



Quem se prepara, não para.

Engenharia de Software

2º período

Professora: Michelle Hanne

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 do Ciclo de Vida

A escolha de um modelo de processo é fortemente dependente das características do projeto. Os principais modelos de ciclo de vida podem ser agrupados em três categorias principais:

- **modelos sequenciais**
- **modelos incrementais**
- **modelos evolutivos**

Modelo em Cascata

- O principal modelo da categoria sequencial é o modelo em cascata, a partir do qual outros modelos foram propostos.
- **O Modelo em Cascata (waterfall) também chamado de “modelo de ciclo de vida clássico”, organiza as atividades do processo de desenvolvimento de forma sequencial.**

Modelo em Cascata

Comunicação

- Levantamento de necessidades iniciais

Planejamento

- Estimativa, Cronograma e Acompanhamento

Modelagem

- Análise e Projeto

Construção

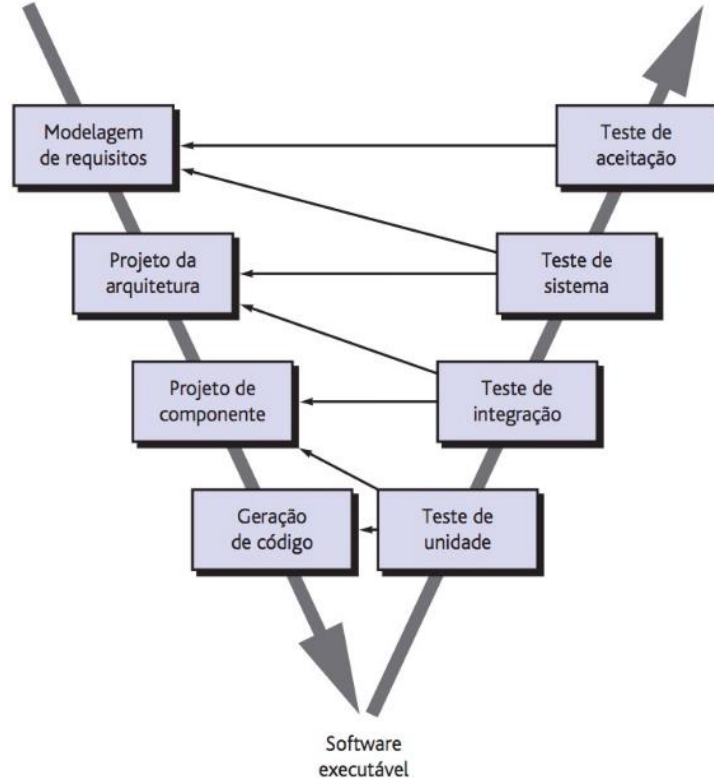
- Codificação e Testes

Entrega e Implantação

- Entrega, Suporte e Feedback

Modelo em Cascata. Baseado em Pressman, 2016.

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.

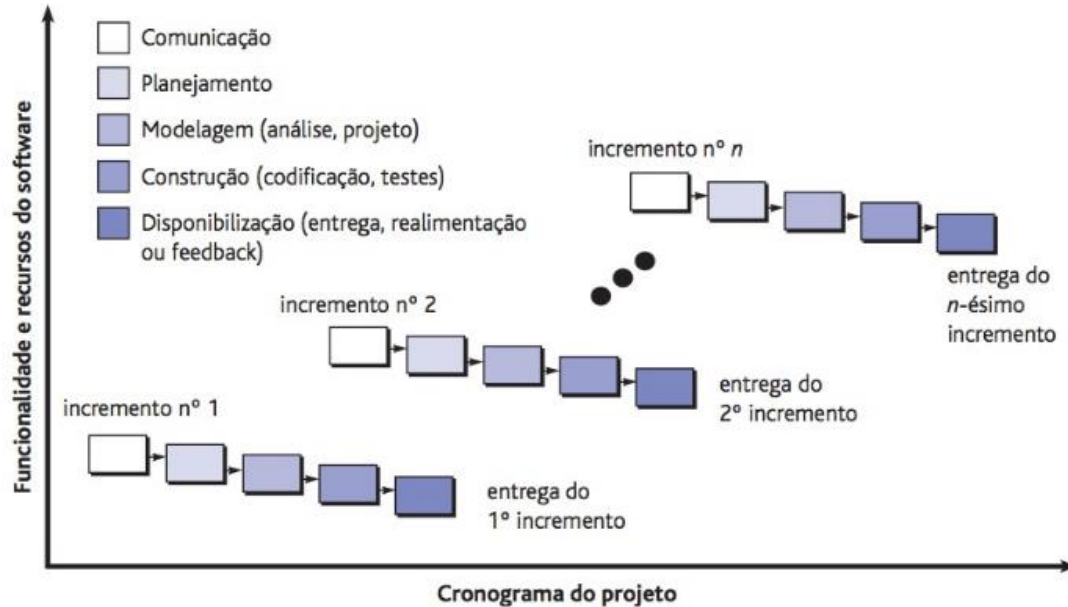
Modelo em V. Fonte: Pressman, 2016.

Modelo Incremental

- O modelo incremental combina os fluxos de processo linear e paralelo. O modelo incremental aplica sequencias lineares de forma escalonada, à medida que o tempo vai avançando. **Produzindo “incrementos” entregáveis do software.**

Fonte: Pressman, 2016

Modelo Incremental



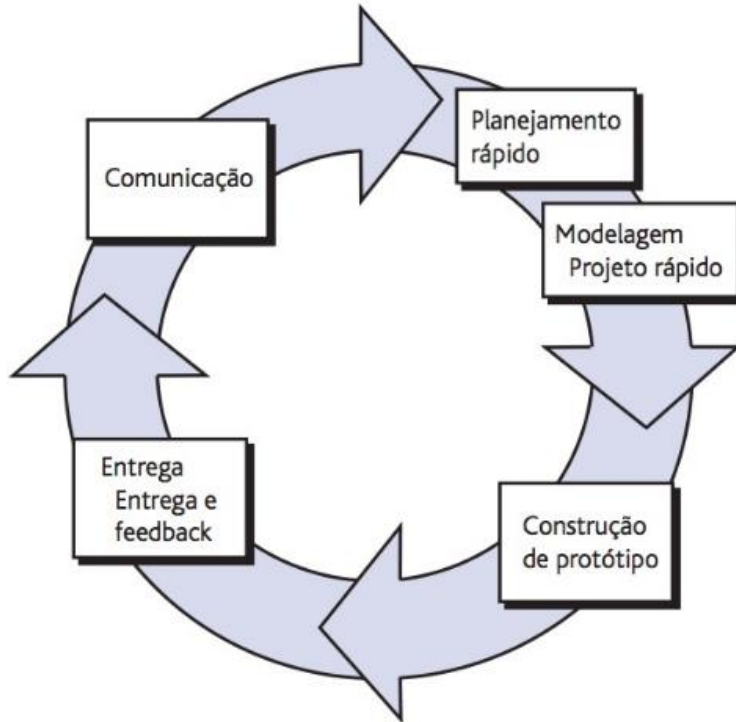
Modelo Incremental. Fonte: Pressman, 2016.

Prototipação

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

Fonte: Pressman, 2016.

Prototipação



A Prototipação é usada no MVP (Minimum Viable Product)

Paradigma da Prototipação. Fonte: Pressman, 2016.

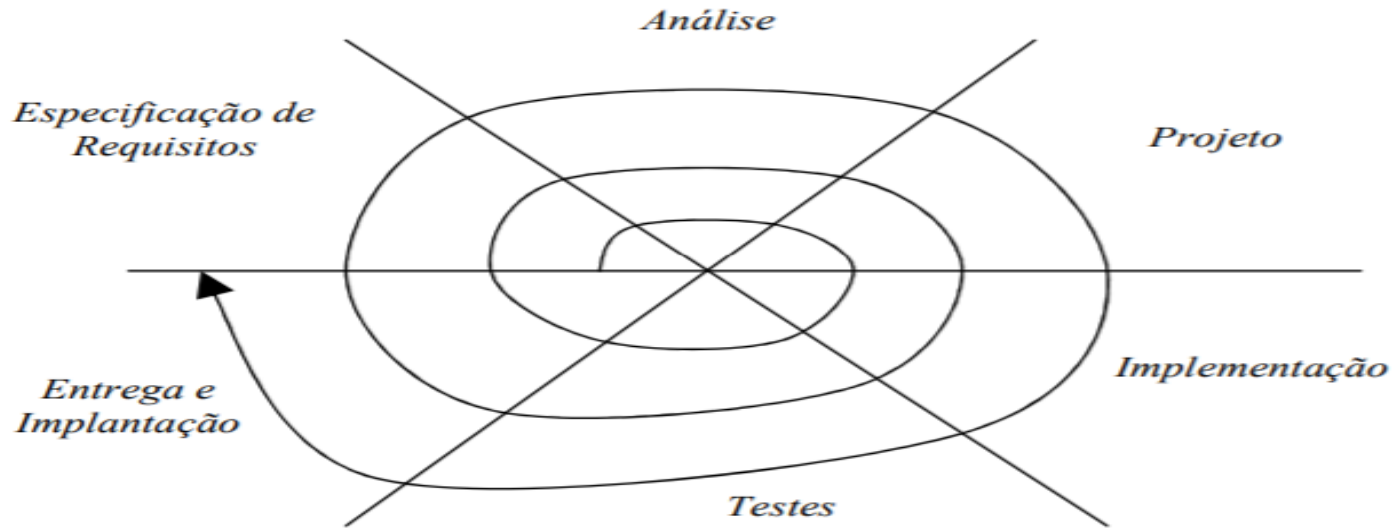
Modelos Evolucionários ou Evolutivos

Sistemas de software, como quaisquer sistemas complexos, evoluem ao longo do tempo.

Os requisitos, muitas vezes, são difíceis de serem estabelecidos ou mudam com frequência ao longo do desenvolvimento.

Os modelos evolucionários ou evolutivos lidam com incertezas e absorvem melhor as contínuas mudanças dos sistemas.

Modelo Espiral



Modelo em Cascata. Fonte: Falbo (2005).

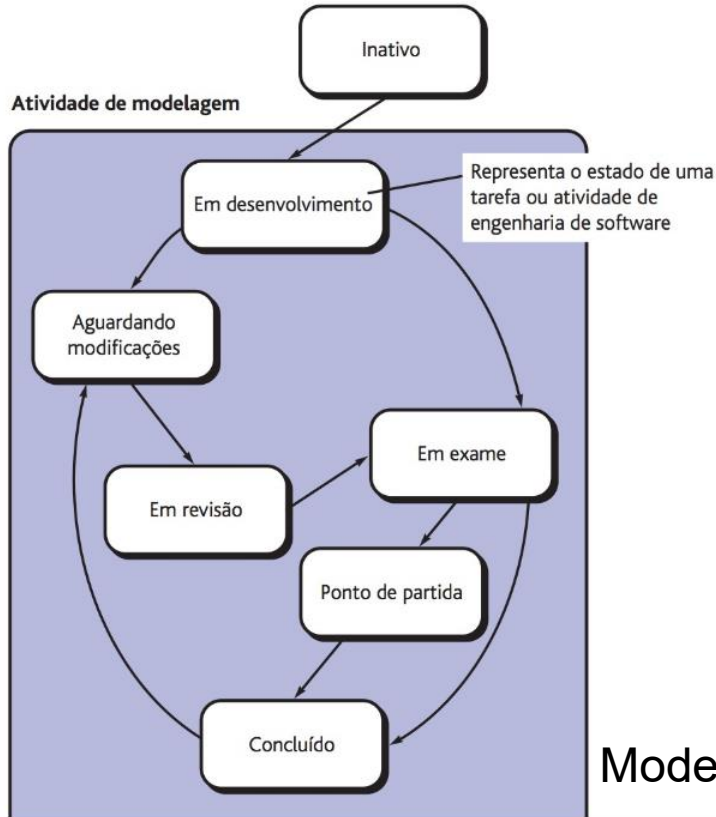
Modelo Espiral

- O sistema é desenvolvido em ciclos, sendo que nos primeiros ciclos nem sempre todas as atividades são realizadas.
- Por exemplo, o produto resultante do primeiro ciclo pode ser uma especificação do produto ou um estudo de viabilidade.
- As etapas subsequentes ao longo da espiral podem ser usadas para desenvolver protótipos, chegando progressivamente a versões operacionais do software, até se obter o produto completo

Modelo de Processo Especializado

- Incluem características de um ou mais dos modelos tradicionais, usado quando se opta por uma abordagem de engenharia de software especializada ou definida de forma restrita:
 - Desenvolvimento baseado em Componentes
 - O modelo de métodos formais
 - Desenvolvimento de software orientado a aspectos

Modelo de Processo Concorrente



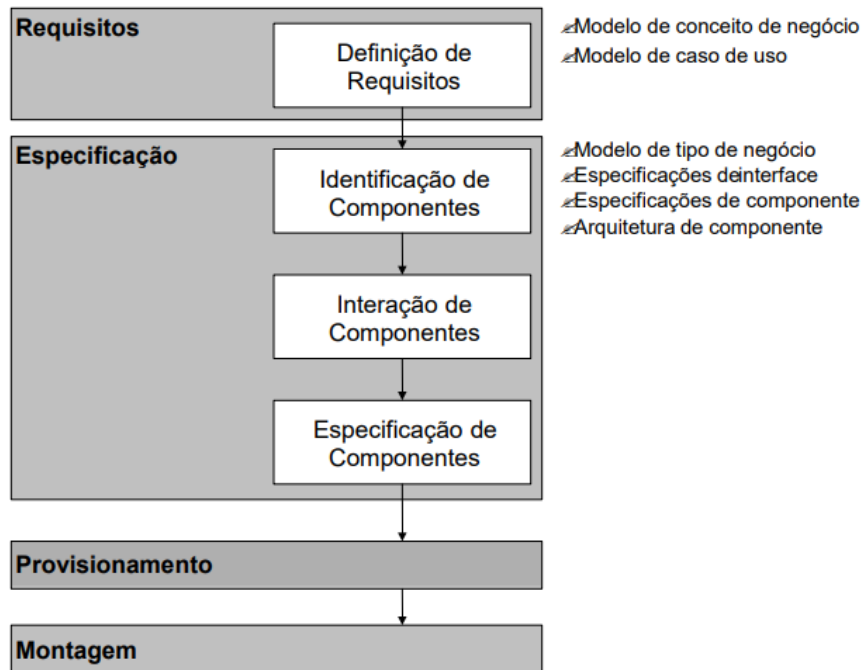
Define uma rede de processos. Cada atividade, ação ou tarefa na rede existe simultaneamente. Eventos gerados disparam transições entre os estados associados a cada atividade.

Modelo de Processo Concorrente. Fonte: Pressman, 2016.

Modelo Baseado em Componentes

- O modelo de desenvolvimento baseado em componentes incorpora muitas das características do modelo espiral.
- Permitem que o componente seja integrado ao software a ser desenvolvido.
- Componentes de software comercial de prateleira

Modelo Baseado em Componentes



Modelo Baseado em Componentes. Fonte:

https://edisciplinas.usp.br/pluginfile.php/130351/mod_resource/content/1/DSBC_UML_Components_1-2-3.pdf

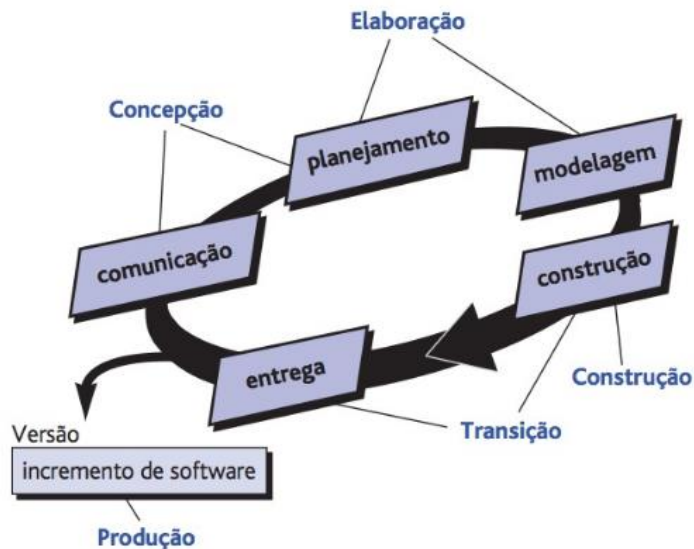
Desenvolvimento de Software Orientado a Aspecto

- **AOSD** (*aspect-oriented software development*), Desenvolvimento de Software Orientado a Aspecto que representam preocupações do cliente que transcendem várias funções, recursos e informações do sistema.
- Usam características tanto dos modelos de processo evolucionário quanto de processo concorrente.

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*)



Modelo Processo Unificado. Fonte: Pressman, 2016.

Processo Unificado Racional (*RUP*, *Rational Unified Process*)

- A fase de concepção do PU inclui a atividade de comunicação com o cliente e a de planejamento.
- Identificar as necessidades de negócio para o software, propõe-se uma arquitetura rudimentar para o sistema e desenvolve-se um **planejamento para a natureza iterativa e incremental do projeto decorrente.**

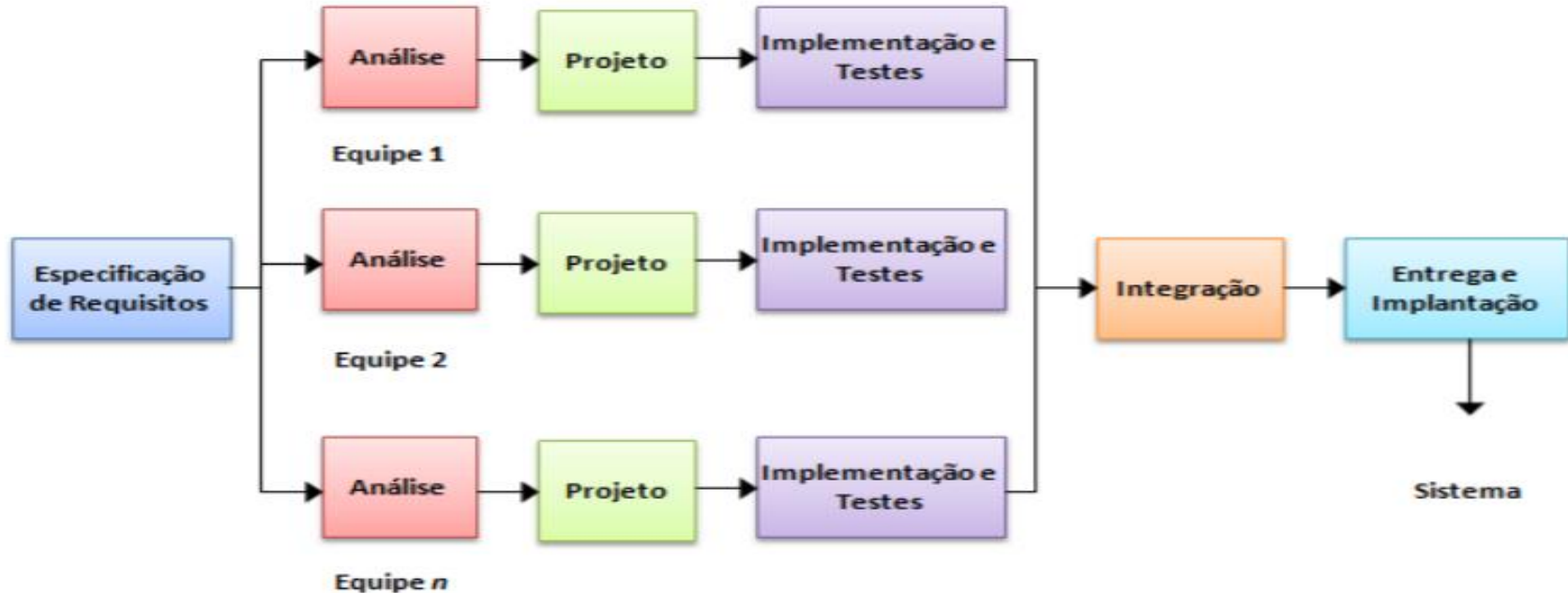
Processo de Software

- **O Processo de Software Pessoal (*PSP, Personal Software Process*)** enfatiza a medição pessoal, tanto do artefato de software gerado quanto da qualidade resultante dele. Além disso, responsabiliza o profissional pelo planejamento de projetos e lhe permite controlar a qualidade de todos os artefatos de software desenvolvidos.
- **Processo de Software de Equipe (*TSP, Team Software Process*)**. O objetivo do TSP é criar uma equipe de projetos “autodirigida”.

Modelo RAD (*Rapid Application Development*)

- O **Rapid Application Development (RAD)** é um modelo de processo de software incremental que enfatiza um ciclo de desenvolvimento curto.
- O Modelo RAD é uma adaptação, de alta velocidade, do modelo em cascata, no qual a agilidade é conseguida com o uso de uma abordagem de construção baseada em componentes.

Modelo RAD (*Rapid Application Development*)



Modelo RAD. Fonte: Falbo, 2005.

Modelos de Processo de Software - Resumo

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.

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.

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 company, 2010.