<Nome do Projeto>

Especificação dos Requisitos de Software

Para <Subsistema ou Recurso>

Versão <1.0>

[Observação: O template a seguir é fornecido para uso com o Rational Unified Process. O texto em azul exibido entre colchetes e em itálico (style=InfoBlue) foi incluído para orientar o autor e deve ser excluído antes da publicação do documento. Qualquer parágrafo inserido após esse estilo será definido automaticamente como normal (style=Body Text).]

[Para personalizar campos automáticos no Microsoft Word (que exibem um plano de fundo cinza quando selecionados), selecione File>Properties e substitua o conteúdo dos campos Title, Subject e Company pelas informações adequadas a esse documento. Depois de fechar a caixa de diálogo, para atualizar os campos automáticos no documento inteiro, selecione Edit>Select All (ou Ctrl-A) e pressione F9 ou simplesmente clique no campo e pressione F9. Isso deve ser feito separadamente para Cabeçalhos e Rodapés. Alt-F9 alterna entre a exibição de nomes de campos e a do conteúdo de campos. Consulte a ajuda do Word para obter mais informações sobre como trabalhar com campos.]

Histórico da Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| <dd/mmm/aa> | <x.x> | <detalhes> | <nome> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Índice Analítico

1. Introdução 4

1.1 Finalidade 4

1.2 Escopo 4

1.3 Definições, Acrônimos e Abreviações 4

1.4 Referências 4

1.5 Visão Geral 4

2. Descrição Geral 5

3. Requisitos Específicos 5

3.1 Funcionalidade 5

3.1.1 RF001 - Armazenar clientes 6

3.1.2 RF002 - Agrupar clientes 6

3.1.3 RF003 – Acessar os pedidos 6

3.1.4 RF004 – Preparar entrega 6

3.1.5 RF004 – Traçar itinerário e rota de entrega 6

3.1.6 RF005 – Designar saída 6

3.1.7 RF006 – Seguir itinerário 6

3.2 Usabilidade 6

3.2.1 <Requisito de Usabilidade Um> 7

3.3 Confiabilidade 7

3.3.1 <Requisito de Confiabilidade Um> 7

3.4 Desempenho 7

3.4.1 <Requisito de Desempenho Um> 7

3.5 Suportabilidade 8

3.5.1 <Requisito de Suportabilidade Um> 8

3.6 Restrições de Design 8

3.6.1 <Restrição de Design Um> 8

3.7 Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line 8

3.8 Componentes Adquiridos 8

3.9 Interfaces 8

3.9.1 Interfaces do Usuário 8

3.9.2 Interfaces de Hardware 8

3.9.3 Interfaces de Software 8

3.9.4 Interfaces de Comunicação 9

3.10 Requisitos de Licenciamento 9

3.11 Observações Legais, de Copyright e Outras 9

3.12 Padrões Aplicáveis 9

4. Informações de Suporte 9

Especificação dos Requisitos de Software

# Introdução

[A introdução da **Especificação de Requisitos de Software (SRS)** fornece uma visão geral de toda a **SRS**. Ela contém a finalidade, o escopo, as definições, os acrônimos, as abreviações, as referências e a visão geral da **SRS**.]

[Observação: A **SRS** captura todos os requisitos de software do sistema ou de uma parte do sistema. A seguir, há um esquema de uma **SRS** típica para um projeto que utiliza somente requisitos em estilo de linguagem natural tradicional — sem **modelagem de casos de uso**.Essa **SRS** captura todos os requisitos em um único documento, com seções aplicáveis inseridas a partir das Especificações Suplementares (que não serão mais necessárias). Para ter acesso a um template de uma **SRS** que utilize a modelagem de casos de uso, que consiste em um pacote contendo Casos de Uso do modelo de casos de uso e Especificações Suplementares aplicáveis, assim como outras informações de suporte, consulte o arquivo rup\_srsuc.dot.]

[É possível organizar a **SRS** de várias maneiras diferentes. Consulte o padrão [IEEE830-1998] para obter explicações mais detalhadas, assim como outras opções de organização de uma **SRS**.]

## Finalidade

[Especifique a finalidade desta **SRS**. A **SRS** descreve totalmente o comportamento externo do aplicativo ou do subsistema identificado. Ela também descreve requisitos não funcionais, restrições de design e outros fatores necessários para fornecer uma visão completa e abrangente dos requisitos do software.]

Este documento tem como finalidade definir os requisitos do sistema.

## Escopo

[Uma breve descrição do aplicativo de software ao qual se aplica a **SRS**, do recurso ou de outro agrupamento de subsistemas, do(s) modelo(s) de Casos de Uso associado(s) a ela e de tudo o que for afetado ou influenciado por este documento.]

O escopo deste documento é o projeto Right Rout, que visa integrar os setores do negócio e melhorar o sistema de entregas da loja.

## Definições, Acrônimos e Abreviações

[Esta subseção fornece as definições de todos os termos, acrônimos e abreviações necessárias à adequada interpretação da **SRS**. Essas informações podem ser fornecidas mediante referência ao Glossário do projeto.]

## Referências

[Esta subseção fornece uma lista completa de todos os documentos mencionados em qualquer outra parte da **SRS**. Identifique cada documento por título, número do relatório (se aplicável), data e organização de publicação. Especifique as fontes a partir das quais as referências podem ser obtidas. Essas informações podem ser fornecidas por um anexo ou outro documento.]

## Visão Geral

[Esta subseção descreve o que o restante da **SRS** contém e explica como o documento está organizado.]

O restante do documento apresenta os requisitos do sistema conforme a sua categoria.

# Descrição Geral

[Esta seção da **SRS** descreve os fatores gerais que afetam o produto e seus requisitos. Ela não define requisitos específicos. Em vez disso, ela fornece uma base para esses requisitos, que serão definidos detalhadamente na Seção 3, e facilita sua compreensão. Inclua itens como:

• perspectiva do produto

• funções do produto

• características do usuário

• restrições

• suposições e dependências

• subconjuntos de requisitos]

O sistema a ser implementado deve ser integrado com o sistema de gerenciamento de estoque Mundo Azul, do qual a loja já faz uso. Ele deve receber e armazenar os dados dos clientes, com enfoque no endereço, pois a partir dele deve ocorrer o agrupamento dos clientes de acordo com sua sub-região.

O sistema deve calcular o montante de pedidos necessário para cada saída, assim como a quantidade de saídas necessárias durante o dia, não excedendo a capacidade máxima de transporte e sem que o prazo dado ao cliente seja extrapolado.

O sistema deve conter um comando que permita notificar o entregador quando uma demanda de entrega estiver pronta e em horário de saída. Deve ser disponibilizada uma versão que integre os entregadores ao processo, de maneira que ele possa abrir um app no seu celular e acompanhar o itinerário traçado pelo sistema, contendo a rotina de entregas mais curta e com maior otimização do tempo.

# Requisitos Específicos

[Esta seção da **SRS** contém todos os requisitos de software em um nível de detalhamento suficiente para possibilitar que os designers projetem um sistema que satisfaça esses requisitos e que os testadores verifiquem se o sistema satisfaz esses requisitos. Quando for utilizada a modelagem de casos de uso, esses requisitos serão capturados nos Casos de Uso e nas especificações suplementares aplicáveis. Se a modelagem de casos de uso não for utilizada, o esquema das especificações suplementares poderá ser inserido diretamente nesta seção, conforme mostrado abaixo.]

## Funcionalidade

[Esta seção descreve os requisitos funcionais do sistema que são expressos no estilo de linguagem natural. Para muitos aplicativos, isso poderá constituir o volume do pacote **SRS**. Deve-se refletir muito para organizar esta seção. Normalmente, ela é organizada por recurso, mas métodos de organização alternativos como, por exemplo, organização por usuário ou organização por subsistema, também podem ser apropriados. Os requisitos funcionais podem incluir conjuntos de características, recursos e segurança.

Quando as ferramentas de desenvolvimento de aplicativos, como ferramentas de requisitos, ferramentas de modelagem, entre outras, forem utilizadas para capturar a funcionalidade, esta seção do documento fará referência à disponibilidade desses dados, indicando o local e o nome da ferramenta usada para capturar os dados.]

### RF001 - Armazenar clientes

[A descrição do requisito deve ser feita aqui.]

O sistema deve armazenar os dados dos clientes a partir do atendimento ou por inserção do gerente, se o cliente já estiver na base de dados o sistema deve dar a opção de atualizar dados, se for necessário. Disponibilizando também as opções de consultar e exclui clientes.

### RF002 - Agrupar clientes

O sistema deve conter e agrupar os clientes em 4 agrupamentos, conforme sua região geográfica.

### RF003 – Acessar os pedidos

O sistema deve permitir que os pedidos sejam registrados pelo atendente, todo pedido deve conter obrigatoriamente a identificação do cliente.

### RF004 – Preparar entrega

O sistema deve calcular a quantidade de saídas para entrega baseado nos seguintes fatores:

* O prazo de 24 horas para entrega do pedido deve ser cumprido sem exceção;
* A capacidade máxima de transporte é de 20 pedidos por saída;
* Os montantes só devem sair para entrega com menos de 10 pedidos se o prazo para entrega estiver vencendo, caso contrário o sistema deve aguardar até que sejam registrados novos pedidos para designar a saída;
* Quando um montante estiver em condições de saída o sistema deve notificar o responsável pela logística.

### RF004 – Traçar itinerário e rota de entrega

O sistema deve traçar o melhor itinerário baseado na localização de cada cliente e entregar este itinerário em forma de rota de entrega.

### RF005 – Designar saída

O sistema deve permitir que o setor de logística contate o entregador de forma integrada;

### RF006 – Seguir itinerário

O sistema deve permitir que o entregador siga o itinerário durante o percurso, apontando na tela as entregas que já foram efetuadas.

## Usabilidade

[Esta seção contém todos os requisitos que afetam a usabilidade. Por exemplo,

* especifique o tempo de treinamento necessário para que usuários normais e usuários com conhecimentos avançados se tornem produtivos em operações específicas
* especifique períodos de tempo mensuráveis para tarefas típicas ou baseie os requisitos de usabilidade do novo sistema em outros sistemas que os usuários conheçam e gostem
* especifique requisitos de forma que estejam em conformidade com padrões de usabilidade comuns como, por exemplo, os padrões CUA da IBM ou os padrões GUI da Microsoft]

### <Requisito de Usabilidade Um>

[A descrição do requisito deve ser feita aqui.]

## Confiabilidade

[Os requisitos de confiabilidade do sistema devem ser especificados aqui. A seguir, há algumas sugestões:

* Disponibilidade — especifique a porcentagem de tempo disponível (xx.xx%), as horas de uso, o acesso à manutenção, as operações de modo degradado, etc.
* Tempo Médio entre Falhas (MTBF) — normalmente especificado em horas, mas também poderá ser especificado em termos de dias, meses ou anos.
* Tempo Médio para Reparo (MTTR) — quanto tempo o sistema poderá ficar sem funcionar após uma falha?
* Exatidão — especifique a precisão (resolução) e a exatidão (através de algum padrão conhecido) necessárias na saída do sistema.
* Taxa Máxima de Erros ou Defeitos — geralmente expressa em termos de erros por milhares de linhas de código (erros/KLOC) ou de erros por ponto de função (erros/ponto de função).
* Taxa de Erros ou Defeitos — categorizada em termos de erros pouco importantes, importantes e críticos: o(s) requisito(s) deve(m) definir o que se entende por um erro “crítico”; por exemplo, a perda total de dados ou uma total incapacidade de usar determinadas partes da funcionalidade do sistema.]

### <Requisito de Confiabilidade Um>

[A descrição do requisito deve ser feita aqui.]

## Desempenho

[As características de desempenho do sistema devem ser descritas nesta seção. Inclua tempos de resposta específicos. Quando aplicável, faça referência, por nome, aos Casos de Uso relacionados.

* Tempo de resposta de uma transação (médio, máximo)
* Taxa de transferência como, por exemplo, transações por segundo
* Capacidade como, por exemplo, o número de clientes ou de transações que o sistema pode acomodar
* Modos de degradação (o modo aceitável de operação quando o sistema tiver sido degradado de alguma maneira)
* A utilização de recursos como, por exemplo, memória, disco, comunicações, etc.

### <Requisito de Desempenho Um>

[A descrição do requisito deve ser feita aqui.]

## Suportabilidade

[Esta seção indica todos os requisitos que irão aprimorar a suportabilidade ou a manutenibilidade do sistema que está sendo criado, incluindo padrões de codificação, convenções de nomeação, bibliotecas de classes, acesso à manutenção e utilitários de manutenção.]

### <Requisito de Suportabilidade Um>

[A descrição do requisito deve ser feita aqui.]

## Restrições de Design

[Esta seção indica todas as restrições de design referentes ao sistema que está sendo criado. As restrições de design representam decisões de design que foram impostas e devem ser obedecidas. Entre os exemplos desse tipo de restrição estão linguagens de software, requisitos de processo de software, uso prescrito de ferramentas de desenvolvimento, restrições de design e de arquitetura, componentes comprados, bibliotecas de classes, etc.]

### <Restrição de Design Um>

[A descrição do requisito deve ser feita aqui.]

## Requisitos de Sistema de Ajuda e de Documentação de Usuário On-line

[Descreve os requisitos, se houver, de documentação de usuário on-line, sistemas de ajuda, observações sobre ajuda, etc.]

## Componentes Adquiridos

[Esta seção descreve todos os documentos comprados para serem usados com o sistema, quaisquer restrições de utilização ou de licenciamento aplicáveis, e quaisquer padrões associados de compatibilidade e de interoperabilidade ou de interface.]

## Interfaces

[Esta seção define as interfaces que devem ser suportadas pelo aplicativo. Ela deve conter especificidades, protocolos, portas e endereços lógicos adequados, entre outros, para que o software possa ser desenvolvido e verificado em relação aos requisitos de interface.]

### Interfaces do Usuário

[Descreva as interfaces de usuário que deverão ser implementadas pelo software.]

### Interfaces de Hardware

[Esta seção define todas as interfaces de hardware que devem ser suportadas pelo software, incluindo a estrutura lógica, os endereços físicos, o comportamento esperado, etc.]

### Interfaces de Software

[Esta seção descreve as interfaces de software para outros componentes do sistema de software. Poderão ser componentes comprados, componentes reutilizados de outro aplicativo ou componentes que estejam sendo desenvolvidos para subsistemas fora do escopo desta **SRS**, mas com os quais esse aplicativo de software deve interagir.]

### Interfaces de Comunicação

[Descreva todas as interfaces de comunicação com outros sistemas ou dispositivos como, por exemplo, redes locais, dispositivos seriais remotos, etc.]

## Requisitos de Licenciamento

[Esta seção define todos os requisitos de imposição de licenciamento ou outros requisitos de restrição de utilização que deverão ser exibidos pelo software.]

## Observações Legais, de Copyright e Outras

[Esta seção descreve todos os avisos legais necessários, garantias, observações sobre direitos autorais, observações sobre patentes, logomarcas, marcas comerciais ou problemas de conformidade com logotipos referentes ao software.]

## Padrões Aplicáveis

[Esta seção descreve, por meio de referências, todos os padrões aplicáveis e as seções específicas desses padrões que se aplicam ao sistema que está sendo descrito. Entre esses padrões estão incluídos, por exemplo, padrões legais, de qualidade e reguladores, padrões de indústria referentes à usabilidade, interoperabilidade, internacionalização, compatibilidade com o sistema operacional, etc.]

# Informações de Suporte

[As informações de suporte facilitam o uso da **SRS**. Elas incluem:

* Índice analítico
* Índice
* Apêndices

Poderão estar incluídos roteiros de caso de uso ou protótipos da interface do usuário. Quando forem incluídos apêndices, a **SRS** deverá especificar explicitamente se os apêndices deverão ou não ser considerados parte integrante dos requisitos.]