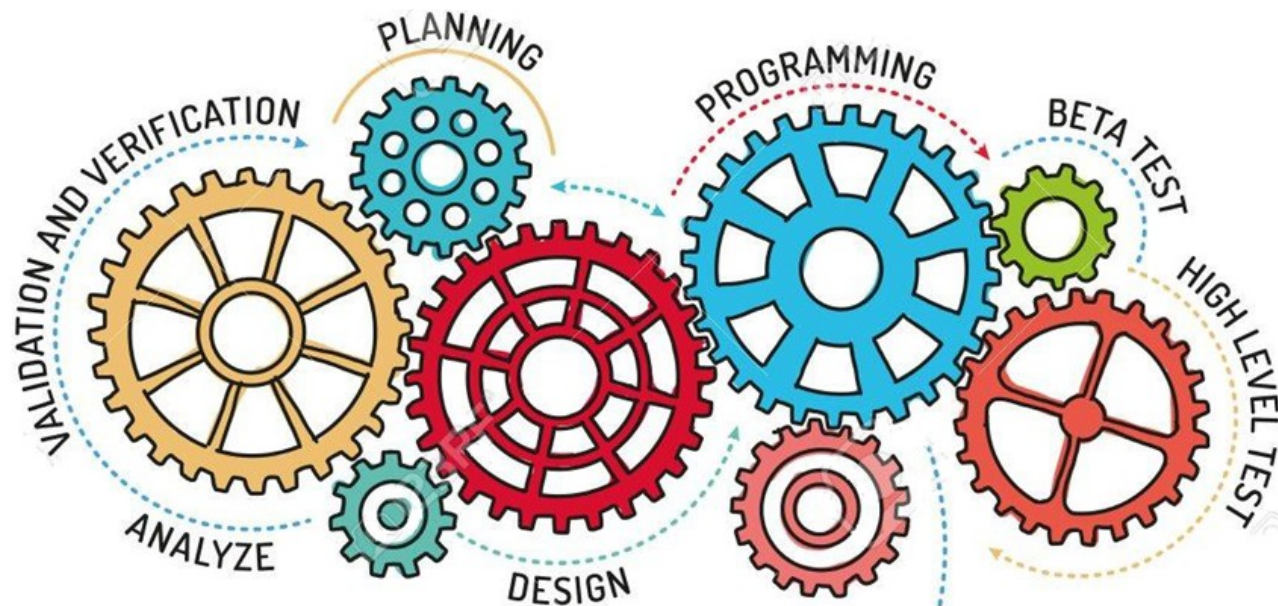


Engenharia de Software



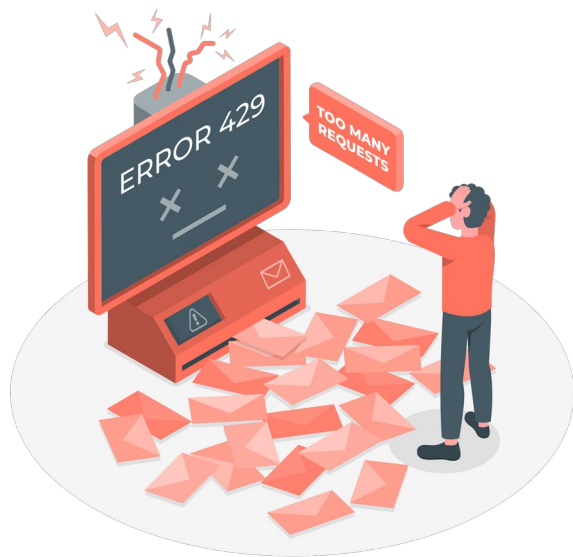
e-mail: clenio.silva@uniube.br



Uniube

Exemplos de falhas de software

- O que podemos entender como falhas em um software?
 - O que as falhas em um software podem causar?
 - Quais cenários podem sofrer impacto dado uma falha de software?



Exemplos de falhas de software

- **Y2K Bug (Bug do Milênio):**
 - No final dos anos 90, muitos sistemas computacionais usavam apenas dois dígitos para representar o ano (por exemplo, "99" para 1999). Isso causou preocupações de que, ao virar o milênio, sistemas poderiam interpretar o ano 2000 como 1900, gerando erros e falhas.
- **Bug do Therac-25:**
 - O Therac-25 era uma máquina de radioterapia usada no tratamento de câncer. Entre os anos 1980 e 1987, erros de software causaram doses de radiação excessivas, resultando em vários casos de morte e ferimentos graves.
- **Falha do Software do Boeing 737 MAX:**
 - O sistema MCAS (Maneuvering Characteristics Augmentation System) do Boeing 737 MAX foi projetado para evitar a estolagem, mas falhou em reconhecer condições de voo corretamente, levando a duas quedas de aviões (Lion Air Flight 610 e Ethiopian Airlines Flight 302).

Crise do Software

- Uma série de desafios e problemas encontrados no processo de desenvolvimento e gerenciamento de software:
 - Década de 1960 – termo “crise do software” foi popularizado dado os problemas enfrentados:
 - dificuldade em estimar custos;
 - prazos
 - falta de métodos sistemáticos para o desenvolver software;
 - Software difícil de manter

Aspectos da Crise de Software

- **Complexidade:** Com a evolução tecnológica e complexidade dos softwares aumenta a dificuldade para os desenvolvedores entenderem e gerenciarem todos os aspectos de um projeto.
- **Qualidade e Confiabilidade:** Sistemas falharam em atender aos requisitos e expectativas dos usuários, resultando em software com bugs, falhas e vulnerabilidades de segurança.
- **Custos e Prazos:** desenvolvimento de software muitas vezes excedeu orçamentos e cronogramas planejados, levando a problemas financeiros e atraso na entrega de produtos.
- **Manutenção:** O custo e a complexidade de manter e atualizar software existente frequentemente superam o custo inicial de desenvolvimento

Processo de Software

- **O que é?**
 - O processo de software refere-se às etapas e práticas sistemáticas usadas para desenvolver, manter e gerenciar software.
- **Metodologia do processo**
 - Uma metodologia de processo estabelece o alicerce para um processo de engenharia de software por meio da identificação de atividades metodológicas aplicáveis a todos os projetos de software.

Componentes e Fases do Processo de Software

- **Planejamento:**
 - Objetivo: definição do escopo, requisitos e metas do projeto.
 - Atividades: criação do cronograma, orçamento, recursos necessários e avaliação de riscos.
 - Resultados: Documento de requisitos, especificações de sistema.
- **Análise de Requisitos**
 - Objetivo: Identificar e documentar as necessidades e expectativas dos stakeholders.
 - Atividades: Coleta de requisitos, análise de requisitos, documentação e validação.
 - Resultados: Documento de requisitos, especificações de sistema.

Componentes e Fases do Processo de Software

- Design

- Objetivo: Criar uma arquitetura e design detalhado para o software.
- Atividades: Design de arquitetura, design de interfaces, modelagem de dados e processos.
- Resultados: Documentos de design, protótipos, modelos de arquitetura.

- Desenvolvimento

- Objetivo: Codificar e construir o software conforme o design especificado.
- Atividades: Programação, integração de componentes, revisão de código.
- Resultados: Código fonte, binários, e documentação técnica.

Componentes e Fases do Processo de Software

- **Teste**

- Objetivo: Verificar se o software atende aos requisitos e identificar e corrigir defeitos.
- Atividades: Criação de casos de teste, execução de testes (unitários, integração, sistema e aceitação), e relatórios de defeitos.
- Resultados: Relatórios de teste, software validado e corrigido.

- **Implantação**

- Objetivo: Colocar o software em operação e disponibilizá-lo para os usuários finais.
- Atividades: Preparação do ambiente de produção, instalação, configuração e treinamento de usuários.
- Resultados: Software operacional, usuários treinados, documentação de implantação.

Componentes e Fases do Processo de Software

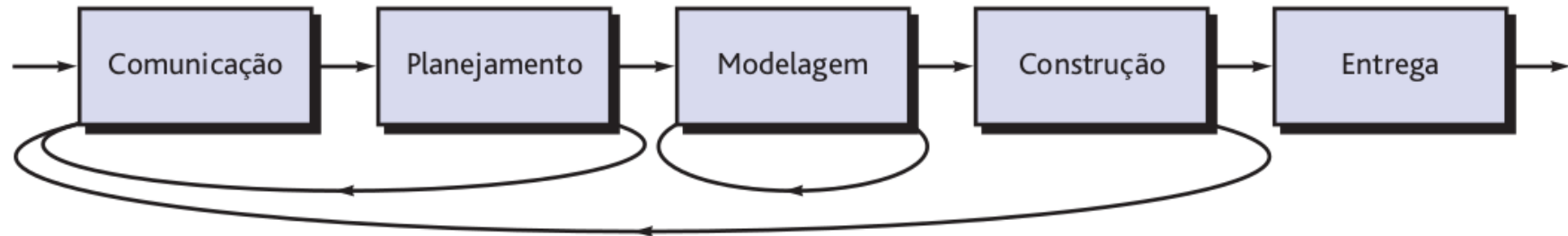
- Manutenção
 - Objetivo: Corrigir defeitos, melhorar o software e adaptá-lo às mudanças no ambiente e requisitos.
 - Atividades: Correção de erros, atualizações e aprimoramentos, suporte contínuo.
 - Resultados: Versões atualizadas do software, correções de bugs, novos recursos.

Modelo Cascata



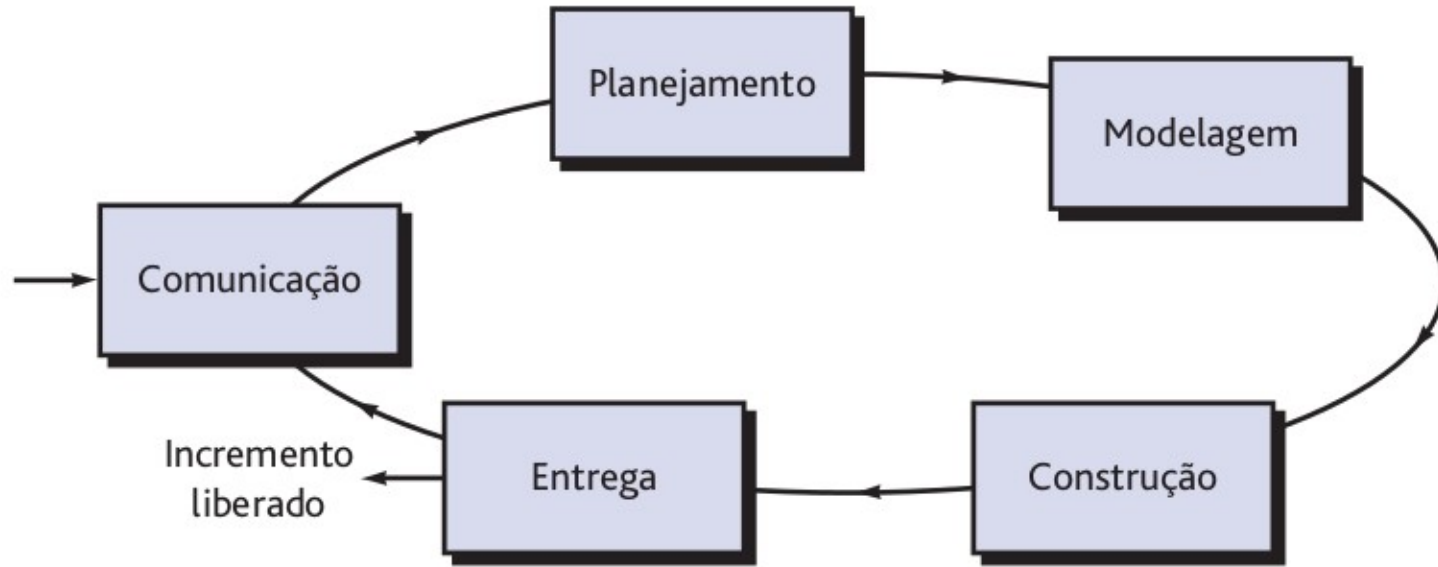
(a) Fluxo de processo linear

Modelo Iterativo Incremental



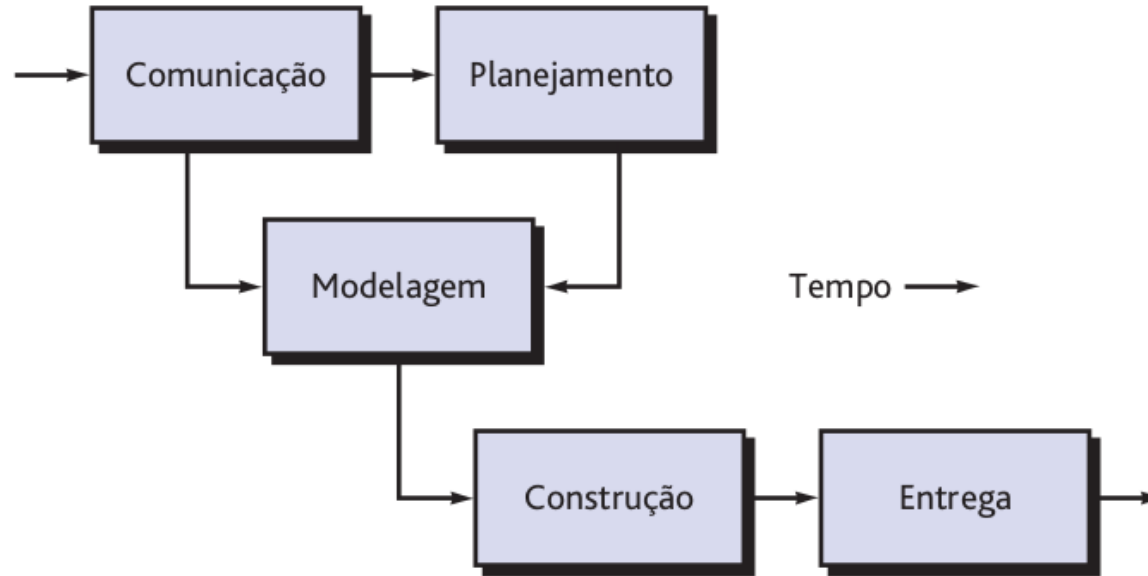
(b) Fluxo de processo iterativo

Modelo evolucionário



(c) Fluxo de processo evolucionário

Modelo paralelo



(d) Fluxo de processo paralelo

Atividades de apoio

As atividades metodológicas são complementadas por atividades de apoio:

- **Controle e acompanhamento do projeto**
 - possibilita que a equipe avalie o progresso em relação ao plano do projeto e tome medidas para cumprir o cronograma
- **Administração de riscos**
 - avalia riscos que possam afetar o resultado do projeto
- **Garantia da qualidade de software**
 - define e conduz as atividades que garantem a qualidade do software
- **Revisões técnicas**
 - Avaliam artefatos tentando identificar e eliminar erros antes que se propaguem para próxima atividade

Atividades de apoio

As atividades metodológicas são complementadas por atividades de apoio:

- **Medição**
 - define e coleta medidas. Auxilia na entrega de acordo com os requisitos.
- **Gerenciamento da configuração de software**
 - gerencia os efeitos das mudanças ao longo do processo.
- **Gerenciamento da capacidade de reutilização**
 - define critérios para a reutilização de artefatos (componentes de software) e estabelece mecanismos para obtenção de componentes reutilizáveis
- **Preparo e produção de artefatos de software**
 - atividades necessárias para criar artefatos como, modelos, documentos, logs, formulários e listas.

Avaliação e aperfeiçoamento de processos

Somente a existência de um processo de software não garante que o software será entregue dentro do prazo acordado ou que apresentará qualidade de longo prazo.

Com isso ao longo das últimas décadas foi proposta uma série de diferentes abordagens de avaliação e aperfeiçoamento dos processos de software

Avaliação e aperfeiçoamento de processos

- **SCAMPI (Standard CMMI Assessment Method for Process Improvement)** ¹ - fornece um modelo de avaliação do processo de cinco etapas, contendo cinco fases:
 - Início
 - Diagnóstico
 - Estabelecimento
 - Atuação
 - aprendizado

Avaliação e aperfeiçoamento de processos

- **CBA IPI (CMM-Based Appraisal for Internal Process Improvement) ¹**
 - fornece uma técnica de diagnóstico para avaliar a maturidade relativa de uma organização de software.
- **SPICE (ISO/IEC15504)**
 - Padrão que define um conjunto de requisitos para avaliação do processo de software. Sua finalidade é auxiliar as organizações no desenvolvimento de uma avaliação objetiva da eficácia de um processo qualquer de software [ISO08].

Avaliação e aperfeiçoamento de processos

- **ISO 9001:2000 para Software** – padrão genérico aplicável a qualquer organização que queira aperfeiçoar a qualidade global de produtos, sistemas ou serviços fornecidos. Padrão aplicável diretamente a organizações e empresas de software.

Resumo

Um modelo de processo genérico para engenharia de software consiste em um conjunto de atividades metodológicas e de apoio, ações e tarefas a realizar.

Cada modelo de processo, entre os vários existentes, pode ser descrito por um fluxo de processo diferente – uma descrição de como as atividades metodológicas, ações e tarefas são organizadas, sequencial e cronologicamente.

Referências

- PRESSMAN, Roger S. Engenharia de software: Uma Abordagem Profissional. 8.ed. Porto Alegre: AMGH, 2016. 968p.