

PLANO DE GESTÃO DE CONFIGURAÇÃO DE SOFTWARE (SOFTWARE CONFIGURATION MANAGEMENT - SCM)

1.0 Introdução

Esta seção fornece uma visão geral do plano de SCM.

1.1 Escopo e objetivo das atividades de SCM

Este documento descreve as práticas para a realização das atividades de controle de configuração de software, detalhando a infraestrutura utilizada para o desenvolvimento do projeto Agile Fichas, mantendo a integridade dos dados e controlando as versões do sistema.

1.2 Papel organizacional da SCM

A equipe é constituída de cinco integrantes, em que a Tabela 1 define o quadro organizacional, sendo dividido na seguinte ordem:

Nome	Cargo	Atribuições
Jefferson	Gerente	<ul style="list-style-type: none">- Acompanhar o andamento da equipe em relação ao projeto;- Organizar as reuniões;- Realizar uma pré análise do projeto e delegar atribuições aos membros da equipe;
Gerson	Analista	<ul style="list-style-type: none">- Estabelece reuniões com o cliente;- Define as ferramentas e metodologias que serão adotadas pela equipe;
Felipe	Desenvolvedor	<ul style="list-style-type: none">- Desenvolve os requisitos estabelecidos;- Insere comentários ao código para facilitar a documentação futura;
João	Testador de Software	<ul style="list-style-type: none">- Estabelece cenários de validações e as corrige se necessário;- Documenta de forma completa o projeto com base nos comentários dos desenvolvedores;
Tiago	Desenvolvedor	<ul style="list-style-type: none">- Desenvolve os requisitos estabelecidos;- Insere comentários ao código para facilitar a documentação futura;

Tabela 1. Quadro Organizacional

2.0 Tarefas de SCM

Esta seção detalha todas as tarefas importantes de SCM.

2.1 Identificação

Os documentos produzidos devem sempre seguir o mesmo padrão textual definido neste documento, assim todos os integrantes irão obter um mesmo entendimento do contexto analisado.

Padrão de nomenclatura dos arquivos:

PROJETO_DESCRIÇÃO_VERSÃO_DATA

Exemplo:

AGILEFICHAS_RG_1.0_14112020

Observações:

- Todos os caracteres devem ser em maiúsculo.
- A descrição deve ser utilizada a sigla definida pela tabela descritiva, demonstrada na Tabela 2.

Tabela de siglas e descrições para utilização na nomenclatura dos arquivos:

Descrição do documento	Sigla
Reunião geral com a equipe	RG
Registro de Não Conformidade de Processo	RNCP
Testes de Software	TS
Configuração de Software	CS
Qualidade do Software	QS
Projeto de Software	PS
Relatório de Alterações	RA

Tabela 2. Descrição dos documentos

2.2 Controle de Configuração

As alterações futuras que sejam necessárias implementar no projeto, deverão ser autorizadas pelo gerente e o analista mediante a um chamado(ticket) ou via email, que contenha a seguinte descrição:

Campo	Valor
Componente	nome do componente
Versão	versão atual que se encontra o componente
Plataforma	Android ou IOS
OS	Linux ou Windows
Prioridade	É dividido em 3 níveis, sendo eles: <ul style="list-style-type: none">- Baixo- Médio- Alta
Tipo de Alteração	Possui as seguintes categorias: <ul style="list-style-type: none">- Correção de bug: Corrige defeito na aplicação;- Implementação: Adiciona nova funcionalidade;- Documentação: Altera a documentação do projeto
CC	Lista emails das pessoas envolvidas com o chamado
Assunto	Nome do problema ou da requisição
Descrição	Explicar detalhadamente o chamado

Tabela 3. Composição do Chamado(ticket)

O fluxo de um chamado aberto para alteração ou implementação de um recurso referente ao projeto, segue a ordem abaixo:

1. Validação dos campos do chamado (falta de informações ou fora de contexto, chamado é finalizado com o motivo).
2. Deferimento ou indeferimento por parte do gerente e analista (Caso

- indeferido chamado é finalizado com o motivo).
3. Alocação do chamado na fila da equipe responsável.
 4. Equipe trata o chamado e gera um relatório com alterações realizadas.
 5. Analista revisa o relatório e publica a nova versão.

2.3 Controle de versões

O controle de versões será feito com o auxílio da ferramenta git para o código e a documentação. Cada lançamento do software será descrito com o esquema <XX>.<YY>.<ZZ>, onde cada elemento é representado da seguinte maneira:

Tipo de versão	Descrição
<XX>	Versão final a ser entregue ao cliente, releases do projeto
<YY>	Campo que representa as novas versões acrescentadas a uma release já existente, este campo é incrementado, e somente é zerado quando é realizado uma nova release.
<ZZ>	Representa a quantidade de releases de correção para os releases <XX>.<YY>, é zerado quando é iniciado uma nova release <YY> ou <XX>

Tabela 4. Versionamento do Sistema

O versionamento da documentação é descrita em <XX>.<YY>, sendo que <XX> representa a versão mais recente do documento e <YY> são versões em andamento que ainda não foram homologadas, ao serem homologadas inicia uma nova numeração em <XX> e <YY> é zerado.

Observações:

- O documento inicial deve começar em 00.00;
- A cada alteração deve ser incrementado o <YY> referente ao seu release, por exemplo: 00.01, 00.02 e assim por diante.
- Ao iniciar um novo release 01.00 seu <YY> deve ser zerado e incrementado caso seja necessário modificações.

2.4 Comunicação do status das configurações (CSA)

Ao ser homologada uma nova mudança de documentação ou de software de alta criticidade, a equipe será notificada por meio do sistema de intranet e a lista de email, onde conterà as informações dos impactos gerados e haverá também a data e hora de uma reunião que será realizada por meio do Microsoft Teams, com todos os membros da equipe envolvidos para sanar eventuais dúvidas e esclarecimentos.

Eventuais alterações de baixa criticidade será apenas gerado um relatório com os detalhes a respeito e enviado por email e notificados via intranet para todos os integrantes envolvidos.

2.5 Auditorias e revisões

As auditorias e revisões devem ser feitas pelo auditor, com o objetivo de validar se todas as alterações no projeto não afetem nenhum outro ponto que já tenha sido desenvolvido, e encontrar possíveis inconsistências, como por exemplo um dado inserido no projeto que não poderia ter sido inserido por aquele caminho ou aquela pessoa.

Além de averiguar se as especificações da alteração em questão foi realizada/preenchida corretamente, para que não falte informação no momento de desenvolver.

O resultado destas auditorias deve ser registrado em formato de relatório e armazenado através do repositório já utilizado pelo grupo para versionamento dos dados.

3.0 Recursos para SCM

Ferramenta	Descrição.
Git	Ferramenta para versionamento.

MySQL	SGBD a ser utilizado.
IntelliJ	IDE a ser utilizada.
Google Drive	Organização de alguns documentos e planilhas.