

---

# **Plano de Gerência de Configuração**

**Projeto UFRPE – Sistema Geral de Controle**

Versão 1.01

## Histórico de Alterações

Data	Versão	Descrição	Autor

## Índice

1.	Introdução	4
1.1	Público Alvo	4
1.2	Glossário	4
2.	Organização	5
2.1	Ferramentas	5
2.2	Linguagens	5
2.3	Localização dos artefatos	6
2.4	Repositório	6
2.5	<i>Baselines</i> do Projeto	7
3.	Identificação de Configuração	7
3.1	Versionamento	7
3.2	Nomenclatura	8
3.2.1	Documentos	8
3.2.2	Código	9
3.2.3	Commit	11
3.2.4	Criação do Scrumy	11
4.	Configuração e Controle de Mudanças	11
4.1	Procedimentos de Mudança	11
5.	Auditoria de Configuração	12

# 1. Introdução

Este documento descreve planejamento para a gerência de configuração do projeto **UFRPE – DisAê** incluindo a identificação dos itens de configuração, estrutura adotada para o repositório dos itens, padrões de nomenclatura, ferramentas de apoio e outras informações relacionadas.

## 1.1 Público Alvo

*Stakeholders* do projeto, especialmente os desenvolvedores e engenheiros de configuração.

## 1.2 Glossário

Esta seção explica o conceito de alguns termos importantes que serão mencionados no decorrer deste documento.

**Tabela 1 – Glossário**

Termo	Definição
Baseline	Marco no desenvolvimento do projeto, composta de um conjunto de artefatos aprovado, estáveis e consistentes entre si.
Branches	Caminho alternativo para o desenvolvimento em paralelo, criado através de rótulos aplicado à determinada versão de um artefato.
CR	Do inglês, <i>Change Request</i> , solicitação de mudanças.
IC	Item de Configuração, ou seja, qualquer artefato do projeto que será submetido à gerência de configuração.
Label, Tag, Rótulo	Marca usada para identificar facilmente uma versão específica de um artefato ou baseline.
CCB	Do inglês, <i>Software Configuration Control Board</i> ou <i>Change Control Board</i> , grupo responsável por autorizar modificações nos itens de configuração e estabelecimento de baselines.
SCM, CM	Do inglês, <i>Software Configuration Management</i> .
Release	Conjunto de itens que é enviado para o cliente. Contém arquivos de instalação, arquivos de dados, programas de instalação e documentação. Cada release pode incluir novas funcionalidades ou alguma mudança decorrente de customização, desenvolvimento, testes ou mudança de legislação.
Build	Conjunto de itens que é submetido à homologação interna antes de ser liberado para um cliente através de um release.

## 2. Organização

### 2.1 Ferramentas

Esta seção descreve a infraestrutura que será utilizada para a realização das atividades de gerência de configuração e mudanças.

**Tabela 2 – Ferramentas, Ambientes e Infraestrutura**

Ferramenta	Versão	Propósito
code.google.com.br	-	Ambiente para promoção de código livre, interação com desenvolvedores e gerenciamento de projetos
MySQL	5.6	SGBD
Apache	2.4	Servidor HTTP
Eclipse	4.2	Ferramenta de Desenvolvimento
TortoiseSVN	1.7	Cliente SVN
GoogleGroups	-	Comunicação da equipe
Facebook	-	Comunicação da equipe

### 2.2 Linguagens

PHP	5.4	Linguagem Backend
Java para Androide	2.3	Linguagem Frontend
XML	-	Linguagem Frontend
SQL	5.6	Linguagem Backend

### 2.3 Localização dos artefatos

Esta seção descreve a localização onde os artefatos estão armazenados.

Diretório	Descrição
<i>Tags</i>	Local que armazena a versão rigorosamente testada ao final de uma sprint. Esta cópia testada do sistema dá origem a um <i>release</i> ou a uma versão liberada para uso. É esse diretório <i>tags</i> que é empacotado e enviado para uso do cliente.
<i>Trunk</i>	A pasta <i>trunk</i> contém todos os artefatos do projeto em desenvolvimento. Neste diretório, também estão todas as atualizações efetuadas no dia-a-dia. Este diretório é responsável por armazenar a estrutura principal do projeto, por meio dos diretórios <i>docs</i> e <i>src</i> .

### 2.4 Repositório

Tabela 3 - Repositório

Informações Gerais do Repositório	
Tipo do Repositório	GoogleCode
URL do Repositório	<a href="https://dizae.googlecode.com/svn/trunk/">https://dizae.googlecode.com/svn/trunk/</a>
Repositório do projeto	<a href="https://code.google.com/p/dizae/">https://code.google.com/p/dizae/</a>
Acesso ao Repositório	Navegador Web
Estrutura do Repositório	
<ul style="list-style-type: none"><li>• <i>//doc/planejamento/...</i><ul style="list-style-type: none"><li>○ Plano Gerencia de Configuração;</li><li>○ Canvas;</li><li>○ Plano de Projeto (cronograma geral, riscos);</li><li>○ Documento de Requisitos;</li></ul></li><li>• <i>//doc/teste/...</i><ul style="list-style-type: none"><li>○ Plano de Testes</li><li>○ Resultados de teste</li></ul></li><li>• <i>//src/...</i><ul style="list-style-type: none"><li>○ Código fonte</li></ul></li></ul>	

## 2.5 Baselines do Projeto

Esta seção descreve quais *Baselines* são geradas para o projeto.

**Tabela 4 – Baselines do Projeto**

Baseline	Descrição	Padrão
Requisitos	Marcado assim que for concluída análise de requisitos da iteração.	SGC_REQ_ <ITERAÇÃO>
Análise de Projeto	Marcado quando forem concluídos a análise e o projeto de cada iteração.	SGC_ARCH_<ITERAÇÃO>
Build	Criada a cada <i>build</i> para o software.	SGC_BUILD_<BUILD>
Release	Criado a cada <i>release</i> do software.	SGC_RELEASE_<VERSÃO>
Documentos	Criado após a aprovação de um documento.	SGC_<NOME>

Onde,

<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.

## 3. Identificação de Configuração

### 3.1 Versionamento

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

X.YZ

Onde,

X                    é um número decimal que representa uma versão final do artefato;  
YZ                   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 primeira versão do artefato deve ser 0.01;

- A cada modificação no artefato, o valor YZ deve ser incrementado;
- Após cada aprovação do artefato, a versão X deve ser incrementada de uma unidade e o valor YZ retorna para 00, sendo assim gerada uma nova versão oficial;

Para que a versão de um artefato seja modificada é necessária a aprovação dos gerentes responsáveis pelo artefato respectivo.

## 3.2 Nomenclatura

### 3.2.1 Documentos

Regra:

**SGC\_<ACRONIMO>\_[<DESCRICAO>].<EXT>**

Onde,

<ACRONIMO>            É o acrônimo utilizado;

<DESCRICAO>            É uma breve descrição do arquivo, **se necessário**. Esta descrição só pode conter caracteres maiúsculos e não acentuados (A-Z), números (0-9) e hífen (-);

<EXT>                    É a extensão do documento.

**Tabela 5 - Acrônimos**

<b>Acrônimo</b>	<b>Descrição</b>
SPMP	Plano de Gerência de Projeto de Software
SCMP	Plano de Gerência de Configuração de Software
SQAP	Plano de Garantia da Qualidade de Software
STP	Plano de Testes de Software
TRK	Relatórios (status, métricas, auditorias, resultado de testes)
ADT	Auditoria
UC	Documento de Casos de Uso
REQ	Documento de Requisitos
ARCH	Documento de Arquitetura
CD	Diagrama de Classes
ER	Diagrama Entidade Relacionamento
DF	Diagrama de Fluxo de Dados
SD	Diagrama de Sequência
ACT	Diagrama de Atividades



TC	Casos de Teste
DBM	Modelagem de Banco de Dados
BKI	Itens de <i>Backlog</i>
MEET	Atas de Reuniões
SPM	Planejamento de <i>Sprints</i>
TTC	Tabela de Horários dos Consultores
CV	Canvas

### 3.2.2 Código

#### 3.2.2.1 Classes

##### Regra:

**<NOME\_CLASSE\_CAMEL\_CASE>**

##### Onde,

<NOME\_CLASSE\_CAMEL\_CASE> É o nome da classe iniciada com maiúscula. Não pode conter caracteres acentuados (A-Z), números (0-9) e hífen (-). Para palavras compostas, cada palavra é iniciada com maiúsculas unidas sem espaço.

#### 3.2.2.2 Métodos

##### Regra:

**<NOME\_METODO>**

##### Onde,

<NOME\_METODO> É o nome do método iniciado com minúscula. Não pode conter caracteres acentuados (A-Z) e hífen (-). Para palavras compostas, a partir da segunda palavra, é iniciada com maiúsculas unidas sem espaço.

#### 3.2.2.3 Parâmetros

##### Regra:

**<NOME\_PARAMETRO>**

##### Onde,

<NOME\_PARAMETRO> É o nome do parâmetro iniciado com minúscula. Não pode conter caracteres

acentuados (A-Z) e hífen (-). Para palavras compostas, a partir da segunda palavra, é iniciada com maiúsculas unidas sem espaço.

#### 3.2.2.4 *Variáveis*

**Regra:**

**<NOME\_VARIAVEL>**

**Onde,**

<NOME\_VARIAVEL> É o nome da variável iniciada com minúscula. Não pode conter caracteres acentuados (A-Z) e hífen (-). Para palavras compostas, a partir da segunda palavra, é iniciada com maiúsculas unidas sem espaço.

Observação: Para constantes, todas as letras são maiúsculas.

#### 3.2.2.5 *Comentários*

**Regra:**

**<DESCRICA0>**

**Onde,**

<DESCRICA0> É o comentário com minúsculas e quebra linhas.

#### 3.2.2.6 *Banco de Dados*

##### 3.2.2.6.1 *Tabelas*

**Regra:**

**<NOME\_TABELA\_CAMEL\_CASE>**

**Onde,**

<NOME\_TABELA\_CAMEL\_CASE> É o nome da tabela iniciada com maiúscula. Não pode conter caracteres acentuados (A-Z), números (0-9) e hífen (-). Para palavras compostas, cada palavra é iniciada com maiúsculas unidas sem espaço.

#### 3.2.2.6.2 Colunas

##### Regra:

<NOME\_COLUNA>\_<NOME\_TABELA>

##### Onde,

<NOME\_COLUNA>           É o nome da coluna com minúsculas;

<NOME\_TABELA>           É o nome da tabela com minúsculas.

#### 3.2.3 Commit

##### Regra:

SPRINT <NÚMERO DA SPRINT> - <TÍTULO> - <RESUMO>

##### Onde,

<NÚMERO DA SPRINT>    É o número da *Sprint* atual;

<TÍTULO>                É o título da atividade desenvolvida;

<RESUMO>                É um resumo da descrição das alterações feitas.

#### 3.2.4 Criação do Scrumy

##### Regra:

www.scrumy.com/<NOME DO GRUPO>-sprint<NUMERO>

##### Onde,

<NOME DO GRUPO>        É o nome do grupo responsáveis pelos requisitos descritos na página;

<NUMERO>                É o número da *Sprint* atual.

## 4. Configuração e Controle de Mudanças

### 4.1 Procedimentos de Mudança

As mudanças nos itens de configuração do projeto devem ser discriminadas no *commit*.

## **5. 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 devem ser armazenados no repositório do projeto e devem ser acompanhados pelos Gerentes de Configuração.