

Relatório Final



Instituto Politécnico de Coimbra

Instituto Superior de Engenharia de Coimbra

Departamento de Engenharia Informática e de Sistemas

Engenharia Informática - Curso Europeu

Bases de Dados

2021/2022

Cap 1 - Tema

Sistema de Voluntariado

Cap 2 - Identificação dos Autore

Autore

Nº Aluno	Prática	Nome	Email
2020130980	P6	Pedro Praça	a2020130980@isec.pt

Cap 3 - Introdução

O objetivo deste nosso projeto é criar um sistema que agilize a gestão de uma qualquer instituição de voluntariado.

Este sistema irá conter toda a informação dos doadores(particulares, anónimos, e empresas) e seus donativos (anónimos, particulares incluindo empresas), voluntários(sabendo se estão disponíveis ou indisponíveis, as suas informações pessoais, a sua função, horas investidas na organização), entidades carenciadas, caracterizadas por tipo, e os apoios que estas irão receber(a preencher, a sua data, o respetivo valor), parceiros (nome, a sua descrição e se está atualmente ativo ou inativo) e atividades(já realizadas ou que serão realizadas no futuro, assim como o seu estado, ativo ou inativo).

O projeto "Prós Outros" foi criado em 2021, a meio de uma pandemia, quando a quantidade de pessoas a necessitarem de apoios materiais e monetários aumentou exponencialmente e a maioria dos projetos existentes não tinham capacidade para ajudar todas as entidades carenciadas.

No capítulo 4, faz-se um enquadramento da relevância da proposta da criação da instituição de voluntariado, descreve-se a situação atual, os principais problemas encontrados e, por fim, descreve-se a proposta que irá permitir solucionar esses problemas e atingir o objetivo desejado.

No capítulo 5, são descritas as funcionalidades da aplicação que irá interagir com a base de dados, apresentando-se os seus módulos e detalhando as suas restrições de funcionamento.

Cap 4 - Enquadramento da Proposta

Neste capítulo faz-se um enquadramento da proposta no âmbito de uma instituição de voluntariado denominada "Prós Outros".

Na secção 4.1 descreve-se o funcionamento actual desta organização de voluntariado, apresentando-se na secção 4.2 o conjunto de problemas existentes actualmente na organização e que se pretende solucionar.

A secção 4.3 descreve a proposta de solução que permitirá solucionar os problemas existentes e satisfazer as necessidades encontradas.

Sec 4.1 - Diagnóstico da Situação Atual

Atualmente a instituição de voluntariado "Prós Outros" depara-se com problemas de organização, dada a dimensão que está a atingir neste tempo de pandemia.

O aumento do número de voluntários, de pessoas e entidades carenciadas e em virtude da necessidade de recolha de fundos e dinamização de atividades, a burocracia (registo de dados de pessoas e entidades, voluntários, etc...) tem gerado demasiado papel, tomado demasiado tempo e, em suma, causando alguma entropia no dia a dia da instituição.

Chegou-se assim à conclusão que é necessário desenvolver um sistema de informação para gerir a atividade da instituição.

Este projeto de base de dados será parte integrante deste sistema de informação.

Sec 4.2 - Problemas encontrados

Como referido anteriormente, o maior problema encontra-se na organização de todos os dados que atualmente é necessário registar, organizar, e, posteriormente listar. De seguida elencam-se com algum pormenor esses mesmos problemas.

1. Voluntários: apenas existe registo em papel. Qualquer necessidade de obter uma lista que obedeça a um qualquer critério é extremamente morosa, tornando mais complexa a distribuição de voluntários pelas atividades. Também é complicado saber quais estão e quando estão ao "serviço", sendo assim mais difícil saber quando estão disponíveis.
2. Pessoas e entidades carenciadas: também apenas existe registo em papel, sendo portanto igualmente complicado obter listagens filtradas, e gerir prioridades de apoio.
3. Donativos e peditório: é difícil e moroso obter dados em tempo real dos donativos recebidos, filtrados ou não por período, por atividades, por entidades, etc.
4. Atividades: a gestão da calendarização das atividades é complicada bem como a distribuição dos voluntários pelas atividades dinamizadas.

Com esta base de dados pretende-se colmatar as dificuldades acima referidas, agilizando assim toda a organização da instituição.

Sec 4.3 - Descrição da Solução Proposta

O sistema de informação a implementar será baseado numa arquitetura cliente/servidor constituída por uma aplicação desenvolvida para ambiente *web*, hospedado num servidor acessível por todas as delegações e funcionários da instituição a qualquer momento, mediante credenciais de acesso e privilégios consoante perfil de utilizador, evitando assim a necessidade de instalações locais. Independentemente desse facto, é possível a implementação de aplicações em arquiteturas diferentes com suporte na mesma base de dados.

A base de dados que servirá de suporte ao sistema de informação deve permitir à instituição:

- gerir (registar, alterar, imprimir dados) voluntários, pessoas/entidades carenciadas, donativos à instituição, recibos de donativos, atividades dinamizadas, quer de cariz social, quer de angariação de fundos, entidades parceiras e dados de peditórios
- gestão de peditórios: atribuição de cofres a voluntários/entidades, apuramento de verbas angariadas
- associar voluntários e/ou entidades parceiras a atividades, mediante disponibilidade dos voluntários/entidades para as datas de realização das atividades
- obter listagens respeitando determinados critérios/filtros, incluindo listagens de:
 - voluntários por distrito/concelho de residência, funções que desempenham, atividades às quais estão associados;
 - donativos recebidos por atividade e/ou por voluntário, por região, por tipo de doador e intervalo de tempo;
 - atividades realizadas, por voluntário, região, intervalo temporal, tipo de atividade;
 - recibos gerados, por intervalo temporal;

Com suporte nesta base de dados, qualquer aplicação a desenvolver permitirá, rápida e facilmente, efetuar uma gestão dinâmica e eficiente da instituição de voluntariado que a utilizará, neste caso a "Prós Outros".

Cap 5 - Funcionalidades da Aplicação

Neste capítulo são descritas as funcionalidades e restrições de implementação que devem ser tidas em consideração aquando da implementação da aplicação que utilizará esta base de dados.

Na secção 5.1 apresentam-se os requisitos e restrições a considerar aquando da implementação da aplicação. Na secção 5.2 descreve-se a estrutura da aplicação, os módulos a desenvolver e respetivas funcionalidades.

Sec 5.1 - Estrutura da Aplicação

A aplicação deve utilizar uma arquitetura cliente/servidor que irá questionar o servidor de base de dados para aceder à informação desejada. Esta aplicação será instalada em cada uma das máquinas clientes, sendo o acesso à base de dados efetuado através da utilização de *drivers* nativos do Oracle.

Para permitir futuras evoluções, esta deve estar estruturada de uma forma modular, que permita facilmente adicionar novos módulos à aplicação. A utilização de uma arquitetura modular irá permitir a implementação faseada dos vários módulos que são propostos, e a possibilidade de futuramente ser possível adicionar novos, além dos já existentes, na mesma interface comum. Esta modularidade permitirá evitar a existência de várias aplicações, com objetivos distintos, mas que necessariamente irão possuir funcionalidades redundantes.

A aplicação deve ser constituída por um conjunto de módulos que permitam manipular os dados de cada uma das operações do sistema de voluntariado. A cada operação deve corresponder um módulo da aplicação que será responsável por interagir com o utilizador e manipular os dados das tabelas associadas à operação em causa. Deve ter uma interface simples e agradável, em que o acesso a cada um dos módulos da aplicação esteja acessível através de um menu. Deve existir na interface principal um conjunto de atalhos que permita o acesso rápido aos módulos mais utilizados.

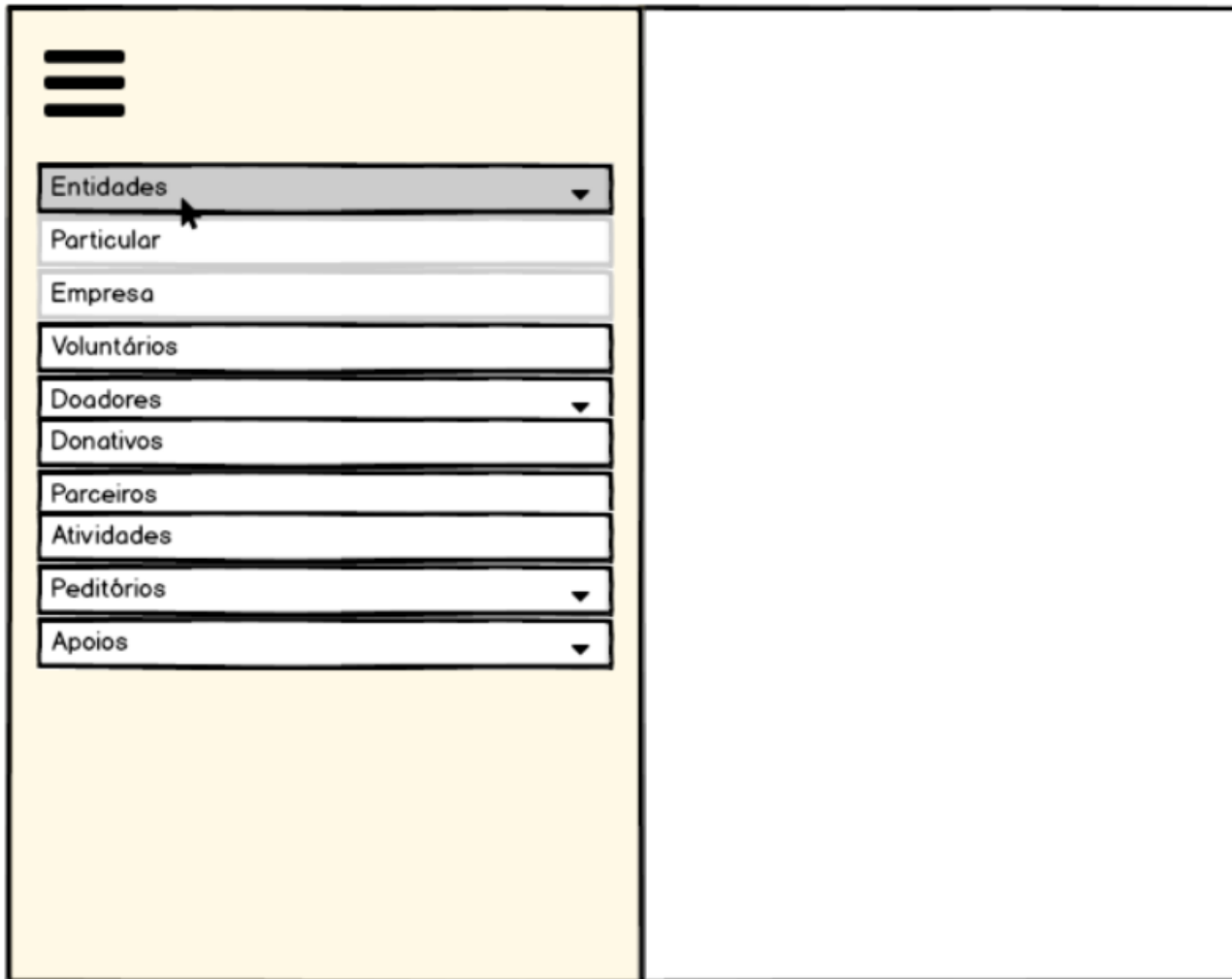
Os módulos a implementar são os seguintes:

- Gestão das entidades:
 - § Acesso a uma lista com os tipos de entidades
 - § Acessos aos diversos tipos de apoios fornecidos as mesmas (fazendo referência ao estado do mesmo)
 - § Acesso aos tipos de apoios cedidos pelas entidades (fazendo referência ao estado do mesmo)
- Gestão de uma lista de voluntários:
 - § Acesso aos peditórios e aos seus respetivos cofres
 - § Acesso a uma lista de atividades praticadas pelos mesmos e os respetivos tipos
 - § Acesso aos distritos, aos concelhos e aos códigos postais
- - Gestão dos parceiros:
 - § Acesso aos peditórios realizados pelos respetivos parceiros

- § Acesso as atividades em que colaboram e os respetivos tipos de atividades
- § Acesso aos cofres associados a cada parceiro
- § Acesso ao distrito, concelho e código postal
- Gestão dos doadores:
 - § Acesso aos tipos de entidades doadoras (que é subdividida entre doadores particulares e doadores não anónimos)
 - § Acesso a uma lista de donativos
- Gestão das atividades:
 - § Acesso aos diferentes tipos de atividades (fazendo referência ao estado do mesmo)
- Gestão dos donativos:
 - § Acesso aos tipos de donativos (fazendo referência ao estado do mesmo)
 - § Acesso a uma lista de recibos

Sec 5.2 - Módulos da aplicação e protótipos





A sidebar menu is displayed on a light yellow background. At the top left is a hamburger menu icon consisting of three horizontal black bars. Below it is a list of items, each in a white box with a black border. The first item, 'Entidades', is highlighted with a grey background and has a small black downward arrow on its right side. A mouse cursor is pointing at the 'Entidades' item. The other items are 'Particular', 'Empresa', 'Voluntários', 'Doadores', 'Donativos', 'Parceiros', 'Atividades', 'Peditórios', and 'Apoios'. The items 'Doadores', 'Peditórios', and 'Apoios' also have small black downward arrows on their right sides. The sidebar is part of a larger interface, with a large white rectangular area to its right.

- Entidades ▼
- Particular
- Empresa
- Voluntários
- Doadores ▼
- Donativos
- Parceiros
- Atividades
- Peditórios ▼
- Apoios ▼

- Gestão de entidades - particulares

Após a selecção da opção “Gestão de Entidades - Particulares” no menu principal da aplicação deve ser apresentada a interface abaixo (Figura 1):

Gestão de Entidades - Particulares

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Ficha de Utente

Nome *

Morada

Código Postal

 -

Data de Nascimento *

/ /

Telefone

E-mail

Genero *

☐ F

☐ M

☐ Outro

Telemóvel

NIF

Página Inicial

Figura 1 – interface da gestão de entidades particulares

O módulo de gestão de entidades deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação das entidades.

Nomeadamente: -Inserir a informação de uma nova entidade (empresa ou particular).

-Inserir a informação da morada juntamente com o código postal (empresa ou entidade).

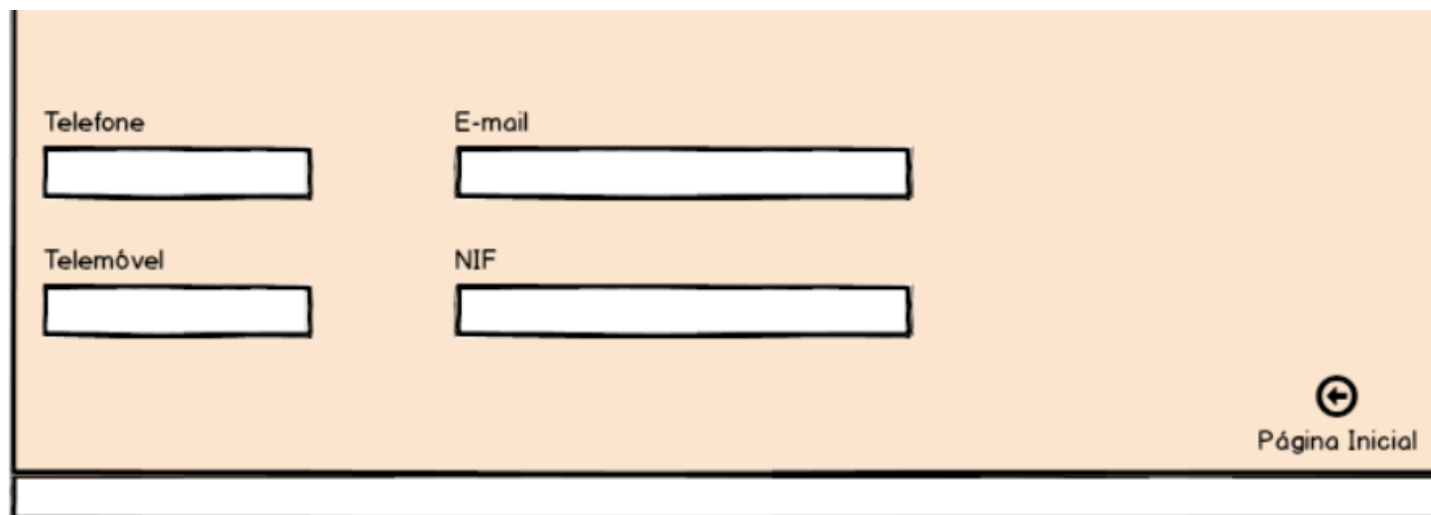
-Se for uma entidade particular é necessário a inserção.

-Inserir o telefone, e-mail, telemóvel e NIF (empresa ou particular).

-Se for uma entidade particular é necessário ainda preencher o género

- Gestão de Entidades - Empresas

The screenshot displays a web interface titled "Gestão de Entidades - Empresas". At the top, there are three buttons: "LISTAR", "ADICIONAR/EDITAR" (highlighted in yellow), and "ELIMINAR". Below these buttons is a section titled "Ficha de Utente" with a light orange background. This section contains a form with the following fields: "Nome *" (with a red asterisk indicating it is required), a text input field, "Morada" (with a text input field), and "Código Postal" (with two separate input fields separated by a hyphen). The interface is designed for managing company information.



A interface de gestão de entidades - empresas é apresentada num formulário com fundo laranja. No topo, há dois campos de entrada: 'Telefone' e 'E-mail'. Abaixo deles, encontram-se os campos 'Telemóvel' e 'NIF'. No canto inferior direito, há um botão com um ícone de seta para trás e o texto 'Página Inicial'. A interface é simples e funcional, com uma barra branca na base.

Figura 2 – interface da gestão de entidades - empresas

O módulo de gestão de entidades deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação das entidades.

Nomeadamente:

- Inserir a informação de uma nova entidade (empresa ou particular).
- Inserir a informação da morada juntamente com o código postal (empresa ou entidade).
- Se for uma entidade particular é necessário a inserção.
- Inserir o telefone, e-mail, telemóvel e NIF (empresa ou particular).
- Se for uma entidade particular é necessário ainda preencher o género

- Gestão de Voluntários

Gestão de Voluntários

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Ficha de Voluntário

Nome *

Morada

Data de Nascimento *

/ /

Telefone

Telemóvel

Código Postal

-

E-mail

NIF *

Função *

Estado *

☐ Ativo

Genero *

☐ F

☐ M

☐ Outro

Número Horas *

Página Inicial

Figura 3 – interface da gestão de voluntários

O módulo de gestão de voluntários deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos voluntários.

Nomeadamente:

- Inserir a informação de um novo voluntário.
- Editar informação sobre voluntários já existentes. Para isso o utilizador deve seleccionar o voluntário que pretende alterar.
- Ao adicionar ou editar os voluntários irá ser verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação de voluntários que existem na base de dados.
- Eliminar voluntários que já não queiram fazer parte do projeto.

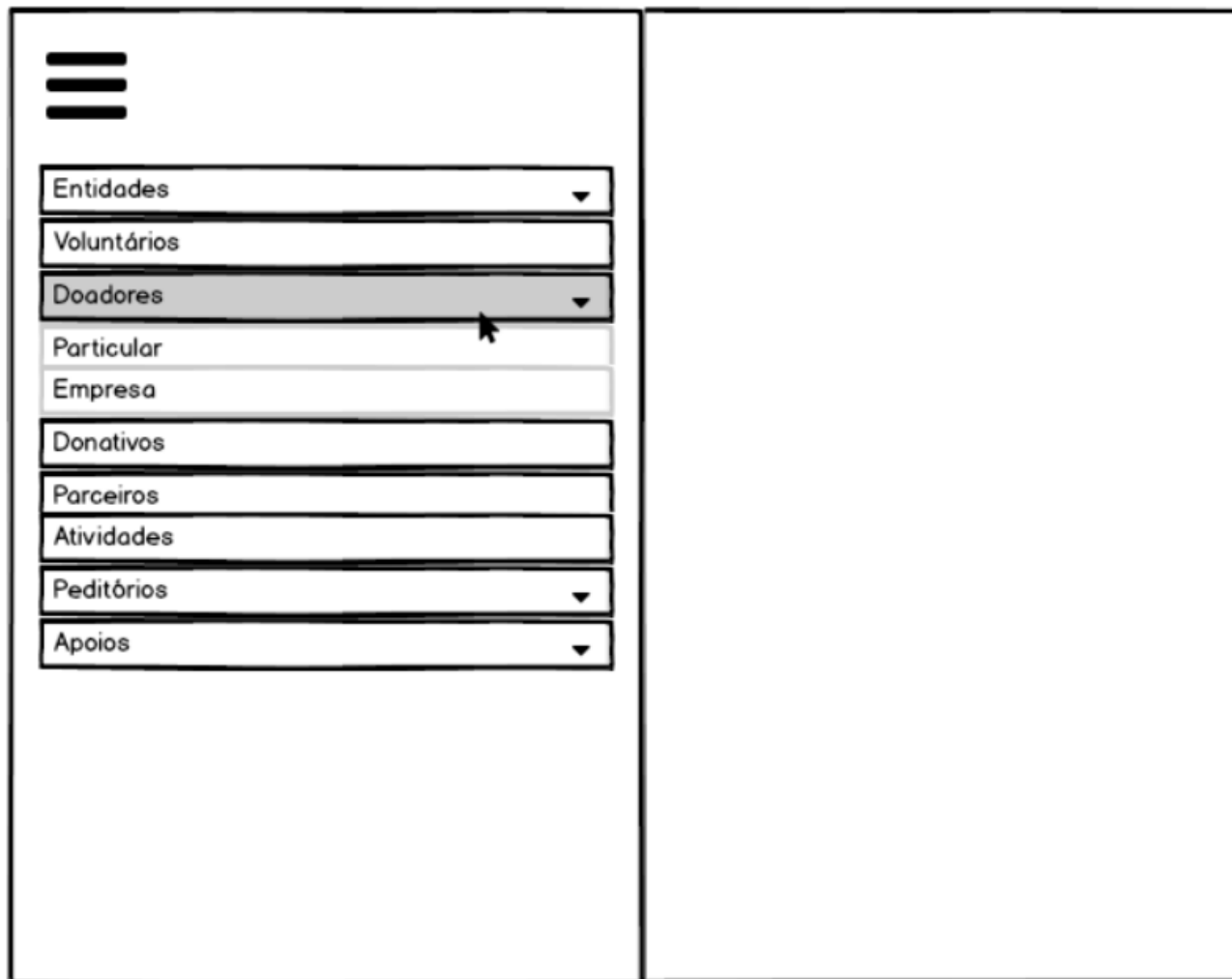


Figura 4 – interface do menu

- Gestão de Doadores - Particulares

O módulo de gestão de doadores deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos doadores.

Nomeadamente:

- Inserir a informação de um novo doador, seja este particular ou uma empresa.
- Editar informação sobre doadores já existentes. Para isso o utilizador deve seleccionar qual o doador que pretende alterar os dados.
- Ao adicionar ou editar os doadores irá ser verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação sobre doadores que existem na base de dados.
- Colocar o estado do doador como inativo quando este desejar deixar o projeto.

Gestão de Doadores - Particulares

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Ficha de Doador

Nome *

Morada

Código Postal

 -

Data de Nascimento *

/ /

Telefone

E-mail

Genero *

☐ F

☐ M

☐ Outro

Telemóvel

NIF

Página Inicial

Figura 5 – interface da gestão de doadores - prticulares

- Gestão de Doadores - Empresas

Gestão de Doadores - Empresas

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Ficha de Empresa

Nome *

Morada

Código Postal


 -

Telefone

E-mail

Telemóvel

NIF



Página Inicial

Figura 6 – interface da gestão de doadores - empresas

O módulo de gestão de doadores deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos doadores.

Nomeadamente:

- Inserir a informação de um novo doador, seja este particular ou uma empresa.
- Editar informação sobre doadores já existentes. Para isso o utilizador deve seleccionar qual o doador que pretende alterar os dados.
- Ao adicionar ou editar os doadores irá ser verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação sobre doadores que existem na base de dados.
- Colocar o estado do doador como inativo quando este desejar deixar o projeto.

- Gestão de Donativos

Gestão de Donativos

LISTAR

ADICIONAR/EDITAR

ELIMINAR


Donativos

Nome *

Designação *

Data de Doação *

/ /



Estado *

☐ Ativo


Página Inicial

Figura 7 – interface da gestão de donativos

Esta gestão de voluntários deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos voluntários.

Nomeadamente:

- O nome da pessoa ou uma empresa, se for anonimo não terá de preencher o campo;
- A sua designação;
- A data da doação do mesmo;
- O seu estado, ativo ou inativo;

- Gestão de Parceiros

Gestão de Parceiros

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Ficha de Parceiros

Nome *

Morada

Código Postal

 -

Descrição

Telefone

E-mail

Estado *

☐ Ativo

Telemóvel

NIF

⬅

Página Inicial

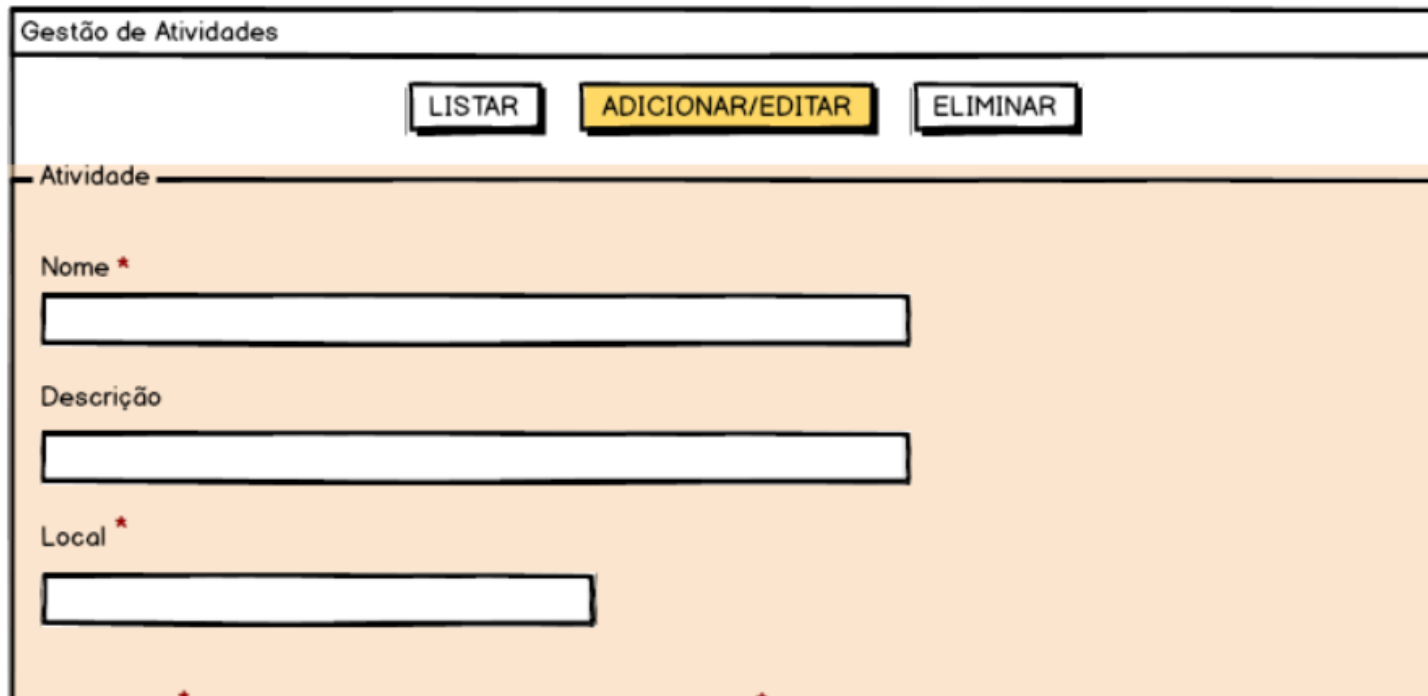
Figura 8 – interface da gestão de parceiros

O módulo de gestão de parceiros deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos parceiros.

Nomeadamente:

- Inserir a informação de um novo parceiro.
- Editar informação sobre parceiros já existentes. Para isso o utilizador deve seleccionar o nome do parceiro cujos dados pretende alterar.
- Ao adicionar ou editar parceiros irá ser verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação sobre parceiros que existem na base de dados.
- Colocar o estado do parceiro como inativo quando este desejar deixar o projeto.

Gestão de Atividades



Gestão de Atividades

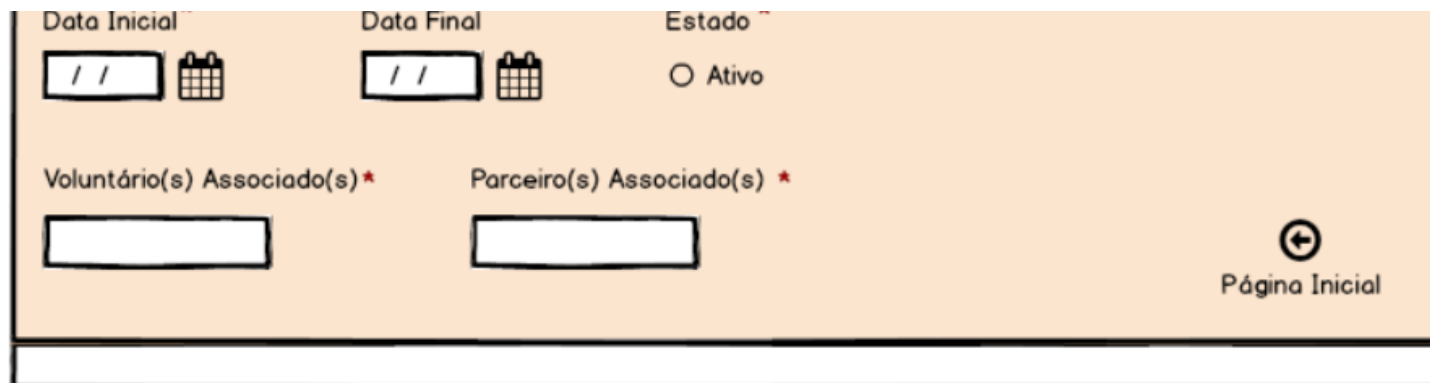
LISTAR ADICIONAR/EDITAR ELIMINAR

Atividade

Nome *

Descrição

Local *



A interface da gestão de atividades é apresentada em um formulário com fundo amarelo. No topo, há três campos: 'Data Inicial' e 'Data Final', cada um com um campo de entrada de texto contendo '/' e um ícone de calendário; e 'Estado', com um botão de rádio rotulado 'Ativo'. Abaixo, há dois campos rotulados 'Voluntário(s) Associado(s) *' e 'Parceiro(s) Associado(s) *', cada um com um campo de entrada de texto. No canto inferior direito, há um botão com um ícone de seta para a esquerda e o texto 'Página Inicial'.

Figura 9 – interface da gestão de atividades

O módulo de gestão de atividades deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação das atividades. Nomeadamente:

- Inserir a informação sobre novas atividades.
- Editar informação sobre atividades já existentes. Para isso o utilizador deve selecionar a atividade que pretende alterar.
- Ao adicionar ou editar atividades irá ser verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação sobre atividades que existem na base de dados.
- Eliminar a atividade.

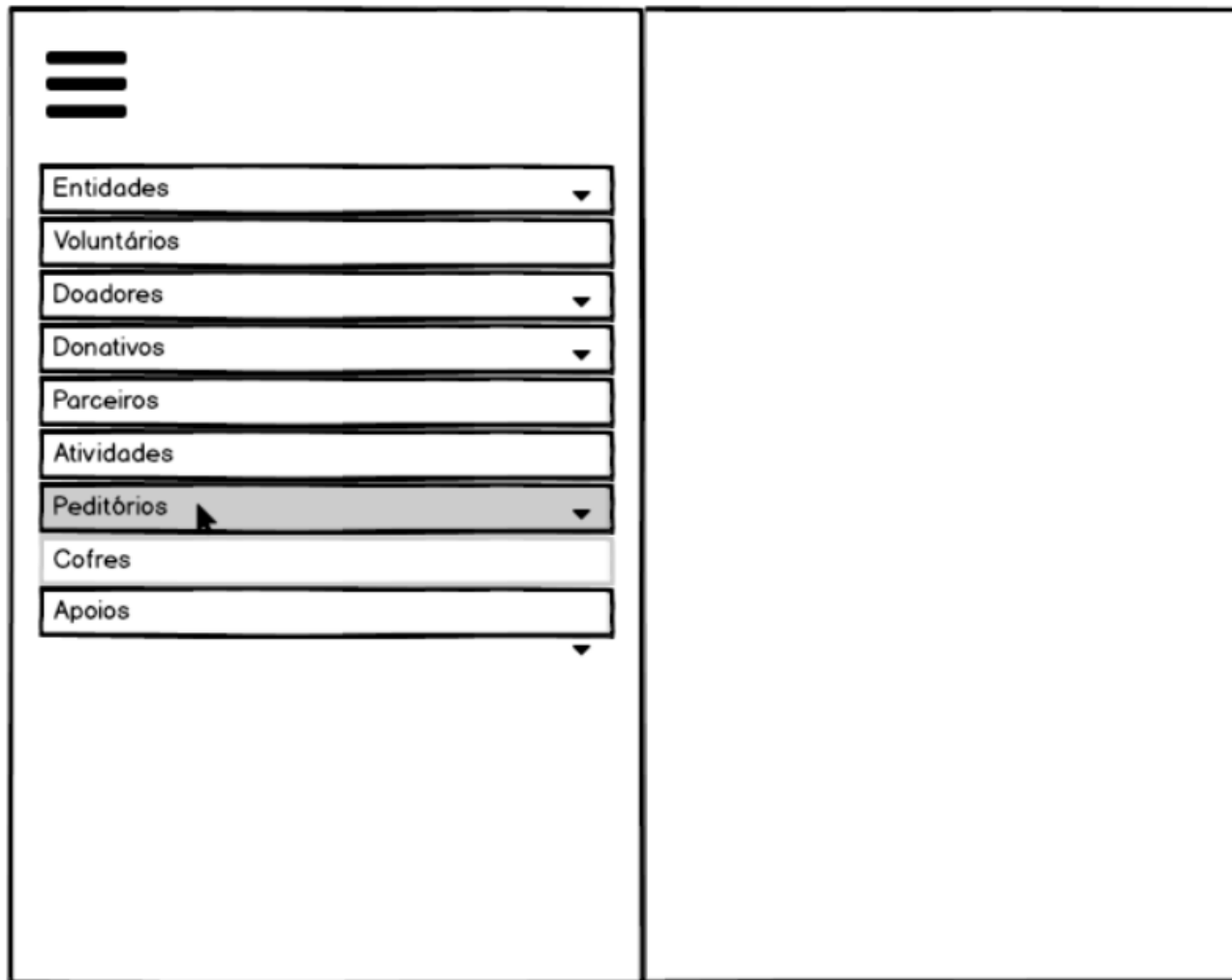


Figura 10 – interface do menu

- Gestão de Peditórios

Gestão de Peditórios

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Peditórios

Nome *

Data Inicial *

Data Final

Voluntário(s) Associado(s) *

Parceiro(s) Associado(s) *

Página Inicial

Figura 11 – interface de peditórios

O módulo de gestão de peditórios deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos peditórios.

Nomeadamente:

- Inserir a informação sobre novos peditórios.
- Editar informação sobre peditórios já existentes. Para isso o utilizador deve seleccionar o peditório que pretende alterar.
- Ao adicionar ou editar peditórios será verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação sobre peditórios que existem na base de dados.
- Eliminar a peditórios.

- Gestão de Cofres

Gestão de Cofres

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Cofres

Montante

Número de Cofre*

Data de Entrega*

/ /

Data de Devolução

/ /

Voluntários(s) Associado(s) ▼*

Ana Santos

Beatriz Gosta

Bruno Aleixo

Parceiros(s) Associado(s) ▼*

Fórum Coimbra

Coimbra Shopping

Alma Shopping

Página Inicial

Figura 12 – interface da gestão de cofres

O módulo de gestão de cofres deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos voluntários.

Nomeadamente:

- Inserir o montante
- Adicionar o número do cofre
- Juntamente com a data de entrega e data de devolução
- Ainda é necessária o preenchimento do campo voluntário/a associado/a e o campo do parceiro/a associado/a

The image shows a web application interface for managing safes. On the left, there is a vertical sidebar menu with the following items: 'Entidades', 'Voluntários', 'Doadores', 'Donativos', 'Parceiros', 'Atividades', 'Peditórios', 'Apoios' (which is highlighted and has a mouse cursor pointing to it), 'Particulares', and 'Empresas'. The 'Apoios' item has a small downward arrow on its right side. The rest of the interface is a large, empty rectangular area.



Figura 13 – interface do menu

- Gestão de Apoios - Particulares

Gestão de Apoios

LISTAR

ADICIONAR/EDITAR

ELIMINAR

Apoio Particular

Nome

Entidade Carenciada*

Designação*

Valor*

Observações

Estado*

☐ Ativo

Data de Apoio*

/ /


Página Inicial

Figura 14 – interface da gestão de apoios - particulares

O módulo de gestão de apoios deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos apoios.

Nomeadamente:

- Inserir a informação sobre novos apoios, fazendo a distinção entre apoios a particulares e apoios a empresas.
- Editar informação sobre apoios já existentes. Para isso o utilizador deve selecionar o apoio que pretende alterar.
- Ao adicionar ou editar apoios será verificado se a informação introduzida satisfaz as restrições definidas na base de dados.
- Pesquisar a informação sobre apoios que existem na base de dados.
- Eliminar a apoios.

- Gestão de Apoios - Empresas

Gestão de Apoios

LISTAR ADICIONAR/EDITAR ELIMINAR

Apoio Empresa

Nome

Entidade Carenciada *

Designação *

Valor *

Observações

Data de Apoio *

Estado *

Ativo



Figura 15 – interface da gestão de apoios - empresas

O módulo de gestão de apoios deve permitir na mesma interface todas as operações possíveis de executar relativamente à informação dos apoios.

Nomeadamente:

- Inserir a informação sobre novos apoios, fazendo a distinção entre apoios a particulares e apoios a empresas.
- Editar informação sobre apoios já existentes. Para isso o utilizador deve seleccionar o apoio que pretende alterar.
- Ao adicionar ou editar apoios será verificado se a informação introduzida satisfaz as restrições definidas na base de dados. -Pesquisar a informação sobre apoios que existem na base de dados.
- Eliminar a apoios.

Cap 6 - Análise de Dados

Neste capítulo pretende-se descrever detalhadamente todas as entidades envolvidas na solução proposta. Na subsecção 6.1 são descritas todas as entidades (respectivos atributos e restrições), na secção 6.2 são descritas todos os relacionamentos existentes entre as diversas entidades, apresentando-se por fim, na secção 6.3, o modelo de Entidade / Relacionamento completo, incluindo todas as entidades e relações relevantes.

Sec 6.1 - Entidades

Nesta secção vão ser descritas todas as Entidades relevantes para o sistema de voluntariado. Após uma análise aprofundada deste sistema, constatou-se a necessidade das seguintes Entidades:

- Entidades Carenciadas
- Voluntários
- Doadores

- Doador não Anónimo
- Doador Particular
- Peditórios
- Cofres Peditório
- Donativos
- Atividades
- Recibos
- Parceiros

Sec 6.1.1 - Entidades Carenciadas

Entidade que consiste nas pessoas e entidades carenciadas, que poderão receber apoios.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_CARENCIADOS	ID_CARENCIADOS	Long integer	N	X	Identificador/chave primária da entidade carenciada
NOME_ENTIDADE	NOME_ENTIDADE	Variable characters (70)	N		Nome da entidade
N_TELEMOVEL	N_TELEMOVEL	Variable characters (15)	S		Número de telemóvel
N_TELEFONE	N_TELEFONE	Variable characters (15)	S		Número de telefone
E_MAIL	E_MAIL	Variable characters (15)	S		email
MORADA	MORADA	Variable characters (100)	N		Morada
NIF	NIF	Integer	N		Número de Identificação Fiscal

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 -> Entidade2 Role Cardinality	Entidade2 -> Entidade1 Role Cardinality
APOIO_ÀS_ENTIDADES	APOIO_AS_ENTIDADES	X	APOIOS_ENTIDADES	ENTIDADES_CARENCIADAS	1,n	1,1
CODIGOPOSTAL_DA_ENTIDADE	CODIGOPOSTAL_DA_ENTIDADE	X	ENTIDADES_CARENCIADAS	CODIGO_POSTAL	1,n	1,1
CONCELHO_DA_ENTIDADE	CONCELHO_DA_ENTIDADE	X	ENTIDADES_CARENCIADAS	CONCELHO	1,n	1,1
DISTRITO_DA_ENTIDADE	DISTRITO_DA_ENTIDADE	X	ENTIDADES_CARENCIADAS	DISTRITO	1,n	1,1
TIPOS_DE_ENTIDADES_CARENCIADAS	TIPOS_DE_ENTIDADES_CARENCIADAS	X	ENTIDADES_CARENCIADAS	TIPO_ENTIDADES	1,n	1,1

Sec 6.1.2 - Entidade Voluntários

Tabela que lista todos os voluntários.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_VOLUNTARIO	ID_VOLUNTARIO	Long integer	N	X	Identificador/chave primária da do voluntário
NOME_VOLUNTARIO	NOME_VOLUNTARIO	Variable characters (70)	N		Nome do voluntário
DATA_NASCIMENTO	DATA_NASCIMENTO	Date	N		Data de Nascimento
MORADA	MORADA	Variable characters (100)	S		Morada do voluntário
GENERO	GENERO	Short integer	N		1 - Feminino 2 - Masculino 3 - Outro
NIF	NIF	Integer	S		Número de Identificação Fiscal

FUNCAO	FUNCAO	Variable characters (75)	N		A função desempenhada pelo voluntário, por ex:-apoio domiciliário, transporte de utentes, apoio hospitalar, etc..
N_HORAS	N_HORAS	Integer	N		Número de horas trabalhadas
TELEFONE	TELEFONE	Variable characters (15)	S		Telefone do voluntário
TELEMOVEL	TELEMOVEL	Variable characters (15)	S		Telemóvel do voluntário
E_MAIL	E_MAIL	Variable characters (15)	S		Email do voluntário
ESTADO_VOLUNTARIO	ESTADO_VOLUNTARIO	Number (1)	N		Estado do registo voluntário: 1- ativo, 2- inativo

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
CODIGOPOSTAL_DOS_VOLUNTARIOS	CODIGOPOSTAL_DOS_VOLUNTARIOS	X	VOLUNTARIOS	CODIGO_POSTAL	1,n	1,1
COFRES_ATRIBUIDOS_AOS_VOLUNTARIOS	COFRES_ATRIBUIDOS_AOS_VOLUNTARIOS	X	COFRES_PEDITORIO	VOLUNTARIOS	1,n	1,1
CONCELHO_DOS_VOLUNTARIOS	CONCELHO_DOS_VOLUNTARIOS	X	VOLUNTARIOS	CONCELHO	1,n	1,1
DISTRITO_DOS_VOLUNTARIOS	DISTRITO_DOS_VOLUNTARIOS	X	VOLUNTARIOS	DISTRITO	1,n	1,1
PARTICIPAM_EM	PARTICIPAM_EM	X	ATIVIDADES	VOLUNTARIOS	1,n	1,n
PARTICIPAM_EM	PARTICIPAM_EM_2	X	PEDITORIOS	VOLUNTARIOS	1,n	1,n

Sec 6.1.3 - Entidade Doadores

Tabela que identifica os doadores, se for anónimo nenhuma das tabelas de herança será preenchida

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_DOADOR	ID_DOADOR	Integer	N	X	Identificador/chave primária da entidade doadora

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
DOADORES_DONATIVOS	DOADORES_DONATIVOS	X	DONATIVOS	DOADORES	1,n	1,1
TIPO_ENTIDADE_DOADORA	TIPO_ENTIDADE_DOADORA	X	TIPO_ENTIDADES	DOADORES	1,1	1,n

Sec 6.1.3.1 - Entidade Doador não Anónimo

Tabela que lista todos os doadores não anónimos

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
NOME_DOADOR	NOME_DOADOR	Variable characters (70)	N		Nome do doador
N_TELEMOVEL	N_TELEMOVEL	Variable characters (15)	S		Número de telemóvel
N_TELEFONE	N_TELEFONE	Variable characters (15)	S		Número de telefone
E_MAIL	E_MAIL	Variable characters (15)	S		Email
MORADA	MORADA	Variable characters (100)	N		Morada

NIF	NIF	Integer	N		Número de Identificação Fiscal
-----	-----	---------	---	--	--------------------------------

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
CODIGOPOSTAL_DOS_DOADORES	CODIGOPOSTAL_DOS_DOADORES	X	DOADORES_NAO_ANONIMO	CODIGO_POSTAL	0,n	0,1
CONCELHO_DOS_DOADORES	CONCELHO_DOS_DOADORES	X	DOADORES_NAO_ANONIMO	CONCELHO	0,n	0,1
DISTRITO_DOS_DOADORES	DISTRITO_DOS_DOADORES	X	DISTRITO	DOADORES_NAO_ANONIMO	0,1	0,n

Sec 6.1.3.2 - Entidade Doador Particular

Entidade que vai conter os dados do doador do tipo particular.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
DATA_NASCIMENTO	DATA__NASCIMENTO	Date	N		Data de nascimento
GENERO	GENERO	Short integer	N		1 - Feminino 2 - Masculino 3 - Outro

Sec 6.1.4 - Entidade Peditórios

Tabela que lista todos os peditórios.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_PEDITORIO	ID_PEDITORIO	Integer	N	X	Chave primária da tabela peditórios
DESIGNACAO_PEDITORIOS	DESIGNACAO_PEDITORIOS	Variable characters (100)	N		
DATA_INI	DATA_INI	Date	N		Data inicial
DATA_FIM	DATA_FIM	Date	N		Data final

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
PARTICIPAM_EM	PARTICIPAM_EM_3	X	PEDITORIOS	PARCEIROS	1,n	1,n
PARTICIPAM_EM	PARTICIPAM_EM_2	X	PEDITORIOS	VOLUNTARIOS	1,n	1,n
TÊM	TEM	X	COFRES_PEDITORIO	PEDITORIOS	1,n	1,1

Sec 6.1.5 - Entidade Cofres Peditório

Tabela onde são registados os detalhes de cada cofre vindo dos peditórios.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_COFRE	ID_COFRE	Long integer	N	X	Chave primária dos cofres do peditório
MONTANTE_PEDITORIO	MONTANTE_PEDITORIO	Number (8,2)	N		Valor monetário dentro do cofre, angariado num peditório
DATA_ENTREGA	DATA_ENTREGA	Date	N		Data de entrega do peditório

DATA_DEVOLUCAO	DATA_DEVOLUCAO	Date	S		Data de devolução do cofre
NUMERO_COFRE	NUMERO_COFRE	Integer	N		Número correspondente a cada cofre

Sec 6.1.6 - Entidade Donativos

Tabela onde são registados os donativos recebidos.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_DONATIVO	ID_DONATIVO	Long integer	N	X	Chave primária
MONTANTE	MONTANTE	Number (8,2)	N		Valor monetário correspondente ao donativo
DATA_DOACAO	DATA_DOACAO	Date	N		Data da doação

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
DOADORES_DONATIVOS	DOADORES_DONATIVOS	X	DONATIVOS	DOADORES	1,n	1,1
DONATIVO_É_DO_TIPO	DONATIVO_E_DO_TIPO	X	DONATIVOS	TIPO_DONATIVOS	1,n	1,1
RECIBOS_DOS_DONATIVOS	RECIBOS_DOS_DONATIVOS	X	RECIBOS	DONATIVOS	1,n	1,1

Sec 6.1.7- Entidade Atividades

Tabela onde estão registados os detalhes das atividades.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_ATIVIDADE	ID_ATIVIDADE	Integer	N	X	Chave primária da tabela das atividades
DESIGNACAO	DESIGNACAO	Variable characters (100)	N		
DATA_INI	DATA_INI	Date	N		Data inicial
DATA_FIM	DATA_FIM	Date	N		Data final
LOCAL	LOCAL	Variable characters (255)	N		Local onde foi realizada a atividade
ESTADO_ATIVIDADE	ESTADO_ATIVIDADE	Number (1)	N		Estado do registo das atividades: 1- planeada, 2- cancelada, 3-concluída

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
ATIVIDADE_É_DO_TIPO	ATIVIDADE_E_DO_TIPO	X	ATIVIDADES	TIPO_ATIVIDADES	1,n	1,1
COLABORAM_EM	COLABORAM_EM	X	ATIVIDADES	PARCEIROS	1,n	1,n
PARTICIPAM_EM	PARTICIPAM_EM	X	ATIVIDADES	VOLUNTARIOS	1,n	1,n

Sec 6.1.8 - Entidade Recibos

Tabela que contém o id do recibos assim como a sua data de emissão.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_RECIBO	ID_RECIBO	Long integer	N	X	Chave primária

DATA_RECIBO	DATA_RECIBO	Date & Time	N		Data de emissão do recibo
-------------	-------------	-------------	---	--	---------------------------

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
RECIBOS_DOS_DONATIVOS	RECIBOS_DOS_DONATIVOS	X	RECIBOS	DONATIVOS	1,n	1,1

Sec 6.1.9 - Entidade Parceiros

Tabela que lista todos os parceiros.

Atributos relevantes da Entidade e restrições dos seus atributos:

Nome do Atributo	Código	Tipo de Dados	Aceita Nulos?	Chave Primária	Observações
ID_PARCEIRO	ID_PARCEIRO	Short integer	N	X	Chave primária da tabela dos parceiros
NOME_PARCEIRO	NOME_PARCEIRO	Variable characters (70)	N		Nome a associado a cada parceiro
DESCRICAO_PARCEIROS	DESCRICAO_PARCEIROS	Variable characters (100)	S		
ESTADO_PARCEIRO	ESTADO_PARCEIRO	Number (1)	N		Estado do registo parceiro: 1- ativo, 2- inativo

Relacionamentos da Entidade:

Nome do Atributo	Código	Generate	Entidade2	Entidade1	Entidade1 - > Entidade2 Role Cardinality	Entidade2 - > Entidade1 Role Cardinality
CODIGOPOSTAL_DOS_PARCEIROS	CODIGOPOSTAL_DOS_PARCEIROS	X	PARCEIROS	CODIGO_POSTAL	1,n	1,1
COFRES_ATRIBUIDOS_AOS_PARCEIROS	COFRES_ATRIBUIDOS_AOS_PARCEIROS	X	PARCEIROS	COFRES_PEDITORIO	1,n	1,1

COLABORAM_EM	COLABORAM_EM	X	ATIVIDADES	PARCEIROS	1,n	1,n
CONCELHO_DOS_PARCEIROS	CONCELHO_DOS_PARCEIROS	X	PARCEIROS	CONCELHO	1,n	1,1
DISTRITO_PARCEIROS	DISTRITO_PARCEIROS	X	PARCEIROS	DISTRITO	1,n	1,1
PARTICIPAM_EM	PARTICIPAM_EM_3	X	PEDITORIOS	PARCEIROS	1,n	1,n

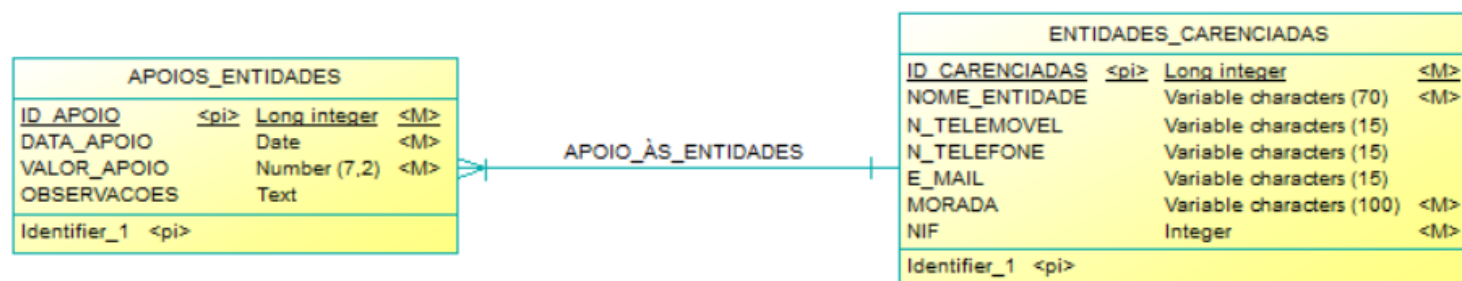
Sec 6.2 - Relacionamentos

Nesta secção são descritos todos os relacionamentos existentes entre as várias entidades. Após uma análise aprofundada do sistema de voluntariado, constatou-se a necessidade dos seguintes relacionamentos:

- APOIO_ÀS_ENTIDADES
- DOADORES_DONATIVOS
- COFRES_ATRIBUIDOS_AOS_VOLUNTARIOS
- PARTICIPAM_EM
- COLABORAM_EM
- COFRES_ATRIBUIDOS_AOS_PARCEIROS
- RECIBOS_DOS_DONATIVOS

Sec 6.2.1 - Relacionamento: APOIO_ÀS_ENTIDADES

Este relacionamento pretende expressar o relacionamento existente entre as Entidades APOIOS_ENTIDADES e ENTIDADES_CARENCIADAS. O objectivo é expressar o/os apoio(s) prestados a entidades.



Entidade 1	ENTIDADES_CARENCIADAS
Entidade 2	APOIOS_ENTIDADES

Entidade 1 ⇒Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒Entidade 1 Cardinalidade	1,1
Entidade 1 ⇒Entidade 2 Obrigatório	Sim
Entidade 2 ⇒Entidade 1 Obrigatório	Sim
Observações	Relação entre as entidades ENTIDADES_CARENCIADAS e APOIOS_ENTIDADES que desgina o/os apoio(s) prestados a entidades

Sec 6.2.2 - Relacionamento: DOADORES_DONATIVOS

Este relacionamento pretende expressar o relacionamento existente entre as Entidades DONATIVOS e DOADORES. O objectivo é expressar os donativos feitos pelos doadores.~



Entidade 1	DOADORES
Entidade 2	DONATIVOS
Entidade 1 ⇒Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒Entidade 1 Cardinalidade	1,1
Entidade 1 ⇒Entidade 2 Obrigatório	Sim
Entidade 2 ⇒Entidade 1 Obrigatório	Sim

Observações	Relação entre as entidades DOADORES e DONATIVOS, que designa os donativos feitos pelos doadores
-------------	---

Sec 6.2.3 - Relacionamento: COFRES_ATRIBUIDOS_AOS_VOLUNTARIOS

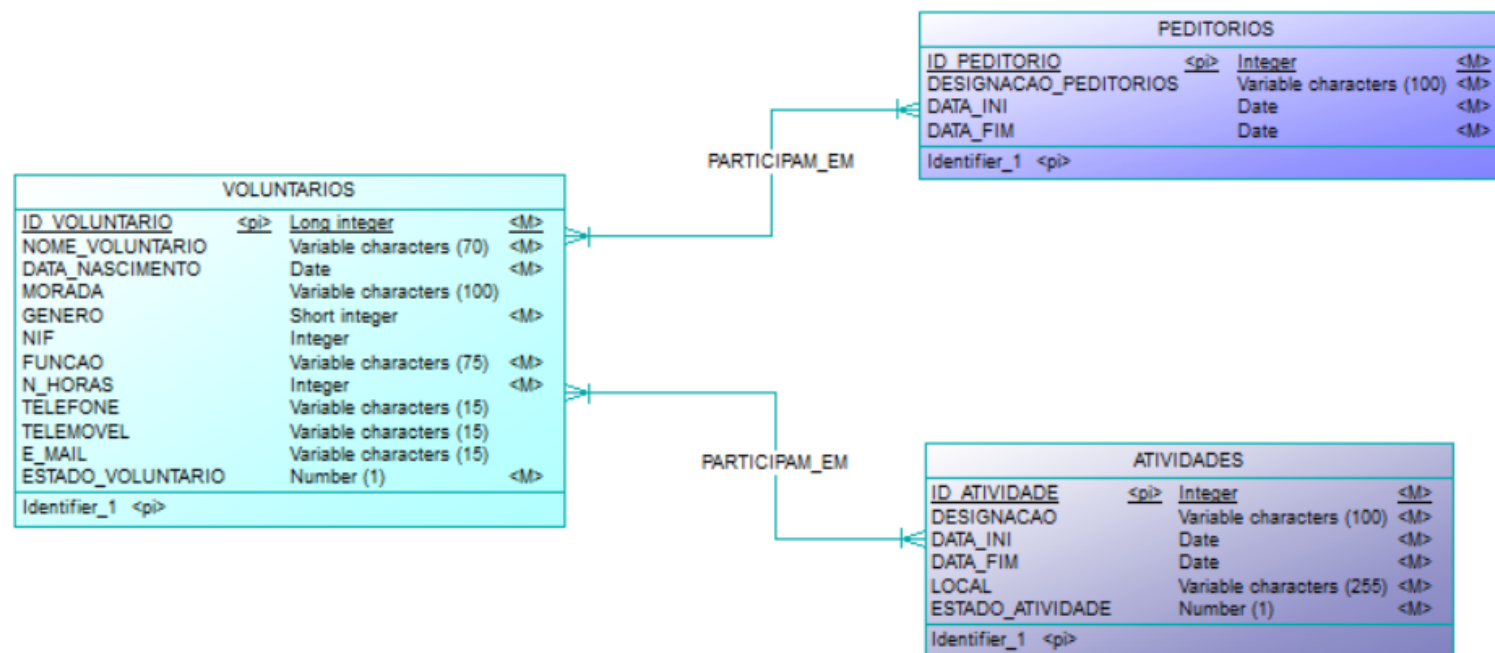
Este relacionamento pretende expressar o relacionamento existente entre as Entidades VOLUNTARIOS e COFRES_PEDITORIO. O objectivo é expressar a atribuição dos cofres aos voluntários.



Entidade 1	VOLUNTARIOS
Entidade 2	COFRES_PEDITORIO
Entidade 1 ⇒Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒Entidade 1 Cardinalidade	1,1
Entidade 1 ⇒Entidade 2 Obrigatório	Sim
Entidade 2 ⇒Entidade 1 Obrigatório	Sim
Observações	Relação entre as entidades VOLUNTARIOS e COFRES_PEDITORIO que designa a atribuição dos cofres aos voluntários

Sec 6.2.4 - Relacionamento: PARTICIPAM_EM

Este relacionamento pretende expressar o relacionamento existente entre as Entidades VOLUNTARIOS e PEDITORIOS, e entre as Entidades VOLUNTARIOS e ATIVIDADES. O objectivo é expressar a participação dos voluntários nos peditórios e nas atividades, respetivamente.



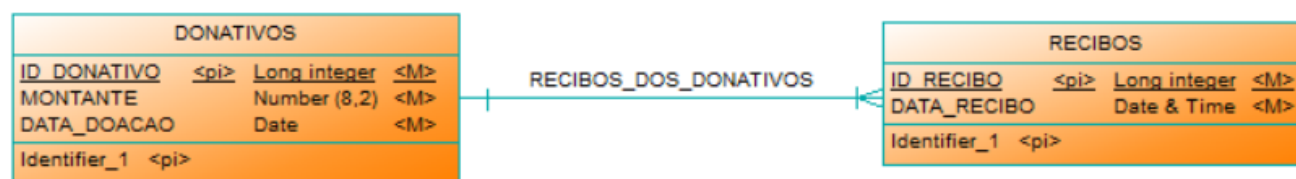
Entidade 1	VOLUNTARIOS
Entidade 2	PEDITORIOS
Entidade 1 ⇒ Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒ Entidade 1 Cardinalidade	1,n
Entidade 1 ⇒ Entidade 2 Obrigatório	Sim
Entidade 2 ⇒ Entidade 1 Obrigatório	Sim

Observações	Relação entre as entidades VOLUNTARIOS e PEDITORIOS, que designa a participação dos voluntários Nãos peditórios
-------------	---

Entidade 1	VOLUNTARIOS
Entidade 2	ATIVIDADES
Entidade 1 ⇒Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒Entidade 1 Cardinalidade	1,n
Entidade 1 ⇒Entidade 2 Obrigatório	Sim
Entidade 2 ⇒Entidade 1 Obrigatório	Sim
Observações	

Sec 6.2.5 - Relacionamento: RECIBOS_DONATIVOS

Este relacionamento pretende expressar o relacionamento existente entre as Entidades DONATIVOS e RECIBOS. O objectivo é relacionar os recibos correspondentes aos donativos.



Entidade 1	DONATIVOS
Entidade 2	RECIBOS
Entidade 1 ⇒Entidade 2 Cardinalidade	1,n

Entidade 2 ⇒Entidade 1 Cardinalidade	1,1
Entidade 1 ⇒Entidade 2 Obrigatório	Sim
Entidade 2 ⇒Entidade 1 Obrigatório	Sim
Observações	Relação entre as entidades DONATIVOS e RECIBO, que relaciona os recibos correspondentes aos donativos

Sec 6.2.6 - Relacionamento: COLABORAM_EM

Este relacionamento pretende expressar o relacionamento existente entre as Entidades PARCEIROS e ATIVIDADES. O objectivo é expressar a colaboração dos parceiros nas atividades realizadas pelo sistema de voluntariado.

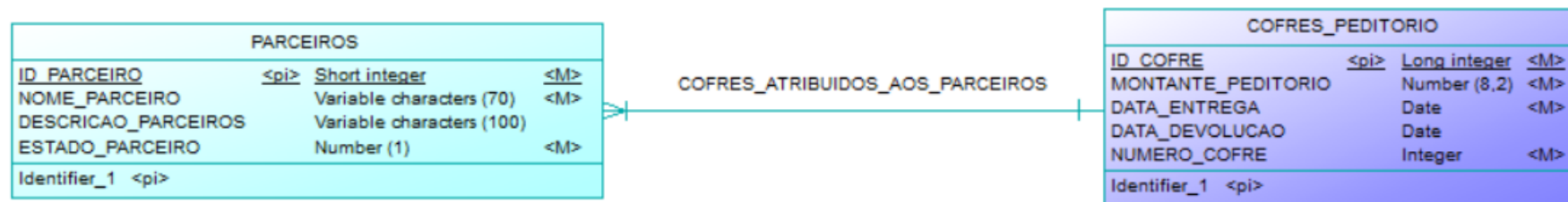


Entidade 1	PARCEIROS
Entidade 2	ATIVIDADES
Entidade 1 ⇒Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒Entidade 1 Cardinalidade	1,n
Entidade 1 ⇒Entidade 2 Obrigatório	Sim
Entidade 2 ⇒Entidade 1 Obrigatório	Sim

Observações	Relação entre as entidades PARCEIROS e ATIVIDADES, que designa a colaboração dos parceiros nas atividades realizadas pelo sistema de voluntariado
-------------	---

Sec 6.2.7 - Relacionamento: COFRES_ATRIBUIDOS_AOS_PARCEIROS

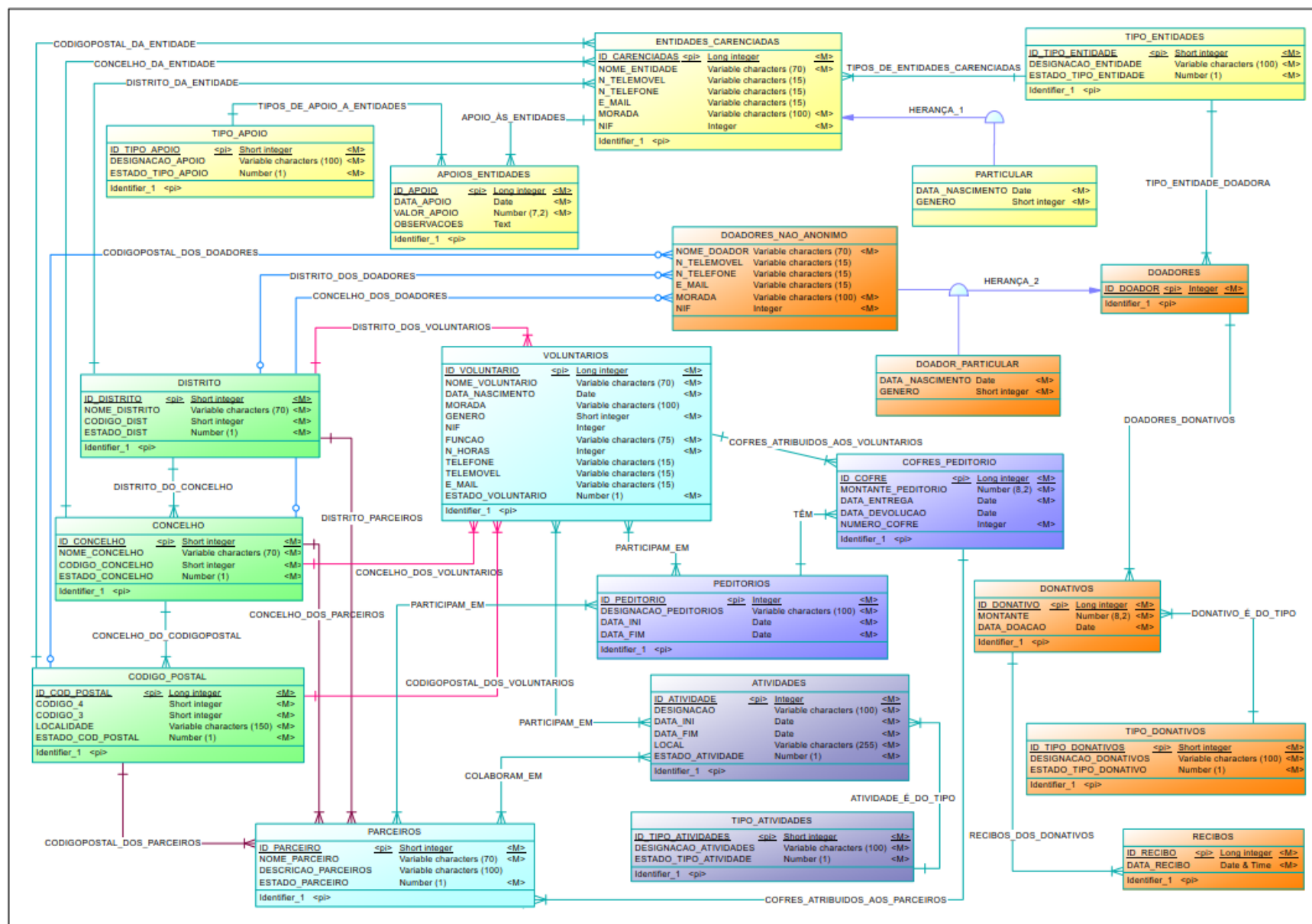
Este relacionamento pretende expressar o relacionamento existente entre as Entidades PARCEIROS e COFRES_PEDITORIO. O objectivo é expressar a atribuição dos cofres do peditório aos parceiros.



Entidade 1	COFRES_PEDITORIO
Entidade 2	PARCEIROS
Entidade 1 ⇒ Entidade 2 Cardinalidade	1,n
Entidade 2 ⇒ Entidade 1 Cardinalidade	1,1
Entidade 1 ⇒ Entidade 2 Obrigatório	Sim
Entidade 2 ⇒ Entidade 1 Obrigatório	Sim
Observações	Relação entre as entidades COFRES_PEDITORIOS e PARCEIROS, que designa a atribuição dos cofres do peditório aos parceiros

Sec 6.3 - Diagrama do Modelo Conceptual

O modelo de Entidade Relacionamento completo é o seguinte.



Cap 7 - Modelo Físico

Neste capítulo documenta-se detalhadamente todas as tabelas geradas para o Sistema de Gestão de Bases de Dados Relacional (SGBDR) Oracle, a partir do modelo Relacional definido no capítulo 6. Na subsecção 7.1, é descrita cada uma das tabelas (respectivos atributos e restrições de integridade), apresentando-se na secção 7.2 o modelo Físico (ou modelo de tabelas) completo, composto por todas as tabelas e restrições de integridade referencial. A secção 7.3 apresenta o código SQL necessário para criar as tabelas e as restrições definidas.

Sec 7.1 - Tabelas

Nesta secção são descritas as tabelas necessárias para implementar o negócio de venda de livros da livraria. Estas tabelas foram extraídas, tomando em consideração o modelo Entidade / Relacionamento definido no capítulo 6. As tabelas são as seguintes:

- Entidades Carenciadas
- Voluntários
- Doadores
- Doador não Anónimo
- Doador Particular
- Peditórios
- Cofres Peditório
- Donativos
- Atividades
- Recibos
- Parceiros

Sec 7.1.1 - Tabela Entidades Carenciadas

Entidade que consiste nas pessoas e entidades carenciadas, que poderão receber apoios

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_CARENCIADOS	INTEGER	CKC_ID_CARENCIADOS_ENTIDADE	X	
ID_TIPO_ENTIDADE	SMALLINT	CKC_ID_TIPO_ENTIDADE_ENTIDADE		X
ID_CONCELHO	SMALLINT	CKC_ID_CONCELHO_ENTIDADE		X
ID_COD_POSTAL	INTEGER	CKC_ID_COD_POSTAL_ENTIDADE		X
ID_DISTrito	SMALLINT	CKC_ID_DISTrito_ENTIDADE		X
NOME_ENTIDADE	VARCHAR2(70)	CKC_NOME_ENTIDADE_ENTIDADE		
N_TELEMOVEL	VARCHAR2(15)	CKC_N_TELEMOVEL_ENTIDADE		

N_TELEFONE	VARCHAR2(15)	CKC_N_TELEFONE_ENTIDADE		
E_MAIL	VARCHAR2(15)	CKC_E_MAIL_ENTIDADE		
MORADA	VARCHAR2(100)	CKC_MORADA_ENTIDADE		
NIF	INTEGER	CKC_NIF_ENTIDADE		

Sec 7.1.2 - Tabela Voluntários

Tabela que lista todos os voluntários

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_VOLUNTARIO	INTEGER	CKC_ID_VOLUNTARIO_VOLUNTAR	X	
ID_DISTrito	SMALLINT	CKC_ID_DISTrito_VOLUNTAR		X
ID_CONCELHO	SMALLINT	CKC_ID_CONCELHO_VOLUNTAR		X
ID_COD_POSTAL	INTEGER	CKC_ID_COD_POSTAL_VOLUNTAR		X
NOME_VOLUNTARIO	VARCHAR2(70)	CKC_NOME_VOLUNTARIO_VOLUNTAR		
DATA_NASCIMENTO	DATE	CKC_DATA_NASCIMENTO_VOLUNTAR		
MORADA	VARCHAR2(100)	CKC_MORADA_VOLUNTAR		
GENERO	SMALLINT	CKC_GENERO_VOLUNTAR		
NIF	INTEGER	CKC_NIF_VOLUNTAR		
FUNCAO	VARCHAR2(75)	CKC_FUNCAO_VOLUNTAR		
N_HORAS	INTEGER	CKC_N_HORAS_VOLUNTAR		
TELEFONE	VARCHAR2(15)	CKC_TELEFONE_VOLUNTAR		
TELEMOVEL	VARCHAR2(15)	CKC_TELEMOVEL_VOLUNTAR		
E_MAIL	VARCHAR2(15)	CKC_E_MAIL_VOLUNTAR		
ESTADO_VOLUNTARIO	NUMBER(1)	CKC_ESTADO_VOLUNTARIO_VOLUNTAR		

Sec 7.1.3 - Tabela Doadores

Tabela que identifica os doadores, se for anónimo nenhuma das tabelas de herança será preenchida

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_DOADOR	INTEGER	CKC_ID_DOADOR_DOADORES	X	
ID_TIPO_ENTIDADE	SMALLINT	CKC_ID_TIPO_ENTIDADE_DOADORES		X

Sec 7.1.3.1 - Tabela Doador não Anónimo

Tabela que lista todos os doadores não anónimos

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_DOADOR	INTEGER	CKC_ID_DOADOR_DOADORES	X	X
ID_CONCELHO	SMALLINT	CKC_ID_CONCELHO_DOADORES		X
ID_COD_POSTAL	INTEGER	CKC_ID_COD_POSTAL_DOADORES		X
ID_DISTRITO	SMALLINT	CKC_ID_DISTRITO_DOADORES		X
NOME_DOADOR	VARCHAR2(70)	CKC_NOME_DOADOR_DOADORES		
N_TELEMOVEL	VARCHAR2(15)	CKC_N_TELEMOVEL_DOADORES		
N_TELEFONE	VARCHAR2(15)	CKC_N_TELEFONE_DOADORES		
E_MAIL	VARCHAR2(15)	CKC_E_MAIL_DOADORES		
MORADA	VARCHAR2(100)	CKC_MORADA_DOADORES		
NIF	INTEGER	CKC_NIF_DOADORES		

Sec 7.1.3.2 - Tabela Doador Particular

Tabela onde são registados os detalhes de cada cofre vindo dos peditórios

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_DOADOR	INTEGER	CKC_ID_DOADOR_DOADOR_P	X	X
DATA_NASCIMENTO	DATE	CKC_DATA__NASCIMENTO_DOADOR_P		
GENERO	SMALLINT	CKC_GENERO_DOADOR_P		

Sec 7.1.4 - Tabela Peditórios

Tabela que lista todos os peditórios

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_PEDITORIO	INTEGER	CKC_ID_PEDITORIO_PEDITORI	X	
DESIGNACAO_PEDITORIOS	VARCHAR2(100)	CKC_DESIGNACAO_PEDITO_PEDITORI		
DATA_INI	DATE	CKC_DATA_INI_PEDITORI		
DATA_FIM	DATE	CKC_DATA_FIM_PEDITORI		

Sec 7.1.5 - Tabela Cofres Peditório

Tabela onde são registados os detalhes de cada cofre vindo dos peditórios

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_COFRE	INTEGER	CKC_ID_COFRE_COFRES_P	X	
ID_PEDITORIO	INTEGER	CKC_ID_PEDITORIO_COFRES_P		X
ID_VOLUNTARIO	INTEGER	CKC_ID_VOLUNTARIO_COFRES_P		X
MONTANTE_PEDITORIO	NUMBER(8,2)	CKC_MONTANTE_PEDITORI_COFRES_P		
DATA_ENTREGA	DATE	CKC_DATA_ENTREGA_COFRES_P		
DATA_DEVOLUCAO	DATE	CKC_DATA_DEVOLUCAO_COFRES_P		

NUMERO_COFRE	INTEGER	CKC_NUMERO_COFRE_COFRES_P		
--------------	---------	---------------------------	--	--

Sec 7.1.6 - Tabela Donativos

Tabela onde são registados os donativos recebidos

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_DONATIVO	INTEGER	CKC_ID_DONATIVO_DONATIVO	X	
ID_TIPO_DONATIVOS	SMALLINT	CKC_ID_TIPO_DONATIVOS_DONATIVO		X
ID_DOADOR	INTEGER	CKC_ID_DOADOR_DONATIVO		X
MONTANTE	NUMBER(8,2)	CKC_MONTANTE_DONATIVO		
DATA_DOACAO	DATE	CKC_DATA_DOACAO_DONATIVO		

Sec 7.1.7 - Tabela Atividades

Tabela onde estão registados os detalhes das atividades

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_ATIVIDADE	INTEGER	CKC_ID_ATIVIDADE_ATIVIDAD	X	
ID_TIPO_ATIVIDADES	SMALLINT	CKC_ID_TIPO_ATIVIDADE_ATIVIDAD		X
DESIGNACAO	VARCHAR2(100)	CKC_DESIGNACAO_ATIVIDAD		
DATA_INI	DATE	CKC_DATA_INI_ATIVIDAD		
DATA_FIM	DATE	CKC_DATA_FIM_ATIVIDAD		
LOCAL	VARCHAR2(255)	CKC_LOCAL_ATIVIDAD		
ESTADO_ATIVIDADE	NUMBER(1)	CKC_ESTADO_ATIVIDADE_ATIVIDAD		

Sec 7.1.8 - Tabela Recibos

Tabela que contém o id do recibos assim como a sua data de emissão

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_RECIBO	INTEGER	CKC_ID_RECIBO_RECIBOS	X	
ID_DONATIVO	INTEGER	CKC_ID_DONATIVO_RECIBOS		X
DATA_RECIBO	DATE	CKC_DATA_RECIBO_RECIBOS		

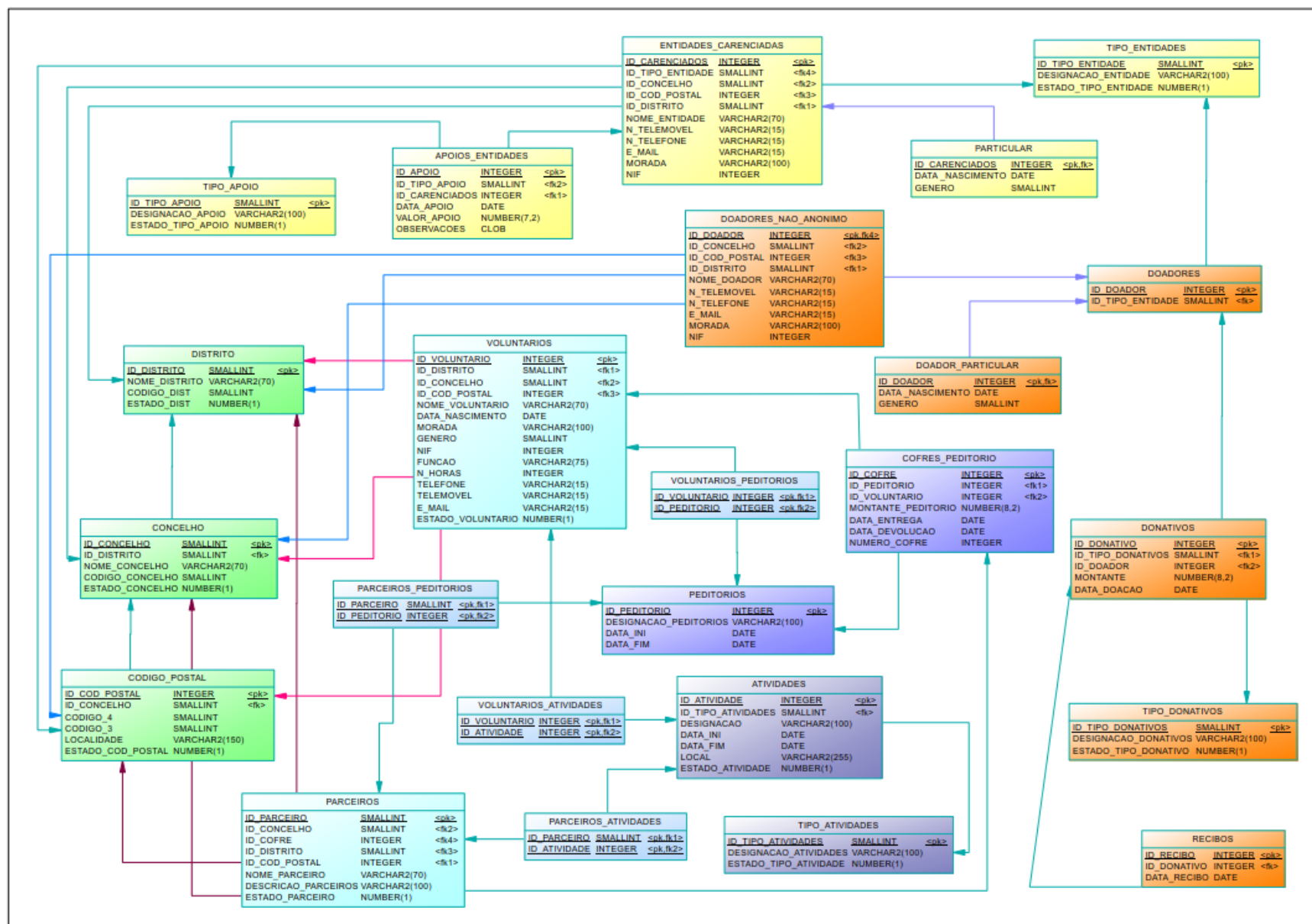
Sec 7.1.9 - Tabela Parceiros

Tabela que lista todos os parceiros

Name	Tipo de Dados	Check Constraint Name	Chave Primária	Chave Forasteira
ID_PARCEIRO	SMALLINT	CKC_ID_PARCEIRO_PARCEIRO	X	
ID_CONCELHO	SMALLINT	CKC_ID_CONCELHO_PARCEIRO		X
ID_COFRE	INTEGER	CKC_ID_COFRE_PARCEIRO		X
ID_DISTRITO	SMALLINT	CKC_ID_DISTRITO_PARCEIRO		X
ID_COD_POSTAL	INTEGER	CKC_ID_COD_POSTAL_PARCEIRO		X
NOME_PARCEIRO	VARCHAR2(70)	CKC_NOME_PARCEIRO_PARCEIRO		
DESCRICAO_PARCEIROS	VARCHAR2(100)	CKC_DESCRICAO_PARCEIR_PARCEIRO		
ESTADO_PARCEIRO	NUMBER(1)	CKC_ESTADO_PARCEIRO_PARCEIRO		

Sec 7.2 - Diagrama do Modelo Físico

O modelo Físico (ou diagrama de tabelas) completo com todas as tabelas é o seguinte.



Sec 7.3 - Script de criação da Base de Dados

Nesta secção apresentam-se as instruções SQL necessárias para criar as tabelas descritas anteriormente na secção 7.1 no SGBDR Oracle. O código SQL apresentado permite criar as tabelas, as restrições de integridade (chave primária, integridade referencial) suportadas pelo SGBD, assim como as validações de dados definidas e valores por omissão.

```
/*=====*/  
/* Table: APOIOS_ENTIDADES */  
/*=====*/
```

```
create table APOIOS_ENTIDADES  
(  
  ID_APOIO      INTEGER      not null,  
  ID_TIPO_APOIO SMALLINT     not null,  
  ID_CARENCIADOS INTEGER      not null,  
  DATA_APOIO   DATE         not null,  
  VALOR_APOIO   NUMBER(7,2)   not null,  
  OBSERVACOES   CLOB,  
  constraint PK_APOIOS_ENTIDADES primary key (ID_APOIO)  
);
```

```
/*=====*/  
/* Table: ATIVIDADES */  
/*=====*/
```

```
create table ATIVIDADES  
(
```

```

ID_ATIVIDADE      INTEGER      not null,
ID_TIPO_ATIVIDADES SMALLINT      not null,
DESIGNACAO        VARCHAR2(100)  not null,
DATA_INI          DATE           not null,
DATA_FIM          DATE           not null,
LOCAL            VARCHAR2(255)    not null,
ESTADO_ATIVIDADE  NUMBER(1)      not null,
constraint PK_ATIVIDADES primary key (ID_ATIVIDADE)
);

```

```

/*=====*/

```

```

/* Table: CODIGO_POSTAL          */

```

```

/*=====*/

```

```

create table CODIGO_POSTAL

```

```

(
  ID_COD_POSTAL      INTEGER      not null,
  ID_CONCELHO        SMALLINT      not null,
  CODIGO_4           SMALLINT      not null,
  CODIGO_3           SMALLINT      not null,
  LOCALIDADE         VARCHAR2(150) not null,
  ESTADO_COD_POSTAL  NUMBER(1)     not null,
  constraint PK_CODIGO_POSTAL primary key (ID_COD_POSTAL)
);

```

```

/*=====*/

```

```
/* Table: COFRES_PEDITORIO */
```

```
/*=====*/
```

```
create table COFRES_PEDITORIO
```

```
(
```

```
    ID_COFRE      INTEGER      not null,
```

```
    ID_PEDITORIO  INTEGER      not null,
```

```
    ID_VOLUNTARIO INTEGER      not null,
```

```
    MONTANTE_PEDITORIO NUMBER(8,2) not null,
```

```
    DATA_ENTREGA DATE          not null,
```

```
    DATA_DEVOLUCAO DATE,
```

```
    NUMERO_COFRE  INTEGER      not null,
```

```
    constraint PK_COFRES_PEDITORIO primary key (ID_COFRE)
```

```
);
```

```
/*=====*/
```

```
/* Table: CONCELHO */
```

```
/*=====*/
```

```
create table CONCELHO
```

```
(
```

```
    ID_CONCELHO    SMALLINT     not null,
```

```
    ID_DISTRITO    SMALLINT     not null,
```

```
    NOME_CONCELHO  VARCHAR2(70) not null,
```

```
    CODIGO_CONCELHO SMALLINT     not null,
```

```
    ESTADO_CONCELHO NUMBER(1)   not null,
```

```
constraint PK_CONCELHO primary key (ID_CONCELHO)
```

```
);
```

```
/*=====*/
```

```
/* Table: DISTRITO */
```

```
/*=====*/
```

```
create table DISTRITO
```

```
(
```

```
    ID_DISTRITO    SMALLINT        not null,
```

```
    NOME_DISTRITO  VARCHAR2(70)    not null,
```

```
    CODIGO_DIST    SMALLINT        not null,
```

```
    ESTADO_DIST    NUMBER(1)       not null,
```

```
    constraint PK_DISTRITO primary key (ID_DISTRITO)
```

```
);
```

```
/*=====*/
```

```
/* Table: DOADORES */
```

```
/*=====*/
```

```
create table DOADORES
```

```
(
```

```
    ID_DOADOR      INTEGER         not null,
```

```
    ID_TIPO_ENTIDADE SMALLINT      not null,
```

```
    constraint PK_DOADORES primary key (ID_DOADOR)
```

```
);
```

```
/*=====*/
```

```
/* Table: DOADORES_NAO_ANONIMO */
```

```
/*=====*/
```

```
create table DOADORES_NAO_ANONIMO
```

```
(
```

```
  ID_DOADOR      INTEGER      not null,
```

```
  ID_CONCELHO     SMALLINT,
```

```
  ID_COD_POSTAL   INTEGER,
```

```
  ID_DISTRITO     SMALLINT,
```

```
  NOME_DOADOR     VARCHAR2(70)  not null,
```

```
  N_TELEMOVEL     VARCHAR2(15),
```

```
  N_TELEFONE      VARCHAR2(15),
```

```
  E_MAIL          VARCHAR2(15),
```

```
  MORADA          VARCHAR2(100) not null,
```

```
  NIF             INTEGER      not null,
```

```
  constraint PK_DOADORES_NAO_ANONIMO primary key (ID_DOADOR)
```

```
);
```

```
/*=====*/
```

```
/* Table: DOADOR_PARTICULAR */
```

```
/*=====*/
```

```
create table DOADOR_PARTICULAR
```

```
(
```

```
ID_DOADOR      INTEGER      not null,  
DATA__NASCIMENTO  DATE        not null,  
GENERO          SMALLINT     not null,  
constraint PK_DOADOR_PARTICULAR primary key (ID_DOADOR)  
);
```

```
/*=====*/  
/* Table: DONATIVOS */  
/*=====*/
```

```
create table DONATIVOS  
(  
    ID_DONATIVO      INTEGER      not null,  
    ID_TIPO_DONATIVOS  SMALLINT     not null,  
    ID_DOADOR         INTEGER      not null,  
    MONTANTE          NUMBER(8,2)   not null,  
    DATA_DOACAO       DATE        not null,  
    constraint PK_DONATIVOS primary key (ID_DONATIVO)  
);
```

```
/*=====*/  
/* Table: ENTIDADES_CARENCIADAS */  
/*=====*/
```

```
create table ENTIDADES_CARENCIADAS
```

```
(
  ID_CARENCIADOS    INTEGER      not null,
  ID_TIPO_ENTIDADE  SMALLINT     not null,
  ID_CONCELHO       SMALLINT     not null,
  ID_COD_POSTAL     INTEGER      not null,
  ID_DISTRITO       SMALLINT     not null,
  NOME_ENTIDADE     VARCHAR2(70)  not null,
  N_TELEMOVEL       VARCHAR2(15),
  N_TELEFONE        VARCHAR2(15),
  E_MAIL            VARCHAR2(15),
  MORADA            VARCHAR2(100)  not null,
  NIF               INTEGER      not null,
  constraint PK_ENTIDADES_CARENCIADAS primary key (ID_CARENCIADOS)
);
```

```
/*=====*/
```

```
/* Table: PARCEIROS */
```

```
/*=====*/
```

```
create table PARCEIROS
```

```
(
  ID_PARCEIRO       SMALLINT     not null,
  ID_CONCELHO       SMALLINT     not null,
  ID_COFRE          INTEGER      not null,
  ID_DISTRITO       SMALLINT     not null,
```

```
ID_COD_POSTAL    INTEGER        not null,  
NOME_PARCEIRO    VARCHAR2(70)    not null,  
DESCRICAO_PARCEIROS VARCHAR2(100),  
ESTADO_PARCEIRO  NUMBER(1)      not null,  
constraint PK_PARCEIROS primary key (ID_PARCEIRO)  
);
```

```
/*=====*/  
/* Table: PARCEIROS_ATIVIDADES          */  
/*=====*/
```

```
create table PARCEIROS_ATIVIDADES  
(  
    ID_PARCEIRO    SMALLINT      not null,  
    ID_ATIVIDADE    INTEGER       not null,  
    constraint PK_PARCEIROS_ATIVIDADES primary key (ID_PARCEIRO, ID_ATIVIDADE)  
);
```

```
/*=====*/  
/* Table: PARCEIROS_PEDITORIOS          */  
/*=====*/
```

```
create table PARCEIROS_PEDITORIOS  
(  
    ID_PARCEIRO    SMALLINT      not null,  
    ID_PEDITORIO    INTEGER       not null,
```



```
constraint PK_PARCEIROS_PEDITORIOS primary key (ID_PARCEIRO, ID_PEDITORIO)
```

```
);
```

```
/*=====*/
```

```
/* Table: PARTICULAR */
```

```
/*=====*/
```

```
create table PARTICULAR
```

```
(
```

```
  ID_CARENCIADOS    INTEGER        not null,
```

```
  DATA__NASCIMENTO DATE          not null,
```

```
  GENERO            SMALLINT       not null,
```

```
  constraint PK_PARTICULAR primary key (ID_CARENCIADOS)
```

```
);
```

```
/*=====*/
```

```
/* Table: PEDITORIOS */
```

```
/*=====*/
```

```
create table PEDITORIOS
```

```
(
```

```
  ID_PEDITORIO      INTEGER        not null,
```

```
  DESIGNACAO_PEDITORIOS VARCHAR2(100) not null,
```

```
  DATA_INI         DATE          not null,
```

```
  DATA_FIM         DATE          not null,
```

```
  constraint PK_PEDITORIOS primary key (ID_PEDITORIO)
```

```
);
```

```
/*=====*/
```

```
/* Table: RECIBOS */
```

```
/*=====*/
```

```
create table RECIBOS
```

```
(
```

```
  ID_RECIBO      INTEGER      not null,
```

```
  ID_DONATIVO    INTEGER      not null,
```

```
  DATA_RECIBO   DATE         not null,
```

```
  constraint PK_RECIBOS primary key (ID_RECIBO)
```

```
);
```

```
/*=====*/
```

```
/* Table: TIPO_APOIO */
```

```
/*=====*/
```

```
create table TIPO_APOIO
```

```
(
```

```
  ID_TIPO_APOIO  SMALLINT     not null,
```

```
  DESIGNACAO_APOIO VARCHAR2(100) not null,
```

```
  ESTADO_TIPO_APOIO NUMBER(1) not null,
```

```
  constraint PK_TIPO_APOIO primary key (ID_TIPO_APOIO)
```

```
);
```

```
/*=====*/
```

```
/* Table: TIPO_ATIVIDADES */
```

```
/*=====*/
```

```
create table TIPO_ATIVIDADES
```

```
(
```

```
    ID_TIPO_ATIVIDADES SMALLINT      not null,
```

```
    DESIGNACAO_ATIVIDADES VARCHAR2(100) not null,
```

```
    ESTADO_TIPO_ATIVIDADE NUMBER(1) not null,
```

```
    constraint PK_TIPO_ATIVIDADES primary key (ID_TIPO_ATIVIDADES)
```

```
);
```

```
/*=====*/
```

```
/* Table: TIPO_DONATIVOS */
```

```
/*=====*/
```

```
create table TIPO_DONATIVOS
```

```
(
```

```
    ID_TIPO_DONATIVOS SMALLINT      not null,
```

```
    DESIGNACAO_DONATIVOS VARCHAR2(100) not null,
```

```
    ESTADO_TIPO_DONATIVO NUMBER(1) not null,
```

```
    constraint PK_TIPO_DONATIVOS primary key (ID_TIPO_DONATIVOS)
```

```
);
```

```
/*=====*/
```

```
/* Table: TIPO_ENTIDADES          */
/*=====*/

create table TIPO_ENTIDADES
(
  ID_TIPO_ENTIDADE  SMALLINT      not null,
  DESIGNACAO_ENTIDADE VARCHAR2(100) not null,
  ESTADO_TIPO_ENTIDADE NUMBER(1)   not null,
  constraint PK_TIPO_ENTIDADES primary key (ID_TIPO_ENTIDADE)
);

/*=====*/
/* Table: VOLUNTARIOS            */
/*=====*/

create table VOLUNTARIOS
(
  ID_VOLUNTARIO    INTEGER      not null,
  ID_DISTrito     SMALLINT      not null,
  ID_CONCELHO      SMALLINT      not null,
  ID_COD_POSTAL    INTEGER      not null,
  NOME_VOLUNTARIO  VARCHAR2(70)  not null,
  DATA_NASCIMENTO DATE          not null,
  MORADA           VARCHAR2(100),
  GENERO           SMALLINT      not null,
  NIF              INTEGER,
  FUNCAO           VARCHAR2(75)  not null,
```

```
N_HORAS      INTEGER      not null,  
TELEFONE      VARCHAR2(15),  
TELEMOVEL     VARCHAR2(15),  
E_MAIL        VARCHAR2(15),  
ESTADO_VOLUNTARIO  NUMBER(1)      not null,  
constraint PK_VOLUNTARIOS primary key (ID_VOLUNTARIO)
```

```
);
```

```
/*=====*/
```

```
/* Table: VOLUNTARIOS_ATIVIDADES          */
```

```
/*=====*/
```

```
create table VOLUNTARIOS_ATIVIDADES
```

```
(
```

```
  ID_VOLUNTARIO    INTEGER      not null,
```

```
  ID_ATIVIDADE     INTEGER      not null,
```

```
  constraint PK_VOLUNTARIOS_ATIVIDADES primary key (ID_VOLUNTARIO, ID_ATIVIDADE)
```

```
);
```

```
/*=====*/
```

```
/* Table: VOLUNTARIOS_PEDITORIOS          */
```

```
/*=====*/
```

```
create table VOLUNTARIOS_PEDITORIOS
```

```
(
```

```
  ID_VOLUNTARIO    INTEGER      not null,
```

```
ID_PEDITORIO    INTEGER    not null,  
constraint PK_VOLUNTARIOS_PEDITORIOS primary key (ID_VOLUNTARIO, ID_PEDITORIO)  
);
```

```
alter table APOIOS_ENTIDADES  
add constraint FK_APOIOS_E_APOIO_AS__ENTIDADE foreign key (ID_CARENCIADOS)  
references ENTIDADES_CARENCIADAS (ID_CARENCIADOS)
```

```
alter table APOIOS_ENTIDADES  
add constraint FK_APOIOS_E_TIPOS_DE__TIPO_APO foreign key (ID_TIPO_APOIO)  
references TIPO_APOIO (ID_TIPO_APOIO)
```

```
alter table ATIVIDADES  
add constraint FK_ATIVIDAD_ATIVIDADE_TIPO_ATI foreign key (ID_TIPO_ATIVIDADES)  
references TIPO_ATIVIDADES (ID_TIPO_ATIVIDADES)
```

```
alter table CODIGO_POSTAL  
add constraint FK_CODIGO_P_CONCELHO__CONCELHO foreign key (ID_CONCELHO)  
references CONCELHO (ID_CONCELHO)
```

```
alter table COFRES_PEDITORIO  
add constraint FK_COFRES_P_COFRES_AT_VOLUNTAR foreign key (ID_VOLUNTARIO)  
references VOLUNTARIOS (ID_VOLUNTARIO)
```

```
alter table COFRES_PEDITORIO  
add constraint FK_COFRES_P_TEM_PEDITORI foreign key (ID_PEDITORIO)
```

```
references PEDITORIOS (ID_PEDITORIO)
```

```
alter table CONCELHO
```

```
add constraint FK_CONCELHO_DISTRITO__DISTRITO foreign key (ID_DISTRITO)  
references DISTRITO (ID_DISTRITO)
```

```
alter table DOADORES
```

```
add constraint FK_DOADORES_TIPO_ENTI_TIPO_ENT foreign key (ID_TIPO_ENTIDADE)  
references TIPO_ENTIDADES (ID_TIPO_ENTIDADE)
```

```
alter table DOADORES_NAO_ANONIMO
```

```
add constraint FK_DOADORES_CODIGOPOS_CODIGO_P foreign key (ID_COD_POSTAL)  
references CODIGO_POSTAL (ID_COD_POSTAL)
```

```
alter table DOADORES_NAO_ANONIMO
```

```
add constraint FK_DOADORES_CONCELHO__CONCELHO foreign key (ID_CONCELHO)  
references CONCELHO (ID_CONCELHO)
```

```
alter table DOADORES_NAO_ANONIMO
```

```
add constraint FK_DOADORES_DISTRITO__DISTRITO foreign key (ID_DISTRITO)  
references DISTRITO (ID_DISTRITO)
```

```
alter table DOADORES_NAO_ANONIMO
```

```
add constraint FK_DOADORES_HERANCA_3_DOADORES foreign key (ID_DOADOR)  
references DOADORES (ID_DOADOR)
```

```
alter table DOADOR_PARTICULAR
```

```
add constraint FK_DOADOR_P_HERANCA_2_DOADORES foreign key (ID_DOADOR)  
references DOADORES (ID_DOADOR)
```

```
alter table DONATIVOS
```

```
add constraint FK_DONATIVO_DOADORES__DOADORES foreign key (ID_DOADOR)  
references DOADORES (ID_DOADOR)
```

```
alter table DONATIVOS
```

```
add constraint FK_DONATIVO_DONATIVO__TIPO_DON foreign key (ID_TIPO_DONATIVOS)  
references TIPO_DONATIVOS (ID_TIPO_DONATIVOS)
```

```
alter table ENTIDADES_CARENCIADAS
```

```
add constraint FK_ENTIDADE_CODIGOPOS_CODIGO_P foreign key (ID_COD_POSTAL)  
references CODIGO_POSTAL (ID_COD_POSTAL)
```

```
alter table ENTIDADES_CARENCIADAS
```

```
add constraint FK_ENTIDADE_CONCELHO__CONCELHO foreign key (ID_CONCELHO)  
references CONCELHO (ID_CONCELHO)
```

```
alter table ENTIDADES_CARENCIADAS
```

```
add constraint FK_ENTIDADE_DISTRITO__DISTRITO foreign key (ID_DISTRITO)  
references DISTRITO (ID_DISTRITO)
```

```
alter table ENTIDADES_CARENCIADAS
```

```
add constraint FK_ENTIDADE_TIPOS_DE__TIPO_ENT foreign key (ID_TIPO_ENTIDADE)
```



```
references TIPO_ENTIDADES (ID_TIPO_ENTIDADE)
```

```
alter table PARCEIROS
```

```
add constraint FK_PARCEIRO_CODIGOPOS_CODIGO_P foreign key (ID_COD_POSTAL)  
references CODIGO_POSTAL (ID_COD_POSTAL)
```

```
alter table PARCEIROS
```

```
add constraint FK_PARCEIRO_COFRES_AT_COFRES_P foreign key (ID_COFRE)  
references COFRES_PEDITORIO (ID_COFRE)
```

```
alter table PARCEIROS
```

```
add constraint FK_PARCEIRO_CONCELHO__CONCELHO foreign key (ID_CONCELHO)  
references CONCELHO (ID_CONCELHO)
```

```
alter table PARCEIROS
```

```
add constraint FK_PARCEIRO_DISTRITO__DISTRITO foreign key (ID_DISTRITO)  
references DISTRITO (ID_DISTRITO)
```

```
alter table PARCEIROS_ATIVIDADES
```

```
add constraint FK_PARCEIRO_COLABORAM_PARCEIRO foreign key (ID_PARCEIRO)  
references PARCEIROS (ID_PARCEIRO)
```

```
alter table PARCEIROS_ATIVIDADES
```

```
add constraint FK_PARCEIRO_COLABORAM_ATIVIDAD foreign key (ID_ATIVIDADE)  
references ATIVIDADES (ID_ATIVIDADE)
```

```
alter table PARCEIROS_PEDITORIOS
```

```
add constraint FK_PARCEIRO_PARTICIPA_PARCEIRO foreign key (ID_PARCEIRO)  
references PARCEIROS (ID_PARCEIRO)
```

```
alter table PARCEIROS_PEDITORIOS
```

```
add constraint FK_PARCEIRO_PARTICIPA_PEDITORIO foreign key (ID_PEDITORIO)  
references PEDITORIOS (ID_PEDITORIO)
```

```
alter table PARTICULAR
```

```
add constraint FK_PARTICUL_HERANCA_1_ENTIDADE foreign key (ID_CARENCIADOS)  
references ENTIDADES_CARENCIADAS (ID_CARENCIADOS)
```

```
alter table RECIBOS
```

```
add constraint FK_RECIBOS_RECIBOS_D_DONATIVO foreign key (ID_DONATIVO)  
references DONATIVOS (ID_DONATIVO)
```

```
alter table VOLUNTARIOS
```

```
add constraint FK_VOLUNTAR_CODIGOPOS_CODIGO_P foreign key (ID_COD_POSTAL)  
references CODIGO_POSTAL (ID_COD_POSTAL)
```

```
alter table VOLUNTARIOS
```

```
add constraint FK_VOLUNTAR_CONCELHO__CONCELHO foreign key (ID_CONCELHO)  
references CONCELHO (ID_CONCELHO)
```

```
alter table VOLUNTARIOS
```

```
add constraint FK_VOLUNTAR_DISTRITO__DISTRITO foreign key (ID_DISTRITO)
```

```
references DISTRITO (ID_DISTRITO)
```

```
alter table VOLUNTARIOS_ATIVIDADES
```

```
add constraint FK_VOLUNTAR_PARTICIPA_ATIVIDAD foreign key (ID_ATIVIDADE)
```

```
references ATIVIDADES (ID_ATIVIDADE)
```

```
alter table VOLUNTARIOS_ATIVIDADES
```

```
add constraint FK_VOLUNTAR_VOLUNTARI_VOLUNTAR foreign key (ID_VOLUNTARIO)
```

```
references VOLUNTARIOS (ID_VOLUNTARIO)
```

```
alter table VOLUNTARIOS_PEDITORIOS
```

```
add constraint FK_VOLUNTAR_PEDITORIO_PEDITORI foreign key (ID_PEDITORIO)
```

```
references PEDITORIOS (ID_PEDITORIO)
```

```
alter table VOLUNTARIOS_PEDITORIOS
```

```
add constraint FK_VOLUNTAR_VOL_PART__VOLUNTAR foreign key (ID_VOLUNTARIO)
```

```
references VOLUNTARIOS (ID_VOLUNTARIO)
```

Cap 8 - Pesquisas SQL

Enuncie as pesquisas mais relevantes, os módulos a que estão associadas e o seu contexto de utilização.

Escreva-as em SQL.

Por exemplo:

Módulo de Listagens Periódicas

Q1 - Volume de vendas de cada livro no mês corrente, ordenado pelo volume de vendas (os livros que mais venderam devem ser apresentados em primeiro).

```
SELECT TITULO, SUM(PRECO_UNITARIO * QUANTIDADE) "VENDAS DO MES"  
FROM LIVROS L, VENDAS V  
WHERE L.CODIGO_LIVRO = V.CODIGO_LIVRO  
AND TO_CHAR(DATA_VENDA, 'MM-YYYY') = TO_CHAR(SYSDATE, 'MM-YYYY')  
GROUP BY TITULO  
ORDER BY "VENDAS DO MES" DESC
```

Q1 - Mostra a atividade realizada pelo voluntário António Mendes, que teve lugar no distrito de Leiria.

```
SELECT designacao,  
FROM voluntarios, atividades, voluntarios_atividades, distrito  
WHERE distrito.id_distrito= voluntarios.id_distrito  
AND voluntarios.id_voluntario=voluntarios_atividades.id_voluntario  
AND voluntarios_atividades.id_atividade=atividades.id_atividade  
AND upper(distrito)=LEIRIA;
```

Q2 - Mostre o nome, o id e a função relativa aos voluntários que participaram na Corrida Solidária Contra ao Cancro, que teve início no dia 25/11/2017.

```
SELECT nome_voluntario, id_voluntario, funcao  
FROM voluntarios, voluntarios_atividades, atividades  
WHERE voluntarios.id_voluntario = voluntarios_atividades.id_voluntario  
AND voluntarios_atividades.id_atividade = atividades.id_atividade  
AND lower(designacao) = "corrida solidária contra ao cancro"  
AND to_char(data_inicio, 'DD/MM/YYYY') = "25/11/2017";
```

Q3 - Mostre os montantes doados, por ordem crescente, pelos doadores do distrito de Coimbra.

```
SELECT montante  
FROM donativos, doadores, doadores_nao_anonimo, distrito
```

```
WHERE distrito.id_distrito = doadores_ao_anonimo.id_distrito  
AND doadores_ao_anonimo.id_doador = doadores.id_doador  
AND doadores.id_doador = donativos.id_doador  
AND lower(nome_distrito) = "coimbra"  
ORDER BY montante;
```

Q4 - Mostra o voluntário que trabalha há mais tempo na associação

```
SELECT nome_voluntario  
FROM voluntarios  
WHERE n_horas=(SELECT max(n_horas)  
                FROM voluntarios);
```

Q5 - Mostra o tipo de apoio recebido pelas entidades carenciadas, ordenado alfabeticamente pelo nome da entidade

```
SELECT nome_entidade, designacao_apoio  
WHERE tipo_apoio.id_tipo_apoio=apoios_entidades.id_tipo_apoio  
AND apoios_entidades.id_carenciados=entidades_carenciadas.id_carenciados  
ORDER BY nome_entidade;
```

Cap 9 - Conclusões

Este projeto permitiu-nos perceber melhor como funciona o desenvolvimento de uma base de dados quando relacionada com um problema real, desde a lógica necessária até aos comandos de SQL específicos.

A nossa base de dados tem o objetivo de facilitar o armazenamento de dados para uma instituição de voluntariado, tornando-a o mais competente possível. Dado o aumento de pessoas e entidades carenciadas, devido também à pandemia que enfrentamos, pensamos que a criação deste novo sistema é relevante.

