Mestrado Integrado em Engenharia Informática e Computação

U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

PharmaCRM - Sales Force Automation

"Information Systems" Class: 4MIEIC05 **Group ID**: D

André Lago - up201303313 - up201303313@fe.up.pt
Diogo Moura - up201304068 - up201304068@fe.up.pt
Gustavo Silva - up201304143 - up201304143@fe.up.pt
Inês Carneiro - up201303501 - up201303501@fe.up.pt
Sérgio Domingues - up201304367 - up201304367@fe.up.pt

Index

Project overview

PharmaLabs is a pharmaceutical company, that sells pharmaceutical products, such as medicine and health products. Their clients are always other companies/businesses such as hospitals, offices, drug stores or similar companies. To persuade these customers into buying their products, PharmaLabs has a vast sales force team, responsible for raising customers and investors as well as collecting customer's information, registering sales orders, etc.

The project consists of a *web app* - PharmaCRM - to manage the above mentioned sales force team in order to automate it as much as possible, with the integration of the Primavera ERP. The main purpose of the app is to make the planning, scheduling and actual work of the sales force team more efficient.

As mentioned in the previous report (the specification document), PharmaCRM has two types of users:

- **Sales representatives**
  - Consult daily activities and places to visit
  - Consult personal goal and KPI (key performance indicators)
  - Consult list and details of clients
  - Manage leads, activities and orders
- **Sales managers**
  - Consult all items available for sales representatives
  - Consult company and sales representative KPI
  - Manage the goals of sales representatives

The core views described in the next section allow the users to perform the actions just mentioned in a simple, user-friendly way.

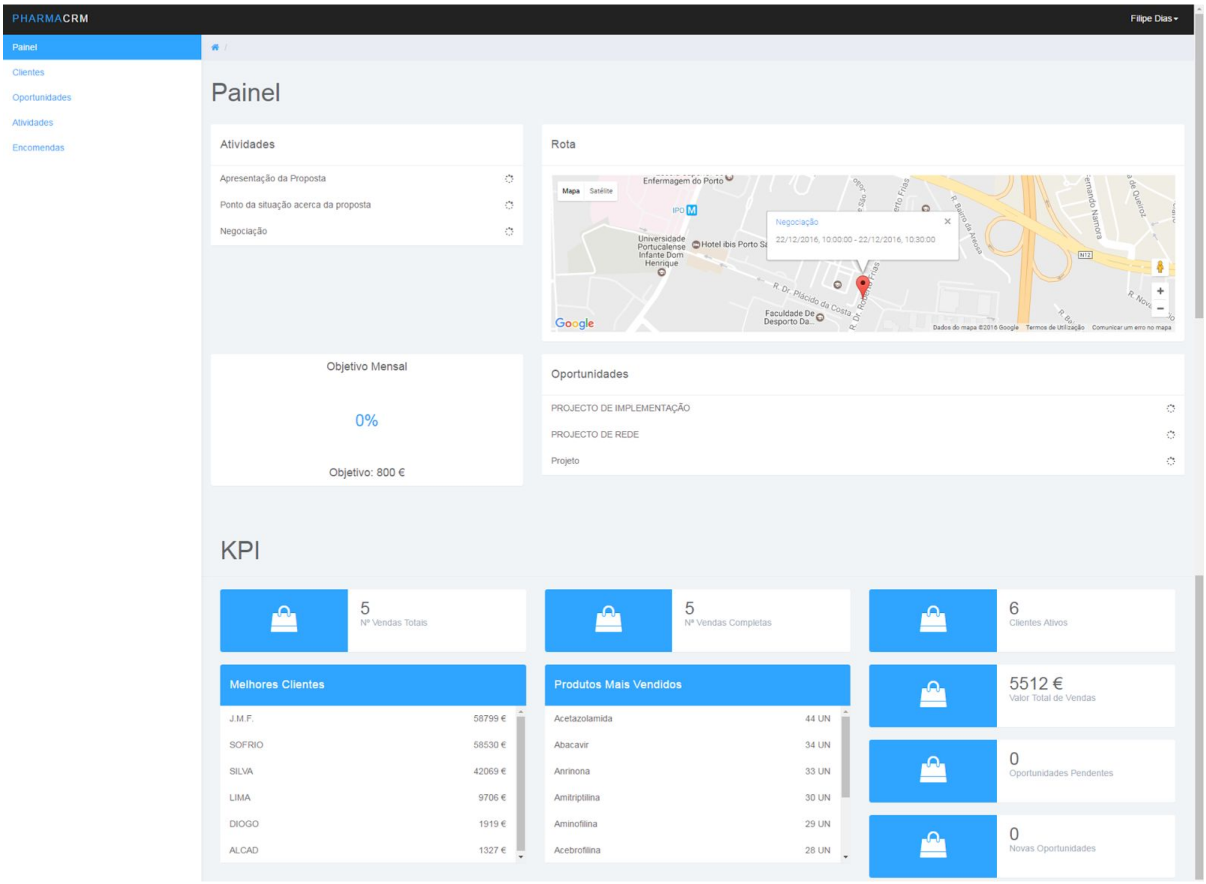
To better understand the concepts described by the core views and interoperability tables, we'll explain what every important term means:

- **Client** - potential or known customer of the company, who may have previously completed transactions with PharmaLabs or not. To ease the work of the sales representative, PharmaCRM makes no distinction between prospect client/contact and effective client.
- **Sales representative** - employee who meets and negotiates with clients and potential targets. Represents the main user of the system, who wants to get and create tasks and see customer information.
- **Sales manager** - employee who manages the sales representatives by viewing the key performance indicators for both the company and specific sales representatives. Inside PharmaCRM, this employee is called the Administrator.
- **Activity**- appointment with a customer or potential target, which can be either presential (in either of the attendee's offices) or via video call, phone call, email, proposal, task,commitment with client or proposal submission (which is assumed to take place in the company's employee's office).

- **Sales order** - allocates stock and allows the client to purchase stock. This means that an agreement has been reached with a customer for the purchase of well-defined quantities of products for an agreed cost.
- **Sale opportunity (lead)** - represents an ongoing negotiation with a customer, for certain products at a certain price. There's a possibility to reserve stock for a sales opportunity so that when an agreement is reached the opportunity can be converted into a sales order without the risk of insufficient stock.

Core views

Sales representative dashboard



Painel

Atividades

Apresentação da Proposta

Ponto da situação acerca da proposta

Negociação

Rota

Negociação

22/12/2016, 10:00:00 - 22/12/2016, 10:30:00

Objetivo Mensal

0%

Objetivo: 800 €

Oportunidades

PROJECTO DE IMPLEMENTAÇÃO

PROJECTO DE REDE

Projeto

KPI

5
Nº Vendas Totais

5
Nº Vendas Completas

6
Clientes Ativos

Melhores Clientes	
J.M.F.	58799 €
SOFRIO	58530 €
SILVA	42069 €
LIMA	9706 €
DIOGO	1919 €
ALCAD	1327 €

Produtos Mais Vendidos	
Acetazolamida	44 UN
Abacavir	34 UN
Anrinona	33 UN
Amitriptilina	30 UN
Aminofilina	29 UN
Acebrofilina	28 UN

5512 €
Valor Total de Vendas

0
Oportunidades Pendentes

0
Novas Oportunidades

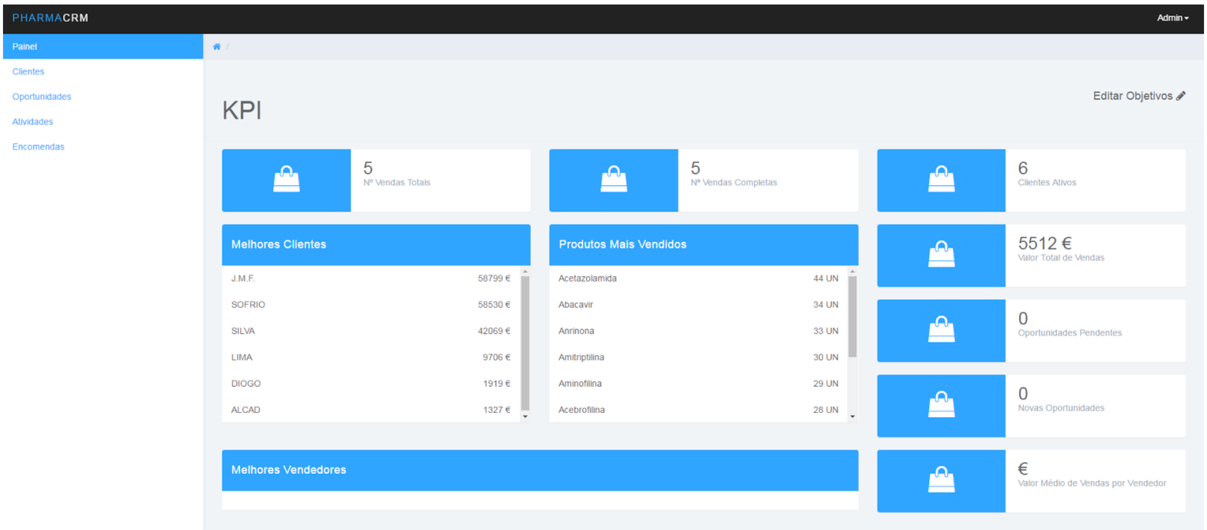
Fig.1 - Core View #1 - Sales Representative Dashboard

On the dashboard, the sales representative can see the activities he has scheduled for that day, a map with the places to visit during that day, his current status on his monthly goal, the current leads he is responsible for and his KPIs.

Inpaths	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

OutPaths	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders
	http://localhost:8080/#/customers/<codCliente>
	http://localhost:8080/#/leads/<id>
	http://localhost:8080/#/activities/<id>

Sales manager dashboard



5 Nº Vendas Totais

5 Nª Vendas Completas

6 Clientes Ativos

5512 € Valor Total de Vendas

0 Oportunidades Pendentes

0 Novas Oportunidades

€ Valor Médio de Vendas por Vendedor

Melhores Clientes	
J.M.F.	58799 €
SOFRIO	58530 €
SILVA	42069 €
LIMA	9706 €
DIOGO	1919 €
ALCAD	1327 €

Produtos Mais Vendidos	
Acetazolamida	44 UN
Abacavir	34 UN
Anrinona	33 UN
Amitriptilina	30 UN
Aminofilina	29 UN
Acebrofilina	28 UN

Melhores Vendedores

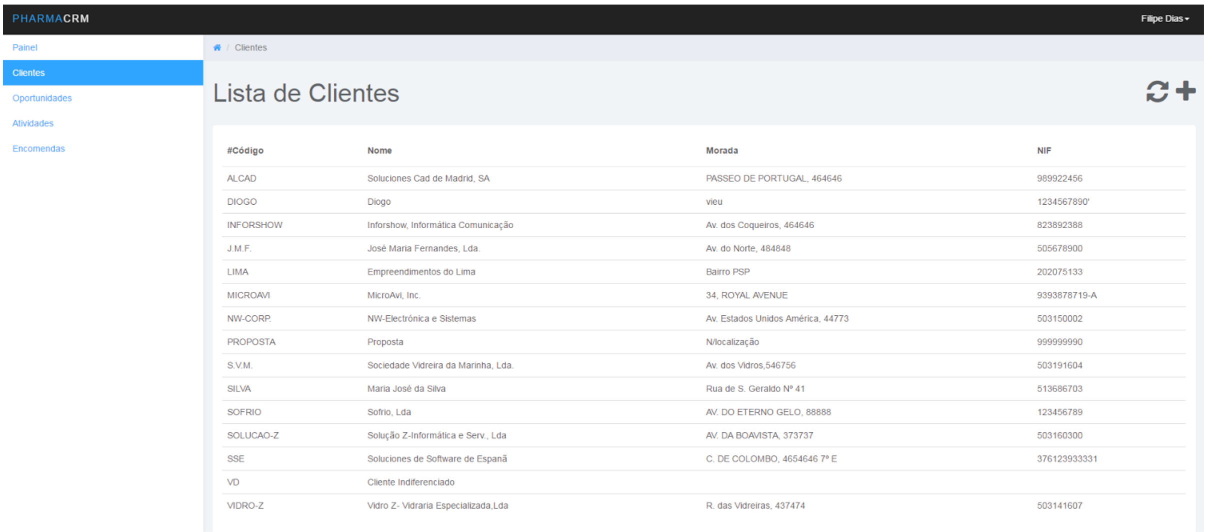
Fig.2 - Core View #2 - Sales Manager Dashboard

On the dashboard, the sales manager can see the overall KPIs of the company, as well as see and edit each sales representative's monthly goal.

Inpaths	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

OutPaths	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders
	http://localhost:8080/#/customers/<codCliente>

Customers list



Lista de Clientes

#Código	Nome	Morada	NIF
ALCAD	Soluciones Cad de Madrid, SA	PASSEO DE PORTUGAL, 464646	989922456
DIOGO	Diogo	vieu	1234567890'
INFORSHOW	Inforshow, Informática Comunicação	Av. dos Coqueiros, 464646	823892388
J.M.F.	José Maria Fernandes, Lda.	Av. do Norte, 484848	505678900
LIMA	Empreendimentos do Lima	Bairro PSP	202075133
MICROAVI	MicroAvi, Inc.	34, ROYAL AVENUE	9393878719-A
NW-CORP.	NW-Electrónica e Sistemas	Av. Estados Unidos América, 44773	503150002
PROPOSTA	Proposta	N/localização	999999990
S.V.M.	Sociedade Vidreira da Marinha, Lda.	Av. dos Vidros,546756	503191604
SILVA	Maria José da Silva	Rua de S. Geraldo Nº 41	513686703
SOFRIO	Sofrio, Lda	AV. DO ETERNO GELO, 88888	123456789
SOLUCAO-Z	Solução Z-Informática e Serv., Lda	AV. DA BOAVISTA, 373737	503160300
SSE	Soluciones de Software de Espanã	C. DE COLOMBO, 4654646 7º E	376123933331
VD	Cliente Indiferenciado		
VIDRO-Z	Vidro Z- Vidraria Especializada,Lda	R. das Vidreiras, 437474	503141607

Fig.3 - Core View #3 - Customers List

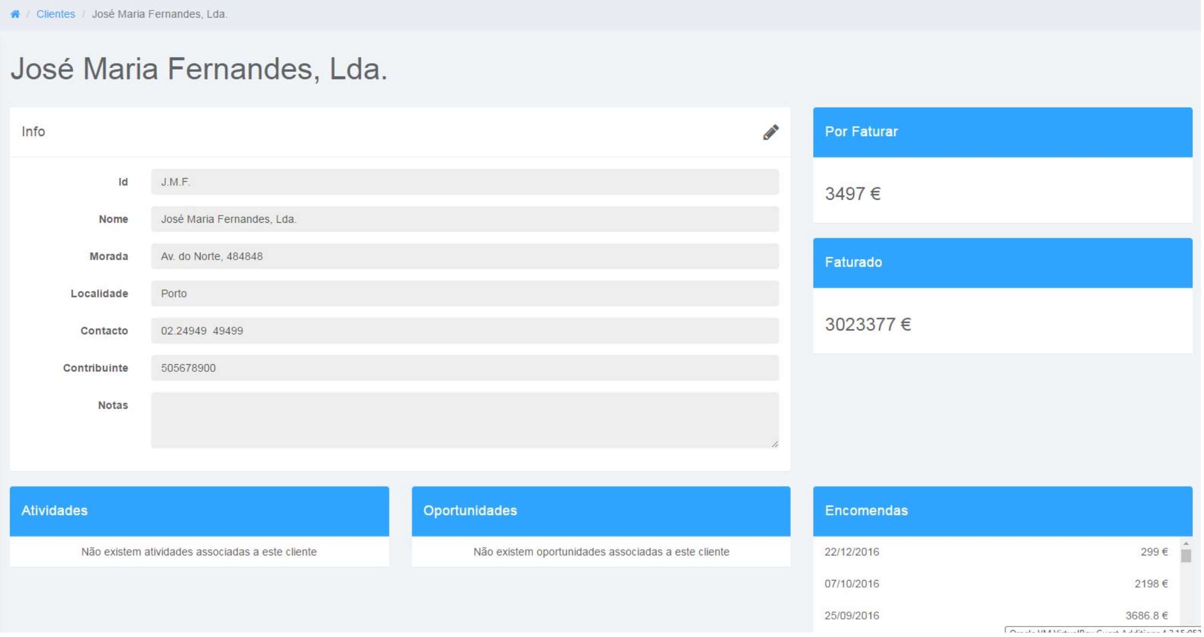
This view lists all the company's customers, along with some of their information and links to their details pages.

It is also possible to create a new customer with the "+" button.

Inpaths	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders
	http://localhost:8080/#/new

OutPaths	http://localhost:8080/#/customers/<codCliente>
	http://localhost:8080/#/
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

Customer details



José Maria Fernandes, Lda.

Info

Id	J.M.F.
Nome	José Maria Fernandes, Lda.
Morada	Av. do Norte, 484848
Localidade	Porto
Contacto	02.24949 49499
Contribuinte	505678900
Notas	

Por Faturar

3497 €

Faturado

3023377 €

Atividades

Não existem atividades associadas a este cliente

Oportunidades

Não existem oportunidades associadas a este cliente

Encomendas

22/12/2016	299 €
07/10/2016	2198 €
25/09/2016	3686.8 €

Fig.4 - Core View #4 - Customers Details

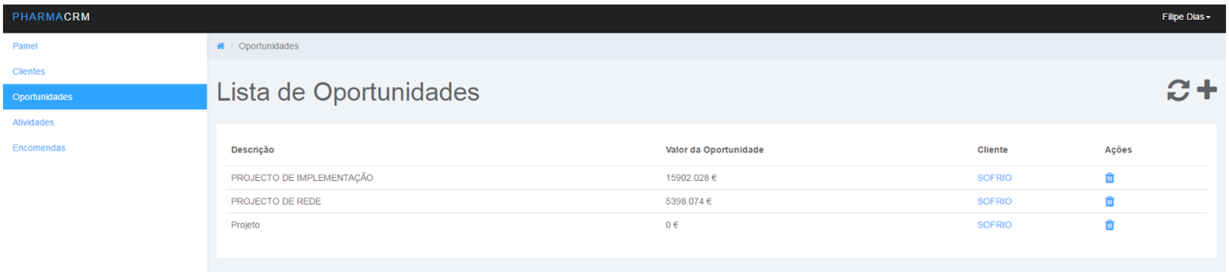
This view lists all details of a specific customer, such as name, address, contact and others. Besides that information, the view shows other relevant data about the customer, such as received and not received amount of money and associated activities, leads and orders.

It is also possible to edit the customer’s details.

Inpaths	http://localhost:8080/#/customers/<codCliente>

OutPaths	http://localhost:8080/#/leads/<id>
	http://localhost:8080/#/salesorders/<id>
	http://localhost:8080/#/activities/<id>
	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

Leads list



Lista de Oportunidades

Descrição	Valor da Oportunidade	Cliente	Ações
PROJECTO DE IMPLEMENTAÇÃO	15902.028 €	SOFRIO	🗑
PROJECTO DE REDE	5398.074 €	SOFRIO	🗑
Projeto	0 €	SOFRIO	🗑

Fig.5 - Core View #5 - Leads List

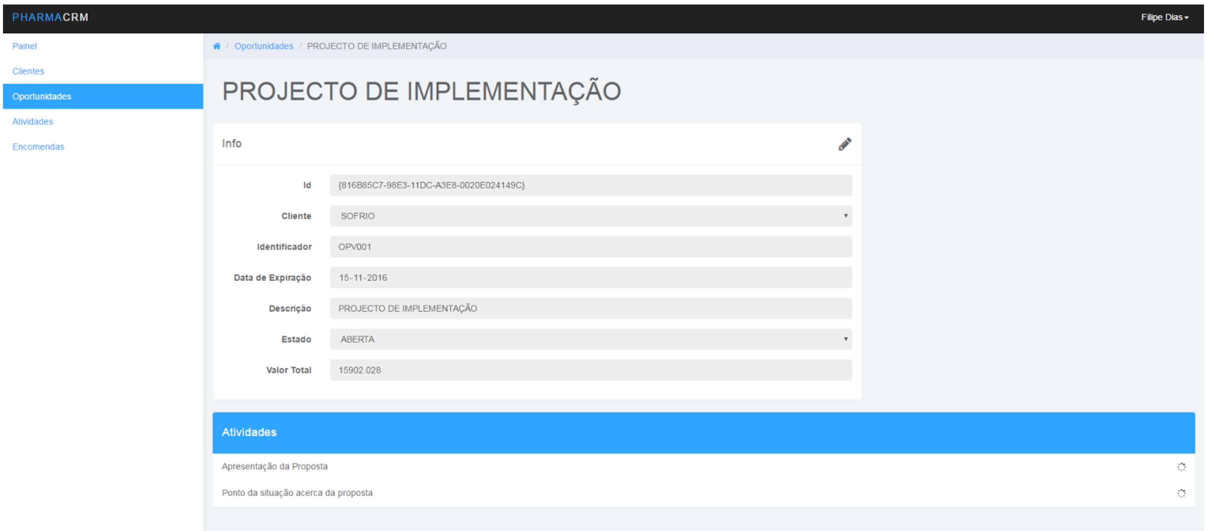
This view lists all leads, as well as some of their details and links to their complete pages. It is also possible to create new leads through this view.

Inpaths	http://localhost:8080/#/leads
	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/activities

	http://localhost:8080/#/salesorders

OutPaths	http://localhost:8080/#/leads/<id>
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

Lead details



PROJECTO DE IMPLEMENTAÇÃO

Info

Id	{816B85C7-98E3-11DC-A3E8-0020E024149C}
Cliente	SOFRIO
Identificador	OPV001
Data de Expiração	15-11-2016
Descrição	PROJECTO DE IMPLEMENTAÇÃO
Estado	ABERTA
Valor Total	15902.028

Atividades

- Apresentação da Proposta
- Ponto da situação acerca da proposta

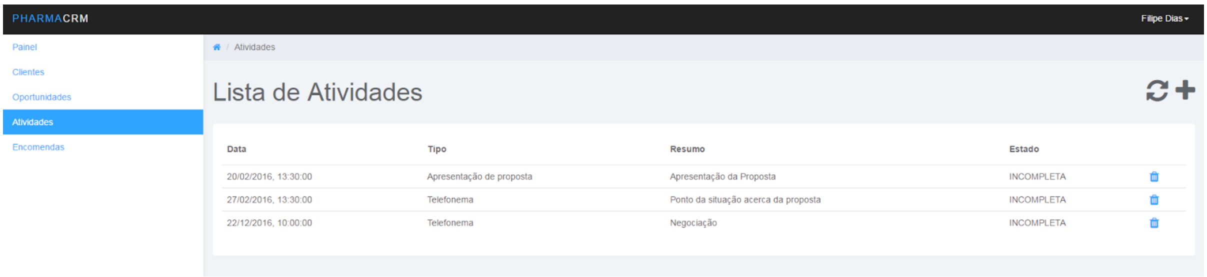
Fig.6 - Core View #6 - Lead Details

This view displays the complete details of a lead, as well as activities associated with that lead. It is also possible to edit the details of a lead through this view.

Inpaths	http://localhost:8080/#/leads/<id>

OutPaths	http://localhost:8080/#/activities/<id>
	http://localhost:8080/#/
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders
	http://localhost:8080/#/customers

Activities list



Lista de Atividades

Data	Tipo	Resumo	Estado
20/02/2016, 13:30:00	Apresentação de proposta	Apresentação da Proposta	INCOMPLETA
27/02/2016, 13:30:00	Telefonema	Ponto da situação acerca da proposta	INCOMPLETA
22/12/2016, 10:00:00	Telefonema	Negociação	INCOMPLETA

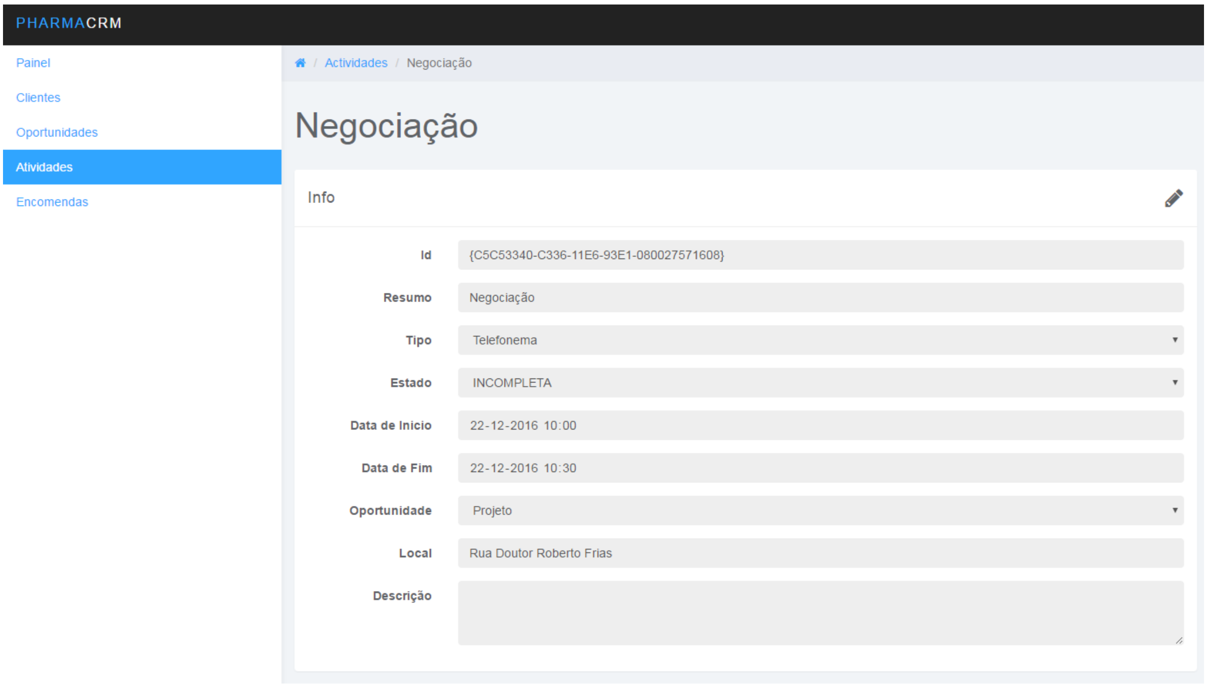
Fig.7 - Core View #7 - Activities List

This view displays a list of all activities, as well as some details about them and links to their individual pages. It is also possible to create new activities through this view.

Inpaths	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/salesorders

OutPaths	http://localhost:8080/#/activities/<id>
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/salesorders

Activity details



Negociação

Info

Id	{C5C53340-C336-11E6-93E1-080027571608}
Resumo	Negociação
Tipo	Telefonema
Estado	INCOMPLETA
Data de Inicio	22-12-2016 10:00
Data de Fim	22-12-2016 10:30
Oportunidade	Projeto
Local	Rua Doutor Roberto Frias
Descrição	

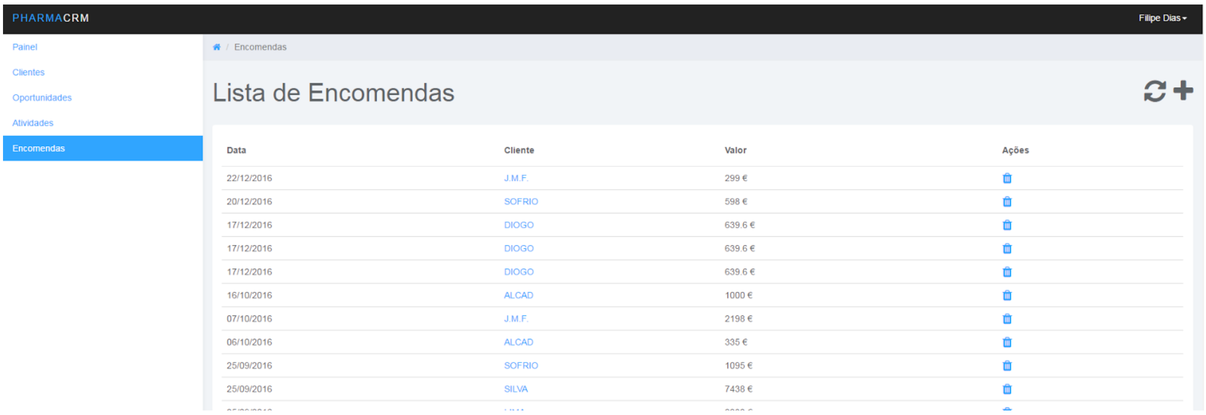
Fig.8 - Core View #8 - Activity Details

This view displays the complete details of an activity, such as it’s name, type, status, start and end dates, location and opportunity. It is also possible to edit the details of the activity through this view.

Inpaths	http://localhost:8080/#/activities/<id>

OutPaths	http://localhost:8080/#/
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders
	http://localhost:8080/#/customers

Order list



Lista de Encomendas

Data	Cliente	Valor	Ações
22/12/2016	J.M.F.	299 €	🗑
20/12/2016	SOFRIO	598 €	🗑
17/12/2016	DIOGO	639.6 €	🗑
17/12/2016	DIOGO	639.6 €	🗑
17/12/2016	DIOGO	639.6 €	🗑
16/10/2016	ALCAD	1000 €	🗑
07/10/2016	J.M.F.	2198 €	🗑
06/10/2016	ALCAD	335 €	🗑
25/09/2016	SOFRIO	1095 €	🗑
25/09/2016	SILVA	7438 €	🗑

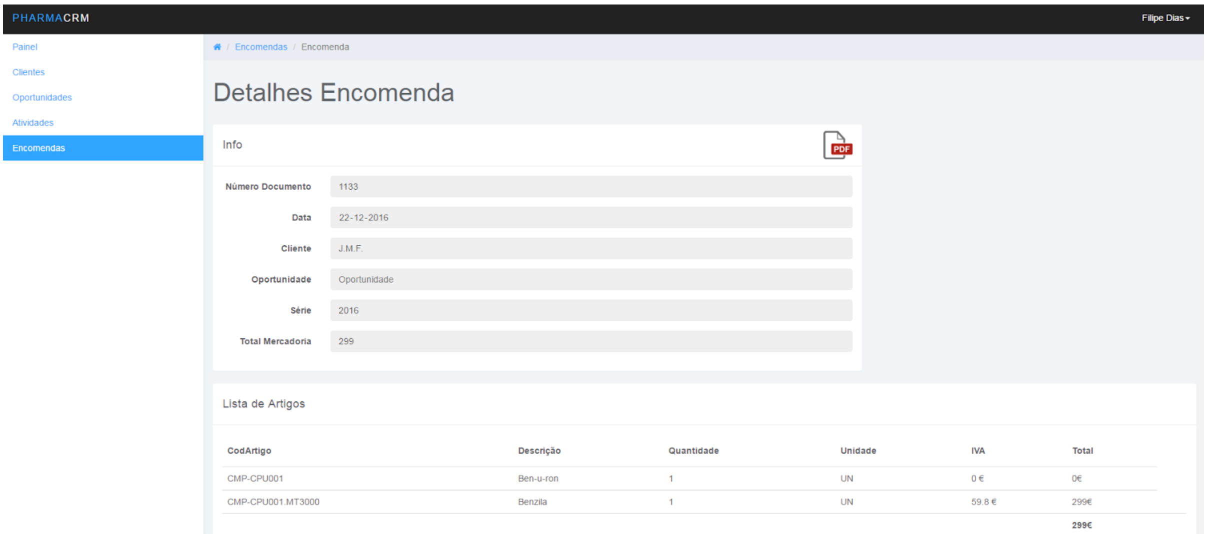
Fig.9 - Core View #9 - Order List

This view displays the list of orders placed, sorted from the most recent to the oldest, along as some of their details and links to their individual pages. It is also possible to create new orders through this view.

Inpaths	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

OutPaths	http://localhost:8080/#/leads/<id>
	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

Order details



Detalhes Encomenda

Info

Número Documento	1133
Data	22-12-2016
Cliente	J.M.F.
Oportunidade	Oportunidade
Série	2016
Total Mercadoria	299

Lista de Artigos

CodArtigo	Descrição	Quantidade	Unidade	IVA	Total
CMP-CPU001	Ben-u-ron	1	UN	0 €	0€
CMP-CPU001.MT3000	Benzila	1	UN	59.8 €	299€
					299€

Fig.10 - Core View #10 - Order Details

This view displays the complete details of a sales order, including the total value, discount (if any exists) and list of products and their quantities. It is also possible to extract the sales order to a printable file by clicking on the PDF icon on the top right.

Inpaths	http://localhost:8080/#/leads

OutPaths	http://localhost:8080/#/
	http://localhost:8080/#/customers
	http://localhost:8080/#/leads
	http://localhost:8080/#/activities
	http://localhost:8080/#/salesorders

Other features

One of the added features was the possibility to have individual goals for the value of orders made for each sales representative, a goal set by any sales manager. This feature was implemented with a separate database managed by our API. The paths used for this feature are described in the “Interoperability with Primavera” section, although they are not interacting directly with Primavera data or features.

It was also implemented a form of user authentication with use of the external database to store user’s credentials.

Another feature was the integration of a map using Google Map's API for the visualization of the daily activities' location (Fig.1), this allows the sales representative to easily get the best way to get to his meetings.

Closing a lead has some constraints, in order to close it as "won" for instance, implies that there is an order linked to that lead, otherwise the sales representative won't be able to finish it as won.

The implementation of **K**ey **P**erformance **I**ndicators (Figs. 1,2) adds value to this product by giving important statistics and performance metrics such as the total number of sales, total order revenue, top customers, top selling products, pending sales opportunities, etc, in general it provides an overall business performance overview.

Interoperability with Primavera

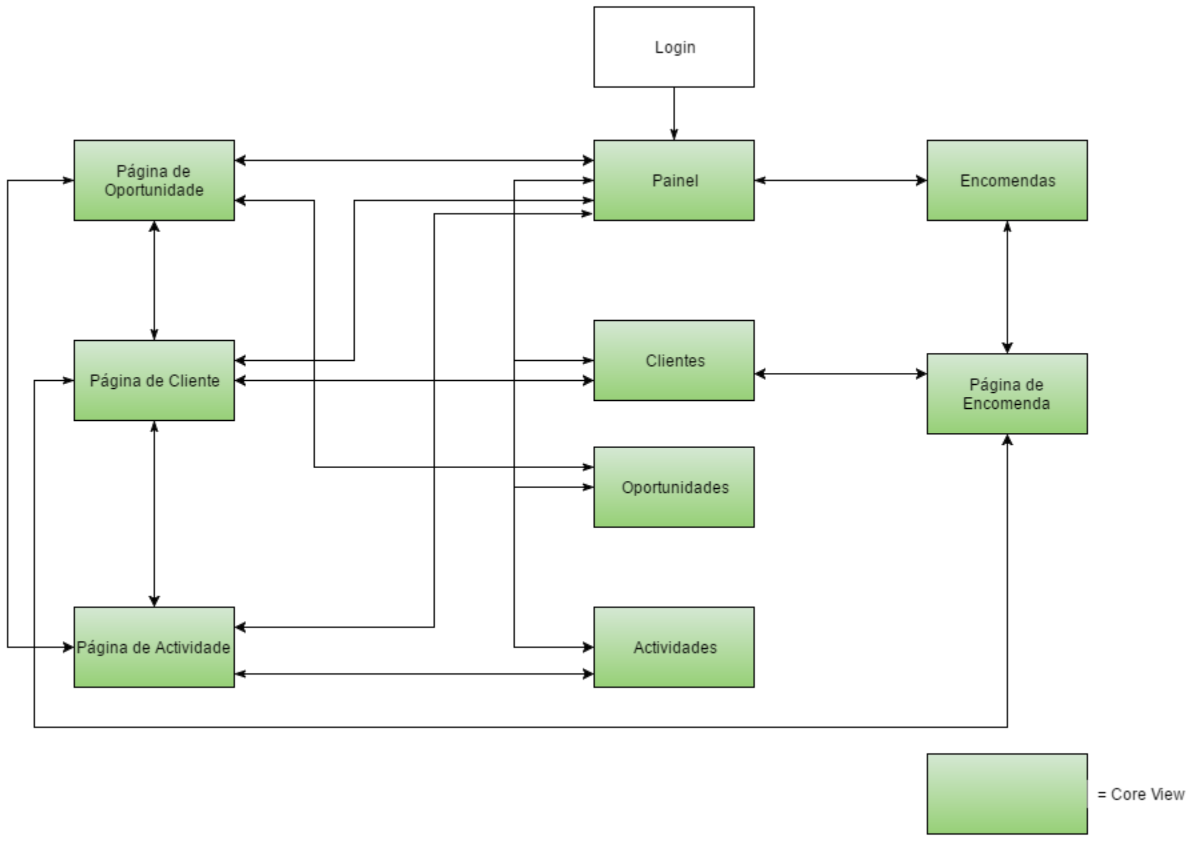
To operate with the Primavera ERP, we designed an API that simplifies interaction with the Primavera database. In this section, we will describe the different methods and paths that can be used with our API. It should be noted that in case of error, the API returns error codes (3XX ou 4XX) with the matching error message, otherwise it returns a 2XX code.

The following web services were implemented, and are described in the attachment section:

- getArtigos - list of products
- getArtigo - details of product
- getAtividades - list of activities
- getAtividade - details of activity
- criaAtividade - creates an activity
- atualizaAtividade - updates an activity
- apagaAtividade - deletes an activity
- getTiposAtividade - list of activity types
- getTipoAtividade - details of activity type
- getCatalogo - catalogue (list of short product informations)
- getCatalogoArtigo - few detauls of a product
- getClientes - list of clients
- getCliente - details of a client
- criaCliente - creates a client
- atualizaCliente - updates a client
- apagaCliente - deletes a client
- getClienteEncomendas - list of sales orders from a client
- getClienteAtividades - list of activities for a client
- getClienteOportunidades - list of leads/opportunities with a client
- getEncomendas - list of sales orders
- getEncomenda - details of a sales order
- criaEncomenda - creates a sales order
- anulaEncomenda - invalidates a sales order

- getExtratoEncomenda - returns a PDF transcript of a sales order
- getKPIs - global company KPIs
- getVendedorKPIs - KPIs of a specific sales representative
- getObjetivos - list of goals of all sales representatives
- getObjetivoVendedor - goal of a sales representative
- defineObjetivoVendedor - sets the goal of a sales representative
- getOportunidades - list of opportunities/leads
- getOportunidade - details of an opportunity/lead
- getAtividadesDeOportunidade - list of activities associated with a lead
- criaOportunidade - creates a lead
- atualizaOportunidade - updates a lead
- apagaOportunidade - deletes a lead
- getStockArtigos - list of stock entries for all products
- getStockArtigo - stock of a product (divided per stock location)
- getVendedores - list of sales representatives
- getVendedor - details of a sales representative
- getVenvedorAtividades - list of activities a sales representative has scheduled
- getVendedorEncomendas - list of sales orders a sales representative is responsible for
- getVendedorOportunidades - list of leads a sales representative is responsible for
- getVendedorObjetivo - goal of a sales representative
- getVendedorKPI - KPIs of a sales representative

Paths



System Architecture

The system follows a client-server architecture which is specified in the deployment diagram below.

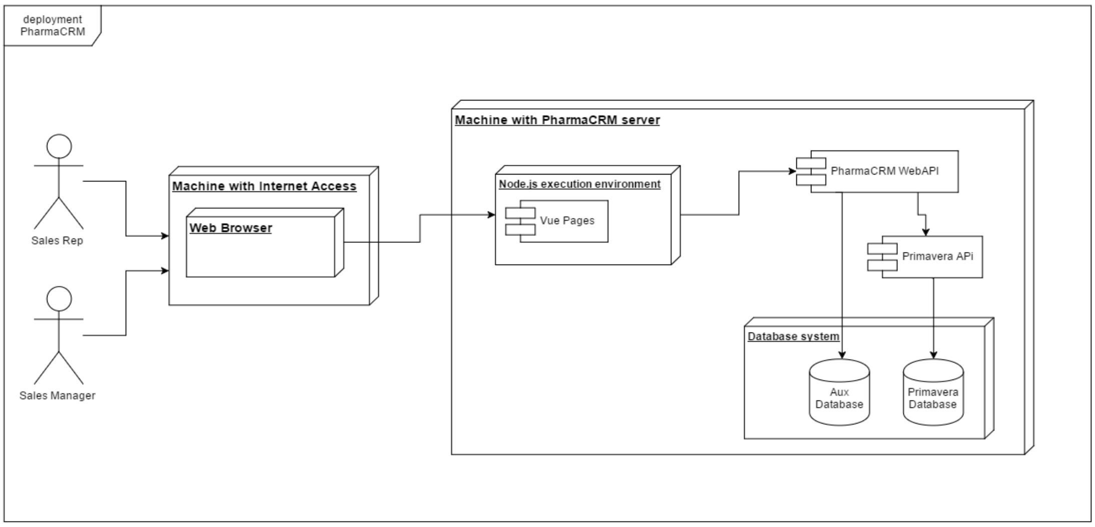


Fig. 11 - PharmaCRM Deployment diagram

Project Specifications vs. Delivered Project

Overall, we managed to accomplish most of the requirements that were proposed in the project specification. However some things were changed, as described below.

- The “Leads” section is now named “Opportunities” to better describe it. A sales opportunity is created as soon as a sales representative has a new contact and is aware of the needs of that person.
- We did not implement the search bar, which would allow a sales representative or an admin to quickly search activities, opportunities and sales orders by partial text matching.
- The KPIs chosen to be shown in the dashboard suffered slight changes.
- The final implementation allows the creation of new clients, instead of only allowing to edit them, as proposed in the project specification.
- Small changes were made to the interoperability with Primavera functions, to properly fulfill the system needs while taking into account the requirements of the Primavera software.

Lessons Learned

One of the most important aspects we learned with this project is the steps of work of a sales representative and how an ERP such as Primavera can help in that process.

Another key knowledge we obtained from this project is the critical role of the key performance indicators displayed in the dashboard page. The idea of having analytics of the whole system makes a company much more likely to substantially outperform their industry peers. The sales representative is now much more capable of making a self-examination of his work and results, as well as knowing what products sell better and what clients bring the most profit. The sales manager, being able to view KPIs globally or for a single sales representative, is also able to make much better managerial decisions.

Attachments

Attachment 1: web services tables

Webservice ID	getArtigos
Description	Returns the list of products available
Path + arguments	/api/artigos
Method	GET
Core View(s)	#1, #9, #10

Output	{
	{
	Codigo: <codigo>,
	Descricao: <descrição>,
	StockAtual: <stock>,
	PrecoUltimo: <preco_ultimo>,
	PrecoMedio: <preco_medio>,
	Iva: <iva>,
	PrazoEntrega: <prazo_entrega>,
	PVPs:
	{
	<pvp_0>,
	…
	}
	unidade: <unidade>,
	descricaoUnidade: <descricao_da_unidade>,
	desconto: <desconto>
	},
	...
	}

Webservice ID	getArtigo
Description	Returns a product with the given ID
Path + arguments	/api/{id}
Method	GET

Core View(s)	#9, #10
Input Example	{id} = A0001

Output	{
	Codigo: <codigo>,
	Descricao: <descrição>,
	StockAtual: <stock>,
	PrecoUltimo: <preco_ultimo>,
	PrecoMedio: <preco_medio>,
	Iva: <iva>,
	PrazoEntrega: <prazo_entrega>,
	PVPs:
	{
	<pvp_0>,
	…
	}
	unidade: <unidade>,
	descricaoUnidade: <descricao_da_unidade>,
	desconto: <desconto>
	}

Webservice ID	getAtividades
Description	Returns a list of activities (tasks) of the sales representatives
Path + arguments	/api/atividades
Method	GET
Core View(s)	#7
Output	{ { id: <id>, idTipoAtividade: <id_tipo_atividade>, estado: <estado>, dataInicio: <data_inicio>, dataFim: <data_fim>, resumo: <resumo>, descricao: <descricao>, idTipoEntidadePrincipal: <id_tipo_entidade_principal>, entidadePrincipal: <entidade_principal>, idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>, local: <local>

	}, ... }

Webservice ID	getAtividade
Description	Returns an activity with a specific ID
Path + arguments	/api/atividades/{id}
Method	GET
Core View(s)	#8
Input Example	{id} = 123

Output	{
	id: <id>,
	idTipoAtividade: <id_tipo_atividade>,
	estado: <estado>,
	dataInicio: <data_inicio>,
	dataFim: <data_fim>,
	resumo: <resumo>,
	descricao: <descricao>,
	idTipoEntidadePrincipal: <id_tipo_entidade_principal>,
	entidadePrincipal: <entidade_principal>,
	idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>,
	local: <local>
	}

Webservice ID	criaAtividade
Description	Creates an activity
Path + arguments	/api/atividades
Method	POST
Core View(s)	#7
Input Example	{ idTipoAtividade: <id_tipo_atividade>, estado: <estado>,

	dataInicio: <data_inicio>, dataFim: <data_fim>, resumo: <resumo>, descricao: <descricao>, idTipoEntidadePrincipal: <id_tipo_entidade_principal>, entidadePrincipal: <entidade_principal>, idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>, local: <local> }
Output	200 if OK

Webservice ID	atualizaAtividade
Description	Updates an activity to the given parameters
Path + arguments	/api/atividades/{id}
Method	PUT
Core View(s)	#8

Input Example	{
	id: <id>,
	idTipoAtividade: <id_tipo_atividade>,
	estado: <estado>,
	dataInicio: <data_inicio>,
	dataFim: <data_fim>,
	resumo: <resumo>,
	descricao: <descricao>,
	idTipoEntidadePrincipal: <id_tipo_entidade_principal>,
	entidadePrincipal: <entidade_principal>,
	idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>,
	local: <local>
	}

Output	200 if OK

Webservice ID	apagaAtividade
Description	Deletes an activity
Path + arguments	/api/atividades/{id}

Method	DELETE
Core View(s)	#8
Input Example	{id} = 123
Output	200 if OK

Webservice ID	getTiposAtividade
Description	Returns a list of activity types
Path + arguments	/api/atividades/tipos
Method	GET
Core View(s)	#7

Output	{
	  {
	    id: <id>,
	    descricao: <descricao>
	  },
	  ...
	}

Webservice ID	getTipoAtividade
Description	Returns an activity type with a specific ID
Path + arguments	/api/atividades/tipos/{id}
Method	GET
Core View(s)	#8
Input Example	{id} = 123

Output	{
	  id: <id>,
	  descricao: <descricao>
	}

Webservice ID	getCatalogo
Description	Returns the available catalogue (simpler list of products)
Path + arguments	/api/catalogo
Method	GET
Core View(s)	#9

Output	{
	  {
	    Codigo: <codigo>,
	    Descricao: <descrição>
	  },
	  ...
	}

Webservice ID	getCatalogoArtigo
Description	Returns the catalogue version of the product with a specific ID
Path + arguments	/api/catalogo/{id}
Method	GET
Core View(s)	#9
Input Example	{id} = A0001

Output	{
	  Codigo: <codigo>,
	  Descricao: <descrição>
	}

Webservice ID	getClientes
Description	Returns a list of customers
Path + arguments	/api/clientes

Method	GET
Core View(s)	#3

Output	{
	{
	CodCliente: <codigo_cliente>,
	Morada: <morada>,
	Localidade: <localidade>,
	Notas: <notas>,
	Nome: <nome>,
	NumContribuinte: <num_contribuinte>,
	Telefone: <telefone>,
	FaturacaoCliente:
	{
	ValorFaturado: <valor_faturado>,
	ValorPorFaturar: <valor_por_faturar>
	}
	},
	...
	}

Webservice ID	getCliente
Description	Returns a client with a specific ID
Path + arguments	/api/clientes/{id}
Method	GET
Core View(s)	#4
Input Example	{id} = 123
Output	{ CodCliente: <codigo_cliente>, Morada: <morada>, Localidade: <localidade>, Notas: <notas>, Nome: <nome>, NumContribuinte: <num_contribuinte>, Telefone: <telefone>, FaturacaoCliente: { ValorFaturado: <valor_faturado>,

	ValorPorFaturar: <valor_por_faturar> } }

Webservice ID	criaCliente
Description	Creates a new client with the given parameters
Path + arguments	/api/clientes
Method	POST
Core View(s)	#3

Input Example	{
	CodCliente: <codigo_cliente>,
	Morada: <morada>,
	Localidade: <localidade>,
	Notas: <notas>,
	Nome: <nome>,
	NumContribuinte: <num_contribuinte>,
	Telefone: <telefone>
	}

Output	200 if OK

Webservice ID	atualizaCliente
Description	Updates a client with the given parameters
Path + arguments	/api/clientes/{id}
Method	PUT
Core View(s)	#4
Input Example	{ CodCliente: <codigo_cliente>, Morada: <morada>, Localidade: <localidade>, Notas: <notas>, Nome: <nome>, NumContribuinte: <num_contribuinte>,

	Telefone: <telefone> }
Output	200 if OK

Webservice ID	apagaCliente
Description	Deletes a client with a specific ID
Path + arguments	/api/clientes/{id}
Method	DELETE
Core View(s)	#4
Input Example	{id} = 123
Output	200 if OK

Webservice ID	getClienteEncomendas
Description	Returns a list of orders by a client with a specific ID
Path + arguments	/api/clientes/{id}/encomendas
Method	GET
Core View(s)	#4
Input Example	{id} = 123
Output	{ { Faturada: <faturada>, Entidade: <entidade>, idOportunidade: <id_oportunidade>, NumeroDocumento: <num_documento>, Data: <data>, TotalMercadoria: <total_mercadoria>, Serie: <serie>, idResponsavel: <id_responsavel>, Filial: <filial>, Anulada: <anulada>, idInterno: <id_interno>,

	LinhasDocumento: {
	{
	CodigoArtigo: <codigo_artigo>,
	DescricaoArtigo: <descricao_artigo>,
	IdCabecaDocumento: <id_cabeca_documento>,
	Quantidade: <quantidade>,
	Unidade: <unidade>,
	Desconto: <desconto>,
	PrecoUnitario: <preco_unit>,
	TotalLiquido: <total_liquido>,
	TotalIliquido: <total_iliquido>,
	TotalIVA: <total_iva>
	},
	...
	}
	},
	...
	}

Webservice ID	getClienteAtividades
Description	Returns a list of activities associated with a client with a specific ID
Path + arguments	/api/clientes/{id}/atividades
Method	GET
Core View(s)	#4
Input Example	{id} = 123
Output	{ { id: <id>, idTipoAtividade: <id_tipo_atividade>, estado: <estado>, dataInicio: <data_inicio>, dataFim: <data_fim>, resumo: <resumo>, descricao: <descricao>, idTipoEntidadePrincipal: <id_tipo_entidade_principal>, entidadePrincipal: <entidade_principal>, idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>, local: <local>

	}, ... }

Webservice ID	getClienteOportunidades
Description	Returns a list of leads associated with a client with a specific ID
Path + arguments	/api/clientes/{id}/oportunidades
Method	GET
Core View(s)	#4
Input Example	{id} = 123

Output	{
	{
	id: <id>,
	codigo: <codigo>,
	descricao: <descricao>,
	numEncomenda: <num_encomenda>,
	entidade: <entidade>,
	tipoEntidade: <tipo_da_entidade>,
	dataCriacao: <data_criacao>,
	dataExpiracao: <data_expiracao>,
	vendedor: <vendedor>,
	valorTotalOV: <valor_total_da_oportunidade>,
	estado: <estado>
	},
	...
	}

Webservice ID	getEncomendas
Description	Returns a list of heads of sales orders (without included products)
Path + arguments	/api/encomendas
Method	GET
Core View(s)	#9

Output	{
	 {
	  Faturada: <faturada>,
	  Entidade: <entidade>,
	  idOportunidade: <id_oportunidade>,
	  NumeroDocumento: <num_documento>,
	  Data: <data>,
	  TotalMercadoria: <total_mercadoria>,
	  Serie: <serie>,
	  idResponsavel: <id_responsavel>,
	  Filial: <filial>,
	  Anulada: <anulada>,
	  idInterno: <id_interno>
	 },
	 ...
	}

Webservice ID	getEncomenda
Description	Returns a sales order with a specific ID
Path + arguments	/api/encomendas/{id}
Method	GET
Core View(s)	#10
Input Example	{id} = 15D3E72F-7A0A-11E6-A55F-080027184ECD

Output	{
	 Faturada: <faturada>,
	 Entidade: <entidade>,
	 idOportunidade: <id_oportunidade>,
	 NumeroDocumento: <num_documento>,
	 Data: <data>,
	 TotalMercadoria: <total_mercadoria>,
	 Serie: <serie>,
	 idResponsavel: <id_responsavel>,
	 Filial: <filial>,
	 Anulada: <anulada>,
	 idInterno: <id_interno>,
	 LinhasDocumento: {
	  {
	   CodigoArtigo: <codigo_artigo>,

	DescricaoArtigo: <descricao_artigo>, IdCabecaDocumento: <id_cabeca_documento>, Quantidade: <quantidade>, Unidade: <unidade>, Desconto: <desconto>, PrecoUnitario: <preco_unit>, TotalLiquido: <total_liquido>, TotalIliquido: <total_iliquido>, TotalIVA: <total_iva> }, ... } }

Webservice ID	criaEncomenda
Description	Creates a sales order with the given parameters
Path + arguments	/api/encomendas
Method	POST
Core View(s)	#9

Input Example	{
	Faturada: <faturada>,
	Entidade: <entidade>,
	idOportunidade: <id_oportunidade>,
	Serie: <serie>,
	idResponsavel: <id_responsavel>,
	Filial: <filial>,
	LinhasDocumento: {
	{
	CodigoArtigo: <codigo_artigo>,
	Quantidade: <quantidade>,
	Desconto: <desconto>,
	PrecoUnitario: <preco_unit>
	},
	...
	}
	}

Output	200 if OK

Webservice ID	anulaEncomenda
Description	Invalidades the sales order with a specific ID
Path + arguments	/api/encomendas/{id}
Method	DELETE
Core View(s)	#9
Input Example	{id} = 15D3E72F-7A0A-11E6-A55F-080027184ECD
Output	200 if OK

Webservice ID	getExtratoEncomenda
Description	Returns a PDF transcript with the details of a specific sales order
Path + arguments	/api/encomendas/{id}/pdf
Method	GET
Core View(s)	#10
Input Example	{id} = 15D3E72F-7A0A-11E6-A55F-080027184ECD
Output	HTTP response with a PDF file as content

Webservice ID	getKPIs
Description	Returns the KPI of the company
Path + arguments	/api/kpi
Method	GET
Core View(s)	#2
Output	{ NumTotalVendas: <num_vendas>, ValorTotalVendas: <valor_vendas>, ProdutosMaisVendidos:

	{
	<nome_artigo>: <valor_total_vendas>,
	...
	},
	NumVendasCompletas: <num_vendas_completas>,
	NumOportunidadesPendentes: <num_oportunidades_pendentes>,
	NovasOportunidades: <novas_oportunidades>,
	NumClientesAtivos: <num_clientes_ativos>,
	MelhoresClientes:
	{
	<nome_cliente>: <valor_total_vendas>,
	...
	}
	ValorMedioVendasPorVendedor: <valor_vendas_por_vendedor>,
	MelhoresVendedores:
	{
	<nome_vendedor>: <valor_total_vendas>,
	...
	}
	}

Webservice ID	getVendedorKPIs
Description	Returns the KPI of the sales representative with a specific ID
Path + arguments	/api/kpi/{id}
Method	GET
Core View(s)	#1
Input Example	{id} = 1
Output	{ idVendedor: <id_vendedor>, NumTotalVendas: <num_vendas>, ValorTotalVendas: <valor_vendas>, ProdutosMaisVendidos: { <nome_artigo>: <valor_total_vendas>, ... }, NumVendasCompletas: <num_vendas_completas>, NumOportunidadesPendentes: <num_oportunidades_pendentes>,

	NovasOportunidades: <novas_oportunidades>,
	NumClientesAtivos: <num_clientes_ativos>,
	MelhoresClientes:
	{
	<nome_cliente>: <valor_total_vendas>,
	...
	}
	}

Webservice ID	getObjetivos
Description	Returns a list of goals for all sales representatives
Path + arguments	/api/objetivos
Method	GET
Core View(s)	#9, #10

Output	{
	{
	Vendedor: <vendedor_id>,
	Valor: <valor>,
	ValorCumprido: <valor_cumprido>
	},
	...
	}

Webservice ID	getObjetivoVendedor
Description	Returns the goal of the sales representative with a specific ID
Path + arguments	/api/objetivos/{id}
Method	GET
Core View(s)	#9
Input Example	{id} = 1
Output	{ Vendedor: <vendedor_id>, Valor: <valor>,

	ValorCumprido: <valor_cumprido> }

Webservice ID	defineObjetivoVendedor
Description	Sets the goal of the sales representative with a specific ID
Path + arguments	/api/objetivos/{id}
Method	PUT
Core View(s)	#10

Input Example	{
	  Vendedor: <vendedor_id>,
	  Valor: <valor>
	}

Output	200 if OK

Webservice ID	getOportunidades
Description	Returns the list of leads/opportunities available
Path + arguments	/api/oportunidades
Method	GET
Core View(s)	#5

Output	{
	  {
	    id: <id>,
	    codigo: <codigo>,
	    descricao: <descricao>,
	    numEncomenda: <num_encomenda>,
	    entidade: <entidade>,
	    tipoEntidade: <tipo_da_entidade>,
	    dataCriacao: <data_criacao>,
	    dataExpiracao: <data_expiracao>,
	    vendedor: <vendedor>,
	    valorTotalOV: <valor_total_da_oportunidade>,
	    estado: <estado>

	}, ... }

Webservice ID	getOportunidade
Description	Returns the lead/opportunity with a specific ID
Path + arguments	/api/oportunidades/{id}
Method	GET
Core View(s)	#6
Input Example	{id} = 123

Output	{
	id: <id>,
	codigo: <codigo>,
	descricao: <descricao>,
	numEncomenda: <num_encomenda>,
	entidade: <entidade>,
	tipoEntidade: <tipo_da_entidade>,
	dataCriacao: <data_criacao>,
	dataExpiracao: <data_expiracao>,
	vendedor: <vendedor>,
	valorTotalOV: <valor_total_da_oportunidade>,
	estado: <estado>
	}

Webservice ID	getAtividadesDeOportunidade
Description	Returns a list of activities associated with the lead with a specific ID
Path + arguments	/api/oportunidades/{id}/atividades
Method	GET
Core View(s)	#6
Input Example	{id} = 123
Output	{

	{
	id: <id>,
	idTipoAtividade: <id_tipo_atividade>,
	estado: <estado>,
	dataInicio: <data_inicio>,
	dataFim: <data_fim>,
	resumo: <resumo>,
	descricao: <descricao>,
	idTipoEntidadePrincipal: <id_tipo_entidade_principal>,
	entidadePrincipal: <entidade_principal>,
	idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>,
	local: <local>
	},
	...
	}

Webservice ID	criaOportunidade
Description	Creates a new lead/opportunity
Path + arguments	/api/oportunidades
Method	POST
Core View(s)	#5

Input Example	{
	codigo: <codigo>,
	descricao: <descricao>,
	numEncomenda: <num_encomenda>,
	entidade: <entidade>,
	tipoEntidade: <tipo_da_entidade>,
	dataCriacao: <data_criacao>,
	dataExpiracao: <data_expiracao>,
	vendedor: <vendedor>,
	estado: <estado>
	}

Output	200 if OK

Webservice ID	atualizaOportunidade

Description	Updates the lead with a specific ID
Path + arguments	/api/oportunidades/{id}
Method	PUT
Core View(s)	#6

Input Example	{
	id: <id>,
	codigo: <codigo>,
	descricao: <descricao>,
	numEncomenda: <num_encomenda>,
	entidade: <entidade>,
	tipoEntidade: <tipo_da_entidade>,
	dataCriacao: <data_criacao>,
	dataExpiracao: <data_expiracao>,
	vendedor: <vendedor>,
	estado: <estado>
	}

Output	200 if OK

Webservice ID	apagaOportunidade
Description	Deletes the lead with a specific ID
Path + arguments	/api/oportunidades/{id}
Method	DELETE
Core View(s)	#6
Input Example	{id} = 123
Output	200 if OK

Webservice ID	getStockArtigos
Description	Returns the stock for all products available
Path + arguments	/api/stock
Method	GET

Core View(s)	#9, #10

Output	{
	{
	CodigoArtigo: <cod_artigo>,
	Stocks: {
	{
	Armazem: <armazem>,
	Lote: <lote>,
	StockAtual: <stock_atual>,
	Localizacao: <localizacao>,
	QuantidadeReservada: <quantidade_reservada>,
	UltimaContagem: <ultima_contagem>;
	DataUltimaContage: <data_ultima_contagem>,
	PrecoMedio: <preco_medio>,
	PrecoUltimo: <preco_ultimo>,
	BloqueadoInventario: <bloqueado_inventario>
	},
	...
	}
	},
	…
	}

Webservice ID	getStockArtigo
Description	Returns the stock of the product with a specific ID
Path + arguments	/api/stock/{id}
Method	GET
Core View(s)	#9, #10
Input Example	{id} = A0001

Output	{
	CodigoArtigo: <cod_artigo>,
	Stocks: {
	{
	Armazem: <armazem>,
	Lote: <lote>,
	StockAtual: <stock_atual>,
	Localizacao: <localizacao>,
	QuantidadeReservada: <quantidade_reservada>,

	UltimaContagem: <ultima_contagem>; DataUltimaContage: <data_ultima_contagem>, PrecoMedio: <preco_medio>, PrecoUltimo: <preco_ultimo>, BloqueadoInventario: <bloqueado_inventario> }, ... } }

Webservice ID	getVendedores
Description	Returns a list of sales representatives
Path + arguments	/api/vendedores
Method	GET
Core View(s)	#2

Output	{
	{
	cod: <codigo>,
	nome: <nome>
	},
	…
	}

Webservice ID	getVendedor
Description	Returns the sales representative with a specific ID
Path + arguments	/api/vendedores/{id}
Method	GET
Core View(s)	#1, #2
Input Example	{id} = 123
Output	{ cod: <codigo>, nome: <nome>

	}

Webservice ID	getVendedorAtividades
Description	Returns a list of activities/tasks for a specific sales representative. The date parameters are optional, and allow a timespan to be defined, where activities should be.
Path + arguments	/api/vendedores/{id}/atividades?dataInicio={dataIni}&dataFim={dataFim}
Method	GET
Core View(s)	#1
Input Example	{id} = 123 {dataIni} = 2010-11-15 {dataFim} = 2016-11-15

Output	{
	  {
	    id: <id>,
	    idTipoAtividade: <id_tipo_atividade>,
	    estado: <estado>,
	    dataInicio: <data_inicio>,
	    dataFim: <data_fim>,
	    resumo: <resumo>,
	    descricao: <descricao>,
	    idTipoEntidadePrincipal: <id_tipo_entidade_principal>,
	    entidadePrincipal: <entidade_principal>,
	    idCabecalhoOportunidadeVenda: <id_cabecalho_op_venda>,
	    local: <local>
	  },
	  ...
	}

Webservice ID	getVendedorEncomendas
Description	Returns a list of sales orders a sales representative is responsible for
Path + arguments	/api/vendedores/{id}/encomendas
Method	GET

Core View(s)	#1, #9
Input Example	{id} = 123

Output	{
	{
	Faturada: <faturada>,
	Entidade: <entidade>,
	idOportunidade: <id_oportunidade>,
	NumeroDocumento: <num_documento>,
	Data: <data>,
	TotalMercadoria: <total_mercadoria>,
	Serie: <serie>,
	idResponsavel: <id_responsavel>,
	Filial: <filial>,
	Anulada: <anulada>,
	idInterno: <id_interno>,
	LinhasDocumento: {
	{
	CodigoArtigo: <codigo_artigo>,
	DescricaoArtigo: <descricao_artigo>,
	IdCabecaDocumento: <id_cabeca_documento>,
	Quantidade: <quantidade>,
	Unidade: <unidade>,
	Desconto: <desconto>,
	PrecoUnitario: <preco_unit>,
	TotalLiquido: <total_liquido>,
	TotalIliquido: <total_iliquido>,
	TotalIVA: <total_iva>
	},
	...
	}
	},
	...
	}

Webservice ID	getVendedorOportunidades
Description	Returns the leads associated with a specific sales representative
Path + arguments	/api/vendedores/{id}/oportunidades
Method	GET

Core View(s)	#1, #5
Input Example	{id} = 123

Output	{
	{
	id: <id>,
	codigo: <codigo>,
	descricao: <descricao>,
	numEncomenda: <num_encomenda>,
	entidade: <entidade>,
	tipoEntidade: <tipo_da_entidade>,
	dataCriacao: <data_criacao>,
	dataExpiracao: <data_expiracao>,
	vendedor: <vendedor>,
	valorTotalOV: <valor_total_da_oportunidade>,
	estado: <estado>
	},
	...
	}

Webservice ID	getVendedorObjetivo
Description	Returns the goal of the sales representative with a specific ID
Path + arguments	/api/vendedores/{id}/objetivo
Method	GET
Core View(s)	#1
Input Example	{id} = 123

Output	{
	Vendedor: <vendedor_id>,
	Valor: <valor>,
	ValorCumprido: <valor_cumprido>
	}

Webservice ID	getVendedorKPI
Description	Returns the KPIs of the sales representative with a specific ID

Path + arguments	/api/vendedores/{id}/kpi
Method	GET
Core View(s)	#1
Input Example	{id} = 123

Output	{
	idVendedor: <id_vendedor>,
	NumTotalVendas: <num_vendas>,
	ValorTotalVendas: <valor_vendas>,
	ProdutosMaisVendidos:
	{
	<nome_artigo>: <valor_total_vendas>,
	...
	},
	NumVendasCompletas: <num_vendas_completas>,
	NumOportunidadesPendentes: <num_oportunidades_pendentes>,
	NovasOportunidades: <novas_oportunidades>,
	NumClientesAtivos: <num_clientes_ativos>,
	MelhoresClientes:
	{
	<nome_cliente>: <valor_total_vendas>,
	...
	}
	}