**FACULDADE DE TECNOLOGIA SENAC GOIÁS**

**Gestão da Tecnologia da Informação**

LogoSenac

Denise Diniz A. Benevides

**Gerência de Configuração**

Elias Ferreira

GOIÂNIA,

2016Denise Diniz A. Benevides

**Gerência de Configuração**

Relatório apresentado como requisito parcial para obtenção de nota para a aprovação na disciplina Auditoria e qualidade de Software no Curso de Gestão da Tecnologia da Informação, na Faculdade de Tecnologia Senac Goiás.

Elias Ferreira

GOIÂNIA,

2016

# RESUMO

Este trabalho apresenta as características exigíveis para a apresentação dos relatórios nas disciplinas ministradas pela Lucília. O objetivo é disponibilizar aos alunos um modelo de apresentação de relatório técnico-científico para ser utilizado nestas disciplinas. Omitiram-se alguns elementos opcionais descritos nas normas da ABNT, bem como alguns itens mais específicos, simplificando-o.

Palavras-chave: Relatório técnico-científico. Modelo. ABNT.

**SUMÁRIO**

RESUMO 3

1 INTRODUÇÃO 5

2 OBJETIVO 5

3 DESENVOLVIMENTO 5

4 RESULTADOS 5

5 CONCLUSÃO 5

6 REFERÊNCIAS BIBLIOGRÁFICAS 5

APÊNDICE A – Título do apêndice 6

ANEXO A – Título do anexo 7

# INTRODUÇÃO

As mudanças são inevitáveis quando um software de computador é construído. E as mudanças aumentam o nível de confusão entre os engenheiros de software que estão trabalhando num projeto. A confusão surge quando as mudanças não são analisadas antes de serem feitas, registradas antes de serem implementadas, relatadas aos que precisam tomar conhecimento delas e controladas de um modo que melhore a qualidade e reduza o erro. (PRESSMAN,2007).

A mudança é um fato na vida do desenvolvimento de software. Clientes querem modificar requisitos. Desenvolvedores querem modificar a abordagem técnica. Administração quer modificar a abordagem ao projeto. Por que todas essas modificações? A resposta é realmente bastante simples. Á medida que o tempo passa todo o envolvido sabe mais sobre o que precisa, e como colocá-la em prática.

É importante fazer uma distinção clara entre manutenção de software e gerenciamento de configuração. A manutenção é um conjunto de atividades de engenharia de software que acontece depois que o software é entregue ao cliente e posto em operação. O gerenciamento de configuração de software é um conjunto de atividades de controle e rastreamento que começa quando um projeto de desenvolvimento de software se inicia e termina somente quando o software é tirado de operação.

Uma meta primordial da metodologia de engenharia de software é melhorar a facilidade com que as mudanças podem ser acomodadas e reduzir a quantidade de esforço despendido quando as mudanças são feitas.

# OBJETIVO

A meta deste trabalho é apresentar a Gerência de Configuração no contexto da engenharia de software, a importância no controle da evolução de sistemas de software, destacar a influência da Gerência de Configuração na qualidade de sistemas de software. Tem como objetivo definir as atividades e procedimentos da gerência de configuração, de forma a permitir que se tenha um maior controle das mudanças que venham a ocorrer nos projetos durante todo o processo, inclusive da mudança de requisitos.

# DESENVOLVIMENTO

**Gerência de Configuração**

O gerenciamento de configuração é o desenvolvimento e o uso de padrões e procedimentos para o gerenciamento de sistemas de software em desenvolvimento.

O gerenciamento de configuração de software (Software Configuration Management – SCM) é uma atividade abrangente que é aplicada em todo o processo de engenharia de software. Uma vez que uma mudança pode ocorrer a qualquer tempo, as atividades de SCM são desenvolvidas para (1) identificar a mudança; (2) controla a mudança; (3) garantir que a mudança esteja adequadamente implementada; e (4) relatar a mudança a outras pessoas que possam ter interesse nela.

Os procedimentos de gerenciamento de configuração definem como registrar e processar mudanças de sistema e os métodos usados para identificar diferentes versões dele. Ferramentas de gerenciamento de configuração são usadas para armazenar versões de componentes do sistema, sistemas construídos com base nesses componentes e rastrear os releases das versões do sistema para os clientes.

O gerenciamento de configuração é considerado algumas vezes parte de gerenciamento da qualidade de software, tendo o mesmo gerente dividindo as responsabilidades do gerenciamento de qualidade e de configuração. O software é inicialmente liberado pela equipe desenvolvimento para a garantia da qualidade. A equipe de GQA verifica se o sistema possui qualidade aceitável. Dessa maneira, ele se transforma num sistema controlado, ou seja, as mudanças que sofrer devem ser acordadas e registradas antes de serem implementadas. Os sistemas controlados são, algumas vezes, chamados de baseline, porque são considerados os pontos de partida para futuras evoluções controladas do sistema.

Há muitas razões por que os sistemas de existem em diferentes configurações. Configurações podem ser produzidas para diferentes computadores, para diferentes sistemas operacionais, incorporando funções específicas de clientes. Os gerentes de configuração são responsáveis por manter a rastreabilidade das diferenças entre versões de software, para assegurar que as novas versões sejam derivadas de maneira controlada e liberar novas versões para clientes certos no momento certo.

A definição e o uso de padrões de gerenciamento de configuração são essenciais para a certificação da qualidade tanto para o padrão ISO 9000 quanto para os padrões CMM e CMMI (Paulk et AL., 1995; Ahern et AL.,2002; Peach, 1996). Em uma empresa, esses padrões devem ser incorporados no manual da qualidade ou no guia de gerenciamento de configuração. Naturalmente padrões genéricos externos podem ser usados como referência para padrões organizacionais detalhados que configuram um ambiente específico.

Em um processo tradicional de desenvolvimento de software baseado no modelo cascata, o software é entregue para a equipe de gerenciamento de configuração depois que o desenvolvimento está completo e os componentes individuais de software foram testados. Essa equipe, portanto, assume a responsabilidade pela construção do sistema completo e pelo gerenciamento de testes de sistema. Defeitos descobertos durante os testes são encaminhados á equipe de desenvolvimento para serem reparados. Depois de reparados, a equipe de desenvolvimento entrega uma nova versão do componente reparado para a equipe de qualidade. Se a qualidade for aceitável, ele pode se tornar uma nova baseline para os desenvolvimentos futuros do sistema.

Baseline é um conceito de gerenciamento de configuração de software que nos ajuda a controlar as mudanças, sem impedir seriamente as mudanças justificáveis.

Em geral a GCS é aplicada apenas quando existe um processo de desenvolvimento bem definido, com atividades agrupadas em fases, constituídas por objetivos bem definidos e documentados. Neste contexto, GCS atua como um suporte sobre o qual as fases do desenvolvimento são conduzidas e os produtos controlados.

**Itens de configuração**

Em um grande sistema de software, pode haver módulos de milhares de códigos-fonte, scripts de testes, documentos de projeto etc. Eles são produzidos por pessoas diferentes e, quando criados, podem ser denominados com nomes similares ou idênticos. Para manter a rastreabilidade de todas as informações de maneira que o arquivo certo possa ser encontrado quando for necessário, você necessita de um esquema de identificação consistente para todos os itens no sistema de gerenciamento de configuração.

Entende-se como item de configuração “Cada um dos elementos de informação que são criados durante o desenvolvimento de um produto de software, ou que para este desenvolvimento sejam necessários, que são identificados de maneira única e cuja evolução é passível de rastreamento” (Pressman em [PRE 92]).

Nesta definição, tanto os documentos como os arquivos-fonte que compõem um produto de software são Itens de Configuração (IC), assim como também o são as ferramentas de software necessárias para o desenvolvimento.

Se o item de configuração for composto exclusivamente de software, ele é então caracterizado como Item de Configuração de Software (ICSW).

Qualquer IC constitui uma unidade funcional que possui um ciclo de desenvolvimento e acompanhamento de GCS próprios. Qualquer sistema em desenvolvimento deve ser  particionado em itens de configuração, e o seu desenvolvimento é visto como o desenvolvimento de cada um dos ICs. Cada IC, por sua vez pode ser particionado em outro conjunto de ICs e assim sucessivamente, até que se resulte um conjunto de ICs não particionáveis, que são desenvolvidos segundo um ciclo de vida propriamente definido. A configuração de um sistema de software passa a ser definida pela configuração do conjunto dos ICSW que o constitui.

Em cada fase do processo de desenvolvimento, um conjunto bem definido de itens de configuração deve ser estabelecido. Tal conjunto representa um estágio do desenvolvimento, o qual é passível de modificações apenas mediante um mecanismo formal de alterações. A este conjunto é dado o nome de Baselines, ou Configurações Base do sistema.



Figura 1. Tipos de itens de configuração

O desenvolvimento com configurações base pode, então, ser resumido nos seguintes pontos:

* Caracterização do ciclo de vida, identificando-se as fases pelas quais o desenvolvimento do software irá passar e, dentro delas, as atividades a serem realizadas e os produtos a serem desenvolvidos.
* Definição do conjunto de baselines. Para cada baseline planejada, deve se estabelecer quais serão os ICs que a irão compor e quais as condições impostas para seu estabelecimento;
* Baselines representam marcos no processo de desenvolvimento: uma nova baseline é estabelecida no final de cada fase do ciclo de vida do software;
* Durante cada fase, o desenvolvimento dos ICs a ela referentes está sob total controle de seus desenvolvedores, e realiza-se com ampla liberdade, podendo os ICs ser criados e modificados com bastante facilidade;
* Durante cada fase, entretanto, a modificação de uma configuração base anteriormente estabelecida somente pode ser feita de forma controlada, mediante um processo bem definido;
* Ao ser estabelecida cada baseline incorpora integralmente a anterior. Desta forma, em qualquer instante do desenvolvimento, a última baseline estabelecida representa o estado atual do desenvolvimento como um todo;
* O estabelecimento de cada baseline somente é realizado após ser aprovada por procedimentos de consistência interna, verificação e validação.

Desta forma é possível ter um controle sistemático sobre todos os itens de configuração abordados em cada fase do ciclo de vida do software, assim como é possível avaliar mais facilmente o seu grau de evolução.

**Resultados Esperados da Gerência de Configuração MSP.BR**

O propósito do processo Gerência de Configuração é estabelecer e manter a integridade de todos os produtos de trabalho de um processo ou projeto e disponibilizá-los a todos os envolvidos. Resultados esperados:

**GCO 1.** Um Sistema de Gerência de Configuração é estabelecido e mantido;

**GCO 2.** Os itens de configuração são identificados com base em critérios estabelecidos;

**GCO 3.** Os itens de configuração sujeitos a um controle formal são colocados sob baseline;

**GCO 4.** A situação dos itens de configuração e das baselines é registrada ao longo do tempo e disponibilizada;

**GCO 5.** Modificações em itens de configuração são controladas;

**GCO 6.** O armazenamento, o manuseio e a liberação de itens de configuração e baselines são controlados;

**GCO 7.** Auditorias de configuração são realizadas objetivamente para assegurar que as baselines e os itens de configuração estejam íntegros, completos e consistentes.

**Atividades de GCO**

Atividades de controle de configuração incluem a requisição, avaliação, aprovação e implementação de mudanças em itens de configuração. Mudanças podem incluir tanto correções de erros quanto melhoramentos. O grau de formalidade necessário para o processo de controle de mudança depende da linha base afetada e do impacto da mudança na estrutura da configuração.

No decorrer dos projetos, serão geradas revisões na ferramenta Subversion. Nestas revisões serão armazenadas as atualizações no cronograma, as atualizações do demonstrativo de coleta de métricas, os registros de reuniões produzidos e os documentos alterados que a gerência do projeto julgue importante sua inclusão na revisão.

Todos os documentos incluídos nas revisões são revistos pela gerência de configuração antes de sua inclusão na revisão. Pendências encontradas durante a geração destas revisões deverão ser anotadas e poderão ser ajustadas posteriormente, desde que, com o consentimento do gerente do projeto, visto que os artefatos já foram revisados anteriormente pela qualidade no decorrer das atividades.

Ao final de cada fase do projeto e das iterações serão geradas revisões que serão consideradas configurações-base (baseline) do projeto. Para estas revisões todos os documentos incluídos deverão estar revisados e aprovados pelos envolvidos.

A seguir apresentarei uma ferramenta que nos auxilia

# RESULTADOS

Nesta seção, descreve-se os resultados obtidos a partir das implementações e estudos realizados.

# CONCLUSÃO

Descrição das conclusões obtidas com a pesquisa.

# REFERÊNCIAS BIBLIOGRÁFICAS

PRESSMAN, Roger. **Engenharia de Software.** São Paulo: Pearson Makron Books, 1995. 916 p.

SOMMERVILLE, Ian. **Engenharia de Software.** São Paulo: Pearson Makron Books, 12007. 455 p.

FERREIRA, Gladstone. **Gerência de Configuração.** 2004. Disponível em: < http://www.cin.ufpe.br/~gfn/qualidade/gc.html>. Acesso em: 25 MAR. 2016.

SOBRENOME1, Nome1; SOBRENOME2, Nome2; SOBRENOME3, Nome3. Título do artigo. **Nome da Revista,** Cidade Onde O Artigo Foi Publicado, v. 1, n. 1, p.1-3, 01 mar. 2014. Periodicidade. Disponível em: <http://www.enderecodosite.com>. Acesso em: 01 fev. 2015.

# APÊNDICE A – Título do apêndice

Apêndices são documentos elaborados pelo(a) autor(a) deste trabalho.

# ANEXO A – Título do anexo

Anexos são documentos não elaborados pelo(a) autor(a) deste trabalho.