EF – Especificação Funcional

Versão 1.0

11 de novembro de 2021

[2021-153] – FEAC SAP B1 Implementação

# Histórico de Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autor | Notas de Revisão |
| 11/11/2021 | 1.0 | Michele Jesus | Criação Especificação Funcional |

# Lista de Participantes

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Papel | Empresa | E-mail |
| Michele Jesus | Consultora SAP | Six Consult | michele.jesus@sixconsult.com.br |
| Rogério Kloss | Quality SAP | Six Consult | rogerio@sixconsult.com.br |

# Aprovação

|  |  |  |  |
| --- | --- | --- | --- |
| Nome | Papel | Data | Assinatura |
| Paula Dantas Ribeiro | Gerente do Projeto |  |  |

**SUMÁRIO**

[*Histórico de Revisão 2*](#_Toc86845025)

[*Lista de Participantes 2*](#_Toc86845026)

[*Aprovação 2*](#_Toc86845027)

[*Introdução 4*](#_Toc86845028)

[*Objetivo do Documento 4*](#_Toc86845029)

[*Descrição do Processo de Negócio 4*](#_Toc86845030)

[*Descrição Funcional 4*](#_Toc86845031)

[*Propósito do negócio 4*](#_Toc86845032)

[*Tipo de solução 4*](#_Toc86845033)

[*Lista de Requisitos 4*](#_Toc86845034)

[*Detalhamento Funcional 4*](#_Toc86845035)

[*1. Consultas geral de movimentos 5*](#_Toc86845036)

[*Mapeamento de Dados Interno 5*](#_Toc86845037)

[*Mapeamento de Dados Externo 5*](#_Toc86845038)

[*Considerações Técnicas 5*](#_Toc86845039)

[*Tabelas de controle 5*](#_Toc86845040)

[*Layouts de Telas e/ou Relatórios 6*](#_Toc86845041)

[*Autorizações de Acesso Específico 8*](#_Toc86845042)

[*Condições de Teste 8*](#_Toc86845043)

# Introdução

A Especificação Funcional é um documento resultante do esforço colaborativo entre a área operacional e a área de desenvolvimento, visando esclarecer e direcionar o desenvolvimento da solução que adequará o sistema à necessidade operacional da organização.

# Objetivo do Documento

O objetivo do documento é descrever a necessidade com clareza para fácil entendimento das partes envolvidas, garantindo que a solução a ser entregue atenda às necessidades do negócio. Adicionalmente, a Especificação Funcional será utilizada para:

* Determinar o design da solução que será entregue;
* Conferir se o requerimento foi atendido corretamente;
* Medir a qualidade das entregas e validar se os resultados esperados foram alcançados.

# Descrição do Processo de Negócio

Atualmente o cliente tem esta integração nativa no ERP , pois usa tanto o RH como a Folha da RM Sistemas. Com a entrada em produção do SAP, será necessário integrar estas informações do SAP para o RM.

Esta rotina irá deverá ser gerada periodicamente(mensal) por um usuário, que irá selecionar o período da integração

# Descrição Funcional

## Propósito do negócio

O desenvolvimento dessa consulta no SAP Business One visa demonstrar os saldos das contas contábeis. A consulta permitirá a análise do balancete pelo saldo acumulado ou por filtros de centros de custo, permitindo assim o gerenciamento e análises por de diversos cenários.

## Tipo de solução

|  |  |  |
| --- | --- | --- |
| Tipo | | Descrição |
| **X** | **Add-on** | Aplicação executada dentro do SAP Business One. |
|  | **Externa** | Aplicação executada fora do SAP Business One (portal web, aplicação mobile etc.). |
| X | **Integração** | Aplicação de integração com sistemas de terceiros. |
|  | **Relatório / Layout** | Layout de impressão baseado em consulta para extração de informações. |
| **x** | **Consulta** | Consulta de dados (linhas x colunas) que pode ser exportado para Excel. |
|  | **Outro** | Solicitação que não se aplica a nenhum dos itens acima. |

## Lista de Requisitos

Abaixo estão listados os requisitos necessários para gerar o controle orçamentário:

|  |  |  |
| --- | --- | --- |
| ID | Ação/Funcionalidade | Descrição |
| 01 | Cadastros de PN | Criação , preenchimentos de informações, e bloqueio de campos (transactions) |
| 02 | Lançamentos em documentos de marketing e LCM | Nota de Entrada e Saída /LCM |
| 03 | Criação de Procedure/Viewer Dados – Viewer |  |
| 04 | Liberações portas de acesso/serviços do RM | Devem ser feito um teste via postman de conexão ao serviço |

# Detalhamento Funcional

1. Cadastros de PN:

Serã necessários as criações de alguns campos de usuários para indicarmos os dados necessários para integração referente a Automonomos externos.

1. Documentos de Marketing:
2. Serã necessários as criações de alguns campos de usuários para indicarmos os dados necessários para integração referente a Automonomos externos.
3. Consultas:

Irão ler todos os documentos do periodo selecionados de PN aonde a categoria Autonomos externos.

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

1. Documentação Integração:

Conforme abaixo no Mapeamento de Dados externos.

# Mapeamento de Dados Interno

Pontos de atenção para o desenvolvimento:

* Novos campos cadastros de PN
* Travas de preenchimentos obrigatórios
* Criação de Modelos de Documentos

**Mapeamentos de Campos**:

Abaixo detalhamos os campos a serem criados no SAP .

Avaliar a necessidade de ser criada uma tabela para controlarmos os dados centralizados para serem enviado pelo serviço de integração.

Esta tabela será criada pelo evento da transaction notification, cujo a rotina de integração deve ler os dados nesta tabela e controlar status de envio/erros de cada registros.



# Mapeamento de Dados Externo

Abaixo temos a documentação enviada pelo fornecedor para integração do serviço e métodos a serem consumidos;

Project: RM

Module: Folha de Pagamento

Description: Autônomos Externos

Business Objects (DataServerName): FopPagAutonomosExtData

PrimaryKey: CODCOLIGADA;CODIGO

Tabela

Descrição gerada automaticamente

Serviço responsável por efetuar a leitura do DataServer e retornar XML contendo registros de acordo com o contexto e o filtro informado.

Equivale a visão dos cadastros RM. **Create**

[view plaincopy to clipboardprint?](https://api.totvs.com.br/legado/devrm/bo_rm/FopPagAutonomosExtData.html?Objeto=FopPagAutonomosExtData)

1. /\* Programa responsável por integrar com os Webservices do RM 2. Para executar o programa, seguir os passos abaixo:

1. 1) Entrar no Visual Studio e criar um "Service References" apontando para
2. http://localhost:8051/wsDataServer/MEX?wsdl
3. Dúvidas? Consulte https://msdn.microsoft.com/pt-br/library/bb628652.aspx

6.

1. 1.1) Nomear o serviço como DataServer
2. 1.2) Substituir localhost pelo nome ou IP do servidor que executa o serviço do RM. Host
3. Em caso de dúvidas sobre como habilitar o serviço de Webservices do RM, consultar o link abaixo:
4. http://tdn.totvs.com/pages/releaseview.action?pageId=89620766

11.

1. 2) Adicionar os arquivos abaixo ao seu projeto:
2. http://tdn.totvs.com/download/attachments/211064343/Utils.cs
3. http://tdn.totvs.com/download/attachments/211064343/DataClient.cs
4. 2.1) Ajustar os NameSpace da classe DataClient e Utils
5. 3) Adicionar referencia a System.Web
6. Dúvidas? Consulte: https://msdn.microsoft.com/pt-br/library/7314433t(v=vs.90).aspx
7. \*///.
8. **public** **void** ReadView()
9. {
10. // ajuste o nome o servidor e porta. Em caso de dúvidas, consulte o link abaixo:
11. // http://tdn.totvs.com/pages/releaseview.action?pageId=89620766 23. **string** url = "http://localhost:8051";

24.

1. //importante passar no contexto o mesmo código de usuário usado para logar no webservice
2. **string** contexto = "CODSISTEMA=G;CODCOLIGADA=1;CODUSUARIO=mestre";

27.

1. //usuário e senha do aplicativo RM. O mesmo utilizado para logar no sistema e que tenha permissão de
2. //acesso ao cadastro que deseja utilizar
3. **string** usuario = "mestre"; 31. **string** senha = "integracao";

32.

1. //o filtro pode ser qualquer campo da visão, por exemplo CODCOLIGADA=1 AND CODFILIAL = 1
2. **string** filtro = "1=1";

35.

36. **string** recordData;

37.

1. // Retorna as credenciais para acesso ao WS
2. DataClient dataclient = **new** DataClient(url, contexto, usuario, senha);
3. // lê os dados da visão respeitando o filtro passado
4. DataSet ds = dataclient.ReadView("FopPagAutonomosExtData", filtro, **out** recordData);
5. // Pode utilizar o ds tipado para DataSet ou a variável recordData que possui o XML da s olicitação
6. MessageBox.Show(recordData);

44.

45. }

### ReadRecord

Serviço responsável por efetuar a leitura do DataServer e retornar XML contendo o registro correspondente a chave primária e o contexto informado.

Equivale a edição do cadastro do RM.

[view plaincopy to clipboardprint?](https://api.totvs.com.br/legado/devrm/bo_rm/FopPagAutonomosExtData.html?Objeto=FopPagAutonomosExtData)

1. /\* Programa responsável por integrar com os Webservices do RM 2. Para executar o programa, seguir os passos abaixo:

1. 1) Entrar no Visual Studio e criar um "Service References" apontando para
2. http://localhost:8051/wsDataServer/MEX?wsdl
3. Dúvidas? Consulte https://msdn.microsoft.com/pt-br/library/bb628652.aspx

6.

1. 1.1) Nomear o serviço como DataServer
2. 1.2) Substituir localhost pelo nome ou IP do servidor que executa o serviço do RM. Host
3. Em caso de dúvidas sobre como habilitar o serviço de Webservices do RM, consultar o link abaixo:
4. http://tdn.totvs.com/pages/releaseview.action?pageId=89620766

11.

1. 2) Adicionar os arquivos abaixo ao seu projeto:
2. http://tdn.totvs.com/download/attachments/211064343/Utils.cs
3. http://tdn.totvs.com/download/attachments/211064343/DataClient.cs
4. 2.1) Ajustar os NameSpace da classe DataClient e Utils
5. 3) Adicionar referencia a System.Web
6. Dúvidas? Consulte: https://msdn.microsoft.com/pt-br/library/7314433t(v=vs.90).aspx
7. \*/**public** **void** ReadRecord()
8. {
9. // ajuste o nome o servidor e porta. Em caso de dúvidas, consulte o link abaixo:
10. // http://tdn.totvs.com/pages/releaseview.action?pageId=89620766 22. **string** url = "http://localhost:8051";

23.

1. //importante passar no contexto o mesmo código de usuário usado para logar no webser vice
2. **string** contexto = "CODSISTEMA=G;CODCOLIGADA=1;CODUSUARIO=mestre";

26.

1. //usuário e senha do aplicativo RM. O mesmo utilizado para logar no sistema e que te nha permissão de
2. //acesso ao cadastro que deseja utilizar
3. **string** usuario = "mestre"; 30. **string** senha = "integracao";

31.

1. **string** recordData;
2. // Retorna as credenciais para acesso ao WS
3. DataClient dataclient = **new** DataClient(url, contexto, usuario, senha);
4. //O ReadRecord retorna o registro da edição do cadastro RM respeitando a chave primá ria
5. DataSet ds = dataclient.ReadRecord("FopPagAutonomosExtData", "0;0", **out** recordData);

1. // Pode utilizar o ds tipado para DataSet ou a variável recordData que possui o XML da solicitação
2. MessageBox.Show(recordData);
3. }

### SaveRecord

Serviço responsável por efetuar alterações de registros do DataServer de acordo com o XML informado.

[view plaincopy to clipboardprint?](https://api.totvs.com.br/legado/devrm/bo_rm/FopPagAutonomosExtData.html?Objeto=FopPagAutonomosExtData)

1. /\* Programa responsável por integrar com os Webservices do RM 2. Para executar o programa, seguir os passos abaixo:

1. 1) Entrar no Visual Studio e criar um "Service References" apontando para
2. http://localhost:8051/wsDataServer/MEX?wsdl
3. Dúvidas? Consulte https://msdn.microsoft.com/pt-br/library/bb628652.aspx

6.

1. 1.1) Nomear o serviço como DataServer
2. 1.2) Substituir localhost pelo nome ou IP do servidor que executa o serviço do RM. Host
3. Em caso de dúvidas sobre como habilitar o serviço de Webservices do RM, consultar o link abaixo:
4. http://tdn.totvs.com/pages/releaseview.action?pageId=89620766

11.

1. 2) Adicionar os arquivos abaixo ao seu projeto:
2. http://tdn.totvs.com/download/attachments/211064343/Utils.cs
3. http://tdn.totvs.com/download/attachments/211064343/DataClient.cs
4. 2.1) Ajustar os NameSpace da classe DataClient e Utils
5. 3) Adicionar referencia a System.Web
6. Dúvidas? Consulte: https://msdn.microsoft.com/pt-br/library/7314433t(v=vs.90).aspx
7. \*/**public** **void** SaveRecord()
8. {
9. //pode ser o mesmo XML retornado pelo método ReadRecord, alterando os valores que dese ja aplicar no BD
10. **string** xml = @"<FopPagAutonomosExt xmlns="http://tempuri.org/FopPagAutonomosExt.xsd">
11. <PPAGTOAUTONOMOEXT>
12. <CODCOLIGADA>0</CODCOLIGADA>
13. <CODFILIAL>0</CODFILIAL>
14. <CODIGO>0</CODIGO>
15. <CNPJTOMADOR>0</CNPJTOMADOR>
16. <CEITOMADOR>0</CEITOMADOR>
17. <TPTOMADOR>0</TPTOMADOR>
18. <NOMEAUTONOMO>0</NOMEAUTONOMO>
19. <DTNASCIMENTO>2018-08-23T08:03:35.798169</DTNASCIMENTO>
20. <CPF>0</CPF>
21. <NROINSCRICAO>0</NROINSCRICAO>
22. <CBOAUTONOMO>0</CBOAUTONOMO>
23. <CATAUTONOMO>0</CATAUTONOMO>
24. <CODOCORSEFIP>0</CODOCORSEFIP>
25. <DATAEMISSAO>2018-08-23T08:03:35.798169</DATAEMISSAO>
26. <DATAVENCIMENTO>2018-08-23T08:03:35.798169</DATAVENCIMENTO>
27. <DATAINIATIVIDADE>2018-08-23T08:03:35.798169</DATAINIATIVIDADE>
28. <DATAPAGAMENTO>2018-08-23T08:03:35.798169</DATAPAGAMENTO>
29. <NATUREZAATIV>0</NATUREZAATIV>
30. <QTDDEPSF>0</QTDDEPSF>
31. <QTDDEPIRRF>0</QTDDEPIRRF>
32. <VALORBRUTO>0</VALORBRUTO>
33. <VALORINSS>0</VALORINSS>
34. <VALORIRRF>0</VALORIRRF>
35. <RECCREATEDBY>0</RECCREATEDBY>
36. <RECCREATEDON>2018-08-23T08:03:35.798169</RECCREATEDON>
37. <RECMODIFIEDBY>0</RECMODIFIEDBY>
38. <RECMODIFIEDON>2018-08-23T08:03:35.798169</RECMODIFIEDON>
39. </PPAGTOAUTONOMOEXT> 51. </FopPagAutonomosExt>";

52.

1. // ajuste o nome o servidor e porta. Em caso de dúvidas, consulte o link abaixo:
2. // http://tdn.totvs.com/pages/releaseview.action?pageId=89620766 55. **string** url = "http://localhost:8051";

56.

1. //importante passar no contexto o mesmo código de usuário usado para logar no webservi ce
2. **string** contexto = "CODSISTEMA=G;CODCOLIGADA=1;CODUSUARIO=mestre";

59.

1. //usuário e senha do aplicativo RM. O mesmo utilizado para logar no sistema e que
2. //tenha permissão de acesso ao cadastro que deseja utilizar
3. **string** usuario = "mestre"; 63. **string** senha = "integracao";

64.

1. // Retorna as credenciais para acesso ao WS
2. DataClient dataclient = **new** DataClient(url, contexto, usuario, senha);
3. //Salva o registro na base de dados. Executará "Update" caso encontre outro registre que respeite
4. //a chave primária do dataserver, do contrário, executará "Insert".
5. **string**[] retorno = dataclient.SaveRecord("FopPagAutonomosExtData", xml);

70.

71.

MessageBox.Show(retorno.ToString());

72.

}

# Considerações Técnicas

## Tabelas de controle

* OCRD
* OPCH/PCH1 – e auxiliares
* SX\_ PPAGTOAUTONOMOEXT

## Layouts de Telas e/ou Relatórios

Exemplo de apresentação no SAP Business One:

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

Obs: deve ser gerado um log e uma consulta para conerencia dos dados antes da integração.

Criar campo lógico de simulação/integração para controla se será chamado o serviço de integração ou apenas consulta.

Outras consideraçõesNão se aplica.

## Autorizações de Acesso Específico

Não existe restrição quanto as autorizações de acesso pelo SAP.

# Condições de Teste

Para realização dos testes da geração de consultas, é necessário que:

* Cadastros de Alguns PN´s
* Documentos lançados com estes PN e com retenções
* Base Testes