

Quem se prepara, não para.

Análise de Sistemas

3º período

Professora: Michelle Hanne

O que é Software



"Um Software é desenvolvido ou passa por um processo de Engenharia; ele não é fabricado no sentido clássico" (Pressmann, 2016).



"Um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software." (Sommerville, 2010)

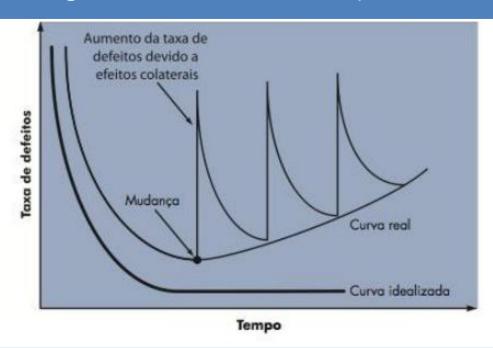


Existem muitos processos de software diferentes, mas todos devem incluir quatro atividades fundamentais para a engenharia de software (SOMMERVILLE, 2011):

- **1. Especificação de software:** A funcionalidade do software e as restrições a seu funcionamento devem ser definidas.
- **2. Projeto e implementação de software:** O software deve ser produzido para atender às especificações.
- **3. Validação de software:** O software deve ser validado para garantir que atenda às demandas do cliente.
- **4. Evolução de software:** O software deve evoluir para atender às necessidades de mudança dos clientes.



"Software não se desgasta, mas sim deteriora" (Pressmann, 2016).





Podem ser dirigidos a planos ou processos ágeis

- Processos dirigidos a planos são aqueles em que todas as atividades são planejadas com antecedência, e o progresso é avaliado por comparação com o planejamento inicial.
- Em processos ágeis, o planejamento é gradativo, e é mais fácil alterar o processo de maneira a refletir as necessidades de mudança dos clientes.



Cascata: organiza as atividades do processo de desenvolvimento de forma sequencial

Incremental: aplica sequencias lineares de forma escalonada, à medida de avanço do tempo

Evolucionário: modelo que absorvem as incertezas, pois trabalham com ciclos de desenvolvimento.

Espiral: O sistema é desenvolvido em ciclos, porém, nem todas as atividades de um ciclo necessitam ser realizadas

Prototipado: A prototipação pode ser utilizada como um modelo de processo isolado, pois é uma técnica a ser implementada no contexto de qualquer um dos modelos de ciclo de vida.

RAD (**Rapid Application Development**) - é um modelo de processo de software incremental que enfatiza um ciclo de desenvolvimento curto.

RUP (*Rational Unified Process*) - Sugere um fluxo de processo iterativo e incremental, proporcionando a sensação evolucionária que é essencial no desenvolvimento de software moderno.

Etapas de Processos de Software

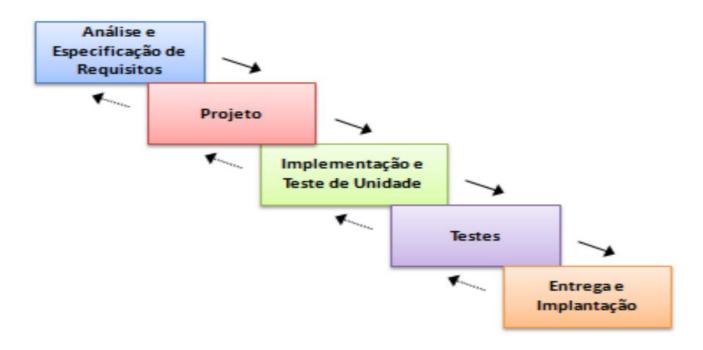


Atividades Genéricas do Processo de Software:

- Comunicação: é a atividade na qual são levantados os requisitos junto ao cliente.
- Planejamento: são elaborados o cronograma, as análises de riscos e os recursos necessários.
- Modelagem: traduz a especificação em formatos simbólicos, tanto em nível de requisito quanto em nível
- de projeto.
- Construção: são gerados os códigos e testes do software.
- Implantação: é quando se coloca o produto em produção, ou seja, é nesta atividade que o cliente recebe o *software*.

Modelo em Cascata

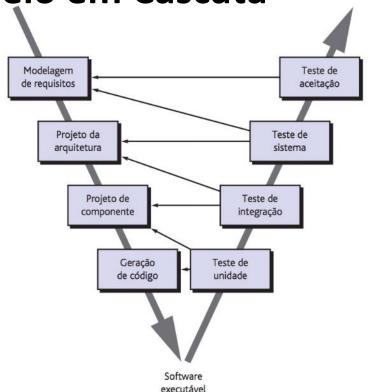




Modelo em Cascata. Fonte: Falbo, 2005.

Modelo em Cascata



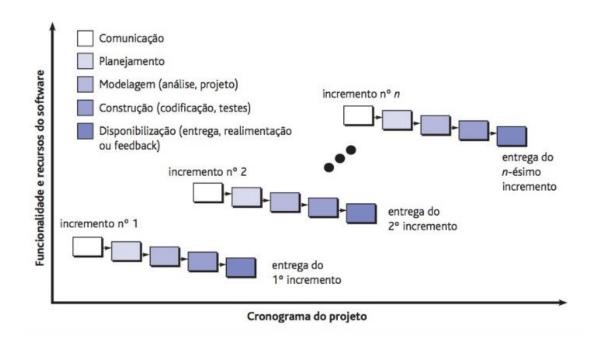


O modelo V descreve a relação entre ações de garantia da qualidade e ações associadas a comunicação, modelagem e atividades de construção iniciais.

Modelo em V. Fonte: Pressman, 2016.

Modelo Incremental

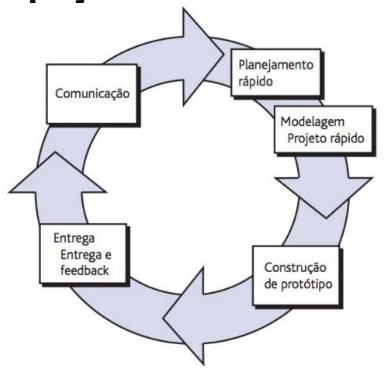




Modelo Incremental. Fonte: Pressman, 2016.

Prototipação



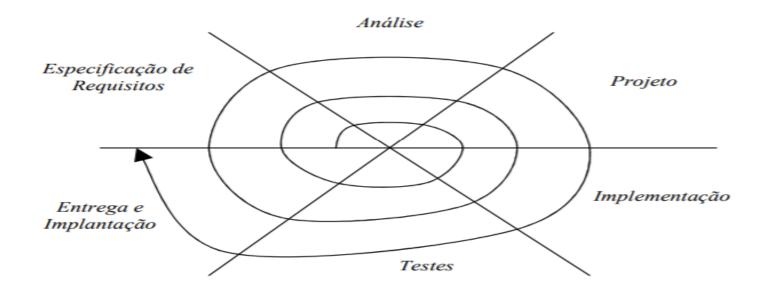


A Prototipação é usada no MVP (Minimum Viable Product)

Paradigma da Prototipação. Fonte: Pressman, 2016.

Modelo Espiral





Modelo em Cascata. Fonte: Falbo (2005).

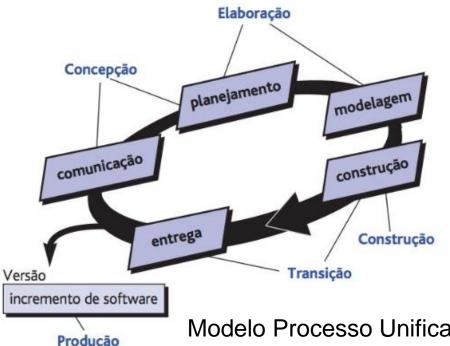
Processo Unificado (PU)



- Aproveitar os melhores recursos e características dos modelos tradicionais de processo de software, mas caracterizando-os de modo a implementar muitos dos melhores princípios do desenvolvimento ágil de software.
- Sugere um fluxo de processo iterativo e incremental, proporcionando a sensação evolucionária que é essencial no desenvolvimento de software moderno.

Processo Unificado Racional (RUP, Rational Unified Process)





Atualmente:

Processo
Unificado Ágil (Agile Unified
Process – AUP)

Modelo Processo Unificado. Fonte: Pressman, 2016.

Participantes do Processo de Software



Os participantes dos projetos são classificados em três categorias (PFLEEGER, 2004):

- Cliente: é quem paga para que seja construído o sistema.
- Usuário: é quem usa o sistema, não necessariamente é o cliente.
- **Desenvolvedor:** é o responsável por desenvolver e manter o sistema.

O que é Análise de Sistemas



Campo de estudo cujo objetivo é encontrar o melhor caminho para a geração de um processo de desenvolvimento de software.

O Analista de Sistemas é
um tradutor entre as
necessidades do usuário e o
programa a ser
desenvolvido pelo
programador.

Problema de Comunicação de Requisitos

Newton
Quem se prepara, não para.

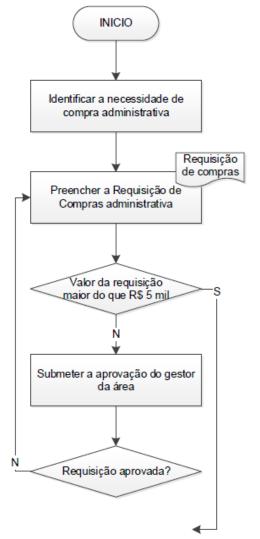
"Os projetos falham devido à omissão de informações importantes e/ ou devido a falha dos envolvidos, na coordenação de seus esforços para criar um produto final bem-sucedido". (Pressman, 2016)



Imagem: https://br.freepik.com/

Modelagem de Sistemas

Análise Estruturada:
 Modelagem Clássica

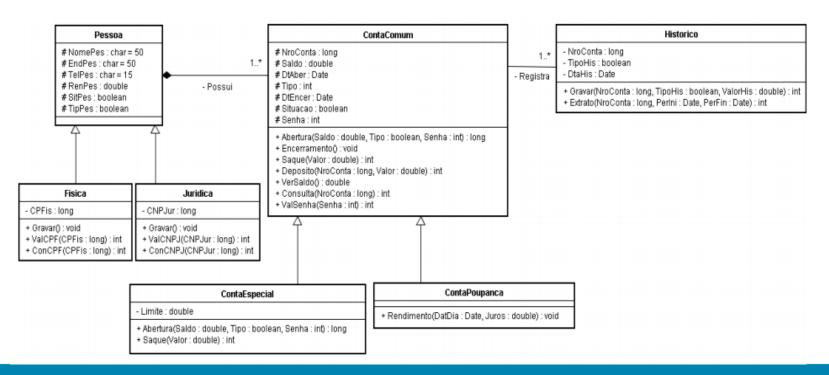




Modelagem de Sistemas

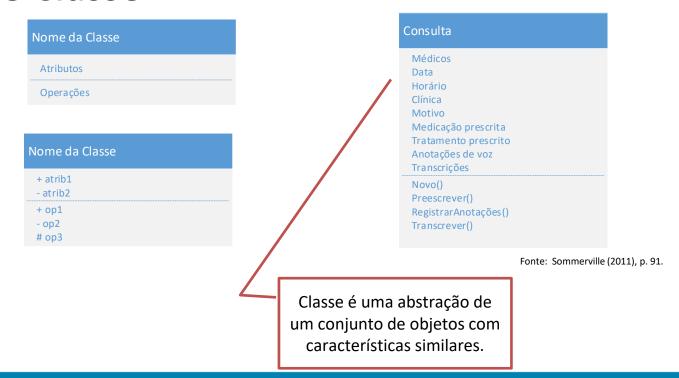


Análise Orientada a Objetos: Representar os objetos do mundo real











- Encapsulamento: a ideia do encapsulamento é de restringir o acesso aos comportamentos do objeto.
- Generalização (herança): permite que um componente herde características de outro componente.
- Polimorfismo: uma operação pode ser utilizada para realizar funções diferentes dentro do sistema.
- **Composição:** permite criar objetos a partir de outro, em outras palavras, um objeto ser composto por outros objetos.



São conexões entre classes:

1. Dependência - uma classe usa a outra.



2. **Generalização** - geral (superclasse) e uma coisa mais específica (subclasse)

Rectangle



3. Associação - classes ou objetos estão interconectados.



Ornamentos para Associações

Multiplicidade

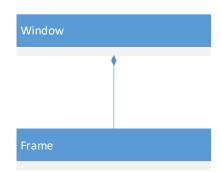




 Agregação: É uma relação do tipo "todo/parte" ou "possui um".



• **Composição:** Um tipo de agregação na qual as partes são inseparáveis do todo.

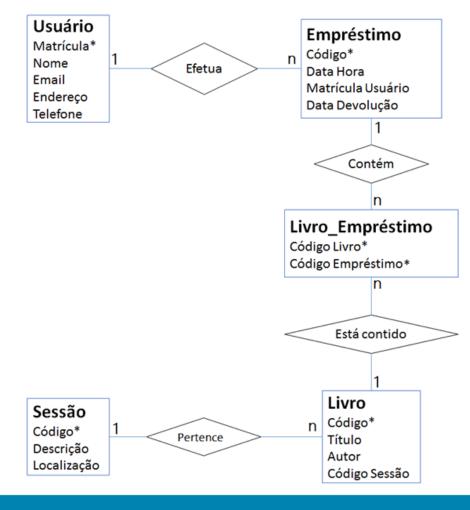


A importância da Modelagem de Sistemas

 Modelagem Conceitual: representa modelos de negócio

Exemplo DER sistema para bibliotecas **Fonte**:

https://www.devmedia.com.br/mod elo-entidade-relacionamento-mer-ediagrama-entidade-relacionamentoder/14332



Ferramentas Case



Permitem a criação de diagramas de forma simples em um ambiente de fácil utilização. Exemplos: Star UML, Astah, ERwin Data Modeler, Draw.io

Referências



M. Michell, Complexity: A guided tour, Oxford University Press, 2009.

PFLEEGER, S. L. **Engenharia de software**: teoria e prática. Belo Horizonte:

Prentice Hall, 2004.

company, 2010.

PRESSMAN, Roger S. MAXIM, Bruce R. **Engenharia de Software - Uma Abordagem Profissional**. 8.ed. Porto Alegre: Amgh Editora, 2016. 968p. ISBN 9788580555332. SOMMERVILLE, Ian. **Engenharia de software**. 8.ed. São Paulo: A. Wesley publishing