

Plano de Gerência de

Configuração de Software

Projeto: SA – Sistema Academico

|  |  |
| --- | --- |
| Data | 25/09/2020 |
| Responsável | Maycon Satel |
| ID do documento | SA\_SCMP\_20200925 |
| Localização | <https://github.com/satel61/sistemaacademico> |
| Versão do Documento | 2.00 |

# Histórico de Revisões

|  |  |  |  |
| --- | --- | --- | --- |
| Data | Versão | Autor | Descrição |
| 25/09/2020 | 1.00 | Maycon Satel | Criação Inicial |
| 28/09/2020 | 2.00 | Maycon Satel | Definição da politica de branch |

Revisores

|  |  |  |
| --- | --- | --- |
| Nome | Função | Data |
| Maycon Satel | Gerente de Projeto | 28/09/2020 |

Sumário

[1. Introdução 5](#_Toc52103837)

[1.1. Objetivos 5](#_Toc52103838)

[1.2. Escopo 5](#_Toc52103839)

[1.3. Glossário 5](#_Toc52103840)

[1.4. Referências 5](#_Toc52103841)

[2. Organização 5](#_Toc52103842)

[2.1. Identificação de Documentos 5](#_Toc52103843)

[2.2. Versão de Documentos 6](#_Toc52103844)

[2.3. Localização de Artefatos 7](#_Toc52103845)

[2.4. Baselines do Projeto 8](#_Toc52103846)

[2.5. Branches 8](#_Toc52103847)

[2.5.1. Branches de Documento 8](#_Toc52103848)

[2.5.2. Branches de Código 9](#_Toc52103849)

[3. Controle de Configuração 12](#_Toc52103850)

[3.1. Procedimentos de Mudança 12](#_Toc52103851)

[3.1.1. Criando Solicitação de Mudança (CRs) 12](#_Toc52103852)

[3.1.2. Ciclo de vida das Solicitações de Mudança (CRs) 12](#_Toc52103853)

[4. Auditoria de Configuração 14](#_Toc52103854)

[5. Plano de Contingência 14](#_Toc52103855)

[6. Ferramentas 15](#_Toc52103856)

## Introdução

A intenção do Gerenciamento de Configuração é estabelecer e manter a integridade dos produtos do projeto durante seu ciclo de vida. As principais atividades envolvem identificar a configuração de software, manter sua integridade durante o projeto e controlar sistematicamente as mudanças. Esse plano contém todas as informações referentes ao sistema de gerencia de configuração para o projeto SA (Sistema Academico).

### Objetivos

Este documento tem por objetivo principal descrever a organização, nomenclatura e regras para versionamento do projeto SA.

### Escopo

Este documento detalha toda a infra-estrutura utilizada durante o desenvolvimento do projeto SA.

### Glossário

**Tabela 1 — Glossário**.

|  |  |
| --- | --- |
| **Termo** | **Descrição** |
| **Baseline** | Conjunto de artefatos que recebe uma aprovação de estabilidade. Um baseline é usado como uma base no desenvolvimento das próximas fases dos artefatos e tem suas modificações controladas por um  processo. |
| **CR** | Solicitação de Mudança (Change Request) |
| **CVS** | Sistema de Controle de Versão (Control Version System) |
| **SCMP** | Plano de Gerência de Configuração de Software (Software Configuration Management Plan) |

### Referências

1. Open eXperience Environment Process. Disponível em: [http://php.cin.ufpe.br/~oxe/index.php?option=com\_content&task=view&id=4&It](http://php.cin.ufpe.br/%7Eoxe/index.php?option=com_content&amp;task=view&amp;id=4&amp;It) emid=10
2. WinCVS. Disponível em [www.wincvs.org,](http://www.wincvs.org/) último acesso: maio/2005.
3. Eclipse. Disponível em [www.eclipse.org,](http://www.eclipse.org/) último acesso: maio/2005.
4. Modelo de Relatório de Auditoria de Gerência de Configuração de Software - SA\_TRK\_20200925\_SCM\_AUDIT.xls.
5. Simulare Software Configuration Management Plan. Disponível em: [http://xsimulare.tigris.org/site/Documents.htm,](http://xsimulare.tigris.org/site/Documents.htm) último acesso: maio/2005.

## Organização

### Identificação de Documentos

Todos os itens de configuração (exceto código fonte) devem ser identificados baseados na nomenclatura descrita a seguir:

<PROJETO>\_<ID\_ARTEFATO>

em que:

<PROJETO> é o nome do projeto;

<ID\_ARTEFATO> é a identificação do artefato (ver Tabela 2);

É importante salientar que todas as letras dos nomes que compõem os documentos devem estar em caixa alta.

A Tabela 2 apresenta os artefatos que podem ser gerados no processo e suas respectivas identificações.

Tabela 2 — Artefatos e suas respectivas identificações.

|  |  |
| --- | --- |
| Artefato | Identificação |
| ESPECIFICACAO DO SISTEMA | SAES |
| ESPECIFICACAO DE REQUISITOS DE SISTEMA | SAERS |
| ESPECIFICACAO DE PROJETO | SAEP |
| PROGRAMA EXECUTAVEL E MODULOS INTERLIGADOS | SAPEMI |
| DESCRICAO DO BANCO DE DADOS | SADBD |
| ESQUEMA E ESTRUTURA DE ARQUIVO | SAEEA |
| CONTEUDO INICIAL | SACI |
| DOCUMENTOS DE MANUTENCAO | SADM |
| PEDIDOS DE MUDANCA | SAPM |
| FERRAMENTAS DE PRODUCAO DE SOFTWARE | SAFPS |

### Versão de Documentos

Esta seção descreve padrões que definem o versionamento dos artefatos. Todos os artefatos devem ter um número de versão segundo o padrão descrito a seguir:

X.Y

em que:

X é um número decimal que representa uma versão final do artefato;

Y é um número decimal que representa um draft da versão X do artefato.

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

* A primeira versão do artefato deve ser 1.0;
* A cada modificação no artefato, o valor Y deve ser incrementado;
* Após cada aprovação do artefato, a versão X deve ser incrementada de uma unidade e o valor Y retorna para 0, sendo assim gerada uma nova versão oficial;

Para que a versão de um artefato seja modificada é necessária a aprovação do moderador que por sua vez deve verificar se o retrabalho (rework) dos issues levantados nas inspeções foi feito corretamente.

### Localização de Artefatos

Esta seção descreve a localização onde os artefatos estão armazenados. É apresentada uma versão de visão gráfica dos diretórios baseado na estrutura de diretórios em que cada artefato deve ser armazenado.



Figura 1 — Estrutura de diretórios do CVS.

Tabela 3 — Detalhamento da estrutura de diretórios do CVS.

|  |  |
| --- | --- |
| Diretórios | Conteúdo |
| codigo | Diretórios relacionados ao código fonte, bibliotecas utilizadas, scripts para compilação e documentação. |
| src | Código fonte |
| Doc gerencia configuracao | Documentos relacionados a gerencia de configuracao |
| Doc gerencia de projeto | Documentos relacionados à gerência de projeto. |
| Doc projeto | Documentos relacionados ao projeto. |
| Banco de dados | Documentos relacionado ao banco de dados. |

### Baselines do Projeto

Esta seção descreve quais baselines são geradas para o projeto.

Tabela 4 — Baselines do Projeto.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Baseline | Descrição | | | | Padrão |
| Requisitos | Deve ser marcado assim que for concluída a análise de requisitos  da iteração. | | | | SA\_REQ\_<iteração> |
| Análise e Projeto | Deve ser marcado assim que forem concluídos a análise e o  projeto de cada iteração. | | | | SA\_ARCH\_ <iteração> |
| Build | Criado a cada geração de build para o software. | | | | SA\_BUILD\_<build> |
| Release | Criado a  software. | cada | release | do | SA\_RELEASE\_<versão> |
| Documentos | Criado após a aprovação de um documento. | | | | Ver Seção 2.1 |

em que:

<iteração> é o número da iteração, sendo utilizados para identificação dois dígitos começando em 01 e sendo incrementado de uma unidade a cada nova iteração.

<build> é o número da build, sendo utilizados para a identificação três dígitos começando em 001 e sendo incrementado de uma unidade a cada nova build.

<versão> é o número da versão lançada (ver seção 2.2).

### Branches

Esta seção descreve as políticas de branches utilizada no processo. A política de branches para documentação é descrita na Seção 2.5.1 e a política para branches de código é descrita na Seção 2.5.2.

#### Branches de Documento

Com o intuito de possuir uma melhor dinâmica de mudanças de documento, é importante que tenhamos uma boa política de branches. Este documento servirá de base para criação de branches e como serão organizadas.

Aqui será descrito como funcionará as branches do projeto:

**Branch master** – Deve ter numero de versao segundo o padrao descrito a seguir:

B.CC

Em que:

B é um numero inteiro que representa uma versao final do documento.

CC é um numero decimal que representa um draft da versao B do documento.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 1.01;
* Após as modificacoes das features e aprovacao de um novo release o valor de CC deve ser incrementado em uma unidade.

**Branch develop**: – Deve ter numero de versao segundo o padrao descrito a seguir:

B.CC.DD

Em que:

B é um numero inteiro que representa uma versao final do documento.

CC é um numero decimal que representa um draft da versao B do documento.

DD é um numero decimal que representa um draft da versao CC do documento.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 1.01.01;
* Após as modificacoes do documento o valor de DD deve ser incrementado em uma unidade.

**Branches feature** - Deve ter numero de versao segundo o padrao descrito a seguir:

B.CC.DD.EE

Em que:

B é um numero inteiro que representa uma versao final do documento.

CC é um numero decimal que representa um draft da versao B do documento.

DD é um numero decimal que representa um draft da versao CC do documento.

EE é um numero decimal que representa um draft da versao DD do documento.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 1.01.01.01;
* Após as modificacoes do documento o valor de EE deve ser incrementado em uma unidade.

#### Branches de Código

Com o intuito de possuir uma melhor dinâmica de mudanças de código, é importante que tenhamos uma boa política de branches. Este documento servirá de base para criação de branches e como serão organizadas.

Aqui será descrito como funcionará as branches do projeto:

**Branch master** – Deve ter numero de versao segundo o padrao descrito a seguir:

XX

Em que:

XX é um numero inteiro que representa uma versao final do codigo.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 01;
* Após as modificacoes das features e aprovacao de um novo release o valor de XX deve ser incrementado em uma unidade.

**Branch develop**: Deve ter numero de versao segundo o padrao descrito a seguir:

XX.YY

Em que:

XX é um numero inteiro que representa a ultima versao do release master

YY é um numero inteiro que representa um draft da versao XX do codigo.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 01.01;
* A cada modificação no codigo, o valor de YY deve ser incrementado em um unidade.
* XX deve ser o numero atual da release em branch master, jamais podera ser valores de versoes antigas de release.

**Branches feature** - Deve ter numero de versao segundo o padrao descrito a seguir:

XX.YY.ZZZ

Em que:

XX é um numero inteiro que representa a ultima versao do release master

YY é um numero inteiro que representa a ultima versao do branch develop

ZZZ é um valor alfabetico que representa um draft da versao X.Y do codigo.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 01.01.AAA;
* A cada modificação no codigo, o valor de ZZZ deve ser incrementado na proxima letra do alfabeto.
* X deve ser o numero atual da release em branch master, jamais podera ser valores de versoes antigas de release.

**Branches hotfix** - : Deve ter numero de versao segundo o padrao descrito a seguir:

XX.WWW

Em que:

XX é um numero inteiro que representa a ultima versao do release master.

WWW é um numero inteiro que representa um draft da versao XX do codigo.

O numero de versao do codigo muda de acordo com as regras descritas a seguir:

* A primeira versao do codigo deve ser 01.001;
* A cada modificação no codigo, o valor de WWW deve ser incrementado em um unidade.

## Controle de Configuração

### Procedimentos de Mudança

Mudanças nos itens de configuração do projeto devem estar sempre associadas a uma ou mais solicitações de mudanças. Dessa forma, para toda e qualquer modificação nos itens de configuração, uma CR deve ser aberta conforme descrito na seção 3.1.1.

#### Criando Solicitação de Mudança (CRs)

As solicitações de mudanças devem ser criadas através da modelo pelo link [formulariodemudanca.xlsx](formulario%20de%20mudanca%20pbl%2004.xlsx). O preenchimento dos campos deve ser feito seguindo as orientações descritas a seguir.

Tabela 5 — Campos e valores a serem preenchidos em uma CR.

|  |  |
| --- | --- |
| Campo | Valor |
| Projeto | SA |
| Solicitante da mudança | Pessoa responsavel |
| Mudança Solicitada | Descrição do que sera mudado |
| Numero | Numero de controle da mudança |
| Data | Data atual da solicitacao |
| Analista de Mudança | Responsavel pela avaliação |
| Componentes Afetados | Quais artefatos serao afetados pela mudança |
| Avaliação da Mudança | Descrição da Avaliação da Mudança |
| Prioridade da Mudança | Descricao do nivel de esforço |
| Esforço Estimado | Descrição do tempo para mudança |
| Decisão | Descrição da decisao tomada |
| Comentarios | Descrição adicional |

#### Ciclo de vida das Solicitações de Mudança (CRs)

Esta seção descreve os possíveis estados, as possíveis soluções e o ciclo de vida de uma de uma solicitação de mudanças. A tabela a seguir, descreve os estados possíveis para uma solicitação de mudanças (CR).

Tabela 6 — Estados possíveis para uma CR.

|  |  |  |
| --- | --- | --- |
| Estado/  Responsável | Descrição do estado | Atividades realizadas |
| NOVO  Qualquer pessoa1 | Uma CR foi especificada ou re- atribuída. | * CR é atribuída. * Colaborador aceita a CR. |
| INICIO  Desenvolvedor | Uma CR foi aceita pelo responsável e será  implementada. | * Executar as modificações solicitadas. |
| RESOLVIDO  Desenvolvedor | Uma CR foi resolvida e está esperando pela verificação do SQA. | * Modificações são validadas conforme requerido pelo processo.   + Revisão, para documentos.   + Testes, para código. |
| VERIFICADO  Testado/SQA | Modificações foram aceitas pela validação (revisão ou teste). | * Disponibilização do documento.   + Publicação de documento.   + Geração de build. |
| FECHADO Gerente de Configuração | CR foi completa, mas não necessariamente as modificações solicitadas foram realizadas.  Caso a mudança tenha sido aceita, esta deve estar disponível (publicada/release feito) para que a CR possa ser  fechada. | Nenhuma ação já que este estado é final. Caso a solução não seja satisfatória, uma nova CR deve ser aberta (possivelmente referenciando esta CR). |
| REABERTO | A CR foi resolvida, mas a solução adotada foi julgada  incorreta. | Ver estado NOVO. |

O Ciclo de uma nova CR segue o fluxo presente na Figura 2. Note que é durante o estado “Assigned” que são feitas: análise de requisito, análise e projeto, codificação e testes unitários da mudança.

Tabela 7 — Possíveis estados finais para solução de CRs.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Resolução | Descrição | | | | |
| CONCERTADO | Uma solução para repositório e testada. | o issue | é | checada | no |
| DUPLICADO | O problema é uma  existente. | duplicata | de | um issue | já |
| TRABALHODURO | Todas as tentativas de reproduzir o erro não tiveram sucesso, mesmo depois de seguir os passos descritos na CR. Se mais informações aparecerem depois, essa CR pode ser re-asignada  para outro colaborador. | | | | |
| INVALIDO | O problema descrito não é um issue. | | | | |

O Ciclo de uma nova CR segue o fluxo na Figura 2. Durante o estado submetida que são feitas: análise de requisito, análise e projeto,codificação e testes unitários.

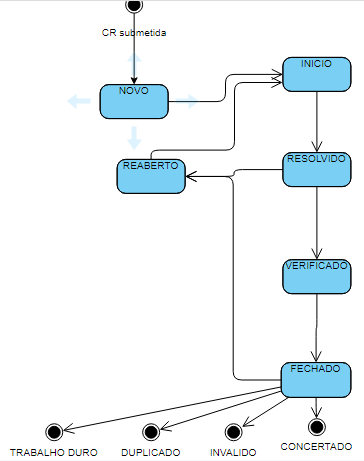


Figura 2 — Máquina de estados de uma CR.

## Auditoria de Configuração

As auditorias de configuração devem ser feitas para cada ciclo do processo de desenvolvimento de forma a garantir que o processo de gerência de configuração vem sendo aplicado corretamente. Os artefatos gerados baseados no template

[4] devem ser armazenados no repositório do projeto e devem ser acompanhados pelos Gerentes de Qualidade e pelos Gerentes de Projeto.

## Plano de Contingência

Uma vez por semana será feito um backup (mirror) da versão mais recente dos artefatos que se encontram no CVS na máquina de dois membros participantes do projeto.

## Ferramentas

A Tabela 8 descreve as ferramentas que serão utilizadas no processo de desenvolvimento do projeto SA.

Tabela 8 — Ferramentas utilizadas no processo de desenvolvimento do projeto SA.

|  |  |
| --- | --- |
| Ferramenta | Descrição |
| Eclipse 2020-06(4.16.0) | IDE a ser utilizado para o desenvolvimento da aplicação. |
| MySQL 4.1 | SGBD a ser utilizado no piloto. |
| Visual Paradigm Online | Modelagem UML. |
| Dbeaver 7.2.1 | SGBD a ser utilizado no projeto. |
| WhatsApp 2.20.199.14 | Ferramenta de comunicação. |