



INSTITUTO DE
INFORMÁTICA
UFG



ENGENHARIA
DE SOFTWARE

Atividade: Dia de Release
Tarefa: Teste de Pré Release

Versão 0.1

Atividade: Dia de Release	Versão: 0.1
Tarefa: Teste de Pré Release	Data: 19/12/2012
LMP-P03.3.4	

Histórico da Revisão

Data	Versão	Descrição	Autor
03/12/2012	0.1	Elaboração Inicial	Emerson José Porfírio
15/12/2012	0.1	Diagrama da Tarefa	Emerson José Porfírio

Atividade: Dia de Release	Versão: 0.1
Tarefa: Teste de Pré Release	Data: 19/12/2012
LMP-P03.3.4	

Sumário

1.	Objetivos	4
1.1	Escopo	4
2.	Introdução	4
3.	Tarefa: Teste de Pré Release	4
4.	Metas	4
5.	<i>Input</i>	5
5.1	Pré-condições	5
5.2	Entradas	5
6.	<i>Output</i>	5
6.1	Pós-condições	5
6.2	Saídas	5
7.	Diagrama da tarefa: Teste de Pré Release	5
8.	Passos	6
9.	Templates	6
10.	Papéis	6
11.	Padrões Relacionados	6
12.	Riscos	7
13.	Referências	7

Atividade: Dia de Release	Versão: 0.1
Tarefa: Teste de Pré Release	Data: 19/12/2012
LMP-P03.3.4	

Tarefa: Teste de Pré Release

1. Objetivo

Apresentar e documentar a tarefa Teste de Pré Release da atividade Dia de Release (Processo Produzir) que faz parte do LMP – Logiciel Mobile Process a ser utilizado pelo Grupo de Estudo Logiciel como trabalho prático para as disciplinas de Integração I e de Desenvolvimento de Software para Dispositivos Móveis do curso de Bacharelado em Engenharia de Software do INF - UFG.

1.1 Escopo

Tarefa 4 da atividade Dia de Release do LMP – Logiciel Mobile Process (Processo Produzir).

2. Introdução

O processo de desenvolvimento LMP – Logiciel Mobile Process abrange atividades acadêmicas referentes aos processos de engenharia de software do INF-UFG. Este oferecerá o apoio ao processo de desenvolvimento para dispositivos móveis, permitindo que o mesmo seja realizado de acordo com o planejamento de tempo e de recursos e com os requisitos funcionais e de qualidade definidos para os projetos propostos.

O modelo foi baseado no Processo MobileD e no RUP, além de seguir as orientações do Guia do MPS.Br 2011 (nível F).

3. Tarefa: Teste de Pré Release

A finalidade desta tarefa é certificar-se de que o software a ser produzido está pronto para os testes de aceitação e *release*.

4. Metas

O objetivo do **Teste de Pré Release** é:

1. Verificar se o software está pronto para o teste de aceitação executando todos os testes e documentando-os por escrito; e
2. Encontrar possíveis defeitos no software.

Atividade: Dia de Release	Versão: 0.1
Tarefa: Teste de Pré Release	Data: 19/12/2012
LMP-P03.3.4	

5. Inputs

5.1 Pré-condições

1. Os testes de aceitação foram executados. O Teste de Pré-Release não pode ser feito se não existem testes de aceitação disponíveis.
2. O software de trabalho é necessário para a realização dos Testes de Pré-lançamento. Isto significa que deve haver algo para testar, se as estórias para a iteração não foram terminadas e não há nenhuma funcionalidade no sistema, o teste de pré-release não pode ser feito.

5.2 Entrada

1. Os testes de aceitação definidos no Dia de Planejamento utilizando o padrão Teste de Aceitação são executados no teste de pré-release.

6. Outputs

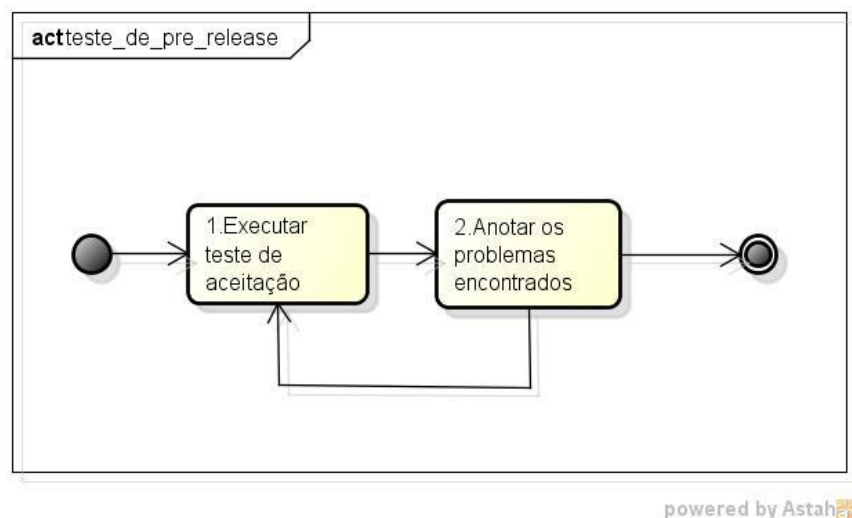
6.1 Pós-condições

1. Todos os testes de aceitação que foram escritos para o sistema são executados. Também os testes de aceitação que foram escritos para as iterações anteriores são executados para garantir a funcionalidade de todo o sistema.
2. Os defeitos encontrados são documentados para uso posterior.

6.2 Saída

1. A **Lista de defeitos** é atualizada para conter todos os defeitos encontrados.

7. Diagrama da tarefa: Teste de Pré Release



Atividade: Dia de Release	Versão: 0.1
Tarefa: Teste de Pré Release	Data: 19/12/2012
LMP-P03.3.4	

8. Passos

Os passos individuais para o Teste de Pré Release são:

1. **O Teste de Aceitação é executado.** Isto é feito para se certificar de que o software está pronto para o Teste de aceitação e que grandes defeitos não sejam encontrados.
2. **Todos os defeitos são anotados.** Todas as questões são documentadas na lista de defeitos para uso posterior. A lista pode ser utilizada em testes de aceitação para fornecer as informações aos clientes dos possíveis defeitos no software, de modo que ele não tenha de testar um sistema com defeito.

9. Templates

1. **Documento de Teste de Aceitação.** Um documento onde todos os testes de aceitação são armazenados. O modelo possui campos para identificação do teste, número e nome de estória, data em que o teste foi escrito, data em que o teste foi executado, se os testes passaram, os insumos e os resultados esperados.
2. **Lista de Defeitos.** Uma lista com todos os defeitos encontrados, a data e a tarefa onde foram corrigidos. A lista permite que os dados a seguir possam ser gravados a partir de todos os erros encontrados: o número, a gravidade, a descrição, a fase de testes onde foi encontrado, a data em que foi encontrado, o ID da tarefa onde foi fixado e observações.

10. Papéis

1. Os **Membros da equipe do projeto** executam os testes de aceitação e anotam os defeitos encontrados
2. O **Gerente de Projeto ou o Engenheiro de Testes** responsável irá dividir os testes de aceitação para cada desenvolvedor executar.

11. Padrões Relacionados

Outros padrões que fazem parte deste ou são associados com a atividade são identificados abaixo:

- **Teste de Pré-Release:** faz parte do Dia de Release
- **Teste de Aceitação:** Teste de pré-release é realizado com o Teste de Aceitação

Atividade: Dia de Release	Versão: 0.1
Tarefa: Teste de Pré Release	Data: 19/12/2012
LMP-P03.3.4	

12. Riscos

Os possíveis riscos que podem resultar de **Teste de Pré Release**, bem como as soluções incluindo ações preventivas para evita-los e medidas a tomar para minimizar seus efeitos são discutidos aqui:

- **Quando o defeito é encontrado no teste de pré-release a equipe não tenta consertá-lo antes do Teste de Aceitação.** Geralmente não há tempo suficiente para corrigir o defeito e testar o software novamente para encontrar os possíveis defeitos introduzidos na correção.
- **Nem todos os defeitos são encontrados durante os testes de pré-release.** É possível que todos os defeitos não sejam encontrados no teste, embora o objetivo da prática seja encontrar todos os defeitos, raramente é possível que o sistema seja completamente testado.

13. Referências

- <http://agile.vtt.fi/mobiled.html>