
Processo de Gerência de Configurações

Lucas Joaquim, Josiel Faleiros e Rafael Santos
Universidade Tecnológica Federal do Paraná - Câmpus Cornélio Procópio

Sumário

1	Introdução	3
1.1	Propósito do processo de gerência de configuração	3
2	Práticas Gerais	3
3	Baselines	3
3.1	Identificando Items de Configuração	3
3.2	Baselines do projeto	4
4	Controle de Mudanças	4
4.1	Ferramentas	4
4.2	Repositórios	4
4.3	Políticas de versionamento	5
4.4	Auditoria	5
4.5	Processo para adesão de novos desenvolvedores	5
4.6	Processo para adicionar novas funcionalidades	6
4.7	Processo para fazer atualizações	7
4.8	Processo para correção de bugs	8
4.9	Processo para criação de distribuições	9
5	Controle dos clientes	9

1 Introdução

Nesse documento foi especificado um processo de gerência de configuração para a empresa Software Supimpa Tecnologia (2ST). Usando como base as normas da CMMI-DEV. A empresa possui o produto de software Klassic e conta com 10 desenvolvedores.

1.1 Propósito do processo de gerência de configuração

O processo de gerência de configuração tem como objetivo gerenciar as alterações através de cinco atividades segundo o livro de engenharia de software do Pressman, sendo elas a identificação dos artefatos que precisam ser alterados, estabelecendo relações entre eles, definição de mecanismos para gerenciar diferentes versões desses artefatos, controle das alterações impostas, auditoria e relatório das alterações realizadas.

2 Práticas Gerais

Segundo o livro de Engenharia de Software do Pressman, para identificar todos os itens que coletivamente definem a configuração do software, gerenciar alterações de um ou mais destes itens, facilitar a construção de diferentes versões de uma aplicação e assegurar que a qualidade do software seja mantida a medida que a configuração evolui, é necessária a execução das 5 tarefas listadas abaixo:

- Identificação.
- Controle de alterações.
- Controle de versão.
- Auditoria de configuração.
- Relatos.

3 Baselines

Nessa seção serão definidos identificados os itens de configuração e as baselines presentes no projeto que serão versionados e controlados.

3.1 Identificando Items de Configuração

Na seguinte tabela constam os itens de configuração e seus respectivos identificadores.

Item	Identificador
Diagrama de Caso de Uso	UC
Diagrama de Classe	CD
Diagrama de Sequência	SD
Diagrama de Pacotes	PD
Planilha clientes software	PCS
Código fonte	CF
Documento LaTeX	TEX

Tabela 1: Itens sob controle

3.2 Baselines do projeto

Baseline	Itens relacionados	status
Plano de Gerência de Configuração	TEX	em desenvolvimento
Documentação	CD, SD, UC, PD, PCS	em desenvolvimento
Produto	Código fonte	em desenvolvimento

Tabela 2: Baseline

4 Controle de Mudanças

4.1 Ferramentas

No ciclo de desenvolvimento do projeto serão utilizadas as seguintes ferramentas para a criação e versionamento dos artefatos, bem como meios para realização de auditoria.

- Git - é um sistema de controle de versão distribuído e um sistema de gerenciamento de código fonte, com ênfase em velocidade.
- Github - é uma plataforma de hospedagem de código-fonte com controle de versão usando o Git.
- Astah - é um software para modelagem uml.
- Bpmn.io - é um software para diagramação online.

4.2 Repositórios

Abaixo constam os repositórios do projeto:

- Documentation - Contém toda a documentação do projeto. <https://github.com/ProdutoInova/Documenta>
- ProdutoInova - Contém todo o código do projeto. <https://github.com/ProdutoInova/Inova.git>.
- modulo-cidade - Contém a implementação do módulo cidade. <https://github.com/ProdutoInova/modulo-cidade.git>.

4.3 Políticas de versionamento

Será usado o seguinte formato para versionamento: 0.0.0. Para o incremento dos números deverão ser seguidas as seguintes políticas:

- O número mais a esquerda é incrementado quando são realizadas uma ou mais mudanças que resultam na perda de compatibilidade com a versão anterior.
- O número do centro é incrementado quando é adicionada uma ou mais funcionalidades, mas mantendo a compatibilidade com a versão anterior.
- O número mais a direita é incrementado quando é realizada correções de bugs, também mantendo a compatibilidade com a versão anterior.

4.4 Auditoria

Será utilizado o audit log do Github para avaliação das ações realizadas pelos membros do projeto. O audit log inclui detalhes como quem executou a ação, qual foi a ação e quando foi realizada.

É possível realizar as seguintes ações:

- Em que repositório uma ação foi realizada.
- O usuário que realizou a ação.
- A ação que foi realizada.
- A data e hora em que a ação ocorreu.

4.5 Processo para adesão de novos desenvolvedores

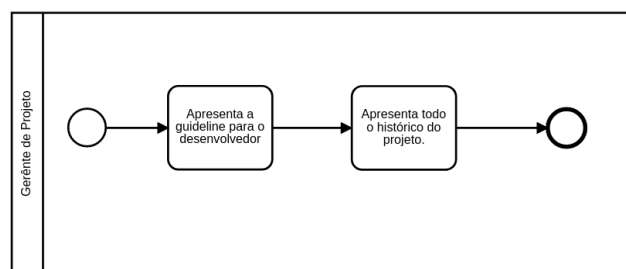


Figura 1: Adesão de novos desenvolvedores.

4.6 Processo para adicionar novas funcionalidades

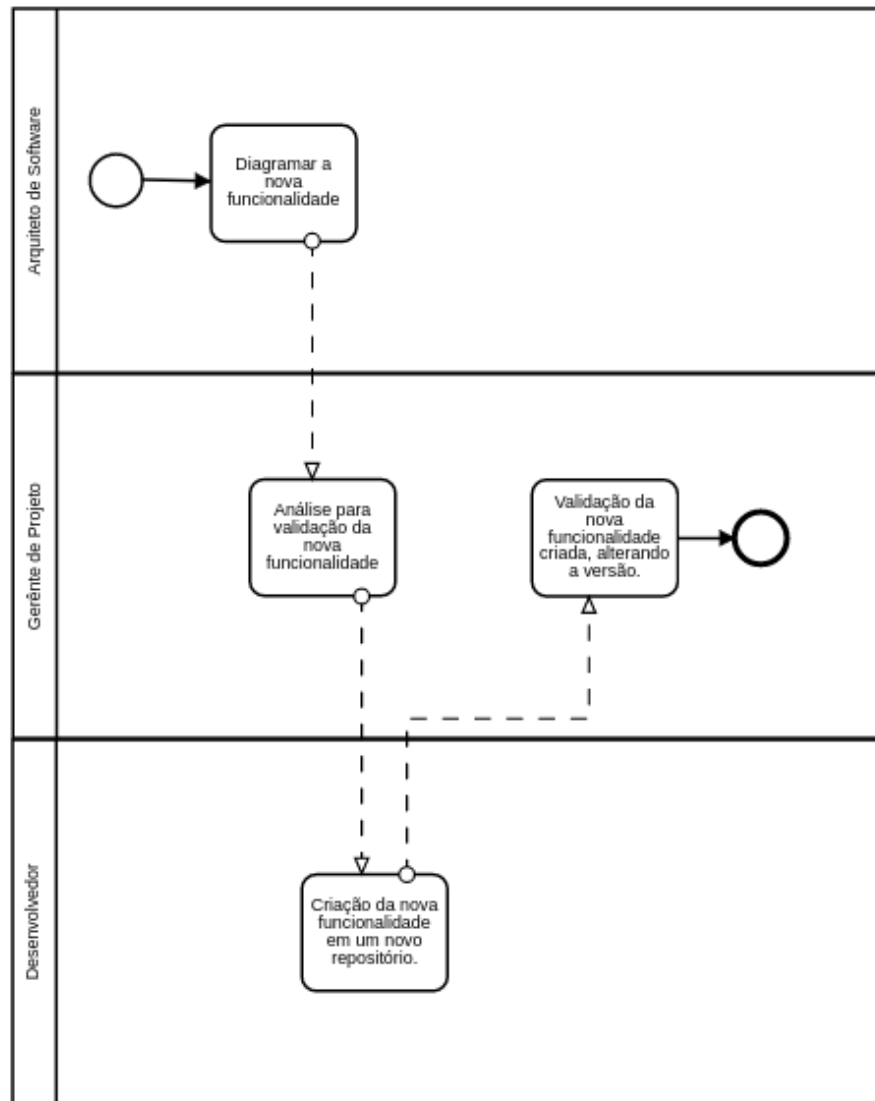


Figura 2: Adição de novas funcionalidades.

4.7 Processo para fazer atualizações

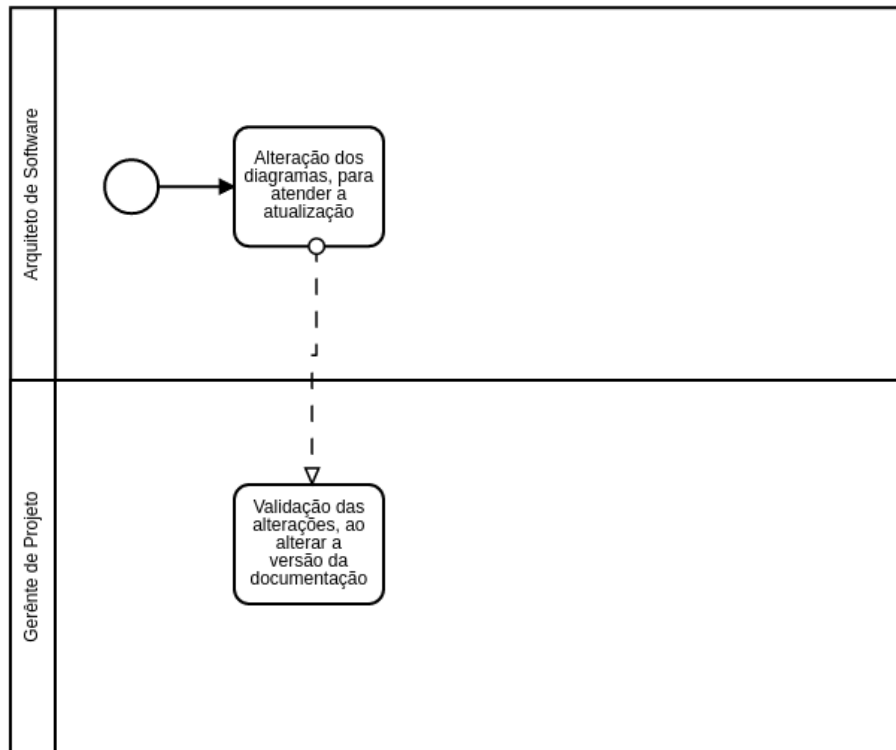


Figura 3: Fazer atualizações.

4.8 Processo para correção de bugs

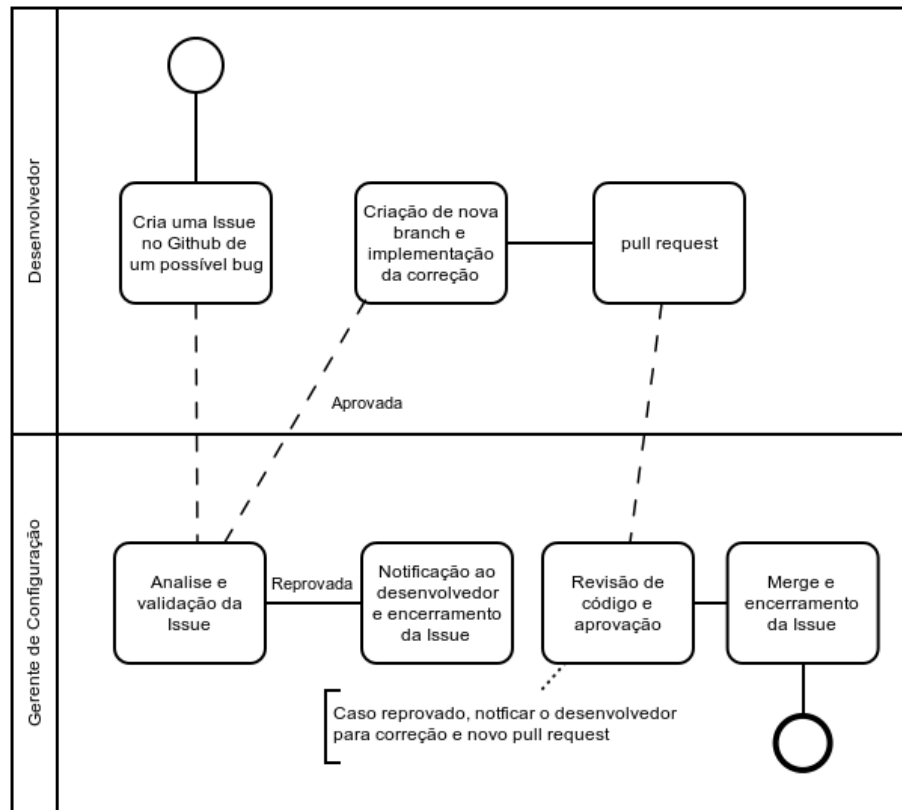


Figura 4: Correção de bugs.

4.9 Processo para criação de distribuições

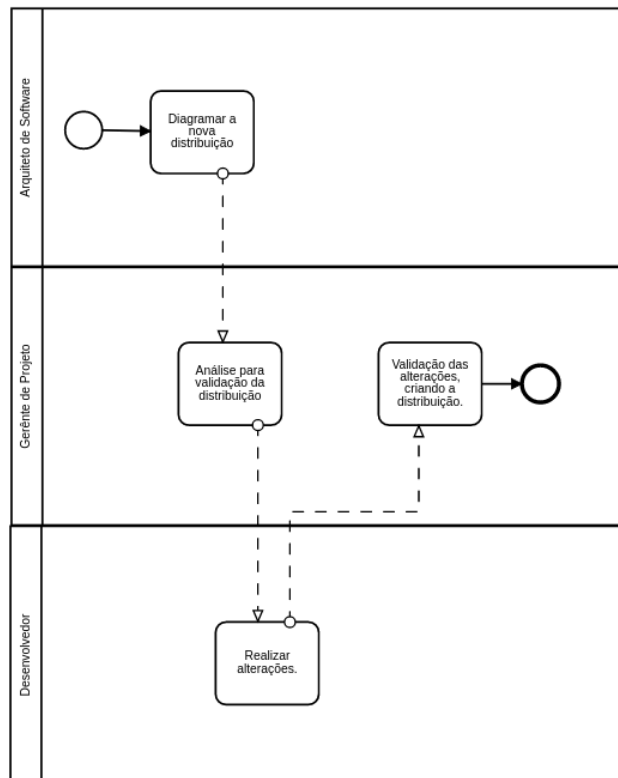


Figura 5: Criação de distribuições.

5 Controle dos clientes

Cada cliente pode possuir funcionalidades diferentes, tendo então uma tabela para controle de quais clientes possuem quais funcionalidades e suas versões.

cliente	baseline version	funcionalidade	versão
cliente a	1.0.0	módulo 1	1.0.0
		módulo 2	1.0.0
cliente b	1.0.0	módulo 2	1.0.0
		módulo 5	1.0.0

Figura 6: Controle clientes x software.

Lista de Figuras

1	Adesão de novos desenvolvedores.	5
2	Adição de novas funcionalidades.	6
3	Fazer atualizações.	7
4	Correção de bugs.	8
5	Criação de distribuições.	9
6	Controle clientes x software.	9

Lista de Tabelas

1	Itens sob controle	4
2	Baseline	4