

UNIVERSIDADE VEIGA DE ALMEIDA

Alline Marjorie Gueiros Camera Coelho

**APLICATIVO DE MONTAGEM DE PLANOS
ALIMENTARES FLEXÍVEIS: UMA ABORDAGEM
EM SISTEMAS DE INFORMAÇÃO PARA PLANOS
ALIMENTARES DE MAIOR ADESÃO**

Rio de Janeiro, RJ

2025

Alline Marjorie Gueiros Camera Coelho

**APLICATIVO DE MONTAGEM DE PLANOS
ALIMENTARES FLEXÍVEIS: UMA ABORDAGEM
EM SISTEMAS DE INFORMAÇÃO PARA PLANOS
ALIMENTARES DE MAIOR ADESÃO**

Trabalho de Conclusão de Curso apresentado
à Universidade Veiga de Almeida como
pré-requisito à obtenção do título de
Bacharel em Sistemas de Informação sob
orientação do Prof. Claudio Fico Fonseca.

Rio de Janeiro, RJ

2025

À Matheus, por ser o meu lar em meio a um mundo cheio de horrores.

RESUMO

Em redes sociais e na internet, prevalece uma enxurrada de informações - e frequentemente, desinformação - sobre alimentação, e especificamente sobre como comer bem visando perda de peso. É frequente a apresentação de alimentos específicos como grandes vilões ou poderosos aliados.

Entretanto, a realidade é simples: um plano alimentar com déficit calórico - onde o indivíduo gasta mais calorias do que consome - conduz à perda de peso. As dificuldades emergem dos desafios de planejar a própria alimentação de forma realista, seguir tal plano, e registrar os consumos de alimentos para acompanhamento e ajustes. Estes fatores fazem com que o maior problema de qualquer dieta seja simplesmente seguir a dieta.

Este projeto visa amenizar tais dificuldades através de um sistema de gestão de planos alimentares, que eventualmente contará com um aplicativo móvel. Como diferencial, o sistema compõe uma vasta base de dados alimentícia, e faz uso desta para sugerir opções de refeições saudáveis, dentro dos objetivos do usuário e respeitando as preferências alimentícias deste. Com isso, é obtido um plano alimentar customizado, realista, e de fácil acompanhamento, e consequentemente mais fácil de seguir.

Palavras-chave: Alimentação, Dieta, Sistema de informação, Aplicativo Móvel, Análise de Dados.

ABSTRACT

On social media, it is easy to be overwhelmed by the amount of information - and often, misinformation - regarding nutrition, dieting, and weight loss. Many types of food are often painted as villains or heroes in regards to weight loss.

However, the reality is simple: a diet with a caloric deficit - that is, one at which a person spends more calories than they ingest - inevitably leads to weight loss. The difficulties emerge from the challenges of planning a realistic diet, following such a diet, and adjusting it as you face unexpected circumstances. Due to this, the greatest challenge on any diet is often simply sticking to it.

This project aims to alleviate such difficulties by building a nutritional plan management system, which will eventually power a mobile application. As its main feature, the system makes use of a vast database of foods and their nutritional data in order to suggest meal options that are healthy, respectful of an user's culinary preferences, and that contribute towards their dieting goals. Through this, a custom-made nutritional plan can be formulated, which is more realistic and easier to follow, therefore leading to increased dietary success.

Keywords: Nutrition, Dieting, Information Systems, Mobile applications, Data Analysis.

SUMÁRIO

1 INTRODUÇÃO	3
2 DEFINIÇÃO DO PROJETO	5
2.1 Tema	5
2.2 Justificativa	5
2.3 Problema de Pesquisa	5
2.4 Hipóteses	5
3 OBJETIVOS	6
3.1 Objetivo Geral	6
3.2 Objetivos Específicos	6
4 EMBASAMENTO TEÓRICO	7
5 PROJETO DE BANCO DE DADOS	8
5.1 Projeto Lógico de Banco de Dados	8
5.2 Projeto Físico de Banco de Dados	8
6 PROJETO DE INTERFACE GRÁFICA	9
6.1 Hierarquia das telas	9
6.1.1 Hierarquia das telas do sistema	9
6.1.2 Padronização de botões, ícones e outros atalhos.	11
7 PROJETO DE IMPLANTAÇÃO	16
7.1 Procedimentos de Implantação	16
7.1.1 Carga inicial	16
7.1.2 Execução	16
7.1.3 Backup e Restauração	16
7.1.4 Recuperação de acesso	16
7.2 Ambiente do sistema	17
7.3 Demonstração de implementação	18
7.3.1 Telas de cadastro	18
7.3.2 Operação que envolve cálculo do sistema	27
7.3.3 Relatórios	29
8 CRONOGRAMA DE IMPLEMENTAÇÃO	31
9 CONSIDERAÇÕES FINAIS	32
10 REFERÊNCIAS	33

1 INTRODUÇÃO

A alimentação exerce papel central na saúde e no bem-estar do ser humano, sendo determinante na prevenção de disfunções (KANDEL, 2019). Entretanto, a prevalência de obesidade na população (25% da população mundial é obesa) sugere a grande dificuldade presente em alimentar-se bem (OKUNOGBE, 2022).

Um dos elementos que gera esta dificuldade é a presença de desinformação sobre dietas, em especial em redes sociais e mitos culturais; que frequentemente associam alimentos específicos à ganho ou perda de peso (FERNÁNDEZ, 2025). Entretanto, a realidade é que o fator determinante para a perda de peso é a adesão a uma dieta que possua déficit calórico (ingestão de calorias menor do que as calorias consumidas) (REDMAN, 2007). Para atingir este déficit, indivíduos que desejam perder peso frequentemente elaboram planos alimentares ; que, dado o consumo calórico estimado do indivíduo e outras informações, discriminam o número de refeições por dia e sua composição para que o déficit calórico seja alcançado (PLANO..., 2025).

Porém, esta rigidez é associada a uma baixa adesão à dieta (GIBSON, 2017). Sabendo que o fator relevante é somente a preservação de um déficit calórico, é possível ajustar a dieta para que ela conteplane diferentes combinações de alimentos em cada refeição, sem que a refeição exceda os limites calóricos desejados e ainda preserve um equilíbrio em sua composição. Tais dietas são conhecidas como flexíveis; e são associadas a uma chance maior de adesão (GONSALVES, 2023), pois melhor contemplam as dificuldades do dia-a-dia como custos, eventos sociais alimentícios, indisponibilidade de alimentos específicos, preferências alimentares individuais, limitações de tempo, etc.

Aplicativos móveis são frequentemente utilizados como uma ferramenta de planejamento alimentar, removendo o atrito e esforço cognitivo associado à decidir a composição de cada refeição individualmente (COUGHLIN, 2025).

Entretanto, a elaboração de um plano alimentar flexível de qualidade costuma necessitar o envolvimento de um nutricionista (AMIRI, 2023), que contempla previamente todas as combinações possíveis para cada refeição. Na ausência deste profissional, o indivíduo precisa elaborar o plano alimentar por conta própria; um processo extenso e para o

qual ele frequentemente não está habilitado e, dado os elementos de desinformação previamente citados, conduz à dietas ineficazes.

O projeto visa a elaboração de um sistema que, munido de informações, objetivos e preferências alimentícias do usuário, seja capaz de fornecer sugestões para a elaboração de cada refeição do plano, e seja também capaz de sugerir refeições quando se faz necessário divergir do planejado. Tais sugestões se farão possíveis dada a manutenção de uma base de dados de alimentos e suas informações nutricionais; que será preenchida pelos planos alimentares contribuídos pelos próprios usuários.

Espera-se que este sistema auxiliará na elaboração de planos alimentares flexíveis, de qualidade, e de alta adesão; sem necessitar a intervenção e acompanhamento de um nutricionista e facilitando a obtenção dos resultados desejados pelo usuário.

2 DEFINIÇÃO DO PROJETO

2.1 Tema

Desenvolvimento de um aplicativo para montagem de planos alimentares flexíveis: uma abordagem em Sistemas de Informação.

2.2 Justificativa

Planos alimentares flexíveis têm maior chance de adesão, e a adesão é associada positivamente à perda de peso. Dada a onipresença de dispositivos móveis, aplicativos móveis são uma ferramenta comum para elaboração e acompanhamento de planos alimentares.

2.3 Problema de Pesquisa

Como desenvolver um sistema de informação, em formato de aplicativo móvel, que possibilite a criação e adaptação de planos alimentares flexíveis personalizados para usuários de acordo com suas necessidades e sem requerer o envolvimento de terceiros?

2.4 Hipóteses

- A capacidade do aplicativo produzir sugestões em tempo real quando não é possível aderir ao plano alimentar reduzirá o dano causado por eventos de não-adesão;
- Um plano alimentar personalizado baseado em informações e preferências individuais contribui para maior eficácia de dietas;
- Quanto mais usuários utilizam o app, mais dados sobre alimentos e suas características ingressarão na base de dados, aumentando a qualidade das sugestões de refeições para todos os usuários.

3 OBJETIVOS

3.1 Objetivo Geral

Desenvolver um aplicativo para criação e gerenciamento de planos alimentares flexíveis, que facilite a personalização das dietas de acordo com os objetivos, preferências, necessidades e restrições alimentares do usuário.

3.2 Objetivos Específicos

- Analisar os requisitos funcionais e não funcionais necessários para um sistema de montagem de planos alimentares flexíveis;
- Modelar a arquitetura do sistema;
- Montar fluxos de uso;
- Implementar um banco de dados de alimentos com informações nutricionais;
- Obter dados do usuário como seu objetivo, peso, altura, perfil de atividade física, preferências e restrições alimentares por um formulário dentro do aplicativo;
- Montar um plano alimentar orientado aos dados obtidos do usuário por formulário;
- Oferecer dentro do plano alimentar substituições para alimentos respeitando os macronutrientes de cada refeição;
- Projetar uma interface intuitiva para facilitar a interação do usuário na criação e gerenciamento de plano alimentar;
- Documentar o processo de desenvolvimento.

4 EMBASAMENTO TEÓRICO

A ideia do projeto é mostrar como a tecnologia pode ser usada para melhorar o acesso e a adesão à planos alimentares, unindo conhecimentos de duas áreas - sistemas de informação e nutrição.

A nutrição possui um papel essencial na vida das pessoas, ingerir alimentos em quantidade e proporção adequados e manter bons hábitos alimentares ao longo de sua vida é fundamental para manter o estado de saúde física e mental em equilíbrio (GONSALVES, 2023).

Uma alimentação equilibrada busca atender às necessidades nutricionais de cada indivíduo, considerando aspectos fisiológicos, culturais e emocionais. A adesão a planos alimentares costuma falhar quando há restrições excessivas; por isso, a dieta flexível é uma boa solução para a perda de peso. Ser uma pessoa saudável é se alimentar bem, estar bem psicologicamente, conseguir ter um equilíbrio e, por consequência, o corpo responderá de uma maneira mais saudável e de acordo com esse estilo de vida (GONSALVES, 2023).

Segundo POLLONI (2001), um sistema de informação é “qualquer sistema que processe informações e produza resultados para um fim específico, em que cada um de seus sistemas integre um sistema automatizado de uma organização.” No contexto da saúde e nutrição, sistemas de informação permitem automatizar e personalizar o gerenciamento de dados nutricionais, facilitando o acompanhamento de metas e hábitos alimentares.

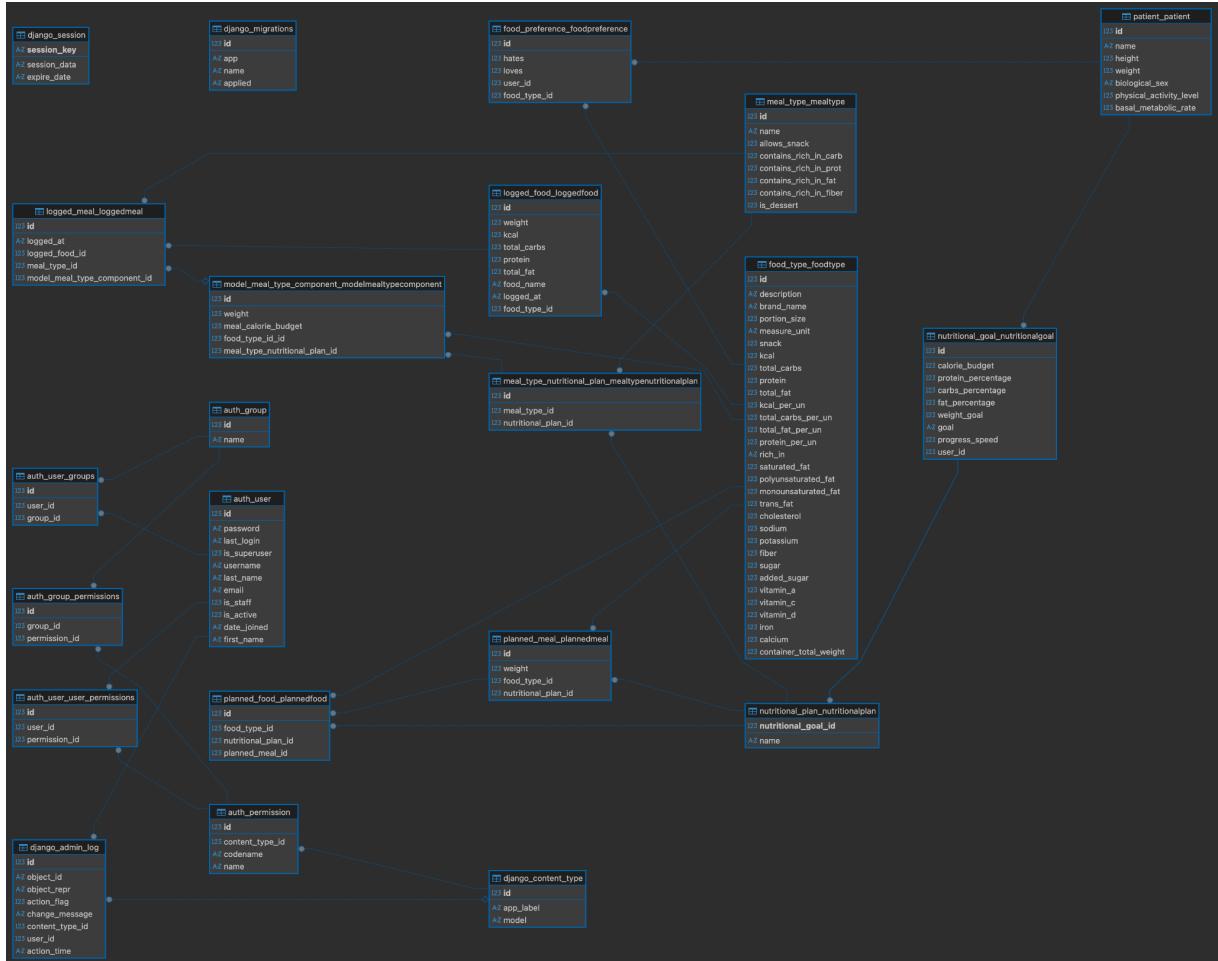
Aplicativos móveis são soluções úteis e de baixo custo para auxiliar pessoas na jornada da perda de peso (COUGHLIN, 2025).

O processo de levantamento de requisitos é essencial para o desenvolvimento de sistemas e de softwares. O levantamento de requisitos é a base para a construção de um software que atenda às necessidades dos usuários e seja capaz de agregar valor ao negócio (SOMMERVILLE, 2018).

Este trabalho se fundamenta em teorias e práticas consolidadas na nutrição, sistemas de informação e engenharia de software, buscando impactar positivamente a vida de pessoas com dificuldades em se alimentar de forma equilibrada.

5 PROJETO DE BANCO DE DADOS

5.1 Projeto Lógico de Banco de Dados



5.2 Projeto Físico de Banco de Dados

```
PRAGMA foreign_keys=OFF;
```

```
BEGIN TRANSACTION;
```

```
CREATE TABLE IF NOT EXISTS "django_migrations" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "app" varchar(255) NOT NULL, "name" varchar(255) NOT NULL, "applied" datetime NOT NULL);
```

```
CREATE TABLE IF NOT EXISTS "auth_group_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "group_id" integer NOT NULL REFERENCES "auth_group" ("id") DEFERRABLE INITIALLY DEFERRED, "permission_id" integer NOT NULL REFERENCES "auth_permission" ("id") DEFERRABLE INITIALLY DEFERRED);
```

```
CREATE TABLE IF NOT EXISTS "auth_user_groups" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ("id") DEFERRABLE INITIALLY DEFERRED, "group_id" integer NOT NULL REFERENCES "auth_group" ("id") DEFERRABLE INITIALLY DEFERRED);
```

```
CREATE TABLE IF NOT EXISTS "auth_user_user_permissions" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "user_id" integer NOT NULL REFERENCES "auth_user" ("id") DEFERRABLE INITIALLY DEFERRED, "permission_id" integer NOT NULL REFERENCES "auth_permission" ("id") DEFERRABLE INITIALLY DEFERRED);
```

```
CREATE TABLE IF NOT EXISTS "django_admin_log" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "object_id" text NULL, "object_repr" varchar(200) NOT NULL, "action_flag" smallint unsigned NOT NULL CHECK ("action_flag" >= 0), "change_message" text NOT NULL, "content_type_id" integer NULL REFERENCES "django_content_type" ("id") DEFERRABLE INITIALLY DEFERRED, "user_id" integer NOT NULL REFERENCES "auth_user" ("id") DEFERRABLE INITIALLY DEFERRED, "action_time" datetime NOT NULL);
```

```
CREATE TABLE IF NOT EXISTS "django_content_type" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "app_label" varchar(100) NOT NULL, "model" varchar(100) NOT NULL);
```

```
CREATE TABLE IF NOT EXISTS "auth_permission" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "content_type_id" integer NOT NULL REFERENCES "django_content_type" ("id") DEFERRABLE INITIALLY DEFERRED, "codename" varchar(100) NOT NULL, "name" varchar(255) NOT NULL);
```

```
CREATE TABLE IF NOT EXISTS "auth_group" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "name" varchar(150) NOT NULL UNIQUE);
```

```
CREATE TABLE IF NOT EXISTS "auth_user" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "password" varchar(128) NOT NULL, "last_login" datetime NULL, "is_superuser" bool NOT NULL, "username" varchar(150) NOT NULL UNIQUE, "last_name" varchar(150) NOT NULL, "email" varchar(254) NOT NULL, "is_staff" bool NOT NULL, "is_active" bool NOT NULL, "date_joined" datetime NOT NULL, "first_name" varchar(150) NOT NULL);
```

```
CREATE TABLE IF NOT EXISTS "patient_patient" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "name" varchar(50) NOT NULL, "height" integer NOT NULL, "weight" real NOT NULL, "biological_sex" varchar NOT NULL, "physical_activity_level" real NOT NULL, "basal_metabolic_rate" real NULL);
```

```
CREATE TABLE IF NOT EXISTS "logged_food_loggedfood" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "weight" integer NOT NULL, "kcal" real NULL, "total_carbs" real NULL, "protein" real NULL, "total_fat" real NULL, "food_name" varchar NULL, "logged_at" datetime NOT NULL, "food_type_id" bigint NOT NULL REFERENCES "food_type_foodtype" ("id") DEFERRABLE INITIALLY DEFERRED);
```

```

CREATE TABLE IF NOT EXISTS
"model_meal_type_component_modelmealtypecomponent" ("id" integer NOT NULL PRIMARY KEY
AUTOINCREMENT, "weight" real NOT NULL, "meal_calorie_budget" real NULL, "food_type_id" bigint
NOT NULL REFERENCES "food_type_foodtype" ("id") DEFERRABLE INITIALLY DEFERRED,
"meal_type_nutritional_plan_id" bigint NOT NULL REFERENCES
"meal_type_nutritional_plan_mealtypenutritionalplan" ("id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "logged_meal_loggedmeal" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "logged_at" datetime NOT NULL, "logged_food_id" bigint NOT NULL REFERENCES "logged_food_loggedfood" ("id") DEFERRABLE INITIALLY DEFERRED, "meal_type_id" bigint NOT NULL REFERENCES "meal_type_mealtype" ("id") DEFERRABLE INITIALLY DEFERRED, "model_meal_type_component_id" bigint NULL REFERENCES "model_meal_type_component_modelmealtypecomponent" ("id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "nutritional_goal_nutritionalgoal" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "calorie_budget" integer NOT NULL, "protein_percentage" real NOT NULL, "carbs_percentage" real NOT NULL, "fat_percentage" real NOT NULL, "weight_goal" real NOT NULL, "goal" varchar NOT NULL, "progress_speed" integer NOT NULL, "user_id" bigint NOT NULL REFERENCES "patient_patient" ("id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "nutritional_plan_nutritionalplan" ("name" varchar(30) NOT NULL, "nutritional_goal_id" bigint NOT NULL PRIMARY KEY REFERENCES "nutritional_goal_nutritionalgoal" ("id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "meal_type_mealtype" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "name" varchar(200) NOT NULL, "allows_snack" bool NOT NULL, "contains_rich_in_carb" bool NOT NULL, "contains_rich_in_prot" bool NOT NULL, "contains_rich_in_fat" bool NOT NULL, "contains_rich_in_fiber" bool NOT NULL, "is_dessert" bool NOT NULL);

```

```

CREATE TABLE IF NOT EXISTS "meal_type_nutritional_plan_mealtypenutritionalplan" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "meal_type_id" bigint NOT NULL REFERENCES "meal_type_mealtype" ("id") DEFERRABLE INITIALLY DEFERRED, "nutritional_plan_id" bigint NOT NULL REFERENCES "nutritional_plan_nutritionalplan" ("nutritional_goal_id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "planned_meal_plannedmeal" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "weight" integer NOT NULL, "food_type_id" bigint NOT NULL REFERENCES "food_type_foodtype" ("id") DEFERRABLE INITIALLY DEFERRED, "nutritional_plan_id" bigint NOT NULL REFERENCES "nutritional_plan_nutritionalplan" ("nutritional_goal_id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "planned_food_plannedfood" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "food_type_id" bigint NOT NULL REFERENCES "food_type_foodtype" ("id") DEFERRABLE INITIALLY DEFERRED, "nutritional_plan_id" bigint NOT NULL REFERENCES "nutritional_plan_nutritionalplan" ("nutritional_goal_id") DEFERRABLE INITIALLY DEFERRED, "planned_meal_id" bigint NOT NULL REFERENCES "planned_meal_plannedmeal" ("id") DEFERRABLE INITIALLY DEFERRED);

```

```

CREATE TABLE IF NOT EXISTS "django_session" ("session_key" varchar(40) NOT NULL PRIMARY KEY, "session_data" text NOT NULL, "expire_date" datetime NOT NULL);

```

```

CREATE TABLE IF NOT EXISTS "food_type_foodtype" ("id" integer NOT NULL PRIMARY KEY AUTOINCREMENT, "description" varchar(100) NOT NULL, "brand_name" varchar(100) NULL, "portion_size" integer NOT NULL, "measure_unit" varchar NOT NULL, "snack" bool NOT NULL, "kcal" real NOT NULL, "total_carbs" real NOT NULL, "protein" real NOT NULL, "total_fat" real NOT NULL, "kcal_per_un" real NULL, "total_carbs_per_un" real NULL,

```

```

"total_fat_per_un" real NULL, "protein_per_un" real NULL, "rich_in" text NULL CHECK
((JSON_VALID("rich_in") OR "rich_in" IS NULL)), "saturated_fat" real NULL, "polyunsaturated_fat"
real NULL, "monounsaturated_fat" real NULL, "trans_fat" real NULL, "cholesterol" real NULL,
"sodium" real NULL, "potassium" real NULL, "fiber" real NULL, "sugar" real NULL, "added_sugar"
real NULL, "vitamin_a" real NULL, "vitamin_c" real NULL, "vitamin_d" real NULL, "iron" real
NULL, "calcium" real NULL, "container_total_weight" integer NULL);

CREATE TABLE IF NOT EXISTS "food_preference_foodpreference" ("id" integer NOT
NULL PRIMARY KEY AUTOINCREMENT, "hates" bool NOT NULL, "loves" bool NOT NULL,
"user_id" bigint NOT NULL REFERENCES "patient_patient" ("id") DEFERRABLE INITIALLY
DEFERRED, "food_type_id" bigint NOT NULL REFERENCES "food_type_foodtype" ("id")
DEFERRABLE INITIALLY DEFERRED);

CREATE UNIQUE INDEX
"auth_group_permissions_group_id_permission_id_0cd325b0_uniq" ON "auth_group_permissions"
("group_id", "permission_id");

CREATE INDEX "auth_group_permissions_group_id_b120cbf9" ON
"auth_group_permissions" ("group_id");

CREATE INDEX "auth_group_permissions_permission_id_84c5c92e" ON
"auth_group_permissions" ("permission_id");

CREATE UNIQUE INDEX "auth_user_groups_user_id_group_id_94350c0c_uniq" ON
"auth_user_groups" ("user_id", "group_id");

CREATE INDEX "auth_user_groups_user_id_6a12ed8b" ON "auth_user_groups" ("user_id");

CREATE INDEX "auth_user_groups_group_id_97559544" ON "auth_user_groups"
("group_id");

CREATE UNIQUE INDEX
"auth_user_user_permissions_user_id_permission_id_14a6b632_uniq" ON
"auth_user_user_permissions" ("user_id", "permission_id");

CREATE INDEX "auth_user_user_permissions_user_id_a95ead1b" ON
"auth_user_user_permissions" ("user_id");

CREATE INDEX "auth_user_user_permissions_permission_id_1fb5f2c" ON
"auth_user_user_permissions" ("permission_id");

CREATE INDEX "django_admin_log_content_type_id_c4bce8eb" ON "django_admin_log"
("content_type_id");

CREATE INDEX "django_admin_log_user_id_c564eba6" ON "django_admin_log"
("user_id");

CREATE UNIQUE INDEX "django_content_type_app_label_model_76bd3d3b_uniq" ON
"django_content_type" ("app_label", "model");

CREATE UNIQUE INDEX "auth_permission_content_type_id_codename_01ab375a_uniq" ON
"auth_permission" ("content_type_id", "codename");

CREATE INDEX "auth_permission_content_type_id_2f476e4b" ON "auth_permission"
("content_type_id");

CREATE INDEX "logged_food_loggedfood_food_type_id_08d9ab4e" ON
"logged_food_loggedfood" ("food_type_id");

```

```

CREATE INDEX
"model_meal_type_component_modelmealtypecomponent_food_type_id_id_5ddfa546" ON
"model_meal_type_component_modelmealtypecomponent" ("food_type_id");

CREATE INDEX
"model_meal_type_component_modelmealtypecomponent_meal_type_nutritional_plan_id_bcb3a92e" ON
ON "model_meal_type_component_modelmealtypecomponent" ("meal_type_nutritional_plan_id");

CREATE INDEX "logged_meal_loggedmeal_food_id_d18ce983" ON
"logged_meal_loggedmeal" ("logged_food_id");

CREATE INDEX "logged_meal_loggedmeal_meal_type_id_6bbb9b13" ON
"logged_meal_loggedmeal" ("meal_type_id");

CREATE INDEX "logged_meal_loggedmeal_model_meal_type_component_id_460b5701" ON
ON "logged_meal_loggedmeal" ("model_meal_type_component_id");

CREATE INDEX "nutritional_goal_nutritionalgoal_user_id_8eb94140" ON
"nutritional_goal_nutritionalgoal" ("user_id");

CREATE INDEX
"meal_type_nutritional_plan_mealtypenutritionalplan_meal_type_id_ce99c48a" ON
"meal_type_nutritional_plan_mealtypenutritionalplan" ("meal_type_id");

CREATE INDEX
"meal_type_nutritional_plan_mealtypenutritionalplan_nutritional_plan_id_d0b04c47" ON
"meal_type_nutritional_plan_mealtypenutritionalplan" ("nutritional_plan_id");

CREATE INDEX "planned_meal_plannedmeal_food_type_id_94dfb6ee" ON
"planned_meal_plannedmeal" ("food_type_id");

CREATE INDEX "planned_meal_plannedmeal_nutritional_plan_id_fb9757c7" ON
"planned_meal_plannedmeal" ("nutritional_plan_id");

CREATE INDEX "planned_food_plannedfood_food_type_id_c31772e6" ON
"planned_food_plannedfood" ("food_type_id");

CREATE INDEX "planned_food_plannedfood_nutritional_plan_id_2030cb67" ON
"planned_food_plannedfood" ("nutritional_plan_id");

CREATE INDEX "planned_food_plannedfood_planned_meal_id_b9e41ad4" ON
"planned_food_plannedfood" ("planned_meal_id");

CREATE INDEX "django_session_expire_date_a5c62663" ON "django_session"
("expire_date");

CREATE INDEX "food_preference_foodpreference_user_id_7f4839d2" ON
"food_preference_foodpreference" ("user_id");

CREATE INDEX "food_preference_foodpreference_food_type_id_3aec84ba" ON
"food_preference_foodpreference" ("food_type_id");

COMMIT;

```

6 PROJETO DE INTERFACE GRÁFICA

6.1 Hierarquia das telas

6.1.1 Hierarquia das telas do sistema

- Módulo de Tipo de Alimento (Food Type)
 - Listagem de Tipos de Alimentos
 - Exibe todos os registros cadastrados
 - Permite pesquisar, ordenar e selecionar itens
 - A partir desta tela, o usuário pode acessar:
 - Adicionar novo Tipo de Alimento
 - Editar um Tipo de Alimento existente
 - Excluir um ou mais Tipos de Alimentos
 - Visualizar dados do Tipo de Alimento
 - Adicionar Tipo de Alimento
 - Tela com formulário solicitando o nome do alimento, sua marca, informações nutricionais, medidas, porção e unidade de medida
 - Também exibe campos derivados que serão calculados automaticamente após o cadastro (ex.: carboidratos por uma unidade de medida)
 - Editar Tipo de Alimento
 - É o mesmo formulário da adição, carregando dados existentes
 - Os campos derivados aparecem agora com seus valores calculados
 - Visualização de Tipo de Alimento
 - Tela de leitura com todos os dados cadastrados e derivados
 - Excluir Tipo de Alimento
 - Tela padrão de confirmação antes da remoção definitiva
- Módulo de Alimento Consumido (Logged Food)
 - Listagem de Alimentos Consumidos
 - Exibe todos os registros cadastrados
 - Permite pesquisar, ordenar e selecionar itens
 - A partir desta tela, o usuário pode acessar:

- Adicionar novo Alimento Consumido
- Editar um Alimento Consumido existente
- Excluir um ou mais Alimentos Consumidos
- Visualizar dados do Alimento Consumido
- Adicionar Alimento Consumido
 - Tela com formulário solicitando o peso do alimento e um Tipo de Alimento
 - Também exibe campos derivados que serão calculados automaticamente após o cadastro (ex.: carboidratos pelo peso do alimento cadastro)
- Editar Alimento Consumido
 - É o mesmo formulário da adição, carregando dados existentes
 - Os campos derivados aparecem agora com seus valores calculados
- Visualização de Alimento Consumido
 - Tela de leitura com todos os dados cadastrados e derivados
- Excluir Alimento Consumido
 - Tela padrão de confirmação antes da remoção definitiva

6.1.2 Padronização de botões, ícones e outros atalhos.

O sistema backoffice foi construído utilizando o framework Django, que disponibiliza uma interface padronizada e consistente pelo Django Admin.

- Botões

- “+” Add



- Add [nome da entidade] “+”

ADD FOOD TYPE +

- “lápis” Change

Change

- Save e suas variações (“Save and add another” e “Save and continue editing”)

SAVE **Save and add another** **Save and continue editing**

- Delete

Delete

- History

HISTORY

- Go

Action: ----- 0 of 1 selected

FOOD NAME

Chocolate branco - Ferrero Rocher

- Ícones

- “+”, “lápis”, “x”

My actions

- 📝 Chocolate branco — Ferrero Rocher
Food type
- 📝 Queijo minas frescal — Fazenda Bela Vista
Food type
- + Queijo Minas Frescal — Fazenda Bela Vista
Food type
- + Pão de forma cenoura com chia sem glúten — Equilibre
Food type
- + YoPro Morango — Danone
Food type
- + Pão de mel — Belive
Food type
- + Chocolate branco - Ferrero Rocher
Food
- 📝 Chocolate branco — Ferrero Rocher
Food type
- ✖ Chocolate branco - Ferrero Rocher
Food

- *Breadcrumbs* (migalhas de pão - indicador de caminho navegado pela interface)

Home > Food > Foods

- Seta para baixo ou para o lado nos filtros

↓ By description

All

Chocolate branco
Pão de forma cenoura com chia
sem glúten
Pão de mel
Queijo minas frescal
YoPro Morango

→ By brand name

- Opção para mostrar ou não mostrar counts



- “Olho” para visualizar um food type de dentro de logged food
 - “Lápis” e “+” aqui também para editar e adicionar food type

Food type:

- “Sol”, “lua”, “meio a meio”
-
- Setas duplas “<<” para fazer a barra lateral esquerda sumir ou aparecer
-

- Padrões de tela
 - Layout responsivo
 - Páginas com *breadcrumbs* (migalhas de pão) para orientar o usuário
 - Estilo minimalista com predominância das cores:
 - Branco e preto (no modo noturno) (fundos)
 - Azul (ações primárias)
 - Cinza (ações secundárias)
 - Vermelho (ações de exclusão)
- Comportamentos padrão e atalhos
 - Interação uniforme em todas as telas de cadastro
 - Campos alinhados verticalmente
 - Confirmações em ações destrutivas
 - Pesquisa integrada nas listagens
 - Ordenação de dados ao clicar no nome das colunas
 - Selecionar múltiplos elementos e aplicar ações de uma vez só neles

7 PROJETO DE IMPLANTAÇÃO

7.1 Procedimentos de Implantação

7.1.1 Carga inicial

Para configuração inicial do sistema, espera-se a execução dos seguintes comandos:

```
pip install .  
python manage.py migrate  
python manage.py seed_initial_food_data
```

Adicionalmente, utilize o seguinte comando para criação do usuário inicial (será solicitado o nome e senha desejado para este usuário):

```
python manage.py createsuperuser
```

7.1.2 Execução

Execute o projeto com o seguinte comando:

```
python manage.py runserver
```

7.1.3 Backup e Restauração

Um backup dos dados do projeto pode ser gerado - em um arquivo do formato JSON chamado “dump” - com o uso do seguinte comando:

```
python manage.py dumpdata > dump.json
```

Um backup dos dados - contido em um arquivo do formato JSON chamado “dump” - pode ser carregado com o seguinte comando:

```
python manage.py loaddata > dump.json
```

7.1.4 Recuperação de acesso

Para recuperação da senha de acesso de um usuário, utilize o seguinte comando (neste exemplo, o nome do usuário desejado é “admin”):

```
python manage.py changepassword admin
```

7.2 Ambiente do sistema

Para execução do software, é requerido um equipamento que atenda os seguintes requerimentos:

- Python 3.12 ou superior instalado;
- Mínimo de 4GB de RAM;
- Capacidade de executar comandos de terminal no usuário que executará o sistema;
- Porta TCP/IP de número 8000 aberta e disponível para acesso do público.

No ambiente final de execução do sistema, próximo do seu lançamento ante ao público (veja item 8 - Cronograma de implementação), é esperado que o sistema esteja em execução em uma máquina virtual Linux, acessível na internet, fornecida por um provedor de hospedagem.

7.3 Demonstração de implementação

7.3.1 Telas de cadastro

As seguintes telas demonstram as operações de cadastro, alteração, exclusão e consulta de dois tipos de dados do sistema.

- Adição de Paciente (Patient)

Add patient

Name:	Emma
Height:	155
Weight:	50
Biological sex:	female
Physical activity level:	moderadamente ativo (exercício/esporte moderado 3-5x/semana)
<input type="button" value="SAVE"/> <input type="button" value="Save and add another"/> <input type="button" value="Save and continue editing"/>	

 The patient "Emma" was added successfully.

Select patient to change

Action: ----- Go 0 of 2 selected

	NAME	HEIGHT	WEIGHT	BIOLOGICAL SEX	PHYSICAL ACTIVITY LEVEL
<input type="checkbox"/>	Emma	155	50.0	female	moderadamente ativo (exercício/esporte moderado 3-5x/semana)
<input type="checkbox"/>	Ahri	160	80.0	female	pouco ativo (exercício/esporte leve 1-3x/semana)

2 patients

- Alterar Paciente (Patient)

Change patient

Ahri

Name:	Ahri	HISTORY
Height:	160	
Weight:	80.0	
Biological sex:	female	
Physical activity level:	sedentário (pouco ou nenhum exercício)	

Buttons: SAVE | Save and add another | Save and continue editing | Delete

✓ The patient "Ahri" was changed successfully.

Select patient to change

Action: ----- Go 0 of 2 selected

<input type="checkbox"/>	NAME	HEIGHT	WEIGHT	BIOLOGICAL SEX	PHYSICAL ACTIVITY LEVEL
<input type="checkbox"/>	Emma	155	50.0	female	moderadamente ativo (exercício/esporte moderado 3-5x/semana)
<input type="checkbox"/>	Ahri	160	80.0	female	sedentário (pouco ou nenhum exercício)

2 patients

- Visualização de Paciente (Patient)

View patient

Ahri

Name:	Ahri
Height:	160
Weight:	80.0
Biological sex:	female
Physical activity level:	pouco ativo (exercício/esporte leve 1-3x/semana)

Close

- Exclusão de Paciente (Patient)

Delete

Are you sure you want to delete the patient "Emma"? All of the following related items will be deleted:

Summary

- Patients: 1

Objects

- Patient: [Emma](#)

Yes, I'm sureNo, take me back

✓ The patient "Emma" was deleted successfully.

Select patient to change

Action: ----- 0 of 1 selected

<input type="checkbox"/>	NAME	HEIGHT	WEIGHT	BIOLOGICAL SEX
<input type="checkbox"/>	Ahri	160	80.0	female

1 patient

- Adição de Tipo de Alimento (Food Type)

Add food type

Description:	Pense Zero - iogurte Grego Tradicional
Brand name:	Batavo
Portion size:	100 <input style="width: 20px; height: 20px;" type="button" value="^"/>
Measure unit:	gram <input style="width: 20px; height: 20px;" type="button" value="^"/>
<input type="checkbox"/> Snack	
Kcal:	61 <input style="width: 20px; height: 20px;" type="button" value="^"/>
Total carbs:	5.9 <input style="width: 20px; height: 20px;" type="button" value="^"/>
Protein:	4.8 <input style="width: 20px; height: 20px;" type="button" value="^"/>
Total fat:	2.0 <input style="width: 20px; height: 20px;" type="button" value="^"/>
Container total weight:	100 <input style="width: 20px; height: 20px;" type="button" value="^"/>
Kcal a cada unidade de medida: —	
Carboidratos a cada unidade de medida: —	
Proteína a cada unidade de medida: —	
Gordura a cada unidade de medida: —	
Rico em: —	
<input style="background-color: #0070C0; color: white; padding: 5px; margin-right: 10px;" type="button" value="SAVE"/> <input style="background-color: #0070C0; color: white; padding: 5px;" type="button" value="Save and add another"/> <input style="background-color: #0070C0; color: white; padding: 5px;" type="button" value="Save and continue editing"/>	

✓ The food type "Pense Zero - iogurte Grego Tradicional — Batavo" was added successfully.

Select food type to change

Action:	-----	Go	0 of 13 selected
<input type="checkbox"/>	DESCRIPTION	BRAND NAME	
<input type="checkbox"/>	Pense Zero - iogurte Grego Tradicional	Batavo	
<input type="checkbox"/>	Banana	-	
<input type="checkbox"/>	Molho de tomate tradicional (sachê)	Pomarola	
<input type="checkbox"/>	Carne bovina patinho sem gordura grelhada	-	
<input type="checkbox"/>	Macarrão espaguete sem glúten	Urbano	
<input type="checkbox"/>	Pão de forma cenoura com chia sem glúten	Équilibre	
<input type="checkbox"/>	Barra de chocolate branco com avelã	Ferrero Rocher	
<input type="checkbox"/>	Queijo minas frescal	-	
<input type="checkbox"/>	Feijão preto cozido	-	
<input type="checkbox"/>	Peito de frango grelhado sem pele	-	
<input type="checkbox"/>	Arroz branco cozido	-	
<input type="checkbox"/>	YoPro Bebida Láctea Morango	Danone	
<input type="checkbox"/>	Pão de mel	Belive	

13 food types

- Alterar Tipo de Alimento (Food Type)

Change food type

Banana

Description:	Banana	
Brand name:		
Portion size:	100	▼
Measure unit:	milliliter ▼	
<input type="checkbox"/> Snack		
Kcal:	98.0	▼
Total carbs:	26.0	▼
Protein:	1.3	▼
Total fat:	0.1	▼
Container total weight:	▼	
Kcal a cada unidade de medida:	0.98 kcal	
Carboidratos a cada unidade de medida:	0.26 g	
Proteína a cada unidade de medida:	0.01 g	
Gordura a cada unidade de medida:	0.00 g	
Rico em:	carb (26.00%)	
<input type="button" value="SAVE"/> <input type="button" value="Save and add another"/> <input type="button" value="Save and continue editing"/>		

 The food type "Banana" was changed successfully.

Select food type to change

Action: 0 of 13 selected

<input type="checkbox"/>	DESCRIPTION	BRAND NAME	PORTION SIZE	MEASURE UNIT
<input type="checkbox"/>	Pense Zero - Iogurte Grego Tradicional	Batavo	100	gram
<input type="checkbox"/>	Banana	-	100	milliliter

- Consultar Tipo de Alimento (Food Type)

View food type

Barra de chocolate branco com avelã — Ferrero Rocher

Description:	Barra de chocolate branco com avelã
Brand name:	Ferrero Rocher
Portion size:	25
Measure unit:	gram
Snack:	✓
Kcal:	143.0
Total carbs:	13.0
Protein:	2.0
Total fat:	9.0
Container total weight:	90
Kcal a cada unidade de medida:	5.72 kcal
Carboidratos a cada unidade de medida:	0.52 g
Proteína a cada unidade de medida:	0.08 g
Gordura a cada unidade de medida:	0.36 g
Rico em:	carb (52.00%), fat (36.00%)

Close

- Excluir Tipo de Alimento (Food Type)

Delete

Are you sure you want to delete the food type "Pense Zero - iogurte Grego Tradicional — Batavo"? All of the following related items will be deleted:

Summary

- Food types: 1

Objects

- Food type: Pense Zero - iogurte Grego Tradicional — Batavo

[Yes, I'm sure](#)

[No, take me back](#)

 The food type "Pense Zero - iogurte Grego Tradicional — Batavo" was deleted successfully.

Select food type to change

Action: [Go](#) 0 of 12 selected

<input type="checkbox"/>	DESCRIPTION	BRAND NAME
<input type="checkbox"/>	Banana	-
<input type="checkbox"/>	Molho de tomate tradicional (sachê)	Pomarola
<input type="checkbox"/>	Carne bovina patinho sem gordura grelhada	-
<input type="checkbox"/>	Macarrão espaguete sem glúten	Urbano
<input type="checkbox"/>	Pão de forma cenoura com chia sem glúten	Équilibre
<input type="checkbox"/>	Barra de chocolate branco com avelã	Ferrero Rocher
<input type="checkbox"/>	Queijo minas frescal	-
<input type="checkbox"/>	Feijão preto cozido	-
<input type="checkbox"/>	Peito de frango grelhado sem pele	-
<input type="checkbox"/>	Arroz branco cozido	-
<input type="checkbox"/>	YoPro Bebida Láctea Morango	Danone
<input type="checkbox"/>	Pão de mel	Belive

12 food types

7.3.2 Operação que envolve cálculo do sistema

Na tela seguinte, é exibido um dado do tipo “Tipo de Alimento” (Food Type). Nota-se a presença de informações nutricionais computadas por unidade de medida do alimento (por exemplo, por grama ou mililitro), que são calculadas pelo sistema baseado nos dados fornecidos pelo usuário. Tais dados estão listados nos últimos campos destacados em vermelho.

Após completar o cadastro de um dado do tipo “Alimento Consumido” (Logged Food), que requer selecionar um dado do tipo “Tipo de Alimento” (Food Type), as informações nutricionais dele também são calculadas pelo sistema baseados no peso informado

View food type

Pense Zero - iogurte Grego Tradicional — Batavo

Description:	Pense Zero - iogurte Grego Tradicional
Brand name:	Batavo
Portion size:	100
Measure unit:	gram
Snack:	<input checked="" type="checkbox"/>
Kcal:	61.0
Total carbs:	5.9
Protein:	4.8
Total fat:	2.0
Container total weight:	100
Kcal a cada unidade de medida:	0.61 kcal
Carboidratos a cada unidade de medida:	0.06 g
Proteína a cada unidade de medida:	0.05 g
Gordura a cada unidade de medida:	0.02 g
Rico em:	—

Add logged food

Weight: 25

Food type: Barra de chocolate branco com avelã — Ferrero Rocher   

Kcal: —

Carboidratos: —

Proteína: —

Gordura: —

SAVE **Save and add another** **Save and continue editing**

View logged food

Barra de chocolate branco com avelã — Ferrero Rocher

Weight:	25
Food type:	Barra de chocolate branco com avelã — Ferrero Rocher
Kcal:	143.00
Carboidratos:	13.00 g
Proteína:	2.00 g
Gordura:	9.00 g

Close

7.3.3 Relatórios

Nas capturas de tela abaixo, é possível observar a capacidade do sistema de gerar relatórios. Vale notar o sistema de filtros, na barra à esquerda, que permite afunilar os resultados exibidos.

Select patient to change

Action:	NAME	HEIGHT	WEIGHT	BIOLOGICAL SEX	PHYSICAL ACTIVITY LEVEL	BASAL METABOLIC RATE
<input type="checkbox"/>	Emma	155	50.0	female	moderadamente ativo (exercício/esporte moderado 3-5x/semana)	-
<input type="checkbox"/>	Ahri	160	80.0	female	sedentário (pouco ou nenhum exercício)	-

2 patients

FILTER	
<input checked="" type="radio"/> Show counts	
↓ By name	All Ahri Emma
↓ By biological sex	All female male
↓ By physical activity level	All sedentário (pouco ou nenhum exercício) pouco ativo (exercício/esporte leve 1-3x/semana) moderadamente ativo (exercício/esporte moderado 3-5x/semana) muito ativo (exercício/esporte pesado 6-7x/semana) extremamente ativo (exercício/esporte pesado e trabalho físico intenso diariamente ou treino de 2x/dia)

Select food type to change

Action:	DESCRIPTION	BRAND NAME	PORTION SIZE	MEASURE UNIT	KCAL	SNACK
<input type="checkbox"/>	Pense Zero - iogurte Grego Tradicional	Batavo	100	gram	61.0	×
<input type="checkbox"/>	Banana	-	100	milliliter	98.0	×
<input type="checkbox"/>	Molho de tomate tradicional (sachê)	Pomarola	60	gram	15.0	×
<input type="checkbox"/>	Carne bovina patinho sem gordura grelhada	-	100	gram	219.0	×
<input type="checkbox"/>	Macarrão espaguete sem glúten	Urbano	80	gram	290.0	×
<input type="checkbox"/>	Pão de forma cenoura com chia sem glúten	Équilibre	50	gram	114.0	×
<input type="checkbox"/>	Barra de chocolate branco com avelã	Ferrero Rocher	25	gram	143.0	✓
<input type="checkbox"/>	Queijo minas frescal	-	100	gram	264.0	×
<input type="checkbox"/>	Feijão preto cozido	-	100	gram	77.0	×
<input type="checkbox"/>	Peito de frango grelhado sem pele	-	100	gram	159.0	×
<input type="checkbox"/>	Arroz branco cozido	-	100	gram	128.0	×
<input type="checkbox"/>	YoPro Bebida Láctea Morango	Danone	250	milliliter	152.0	✓
<input type="checkbox"/>	Pão de mel	Belive	45	gram	162.0	✓

13 food types

FILTER	
<input checked="" type="radio"/> Show counts	
→ By description	All Batavo Belive Danone Ferrero Rocher Pomarola Urbano Équilibre -
↓ By brand name	
→ By portion size	
→ By measure unit	
→ By kcal	
→ By snack	

Select logged food to change

Action: ----- Go 0 of 5 selected

<input type="checkbox"/> FOOD NAME	WEIGHT	KCAL	TOTAL CARBS	PROTEIN	TOTAL FAT	LOGGED AT
<input type="checkbox"/> Barra de chocolate branco com avelã — Ferrero Rocher	25	143.0	13.0	2.0	9.0	Dec. 1, 2025, 12:59 a.m.
<input type="checkbox"/> Peito de frango grelhado sem pele	90	143.1	0.0	28.8	2.2	Dec. 1, 2025, 12:26 a.m.
<input type="checkbox"/> Feijão preto cozido	70	53.9	9.8	3.1	0.4	Dec. 1, 2025, 12:26 a.m.
<input type="checkbox"/> Arroz branco cozido	80	102.4	22.5	2.0	0.2	Dec. 1, 2025, 12:26 a.m.
<input type="checkbox"/> Pão de mel — Belive	45	162.0	16.0	6.0	10.0	Dec. 1, 2025, 12:07 a.m.

5 logged foods

ADD LOGGED FOOD +

FILTER

- Show counts
- By logged at
 - Any date
 - Today
 - Past 7 days
 - This month
 - This year

Select logged meal to change

Action: ----- Go 0 of 4 selected

<input type="checkbox"/> LOGGED FOOD	MEAL TYPE	MODEL MEAL TYPE COMPONENT	LOGGED AT
<input type="checkbox"/> Peito de frango grelhado sem pele	Almoço	Plano Emagrecimento (Mês 1) - Almoço	Dec. 1, 2025, 12:27 a.m.
<input type="checkbox"/> Feijão preto cozido	Almoço	Plano Emagrecimento (Mês 1) - Almoço	Dec. 1, 2025, 12:26 a.m.
<input type="checkbox"/> Arroz branco cozido	Almoço	Plano Emagrecimento (Mês 1) - Almoço	Dec. 1, 2025, 12:26 a.m.
<input type="checkbox"/> Pão de mel — Belive	Lanche da tarde	Plano Emagrecimento (Mês 1) - Cesta	Dec. 1, 2025, 12:07 a.m.

4 logged meals

ADD LOGGED MEAL +

FILTER

- Show counts
- By logged food
- By meal type
 - All
 - Café da manhã
 - Almoço
 - Lanche da tarde
 - Docinho pós-almoço
 - Jantar
 - Cesta
- By model meal type component
- By logged at
 - Any date
 - Today
 - Past 7 days
 - This month
 - This year

8 CRONOGRAMA DE IMPLEMENTAÇÃO

No momento, o projeto encontra-se em sua primeira etapa, “Modelagem de dados e backoffice”.

Etapa	Resultado esperado	Duração
Modelagem de dados e backoffice	Backoffice funcional	4 semanas
Coleta de dados alimentícios	Base de dados inicial que alimentará o motor	1 semana
Motor de sugestão de refeições	Concepção técnica da proposta principal do projeto	8 semanas
Aplicativo móvel: cadastro de alimentos	Aplicativo móvel básico	4 semanas
Provisionamento de estrutura para período de testes	Aplicativo funcional e disponível para público seletivo para fins de testes	3 semanas
Aplicativo móvel: elaboração de plano alimentar com motor de sugestões de refeições	Exposição ao usuário da maior proposta de valor do projeto	8 semanas
Aplicativo móvel: diário de refeições	Funcionalidade de registro de alimentos consumidos	8 semanas
Aplicativo móvel: relatórios e acompanhamento	Capacidade do usuário acompanhar seu progresso	4 semanas
Mecanismos de autenticação e persistência de dados através de dispositivos	Capacidade do usuário utilizar o aplicativo em múltiplos dispositivos e ter seus dados protegidos	5 semanas
Aplicativo móvel: ajustes de plano alimentar	Capacidade do usuário ajustar, arquivar, e gerenciar planos alimentares	3 semanas
Aplicativo móvel: insights de diário alimentar	Exposição ao usuário de padrões de comportamento e pontos de atenção baseado no seu diário alimentar	4 semanas
Ajustes precedendo lançamento público	Vários ajustes baseados em feedback coletado durante o período de testes	2 semanas
Provisionamento para lançamento público	Estrutura pronta para lançamento amplo do aplicativo	3 semanas

9 CONSIDERAÇÕES FINAIS

Inúmeras são as dificuldades em perder peso. Nossa sistema, através de sua capacidade de fornecer planos alimentares customizados, realistas, e de fácil acompanhamento, visa minimizar ao máximo estas dificuldades.

Ao longo deste projeto, concluímos a primeira etapa de implementação do sistema almejado. A modelagem de dados, detalhada no capítulo 5, mostrou-se desafiadora, porém resultou em uma base de dados versátil e completa; que se demonstrou capaz de acomodar os vários casos de uso esperados. O uso do framework Django facilitou a criação de um backoffice simples porém suficiente para este estágio do projeto.

Esperamos que, conforme avançarmos nas demais etapas planejadas, o sistema prove sua viabilidade e valor.

10 REFERÊNCIAS

AMIRI, Maryam; LI, Juan; HASAN, Wordh. Personalized flexible meal planning for individuals with diet-related health concerns: System design and feasibility validation study. **JMIR formative research**, v. 7, p. e46434, 2023.

COUGHLIN, Steven S. *et al.* Smartphone applications for promoting healthy diet and nutrition: A literature review. **Jacobs journal of food and nutrition**, v. 2, n. 3, p. 021, 2015.

FERNÁNDEZ Segado, Sergio. Disinformation about diet and nutrition on social networks: a review of the literature. **Nutrición Hospitalaria**, Volume 42 N 2, p. 366-375, 2025.

GONSALVES, Maria Eduarda Cabral Pellicione Sulz; BARROS, Marília Vieira de Camargo; SILVA, Maria Cláudia da. A adoção de dieta flexível frente a adoção de dieta restritiva: Uma revisão de literatura. **Research, Society and Development**, v. 12, n. 13, p. e79121344189, 2023.

GIBSON, Alice A.; SAINSBURY, Amanda. Strategies to improve adherence to dietary weight loss interventions in research and real-world settings. **Behavioral sciences**, v. 7, n. 3, p. 44, 2017.

KANDEL, Sean. An Evidence-based Look at the Effects of Diet on Health. **Cureus**, vol 11,5 e4715, 2019.

OKUNOGBE, Adeyemi et al. Economic impacts of overweight and obesity: current and future estimates for 161 countries. **BMJ Global Health**, vol. 7,9, 2022.

Plano alimentar: o papel do nutricionista. Disponível em: <<https://www.endocrino.org.br/plano-alimentar-o-papel-do-nutricionista/>>. Acesso em: 15 sept. 2025.

POLLONI, Enrico G.F. **Sistemas de Informação**. São Paulo: Futura, 2001

REDMAN, Leanne M. *et al.* Effect of calorie restriction with or without exercise on body composition and fat distribution. **The journal of clinical endocrinology and metabolism**, v. 92, n. 3, p. 865–872, 2007.

SOMMERVILLE, I. (2018). **Engenharia de Software**. 10^a ed. Pearson Education.
Disponível em: <https://www.academia.edu/50882590/Sommerville_Software_Engineering_10ed>. Acesso em: 19 out. 2025.