Academicci

Plano de Gerência de Configuração

Versão 0.000

Histórico de Revisão

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 28/02/18 | 0.000 | Criação do Documento. | Raphael Guedes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sumário

1. Introdução 4

1.1 Finalidade 4

1.2 Escopo 4

1.3 Referências 4

1.4 Visão Geral 4

2. Gerência de Configuração de Software 4

2.1 Organização, Responsabilidades e Interfaces 4

2.2 Ferramentas, Ambiente e Infraestrutura 4

2.2.1 Ferramentas 5

2.2.2 Ambiente 5

2.2.3 Infraestrutura 5

3. O Programa de Gerenciamento de Configuração 6

3.1 Identificação da Configuração 6

3.1.1 Métodos de Identificação 6

3.1.2 Métodos de Versionamento 7

3.1.3 Baselines do Projeto 7

3.1.4 Estrutura do Repositório 7

3.2 Controle de Configuração e Mudança 8

3.2.1 Processamento e Aprovação de Solicitações de Mudança 8

3.2.2 Comitê de Controle de Mudança (CCB) 8

3.3 Estimativa de Status da Configuração 9

3.3.1 Processo de Armazenamento de Mídia e Liberação do Projeto 9

4. Marcos 9

4.1 Alteração do Plano de Gerência de Configuração 9

5. Treinamento e Recursos 9

Plano de Gerência de Configuração

# Introdução

## Finalidade

O objetivo deste documento é criar um padrão a ser seguido por todos os membros da equipe com o intuito de garantir o maior controle do produto no decorrer do projeto.

Para que isso aconteça serão detalhados os recursos necessários (equipes, ferramentas e ambiente), as responsabilidades atribuídas e o cronograma de atividades.

## Escopo

Este Plano de Gerência de Configuração é destinado para todos os integrantes da equipe de desenvolvimento do projeto **Academicci**, e abrange o controle e gerenciamento da configuração do projeto **Academicci**.

## Referências

* Disciplina de Gestão de Configuração e Mudanças OpenUP v 1.0. Disponível em: <http://www.hectordufau.com.br/openup/index.htm>
* Documento de Glossário. Disponível em
* Documento de Plano de Projeto
* Documento de Arquitetura de Software
* Sistema de Versionamento SEMVER. Disponível em: <https://semver.org/lang/pt-BR/>

## Visão Geral

A primeira seção introduz o seu objetivo, escopo e documentos que servem de referência para seu entendimento.

# Gerência de Configuração de Software

## Organização, Responsabilidades e Interfaces

|  |  |
| --- | --- |
| **Função** | **Responsabilidades** |
| Scrum Master (Gerente de Projeto) | * Gerenciar as atividades referentes ao projeto. * Garantir que o processo seja executado da forma correta. * Remover os impedimentos do projeto. |
| Product Owner (Gerente de Configuração) | * Elaborar e manter as Políticas de Gerência de Configuração; * Controlar a estrutura do diretório, garantindo sua devida organização |

## Ferramentas, Ambiente e Infraestrutura

[Descreva o ambiente de computação e as ferramentas de software a serem utilizadas para desempenhar as funções de CM em todo o ciclo de vida do projeto ou produto.

Descreva as ferramentas e os procedimentos necessários utilizados para o controle de versão dos itens de configuração gerados no ciclo de vida do projeto ou produto.

As questões envolvidas na configuração do ambiente de CM incluem:

* tamanho previsto dos dados do produto
* distribuição da equipe do produto
* localização física dos servidores e clientes]

### Ferramentas

|  |  |  |
| --- | --- | --- |
| **Ferramenta** | **Versão** | **Descrição** |
| Astah Community | *Versão da ferramenta utilizada no desenvolvimento* | Modelagem de Diagramas UML |
| Astah Professional | *Versão da ferramenta utilizada no desenvolvimento* | Modelagem de Diagramas UML |
| Corel Draw X7 | *Versão da ferramenta utilizada no desenvolvimento* | Ferramenta de Edição de Imagens Vetoriais |
| Git | *Versão da ferramenta utilizada no desenvolvimento* | Controle de Versão |
| *IDE de desenvolvimento* | *Versão da ferramenta utilizada no desenvolvimento* |  |
| Java SE Development Kit | *Versão da ferramenta utilizada no desenvolvimento* | Ambiente de Desenvolvimento da Aplicação |
| Microsoft Office Excel | *Versão da ferramenta utilizada no desenvolvimento* | Editor de Planilhas |
| [Photoshop CC](https://www.adobe.com/br/products/photoshop.html) | *CC 2017* | Ferramenta de Edição de Imagens |
| *Servidor de Aplicação* | *Versão da ferramenta utilizada no desenvolvimento* |  |
| [Target Process](https://www.targetprocess.com/) | Mais recente | Ferramenta de Controle do Projeto |
| [WhatsApp](https://www.whatsapp.com/) | Android 5.0+ | Ferramenta de Comunicação |
| Windows | 10 Pro | Sistema Operacional |
| Pencil |  | Ferramenta para prototipação de interfaces |

### Ambiente

O sistema de desenvolvimento deve ter a configuração mínima de:

* 4 GB de Memória RAM.
* 20 GB de espaço livre em disco.
* Conexão com a internet.

O ambiente de execução deve ter a configuração mínima de:

* Navegador que suporte HTML 5, CSS 3 e JavaScript.
* Memória RAM mínima, necessária para suportar a execução do navegador de internet
* Conexão com internet.

### Infraestrutura

# O Programa de Gerenciamento de Configuração

## Identificação da Configuração

### Métodos de Identificação

Todos os artefatos gerados, com exceção de código fonte, neste projeto terão a seguinte nomenclatura:

<PROJECT>\_<ARTEFACT\_ID>\_<Artefact\_SubName>.<EXT>

**Exemplo:** MeuProjeto\_DAS\_DocumentoArquitetura

Onde:

|  |  |
| --- | --- |
| **Identificador** | **Descrição** |
| <PROJECT> | Nome do projeto em desenvolvimento, em caixa alta [obrigatório] |
| <ARTEFACT\_ID> | Identificador do artefato de acordo com a tabela de identificadores, em caixa alta [obrigatório] |
| <Artefact\_SubName> | Nome do artefato em particular. Somente necessário caso haja um nome em específico, como no caso de UCEs. [opcional] |
| <EXT> | Extensão do arquivo |

**Observações:** Não deve haver acentuação ortográfica em nenhum dos termos que compõem o nome do item de configuração. E os identificadores são compostos de 2 a 4 letras que resumem o nome do artefato.

|  |  |
| --- | --- |
| **Artefato** | **ID** |
| Checklist Scrum | **CHKS** |
| Diagrama de Casos de Uso | **DCU** |
| Diagrama de Classes | **DCL** |
| Diagrama Entidade-Relacionamento | **DER** |
| Documento de Arquitetura de Software | **DAS** |
| Documento de Casos de Teste | **DCT** |
| Documento de Especificação de Interfaces | **DEI** |
| Documento de Especificação de Mensagens | **DEM** |
| Documento de Glossário | **DGLO** |
| Documento de Planejamento do Sprint | **DPS** |
| Documento de Plano de Projeto | **DPP** |
| Documento de Regras de Negócio | **DRN** |
| Documento de Visão | **DVIS** |
| Gráfico Burndown do Projeto | **GBP** |
| Gráfico Burndown do Sprint | **GBS** |
| Lista de Riscos | **LR** |
| Modelo Entidade-Relacionamento | **MER** |
| Modelo de Processo | **MDP** |
| Plano de Gerência de Configuração | **PGC** |
| Product Vision Box | **PVB** |
| Script de Teste | **SPT** |
| Template | **TPT** |

### Métodos de Versionamento

**Documentação:**

O versionamento dos documentos será feito através de um identificador único atribuído ao documento. Como a seguir:

X.YYY

**Exemplo:** 5.45

Onde:

**X**: número inteiro que representa uma versão da baseline atual do sistema;

**YYY**: número decimal que representa a versão de edição do artefato.

O número de artefatos muda de acordo com as regras descritas a seguir:

* A primeira versão do artefato recebe versão 0.000;
* A cada modificação no artefato, o valor de YYY deve ser incrementado em uma unidade.
* Após cada baseline, a versão X deve ser incrementada em uma unidade.

Para que a versão de um artefato seja modificada é necessária a aprovação da pessoa com papel responsável.

**Código:**

Ver SEMVER (Item *1.3 Referências*).

### Baselines do Projeto

As Baselines do projeto serão definidas ao término de cada Sprint. A fim de garantir o acompanhamento do progresso de desenvolvimento do projeto. As baselines do projeto estarão disponíveis na plataforma de compartilhamento de código Open Source, GitHub.

Cada baseline conterá uma imagem da versão corrente do sistema, definida no momento de sua criação;

### Estrutura do Repositório

O repositório utilizado no projeto **Academicci** será o GitHub (Ver item 1.3 Referências) e toda os itens de configuração estarão disponíveis para consulta publicamente.

A organização do diretório da documentação segue a estrutura de disciplinas do framework open source OpenUP.

|  |  |
| --- | --- |
| **Diretório** | **Conteúdo** |
| **.** | README do Projeto . |
| **.** /<Nome\_Projeto> | Diretório de código fonte do sistema. |
| **.** /documentacao | Artefatos relacionados à documentação do sistema. |
| **.** /documentacao/arquitetura | Artefatos relacionados à arquitetura do sistema. |
| **.**/documentacao/gerencia\_config | Artefatos relacionados à gerência de configuração do sistema. |
| **.** /documentacao/processo | Artefatos relacionados ao processo de desenvolvimento e gerência do projeto. |
| **.** /documentacao/requisitos | Artefatos relacionados aos requisitos do sistema. |
| **.** /documentacao/templates | Modelos dos artefatos utilizados no projeto. |
| **.** /documentacao/teste | Artefatos relacionados à execução dos diferentes níveis de teste do sistema. |

## Controle de Configuração e Mudança

### Processamento e Aprovação de Solicitações de Mudança

As solicitações de mudança serão controladas a partir da funcionalidade de ISSUES, disponível na plataforma do GitHub.

As solicitações serão realizadas e de acordo com o fluxo a seguir serão aprovadas ou rejeitadas:

* O membro do time abre uma issue no GitHub e atribui uma classificação;
* O Comitê de Controle de Mudança avalia os impactos da efetuação da mudança no projeto;
  + Caso seja aprovado, a mudança é realizada.
    - O sistema é testado após a mudança.
    - Caso seja gerada alguma inconsistência, o sistema é retornado para o estado anterior à mudança;
  + Caso seja reprovado, a mudança não é realizada.

### Comitê de Controle de Mudança (CCB)

O CCB será formado pelo **Scrum Master** o **Product Owner** e um **Representante do Time de Desenvolvimento**.

O CCB tem a responsabilidade de avaliar, aprovar ou rejeitar as solicitações de mudanças surgidas durante o desenvolvimento do sistema.

## Estimativa de Status da Configuração

### Processo de Armazenamento de Mídia e Liberação do Projeto

O repositório do projeto será mantido em nuvem através do GitHub e uma cópia atualizada estará disponível para consulta em uma pasta compartilhada no Google Drive

# Marcos

Os marcos do projeto estão definidos como a finalização dos sprints, no qual é criada uma nova baseline.

## Alteração do Plano de Gerência de Configuração

Este artefato sofrerá alteração quando a estrutura do diretório não atender, satisfatoriamente, o desenvolvimento do projeto. Todavia a sua alteração deverá ser aprovada pelo CCB.

# Treinamento e Recursos

Para o bom aproveitamento do trabalho de Gerência de Configuração é necessário que toda a equipe saiba:

* Os fundamentos do controle de versão de itens de configuração de software.
  + Disponível em: PRESSMAN, Roger S. Engenharia de Software: uma abordagem profissional. 7. ed. São Paulo: Pearson Makron Books, 2011.
* Conhecimento do cliente de controle de versão Git
  + Disponível em: <https://git-scm.com/book/pt-br/v2>
* Conhecimento da linguagem de marcação Markdown;
  + Disponível em: <https://daringfireball.net/projects/markdown/>
* Fundamentos da utilização da ferramenta GitHub;
  + Disponível em: <https://guides.github.com/activities/hello-world/>