

# Plano de Gerência de Configuração

## Introdução

O Plano de Gerenciamento de Configuração estabelece e mantém a integridade dos artefatos gerados no projeto **SWSchool**, permitindo o acompanhamento destes itens durante todo o ciclo de vida do projeto, e preservando o histórico de evolução dos mesmos. Auxiliando a gerenciar o estado dos itens de configuração dos sistemas, controlar as mudanças em itens de configuração e rastrear modificações nos itens de configuração ao longo do tempo.

## Propósito

Este documento tem como propósito organizar a evolução do projeto **SWSchool** mantendo um planejamento dos artefatos gerados no que diz respeito a nomenclatura e regras para o versionamento, estrutura de diretórios, identificação e responsabilidades.

## Escopo

Este documento descreve toda a infra-estrutura utilizada durante o desenvolvimento do projeto **SWSchool**.

## Estrutura de Diretórios

Diretório	SubDiretório	Artefatos
<b>docs</b>		Todos os documentos gerados pela gerência do projeto
<b>Project</b>	App	Arquivos e pastas da aplicação
	libs	Arquivos e pastas da biblioteca do framework usado (cakePHP)
	plugins	Arquivos e pastas de plugins adicionais
	vendors	
<b>database</b>	Tabelas	Arquivos contendo as instruções para criação das tabelas SQL
	Inserções	Arquivos contendo as instruções para popular o banco

## Organização Estrutural

### Identificação de Documentos

Todos os documentos gerados neste projeto terão a seguinte nomenclatura:

**<ACRONIMO>\_<TEXTO\_CAMEL\_CASE>.<EXT>** onde:

Parte da nomenclatura	Regra
<b>ACRONIMO</b>	Acrônimo definido abaixo que irá representar o artefato em questão. Servirá como <b>identificador único</b> .
<b>TEXTO_CAMEL_CASE</b>	Texto livre usando palavras compostas. Maiores informações, <a href="#">clique aqui</a> . Obs: texto opcional
<b>EXT</b>	Extensão do artefato em questão. Ex: doc, xls, etc.

### Acrônimos

Abaixo são descritos os acrônimos de cada artefato do projeto juntamente com seu significado além de uma descrição do propósito do documento que tal acrônimo representa.

Acrônimo	Significado
<a href="#">PGC</a>	Plano de Gerência de Configuração
<a href="#">Auditoria</a>	Auditoria
<a href="#">UseCases</a>	Documento de casos de usos
<a href="#">REQ</a>	Documento de requisitos
<a href="#">DiagClass</a>	Diagrama de classes
<a href="#">DiagER</a>	Diagrama Entidade Relacionamento
<a href="#">DiagFluxoAtiv</a>	Diagrama de fluxo de dados
<a href="#">ModelagemBD</a>	Modelagem de banco de dados
<a href="#">Backlog</a>	Itens de Backlog

<a href="#">MapMental</a>	Mapa Mental
<a href="#">PerfilAlunos</a>	Perfil do alunos
<a href="#">Layout</a>	Design de Interfaces de páginas do sistema
<a href="#">DocVisao</a>	Documento de visão
<a href="#">ARN</a>	Atas de Reuniões
<a href="#">SprintPlanning</a>	Planejamento de sprints
<a href="#">THC</a>	Tabela de horários dos consultores

## Identificação de Bases de Dados

### Tabelas

Dentro do diretório database -> tabelas serão armazenados todos os arquivos com função de gerar as tabelas SQL do projeto, estes arquivos deverão seguir a seguinte nomenclatura:

**CreatingTables\_<VERSAO\_DO\_ARQUIVO>.sql**

Parte da nomenclatura	Regra
<b>VERSAO_DO_ARQUIVO</b>	Versão atual do arquivo SQL, caso exista mais de um arquivo será considerado apenas o arquivo com maior versão encontrada.

### Inserções

Dentro do diretório database -> Inserções serão armazenados todos os arquivos com função de popular as tabelas SQL do projeto, estes arquivos deverão seguir a seguinte nomenclatura:

**Populatetables\_<VERSAO\_DO\_ARQUIVO>.sql**

Parte da nomenclatura	Regra
-----------------------	-------

<b>VERSAO_DO_ARQUIVO</b>	Versão atual do arquivo SQL, caso exista mais de um arquivo será considerado apenas o arquivo com maior versão encontrada.
--------------------------	--

### **Padrão de commit**

Ao commitar uma alteração no repositório é obrigatório informar a descrição do commit, obedecendo a seguinte regra:

**<Versão da Sprint> - <Título da tarefa> - <Resumo/descrição das alterações feitas>**

Exemplo:

*Sprint 5 - Relatório de consultores cadastrados - editado controller consultores, adicionado a função XXXXX para formação das consultas*

### **Padrão de Criação do Scrummy**

OBS: Antes de criar o scrummy terá que ser cadastrada a tarefa (Elaboração do scrummy) no redmine.

O scrummy deve ser criado seguindo o seguinte padrão:

**[www.scrummy.com/sws-2012-2-NomeDoGrupo-NumeroDaSprint](http://www.scrummy.com/sws-2012-2-NomeDoGrupo-NumeroDaSprint)**.

Assim que criado ele deve ser publicado no grupo do facebook para que todos possam visualizar

### **Padrão de comunicação**

A comunicação entre os membros de processo e gerencia de projeto tratando especificamente do projeto SWSchool, devem ser feitas seguindo as seguintes regras:

E-mail sobre particularidades do projeto devem vir com [Tag] na frente do assunto, definindo se a mensagem se trata de uma descrição de um problema, uma sugestão, ou não entendimento de algumas especificidade. Ex.: [Atividade][Sprint-3] “não consegui compreender a atividade ‘criar layout’ da Sprint 3, significa fazer um protótipo ou desenvolver tudo?”

Sugestões de tag: [cronograma], [impedimento], [GConfig], [GDados], [Backlog], etc

O grupo formado no facebook denominado “Gerencia/Processo de desenvolvimento de Software 2012.2” tem como objetivo um contato mais dinâmico entre os membros. Devendo ser utilizado para a disponibilização de informações para todos os usuários.

### **Baselines do Projeto**

As baselines serão criadas a cada final de uma sprint. Todo o repositório será replicado para pasta com a seguinte nomenclatura:

**SWSchool\_baseline\_sprint\_<numero\_da\_sprint>**

### **Instruções para instalação da aplicação**

Faça o update do projeto na pasta www/htdocs do seu servidor web. Crie, um banco de dados chamado swsdb, após isso execute os arquivos SQL de criação de tabelas e população na base de dados criada, esses arquivos serão encontrados na pasta database do projeto, logo após, vá em Project -> app -> Config -> database.php

- Na linha 65 ('host' => 'localhost') em localhost defina o ip do seu servidor web (por padrão deixe como está).
- Na linha 66 ('login' => 'xxxx') forneça o login da sua base de dados.
- Na linha 67 ('password' => 'xxx') forneça o password da sua base de dados.
- Na linha 68 ('database' => 'swsdb') defina a base de dados usada (segundo esse tutorial o padrão é swsdb)

Em seu browser digite localhost/<para do repositório>/Project digite admin/admin no login e senha de acesso.

Para a maioria dos usuários essas configuração são o bastante.

### **Ferramentas utilizadas no projeto**

#### **Modelo de Desenvolvimento**

MVC

#### **Framework**

CakePHP

#### **Linguagem Backend**

- PHP 5.4

#### **Linguagem Frontend**

HTML5 + CSS3

#### **SGBD**

- MySQL 5.5

#### **Ferramenta de gerenciamento de projeto :**

- Redmine+Scrum

#### **Framework de persistência de dados :**

- Lumine

### **IDE de desenvolvimento**

- Eclipse for PHP
- Notepad++
- Sublimetext 2
- Dreamweaver CS5
- Netbeans

### **ferramentas Mysql**

[Mysql Workbench 5.2](#)

[phpMyAdmin 3.4.11](#)

### **Cliente SVN**

TortoiseSVN 1.7.10

### **Cliente FTP**

FileZilla 3.6.0

### **Bug Tracking (Ferramenta para auxílio de testes)**

Mantis Bug Tracker

Selenium