



DEPARTAMENTO DE DESENVOLVIMENTO DE SOFTWARE
PROPOSTA TÉCNICA DE SOFTWARE

EXTRA DELIVERY

DESCRIÇÃO DOS WEBSERVICES DE INTEGRAÇÃO

COLETOR

1. CONTROLE DE VERSÃO

PROPOSTA TÉCNICA DE SOFTWARE
VERSÃO - 1.0.0.0

AUTOR	DESCRIÇÃO	OBSERVAÇÃO
Ricardo Ferreira Alves	Criação da Proposta Técnica	

2

INTRODUÇÃO

ESTE TÓPICO DESCREVE O OBJETIVO DESTE DOCUMENTO E SUA IMPORTANCIA NO PROCESSO INICIAL DA AQUISIÇÃO DE UMA SOLUÇÃO DE SOFTWARE DA SEAL SISTEMAS.

2. INTRODUÇÃO

Visando sempre a transparência nos serviços prestados, a Seal Sistemas através deste documento, tem o objetivo de levantar os requisitos funcionais e não funcionais, os pré-requisitos, relativos à parte sistêmica do projeto de **AUTOMAÇÃO DO PROCESSO DE SEPARAÇÃO DE PRODUTOS NAS LOJAS** para o cliente **GPA**.

É de extrema importância e também explicitado neste documento as necessidades, os problemas atuais e os objetivos a serem atingidos para satisfazer o cliente **GPA** visando à melhoria dos seus processos que irá com certeza refletir no melhor atendimento a seus clientes tornando-o mais competitivo no mercado atual.

Este documento deve ter aprovação pelo cliente, visando à segurança que o escopo de software ofertado atende em 100% as suas necessidades, caso contrário, o documento deve ser revistos e adicionado no controle de versão às revisões sugeridas bem como a atualização do documento.

No caso de surgimento de novas revisões neste documento (novos requisitos, alterações de escopo, pré-condições, pós-condições, funcionalidades), podem ser aplicadas revisões na proposta comercial, alterando-se os valores e/ou em tempo de desenvolvimento enviados em versões anteriores deste documento.

3

ESCOPO DO PRODUTO

ESTE TÓPICO DESCREVE O OBJETIVO DA SOLUÇÃO, AS NECESSIDADES DO CLIENTE, OS REQUISITOS MÍNIMOS ACEITOS PELO CLIENTE, OS SOFTWARES AUXILIARES E OS RESPONSÁVEIS INICIAIS PELO PROJETO.

3. ESCOPO DO PRODUTO

PRODUTO OFERTADO	
NOME DO PRODUTO	MIDDLEWARE KAIROS
VERSÃO DO PRODUTO	VERSÃO 3.0.0.0
TECNOLOGIAS QUE SERÃO UTILIZADOS NO PROJETO	COLETORES DE DADOS
MISSÃO DO PRODUTO	SOFTWARE MIDDLEWARE DE AUTOMATIZAÇÃO DE PROCESSOS DA CADEIA DE SUPRIMENTO

PROCESSOS ENVOLVIDOS NO PROJETO PROPOSTO	
ID	PROCESSO
1	SEPARAÇÃO DE MERCADORIAS NAS LOJAS

SOFTWARE AUXILIARES			
SOFTWARE	FABRICANTE	RESPONSABILIDADE	FUNCIONALIDADE

RESTRIÇÕES AO PROJETO

ID	TIPO	RESTRIÇÃO
1	AMBIENTE	DEVERÁ SER REALIZADO INTEGRAÇÃO COM O SISTEMA DE RETAGUARDA DO GPA VIA WEBSERVICES

4

REQUISITOS NÃO-FUNCIONAIS

UM **REQUISITO NÃO-FUNCIONAL**, DESCREVEM APENAS ATRIBUTOS DO SISTEMA OU ATRIBUTOS RELACIONADOS AO AMBIENTE, OS REQUISITOS NÃO-FUNCIONAIS SÃO OS QUE DESCREVEM: FUNCIONALIDADE, USABILIDADE, CONFIABILIDADE, DESEMPENHO, SUPORTABILIDADE, DESIGN, IMPLEMENTAÇÃO, INTERFACE E REQUISITO FÍSICO

4. REQUISITOS NÃO-FUNCIONAIS

DEFINIÇÃO DOS REQUISITOS NÃO-FUNCIONAIS

RNF	REQUISITO	CATEGORIA	PRIORIDADE	ESTABILIDADE	ESTADO
1	FÁCIL UTILIZAÇÃO	USABILIDADE	ESSENCIAL	ALTA	COMPLETO
2	RAPIDA ATUALIZAÇÃO NO SERVIDOR	DESEMPENHO	ESSENCIAL	ALTA	COMPLETO
3	INTEGRIDADE NAS INFORMAÇÕES	CONFIABILIDADE	ESSENCIAL	ALTA	COMPLETO
4	INTERFACE CLARA DE FÁCIL ENTENDIMENTO	USABILIDADE	ESSENCIAL	ALTA	COMPLETO

IDS – RNF X

CATEGORIAS – QUALIDADE | TÉCNICO | SEGURANÇA | DESEMPENHO | USABILIDADE | CONFIABILIDADE | FÍSICO

PRIORIDADES – ESSENCIAL | DESEJÁVEL | OPCIONAL

ESTABILIDADE – ALTA | MÉDIA | BAIXA

ESTADO/ENTENDIMENTO – COMPLETO | INCOMPLETO

5

REQUISITOS FUNCIONAIS

UM **REQUISITO FUNCIONAL** É DEFINIDO COMO UMA CONDIÇÃO OU UMA CAPACIDADE COM A QUAL O SISTEMA PROPOSTO DEVE ESTAR DE ACORDO. OS REQUISITOS FUNCIONAIS ESPECIFICAM AÇÕES QUE O SISTEMA DEVE SER CAPAZ DE EXECUTAR, SEM LEVAR EM CONSIDERAÇÃO AS RESTRIÇÕES FÍSICAS, DESTA FORMA OS REQUISITOS FUNCIONAIS ESPECIFICAM PORTANTO, O COMPORTAMENTO DE ENTRADA E SAÍDA DE UM SISTEMA.

5. REQUISITOS FUNCIONAIS

5.1. SEPARAÇÃO DE PRODUTOS NAS LOJAS

Para atendimento do processo de separação nas lojas no projeto Extra Delivery, se faz necessário o GPA disponibilize os webservices para a integração com os coletores de dados, pois o projeto foi concebido sem a utilização da interface de gestão Kairos, somente o middleware que será a ponte entre o sistema Legado e os coletores de dados.

Desta forma, a Seal Sistemas descreve neste documento alguns webservices que o GPA deverá fornecer e customizar para o projeto. Os métodos a seguir são sugestões para dar direção ao desenvolvimento do lado da TI GPA.

Principais WebServices

```
/// <summary>
/// Método: Login
/// Deverá validar as credenciais do usuario quando logado no sistema
/// </summary>
/// <param name="usuario">login do usuário</param>
/// <param name="senha">senha do usuário</param>
/// <returns>Se existir o usuário deverá retornar estrutura Usuario caso
contrário NULL</returns>
Usuario Login(String usuario, String senha);
```

```
/// <summary>
/// Método: ProximoPedido
/// Deverá retornar a lista de produtos, respectivas localizações e
quantidades a serem separadas de acordo com
/// o perfil o separador
/// </summary>
/// <param name="usuario">Estrutura Usuario obtida no método
Login</param>
/// <returns>Retorno a Estrutura Pedido conforme descrição do
método</returns>
Pedido ProximoPedido(Usuario usuario);
```

```

/// <summary>
/// Método: AtualizaItemSeparado
/// Método responsável em atualizar a situação do item no sistema Legado
/// </summary>
/// <param name="usuario">Estrutura Usuario obtida no método
Login</param>
/// <param name="idPedido">Id do Pedido para que o legado saiba qual
pedido será atualizado
/// esta informação deve vir na estrutura Pedido do método
ProximoPedido</param>
/// <param name="idCodigoProduto">Id do Produto que será atualizado no
legado, esta informação
/// deve vir na estrutura Pedido do método ProximoPedido</param>
/// <param name="quantidadeSeparada">Quantidade informado pelo
separador</param>
/// <param name="mensagem">Mensagem de retorno do legado para o coletor
quando houver necessidade</param>
/// <returns>0 = item atualizado com sucesso
/// menor que 0 = erro que será mostrado na tela do coletor, o
erro deve preencher o campo mensagem do método</returns>
int AtualizaItemSeparado(Usuario usuario, int idPedido, int
idCodigoProduto, decimal quantidadeSeparada, out string mensagem);

```

```

/// <summary>
/// Método: SolicitarVerificacaoItemSimilar
/// Método que informa ao legado para atendente ligar para o Cliente e
verificar se o produto solicitado não pode ser alterado por
/// um similar
/// </summary>
/// <param name="usuario">Usuario que realizou a solicitação</param>
/// <param name="idPedido">Identificação do pedido</param>
/// <param name="idCodigoProduto">Identificação do produto a ser
alterado</param>
/// <param name="quantidadeSeparada">Quantidade separada</param>
/// <param name="mensagem">Mensagem de retorno do legado para coletor
quando houver necessidade</param>
/// <returns>0 = Solicitado com sucesso
/// menor que 0 = erro na solicitação, erro deve preencher o campo
mensagem do método</returns>
int SolicitarVerificacaoItemSimilar(Usuario usuario, int idPedido, int
idCodigoProduto, decimal quantidadeSeparada, out string mensagem);

```

```

/// <summary>
/// Método: VerificaSituacaoItemSimilar
/// Método que verifica se a análise de um determinado produto já foi
realizada com relação a item Similar
/// </summary>
/// <param name="usuario">Usuário que solicita a verificaçã/o</param>
/// <param name="idPedido">Identificação do pedido</param>
/// <param name="idCodigoProduto">Identificação do Código do
Produto</param>
/// <param name="status">0 = Não Analisado
/// -1 = Cancelar produto
/// -2 = Substituir por similar</param>
/// <returns>Se status for -2 o método deverá retornar o Produto aprovado
pelo cliente para substituição</returns>
Produto VerificarSituacaoItemSimilar(Usuario usuario, int idPedido, int
idCodigoProduto, out int status);

```

```

/// <summary>
/// Método: CancelaItem
/// Método responsável em cancelar um item do pedido e notificar o
legado.
/// </summary>
/// <param name="usuario">Usuario que realizou o cancelamento</param>
/// <param name="idPedido">Identificação do pedido</param>
/// <param name="idCodigoProduto">Identificação do produto</param>
/// <param name="mensagem">Mensagem de retorno do legado para coletor
quando houver necessidade</param>
/// <returns>0 = cancelamento realizad com sucesso
/// menos que 0, erro no cancelamento, sera retornado o valor que estiver
no campo mensagem</returns>
int CancelaItem(Usuario usuario, int idPedido, int idCodigoProduto, out
String mensagem);

```

```

/// <summary>
/// Método: Fechar Pedido
/// Método responsável em fechar o pedido em andamento
/// </summary>
/// <param name="usuario">Usuario que está finalizando o pedido</param>
/// <param name="idPedido">Identificação do pedido</param>
/// <param name="mensagem">Mensagem de retorno do legado para o coletor
quando houver necessidade</param>
/// <returns>0 = pedido fechado com sucesso
/// menos que 0 = erro no fechamento, sera retornado o valor que estiver
no campo mensagem</returns>
int FecharPedido(Usuario usuario, int idPedido, String mensagem);

```

6

INTEGRAÇÕES E ARQUITETURA

ESTE TÓPICO DESCREVE AS INTEGRAÇÕES PREVISTAS ENTRE O SISTEMA SEAL E O SISTEMA LEGADO (SE HOUVER) E A ARQUITETURA ENTRE OS PRINCIPAIS COMPONENTES DA SOLUÇÃO OFERTADA, DESCREVENDO TAMBÉM OS RESPONSÁVEIS DA AQUISIÇÃO OU FORNECIMENTO DOS SOFTWARES OU HARDWARES ENVOLVIDOS.

6. INTEGRAÇÕES E ARQUITETURA

6.1. INTEGRAÇÕES

INTEGRAÇÃO			RECOMENDAÇÕES/OBS.
PRINCIPAL MEIO DE INTEGRAÇÃO	WEBSERVICES		
ERP/WMS UTILIZADO	GPA-LEGADO		
VERSÃO			
OUTROS MEIOS DE INTEGRAÇÃO PREVISTAS	NÃO		
CLIENTE POSSUI EQUIPE DE DESENVOLVIMENTO INTERNO	SIM		

7

ACEITE DO DOCUMENTO

TERMO DE ACEITE DAS CONDIÇÕES DA PROPOSTA TÉCNICA APRESENTADA

7. ACEITE DO DOCUMENTO

Firmamos através deste documento, um “de acordo” com os objetivos, escopo, requisitos funcionais, requisitos não-funcionais, integração e arquitetura a serem implementadas no projeto de **SEPARAÇÃO DE PRODUTOS NAS LOJAS** do cliente **GPA**.

RESPONSÁVEL
GPA

RESPONSÁVEL
SEAL SISTEMAS E TECNOLOGIA DA INFORMAÇÃO