i característiques que proporciona el framework. Aquesta etapa afegida a la planificació inicial requereix explorar la documentació oficial, fer cursos específics i buscar solucions a problemes que sorgeixen durant el desenvolupament.

Aquesta experiència posa de manifest la importància de tenir un coneixement adequat de les eines i els frameworks utilitzats en un projecte abans de començar la seva realització, a causa dels possibles endarreriments que pot crear. Per tant, és crucial reconèixer com la manca de coneixement suficient va afectar la planificació inicial i va comportar un increment en el temps dedicat al desenvolupament del FrontEnd.

10. Possibles ampliacions

Les possibles ampliacions que es presenten incumbeixen tant objectius que no s'han pogut assolir com objectius que no han sigut proposats com ampliacions que requereixen pressupost o temps de producció més elevat. Les possibles ampliacions que es presenten són:

- Afegir una funció de connectivitat amb el calendari, per poder tenir un control de la dieta dia a dia i poder marcar, quines receptes s'han ingerit i quin dia. Aquesta funcionalitat també afegiria l'opció de veure quantes calories s'han ingerit en un dia i quantes queden per ingerir, el mateix cas aplica amb les macros.
- Millorar la presentació de l'aplicació, a causa de la inexperiència amb el framework Ionic React no es pot profunditzar en una interfície d'usuari com es vol. Aquesta ampliació pot incloure l'ús de dissenys més atractius, l'afegiment d'elements gràfics complementaris, la millora de la font i la disposició dels elements a les diferents pantalles.
- Implementar una base de dades pròpia per allotjar les receptes en lloc de dependre d'API de tercers. Aquesta acció proporcionaria més control i flexibilitat sobre les receptes, permetent així afegir, modificar i eliminar receptes d'usuaris.
- Possibilitat de tenir diverses dietes en un mateix usuari. Actualment, cada usuari només té una dieta assignada en cada moment i si es crea una de nova s'elimina l'anterior. No obstant això, es podria considerar ampliar la funcionalitat per permetre als usuaris tenir múltiples dietes. Això és útil pels usuaris que tenen objectius diferents en diferents etapes o que volen explorar diverses opcions de dieta.
- Hosting tant del FrontEnd com del BackEnd. Per garantir un funcionament òptim i un accés constant a l'aplicació, seria recomanable considerar el host tant del BackEnd com del FrontEnd en un servidor fiable i escalable. Això asseguraria que l'aplicació estigui disponible en línia en tot moment i es pugui accedir-hi des de diferents dispositius i ubicacions.

En resum, aquestes possibles ampliacions del projecte poden millorar tant la funcionalitat com l'experiència global d'usuari. L'addició d'una funció de calendari, la millora de la

presentació, la creació d'una base de dades pròpia, la possibilitat de tenir diverses dietes i l'allotjament en un servidor fiable són passos que poden contribuir a l'èxit i l'evolució continuada del projecte.

11. Bibliografia

- [1] Ó. J. Socas González, "Aplicación móvil para la generación y preparación de menús saludables personalizados", treball de grau, Universidad de La Laguna, San Cristóbal, Tenerife, 2019. Accedit el 30 de novembre de 2022. [En línia]. Disponible: <https://riull.ull.es/xmlui/bitstream/handle/915/15451/Aplicacion%20movil%20para%20la%20generacion%20y%20preparacion%20de%20menus%20saludables%20personalizados.pdf?sequence=1&isAllowed=y>
- [2] D. Suárez Cáceres, "Nutrain: Aplicación web de nutrición, entrenamiento y otros aspectos de la salud", treball de grau, Universidad de Las Palmas de Gran Canaria, Las Palmas. Accedit el 30 de novembre de 2022. [En línia]. Disponible: https://accedacris.ulpgc.es/bitstream/10553/77855/2/0771835_00000_0000.pdf
- [3] J. Alaguero Martín, "Dielthy: Una web para dietes especiales", treball de grau, Universidad de Valladolid, Valladolid. Accedit el 30 de novembre de 2022. [En línia]. Disponible: <https://uvadoc.uva.es/handle/10324/43958>
- [4] "API Edamam". <https://www.edamam.com>.
- [5] "Piràmide de l'alimentació saludable". Canal Salut. <https://canalsalut.gencat.cat/ca/vida-saludable/alimentacio/piramide-alimentacio-saludable/> (Accedit el 12 de gener de 2023).
- [6] M. R. C, Clean code: A Handbook of Agile Software Craftsmanship. Upper Saddle River, NJ: Prentice Hall, 2008.
- [7] "Introduction to Ionic | Ionic Documentation". Ionic Framework - The Cross-Platform App Development Leader. <https://ionicframework.com/docs/> (Accedit el 6 de febrer de 2023).
- [8] Microsoft. "Visual Studio Code - Code Editing. Redefined". Visual Studio Code - Code Editing. Redefined. <https://code.visualstudio.com> (Accedit el 26 de novembre de 2022).
- [9] "¿Qué es Xamarin? - Xamarin". Microsoft Learn: Build skills that open doors in your career. <https://learn.microsoft.com/es-es/xamarin/get-started/what-is-xamarin> (Accedit el 15 de desembre de 2022).

- [10] "Comprar IntelliJ IDEA Ultimate: Precios y licencias, Descuentos - Suscripción a JetBrains Toolbox". JetBrains. <https://www.jetbrains.com/es-es/idea/buy/?section=commercial&billing=yearly> (Accedit el 19 de gener de 2023).
- [11] "Spring | Home". Home. <https://spring.io> (Accedit el 30 de diciembre de 2022).
- [12] España, Jefatura del Estado, "Ley Orgánica n.º 3/2018, Ley Orgánica 3/2018, de 5 de diciembre, de Protección de Datos Personales y garantía de los derechos digitales", Boletín Oficial del Estado, n.º 294, 6 de diciembre de 2018. Accedit el 2 de febrer de 2023. [En línia]. Disponible: <https://www.boe.es/buscar/act.php?id=BOE-A-2018-16673>
- [13] "Free Database Modeling Tool Online | NoSQL DB Modeler, Entity Relationship Diagram". NoSQL DB Schema Modeling. <https://nosqldbm.ru> (accedit el 6 d'abril de 2023).
- [14] "Firebase". Firebase. <https://firebase.google.com/?hl=es> (accedit el 6 de març de 2023).
- [15] "Figma: the collaborative interface design tool." Figma. <https://www.figma.com> (accedit el 29 demarç de 2023).
- [16] "JSON Crack - Crack your data into pieces". <https://jsoncrack.com> (accedit el 14 de març de 2023).