



# Engenharia de Software

---

Alessandreia Marta de Oliveira  
alessandreia@gmail.com



# **Gerência de Projetos**

---



# Gerência de Projetos

---

- É a primeira “camada” do processo de ES
  - Ele abrange todo o processo de desenvolvimento do começo ao fim (atividades que envolvem medição, estimativas, análise de erros, programação de atividades, monitoração e controle)
- Começa antes do trabalho técnico, prossegue à medida que o software se desenvolve do modelo conceitual para a realidade e encerra somente quando o software se torna absoluto



# Foco da Gerência

---

- Pessoas
- Problema
- Processos



# Problema

---

- Antes de planejar é necessário:
  - Definir os objetivos e o escopo do projeto
  - Considerar as alternativas de solução
  - Identificar as restrições técnicas e gerenciais



# Processo

---

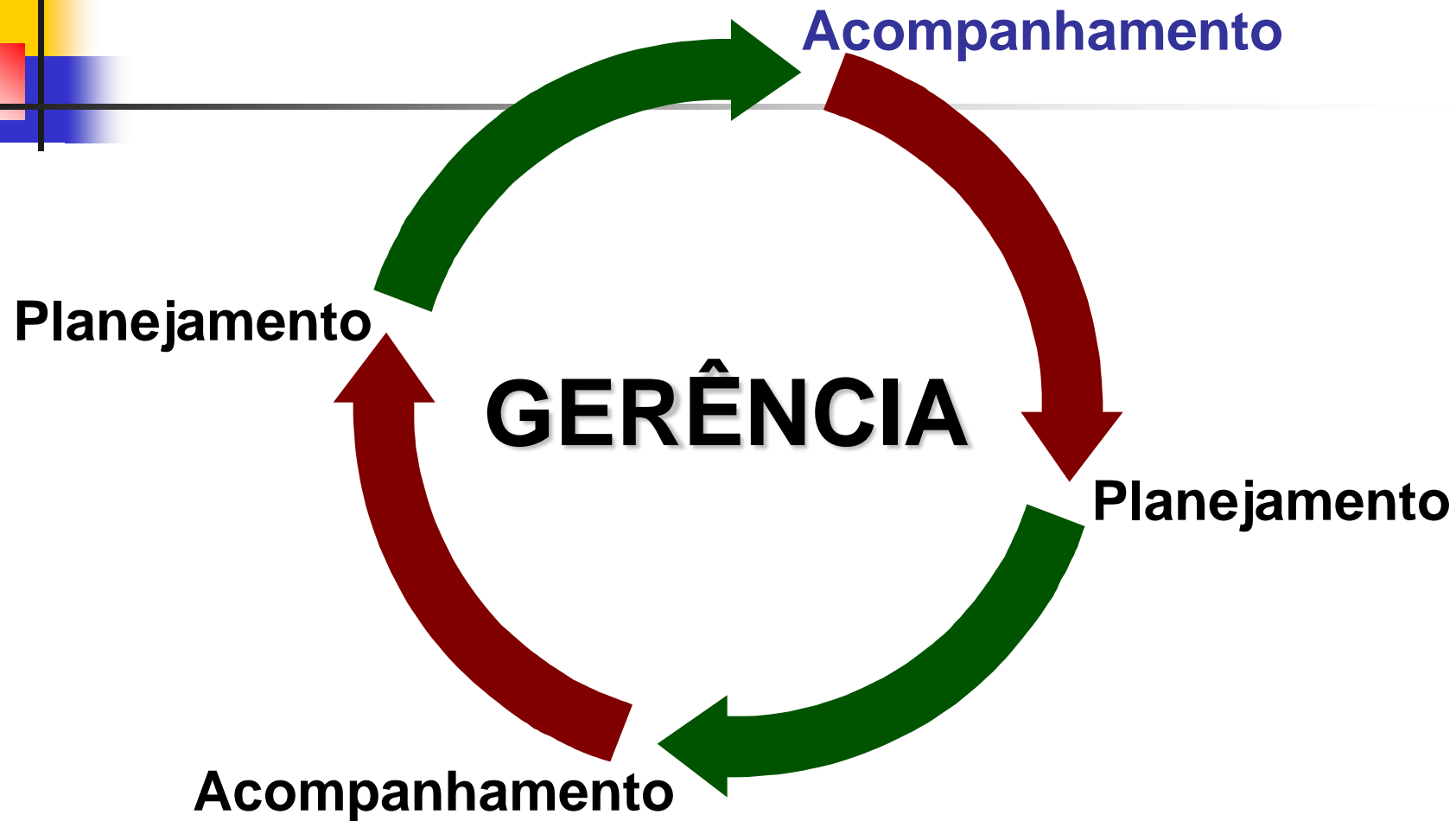
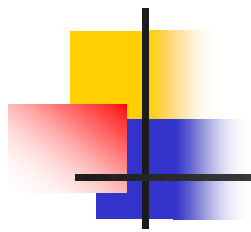
- Fornece um *framework* a partir do qual pode-se estabelecer um plano para o desenvolvimento do projeto



# Gerência de Projetos de Software

---

- Elaboração de propostas
- Planejamento e programação de projetos
- Custo do projeto
- Monitoramento e revisões do projeto
- Seleção e avaliação de pessoal
- Elaboração de relatórios e apresentações







# Plano de projeto

---

- Introdução
- Organização do projeto
  - Equipe e papéis
- Análise de riscos
- Requisitos de hardware e software



# Plano de projeto

---

- Estrutura analítica
  - Divisão do trabalho em atividades, marcos e produtos a serem entregues
- Programação de projeto
  - Descreve a dependência entre atividades, tempo estimado e alocação de pessoas
- Mecanismos de monitoramento e de elaboração de relatórios



# Escopo

---

- Primeira atividade no planejamento de projetos de software é a definição do escopo
- Deve ser claro e compreensível tanto em nível técnico como administrativo
- O escopo deve ser delimitado:
  - Dados quantitativos são declarados explicitamente
  - As restrições e limitações são anotadas
  - Fatores esclarecedores são descritos



# Escopo

---

- O escopo do software descreve:
  - Funções (usar refinamento para detalhar)
  - Desempenho (requisitos de processamento e de tempo de resposta)
  - Restrições (limites impostos pelo hardware externo, memória disponível, outros sistemas existentes)
  - Interfaces (considerar natureza e complexidade de cada interface de hardware, software, pessoas que utilizam o software e outros terminais e procedimentos que precedem ou sucedem o software como uma série seqüencial de operações)
  - Confiabilidade (observar as características específicas como parâmetro)



# Estimativas

---

- Muitas vezes cronograma e orçamento são estabelecidos por razões de negócio
- O papel das estimativas muitas vezes é verificar a viabilidade do que foi estabelecido
- Para estimar é necessário ter-se:
  - Escopo do projeto
  - Visão de alto nível das funcionalidades
  - Avaliação da dificuldade e complexidade
- Técnicas
  - Pontos por Função



# Estimativas de Custos

---

- Nenhuma alternativa é melhor do que a outra sob todos os aspectos
  - Os pontos fortes e fracos das técnicas se complementam
  - O melhor é se utilizar uma combinação das técnicas e comparar e interagir as estimativas obtidas de cada uma



# Cronograma

---

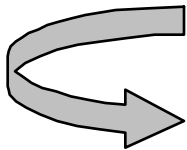
- Todo software tem uma programação de atividades para sua realização, mas nem todos são criados igualmente
- A programação do projeto de software, como qualquer outro projeto, inclui identificação de:
  - Um conjunto de tarefas, estabelecimento das interdependências entre as tarefas, estimativa do esforço associado a cada tarefa e atribuição de pessoas e recursos
  - Uma “rede de tarefas” é criada



# Elaboração de Cronograma

---

- Realizada a partir de:
  - Processo de software
  - Estimativas
  - Análise de riscos
  - Recursos humanos disponíveis



- Identificação de:

- Independência e paralelismo de tarefas
- Tempo





# Recursos

---

- As ferramentas - hardware e software - sustentam o desenvolvimento e no nível mais elevado temos os recursos humanos - pessoas
- Cada recurso deverá ser especificado segundo as seguintes características:
  - Descrição do recurso
  - Declaração da disponibilidade
  - Tempo cronológico em que o recurso será exigido
  - Tempo da aplicação ou uso



# Recursos Humanos

---

- O planejador deve avaliar o escopo e selecionar as habilidades exigidas para o desenvolvimento como postos organizacionais (gerente, engenheiro de software, etc) e especialidades (telecomunicações, banco de dados, microprocessador, etc)
- O número de pessoas exigido só pode ser determinado depois que uma estimativa do esforço de desenvolvimento for feito (pessoa-mês, pessoa-ano)



# Recursos de Hardware

---

- Considerar
  - Hardware de desenvolvimento: computador e periféricos usados durante o desenvolvimento
  - Hardware de produção
  - Outros elementos: máquinas e equipamentos específicos necessários como parte de uma etapa de desenvolvimento



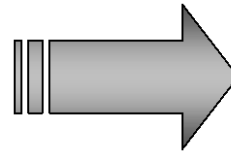
# Recursos de Software

---

- Todos os recursos do ambiente de desenvolvimento de software a ser utilizado
  - Ferramentas CASE
    - Ferramentas de Planejamento de Sistemas de Informações
    - Ferramentas de Gerenciamento de Projetos
    - Ferramentas de Apoio, Análise e Projeto
    - Ferramentas de Programação, de Integração e Testes
    - Ferramentas de Construção de Protótipos e Simulação
    - Ferramentas de Manutenção
    - Entre outras.

# Acompanhamento e Controle

- Foco do Controle



**Qualidade  
Alterações**

- Acompanhamento e Controle são mais efetivos quando integrados ao processo de software
  - Processo fornece marcos e pontos de controle que devem ser usados para acompanhamento do projeto



# **Análise de Riscos**

---



# Planejamento de Projetos

---

- Planejar é decidir antecipadamente o que fazer, como fazer, quando fazer e quem deve fazer
- Todo projeto deve começar com um Plano
- O Plano do Projeto tem como foco os objetivos do projeto, as ações necessárias para atingir estes objetivos e os potenciais riscos e problemas que podem afetar se atingir estes objetivos



# Por que planejar?

---

- evitar o fracasso
- prever custos, recursos, prazos e riscos
- analisar alternativas
- organizar
- preparar-se para alterações
- poder acompanhar o andamento do projeto
- planejar melhor da próxima vez





# Quando planejar?

---

- o planejamento começa de forma macroscópica no início do projeto
- o planejamento é revisto e detalhado ao longo do projeto



# Plano do Projeto

---

- Introdução do Projeto (histórico, escopo, objetivos e público alvo)
- Plano do Processo de Desenvolvimento
- Plano de Recursos e Produtos
- Plano de Riscos
- Plano de Organização



# Plano do Projeto

---

- Plano de Documentação
- Plano de Acompanhamento
- Plano de Controle da Qualidade
- Plano de Treinamento
- Plano de Gerência de Configuração
- Plano de Implantação e Operação
- Glossário



# Gerenciamento de Riscos

---

- Risco: “possibilidade de sofrer dano ou perda; perigo”
- Para projetos de desenvolvimento de software o dano ou perda pode ser:
  - diminuição da qualidade do produto
  - aumento de custos
  - atraso no cronograma
  - falha total do programa



# Plano de Riscos

---

- Relaciona-se com a análise de aspectos desconhecidos do projeto
  - São esses aspectos que podem fazer com o projeto fracasse!
- Risco
  - Fator, elemento, acontecimento, qualquer coisa que, se concretizada, pode interferir no sucesso do projeto

# Atividades da Gerência de Riscos



---

- Identificar riscos
- Avaliar riscos
- Classificar riscos
- Hierarquizar riscos
- Planejar como mitigar riscos
- Acompanhar riscos e Plano
- Rever e ajustar Planos

# Atividades da Gerência de Riscos



---

- Identificar riscos
  - O primeiro passo é listar os riscos e torná-los visíveis para todos
  - Nada tem mais sucesso em tornar gerentes interessados nos riscos do que uma lista de 100 ou mais riscos, analisados e hierarquizados
  - Quando está no papel os gerentes se sentem obrigados a tomar medidas

# Atividades da Gerência de Riscos



---

- Avaliar riscos
  - Avaliação qualitativa ou quantitativa?
  - Há casos onde se pode obter a probabilidade de um evento futuro ocorrer, mas são raros.
  - Em geral uma quantificação prematura não é necessária.
  - Cuidado para não gastar mais tempo avaliando o risco, do que o tempo que seria gasto se este ocorrer.





# Atividades da Gerência de Riscos

---

- Projeção de Riscos
- Classifica os riscos de duas maneiras:
  - a probabilidade de que o risco seja real
  - as conseqüências dos problemas associados ao risco, caso ele ocorra – os efeitos.

# Atividades da Gerência de Riscos



---

- Projeção de Riscos
- Para calcular a probabilidade do risco ocorrer pode-se usar a escala:
  - Muito baixa, baixa, moderada, alta e muito alta
- Para avaliar os efeitos do risco pode-se usar a escala:
  - Catastrófico, sério, tolerável, insignificante



# Atividades da Gerência de Riscos

---

- Avaliação dos Riscos
  - Examina mais detalhadamente a precisão das estimativas do risco ocorrer, determina uma ordem de prioridade para os riscos que foram descobertos e define maneiras de controlar e ou evitar riscos que tem probabilidade de ocorrer

# Atividades da Gerência de Riscos



---

- Avaliação dos Riscos
  - Para a maioria dos projetos de software o custo, os prazos e o desempenho representam níveis de risco referentes, momento (break point) em que o projeto pode parar caso estes riscos ocorram ultrapassando o custo, o cronograma, degradação do desempenho ou outra combinação entre eles

# Atividades da Gerência de Riscos



---

- Classificar riscos
  - um único risco pode dar origem a várias estratégias de controle de risco.
  - Classificar/agrupar declarações de riscos em categorias baseando-se em características comuns ajuda a encontrar riscos globais que podem ser resolvidos juntos.

# Atividades da Gerência de Riscos



---

- Hierarquizar riscos
  - Deve-se tratar primeiro os riscos mais importantes (maior probabilidade de ocorrer e maior impacto) e procurar ver para quais destes se tem recursos para mitigar
  - Em geral é difícil lidar com mais de 10 riscos

# Atividades da Gerência de Riscos



---

- Planejar como mitigar riscos
  - Nem todos os riscos podem ser mitigados
  - Deve-se identificar:
    - riscos com os quais se pode conviver caso se tornem realmente problemas
    - riscos para os quais deve-se designar um responsável mais capaz de gerenciá-lo

# Atividades da Gerência de Riscos



---

- Acompanhar riscos e Plano
  - Com um objetivo claro para mitigar riscos, pode-se determinar se o esforço para mitigar riscos está tendo efeito
  - Podem ser necessários relatórios e dados para acompanhar o status de riscos críticos
  - Relatórios devem ser escritos evitando-se a comunicação oral





# Gerenciamento e Monitoramento dos Riscos

---

- Os passos de gerenciamento dos riscos são desenvolvidos a partir da descrição, probabilidade e impacto de cada risco
- Como a administração de riscos acarreta custos é essencial uma análise de custo-benefício para avaliação de quando os benefícios advindos das atividades tomadas para evitá-los são ultrapassados pelos custos associados à implementação dos mesmos.



# Gerenciamento e Monitoramento dos Riscos

---

- A fase de monitoramento dos riscos é uma atividade de rastreamento do projeto: avaliar se um risco previsto de fato ocorreu, garantir que os passos de reversão estão sendo adequadamente aplicados e coletar informações que possam ser usadas em análises de riscos futuras



# Atividades da Gerência de Riscos

---

- Rever e ajustar Planos
  - Controlar riscos significa
    - alterar a estratégia para mitigar erros se ela não se mostrar efetiva
    - tomar medidas quando um risco se tornar importante a ponto de de requerer ações para mitigá-lo
    - seguir um planejamento de contingências
    - encerrar um risco quando este não mais existir



# Taxonomia de Riscos

---

- Riscos de Estimativas
  - Os problemas ao se tentar desenvolver um produto difícil ou com menos pessoal do que o desejado são seriamente exacerbados quando não existe tempo suficiente. A taxa de solução de defeitos é subestimada. O tamanho do software é subestimado. Planejamento baseado em estimativas inadequadas



# Taxonomia de Riscos

---

- Consequências
  - custos mais altos
  - atraso no cronograma
  - entrega de um produto não adequado



# Taxonomia de Riscos

---

- Riscos da Tecnologia
  - Problemas relacionados a sistema operacional, hardware e ambiente de desenvolvimento imaturos
  - Banco de dados que não processa o esperado
  - Erros devidos a upgrades
  - Componentes reutilizados com defeito.
  - Ferramentas case geram código ineficiente.



# Taxonomia de Riscos

---

- Riscos de Pessoal
  - Principal fonte de riscos em projetos
    - Impossibilidade de recrutar pessoal com a habilidade requerida
    - Pessoas importantes não disponíveis em períodos cruciais (doenças por exemplo)
    - O treinamento necessário não está disponível
    - Pessoal pouco motivado
    - Relacionamento pouco satisfatório entre membros.



# Taxonomia de Riscos

---

- Riscos do Processo
  - Problemas relacionados a métodos e ferramentas
    - Os riscos aumentam consideravelmente se ferramentas e métodos comprovadamente adequados não são usados
  - Problemas decorrentes de não detectar erros no início do projeto afetam custo e tempo das fases finais
  - Problemas de comunicação quando o projeto é desenvolvido em vários locais
  - Processo de desenvolvimento ad hoc





# Taxonomia de Riscos

---

- Risco organizacional
  - A organização está organizada de modo que diferentes gerencias são responsáveis pelo projeto
  - Problemas financeiros organizacionais forçam reduções no orçamento do projeto
  - Fofocas na empresa
  - Falta de iniciativa.



# Taxonomia de Riscos

---

- Riscos de requisitos
  - São propostas mudanças nos requisitos, que exigem significativo retrabalho
  - Os clientes não compreendem o impacto das mudanças nos requisitos
  - Muitas reclamações dos clientes.



# Taxonomia de Riscos

---

- Riscos de ferramentas
  - Relutância dos membros da equipe em adotar ferramentas
  - Reclamações sobre ferramentas CASE
  - Solicitações de estações de trabalho com maior capacidade



# Riscos

---

- Os dez riscos mais comuns
  - Perda de pessoal
  - Cronograma e custos não realísticos
  - Desenvolvimento de funções erradas
  - Desenvolvimento da interface com o usuário errada
  - Subjulgamento dos requisitos



# Riscos

---

- Mudanças constantes dos requisitos
- Perda no fornecimento de componentes externos
- Perda na performance de tarefas externas
- Perda de desempenho em tempo real
- Incompatibilidade com técnicas de ES



# Importante

---

- Guardar informações sobre os riscos de um projeto
  - Documentar o que se aprendeu
  - Banco de Dados de Riscos



# Plano de Risco

---

- Elementos
  - Descrição do risco
  - Tipo de risco
  - Probabilidade
  - Efeito
  - Estratégia de administração



# Manutenção

---





# Manutenção

---

- A manutenção de software pode ser responsável por até 70% do custo de um projeto.
- As mudanças no software são inevitáveis por isso é necessário desenvolver mecanismos para avaliar, controlar e fazer modificações.
- Uma meta primordial da Engenharia de Software é facilitar as mudanças e reduzir o esforço despendido em manutenção.



# Tipos de manutenção

---

- Corretiva:
  - Processo que inclui o diagnóstico e a correção de erros, na maioria das vezes não detectados na fase de testes.
- Adaptativa
  - Processo que inclui a modificação do software para que ele tenha interface adequada com o ambiente de hardware e software, em constante mutação.



# Tipos de manutenção

---

- Perfectiva

- Processo de modificação de um software para incorporar novas funcionalidades e ampliações gerais solicitadas pelos usuários.

- Preventiva

- Processo de modificação do software para melhorar a confiabilidade ou a manutenibilidade futura ou mesmo para oferecer uma base melhor para futuras ampliações



# Características da Manutenção

---

- Atividades específicas exigidas
- Custos elevados
- Problemas específicos



# Gerência de Configuração e Manutenção

---

- Quando existe gerência de configuração a documentação do software existe e portanto a manutenção é facilidade e estruturada seguindo o seguinte ciclo de aplicação:



# Ciclo da Manutenção

---

- Avaliar o projeto
- Planejar a abordagem
- Modificar o projeto
- Recodificar
- Rever
- Liberar



# Problemas da Manutenção

---

- Insatisfação do cliente do cliente quando solicitações aparentemente legítimas de reparo ou modificações não podem ser encaminhadas a tempo
- Baixa qualidade do software como resultado de mudanças que introduzem erros latentes no software
- Baixa valorização da atividade de manutenção.



# Problemas da Manutenção

---

- PAGUE AGORA OU PAGUE MUITO MAIS DEPOIS.
- Documentação inadequada do software
- Dificuldade de rastreamento do desenvolvimento do software
- Poucos elementos da gerencia de configuração e dificuldade de entendimento do código de outro programador.





# Problemas da Manutenção

---

- Mobilidade do pessoal da área de software.
- Documentação ruim ou não existente.
- A maioria do software não é projetado para sofrer mudanças (sem independência funcional ou classes de objetos).
- Pouca valorização do trabalho de manutenção.



# Manutenibilidade

---

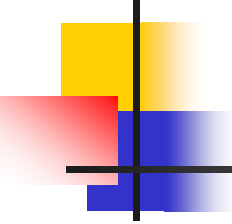
- Fatores que facilitam a manutenção do software, associados ao ambiente de desenvolvimento:
  - Pessoal qualificado
  - Facilidade de manuseio do software
  - Uso de Linguagem de Programação padronizada
  - Uso de Sistemas Operacionais padronizados
  - Documentação padronizada
  - Casos de teste
  - Disponibilidade de um computador adequado para realizar a manutenção.



# Efeitos da Manutenção

---

- A modificação de software é perigosa.
- Toda vez que uma mudança é introduzida num procedimento lógico complexo, o potencial de erros cresce.



# Cuidados com as Codificações

---

- Mudanças que podem causar erros:
  - Subprograma é apagado ou modificado
  - Label é apagado ou modificado
  - Identificador é apagado ou modificado
  - Melhorias de desempenho
  - Modificação na abertura e fechamento de arquivos
  - Modificação de operadores lógicos
  - Modificação de testes lógicos.



# Cuidados com os Dados

---

- Mudanças que podem causar erros:
  - Redefinição de constantes e variáveis locais e globais
  - Redefinição de formatos de registros ou arquivos
  - Aumento ou diminuição no tamanho de um array ou outra estrutura
  - Rearranjo de argumentos de subprogramas.



# Cuidados com a Documentação

---

- A manutenção deve concentrar-se em toda configuração do software e não somente nas modificações do código.
- As modificações devem ser refletidas na documentação e nos manuais do usuário.



# A Engenharia Reversa e a Reengenharia

---

- A engenharia reversa extrai informações do projeto do código-fonte quando nenhuma outra documentação encontra-se á disposição e refaz o projeto.
- A reengenharia pega as informações obtidas e reestrutura o programa para conseguir uma qualidade mais alta e melhor manutenibilidade.



# Gerência de Configuração de Software

---

- Atividade de apoio aplicada ao longo de todo o processo de software
- É a arte de organizar, identificar e controlar modificações em um software que está em desenvolvimento.
- Objetivo: gerenciar e manter a integridade dos produtos gerados ao longo do ciclo de vida do software, maximizando a produtividade e minimizando a quantidade de erros.



# Gerência de Configuração de Software

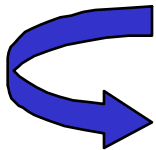


---

- É o gerenciamento de mudanças no sistema, quando é feita uma alteração no sistema o objetivo da gerência de configuração é garantir que as mudanças sejam incorporadas de maneira controlada.
- Um esquema formal de denominação de documentos deve ser estabelecido e utilizado como base para o controle das diferentes versões de todos os documentos do projeto.
- A equipe deve ser apoiada por um banco de dados e os projetos devem ter um meio formal de solicitar mudanças no sistema.

# Gerência de Configuração X Manutenção

- Manutenção: atividades de Engenharia de Software que ocorrem depois que o software é colocado em operação
- Gerência de Configuração: atividades de controle que começam no início do projeto e só terminam quando o software sai de operação.



- É um elemento importante para garantia da qualidade de software