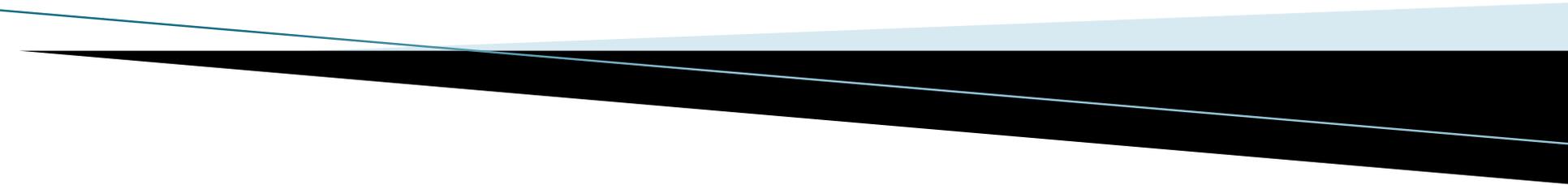


ANÁLISE DE SISTEMAS (processo de desenvolvimento de sistemas)

por

Antônio Maurício Pitangueira



Um Pouco de Estatística...

- Um estudo com milhares de projetos de software mostrou que

- 25% dos requisitos de um sistema mudam ao longo do tempo
- Grandes projetos: 35% a 50%!
- Ciclo Cascata:
 - o 45% de coisas não usadas!
 - o 400% de requisitos alterados até o final do projeto!

- o Conclusão:

- Desenvolvimento de software é um domínio de grandes mudanças e de alta instabilidade!!

▶ Levantamento de requisitos

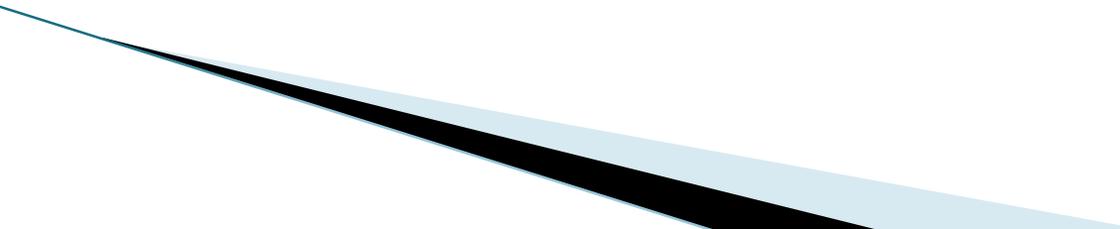
▶ Análise de requisitos

▶ Projeto

▶ Implementação

▶ Testes

▶ Implantação

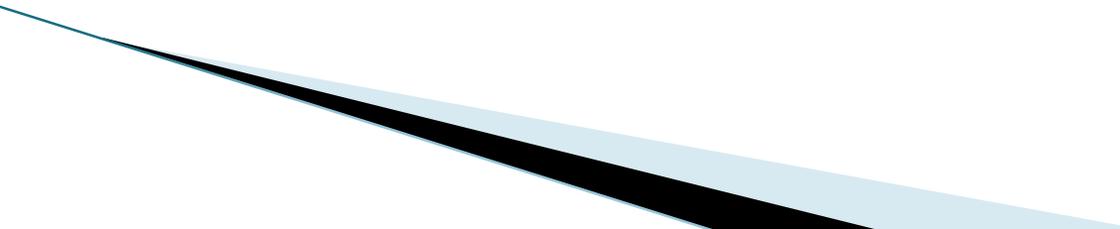
- ▶ Corresponde à etapa de compreensão do problema aplicada ao desenvolvimento de software
 - ▶ O principal objetivo do levantamento de requisitos
 - ▶ É que usuários e desenvolvedores tenha a mesma visão do problema a ser resolvido
 - ▶ Nessa etapa, os desenvolvedores, juntamente com os clientes, tentam levantar e definir as necessidades dos futuros usuários do sistema a ser desenvolvido.
- 

Levantamento de requisitos

- ▶ Formalmente, um 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.
- ▶ Durante o levantamento de requisitos, a equipe de desenvolvimento tenta entender o domínio que deve ser automatizado pelo software

Levantamento de requisitos

- ▶ O levantamento de requisitos compreende também um estudo exploratório das necessidades dos usuários e da situação do sistema atual(se existir)
- ▶ Possíveis técnicas:
 - Leitura de obras de referência;
 - Observação do ambiente do usuário;
 - Realização de entrevistas com o usuário;
 - Entrevistas com especialistas do domínio;
 - Reutilização de análises anteriores;

- ▶ Requisitos funcionais: definem as funcionalidades do sistema.
 - Ex: “ o sistema deve permitir que cada professor realize o lançamento de notas das turmas nas quais lecionou”
 - “O sistema deve permitir que um aluno realize a sua matrícula nas disciplinas oferecidas em um semestre letivo”
 - “os coordenadores de escola devem poder obter o número de aprovações, reprovações e trancamentos em cada disciplina oferecida em um determinado período”
- 

- ▶ Requisitos não-funcionais: declaram as características de qualidade que o sistema deve possuir e que estão relacionadas às suas funcionalidades. Alguns tipos:
 - Confiabilidade: corresponde a medidas quantitativas da confiabilidade do sistema, tais como tempo médio entre falhas, recuperação de falhas ou quantidade de erros por milhares de linhas de código-fonte;
 - Desempenho: requisitos que definem tempos de resposta esperados para as funcionalidades do sistema
 - Portabilidade: restrições sobre as plataformas de hardware e de software nas quais o sistema será implantado e sobre o grau de facilidade para transportar o sistema para outras plataformas
- 

- Segurança: limitações sobre a segurança do sistema em relação acessos não-autorizados;
 - Usabilidade: requisitos que se relacionam ou afetam a usabilidade do sistema. Exemplos incluem requisitos sobre a facilidade de uso e a necessidade ou não de treinamento dos usuários.
- 

- ▶ Requisitos normativos: declarações de restrições impostas sobre o desenvolvimento do sistema.
 - Ex: restrições que definem a adequação a custos e prazos, a plataforma tecnológica, aspectos legais, componentes de hw e se a serem adquiridos,...
- 

2- Análise

- ▶ Formalmente, o termo análise corresponde a “quebrar” um sistema em seus componentes e estudar como tais componentes interagem com o objetivo de entender como esse sistema funciona;
- ▶ No nosso contexto, esta é a fase em que os analistas realizam um estudo detalhado dos requisitos levantados na atividade anterior. **A partir desse estudo, são construídos modelos para representar o sistema a ser construído**

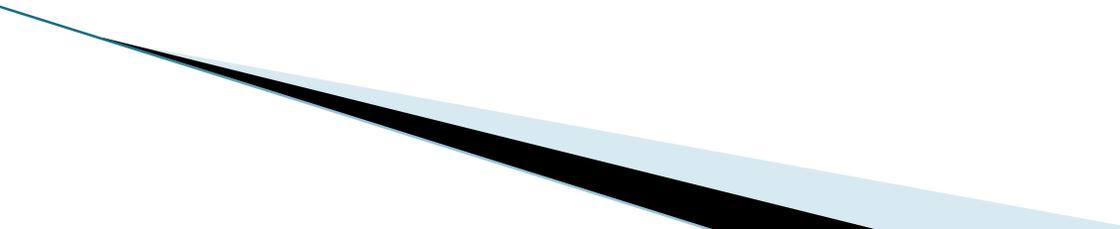
2- Análise

- ▶ Nessa atividade, o foco de interesse é tentar construir uma estratégia de solução sem se preocupar com a maneira *como* essa estratégia será realizada.
- ▶ Em outras palavras, é necessário saber o que o sistema proposto deve fazer para, então, definir como esse sistema irá fazê-lo.
- ▶ **OBS: os modelos construídos na fase de análise devem ser cuidadosamente validados e verificados.**

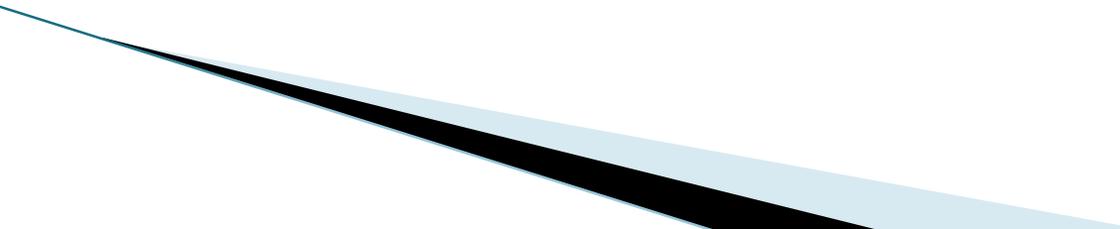
2- Análise

- ▶ O objetivo da validação é assegurar que as necessidades do cliente estão sendo atendidas pelo sistema: será que o software correto está sendo construído?
- ▶ Quando um usuário valida um modelo, quer dizer que entendeu o modelo construído e que, segundo esse entendimento, o modelo reflete suas necessidades com relação ao sistema a ser desenvolvido.

2- Análise

- ▶ Já a verificação tem o objetivo de analisar se os modelos construídos estão em conformidade com os requisitos definidos: será que o sw está sendo construído corretamente?
 - ▶ OBS: **diferentemente da validação(que é uma atividade de análise), a verificação é uma etapa típica da fase de projeto)**
- 

2- Análise

- ▶ Em um processo de desenvolvimento OO, um dos resultados da análise é o modelo de objetos que representa as classes componentes do sistema..
 - ▶ A fase de análise pode ser subdividida em **análise do domínio** (ou análise do negócio) e **análise da aplicação**
- 

2- Análise

▶ ANÁLISE DO DOMÍNIO

- Primeiro objetivo: identificar e modelar os objetos do mundo real que, de alguma forma, serão processados pela aplicação em desenvolvimento.
- Ex: um aluno é um objeto do mundo real que um sistema de controle acadêmico deve processar.
 - assim, um aluno é um objeto do domínio.
 - Cite outros objetos para este domínio..
- Outro objetivo é identificar as regras do negócio e os processos do negócio realizados pela organização.

2- Análise

▶ ANÁLISE DA APLICAÇÃO

- Tem como objetivo identificar objeto de análise que normalmente não fazem sentido para os especialistas do domínio, mas que são necessários para suprir as funcionalidades do sistema em questão.
- Esses objetos têm a ver com os aspectos computacionais de alto nível da aplicação. Ex: uma tela de inscrição em disciplinas.]
- De uma forma geral, qualquer objeto necessário para que o sistema em desenvolvimento possa se comunicar com seu ambiente é um objeto da aplicação.

2- Análise

- ▶ Sendo assim, a análise de domínio identifica objetos do domínio e a análise da aplicação identifica objetos da aplicação.
- ▶ Por que isto?
- ▶ Principais ferramentas(diagramas de casos de uso e de classes)
- ▶ Outros também utilizados: diagrama de interação, de estados e de atividades
- ▶ **Em suma, o foco principal da análise são os aspectos lógicos e independentes de implementação de um sistema(os requisitos)**



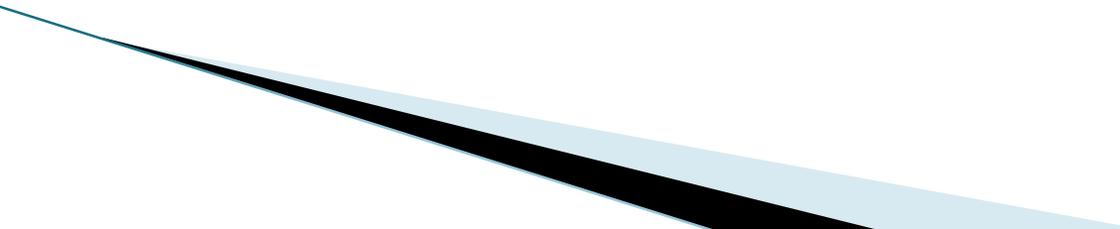
- ▶ Na fase de projeto, determina-se “**como**” o sistema funcionará para atender aos requisitos, de acordo com os recursos tecnológicos existentes (a fase de projeto considera os aspectos físicos e dependentes de implementação)
- ▶ Então, aos modelos construídos na análise são adicionadas as denominadas “restrições de tecnologia”.

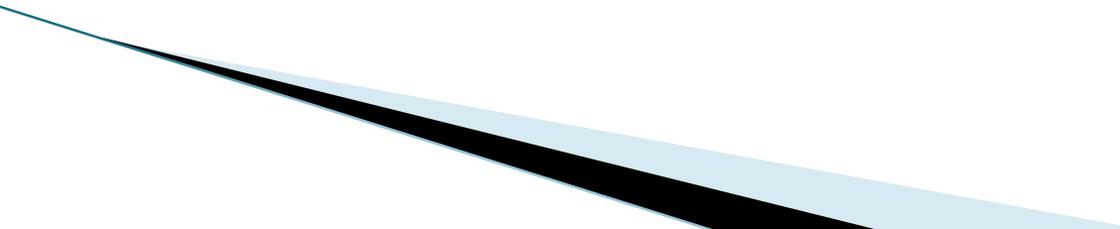


- ▶ Exemplos de aspectos a serem considerados na fase de projeto: arquitetura do sistema, padrão de interface gráfica, a linguagem de programação, o gerenciador de banco de dados, etc.
- ▶ Esta fase produz uma descrição computacional do que o software deve fazer e deve ser coerente com a descrição feita na análise.



- ▶ A FASE DE PROJETO É DIRECIONADA PELOS MODELOS CONSTRUÍDOS NA FASE DE ANÁLISE E PELO PLANEJAMENTO DO SISTEMA.
- ▶ O projeto consiste em duas atividades principais:
 - A) projeto da arquitetura
 - B) projeto detalhado

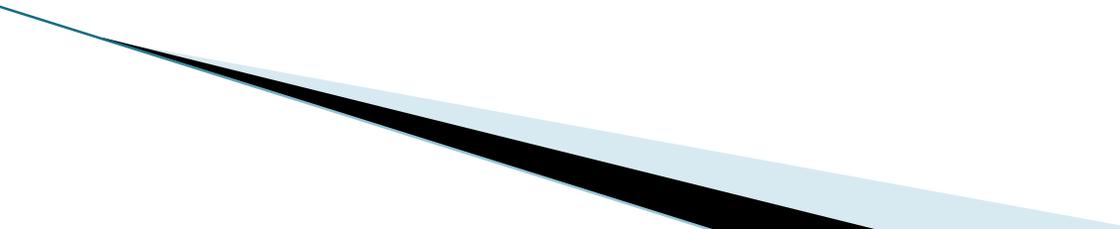
- ▶ Consiste em distribuir as classes de objetos relacionadas do sistema em subsistemas e seus componentes.
 - ▶ Consiste também em distribuir esses componentes fisicamente pelos recursos de Hw disponíveis.
 - ▶ Diagramas de implementação(UML) são comumente utilizados nesta fase
- 

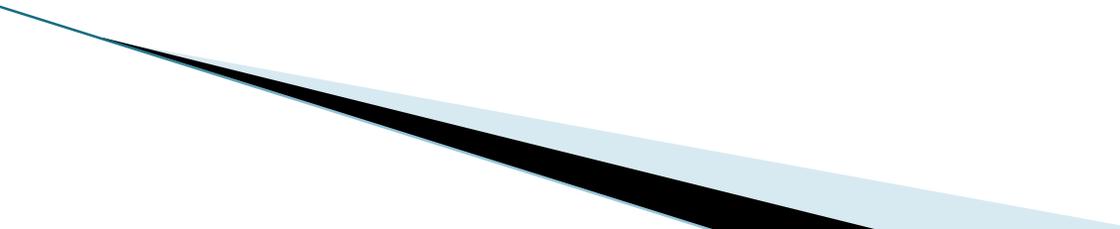
- ▶ Aqui são modeladas as colaborações entre os objetos de cada módulo com o objetivo de realizar as funcionalidades do módulo.
 - ▶ Também são realizados o projeto da interface com o usuário e o projeto de banco de dados.
 - ▶ Aspectos de mapeamento dos modelos de análise para artefatos de sw são também considerados.
 - ▶ Projeto dos algoritmos
- 

ANÁLISE

- ▶ Enfatiza a investigação de um problema e os requisitos, ao invés da solução.
 - Ex: Se um novo sistema é desejado, como ele será usado? Quais são suas Funcionalidades?
- ▶ -Análise OO:
 - Investigação de quais são os objetos do domínio. Há uma ênfase em encontrar e descrever os objetos ou conceitos do domínio do problema.

PROJETO

- ▶ Enfatiza uma solução conceitual (em software ou hardware) que satisfaz os requisitos, ao invés de sua implementação.
 - ▶ Ênfase na definição de objetos e como eles colaboram para atender um requisito.
 - ▶ O projeto ao final pode ser implementado.
- 

- ▶ Nesta fase, o sistema é codificado, ou seja, ocorre a tradução da descrição computacional obtida na fase de projeto em código executável mediante o uso de uma ou mais linguagens.
 - ▶ A implementação pode reutilizar os componentes de software, bibliotecas de classe e frameworks no produto de software
- 

5 - Testes

- ▶ Verificação do sistema construído, levando-se em conta a especificação feita na fase de projeto.
 - Produto desta fase: relatório de testes com informações sobre erros detectados no software
- ▶ Após a atividade de testes, os diversos módulos do sistema são integrados, resultando finalmente no produto de sw.

- ▶ O sistema é :
 - Empacotado
 - Distribuído
 - Instalado no ambiente do usuário.

 - ▶ Manuais são escritos, arquivos são carregados e os dados são importados para o sistema.

 - ▶ Usuários são treinados
- 