Plano de Teste

**Projeto: eveRemind**

**Histórico de Alterações**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| **31/05/2015** | **1.0** | **Preenchimento do Plano parte 1** | **Moisés e Igor** |
| **02/05/2015** | **1.0** | **Preenchimento do Plano parte 2** | **Moisés e Igor** |
|  |  |  |  |

# **Introdução**

## **1.1 Escopo**

O escopo de testes para o sistema eveRemind engloba os produtos de software das seguintes áreas de processos: Requisitos, Arquitetura, Implementação e Manutenção. Os itens detalhados de cada área de processo estão documentados na seção 2.1, Itens de Teste.

## **1.2 Referências**

Externas:

* Norma IEEE 1012:2004
* Norma IEEE 829:2008
* Guia de Maturidade de Processo MPS-BR nível D

Internas:

* Plano de Verificação e Validação

### **1.3 Notações, Siglas, Acrônimos e Abreviações**

|  |  |
| --- | --- |
| Abreviação | Significado |
| PVVS | Plano de Verificação e Validação de Software |
| V&V | Verificação e Validação |
| CT | Caso de Teste |
| EOR | Especificação de Objetivos e Requisitos |
| SAR | Solicitação de Alteração de Requisito |
| HSARs | Histórico de Solicitação de Alteração de Requisitos |
| RM/RP | Requisição de Mudança/Relato de Problema |
| EAS | Especificação Arquitetural de Software |

# **2. Detalhes do Plano de Teste**

## **2.1 Itens de Teste**

Os itens de teste a serem testados estarão identificados na tabela abaixo, indicando o tipo de método a ser aplicado neste:

Métodos:

1. Inspeção de Software

2. Teste de Estrutura

3. Teste Funcional

|  |  |  |  |
| --- | --- | --- | --- |
|  | 1 | 2 | 3 |
| **Requisitos** |  |  |  |
| EOR | X |  |  |
| SARs | X |  |  |
| HSARs | X |  |  |
| **Arquitetura** |  |  |  |
| EAS | X |  |  |
| Diagramas de Classe | X |  |  |
| Diagramas de Componentes | X |  |  |
| Implementação |  |  |  |
| Código fonte (html, java, html, css, xml, jsp, javascript) |  | X | X |
| Manutenção |  |  |  |
| RM/RP | X |  |  |
| Análise de RM/RP | X |  |  |
| Plano de Migração | X |  |  |
| Registro dos Pedidos de Modificação | X |  |  |

Itens de Configuração de Software que serão excluídos dos itens de teste:

* Gerência de Requisitos: Documento de Validação dos Requisitos, Plano de Gerência de Requisitos e Plano de Revisão de Requisitos
* Gerência de Projeto: todos os artefatos de Gerência de Projeto.
* Gerência de Configuração: Plano de Gerência de Configuração e Relatório de Auditoria Física.
* Manutenção: Plano de Manutenção, Plano de Aposentadoria, Procedimentos de Resolução e Resultado da Aposentadoria.
* Garantia da Qualidade: todos os artefatos da Garantia da Qualidade.
* Reuniões de Definição do Processo.

## **2.2 Características a serem testadas**

As características a serem testadas em cada área de processo são:

* Gerência de Requisitos: acurácia, consistência, compreensibilidade, não redundância, não ambiguidade, relevância, viabilidade e testabilidade.
* Arquitetura: coerência com todos os requisitos funcionais e não-funcionais verificados e validados, integridade da construção dos diagramas de visão (regras do UML respeitadas), compreensibilidade, não redundância, viabilidade e testabilidade.
* Implementação: coerência com todos os requisitos funcionais e não-funcionais; coerência com as especificações arquiteturais, funcionalidades esperadas.
* Manutenção: coerência com todos os requisitos funcionais e não-funcionais; coerência com as especificações arquiteturais.

## **2.3 Características a não serem testadas**

Não serão testadas as características de código ligadas ao teste estrutural (caixa branca), testes de Unidade e de Integração de componentes do sistema.

## **2.4 Critérios de sucesso / falha para itens de teste**

Para cada item de teste citado na seção 2.1 os critérios de sucesso e falha são:

* Gerência de Requisitos:
  + Sucesso: Todos os artefatos foram verificados e validados e nenhuma anomalia em relação às características a serem testadas nos requisitos foi encontrada.
  + Falha: Uma não-conformidade foi detectada.
* Arquitetura:
  + Sucesso: O EAS deve estar coerente, compreensível, viável e testável. Os diagramas devem estar utilizando a sintaxe e semântica correta da UML
  + Falha: Uma não-conformidade foi detectada tanto na questão de estrutura da arquitetura quanto na coerência com os requisitos estabelecidos.
* Implementação:
  + Sucesso: Todos os casos de teste planejados foram executados e seus resultados foram conforme o esperado.
  + Falha: Um caso de teste revelou uma fragilidade, comportamento inesperado ou anômalo em relação ao previsto.
* Manutenção:
  + Sucesso: Todos os artefatos de manutenção foram verificados e validados.
  + Falha: Foi encontrado alguma anomalia nos artefatos de manutenção.

## **2.5 Entregáveis de Teste**

Os entregáveis de teste são: Plano de Teste, Casos de Teste, Resultados de Teste, Relatório de Atividades de Verificação e Validação e Relato (s) de Anomalia (s).

## **2.6 Procedimentos de Teste**

* Inspeção de software:
  + Objetivo: Verificar se não existem inconsistências nos documentos a serem testados.
  + Tipo de teste: Manual
  + Requisitos: As etapas do processo foram concluídas e os documentos gerados entregues ao Analista de Teste. A inspeção pode ser executada em qualquer editor de texto rico.
  + Fluxo:
    - O analista de teste deve receber o documento a ser verificado;
    - Verificar se há erros ortográficos/inconsistências no documento;
    - Caso encontre alguma inconsistência, relatar para o responsável do documento via Relato de Anomalia, para que então o documento seja corrigido.
* Teste Funcional:
  + Objetivo: Testar todas as funcionalidades, resultados e comportamentos esperados na realização dos casos de teste.
  + Tipo de teste: Manual
  + Requisitos: Funcionalidades do sistema foram implementadas. O teste deverá ser realizado em um navegador web.
  + Fluxo:
    - O testador deve planejar os casos de teste, baseando-se nas regras de negócio e requisitos já estabelecidos.
    - Executar os casos de teste e documentar os resultados
    - Caso encontre alguma inconsistência, associar um Relato de Anomalia para este e entregar ao programador.
* Teste de Estrutura:
  + Objetivo: Testar se todos os links estão funcionando conforme o esperado
  + Tipo de teste: Manual
  + Requisitos: Deve ser testado em um navegador web
  + Fluxo:
    - O testador deve verificar se todos os links existentes no sistema estão funcionando corretamente.
    - Caso encontre algum link quebrado ou indicando uma página não esperada relatar aos programadores por meio de um Relato de Anomalia.

# **3. Gerenciamento de Testes**

## **3.1 Ambiente / Infraestrutura**

Hardware: 2 Estações de trabalho com as seguintes configurações mínimas: Intel Core i3, 4GB RAM, 500 GB HD.

Software: Microsoft Office 2013 e navegador web (Mozilla Firefox, Google Chrome, Opera)

Pessoal: 2 Testadores/Analistas de Teste

## **3.2 Responsabilidades e autoridade**

As responsabilidades e autoridades estão referenciadas no [Processo de VER-VAL](../../../Processo%20Genérico/6-Verificacao%20e%20Validacao/VER-VAL-Verificação%20e%20Validação.docx#papeis) (Seção 4).

## **3.3 Treinamento**

O treinamento, para todos os papéis neste projeto, foi realizado via autodidata em ritmo pessoal baseado em computador e formação tradicional em sala de aula.

## **3.4 Cronograma**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Requisitos | Arquitetura | Implementação | Manutenção |
| Inspeção de Software | Início: 03/06/2015 Término: 04/06/2015 | Início: 05/06/2015 Término: 27/06/2015 | Não se aplica | Início: 06/06/2015 Término: 30/06/2015 |
| Teste Funcional | Não se aplica | Não se aplica | Início: 01/06/2015  Término: 30/06/2015 | Não se aplica |
| Teste de Estrutura | Não se aplica | Não se aplica | Início: 01/06/2015 Término: 30/06/2015 | Não se aplica |

# **4. Geral**

## **4.1 Métricas**

As métricas utilizadas para este projeto estão referenciadas no [Processo de VER-VAL](../../../Processo%20Genérico/6-Verificacao%20e%20Validacao/VER-VAL-Verificação%20e%20Validação.docx#metricas) (Seção 5).

## **4.2 Cobertura de Teste**

* Inspeção de Software:

Para a esta técnica de teste, o critério de suficiência de teste é definido por:

De modo que x é maior ou igual a 100%

* Teste Funcional:

Para a esta técnica de teste, o critério de suficiência de teste é definido que 100% dos casos de uso e requisitos funcionais do sistema foram abordados pelos casos de teste.

* Teste de Estrutura:

Para a esta técnica de teste, o critério de suficiência de teste é definido que 100% dos links utilizados em todas as páginas do sistema foram testados.