

**Code*Girls***



Thais Cardozo Pinheiro - thaiscp59@gmail.com  
Luiza de Souza - luizadesouza140@gmail.com

### **Problema**

A Nutrilife contém um alto número de pacientes, fazendo com que o tempo de espera seja longo, pois realizam cadastro, consultas e acompanhamentos de forma presencial.

Por conta disso, a clínica está enfrentando dificuldades para atender seus pacientes, também possui necessidade de adicionar funcionários terceirizados no plano de funcionamento, profissionais na área de educação física que englobam em alguns planos (pacotes).

Por esta razão, está buscando soluções.

### **Solução**

Criamos um aplicativo que irá facilitar o contato do profissional com o cliente, de forma que o acompanhamento, consultas e informações estejam disponíveis para o paciente ter acompanhamento e orientação digitalmente.

Necessitando de um software que possa ser acessado em computadores e dispositivos móveis (android/ IOS) conectados a internet.

Assim será desenvolvido uma plataforma(aplicativo), que será utilizado a linguagem “Dart”, o banco de dados será o “MySQL”, Este sistema terá um visual dinâmico e intuitivo, conterá rápido processamento. Utilizando o framework “Flutter” desenvolvemos o visual do seu visual.



## **Levantamento de Requisitos**

- a.** aluno: poderá fazer login; acessar dados da conta; alterar senha; ativar/desativar modo escuro; criar e personalizar notificações; consultar plano alimentar; consultar treino; utilizar IMC; utilizar chat de comunicação; registrar fotos.
- b.** Funcionário(recepção): poderá gerar o código de validação do aplicativo; cadastrar, consultar, alterar e excluir dados de alunos.
- c.** funcionário: poderá gerar o código de validação do aplicativo; cadastrar, consultar, alterar e excluir dados de alunos; Criar e alterar Plano alimentar; visualizar imagens; Utilizar Chat de Comunicação.
- d.** Funcionário(educador): poderá criar e alterar planos de exercícios; consultar alunos; utilizar chat de comunicação.

## **Funcionais**

### **CADASTRO DE USUÁRIO**

- Cadastro do cliente seguido de um código de liberação concedido pela clínica;
- Login de usuários existentes. Para realizar o cadastro no aplicativo, o paciente deve ser maior de idade ou acompanhado com um responsável maior de 18 anos.

### **RELATÓRIOS**

- Notificações personalizadas;
- Os profissionais poderão fixar e alterar em um painel a dietas e treinos.

**OPCIONAIS** • Registro de peso e altura onde pode calcular seu IMC diariamente.

### **ACOMPANHAMENTO**

- Permitir que usuários atualizem informações pessoais;
- Registro de fotos do cliente para acompanhamento de refeições;
- Visualização de planos alimentares personalizados.

### **INTERAÇÃO**

- Chat para a comunicação direta entre profissional e paciente.
- RECURSOS
- Permitir que usuários ajustem preferências de notificação e privacidade.

### **SUPORTE TÉCNICO**

- Seção para FAQs e suporte técnico para resolução de problemas.

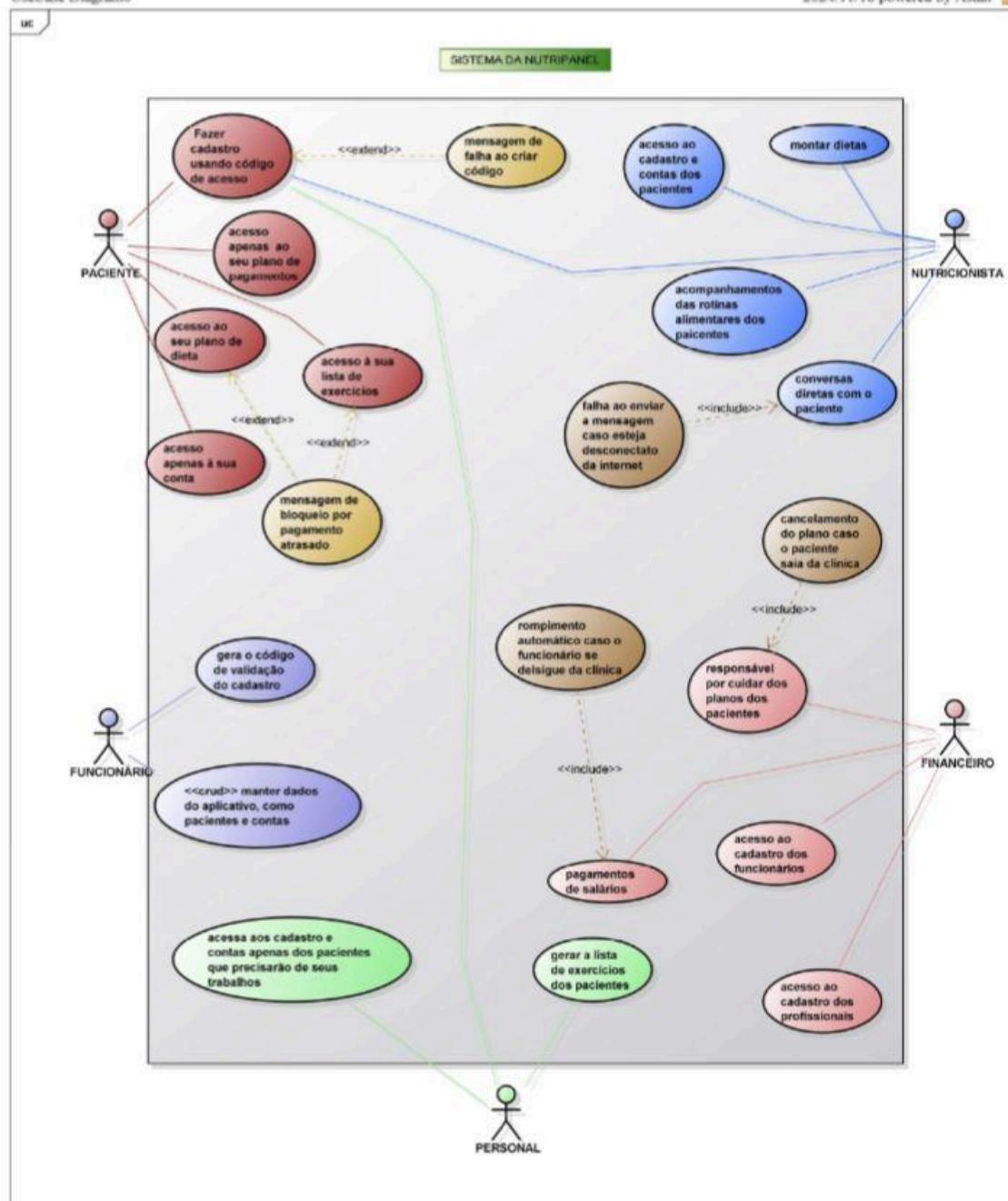
## **Funções**

- **Nutricionista:** Gerar o código de validação do aplicativo; atender consultar presencial; acessar ficha paciente; cadastrar, consultar, alterar e excluir dados de alunos na plataforma; Criar e alterar Plano alimentar.
- **Educador Físico(terceiro):** Contratado para fazer parte do plano (opcional do paciente); receber ficha do paciente; criar plano de treino na plataforma; atendimento online.
  - **Limpeza:** Responsável pela limpeza do prédio (interna).
  - **Recepção:** Cadastrar pacientes no aplicativo e na ficha da empresa; marcar consultas e agendar horários.
  - **financeiro:** se responsabiliza pelos planos escolhidos pelo paciente; pela forma de pagamento; distribuição dos salários dos funcionários e profissionais; acesso aos seus cadastros.

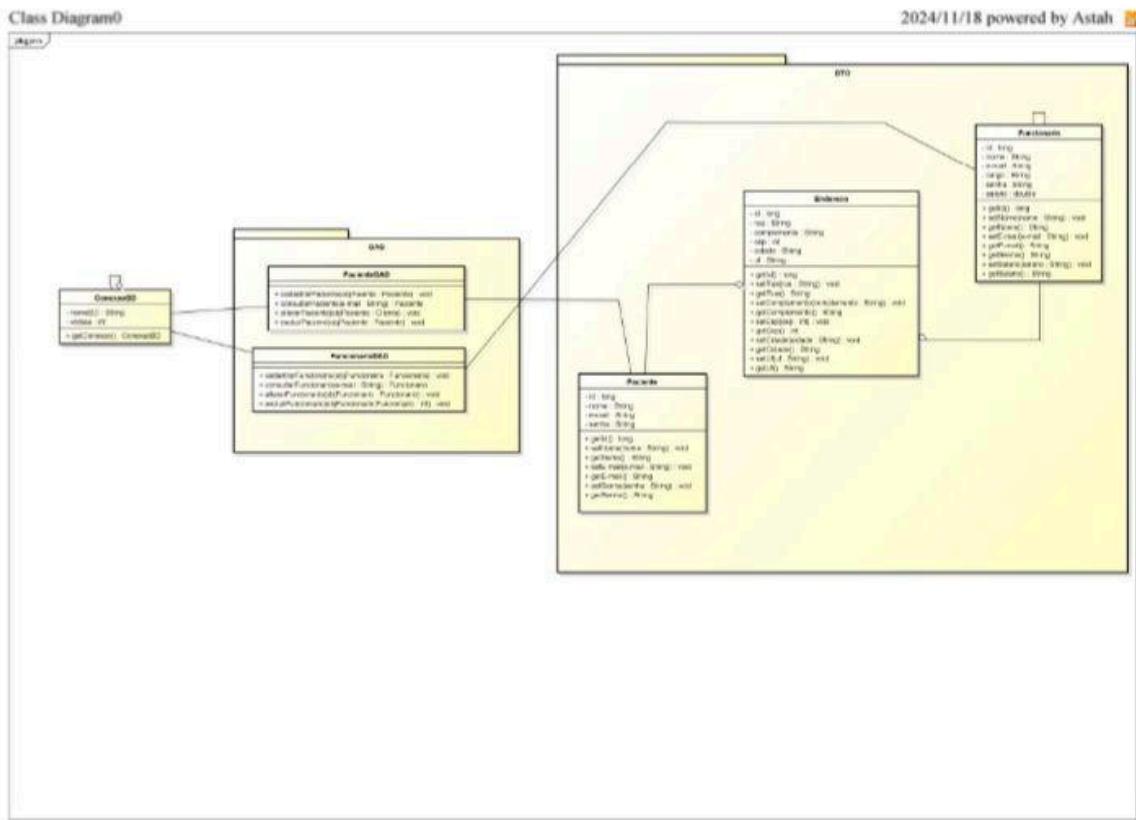
## Diagrama de caso de uso

UseCase Diagram6

2024/11/18 powered by Astah



## Diagrama de Classe



```

1 package ConexaoBD;
2 public class ConexaoBD {
3     private String nomeBD;
4     private int versao;
5     public ConexaoBD getConexaoBD(){
6         return null;
7     }
8 }

```

```

1 package DTO;
2 public class Endereco {
3     private long id;
4     private String rua;
5     private String complemento;
6     private int cep;
7     private String cidade;
8     private String uf;
9
10    public long getId() {
11        return id;
12    }
13    public void setId(long id) {
14        this.id = id;
15    }
16    public String getRua() {
17        return rua;
18    }
19    public void setRua(String rua) {
20        this.rua = rua;
21    }
22    public String getComplemento() {

```

```
23 return complemento;
24 }
25 public void setComplemento(String complemento) {
26 this.complemento = complemento;
27 }
28 public int getCep() {
29 return cep;
30 }
31 public void setCep(int cep) {
32 this.cep = cep;
33 }
34 public String getCidade() {
35 return cidade;
36 }
37 public void setCidade(String cidade) {
38 this.cidade = cidade;
39 }
40 public String getUf() {
41 return uf;
42 }
43 public void setUf(String uf) {
44 this.uf = uf;
45 }
46 }
```

```
1 10package DAO;
2 import DTO.Paciente;
3 public class PacienteDAO {
4 public void cadastrarPaciente (Paciente objPaciente){
5 }
6 public Paciente consultarPaciente(String email){
7 return null;
8 }
9 public void alterarPaciente(Paciente objPaciente){
10 }
11 public void excluirPaciente(Paciente objPaciente){
12 }
13}
```

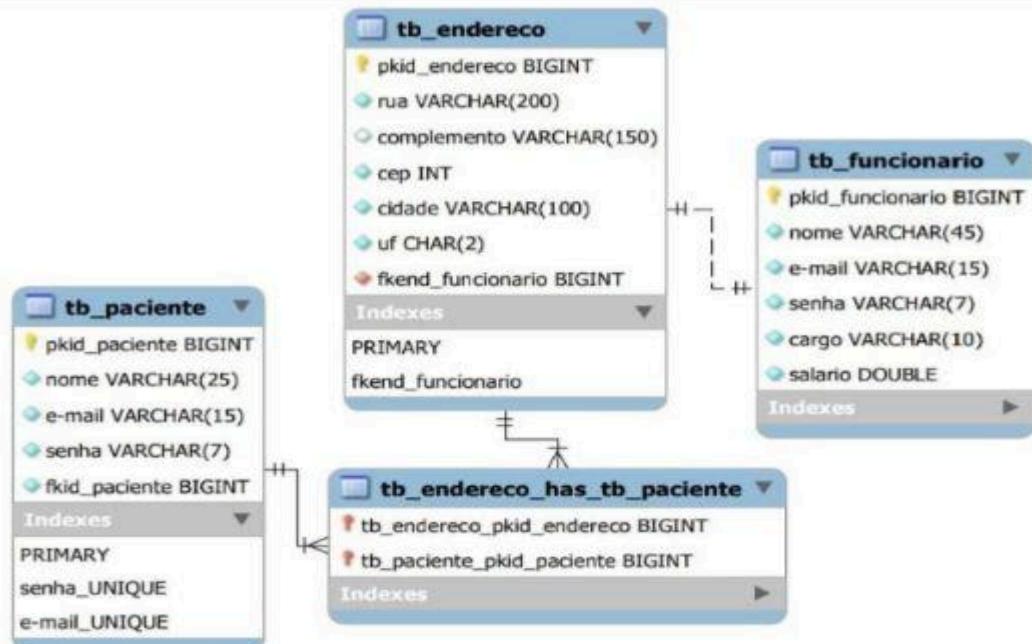
```
1 package DAO;
2 import DTO.Funcionario;
3 public class FuncionarioDAO {
4 public void cadastrarFuncionario (Funcionario objFuncionario){
5 }
6 public Funcionario consultarFuncionario(String email){
7 return null;
8 }
9 public void alterarCliente(Funcionario objFuncionario){
10 }
11 public void excluirFuncionario(Funcionario objFuncionario){
12 }
13 }
```

```
1 package funcionario;
2 public class funcionario {
3 private long id;
4 private String nome;
5 private String email;
6 private String cargo;
7 private String senha;
8 private Endereco objEndereco;
9 private double salario;
10 public long getId() {
11 return id;
12 }
```

```
13 public void setId(long id) {
14     this.id = id;
15 }
16 public String getNome() {
17     return nome;
18 }
19 public void setNome(String nome) {
20     this.nome = nome;
21 }
22 public String getEmail() {
23     return email;
24 }
25 public void setEmail(String email) {
26     this.email = email;
27 }
28 public String getCargo() {
29     return cargo;
30 }
31 public void setCargo(String cargo) {
32     this.cargo = cargo;
33 }
34 public String getSenha() {
35     return senha;
36 }
37 public void setSenha(String senha) {
38     this.senha = senha;
39 }
40 public double getSalario() {
41     return salario;
42 }
43 public void setSalario(double salario) {
44     this.salario = salario; 45 } 46 }

1 package paciente;
2 public class paciente {
3     private long id;
4     private String nome;
5     private String email;
6     private String senha;
7     Endereco objEndereco;
8     public long getId() {
9         return id;
10    }
11    public String getNome() {
12        return nome;
13    }
14    public void setNome(String nome) {
15        this.nome = nome;
16    }
17    public String getEmail() {
18        return email;
19    }
20    public void setEmail(String email) {
21        this.email = email;
22    }
23    public String getSenha() {
24        return senha;
25    }
26    public void setSenha(String senha) {
27        this.senha = senha;
28    }
29 }
```

## Banco de Dados



```

1 CREATE TABLE tb_funcionario ( pkid_funcionario BIGINT PRIMARY KEY AUTO_INCREMENT,
2 nome VARCHAR(45) NOT NULL,
3 email VARCHAR(15) NOT NULL,
4 senha VARCHAR(7) NOT NULL,
5 cargo VARCHAR(10) NOT NULL,
6 salario DECIMAL(10, 2) NOT NULL );

```

```

1 CREATE TABLE tb_endereco (
2 pkid_endereco BIGINT PRIMARY KEY AUTO_INCREMENT,
3 rua VARCHAR(200) NOT NULL,
4 complemento VARCHAR(150),
5 cep INT NOT NULL,
6 cidade VARCHAR(100) NOT NULL,
7 uf VARCHAR(2) NOT NULL,
8 fk_funcionario BIGINT,
9 FOREIGN KEY (fk_funcionario) REFERENCES tb_funcionario(pkid_funcionario) );

```

```

1 CREATE TABLE tb_paciente (
2 pkid_paciente BIGINT PRIMARY KEY AUTO_INCREMENT,
3 nome VARCHAR(25) NOT NULL,
4 email VARCHAR(15) NOT NULL,
5 senha VARCHAR(7) NOT NULL );

```

```

1 CREATE TABLE tb_endereco_has_tb_paciente (
2 tb_endereco_pkid_endereco BIGINT,
3 tb_paciente_pkid_paciente BIGINT, PRIMARY KEY (tb_endereco_pkid_endereco, tb_paciente_pkid_paciente),
4 FOREIGN KEY (tb_endereco_pkid_endereco) REFERENCES tb_endereco(pkid_endereco)
5 ON DELETE CASCADE ON UPDATE CASCADE,
6 FOREIGN KEY (tb_paciente_pkid_paciente) REFERENCES tb_paciente(pkid_paciente)
7 ON DELETE CASCADE ON UPDATE CASCADE

```

## Telas - Celular/Computador

The image displays two versions of the Nutripanel application interface: a mobile version on the left and a desktop/computer version on the right.

**Mobile Version (Left):**

- Header:** Nutripanel logo featuring an apple with a measuring tape.
- Login Form:** Fields for "digite seu nome" (name), "digite seu e-mail" (email), and "digite seu código" (code). A "faça o seu login" (login) button is at the bottom.
- User Profile:** Shows a placeholder user icon and the text "Usuário Paínel Nutrição".
- Navigation Options:** Icons for "Configurações" (Settings), "Lembretes e notificações" (Reminders and notifications), "Plano Alimentar" (Food plan), "Medidas IMC" (IMC measurements), "Chat" (messaging), and "Espaço Treino" (Training space).

**Desktop/Computer Version (Right):**

- Header:** Nutripanel logo featuring an apple with a measuring tape.
- Login Form:** Fields for "digite seu nome" (name), "digite seu e-mail" (email), and "digite seu código" (code). A "faça o seu login" (login) button is at the bottom.
- User Profile:** Shows a placeholder user icon and the text "Usuário Paínel Nutrição".
- Navigation Options:** Icons for "Configurações" (Settings), "Lembretes e notificações" (Reminders and notifications), "Plano Alimentar" (Food plan), "Medidas IMC" (IMC measurements), "Chat" (messaging), and "Espaço Treino" (Training space).

The image shows a mobile messaging interface between a patient and a nutritionist named Dra. Melissa.

**Dra. Melissa Online**

**Message**

**ACOMPANHAMENTO ALIMENTAR**

**Café da manhã**

- 2 ovos, 1 fatia de pão e uma banana

**Almoço**

- 100g de arroz, 80g de feijão, 200g de frango + Saladas a gosto

**Café da tarde**

- iogurte natural com whey, 1 tapioca com 2 banana

**Jantar**

- 2 ovos cozidos

**Message**

**ACOMPANHAMENTO ALIMENTAR**

**Café da manhã**

- 2 ovos, 1 fatia de pão e uma banana

**Almoço**

- 100g de arroz, 80g de feijão, 200g de frango + Saladas a gosto

**Café da tarde**

- iogurte natural com whey, 1 tapioca com 2 banana

**Almoço**

- 100g de arroz, 80g de feijão, 200g de frango + Saladas a gosto

**Jantar**

- 2 ovos cozidos

**Profile**



Usuário  
exemploUser@gmail.com

General Settings

- Mode
- Alterar senha
- Linguagem
- Permissões

Informações

- Privacidade
- Termos de uso
- Ajuda

**Profile**



Usuário  
exemploUser@gmail.com

Mode

Alterar senha

Linguagem

Permissões

Privacidade

Termos de uso

Ajuda



### Calculo IMC

**Feminino** **Masculino**

IDADE  
22

ALTURA  
1,68

PESO  
73

SEU IMC:  
25.86 sobrepeso

**CALCULAR** **LIMPAR**

### Calculo IMC

**Feminino** **Masculino**

IDADE  
22

ALTURA  
1,68

PESO  
73

SEU IMC:  
25.86 sobrepeso

**CALCULAR** **LIMPAR**

# Usuário

Páñel Ed. Fisica

*healthy life*

- Cronograma de treino
- Lembretes e notificações
- Chat
- ←Espaço Nutrição

# Usuário

Páñel Ed. Fisica

*healthy life*

- Lembretes e notificações
- Chat
- Cronograma de treino
- ←Espaço Nutrição

**Cronograma de Treino**

**Segunda - Quadriceps**

Agachamento Livre - 3 series, 12 repetições;  
Cadeira Extensora - 4 series, 12x2, 10 e 8 reps;  
Cadeira adutora - 3 series, 12 repetições;  
Legpress - 4 series, 10 repetições

**Terça - Peito, Ombro e Triceps**

Supino reto - 4 serie, 10 repetições;  
Crucifixo invertido - 3 serie, 12 repetições;  
Triceps na polia - 3 series, 15 repetições;  
Desenvolvimento c/alter - 3 series, 12 reps

**Quarta - Costas e Biceps**

Rosca martelo - 3 series, 20 repetições;  
Remada curvada - 3 series, 12 repetições;  
Puxada alta - 3 series, 15 repetições;  
Extensão de lombar - 3 series, 10 repetições.

**Quinta - Posterior e Gluteo**

Agachamento sumo - 4 series, 15 reps;  
Cadeira Abdutora - 4 series, 3x 15, 1x falha;  
Stiff - 3 series, 15 repetições;  
Bulgaro - 3 series, 12 repetições.

**Cronograma de Treino**

**Segunda - Quadriceps**

Agachamento Livre - 3 series, 12 repetições;  
Cadeira Extensora - 4 series, 12x2, 10 e 8 reps; Cadeira adutora - 3 series, 12 repetições;  
Legpress - 4 series, 10 repetições

**Terça - Peito, Ombro e Triceps**

Supino reto - 4 serie, 10 repetições;  
Crucifixo invertido - 3 serie, 12 repetições;  
Triceps na polia - 3 series, 15 repetições;  
Desenvolvimento c/alter - 3 series, 12 reps

**Quarta - Costas e Biceps**

Rosca martelo - 3 series, 20 repetições;

**Quinta - Posterior e Gluteo**

Agachamento sumo - 4 series, 15