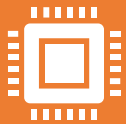


O que é um Software ?

“Consiste em instruções, estrutura de dados e documentação.”

Engenharia de Software



Em meados dos anos 70, surgiram os primeiros elementos da engenharia de software, devido ao evento chamado “crise do software”.



Os computadores evoluíam cada vez mais rápido com a introdução dos chips e os softwares não estavam acompanhando no mesmo ritmo essa evolução.



Pode ser vista como uma abordagem de desenvolvimento de software elaborada com disciplina e métodos bem definidos.

Engenheira de software

- [Sommerville 2011] relata os maiores problemas enfrentados na época:
 - I. Estouro de orçamento;
 - II. Prazos não cumpridos;
 - III. Software que não atendem aos requisitos do usuário;
 - IV. Projetos com poucos elementos para permitir sua gestão e código fonte de difícil manutenção.

O que é PROCESSO DE SOFTWARE ?

“Definimos processo de software como uma metodologia para as atividades, ações e tarefas necessárias para desenvolver um software de alta qualidade.” (PRESSMAN, 2016, p. 30)

“Um processo de software é um conjunto de atividades relacionadas que levam à produção de um produto de software.” (SOMMERVILLE, 2011, pg. 18)

“O processo de engenharia de software não é rígido nem deve ser seguido à risca. isso, ele deve ser ágil e adaptável (ao problema, ao projeto, à equipe e à cultura organizacional).” (PRESSMAN, 2016, p. 18)

Processos de Software

São complexos e dependem de vários fatores como:

- Produto a ser desenvolvido;
- Time;
- Recursos;
- Não existe um processo ideal, ele pode ser adaptado de acordo com o contexto, podendo ser mais formal ou mais flexível

Atividades(PRESSMAN)

Atividade	Definição
Comunicação	Entender os objetivos dos envolvidos para o projeto e reunir requisitos que ajudem a definir os recursos e as funções do software.
Planejamento	Cria um “mapa” (plano de projeto de software) descrevendo as tarefas técnicas a serem conduzidas, os riscos prováveis, os recursos necessários , os produtos resultantes a ser produzidos e um cronograma de trabalho .
Modelagem	Criação de modelos para entender melhor as necessidades do software e do projeto que vai atender a essas necessidades. Cria-se um “esboço” para que se possa ter uma ideia do todo.
Construção	Essa atividade combina geração de código (manual ou automatizada) e testes necessários para revelar erros na codificação.
Entrega	Entrega do software e avaliação pelo cliente

Atividades de Apoio (PRESSMAN)

Atividade	Definição
Controle e acompanhamento do projeto	Possibilita que a equipe avalie o progresso em relação ao plano do projeto e tome as medidas necessárias para cumprir o cronograma .
Administração de riscos	Avalia riscos que possam afetar o resultado ou a qualidade do produto/projeto .
Garantia da qualidade	Define e conduz as atividades que garantem a qualidade do software .
Revisões técnicas	Avaliam artefatos da engenharia de software, tentando identificar e eliminar erros antes que se propaguem para a atividade seguinte .
Medição	Define e coleta medidas (do processo, do projeto e do produto).
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 e estabelece mecanismos para a obtenção de componentes reutilizáveis .
Preparo e produção de artefatos de software	Engloba as atividades necessárias para criar artefatos como, por exemplo, modelos, documentos, logs, formulários e listas.

Atividades Fundamentais (SOMMERVILLE)

Atividade	Definição
Especificação de Software	A funcionalidade do software e as restrições a seu funcionamento devem ser definidas.
Projeto e Implementação de Software	O software deve ser produzido para atender às especificações.
Validação de Software	O software deve ser validado para garantir que atenda às demandas do cliente.
Evolução de Software	O software deve evoluir para atender às necessidades de mudança dos clientes .

FLUXO DE PROCESSO

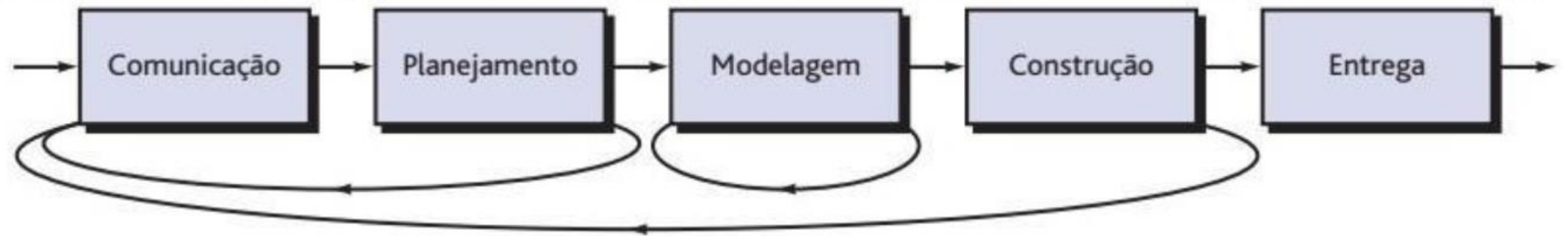
“Descreve como são organizadas as atividades metodológicas, bem como as ações e tarefas que ocorrem dentro de cada atividade em relação à sequência e ao tempo” (PRESSMAN, 2016, p. 31)

- O fluxo de um processo pode ser: Linear, iterativo, evolucionário ou paralelo.
- Não confundir com Modelos de Processos (Fluxo + Padrões + Organização do Trabalho).



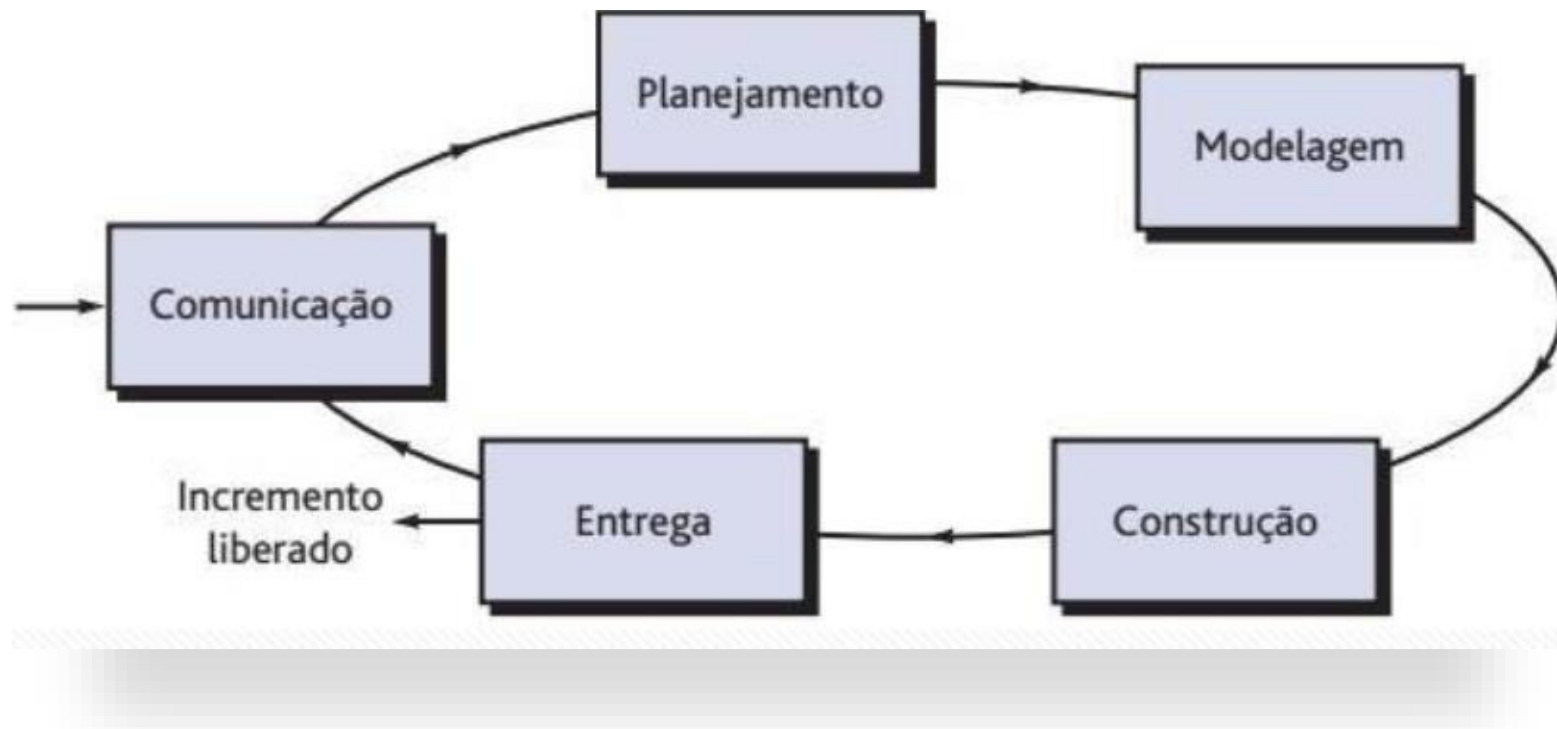
Fluxo de Processo LINEAR

Executa cada uma das cinco atividades metodológicas em **sequência**, começando com a comunicação e culminando com a entrega. Cada uma das atividades é executada **somente uma vez**.



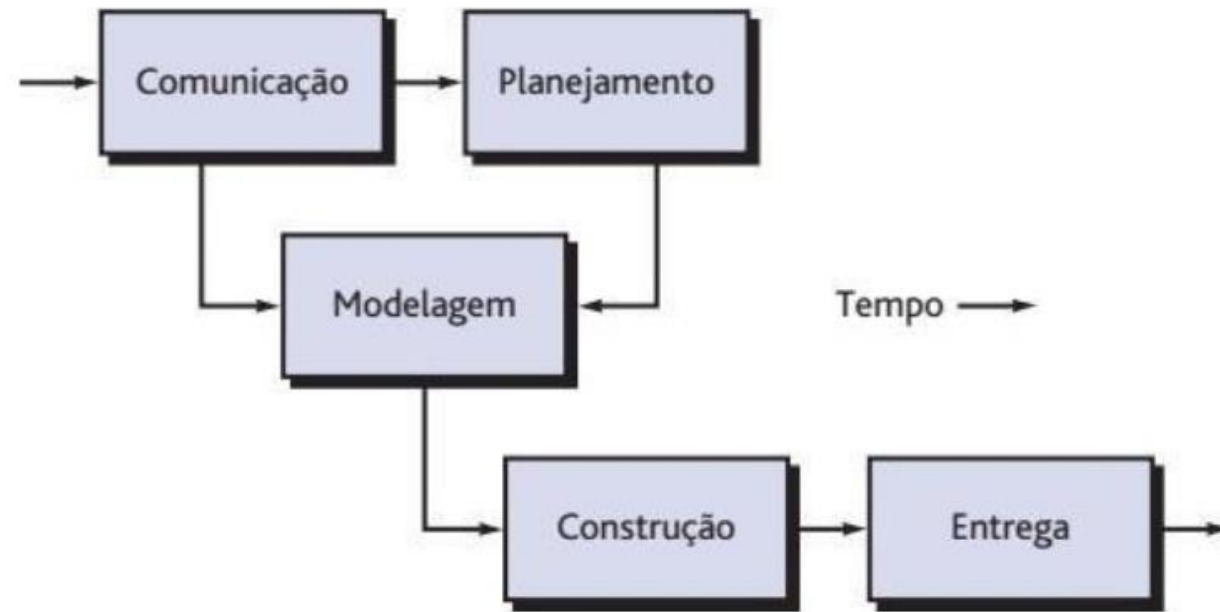
Fluxo de Processo ITERATIVO

- Repete uma ou mais das atividades metodológicas antes de prosseguir para a seguinte
- A repetição pode ocorrer em uma única atividade ou em conjuntos de atividades.



Fluxo de Processo EVOLUCIONÁRIO

- Executa as atividades de forma “**circular**”.
- **Cada volta** pelas cinco atividades conduz a uma **versão mais completa do software**.



Fluxo de Processo PARALELO

- Executa **uma ou mais atividades em paralelo** com outras (por exemplo, a modelagem para um aspecto do software poderia ser executada em paralelo com a construção de outro aspecto do software).

O que é Modelo de Processo de Software?

Um modelo de processo fornece um **guia específico** para o trabalho de engenharia de software. Ele define o fluxo de **todas as atividades, ações e tarefas**, o grau de **iteração**, os **artefatos** e a **organização do trabalho** a ser feito.

(PRESSMAN, 2016, p. 40)

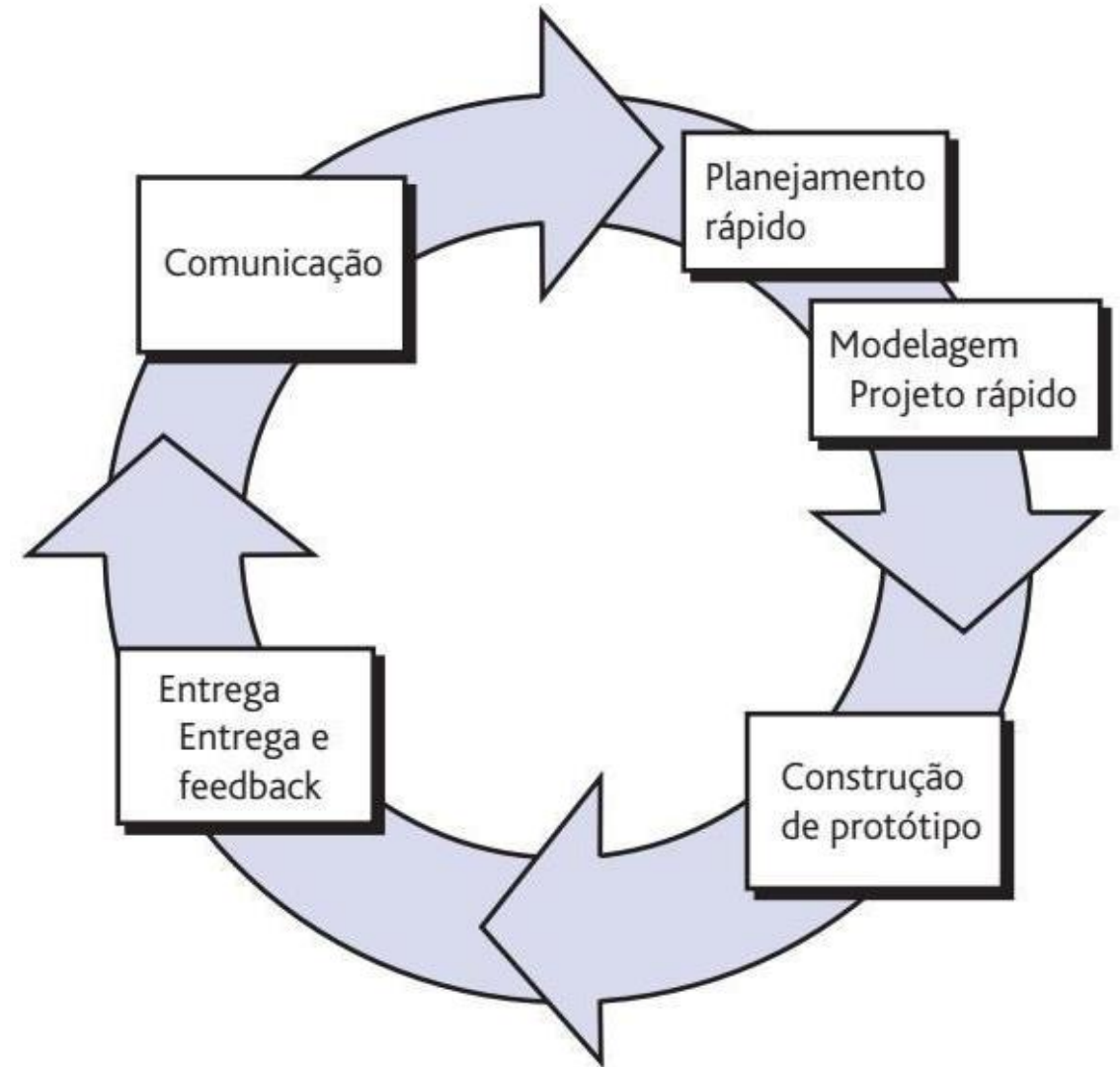
- Modelos de processo **prescritivos** (tradicionais ou dirigidos a planos)
- Os modelos de processo não prescritivos são denominados “**ágeis**”.
- Concentram-se em **estruturar e ordenar** o desenvolvimento do software.

Modelos de Processo Evolucionário

- É utilizado em situações onde há pressão pela **implantação rápida** do software ou os requisitos essenciais do produto estão bem compreendidos, mas os **detalhes ainda precisam ser definidos**.
- Modelos evolucionários são **iterativos**.
- Apresentam características que possibilitam desenvolver **versões cada vez mais completas** do software.
- Os dois modelos mais comuns de processos evolucionários são: **Prototipação** e **Modelo Espiral**.

Prototipação

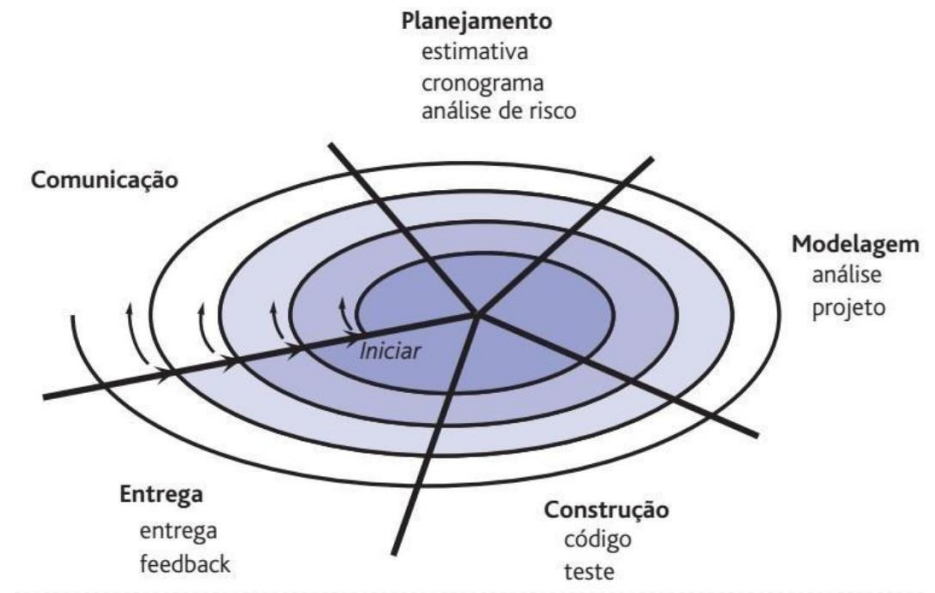
- Um protótipo é uma **versão inicial** de um sistema de software, usado para **demonstrar conceitos, experimentar opções de projeto e descobrir mais sobre o problema** e suas possíveis soluções.
- O desenvolvimento **rápido e iterativo** com **custos controlados**.
- Prototipação possa ser utilizada como um **modelo de processo isolado**, mas ela é mais comumente utilizada como uma **técnica** a ser implementada no contexto de qualquer um **dos modelos de processo**.



Prototipação

Vantagens	Desvantagens
Facilita a definição dos requisitos.	Os envolvidos enxergam o que parece ser uma versão operacional do software.
Reduz os riscos e incertezas do desenvolvimento.	Pressão para aproveitar o código escrito.
A experiência obtida pode reduzir o custo das etapas seguintes.	Retrabalho
	Pode aumentar o custo do desenvolvimento.

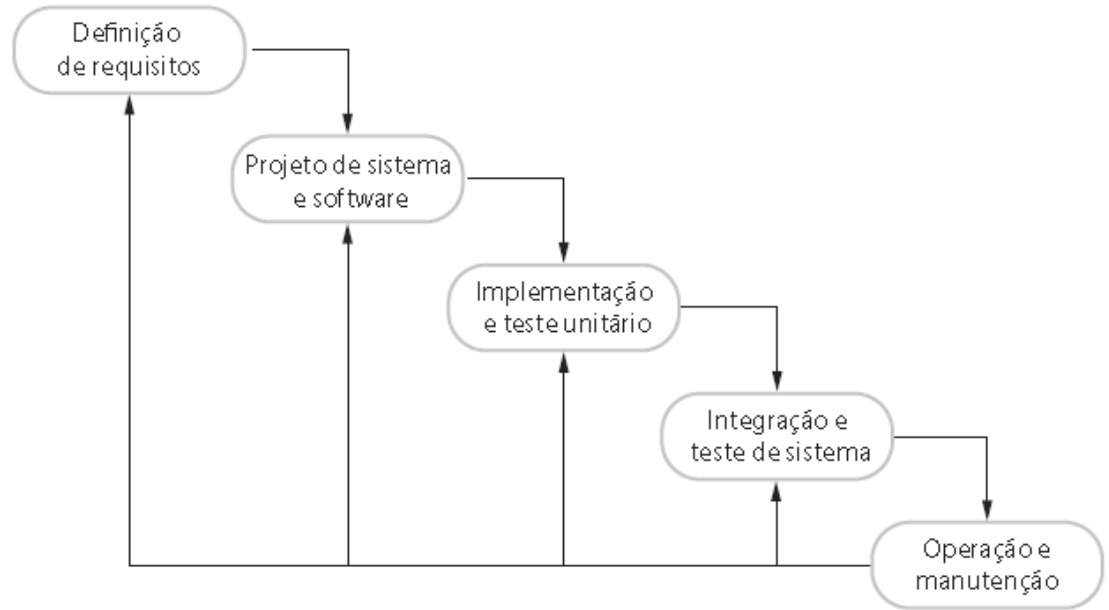
Modelo Espiral



- É um modelo de processo de software evolucionário que une a natureza iterativa da prototipação aos aspectos sistemáticos e controlados do modelo cascata. A principal diferença entre o modelo espiral e outros modelos de processo de software é seu reconhecimento explícito do risco.
- As versões iniciais podem ser um modelo de papel ou protótipo. As últimas são cada vez mais completas do sistema.
- Diferentemente de outros modelos de processo, que terminam quando o software é entregue, o modelo espiral pode ser adaptado para ser aplicado ao longo da vida do software.
- O modelo é uma abordagem realista para o desenvolvimento de softwares de grande porte.
- No caso de um orçamento fixo, o modelo espiral pode ser um problema.

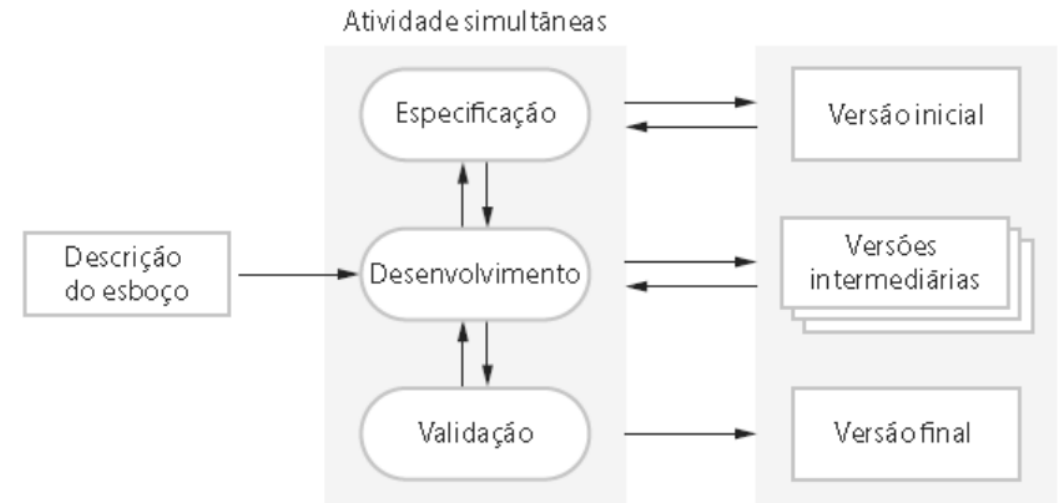
Modelo de Processos (Sommerville)

O modelo em cascata



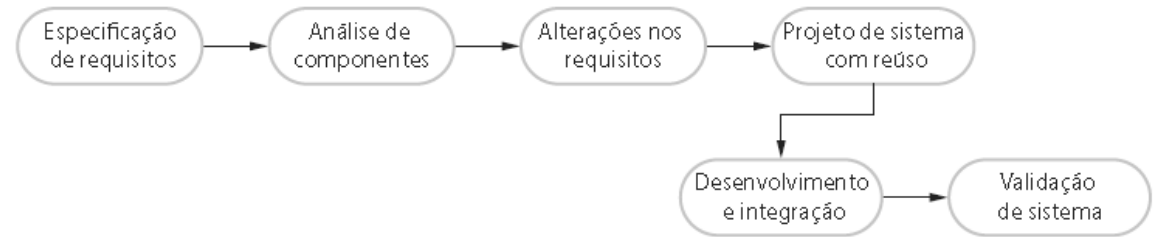
Modelo de Processos (Sommerville)

Desenvolvimento incremental



Modelo de Processos (Sommerville)

Engenharia de software orientada a reuso





Desenvolvimento Baseado em Componentes

- Na maioria dos projetos de software, há algum **reúso de software**. Isso acontece muitas vezes informalmente, quando as pessoas envolvidas no projeto sabem de projetos ou códigos semelhantes ao que é exigido.

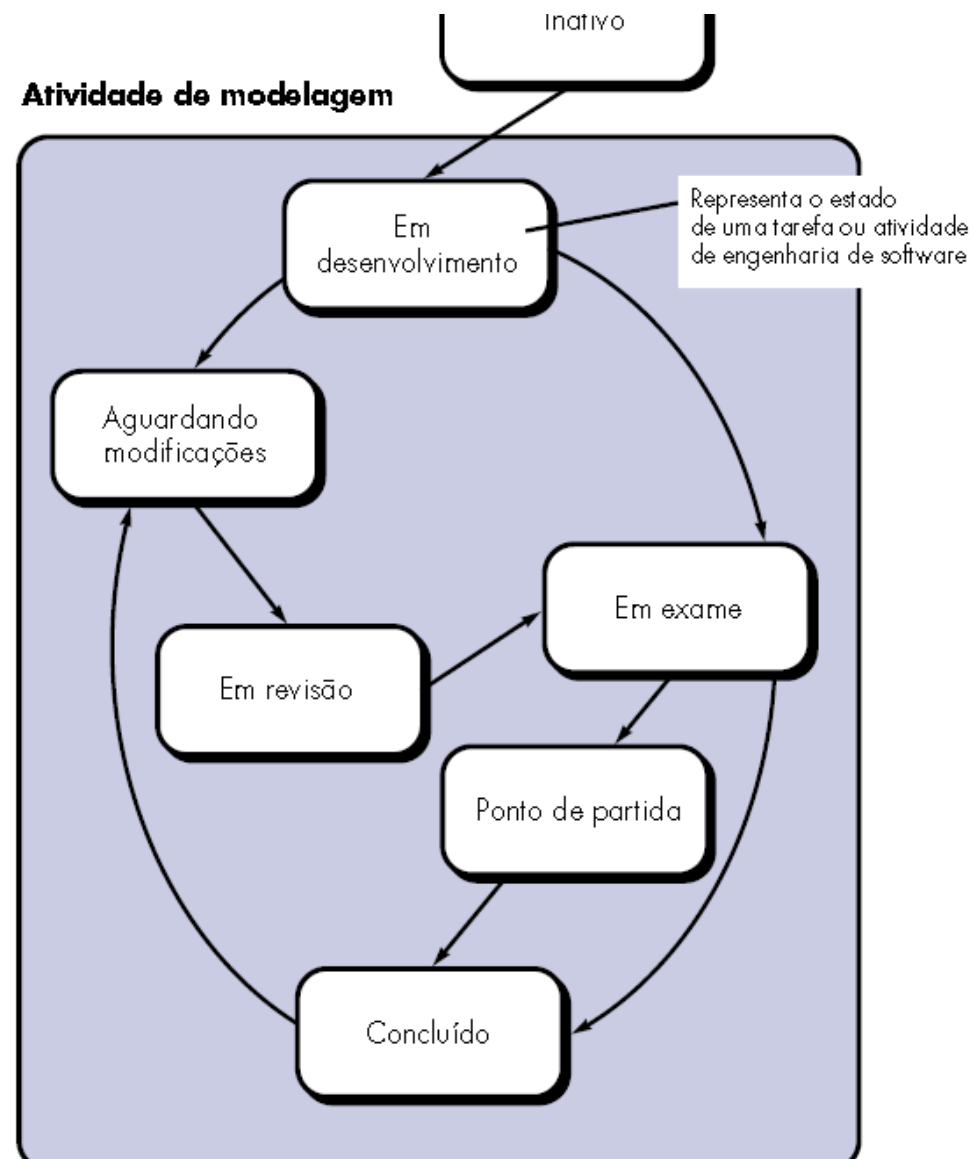
- Abordagens orientadas a reúso dependem de uma ampla **base de componentes reusáveis** de software e de um **framework de integração** para a composição desses componentes.

- Tem a vantagem óbvia de reduzir a quantidade de software a ser desenvolvido, no entanto, pode comprometer atendimento aos requisitos do cliente.

Modelos Evolucionários

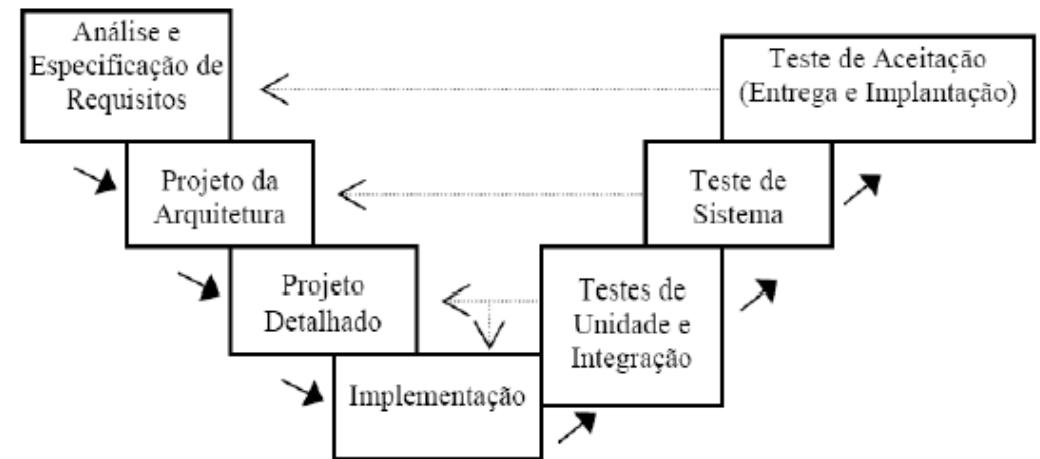
- Caracterizados por contínuas modificações, prazos muito apertados e por uma ênfase na satisfação do cliente—usuário.
- Em muitos casos, o tempo de colocação de um produto no mercado é o requisito mais importante a ser gerenciado.
- **Prototipação e Espiral** são os mais conhecidos.
- Outros menos conhecidos como o desenvolvimento concorrente define uma série de eventos que irão disparar transições de estado para estado para cada uma das atividades, ações ou tarefas da engenharia de software.

Modelo de Processo Concorrente



Modelo em V

- **Observação:** O modelo em V é uma variação do modelo cascata que procura enfatizar a estreita relação entre as atividades de teste e as demais fases do processo. Os modelos sequenciais pressupõem que o sistema é entregue completo, após a realização de todas as atividades de desenvolvimento. Hoje em dia os clientes não estão dispostos esperar a entrega no final, ainda mais quando se trata de um sistema grande, por isso a entrega por partes, se tornou mais comum e disso surgiram outros processos.



Processo Unificado RUP

Fases no Rational Unified Process

