**Plano De Gerenciamento De Configuração Da GC**

**Controle de Versões**

*<Inserir os dados das versões.>*

| **Versão** | **Data** | **Autor** | **Notas da Revisão** |
| --- | --- | --- | --- |
| **VC1.0** | **19/04/23** | **Wiuver Ribeiro** | **Primeiro Commit** |

1. **Introdução**

A atividade tem por fim documentar toda a trajetória até o presente momento, sendo ensinado conceitos que são usados no dia a dia de quem trabalha com Tecnologia,

1.1 **Finalidade**

Tem por finalidade ressaltar tudo aprendido durantes as aulas, sobre os conceitos do git, entre eles modificações, controle de versões, e o gerenciamento de configurações.

1.2 **Escopo**

Documento serve como referência para todos as pessoas que trabalham com tecnologia no geral.

1.3 **Definições, Acrônimos e Abreviações**

O documento tem referências que podem ser seguidas em ordem crescente para ser feito um maior aproveitamento de todo o conteúdo do mesmo.]

**Baseline:** é um ponto de referência para o monitoramento do desenvolvimento das tarefas relacionadas ao projeto.

**Branch**:O branch local, como o próprio nome sugere, consiste em ramificações do código que apenas um desenvolvedor tem acesso, ficando disponível localmente em seu computador.

**Check-out:** O comando git checkout é uma ferramenta essencial para a operação padrão do Git. É uma contraparte do git merge

**Change Request:** Verificar modificações nos arquivos feitas recentemente.

**Change Control Board:** é um grupo de pessoas dentro de um equipe de projeto quem é responsável por tomar a decisão final sobre se e quando devemos fazer certas alterações em um sistema dentro de um projeto.

**Gerenciamento de configurações**:O Git é o sistema de controle de versão líder do setor para rastrear alterações de código.

**1.4 Referências**

[*https://www.projectbuilder.com.br/blog/baseline-de-projetos/#:~:text=A%20baseline%20%C3%A9%20um%20ponto,das%20tarefas%20relacionadas%20ao%20projeto*](https://www.projectbuilder.com.br/blog/baseline-de-projetos/#:~:text=A%20baseline%20%C3%A9%20um%20ponto,das%20tarefas%20relacionadas%20ao%20projeto)*.*

*Referência 02:* [*https://tecnoblog.net/responde/o-que-e-branch-em-programacao/*](https://tecnoblog.net/responde/o-que-e-branch-em-programacao/)

[*https://www.atlassian.com/br/git/tutorials/using-branches/git-checkout#:~:text=Em%20resumo%2C%20o%20git%20checkout,uma%20contraparte%20do%20git%20merge%20*](https://www.atlassian.com/br/git/tutorials/using-branches/git-checkout#:~:text=Em%20resumo%2C%20o%20git%20checkout,uma%20contraparte%20do%20git%20merge%20)*.*

*https://edisciplinas.usp.br/mod/resource/view.php?id=1723652*

1.5 **Visão Geral**

O documento será organizado em seções para cada tópico a ser discutido.

2. **Gerenciamento de Configuração de Software**

**Abordagem de ferramentas e padrões**

**2.1 Organização, Responsabilidades e Interfaces**

*<descreva quem será o responsável pela execução das diversas atividades de Gerenciamento de Configuração (CM) descritas no Processo de Gerenciamento da Configuração>*

| **Papel** | **Responsabilidade** |
| --- | --- |
| **Gerência de Projeto** | **Jira** |
| **Gestão de Requisitos** | **GitLab** |
| **Análise de Projeto** |  |
| **Construção** | **Apache, Netbeans e MYSQL** |
| **Teste de software** | **jira** |
| **SGBD** | **MySQL, PostrgreSQL, NOSQL** |

**2.2 Ferramentas, Ambiente e Infraestrutura**

3.  **O Programa de Gerenciamento de Configuração**

**3.1  Identificação da Configuração**

**3.1.1 Métodos de Identificação**

*<Descreva como os artefatos do projeto ou produto devem ser nomeados, marcados e numerados. O esquema de identificação deve abranger o hardware, o software do sistema, os produtos de terceiros e todos os artefatos de desenvolvimento de aplicativos listados na estrutura de diretórios do produto; por exemplo, planos, modelos, componentes, software de teste, resultados e dados, executáveis e assim por diante>*

3.1.2 **Baselines do Projeto**

*<As baselines funcionam como um padrão oficial no qual os trabalhos subsequentes são baseados. Somente mudanças autorizadas podem ser efetuadas nas baselines.*

*Descreva em que pontos do ciclo de vida do projeto ou produto as baselines devem ser estabelecidas. As baselines mais comuns devem ser definidas ao final de cada uma das fases de Iniciação, Elaboração, Construção e Transição. Elas também podem ser geradas no final de iterações ocorridas dentro das várias fases ou com frequência ainda maior.*

*Descreva quem autoriza uma baseline e o que ela contém>*

3.2   **Controle de Configuração e Mudança**

3.2.1 **Processamento e Aprovação de Solicitações de Mudança**

*<Descreva o processo pelo qual os problemas e as mudanças são submetidos, revisados e dispostos>*

3.2.2  **Comitê de Controle de Mudança (CCB)**

*<Descreva a participação e os procedimentos para processar solicitações e aprovações de mudança a serem seguidos pelo CCB>*

3.3 **Estimativa do Status de Configuração**

3.3.1 **Processo de Armazenamento de Mídia e Liberação do Projeto**

*<Descreva as políticas de retenção e os planos de backup, erros irreversíveis e recuperação. Descreva também como a mídia deve ser mantida on-line, off-line, tipo de mídia e formato. O processo de liberação deve descrever o conteúdo do release, a quem ele se destina e se há quaisquer problemas conhecidos e instruções de instalação>*

3.3.2   **Relatórios e Auditorias**

*<Descreva o conteúdo, o formato e a finalidade dos relatórios e auditorias de configuração solicitados.*

*Os relatórios são usados para avaliar a "qualidade do produto" em qualquer fase do ciclo de vida do projeto ou produto. Os relatórios sobre defeitos com base em solicitações de mudança podem fornecer alguns indicadores de qualidade proveitosos e, dessa forma, alertar a administração e os desenvolvedores para determinadas áreas prioritárias do desenvolvimento. Geralmente os defeitos são classificados por prioridade (alta, média e baixa) e podem ser reportados com base nos seguintes aspectos:*

* *Vencimento (Relatórios Baseados em Períodos): Há quanto tempo defeitos de diversos tipos estão pendentes? Qual é o "tempo de retardo" de quando são encontrados defeitos no ciclo de vida em comparação com o tempo necessário para corrigi-los?*
* *Distribuição (Relatórios Baseados em Contagens): Existem quantos defeitos nas diversas categorias por proprietário, prioridade ou estado de correção?*
* *Tendência (Relatórios Relacionados a Períodos e Contagens): Qual é o número acumulado de defeitos encontrados e corrigidos no decorrer do tempo? Qual é a classificação dos defeitos detectados e corrigidos? Qual é a "lacuna de qualidade" em termos de defeitos pendentes versus defeitos corrigidos? Qual é a média de tempo de correção de um defeito?>*

4.  **Marcos**

*<Identifique os marcos internos e de cliente relacionados ao esforço do Plano de Gerenciamento da Configuração do projeto ou produto. Esta seção deve incluir detalhes sobre quando o Plano de Gerenciamento da Configuração deve ser atualizado>*

5. **Treinamento e Recursos**

*<Descreva as ferramentas de software, o pessoal e o treinamento necessários para implementar as atividades do Gerenciamento da Configuração especificadas>*

6. **Controle de Software de Subcontratados e Fornecedores**

*<Descreva de que forma o software desenvolvido fora do ambiente do projeto será incorporado>*

Aprovado em \_\_\_ de \_\_\_\_\_\_\_\_\_\_ de \_\_\_\_\_.

<nome completo da autoridade máxima da Estatal >

<cargo da autoridade máxima da Estatal >

**Observações:**

**Finalidade**

A finalidade do Plano Gerenciamento da Configuração é definir ou fazer referência às etapas e atividades que descrevem como o Gerenciamento de Controle de Configuração e Mudança é executado no desenvolvimento de um software.

**Ocorrência**

O Plano CM é elaborado no início da fase de Elaboração, depois que a provisão de fundos for aprovada para o prosseguimento do projeto. É recomendável que você o reveja no início de cada fase e o atualize de forma apropriada. O Plano CM precisa ser arquivado para que esteja disponível para as atividades de manutenção posteriores à implantação, especialmente para fornecer orientação sobre onde determinados componentes de software podem ser armazenados.

**Responsabilidade**

O Papel: O Gerente de Configuração é responsável pela integridade do Plano Gerenciamento da Configuração e por garantir que ele abranja:

* As atividades a serem executadas;
* O cronograma de atividades;
* As responsabilidades atribuídas;
* Os recursos necessários (equipes, ferramentas, ambiente e infraestrutura).

**Adaptação**

Este plano contém informações que podem ser cobertas em maior ou menor nível de detalhamento por outros planos.  As abordagens a seguir podem ser usadas para lidar com essa possível sobreposição:

* Fazer referência ao conteúdo em outro plano.
* Fornecer a visão geral em outro plano e mais detalhes neste plano.  As referências provenientes desses outros planos para o Plano de Gerenciamento de Configuração também podem ser úteis.  Isso normalmente funciona bem em projetos grandes, que tenham uma organização separada responsável pelo gerenciamento de configuração.
* Adaptar as seções de documento para cobrir somente as áreas que ainda não foram abordadas em nenhum outro lugar.