

PORTAL SGMS

Sistema Web de Apoio ao Processo de Gerência e Controle na Manutenção de Software

Aluno: Thiago Ribeiro Vieira

Orientador: Luciana Pereira de Araújo

Roteiro

- **Introdução**
- **Objetivos**
- **Fundamentação Teórica**
 - Ciclo de vida do desenvolvimento de software
 - Modelo Cascata
 - Processos na manutenção de software
 - Scrum
- **Trabalhos Correlatos**
 - Qualitor
 - Software de apoio à gerência de solicitação de mudanças
 - Software de apoio à manutenção de sistemas baseado em normas de qualidade
- **Requisitos (RF / RNF)**
- **Especificação**
 - Diagrama de casos de uso
 - Diagrama de atividade
- **Implementação**
 - Model-view-controller
- **Operacionalidade da Implementação**
- **Resultados e Discussões**
- **Conclusões e Sugestões**
- **Demonstração**

Introdução

- A manutenção de software é uma das fases em Engenharia de Software do ciclo do desenvolvimento, a qual engloba as etapas de correção, adaptação e evolução (MAGELA, 2006 apud WEBER, 2014, p. 12).
- Um dos problemas que a equipe de manutenção enfrenta está relacionada ao gerenciamento das atividades da equipe.
- “O problema do alto custo referente a manutenção não passa de uma má comunicação entre a equipe responsável por realizar a manutenção” (APRIL e ALAN, 2008 apud SANTOS, 2015, p. 34).
- Diante deste cenário, este trabalho apresenta o desenvolvimento de um portal web para atender os segmentos que visam controlar a manutenção de software em uma empresa com vários produtos ou projetos.

Objetivos

- O objetivo geral do trabalho é o desenvolvimento de um sistema web para atender os segmentos que visam planejar, controlar e gerenciar a manutenção de software.

Os objetivos específicos do trabalho são:

- a) oferecer um módulo para o acompanhamento e gerenciamento dos processos no ciclo de vida da manutenção de software por cada um dos papéis;
- b) permitir a centralização de indicadores de performance e atividade da manutenção nos seus processos;
- c) garantir o andamento e a integridade da gestão das solicitações ou demandas nas determinadas áreas de serviço, como suporte, desenvolvimento, teste e documentação.

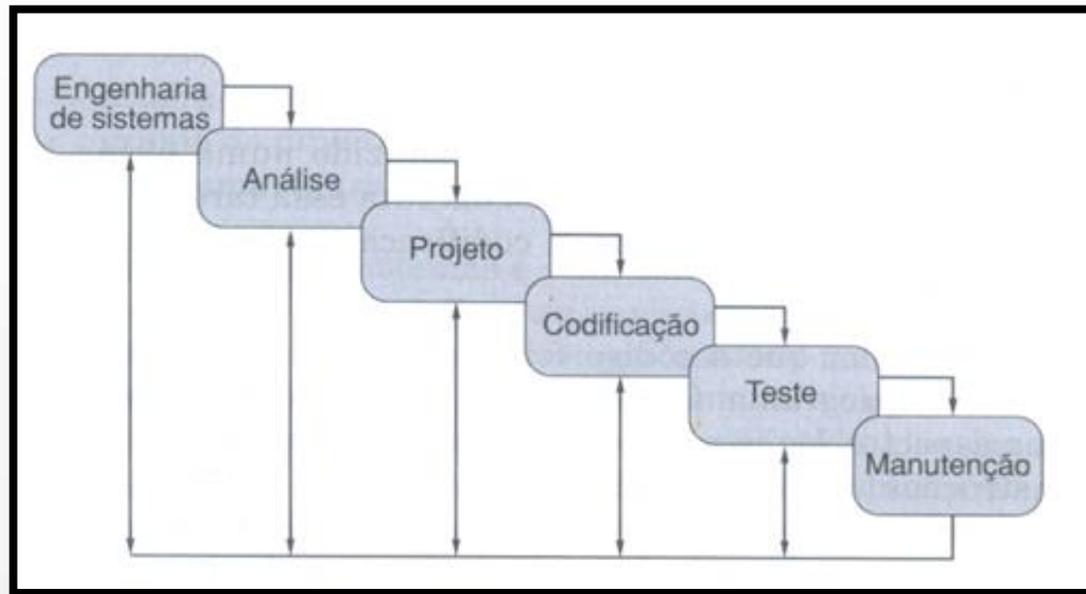
Fundamentação Teórica

Ciclo de vida do desenvolvimento de software

- O ciclo de vida do software descreve como um software deve ser desenvolvido, onde está envolvido desde o início da ideia de sua criação até a sua descontinuação.
- O modelo de ciclo de vida é a primeira escolha a ser feita no processo de software.
- A partir desta escolha define-se a maneira mais adequada de obter as necessidades do cliente (MACÊDO; SPÍNDOLA, 2016).
- Exemplos de ciclos de vida de software são: Espiral, Cascata, Ciclo de Vida Iterativo, entre outros.

Fundamentação Teórica: Modelo Cascata

- O modelo cascata consiste de fases que são realizadas sequencialmente, de forma que uma atividade deve ser iniciada após o término do seu antecessor.



- A Imagem demonstra as etapas discutidas por Pressman (2006), sendo que esse modelo sugere uma abordagem sequencial para o desenvolvimento de software.

Fundamentação Teórica: Processos na manutenção de software

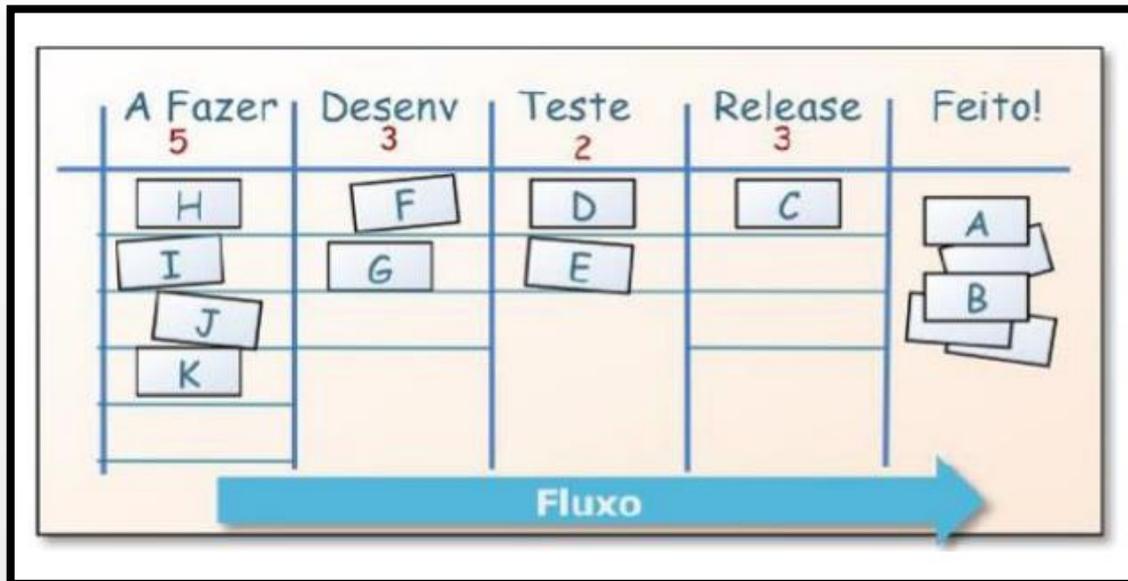
- Dentro do processo de manutenção são estabelecidos papéis para desempenhar atividades.
- Os papéis não são indivíduos e sim descrevem como os indivíduos atribuídos aos papéis se comportarão no contexto de um processo.
- Os papéis a serem desempenhados no processo são o de usuário (solicitante), suporte, desenvolvimento, testes e a documentação.
- Manutenção de software começa com um pedido para mudar o sistema, geralmente um pedido do usuário.
- Sempre que surgir alguma dúvida ou algum problema for identificado, o usuário deverá descrever os procedimentos que está realizando, conforme seu conhecimento de uso do produto.

Fundamentação Teórica: Processos na manutenção de software

- Os pedidos abertos ao suporte devem ser analisados, ou seja, ao se constatar o erro no sistema.
- O implementador é quem procede a correção dos defeitos identificados e também realiza análise de impacto da correção.
- O teste de software pode ser considerado como uma fase do processo de desenvolvimento de software, cujo objetivo é atingir um nível de qualidade.
- Por fim, segundo Santos (2015, p.96) “a última atividade, em relação à demanda, é verificar se o catálogo foi devidamente atualizado. Esta é parte essencial da documentação relativa à demanda”.

Fundamentação Teórica: Scrum

- Dentro do processo de manutenção podemos aplicar o Scrum.
- O Scrum visa promover um modelo de auto-organização da equipe, aonde todos são responsáveis pela qualidade, além de fornecer maior visibilidade e melhores níveis de detalhamento (GRIMHEDEN, 2013).
- No caso da manutenção é aplicado o Scrum em Bug *Fixing*, um tipo de projeto destinado somente para correções do produto.



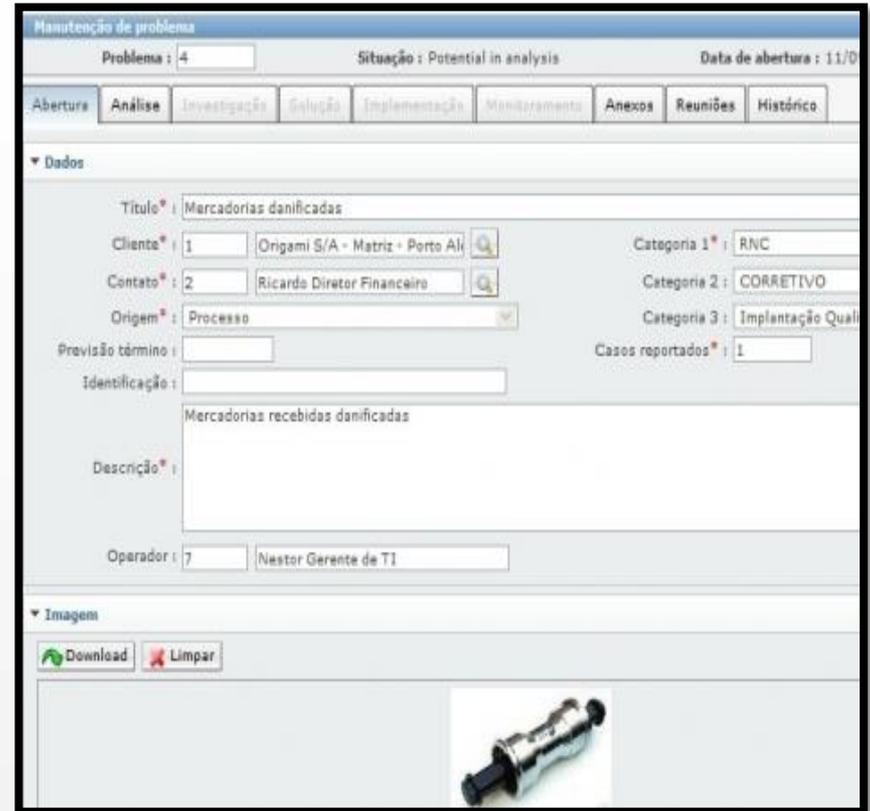
Trabalhos Correlatos

Foram selecionados uma ferramenta de mercado e dois trabalhos acadêmicos como trabalhos correlatos, sendo eles:

- **Qualitor**
- **Software de apoio à gerência de solicitação de mudanças**
- **Software de apoio à manutenção de sistemas baseado em normas de qualidade**

Trabalhos Correlatos: Qualitor

- O Qualitor é um software para gerenciamento de atendimento, podendo ser utilizado em *service-desk* ou *help-desk* (QUALITOR, 2016).
- É um sistema correlato ao portal, entretanto só atendendo uma parte do ciclo da manutenção de software.
- A ferramenta tem o objetivo de trabalhar na abertura e gestão de demanda para o suporte e atendimento.
- Linguagem PHP e banco de dados PostgreSQL.



The screenshot displays the 'Manutenção de problema' (Problem Maintenance) interface. At the top, it shows 'Problema: 4', 'Situação: Potential in analysis', and 'Data de abertura: 11/0'. Below this is a navigation bar with tabs: 'Abertura', 'Análise', 'Investigação', 'Solução', 'Implementação', 'Monitoramento', 'Anexos', 'Reuniões', and 'Histórico'. The main section is titled 'Dados' and contains several fields: 'Titulo*' (Mercadorias danificadas), 'Cliente*' (1 Origami S/A - Matriz - Porto Al), 'Contato*' (2 Ricardo Diretor Financeiro), 'Origem*' (Processo), 'Previsão término:', 'Identificação:', 'Descrição*' (Mercadorias recebidas danificadas), and 'Operador:' (7 Nestor Gerente de TI). On the right side, there are 'Categoria 1*' (RNC), 'Categoria 2*' (CORRETIVO), 'Categoria 3*' (Implantação Quali), and 'Casos reportados*' (1). At the bottom, there is an 'Imagem' section with 'Download' and 'Limpar' buttons, and a small image of a pen.

Trabalhos Correlatos: Software de apoio à gerência de solicitação de mudanças

- Software desenvolvido por Fabrício Oliveira em seu trabalho de conclusão de curso.
- Implementado na linguagem Java que aborda um processo de gerência de solicitações de mudanças.
- Esse software atende as normas do ISO/IEC 15504, auxiliando as empresas na criação, definição e gerenciamento das solicitações de alteração de sistemas.
- O analista recebe as solicitações abertas, dá o seu parecer a respeito e assim encaminha ao testador.
- O testador aprova o teste e a partir desse ponto, o solicitante é notificado e o caso é encerrado.

The screenshot shows a web-based form titled 'Formulário de Solicitação de Mudança'. The form contains the following fields and values:

- Nr. Solicitação:** 10000
- Data:** 01/11/2006
- Hora:** 14:41:56
- Título:** PROBLEMAS NO TELAS DE MALHAS
- Descrição:** PROBLEMAS AO SALVAR NOVO ITEM DE MALHAS.
- Observacao:** QUANDO SELECIONO O BOTÃO SALVAR A TELA FECHA.
- Solicitante:** FABRICIO OLIVEIRA
- Sistema:** SISTEMA INDUSTRIAL
- Versão:** 2.1.1
- Módulo:** BENEFICIAMENTO
- Prioridade:** BAIXA
- Funcionalidade:** PA - MALHAS
- Classificacao:** CORRETIVA

At the bottom of the form, there is a navigation bar with several icons: a left arrow, a double left arrow, a right arrow, a double right arrow, a document icon, a trash icon, a printer icon, and a red circle icon.

Trabalhos Correlatos: Software de apoio à manutenção de sistemas baseado em normas de qualidade

- Software desenvolvido por Charles Hoppe em seu trabalho de conclusão de curso.
- Na implementação e codificação do sistema foi utilizado o Delphi 3.0.
- O software tem o objetivo de auxílio ao processo de manutenção de sistemas, nas etapas de solicitação e análise das pendências, bem como no registro das manutenções do sistema.
- Este tipo de controle ajuda a equipe de manutenção de sistemas e até mesmo os desenvolvedores de novas aplicações com feedback de problemas passados para auxiliá-los em dificuldades futuras (HOPPE, 1999, p. 105).

The screenshot shows a Windows-style application window titled "Manutenção de Pendências de Correção". The interface includes several input fields and radio button options. The fields are filled with the following data: Número: 9; Cliente: 6 Empresa Venda de Produtos; Data: 20/12/1999; Solicitante: José da Silva; Sistema: 7 Sistema FiscalPrint; Rotina: Frente de Caixa; Tipo de Problema: 2 Erro de Leitura. The "Descrição do Problema" field contains the text: "Exibe mensagem de erro quando se tenta digitar o código do produto. MENSAGEM: 'Erro de Leitura: CADPROD.DB'". The "Efeito do Problema" field contains: "Aborta o programa". On the right side, there are two groups of radio buttons: "Situação da Solicitação" with options "Recebida" (selected), "Em Análise", "Aprovada", and "Reprovada"; and "Gravidade para o Cliente" with options "Alta" (selected), "Média", and "Baixa". Below these, there is another "Urgência" group with options "Alta" (selected), "Média", and "Baixa". At the bottom, there are date fields for "Data Solicitada" (20/12/1999) and "Data de Resposta" (20/12/1999), and a "Data Encaminhada para análise" field (20/12/1999). A red checkmark icon and the word "Análise" are visible in the bottom right corner.

Requisitos

Requisitos Funcionais

RF02: O sistema deverá permitir cadastrar uma solicitação.

RF03: O sistema deverá permitir consultar uma solicitação.

RF04: O sistema deverá permitir escalar a solicitação para o nível superior.

RF07: O sistema deverá permitir revisar a solicitação.

RF11: O sistema deverá permitir cadastrar projetos.

RF12: O sistema deverá permitir cadastrar usuário.

RF15: O sistema deverá permitir configurar o perfil do usuário.

RF17: O sistema deverá permitir o gerente a consultar indicadores sobre o processo no ciclo da manutenção de software.

RF19: O sistema deverá demonstrar todas as solicitações pendentes em seu nome, aguardando sua ação.

Requisitos

Requisitos Não Funcionais

RNF1: O sistema deve utilizar o banco de dados SQL Server 2012 ou superior.

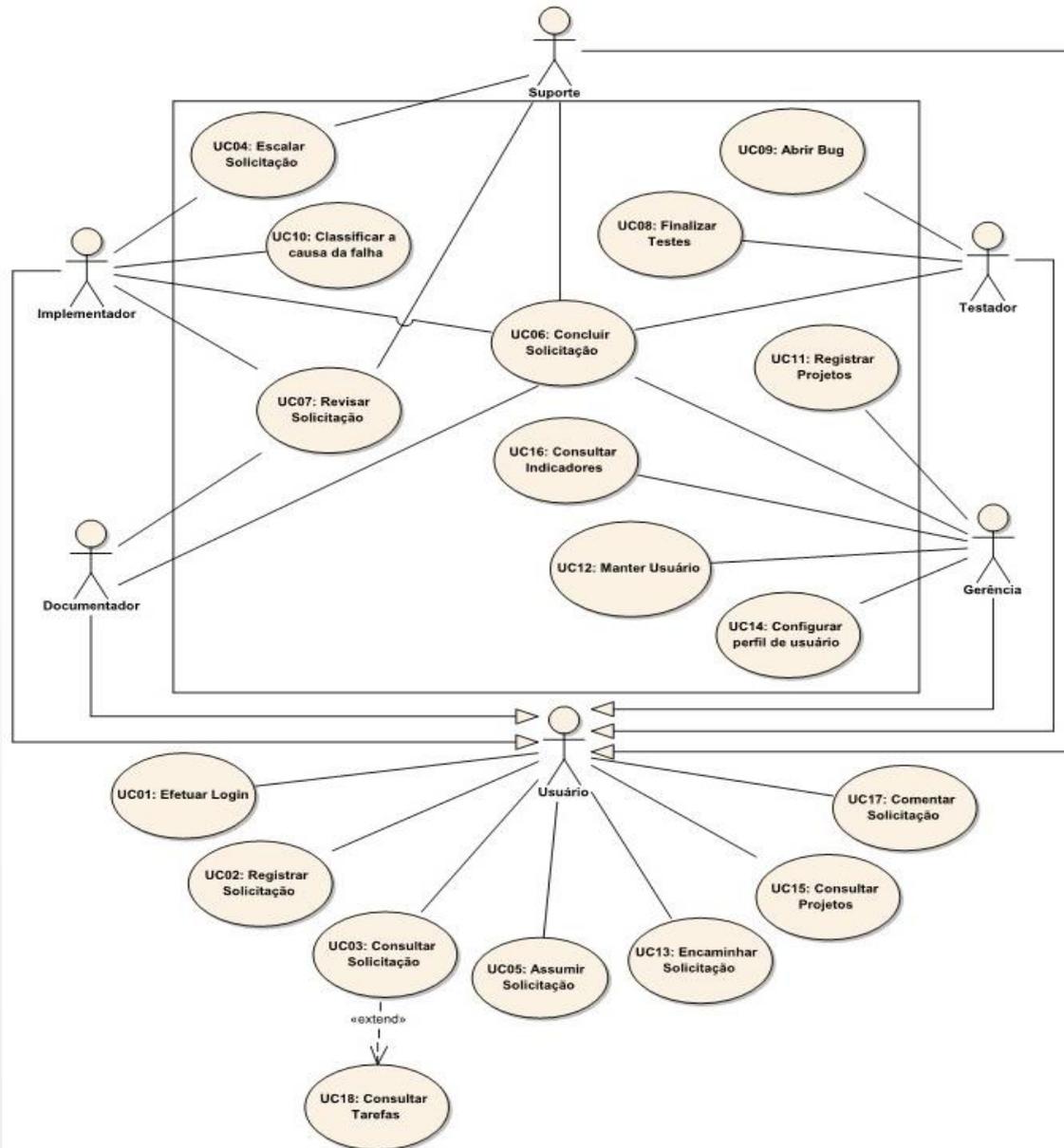
RNF2: O sistema deve ser acessado através de um web browser (Internet Explorer 8.0 ou superior e Google Chrome).

RNF03: O ambiente de desenvolvimento será o Visual Studio 2012 ou superior.

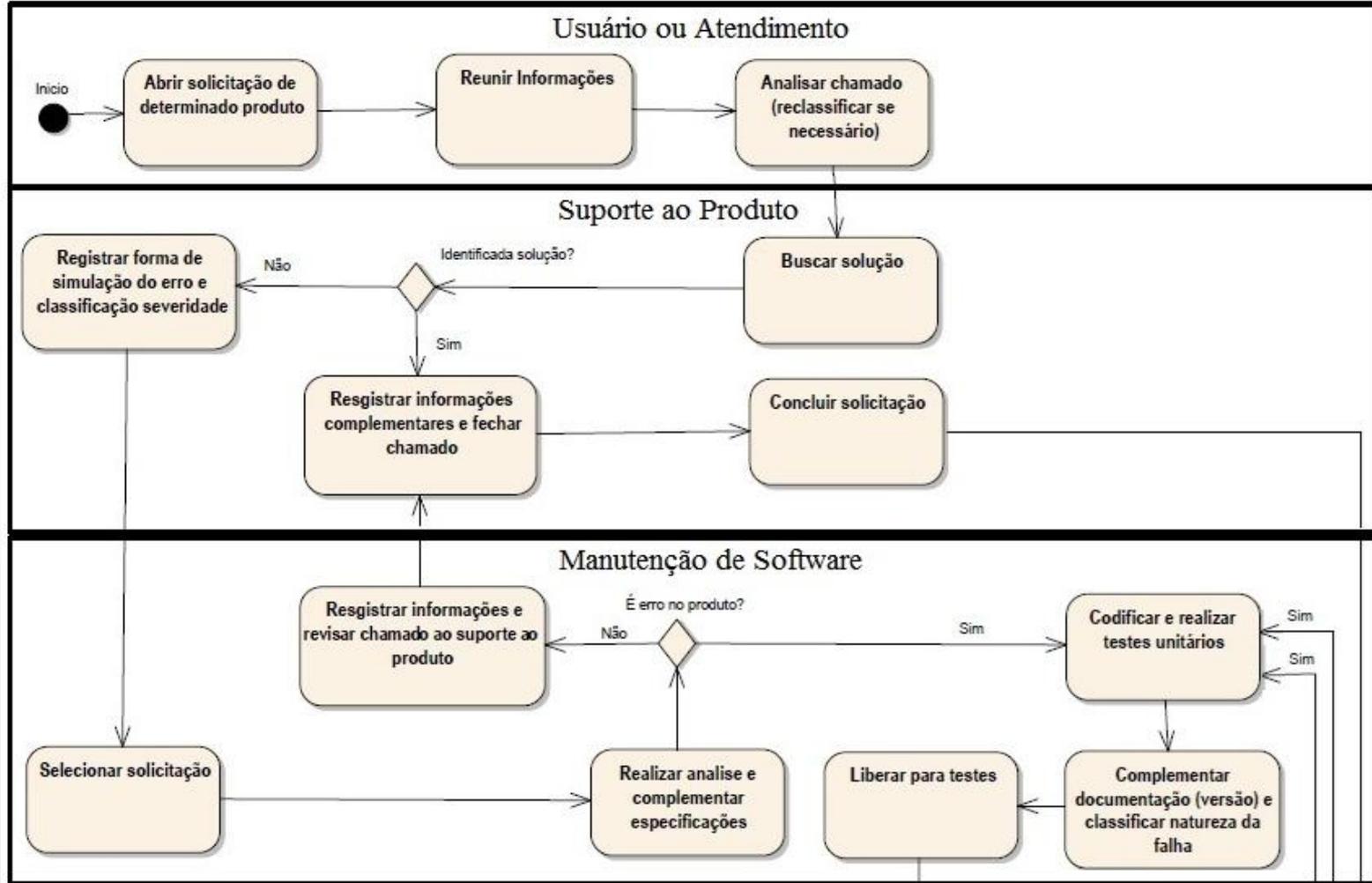
RNF04: O sistema deve utilizar a linguagem C# e NET framework que serão utilizados no sistema de gerenciamento na web.

RNF05: O sistema deve apresentar mensagens de erro de modo evidente e intuitivo, fazendo com que o usuário identifique sua origem e como proceder após sua ocorrência.

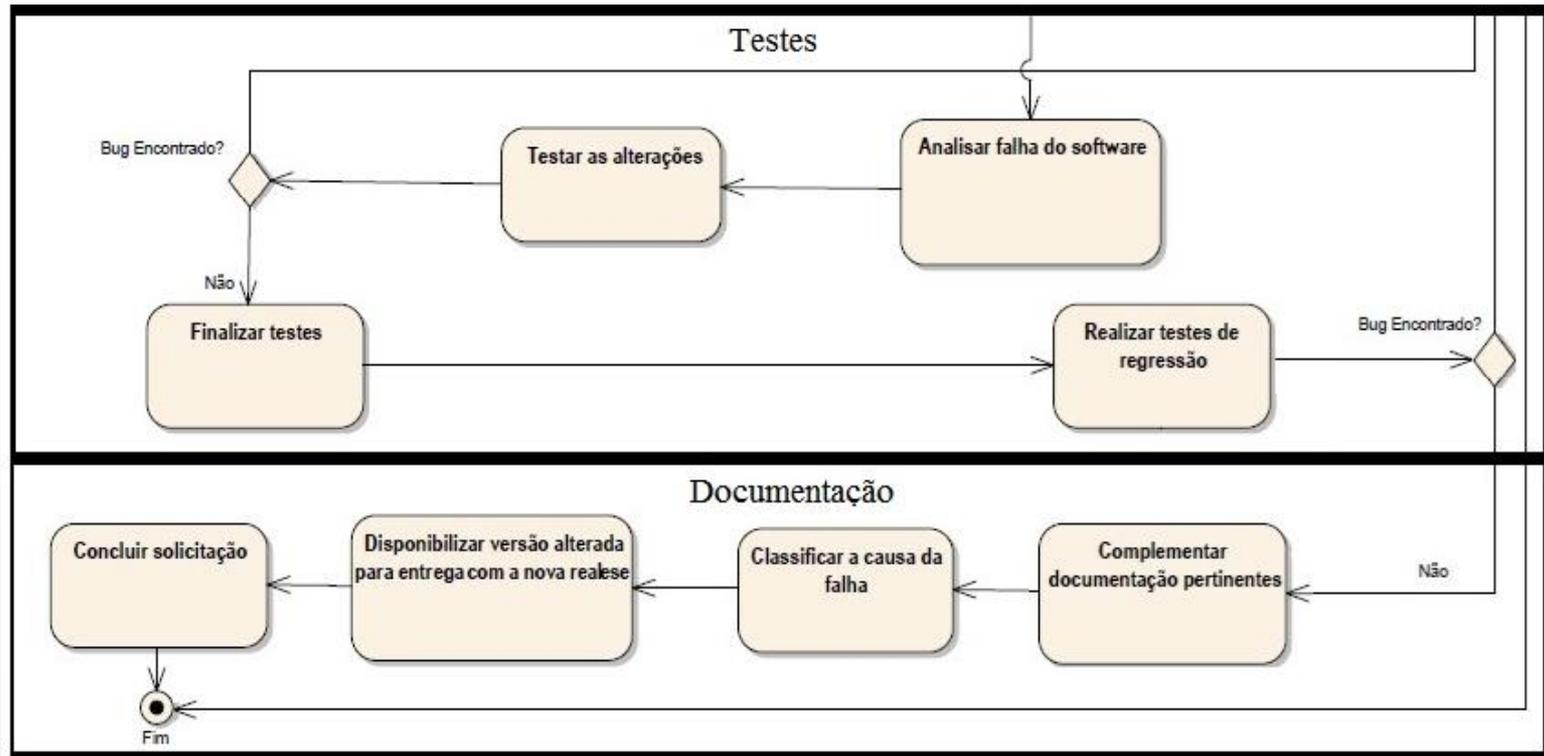
Especificação: Diagrama de casos de uso



Especificação: Diagrama de atividade



Especificação: Diagrama de atividade



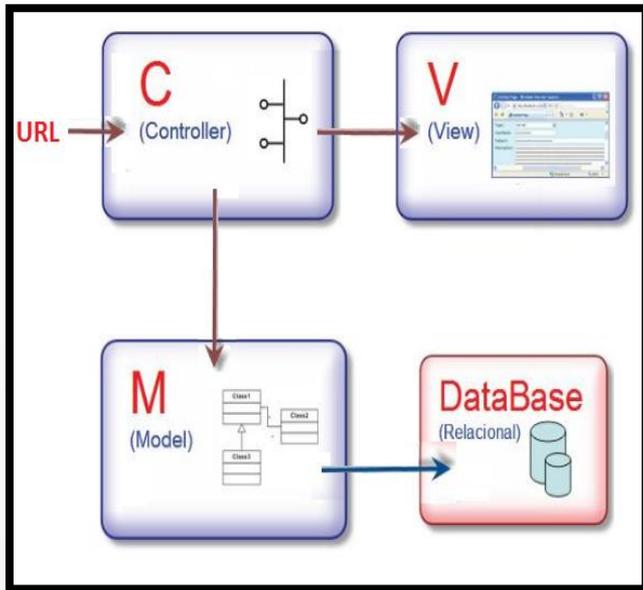
Implementação

Técnicas e ferramentas utilizadas para o sistema desenvolvido

- O sistema foi desenvolvido utilizando a linguagem C# com *Framework .Net*. Para o desenvolvimento aplicou-se programação Orientada a Objeto (OO) para manipulação de objetos, declaração de atributos e métodos.
- A manipulação de dados vindos da base SQL Server criou-se um pacote intermediário com classes.
- Como padrão de arquitetura de software, foi aplicado o modelo Design *Pattern Model-view-controller* (MVC).

Implementação

Técnicas e ferramentas utilizadas para o sistema desenvolvido



Model

O modelo (*model*) consiste nos dados da aplicação, regras de negócios, lógica e é responsável pela manipulação de dados.

View

Na visão (*view*) encontram-se as saídas de representação dos dados, como as páginas em HTML criadas.

Controller

O controlador (controller) é formado pelas as classes responsáveis em realizar a mediação.

Implementação: model-view-controller

The image displays the Visual Studio Solution Explorer for a project named 'Solicitacao' (2 projects). The structure is organized as follows:

- Solicitacao**
 - Properties
 - References
 - App_Data
 - App_Start
 - Content
 - Controllers
 - fonts
 - Models
 - Scripts
 - Views
 - ApplicationInsights.config
 - favicon.ico
 - Global.asax
 - packages.config
 - Project_Readme.html
 - Startup.cs
 - Web.config
 - Solicitacao.Repositorio**
 - Properties
 - References
 - BancoDados

The right side of the image shows three detailed views of the MVC folders:

- Controllers**
 - GerenciaController.cs
 - HomeController.cs
 - LoginController.cs
 - ProjetosController.cs
 - SolicitacaoController.cs
 - TarefaController.cs
- Models**
 - CausaFalhaModels.cs
 - EmailModels.cs
 - EventoModels.cs
 - LoginModels.cs
 - OperacaoModels.cs
 - ProjetoModels.cs
 - RegrasModels.cs
 - ResponsavelModels.cs
 - SolicitacaoModels.cs
- Views**
 - Gerencia**
 - CriarProjeto.cshtml
 - Index.cshtml
 - Indicadores.cshtml
 - Responsavel.cshtml
 - Usuario.cshtml
 - Home
 - Login
 - Projetos
 - Shared
 - Solicitacao
 - Tarefa

Implementação: Model

Model

A classe *model* ProjetoModels constitui quais propriedades são persistentes em um projeto da manutenção no sistema, sendo elas código do projeto, nome, descrição, responsável e situação.

```
namespace Solicitacao.Models
{
    8 references
    public class ProjetoModels
    {
        public int CodigoProjeto { get; set; }

        [Required]
        [Display(Name = "Nome do Projeto")]

        public string Nome { get; set; }

        [Required]
        [Display(Name = "Descrição")]

        public string Descricao { get; set; }

        [Required]
        [Display(Name = "Responsável")]

        public string Responsavel { get; set; }
    }
}
```

Implementação: View

View

Em seguida a view é representada em páginas em HTML, nele inserimos qual a model será composta, ou seja, quais campos serão manipulados na página.

```
@model Solicitacao.Models ProjetoModels
@{
    ViewBag.Title = "Criar Novo Projeto";
    List<string> ListResposanvel = ViewBag.ListResposanvel;
}

<h2>Novos Projeto</h2>

<p class="text-success">@ViewBag.StatusMessage</p>
<div>
    <h4>Informações sobre o projeto</h4>
    <hr />
    <dl class="dl-horizontal">
        @using (Html.BeginForm("CriarProjeto", "Gerencia", FormMethod.Post))
        {
            @Html.ValidationSummary("", new { @class = "text-danger" })
            <div>
                <fieldset>
                    <div class="editor-label">
                        @Html.LabelFor(m => m.Nome, new { @class = "col-md-2 control-label" })
                    </div>
                    <div class="editor-field">
                        @Html.TextBoxFor(m => m.Nome, new { @class = "form-control" })
                    </div>
                </fieldset>
            </div>
        }
    </dl>

```

Implementação: Controller

Controller

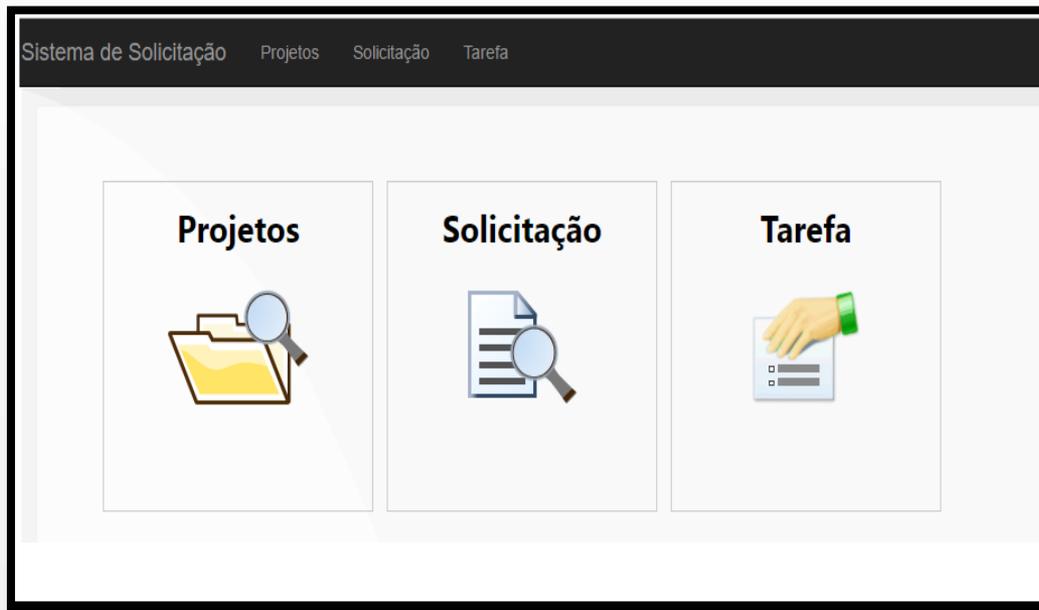
- A controller é responsável por receber todas as requisições do usuário. Seus métodos chamados actions são responsáveis por uma página.

```
[HttpPost]
References
public ActionResult CriarProjeto(ProjetoViewModels projeto)
{
    if (regras.VerificarUsuarioLogado())
    {
        if (ModelState.IsValid)
        {
            banco.RegistrarProjeto(projeto.Nome,
                                   regras.CodigoUsuario(projeto.Responsavel),
                                   projeto.Descricao);
            ModelState.AddModelError("Mensagem", "Projeto criado com sucesso.");
        }
        else
        {
            ModelState.AddModelError("Mensagem", "Não foi possível criar novo projeto.");
        }

        return View("CriarProjeto");
    }
    else
    {
        return RedirectToAction("Login", "Login");
    }
}
```

Operacionalidade da Implementação

- Cada membro do sistema tem funções definidas e para fazer uso deste deverá estar cadastrado no sistema.
- O gerente dos projetos tem acesso a todas as funcionalidades do sistema, além do seu ambiente de gerencia onde é responsável por manter todos os cadastros básicos para que uma solicitação possa ser criada.



Operacionalidade da Implementação

- Esta tela contempla todas as solicitações abertas com informativos em relação ao seu estado atual. Disponibiliza filtros para o usuário buscar e filtrar as solicitações que deseja atuar ou visualizar.

Sistema de Solicitação | Projetos | Solicitação | Tarefa | Gerência | Bem-Vindo thiago.vieira | Sair

Solicitação

Código: Nome:

Situação: Responsável:

Projeto: [Abrir Solicitação](#)

Código	Nome	Responsável	Situação	Data Abertura
0009	Erro ao visualizar Nota Fiscal 34694-6	thiago.vieira	Desenvolvimento	17/10/2016 15:05:44
0008	Erro ao importar RPS	thiago.vieira	Desenvolvimento	17/10/2016 14:04:25
0007	Ajustar tela de Indicadores	Equipe.Suporte	Suporte	17/10/2016 14:03:21
0006	Label da operação inconsistente	thiago.vieira	Desenvolvimento	14/10/2016 13:30:10
0005	Erro nos caracter especiais	viviane.degan	Concluída	14/10/2016 13:23:06
0004	Erro no cadastrar	****	Aceita	12/10/2016 14:18:00
0003	Falha ao cadastrar novo aluno	****	Aceita	01/10/2015 08:56:44
0002	Falha ao enviar e-mail com a porta 587	****	Aceita	12/10/2016 13:51:34
0001	Erro ao gerar nota fiscal	****	Aceita	12/10/2016 13:41:26

1

Sistema de Solicitação | Projetos | Solicitação | Tarefa | Gerência | Bem-Vindo thiago.vieira | Sair

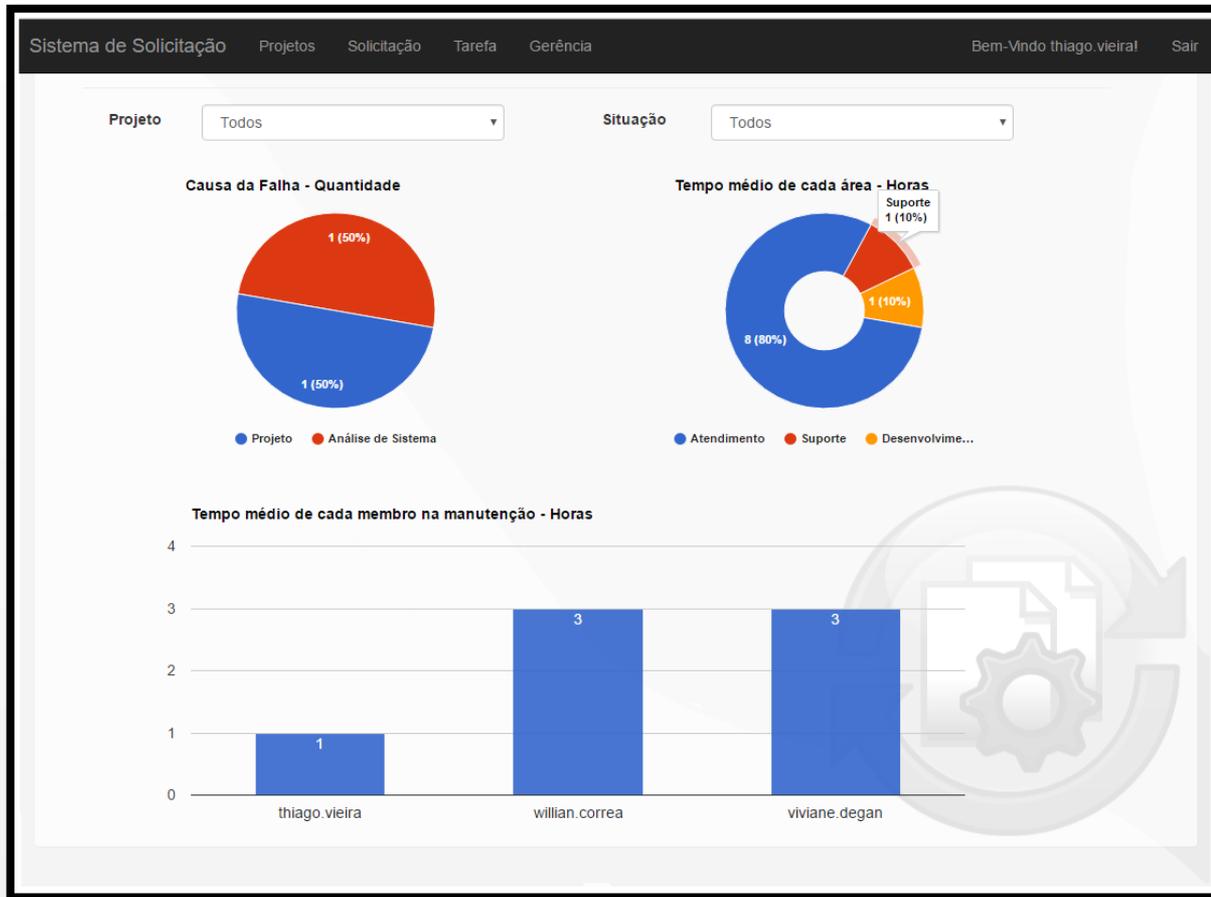
9 - Erro ao visualizar Nota Fiscal 34694-6

Solicitação do projeto: [NFSe - Nota Fiscal de Serviços Eletrônica](#)

Situação: [Desenvolvimento](#)
Responsável: thiago.vieira

Inclusão	Autor	Situação	Operação	Descrição
17/10/2016 15:06:41	Equipe.Desenvolvimento	Desenvolvimento	Encaminhar	Encaminhado para thiago.vieira.
17/10/2016 15:06:09	Equipe.Suporte	Suporte	Escalar	Segue para análise e correção.
17/10/2016 15:05:44	Equipe.Suporte	Atendimento	Inclusão	Ao clicar em visualizar a nota fiscal 34694-6 o sistema fica carregando e fica em loop.

Operacionalidade da Implementação



- Tela responsável por dar uma dimensão de como está a performance dos processos na manutenção de software.

Resultados e Discussões

- O objetivo proposto por este trabalho foi atingido uma vez que foi desenvolvido um sistema que contempla os requisitos identificados.
- O objetivo específico que tinha por necessidade o acompanhamento e gerenciamento por cada um dos papéis, foi atingido, pois o sistema desenvolvido possibilita a troca de informações entre os envolvidos no processo de manutenção, permitindo que as solicitações sejam gerenciadas, acompanhadas e controladas até a sua conclusão.
- O segundo objetivo específico que tinha como necessidade disponibilizar indicadores que acompanharão a performance da manutenção, foi atingido, pois o sistema disponibiliza ao gerente da manutenção de software um módulo dedicado a obter esse resultado, no qual apresenta os indicadores.
- O último objetivo específico que tinha como necessidade de garantir o andamento e a integridade da gestão das solicitações em determinada área de serviço, foi atingido, no qual o sistema proporciona o trabalho em equipe, através de um mecanismo de controle de solicitações, que notifica via e-mail.

Resultados e Discussões

- Em relação aos trabalhos correlatos, a ferramenta descrita sobre o Qualitor serviu como base na abertura e gestão de demandas para o suporte e atendimento para determinado produto.
- Os dois trabalhos acadêmicos (ambos aplicações desktop), foram utilizados como base para a troca de informações entre os envolvidos no processo e para a análise das pendências, bem como no registro das manutenções do sistema.

Áreas Atendidas	Qualitor	Soft. Solicitação Mudança	Soft. Apoio à manutenção	Trabalho Desenvolvido (Portal Web)
Suporte	X			X
Desenvolvimento		X	X	X
Testes		X		X
Documentação				X

Resultados e Discussões

- O sistema foi publicado em uma rede local em uma empresa do setor de software da região. No qual três membros desta empresa realizaram uma avaliação heurística proposto e definida por Nielsen.

H1: Visibilidade do sistema	O Sistema fornece um feedback adequado aos usuários. Mantém-se o usuário informado sobre o que está ocorrendo?
H2: Compatibilidade do sistema com o mundo real	Sistema utiliza termos familiares ao usuário ao invés de termos orientados ao software?
H3: Consistência e mapeamento	A interação do usuário com o sistema está de acordo com o contexto?
H4: Reconhecimento ao invés de memorização	O sistema propõe de instruções visíveis ou ser de fácil recuperação quando necessárias?
H5: Flexibilidade e eficiência de uso	Permitir que os usuários configurem o sistema de acordo com a necessidade do contexto?
H6: Design estético e minimalista	Exibe apenas as informações que sejam importantes e necessárias?
H7: Gerenciamento de erros	O sistema apresenta as mensagens de erro de forma clara, indicando o problema e sugerindo uma solução?
H8: Facilidade de entrada, visualização e tela	O sistema exige e apresenta apenas informações cruciais sobre o sistema?
H9: Convenções estéticas, sociais e privativas	O sistema propõe e deixa claro que as informações do usuário estão seguras?
H10: Fornece comunicação de artefatos compartilhados	O sistema disponibiliza informações sobre as ações dos outros usuários?
H11: Fornecer proteção	O sistema propõe de um mecanismo de segurança?
H12: Gerenciamento de colaboração	O sistema possibilita o trabalho em conjunto no processo de manutenção de software?

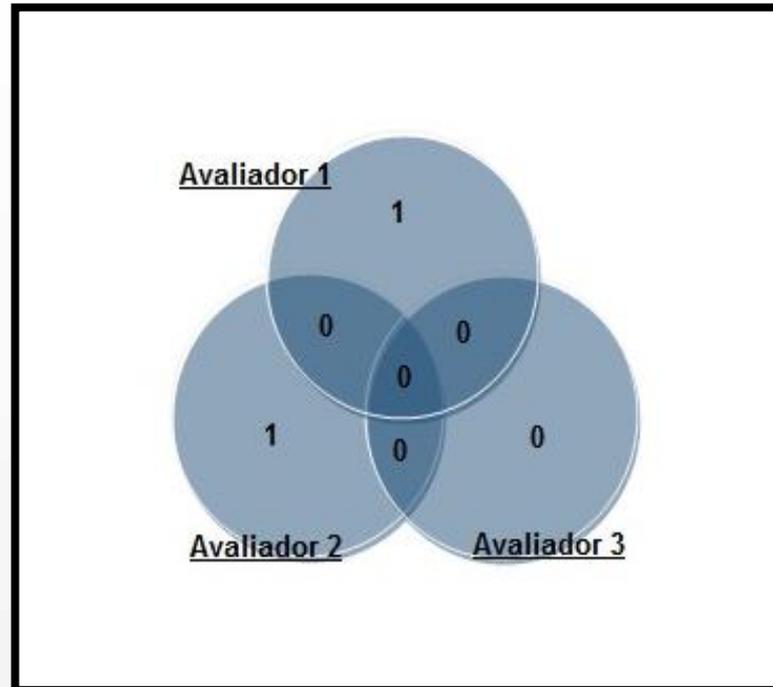
O objetivo dessa avaliação é identificar problemas de usabilidade do sistema e o seu grau de gravidade, como:

- (1) Baixa
- (2) Média
- (3) Alta
- (4) Altíssima

Resultados e Discussões

- A avaliação heurística encontrou 2 problemas de usabilidade.

Heurística	Problema	Gravidade
H1	--	--
H2	1	2
H3	--	--
H4	--	--
H5	--	--
H6	--	--
H7	2	2
H8	--	--
H9	--	--
H10	--	--
H11	--	--
H12	--	--



- Dois avaliadores encontraram problemas distintos, sendo que um dos avaliadores não encontrou nenhum problema, tendo todos os requisitos de usabilidade atendidas.

Conclusões

- Neste trabalho é apresentado um sistema para atender os segmentos que visam planejar e controlar a manutenção de software em uma organização com vários produtos ou projetos.
- Pode-se afirmar que os objetivos foram totalmente cumpridos, pois a aplicação promove a troca de informações entre os envolvidos no processo de manutenção, permitindo que as solicitações sejam gerenciadas, acompanhadas e controladas até a sua conclusão.
- Dessa forma, é possível garantir o andamento e a integridade das solicitações dentro do ciclo de vida na manutenção.
- A contribuição deste trabalho é permitir que qualquer organização possa melhorar as áreas de comunicação e troca de informações dentro de cada área contemplada na manutenção de software, sendo que um dos problemas que a equipe de manutenção enfrenta está relacionada ao gerenciamento das atividades da equipe (WEBER, 2014).
- Com a utilização do sistema, esperasse uma facilidade na visibilidade do andamento dos processos na manutenção de software, além de um acompanhamento mais simples ser gerenciado e controlado.

Conclusões

- Outro item importante é que os gerentes responsáveis pela gestão da manutenção podem monitorar e visualizar a performance de cada membro em suas respectivas áreas, além de um diagnóstico das causas das falhas em seus projetos ou produtos.
- Contudo as solicitações abertas no ciclo de manutenção do software apresentam limitações, sendo:
 - a) não é possível agrupar ou vincular solicitações com o mesmo problema reportado;
 - b) as solicitações não possuem severidade, ou seja, não se sabe qual solicitação deverá ser priorizada.

Sugestões

- Como sugestões para possíveis extensões ao trabalho desenvolvido citam-se:
 - a) gerar novos relatórios para auxiliar nas estatísticas e no gerenciamento das solicitações;
 - b) agregar o modelo de acordo de nível de serviço (SLA) sobre as solicitações reportadas;
 - c) desenvolver um mecanismo capaz de identificar o membro mais adequado para o tratamento de determinada atividade;
 - d) permitir agrupar solicitações com o mesmo problema reportados;
 - e) apresentar sugestões de correções com base em problemas semelhantes encontrados em outras solicitações;
 - f) permitir que os profissionais troquem mensagens através do sistema, de modo que as conversas relacionadas a uma solicitação fique armazenada no sistema;
 - g) alertar os profissionais quando houver uma nova solicitação que seja correspondente ao seu perfil.