

Aula 01 - Fundamentos de Manutenção de Software

Disciplina: Manutenção de Software

Prof. Me. João Paulo Biazotto

Agenda

- Revisão de processos de software
- Tipos de manutenção
- Leis de Lehman

Processo de Software

Processo

Requisitos

Projeto (análise,
projeto, arquitetura)

Implementação

Manutenção

Processo de Software

- **Especificação:** Defina a funcionalidade do software e as restrições sobre sua operação.
- **Projeto e implementação:** O software deve ser desenvolvido conforme a especificação.
- **Validação:** O software deve ser validado para assegurar que atenda às expectativas do cliente.
- **Evolução:** O software deve evoluir para atender aos novos requisitos que surgirem.

Manutenção/Evolução

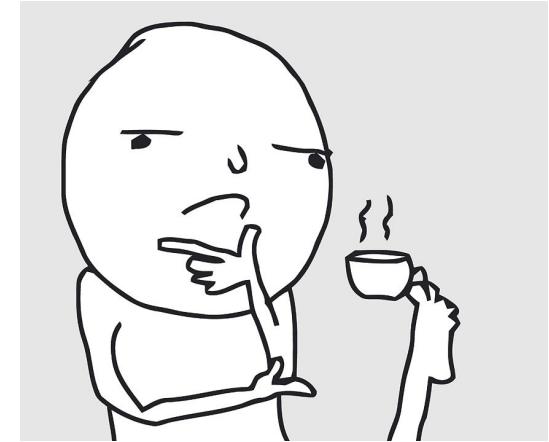
- Disciplina responsável por lidar com as **mudanças relacionadas** ao software depois da sua **entrega**.
- Software: código + documentação (requisitos, análise, projeto, manuais, etc)
- “Ato de manter uma entidade em bom estado de reparo, eficiência ou validade, para evitar falhas ou declínio” (IEEE)

Desenvolvimento x Manutenção

- Adicionar uma nova funcionalidade durante o desenvolvimento é mais fácil que durante a manutenção
- Manutenção de software deve respeitar certos parâmetros e restrições existentes

Funcionalidades

Por que é mais fácil adicionar uma funcionalidade durante o desenvolvimento?



Desenvolvimento x Manutenção

- Quando realiza-se mudança em uma casa
 - Arquitetos e construtores devem ter cuidado para não enfraquecer a estrutura da casa.
 - Custo de um novo cômodo pode ser barato, mas o custo por metro quadrado é muito mais alto.
 - Isso ocorre pois é necessário a remoção de paredes, alteração do encanamento/fiação, etc.

Desenvolvimento x Manutenção

- Em software, ao projetar uma nova funcionalidade, o mantenedor deve investigar o sistema atual para abstrair sua arquitetura e detalhes de baixo nível:
 - Realizar como a mudança será acomodada
 - Prever o impacto da mudança (efeito cascata)
 - Determinar as características necessárias para o trabalho

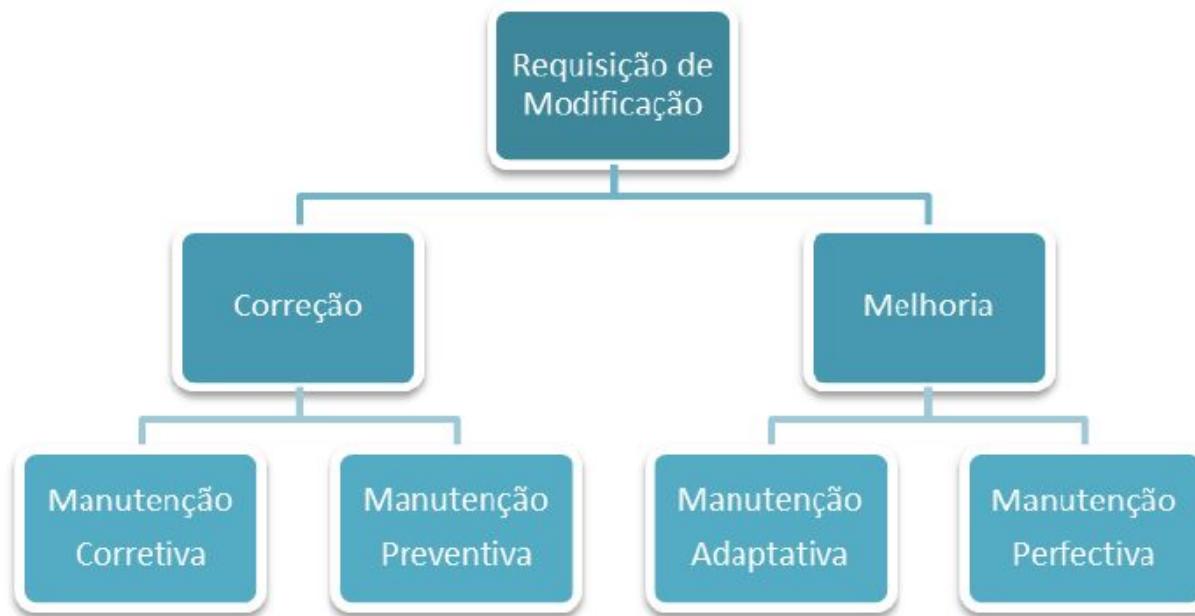
Quando realizar manutenção de software?

- Adicionar novas funcionalidades
- Corrigir defeitos
- Melhorar design
- Comunicar com outros sistemas
- Migrar de SO, BD, bibliotecas, etc
- Adaptar a diferentes hardwares
- Adaptar a leis, regras de negócio, etc
- Refatorar o código

Categorias de Manutenção

- Manutenção Corretiva
- Manutenção Preventiva
- Manutenção Adaptativa
- Manutenção perfectiva ou Evolutiva

Categorias de Manutenção



Manutenção Corretiva

- Modificações no software para corrigir defeitos
- Ex. defeitos em requisitos, projeto, código
- Pode gerar outros problemas como aumento de complexidade e outros efeitos cascatas

Manutenção Preventiva

- Modificações no software para prevenir potenciais problemas no futuro
- Lida com o deterioramento de estruturas, previne falhas e melhorar a manutenibilidade
- Torna os programas mais fáceis de entender e facilita trabalhos de manutenção futuros
- Ex. reestruturação de código, otimização de código, refatoração, atualização de documentação

Manutenção Adaptativa

- Adaptações para manter o software usável devido às alterações no ambiente externo
- Ocorre pois o ambiente está em constante evolução, mesmo quando defeitos não existem
- Ex. alterações no SO, BD, servidor, compilador, bibliotecas, frameworks, hardware
- Se SO é atualizado, mudanças podem ser realizadas no software para acomodar o novo SO

Manutenção Evolutiva

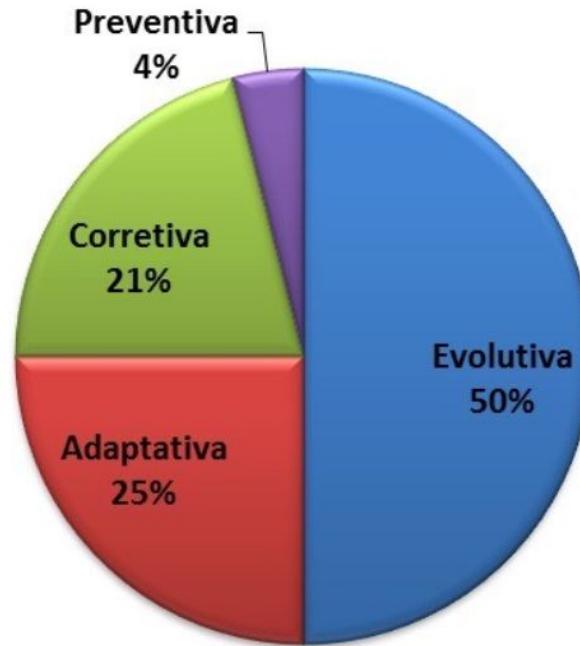
- Modificações para fornecer melhorias aos usuários
- Expande os requisitos do sistema
- Quando o software se torna útil, os usuários solicitam melhorias além do escopo inicial
- Ex. novas funcionalidades

Categorias de Manutenção

Quanto de cada tipo de manutenção é mais feita?



Categorias de Manutenção



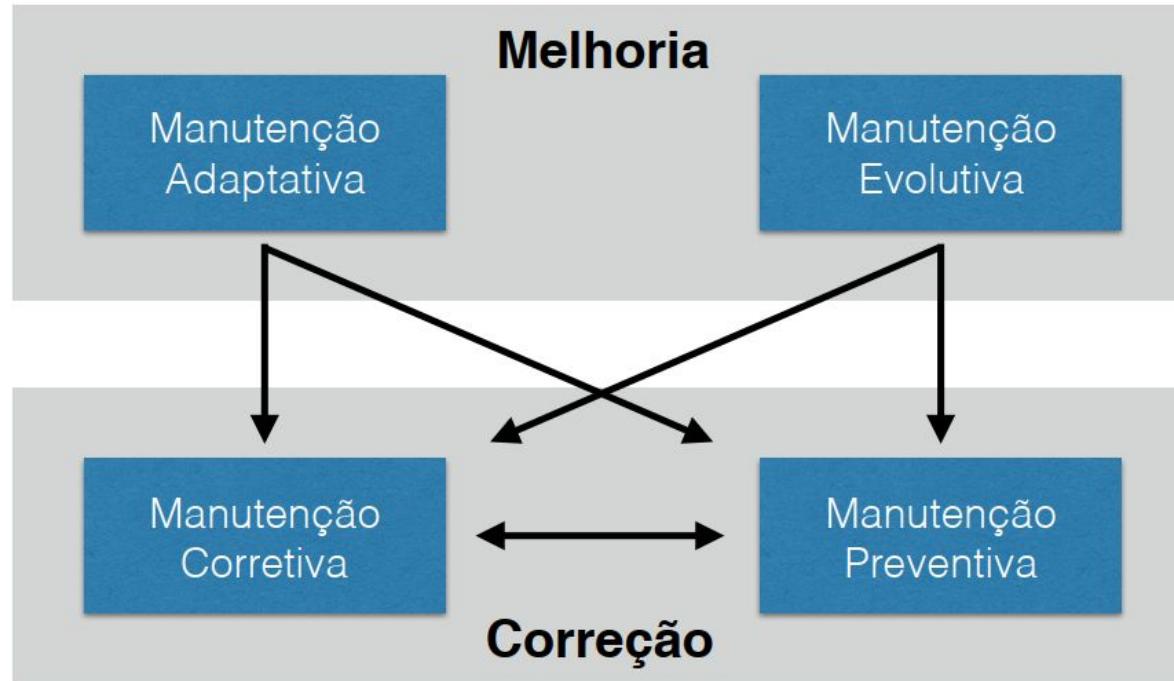
Categorias de Manutenção

- Manutenção **corretiva** e **evolutiva** são mais visíveis e trazem valor direto para o usuário
- Manutenção **preventiva** e **adaptativa** trazem valor indireto para o usuário (por quê?)

Relação entre os tipos de manutenção

- Manutenções são categorizadas de forma individual, mas na prática estão interligadas
- Ao modificar o código devido a uma nova biblioteca (**adaptativa**), defeitos podem ser introduzidos. Logo, esses defeitos devem ser corrigidos (**corretiva**)
- A introdução de uma nova funcionalidade (**evolutiva**), pode requerer que o código seja refatorado antes para facilitar sua implementação (**preventiva**)

Relação entre os tipos de manutenção



Leis de Lehman

- O termo manutenção de software data dos anos 60/70
- Lehman após 20 anos de pesquisas, formulou as Leis da evolução (8 leis da evolução)

Leis de Lehman

- **Mudança contínua:** software deve ser continuamente adaptado, senão torna-se menos satisfatório
- **Complexidade crescente:** se não forem tomadas medidas para reduzir complexidade, ela irá aumentar progressivamente
- **Crescimento contínuo:** software deve ter funcionalidades ampliadas para manter a satisfação dos seus usuários; o projeto inicial não inclui tudo e deve ser aumentado
- **Declínio de qualidade:** software perde qualidade se não é alterado para refletir as mudanças do ambiente externo

Leis de Lehman

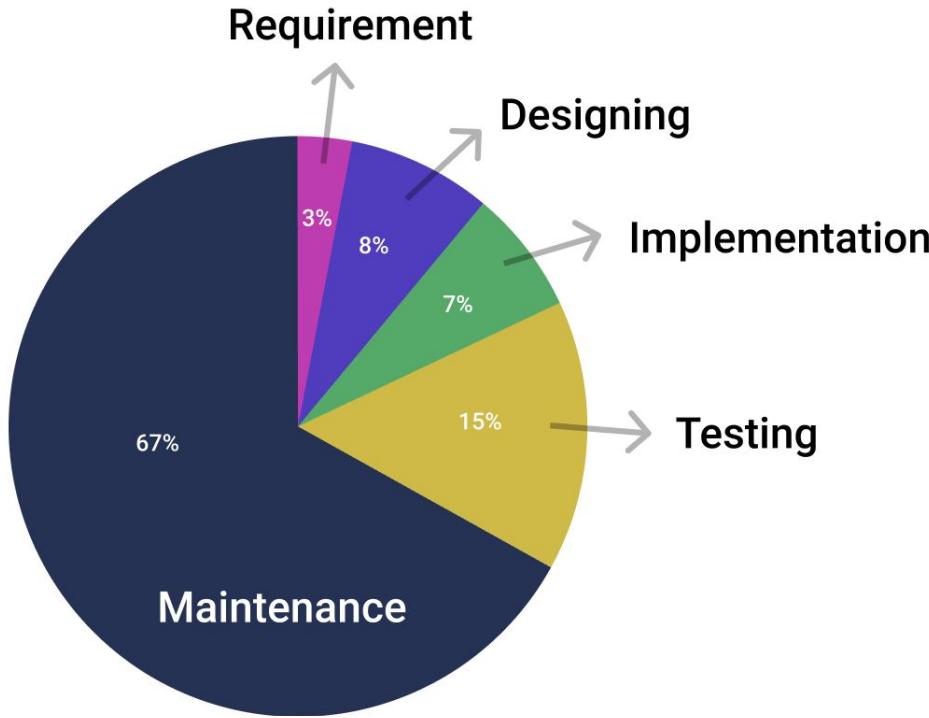
- **Auto regulação:** software tende a se auto regular, seguindo padrões previsíveis e comportamentos disciplinados.
- **Conservação da estabilidade organizacional:** O software tende a se manter em um nível constante em relação à organização
- **Conservação da familiaridade:** Muitas mudanças no software afetam a familiaridade dos usuários com o software
- **Sistema de feedback:** Cada nova versão, gera feedback e novas informações e ajustes

Custo de Manutenção de Software

Qual o custo de manutenção do software?



Custo de Manutenção de Software



Custo de Manutenção de Software

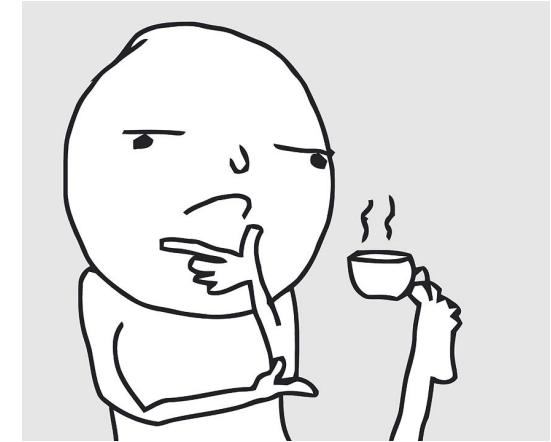
Por que manutenção de software é tão caro?



Custo de Manutenção de Software

Por que manutenção de software é tão caro?

BAIXA MANUTENIBILIDADE!



Manutenibilidade

- Facilidade de manutenção de um sistema
- Difícil de quantificar
- Existem métricas e aspectos do sistema podem ser medidos (ex: complexidade, portabilidade)
- Entretanto não existe fórmula mágica.

Manutenibilidade

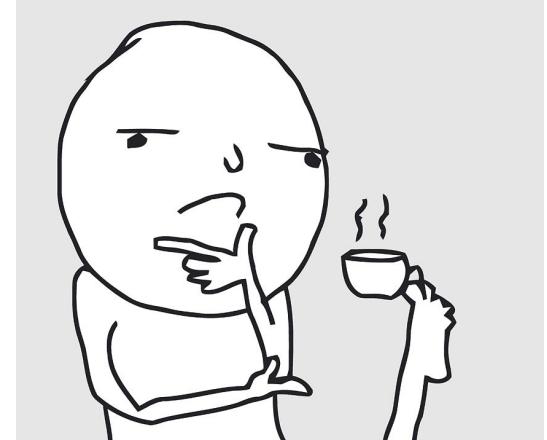
O que impacta a manutenibilidade de um sistema?



Manutenibilidade

O que impacta a manutenibilidade de um sistema?

- Qualidade de código
- Arquitetura
- Acoplamento
- Etc...



Métricas de Manutenibilidade

- Métricas de Chidamber-Kemerer (CK): específicas para OO
 - Profundidade de Herança (DIT)
 - Número de filhos (NOC)
 - Acoplamento entre objetos (CBO)
 - Falta de coesão em métodos (LCOM)
 - Métodos ponderados por classe (WMC)
 - Resposta para classe (RFC)

DÚVIDAS?