

Engenharia de Software I



Rogério Eduardo Garcia
(rogerio.garcia@unesp.br)

Bacharelado em
Ciência da
Computação
2022

Bibliografia Básica



- PRESSMAN, R. S. Engenharia de Software, 6ª Edição, McGraw-Hill, Boston, 2001.
- SOMMERVILLE, I. Engenharia de Software, 6ª Edição, Addison-Wesley, São Paulo, 2003.
- PETERS, J. F.; PEDRYCZ, W. Engenharia de Software: teoria e prática, Editora Campus, Rio de Janeiro, 2001.
- PFLEEGER, S. L. Engenharia de Software, Teoria e Prática. Pearson Brasil, 2004.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

2

Metodologia



- Aulas expositivas teórico-práticas;
- Exercícios práticos;
- Projetos individuais e/ou em grupo;
- Seminários, individuais e em grupo, sobre tópicos abordados e relacionados.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

3

Avaliação



- As notas de todas as atividades – entre 0 (zero) e 10,0 (dez) – serão atribuídas individualmente, mesmo em atividades em grupo;
- A média final será calculada da seguinte maneira:
 - $MA = (NP1 + 2 * NP2) / 3$
 - $Mt = (NT1 + NT2 + \dots + NTn) / n$
 - $MT = (7 * NPJ + 3 * Mt)$
 - Média Final:
 - $MF = (MA + MT) / 2$ SE E SOMENTE SE $(MA \geq 5 \text{ E } MT \geq 5)$
 - Caso contrário $(MP < 5 \text{ OU } MT < 5)$ $MF =$ Menor Nota
 - Sendo:
 - $MF =$ Média Final.
 - $MA =$ Média de Provas
 - $MT =$ Média de Trabalhos e Projeto
 - $Mt =$ Média de Trabalhos

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

4

Tópicos da Disciplina



- Introdução: Histórico do desenvolvimento da área de Engenharia de Software. Conceitos básicos e terminologia.
- Objetivos da Engenharia de Software: Finalidade e aplicabilidade dos métodos e técnicas da Engenharia de Software, fornecendo uma visão ampla e abrangente da área. Qualidade de Software (ISO 9126)
- Processo de software: Definição de processo de software. Comparação entre modelos de ciclo de vida. Descrição das atividades no desenvolvimento de software.
- Análise de Requisitos de software: Técnicas de especificação de requisitos. Tipos de requisitos. Modelo de sistemas.
- Projeto de software: Arquitetura de software. Modelos para arquitetura de sistemas. Sistema de tempo real. Interface com o usuário.
- Verificação e validação de software: Distinção entre verificação e validação. Planejamento. Testes de software. Sistemas críticos.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

5

COVID-19



- Protocolo (diretrizes)
 - www.fct.unesp.br
- Inquérito de Sintomas
 - www.fct.unesp.br



Rede Virtual Multiprofissional de apoio à Saúde

- <https://www.ecaresentinel.com.br/>

Cronograma

Software e Engenharia de Software

- TÓPICOS**
- A importância do Software
 - Software
 - Aplicações de Software
 - Mitos de Software
 - Processo de Software
 - Modelos de Processo de Desenvolvimento de Software

SOFTWARE

O que é isso?

SOFTWARE

- “Software é um lugar onde sonhos são plantados e pesadelos são colhidos, um pântano abstrato e místico onde demônios terríveis competem com mágicas panaceias, um mundo de lobisomem e balas de prata”

Brad J. Cox

SOFTWARE

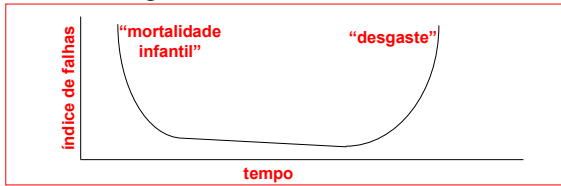
- INSTRUÇÕES que quando executadas produzem a função e o desempenho desejados
- ESTRUTURAS DE DADOS que possibilitam que os programas manipulem adequadamente a informação
- DOCUMENTOS que descrevem a operação e o uso dos programas

Características do Software

- Desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico
- Não se desgasta, mas se deteriora

Características do Software

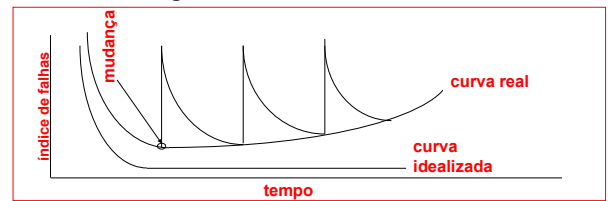
- Desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico
- Não se desgasta, mas se deteriora



Curva de Falhas do Hardware

Características do Software

- Desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico
- Não se desgasta, mas se deteriora



Curva de Falhas do Software

Características do Software

- Desenvolvido ou projetado por engenharia, não manufaturado no sentido clássico
- Não se desgasta, mas se deteriora
- A maioria é feita sob medida em vez de ser montada a partir de componentes existentes

Aplicações do Software

- Básico
- De Tempo Real
- Comercial
- Científico e de Engenharia
- Embutido
- De Computador Pessoal
- De Inteligência Artificial

Evolução do Software

(1950 - 1965)

- O hardware sofreu contínuas mudanças
- O software era uma arte "secundária" para a qual havia poucos métodos sistemáticos
- O hardware era de propósito geral
- O software era específico para cada aplicação
- Não havia documentação

Evolução do Software

(1965 - 1975)

- Multiprogramação e sistemas multiusuários
- Técnicas interativas
- Sistemas de tempo real
- 1ª geração de SGBD's
- Produto de software - software houses
- Bibliotecas de Software

Evolução do Software

(1965 - 1975)

- Cresce o número de sistemas baseado em computador
- Manutenção quase impossível

..... **CRISE DE SOFTWARE**

Evolução do Software

(1965 - 19

AFLIÇÃO CRÔNICA

- Cresce o número de sistemas baseado em computador
- Manutenção quase impossível

CRISE DE SOFTWARE

Refere-se a um conjunto de problemas encontrados no desenvolvimento de software

..... **CRISE DE SOFTWARE**

Evolução do Software

(1975 - hoje)

- Sistemas distribuídos
- Redes locais e globais
- Uso generalizado de microprocessadores - produtos inteligentes
- Hardware de baixo custo
- Impacto de consumo

Crise de Software - problemas

- 1) As estimativas de prazo e de custo frequentemente são imprecisas
 - “Não dedicamos tempo para coletar dados sobre o processo de desenvolvimento de software”
 - “Sem nenhuma indicação sólida de produtividade, não podemos avaliar com precisão a eficácia de novas ferramentas, métodos ou padrões”

Crise de Software - problemas

- 2) Insatisfação do cliente com o sistema concluído
 - “Os projetos de desenvolvimento de software normalmente são efetuados apenas com um vago indício das exigências do cliente”

Crise de Software - problemas

- 3) A qualidade de software às vezes é menos que adequada
 - Só recentemente começam a surgir conceitos quantitativos sólidos de garantia de qualidade de software

Crise de Software - problemas



- 4) O software existente é muito difícil de manter
- A tarefa de manutenção devora o orçamento destinado ao software
 - A facilidade de manutenção não foi enfatizada como um critério importante

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

25

Causas dos problemas associados à crise de software



1) PRÓPRIO CARÁTER DO SOFTWARE

- O software é um elemento de sistema lógico e não físico. Consequentemente o sucesso é medido pela qualidade de uma única entidade e não pela qualidade de muitas entidades manufaturadas

O software não se desgasta, mas se deteriora

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

26

Causas dos problemas associados à crise de software



2) FALHAS DAS PESSOAS RESPONSÁVEIS PELO DESENVOLVIMENTO DE SOFTWARE

- Gerentes sem nenhum *background* em software
- Profissionais da área de software têm pouco treinamento formal em novas técnicas para o desenvolvimento de software
- Resistência a mudanças

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

27

Causas dos problemas associados à crise de software



3) MITOS DO SOFTWARE

- Propagaram desinformação e confusão
 - Administrativos
 - Cliente
 - Profissional

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

28

Mitos do Software

ADMINISTRATIVOS:

Mito 1:

- Já temos um manual repleto de padrões e procedimentos para a construção de software.
- Isso não oferecerá ao meu pessoal tudo o que eles precisam saber?



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

29

Mitos do Software

ADMINISTRATIVOS:

Mito 1:

- **Realidade:**
 - Será que o manual é usado?
 - Os profissionais sabem que ele existe?
 - Ele reflete a prática moderna de desenvolvimento de software?
 - Ele é completo?



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

30

Mitos do Software

ADMINISTRATIVOS:



Mito 2:

- Meu pessoal tem ferramentas de desenvolvimento de software de última geração.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

31

Mitos do Software

ADMINISTRATIVOS:



Mito 2:

- Meu pessoal tem ferramentas de desenvolvimento de software de última geração.

Realidade:

- É preciso muito mais do que os mais recentes computadores e ferramentas para se fazer um desenvolvimento de software de alta qualidade.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

32

Mitos do Software

ADMINISTRATIVOS:



Mito 3:

- Se nós estamos atrasados nos prazos, podemos adicionar mais programadores e tirar o atraso.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

33

Mitos do Software

ADMINISTRATIVOS:



Mito 3:

- Se nós estamos atrasados nos prazos, podemos adicionar mais programadores e tirar o atraso.

Realidade:

- O desenvolvimento de software não é um processo mecânico igual à manufatura. Acrescentar pessoas em um projeto torna-o ainda mais atrasado.
- Pessoas podem ser acrescentadas, mas somente de uma forma planejada.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

34

Mitos do Software

CLIENTE:



Mito 1:

- Uma declaração geral dos objetivos é suficiente para se começar a escrever programas - podemos preencher os detalhes mais tarde.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

35

Mitos do Software

CLIENTE:



Mito 1:

- Uma declaração geral dos objetivos é suficiente para se começar a escrever programas - podemos preencher os detalhes mais tarde.

Realidade:

- Uma definição inicial ruim é a principal causa de fracassos dos esforços de desenvolvimento de software.
- É fundamental uma descrição formal e detalhada do domínio da informação, função, desempenho, interfaces, restrições de projeto e critérios de validação.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

36

Mitos do Software

CLIENTE:



Mito 2:

- Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodadas, porque o software é flexível.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

37

Mitos do Software

CLIENTE:



Mito 2:

- Os requisitos de projeto modificam-se continuamente, mas as mudanças podem ser facilmente acomodadas, porque o software é flexível.
- Realidade:** Uma mudança, quando solicitada tardiamente num projeto, pode ser maior do que a ordem de magnitude mais dispendiosa da mesma mudança solicitada nas fases iniciais.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

38

Mitos do Software



MAGNITUDE DAS MUDANÇAS

FASES	CUSTO DE MANUTENÇÃO
DEFINIÇÃO	1x
DESENVOLVIMENTO	1.5 a 6x
MANUTENÇÃO	60 a 100x

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

39

Mitos do Software

PROFISSIONAL:



Mito 1:

- Assim que escrevermos o programa e o colocarmos em funcionamento nosso trabalho estará completo.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

40

Mitos do Software

PROFISSIONAL:



Mito 1:

- Assim que escrevermos o programa e o colocarmos em funcionamento nosso trabalho estará completo.
- Realidade:** Os dados da indústria indicam que entre 50 e 70% de todo esforço gasto num programa serão despendidos depois que ele for entregue pela primeira vez ao cliente.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

41

Mitos do Software

PROFISSIONAL:



Mito 2:

- Enquanto não tiver o programa "funcionando", eu não terei realmente nenhuma maneira de avaliar sua qualidade.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

42

Mitos do Software

PROFISSIONAL:

Mito 2:

- Enquanto "função"

Realidade:

- Um programa funcionando é somente uma parte de uma Configuração de Software que inclui todos os itens de informação produzidos durante construção e manutenção do software.

Evolução do Software

- (1965 - 1975)
- Cresce o número de sistemas baseado em computador
- Manutenção quase impossível

CRISE DE SOFTWARE

Refere-se a um conjunto de **problemas** encontrados no desenvolvimento de software

Resposta à Crise de Software

Engenharia de Software

- A aplicação de uma abordagem sistemática, disciplinada e possível de ser medida para o desenvolvimento, operação e manutenção do software (IEEE)

Resposta à Crise de Software

PROCESSO DE SOFTWARE

- A aplicação de uma **abordagem** sistemática, disciplinada e possível de ser medida para o desenvolvimento, operação e manutenção do software (IEEE)

Engenharia de Software

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



.....“a construção por múltiplas pessoas de um software de múltiplas versões”

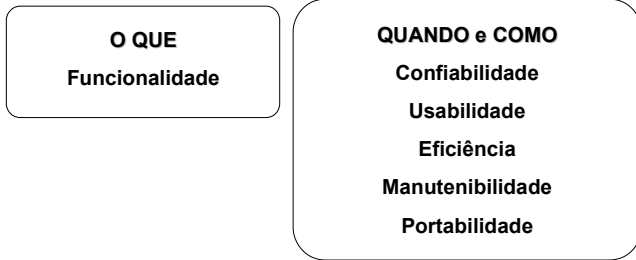
[Parnas 1987]

Qualidade

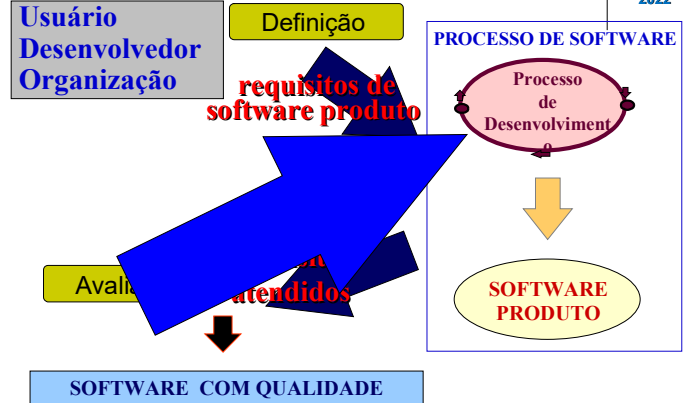
Software & Processo de Software

Qualidade de Software

Norma ISO/IEC 9126
Características



Qualidade de Software

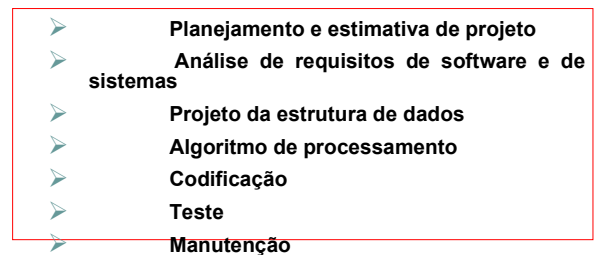


O Processo de Software

- Abrange um conjunto de três elementos fundamentais: **Métodos**, **Ferramentas** e **Procedimentos** para projetar, construir e manter grandes sistemas de software de forma profissional

O Processo de Software

- MÉTODOS:** proporcionam os detalhes de como fazer para construir o software



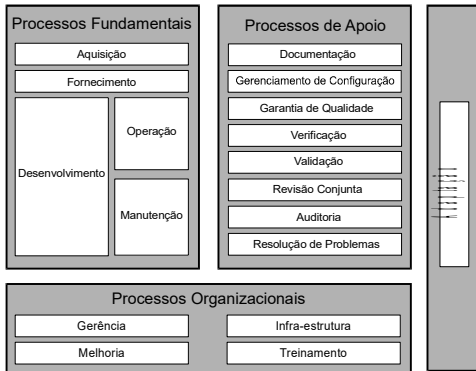
O Processo de Software

- FERRAMENTAS:** dão suporte automatizado aos métodos
 - Existem atualmente ferramentas para sustentar cada um dos métodos
 - Quando as ferramentas são integradas, é estabelecido um sistema de suporte ao desenvolvimento de software chamado *CASE - Computer Aided Software Engineering*

O Processo de Software

- PROCEDIMENTOS:** constituem o elo de ligação entre os métodos e ferramentas
 - Sequência em que os métodos devem ser aplicados
 - Produtos que se exige que sejam entregues
 - Controles que ajudam assegurar a qualidade e coordenar as alterações
 - Marcos de referência que possibilitam administrar o progresso do software

ISO 12207: Estrutura



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

55

Processo de Software com Qualidade

- A *Qualidade do Processo de Software* está relacionada à extensão na qual um processo de software específico é **eficiente** e é explicitamente **definido, gerenciado, medido e controlado**.
- A *Qualidade de Processo de Software* também implica em um **potencial** para crescimento na capacidade do processo de software e a **consistência** com a qual ele é aplicado em projetos por toda a organização.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

56

Processo de Software com Qualidade (SOMMERVILLE)

- Inteligibilidade
 - O processo é definido e inteligível
- Visibilidade
 - O progresso do processo é visível externamente
- Suportabilidade
 - O processo pode ser apoiado por ferramentas CASE

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

57

Processo de Software com Qualidade (SOMMERVILLE)

- Aceitabilidade
 - O processo é aceito por todos envolvidos nele
- Confiabilidade
 - Os erros do processo são descobertos antes que resultem em erros no produto
- Robustez
 - O processo pode continuar a despeito de problemas inesperados

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

58

Processo de Software com Qualidade (SOMMERVILLE)

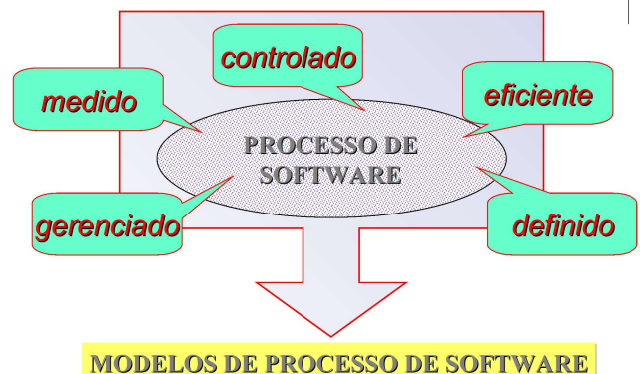
- Manutenibilidade
 - O processo pode evoluir para atender alterações de necessidades organizacionais
- Velocidade
 - Quão rápido o sistema pode ser produzido

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

59

Processo de Software com Qualidade



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

60

Fases Genéricas dos Modelos de Processo de ENGENHARIA



- Especificação - estabelecer os requisitos e restrições do sistema
- Projeto - produzir um modelo documentado do sistema
- Implementação - construir o sistema
- Teste - verificar se o sistema atende às especificações requeridas
- Instalação - liberar o sistema para o cliente e garantir que ele seja operacional
- Manutenção - eliminar defeitos e evoluir o sistema conforme demanda

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

61

Fases Genéricas dos Modelos de Processo de SOFTWARE



- Independentemente da natureza do projeto e aplicação os modelos de processo de software possuem:
 - Fase de definição
 - Fase de desenvolvimento
 - Fase de manutenção
 - Atividades de apoio

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

62

Fase de Definição do Processo de Software



Focaliza "**o que**" será desenvolvido

- Que informação vai ser processada
- Que função e desempenho são desejados
- Que comportamento pode ser esperado do sistema
- Que interfaces vão ser estabelecidas
- Que restrições de projeto existem
- Que critérios de validação são exigidos para definir um sistema bem sucedido
- Que tarefas serão realizadas

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

63

Fase de Definição do Processo de Software



Focaliza "**o que**" será desenvolvido

- Três tarefas principais ocorrem de alguma forma:
 - engenharia de sistemas
 - planejamento do projeto de software
 - análise de requisitos
- Que tarefas serão realizadas

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

64

Fase de Desenvolvimento do Processo de Software



Focaliza "**como**" o software será desenvolvido

- Como os dados vão ser estruturados
- Como a função vai ser implementada como uma arquitetura de software
- Como os detalhes procedimentais vão ser implementados
- Como as interfaces vão ser caracterizadas
- Como o projeto será traduzido em uma linguagem de programação
- Como os testes serão efetuados

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

65

Fase de Desenvolvimento do Processo de Software



Focaliza "**como**" o software será desenvolvido

- Três tarefas técnicas específicas deverão ocorrer sempre:
 - projeto de software
 - geração de código
 - Inspeção e teste de software
- Como o projeto será traduzido em uma linguagem de programação
- Como os testes serão efetuados

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

66

Fase de Manutenção do Processo de Software



Focaliza as "**mudanças**" que ocorrerão depois que o software for liberado para uso operacional

- A fase de manutenção re replica os passos das fases de definição e desenvolvimento, mas faz isso no contexto de um software existente.

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

67

Fase de Manutenção do Processo de Software



Focaliza as "**mudanças**" que ocorrerão depois que o software for liberado para uso operacional

- **As mudanças estão associadas com:**
 - **Correção** de erros/defeitos
 - **Adaptações** exigidas conforme o ambiente do software evolui
 - Mudanças devido a **melhoramentos** ocorridos por alterações nos requisitos dos clientes

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

68

Atividades de Apoio ao Processo de Software



- As três fases genéricas do processo de software são complementadas por uma série de **atividades de apoio**
- As atividades de apoio são aplicadas durante toda a engenharia do software

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

69

Atividades de Apoio ao Processo de Software



- As três fases genéricas do processo de software são complementadas por uma série de **atividades de apoio**

Atividades típicas nessa categoria são:

- Controle e Acompanhamento do Projeto de Software
- Revisões Técnicas Formais
- Garantia de Qualidade de Software
- Gerenciamento de Configuração de Software
- Preparação e Produção de Documentos
- Gerenciamento de Reusabilidade
- Medidas

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

70

Modelos de Processo de Desenvolvimento de Software



- Existem vários *modelos de processo de desenvolvimento de software* (ou *paradigmas de engenharia de software*)
- Cada um representa uma tentativa de colocar ordem em uma atividade inerentemente caótica
- Pode-se citar os seguintes **modelos de processo de desenvolvimento de software**

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

71

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

72

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado **Modelo Cascata**
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

73

O Modelo Cascata



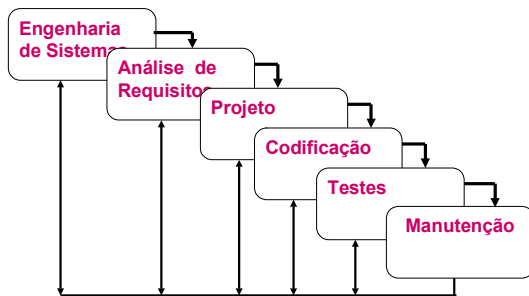
- Modelo mais antigo e o mais amplamente usado da engenharia de software
- Modelado em função do ciclo da engenharia convencional
- Requer uma abordagem sistemática, sequencial ao desenvolvimento de software
- O resultado de uma fase se constitui na entrada da outra

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

74

O Modelo Cascata

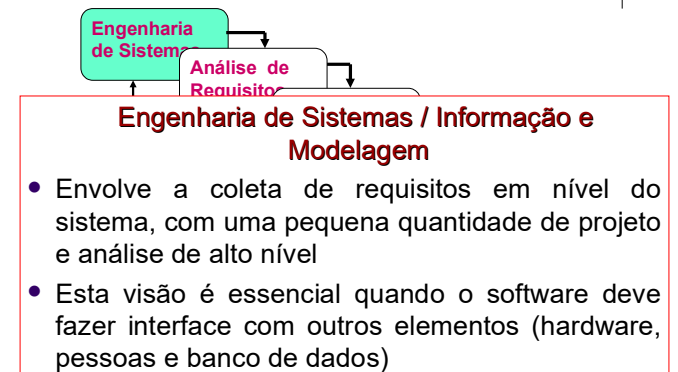


05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

75

O Modelo Cascata

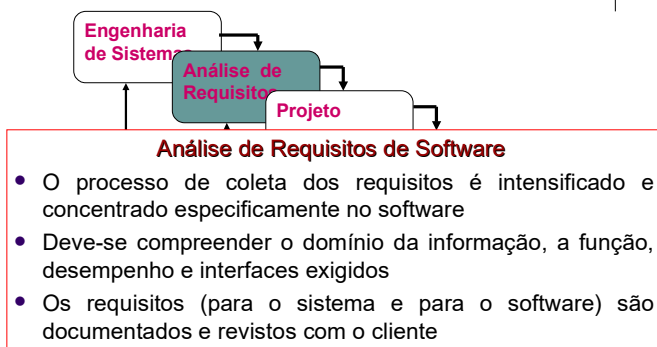


05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

76

O Modelo Cascata

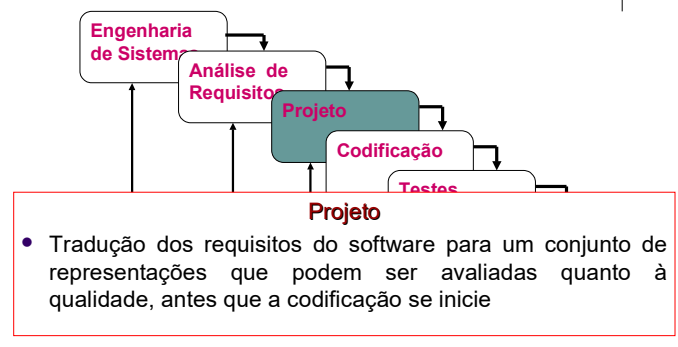


05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

77

O Modelo Cascata

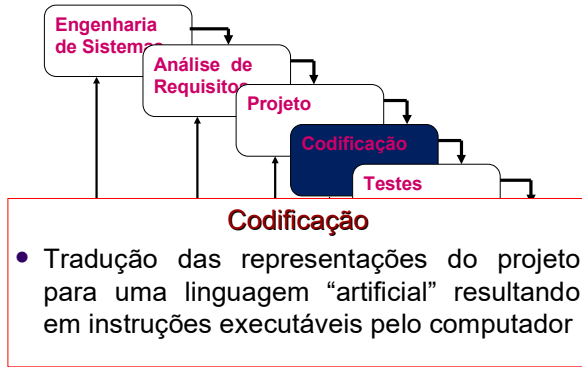


05/04/22

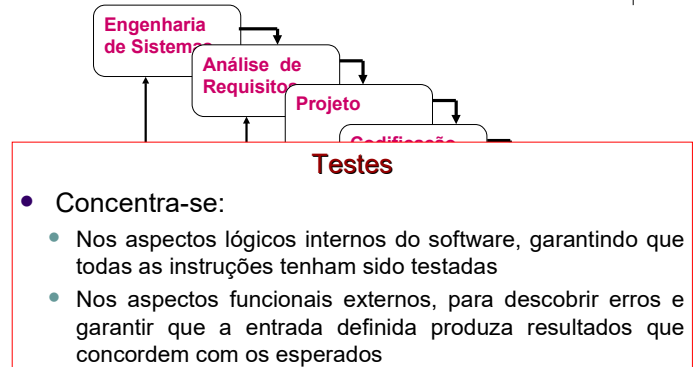
Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

78

O Modelo Cascata



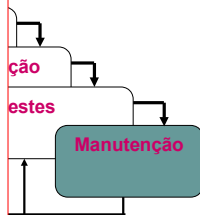
O Modelo Cascata



O Modelo Cascata

Manutenção

- Provavelmente o software deverá sofrer mudanças depois que for entregue ao cliente
- Causas das mudanças: erros, adaptação do software para acomodar mudanças em seu ambiente externo e exigência do cliente para acréscimos funcionais e de desempenho



Problemas com o Modelo Cascata

- Projetos reais raramente seguem o fluxo “sequencial” que o modelo propõe
- Logo no início é difícil estabelecer explicitamente todos os requisitos. No começo dos projetos sempre existe uma incerteza natural
- O cliente deve ter paciência. Uma versão executável do software só fica disponível em uma etapa avançada do desenvolvimento

Problemas com o Modelo Cascata

- Projetos reais raramente seguem o fluxo “sequencial” que o modelo propõe
 - Logo no início é difícil estabelecer explicitamente todos os requisitos. No começo dos projetos sempre existe uma incerteza natural
 - O cliente deve ter paciência. Uma versão executável do software só fica disponível em uma etapa avançada do desenvolvimento
- Embora o Modelo Cascata tenha fragilidades, ele é significativamente melhor do que uma abordagem casual ao desenvolvimento de software**

O Modelo Cascata

- O modelo Cascata trouxe contribuições importantes para o processo de desenvolvimento de software:
 - Imposição de disciplina, planejamento e gerenciamento
 - A implementação do produto deve ser postergada até que os objetivos tenham sido completamente entendidos

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- **O Modelo de Prototipação**
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

85

O Modelo de Prototipação



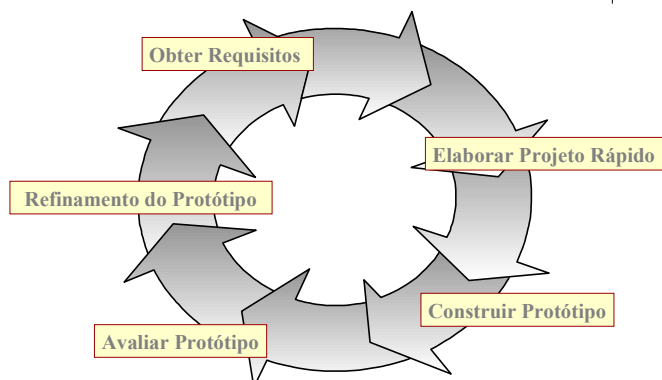
- O objetivo é entender os requisitos do usuário e, assim, obter uma melhor definição dos requisitos do sistema
- Possibilita que o desenvolvedor crie um modelo (protótipo) do software que deve ser construído
- Adequado para quando o cliente não definiu detalhadamente os requisitos

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

86

O Modelo de Prototipação para obtenção dos requisitos

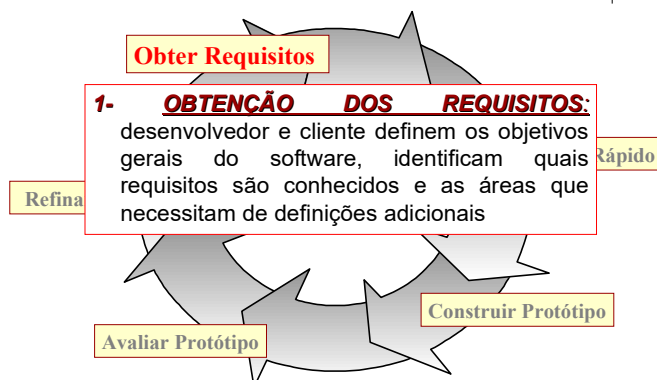


05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

87

O Modelo de Prototipação para obtenção dos requisitos

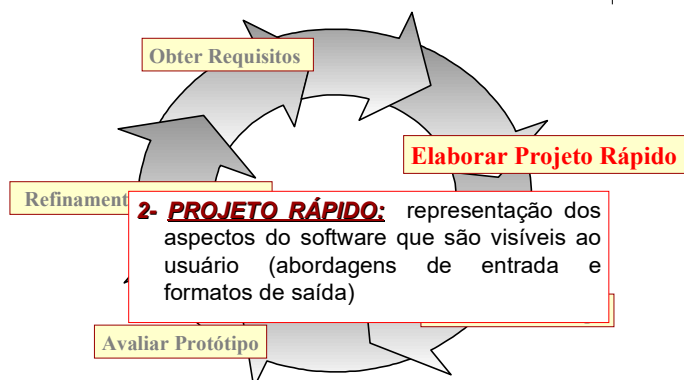


05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

88

O Modelo de Prototipação para obtenção dos requisitos



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

89

O Modelo de Prototipação para obtenção dos requisitos



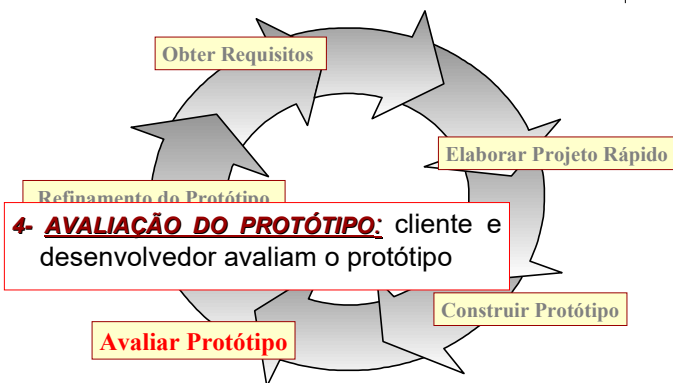
05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

90

O Modelo de Prototipação

para obtenção dos requisitos



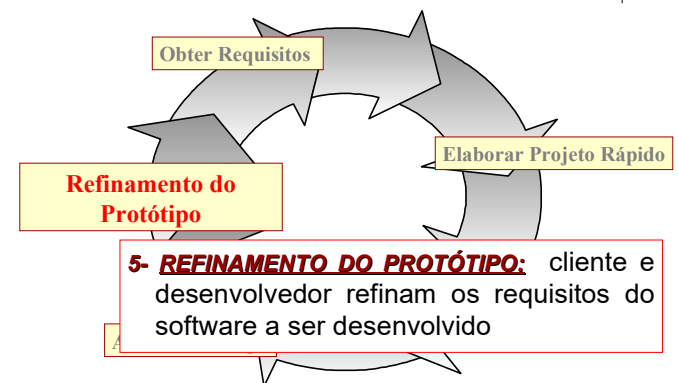
05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

91

O Modelo de Prototipação

para obtenção dos requisitos



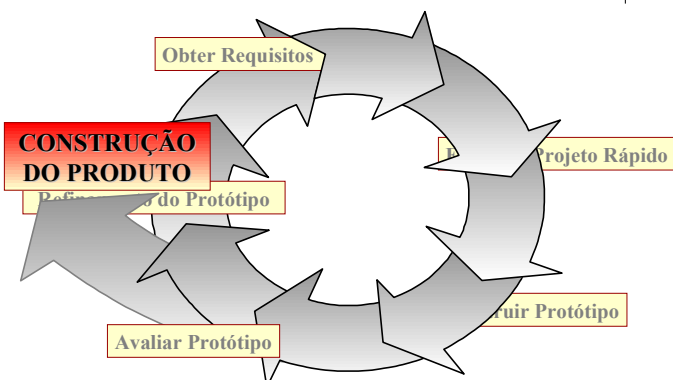
05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

92

O Modelo de Prototipação

para obtenção dos requisitos



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

93

O Modelo de Prototipação

para obtenção dos requisitos



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

94

Problemas com a Prototipação

- Cliente não sabe que o software que ele vê não considerou, durante o desenvolvimento, a qualidade global e a manutenibilidade a longo prazo
- Desenvolvedor frequentemente faz uma implementação comprometida (utilizando o que está disponível) com o objetivo de produzir rapidamente um protótipo

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

95

Comentários sobre o Modelo de Prototipação

- Ainda que possam ocorrer problemas, a prototipação é um ciclo de vida eficiente
- A chave é definir-se as regras do jogo logo no começo
- O cliente e o desenvolvedor devem ambos concordar que o protótipo seja construído para servir como um mecanismo a fim de definir os requisitos

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

96

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
 - O Modelo de Desenvolvimento Concorrente
- Modelos de Métodos Formais
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

97

O Modelo RAD



- **RAD (Rapid Application Development)** é um modelo sequencial linear que enfatiza um **ciclo** de desenvolvimento extremamente **curto**
- O desenvolvimento rápido é obtido usando uma **abordagem** de construção baseada em **componentes**

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

98

O Modelo RAD



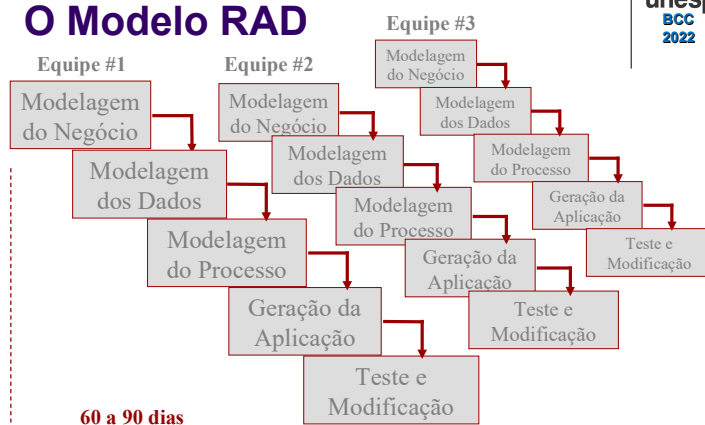
- Os **requisitos** devem ser bem **entendidos** e o **alcance** do projeto **restrito**
- O modelo RAD é usado principalmente para aplicações de **sistema de informação**
- Cada função principal pode ser direcionada para uma equipe RAD separada e então integrada para formar o todo

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

99

O Modelo RAD



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

100

O Modelo RAD



Desvantagens:

- Exige recursos humanos suficientes para todas as equipes
- Exige que desenvolvedores e clientes estejam comprometidos com as atividades de "fogo-rápido" a fim de terminar o projeto num prazo curto

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

101

O Modelo RAD



- Nem todos os tipos de aplicação são apropriadas para o RAD:
 - Deve ser possível a modularização efetiva da aplicação
 - Se alto desempenho é uma característica e o desempenho é obtido sintonizando as interfaces dos componentes do sistema, a abordagem RAD pode não funcionar

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

102

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
 - O Modelo de Desenvolvimento Concorrente
- Modelos de Métodos Formais
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

103

Modelos Evolutivos de Processo



- Existem **situações** em que a engenharia de software necessita de um modelo de processo que possa **acomodar** um produto que **evolui** com o tempo

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

104

Modelos Evolutivos de Processo



- Quando os requisitos de produto e de negócio mudam conforme o desenvolvimento procede
- Quando uma data de entrega apertada (mercado) - impossível a conclusão de um produto completo
- Quando um conjunto de requisitos importantes é bem conhecido, porém os detalhes ainda devem ser definidos

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

105

Modelos Evolutivos de Processo



- Modelos evolutivos são **iterativos**
- Possibilitam o desenvolvimento de **versões** cada vez mais completas do software

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

106

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
 - O Modelo de Desenvolvimento Concorrente
- Modelos de Métodos Formais
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

107

O Modelo Incremental



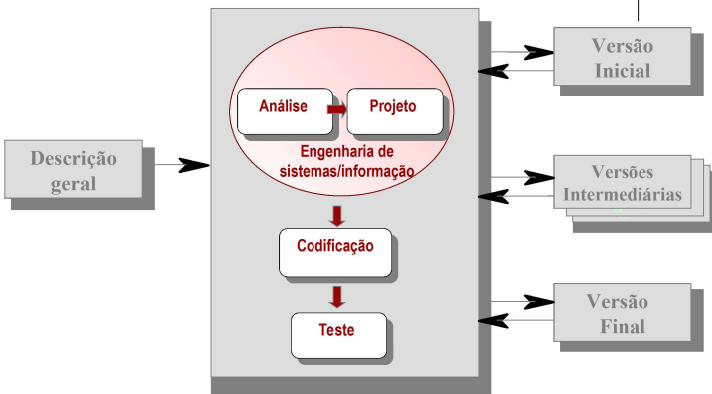
- O modelo incremental combina elementos do modelo cascata (aplicado repetidamente) com a filosofia iterativa da prototipação
- O objetivo é trabalhar junto do usuário para descobrir seus requisitos, de maneira incremental, até que o produto final seja obtido

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

108

O Modelo Incremental (SOMMERVILLE)



O Modelo Incremental

- A versão inicial é frequentemente o **núcleo** do produto (a parte mais importante)
 - A evolução acontece quando novas características são adicionadas à medida que são sugeridas pelo usuário
- Este modelo é importante quando é difícil estabelecer *a priori* uma especificação detalhada dos requisitos

O Modelo Incremental

- O modelo incremental é mais apropriado para sistemas pequenos
- As novas versões podem ser planejadas de modo que os riscos técnicos possam ser administrados (Ex. disponibilidade de determinado hardware)

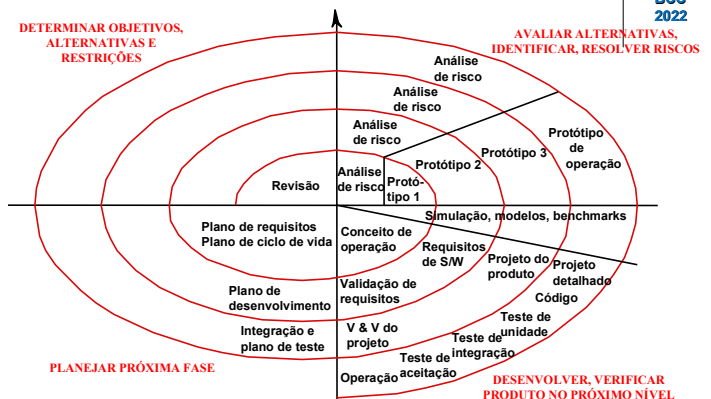
Modelos de Processo de Desenvolvimento de Software

- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - **O Modelo Espiral**
 - O Modelo de Montagem de Componentes
- Técnicas de Quarta Geração

O Modelo Espiral

- O modelo espiral **acopla** a natureza **iterativa** da **prototipação** com os aspectos controlados e **sistemáticos** do modelo **cascata**
- O modelo espiral é dividido em uma série de **atividades de trabalho** ou **regiões de tarefa**
- Existem tipicamente de **3 a 6** regiões de tarefa

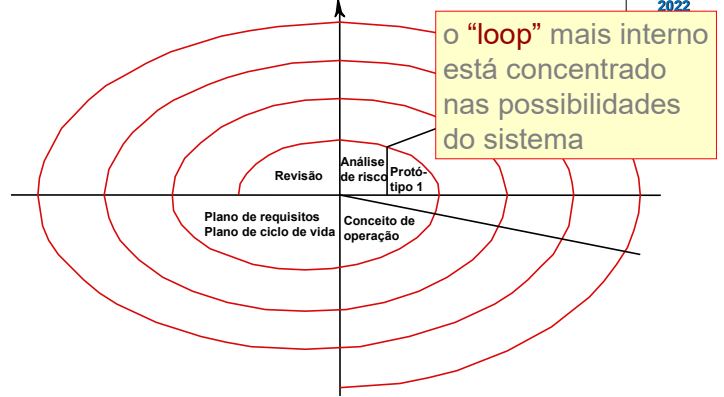
O Modelo Espiral (com 4 regiões)



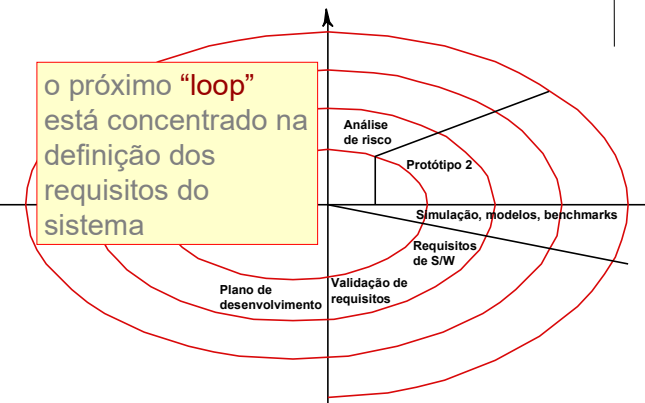
O Modelo Espiral



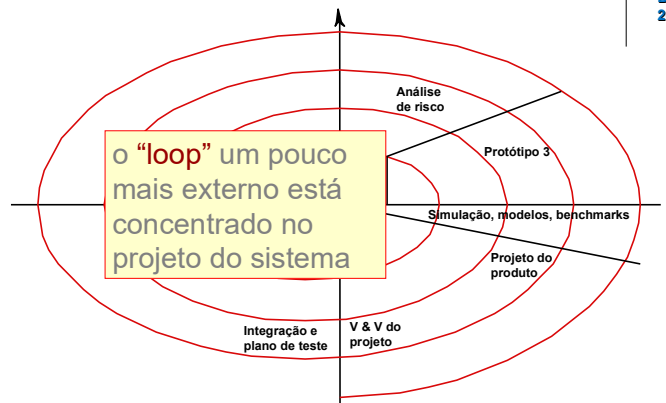
O Modelo Espiral



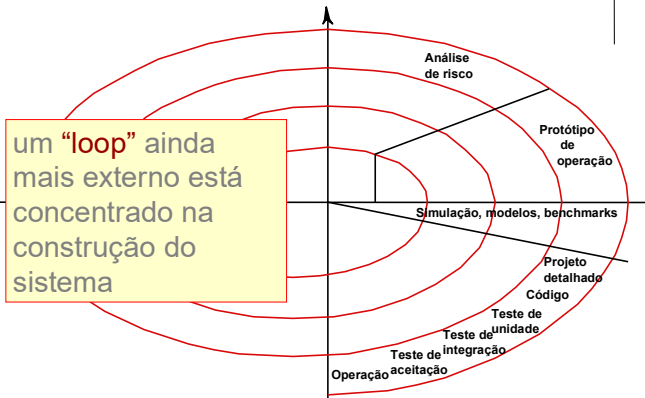
O Modelo Espiral



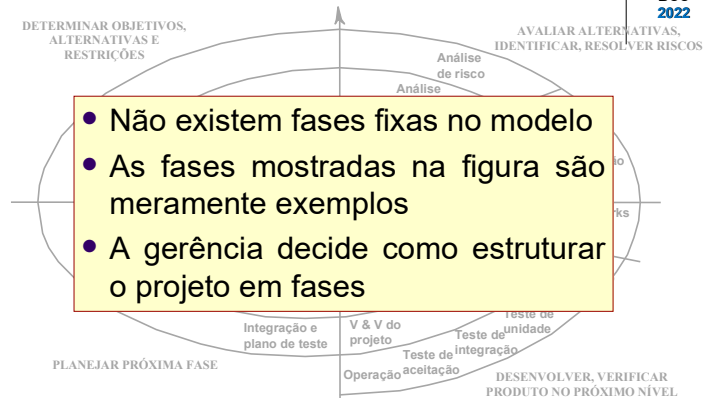
O Modelo Espiral



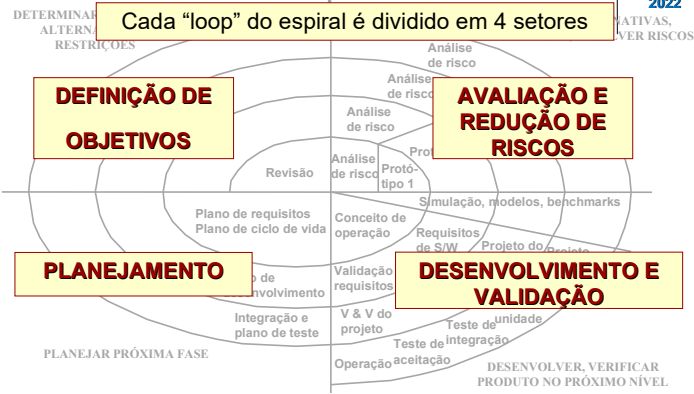
O Modelo Espiral



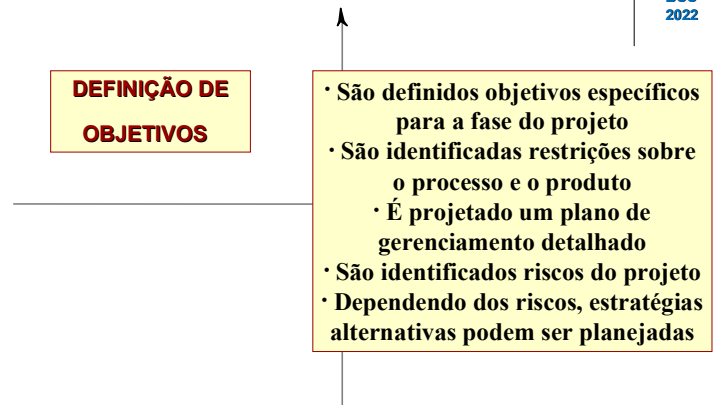
O Modelo Espiral



O Modelo Espiral (com 4 regiões)



O Modelo Espiral (com 4 regiões)



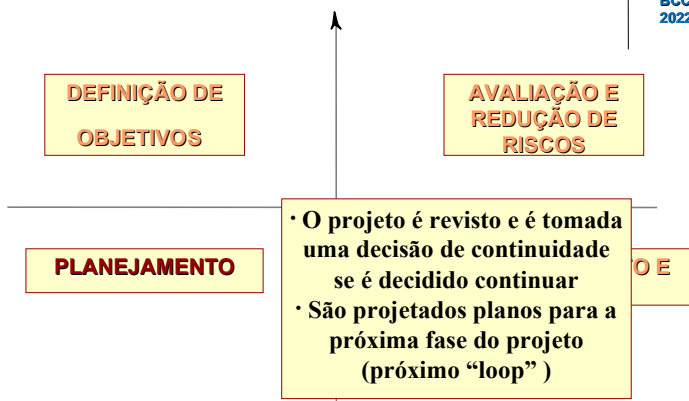
O Modelo Espiral (com 4 regiões)



O Modelo Espiral (com 4 regiões)



O Modelo Espiral (com 4 regiões)



O Modelo Espiral

- Engloba as melhores características do ciclo de vida Clássico e da Prototipação, adicionando um novo elemento: a **Análise de Risco**
- Segue a abordagem de passos sistemáticos do Ciclo de Vida Clássico incorporando-os numa estrutura **iterativa** que reflete mais realisticamente o mundo real
- Usa a **Prototipação**, em qualquer etapa da evolução do produto, como mecanismo de redução de riscos

Comentários sobre o Modelo Espiral



- É, atualmente, a abordagem mais realística para o desenvolvimento de software em grande escala
- Usa uma abordagem que capacita o desenvolvedor e o cliente a entender e reagir aos riscos em cada etapa evolutiva
- Pode ser difícil convencer os clientes que uma abordagem "evolutiva" é controlável

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

127

Comentários sobre o Modelo Espiral



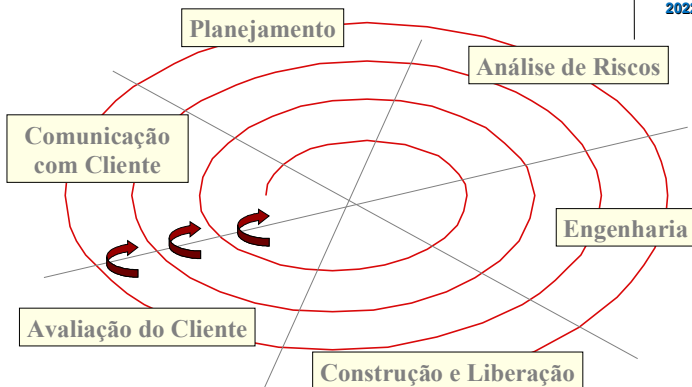
- Exige considerável experiência na determinação de riscos e depende dessa experiência para ter sucesso
- O modelo é relativamente novo e não tem sido amplamente usado. Demorará muitos anos até que a eficácia desse modelo possa ser determinada com certeza absoluta

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

128

O Modelo Espiral (com 6 regiões)



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

129

Modelos de Processo de Desenvolvimento de Software



- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - **O Modelo de Montagem de Componentes**
- Técnicas de Quarta Geração

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

130

O Modelo de Montagem de Componentes



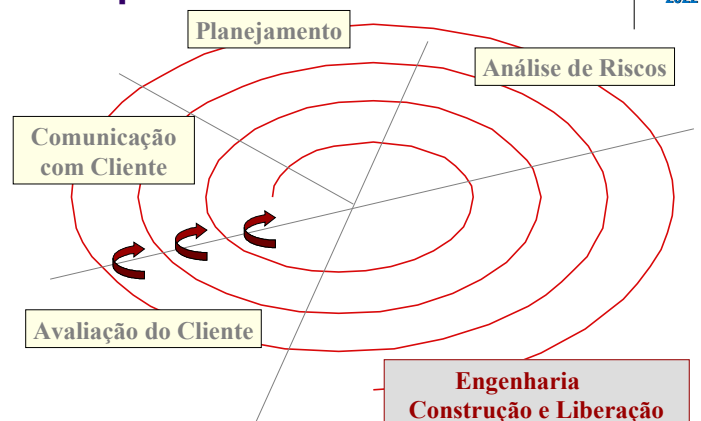
- Utiliza **tecnologias orientadas a objeto**
- Quando projetadas e implementadas apropriadamente as **classes** orientadas a objeto são **reutilizáveis** em diferentes aplicações e arquiteturas de sistema
- O modelo de montagem de componentes incorpora muitas das características do **modelo espiral**

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

131

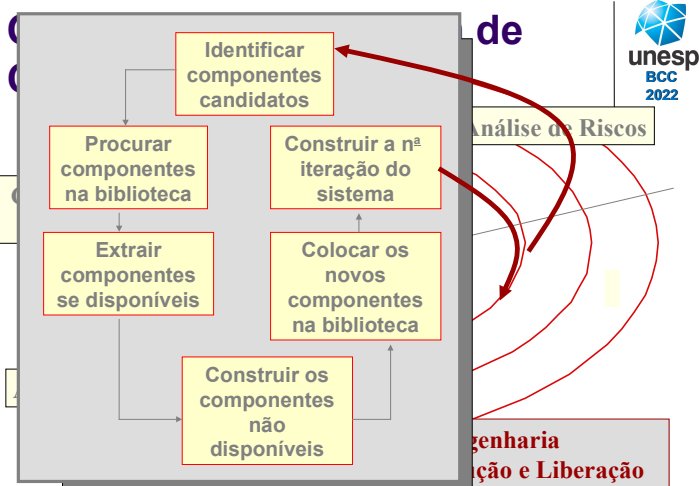
O Modelo de Montagem de Componentes



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

132



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

133

O Modelo de Montagem de Componentes

- O modelo de montagem de componentes conduz ao **reúso** do software
- A **reusabilidade** fornece uma série de benefícios:
 - Redução de 70% no tempo de desenvolvimento
 - Redução de 84% no custo do projeto
 - Índice de produtividade de 26.2 (normal da indústria é de 16.9)
- Esses resultados dependem da **robustez** da biblioteca de componentes

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

134

Modelos de Processo de Desenvolvimento de Software

- O Modelo Sequencial Linear
 - Também chamado Modelo Cascata
- O Modelo de Prototipação
- O Modelo RAD (Rapid Application Development)
- Modelos Evolutivos de Processo de Software
 - O Modelo Incremental
 - O Modelo Espiral
 - O Modelo de Montagem de Componentes
- **Técnicas de Quarta Geração**

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

135

Técnicas de 4ª Geração

- Concentra-se na capacidade de se especificar o software a uma máquina em um nível que esteja próximo à linguagem natural
- Engloba um conjunto de ferramentas de software que possibilitam que:
 - O **sistema seja especificado em uma linguagem de alto nível**
 - O **código fonte seja gerado automaticamente a partir dessas especificações**

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

136

Ferramentas do Ambiente das Técnicas de 4ª Geração

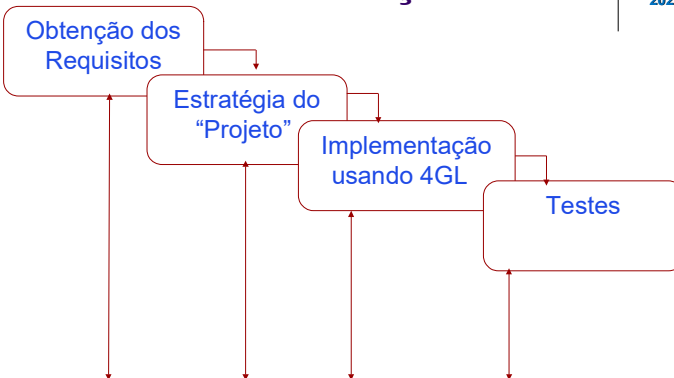
- O ambiente de desenvolvimento de software que sustenta o ciclo de vida de 4ª geração inclui as ferramentas:
 - Linguagens não procedimentais para consulta de banco de dados
 - Geração de relatórios
 - Manipulação de dados
 - Interação e definição de telas
 - Geração de códigos
 - Capacidade gráfica de alto nível
 - Capacidade de planilhas eletrônicas

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

137

Técnicas de 4ª Geração



05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

138

Técnicas de 4ª Geração

Obtenção dos
Requisitos

OBTENÇÃO DOS REQUISITOS:

- O cliente descreve os requisitos os quais são traduzidos para um protótipo operacional
 - O cliente pode estar inseguro quanto aos requisitos
 - O cliente pode ser incapaz de especificar as informações de um modo que uma ferramenta 4GL possa consumir
 - As 4GLs atuais não são sofisticadas suficientemente para acomodar a verdadeira "linguagem natural"

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

139

Técnicas de 4ª Geração

Obtenção dos
Requisitos

ESTRATÉGIA DO "PROJETO":

- Para pequenas aplicações é possível mover-se do passo de Obtenção dos Requisitos para o passo de Implementação usando uma linguagem de quarta geração
- Para grandes projetos é necessário desenvolver uma estratégia de projeto. De outro modo ocorrerão os mesmos problemas encontrados quando se usa abordagem convencional (baixa qualidade)

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

140

Técnicas de 4ª Geração

Obtenção dos
Requisitos

Estratégia do
"Projeto"

Implementação
usando 4GL

IMPLEMENTAÇÃO USANDO 4GL:

- Os resultados desejados são representados de modo que haja geração automática de código. Deve existir uma estrutura de dados com informações relevantes e que seja acessível pela 4GL

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

141

Técnicas de 4ª Geração

Obtenção dos
Requisitos

Estratégia do
"Projeto"

Implementação
usando 4GL

Testes

TESTES:

- O desenvolvedor deve efetuar testes e desenvolver uma documentação significativa. O software desenvolvido deve ser construído de maneira que a manutenção possa ser efetuada prontamente

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

142

Comentários sobre as Técnicas de 4ª Geração

• PROPONENTES:

- Redução dramática no tempo de desenvolvimento do software (aumento de produtividade)

• OPONENTES:

- As 4GL atuais não são mais fáceis de usar do que as linguagens de programação
- O código fonte produzido é ineficiente
- A manutenibilidade de sistemas usando técnicas 4GL ainda é questionável

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

143

Para escolha de um Modelo de Processo de Software:

- **Natureza** do projeto e da aplicação
- **Métodos** e **ferramentas** a serem usados
- **Controles** e **produtos** que precisam ser entregues

05/04/22

Ciência da Computação - Engenharia de Software I - Rogério Eduardo Garcia

144

- A importância do Software
- Software
- Aplicações do Software
- Mitos do Software
- Processo de Software
- Modelos de Processo de Desenvolvimento de Software