**DCC060 – Trabalho Prático**

**GRUPO: Lohan Ferreira, Lucas Carvalho**

**1. INTRODUÇÃO**   
Desenvolvimento conceitual e lógico de um sistema (banco de dados) para um app que gerencia clínicas nutricionistas onde podem ser encontrados todos os clientes sendo atendidos, nutricionistas trabalhando em clinicas, dietas e tratamentos sendo realizados assim como receitas e produtos para auxilio dessas dietas e seus fornecedores.  
  
**2. REQUISITOS DO PROJETO**   
O sistema deve ser capaz de armazenar informações sobre as várias clínicas participantes (guardando código, nome da clínica e endereço) além de todas as nutricionistas que trabalham em cada uma delas (contendo CPF, nome e sua especialização).

O sistema deve permitir o cadastro de pacientes (CPF, nome, endereço, telefones e planos de saúde que possui) e permitir que os mesmos sejam capazes de marcar consultas ( armazenando a data) com as nutricionistas, e se desejar iniciar um tratamento .

O sistema deve permitir a escolha do tipo de tratamento a ser seguido pelo cliente e guardar informações importantes referente ao tipo de tratamento.

O sistema deve permitir o envio de mensagens entre pacientes e nutricionistas guardando data de envio.

O sistema deve permitir aos pacientes o cadastro de uma conta para acesso ao sistema.

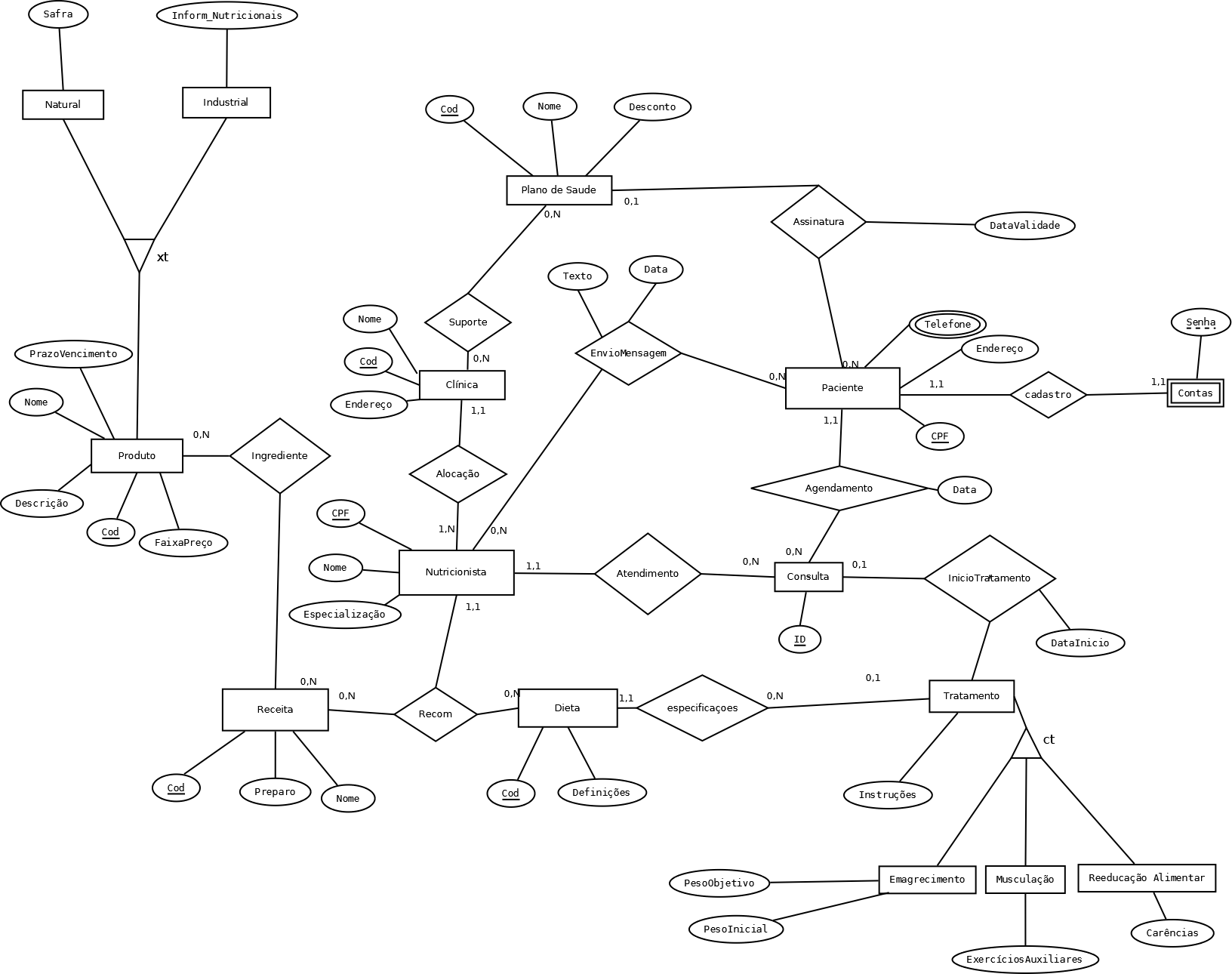
O sistema deve permitir o cadastro de dietas que serão seguidas pelos pacientes em tratamento bem como permitir que nutricionistas recomendem receitas adicionais para uma dieta em específico.

O sistema deve possuir uma lista de planos de saúde aceitos pelas diversas clínicas permitindo descontos ou coberturas nos tratamentos.

As receitas recomendadas por nutricionistas devem ter seus produtos ( ou ingredientes) armazenados.⋈ {\displaystyle \bowtie } ⋈ {\displaystyle \bowtie }

|  |
| --- |
| **Regras de Negócio** |
| 1. O Cpf das nutricionistas e dos pacientes devem ser compostos por 11 dígitos numéricos. 2. O desconto máximo permitido por plano de saúde é de 50% do valor do tratamento. 3. Ao cadastrar um produto no sistema, seu prazo de vencimento deve ser de no mínimo 1 mês após a data de cadastro. 4. Descrições de produtos não devem conter mais que 200 caracteres. 5. Preparo de receitas não devem conter mais que 300 caracteres. 6. A data de início de um tratamento deve ser no máximo 1 semana após a consulta que deu início ao tratamento. 7. A safra de um produto natural deve ser de no máximo 1 ano anterior a data de cadastro do produto. 8. Para receber desconto num tratamento, o plano de saúde de um cliente deve ainda ser válido (não ter ultrapassado a data de validade). 9. Nos tratamentos para emagrecimento o peso objetivo não deve ser maior que o peso início. 10. Contas cadastradas pelos pacientes devem possuir senha de no máximo 9 digitos e no mínimo 6 digitos, sendo todos numéricos (0 a 9). |

**3. MODELAGEM RELACIONAL**

****

**3.2 Modelagem lógica**

Paciente(CPF,Nome,Endereço,DataNascimento,Cod\_Plan,DataValidade)

Cod\_Plan referencia plano de saúde

Telefones(CpfP,Número)

CpfP referencia paciente

Clínica(Cod,Nome,Endereço)

Plano de Saude (Cod,Nome,ValorDesconto)

PlanosAceitos(Cod\_Plan,Cod\_Clin)

Cod\_Plan referencia plano de saúde

Cod\_Clin referencia clinica

Nutricionista(CPF,Cod,Nome,Especializacao)

Cod referencia Clínica

EnvioMensagem(CpfP,CpfN,Texto,Data)

CpfP referencia Paciente

CpfN referencia Nutricionista

Conta(CpfP,Senha)

CpfP referencia Paciente

Consulta(CpfP,CpfN,ID.Data)

CpfP referencia Paciente

CpfN referencia Nutricionista

Dieta(Cod,Definições)

Tratamento(,CpfP,CpfN,ID ,Cod\_Dieta,Instrucoes,Data\_Inicio)

Cod\_Dieta referencia Dieta, e este campo não pode ser Null.

CpfP,CpfN ,IDreferenciam Consulta

Emagrecimento(CpfP,CpfN,ID ,PesoObjetivo)

CpfP,CpfN,ID referenciam Tratamento

Musculação(CpfP,CpfN ,ExerciciosAuxiliares)

CpfP,CpfN,ID referenciam Tratamento

ReeducaçãoAlimentar(CpfP,CpfN,Carencias)

CpfP,CpfN,ID referenciam Tratamento

Receita(Cod,CpfN,Preparo,Nome)

CpfN referencia Nutricionista

Recomendação(Cod\_Dieta,CpfN,Cod\_R)

Cod\_Dieta referência Dieta

CpfN referencia Nutricionista

Cod\_R referencia receita

Cada dieta possui no máximo 4 receitas recomendadas.

Produto(Cod, Descrição,Faixa de Preço,Nome)

ReceitaProduto(Cod\_P,Cod\_R)

Cod\_P referencia Produto

Cod\_R referencia Receita

Natural(Cod,Safra)

Cod referencia produto

Industrial(Cod,Inform\_Nutricionais)

Cod referencia produto

**4 CONSULTAS EM ÁLGEBRA RELACIONAL**

|  |
| --- |
| **Consulta 1: “Quais nutricionistas trabalham na clínica ‘X’?”** |
| ΠNome(Nutricionista ⋈Cod (σNome=’X’Clínica)) |
|  |

|  |
| --- |
| **Consulta 2: “Planos de Saúde aceito por todas as clínicas”** |
| Π Nome(PlanodeSaude ⋈cod=cod\_plan (PlanosAceitos / (πcod\_clin PlanosAceitos) |

|  |
| --- |
| **Consulta 3: “Planos de Saúde aceitos pela clinica ‘X’?”** |
| Π Nome(PlanodeSaude ⋈cod=cod\_plan (PlanosAceitos ⋈cod\_clin = cod(ΠCod(σnome=’x’Clínica)))) |

|  |
| --- |
| **Consulta 4: “Nome dos pacientes que consultaram com a nutricionista ‘A’?”** |
| Π Nome (Paciente ⋈Cpf(Π CpfP(Consulta ⋈CpfN=Cpf (ΠCpf(σnome=’A’Nutricionista))))) |

|  |
| --- |
| **Consulta 5: “Planos de Saúde aceitos somente pela Clinica ‘X’?”** |
| ρ(Planos,((Π Cod\_plan (PlanosAceitos ⋈ cod\_clin = cod(ΠCod(σnome=’x’Clínica)))) - Π Cod\_plan(PlanosAceitos ⋈cod\_plan = cod\_plan^cod\_clin <> cod\_clin(PlanosAceitos ⋈cod\_clin = cod(ΠCod(σnome=’x’Clínica)))))  Π Nome(PlanodeSaude ⋈cod=cod\_plan Planos) |

|  |
| --- |
| **Consulta 6: “Nome dos pacientes em tratamento com a nutricionista ‘A’?”** |
| Π Nome (Paciente ⋈Cpf(Π CpfP(Tratamento ⋈CpfN=Cpf (ΠCpf(σnome=’A’Nutricionista))))) |

|  |
| --- |
| **Consulta 7: “Receitas recomendadas pela nutricionista ‘A’?”** |
| Π Nome,Preparo (Receita ⋈Cod(Π Cod\_R(Recomendação ⋈CpfN=Cpf (ΠCpf(σnome=’A’Nutricionista))))) |
|  |

|  |
| --- |
| **Consulta 8: “Produtos usados em exatamente uma receita?”** |
| (Π Cod\_P Produto) – (Π Cod\_P(ProdutoReceita ⋈ Cod\_R<>Cod\_R ^ Cod\_P = Cod\_P ProdutoReceita)) |

**5 MODELO FÍSICO**

**5.1 Tabelas**

As tabelas criadas seguem conforme o modelo lógico descrito na seção 3.

**5.2 Índices**Foram criados índices em tabela hash para diversas tabelas que dispõem to atributo “Nome” , como pacientes, nutricionistas e produtos,para agilizar pesquisas dos mesmos.

**5.3 Gatilhos e Funções**Foram criados gatilhos para verificações relacionadas às regras de negócio como validade dos dados inseridos e cálculo de desconto.  
  
**6. CARGA E VERIFICAÇÃO**

O banco de dados assim como sua carga de dados pode ser encontrada no arquivo bank.sql .

**7.2 Verificação**(INSERTs, UPDATEs e DELETEs que demonstram o correto comportamento dos controles de integridade criados)

--Verificação de inserção com cpf invalido;

Insert into paciente values (300,'Jota','Rua Labras Silveira', '2002-10-23');

--Inserindo dados pra outros testes;

insert into paciente values (30056078113,'Jota','Rua Labras --Silveira','2002-10-23');

insert into clinica values (33,'Clinica de Testes','Rua Sao Pedro');

insert into nutricionista values (12075728630,33,'Mariana');

insert into consulta values (130, 30056078113, 12075728630, '2017-11-19');

--Verificação da data limite para iniciar tratamento

insert into tratamento values (NULL, 30056078113, 12075728630, 130, '2017-11-30', NULL, 50);

insert into tratamento values (NULL, 30056078113, 12075728630, 130, '2017-11-20', NULL, 50);

--Verificando integridade de um tipo de tratamento

Insert into table emagrecimento (30056078113, 12075728630, 130,90, 130);

Insert into table emagrecimento (30056078113, 12075728630, 130,140, 90);

--Verificação descontos de planos de saude

update plano\_saude

set desconto = 64

where cod = 1

--Verificação de produtos

update prod\_industrial

set validade = '2016-01-01'

update prod\_natural

set safra = '2016-01-01'

**7. CONSULTAS EM SQL**

|  |
| --- |
| **Consulta 1: “Quais nutricionistas trabalham na clínica ‘A’?”** |
| select n.nome from nutricionista n , clinica c where n.cod = c.cod and c.nome = 'Clinica A' |

|  |
| --- |
| **Consulta 2: “Planos de Saúde aceito por todas as clínicas”** |
| select p.nome  from plano\_saude p  natural join planos\_aceitos pa  where p.cod = pa.cod\_plan  group by p.nome  having count (pa.cod\_clin) =(select count(\*) from clinica) |

|  |
| --- |
| **Consulta 3: “Planos de Saúde aceitos pela clinica ‘A’?”** |
| select p.nome  from plano\_saude p , planos\_aceitos pa, clinica c  where p.cod = pa.cod\_plan and pa.cod\_clin = c.cod and c.nome = 'Clinica A' |

|  |
| --- |
| **Consulta 4: “Nome dos pacientes que consultaram com a nutricionista ‘A’?”** |
| select p.nome  from paciente p , consulta c, nutricionista n  where c.cpfp = p.cpf and c.cpfn = n.cpf and n.nome = 'Vania' |

|  |
| --- |
| **Consulta 5: “Planos de Saúde aceitos somente pela Clinica ‘C’?”** |
| select p.nome from (  select distinct pa.cod\_plan, pa.cod\_clin from planos\_aceitos pa, clinica c  where pa.cod\_clin = c.cod and c.nome = 'Clinica C'  except  (select distinct p1.cod\_plan, p1.cod\_clin from planos\_aceitos p1 , planos\_aceitos p2  where p1.cod\_plan = p2.cod\_plan and p1.cod\_clin != p2.cod\_clin) ) as pc, plano\_saude p  where p.cod = pc.cod\_plan |

|  |
| --- |
| **Consulta 6: “Nome dos pacientes em tratamento com a nutricionista ‘Vania’?”** |
| select p.nome  from nutricionista n, tratamento t, paciente p  where t.cpfn = n.cpf and t.cpfp = p.cpf and n.nome = 'Vania' |

|  |
| --- |
| **Consulta 7: “Receitas recomendadas pela nutricionista ‘A’?”** |
| select r.nome  from receita r, recomendacao rc, nutricionista n  where r.cod = rc.cod\_r and rc.cpfn = n.cpf and n.nome = 'Lara Croft' |

|  |
| --- |
| **Consulta 7: “Receitas recomendadas pela nutricionista ‘A’?”** |
| select r.nome  from receita r, recomendacao rc, nutricionista n  where r.cod = rc.cod\_r and rc.cpfn = n.cpf and n.nome = 'Lara Croft' |

|  |
| --- |
| **Consulta 7: “Receitas recomendadas pela nutricionista ‘A’?”** |
| select r.nome  from receita r, recomendacao rc, nutricionista n  where r.cod = rc.cod\_r and rc.cpfn = n.cpf and n.nome = 'Lara Croft' |

|  |
| --- |
| **Consulta 8: “Produtos usados em exatamente uma receita?”** |
| select distinct rc.cod\_p from receita\_produto rc  except  (select distinct rc1.cod\_p from receita\_produto rc1 , receita\_produto rc2  where rc1.cod\_p = rc2.cod\_p and rc1.cod\_r != rc2.cod\_r) |

|  |
| --- |
| **Consulta 9: “Tratamentos e suas dietas”** |
| select t.id, d.cod  from tratamento t  left join dieta d on t.cod\_dieta = d.cod  order by id |

**8. TELAS DA APLICAÇÃO**

(mostrar algumas telas da aplicação e relacioná-las com os requisitos de negócio da seção 2. O sistema não precisará ser completo. Basta implementar algumas telas de inserção e resultados de consultas que sejam interessantes para a aplicação que está desenvolvendo, de forma a demonstrar que é capaz de criar uma aplicação que se comunique com um SGBD relacional. O código do sistema deve ser enviado em conjunto com o relatório.)