



U N I V E R S I D A D E  
**LUSÓFONA**

ULHT - Universidade Lusófona de Humanidades e Tecnologias

LEI - Licenciatura em Engenharia Informática



**CRM: Customer Relationship Management**

**Gestão de Clinicas Veterinárias**

Elaborado Por:

Pedro Miguel dos Reis Calado Garcia

Aluno nº20083916

Orientador TFC: Rui Ribeiro

## Índice

<b>Resumo.....</b>	<b>Pág.2</b>
<b>Abstract.....</b>	<b>Pág.3</b>
<b>Introdução.....</b>	<b>Pág.4</b>
<b>Tecnologias.....</b>	<b>Pág.5</b>
Tecnologias de Suporte.....	Pág.5
Virtual Box.....	Pág.5
Linux Ubuntu.....	Pág.5
Eclipse Kepler.....	Pág.6
Odoo.....	Pág.6
PostgreSQL.....	Pág.8
<b>Método.....</b>	<b>Pág.9</b>
Requisitos.....	Pág.9
Módulos Utilizados.....	Pág.11
Base de dados.....	Pág.12
Pagina principal.....	Pág.13
Pagina eventos.....	Pág.13
BackOffice.....	Pág.14
Pagina Animal.....	Pág.14
Pagina Clientes.....	Pág.15
Pagina Faturas.....	Pág.15
Pagina Campanhas de Marketing.....	Pág.16
Pagina Telefonemas.....	Pág.16
Pagina Artigos.....	Pág.17
Gestão de Utilizadores.....	Pág.17
Eventos.....	Pág.18
<b>Conclusão.....</b>	<b>Pág.19</b>
<b>Bibliografia.....</b>	<b>Pág.20</b>
<b>Anexos.....</b>	<b>Pág.21</b>

## Resumo

O objetivo deste trabalho consistiu em estudar as potencialidades do Odoo, num desenvolvimento de um CRM de gestão de clínicas veterinárias através das tecnologias disponibilizadas pelo Odoo como as linguagens de programação adjacentes, o Python, HTML, CSS, JavaScript e PostgreSQL. Este CRM incide sobre a relação da instituição prestadora de serviço com os seus clientes, de forma personalizada através de um conjunto de interações implementadas no projeto com objetivo de auxiliar as clínicas veterinárias a angariar e fidelizar novos clientes ou atuais na busca de atingir satisfação total, através do melhor entendimento das suas necessidades e expectativas.

## Abstract

The purpose of this final project was consisted in study of Odoo capabilities in a development of a veterinary clinical management CRM through technology that provided by Odoo as the adjacent programming languages like Python, HTML, CSS, JavaScript and PostgreSQL. This CRM focuses on the relationship between the service provider institution with its customers in a personalized way through a set of interactions implemented in the project in order to assist veterinary clinics to raise and retain new customers or current in the quest to achieve total satisfaction, through better understanding of their needs and expectations.

## Introdução

Este trabalho foi desenvolvido no âmbito da disciplina de Trabalho Final de Curso e consistiu no aprofundamento das tecnologias utilizadas em Odoo, que ao longo dos últimos anos se tem vindo a assistir cada vez mais uma grande aposta neste conceito, tanto para pequenas empresas como para grandes, tendo como exemplo a Auchan, a Danone ou a Singer, grandes empresas que hoje em dia gerem o seu negócio recorrendo ao Odoo.

O Odoo é uma Plataforma web, de simples utilização e acessível a partir de qualquer browser (Mozilla Firefox, Google Chrome, Opera, Safari), contém uma framework que lida com solicitações do lado do cliente, usa base de dados Postgresql para guardar os dados (todas as tabelas são criadas automaticamente pelo Odoo) e é constituído por módulos que implementam toda a lógica do negócio em linguagem Python e as vistas em XML.

Para este projeto foi escolhido Odoo para a criação de um CRM para clínicas veterinárias, onde o objetivo principal é estabelecer relações personalizadas e duradouras com os clientes, e alimenta-las permanentemente através de um conjunto de interações de marketing que sejam relevantes e acrescentem valor.

Esta ambição obriga a criar e a gerir um sistema de informação de clientes capaz de sustentar todos os tipos de contactos que são estabelecidos com os clientes:

- Contactos inbound, despoletados pelo próprio cliente (por telefone, email, website e presencialmente);
- Campanhas outbound de iniciativa da clinica veterinária (maioritariamente email-marketing e telemarketing)

Um sistema de informação de clientes robusto permitirá tratar todos os clientes de modo relevante e personalizado, adequando a oferta ao perfil de cada um, através do melhor entendimento das suas necessidades e expectativas.

## Tecnologias

Neste capítulo são apresentadas algumas tecnologias de suporte utilizadas para o desenvolvimento deste projeto.

### Tecnologias de Suporte

Relativamente as tecnologias foi utilizado o SO Linux Ubuntu a correr numa máquina virtual pelo Virtual Box da Oracle, IDE Eclipse foi utilizada para o desenvolvimento a linguagem Python e o XML para a construção de módulos de que implementam toda a lógica do negócio e implementa as vistas e para a base de dados da aplicação foi utilizado o PostgreSQL, para guardar os dados (todas as tabelas são criadas automaticamente pelo Odoo) e por fim uma aplicação JavaScript que Comunica com o servidor e exibe a interface gráfica do user.

#### Virtual Box

É um criador/gerenciador de máquinas virtuais que pode ser utilizado em diversos sistemas operativos como Linux, OS X, Windows XP, Solaris e OpenSolaris. Utilizado no projeto para correr o Linux.

#### Linux Ubuntu

É sistema operativo open source, construído a partir do núcleo do Linux, baseado no Debian. Utilizado no projeto para instalar o Odoo, Eclipse, PostgreSQL e servir de servidor.

## Eclipse

Eclipse é um IDE para desenvolvimento java, que suporta várias outras linguagens a partir de plugins como C/C++,PHP/Python e plataforma Androide. Segue o modelo open source. Atualmente faz parte do kit de desenvolvimento de software recomendado para developers Androide. Foi utilizado no projeto para criação e edição de módulos em Python.

## Odoo

É uma solução de gestão empresarial ERP completo, com um sistema CRM. É baseado na arquitetura MVC e implementa um cliente e um servidor, sendo a comunicação entre o cliente e o servidor por interface XML-RPC. O Software é open source.

- **Acesso Web**

Plataforma web, de simples utilização e acessível a partir de qualquer browser (Mozilla, Firefox, Google Chrome, Opera, Safari).

- **Gestão Documental Integrada**

Permite anexar qualquer tipo de documento a qualquer tipo de objeto do Odoo, de uma forma simples e centralizada, permitindo igualmente a gestão de pastas e controlo de acesso aos documentos.

- **Reporting Integrado**

Possibilidade de criação de relatórios customizáveis sobre toda a informação constante no ERP, através de uma ferramenta de criação de reports.

- **Motor de Business Intelligence**

Permite a fácil criação e modificação de dashboards de controlo de processos, de SLA's e de KPI's, com drilldown sobre a informação e manipulação de cubos, permitindo acesso rápido a todos os dados de negócio constantes no ERP.

- **Gestão de Permissões**

Gestão integrada de permissões, através da criação de grupos e perfis de utilização, possibilitando a configuração manual de acesso personalizado a todos os elementos do sistema, tais como objetos, ações, dados, relatórios, vistas.

- **Integração com Email**

Integração dos workflows com email, permitindo a configuração de ações condicionais e procedimentos de execução automatizados para cada passo/transição dos diversos workflows existentes.

- **Base de Dados Postgres**

Ao utilizar uma base de dados Postgres SQL permite a redução do custo total de operação do sistema, mantendo todas as garantias de fiabilidade e segurança da informação.

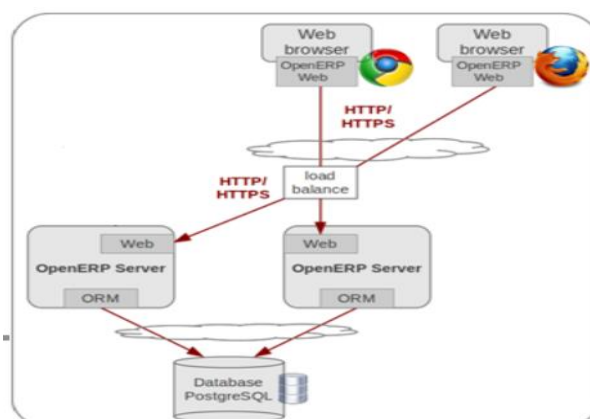
- **Importação/Exportação**

Importa e exporta informação de uma série de fontes de dados, tais como ficheiros CSV, ficheiros XML, tabelas Excel.

- **Arquitetura Odoo**

Arquitetura Cliente - Servidor, contendo 3 camadas:

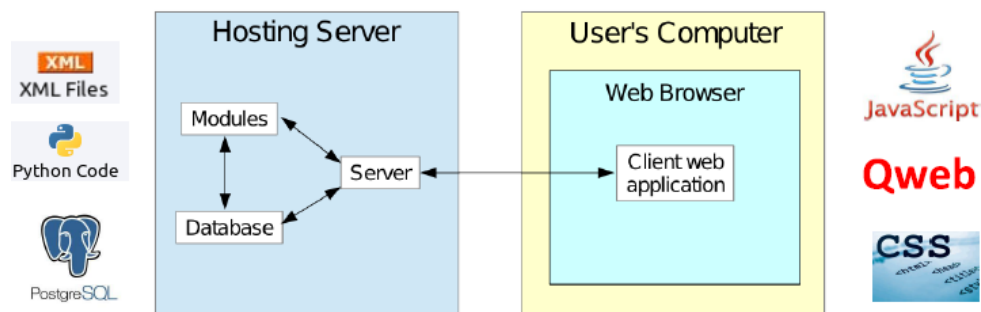
- Componente da interface do utilizador que permite a comunicação com o servidor
- Componente aplicacional onde contem toda a lógica de negócio
- Componente de armazenamento de dados





- **Tecnologias Odoo**

- Servidor: contém a framework que lida com solicitações do lado Cliente
- Database: usado o Postgresql para guardar os dados (todas as tabelas são criadas automaticamente pelo Odoo)
- Modules: Módulos que implementam toda a lógica do negócio em linguagem Python e as vistas em xml. É nesta camada que está presente o ORM (gere todas as operações de um objeto).
- Client web application: Comunica com o servidor e exibe a interface gráfica do user através de uma aplicação Javascript



## PostgreSQL

É um sistema gerenciador de bases de dados open source dos mais avançados, contando com recursos como:

- Consultas complexas
- Foreign Key
- Integridade transacional
- Controle de concorrência multi-versão
- Suporte ao modelo híbrido objeto-relacional
- Triggers
- Views
- Stored Procedure
- Indexação por texto

## Método

Inicialmente houve uma conversa com o professor Rui Ribeiro (orientador de projeto), onde se falou um pouco o que seria o Odoo e a melhor forma de estudar esta nova tecnologia dado estar a decorrer na universidade uma ação de formação da Think Open sobre a tecnologia Odoo, na qual foi necessário estar inscrito e agendou-se uma reunião com o potencial cliente para a definição de requisitos.

Ficou definido que a abordagem consistia em focar o projeto no estabelecimento de relações personalizadas e duradouras com os clientes, e alimenta-las permanentemente através de um conjunto de interações de marketing que fossem relevantes e acrescentassem valor através da criação de uma solução em Odoo para gerir um sistema de informação de clientes capaz de sustentar todos os tipos de contactos que são estabelecidos com os clientes.

## Requisitos

Para cumprir os objetivos traçados em termos de conquista, satisfação e fidelização de clientes, definiu-se os requisitos:

- Base de dados deverá reunir toda a informação útil relacionada com cada cliente e animal.
- Base de dados para clientes

	Dono 1	Dono 2	Outros / agregado
Nome			2 filhos 7 e 11 anos
NIF			Avó 85 anos visita frequente
Telefone			
Email			
Morada			
Profissão			
Data nasc.			
Idade			
Perfil psicográfico	O Acº é considerado membro da família, vivem em vivenda, praticam actividades na natureza, interessado em receber info s/ seminários e workshops; etc.		
Data registo	1º registo do cliente		

- Base de dados para animais

	Pet 1	Pet 2	... Pet x
Espécie	cão	Gato	
Nome	Baloo	Dentuça	
Nº identificação	xxxxxxx	yyyyyyyy	
Raça	Labrador	rafeiro	
Data nasc.			
Idade			
Socializou? Desde que idade?			
Episódios de agressividade?			
Fobias?			
Rotinas diárias?			
Férias?			

- Ligação dono » animal e animal » dono
- Base de dados para produtos do veterinário
- Ferramentas que concorram para a automatização de ações segmentadas de marketing
- Contactos inbound, despoletados pelo próprio cliente (por telefone, email, website, redes sociais e presencialmente);
- Campanhas outbound de iniciativa da clinica veterinária (maioritariamente email-marketing e telemarketing)
- Criação de eventos
- Possibilidade de extrair informação analítica relevante para avaliação do desempenho do negócio

	Total (todos pets)	Pet 1	Pet 2	Pet x
Nº total de serviços comprados ( $\Sigma$ )	12	6	6	...
Workshops & semin		2015.06.01 2015.09.12 2015.10.05		
Treino individual				
Treino domicilio			2015.06.01 2015.09.12 2015.10.05	
Socialização			2015.06.01 2015.09.12 2015.10.05	
Consultas		2015.06.01 2015.09.12 2015.10.05		
Terapias				
Satisfação	Histórico de resultados dos inquéritos (+ média) e de reclamações			

- Sistema de faturação
- Gestão de inscrição em eventos

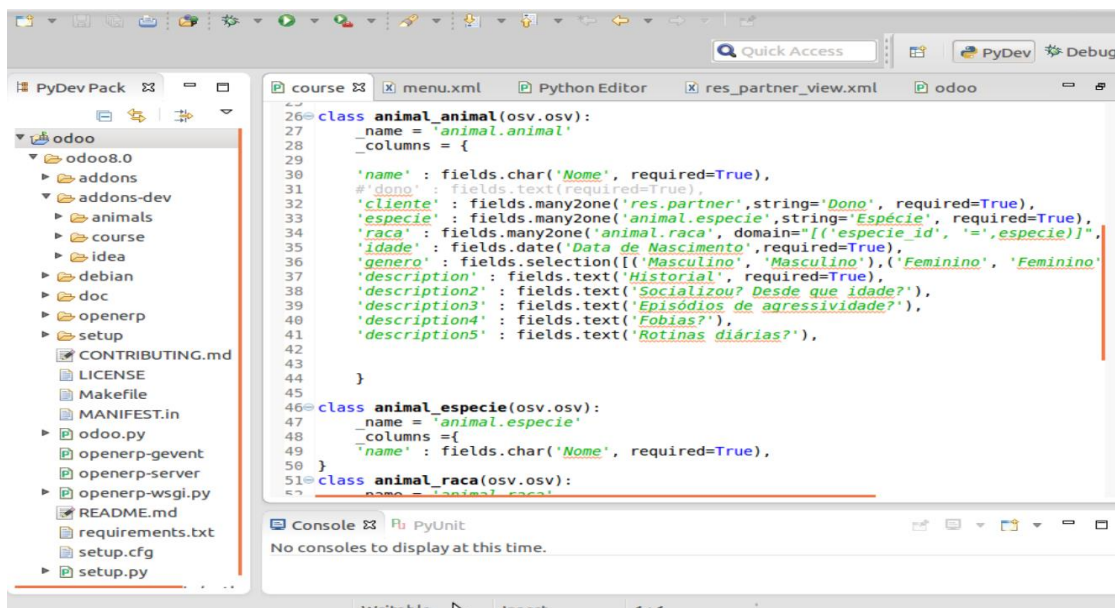
### Módulos utilizados

Apos instalação do Odoo selecionou-se os modelos que seriam necessários para atender os requisitos, o Odoo tem mais de 4000 módulos disponíveis mas é necessário a criação de módulos adicionais aos existentes para colmatar as lacunas do negocio mais especificas.

Para o projeto foram utilizados os módulos:

- Crm
- Rede social
- Online billing
- Gestão de vendas
- Instant messaging
- Calendário
- Animal
- Website builder
- Online events
- Organização de eventos
- Vendas dos eventos
- Portugal – chart accounts
- Mass mailing campaigns
- Portal
- Faturação electronica

Criou-se um módulo adaptado às necessidades do negócio como o exemplo do módulo animal de modo a que o Odoo criasse tabelas na base de dados para armazenar todos os dados de informação do animal bem como toda a sua integração com o cliente e restantes módulos.



Código Python na criação do módulo animal

## Base de Dados

Utilizou-se o PostgreSQL para guardar os dados, onde a partir da instalação dos módulos, todas as tabelas foram criadas automaticamente pelo Odoo de modo a guardar toda a informação necessária ao funcionamento de uma clínica veterinária.

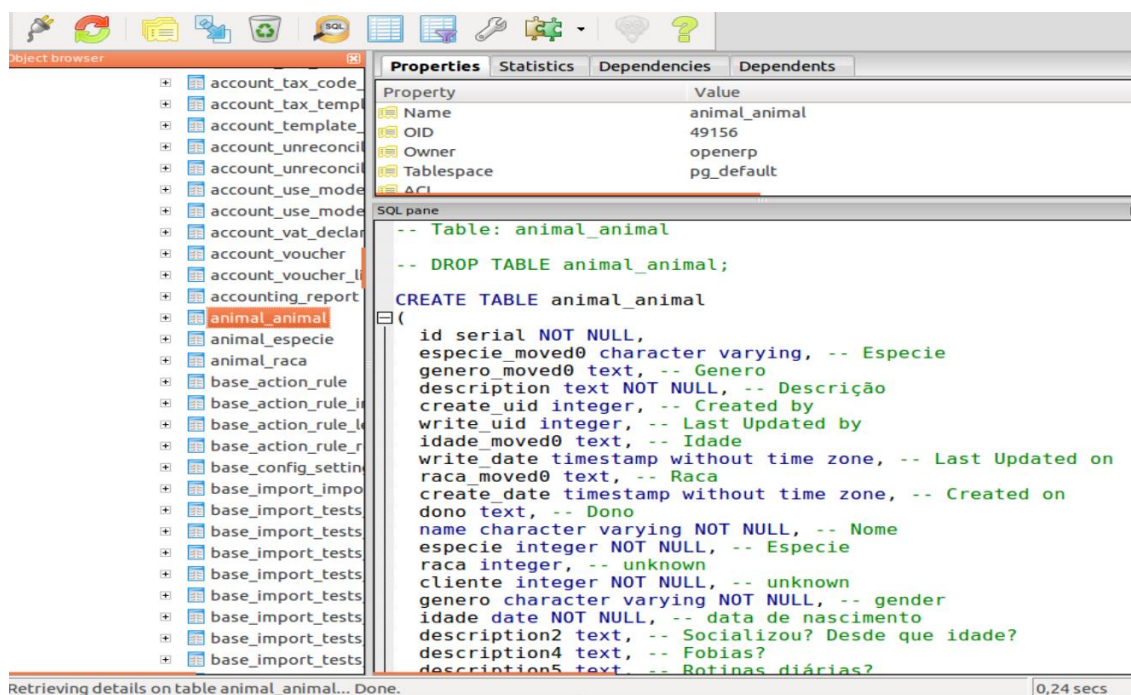
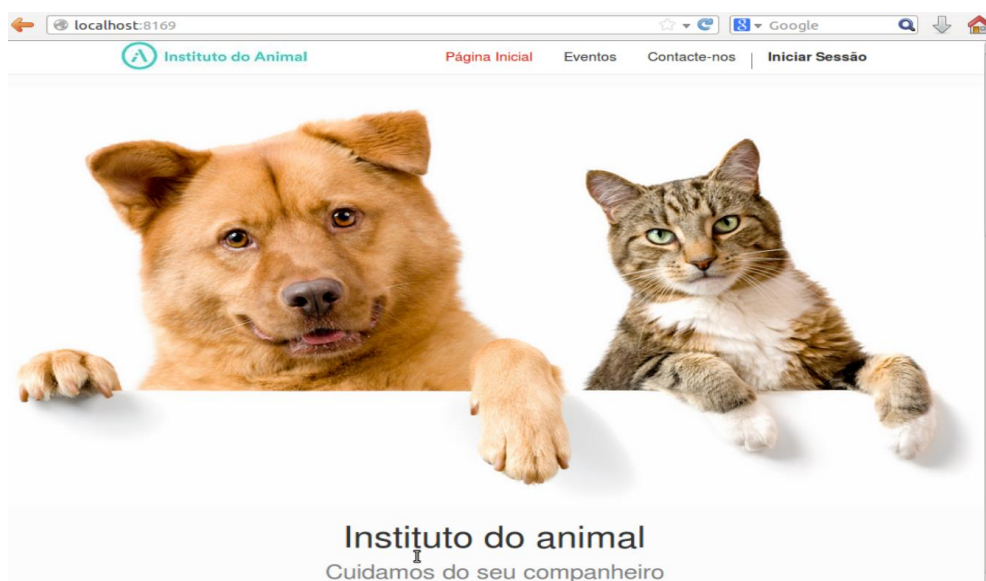


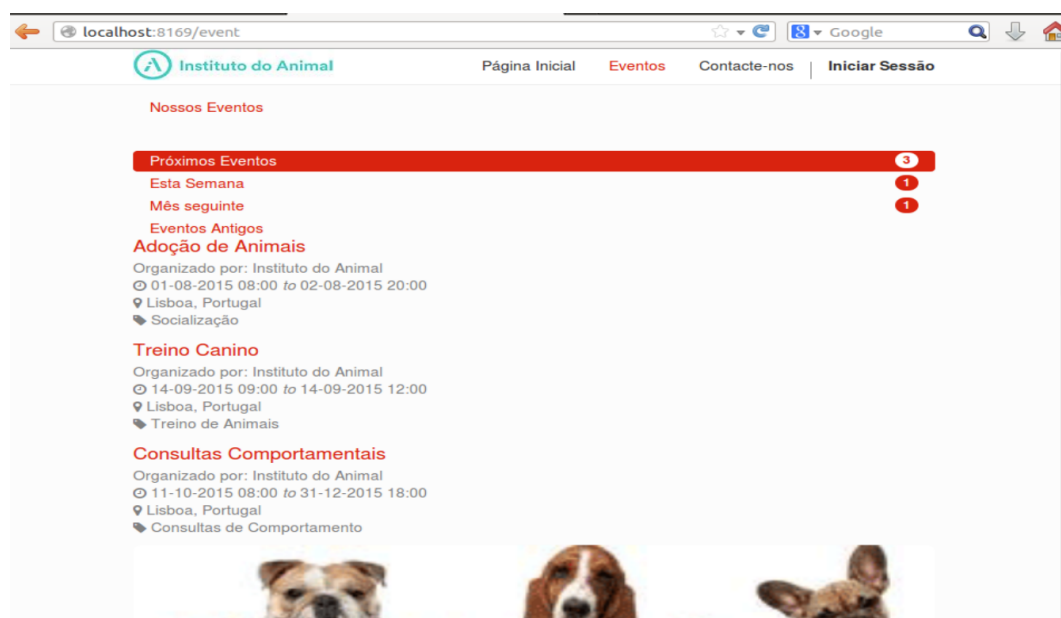
Tabela animal no PostgreSQL

## Página Principal



A página principal tem toda a informação sobre a clínica veterinária, acesso aos contactos, eventos e zona de login para aceder ao BackOffice se for colaborador.

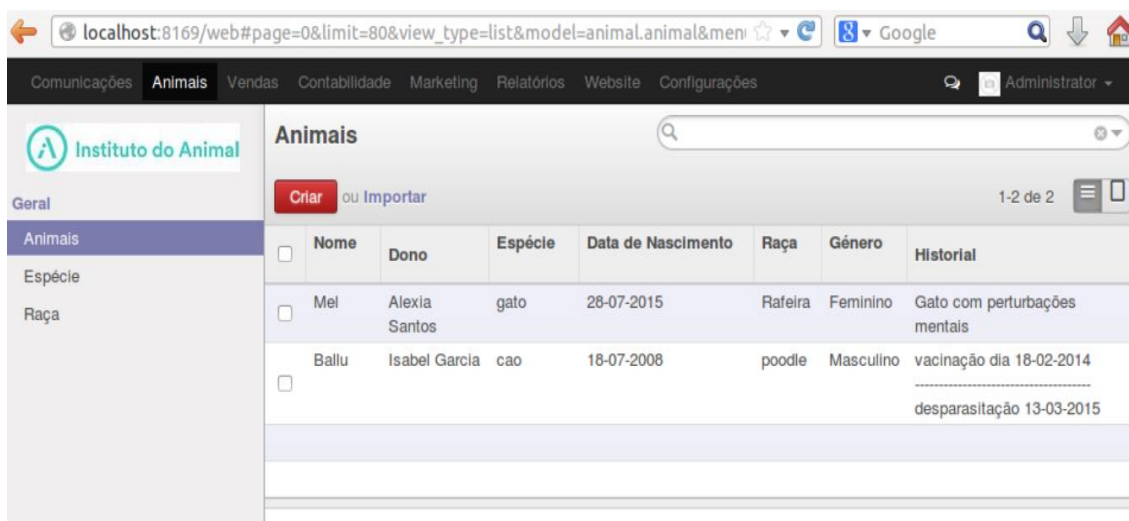
## Página de Eventos



Página de eventos mostra todos os eventos publicados através do BackOffice que estão disponíveis para inscrição do cliente contendo a descrição do evento, data, local, lotação e organizador.

## BackOffice

### Página Animal



localhost:8169/web#page=0&limit=80&view\_type=list&model=animal.animal&men

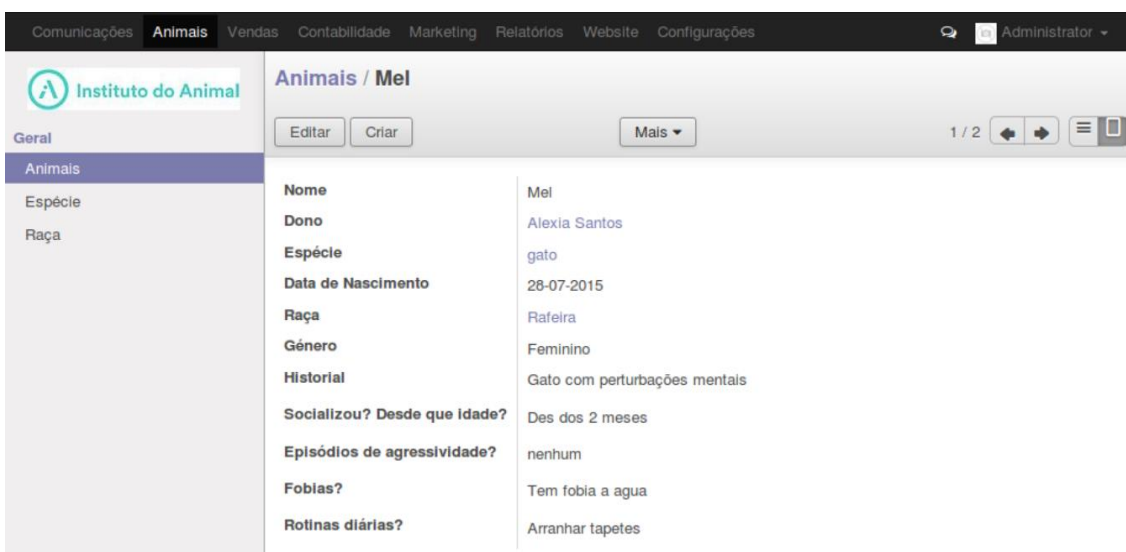
Comunicações **Animais** Vendas Contabilidade Marketing Relatórios Website Configurações

Administrador

**Animais**

[Criar](#) ou [Importar](#) 1-2 de 2

<input type="checkbox"/>	Nome	Dono	Espécie	Data de Nascimento	Raça	Género	Historial
<input type="checkbox"/>	Mel	Alexia Santos	gato	28-07-2015	Rafeira	Feminino	Gato com perturbações mentais
<input type="checkbox"/>	Ballu	Isabel Garcia	cao	18-07-2008	poodle	Masculino	vacinação dia 18-02-2014 desparasitação 13-03-2015



Comunicações **Animais** Vendas Contabilidade Marketing Relatórios Website Configurações

Administrador

**Animais / Mel**

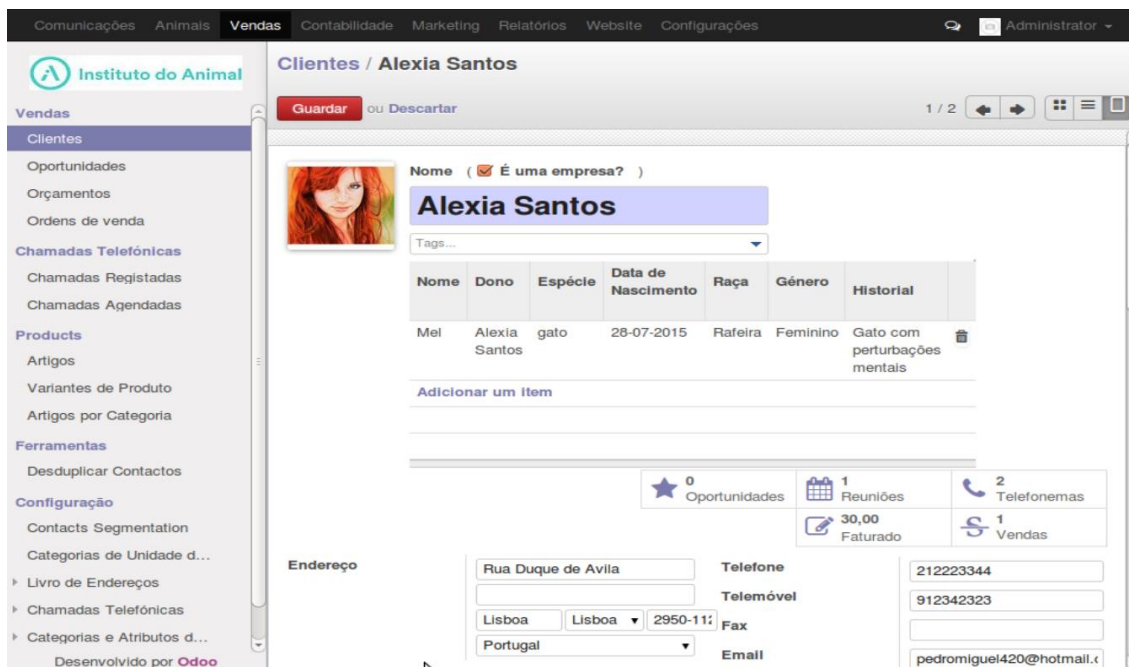
[Editar](#) [Criar](#) [Mais](#) 1 / 2

<b>Nome</b>	Mel
<b>Dono</b>	Alexia Santos
<b>Espécie</b>	gato
<b>Data de Nascimento</b>	28-07-2015
<b>Raça</b>	Rafeira
<b>Género</b>	Feminino
<b>Historial</b>	Gato com perturbações mentais
<b>Socializou? Desde que idade?</b>	Des dos 2 meses
<b>Episódios de agressividade?</b>	nenhum
<b>Fobias?</b>	Tem fobia a agua
<b>Rotinas diárias?</b>	Arranhar tapetes

Página do animal, criada pelo módulo animal, onde podemos inserir novos animais da clínica, associar a um dono, com todas as informações definidas na tabela apresentada nos requisitos, no menu espécie podemos introduzir todo o tipo de espécies que a clínica trata, desde cães a leões e no menu Raça definimos todas as raças por espécie que existente.



## Página de Clientes



**Clientes / Alexia Santos**

Guardar ou Descartar

1 / 2

Nome ( ☒ É uma empresa? )

**Alexia Santos**

Tags...

Nome	Dono	Espécie	Data de Nascimento	Raça	Género	Historial
Mel	Alexia Santos	gato	28-07-2015	Rafeira	Feminino	Gato com perturbações mentais

Adicionar um item

0 Oportunidades 1 Reuniões 2 Telefonemas 30,00 Faturado 1 Vendas

Endereço

Rua Duque de Avila

Lisboa Lisboa 2950-11 Portugal

Telefone

212223344

Telemóvel

912342323

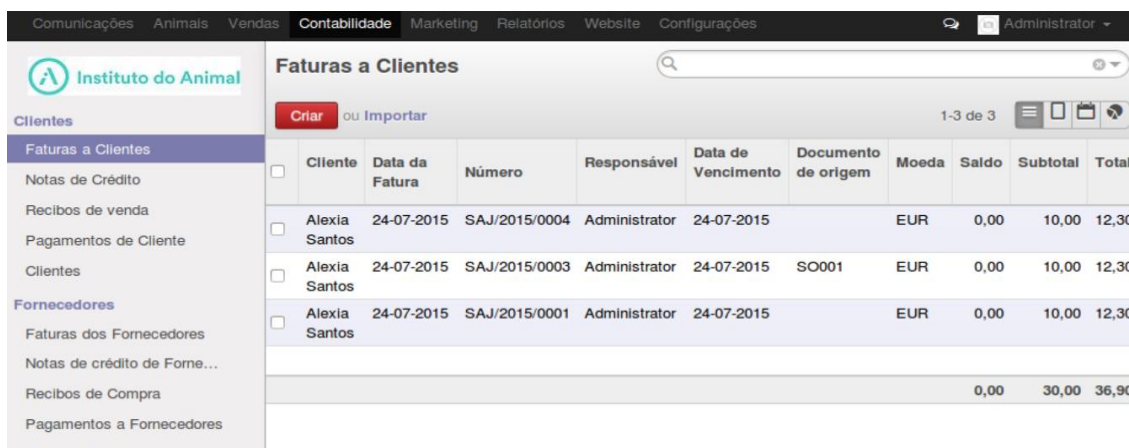
Fax

Email

pedromiguel420@hotmail.c

A página de cliente contém toda a informação do cliente tal como na tabela dos requisitos, com a possibilidade de adicionar animais à sua ficha, visualizar e editar todos os animais que lhe estão associados, tem toda a informação de marcações que tem o cliente, quanto é que já foi faturado, telefonemas efetuados e recebidos bem como o número de vendas já realizado.

## Página de Faturas



**Faturas a Clientes**

Criar ou Importar

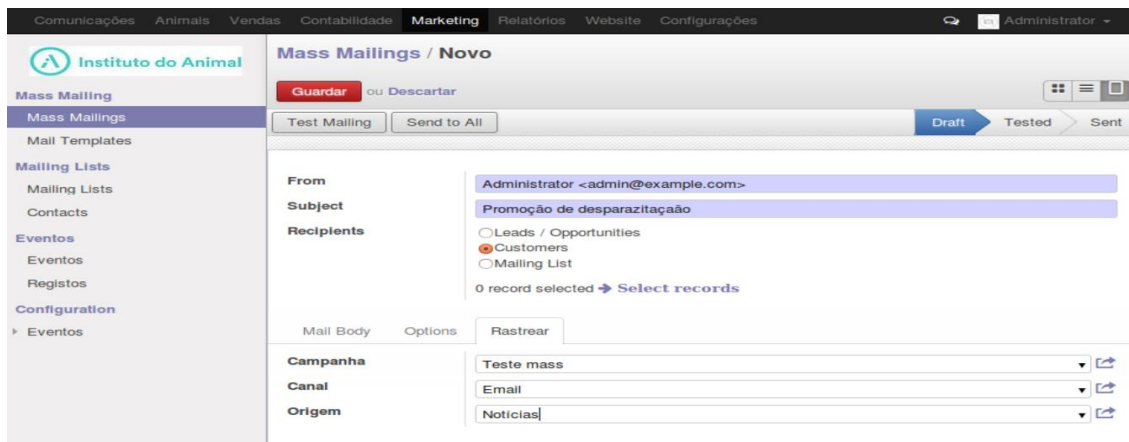
1-3 de 3

	Cliente	Data da Fatura	Número	Responsável	Data de Vencimento	Documento de origem	Moeda	Saldo	Subtotal	Total
<input type="checkbox"/>	Alexia Santos	24-07-2015	SAJ/2015/0004	Administrator	24-07-2015		EUR	0,00	10,00	12,30
<input type="checkbox"/>	Alexia Santos	24-07-2015	SAJ/2015/0003	Administrator	24-07-2015	SO001	EUR	0,00	10,00	12,30
<input type="checkbox"/>	Alexia Santos	24-07-2015	SAJ/2015/0001	Administrator	24-07-2015		EUR	0,00	10,00	12,30
								0,00	30,00	36,90

Página de faturas a clientes e fornecedores apresenta todas as faturas que foram passadas a clientes ou a fornecedores, com possibilidade de criação ou edição.



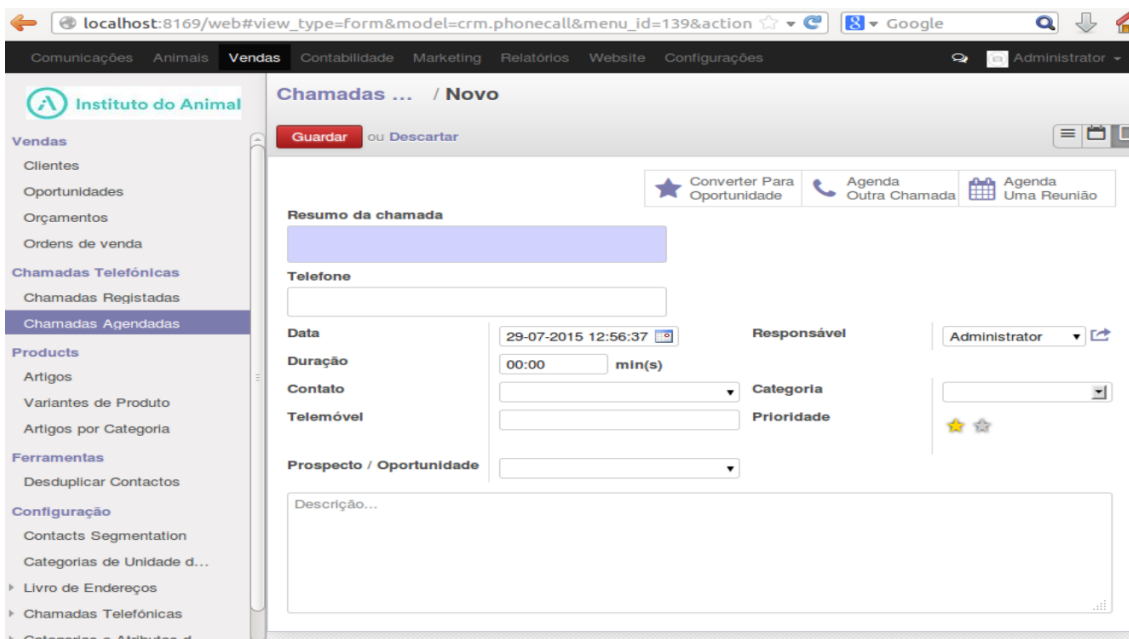
## Página de Campanhas de Marketing



The screenshot shows the 'Mass Mailings / Novo' form. The top navigation bar includes 'Comunicações', 'Animais', 'Vendas', 'Contabilidade', 'Marketing', 'Relatórios', 'Website', and 'Configurações'. The left sidebar has a tree view with 'Mass Mailings' selected. The main form area has a 'Guardar' or 'Descartar' button at the top. Below it are 'Test Mailing' and 'Send to All' buttons. The form fields include: 'From' (Administrator <admin@example.com>), 'Subject' (Promoção de desparasitação), 'Recipients' (radio buttons for Leads / Opportunities, Customers (selected), and Mailing List), and a 'Rastrear' button. At the bottom, there are fields for 'Campanha' (Teste mass), 'Canal' (Email), and 'Origem' (Noticias).

Página de campanhas de marketing por correio eletrónico, onde se pode seleccionar os clientes específicos que se quer enviar ou para um grupo pré definido como por exemplo só quem tem gatos ou então mandar para quem tem apenas marcações de consultas.

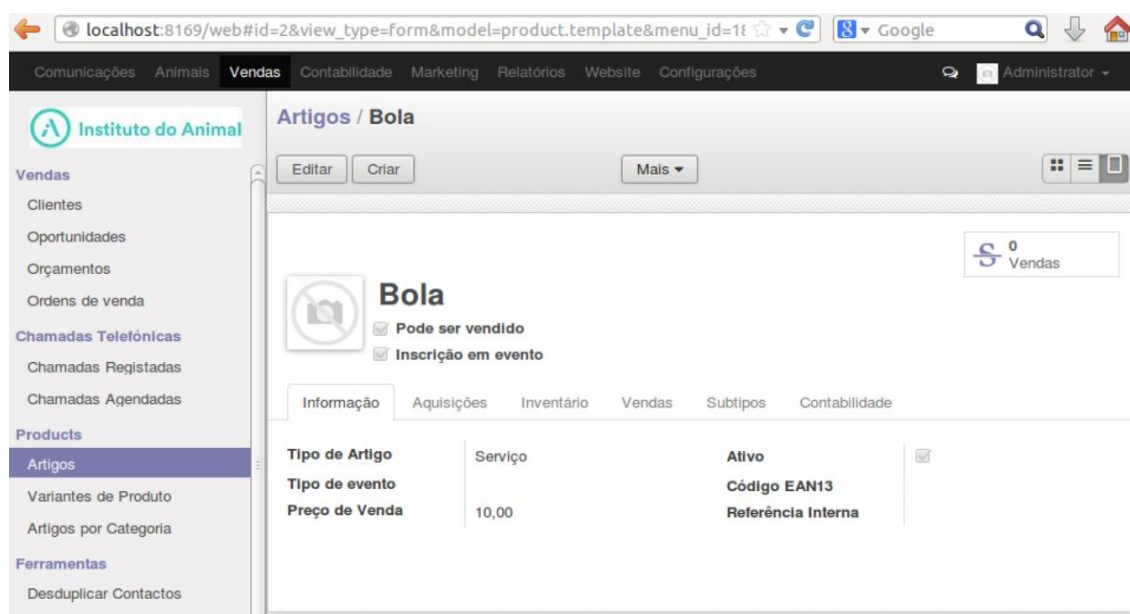
## Página de Telefonemas



The screenshot shows the 'Chamadas ... / Novo' form. The top navigation bar is the same as the previous page. The left sidebar has a tree view with 'Chamadas Agendadas' selected. The main form area has a 'Guardar' or 'Descartar' button at the top. Below it are three buttons: 'Converter Para Oportunidade', 'Agenda Outra Chamada', and 'Agenda Uma Reunião'. The form fields include: 'Resumo da chamada' (text area), 'Telefone' (text field), 'Data' (29-07-2015 12:56:37), 'Responsável' (Administrator), 'Duração' (00:00 min(s)), 'Contato' (text field), 'Categoria' (text field), 'Telemóvel' (text field), 'Prioridade' (radio buttons), 'Prospecto / Oportunidade' (text field), and 'Descrição...' (text area).

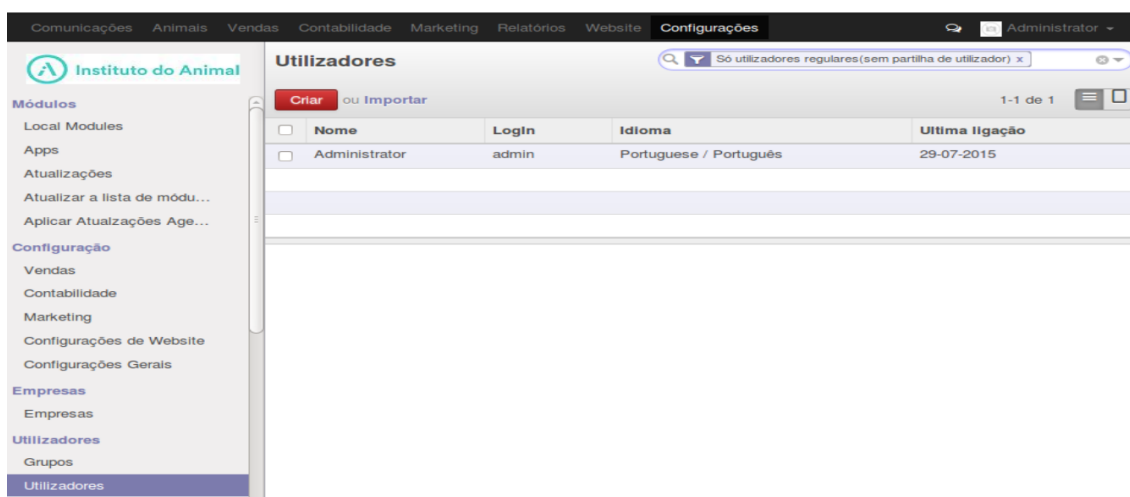
Nesta página podem ser registadas ou agendadas chamadas. Podem ser agendadas chamadas com fim de telemarketing ou simplesmente para recordar consultas marcadas, no caso de registo de chamadas, serve para criar uma base com toda a informação do cliente desde sempre de modo a conhecer melhor o cliente e ter toda a informação da relação que estabelece com a clínica veterinária.

## Página Artigos



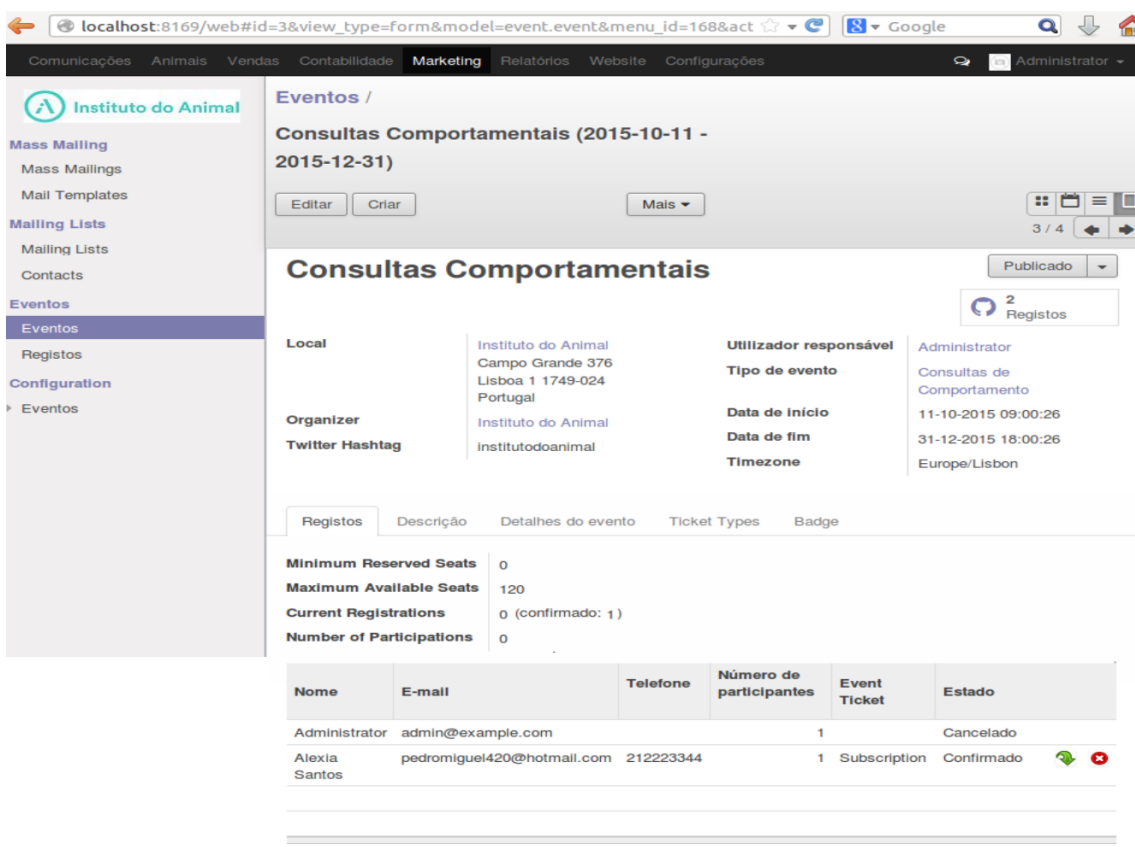
Na página artigos é possível listar todos os artigos existentes na clínica, criar novos artigos, com o número de unidades existentes, verificar quantos já foram vendidos e quantos estão em stock.

## Gestão de Utilizadores



Na gestão de utilizadores é possível criar todos os colaboradores com acesso ao BackOffice, com permissões diferentes consoante o tipo de função que ocupam na clínica.

## Eventos



localhost:8169/web#id=3&view\_type=form&model=event.event&menu\_id=168&act

Comunicações Animais Vendas Contabilidade Marketing Relatórios Website Configurações Administrator

**Instituto do Animal**

Mass Mailing  
Mass Mailings  
Mail Templates

Mailing Lists  
Mailing Lists  
Contacts

Eventos  
Eventos  
Registos

Configuration  
Eventos

**Eventos /**  
**Consultas Comportamentais (2015-10-11 - 2015-12-31)**

Editar Criar Mais

3 / 4

**Consultas Comportamentais** Publicado

2 Registos

**Local** Instituto do Animal  
Campo Grande 376  
Lisboa 1 1749-024  
Portugal

**Organizer** Instituto do Animal

**Twitter Hashtag** institutodoanimal

**Utilizador responsável** Administrator

**Tipo de evento** Consultas de Comportamento

**Data de início** 11-10-2015 09:00:26

**Data de fim** 31-12-2015 18:00:26

**Timezone** Europe/Lisbon

Registos Descrição Detalhes do evento Ticket Types Badge

**Minimum Reserved Seats** 0

**Maximum Available Seats** 120

**Current Registrations** 0 (confirmado: 1)

**Number of Participations** 0

Nome	E-mail	Telefone	Número de participantes	Event Ticket	Estado
Administrator	admin@example.com		1		Cancelado
Alexia Santos	pedromiguel420@hotmail.com	212223344	1	Subscription	Confirmado

Esta página possibilita a criação e publicação de eventos no website na página de eventos, permitindo gerir registos dos clientes, definir a lotação, informação e descrição do próprio evento.

## Conclusão

Apos o desenvolvimento em tecnologia Odoo pôde-se concluir as vantagens da utilização desta tecnologia, não só pelo facto de ser open source e não ter qualquer custo para quem desenvolve mas também por diversas vantagens como a sua flexibilidade enorme para customizações e adaptações, bastante rápido de aprender e fácil de utilizar, ter inúmeros módulos já existentes e facilmente moldáveis ao nosso negócio e com uma gestão integrada de permissões simples e intuitiva, através da criação de grupos e perfis de utilização e por fim acessível em qualquer lugar e em qualquer dispositivo o que é uma vantagem enorme podendo alargar o negocio não só a uma plataforma específica mas sim a todo o mercado.

O CRM criado irá auxiliar o aumento da taxa de fidelização da clientela (que custa 5 vezes menos do que conquistar novos), economizar tempo graças à automatização de tarefas (aumentar a produtividade), otimizar a colaboração entre os diversos serviços da empresa (comercial, marketing, serviço pós-venda) e Contribuir para vantagem competitiva da empresa.

Num trabalho futuro pode ser viável a integração de animais de produção, numa escala maior, mais direccionado para as empresas com módulos de faturação certificados.

Este foi um projeto bastante gratificante que me permitiu a aprendizagem de novas tecnologias e novos mundos na área do desenvolvimento e certamente que os conhecimentos adquiridos neste trabalho serão compensadores num futuro próximo.

## Bibliografia

- [1] Odoo, <https://www.odoo.com/documentation/8.0/howtos/backend.html>
- [2] Tutoriais Python, <https://docs.python.org/2/tutorial/>
- [3] Wikipedia, <http://en.wikipedia.org/>

## Anexos

Todos os ficheiros que se seguem em baixo pertencem ao módulo criado de raiz com o nome “Animal” para responder as necessidades do negócio proposto, colmatando as lacunas em falta nas opções oferecidas pelo Odoo. Este módulo consiste na associação do animal ao cliente e cliente animal, introdução e criação de novos animais com todos os campos solicitados nos requisitos, alterações na página de clientes conforme requisitos, bem como o pedido de criação (feito automaticamente pelo Odoo) das tabelas Animal, Espécie e Raça na base de dados PostgreSQL.

### **course\_security:**

```
<?xml version="1.0"?>
<openerp>
<data>

  <record model="res.groups" id="base.group_tool_user">
    <field name="name">User</field>
    <field name="category_id" ref="base.module_category_tools"/>
  </record>

</data>
</openerp>
```

## \_\_openerp\_\_.py:

Código que vai permitir encontrar o módulo que foi adicionado ao Odoo para poder instalar na aplicação que esta a ser criada, permitindo descrever todo o módulo e para que serve para o utilizador saber o que esta a instalar.

```
{
    'name': 'Animal',
    'version': '1.0',
    'category': 'Tools',
    'summary': 'Courses, Sessions, Subscriptions',
    'description': """
Courses, Sessions, Subscriptions
=====

este module manages the definition of course, the organization,
sessions, and the subscriptions of participants.
""",
    'depends': ['base','crm'],
    'data': ['view/menu.xml'],
    'images': [],
    'demo': [],
    'installable': True,
    'application': True,
}
```

## menu.xml:

O código que se segue contém a view em XML de todo o módulo Animal desde a ordem com que os menus Animal, Espécie e Raça estão ordenados, qual o nome do menu principal, todos os campos que aparecem na view de criação ou de edição de qualquer uma das temáticas bem como afeta a view do cliente, alterando assim a view do módulo res.partner (que pertence de raiz ao Odoo) para conter a informação definida nos requisitos do cliente e adicionar a informação do animal à view do cliente.

```
<?xml version="1.0" encoding="UTF-8"?>
<openerp>
  <data>

    <menuitem name="Animais" id="menu_root" sequence="19"/>

    <menuitem name="Geral" id="menu_general" parent="menu_root"/>

<!-- Idea Category Form View -->
  <record model="ir.ui.view" id="view_animal_animal_form">
    <field name="name">animal.animal.form</field>
    <field name="model">animal.animal</field>
    <field name="arch" type="xml">
      <form string="Category of Ideas" version="7.0">
        <group>
          <field name="name"/>
          <field name="cliente"/>
          <!--<field name="dono"/>-->
          <field name="especie"/>
          <field name="idade"/>
          <field name="raca"/>
          <field name="genero"/>
          <field name="description"/>
        </group>
      </form>
    </field>
  </record>
```



```

    <field name="description2"/>
    <field name="description3"/>
    <field name="description4"/>
    <field name="description5"/>
  </group>
</form>
</field>
</record>

```

```
<!-- Idea Category Tree View -->
```

```

<record model="ir.ui.view" id="view_animal_animal_tree">
  <field name="name">animal.animal.tree</field>
  <field name="model">animal.animal</field>
  <field name="arch" type="xml">
    <tree string="Category of ideas">
      <field name="name"/>
      <field name="cliente"/>
      <!--<field name="dono"/>-->
      <field name="especie"/>
      <field name="idade"/>
      <field name="raca"/>
      <field name="genero"/>
      <field name="description"/>
      <!--<field name="description2"/>-->
      <!--<field name="description3"/>-->
      <!--<field name="description4"/>-->
      <!--<field name="description5"/>-->
    </tree>
  </field>
</record>

```

```

<record model="ir.actions.act_window" id="action_courses">
  <field name="name">Animais</field>

```

```
<field name="res_model">animal.animal</field>
<field name="view_type">form</field>
<field name="view_mode">tree,form</field>
</record>
```

```
<menuitem name="Animais" id="menu_courses" parent="menu_general"
sequence="1" action="action_courses"/>
```

```
<!-- Idea Category Form View -->
<record model="ir.ui.view" id="view_animal_especie_form">
  <field name="name">animal.especie.form</field>
  <field name="model">animal.especie</field>
  <field name="arch" type="xml">
    <form string="Category of Ideas">
      <group>
        <field name="name"/>
      </group>
    </form>
  </field>
</record>
```

```
<!-- Idea Category Tree View -->
<record model="ir.ui.view" id="view_animal_especie_tree">
  <field name="name">animal.especie.tree</field>
  <field name="model">animal.especie</field>
  <field name="arch" type="xml">
    <tree string="Category of ideas">
      <field name="name"/>
    </tree>
  </field>
</record>
```

```
<record model="ir.actions.act_window" id="action_especie">
```

```
  <field name="name">Espécie</field>
```

```
  <field name="res_model">animal.especie</field>
```

```
  <field name="view_type">form</field>
```

```
  <field name="view_mode">tree,form</field>
```

```
</record>
```

```
<menuitem name="Espécie" id="menu_especie" parent="menu_general"
```

```
  sequence="2" action="action_especie"/>
```

```
<!-- Idea Category Form View -->
```

```
<record model="ir.ui.view" id="view_animal_raca_form">
```

```
  <field name="name">animal.raca.form</field>
```

```
  <field name="model">animal.raca</field>
```

```
  <field name="arch" type="xml">
```

```
    <form string="Category of Ideas" version="7.0">
```

```
      <group>
```

```
        <field name="name"/>
```

```
        <field name="especie_id"/>
```

```
      </group>
```

```
    </form>
```

```
  </field>
```

```
</record>
```

```
<!-- Idea Category Tree View -->
```

```
<record model="ir.ui.view" id="view_animal_raca_tree">
```

```
  <field name="name">animal.raca.tree</field>
```

```
<field name="model">animal.raca</field>
<field name="arch" type="xml">
  <tree string="Category of ideas">
    <field name="name"/>
    <field name="especie_id"/>
  </tree>
</field>
</record>

<record model="ir.actions.act_window" id="action_raca">
  <field name="name">Raça</field>
  <field name="res_model">animal.raca</field>
  <field name="view_type">form</field>
  <field name="view_mode">tree,form</field>

</record>

<menuitem name="Raça" id="menu_raca" parent="menu_general"
  sequence="3" action="action_raca"/>

<record model="ir.ui.view" id="view_res_partner_form_inherit">
  <field name="name">res.partner.inherit.form</field>
  <field name="model">res.partner</field>
  <field name="inherit_id" ref="base.view_partner_form" />
  <field name="arch" type="xml">
    <field name="category_id" position="after">
      <field name="animais"/>
    </field>

    <field name="website" position="replace">
```

```
<field name="website" invisible="1"/>
</field>

<field name="title" position="after">
  <field name="desc"/>
</field>

<field name="desc" position="after">
  <field name="desc2"/>
</field>

<field name="desc2" position="after">
  <field name="age"/>
</field>
</field>
</record>

</data>
</openerp>
```

## course.py:

Todo o código Python que define as permissões de cada campo do modulo Animal, como responde, se são campos obrigatórios ou não, se são numéricos ou de texto se herdam informações de outros menus ou não. Ou seja, define o tipo ou a ação que cada campo da view pode ter.

```
#from openerp import models, fields
from openerp.osv import osv
from openerp.osv import fields

class animal_animal(osv.osv):
    _name = 'animal.animal'
    _columns = {

        'name' : fields.char('Nome', required=True),
        #'dono' : fields.text(required=True),
        'cliente' : fields.many2one('res.partner',string='Dono', required=True),
        'especie' : fields.many2one('animal.especie',string='Espécie', required=True),
        'raca' : fields.many2one('animal.raca', domain=" [('especie_id',
        '=' ,especie)]",string='Raça'),
        'idade' : fields.date('Data de Nascimento',required=True),
        'genero' : fields.selection([ ('Masculino', 'Masculino'), ('Feminino',
        'Feminino') ],string='Género',required=True),
        'description' : fields.text('Historial', required=True),
        'description2' : fields.text('Socializou? Desde que idade?'),
        'description3' : fields.text('Episódios de agressividade?'),
        'description4' : fields.text('Fobias?'),
        'description5' : fields.text('Rotinas diárias?'),

    }
```

```
class animal_especie(osv.osv):
    _name = 'animal.especie'
    _columns = {
        'name' : fields.char('Nome', required=True),
    }

class animal_raca(osv.osv):
    _name = 'animal.raca'

    _columns = {
        'name' : fields.char('Nome', required=True),
        #especie = fields.Many2one(string='Espécie', required=True)

        'especie_id': fields.many2one('animal.especie',string='Espécie'),

    }

class res_partner(osv.osv):
    _name="res.partner"
    _inherit = 'res.partner'

    _columns = {
        'animais':fields.one2many('animal.animal', 'cliente',string='Animal' ),
        'age' : fields.date(string='Data de Nascimento',required=True),
        'desc':fields.text(string='Perfil psicográfico'),
        'desc2':fields.date(string='Data registo'),

    }
```