

UNILASALLE – RJ

GABRIEL CUSTODIO DE AZEVEDO

DESENVOLVIMENTO DO PROJETO DA EMPRESA GH FLUSÃO

NITERÓI
2017

GABRIEL CUSTODIO DE AZEVEDO

DESENVOLVIMENTO DO PROJETO DA EMPRESA GH FLUSÃO

PROFESSOR: GEORGE HAMILTON

DISCIPLINA: ENGENHARIA DE SOFTWARE

TURNOS: NOITE

NITERÓI
2017

Sumário

1.	Levantamento de Requisitos.....	4
1.1	Objetivo	4
1.2	Justificativa.....	4
1.3	Descrição	4
2	Análise	5
2.1	Diagrama de classe Conceitual.....	5
2.2	Lista de evento	5
2.3	Diagrama de caso de uso	6
2.4	Diagrama de Estado	7
2.5	Diagrama de Classe	8
3	Projeto.....	9
3.1	D.E.D.....	9
3.2	Script de criação das tabelas	10

1. Levantamento de Requisitos

1.1 Objetivo

O presente sistema tem por finalidade implementar os processos de controle da empresa **GH FLUSÃO** para melhor funcionamento e visibilidade dos mesmos. Processos esse de cadastrar cliente, funcionários, projetos e departamentos, visando sempre a otimização da **GH FLUSÃO** para que a mesma se mantenha lide absoluta no mercado.

1.2 Justificativa

Em função da crescente demanda de solicitações de projetos, verificou-se a necessidade de melhorar a organização dos documentos dos processos internos da empresa, tão quando os próprios projetos e métodos da empresa no dia-a-dia, para melhorar a eficácia da empresa e possibilitar a eficiência com os clientes.

1.3 Descrição

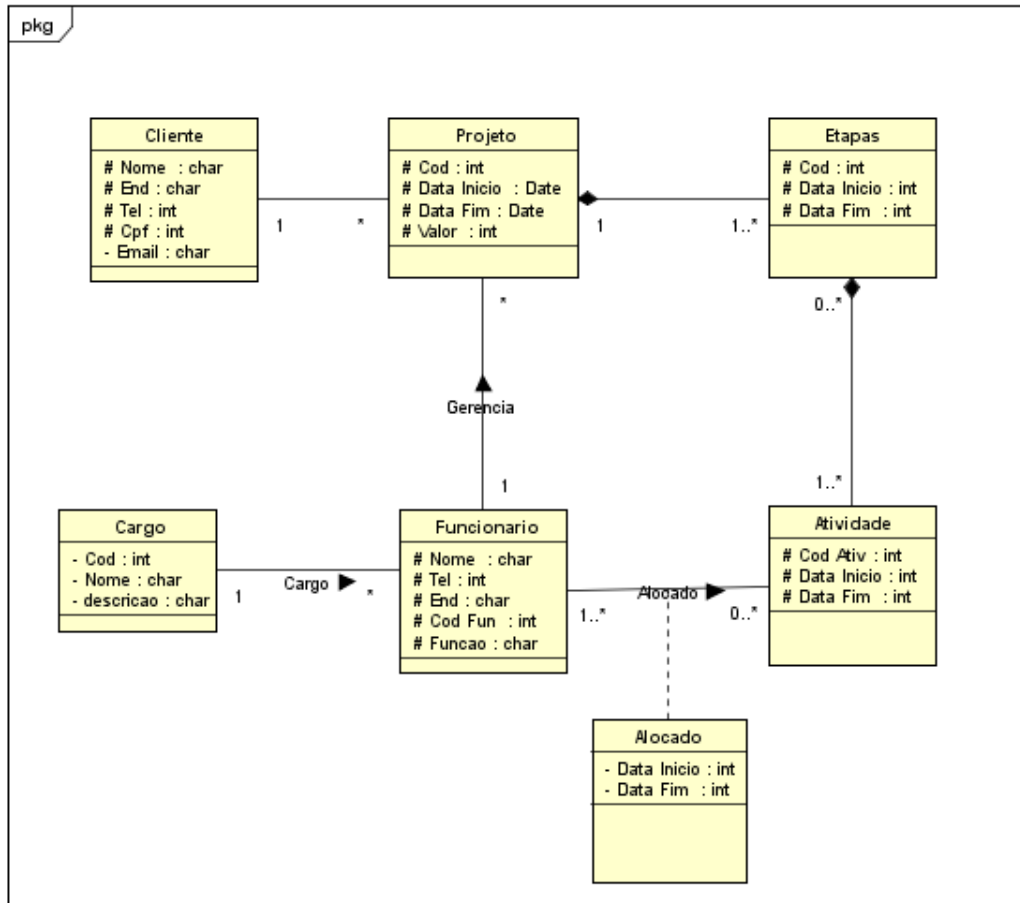
A proposta a seguir tem como objetivo melhorar os processos da **GH FLUSÃO** líder no mercado em que atua, a partir das seguintes premissas:

1. O Sistema deverá manter um controle dos clientes com nome, CPF, endereço, telefone e e-mail que solicitarem projetos
2. O sistema deverá manter um controle dos projetos com código, data e início, data de finalização e valor
3. Cada projeto é composto por etapas, toda etapa possui um código, uma data de início e uma data de finalização
4. Cada etapa é composta por atividades compostas por código, data início e data de finalização
5. O sistema devera manter um cadastro de seus funcionários com nome, código, telefone, endereço e função
6. O sistema deverá manter um cadastro dos cargos de seus funcionários
7. Cada projeto é gerenciado por um funcionário (não existe a necessidade de um histórico de gerentes)
8. Os funcionários podem ser alocados em várias atividades e as atividades podem conter vários funcionários (é necessário manter um histórico de alocação)

Por meio dos requisitos listados acima, foram formados diagramas que auxiliaram para a construção do futuro sistema.

2 Analise

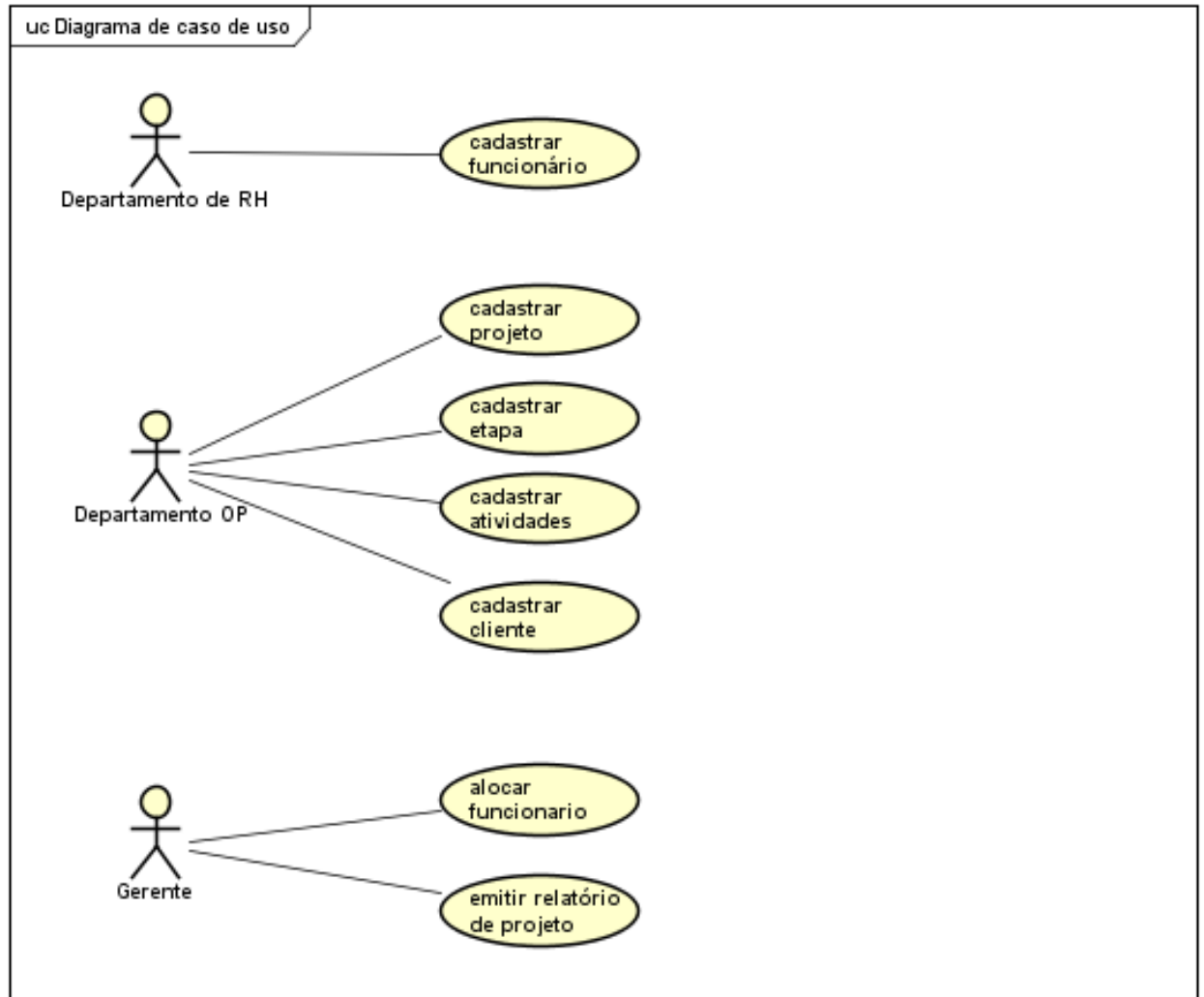
2.1 Diagrama de classe Conceitual



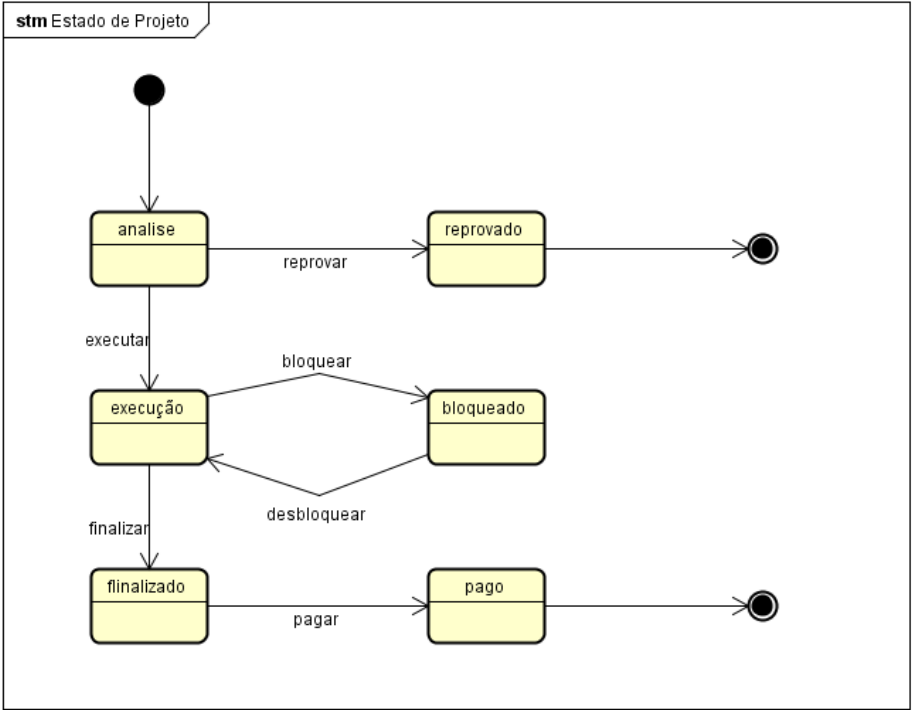
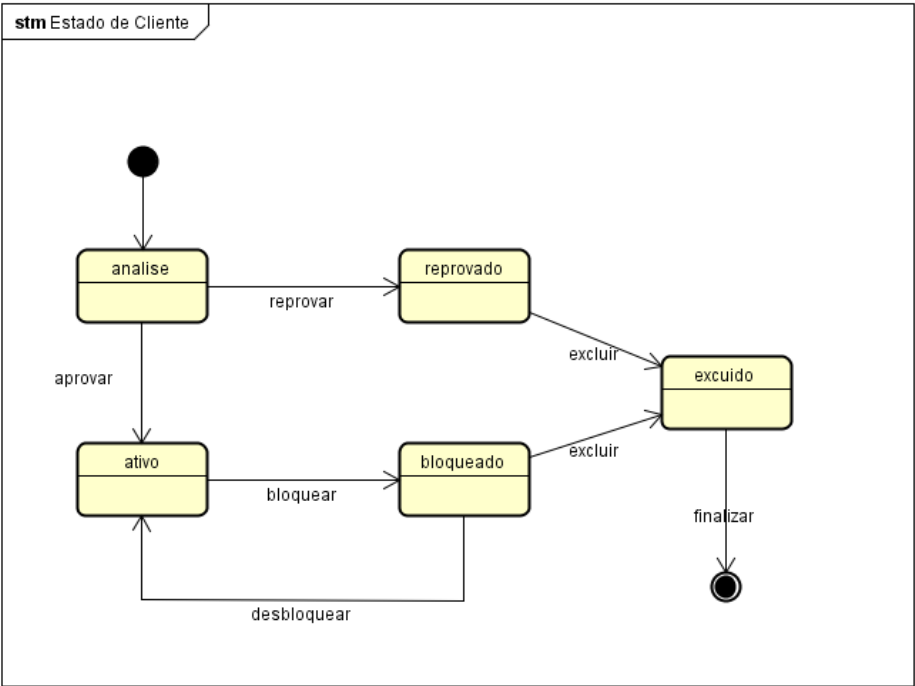
2.2 Lista de evento

1. Departamento de RH Cadastra Funcionários
2. Dep. Operacional cadastra projeto
3. Dep. Operacional cadastra etapas
4. Dep. Operacional cadastra atividades
5. Gerente de projeto aloca funcionários
6. Dep. Operacional cadastra cliente
7. Gerente emite relatório de projeto

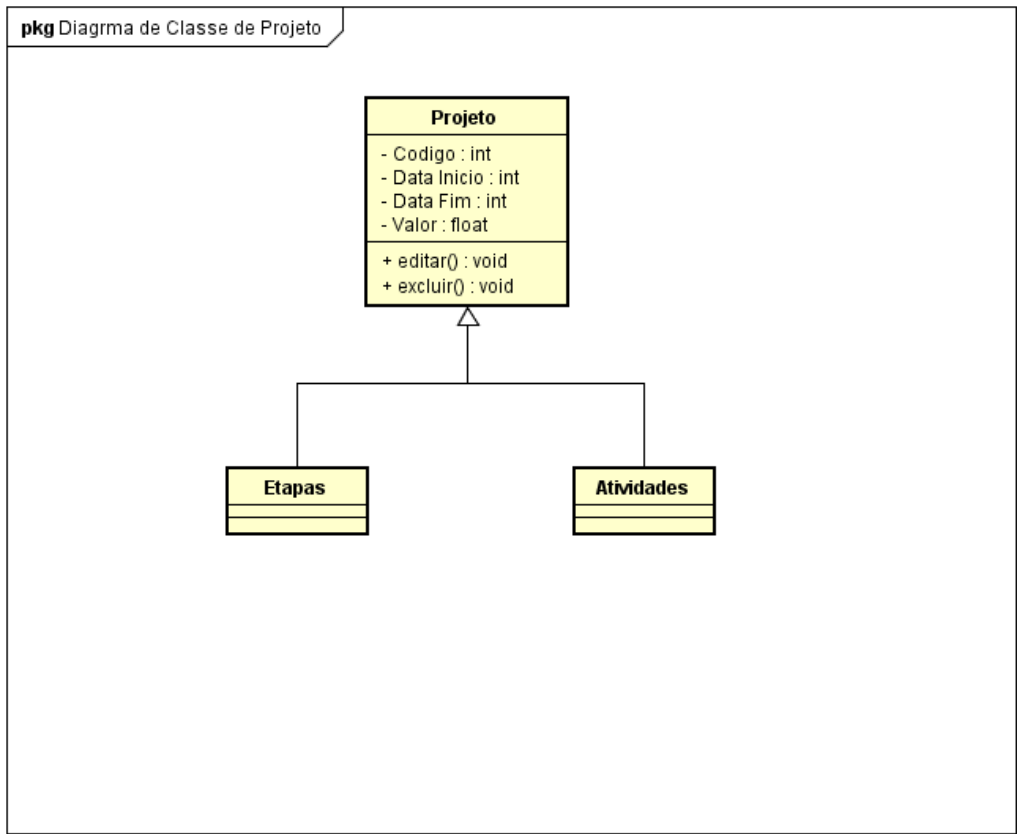
2.3 Diagrama de caso de uso



2.4 Diagrama de Estado

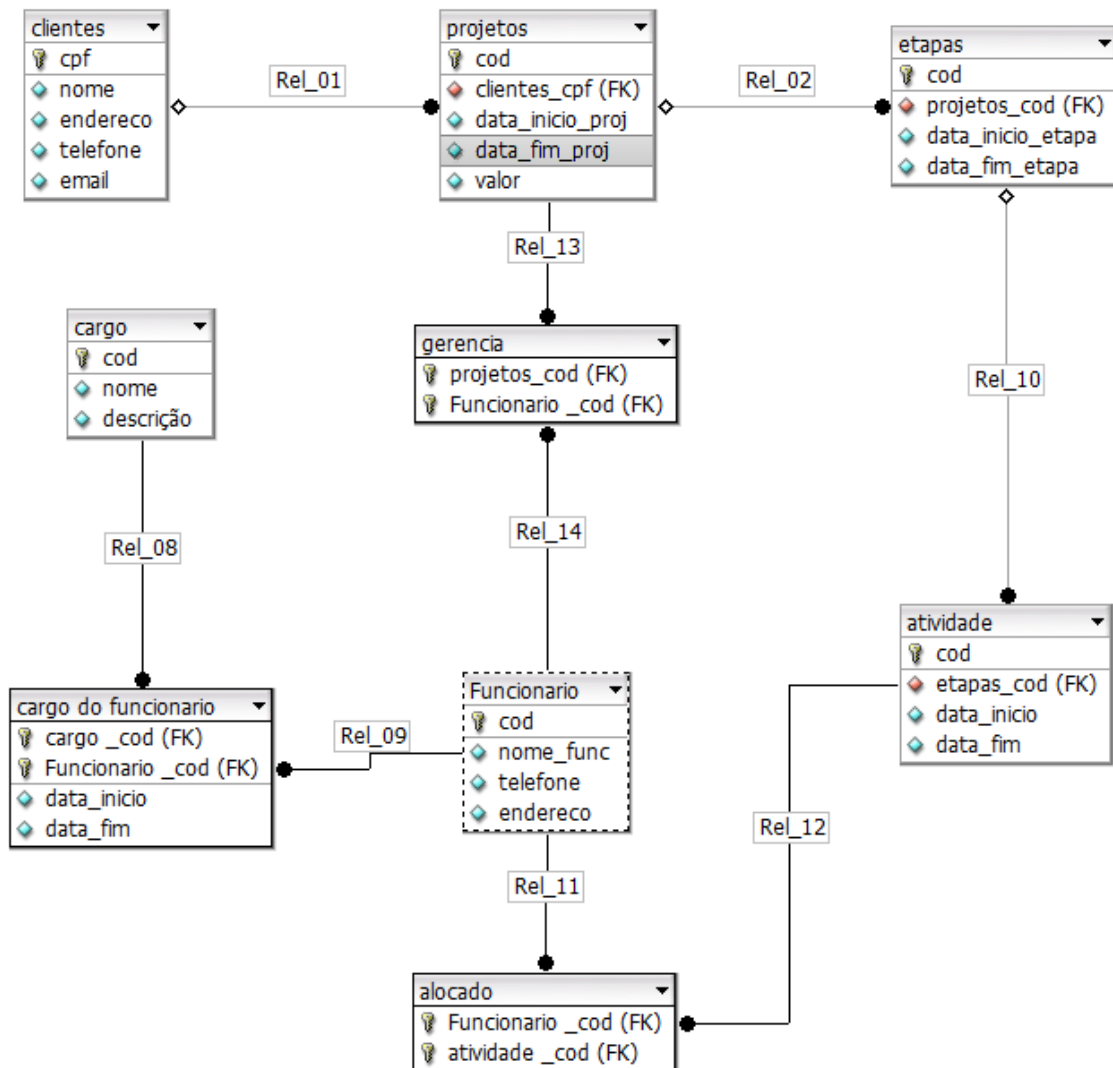


2.5 Diagrama de Classe



3 Projeto

3.1 D.E.D



3.2 Script de criação das tabelas

1. Tabela Clientes

```
CREATE TABLE "CLIENTES"  
(  
    "CPF" NUMBER(15,0) NOT NULL ENABLE,  
    "NOME" VARCHAR2(15) NOT NULL ENABLE,  
    "ENDereco" VARCHAR2(40) NOT NULL ENABLE,  
    "TELEFONE" NUMBER(10,0) NOT NULL ENABLE,  
    "EMAIL" VARCHAR2(40) NOT NULL ENABLE,  
    CONSTRAINT "CLIENTES_PK" PRIMARY KEY ("CPF")  
    USING INDEX ENABLE  
)
```

2. Tabela Projeto

```
CREATE TABLE "PROJETOS"  
(  
    "COD" NUMBER NOT NULL ENABLE,  
    "CLIENTE_CPF" NUMBER NOT NULL ENABLE,  
    "DATA_INICIO" DATE,  
    "DATA_FIM" DATE,  
    "VALOR" NUMBER,  
    CONSTRAINT "PROJETOS_PK" PRIMARY KEY ("COD")  
    USING INDEX ENABLE  
)  
/  
ALTER TABLE "PROJETOS" ADD CONSTRAINT "CPF_FK" FOREIGN KEY ("CLIENTE_CPF")  
    REFERENCES "CLIENTES" ("CPF") ON DELETE CASCADE ENABLE  
/  
  
CREATE OR REPLACE EDITIONABLE TRIGGER "BI_PROJETOS"  
    before insert on "PROJETOS"  
    for each row  
    begin  
        if :NEW."COD" is null then  
            select "PROJETOS_SEQ".nextval into :NEW."COD" from sys.dual;  
        end if;
```

end;

/

ALTER TRIGGER "BI_PROJETOS" ENABLE

/

3. Tabela Etapas

CREATE TABLE "ETAPAS"

("COD" NUMBER NOT NULL ENABLE,
"PROJETOS_COD_FK" NUMBER NOT NULL ENABLE,
"DATA_INICIO" DATE,
"DATA_FIM" DATE,
CONSTRAINT "ETAPAS_PK" PRIMARY KEY ("COD")

USING INDEX ENABLE

)

/

ALTER TABLE "ETAPAS" ADD CONSTRAINT "PROJETOS_COD FK" FOREIGN KEY
("PROJETOS_COD_FK")

REFERENCES "PROJETOS" ("COD") ON DELETE CASCADE ENABLE

/

CREATE OR REPLACE EDITIONABLE TRIGGER "BI_ETAPAS"

before insert on "ETAPAS"

for each row

begin

if :NEW."COD" is null then

select "ETAPAS_SEQ".nextval into :NEW."COD" from sys.dual;

end if;

end;

/

ALTER TRIGGER "BI_ETAPAS" ENABLE

/

4. Tabela Atividades

```

CREATE TABLE "ATIVIDADES"
(
    "COD" NUMBER NOT NULL ENABLE,
    "ETAPAS_COD_FK" NUMBER NOT NULL ENABLE,
    "DATA_INICIO" DATE,
    "DATA_FIM" DATE,
    CONSTRAINT "ATIVIDADES_PK" PRIMARY KEY ("COD")
    USING INDEX ENABLE
)
/

ALTER TABLE "ATIVIDADES" ADD CONSTRAINT "ETAPAS_COD FK" FOREIGN KEY ("ETAPAS_COD_FK")
    REFERENCES "ETAPAS" ("COD") ON DELETE CASCADE ENABLE
/

CREATE OR REPLACE EDITIONABLE TRIGGER "BI_ATIVIDADES"
    before insert on "ATIVIDADES"
    for each row
    begin
        if :NEW."COD" is null then
            select "ATIVIDADES_SEQ".nextval into :NEW."COD" from sys.dual;
        end if;
    end;
/

ALTER TRIGGER "BI_ATIVIDADES" ENABLE
/

```

5. Tabela Funcionários

```

CREATE TABLE "FUNCIONARIOS"
(
    "COD" NUMBER NOT NULL ENABLE,
    "NOME" VARCHAR2(50) NOT NULL ENABLE,
    "TELEFONE" NUMBER,
    "ENDERECO" VARCHAR2(50),
    "FUNCAO" VARCHAR2(50),
    CONSTRAINT "FUNCIONARIOS_PK" PRIMARY KEY ("COD")

```

```

        USING INDEX ENABLE
    )
/

CREATE OR REPLACE EDITIONABLE TRIGGER "BI_FUNCIONARIOS"
    before insert on "FUNCIONARIOS"
    for each row
begin
    if :NEW."COD" is null then
        select "FUNCIONARIOS_SEQ".nextval into :NEW."COD" from sys.dual;
    end if;
end;

/

ALTER TRIGGER "BI_FUNCIONARIOS" ENABLE
/

```

6. Tabela alocado

```

CREATE TABLE "ALOCADO"
(
    "FUNCIONARIO_ALOCADO__FK" NUMBER NOT NULL ENABLE,
    "ATIVIDADE_FK" NUMBER NOT NULL ENABLE
)
/

ALTER TABLE "ALOCADO" ADD CONSTRAINT "ATIVIDADES_FK" FOREIGN KEY ("ATIVIDADE_FK")
    REFERENCES "ATIVIDADES" ("COD") ON DELETE CASCADE ENABLE
/

ALTER TABLE "ALOCADO" ADD CONSTRAINT "FUNCIONARIO_ALOCADO_FK" FOREIGN KEY
("FUNCIONARIO_ALOCADO__FK")
    REFERENCES "FUNCIONARIOS" ("COD") ON DELETE CASCADE ENABLE
/

```

7. Tabela gerencia

```

CREATE TABLE "GERENCIA"
(
    "PROJETOS_COD_FK" NUMBER NOT NULL ENABLE,
    "FUNCIONARIO_COD_FK" NUMBER NOT NULL ENABLE
)
/

ALTER TABLE "GERENCIA" ADD CONSTRAINT "FUNCIONARIOS_FK" FOREIGN KEY ("FUNCIONARIO_COD_FK")
REFERENCES "FUNCIONARIOS" ("COD") ON DELETE CASCADE ENABLE
/

ALTER TABLE "GERENCIA" ADD CONSTRAINT "GERENCIA_FK" FOREIGN KEY ("PROJETOS_COD_FK")
REFERENCES "PROJETOS" ("COD") ON DELETE CASCADE ENABLE
/

```

8. Tabela Cargo

```

CREATE TABLE "CARGO"
(
    "COD" NUMBER NOT NULL ENABLE,
    "NOME" VARCHAR2(20),
    "DESCRIÇÃO" VARCHAR2(100),
    CONSTRAINT "CARGO_PK" PRIMARY KEY ("COD")
    USING INDEX ENABLE
)
/

CREATE OR REPLACE EDITIONABLE TRIGGER "BI_CARGO"
before insert on "CARGO"
for each row
begin
    if :NEW."COD" is null then
        select "CARGO_SEQ1".nextval into :NEW."COD" from sys.dual;
    end if;
end;
/

ALTER TRIGGER "BI_CARGO" ENABLE
/

```

9. Tabela Cargo Funcionário

```
CREATE TABLE "CARGO_FUNCIONARIO"
(
    "COD_CARGO_FK" NUMBER NOT NULL ENABLE,
    "COD_FUNCI_FK" NUMBER NOT NULL ENABLE,
    "DATA_INICIO" DATE,
    "DATA_FIM" DATE
)
/

ALTER TABLE "CARGO_FUNCIONARIO" ADD CONSTRAINT "CARGO_FUNCI_FK" FOREIGN KEY
("COD_FUNCI_FK")
    REFERENCES "FUNCIONARIOS" ("COD") ON DELETE CASCADE ENABLE
/

ALTER TABLE "CARGO_FUNCIONARIO" ADD CONSTRAINT "COD_CARGO FK" FOREIGN KEY
("COD_CARGO_FK")
    REFERENCES "CARGO" ("COD") ON DELETE CASCADE ENABLE
/
```