

❖ Equipe:

- Renan Nunes Viana / 118210164
- Arthur Dantas Porto / 118210628
- Victor Paiva dos Santos / 118210854
- Willy Guimarães Moraes Barros / 118210278

❖ Tema:

- Saúde e Bem Estar

❖ Link GitHub:

- <https://github.com/RenanNunesViana/software-analysis-project-UFCG-2021.2>

Apresentação do projeto (ideia):

O objetivo principal do App é guiar o usuário em sua dieta, ajudando-o a controlar o seu consumo diário de alimentos com objetivo de otimizar o ganho de massa muscular.

O aplicativo irá disponibilizar diversas opções de alimentos para que o usuário monte sua própria dieta de acordo com as necessidades diárias de calorias. Também serão informados os horários para cada refeição.

Requisitos funcionais

[RF01] - O usuário deve fornecer informações para que o sistema possa sugerir uma margem da quantidade de carboidratos, proteínas e calorias.

[RF02] - O sistema deve calcular a quantidade ideal de calorias diárias a serem consumidas.

[RF03] - O sistema deve disponibilizar a criação de um cardápio semanal, sugerindo alimentos presentes no banco de dados.

[RF04] - O sistema deve informar o horário de cada refeição através de um alarme.

[RF05] - O sistema deve ter uma barra de pesquisa, para o usuário buscar alimentos.

Requisitos não funcionais

[RNF01] - O sistema será desenvolvido em java/javaScript utilizando spring boot e a biblioteca react.

[RNF02] - O sistema utilizará banco de dados relacional postgresql.

[RNF03] - O sistema funcionará sem acesso a internet.

[RNF04] - Será desenvolvido para Android e IOS com react native.

[RNF05] - O sistema deve ter tempo de resposta máximo de 1s.

[RNF06] - alimentos são objetos, adicionados manualmente no BD, com nome, valor calórico, carboidratos, proteínas, lipídios, fibra alimentar, cálcio, ferro, sódio, sais, açúcar.

USE CASE: fornecer informações pro sistema
ID:1
<p>Descrição:</p> <p>assim que o sistema perceber que o aplicativo foi aberto pela primeira vez ele ira requisitar a altura, o peso e o sexo do individuo.</p>
<p>Ator primário:</p> <p>Cliente</p>
Ator secundário:
Pré Condições:
<p>Fluxo Principal:</p> <ol style="list-style-type: none"> 1. O caso de uso inicia quando o cliente abre o app pela primeira vez. 2. O sistema fornece um formulário que contem requisições de altura, peso, idade e sexo do cliente. 3. O cliente preenche o formulário com os dados dele. 4. O sistema armazena esses dados.
Pós Condições:
Fluxos Alternativos:

USE CASE: Cálculo de calorias diárias
ID:2
Descrição: O sistema irá calcular a quantidade de calorias necessárias diariamente.
Ator primário: Sistema
Ator secundário: Cliente
Pré Condições: O cliente deve ter fornecido suas informações de acordo com o UC1.
Fluxo Principal: <ol style="list-style-type: none"> 1. O caso inicia quando o cliente termina de fornecer suas informações ou editar seus dados. 2. O sistema fará o cálculo da necessidade de calorias diárias com base na fórmula de Harris-Benedict (taxa de metabolismo basal) * 1,6: <ul style="list-style-type: none"> • TMB Mulher= $655,1 + (9,5 \times \text{Peso (kg)}) + (1,8 \times \text{Altura (cm)}) - (4,7 \times \text{Idade}) \times 1,6$ • TMB Homem= $66,5 + (13,8 \times \text{Peso (kg)}) + (5 \times \text{Altura (cm)}) - (6,8 \times \text{Idade}) \times 1,6$ 4. O sistema guardará as informações em uma tabela no BD.
Pós Condições:
Fluxos Alternativos:

USE CASE: Cardápio semanal
ID:3
Descrição: Criar um cardápio semanal.
Ator primário: Cliente
Ator secundário: Sistema
Pré Condições: O sistema deve ter calculado a quantidade de calorias indicada para serem consumidas diariamente.
Fluxo Principal: <ol style="list-style-type: none"> 1. O caso de uso inicia quando o cliente seleciona a opção "criar cardápio" no menu principal. 2. O sistema fornece um template de cardápio com campos vazios para serem preenchidos com os alimentos sugeridos para cada refeição. 3. ao clicar em um campo vazio referente à refeição, o sistema mostrará os alimentos para que o usuário escolha. 4. A soma calórica dos alimentos selecionados não deve ser menor que o TMB * 1.6 e só poderá ultrapassar esse valor no máximo 5%. 5. O sistema armazena o cardápio criado.
Pós Condições:
Fluxos Alternativos:

USE CASE: Cálculo do cardápio
ID:4
Descrição: Calcular as informações nutricionais do cardápio
Ator primário: Sistema
Ator secundário:
Pré Condições: O cliente deve ter selecionado um cardápio.
Fluxo Principal: <ol style="list-style-type: none"> 1. O caso de uso inicia quando o cliente termina de criar um cardápio. 2. O sistema calcula, para cada refeição no cardápio do cliente, a quantidade de seus nutrientes presentes. 3. Os valores calculados são mostrados ao cliente junto ao cardápio.
Pós Condições:
Fluxos Alternativos:

USE CASE: Hora da refeição
ID:5
Descrição: Alarme informando a hora em que o cliente deve se alimentar
Ator primário: Sistema
Ator secundário: cliente
Pré Condições:
Fluxo Principal: 1. O caso de uso inicia quando o cliente seleciona a opção "Horários" no menu principal. 2. O sistema fornece um campo vazio com para ser preenchido com o horário que o cliente toma seu café da manhã normalmente. 3. O cliente preenche o campo com um horário. 4. O sistema calcula os horários das refeições do cliente a partir do que foi fornecido por ele, sendo as próximas refeições feitas de 3 em 3 horas. 5. O cliente define um alarme de 15 min, 30 min ou 45 min minutos antes do horário de cada refeição.
Pós Condições: define o alarme como silencioso ou não.
Fluxos Alternativos:

USE CASE: Busca de alimentos
ID:6
Descrição: o sistema possui uma barra de pesquisa de alimentos
Ator primário: Sistema
Ator secundário: Cliente
Pré Condições:
Fluxo Principal: 1. O caso de uso inicia quando o cliente seleciona a opção "pesquisar alimento" no menu principal. 2. O sistema fornece uma barra de pesquisa ao cliente. 3. O cliente preenche a barra com o nome do alimento desejado. 4. O sistema procura o alimento no banco de dados e, caso seja encontrado, mostra o alimento e suas informações nutricionais, caso não seja encontrado mostra a mensagem "alimento não encontrado".
Pós Condições:
Fluxos Alternativos: alimento não encontrado. Volta para a página de busca de alimentos, mostrando a mensagem : "alimento não encontrado"

Diagrama de casos de uso

