docss

**Especificação dos Requisitos do Software**

**e Análise do Projeto**

**<RID>**

## Versão <1.0>

*Equipe:*

*<Integrante da equipe:*

*Marcos Silva*

*e OUTROS (COMPLETAR)>*

**Histórico das Revisões**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Sumário**

[Versão <1.0> 1](#_gjdgxs)

[1](#_30j0zll) Introdução 4

[*1.1*](#_1fob9te) *Objetivos deste documento 4*

[*1.2*](#_3znysh7) *Escopo do produto 4*

[1.2.1](#_2et92p0) Nome do produto e de seus componentes principais 4

[1.2.2](#_tyjcwt) Descrição do produto 4

[1.2.3 Missão do produto 4](#_3dy6vkm)

[*1.3*](#_1t3h5sf) *Definições e siglas 4*

[*1.4*](#_4d34og8) *Técnica(s) utilizada(s) para levantamento de requisitos 4*

[2](#_2s8eyo1) Descrição geral do produto 4

[*2.1*](#_17dp8vu) *Perspectiva do produto 5*

[2.1.1](#_3rdcrjn) Diagrama de Contexto 5

[*2.2*](#_26in1rg) *Usuários e sistemas relacionados 5*

[2.2.1](#_lnxbz9) Descrição 5

[3](#_35nkun2) Requisitos específicos 6

[*3.1*](#_1ksv4uv) *Identificação dos requisitos 6*

[*3.2*](#_44sinio) *Prioridades dos requisitos 6*

[*3.3*](#_2jxsxqh) *Descrição dos requisitos 6*

[*Requisitos funcionais 8*](#_z337ya)

[3.3.1 Diagrama de casos de uso 8](#_3j2qqm3)

[3.3.1.1 Casos de Uso: Usuário <Diagrama de Caso de Uso para cada ator envolvido> 8](#_1y810tw)

[3.3.2 Casos de Uso: Usuário <Descrição de todos os casos de uso para cada ator> 8](#_4i7ojhp)

[3.3.2.1 [RU USU01] – Consulta Detento em Outro Sistema 8](#_2xcytpi)

[3.3.2.1.1 Descrição 8](#_1ci93xb)

[3.3.2.1.2 Prioridade 8](#_3whwml4)

[3.3.2.1.4 Pré-condições 9](#_2bn6wsx)

[3.3.2.1.5 Pós-condição 9](#_qsh70q)

[3.3.2.1.6 Fluxos principal 9](#_3as4poj)

[3.3.2.1.6.1 Penitenciárias não encontradas 9](#_1pxezwc)

[3.3.2.1.7 Fluxo secundário 9](#_49x2ik5)

[*3.3*](#_2p2csry) *Requisitos não funcionais 9*

[3.4.1 [RNF 01] – Utilização de Sistema Anterior 9](#_147n2zr)

[3.4.2 [RNF 02] – Relatórios 9](#_3o7alnk)

[3.4.3 [RNF 03] – Tempo de Resposta 10](#_23ckvvd)

[3.4.4 [RNF 04] – Interface Amigável 10](#_ihv636)

[3.4.5 [RNF 05] – Sistema de Ajuda 10](#_32hioqz)

[3.4.6 [RNF 06] – Usuários Simultâneos 10](#_1hmsyys)

[3.4.7 [RNF 07] – Uso do Teclado 10](#_41mghml)

[3.4.8 [RNF 08] – Histórico de Eventos 10](#_2grqrue)

[3.4.9 [RNF 09] – Backup dos Dados 10](#_vx1227)

[3.4.10 [RNF 10] – Fácil Migração 11](#_3fwokq0)

[4.](#_1v1yuxt) Cronograma do Projeto 11

[Anexo A – 12](#_4f1mdlm)

## Introdução

### Objetivos deste documento

O objetivo desse documento é descrever passo a passo o funcionamento do sistema de controle e de estrutura organizacional de manuais dos equipamentos de responsabilidade do setor Transmissão, esse documento é de extrema importância para entender o funcionamento do sistema e saber as condições de cada variável do banco de dados a parte. desde um fluxograma até um diagrama de blocos para simples entendimento. Qualquer soma ou melhorias serão datas em versões do mesmo documento acionando um # seguindo do nome do reviewl da versão melhorada.

Público Alvo: Setor Transmissão

### Escopo do produto

#### Nome do produto e de seus componentes principais

rid (temporário, em discussão):

#Rev 1.0

* 1. Estrutura de Manuais e equipamentos dos devidos postos
* 2. Estrutura de Formulário e Gerenciamento de Relatórios de Preventivas
* 3. Gerenciamento desse sistema por meio do Browser
* 4. Gerenciamento de estatísticas
* 5. Gerenciamento de todas as interface e imagens de blocos, OBS: mudança feita pelo usuário final
* 6. Gerenciamento de gerenciamento de conteúdo (CMS) orientado pelo escopo inicial de Preventivas

#### Descrição do produto

O produto será designado a trabalhar na rede local da TV Gazeta na Fonte Grande onde será dado a um determinado endereço de IP fixo e o acesso garantido a todos os postos onde trafegam cerca de 2 Mbits por segundo. Serviços serão agregados tais como a suíte Apache 2 como serviços de requisição/cabeçalho sob o protocolo HTTP e HTTPS, PHP 7.2 usada originalmente apenas para o desenvolvimento desta aplicação presentes e atuantes no lado do servidor já preparado, capazes de gerar conteúdo dinâmico final e MariaDB, serviço para gerenciamento de toda a estrutura do banco de dados e formulário, que também trabalhará em paralelo com MySQL.

#### 1.2.3 Missão do produto

O produto tem como iniciativa, facilitar o acesso aos manuais deixando-os disponivel em rede, onde possa ser acessado de qualquer local que esteja na rede local onde o servidor será alocado. O propósito é facilitar e aumentar a eficiência de resposta durante as preventivas ou qualquer relatório convicto que venha a ser usado no atual serviço do Setor Alvo. Além de executar alguns serviços de apoio a alguns equipamento no aguardo do #Rev1.1.

### Definições e siglas

|  |  |  |
| --- | --- | --- |
| **Número de ordem** | **Sigla** | **Definição** |
|  | SIGEP | Sistema de gerenciamento e estatística de dados |
|  | SGBD | Sistema de Gerenciamento de Banco de Dados |
|  | CMS | Sistema de Gerenciamento de Conteúdo |
|  | MariaDB | Alternativa Open Source do MySQL |
| **5** | CGI | Interface de gateway comum |

### Técnica(s) utilizada(s) para levantamento de requisitos

Na imagem abaixo é possível verificar a estrutura da técnica para levantamento dos requisitos limitado a esta versão.

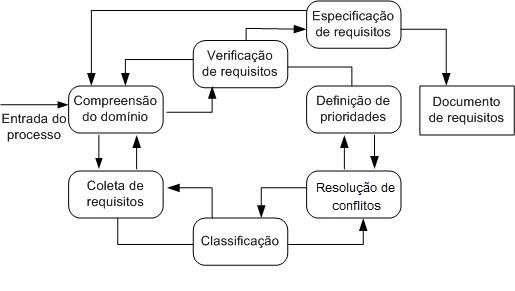
1.4.1 Compreensão do domínio: é a analisar a compreensão do real sentido do domínio da aplicação

1.4.2 Coleta de requisitos: É o processo de interagir com as partes interessadas do sistema para descobrir o real requisito.

1.4.3 Classificação: Essa atividade vai considera o conjunto não estrutura dos requisitos e os organizar em grupos coerentes

1.4.4 Resolução de conflitos: Quanto múltiplos choque de interesse no atual envolvimento do sistema os requisitos apresentaram um certo conflito. Esse passo visam solucionar os problemas de conflitos vigentes

1.4.5 Verificação de requisitos: Os requisitos são verificados para descobrir se estão completos e consistentes esse tópico visam verificar se estão de acordo com as partes interessadas neste desenvolvimento.

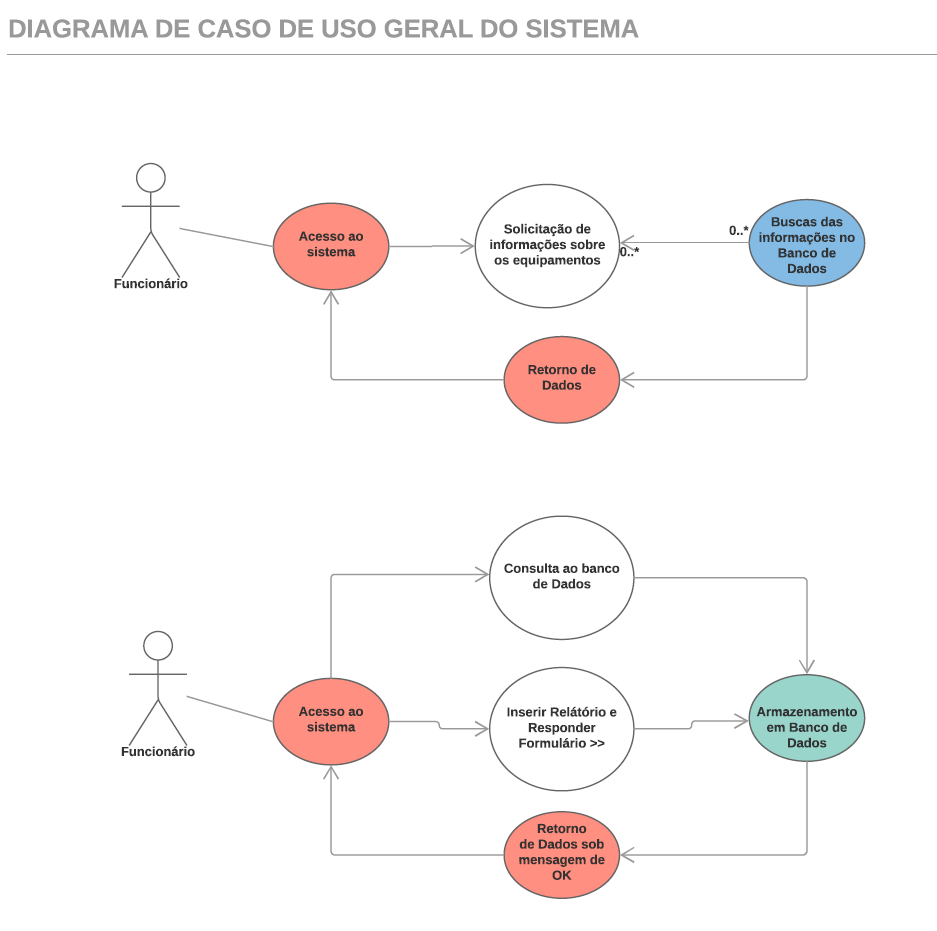


## Descrição geral do produto

### Perspectiva do produto

A descrição geral deste sistema visa descrever o contexto no qual o software será inserido. Já que o atual projeto usa um pouco de RUP, foi definido um diagrama de contexto do qual foram extraídas as informações relevantes sobre as interfaces de software, hardware e comunicação.

#### Diagrama de Contexto



Na figura acima é possível ver o diagrama de caso para algumas situações do cotidiano do usuário limitado a esse Rev1.0.

### Usuários e sistemas relacionados

#### Descrição

|  |  |  |
| --- | --- | --- |
| **Número de ordem** | **Ator** | **Definição** |
|  | Usuário/Funcionário | Qualquer pessoa autorizada a usar o sistema para acesso a funções gerenciais e não gerenciais. |

## Requisitos específicos

### Identificação dos requisitos

Por convenção e para facilitar a identificação dos casos de uso junto aos atores e contextos, a referência é feita de acordo com o esquema abaixo:

**[sigla de subseção | identificação do ator | numeração]**

Os atores são identificados pelas primeiras três letras do seu nome.

### Prioridades dos requisitos

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações: essencial, importante e desejável. Abaixo temos a descrição de significado de cada uma dessas denominações:

|  |  |
| --- | --- |
| **Essencial** | É o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente. |
| **Importante** | É o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim. |
| **Desejável** | É o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada. |

### Descrição dos requisitos

**[RF 01] – Gerenciamento de manuais**

O sistema provê meios de gerenciar os manuais. Há funções para cadastro de um determinado equipamento, remoção e edição de dados dos manuais no sistema, bem como consultas. O sistema armazena os seguintes dados para cada inserção de um novo equipamento e seu manual: Nome Técnico, Numero de Registro/ Ou Identificação, Data de entrada, Foto, e Status (Operante ou Não Operante). O status do equipamento pode ser Operante Reserva, Operante Principal, Não Operante. Cada equipamento tem seu período de preventivas que será determinado pelo usuário (Opta ou Não Opta). Essas ações podem ser: LIMITADO AO REV 1.0, em breve, devendo ser mais detalhadas ao serem inseridas junto ao relatório. A qualquer usuário ou não-usuário é permitido fazer consultas sobre equipamentos e seus manuais, mas as outras funcionalidades são restritas a usuários do tipo 'Funcionário' limitado a esse Rev1.0 É possível gerar um arquivo de relatório sobre qualquer consulta feita sobre os equipamentos.

**[RF 02] – Gerenciamento de Serviços**

O sistema provê meios de gerenciar os serviços oferecidos pelo equipamento. Há funções para cadastro, remoção e edição de dados dos serviços, bem como consultas. Os serviços podem ser realizados por funcionários ou por usuário não cadastrado limitado ao Rev 1.0. Também podem ser realizado serviços internos ou externos. A qualquer usuário é permitido fazer consultas sobre serviços, mas as outras funcionalidades são restritas a usuários do tipo 'Funcionário' limitado ao Rev1.0. É possível gerar um arquivo de relatório sobre qualquer consulta feita sobre serviços.

**[RF 03] – Gerenciamento de Estatísticas (LIMIT Rev1.0)**

O sistema provê meios de gerar estatísticas a respeito dos dados armazenado,. Pode-se calcular o espaço disponível no disco local do servidor. A qualquer usuário é permitido fazer esse tipo de consulta. É possível gerar um arquivo de relatório sobre qualquer consulta feita sobre estatísticas Limitado ao Rev1.0.

**[RF 04] – Estrutura de pasta manuais e estrutura do banco de dados**

O sistema provê meios de uma estrutura de pasta para consulta de manuais conforme descrito abaixo. A qualquer usuário é permitido fazer esse tipo de consulta. É possível gerar um arquivo de relatório sobre qualquer consulta feita sobre listagem de pasta via comando terminal direto do servidor limitado ao Rev1.0.

sistema

├─── manuais

├─── sys

├─── rootcopy\_services

├─── cms

├─── ntp

├─── geracao

├─── database ($MySQL)

│ ├─── 1-Fonte-Grande

│ ├─── 2-Sede

│ ├─── 3-Viana

│ ├─── 4-Serra

│ ├─── 5-Morro-do-Moreno

│ ├─── 6-Morro-do-Ceu

│ ├─── 7-Jaburuna

│ ├─── 8-Kautsky

│ ├─── 9-Arace

│ ├─── 10-Itanhenga

│ ├─── 11-Prefeituras

│ └─── 12-Vitoria

**[RF 05] – Sistema de Gerenciamento de Conteúdo (CMS)**

O sistema provê meios de gerar, adicionar e modificar qualquer coisa dentro do ambiente GUI do sistema através do CMS Sistema de Gerenciamento de Conteúdo, que nele se encontra todas as formas legais e facilitadas de manipulação do banco de dados, e de todas a suíte de software e desenvolvimento de novos plugins para incorporar outras linguagem de programação além delas utilizados HTML5 para interface, JavaScript para manipulamento de content que servirá como spin-off e Perl/Python para geração offline de dos serviços contratados estatísticas a respeito dos dados armazenado,. Pode-se calcular o espaço disponível no disco local do servidor. A qualquer usuário é permitido fazer esse tipo de consulta. É possível gerar um arquivo de relatório sobre qualquer consulta feita sobre estatísticas Limitado ao Rev1.0.

**[RF 06] – Estrutura de pasta do sistema**

O sistema provê meios de uma estrutura de pasta para modificação do sistema com aparato do método de ajax de instância de arquivos paralelos, conforme descrito abaixo. A abertura do sistema de pasta é feita através do protocolo SFTP ou FTP sem certificação ou acesso direto ao diretório raiz via OS (Sistema Operacional), limitado ao review 1.0, o usuário cadastrado é permitido fazer esse tipo de consulta e modificação. É possível gerar um arquivo de base para alteração do ajax do arquivo instanciado no sistema, alterando no bloco de notas ou seguindo a caixa de indexação do sistema HTML ou se preferir upload via FTP ou SFTP feita sobre listagem de pasta via comando terminal direto do servidor limitado ao Rev1.0.

OBS: COLOCAR A LISTAGEM DO DIRETÓRIO RAIZ NESSE LOCAL

**[RF 07] – Interface agradável sobre CSS Style e HTML5**

O sistema provê meios de uma estrutura de pasta para modificação do sistema com aparato do método de ajax de instância de arquivos paralelos, conforme descrito abaixo. As principais classes dentro do Style do design front-end do navegador são

**[RF 08] – Estrutura do Banco de Dados**

O sistema provê meios de uma estrutura de banco de dados para inserção de conteúdo de novos equipamentos e eventuais consultas, para modificação do sistema, com aparato do método de ajax de instância de arquivos paralelos, conforme a Tabela abaixo. A estrutura abaixo contempla o agrupamento em ASCII\_BIN e UTF8\_BIN que podemos obter uma eventual modificação na estrutura sem gerar processos demasiadamente longos do processamento do MySQL/MariaDB. Evitando-se o sobrecarregamento do servidor onde esses serviços estão de inoperabilidade, operando devidamente conforme o que foi programado

# Nome Tipo Agrupamento (Collation) Atributos Nulo Predefinido Comentários Extra Acções

1 Nome text ascii\_bin Não None Muda Muda Elimina Elimina

Mais Mais

2 Prefixo text utf8\_bin Sim None Muda Muda Elimina Elimina

Mais Mais

3 Tipo text utf8\_bin Não None Muda Muda Elimina Elimina

Mais Mais

4 Canal text utf8\_bin Não None Muda Muda Elimina Elimina

Mais Mais

5 Potência int(11) Não None Muda Muda Elimina Elimina

Mais Mais

6 Frequência int(11) Não None Muda Muda Elimina Elimina

Mais Mais

7 Posto text utf8\_bin Não None Muda Muda Elimina Elimina

O objetivo deste tópico é prover informações para formalizar a nomenclatura de objetos de banco de dados, bem como apresentar regras para sua utilização. Evitando assim, o hábito de existir diferentes nomenclaturas dentro de uma mesma aplicação do servidor. O documento apresentar os objetos de banco de dados com três itens sintaxe, regras e exemplo. Isto para facilitar o entendimento do desenvolvedor é possível continuador do progresso de desenvolvimento do sistema. O documento está dividido entre: os objetos para serem utilizados para todos os bancos de dados e uma parte específica para o banco de dados MySQL, além de uma parte referente a uma customização do padrão especificamente para o banco de apoio de indexação dos script back-end Perl/Python.

A chave de cadastro de novos postos deve refletir o nome do banco de dados a ser utilizado por exemplo, o banco de dados provê a localidade de Gazeta Vix → Logo a sintaxe será:

**Sintaxe:**

**gv\_postos**

Onde:

gv\_- indicar que é uma chave do banco de dados Gazeta Vix

postos - Indica o nome da aplicação

Para o nome da aplicação utilizar tudo em minúsculo com no máximo 15 caracteres.

O nome de uma tabela deverá ser sugestivo. Deve-se fazer o uso de nomenclatura orientado a objeto, por exemplo: se no departamento de Transmissão for necessário manter uma tabela de postos, ou manutenção, esta tabela deve ser nomeada identificando claramente seu conteúdo, isto significa que seu nome então deverá ser Posto e seu respectivos dados distribuídos dentro de subcolunas referente a tabela em questão

**Sintaxe**

**TB\_{[A….Z][{a..z}]} TB\_posto,** onde:

TB\_ - Indicará que é uma tabela do banco de dados

xxxxxx ou posto - indicará o nome da tabela.

**Regras**

* Primeira letra em maiúscula, demais em minúsculas. Para cada palabra interna, primeira letra em maiúscula, notação húngara;
* Limite-se a usar 30 caracteres - padrão de grande parte dos bancos de dados. Em caso de dúvida, consulte esta informação no manual do banco de dados que estiver utilizando;
* O nome da tabela deve estar sempre no singular (Exemplo: *Cliente* no lugar de *Clientes*);
* Evite usar abreviações, se necessário use as conhecidas;
* Não utilize acentuação ou caracteres especiais;

**Exemplos:**

**TB\_Pessoa -** Tabela de pessoas;

**TB\_Funcionario -** Tabela de funcionários;

**Nome dos atributos**

Seguindo o mesmo padrão utilizado por tabelas, deve-se identificar a coluna da tabela de maneira clara e descritiva, somente use abreviações quando se tratar de domínio público, exemplo FGTS, fundo de garantia por tempo de serviço.

**Sintaxe:**

{ [A..Z][{a..z}] } XXXXXXXXXX, onde

XXXXXXXX - Indicará o nome do campo

**Regras:**

* Primeira letra em maiúscula, demai sem minúsculas. Para cada palavra interna, primeira letra em maiúscula, notação húngara;
* Não usar preposições;
* para siglas utilizar todas as letras em maiúsculo;
* Usar palavras no singular e sem acentuação;
* usar nome que identifique e individualize o dado dentro do “ambiente” do cliente;
* Dar nomes distintos para dados distintos;
* Utilizar o nome sempre no singular;
* evite utilizar abreviações, caso seja extremamente necessário utilize as de domínio público(consulte a **nota 4** descrita abaixo) ou limite-se a 4 letras;

**Exemplo:**

Sem abreviações

* Nome
* NomeFuncionario
* DataNascimento
* ValorSarial

Com abreviações:

* NomeFuncResp
* GestorResp
* CodTermo

**NOTA 1:** Para cada campo de chave primária utilizar a seguintes regras:

* Identificador-da-chave[nome da tabela]

Exemplos:

|  |  |
| --- | --- |
| Nome da tabela | Identificador |
| TB\_Pessoa | CodPessoa |
| TB\_Laudo | CodLaudo |
| TB\_Historico | CodHistorico  DtHistorico |

**NOTA 2:** Caso a chave seja composta utilize um nome “simples que identifique a composição, evitando usar abreviações.

**NOTA 3:** Nomenclatura - Um aspecto importante quando a **nomenclatura dos dados**, é que a regra para codificação dos nomes seja tal que, a partir do significado dos dados, se possa chegar ao seu nome. Este aspecto é importante para que se possa aproveitar o potencial dos dados armazenados através da identificação da existência ou não dos dados necessários a uma aplica~ção, evitando-se a criação de redundância desnecessárias

Uma estratégia que utilizada par aeste fim, é ter uma regra para formação de nome:

Formato geral de um nome dado

**TIPO-QUALIFICADO 1…. QUALIFICADOR N- IDENTIFICADOR**

Onde: Esse ti

**[RF 09] – Sistema de Usuário e Senha**

O sistema provê os meios com uma página criptografada de login dentro do server-serviço sobre a estrutura PHP que pretendo que ele encaminhe os utilizadores para diferentes páginas dependendo do seu perfil, tais como Funcionário, $Consulta/Não Funcionário. Existe uma tabela login que tratará um campo chamado "perfil" que ou está a 0 ou a 1 tanto para Funcionário e $Consulta/Não Funcionário. O objetivo é encaminhar os utilizadores dependendo do seu perfil para diferentes páginas dentro da estrutura.

**VIEW**

Deve-se utilizado a mesma semântica utilizada para as tabelas. Deve ser prefixada a palavra “VW’² seguindo o nome da tabela