

ACH2028

Qualidade de Software

Aula 02 - Revisão de Engenharia de Software

Prof. Marcelo Medeiros Eler
marceloeler@usp.br

Objetivos da aula

- Apresentar a história da Engenharia de Software e seus objetivos
- Apresentar uma visão geral de um processo de desenvolvimento de software
- Apresentar uma visão geral de dois tipos de processos de desenvolvimento de software
 - Processos Tradicionais (Preditivos)
 - Métodos Ágeis (Adaptativos)

Conceitos

- O que é um software?

Conceitos

- O que é um software? [Michaelis]
 - “Qualquer programa ou grupo de programas que instrui o hardware sobre a maneira como ele deve executar uma tarefa, inclusive sistemas operacionais, processadores de texto e programas de aplicação.”

Conceitos

- O que é um software?
- O que é engenharia?

Conceitos

- O que é um software?
- O que é engenharia? [Michaelis]
 - “Arte de aplicar os conhecimentos científicos à invenção, aperfeiçoamento ou utilização da técnica industrial em todas as suas determinações.”

História

1950 a 1965:

- O hardware sofreu contínua mudança
- O hardware era de propósito geral
- O software era de propósito específico
- Havia poucos profissionais e métodos sistemáticos para o desenvolvimento de software
- A documentação era praticamente inexistente
- A programação predominante era de baixo nível

História

1965 a 1970:

- Multiprogramação e sistemas com múltiplos usuários
- Evolução das linguagens e paradigmas de programação
- Primeira geração dos SGBDs
- Aumento do número de profissionais de software
- Bibliotecas de software
- Aumento do uso de computadores e sistemas de informação
- Aumento da complexidade do software
- Crise do Software

Crise do Software

Refere-se a uma série de problemas encontrados no desenvolvimento de software:

- Estimativas (de esforço, prazo e custo) imprecisas
- Insatisfação do cliente
- Software de baixa qualidade
- Difícil manutenção
- Gerência de projetos ineficiente
- Processo de software ineficiente

Um pouco de história

1968:

- A “NATO Science Committee” organizou uma conferência para discutir os problemas da crise do software e propor soluções
- Foi “criada” a Engenharia de Software como uma tentativa de dar um tratamento de engenharia (mais sistemático, controlado e de qualidade mensurável) ao desenvolvimento de sistemas de software complexos.

Um pouco de história

1968:

- Software Engineering: Report of a conference sponsored by the NATO Science Committee, Garmisch, Germany, 7-11 Oct. 1968, Brussels, Scientific Affairs Division, NATO January 1969. <http://homepages.cs.ncl.ac.uk/brian.randell/NATO/index.html>

Software Engineering

COINS III

VOLUME I

Proceedings of the Third Symposium on Computer and Information Sciences held in Miami Beach, Florida, December, 1969

Edited by

JULIUS T. TOU

Center for Informatics Research
University of Florida
Gainesville, Florida



Academic Press New York · London · 1970

Software Engineering—A New Profession

Julius T. Tou

UNIVERSITY OF FLORIDA
GAINESVILLE, FLORIDA

Throughout history engineering has continually interacted with the rest of human society in a highly determinative and influential way. It has largely freed man from physical drudgery and from the slavery of routine mental tasks. It has provided the tools of new scientific discovery as well as those of new creative art. It has provided the means of bringing to mankind the results of the creative efforts of the great artists, writers, scientists, and philosophers of both ancient and modern times. It has provided the facilities for man to communicate with his fellowmen throughout the world. It has placed men on the moon, and is now concerned with the problem of information explosion. The engineer continually seeks new imaginative solutions to many of the very real problems that confront mankind.

During the last decade, engineering has undergone a continuous transition from the practical to the theoretical. This change had made technology overlap with science. Engineering education today is quite different from engineering education twenty years ago; it bears little resemblance to engineering education at the turn of this century. Perhaps we shall not be able to recognize engineering by the year 2000. Old branches are either updated or phased out, and new branches are created out of necessity. The newest offspring is software engineering which is the theme of this volume. For decades engineering drawing was a subject required by all engineering colleges. Today, it is being replaced in many schools by digital computation and programming as a basic requirement.

Engenharia de Software

- Definição (IEEE):
 - “A aplicação de uma abordagem sistemática, disciplinada e possível de ser medida para o desenvolvimento, operação e manutenção do software”
- Objetivo inicial (tradicional):
 - Construir um software como qualquer outro produto da engenharia tradicional (civil, mecânica, aeronáutica, etc)

Engenharia de Software

Um dos benefícios da criação da Engenharia de Software foi a definição de processos de desenvolvimento de software

Neste contexto, dois tipos de processo se destacam:

- Processos tradicionais (preditivos)
- Métodos ágeis (adaptativos)

Processo de desenvolvimento de software

Antes de diferenciar os dois tipos de abordagem, é importante compreender, mesmo que de forma geral, as atividades de um processo de desenvolvimento de software.

Processo de desenvolvimento de software

Refere-se à abordagem/método/metodologia/filosofia de desenvolvimento de um software

Compreende o conjunto de atividades utilizado para desenvolver um software

Alguns objetivos de um processo de desenvolvimento são:

- Definir quais atividades devem ser executadas ao longo do projeto
- Definir quando, como e por quem tais atividades serão executadas
- Definir quais artefatos serão produzidos e usados em cada atividade
- Prover pontos de controle para verificar o andamento do desenvolvimento
- Padronizar a forma de desenvolver software em uma organização

Processo de desenvolvimento de software

Atividades técnicas

- Análise/Engenharia de Sistemas
- Análise/Engenharia de Requisitos (Especificação)
- Projeto/Arquitetura
- Codificação (Implementação)
- Teste (Validação)
- Manutenção (Evolução)

Atividades gerenciais

- Gerência de Mudanças
- Gerência de Configuração de Software
- Garantia de Qualidade de Software
- Gerência de Projetos

Processo de desenvolvimento de software

Análise/Engenharia de Sistemas

Análise/Engenharia de Requisitos

Projeto/Arquitetura

Codificação

Teste

Manutenção

Análise e Engenharia de Sistemas

- Componentes de SI
- Interfaces com outros sistemas
- Viabilidade econômica, legal e técnica
- Soluções em alto nível

Análise e Engenharia de Requisitos (Especificação)

Identifica as necessidades/restrições do software

Perguntas importantes:

- O que?
- Quais?
- Quem?
- Por que?

Criação de modelos/abstrações para compreender o que é o software

Projeto/Arquitetura

Construção de modelos e abstrações de soluções para apoiar a implementação do software

Os modelos e abstrações devem ter detalhes suficientes para possibilitar a sua implementação física

Pergunta importante:

- Como implementar a solução?

Projeto/Arquitetura

Tipos de projeto:

- Da arquitetura
- De algoritmos
- De dados
- De interface

Codificação (Implementação)

Implementação da solução em uma linguagem de programação

Tradução dos modelos de projeto para linguagem de programação

Ferramentas de apoio, IDEs, geração de código, refatoração

Teste (Validação)

- Execução do software com o objetivo de encontrar falhas
- Quanto mais cedo a falha for encontrada, menor o custo de correção
- Fases de teste mais conhecidas: unidade, integração, sistema, aceitação
- Tipos de testes mais conhecidos: funcional (caixa-preta) e estrutural (caixa-branca)

Manutenção (Evolução)

- Todas as modificações realizadas no software após a sua entrega para o cliente
- Pode ser para corrigir um defeito, acrescentar uma função, melhorar uma função ou adaptar para um novo ambiente de utilização, etc.

Gerência de mudanças e de configuração do software

- Controla as mudanças realizadas no sistema (prioridades, impactos)
- Controla os artefatos gerados e manipulados durante o desenvolvimento do software
- Controla as diferentes versões dos diferentes artefatos utilizados para o lançamento de um software

Garantia de Qualidade de Software

- Atividades para garantir que o produto **correto** está sendo construído
(Qualidade do Produto)
- Atividades para garantir que o produto está sendo construído **corretamente**
(Qualidade do Processo)

Gestão de Projetos

Atividades gerenciais que envolve áreas como a iniciação, o planejamento, a execução, o monitoramento, o controle e o encerramento dos projetos de desenvolvimento

Possui atividades para gerenciar:

- escopo
- custo
- cronograma
- riscos
- recursos
- partes interessadas
- etc...

Engenharia de Software

Tipos de processos que se destacam:

- Processos tradicionais (preditivos)
- Métodos ágeis (adaptativos)

Engenharia de Software

Tipos de processos que se destacam:

- **Processos tradicionais (preditivos)**
- Métodos ágeis (adaptativos)

Processos tradicionais de desenvolvimento

Objetivo inicial (tradicional):

- Construir um software como qualquer outro produto da engenharia tradicional (civil, mecânica, aeronáutica, etc)

Processos tradicionais de desenvolvimento

Produtos da engenharia tradicional:

- Qual é o processo para construir um produto da engenharia tradicional (exemplo: casa, prédio ou ponte)?

Processos tradicionais de desenvolvimento

Produtos da engenharia tradicional:

- Qual é o processo para construir um produto da engenharia tradicional (exemplo: casa, prédio ou ponte)?
- Quais são as premissas de usar este tipo de abordagem para desenvolver um software?

Processos tradicionais de desenvolvimento

Premissas:

- É possível definir e especificar detalhadamente todos os requisitos de um software na fase de concepção
- Os requisitos do software permanecerão os mesmos ou serão pouco alterados durante todo o seu ciclo de desenvolvimento
- Os artefatos gerados pelas atividades de projeto são fundamentais para a codificação do software
- O cliente só usará o produto quando ele estiver totalmente finalizado

Processos tradicionais de desenvolvimento

Trouxeram disciplina para o desenvolvimento de software:

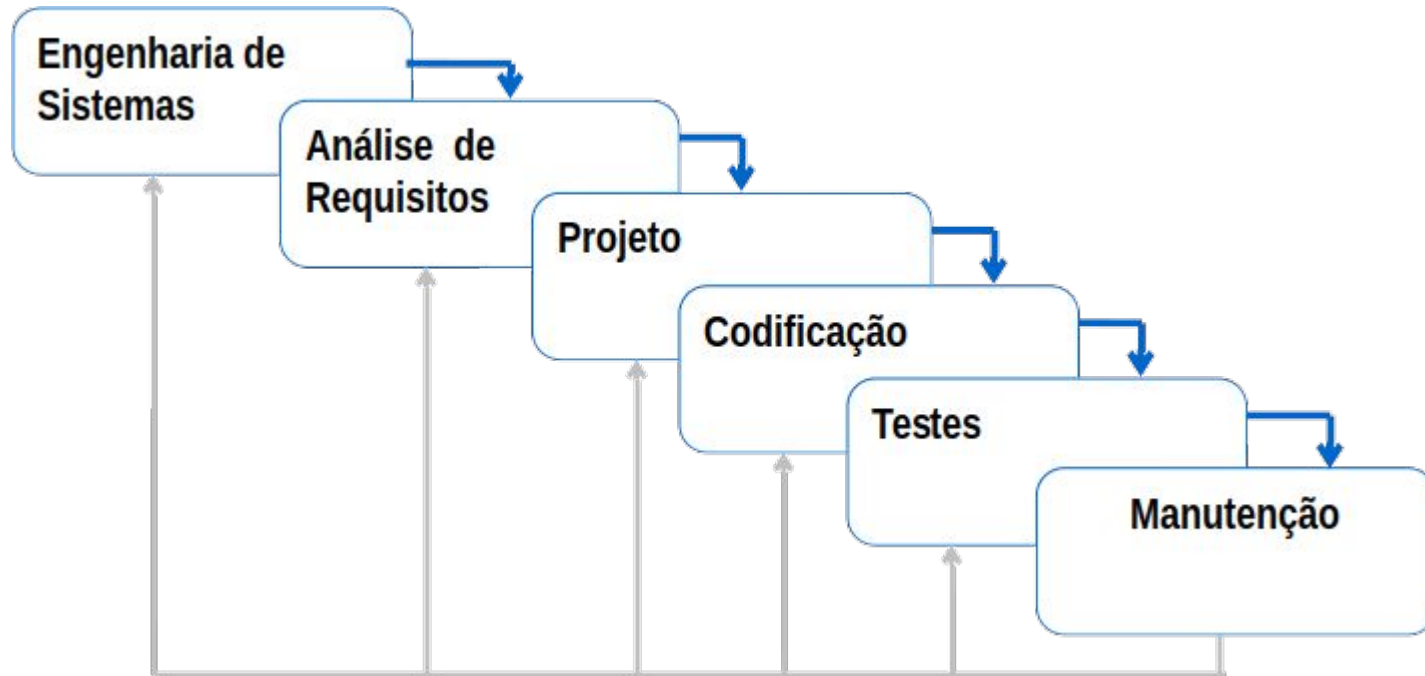
- Planejamento
- Divisão do processo em fases
- Divisão de papéis
- Sequenciamento de atividades
- Estimativas/Medição/Controle
- Documentação rigorosa
- Políticas de controle de alterações

Abordagens de desenvolvimento tradicionais

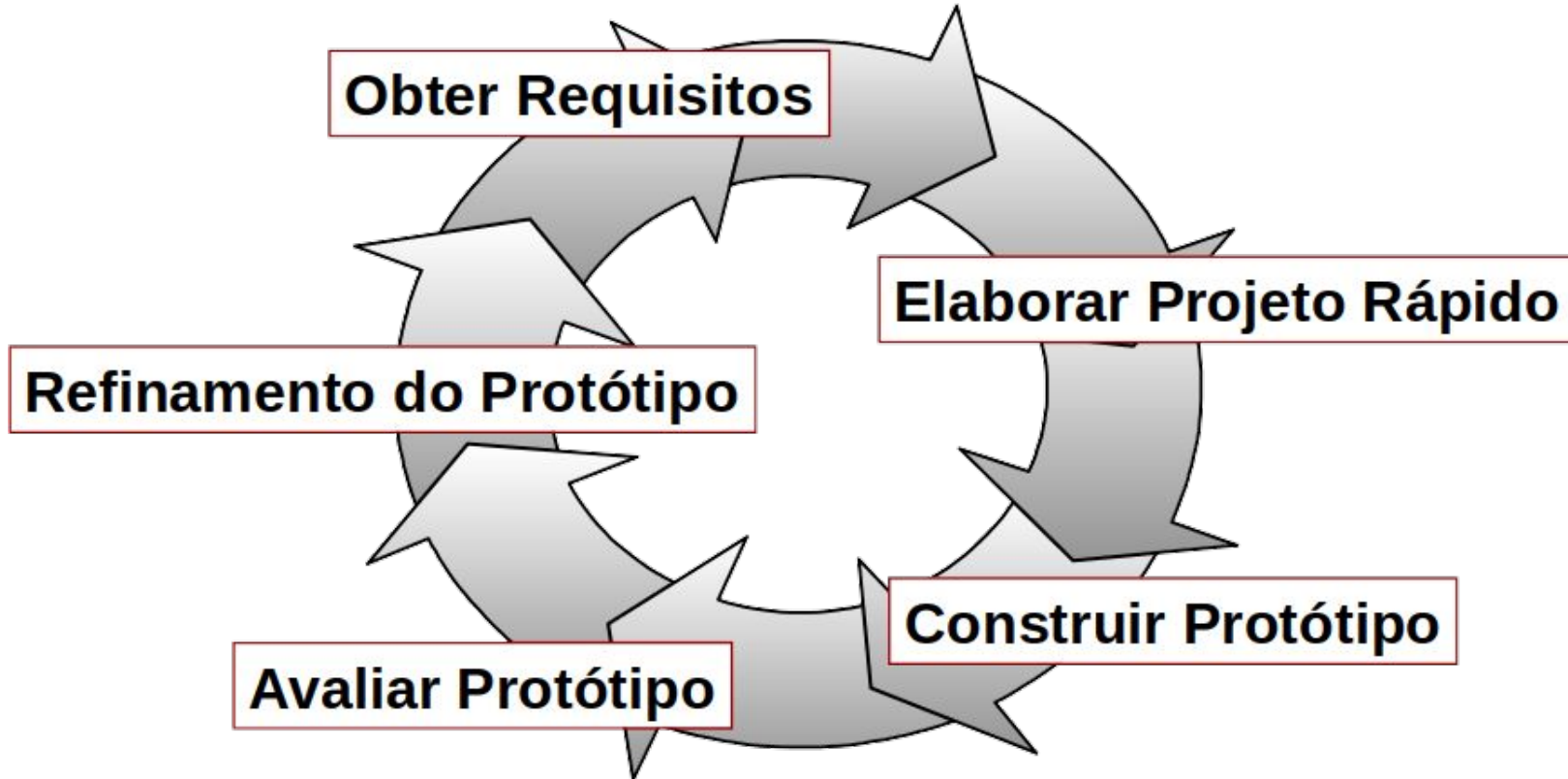
Exemplos

- Modelo Cascata
- Modelo espiral
- Modelo de Prototipação
- Modelos Evolutivos
- Processo Unificado

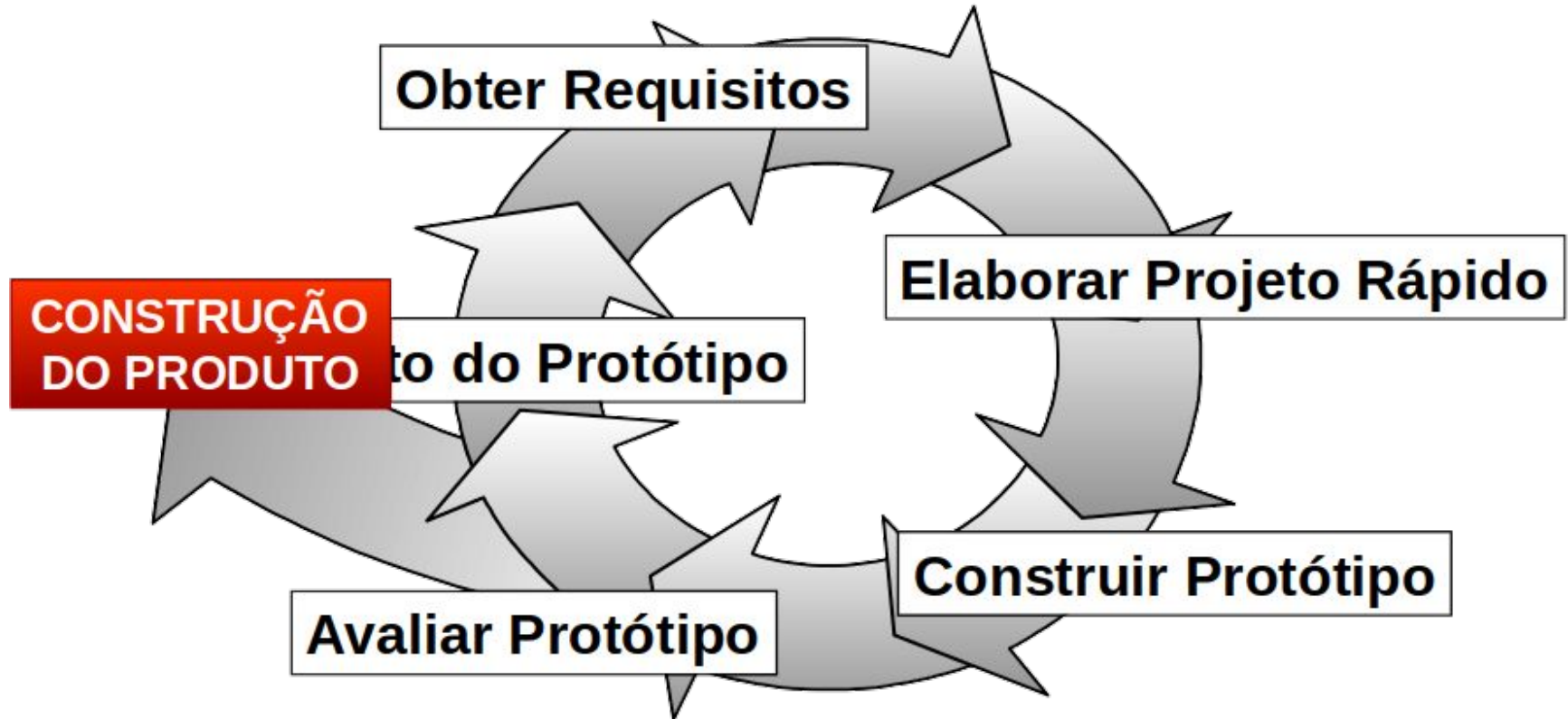
Modelo Cascata



Modelo de Prototipação



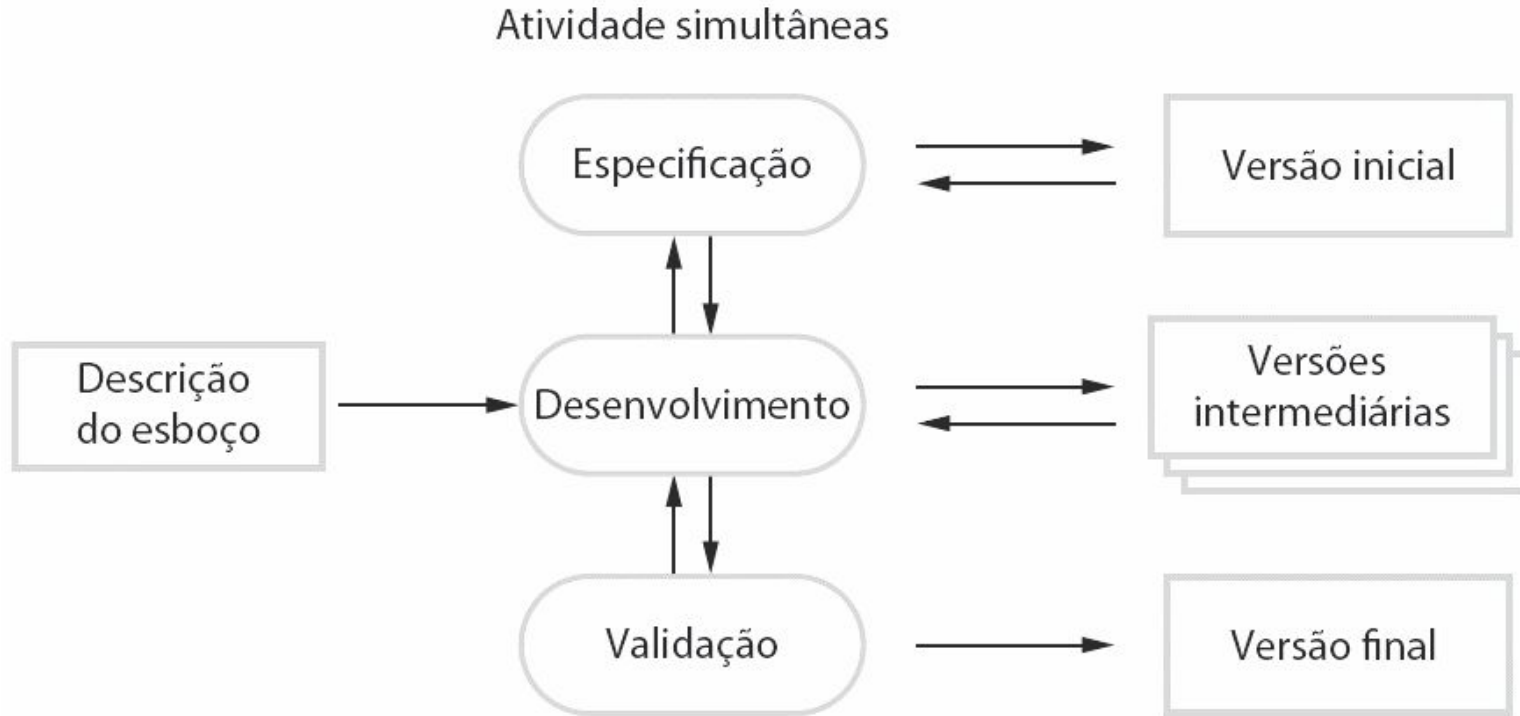
Modelo de Prototipação



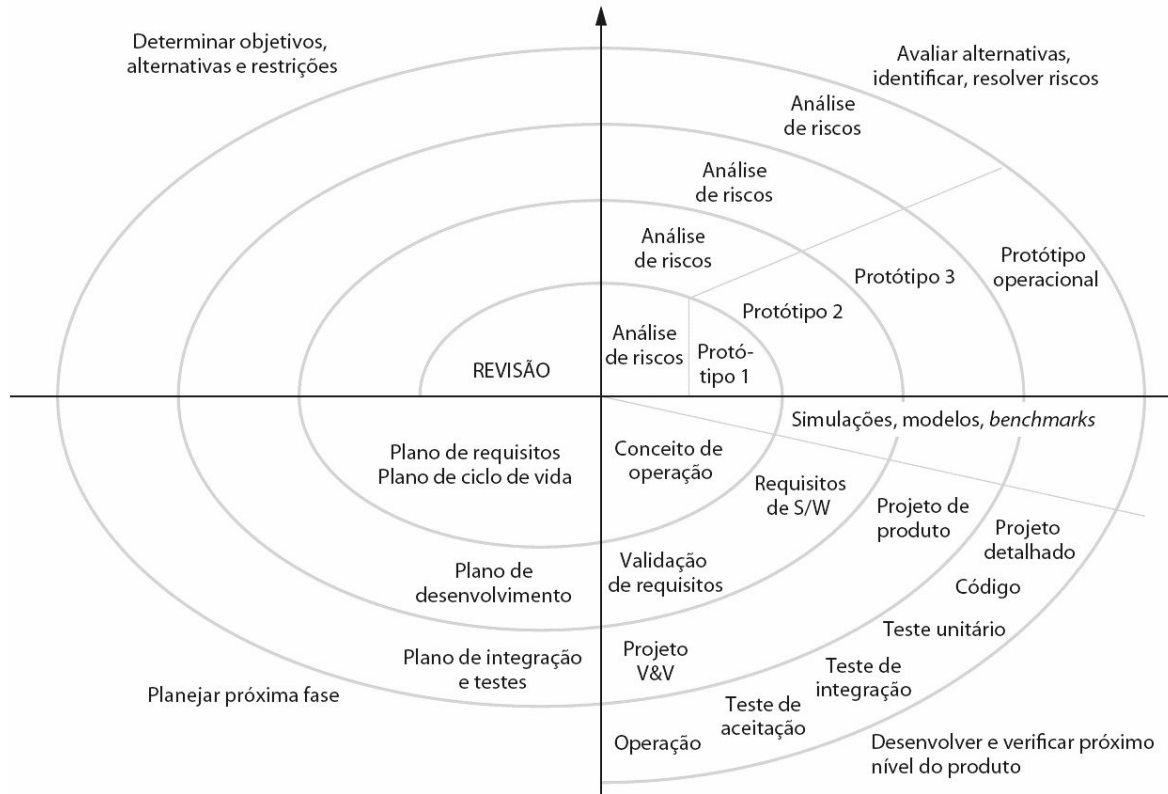
Modelo de Prototipação

- Fred Brooks disse o seguinte: “Planeje jogar uma implementação fora, pois você irá.”
- Brooks refere-se ao fato de que é mais fácil para os clientes entender o que eles querem uma vez que tenham usado um protótipo, e também é mais fácil para os engenheiros entenderem como construir um software uma vez que já o tenham feito uma vez.

Modelo Incremental



Modelo Espiral



Resumo dos métodos tradicionais

- Planejamento rigoroso
- Extensa documentação
- Etapas e papéis bem definidos
- Pontos de controle rígidos
- Foco na documentação e no processo
- Antecipação de todos os detalhes de desenvolvimento

Processos tradicionais de desenvolvimento

O que você achou deste tipo de processo? Alguma crítica?

Processos tradicionais de desenvolvimento

Críticas comuns:

- Divisão distinta de fases no projeto gera inflexibilidade uma vez que raramente os projetos seguem um fluxo sequencial.
- Requisitos totalmente especificados e “congelados” na primeira fase do projeto dificultam futuras mudanças.
- Arquitetura especificada e “congelada” na segunda fase do projeto torna a arquitetura pouco confiável diante de possíveis mudanças de requisitos.

Processos tradicionais de desenvolvimento

Críticas comuns:

- Grande dificuldade de alterações no projeto depois de decisões já tomadas.
- A excessiva documentação torna o processo “pesado”
- O excessivo planejamento “aprisiona” os envolvidos no desenvolvimento

Engenharia de Software

Tipos de processos que se destacam:

- Processos tradicionais (preditivos)
- **Métodos ágeis (adaptativos)**

Métodos ágeis

Muitos profissionais e pesquisadores não estavam contentes com a visão tradicional de que o software é um produto de engenharia tradicional

Modelos alternativos aos processos tradicionais foram criados para “abraçar” a mudança como um fato inevitável no processo de desenvolvimento

Uma das características que diferencia o software de um produto tradicional de engenharia é sua grande capacidade de mudança

Métodos ágeis

Portanto, desenvolvedores deveriam continuamente refinar protótipos com o feedback dos clientes a cada iteração até que o resultado final seja satisfatório

Neste contexto surgiu o manifesto ágil

Manifesto Ágil

Uma reunião realizada entre 17 gurus da comunidade de desenvolvimento para buscar alternativas aos processos de desenvolvimento tradicionais:

- “Estamos descobrindo maneiras melhores de desenvolver software fazendo-o nós mesmos e ajudando outros a fazê-lo.”
- Através deste trabalho, passamos a valorizar:
 - Indivíduos e interações mais que processos e ferramentas
 - Software em funcionamento mais que documentação abrangente
 - Colaboração com o (e do) cliente mais que negociação de contratos
 - Responder a mudanças mais do que seguir um plano
- Mesmo havendo valor nos itens à direita, valorizamos mais os itens à esquerda.

ACH2028

Qualidade de Software

Aula 02 - Revisão de Engenharia de Software

Prof. Marcelo Medeiros Eler
marceloeler@usp.br