

# Projeto de Sistemas de Software

Jordana S. Salamon

[jssalamon@inf.ufes.br](mailto:jssalamon@inf.ufes.br)

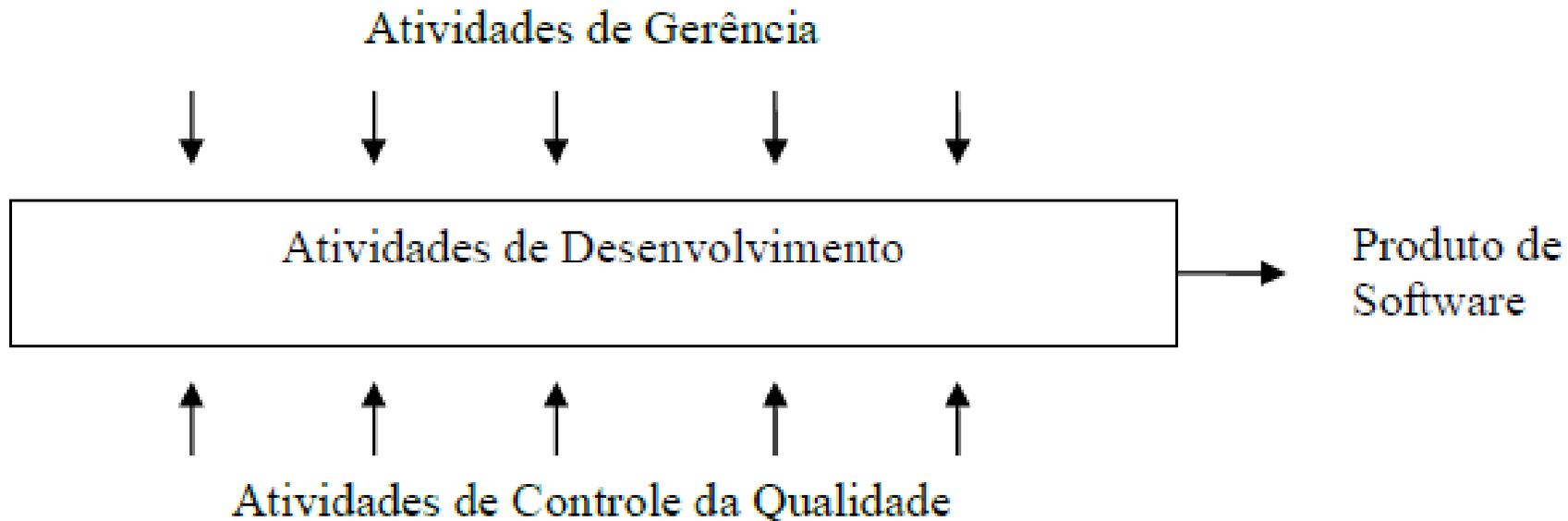
[jordanasalamon@gmail.com](mailto:jordanasalamon@gmail.com)

DEPARTAMENTO DE INFORMÁTICA  
CENTRO TECNOLÓGICO  
UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

# Revisão

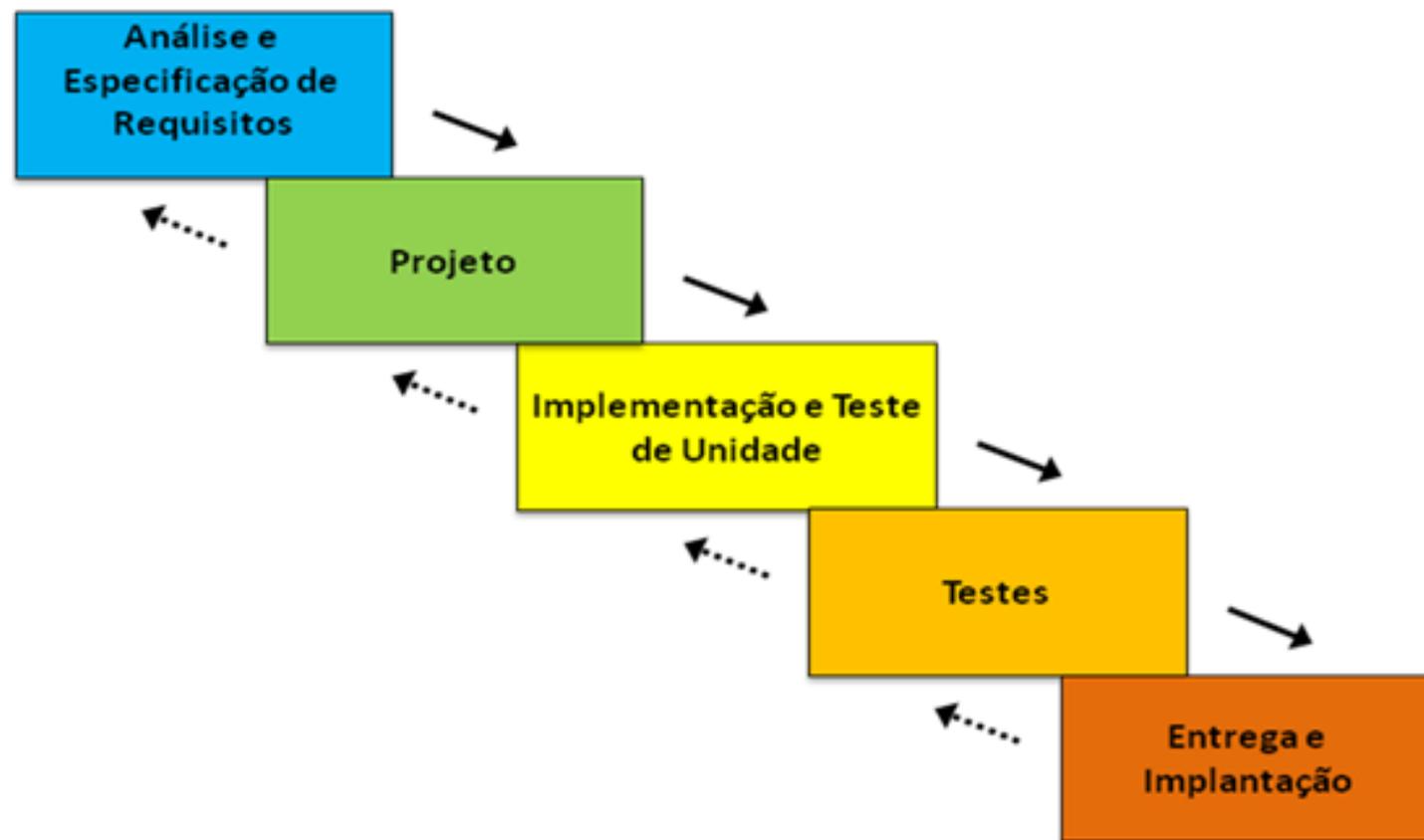
## ► Processo de Software

- Envolve atividades de Desenvolvimento, Gerência e Garantia da Qualidade:



# Revisão

## ► Atividades de Desenvolvimento



# Revisão



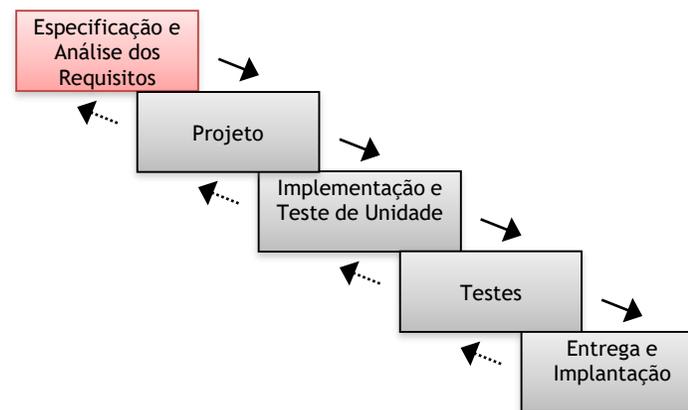
Que requisitos o software deve atender?



- Quem são os envolvidos?
- Quais são suas necessidades em relação ao software?

Mas...

**O que é um requisito?**



# Revisão

## Requisitos

Requisitos são descrições dos serviços que devem ser providos pelo sistema e de suas restrições operacionais (SOMMERVILLE, 2007).

Um requisito é uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar para atingir seus objetivos (PFLEEGER, 2004).

Um requisito é alguma coisa que o produto tem de fazer ou uma qualidade que ele precisa apresentar (ROBERTSON; ROBERTSON, 2006).



# Revisão

## Requisitos

Os requisitos de um sistema definem o que o sistema deve fazer e as circunstâncias sob as quais deve operar.

Ou, em outras palavras...

São as funções que um sistema deve incorporar e as restrições que devem ser satisfeitas.



# Revisão

## Requisitos

Uma das principais medidas do sucesso de um sistema de software é o grau no qual ele atende aos requisitos para os quais foi construído.

# Revisão

## Tipos de Requisitos

**Funcionais:** são declarações de serviços que o sistema deve prover, descrevendo o que o sistema deve fazer (SOMMERVILLE, 2007). Um requisito funcional descreve uma interação entre o sistema e o seu ambiente (PFLEEGER, 2004), podendo descrever, ainda, como o sistema deve reagir a entradas específicas, como o sistema deve se comportar em situações específicas e o que o sistema não deve fazer (SOMMERVILLE, 2007).

*Ex.: O sistema deve registrar locações, indicando o cliente, os itens locados, a data da locação, a data de devolução e o valor da locação.*



# Revisão

## Tipos de Requisitos

**Não Funcionais:** descrevem restrições sobre os serviços ou funções oferecidos pelo sistema (SOMMERVILLE, 2007), as quais limitam as opções para criar uma solução para o problema (PFLEEGER, 2004). Neste sentido, os requisitos não funcionais são muito importantes para a fase de projeto (design), servindo como base para a tomada de decisões nessa fase.

Ex.: A consulta ao acervo da locadora deve estar disponível pela Internet, a partir dos principais navegadores disponíveis no mercado. (requisito de portabilidade)



# Revisão

## Tipos de Requisitos

Os requisitos não funcionais têm origem nas necessidades dos usuários, em restrições de orçamento, em políticas organizacionais, em necessidades de interoperabilidade com outros sistemas de software ou hardware ou em fatores externos como regulamentos e legislações (SOMMERVILLE, 2007).

Assim, os requisitos não funcionais podem ser classificados quanto à sua origem.



# Revisão

## Tipos de Requisitos

Além desses requisitos, é importante considerar também *Requisitos de Domínio*.

*Requisitos de domínio (ou regras de negócio)* são provenientes do domínio de aplicação do sistema e refletem características e restrições desse domínio. Eles são derivados do negócio que o sistema se propõe a apoiar e podem restringir requisitos funcionais existentes ou estabelecer como cálculos específicos devem ser realizados, refletindo fundamentos do domínio de aplicação (SOMMERVILLE, 2011).

*Ex.: Em um sistema de matrícula de uma universidade, uma importante regra de negócio diz que um aluno só pode se matricular em uma turma de uma disciplina se ele tiver cumprido seus pré-requisitos.*



# Revisão

- ▶ Uma vez que requisitos de cliente e de sistema têm propósitos e público alvo diferentes, é útil descrevê-los em documentos diferentes.
- ▶ Existem dois documentos sugeridos pela literatura:
  - ▶ • **Documento de Definição de Requisitos:** deve ser escrito de maneira que o cliente possa entender, i.e., na forma de uma listagem do que o cliente espera que o sistema proposto faça. Ele representa um consenso entre o cliente e o desenvolvedor sobre o que o cliente quer.
  - ▶ • **Documento de Especificação de Requisitos:** refina os requisitos de cliente em termos mais técnicos, apropriados para o desenvolvimento de software, sendo produzido por analistas de requisitos.



# Introdução

- ▶ Com os requisitos pelo menos parcialmente capturados e especificados na forma de modelos, pode-se começar a trabalhar no domínio da solução.
- ▶ Muitas soluções são possíveis para o mesmo conjunto de requisitos e elas são intrinsecamente ligadas a uma dada plataforma de implementação (linguagem de programação, mecanismo de persistência a ser adotado, dispositivos onde o sistema deverá rodar e tipos de interface associados etc.).
- ▶ Em especial, requisitos não funcionais são, muitas vezes, conflitantes e a escolha de quais atributos de qualidade devem ser mais atentamente considerados tem também um forte impacto na escolha da solução.
- ▶ Assim, ao se considerar alternativas de solução, todos esses aspectos têm de ser levados em conta.



# Introdução

- ▶ A fase de projeto tem por objetivo definir e especificar uma solução a ser implementada.
- ▶ É uma fase de tomada de decisão, tendo em vista que muitas soluções são possíveis.
- ▶ Além disso, o projeto é um processo de refinamento.
- ▶ Partindo dos modelos conceituais do sistema, a fase de projeto inicia-se com o **projeto da arquitetura do sistema**, que visa descrever a estrutura de nível mais alto da aplicação, identificando seus principais elementos ou componentes e como eles se relacionam uns com os outros.
- ▶ Uma vez definida a arquitetura, o projeto passa a se concentrar no detalhamento classes no desenvolvimento orientado a objetos).

# Introdução

- ▶ Uma vez especificado o projeto dos elementos da arquitetura, pode dar-se início à implementação, quando as unidades de software do projeto detalhado são implementadas e testadas individualmente (teste de unidade).
- ▶ Gradativamente, os elementos vão sendo integrados e testados (teste de integração), até se obter o sistema, quando o todo deve ser testado (teste de sistema).



# Introdução

- ▶ Por fim, uma vez testado no ambiente de desenvolvimento, o software pode ser colocado em produção.
- ▶ Usuários devem ser treinados, o ambiente de produção deve ser configurado e o sistema deve ser instalado e testado, agora pelos usuários no ambiente de produção (testes de homologação ou aceitação).
- ▶ Caso o software demonstre prover as capacidades requeridas, ele pode ser aceito e a operação iniciada.

# A Fase de Projeto

- ▶ O objetivo da fase de projeto (ou design) é produzir uma solução para o problema identificado e modelado nas fases de levantamento e análise de requisitos, **incorporando a tecnologia aos requisitos e projetando o que será construído na implementação.**
- ▶ Sendo assim, é necessário conhecer a tecnologia disponível e os ambientes de hardware e software onde o sistema será desenvolvido e implantado.
- ▶ Durante o projeto, deve-se decidir como o problema será resolvido, começando em um alto nível de abstração, próximo da análise, e progredindo sucessivamente para níveis mais detalhados até se chegar a um nível de abstração próximo da implementação.

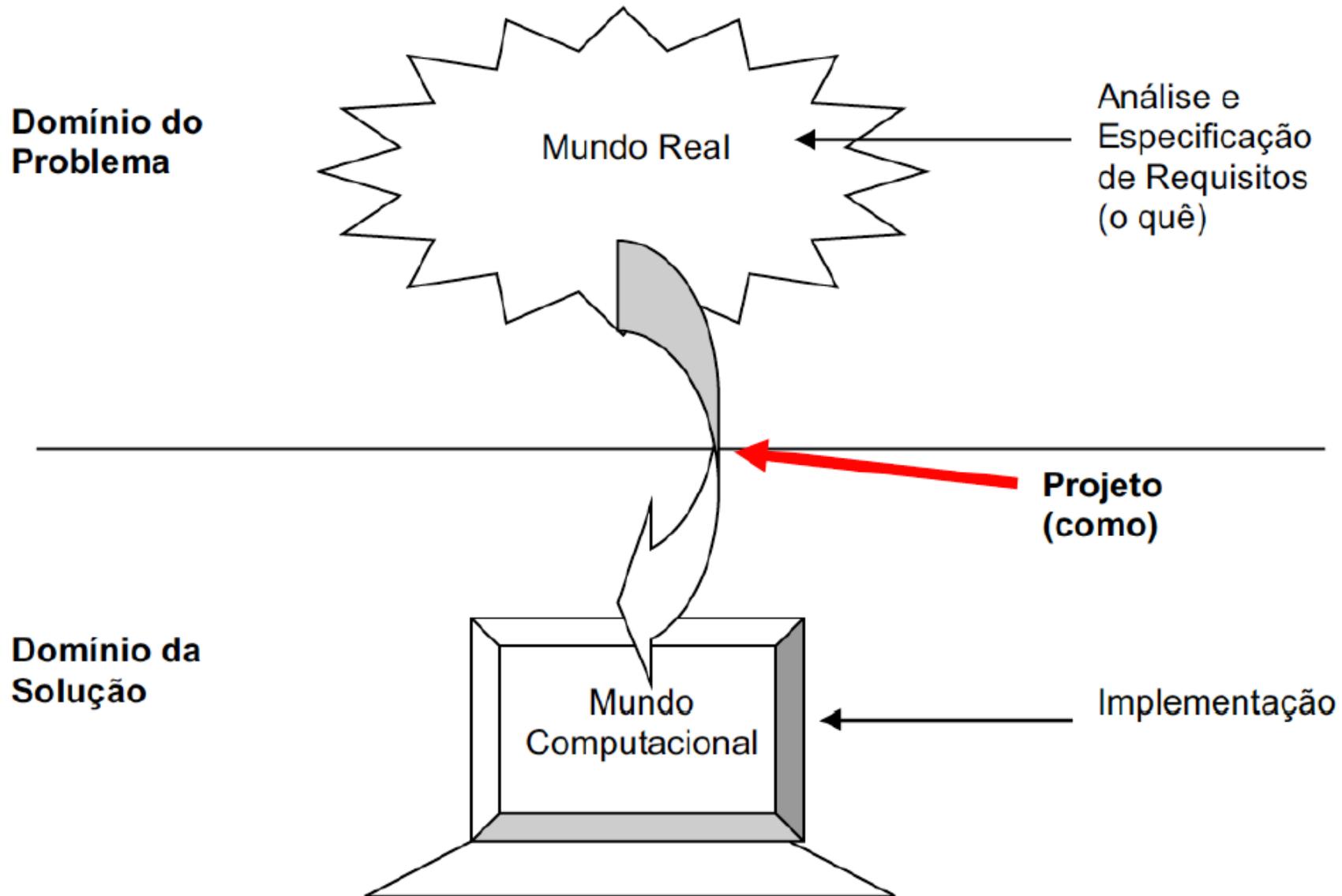


# A Fase de Projeto

- ▶ Corresponde à primeira atividade que leva em conta aspectos tecnológicos.
- ▶ Enquanto a fase de análise pressupõe que a tecnologia é perfeita (capacidade ilimitada de processamento, com velocidade instantânea, capacidade ilimitada de armazenamento, custo zero e não passível de falha), a fase de projeto envolve a modelagem de como o sistema será implementado, agora considerando também os requisitos não funcionais de caráter tecnológico.



# A Fase de Projeto



# A Fase de Projeto

- ▶ Independentemente do paradigma adotado, o processo de projeto envolve as seguintes atividades:
  - **Projeto da Arquitetura do Software:** visa definir os elementos estruturais do software e seus relacionamentos.
  - **Projeto dos Elementos da Arquitetura:** visa projetar em um maior nível de detalhes cada um dos elementos estruturais definidos na arquitetura, o que envolve a decomposição de módulos em outros módulos menores.



# A Fase de Projeto

- **Projeto Detalhado:** tem por objetivo refinar e detalhar os elementos mais básicos da arquitetura do software, i.e., as interfaces, os procedimentos e as estruturas de dados.

Deve-se descrever como se dará a comunicação entre os elementos da arquitetura (interfaces internas), a comunicação do sistema em desenvolvimento com outros sistemas (interfaces externas) e com as pessoas que vão utilizá-lo (interface com o usuário), bem como se devem projetar detalhes de algoritmos e estruturas de dados.

# A Fase de Projeto

- ▶ Considerando a classe de sistemas de informação, de maneira geral, os seguintes elementos estão presentes na arquitetura de um sistema:
- ▶ • **Lógica de Negócio:** é o elemento da arquitetura que trata da lógica de negócio apoiada pelo sistema, englobando tanto aspectos estruturais (classes de domínio derivadas dos modelos conceituais estruturais da fase de análise), quanto comportamentais (classes de lógica de aplicação, que tratam das funcionalidades descritas pelos casos de uso).
- ▶ • **Interface com o Usuário:** é o elemento da arquitetura que trata da interação humano-computador. Envolve tanto as interfaces propriamente ditas (objetos gráficos responsáveis por receber dados e comandos do usuário e apresentar resultados), quanto o controle da interação, abrindo e fechando janelas, habilitando ou desabilitando botões etc.
- ▶ • **Persistência:** é o elemento da arquitetura responsável pelo armazenamento e recuperação de dados em memória secundária (classes que representam e isolam os depósitos de dados do restante do sistema).



That's all Folks!



nemo