

CURSO GESTÃO DA TECNOLOGIA DA INFORMAÇÃO
AUDITORIA E QUALIDADE DE SOFTWARE

Trabalho apresentado para fins de avaliação do Projeto Integrador V da disciplina Auditoria e Qualidade de Software, ministrado pelo Professor: Winder César Mendes.

Alunos: Alexsander Bitencourt de Lima
Guilherme Caetano da Silva
Jose Matheus G. da Silva
Marielen Cristine de A. Leite

Goiânia junho 2021

INTRODUÇÃO

O intuito desse trabalho é colocar em prática os conhecimentos teóricos adquiridos no decorrer desse semestre referente a disciplina de Auditoria e Qualidade de Software.

O gerenciamento de configuração é uma disciplina que governa a identificação, controle, contabilidade de status, auditoria e interface de um determinado produto de software. É um dos muitos processos que ocorrem em um ambiente de desenvolvimento de engenharia no qual vários processos de engenharia, software e manufatura são executados simultaneamente.

Uma atividade de gerenciamento de configuração (CM) pode ser um indivíduo ou grupo designado pela administração de uma entidade para garantir que todas as pessoas que participam do programa ou projeto de software saibam o que está sendo projetado, desenvolvido, construído, testado e entregue. No final, essas mesmas pessoas e os usuários finais precisam saber o que foi desenvolvido, testado e entregue, para que o produto de software seja usado como pretendido e possa ser suportado e mantido em seu ambiente.

OBJETIVO

O objetivo criar/identificar um processo de gerência de configuração e controle de versão release e versões do software que venham agregar na qualidade do software Simulador WEB - Casa Bancária.

METODOLOGIA

Quanto à metodologia para estruturação do estudo e o desenvolvimento com embasamento nas normas e modelos internacionais como a ISO/IEC 12207, ISO/IEC 15504 e o CMMI citam a gerência de configuração de software como requisito para que uma empresa inicie a melhoria de qualidade do processo de desenvolvimento de software.

RESULTADO

Para que o objetivo principal seja alcançado, deverão ser atingidos os seguintes objetivos específicos:

- Revisão dos princípios e conceitos sobre Software Configuration Management (SCM), CMMi e MPS-Br o qual dá apoio ao gerenciamento e desenvolvimento de software;
- Análise de ferramentas disponíveis relacionadas ao controle de versões do desenvolvimento de projetos, buscando verificar a sua aplicabilidade;
- Modelar um sistema de controle de versões utilizando o ambiente colaborativo distribuído.

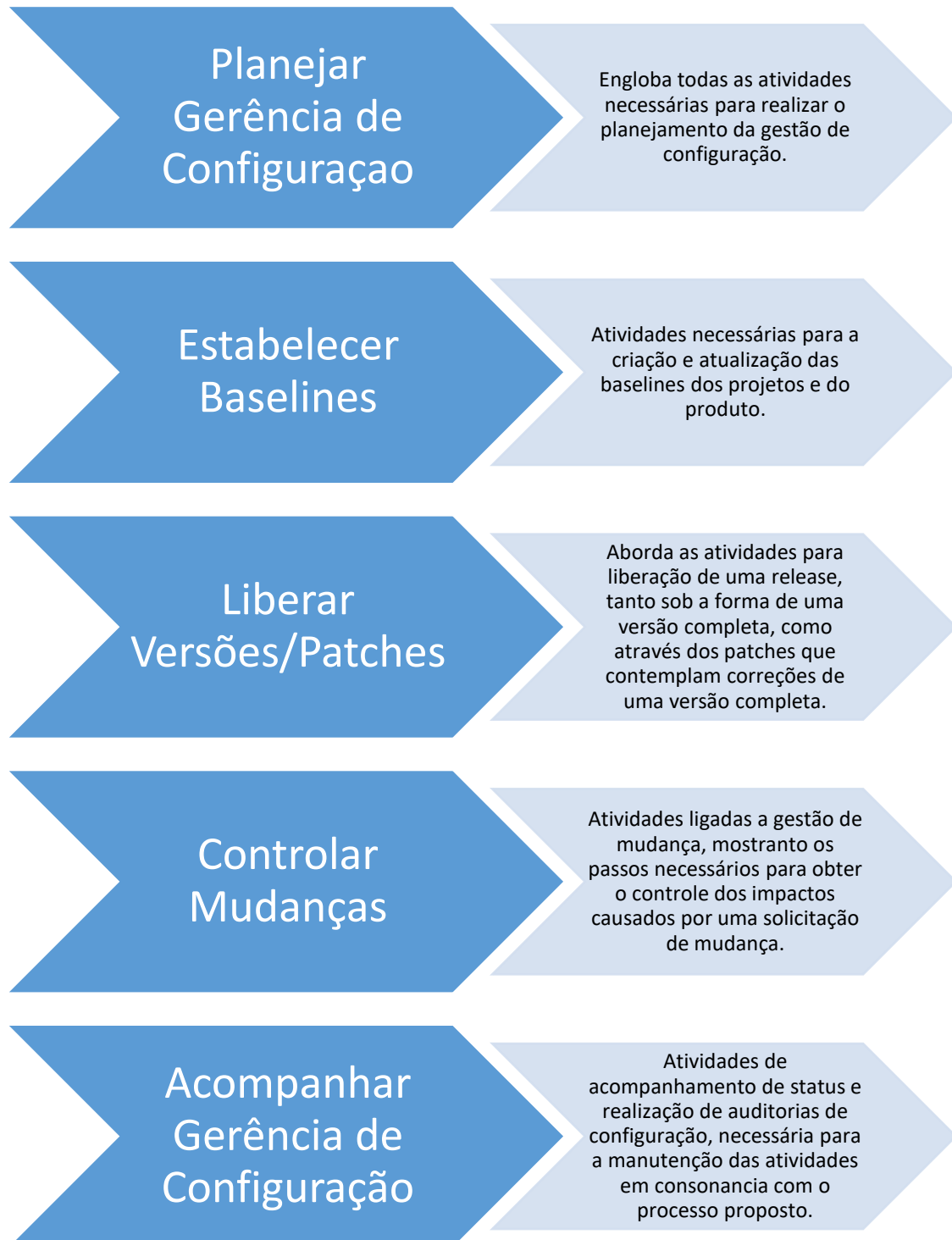
PROCESSO DE GERÊNCIA DE CONFIGURAÇÃO

Ele fornece detalhes de como a equipe de suporte ao sistema gerenciará o controle de itens de configuração ativos em serviço e as alterações que podem ser encomendadas durante a manutenção programada. O plano define as políticas e procedimentos para a gestão de configuração e a infraestrutura necessária para implementá-las.

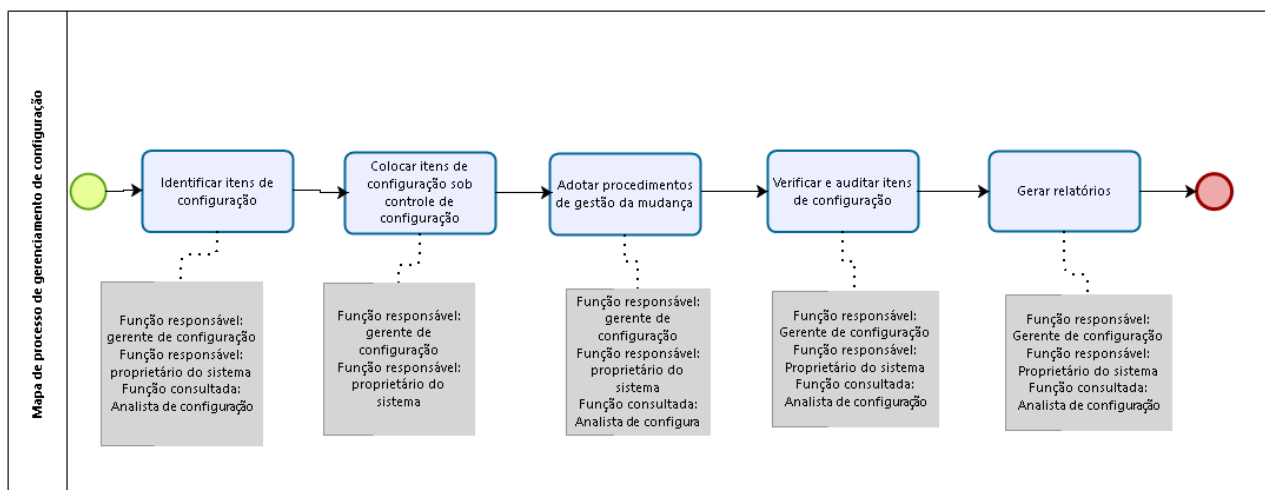
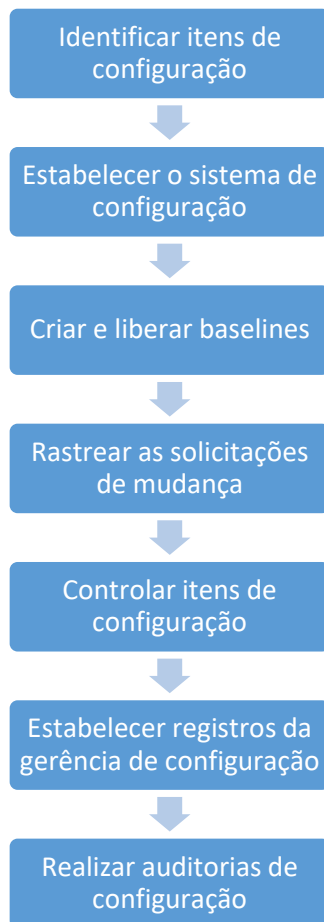
O CMMI trata a gerência de configuração como uma área de processo de suporte, pois seu objetivo é estabelecer e manter a integridade dos produtos de trabalhos gerados pelas áreas de processos de gerenciamento de projetos e engenharia. Esta área de processo envolve três metas específicas:

- Estabelecimento de baselines, para garantir um conjunto de itens de configuração estáveis o suficiente para servirem como base para o trabalho da equipe do projeto;
- Controle e rastreamento de mudanças, através do registro e análise das solicitações de mudança para um projeto, e o acompanhamento de sua resolução;
- Estabelecimento de integridade das baselines, onde, através das auditorias de configuração, seja possível monitorar as baselines, onde, através das auditorias de configuração, seja possível monitorar as baselines estabelecidas e evitar que elas sejam violadas e tornem-se inconsistentes com o planejamento feito para o projeto.

Fluxo da Gerência de Configuração



Práticas:



CONTROLE RELEASE E VERSÕES DO SOFTWARE

O processo de controle da configuração objetiva controlar as alterações do produto, fazendo uso de uma sistemática confiável de avaliação e implementação de mudanças. Abrange a garantia de que, após a definição de uma linha de base, somente as mudanças aprovadas sejam implementadas.

Uma versão do sistema pode ser uma versão distribuída aos clientes. Cada versão do sistema deve incluir novas funcionalidades ou deve ser destinada a uma plataforma de hardware especial. Normalmente há muito mais versões de um sistema do que de lançamento. As versões são criadas com uma organização para desenvolvimento interno ou testes e não são destinadas à liberação aos clientes.

Versão x Release

Uma "versão" de algum software é toda a coleção de arquivos de origem e outros recursos que foram para fazer esse software em algum momento. O software pode passar por muitas versões de teste antes de finalmente ser declarado pronto para lançamento aos clientes.

A Release é uma versão do software publicada após ser testada e passou por todos os ciclos de desenvolvimento, e deveria ser implantada aos clientes. No entanto, pode haver uma liberação interna, que é uma liberação que não se destina a ser implantada. Cada release de sistema deve incluir nova funcionalidade ou se destinar a uma diferente plataforma de hardware.

Para nosso software optamos Git/GitHub por se tratar de uma ferramenta que já temos uma familiaridade. Tem outras ferramentas disponíveis como CVS, Subversion, Mercurial entre outras.

| Ferramentas | Versão | Descrição |
|--------------------|----------------|-----------------------------------|
| Visual Studio Code | - | Editor de texto para codar |
| Git | 2.7.4 ou maior | Ferramenta de versionamento |
| GitHub | - | Servidor de hospedagem do projeto |
| GitHub Pages | - | Página estática para documentação |

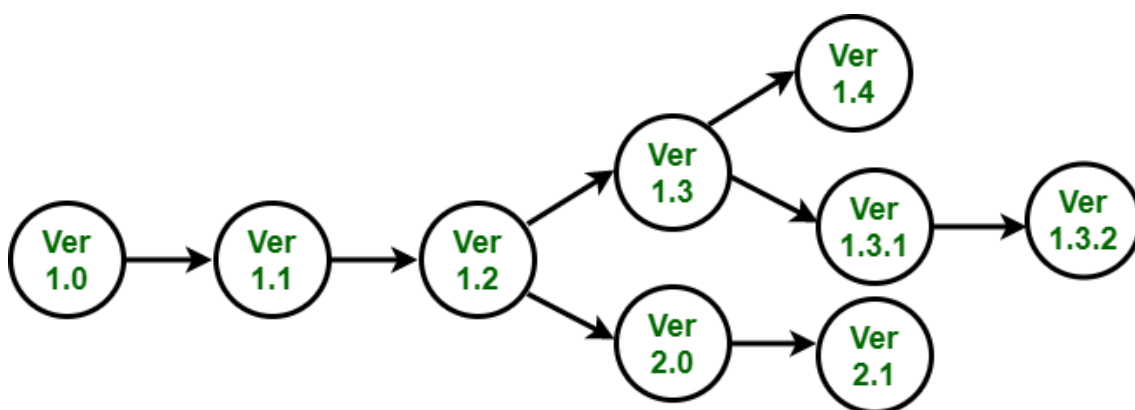


Figura ilustrativa modelo controle de versão

Modelo de Documentação

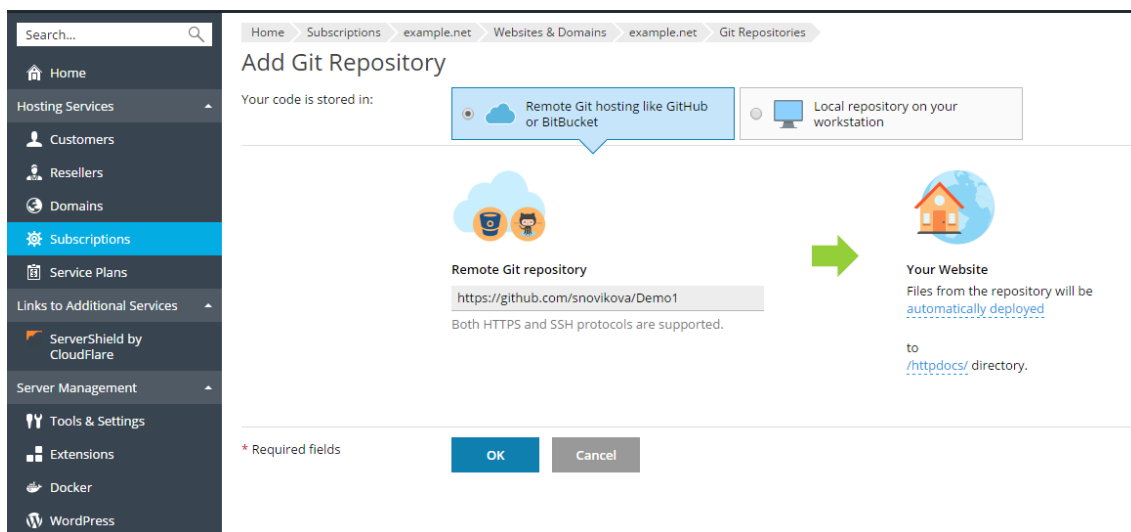
Histórico de Versão

| Número da Versão | Data da Liberação | Data da Implantação |
|--------------------------------|-------------------|---------------------|
| X.Y.Z | dd/MM/yyyy | dd/MM/yyyy |
| Breve descrição da Versão | | |
| texto descritivo | | |
| Itens de verificação da Versão | | |
| texto descritivo | | |
| Responsável pela Liberação: | | nome responsável |

Histórico de Release

| Número do Release | Data da Liberação | Data da Implantação |
|--|-------------------|---------------------|
| X.Y.Z | dd/MM/yyyy | dd/MM/yyyy |
| Breve descrição do Release (Histórico de Modificações) | | |
| texto descritivo | | |
| Responsável pela Liberação: | | nome responsável |

Figuras ilustrativas (GitHub)



Search...

Home > Subscriptions > example.net > Websites & Domains > example.net > Git Repositories

Add Git Repository

Your code is stored in:

- ☒ Remote Git hosting like GitHub or BitBucket
- ☐ Local repository on your workstation

Remote Git repository

<https://github.com/snovikova/Demo1>

Both HTTPS and SSH protocols are supported.

Your Website

Files from the repository will be [automatically deployed](#) to [/httpdocs/](#) directory.


* Required fields

OK Cancel

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

 treinaweb ▾

Repository name

/ repositorio_teste_github_pages ✓

Great repository names are short and memorable. Need inspiration? How about **jubilant-guide**.

Description (optional)

☒  **Public**

Anyone can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **None** ▾



Create repository

GitHub Pages

[GitHub Pages](#) is designed to host your personal, organization, or project pages from a GitHub repository.

Source

GitHub Pages is currently disabled. You must first add content to your repository before you can publish a GitHub Pages site. [Learn more](#).

None ▾

Save

Theme Chooser

Select a theme to publish your site with a Jekyll theme using the `master` branch. [Learn more](#).

Choose a theme

CONCLUSÃO

As práticas recomendadas de gerenciamento de configuração ajudam as organizações a gerenciar, organizar e controlar sistematicamente as alterações nos documentos, códigos e outras entidades durante o ciclo de vida do desenvolvimento de Software. O objetivo principal do processo SCM é aumentar a produtividade com erros mínimos. A principal razão por trás do processo de gerenciamento de configuração é que há várias pessoas trabalhando em software que está continuamente atualizando. O SCM ajuda a estabelecer a concorrência, a sincronização e o controle de versão.