



---

MIEEC | Sistemas de Informação e Bases de Dados – 2014/2015

# **GESTbook®**

Relatório de Arquitetura

---

**André Filipe Pinto Coelho – 201107958**

**António Bastos Pintor - 201101929**

**Hugo Manuel Carvalho Fonseca - 201109210**

**Novembro 28, 2014**

## 1. Introdução

Este projeto será desenvolvido no âmbito da unidade curricular de Sistemas de Informação e Bases de Dados, lecionada no 4.º ano de Engenharia Eletrotécnica e de Computadores (MIEEC), ramo de Telecomunicações, Eletrónica e Computadores, na Faculdade de Engenharia da Universidade do Porto e tem como objetivo principal o desenvolvimento de uma aplicação que apoie os docentes e investigadores da Universidade do Porto a gerirem os seus centros de custos.

Em relação à aplicação, optámos por renomeá-la de “GESTbook” por forma de ir ao encontro do que se faz a nível empresarial, mas também tornar a aplicação original, distinguindo-a das restantes. Ela é também uma excelente oportunidade de negócio, uma vez que ainda não existe no mercado outra aplicação com objetivos idênticos.

A aplicação irá recorrer a bases de dados para fazer a gestão financeira dos centros de custos a que cada docente ou investigador estejam associados/responsabilizados. Assim, se um docente ou investigador pretender fazer uma determinada operação, o centro de custos analisará e registará essa operação. Por exemplo, se o funcionário quiser realizar uma despesa, terá que, inicialmente, pedir uma cabimentação, a qual será aprovada posteriormente pelo centro de custos respetivo.

Na implementação da aplicação serão usados HTML CSS, tecnologias orientadas para base de dados, nomeadamente, *PostgreSQL* e, para a lógica de negócio, *PHP*.

Em relação à estrutura do relatório, este encontra-se dividido em várias secções:

- a. Modelo Relacional, o qual resulta da transformação do modelo da Entidade-Associação. Além disso, é também incluído o modelo Entidade-Associação após correção de erros, sendo esta a versão a implementar no projeto;
- b. Visão geral, na qual é descrita a divisão em módulos da aplicação, bem como grupo de privilégios de cada um deles;
- c. Interface com o Utilizador, onde são representados os diagramas de arquitetura para cada um dos módulos, como também as interfaces existentes num dos módulos;
- d. Bibliografia, na qual estão as várias fontes usadas no documento;
- e. Anexos A e B, nos quais se encontra inserido o código SQL de criação da base de dados e respetivo povoamento, inserindo os dados atuais, e normalização do modelo relacional, respetivamente.

## 2. Modelo Relacional

De seguida, será apresentada a transformação obtida do modelo Entidade-Associação para o modelo Relacional, o qual será um auxiliar bastante útil na criação do script SQL responsável pela nossa base de dados.

**Funcionario** [user\_ID || NIF | nome | categoria | email | tipo | telefone | morada]

**Centro\_Custos** [centro\_custos\_ID || tipo\_custo | nome | periodo | descrição | valor\_orcamentado | nome\_curto | #Responsavel -> Funcionario]

**Operacao** [operacao\_ID || tipo\_movimento | descritivo | tipo\_suporte | montante | data | #Suporta -> Funcionario | #Beneficia -> Funcionario | #Requer -> Cabimentacao]

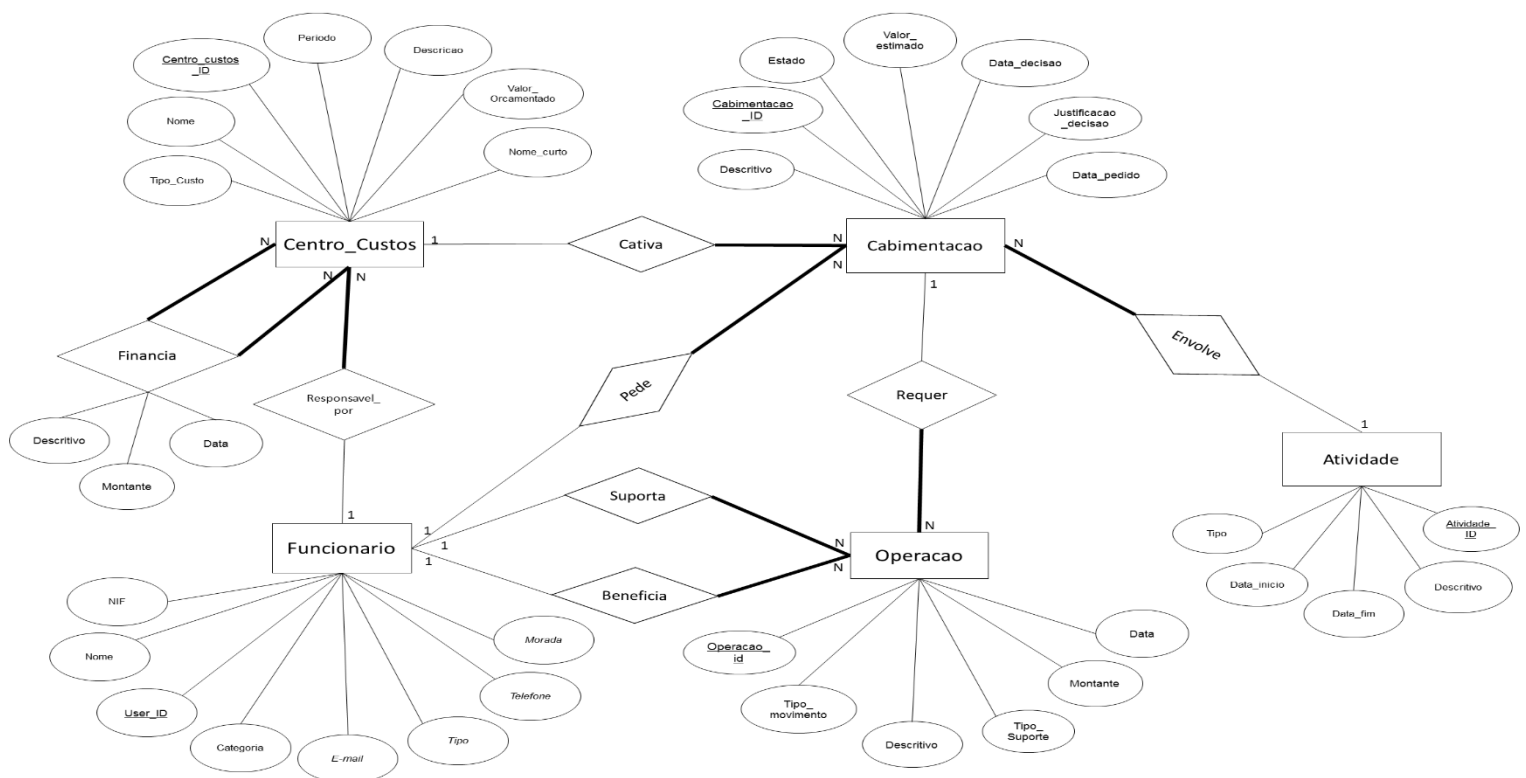
**Cabimentacao** [cabimentacao\_ID || descritivo | estado | valor\_estimado | data\_decisao | justificacao\_decisao | data\_pedido | #Pede -> Funcionario | #Cativa -> Centro\_Custos | #Envolve -> Atividade]

**Atividade** [atividade\_ID || tipo | data\_inicio | data\_fim | descritivo]

**Financia** [#Financiador -> Centro\_custos | #Financiado -> Centro\_custos || montante | data | descritivo]

## 3. Modelo Entidade-Associação

O modelo Entidade-Associação abaixo apresentado será usado na transformação para o modelo Relacional, bem como na construção da base de dados e na forma como os módulos se encontrarão divididos.



## 4. Visão geral

### 4.1. Grupos de Utilizadores

#	Nome	Descrição
<b>V</b>	Visitante	Utilizador não autenticado nem registado.
<b>REG</b>	Registado	Utilizador registado, mas que ainda não efetuou <i>Login</i> .
<b>DOC</b>	Docente	Utilizador registado e com Login efetuado pertencente ao tipo “Docente”.
<b>INV</b>	Investigador	Utilizador registado e com Login efetuado pertencente ao tipo “Investigador”.
<b>ADM</b>	Pessoal Administrativo	Utilizador registado e com Login efetuado, com permissões específicas (ex. aceitar/rejeitar cabimentações).
<b>ADMIN</b>	Administrador	Utilizador registado e com Login efetuado, com permissões específicas (ex. apagar utilizadores)

### 4.2. Módulos

A nossa aplicação irá se dividir em quatro módulos, nomeadamente: Principal, Gestão Financeira, Centro de Custos e Definições. Na secção da Interface do Utilizador será apresentado para cada um dos módulos o respetivo diagrama de arquitetura, acompanhado da respetiva de interface de utilização.

#### 4.2.1. Módulo Principal

Neste módulo será incluída a página principal da aplicação, apresentando duas interfaces distintas conforme o utilizador esteja ou não registado. Assim, inicialmente, será apresentado ao utilizador um formulário para registo ou um formulário para efetuar *login*. Após o registo estar efetuado, será apresentado novamente o mesmo módulo, devendo agora o utilizador efetuar o login na sua conta com os dados inseridos. A partir daí, será então apresentada uma nova interface, a qual conterá ligações de acesso à página de entrada dos módulos a que o utilizador tem acesso, conforme as permissões estabelecidas. Para além disso, será também apresentada uma ligação de *logout*.

#### 4.2.2. Módulo de Gestão Financeira

Neste módulo serão apresentadas as várias atividades, cada uma com as cabimentações respetivas e estas irão incluir as respetivas operações. Assim, inicialmente, será uma apresentada ao utilizador uma interface que apresentará a listagem das atividades existentes, contendo ainda ligações que permite eliminar, editar e criar uma atividade (variando conforme os privilégios do respetivo utilizador), mas também em cada atividade possui uma ligação que avançará para uma outra interface onde contém as cabimentações relativas à atividade selecionada. Na interface das cabimentações serão listadas as cabimentações pertencentes à atividade selecionada, possuindo ainda ligações para editar, eliminar ou pedir uma nova cabimentação, como também uma ligação em que cada cabimentação que permitirá avançar para uma nova interface com os detalhes relativos à cabimentação

selecionada (operações respetivas, descritivo, identificação da cabimentação, entre outros). E, por fim, nesta última interface, serão listadas as informações relativas à operação selecionada.

### **4.2.3. Módulo do Centro de Custos**

Neste módulo serão listados os vários centros de custos existentes, quer aqueles pelos quais utilizador é responsável, quer os restantes centros de custos de outros utilizadores, permitindo ainda operações como criar, editar ou apagar um determinado centro de custos (dependendo, claro, dos privilégios que o utilizador possua). Para além da interface onde serão listados os centros de custos existentes na base de dados e respetivas ligações que permitirão edição, criação ou eliminação de centros, existirá uma outra interface que servirá para, quando selecionado um determinado centro de custos, possa-se fazer pedidos de empréstimo a outros centros de custos (pedidos esses que terão de ser aceites pelos responsáveis do centro de custos destinado). Para se poder tratar das transferências entre centros de custos, será representada numa interface a listagem dos pedidos de empréstimos pendentes (pedidos efetuados e recebidos), podendo nesta interface o utilizador aceitar, editar ou eliminar pedidos de empréstimos. Para além disso, e somente para determinados utilizadores com privilégios, existirá uma interface onde serão listadas todas as cabimentações que aguardam veredito e que poderá ser alterado por este grupo de utilizadores.

### **4.2.4. Módulo das Definições da Conta**

Por último, este módulo será responsável apenas pela alteração de definições pessoais relativamente à conta pessoal do utilizador. Assim, neste módulo poderemos alterar a palavra-passe atual, nome de utilizador, *email*, categoria, tipo, telefone e morada. De notar que este módulo apenas conterá uma interface, na qual poderá se fazer as respetivas alterações.

## **5. Interface com o utilizador**

### **5.1. Descrição Geral dos módulos**

As ligações entre páginas *Web* estão representadas por “seta a cheio” e as invocações entre páginas *Web* estão representadas por “seta a tracejado”. O tipo de cada página está presente no canto inferior direito, sendo:

- **R** – Listagens (*Reports*). Estas páginas contêm informação, geralmente, na forma tabular (por exemplo a lista de todas as cabimentações pedidas por um docente ou investigador) e não permitem a introdução de dados pelo utilizador;
- **A** – Ação (*Action*). Estas páginas são internas à “lógica do negócio” e, normalmente, invocadas a partir da submissão de um formulário;
- **F** – Formulário (*Form*). Estas páginas que contêm informação, geralmente, na forma de *Label* e valor. Permitem a introdução de dados pelos utilizadores.

Os “grupos de privilégio” que podem aceder a cada página encontram-se explicitados no canto inferior esquerdo, os quais se passa a enumerar:

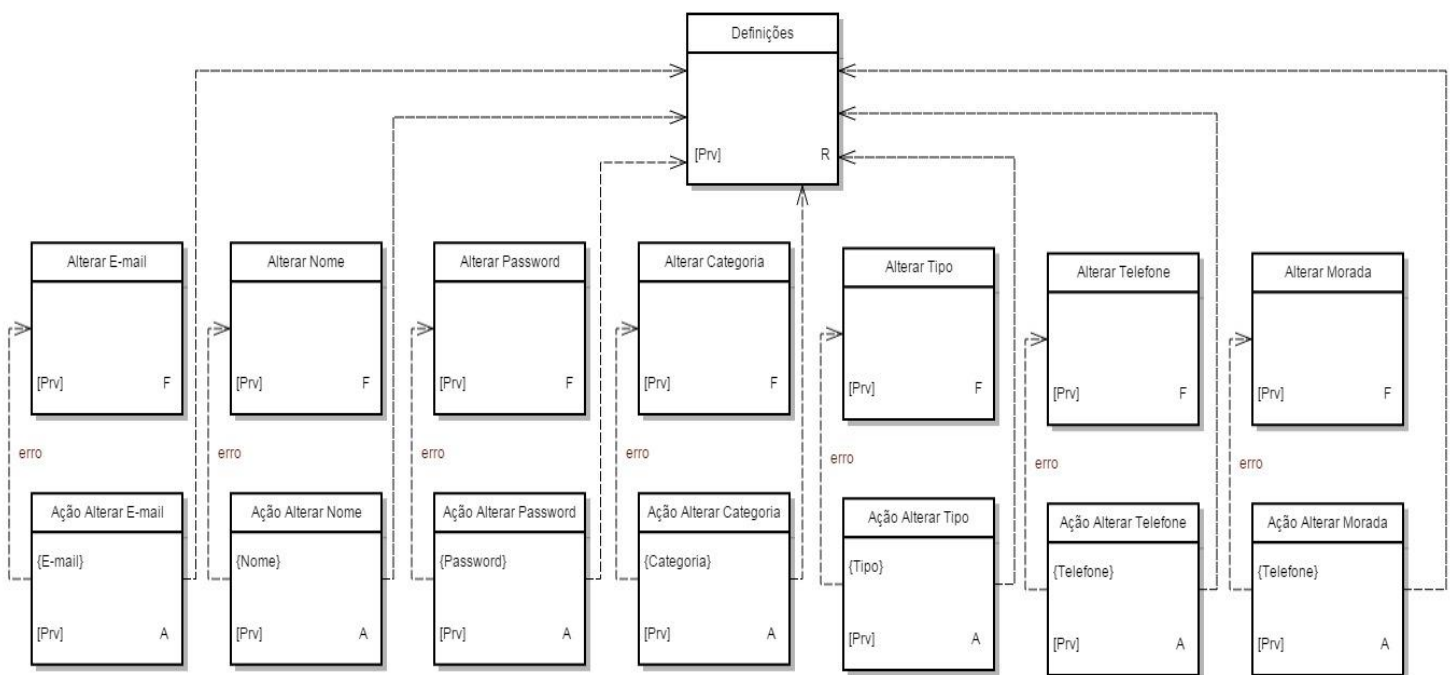
- **[ - ]** – Todos os tipos de funcionários (administrativos, docentes, investigadores);
- **[Prv]** – Administrativos e aqueles cuja informação apresentada na página diz respeito ou é da sua responsabilidade;
- **[Adm]** – Administrativos.
- **[V]** - Visitante

As páginas com comentário (retângulo com canto superior direito “dobrado”), dizem respeito a páginas de outros módulos, mas com ligações ao módulo que o diagrama especifica.

Para representar os parâmetros de entrada das páginas, são usadas chavetas ({dados}).

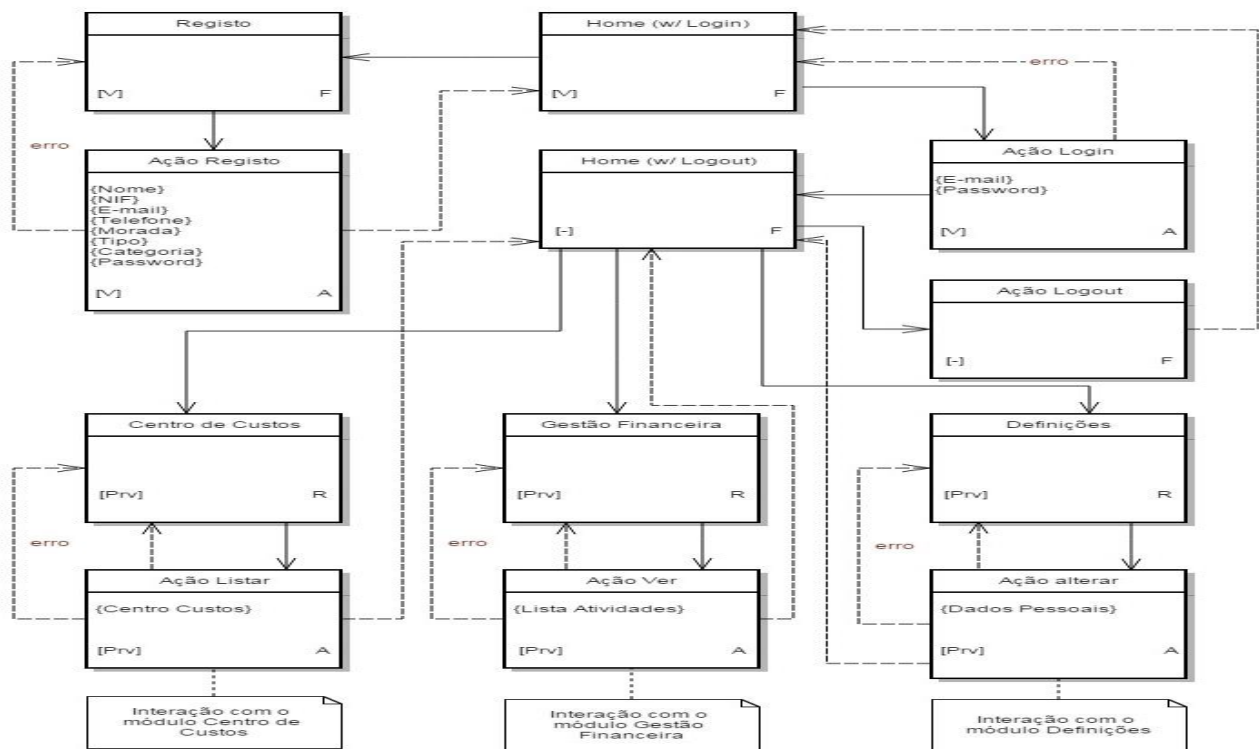
## 5.2. Diagramas de Arquitetura de cada Módulo

### 5.2.1. Módulo “Definições da Conta”

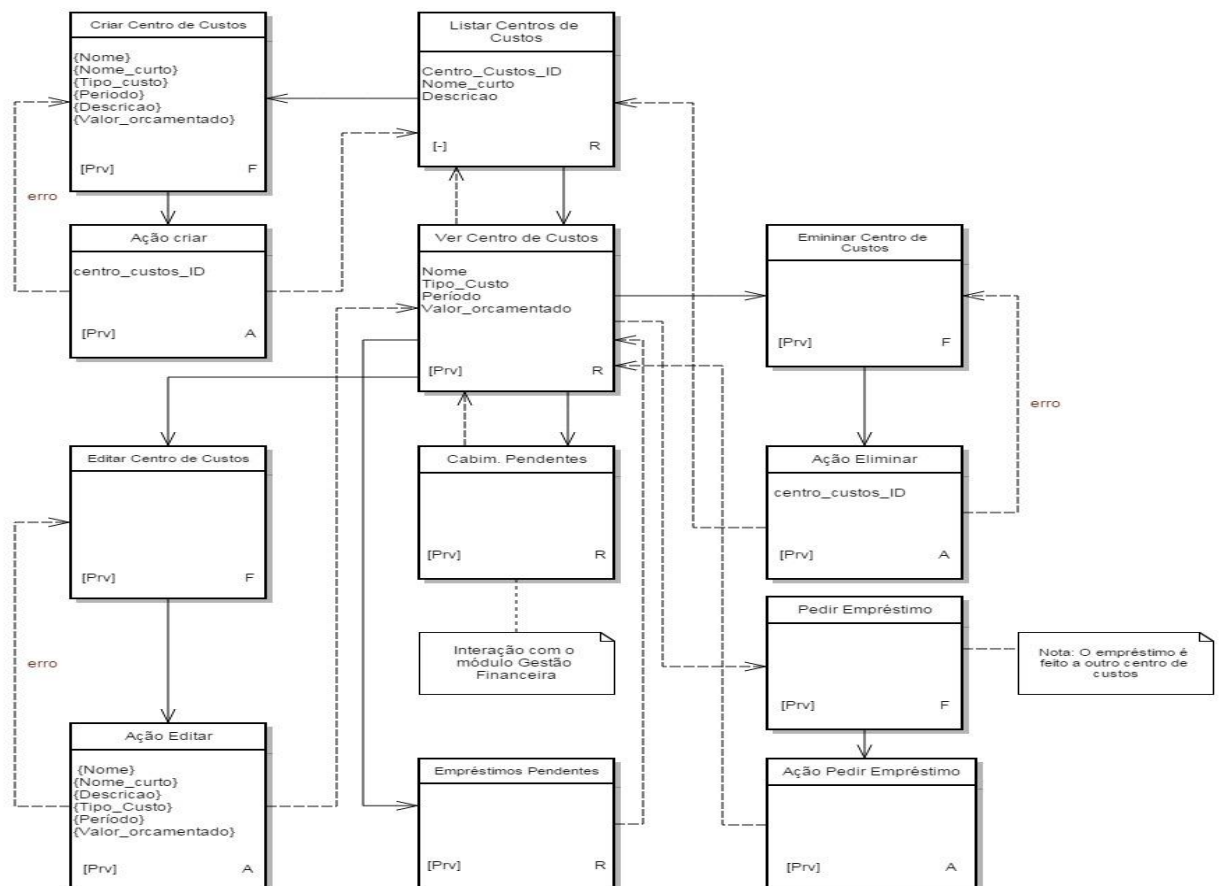


### 5.2.2.

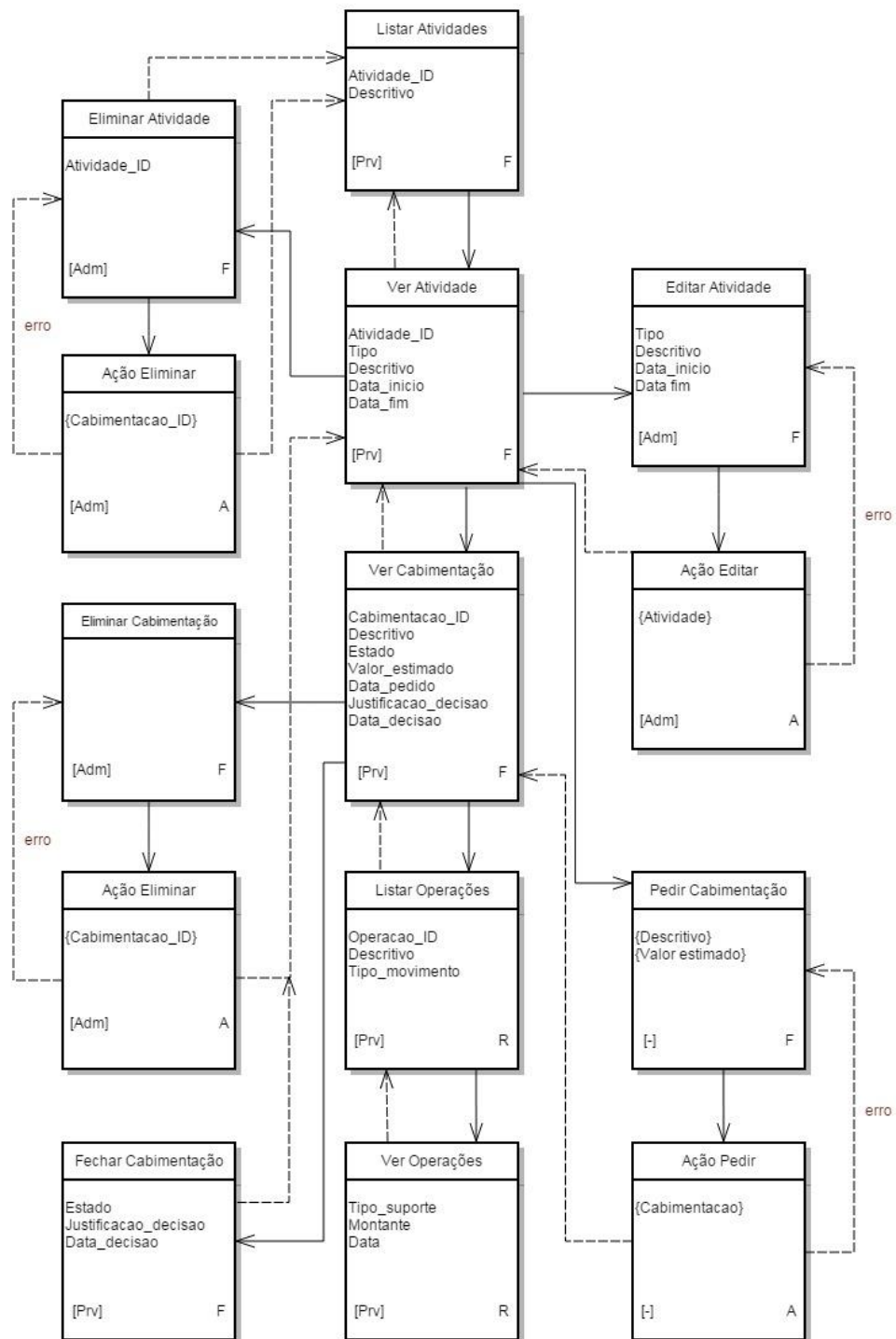
### 5.2.3. Módulo “Principal”



#### **5.2.4. Módulo “Centro de Custos”**



### 5.2.5. Módulo “Gestão Financeira”





### 5.3. Interfaces do Módulo “Centro de Custos”

Note-se que, na construção das interfaces, procurou-se assimilar às do “MyNews”, por forma a podermos usar o seu código e adaptá-lo ao nosso projeto.

GESTbook®

http://fe.up.pt/~ee11141/gestbook

GESTbook :: Gestão de Centros de Custos

MENU

LISTAR CENTROS DE CUSTOS

# ID	Nome Curto	Descrição
1	CC_Est	Responsável pela gestão da educação
2	CC_Proj	Responsável pela gestão de projetos
3	CC_Eve	Responsável pela gestão de eventos
4	CC_Pess	Responsável pela gestão pessoal

[Inserir]

Carlos Soares (Docente) Logout

Copyright © 2014-2015, GESTbook. Todos os direitos reservados

Figura 1 - Interface Inicial do Módulo (Listagem dos CCs existentes)

GESTbook®

http://fe.up.pt/~ee11141/gestbook

GESTbook :: Gestão de Centros de Custos

MENU

criar CENTRO DE CUSTOS

Nome: Centro de Custos - Nova Base de Dados SIGARRA

Nome curto: CC\_SIG

Período de vida: 3 Meses

Valor Orçamentado: €1000,00

Descrição: Atualização da Base de Dados do SIGARRA

Tipo de Custo: Projetos

[Criar] [Cancelar]

Carlos Soares (Docente) Logout

Copyright © 2014-2015, GESTbook. Todos os direitos reservados

Figura 2 - Criação de Novo CC

GESTbook®

http://fe.up.pt/~ee11141/gestbook

GESTbook :: Gestão de Centros de Custos

MENU

EDITAR CENTRO DE CUSTOS

Nome: Centro de Custos - Nova Base de Dados SIGARRA

Nome curto: CC\_SIG

Período de vida: 5 Meses

Valor Orçamentado: €1500,00

Descrição: Atualização da Base de Dados do SIGARRA

Tipo de Custo: Projetos

[Editar] [Cancelar]

Carlos Soares (Docente) Logout

Copyright © 2014-2015, GESTbook. Todos os direitos reservados

Figura 3 - Edição de CC existente

GESTbook®

http://fe.up.pt/~ee11141/gestbook

GESTbook :: Gestão de Centros de Custos

MENU

VER CENTRO DE CUSTOS

ID: 1

Nome: Centro de Custos - Aulas

Nome curto: CC\_Est

Período de vida: 5 Meses

Valor Orçamentado: €20.000,00

Descrição: Gestão financeira das UCs do 1.º semestre

Tipo de Custo: Estudantes

[Editar] [Eliminar]

Carlos Soares (Docente) Logout

Copyright © 2014-2015, GESTbook. Todos os direitos reservados

Figura 4 - Visualizar Informações sobre o CC selecionado

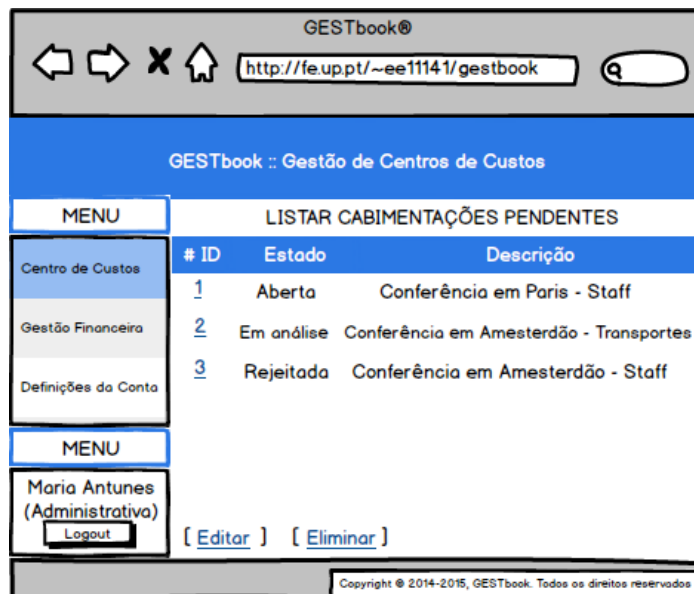


Figura 5 – Visualizar Cabimentações existentes (apenas administrativos conseguem aceder a esta interface)

## **6. Resumo e conclusões**

O relatório de arquitetura que aqui se apresenta relata os últimos passos do “desenho” da aplicação proposta. Constitui, portanto, o último patamar antes da implementação da aplicação.

De uma forma geral, foram cumpridos os pontos detalhados na secção de introdução. Contudo, devemos salientar as dificuldades que a realização destas etapas de projeto acarretaram. A arquitetura de uma aplicação deste tipo, mesmo que a sua complexidade não seja especialmente elevada, suscita inevitavelmente dúvidas, pelo que foi preciso tomar decisões importantes e difíceis, de modo a cumprir os requisitos do “cliente”.

Além da arquitetura, de destacar também o passo da criação da base de dados, cujo código SQL também figura neste relatório e que teve por base os modelos entidade-associação e o conseqüente modelo relacional também presente. Foi ainda posteriormente feito o “povoamento” da base de dados, de modo a despistar erros.

Resta perspetivar a etapa que se segue, ou seja, a implementação da aplicação. Para tal, usar-se-á uma codificação em linguagem PHP, de modo a criar um conjunto de páginas *Web* dinâmicas, que, em conjunto com o sistema de gestão de base de dados PostgreSQL, constituirá o produto final.

## **7. Bibliografia**

Ao longo da elaboração do relatório, foram consultadas diversas fontes de informação. De seguida, serão enumeradas essas fontes:

- [http://pt.wikipedia.org/wiki/Normaliza%C3%A7%C3%A3o\\_de\\_dados#Terceira Forma Normal](http://pt.wikipedia.org/wiki/Normaliza%C3%A7%C3%A3o_de_dados#Terceira_Forma_Normal) [consultado a 27/11/2014]
- [http://www.decom.ufop.br/guilherme/BCC321/geral/bd1\\_normalizacao.pdf](http://www.decom.ufop.br/guilherme/BCC321/geral/bd1_normalizacao.pdf) [consultado a 27/11/2014]
- <http://www.luis.blog.br/terceira-forma-normal-3fn-normalizacao-de-dados.aspx> [consultado a 27/11/2014]

## **ANEXO A**

```
--
-- PostgreSQL database dump
--
--
-- SCHEMA: GESTbook
--
DROP SCHEMA IF EXISTS GESTbook CASCADE;

CREATE SCHEMA GESTbook;

SET client_encoding = 'UTF8';
SET search_path = GESTbook;

--
-- TABLE: funcionário
--
CREATE TABLE funcionarios (
  user_id INTEGER PRIMARY KEY,
  nome CHARACTER VARYING(128) NOT NULL,
  categoria CHARACTER VARYING(64) NOT NULL,
  tipo CHARACTER VARYING(24) NOT NULL,
  telefone CHARACTER VARYING(24) UNIQUE,
  nif CHARACTER VARYING(9) UNIQUE,
  morada CHARACTER VARYING(512) UNIQUE,
  email CHARACTER VARYING(64) UNIQUE
);
--ALTER TABLE ONLY funcionarios;
COMMENT ON TABLE funcionarios IS 'Informacao acerca dos funcionários';
COMMENT ON COLUMN funcionarios.user_id IS 'Número de Identificação';
COMMENT ON COLUMN funcionarios.nome IS 'Nome';
COMMENT ON COLUMN funcionarios.telefone IS 'Número de Telefone';
COMMENT ON COLUMN funcionarios.nif IS 'NIF';
COMMENT ON COLUMN funcionarios.tipo IS 'Tipo';
COMMENT ON COLUMN funcionarios.categoria IS 'Categoria';
COMMENT ON COLUMN funcionarios.morada IS 'Morada';
COMMENT ON COLUMN funcionarios.email IS 'Email';
--
-- TABLE: CentroCustos
--
CREATE TABLE CentroCustos(
  CentroCustos_ID INTEGER PRIMARY KEY,
  Nome CHARACTER VARYING(128) NOT NULL,
  Nome_curto CHARACTER VARYING(24) NOT NULL,
  Descricao CHARACTER VARYING(256) NOT NULL,
  Valor_orcamentado INTEGER NOT NULL,
  Tipo_custo CHARACTER VARYING(24) NOT NULL,
  Periodo INTEGER NOT NULL,
  Responsavel INTEGER REFERENCES funcionarios
);
--ALTER TABLE ONLY CentroCustos
```

```
COMMENT ON TABLE CentroCustos IS 'Informacao acerca dos pedidos
cabimentação';
COMMENT ON COLUMN CentroCustos.CentroCustos_ID IS 'Numero de
identificação do centro';
COMMENT ON COLUMN CentroCustos.Nome IS 'Nome';
COMMENT ON COLUMN CentroCustos.Nome_Curto IS 'Nome Curto';
COMMENT ON COLUMN CentroCustos.Descricao IS 'Descrição';
COMMENT ON COLUMN CentroCustos.Valor_orcamentado IS 'Valor
Orçamentado';
COMMENT ON COLUMN CentroCustos.Tipo_custo IS 'Tipo de custo';
COMMENT ON COLUMN CentroCustos.Periodo IS 'Período';
COMMENT ON COLUMN CentroCustos.Responsavel IS 'Pessoa Responsável
pelo Centro de Custos';

--
-- TABLE: Atividade
--
CREATE TABLE atividade (
Atividade_ID INTEGER PRIMARY KEY,
Tipo CHARACTER VARYING(30) NOT NULL,
Descritivo CHARACTER VARYING(256) NOT NULL,
Data_inicio DATE NOT NULL,
Data_fim DATE NOT NULL
);
--ALTER TABLE ONLY atividade
COMMENT ON TABLE atividade IS 'Informação acerca das Atividades';
COMMENT ON COLUMN atividade.Atividade_ID IS 'Número de Identificação da
Atividade';
COMMENT ON COLUMN atividade.Tipo IS 'Tipo';
COMMENT ON COLUMN atividade.descritivo IS 'Descrição' ;
COMMENT ON COLUMN atividade.Data_inicio IS 'Data de início' ;
COMMENT ON COLUMN atividade.Data_fim IS 'Data de fim' ;

--
-- TABLE: cabimentação
--
CREATE TABLE cabimentacao (
Cabimentacao_ID INTEGER PRIMARY KEY,
Estado CHARACTER VARYING(30) NOT NULL,
Descritivo CHARACTER VARYING(64) NOT NULL,
Valor_Estimado INTEGER NOT NULL,
Data_Pedido DATE NOT NULL,
Data_Decisao DATE NOT NULL,
Justificacao_Decisao CHARACTER VARYING(512) NOT NULL,
Cativa Integer REFERENCES CentroCustos,
Pede Integer REFERENCES funcionarios,
Envolve INTEGER REFERENCES atividade NOT NULL
);
--ALTER TABLE ONLY cabimentacao
COMMENT ON TABLE cabimentacao IS 'Informação acerca dos orçamentos
pedidos pelos funcionários';
```

```
COMMENT ON COLUMN cabimentacao.Cabimentacao_ID IS 'Número de
Identificação da Cabimentação';
COMMENT ON COLUMN cabimentacao.Estado IS 'Estado';
COMMENT ON COLUMN cabimentacao.Descritivo IS 'Descrição' ;
COMMENT ON COLUMN cabimentacao.Valor_Estimado IS 'Valor Estimado';
COMMENT ON COLUMN cabimentacao.Data_Pedido IS 'Data de Pedido';
COMMENT ON COLUMN cabimentacao.Data_Decisao IS 'Data de Decisão';
COMMENT ON COLUMN cabimentacao.Justificacao_Decisao IS 'Justificação da
Decisão';
COMMENT ON COLUMN cabimentacao.Cativa IS 'Centro de Custos a que
Cativa';
COMMENT ON COLUMN cabimentacao.Pede IS 'Funcionário que pediu';
COMMENT ON COLUMN cabimentacao.Envolve IS 'Atividade a que pertence';

--
-- TABLE: Operação
--
CREATE TABLE operacao(
Operacao_ID INTEGER PRIMARY KEY,
Tipo_Movimento CHARACTER VARYING(64) NOT NULL,
Tipo_Suporte CHARACTER VARYING(64) NOT NULL,
Descritivo CHARACTER VARYING(64) NOT NULL,
Montante INTEGER NOT NULL,
Data date NOT NULL,
Beneficiante INTEGER REFERENCES funcionarios,
Requer INTEGER REFERENCES cabimentacao,
Suporta INTEGER REFERENCES funcionarios
);
--ALTER TABLE ONLY operacao
COMMENT ON TABLE operacao IS 'Informacao acerca das operações envolvidas
por uma cabimentação';
COMMENT ON COLUMN operacao.Operacao_ID IS 'Numero de identificação da
Operação';
COMMENT ON COLUMN operacao.Tipo_Movimento IS 'Tipo de movimento';
COMMENT ON COLUMN operacao.Tipo_Suporte IS 'Tipo de Suporte';
COMMENT ON COLUMN operacao.Descritivo IS 'Descrição';
COMMENT ON COLUMN operacao.Montante IS 'Montante';
COMMENT ON COLUMN operacao.Beneficiante IS 'Benefeciante';
COMMENT ON COLUMN operacao.Data IS 'Data efectuada';
COMMENT ON COLUMN operacao.Suporta IS 'Entidade que suporta o custo da
Operação';
COMMENT ON COLUMN operacao.Requer IS 'Cabimentação à qual foi
requerida';
```

```
--
-- TABLE: Financia
--
CREATE TABLE Financia(
```

```
Financiador INTEGER REFERENCES CentroCustos,
Financiado INTEGER REFERENCES CentroCustos,
Descritivo CHARACTER VARYING(128) NOT NULL,
Montante INTEGER,
Data DATE NOT NULL,
PRIMARY KEY (Financiador, Financiado)
);
--ALTER TABLE ONLY Financia

COMMENT ON TABLE Financia IS 'Informação a cerca dos financiamentos entre
Centros de Custo';
COMMENT ON COLUMN Financia.Financiador IS 'Numero de identificação do
centro que vai financiar';
COMMENT ON COLUMN Financia.Financiado IS 'Numero de identificação do
centro que vai ser financiado';
COMMENT ON COLUMN Financia.Descritivo IS 'Descrição';
COMMENT ON COLUMN Financia.Montante IS 'Montante';
COMMENT ON COLUMN Financia.Data IS 'Data de Financiamento ' ;
--
-- Data for Funcionarios; Type: TABLE DATA; Schema: Gestbook;
--
INSERT INTO funcionarios (User_ID, Nome, nif, Categoria, Tipo, Morada,
Telefone, Email) VALUES (1234, 'António Pintor', '213546987', 'Técnico Superior',
'Administrativo', 'Rua D. Afonso II', '915869921', 'ee11098@fe.up.pt');

INSERT INTO funcionarios (User_ID, Nome, nif, Categoria, Tipo, Morada,
Telefone, Email) VALUES (1235, 'André Coelho', '321456978', 'Engenharia',
'Investigador', 'Rua 1º de Maio', '914568265', 'ee11141@fe.up.pt');

INSERT INTO funcionarios (User_ID, Nome, nif, Categoria, Tipo, Morada,
Telefone, Email) VALUES (1236, 'Hugo Fonseca', '369852147', 'Convidado', '
Docente', 'Rua D. Afonso Henriques', '915684259', 'ee11178@fe.up.pt');
--
-- Data for CentrosCusto
--
INSERT INTO CentroCustos (CentroCustos_ID, Nome, Nome_curto, Descricao,
Valor_orcamentado,Tipo_custo, Período, Responsavel) VALUES (555, 'Centro de
Custo de Departamento Electrotécnica', 'CCDE ', 'Centro responsável pela gestão
de eventos a serem realizados pelo Departamento de Eletrotécnica',100000,
'Eventos', 1, 1234);

INSERT INTO CentroCustos (CentroCustos_ID, Nome, Nome_curto, Descricao,
Valor_orcamentado,Tipo_custo, Período, Responsavel) VALUES (424, 'Centro de
Custo de Informática', 'CCI ', ' Centro responsável pela gestão de gastos pessoais
para custos relacionados com a informática',50000, 'Pessoais', 1, 1235);

INSERT INTO CentroCustos (CentroCustos_ID, Nome, Nome_curto, Descricao,
Valor_orcamentado,Tipo_custo, Período, Responsavel) VALUES (951, 'Centro de
Custo de Departamento da Física', 'CCDF ', ' Centro responsável pela gestão de
custos relacionados com Projectos desenvolvidos pelo Departamento da Física',
75000, 'Projetos', 1, 1236);
--
```

-- Data for atividades

--

```
INSERT INTO atividade (Atividade_ID, Descritivo, Tipo, Data_inicio, Data_fim)
VALUES (654, 'Conferência sobre Protocolos de Redes e Internet', 'eventos',
'10/2/2014', '10/7/2014');
```

```
INSERT INTO atividade (Atividade_ID, Descritivo, Tipo, Data_inicio, Data_fim)
VALUES (789, 'Renovação de Equipamentos do departamento', 'estudantes',
'5/7/2014', '5/20/2014');
```

```
INSERT INTO atividade (Atividade_ID, Descritivo, Tipo, Data_inicio, Data_fim)
VALUES (213, 'Estudo para o descobrimento de novos buracos negros nas
galáxias mais próximas', 'projetos', '10/6/2014', '11/3/2014');
```

-- end

--

-- Data for cabimentacao

--

```
INSERT INTO cabimentacao (Cabimentacao_ID, Descritivo, Estado, Data_Pedido,
Data_Decisao, Valor_Estimado, Justificacao_Decisao, Cativa, Pedir, Envolve)
VALUES (75, 'Catering', 'Aberta', '8/1/2014', '10/1/2014', 10000, 'Boa adesão ao
evento', 555, 1234, 654);
```

```
INSERT INTO cabimentacao (Cabimentacao_ID, Descritivo, Estado, Data_Pedido,
Data_Decisao, Valor_Estimado, Justificacao_Decisao, Cativa, Pedir, Envolve)
VALUES (100, 'Material Informático', 'Aberta', '4/10/2014', '5/6/2014', 25000,
'Muito boa ideia', 424, 1235, 789);
```

```
INSERT INTO cabimentacao (Cabimentacao_ID, Descritivo, Estado, Data_Pedido,
Data_Decisao, Valor_Estimado, Justificacao_Decisao, Cativa, Pedir, Envolve)
VALUES (946, 'Projecto sobre Buracos Negros', 'Aberta', '9/23/2014',
'10/5/2014', 5000, 'Grande Equipa de Investigadores', 951, 1236, 213);
```

--

-- Data for Financia

--

```
INSERT INTO Financia (Financiador, Financiado, Descritivo, Montante, Data)
VALUES (555, 424, 'Empréstimo para ajuda a obtenção de novo material', 2000,
'4/25/2014');
```

--

-- Data for operacao

--

```
INSERT INTO operacao (Operacao_ID, Descritivo, Tipo_Movimento,
Tipo_Suporte, Beneficiante, Montante, Data, Suporta, Requer) VALUES (563,
'Contratar Pessoal', 'Despesa', 'Pessoal', 1236, 2000, '10/2/2014', 1234, 75);
INSERT INTO operacao (Operacao_ID, Descritivo, Tipo_Movimento,
Tipo_Suporte, Beneficiante, Montante, Data, Suporta, Requer) VALUES (776,
'Computadores', 'Despesa', 'Pessoal', 1234, 10000, '5/7/2014', 1235, 100);
INSERT INTO operacao (Operacao_ID, Descritivo, Tipo_Movimento,
Tipo_Suporte, Beneficiante, Montante, Data, Suporta, Requer) VALUES (472,
'Contratar Pessoal', 'Despesa', 'Pessoal', 1235, 4000, '10/6/2014', 1236, 946);
```

## **Anexo B**



O modelo relacional desenvolvido para o projeto da aplicação descrito neste relatório, segue uma normalização de dado, a Terceira Forma Normal, que permite uma maior independência dos dados, como também reduzir redundâncias e anomalias destes.

Esta normalização define-se como se alguma relação esteja na segunda forma normal onde cada atributo que não é chave primária, não possua dependências com qualquer outra chave candidata desta mesma relação.

Exemplo:

**Operacao** [operacao\_ID || tipo\_movimento | descritivo | tipo\_suporte | montante | data | #Suporta -> Funcionario | #Beneficia -> Funcionario | #Requer -> Cabimentacao]

**Cabimentacao** [cabimentacao\_ID || descritivo | estado | valor\_estimado | data\_decisao | justificacao\_decisao | data\_pedido | #Pede -> Funcionario | #Cativa -> Centro\_Custos | #Envolve -> Atividade]

Como podemos observar pelo exemplo na relação Operacao temos chaves não primárias não dependentes de outras chaves candidatas, pois não faria sentido colocar como os atributos das operações juntos com os da cabimentacao na mesma relação.