

O PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

Projeto de Programas – PPR0001

O que é Software

- O que é um software?

O que é Software

- O que é um software?
 - *“Software são os programas de computador e a documentação associada” (Sommerville, 2003)*
- Diversos tipos de produtos possuem documentação associada:
 - Casas e edifícios: planta
 - Remédios: bula
 - Eletrodomésticos: especificações de montagem, manual do consumidor
 - Software: especificações de implementação, manual do usuário
- E.g.: um jogo é produzido através de especificações, principalmente do game designer, e pode vir acompanhado de um manual de uso.

Sobre o desenvolvimento de Software

- O desenvolvimento de software é um processo complexo
- Dados levantados pelo *Standish Group* (Chaos Report, 1994):
 - 10% dos projetos terminaram dentro do prazo
 - Atraso médio nos projetos: 1 ano
 - 25% dos projetos são descontinuados antes de chegar ao fim
 - 60% dos projetos saem acima do custo esperado
- Motivos (Goldratt, 2002):
 - Expansão de funcionalidades
 - “Lapidação a ouro” (quando tenta-se fazer detalhes muito elaborados no design ou na implementação que muitas vezes nem foram requisitados)
 - Negligência ao controle de qualidade
 - Cronogramas “super” otimistas

Sobre o desenvolvimento de Software

- Motivos (Goldratt, 2002) (...):
 - Trabalho em muitos projetos ao mesmo tempo
 - **Planejamento pobre**
 - Projetos orientados a pesquisa (resultados da pesquisa são incertos)
 - Pessoal não qualificado para as atividades
 - Distância dos *stakeholders*
- Tentativa de minimizar problemas => estudos sugerem uso de processos ou metodologias de desenvolvimento de software:
 - ICONIX
 - EUP (*Enterprise Unified Process*)
 - RUP (*Rational Unified Process*)
 - XP (*Extreme Programming*)
 - OPEN (*Object-Oriented Process, Environment and Notation*)

Sobre o desenvolvimento de Software

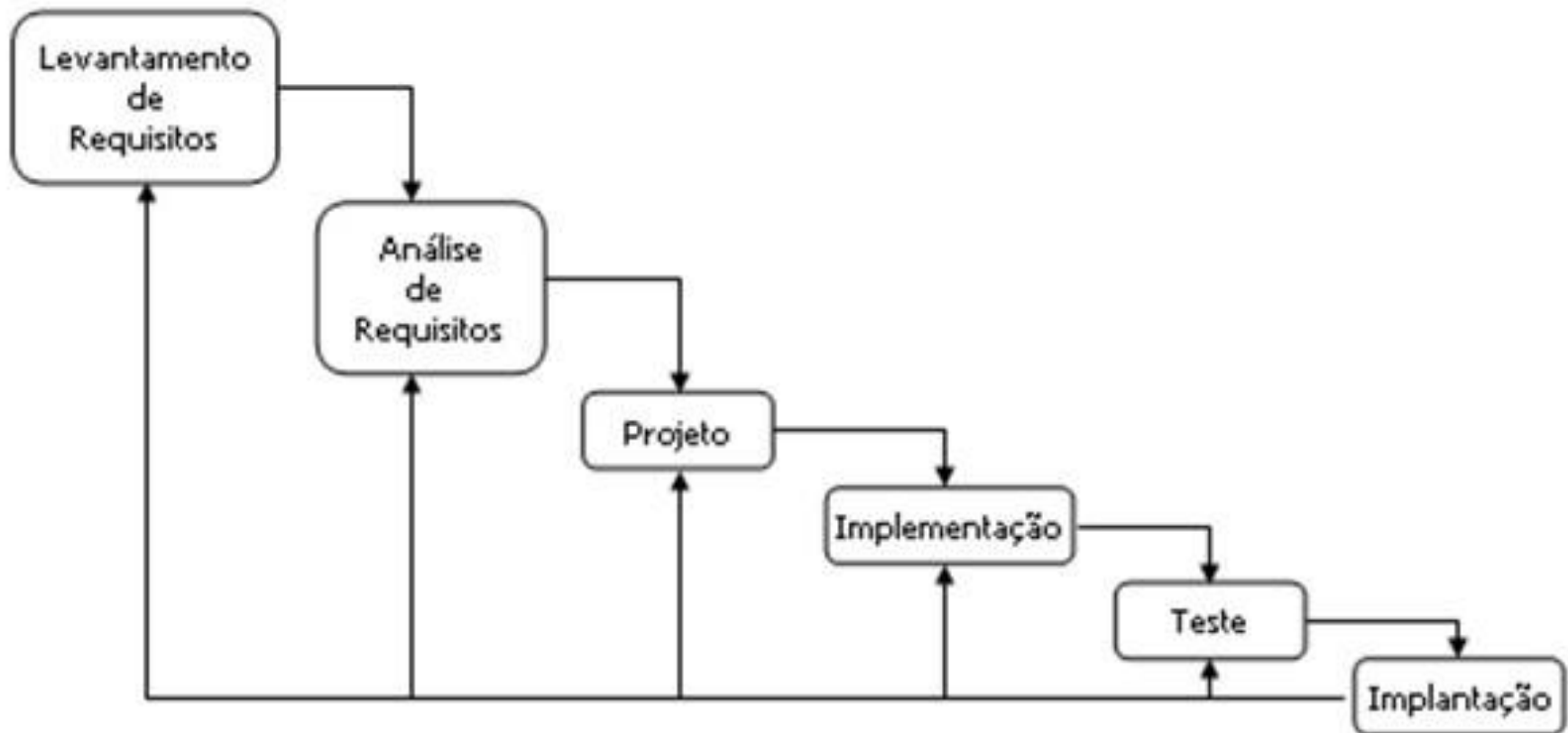
- O objetivo das metodologias de desenvolvimento de software é definir os pontos chaves de um projeto:
 - Quais atividades serão executadas
 - Quando cada atividade será executada
 - Quem executará cada tarefa
 - Definir pontos de controle para verificar andamento do desenvolvimento
 - Definir um padrão para o desenvolvimento

Desenvolvimento de Software

- Atividades típicas no desenvolvimento de software:
 - Levantamento de Requisitos
 - Análise
 - Projeto
 - Implementação
 - Testes
 - Implantação

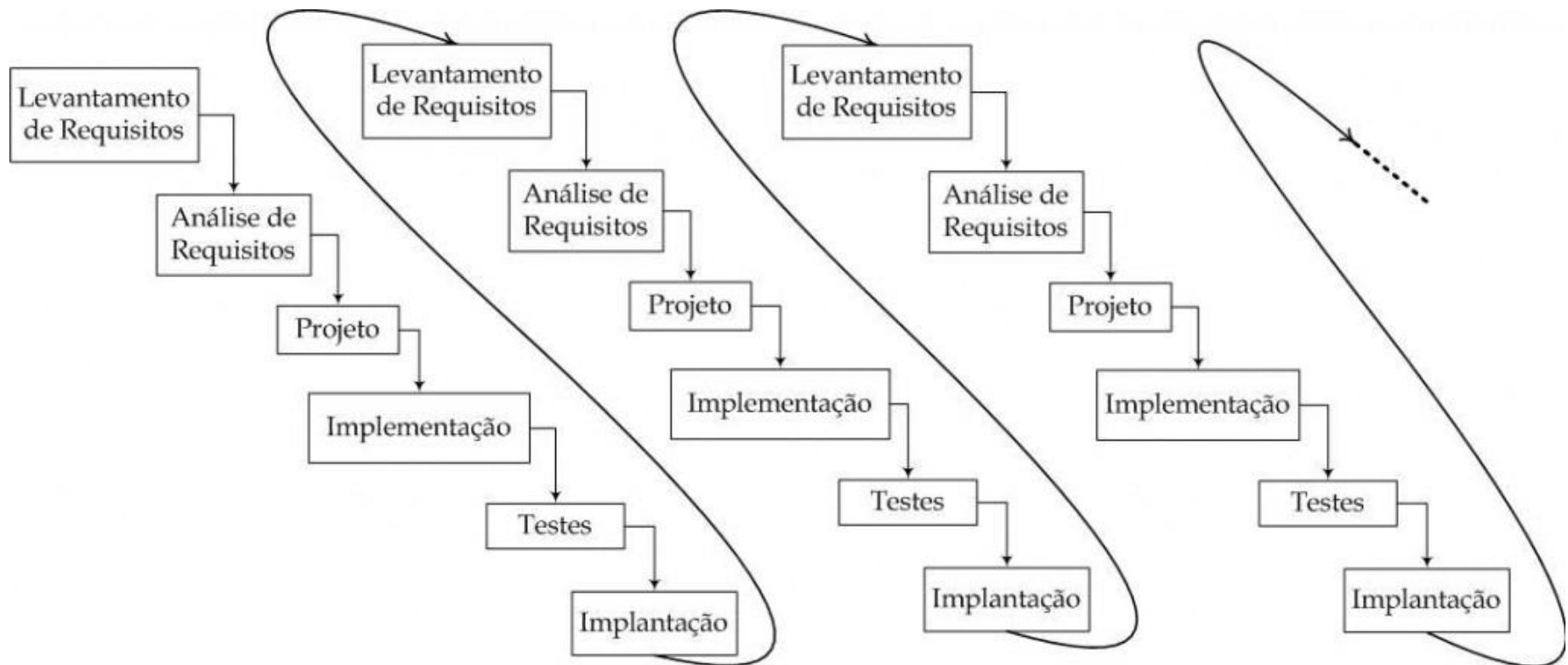
Metodologia de Desenvolvimento

- **Modelo Cascata**



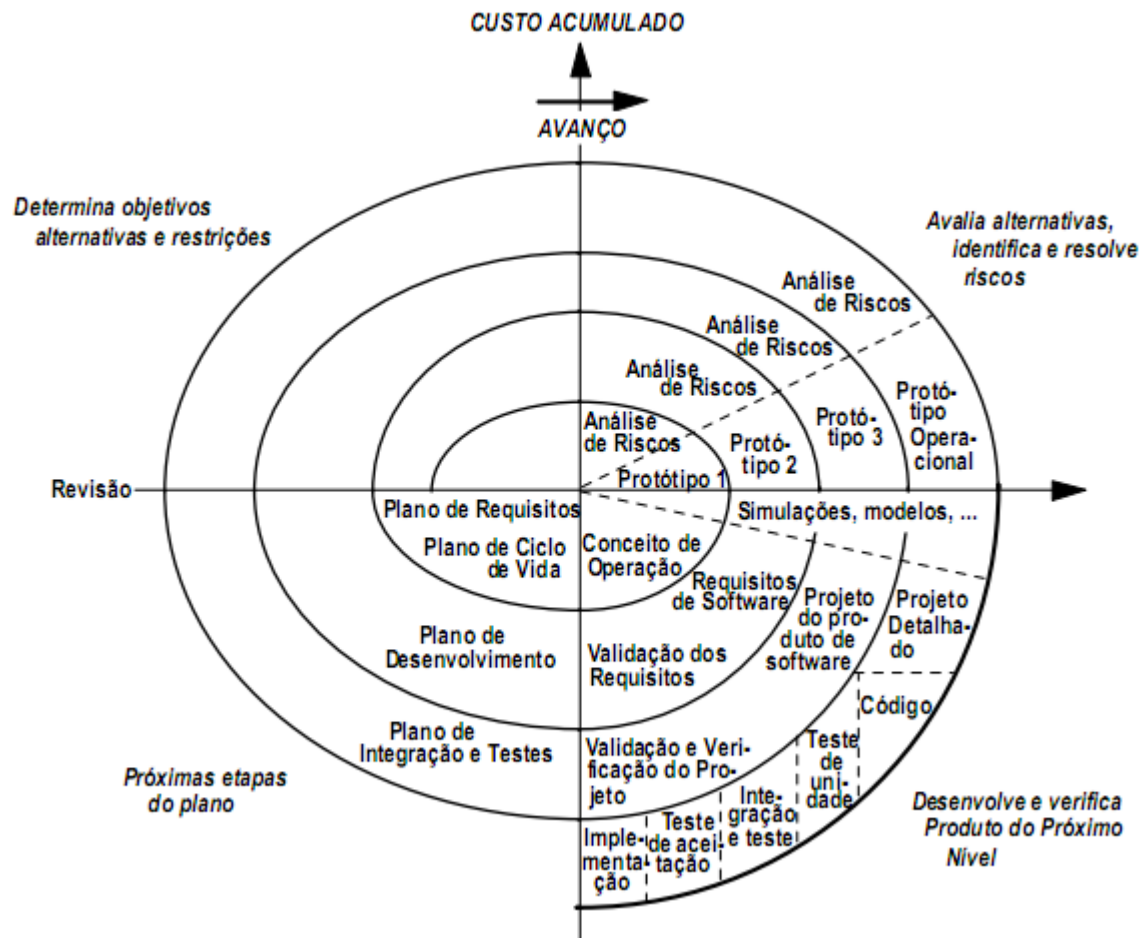
Metodologia de Desenvolvimento

- **Modelo Iterativo Incremental**

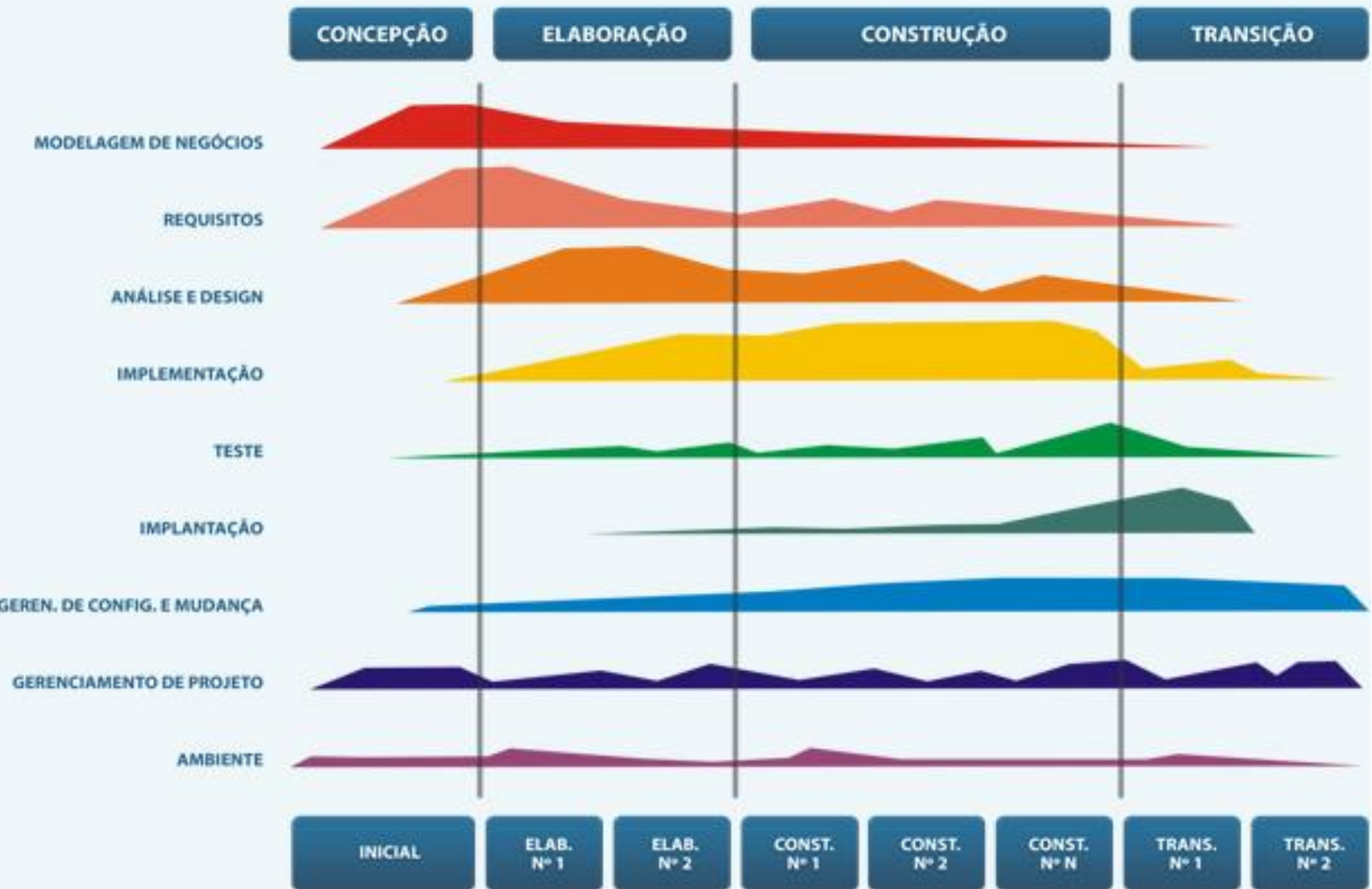


Metodologia de Desenvolvimento

- Modelo Espiral



FASES



ITERAÇÕES

Desenvolvimento de Software

Levantamento de Requisitos

- Compreensão do problema
- Ideia do usuário = Ideia dos desenvolvedores (ver próximo slide)
- Desenvolvedores e usuários discutem as necessidades dos futuros usuários do sistema a ser desenvolvido
- Necessidades = Requisitos
- “Requisito é uma condição ou capacidade que deve ser alcançada ou possuída por um sistema ou componente deste, para satisfazer um contrato, padrão, especificação ou outros documentos formalmente impostos.” (Maciaszek, 2000)

Desenvolvimento de Software



Como o cliente explicou



Como o lider de projeto entendeu



Como o analista planejou



Como o programador codificou



O que os beta testers receberam



Como o consultor de negocios descreveu



Valor que o cliente pagou



Como o projeto foi documentado



O que a assistencia tecnica instalou



Como foi suportado



Quando foi entregue



O que o cliente realmente necessitava

Desenvolvimento de Software

Levantamento de Requisitos

- Domínio do problema ou Domínio do negócio:
 - Parte do mundo real que é relevante para o sistema
 - Quais informações e processos precisam estar no sistema?
- Tipos de Requisitos:
 - Requisitos Funcionais
 - Requisitos Não-Funcionais
 - Requisitos Normativos

Desenvolvimento de Software

Levantamento de Requisitos

- Requisitos Funcionais
 - Definem funcionalidade do sistema
 - E.g. “O sistema deve permitir que cada professor realize o lançamento de notas das turmas nas quais lecionou.”
 - E.g. “Os coordenadores de escola devem poder obter o número de aprovações, reprovações e trancamentos em cada disciplina oferecida em um determinado período.”

Desenvolvimento de Software

Levantamento de Requisitos

- Requisitos Não-Funcionais
 - Definem características de qualidade que o sistema deve possuir
 - Podem estar associadas a funcionalidades
 - Subtipos de requisitos não-funcionais são:
 - Confiabilidade: medidas quantitativas de confiabilidade; e.g. tempo médio entre falhas, recuperação de falhas
 - Desempenho: definem tempos de resposta esperados para determinadas funcionalidades
 - Portabilidade: restrições ou exigências sobre hardware e/ou software
 - Segurança: propriedades de segurança (restrições de senha ou acesso)
 - Usabilidade: relacionados ao uso do software

Desenvolvimento de Software

Levantamento de Requisitos

- Requisitos Normativos
 - Declaração de restrições impostas sobre o desenvolvimento do sistema, que não se enquadram como requisitos não-funcionais
 - E.g. Adequações a custos e prazos;
componentes de hardware e software a serem adquiridos;
comunicação com outro sistema;
restrições de funcionamento da instituição;
restrições de valores;

Desenvolvimento de Software

Levantamento de Requisitos

- Produto resultante: Documento de Requisitos
 - Escrita simples; leitura fácil para técnicos e não-técnicos
 - **Não** contém informações sobre as soluções técnicas
 - É um termo de **consenso** entre equipe técnica e cliente(s)
 - É comum que os requisitos não sejam estáticos:
 - Surgimento de novas necessidades
 - Expectativas dos usuários - ao usar o software, os cliente descobrem requisitos que não tinham pensado.
- Muitos sistemas são abandonados ou tem um custo maior devido a utilização de pouco tempo para esta etapa

Desenvolvimento de Software

Análise de Requisitos

- Uma das primeiras etapas da análise de requisitos é identificar os “atores” => pessoas, dispositivos ou software que irá interagir com o sistema

Documentação dos atores

Aluno: Pessoa vinculada à instituição de ensino, por um período específico de tempo. Apta a fazer matrícula e participar das aulas que serão ministradas durante cada semestre letivo.

Professor: Pessoa contratada pela instituição de ensino para lecionar aulas durante cada semestre letivo.

Coordenador: Professor que tem como atividade principal controlar e gerenciar um curso específico de graduação.

Sistema Financeiro: Sistema utilizado pela instituição de ensino superior para gerenciar pagamentos e devoluções referentes a inscrições e matrículas de Alunos.

Desenvolvimento de Software

Análise de Requisitos

- Deve-se também registrar as “regras de negócio”, com:
 - **Nome e identificador:** um nome e identificador para facilitar a referência e a busca por regras de negócio
 - **Descrição:** a descrição textual da regra de negócio

Documentação das Regras de Negócio

	Quantidade de inscrições possíveis	[RN-01]
Descrição:	Um aluno não pode se inscrever em mais de seis disciplinas por semestre letivo.	
Fonte:	Coordenador da escola de informática	
Histórico:	Data de identificação:	12/07/2002
	Data de atualização:	-

Exemplo

Levantar os requisitos necessários para um sistema acadêmico que permite o controle e gerenciamento de matrícula, frequência e desempenho dos discentes e a organização das disciplinas ofertadas. O sistema acadêmico deverá permitir que os acadêmicos realizem suas matrículas nas turmas de disciplinas disponíveis, considerando restrições de pré-requisitos, número máximo de créditos (9) e limite de alunos por turma. Deverá permitir que chefes de departamento incluam novas disciplinas e novos professores, abram novas turmas para as disciplinas existentes com sala, horário, lotação máxima e professor definidos. As disciplinas só poderão ser ofertadas entre 7:30 e 12:00, e, 13:30 e 21:40, em blocos de 50 minutos por aula (hora-aula). Também deverá ser possível que professores acessem suas turmas e registrem frequência e notas para seus alunos.

Exemplo

O sistema deverá ter uma opção para finalizar o semestre, possibilitando a inclusão das notas de exame. Um aluno deverá ter frequência superior a 75% e deverá ter uma média superior a 3 para realizar exame. Caso sua nota seja maior ou igual a 7 está aprovado (desde que tenha a frequência necessária). Após a digitação das notas de exame o professor deverá finalizar a turma e o sistema mostrará o resultado final. O sistema deverá funcionar nos sistemas operacionais Windows e Linux e deverá ter seu acesso controlado por login e senha.

Atividade

Agora é a sua vez!

Levante os requisitos necessários para o sistema descrito no documento que está disponível na página da disciplina.

Documente também os atores e as regras de negócio do sistema descrito.

Bibliografia

- **Básica:**

BEZERRA, E. Princípios de Análise e Projetos de Sistemas com UML. Rio de Janeiro: Campus, 2003.

PRESSMAN, R.S. Engenharia de Software. São Paulo: Makron Books, 2002.

SOMMERVILLE, I. Engenharia de Software. São Paulo: Addison Wesley, 2003.

- **Complementar:**

WARNIER, J. Lógica de Construção de Programas. Rio de Janeiro: Campus, 1984.

JACKSON, M. Princípios de Projeto de Programas. Rio de Janeiro: Campus, 1988.

PAGE-JONES, M. Projeto Estruturado de Sistemas. São Paulo: McGraw-Hill, 1988.