

Gerador de Burndown Plano de Gerenciamento de Configuração

Versão 1.02

Histórico da Revisão

Data	Versão	Descrição	Autor
13/10/2014	1.0	Criação de documento	Sara Lima
18/10/2014	1.01	Revisão do documento	Wander Cassimiro
18/10/2014	1.02	Alteração final de documento	Sara Lima

Índice Analítico

1. Introdução

1.1 Finalidade

1.2 Escopo

1.3 Definições, Acrônimos e Abreviações

1.4 Referências

1.5 Visão Geral

2. Gerenciamento de Configuração de Software

2.1 Organização, Responsabilidades e Interfaces

2.2 Ferramentas, Ambiente e Infra-estrutura

3. O Programa de Gerenciamento de Configuração

3.1 Identificação da Configuração

3.1.1 Métodos de Identificação

3.1.2 Baselines do Projeto

4. Marcos

5. Controle de Software de Subcontratos e fornecedores

Plano de Gerenciamento de Configuração

1. Introdução

O Plano de Gerenciamento de Configuração contém padrões para manter a integridade do código, descreve atividades, controla mudanças e a evolução do projeto, permitindo acompanhamento do mesmo durante todo seu ciclo de vida.

1.1 Finalidade

Definir padrões a serem seguidos pelos membros envolvidos, com o propósito de garantir maior controle e evolução do sistema ao longo do projeto.

1.2 Escopo

Este Plano de Gerenciamento de Configuração destina-se a todos os integrantes do projeto no desenvolvimento do sistema Gerador de Burndown que abrange todo o controle e gerenciamento da configuração do projeto.

1.3 Definições, Acrônimos e Abreviações

Termo	Descrição
SCRUM	<i>É um processo ágil que permite manter o foco na entrega do maior valor de negócio, no menor tempo possível.</i>
Sprint	<i>Um sprint é a unidade básica de desenvolvimento em Scrum e são um esforço dentro de uma faixa de tempo constante.</i>
Estoria	<i>Produto que precisa ser desenvolvido durante uma Sprint.</i>
PGC	Plano de Gerência de Configuração
GC	Gerência de Configuração
CCM	Controle de Configuração e Mudanças.
Baseline	Grupo de itens de configuração correlatos e íntegros. Serve de base para a próxima etapa de trabalho.

PFM	Pedido Formal de Mudanças
GD	Gerente de Desenvolvimento
LP	Líder do Projeto

1.4 Referências

Template de Plano de Gerenciamento de Configuração

Documentação de código fonte – Gerador de Burndow

1.5 Visão Geral

As próximas seções deste documento estão divididas conforme descrito abaixo.

- **Seção 2** Descreve os papéis, responsabilidades e as ferramentas de software a serem utilizadas na GC;
- **Seção 3** Descreve como os artefatos do projeto devem ser nomeados e criação e divulgação de baselines.
- **Seção 4** Identifica os marcos relacionados ao esforço de CM do projeto.
- **Seção 5** Descreve de que forma o software desenvolvido fora do ambiente do projeto será incorporado.

2. Gerenciamento de Configuração de Software

2.1 Organização, Responsabilidades e Interfaces

<i>Papéis</i>	<i>Responsabilidades</i>
<i>Gerente de Desenvolvimento (GD)</i>	<i>Juntamente com a CCM receber, analisar e aprovar os PFM (Pedido Formal de Mudanças).</i>
<i>Desenvolvedor</i>	<i>Seguir os padrões e procedimentos definidos no Plano de Gerência de Configuração (PGC)</i>
<i>Líder do Projeto (LP)</i>	<i>Planejar as atividades de GC juntamente com o Gerente de Configuração, designar executante, finalizar Sprint, autorizar a criação das configurações pelo Gerente de configuração, checar e coordenar as atividades realizadas juntamente com os desenvolvedores.</i>

<i>Gerente de Configuração</i>	<i>Criar e manter infra-estrutura corporativa (servidores) de GC; Implementar as políticas de Controle de Acesso ao ambiente de GC, Realizar os backups dos repositório de configuração dos projetos, Criar as Baselines</i>
<i>CCM</i>	<i>Estabelecer Processo de Controle de Mudanças. Revisar Solicitação de Mudança (PFM)</i>

2.2 Ferramentas, Ambiente e Infra-estrutura

O ambiente de computação está organizado da seguinte forma:

- Desenvolvimento

É o ambiente que servirá para o desenvolvimento do Sistema. Sua maturidade será atingida quando os requisitos forem supridos e testados pelos desenvolvedores através dos testes unitários.

- Integração

É o ambiente que servirá para os testes de integração. Sua maturidade será atingida quando a comunicação entre os módulos atingir um estágio satisfatório de funcionamento, ou seja, não deverão existir erros de integração entre os subsistemas.

- Banco de Dados

É o ambiente de Banco de dados.

Ferramentas a serem utilizadas no desenvolvimento do projeto:

Software/ Ferramenta	Propósito	Ambiente	Release/Ver são
GIT	Repositório e Controle de Versão	Todos	1.9.4
Source Tree	Suporte ao controle de versão	Desenvolvimento	1.6.5.0
MS-Office	Documentos do Word	Todos	2007, 2010
MS-Project	Edição do cronograma	Desenvolvimento	2007, 2010
Java JDK	Kit básico para o desenvolvimento em Java	Desenvolvimento	
Eclipse	IDE de desenvolvimento	Desenvolvimento	

3. O Programa de Gerenciamento de Configuração

3.1 Identificação da Configuração

3.1.1 Métodos de Identificação

A apresentação dos artefatos que podem ser gerados no processo e suas respectivas identificações.

ID	ARTEFATO
TA	Termo de Abertura
AR	Ata de Reunião
CRO	Cronograma
APD	Ativação de Projeto de Desenvolvimento
ER	Especificação e Requisitos
UC	Detalhamento de Casos de Uso
ARC	Arquitetura do Projeto
PCP	Plano de Comunicação do Projeto
PDS	Plano de Desenvolvimento de Software
PGC	Plano de Gerência de Configuração
PGQ	Plano de Gerência da Qualidade
PGR	Plano de Gerência de Riscos
RAC	Relatório de Avaliação do Cliente
RAP	Relatório de Acompanhamento do Projeto
PFM	Pedido Formal de Modificação
SAI	Solicitação de Análise de Impacto
PTS	Plano de Teste de Software
RT	Relatório de Teste
DAS	Documento de Análise de Software
DPS	Documento de Projeto de Software

3.1.2 *Baselines do Projeto*

As baselines geradas para o projeto só podem ser alteradas mediante procedimentos de controle de mudança (PFM) e a mesma deve ser autorizada pelo GD juntamente com CCM do PGC

<i>Baselines</i>	<i>Descrição</i>
Requisitos	Marcado assim que for concluída análise de requisitos e prototipação da iteração.
Análise e projeto	Marcado quando concluídos a análise, o roteiro de testes o projeto de cada iteração.
Build	Criada a cada build para o software.
Codificação	Marcada assim que concluída a parte de codificação e testes de unidade.
Arquitetura	Marcada assim que concluída a arquitetura do projeto.
Homologação	Marcada assim que concluído o desenvolvimento do projeto.

4. **Marcos**

A cada 15 dias serão feitas entregas e conferências dos marcos através de baselines.

Marcos internos:

Requisitos

Codificação

Testes

Homologação

Produção

O Plano CM deve ser atualizado a cada 15 dias.

5. **Controle de Software de Subcontratados e Fornecedores**

N/A