



# Senior

Tecnologia para Gestão



## Gestão de Relacionamento | CRM

**MANUAL WSDL - Webservices**

Ver 1 – 01 de Dezembro de 2016

Sede: Rua São Paulo, 825 - Victor Konder  
Blumenau/SC - Brasil - CEP 89012-001  
Filiais em: MS, PE, RS, MG, SP e RJ.  
[senior.com.br](http://senior.com.br)



# Conceitos

## 1.1 O que é um Webservice

Web Services são amplamente utilizados como uma solução para integrar Sistemas, auxiliando na comunicação entre aplicações de plataformas diferentes. Com este recurso é possível que novas aplicações possam ser integradas com aplicações já existentes e que Sistemas desenvolvidos em plataformas distintas sejam compatíveis.

O CRM Senior permite integração de todas as suas rotinas e tabelas através da metodologia WSDL (*Web Services Description Language*), que é um padrão homologado pelo W3C baseado em XML que visa padronizar as descrições das funcionalidades oferecidas por web services de forma independente de plataforma ou linguagem.

Os Web Services fornecidos pela ferramenta denominada WConnect são componentes que permitem as aplicações enviar e receber dados do ambiente do CRM Senior do cliente no formato XML, permitindo que quaisquer aplicações podem utilizar sua própria linguagem, que será traduzida para o XML, fazendo com que em certo ponto utilizem a mesma linguagem para comunicação.

## 1.2 – Funcionamento dos Webservices disponíveis no WConnect

O WConnect e seu conjunto de Web Services baseiam-se no protocolo SOAP (Simple Object Access Protocol). O protocolo SOAP utiliza geralmente o HTTP para transferência das informações. Com isto, em vez de usar HTTP para pedir uma página HTML para ser visualizada em um navegador, o SOAP envia uma mensagem de XML através do pedido HTTP e recebe uma resposta, se existir, através da resposta do HTTP.

## 1.3 Estrutura dos Webservices

Os webservices do WConnect são compostos basicamente de 2 itens principais: GETs e SETs.

Os métodos do webservice são os serviços ou rotinas que serão executadas pelo WConnect. Os parâmetros de entrada e saída são informações que, no caso das entradas (SETs) são dados que o WConnect pode utilizar para filtros, ordenações, etc, enquanto que os parâmetros de saída (GETs) normalmente são as matrizes de dados em caso de chamadas que listam os registros de uma determinada tabela do sistema (Ex: array com todas as cidades cadastradas no CRM).

Os parâmetros de entrada e saída podem ser diferentes para cada método disponível no WConnect, e estão listados neste documento. Note que alguns desses parâmetros podem ser obrigatórios, e nestes casos é de extrema importância que estes dados sejam verificados para a correta execução da chamada do webservice.

**ATENÇÃO: para utilização do WConnect, é necessário um usuário nominal “integração”, ATIVO no ambiente do CRM Senior do cliente.**

# Métodos - Set

## 1. setConta

Permite o cadastro de uma conta no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 1.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setConta

### 1.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
contaIdERP	int	ID complementar caso esteja integrando com outro ERP.
contaStatus	int	[Obrig] – 0 = Inativo, 1 = Ativo, 2 = Baixado, 3 = Bloqueado
contaRazaoSocial	string	[Obrig] – Razão Social (PJ) ou Nome (PF)
contaFantasia	string	Nome de Fantasia
contaFilial	string	Possibilidade de filtro pelo Email da conta no CRM Senior
contaGrupoEmpresa	int	Agrupamento por grupo de empresa (ID do Grupo)
contaTipoConta	int	[Obrig] – 1 = Lead, 2 = Prospect, 3 = Cliente, 4 = Outro
contaTipoPessoa	int	[Obrig] – 1 = Jurídica, 2 = Física
contaRating	int	[Obrig] – 1 = A, 2 = B, 3 = C, 4 = D
contaPorte	int	[Obrig] – 0 = Indeterminado, 1 = Microempresa, 2 = Pequeno Porte, 3 = Médio Porte, 4 = Grande Porte
contaEndereco	string	Endereço da Conta
contaNumero	string	Número do Endereço da Conta
contaComplemento	string	Complemento do Endereço da Conta
contaBairro	string	Bairro da Conta

contaCidade	int	[Obrig] – Id IBGE da Cidade. Ex: 4202404 (Blumenau) (Informação contida no getCidades)
contaCidadeUf	string	[Obrig] – UF da Cidade da Conta
contaPais	string	[Obrig] – País da Conta
contaCep	string	CEP da conta (9999-999)
contaInscricaoEstadual	string	Inscrição Estadual da Conta
contaInscricaoMunicipal	string	Inscrição Municipal da Conta
contaCnpj	string	CNPJ da Conta. Padrão COM PONTOS 99.999.999/9999-99. Se Pessoa Jurídica, campo obrigatório
contaCpf	string	CPF da Conta. Padrão COM PONTOS 999.999.999-99. Se Pessoa Física, campo obrigatório
contaRg	string	RG da Conta (PF)
contaAniversario	string	Aniversário da Conta (DD/MM)
contaTelefone	string	Número de Telefone 1
contaTelefone2	string	Número de Telefone 2
contaTelefone3	string	Número de Telefone 3
contaFax	string	Número de Fax
contaCelular	string	Número de Celular
contaUrl	string	Website (URL) da conta
contaEmailGeral	string	[Obrig] - Email geral da conta
contaRecebeCampanha	string	0 = Não, 1 = Sim
contaRecebePesquisa	string	0 = Não, 1 = Sim
contaRamoAtividadeId	string	[Obrig] – ID do Ramo de Atividade da Conta (Informação contida no getRamosAtividade)
contaOrigemId	string	[Obrig] – ID da Origem da Conta (Informação contida no getOrigemConta)
contaObservacoes	string	Observações da Conta

### 1.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	<b>[sucesso]</b> OU <b>[erro]</b> Retorna a mensagem de acordo com o status da consulta.
[mensagem]	Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).
[id_registro]	<b>[99999]</b> ID atribuído ao registro (Caso o retorno seja sucesso).

## 1.4 Exemplo de utilização (Utilizando PHP + NuSOAP)

```
//Adiciona o nuSOAP
require_once('../nusoap/nusoap.php');

//Constantes de conexão
define('WL_CAMINHO_SOAP','http://demo.workcrm.com.br/wconnect/WConnect_WSDL.php?wsd');

//Definição da localização do arquivo WSDL
$client = new nusoap_client(WL_CAMINHO_SOAP, true);

//verifica se ocorreu erro na criação do objeto
$error = $client->getError();

//Caso ocorra erro na conexão ao WSDL
if ($error) {

    echo "Erro no construtor <pre>".$error."</pre>";

}

$client->setDefaultRpcParams(true);

//Define o tempo de resposta
$client->response_timeout = 25;

//Parâmetros a enviar
$params = array('conta' => array('contaIdERP' => '99998',
                                'contaStatus' => '1',
                                'contaRazaoSocial' => 'Work Labs Ltda',
                                'contaFantasia' => 'Work Labs Tecnologia',
                                'contaFilial' => 'Matriz',
                                'contaGrupoEmpresa' => '0',
                                'contaTipoConta' => '2',
                                'contaTipoPessoa' => '1',
                                'contaRating' => '4',
                                'contaPorte' => '3',
                                'contaEndereco' => 'Rua Ingo Hering',
                                'contaNumero' => '20',
                                'contaComplemento' => 'Sala 203',
                                'contaBairro' => 'Centro',
                                'contaCidade' => '4202404',
                                'contaCidadeUf' => 'SC',
                                'contaPais' => 'Brasil',
                                'contaCep' => '89010-205',
```

```

'contaInscricaoEstadual' => '99999999999',
'contaInscricaoMunicipal' => '888888888888',
'contaCnpj' => '99.999.999/9999-99',
'contaCpf' => '999.999.999-99',
'contaRg' => '99999999999',
'contaAniversario' => '02/08',
'contaTelefone' => '(47) 4102-2262',
'contaTelefone2' => '(99) 9999-9999',
'contaTelefone3' => '(99) 9999-9999',
'contaFax' => '(99) 9999-9999',
'contaCelular' => '(99) 99999-9999',
'contaOperadora' => '0',
'contaUrl' => 'www.workcrm.com.br',
'contaEmailGeral' =>

'suporte@workabs.com.br',

'contaRecebeCampanha' => '1',
'contaRecebePesquisa' => '1',
'contaRamoAtividadeId' => '2',
'contaOrigemId' => '1',
'contaObservacoes' => 'Inserido via

webservice'

);

//Classificações
$result = $client->call('setConta', $parametros);

//Verifica se ocorreu falha na chamada do método
if ($client->fault){

    echo "Falha<pre>".print_r($result)."</pre>";

}

else {

    //verifica se ocorreu erro
    $err = $client->getError();

    if ($err) {

        echo "Erro: <pre>".$err."</pre>";

    }

    else {

        //Exibe a string do resultado
        echo "<pre>";
        print_r($result);
        echo "</pre>";

    }

}
}

```

## 2. setContasResponsavel

Permite o cadastro de um Responsável de Conta no CRM Senior. Os campos obrigatórios devem ser respeitadas, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 2.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setContasResponsavel

### 2.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
contaResponsavelIdErp	string	ID da conta do ERP que será atribuído o responsável. Obrigatório se contaResponsavelId, contaResponsavelCnpjCpf e contaResponsavelTipoPessoa em branco.
contaResponsavelId	int	ID da conta do CRM Senior que será atribuído o responsável. Obrigatório se contaResponsavelIdErp, contaResponsavelCnpjCpf e contaResponsavelTipoPessoa em branco.
contaResponsavelCnpjCpf	string	CNPJ ou CPF da conta do CRM Senior que será atribuído o responsável. Obrigatório se contaResponsavelIdErp e contaResponsavelId em branco.
contaResponsavelTipoPessoa	int	1 = Cnpj e 2 = Cpf. Obrigatório se contaResponsavelCnpjCpf preenchido e contaResponsavelIdErp e contaResponsavelId em branco.
contaResponsavelTipoResp	int	[Obrig] – 1 = Usuário e 2 = Grupo.
contaResponsavelRespId	int	[Obrig] – Id do Usuário do CRM Senior ou do Grupo de Usuário do CRM Senior (Informação contida no getUsers ou getGrupos)

### 2.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
-------------	----------



[status]	<b>[sucesso]</b> OU <b>[erro]</b> <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>
[id_registro]	<b>[99999]</b> <i>ID atribuído ao registro (Caso o retorno seja sucesso).</i>

### 3. setContasCaracteristica

Permite o cadastro de uma Característica da Conta no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

#### 3.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setContasCaracteristica

#### 3.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
contaCaracteristicaIdErp	string	[Obrig] - ID da conta do ERP que será atribuído o a característica da conta
contaCaracteristicaId	int	[Obrig] – Id da Categoria da Característica da Conta (Informação contida no getCaracteristica)
contaCaracteristicaDetalheId	int	[Obrig] – Id do Detalhamento da Característica da Conta (Informação contida no getCaracteristicaDetalhe)

#### 3.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	[sucesso] OU [erro] <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>
[id_registro]	[99999] <i>ID atribuído ao registro (Caso o retorno seja sucesso).</i>

## 4. setContatos

Permite o cadastro de um Contato da Conta no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 4.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setContatos

### 4.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
contatoContaIdErp	string	[Obrig] - ID da conta do ERP que será atribuído o a contato da conta
contatoNome	string	[Obrig] – Nome do Contato
contatoTratamento	string	Nome de Tratamento do Contato
contatoDepartamentoId	int	Id do Departamento do Contato (Informação contida em getDepartamentos)
contatoFuncaoId	int	Id da Função do Contato (Informação contida em getFuncoes)
contatoOrigemContatoId	int	Id da Origem do Contato (Informação contida em getOrigemContato)
contatoCep	string	CEP do contato (9999-999)
contatoEndereco	string	Endereço do contato
contatoNumero	string	Número do Endereço da Conta
contatoComplemento	string	Complemento do Endereço da Conta
contatoBairro	string	Bairro da Conta
contatoCidade	int	[Obrig] – Id IBGE da Cidade. Ex: 4202404 (Blumenau) (Informação contida no getCidades)
contatoCidadeUf	string	[Obrig] – UF da Cidade da Conta

contatoPais	string	[Obrig] – País da Conta
contatoCpf	string	CPF da Conta. Padrão COM PONTOS 999.999.999-99
contatoRg	string	RG da Conta (PF)
contatoTelefone	string	Número de Telefone do contato
contatoRamal	string	Número do Ramal do contato
contatoFax	string	Número do Fax do contato
contatoCelular	string	Número do Celular do contato
contatoEmail	string	E-mail do contato
contatoRecebeCampanha	int	0 = Não, 1 = Sim
contatoRecebePesquisa	int	0 = Não, 1 = Sim
contatoSkype	string	Nome Skype do contato
contatoTwitter	String	Endereço Twitter do contato
contatoFacebook	String	Endereço Facebook do contato
contatoAniversario	string	Aniversário do Contato (DD/MM)
contatoDataNascimento	String	Data de Aniversário do Contato (AAAA-MM-DD)
contatoNivelDecisao	Int	1 = Nenhum, 2 = Usuário, 3 = Decisor Financeiro, 4 = Decisor Técnico, 5 = Recomenda
contatoParecerEmpresa	Int	1 = Favorável a empresa, 2 = Indiferente a Empresa, 3 = Não Favorável a empresa
contatoMotivoParecerEmpresa	String	Obrigatório caso contatoParecerEmpresa = 3, descritivo do parecer negativo
contatoObservação	String	Observação sobre o contato

### 4.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
-------------	----------

[status]	<b>[sucesso]</b> OU <b>[erro]</b> <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>
[id_registro]	<b>[99999]</b> <i>ID atribuído ao registro (Caso o retorno seja sucesso).</i>

## 5. setOportunidades

Permite o cadastro de um Contato de uma Oportunidade no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 5.1 Chamada do Método

<b>URL</b>	[ambiente_work_crm]/wconnect/wsd1/WConnect_WSDL.php?wsdl
<b>Método</b>	setOportunidades

### 5.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
oportunidadeData	string	[Obrig] – Data de abertura da oportunidade. Ex: 2016-02-16
oportunidadeHora	string	[Obrig] – Hora de abertura da oportunidade. Ex: 14:14:14
oportunidadeContaId	int	[Obrig] – Id da conta para o qual será aberta a oportunidade. Ex: 222. Informação contida no getContas
oportunidadeContatoId	int	Id do Contato referencia para a oportunidade que será aberta. Ex: 221. Informação contida no getContatos
oportunidadeNome	string	[Obrig] – Breve descrição do que será a oportunidade. Ex: Aquisição do CRM Senior
oportunidadeTipoId	int	[Obrig] – Id do tipo da oportunidade que será aberta. Ex: 1. Informação contida no getTiposOportunidade
oportunidadeSubtipoId	int	Id do subtipo da oportunidade que será aberta. Ex: 1. Informação contida no getSubtiposOportunidade
oportunidadeOrigemId	int	[Obrig] – Id da origem da oportunidade que será aberta. Ex: 1. Informação contida no getOrigemOportunidade

oportunidadeModalidadeNegocioId	int	[Obrig] – Id da modalidade de negócios da oportunidade que será aberta. Ex: 1. Informação contida no getModalidadeNegocios
oportunidadeUsuarioId	int	Id do usuário responsável pela oportunidade que será aberta. Ex: 1. Informação contida no getUsuarios
oportunidadeUsuarioIdERP	int	Id ERP do usuário responsável pela oportunidade que será aberta. Ex: 1.
oportunidadeEstagioId	int	[Obrig] – Etapa da Oportunidade. 1 = Prospectando, 2 = Em Proposta, 3 = Em Análise, 4 = Em Negociação, 5 = Fechada Concretizada, 6 = Fechada Perdida
oportunidadeMotivoPerdaId	int	Id do Motivo de Perda. Ex: 1. Informação contida no getMotivosPerda. Obrigatório se Estagioid = 6.
oportunidadeMotivoFechamentoId	int	Id do Motivo de Fechamento. Ex: 1. Informação contida no getMotivosFechamento. Obrigatório se Estagioid = 5.
oportunidadeProbabilidade	int	[Obrig] Probabilidade da oportunidade. Ex: 50.
oportunidadeNumeroPedido	string	[Obrig] Numero do Pedido da Oportunidade. Ex: 1.
oportunidadeDataPrevistaFechamento	string	[Obrig] Data prevista para fechamento da oportunidade. Ex: 2016-02-16.
oportunidadeDataEfetivaFechamento	string	Data efetiva do fechamento da oportunidade. Ex: 2016-02-16.
oportunidadeDataPrevistaEntrega	string	Data prevista para entrega da oportunidade. Ex: 2016-02-16.
oportunidadeDataEfetivaEntrega	string	Data efetiva da entrega da oportunidade. Ex: 2016-02-16.
oportunidadeValorPrevisto	string	[Obrig] Valor Previsto da Oportunidade. Ex: 500.00
oportunidadeValorFechado	String	Valor Fechado da Oportunidade. Ex: 600.00
oportunidadeValorPrevistoMoeda	string	Valor Previsto da Oportunidade em Moeda Estrangeira. Ex: 500.00
oportunidadeValorFechadoMoeda	string	Valor Fechado da Oportunidade em Moeda Estrangeira. Ex: 500.00
oportunidadeQuantitativoPrevisto	string	Quantitativo previsto para a oportunidade. Ex: 1.

oportunidadeQuantitativoRealizado	string	Quantitativo realizado para a oportunidade. Ex: 1.
oportunidadeHistorico	string	Histórico da oportunidade. Ex: Foi realizado contato com o Cliente.
oportunidadeDescricao	string	Descrição da oportunidade. Ex: Foi vendida um software ao cliente.

### 5.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	<b>[sucesso]</b> OU <b>[erro]</b> <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>
[id_registro]	<b>[99999]</b> <i>ID atribuído ao registro (Caso o retorno seja sucesso).</i>



## 6. setOcorrencias

Permite o cadastro de um Contato de uma Ocorrência no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 6.1 Chamada do Método

URL	[ambiente_work_crm]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setOcorrencias

### 6.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
UsuarioLogin	string	[Obrig] – Usuário de integração “integracao”
UsuarioSenha	string	[Obrig] – Senha do usuário de integração. OBS: Deve ser enviado num hash md5.
ocorrenciaUsuarioId	int	[Obrig] – Id do usuário responsável
ocorrenciaContaId	int	[Obrig] Id da conta
ocorrenciaNome	string	Breve descrição do que será a ocorrência
ocorrenciaTipoId	int	[Obrig] – Id do tipo da ocorrencia que será aberta. Ex: 1. Informação contida no getTiposOcorrencia
ocorrenciaSubtipoId	int	Id do subtipo da ocorrencia que será aberta. Ex: 1. Informação contida no getSubtiposOcorrencia
ocorrenciaOrigemId	int	[Obrig] – Id da origem da ocorrencia que será aberta. Ex: 1. Informação contida no getOrigemOcorrencia
ocorrenciaDepartamentoId	int	[Obrig] – Id do departamento alocado para a ocorrência. Ex: 1. Informação contida no getDepartamentos
ocorrenciaPrioridade	int	Prioridade para a resolução da Ocorrência 1 = alta 2 = média 3 = baixa
ocorrenciaDescricao	string	[Obrig] - Detalhamento da ocorrência

ocorrenciaStatusId	int	[Obrig] – Status da ocorrência. Ex: 1. Informação contida no getStatusOcorrencia.
ocorrenciaDataInicioPrevisto	date	Data para início previsto. Formado YYYY-MM-DD
ocorrenciaHoraInicioPrevisto	time	Hora para início previsto. Formato HH:MM
ocorrenciaDataInicioEfetivo	date	Data para início efetivo. Formado YYYY-MM-DD
ocorrenciaHoraInicioEfetivo	time	Hora para início efetivo. Formato HH:MM
ocorrenciaDataResolucaoPrevisto	date	Data para resolução previsto. Formado YYYY-MM-DD
ocorrenciaHoraResolucaoPrevisto	time	Hora para resolução previsto. Formato HH:MM
ocorrenciaResolvida	bool	Ocorrência Resolvida 0 = Não 1 = Sim
ocorrenciaDataResolucaoEfetivo	date	Data da resolução efetiva. Formado YYYY-MM-DD
ocorrenciaHoraResolucaoEfetivo	time	Hora da resolução início efetiva. Formato HH:MM
ocorrenciaResolucao	string	Detalhamento da resolução da ocorrência
ocorrenciaDisposicaoId	int	Id da disposição da ocorrência. Ex: 1. Informação contida no getDisposicaoOcorrencia.
ocorrenciaCusto	decimal	Custo envolvido na ocorrência. Formato 9999999.99 (ponto separador decimal).
ocorrenciaNFEntrada	string	Numero da NF de entrada
ocorrenciaNFSaida	string	Número da NF de saída
ocorrenciaTransportadoraId	int	Id da transportadora. Ex: 1. Informação contida no getTransportadoras

### 6.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	[sucesso] OU [erro] <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>

[id\_registro]

[99999]

ID atribuído ao registro (Caso o retorno seja sucesso).

## 7. setDocumentosOportunidade

Permite o cadastro de um Documento de uma Oportunidade no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 7.1 Chamada do Método

<b>URL</b>	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
<b>Método</b>	setDocumentosOportunidade

### 7.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
documentoOpOportunidadeId	int	[Obrig] – Id da Oportunidade em que o documento será vinculado. Ex: 1.
documentoOpTipoId	Int	[Obrig] – Id do Tipo de Documento que será adicionado. Ex: 1. Informação contida no getTipoDocumento.
documentoOpData	string	[Obrig] – Data de adição do documento. Ex: 2016-02-16
documentoOpHora	string	[Obrig] – Hora de adição do documento. Ex: 14:14:14
documentoOpTitulo	string	[Obrig] – Título do Documento. Ex: Documento de Integração.
documentoOpNumero	string	Identificação do documento. Ex: 1.
documentoOpRevisão	string	Número de revisão do Documento. Ex: 2
documentoOpTipoEnvio	int	[Obrig] – Tipo de Envio. 1 = Ftp configurado no CRM Senior, 2 = Link Externo.
documentoOpUrlExterna	string	Url de localização do arquivo. Ex: <a href="http://www.meulocal.com.br/8d93kdkdkr/arquivo.pdf">http://www.meulocal.com.br/8d93kdkdkr/arquivo.pdf</a> . Obrigatório se Tipo de Envio = 2.

documentoOpArquivoFtp	string	Nome do arquivo no Ftp configurado no CRM Senior. Ex: arquivo.pdf. Obrigatório se Tipo de Envio = 1.
documentoOpObservacoes	string	Descrição do Documento. Ex: Arquivo que contem dados da Integração.

### 7.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	[sucesso] OU [erro] <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>
[id_registro]	[99999] <i>ID atribuído ao registro (Caso o retorno seja sucesso).</i>

## 8. setProdutosOportunidade

Permite o cadastro de um Produto em uma Oportunidade no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 8.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setProdutosOportunidade

### 8.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
produtoOpOportunidadeId	int	[Obrig] – Id da Oportunidade em que o produto será vinculado. Ex: 1.
produtoOpId	int	[Obrig] – Id do produto que será vinculado a oportunidade. Ex: 1. Informação contida em getProdutos.
produtoOpQuantidade	string	[Obrig] – Quantidade do produto a ser adicionado a oportunidade. Ex: 5.
produtoOpValorVenda	string	[Obrig] – Valor de venda do produto a ser adicionado na oportunidade. Ex: 10.00
produtoOpValorDesconto	string	Valor de desconto do produto a ser adicionado na oportunidade. Ex: 5.00
produtoOpObservacoes	string	Observações sobre a inclusão do produto. Ex: O produto deve ser produzido conforme especificação que segue.

### 8.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	[sucesso] OU [erro] <i>Retorna a mensagem de acordo com o status da consulta.</i>

[mensagem]	Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).
[id_registro]	<b>[99999]</b> ID atribuído ao registro (Caso o retorno seja sucesso).

## 9. setProdutos

Permite o cadastro de um Produto no CRM Senior. Os campos obrigatórios devem ser respeitados, conforme os parâmetros de entrada. O retorno será um Array contendo as chaves abaixo.

### 9.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	setProdutos

### 9.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
produtoNome	string	[Obrig] – Nome do Produto. Ex: Notebook.
produtoPartNumber	string	Part Number do Produto. Ex: 111.234.3232.
produtoCategoriaId	int	[Obrig] – Id da Categoria do Produto. Ex: 1. Informação contida no getCategoryasProduto.
produtoUnidade	int	[Obrig] – Unidade de Medida do Produto. Ex: 1. Informação contida no getUnidadeMedida.
produtoPrecoVenda	string	Valor de venda do produto. Ex: 5.00
produtoQuantidadePadrao	string	Quantidade padrão do produto. Ex: 2.
produtoObservacoes	string	Observações do produto. Ex: O produto possui medida y.

### 9.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	[sucesso] OU [erro] <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>



[id\_registro]

[99999]

ID atribuído ao Produto no CRM Senior (Caso o retorno seja sucesso).

# Métodos - Get

## 1. getConta

Retorna um array com os dados de uma ou mais contas cadastradas no CRM Senior.  
Diversos parâmetros de filtros se aplicam caso necessário:

### 1.1 Chamada do Método

URL	[ambiente_crm_senior]/wconnect/wsd1/WConnect_WSDL.php?wsdl
Método	getContas

### 1.2 Parâmetros de Entrada (Request)

Parâmetro	Tipo	Descrição
ContasId	int	Possibilidade de filtro pelo ID da conta
contaIdERP	int	Possibilidade de filtro pelo ID ERP da conta
contaCNPJ	string	Possibilidade de filtro pelo CNPJ da conta
contaCPF	string	Possibilidade de filtro pelo CPF da conta
contaEmail	string	Possibilidade de filtro pelo Email da conta

### 1.3 Parâmetros de Saída (Response -- Array)

Chave Array	Conteúdo
[status]	[sucesso] OU [erro] <i>Retorna a mensagem de acordo com o status da consulta.</i>
[encontrado]	[TRUE] OU [FALSE] <i>Retorna a mensagem de acordo com o status da consulta.</i>
[mensagem]	<i>Retorna a mensagem de acordo com o status (Mensagem de OK ou o erro apresentado pelo webservice).</i>
[registros]	[registrosArray] <i>Array com o resultado da busca realizada.</i>

## 1.4 Exemplo de utilização (Utilizando PHP + NuSOAP)

```
//Adiciona o nuSOAP
require_once('../nusoap/nusoap.php');

//Constantes de conexão
define('WL_CAMINHO_SOAP','http://demo.workcrm.com.br/wconnect/WConnect_WSDL.php?wsdl');

//Definição da localização do arquivo WSDL
$client = new nusoap_client(WL_CAMINHO_SOAP, true);

//verifica se ocorreu erro na criação do objeto
$error = $client->getError();

//Caso ocorra erro na conexão ao WSDL
if ($error)
{
    echo "Erro no construtor <pre>".$error."</pre>";
}

$client->setDefaultRpcParams(true);

//Define o tempo de resposta
$client->response_timeout = 25;

//Parâmetros a enviar
$params = array('ContasParametros' => array( 'ContasId' => '',
                                             'ContasIdERP' => '',
                                             'ContasCNPJ' => '',
                                             'ContasCPF' => '',
                                             'ContasEmail' => ''));

//Chama o webservice
$result = $client->call('getContas', $params);

//Verifica se ocorreu falha na chamada do metodo
if ($client->fault)
{
    echo "Falha<pre>".print_r($result)."</pre>";
}
else
{
    //verifica se ocorreu erro
    $error = $client->getError();

    if ($error)
    {
        echo "Erro: <pre>".$error."</pre>";
    }

    else
    {
        //Exibe o resultado da requisicao
        echo '<pre>';
        print_r($result);

        echo $result['registros']['registrosArray'];
    }
}
```

```
        echo '</pre>';  
    }  
}
```

Exemplo de retorno: