



Universidade Federal
do Rio de Janeiro
Escola Politécnica

**FERRAMENTA DE ACOMPANHAMENTO DE RELAÇÕES COM
COMUNIDADES ADJACENTES ÀS OPERAÇÕES DE EMPRESA DE
EXPLORAÇÃO DE RECURSOS NATURAIS**

Rodrigo Maestrelli Leobons

Projeto de Graduação apresentado ao Curso de Engenharia de Computação e Informação da Escola Politécnica, Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Engenheiro.

Orientador: Antônio Cláudio Gomez de Sousa

Rio de Janeiro
Setembro de 2013

Universidade Federal do Rio de Janeiro

Escola Politécnica

Departamento de Eletrônica e de Computação

**Ferramenta de Acompanhamento de Relações com
Comunidades Adjacentes às Operações de Empresa de
Exploração de Recursos Naturais**

Autor:

Rodrigo Maestrelli Leobons

Orientador:

Prof. Antônio Cláudio Gomez de Sousa

Examinador:

Prof. Aloysio de Castro Pinto Pedroza

Examinador:

Prof. Flávio Luis de Mello

Leobons, Rodrigo Maestrelli

Ferramenta de Acompanhamento de Relações com Comunidades Adjacentes às Operações de Empresa de Exploração de Recursos Naturais/ Rodrigo Maestrelli Leobons. – Rio de Janeiro: UFRJ/ Escola Politécnica, 2013.

XII, 59 p.: il.; 29,7 cm.

Orientador: Antônio Cláudio Gomez de Sousa

Projeto de Graduação – UFRJ/ Escola Politécnica/ Curso de Engenharia de Computação e Informação, 2013.

Referencias Bibliográficas: p. 54.

1. Introdução 2.Especificação de requisitos de software(ERS). 3. Projeto do Software. 4.Testes e resultados. 5.Conclusões.

I. de Sousa, Antônio Cláudio Gomez. II. Universidade Federal do Rio de Janeiro, Escola Politécnica, Curso de Engenharia de Computação e Informação. III. Ferramenta de Acompanhamento de Relações com Comunidades Adjacentes às Operações de Empresa de Exploração de Recursos Naturais.

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Escola Politécnica – Departamento de Eletrônica e de Computação

Centro de Tecnologia, bloco H, sala H-217, Cidade Universitária

Rio de Janeiro – RJ CEP 21949-900

Este exemplar é de propriedade da Universidade Federal do Rio de Janeiro, que poderá incluí-lo em base de dados, armazenar em computador, microfilmear ou adotar qualquer forma de arquivamento.

É permitida a menção, reprodução parcial ou integral e a transmissão entre bibliotecas deste trabalho, sem modificação de seu texto, em qualquer meio que esteja ou venha a ser fixado, para pesquisa acadêmica, comentários e citações, desde que sem finalidade comercial e que seja feita a referência bibliográfica completa.

Os conceitos expressos neste trabalho são de responsabilidade do autor e do(s) orientador(es).

AGRADECIMENTO

À minha família.

RESUMO

Esse trabalho tem como objetivo mostrar uma ferramenta para monitoramento de relações com as comunidades no entorno de operações de empresas, a fim de evitar o descontentamento da sociedade local para com a empresa. Com a crescente preocupação com a sustentabilidade, sendo o social um de seus tripés, esse sistema se faz muito importante não só para o crescimento sustentável das empresas, como também é uma ferramenta para que a companhia evite custos desnecessários provenientes de manifestações que paralise operações ou que causem danos à imagem. Tem como objetivo criar um banco de dados das manifestações das comunidades, gerando relatórios e mapa de criticidade, a fim de auxiliar na tomada de decisões sobre ações com o intuito de mitigar impactos gerados pela empresa. Nesse projeto será utilizada linguagem com orientação a objeto, PHP, e testes serão conduzidos para verificação das funcionalidades propostas.

Palavras-Chave: manifestação, comunidades, banco de dados

ABSTRACT

This work aims to show a tool for monitoring relations with the communities surrounding the operations of companies, in order to avoid the displeasure of the local society to the company. With the growing concern for sustainability, and the social being one of its tripods, this system becomes very important not only for the sustainable growth of enterprises, it is also a tool for the company to avoid unnecessary costs from manifestations that paralyzes operations or causes damage to the image. It aims to create a database of community manifestations, generating reports and map of criticality, to assist in taking decisions about actions in order to mitigate the impacts generated by the company. This project will use object-oriented language, PHP, and tests will be conducted to verify the proposed features.

Key-words: manifestations, communities, data base

SIGLAS

API – Application Programming Interface (Interface de Programação de Aplicativos)

DD – Dicionário de Dados

DER – Diagrama de Entidades e Relacionamentos

ERS – Especificação de Requisito de Software

HTML – HyperText Markup Language (Linguagem de Marcação de Hipertexto)

PHP – PHP: Hypertext Preprocessor

SQL – Structured Query Language (Linguagem de Consulta Estruturada)

UC – Use Case (Caso de Uso)

UFRJ – Universidade Federal do Rio de Janeiro

UML – Unified Modeling Language (Linguagem de Modelagem Unificada)

Sumário

1. Introdução	1
1.1. Considerações Iniciais	1
1.2. Justificativa	1
1.3. Objetivo	2
1.4. Metodologia	2
1.5. Estruturada monografia	3
2. Especificação de requisitos de software (ERS)	4
2.1. Introdução	4
2.1.1. Finalidade	4
2.1.2. Escopo	4
2.1.3. Definições, Acrônimos e Abreviaturas	4
2.1.4. Referências	5
2.1.5. Resumo	5
2.2. Descrição Geral	5
2.2.1. Perspectiva do Produto	5
2.2.2. Funções do Produto	5
2.2.2.1. Funcionalidades exclusivas dos usuários Administradores	6
2.2.2.2. Funcionalidades exclusivas dos usuários Gerentes	7
2.2.2.3. Funcionalidades exclusivas dos usuários Analistas	8
2.2.2.4. Funcionalidades de uso geral	10
2.2.3. Características do Usuário	14
2.2.4. Restrições	14
2.2.5. Pressupostos e Dependências	15
2.2.6. Postergar Requisitos	15
2.3. Requisitos Específicos	15
2.3.1. Interfaces Externas	15
2.3.1.1. Interfaces dos Usuários	15
2.3.1.2. Interfaces de Hardware	16
2.3.1.3. Interfaces de Software	16
2.3.1.4. Interfaces de Comunicação	16
2.3.2. Requisitos Funcionais	16
2.3.2.1. Casos de Uso	16
2.3.2.2. Diagrama de Classes	32
2.3.3. Requisitos de Desempenho	35
2.3.4. Restrições de Projeto	36

2.3.5.	Atributos	36
2.3.6.	Outros Requisitos.....	37
3.	Projeto do Software.....	38
3.1.	Arquitetura	38
3.2.	Modelo de Dados.....	39
3.2.1.	Diagrama de Entidades e Relacionamentos	39
3.2.2.	Dicionário de Dados.....	40
3.3.	Implementação do sistema.....	44
3.4.	Ferramentas utilizadas	44
4.	Testes e Resultados.....	46
4.1.	Cadastramento de usuários	46
4.2.	Cadastramento de estados e municípios	46
4.3.	Cadastramento de questão crítica	47
4.4.	Cadastramento de comunidades	47
4.5.	Cadastramento de manifestações	47
4.6.	Cadastramento de manifestantes	48
4.7.	Cadastramento de atividades	48
4.8.	Cadastramento de pontos de atenção.....	48
4.9.	Visualização de questão crítica.....	49
4.10.	Visualização de comunidades.....	49
4.11.	Visualização de manifestações	49
4.12.	Visualização de manifestantes.....	50
4.13.	Visualização de atividades.....	50
4.14.	Visualização de pontos de atenção.....	51
4.15.	Geração de mapa	51
4.16.	Geração de relatório.....	52
4.17.	Login e logout	52
5.	Conclusões	53
6.	Bibliografia	54
	Apêndice A.....	55
	Apêndice B	58

Lista de Figuras

Figura 1 – Diagrama de Casos de Uso	17
Figura 2 – Diagrama de Classes	33
Figura 3 – Arquitetura do sistema.....	38
Figura 4 – Diagrama de Entidades e Relacionamentos.....	39
Figura 5 – Tela de login	58
Figura 6 – Tela inicial	58
Figura 7 – Tela com mapa de comunidades	59
Figura 8 – Tela com relatório	59

Lista de Tabelas

Tabela 1 – Dicionário de Dados: CR_Demands.....	40
Tabela 2 – Dicionário de Dados: CR_Issues.....	41
Tabela 3 – Dicionário de Dados: CR_People.....	41
Tabela 4 – Dicionário de Dados: CR_Communities.....	42
Tabela 5 – Dicionário de Dados: CR_City.....	42
Tabela 6 – Dicionário de Dados: CR_State.....	42
Tabela 7 – Dicionário de Dados: CR_Users	43
Tabela 8 – Dicionário de Dados: CR_Tasks	43
Tabela 9 – Dicionário de Dados: CR_AttentionPoints	44
Tabela 10 – Testes e resultados: Cadastramento de usuários.....	46
Tabela 11 – Testes e resultados: Cadastramento de estados e municípios	46
Tabela 12 – Testes e resultados: Cadastramento de questão crítica	47
Tabela 13 – Testes e resultados: Cadastramento de comunidades	47
Tabela 14 – Testes e resultados: Cadastramento de manifestações.....	47
Tabela 15 – Testes e resultados: Cadastramento de manifestantes	48
Tabela 16 – Testes e resultados: Cadastramento de atividades	48
Tabela 17 – Testes e resultados: Cadastramento de pontos de atenção	49
Tabela 18 – Testes e resultados: Visualização de questão crítica	49
Tabela 19 – Testes e resultados: Visualização de comunidades	49
Tabela 20 – Testes e resultados: Visualização de manifestações.....	50
Tabela 21 – Testes e resultados: Visualização de manifestantes.....	50
Tabela 22 – Testes e resultados: Visualização de atividades	51
Tabela 23 – Testes e resultados: Visualização de ponto de atenção.....	51
Tabela 24 – Testes e resultados: Geração de mapa	52
Tabela 25 – Testes e resultados: Geração de relatório.....	52
Tabela 26 – Testes e resultados: Login e logout	52

1. Introdução

1.1. Considerações Iniciais

Empresas de exploração de recursos naturais, sejam elas de mineração ou de petróleo, necessitam de grandes campos para realizarem suas operações de retiradas desses recursos para comercialização. Na grande maioria dos casos, essas áreas são localizadas longe dos grandes centros consumidores, fazendo com que essas companhias precisem de uma grande logística para transportarem esses recursos, seja através de meio rodoviário, ferroviário ou marítimo. Com todas essas operações, impactos também são gerados para a população que vive em seu entorno, sejam estes poeira, barulho ou até mesmo problemas trazidos pelo grande aumento no número populacional das cidades ao longo dessas operações.

Os impactos que são gerados para essas pessoas por esse tipo de empresa podem causar grandes transtornos em suas vidas, gerando uma grande insatisfação dessa população local, ocasionando o descontentamento desta com a companhia exploradora, podendo fazer até com que os habitantes dessas comunidades causem alguma paralização nas operações da empresa, gerando uma perda que poderá chegar a uma grande quantia monetária.

Um dos grandes temas recorrentes nas grandes empresas atualmente é a sustentabilidade, tendo em um dos seus três pilares o social (além dos pilares econômico e ambiental). Sendo assim, o relacionamento com as comunidades é fundamental para a garantia de que a empresa está no caminho de se tornar uma companhia sustentável. Provando ser uma empresa engajada na sustentabilidade, a companhia ganha uma imagem positiva e reduz o risco para os seus investidores, aumentando o interesse em suas ações e possibilitando uma maior rentabilidade.

1.2. Justificativa

A motivação para o desenvolvimento de uma ferramenta que gerencie as manifestações da comunidade surgiu a partir do trabalho desenvolvido no setor de

responsabilidade social de uma grande empresa multinacional do setor de mineração e metais.

Para o projeto desenvolvido para essa companhia, a ferramenta foi desenvolvida em Microsoft Excel, devido ao fato de que a ideia ainda estava em desenvolvimento, para que servisse de modelo para o desenvolvimento de um sistema mais robusto e complexo, que será apresentado nesse projeto.

1.3. Objetivo

O objetivo desse projeto é a implementação de um sistema web para o controle de manifestações das comunidades no entorno de operações de quaisquer tipo de empresa de exploração de recursos naturais, possibilitando uma maior visibilidade das demandas, reclamações e elogios oriundos das pessoas que vivem nessas localidades e que tem suas vidas afetadas de alguma maneira por essas empresas. Com essa base sólida de informações e a geração de relatórios, as empresas poderão tomar decisões críticas de uma maneira mais confiável e precisa.

O sistema deverá ser utilizado em um primeiro momento apenas para a diretoria envolvida com o relacionamento com essas comunidades, podendo ser expandido para consulta de informações a todas as áreas que necessitarem dessas informações para tomadas de decisões.

Além dos objetivos da criação do sistema, esse projeto também visa a expansão dos conhecimentos em relação a sistemas web, modelagem de dados e sistemas de gerenciamento de banco de dados, uma vez que a primeira versão dessa ferramenta foi desenvolvida em Excel, e este seria um aprimoramento dela.

1.4. Metodologia

Para a realização desse projeto, será utilizada a metodologia de orientação a objeto, trazendo uma facilidade maior de incorporação de novas funcionalidades a medida que forem sendo identificadas oportunidades de melhoria ao longo da implementação do sistema proposto.

As ferramentas a serem utilizadas para a conclusão desse projeto foram escolhidas devido a experiência prévia em sua utilização, na facilidade de obtê-las e na curva de aprendizagem favorável.

As linguagens escolhidas para o desenvolvimento foram o PHP e o Java script, devido ao sistema ser baseado na web e essas linguagens serem de fácil utilização e integração com o sistema de gerenciamento de banco de dados.

Uma preocupação recorrente ao longo do desenvolvimento do sistema será a facilidade de utilização, pois devido à experiência prévia de implantação de uma ferramenta mais básica em um cliente, a qual me motivou a realizar esse trabalho, foi possível perceber que era de fundamental importância, pois o usuário tende a não aceitar facilmente a implantação de um novo sistema no seu dia a dia no trabalho. Ao final da implantação do sistema, testes com alguns usuários serão realizados para verificar o desempenho das funcionalidades propostas.

1.5. Estruturada monografia

Além do capítulo atual, introdutório, que expõe o tema e os objetivos desse projeto, esta monografia está dividida em mais quatro capítulos, que possuem o detalhamento sobre a solução apresentada.

No capítulo dois é descrita a especificação de requisitos de software para o sistema proposto, baseada na norma ANSI/IEEE 830 simplificada, que traz todo o detalhamento sobre a ferramenta desse trabalho.

O capítulo três apresenta o projeto do software desenvolvido, explicitando como se deu a elaboração do sistema. É descrita a arquitetura da ferramenta, bem como o modelo de dados implementado, a implementação do sistema e a descrição das ferramentas necessárias para o desenvolvimento.

Já no capítulo quatro, há a apresentação dos testes realizados a partir da versão alpha do sistema, mostrando quais seriam os resultados esperados para cada um deles e o que foi obtido.

Por fim, temos o capítulo cinco, que apresenta as considerações finais do projeto, trazendo as conclusões obtidas com a implementação desse sistema, e sugestões sobre novas atualizações da ferramenta.

2. Especificação de requisitos de software (ERS)

2.1. Introdução

2.1.1. Finalidade

Essa ERS tem como objetivo fornecer informações aos clientes e desenvolvedores sobre as funcionalidades do sistema, como também estabelecer um acordo entre os envolvidos no projeto, com o intuito de compreensão plena de como ocorrerá o funcionamento da ferramenta. Portanto, esse documento contará com a especificação de todos os requisitos que deverão ser atendidos, e também por aqueles que por algum motivo não serão atendidos pelo software.

2.1.2. Escopo

O software em questão é chamado de CR_Model 1.0 e tem como objetivo auxiliar no relacionamento com as comunidades por parte de empresas de exploração de recursos naturais, fornecendo um ambiente para inclusão de manifestações das comunidades no entorno de operações da companhia, gerando relatórios que subsidiem as tomadas de decisões da empresa em relação aos relacionamentos com essas comunidades.

Com a ferramenta, os processos adotados nas relações com comunidades poderão ser aprimorados, criando uma sistematização de informações relativas às manifestações provenientes das comunidades, fornecendo subsídios para uma atuação mais proativa e estratégica da empresa no que diz respeito ao gerenciamento desse relacionamento.

A partir da implantação da ferramenta, o processo de gestão das manifestações das comunidades contará com um sistema online padronizado, possuindo um banco de dados integrado entre todas as operações da empresa.

2.1.3. Definições, Acrônimos e Abreviaturas

ERS – Documento que permite ao cliente descrever suas necessidades a fim de que o desenvolvedor possa compreendê-las.

API – Application Programming Interface (Interface de Programação de Aplicativos).

2.1.4. Referências

Não se aplica.

2.1.5. Resumo

Nos itens a seguir dessa ERS, haverá a apresentação do produto, com a descrição geral, apresentando as funcionalidades do produto, características do usuário do sistema, restrições e requisitos futuros do software.

Será apresentado também nessa especificação os requisitos específicos, definindo as interfaces externas do produto e apresentando os casos de uso e diagrama de classes, assim como os requisitos de desempenho e as restrições de projeto.

2.2. Descrição Geral

2.2.1. Perspectiva do Produto

A maior parte da ferramenta a ser desenvolvida será um sistema independente. Porém, apenas para a geração do mapa de comunidades, será necessária a utilização de API do Google Maps para que os mapas desse aplicativo possam ser utilizados para o georreferenciamento dessas comunidades.

2.2.2. Funções do Produto

Nessa seção são apresentadas as principais funcionalidades do sistema de gestão de comunidades, listados a seguir:

- ➔ Funcionalidades exclusivas dos usuários Administradores:
 - ✓ Cadastramento de usuários;
 - ✓ Cadastramento de estados e municípios;

→ Funcionalidades exclusivas dos usuários Gerentes:

- ✓ Cadastramento de questão crítica;
- ✓ Cadastramento de comunidades;

→ Funcionalidades exclusivas dos usuários Analistas:

- ✓ Cadastramento de manifestações;
- ✓ Cadastramento de manifestantes;
- ✓ Cadastramento de atividades;
- ✓ Cadastramento de pontos de atenção;

→ Funcionalidades de uso geral:

- ✓ Visualização de questão crítica;
- ✓ Visualização de comunidades;
- ✓ Visualização de manifestações;
- ✓ Visualização de manifestantes;
- ✓ Visualização de atividades;
- ✓ Visualização de pontos de atenção;
- ✓ Geração de mapa;
- ✓ Geração de relatório;
- ✓ Login e logout;

2.2.2.1. Funcionalidades exclusivas dos usuários Administradores

2.2.2.1.1. Cadastramento de usuários

- Objetivo da função
 - Gerenciar o cadastro dos usuários, permitindo a criação e edição desses usuários. Os usuários não poderão ser excluídos do sistema, apenas ter o seu acesso cortado, pois estes poderão estar associados a alguma manifestação ou questão crítica inserida no sistema.
- Características da função
 - O sistema poderá ter três tipos diferentes de usuários: analistas, gerentes e administradores.
 - A senha deverá ser gerada automaticamente e enviada ao novo usuário.
 - Os campos para criação dos usuários serão os seguintes:
 - Nome e sobrenome do usuário

- Login
- Telefone
- E-mail
- Cargo (analista, gerente ou administrados)
- Supervisor (se houver)

2.2.2.1.2. Cadastramento de estados e municípios

- Objetivo da função
 - Cadastrar os estados e municípios para que possam ser atrelados às comunidades a serem inseridas. Possíveis alterações de nomes de estados e municípios também poderão ser realizadas.
- Características da função
 - O único campo para a criação do estado deverá ser o nome do estado, e para a criação do município, além do nome do município também deverá haver o campo para a escolha do município ao qual o estado faça parte.

2.2.2.2. Funcionalidades exclusivas dos usuários Gerentes

2.2.2.2.1. Cadastramento de questão crítica

- Objetivo da função
 - Gerenciar o cadastro da questões críticas a serem inseridas no sistema, permitindo a adição de novos itens, bem como sua edição. Se houver alguma questão crítica atrelada a uma manifestação, a questão não poderá ser excluída.
- Características da função
 - O estado, município, comunidade e responsável pela questão crítica deverão estar previamente cadastrados no sistema.
 - Os campos para esse cadastro serão os seguintes:
 - Nome da questão
 - Descrição
 - Severidade (poderá ser leve, moderada, grave e muito grave)
 - Data de Início
 - Data de Conclusão
 - Estado
 - Município

- Comunidade
- Responsável (o responsável deverá ser um usuário com nível gerente)

2.2.2.2.2. Cadastramento de comunidades

- Objetivo da função
 - Gerenciar o cadastro de comunidades, permitindo a inclusão de nova comunidade e posterior edição. As comunidades não poderão ser apagadas.
- Características da função
 - O estado e município da comunidade deverão estar previamente cadastrados.
 - Os campos para esse cadastro serão os seguintes:
 - Nome da comunidade
 - Estado
 - Município
 - Latitude
 - Longitude

2.2.2.3. Funcionalidades exclusivas dos usuários Analistas

2.2.2.3.1. Cadastramento de manifestações

- Objetivo da função
 - Gerenciar o cadastro de manifestações, permitindo a entrada de novas manifestações no sistema, bem como a edição. As manifestações não poderão ser apagadas, para que não haja manipulação de dados no sistema. Uma vez que a manifestação foi aberta, ela poderá ser fechada, apresentando a justificativa para a conclusão.
- Características da função
 - O estado, município e comunidade, e se houver questão crítica, aos quais a manifestação estiver atrelada, deverão estar previamente cadastrados.
 - Os campos para esse cadastro serão os seguintes:
 - Descrição
 - Estado

- Município
- Comunidade
- Data de início
- Data de conclusão
- Tema
- Categoria
- Severidade (poderá ser leve, moderada, alta ou muito alta)
- Probabilidade (poderá ser baixa, moderada, alta, muito alta ou ocorrência)
- Resultado
- Responsável
- Manifestante
- Questão crítica relacionada

2.2.2.3.2. Cadastramento de manifestantes

- Objetivo da função
 - Gerenciar o cadastramento de manifestantes, permitindo a criação e alteração de manifestantes. Os manifestantes inseridos no sistema não poderão ser excluídos, para que haja um histórico destes.
- Características da função
 - A comunidade a qual o manifestante fizer parte deverá estar previamente cadastrada.
 - Os campos para esse cadastro serão os seguintes:
 - Nome e sobrenome do manifestante
 - Endereço
 - Comunidade
 - Telefone
 - E-mail

2.2.2.3.3. Cadastramento de atividades

- Objetivo da função
 - Gerenciar o cadastramento de atividades, permitindo a criação, alteração e exclusão de registros. Para toda nova manifestação inserida no sistema, uma atividade para resposta ao manifestante deverá ser criada automaticamente, com data para resolução de 5

dias após o cadastro, e tendo o usuário responsável pela manifestação como o responsável por essa atividade, que não poderá ser excluída, por ser a atividades obrigatória para toda manifestação.

- Características da função
 - As atividades serão geradas a partir de uma manifestação, que será automaticamente atrelada a essa nova atividade.
 - Os campos para esse cadastro serão os seguintes:
 - Nome da tarefa
 - Descrição
 - Data de início
 - Data de previsão de conclusão
 - Data de conclusão
 - Responsável

2.2.2.3.4. Cadastramento de pontos de atenção

- Objetivo da função
 - Gerenciar o cadastramento de pontos de atenção, permitindo a criação, alteração e exclusão de registros.
- Características da função
 - Os pontos de atenção serão gerados a partir de uma manifestação, que será automaticamente atrelada a essa novo ponto de atenção.
 - Os campos para esse cadastro serão os seguintes:
 - Nome do ponto de atenção
 - Descrição
 - Impacto
 - Plano de ação
 - Status (Em aberto, finalizado)
 - Responsável pela resolução

2.2.2.4. Funcionalidades de uso geral

2.2.2.4.1. Visualização de questão crítica

- Objetivo da função
 - Visualizar a lista de questões críticas, observando suas características. Deverá haver um filtro para que esta visualização

se adeque às expectativas do usuário.

- Características da função
 - A lista de questões críticas padrão que será exibida, mostrará todas as que estiverem em aberto.
 - O filtro poderá ser composto com os seguintes campos:
 - Estado
 - Município
 - Comunidade
 - Responsável

2.2.2.4.2. Visualização de comunidades

- Objetivo da função
 - Visualizar a lista de comunidades, observando suas características. Deverá haver um filtro para que esta visualização se adeque às expectativas do usuário.
- Características da função
 - O filtro poderá ser composto com os seguintes campos:
 - Estado
 - Município

2.2.2.4.3. Visualização de manifestações

- Objetivo da função
 - Visualizar a lista de manifestações, observando suas características. Deverá haver um filtro para que esta visualização se adeque às expectativas do usuário.
- Características da função
 - A lista de manifestações padrão que será exibida, mostrará todas as que estiverem em aberto.
 - O filtro poderá ser composto com os seguintes campos:
 - Estado
 - Município
 - Comunidade
 - Responsável
 - Manifestante

2.2.2.4.4. Visualização de manifestantes

- Objetivo da função
 - Visualizar a lista de manifestantes, observando suas características. Deverá haver um filtro para que esta visualização se adeque às expectativas do usuário.
- Características da função
 - O filtro poderá ser composto com o campo de comunidade.

2.2.2.4.5. Visualização de atividades

- Objetivo da função
 - Visualizar a lista de atividades, observando suas características. Deverá haver um filtro para que esta visualização se adeque às expectativas do usuário.
- Características da função
 - A lista poderá ser gerada a partir da visualização de uma manifestação, sendo que esta lista exibirá todas as atividades em aberto daquela manifestação.
 - O filtro poderá ser composto com o campo do responsável.
 - Na página inicial, será exibida uma lista com as atividades do usuário atual e que estiverem em aberto, por ordem de previsão de conclusão.
 - Se a data de previsão para conclusão da atividade estiver vencida e a atividade estiver em aberto, o campo dessa data deverá aparecer em vermelho para chamar a atenção do usuário quanto ao atraso.

2.2.2.4.6. Visualização de pontos de atenção

- Objetivo da função
 - Visualizar a lista de pontos de atenção, observando suas características. Deverá haver um filtro para que esta visualização se adeque às expectativas do usuário.
- Características da função
 - A lista poderá ser gerada a partir da visualização de uma manifestação, sendo que esta lista exibirá todos os pontos de atenção em aberto daquela manifestação.
 - O filtro poderá ser composto com o campo do responsável e de status.

- Na página inicial, será exibida uma lista com os pontos de atenção do usuário atual e que estiverem em aberto.

2.2.2.4.7. Geração de mapa

- Objetivo da função
 - Gerar um mapa com a localização das comunidades cadastradas, identificando-as com a cor do farol de situação de cada uma. Quando clicarmos em uma comunidade no mapa, deverá aparecer o nome da mesma.
- Características da função
 - Poderá ser realizado um filtro no mapa em relação a estado e/ou município.
 - O critério para a geração do farol deverá ser o seguinte:
 - Vermelho (crítica): a nota deverá ser menor do que cinco.
 - Amarelo (atenção): a nota deverá ser maior ou igual a cinco e menor que sete.
 - Verde (ok): a nota deverá ser maior que sete.
 - A composição da nota para cada comunidade deverá ser estabelecida somando-se as duas notas encontradas a partir das fórmulas abaixo e dividindo-as por dois:

▪

▪

2.2.2.4.8. Geração de relatório

- Objetivo da função
 - Gerar um relatório de gráficos a partir dos dados inseridos no sistema, obtendo alguns indicadores a partir dos filtros especificados pelo usuário.
- Características da função
 - O filtro poderá ser composto pelos seguintes campos:
 - Estado
 - Município

- Comunidade
- Analista
- Gerente
- Os indicadores apresentados serão os abaixo:
 - Percentual e número de manifestações em aberto e fechadas
 - Percentual e número das criticidades das manifestações
 - Percentual e número das categorias das manifestações
 - Percentual e número do temas das manifestações
 - Número de questões críticas em aberto e percentual das criticidades das questões durante o período especificado
 - Número e percentual dos status das atividades

2.2.2.4.9. Login e Logout

- Objetivo da função
 - Criar a sessão para o usuário utilizar o sistema e posteriormente encerrar essa sessão.
- Características da função
 - A proteção do sistema será realizada através de senha de acesso.

2.2.3. Características do Usuário

Os usuários do sistema serão os funcionários da empresa que trabalham no setor encarregado do relacionamento com as comunidades. Por ser um sistema que abrangerá uma empresa que possui operações em todo o território nacional, a cultura e o nível de interação com sistemas dos envolvidos terá uma grande variação. Os usuários serão permanentes, com a adição de novos usuários de acordo com a contratação de novos funcionários para o setor. Caso haja o desligamento de algum funcionário, este terá seu acesso ao sistema desativado, porém o usuário permanecerá no sistema para histórico. Todos os usuários serão treinados para a correta utilização da ferramenta.

2.2.4. Restrições

Por conter informações sigilosas quanto ao relacionamento externo da empresa, a segurança do sistema deverá ser garantida. Por se tratar de um ambiente Web, possuindo um servidor Apache e um sistema gerenciador de banco de dados SQL

Server, o sistema deverá possuir uma política de segurança para garantir que não haja acessos sem autorização e que os dados trafeguem pela rede de maneira segura, que deverá ser garantida através da criptografia do banco de dados. Além disso, por se tratar de uma ferramenta que será utilizada por uma grande empresa e que deverá orientar as ações estratégicas em relação ao trato com as comunidades, deverá ser garantida a fiabilidade do sistema. Também deverá ser realizado um backup das informações a cada 24 horas, para garantir que não haverá perda de dados caso algum incidente venha a ocorrer.

2.2.5. Pressupostos e Dependências

O sistema deverá se utilizar uma versão do Sistema Gerenciador de Banco de Dados SQL Server, da Microsoft, por ser uma ferramenta cuja utilização já é dominada pelo administrador da solução.

2.2.6. Postergar Requisitos

Até o momento, não foram postergados requisitos. Se houver a postergação de algum requisito, deverá ser incluída nessa seção uma explicação sobre este adiamento.

2.3. Requisitos Específicos

2.3.1. Interfaces Externas

2.3.1.1. Interfaces dos Usuários

A interface com o usuário será feita através da web, tendo como página inicial a tela para login no sistema. Após a entrada do usuário no sistema, ele encontra uma página com as informações essenciais que deverá conhecer. O menu de navegação deverá estar na lateral esquerda das telas da ferramenta. A navegação deverá ocorrer de maneira intuitiva, a fim de os usuários possam realizar suas ações a partir de um menor número de etapas. Alguns recursos visuais poderão ser usados para uma interface mais amigável.

2.3.1.2. Interfaces de Hardware

O sistema necessitará de ao menos uma máquina para funcionar como servidor de aplicação e como servidor de dados. Para a conexão com a Internet, serão necessários cabos e um roteador. Poderá haver a separação em duas máquinas para os dois servidores, porém será necessário uma maior quantidade de cabos.

2.3.1.3. Interfaces de Software

Sendo um sistema baseado na Web, onde as iterações do software ocorrem no servidor de aplicação, não há a obrigação de utilização de um sistema operacional proprietário, devendo apenas o usuário possuir um navegador para visualização de sites, o que é padrão em todos os sistemas operacionais.

2.3.1.4. Interfaces de Comunicação

Por ser um sistema baseado na Web, a comunicação utilizada pelo software será dada por meio da Internet, o que obrigará os usuários da ferramenta a possuírem uma boa conexão com a rede. Se o servidor de aplicação e o servidor de dados não estiverem na mesma máquina, deverá haver uma conexão segura entre essas duas máquinas.

2.3.2. Requisitos Funcionais

2.3.2.1. Casos de Uso

Diagrama de Casos de Uso:

A figura a seguir representa os atores e os casos de uso relacionados com cada um destes.

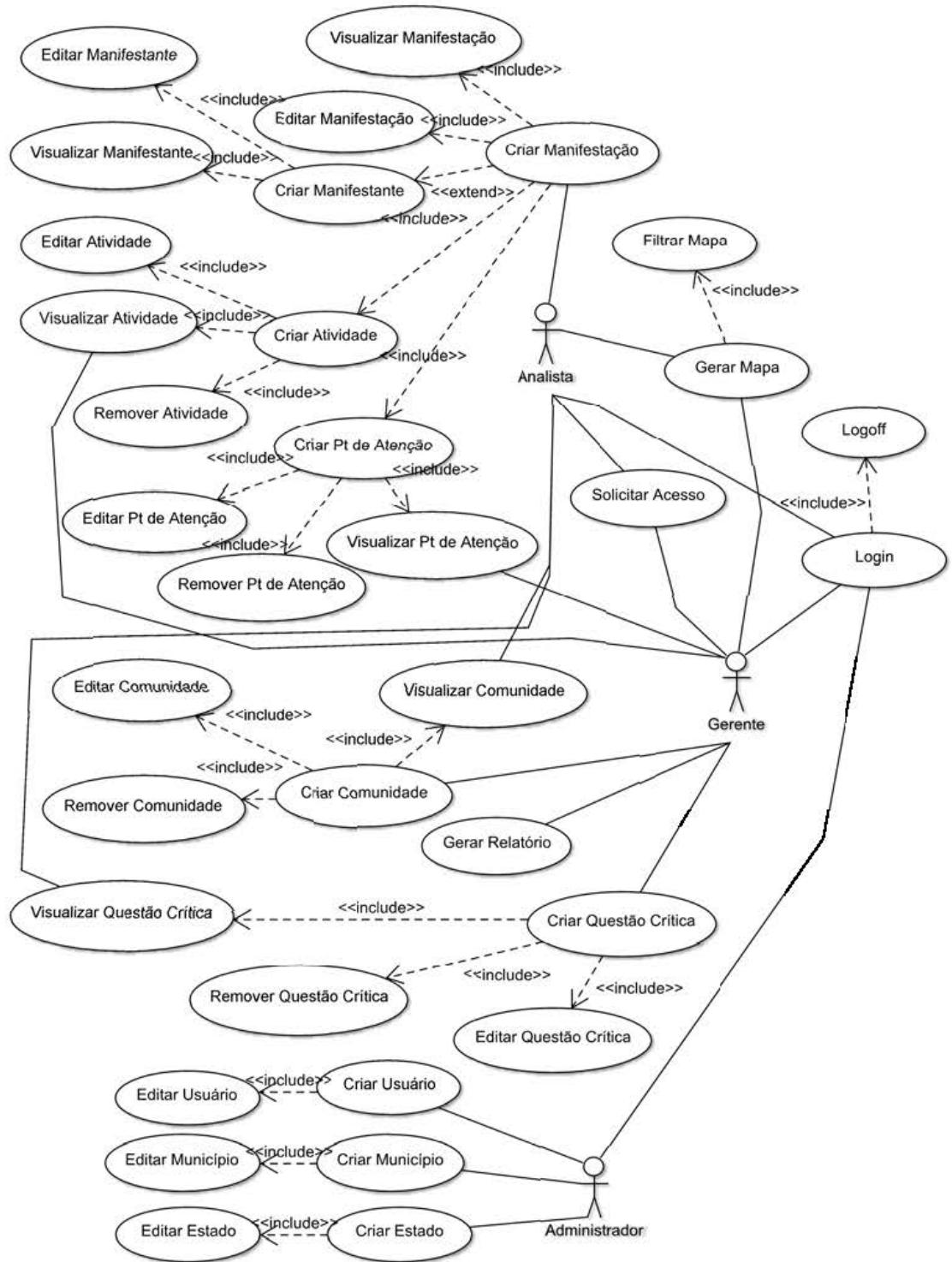


Figura 1 – Diagrama de Casos de Uso

Descrição dos Casos de Uso:

UC01: Criar Atividade

Descrição:	O usuário cria uma atividade para a resolução de uma determinada manifestação
Papéis:	Analista
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de analista e deve existir uma manifestação para a atividade ser relacionada.
Pós-condições:	A atividade é criada com sucesso e atrelada a uma manifestação.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas 2. O usuário insere as informações relativas a nova atividade. 3. O usuário confirma o cadastro da nova atividade. 4. A atividade é cadastrada.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC02: Criar Comunidade

Descrição:	O usuário insere uma nova comunidade no sistema.
Papéis:	Gerente
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de gerente. O estado e município relativos necessitam estar previamente criados.
Pós-condições:	Uma nova comunidade é adicionado ao sistema.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a opção para criação de nova comunidade 2. O usuário insere as informações relativas a nova comunidade. 3. O usuário confirma o cadastro da nova comunidade. 4. A comunidade é cadastrada.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC03: Criar Estado

Descrição:	O usuário insere um novo estado no sistema.
Papéis:	Administrador

Pré-condições:	O usuário necessita estar logado no sistema com o perfil de administrador.
Pós-condições:	Um novo estado é adicionado ao sistema.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a opção para criação de novo estado. 2. O usuário insere as informações relativas ao novo estado. 3. O usuário confirma o cadastro do novo estado. 4. O estado é cadastrado.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC04: Criar Manifestação

Descrição:	O usuário insere uma nova manifestação no sistema.
Papéis:	Analista
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de analista. A comunidade, o município, o estado e o manifestante relativos necessitam estar previamente cadastrados.
Pós-condições:	Uma nova manifestação é incluída nos sistema.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a opção para criação de nova manifestação. 2. O usuário insere as informações relativas a nova manifestação. 3. O usuário confirma o cadastro da nova manifestação. 4. A manifestação é cadastrada.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC05: Criar Manifestante

Descrição:	O usuário insere um novo manifestante no sistema.
Papéis:	Analista
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de analista.
Pós-condições:	Um novo manifestante é cadastrado no sistema.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona a opção para criação de novo manifestante.

2. O usuário insere as informações relativas ao novo manifestante.
3. O usuário confirma o cadastro do novo manifestante.
4. O manifestante é cadastrado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC06: Criar Município

Descrição: O usuário insere um novo município no sistema.

Papéis: Administrador

Pré-condições: O usuário necessita estar logado no sistema com o perfil de administrador. O estado relacionado precisa estar previamente criado.

Pós-condições: Um novo município é inserido no sistema.

Fluxo principal:

1. O usuário seleciona a opção para criação de novo município.
2. O usuário insere as informações relativas ao novo município.
3. O usuário confirma o cadastro do novo município.
4. O município é cadastrado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC07: Criar Ponto de Atenção

Descrição: O usuário cria um ponto de atenção relativo a uma determinada manifestação

Papéis: Analista

Pré-condições: O usuário necessita estar logado no sistema com o perfil de analista e deve existir uma manifestação para o ponto de atenção ser relacionado.

Pós-condições: Um novo ponto de atenção atrelado a uma manifestação é adicionado.

Fluxo principal:

1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas
2. O usuário insere as informações relativas ao novo ponto de atenção.
3. O usuário confirma o cadastro do novo ponto de atenção.

4. O ponto de atenção é cadastrado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC08: Criar Questão Crítica

Descrição: O usuário insere uma nova questão crítica no sistema.

Papéis: Gerente

Pré-condições: O usuário necessita estar logado no sistema com o perfil de gerente. O estado, município e comunidades relativos necessitam estar previamente criados.

Pós-condições: Uma nova questão crítica é inserida no sistema.

Fluxo principal:

1. O usuário seleciona a opção para criação de nova questão crítica.
2. O usuário insere as informações relativas a nova questão crítica.
3. O usuário confirma o cadastro da nova questão crítica.
4. A questão crítica é cadastrada.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se as informações obrigatórias não forem inseridas ou se os dados inseridos forem de tipo diferente do esperado, retorna ao passo (2).

UC09: Criar Usuário

Descrição: O usuário insere um novo usuário no sistema.

Papéis: Administrador

Pré-condições: O usuário necessita estar logado no sistema com o perfil de administrador.

Pós-condições: Um novo usuário é criado com acesso ao sistema e um e-mail é enviado a esse novo usuário com as suas informações de login

Fluxo principal:

1. O usuário seleciona a opção para criação de novo usuário.
2. O usuário insere as informações relativas ao novo usuário.
3. O usuário confirma o cadastro do novo usuário.
4. O usuário é cadastrado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se as informações obrigatórias não forem inseridas, se os dados inseridos forem de tipo diferente do esperado ou se houver um usuário com o mesmo login já previamente cadastrado, retorna ao passo (2).

UC10: Remover Atividade

Descrição: O usuário remove uma atividade para a resolução de uma determinada manifestação

Papéis: Analista

Pré-condições: O usuário necessita estar logado no sistema com o perfil de analista.

Pós-condições: A atividade é apagada do sistema.

Fluxo principal:

1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas
2. O usuário seleciona uma atividade entre as relacionadas a determinada manifestação.
3. O usuário seleciona a opção para remover a atividade.
4. O usuário confirma a exclusão da atividade.
5. A atividade é excluída.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (4), se a atividade for a principal para a manifestação, a opção de deletar é anulada e retorna ao passo (2).

UC11: Remover Ponto de Atenção

Descrição: O usuário remove um ponto de atenção relacionado a uma determinada manifestação

Papéis: Analista

Pré-condições: O usuário necessita estar logado no sistema com o perfil de analista.

Pós-condições: O ponto de atenção é apagado do sistema.

Fluxo principal:

1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas
2. O usuário seleciona um ponto de atenção entre as relacionadas a determinada manifestação.
3. O usuário seleciona a opção para remover o ponto de atenção.
4. O usuário confirma a exclusão do ponto de atenção.

5. O ponto de atenção é excluído.

Fluxo alternativo: Não há.

Fluxos de exceção: Não há.

UC12: Remover Questão Crítica

Descrição: O usuário remove uma questão crítica.

Papéis: Gerente

Pré-condições: O usuário necessita estar logado no sistema com o perfil de gerente.

Pós-condições: A questão crítica é apagada do sistema.

Fluxo principal:

1. O usuário seleciona uma questão crítica a partir da tabela
2. O usuário seleciona a opção para remover a questão crítica.
3. O usuário confirma a exclusão da questão crítica.
4. A questão crítica é excluída.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se houver qualquer manifestação relacionada a essa questão crítica, retorna ao passo (1).

UC13: Editar Atividade

Descrição: O usuário edita uma atividade para a resolução de uma determinada manifestação

Papéis: Analista

Pré-condições: O usuário necessita estar logado no sistema com o perfil de analista.

Pós-condições: A atividade é modificada com as novas informações inseridas.

Fluxo principal:

1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas
2. O usuário seleciona uma atividade entre as relacionadas a determinada manifestação.
3. O usuário modifica os campos que deseja.
4. O usuário confirma a edição da atividade.
5. A atividade é editada.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (4), se alguma informação obrigatória for deixada em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (3).

UC14: Editar Comunidade

Descrição:	O usuário edita uma comunidade no sistema.
Papéis:	Gerente
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de gerente.
Pós-condições:	A comunidade é modificada com os novos dados inseridos.
Fluxo principal:	<ol style="list-style-type: none">1. O usuário seleciona a comunidade na lista de comunidades existentes.2. O usuário edita as informações da comunidade que deseja.3. O usuário confirma a edição da comunidade.4. A comunidade é editada.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se alguma informação obrigatória for deixada em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC15: Editar Estado

Descrição:	O usuário edita um estado no sistema.
Papéis:	Administrador
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de administrador.
Pós-condições:	O estado é editado com as novas informações inseridas.
Fluxo principal:	<ol style="list-style-type: none">1. O usuário seleciona o estado na lista de estados existentes.2. O usuário edita as informações do estado que deseja.3. O usuário confirma a edição do estado.4. O estado é editado.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se alguma informação obrigatória for deixada em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC16: Editar Manifestação

Descrição:	O usuário edita uma determinada manifestação.
Papéis:	Analista

Pré-condições:	O usuário necessita estar logado no sistema com o perfil de analista.
Pós-condições:	A manifestação é modificada com as informações inseridas.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas. 2. O usuário altera as informações relativas a manifestação selecionada. 3. O usuário confirma a edição da manifestação. 4. A manifestação é editada.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se informações obrigatórias forem deixadas em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC17: Editar Manifestante

Descrição:	O usuário edita um manifestante no sistema.
Papéis:	Analista
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de analista.
Pós-condições:	O perfil do manifestante é editado com os novos dados inseridos.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona um manifestante na lista de manifestantes. 2. O usuário altera as informações desejadas relativas ao manifestante. 3. O usuário confirma a edição do manifestante. 4. O manifestante é editado.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Para o passo (3), se informações obrigatórias forem deixadas em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC18: Editar Município

Descrição:	O usuário edita um município no sistema.
Papéis:	Administrador
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de administrador.
Pós-condições:	O município é editado com as novas informações inseridas.
Fluxo principal:	<ol style="list-style-type: none"> 1. O usuário seleciona o município na lista de municípios existentes. 2. O usuário edita as informações do município que deseja.

3. O usuário confirma a edição do município.
4. O município é editado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se informações obrigatórias forem deixadas em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC19: Editar Ponto de Atenção

Descrição: O usuário edita um ponto de atenção relacionado a uma manifestação.

Papéis: Analista

Pré-condições: O usuário necessita estar logado no sistema com o perfil de analista.

Pós-condições: O ponto de atenção é editado com os novos dados inseridos.

Fluxo principal:

1. O usuário seleciona uma manifestação a partir da tabela de manifestações geradas
2. O usuário seleciona um ponto de atenção entre os relacionados a determinada manifestação.
3. O usuário modifica os campos que deseja.
4. O usuário confirma a edição do ponto de atenção.
5. O ponto de atenção é editado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (4), se alguma informação obrigatória for deixada em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (3).

UC20: Editar Questão Crítica

Descrição: O usuário edita uma questão crítica.

Papéis: Gerente

Pré-condições: O usuário necessita estar logado no sistema com o perfil de gerente.

Pós-condições: A questão crítica é editada com as novas informações inseridas.

Fluxo principal:

1. O usuário seleciona uma questão crítica a partir da tabela
2. O usuário modifica os campos desejados para a questão crítica.
3. O usuário confirma a edição da questão crítica.
4. A questão crítica é editada.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se informações obrigatórias forem deixadas em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC21: Editar Usuário

Descrição: O usuário edita um usuário no sistema.

Papéis: Administrador

Pré-condições: O usuário necessita estar logado no sistema com o perfil de administrador.

Pós-condições: O usuário é editado com os novos dados inseridos.

Fluxo principal:

1. O usuário seleciona um usuário na lista de usuários existentes.
2. O usuário edita as informações do usuário que deseja.
3. O usuário confirma a edição do usuário.
4. O usuário é editado.

Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (3), se informações obrigatórias forem deixadas em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (2).

UC22: Filtrar Mapa

Descrição: O usuário filtra o mapa das comunidades por município ou por estado.

Papéis: Gerente
Analista

Pré-condições: O usuário necessita estar logado no sistema.

Pós-condições: O mapa é filtrado a partir dos parâmetros selecionados.

Fluxo principal:

1. O usuário gera o mapa.
2. O usuário seleciona o(s) parâmetro(s) que deseja filtrar.
3. O usuário confirma a realização do filtro.
4. O mapa é filtrado.

Fluxo alternativo: Não há.

Fluxos de exceção: Não há.

UC23: Gerar Mapa

Descrição: O usuário gera o mapa das comunidades.

Papéis:	Gerente Analista
Pré-condições:	O usuário necessita estar logado no sistema.
Pós-condições:	O mapa é gerado com o farol das comunidades.
Fluxo principal:	1. O usuário seleciona o link para o mapa das comunidades. 2. O mapa é gerado.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Não há.

UC24: Gerar Relatório

Descrição:	O usuário gera um relatório com os dados inseridos no sistema.
Papéis:	Gerente
Pré-condições:	O usuário necessita estar logado no sistema com o perfil de gerente.
Pós-condições:	O relatório dos dados inseridos no sistema é gerado.
Fluxo principal:	1. O usuário seleciona a opção para geração de relatório. 2. O usuário escolhe as opções que deseja para a geração do relatório. 3. O usuário submete as opções para geração do relatório. 4. O relatório é gerado.
Fluxo alternativo:	1. O usuário seleciona uma comunidade na lista de comunidades. 2. O usuário escolhe a opção de geração de relatório para aquela comunidade. 3. O relatório daquela comunidade é gerado.
Fluxos de exceção:	Não há.

UC25: Login

Descrição:	O usuário entra no sistema.
Papéis:	Administrador Gerente Analista
Pré-condições:	O usuário necessita ter login e senha cadastrados e ativos.
Pós-condições:	O usuário entra no sistema e sua sessão é iniciada.
Fluxo principal:	1. O usuário entra no site do sistema. 2. O usuário fornece as suas informações de login e senha. 3. O usuário confirma o envio das suas informações para entrada no

sistema.
4. O usuário entra no site do sistema.
Fluxo alternativo: Não há.
Fluxos de exceção: Para o passo (3), se o usuário inserir a senha errada, retorna ao passo (2).

UC26: Logoff

Descrição: O usuário sai do sistema.
Papéis: Administrador
Gerente
Analista
Pré-condições: O usuário necessita estar logado no sistema.
Pós-condições: O usuário sai do sistema e sua sessão é finalizada.
Fluxo principal: 1. O usuário seleciona a opção de logoff.
2. O usuário confirma o logoff.
3. A sessão do usuário é finalizada.
Fluxo alternativo: 1. O usuário fecha o browser que está sendo usado para entrar no sistema.
2. A sessão do usuário é finalizada.
Fluxos de exceção: Não há.

UC27: Solicitar Acesso

Descrição: O usuário solicita acesso ao sistema, descrevendo sua função na empresa.
Papéis: Gerente
Analista
Pré-condições: O usuário necessita entrar no site do sistema.
Pós-condições: A solicitação de acesso é enviada para o administrador do sistema.
Fluxo principal: 1. O usuário entra no site do sistema.
2. O usuário seleciona a opção para solicitar acesso.
3. O usuário insere as suas informações.
4. O usuário confirma o envio da sua solicitação.
5. A solicitação é enviada.
Fluxo alternativo: Não há.

Fluxos de exceção: Para o passo (4), se informações obrigatórias forem deixadas em branco ou se os dados alterados forem de tipo diferente do esperado, retorna ao passo (3).

UC28: Visualizar Atividade

Descrição: O usuário visualiza a lista de atividades.
Papéis: Gerente
Analista
Pré-condições: O usuário necessita estar logado no sistema.
Pós-condições: A lista de atividades é gerada para visualização do usuário.
Fluxo principal: 1. O usuário seleciona o link para as atividades.
2. A lista de atividades é gerada.
Fluxo alternativo: 1. O usuário seleciona uma manifestação.
2. O usuário seleciona a opção para visualizar as atividades relacionadas a essa manifestação.
3. A lista de atividades é gerada.
Fluxos de exceção: Não há.

UC29: Visualizar Comunidade

Descrição: O usuário visualiza a lista de comunidades.
Papéis: Gerente
Analista
Pré-condições: O usuário necessita estar logado no sistema.
Pós-condições: A lista de comunidades é gerada para visualização do usuário.
Fluxo principal: 1. O usuário seleciona o link para as comunidades.
2. A lista de comunidades é gerada.
Fluxo alternativo: Não há.
Fluxos de exceção: Não há.

UC30: Visualizar Manifestação

Descrição: O usuário visualiza a lista de manifestações.
Papéis: Gerente
Analista
Pré-condições: O usuário necessita estar logado no sistema.

Pós-condições: A lista de manifestações é gerada para visualização do usuário.
Fluxo principal: 1. O usuário seleciona o link para as manifestações.
2. A lista de manifestações é gerada.
Fluxo alternativo: Não há.
Fluxos de exceção: Não há.

UC31: Visualizar Manifestante

Descrição: O usuário visualiza a lista de manifestantes.
Papéis: Gerente
Analista
Pré-condições: O usuário necessita estar logado no sistema.
Pós-condições: A lista de manifestantes é gerada para visualização do usuário.
Fluxo principal: 1. O usuário seleciona o link para os manifestantes.
2. A lista de manifestantes é gerada.
Fluxo alternativo: Não há.
Fluxos de exceção: Não há.

UC32: Visualizar Ponto de Atenção

Descrição: O usuário visualiza a lista de pontos de atenção.
Papéis: Gerente
Analista
Pré-condições: O usuário necessita estar logado no sistema.
Pós-condições: A lista de pontos de atenção é gerada para visualização do usuário.
Fluxo principal: 1. O usuário seleciona o link para os pontos de atenção.
2. A lista de pontos de atenção é gerada.
Fluxo alternativo: 1. O usuário seleciona uma manifestação.
2. O usuário seleciona a opção para visualizar os pontos de atenção relacionados a essa manifestação.
3. A lista de pontos de atenção é gerada.
Fluxos de exceção: Não há.

UC33: Visualizar Questão Crítica

Descrição: O usuário visualiza a lista de questões críticas.
Papéis: Gerente

	Analista
Pré-condições:	O usuário necessita estar logado no sistema.
Pós-condições:	A lista de questões críticas é gerada para visualização do usuário.
Fluxo principal:	1. O usuário seleciona o link para as questões críticas. 2. A lista de questões críticas é gerada.
Fluxo alternativo:	Não há.
Fluxos de exceção:	Não há.

2.3.2.2. Diagrama de Classes

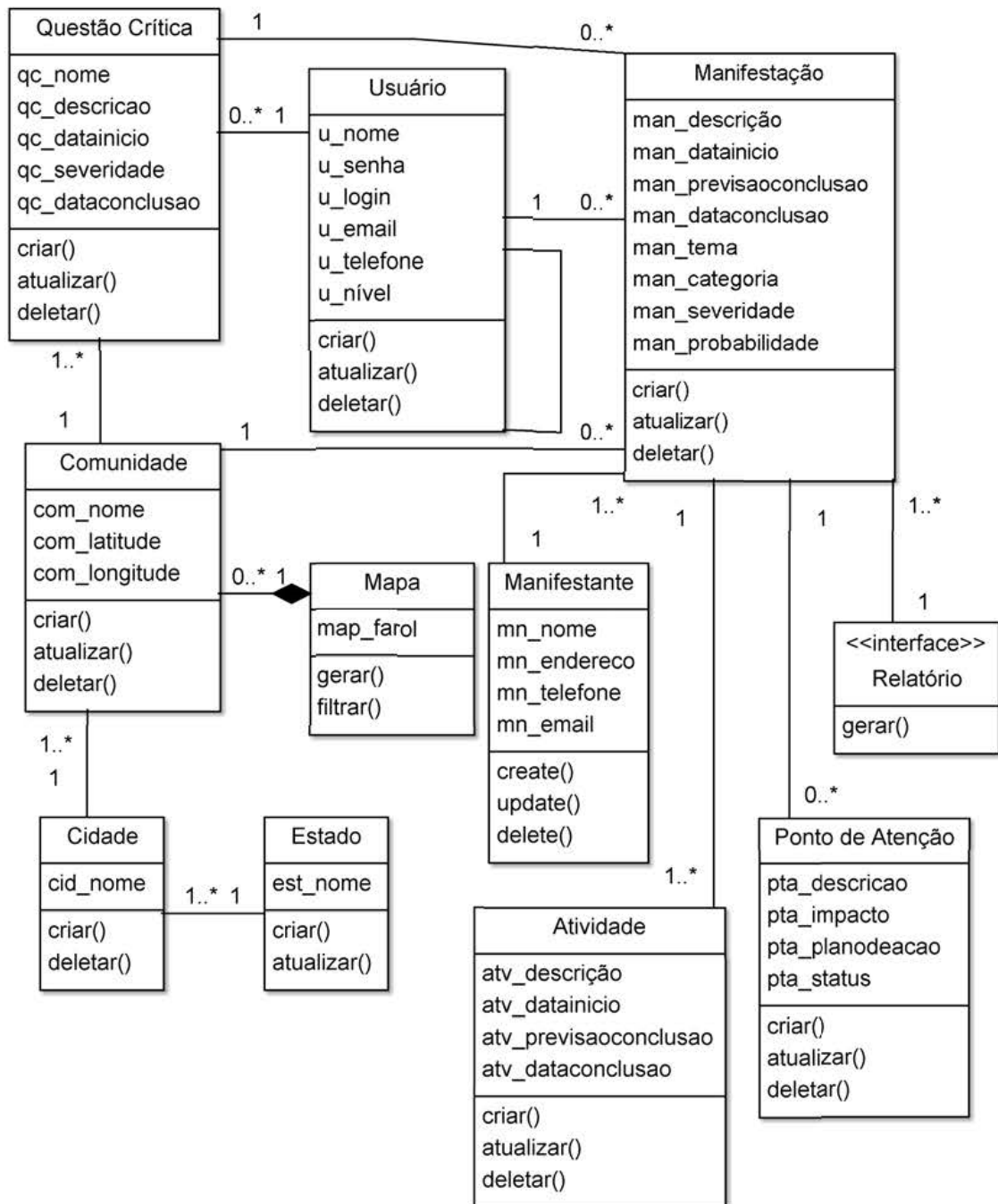


Figura 2 – Diagrama de Classes

2.3.2.2.1. Classe Questão Crítica:

- ✓ qc_nome – nome que identifica a questão crítica
- ✓ qc_descricao – descrição da questão crítica
- ✓ qc_datainicio – data em que a questão crítica foi iniciada
- ✓ qc_severidade – severidade da questão crítica
- ✓ qc_dataconclusao – data em que a questão crítica foi finalizada

2.3.2.2.2. Manifestação:

- ✓ man_descricao – descrição da manifestação
- ✓ man_datainicio – data em que a manifestação foi iniciada
- ✓ man_previsaoconclusao – data prevista para a conclusão da manifestação
- ✓ man_dataconclusao – data em que a manifestação foi finalizada
- ✓ man_tema – tema em que a manifestação está inserida
- ✓ man_categoria – categoria em que a manifestação está inserida
- ✓ man_severidade – severidade da manifestação
- ✓ man_probabilidade – probabilidade da manifestação ocorrer

2.3.2.2.3. Ponto de Atenção:

- ✓ pta_descricao – descrição do ponto de atenção
- ✓ pta_impacto – descrição do possível impacto do ponto de atenção
- ✓ pta_planodeacao – descrição do plano de ação para resolução do ponto de atenção
- ✓ pta_status – status da resolução do ponto de atenção

2.3.2.2.4. Atividade:

- ✓ atv_descricao – descrição da atividade
- ✓ atv_datainicio – data em que a atividade foi iniciada
- ✓ atv_previsaoconclusao – data prevista para a conclusão da atividade
- ✓ atv_dataconclusao – data em que a atividade foi finalizada

2.3.2.2.5. Manifestante:

- ✓ mn_nome – nome do manifestante
- ✓ mn_endereco – endereço do manifestante
- ✓ mn_telefone – telefone do manifestante
- ✓ mn_email – e-mail do manifestante

2.3.2.2.6. Usuário:

- ✓ u_nome – nome do usuário
- ✓ u_senha – senha do usuário
- ✓ u_login – login do usuário
- ✓ u_email – e-mail do usuário
- ✓ u_telefone - telefone do usuário
- ✓ u_nível – nível do usuário dentro da empresa

2.3.2.2.7. Comunidade:

- ✓ com_nome – nome da comunidade
- ✓ com_latidade – latitude da comunidade
- ✓ com_longitude – longitude da comunidade

2.3.2.2.8. Cidade:

- ✓ cid_nome – nome da cidade

2.3.2.2.9. Estado:

- ✓ est_nome – nome do estado

2.3.2.2.10. Mapa:

- ✓ map_farol – farol que identifica a criticidade da comunidade

2.3.3. Requisitos de Desempenho

O tempo de resposta do sistema é essencial para seu uso. Por ser uma ferramenta de controle e geração de relatórios, onde os usuários relatarão as ocorrências do seu dia a dia de trabalho, sendo que estes possuem uma enorme

carga de trabalho sem considerar o uso do site, para que haja um engajamento na utilização, o tempo de resposta deverá ser mínimo.

A quantidade de usuários máxima do sistema deverá ser em torno de 150, que vem a ser o total de funcionários analistas que o site terá no momento da efetivação do seu uso.

2.3.4. Restrições de Projeto

A tecnologia empregada será a linguagem PHP associado a uma base de dados armazenada em um sistema gerenciador de banco de dados SQL Server e um servidor Apache. Por ser um ambiente Web e rodar em um servidor de grande capacidade, a única restrição aplicada ao sistema deverá ser que o usuário tenha acesso a internet.

2.3.5. Atributos

Os seguintes atributos de qualidade podem ser atribuídos ao projeto:

- ➔ Amigabilidade – Por ser um site que será amplamente usado por uma equipe que poderá não possuir grandes conhecimentos técnicos em relação a sistemas computacionais, o ambiente de navegação do sistema deverá ser de fácil utilização e possuir uma interface amigável.
- ➔ Segurança – A segurança da informação do sistema deve ser um atributo essencial, pois a ferramenta guardará dados extremamente confidenciais sobre manifestações da comunidade e as ações efetivas da empresa para a resolução das mesmas. Além desses dados, também serão armazenadas informações sobre os funcionários da empresa que utilizam o sistema.
- ➔ Portabilidade – Sendo um sistema Web, não apresentará dependência de um sistema operacional. Também deverá ser capaz de rodar sem problema em qualquer dos browsers utilizados atualmente.
- ➔ Manutenível – O sistema projetado tem essa característica devido a possuir uma especificação de requisitos bastante detalhada. Além disso, a codificação foi realizada com muitos comentários, afim de facilitar a correção de erros e a expansão, que será de crucial

importância devido ao fato de que sempre é necessário atribuir novas funções para o melhor funcionamento de um sistema WEB, e de que a ferramenta poderá ter novos processos incluídos no futuro.

2.3.6. Outros Requisitos

Não há.

3. Projeto do Software

Para a implementação do sistema, foi utilizada a linguagem PHP, com um base de dados para armazenar as informações inseridas pelos usuários. Para a realização de algumas validações nos formulários a serem preenchidos pelos usuários, foi aplicado Java Script, facilitando essas autenticações, pois com esse tipo de código, a validação é realizada diretamente na máquina do usuário, e as informações só são levadas ao servidor quando validadas, evitando tráfego e utilização desnecessária do servidor.

3.1. Arquitetura

O sistema possui uma arquitetura cliente/servidor, onde a grande maioria das iterações do software ocorre no servidor. As camadas dessa arquitetura podem ser observadas na figura a seguir:

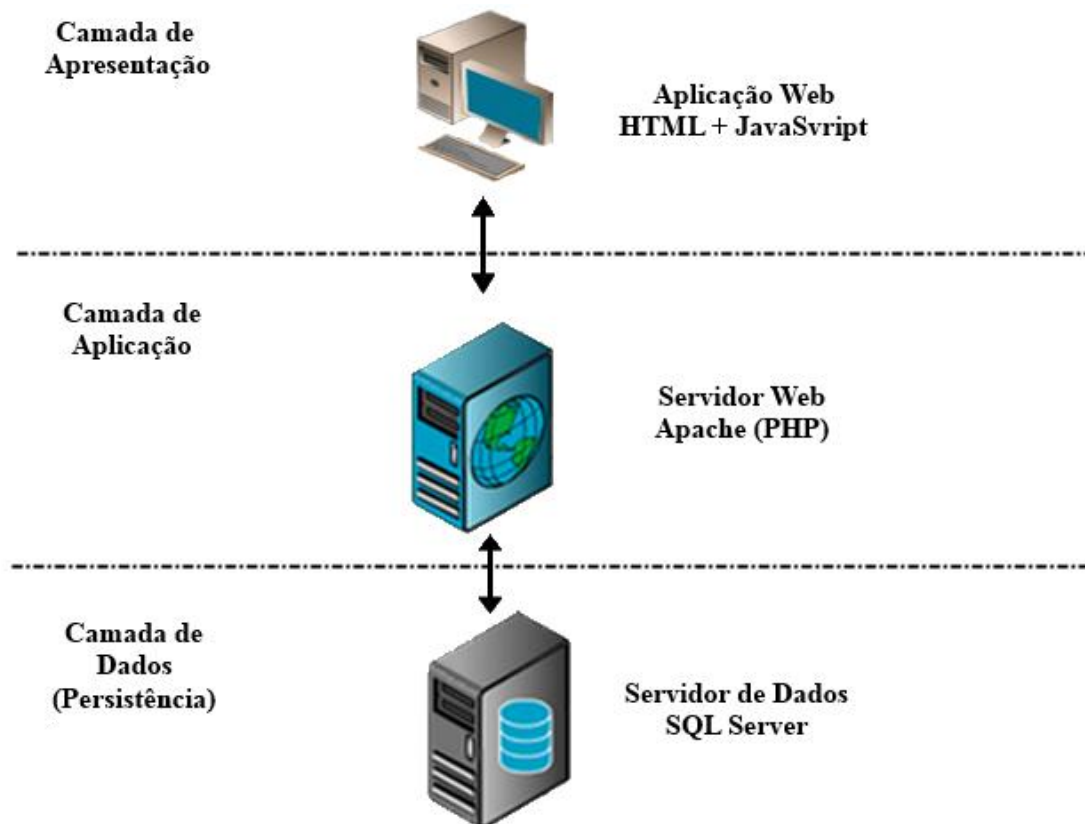


Figura 3 – Arquitetura do sistema

A camada de aplicação apresentará as páginas da ferramenta em HTML, possuindo códigos para validação em Java Script. As páginas serão geradas após a interpretação do código PHP pelo servidor Apache. A camada de dados trará o servidor de dados, que utilizará o sistema gerenciador de banco de dados SQL Server.

3.2. Modelo de Dados

3.2.1. Diagrama de Entidades e Relacionamentos

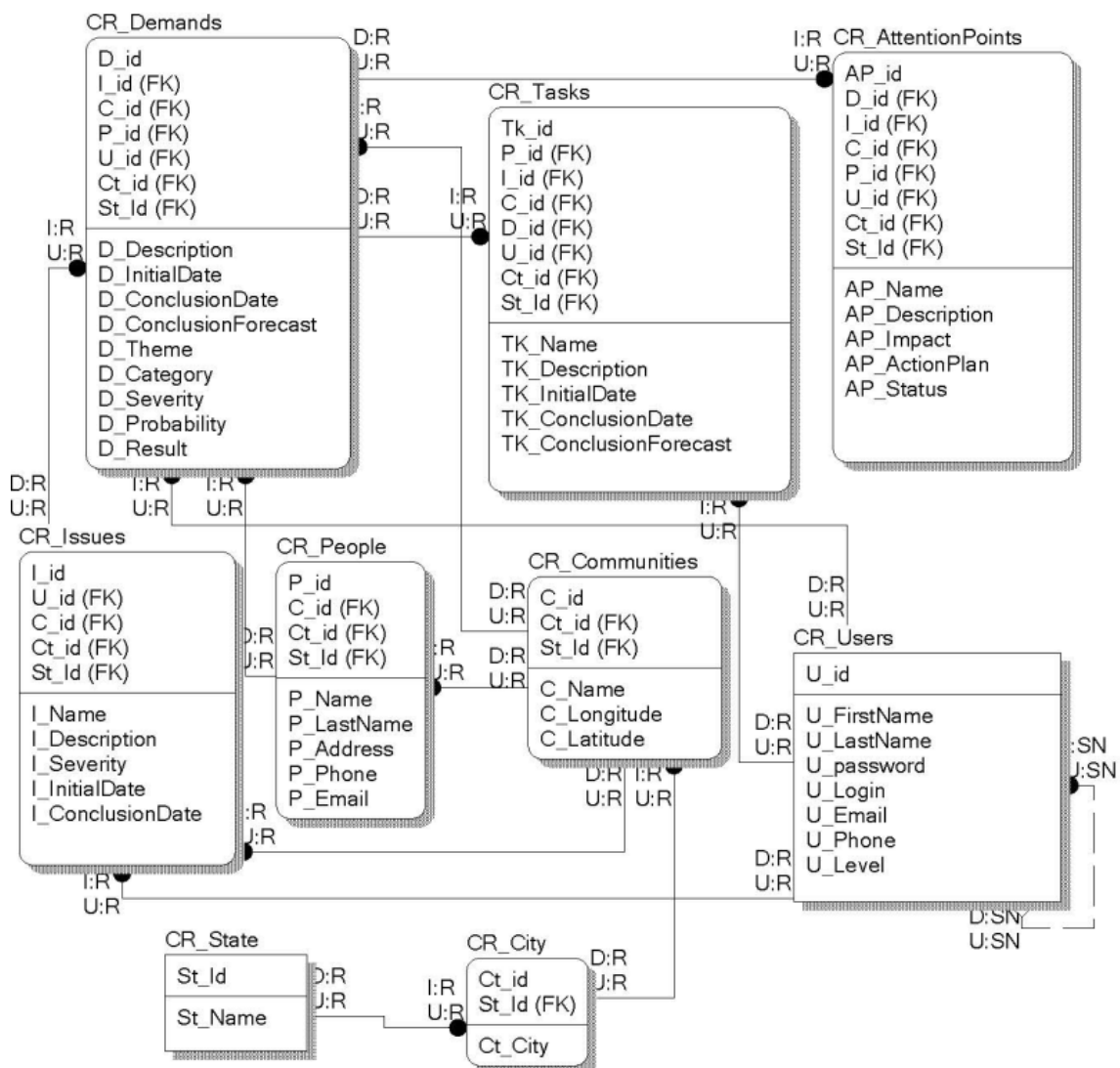


Figura 4 – Diagrama de Entidades e Relacionamentos

Os códigos em SQL para a criação das tabelas especificadas acima se encontram no apêndice desse projeto.

3.2.2. Dicionário de Dados

3.2.2.1. Tabela CR_Demands

Tabela que armazena as manifestações recebidas pelos analistas por parte das comunidades.

Atributo	Descrição
D_Id	Armazena o identificador da manifestação
D_Description	Armazena a descrição da manifestação
D_InitialDate	Armazena a data em que a manifestação foi criada
D_ConclusionDate	Armazena a data em que a manifestação foi concluída
D_Theme	Armazena o tema da manifestação
D_Category	Armazena a categoria da manifestação
D_Severity	Armazena a severidade da manifestação
D_Probability	Armazena a probabilidade da manifestação
D_Result	Armazena a descrição do resultado da manifestação
C_Id	Armazena o número de identificação da comunidade atrelada a manifestação
U_Id	Armazena o número de identificação do usuário atrelado a manifestação
P_Id	Armazena o número de identificação do manifestante atrelado a manifestação
I_Id	Armazena o número de identificação da questão crítica atrelada a manifestação

Tabela 1 – Dicionário de Dados: CR_Demands

3.2.2.2. Tabela CR_Issues

Tabela que armazena as questões críticas identificadas nas comunidades pelos gerentes.

Atributo	Descrição
----------	-----------

I_Id	Armazena o identificador da manifestação
I_Name	Armazena o nome que identifica a questão crítica
I_Description	Armazena a descrição da questão crítica
I_Severity	Armazena a severidade da questão crítica
I_InitialDate	Armazena a data em que a questão crítica foi criada
I_ConclusionDate	Armazena a data em que a questão crítica foi concluída
C_Id	Armazena o número de identificação da comunidade atrelada a questão crítica
U_Id	Armazena o número de identificação do usuário atrelado a questão crítica

Tabela 2 – Dicionário de Dados: CR_Issues

3.2.2.3. Tabela CR_People

Tabela que armazena os manifestantes responsáveis por gerar as manifestações a serem inseridas no sistema pelos analistas.

Atributo	Descrição
P_Id	Armazena o identificador do manifestante
P_FirstName	Armazena o primeiro nome do manifestante
P_LastName	Armazena o último nome do manifestante
P_Address	Armazena o endereço do manifestante
P_Email	Armazena o endereço do manifestante
P_Phone	Armazena o endereço do manifestante
C_Id	Armazena o número de identificação da comunidade atrelada ao manifestante

Tabela 3 – Dicionário de Dados: CR_People

3.2.2.4. Tabela CR_Communities

Tabela que armazena as comunidades de onde as manifestações podem surgir. Os gerentes são os responsáveis pela identificação das comunidades que podem ter relação com as operações da empresa.

C_Id	Armazena o identificador da comunidade
------	----------------------------------------

C_Name	Armazena o nome da comunidade
C_Longitude	Armazena a longitude da comunidade
C_Latitude	Armazena a latitude da comunidade
Ct_Id	Armazena o número de identificação da cidade atrelada a comunidade

Tabela 4 – Dicionário de Dados: CR_Communities

3.2.2.5. Tabela CR_City

Tabela que armazena as cidades onde existem operações da empresa ou que fazem fronteira com essas operações.

Atributo	Descrição
Ct_Id	Armazena o identificador da cidade
Ct_City	Armazena o nome da cidade
St_Id	Armazena o número de identificação do estado atrelado a cidade

Tabela 5 – Dicionário de Dados: CR_City

3.2.2.6. Tabela CR_State

Tabela que armazena os estados onde existem operações da empresa ou que tenham alguma relação com essas operações.

Atributo	Descrição
St_Id	Armazena o identificador do estado
St_State	Armazena o nome do estado

Tabela 6 – Dicionário de Dados: CR_State

3.2.2.7. Tabela CR_Users

Tabela que armazena os usuários do sistema, que são os funcionários da área responsável pela gestão das relações com as comunidades, bem como os administradores do sistema.

Atributo	Descrição
U_Id	Armazena o identificador do usuário
U_FirstName	Armazena o primeiro nome do usuário

U_LastName	Armazena o último nome do usuário
U_Login	Armazena o login do usuário
U_Password	Armazena a senha criptografada do usuário
U_Email	Armazena o endereço de e-mail do usuário
U_Phone	Armazena o telefone do usuário
U_Level	Armazena o nível do usuário, podendo ter os seguintes valores: 1- Analista 2- Gerente 5- Administrador
U_Id_Manager	Armazena o número de identificação do usuário gerente atrelado ao usuário analista

Tabela 7 – Dicionário de Dados: CR_Users

3.2.2.8. Tabela CR_Tasks

Tabela que armazena as atividades a serem realizadas para a resolução de uma determinada manifestação.

Atributo	Descrição
Tk_Id	Armazena o identificador da atividade
Tk_Name	Armazena o nome que identifica a atividade
Tk_Description	Armazena a descrição da atividade
Tk_InitialDate	Armazena a data em que a atividade foi criada
Tk_ConclusionForecast	Armazena a data de previsão para a atividade ser concluída
Tk_ConclusionDate	Armazena a data em que a atividade foi concluída
U_Id	Armazena o número de identificação do usuário atrelado a atividade
D_Id	Armazena o número de identificação da manifestação atrelada a atividade

Tabela 8 – Dicionário de Dados: CR_Tasks

3.2.2.9. Tabela CR_AttentionPoints

Tabela que armazena os pontos de atenção atrelados a alguma manifestação existente no sistema.

Atributo	Descrição
Ap_Id	Armazena o identificador do ponto de atenção
Ap_Name	Armazena o nome que identifica o ponto de atenção
Ap_Description	Armazena a descrição do ponto de atenção
Ap_ConclusionDate	Armazena a data em que o ponto de atenção foi concluído
Ap_ActionPlan	Armazena o plano de ação para resolução do ponto de atenção
Ap_Status	Armazena o status do ponto de atenção
U_Id	Armazena o número de identificação do usuário atrelado ao ponto de atenção
D_Id	Armazena o número de identificação da manifestação atrelada ao ponto de atenção

Tabela 9 – Dicionário de Dados: CR_AttentionPoints

3.3. Implementação do sistema

A implementação do sistema foi realizada utilizando a linguagem PHP para rodar as suas funcionalidades. Foi utilizado Java Script para as validações necessárias dos dados inseridos pelos usuários. Os códigos para a operação das funções do produto, bem como os layouts das telas da ferramenta desenvolvida na versão alpha, se encontram no Apêndice dessa monografia.

3.4. Ferramentas utilizadas

Para a construção do sistema, foram utilizadas algumas ferramentas de apoio para viabilizar a implementação. A descrição dessas ferramentas se encontra a seguir.

- Apache 2.2 – servidor web utilizado para comunicação da máquina que continha o sistema com a Internet. O servidor também foi utilizado para a compilação dos códigos em PHP da ferramenta.
- SQL Server2008 R2– sistema gerenciador de banco de dados, onde toda a base de dados do sistema foi implantada e será armazenada.
- Adobe DreamweaverCS5.5 – aplicação para desenvolvimento web, foi utilizada para a criação das páginas web dos sistema.
- ArgoUML – aplicação para diagramação UML, utilizada para a construção

dos diagramas apresentados no projeto.

- Erwin Data Modeler 7.3 – aplicativo para modelagem de dados, foi utilizado para a construção do modelo de entidades e relacionamentos.

4. Testes e Resultados

Nesta seção serão apresentados os testes realizados na versão alpha do sistema, bem como os resultados obtidos após a execução destes, expondo a comparação entre eles e os resultados esperados.

4.1. Cadastramento de usuários

Ação	Resultado Esperado	Resultado Apresentado
Informação de login de usuário já cadastrado	Exibir de janela com a informação de que não foi possível cadastrar o usuário, pois o login do usuário já está cadastrado, e retornar a tela de cadastramento do usuário.	A janela foi exibida e retornou a tela de cadastramento do usuário.
Informação de login de usuário não cadastrado	Exibir tela de cadastro realizado e incluir novo usuário na lista.	A tela foi exibida e o novo usuário foi incluído na lista de usuários.

Tabela 10 – Testes e resultados: Cadastramento de usuários

4.2. Cadastramento de estados e municípios

Ação	Resultado Esperado	Resultado Apresentado
Informar o nome de um município já cadastrado, para um mesmo estado	Exibir janela dizendo que já existe o município cadastrado e voltar a tela de cadastramento de município.	A janela foi exibida e retornou para tela de cadastro.
Informar o nome de um município novo	Exibir tela de sucesso e incluir município na lista.	A tela esperada foi exibida e o município incluído na lista.
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de município.	A janela foi exibida e permaneceu na tela de cadastro.
Informar um estado já existente	Exibir janela dizendo que já existe o estado cadastrado e voltar a tela de cadastramento de estado	A janela foi exibida e retornou para tela de cadastro.
Informar um novo estado	Exibir tela de sucesso e incluir estado na lista	A tela esperada foi exibida e o estado incluído na lista.

Tabela 11 – Testes e resultados: Cadastramento de estados e municípios

4.3. Cadastramento de questão crítica

Ação	Resultado Esperado	Resultado Apresentado
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de questão crítica	A janela foi exibida e permaneceu na tela de cadastro.
Informar dados inválidos	Exibir janela de quais campos contém dados inválidos e permanecer na tela de cadastro de questão crítica	A janela foi exibida e permaneceu na tela de cadastro.
Informar todos os dados obrigatórios e somente dados válidos	Exibir tela de sucesso e incluir questão crítica na lista.	A tela esperada foi exibida e a questão crítica incluída na lista.

Tabela 12 – Testes e resultados: Cadastramento de questão crítica

4.4. Cadastramento de comunidades

Ação	Resultado Esperado	Resultado Apresentado
Informar o nome de uma comunidade já cadastrada, para um mesmo município	Exibir janela dizendo que já existe a comunidade cadastrada e voltar a tela de cadastramento de comunidade.	A janela foi exibida e retornou para tela de cadastro.
Informar o nome de uma comunidade nova	Exibir tela de sucesso e incluir comunidade na lista.	A tela esperada foi exibida e a comunidade incluída na lista.
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de comunidade.	A janela foi exibida e permaneceu na tela de cadastro.

Tabela 13 – Testes e resultados: Cadastramento de comunidades

4.5. Cadastramento de manifestações

Ação	Resultado Esperado	Resultado Apresentado
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de manifestação	A janela foi exibida e permaneceu na tela de cadastro.
Informar dados inválidos	Exibir janela de quais campos contém dados inválidos e permanecer na tela de cadastro de manifestação	A janela foi exibida e permaneceu na tela de cadastro.
Informar todos os dados obrigatórios e somente dados válidos	Exibir tela de sucesso e incluir manifestação na lista.	A tela esperada foi exibida e a manifestação incluída na lista.

Tabela 14 – Testes e resultados: Cadastramento de manifestações

4.6. Cadastramento de manifestantes

Ação	Resultado Esperado	Resultado Apresentado
Informar o nome de uma comunidade já cadastrada, para um mesmo município	Exibir janela dizendo que já existe a comunidade cadastrada e voltar a tela de cadastramento de comunidade.	A janela foi exibida e retornou para tela de cadastro.
Informar o nome de uma comunidade nova	Exibir tela de sucesso e incluir comunidade na lista.	A tela esperada foi exibida e a comunidade incluída na lista.
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de comunidade.	A janela foi exibida e permaneceu na tela de cadastro.

Tabela 15 – Testes e resultados: Cadastramento de manifestantes

4.7. Cadastramento de atividades

Ação	Resultado Esperado	Resultado Apresentado
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de atividade	A janela foi exibida e permaneceu na tela de cadastro.
Informar dados inválidos	Exibir janela de quais campos contém dados inválidos e permanecer na tela de cadastro de atividade	A janela foi exibida e permaneceu na tela de cadastro.
Informar todos os dados obrigatórios e somente dados válidos	Exibir tela de sucesso e incluir atividade na lista.	A tela esperada foi exibida e a atividade incluída na lista.

Tabela 16 – Testes e resultados: Cadastramento de atividades

4.8. Cadastramento de pontos de atenção

Ação	Resultado Esperado	Resultado Apresentado
Informar somente parte dos dados obrigatórios	Exibir janela de quais dados obrigatórios não foram preenchidos e permanecer na tela de cadastro de ponto de atenção	A janela foi exibida e permaneceu na tela de cadastro.
Informar dados inválidos	Exibir janela de quais campos contém dados inválidos e permanecer na tela de cadastro de ponto de atenção	A janela foi exibida e permaneceu na tela de cadastro.
Informar todos os dados obrigatórios e somente dados válidos	Exibir tela de sucesso e incluir ponto de atenção na lista.	A tela esperada foi exibida e a ponto de atenção incluída na lista.

4.9. Visualização de questão crítica

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para visualização da lista de questões críticas	Visualizar lista com as questões críticas em aberto. No caso de não haver questões em aberto, mostrar mensagem de que não há questão em aberto	A lista foi gerada e somente as questões em aberto foram mostradas. No caso em que não existiam questões em aberto, foi exibida mensagem de que não existiam questões em aberto.
Escolha de parâmetros que possuem alguma questão crítica	Visualizar lista com as questões críticas de acordo com os parâmetros selecionados	A lista foi gerada e somente as questões de acordo com os parâmetros foram mostradas
Escolha de parâmetros que não possuem alguma questão crítica	Mensagem que não existem questões críticas com os parâmetros selecionados	Foi mostrada mensagem de que não existiam questões críticas de acordo com os parâmetros escolhidos

Tabela 18 – Testes e resultados: Visualização de questão crítica

4.10. Visualização de comunidades

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para visualização da lista de comunidades	Visualizar lista com as comunidades. No caso de não haver comunidades cadastradas, mostrar mensagem de que não há comunidade cadastrada	A lista foi gerada. No caso em que não existiam comunidades cadastradas, foi exibida mensagem de que não existiam comunidade cadastrada
Escolha de parâmetros que possuem alguma comunidade	Visualizar lista com as comunidades de acordo com os parâmetros selecionados	A lista foi gerada e somente as comunidades de acordo com os parâmetros foram mostradas
Escolha de parâmetros que não possuem alguma comunidade	Mensagem que não existem comunidades com os parâmetros selecionados	Foi mostrada mensagem de que não existiam comunidades de acordo com os parâmetros escolhidos

Tabela 19 – Testes e resultados: Visualização de comunidades

4.11. Visualização de manifestações

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para	Visualizar lista com as	A lista foi gerada e somente

visualização da lista de manifestações	manifestações em aberto. No caso de não haver manifestações em aberto, mostrar mensagem de que não há manifestação em aberto.	as manifestações em aberto foram mostradas. No caso em que não existiam manifestações em aberto, foi exibida mensagem de que não existiam manifestações em aberto.
Escolha de parâmetros que possuem alguma manifestação	Visualizar lista com as manifestações de acordo com os parâmetros selecionados	A lista foi gerada e somente as manifestações de acordo com os parâmetros foram mostradas
Escolha de parâmetros que não possuem alguma manifestação	Mensagem que não existem manifestações com os parâmetros selecionados	Foi mostrada mensagem de que não existiam manifestações de acordo com os parâmetros escolhidos

Tabela 20 – Testes e resultados: Visualização de manifestações

4.12. Visualização de manifestantes

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para visualização da lista de manifestantes	Visualizar lista com os manifestantes. No caso de não haver manifestantes cadastrados, mostrar mensagem de que não há manifestante cadastrado	A lista foi gerada. No caso em que não existiam manifestantes cadastrados, foi exibida mensagem de que não existiam manifestante cadastrado
Escolha de parâmetros que possuem algum manifestante	Visualizar lista com os manifestantes de acordo com os parâmetros selecionados	A lista foi gerada e somente os manifestantes de acordo com os parâmetros foram mostrados
Escolha de parâmetros que não possuem algum manifestante	Mensagem que não existem manifestantes com os parâmetros selecionados	Foi mostrada mensagem de que não existiam manifestantes de acordo com os parâmetros escolhidos

Tabela 21 – Testes e resultados: Visualização de manifestantes

4.13. Visualização de atividades

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para visualização da lista de atividades	Visualizar lista com as atividades em aberto. No caso de não haver atividades em aberto, mostrar mensagem de que não há atividade em aberto.	A lista foi gerada e somente as atividades em aberto foram mostradas. No caso em que não existiam atividades em aberto, foi exibida mensagem de que não existiam atividades em aberto.

Escolha de parâmetros que possuem alguma atividade	Visualizar lista com as atividades de acordo com os parâmetros selecionados	A lista foi gerada e somente as atividades de acordo com os parâmetros foram mostradas
Escolha de parâmetros que não possuem alguma atividade	Mensagem que não existem atividades com os parâmetros selecionados	Foi mostrada mensagem de que não existiam atividades de acordo com os parâmetros escolhidos

Tabela 22 – Testes e resultados: Visualização de atividades

4.14. Visualização de pontos de atenção

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para visualização da lista de pontos de atenção	Visualizar lista com os pontos de atenção em aberto. No caso de não haver pontos de atenção em aberto, mostrar mensagem de que não há ponto de atenção em aberto.	A lista foi gerada e somente os pontos de atenção em aberto foram mostrados. No caso em que não existiam pontos de atenção em aberto, foi exibida mensagem de que não existiam pontos de atenção em aberto.
Escolha de parâmetros que possuem algum ponto de atenção	Visualizar lista com os pontos de atenção de acordo com os parâmetros selecionados	A lista foi gerada e somente os pontos de atenção de acordo com os parâmetros foram mostrados
Escolha de parâmetros que não possuem algum ponto de atenção	Mensagem que não existem pontos de atenção com os parâmetros selecionados	Foi mostrada mensagem de que não existiam pontos de atenção de acordo com os parâmetros escolhidos

Tabela 23 – Testes e resultados: Visualização de ponto de atenção

4.15. Geração de mapa

Ação	Resultado Esperado	Resultado Apresentado
Solicitação para geração do mapa	Visualizar mapa com os faróis das comunidades de acordo com os cálculos e critérios definidos	O mapa foi gerado, mostrado na tela, com as comunidades identificadas pelos seus respectivos faróis
Escolha de parâmetros que possuem alguma comunidade	Visualizar mapa com os faróis das comunidades que se enquadrem nos parâmetros selecionados, de acordo com os cálculos e critérios definidos	O mapa foi gerado, mostrado na tela, com as comunidades que obedeciam os parâmetros selecionadas, sendo identificadas pelos seus respectivos faróis
Escolha de parâmetros que não possuem	Visualizar o mapa sem nenhuma comunidade aparecendo e	A janela foi exibida, e o mapa foi gerado, mostrado

alguma comunidade	aparecer janela exibindo que não há nenhuma comunidade de acordo com o critério	na tela, sem comunidades.
-------------------	---------------------------------------------------------------------------------	---------------------------

Tabela 24 – Testes e resultados: Geração de mapa

4.16. Geração de relatório

Ação	Resultado Esperado	Resultado Apresentado
Escolha de parâmetros que possuem manifestações cadastradas	Gerar relatório e exibi-lo na tela, de acordo com os parâmetros selecionados	O relatório foi exibido na tela de acordo com os parâmetros selecionados
Escolha de parâmetros que não possuem alguma manifestações cadastradas	Exibir na tela que não há informações de acordo com os parâmetros selecionados	A tela foi exibida, informando não haver informações para os parâmetros selecionados

Tabela 25 – Testes e resultados: Geração de relatório

4.17. Login e logout

Ação	Resultado Esperado	Resultado Apresentado
Login de usuário inexistente	Exibição de janela de usuário não cadastrado e retorno a tela de login	A janela foi exibida e retornou para tela de login
Login de usuário existente com senha incorreta	Exibição de janela de senha incorreta e retorno a tela de login	A janela foi exibida e retornou para tela de login
Login de usuário administrador e senha correta	Iniciar sessão e ir para tela inicial de administrador	Sessão iniciada e exibição da tela inicial de administrador
Login de usuário gerente e senha correta	Iniciar sessão e ir para tela inicial de gerente	Sessão iniciada e exibição da tela inicial de gerente
Login de usuário analista e senha correta	Iniciar sessão e ir para tela inicial de analista	Sessão iniciada e exibição da tela inicial de analista

Tabela 26 – Testes e resultados: Login e logout

5. Conclusões

Com a experiência e o aprendizado adquiridos ao longo do curso, aplicando conceitos de engenharia de software, e somando-se a isso a experiência acumulada na realização de projetos em uma empresa de consultoria, foi possível a realização desse trabalho, que cumpriu os objetivos a que se propunha.

O desenvolvimento dos códigos do sistema não apresentaram problemas, pelo fato de PHP ser de fácil aprendizagem e ter suporte a orientação a objeto. A utilização de java script também foi de simples aprendizagem e implementação.

Quanto às ferramentas utilizadas para elaboração do projeto, a utilização do SQL Server trouxe um esforço adicional ao que seria empregado com o uso do MySQL, devido ao fato de que a linguagem PHP possui uma integração nativa a essa ferramenta, enquanto que usando o SQL Server foi necessário fazer um desenvolvimento mais trabalhoso para obter os mesmos resultados que essa integração fornece. Já o uso do ArgoUML foi fácil, sem maiores complicações, assim como o uso do Erwin, que já havia sido utilizado em trabalhos anteriores. O desenvolvimento a partir do Dreamweaver não apresentou dificuldades, por ser uma ferramenta bastante intuitiva e de fácil aplicação.

Para o futuro, há a possibilidade de ampliação das funcionalidades desse sistema, com a inclusão de um painel com os custos das ações realizadas pela empresa na resolução de manifestações e de possíveis custos evitados com a tomada de decisões prévias a partir dessas ações, agregando um maior valor ao projeto, tornando esse valor mensurável. Também podem ser incluídos novos módulos a fim de que seja possível monitorar mais ações da empresa em relação às suas comunidades adjacentes, como um controle de todas as ações realizadas pela empresa nessas comunidades de forma voluntária.

6. Bibliografia

- ALECRIM, Emerson. Conhecendo o Servidor Apache.2006. Disponível em: <http://www.infowester.com/servapach.php>. Acessado em: 15 dez. 2012.
- COAR, Ken; BOWEN, Rich. Apache Guia Prático. Rio de Janeiro: Alta Books, 2008. 272 p.
- FERREIRA, Felipe. Linguagem PHP – Vantagens e Aplicações. 2010. Disponível em: <http://www.artigonal.com/internet-artigos/linguagem-php-vantagens-e-aplicacoes-1788643.html>. Acessado em: 10 dez. 2012.
- FLANAGAN, David. JavaScript: O Guia Definitivo. 4 ed. São Paulo: Artmed, 2002. 816 p.
- MELO, Alexandro Altair; G.F. NASCIMENTO, Mauricio. PHP Profissional. São Paulo: Novatec, 2007. 464 p.
- PHP GROUP. PHP. 2010. Disponível em: <http://php.net/>. Acessado em: 20 jan. 2013.
- WORLD WIDE WEB CONSORTIUM. Extensible Markup Language (XML). Disponível em: <http://www.w3.org/XML/>. Acessado em: 21 jan. 2013.
- GOOGLE. Google Maps JavaScript API v3. Disponível em: <https://developers.google.com/maps/documentation/javascript/overlays>. Acessado em: 21 jan. 2013.
- GOOGLE. Google Charts. Disponível em: https://google-developers.appspot.com/chart/interactive/docs/basic_customizing_chart. Acessado em: 20 abr. 2013.
- MACORATTI, José Carlos. UML - Diagrama de Classes e objetos. Disponível em: http://www.macoratti.net/net_uml1.htm. Acessado em: 10 fev. 2013.
- FIGUEIREDO, Marcos Leandro; SOARES, Hélio Rubens. Comparação entre a modelagem orientada a objeto e a modelagem estruturada relacional. Disponível em: <http://online.unicaldas.edu.br/portal/publicacao/arq/modelagem.pdf>. Acessado em: 1 dez. 2012.
- ARGOURL. Homepage Online. Disponível em: <http://argouml.tigris.org/>. Acessado em: 12 fev. 2013.

Apêndice A

Criação de tabelas no SQL Server

A seguir são apresentados os códigos utilizados para criar as tabelas do projeto em SQL.

- TabelaCR_Demands

```
createtable CR_Demands
(
  D_Id intPRIMARYKEYIDENTITY,
  D_Name varchar(255)NOTNULL,
  D_Description varchar(800)NOTNULL,
  D_InitialDate dateNOTNULL,
  D_ConclusionForecast dateNOTNULL,
  D_ConclusionDate date,
  D_Theme varchar(255)NOTNULL,
  D_Category varchar(255)NOTNULL,
  D_Severity varchar(255)NOTNULL,
  D_Probability decimal(3,2)NOTNULL,
  D_Result varchar(1024),
  C_Id intFOREIGNKEYREFERENCES CR_Communities(C_Id),
  U_Id intFOREIGNKEYREFERENCES CR_Users(U_Id),
  P_Id intFOREIGNKEYREFERENCES CR_People(P_Id),
  I_Id intFOREIGNKEYREFERENCES CR_Issues(I_Id)
)
```

- TabelaCR_Issues

```
createtable CR_Issues
(
  I_Id intPRIMARYKEYIDENTITY,
  I_Name varchar(255)NOTNULL,
  I_Description varchar(255)NOTNULL,
  I_Severity varchar(255)NOTNULL,
  I_InitialDate dateNOTNULL,
  I_ConclusionDate date,
  C_Id intFOREIGNKEYREFERENCES CR_Communities(C_Id),
  U_Id intFOREIGNKEYREFERENCES CR_Users(U_Id)
)
```

- TabelaCR_People

```
createtable CR_People
(
  P_Id intPRIMARYKEYIDENTITY,
  P_FirstName varchar(255)NOTNULL,
  P_LastName varchar(255)NOTNULL,
  P_Address varchar(255)NOTNULL,
  P_Email varchar(255)NOTNULL,
  P_Phone varchar(255)NOTNULL,
)
```

```
C_Id intFOREIGNKEYREFERENCES CR_Communities(C_Id)
)
```

- TabelaCR_Communities

```
createtable CR_Communities
(
C_Id intPRIMARYKEYIDENTITY,
C_Name varchar(255)NOTNULL,
C_Longitude floatNOTNULL,
C_Latitude floatNOTNULL,
Ct_Id intFOREIGNKEYREFERENCES CR_City(Ct_Id)
)
```

- TabelaCR_City

```
createtable CR_City
(
Ct_Id intPRIMARYKEYIDENTITY,
Ct_City varchar(255)NOTNULL,
St_Id intFOREIGNKEYREFERENCES CR_State(St_Id)
)
```

- TabelaCR_State

```
createtable CR_State
(
St_Id intPRIMARYKEYIDENTITY,
St_Statevarchar(255) NOT NULL
)
```

- TabelaCR_Users

```
createtable CR_Users
(
U_Id intPRIMARYKEYIDENTITY,
U_FirstName varchar(255)NOTNULL,
U_LastName varchar(255)NOTNULL,
U_Login varchar(255)NOTNULL,
U_Password varchar(255)NOTNULL,
U_Email varchar(255)NOTNULL,
U_Phone varchar(255)NOTNULL,
U_Level intNOTNULL,
U_Id_Manager intFOREIGNKEYREFERENCES CR_Users(U_Id)
)
```

- TabelaCR_Tasks

```
createtable CR_Tasks
(
Tk_Id intPRIMARYKEYIDENTITY,
Tk_Name varchar(255)NOTNULL,
Tk_Description varchar(800)NOTNULL,
Tk_InitialDate dateNOTNULL,
Tk_ConclusionForecast dateNOTNULL,
Tk_ConclusionDate date,
U_Id intFOREIGNKEYREFERENCES CR_Users(U_Id),
D_Id intFOREIGNKEYREFERENCES CR_Demands(D_Id),
)
```

- TabelaCR_AttentionPoints

```
createtable CR_AttentionPoints
(
  Ap_Id intPRIMARYKEYIDENTITY,
  Ap_Name varchar(255)NOTNULL,
  Ap_Description varchar(800)NOTNULL,
  Tk_ConclusionDate date,
  Ap_ActionPlanvarchar(800)NOTNULL,
  Ap_Statusvarchar(255)NOTNULL,
  U_Id intFOREIGNKEYREFERENCES CR_Users(U_Id),
  D_Id intFOREIGNKEYREFERENCES CR_Demands(D_Id),
)
```

Apêndice B

Layout do Sistema

A seguir são apresentados os layouts das principais telas da versão alpha implementada.

- Tela de login:



The screenshot shows a login page with a header 'Monitoramento de Relações com Comunidades' and a sub-header 'Sign In'. It contains input fields for 'Login:' and 'Password:', 'Clear' and 'Send' buttons, and a link for non-members. A footer note states: 'Este site foi desenvolvido por RML. Todos os direitos reservados.'

Figura 5 – Tela de login

- Tela inicial:



The screenshot shows a dashboard with a left sidebar menu and a main content area. The sidebar includes links for 'Manifestações da Comunidade', 'Issues', 'Mapa de Comunidades', 'Lista de Comunidades', and 'Lista de Pessoas'. The main content area displays a welcome message, a table of open actions, and a table of attention points.

Tarefa	Descrição	Preciso de Conclusão	Editar
Responder manifestação	Responder à manifestação: Trepidações causadas pela passagem do trem da empresa estão gerando rechaduras nas casas da comunidade	12/07/13	

Pt de Atenção	Impacto	Plano de Ação	Editar
---------------	---------	---------------	--------

Figura 6 – Tela inicial

- Tela com o mapa de comunidades:



Figura 7 – Tela com mapa de comunidades

- Tela com o relatório:

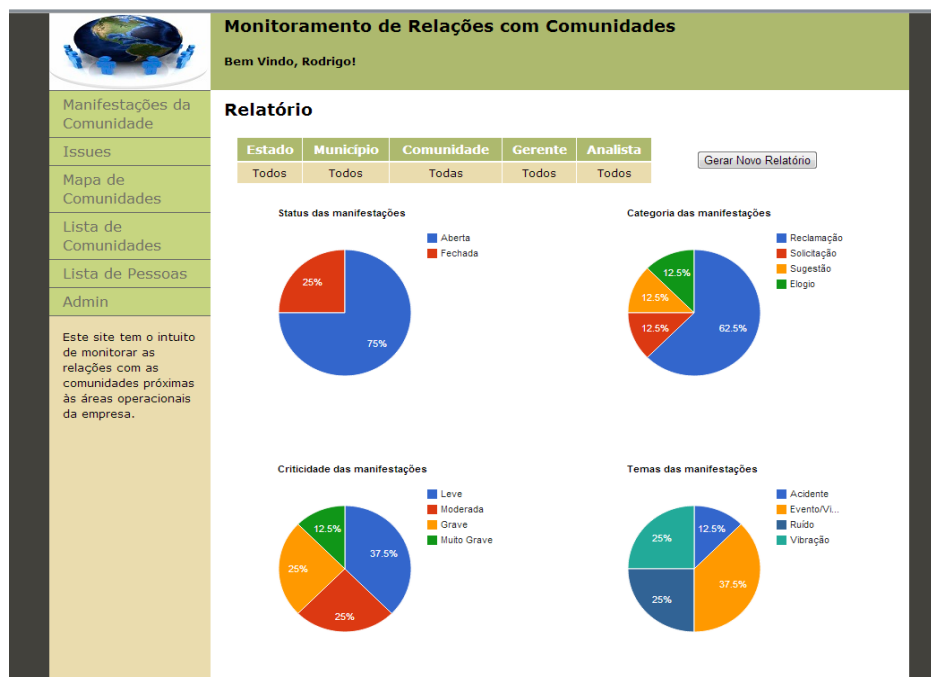


Figura 8 – Tela com relatório