

FACULDADE DE COMPUTAÇÃO E INFORMÁTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Projeto de Software – Aula 1 – 2º SEMESTRE/2018

Análise, Projeto e Desenvolvimento de Sistemas de Software – Modelagem

Objetivos:

- discutir o contexto da disciplina
 - conhecer as fases do desenvolvimento de software
 - aprender sobre a importância da análise e modelagem de sistemas de software
-

Nas disciplinas dos semestre anteriores foi discutido que em consequência do crescimento da importância da informação, surgiu a necessidade de processar, transformar, armazenar e gerenciar as informações de forma adequada e eficiente, assim, dessa necessidade, surgiram os sistemas de informação.

Para Laudon e Laudon (2011) um sistema de informação pode ser definido como um conjunto de componentes inter-relacionados trabalhando juntos para coletar, recuperar, processar, armazenar e distribuir informações com a finalidade de facilitar o planejamento, o controle, a coordenação, a análise e o processo decisório em organizações.

Pode-se perceber que um sistema de informação é uma combinação de pessoas, dados, processos, interfaces entre sistemas, redes de comunicação e tecnologia que interagem com o objetivo de fornecer suporte e melhorar o processo de negócio de uma organização com relação às informações que nela fluem.

Uma característica de sistemas de informação é a complexidade. Desta forma, o sistema de software envolvido também pode ser menos ou mais complexo e também, podem envolver menos ou mais pessoas no seu desenvolvimento e levar mais ou menos tempo para sua construção (desenvolvimento).

A formação necessária para um profissional atuar na indústria de software vai além da formação tecnológica, deve fazer parte desta formação disciplinas de negócio, gestão, estratégia e relações humanas e empresariais, uma vez que, este profissional deverá atuar em um mercado em que se busca obter, através da TI, vantagens competitivas por meio de uma gestão mais eficiente de suas operações e de descobrir meios de colaborar com parceiros comerciais e indústrias para fornecer produtos e serviços para o mercado de forma lucrativa.

Para lidar com a complexidade e traduzir necessidades de negócio e viabilizar estratégias corporativas, desenvolver novas formas de viabilizar produtos e serviços, e até mesmo entreter as pessoas; os profissionais da área de desenvolvimento de software utilizam de métodos, técnicas e ferramentas para lidar com desafios do mundo real em soluções de software.

Nesta disciplina serão estudados métodos, técnicas e ferramentas para viabilizar a comunicação e entendimento uniforme entre os diferentes envolvidos durante as etapas do processo de desenvolvimento de software, que são: concepção do sistema, análise, projeto, implementação e testes, como mostra a figura 1.

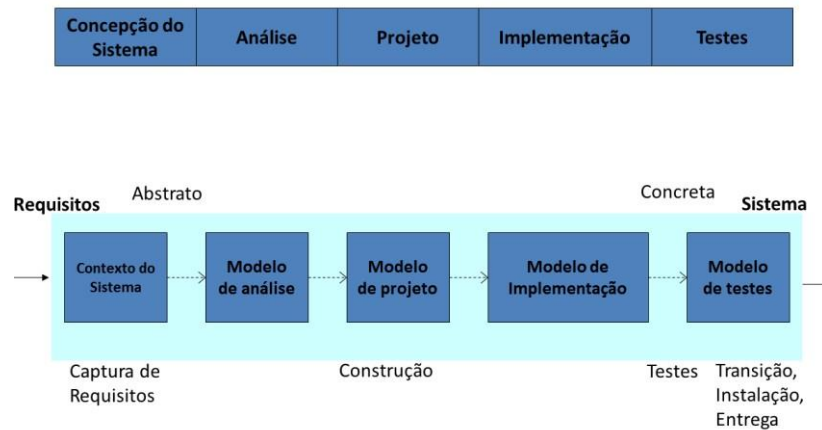


Figura 1: Etapas do processo de desenvolvimento.

Para isso, a **Modelagem de Sistemas de Software** permitirá através da utilização de notação gráfica e textuais irá permitir traduzir problemas do mundo real em modelos que representaram as partes essenciais de um sistema, considerando as várias perspectivas diferentes e complementares. A necessidade de construir modelos do sistema tem o objetivo de:

- Gerenciar a complexidade envolvida
- Possibilitar a comunicação e o entendimento entre as pessoas envolvidas
- Reduzir dos custos no desenvolvimento
- Prever do comportamento futuro do sistema

Afinal,

“Uma figura vale por mil palavras”

Vocês concordam?

Abstração

É o exame seletivo de certos aspectos de um problema.

“A abstração é o modo fundamental que nós humanos lidamos com a complexidade.”
(Booch, 2007)

Observem algumas definições e comentários relevantes sobre abstração:

- Dahl, Dijkstra e Hoare (1972) argumenta que a abstração surge a partir de um reconhecimento de semelhanças entre certos objetos, situações ou processos no mundo real, e a decisão de concentrar-se sobre essas semelhanças e ignorar ao mesmo tempo as diferenças;

- Shaw (1989) define a abstração como uma descrição ou especificação simplificada de um sistema que enfatiza alguns dos detalhes ou propriedades do sistema, enquanto suprime outros. Uma boa abstração é aquela que enfatiza detalhes que são importantes para o leitor ou usuário e suprime detalhes que não são, pelo menos para o momento, , são irrelevantes;
- Berzins, Gray e Naumann (1986) definiram que um conceito é qualificado como uma abstração somente se ele pode ser descrito, entendido, e analisado independentemente do mecanismo que eventualmente será utilizado para realiza-lo;
- Booch (2007) argumenta que uma abstração denota as características essenciais de um objeto que o distinguem de todos os outros tipos de objetos e, assim, proporcionar nitidez definindo os limites conceituais, em relação à perspectiva do espectador.



Abstração foca sobre as características essenciais de algum objeto, relativo a perspectiva do espectador (BOOCH, 2007).

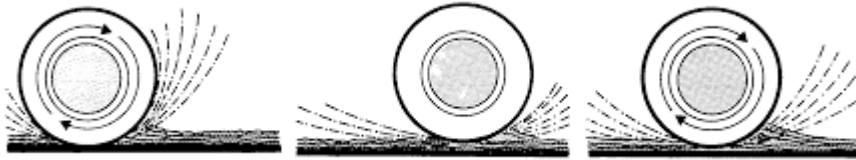
Então vamos aplicar o princípio da abstração?

Algumas características de um pneu de automóvel são seu tamanho, tipo de material, construção interna (aço trançado, por exemplo), desenho dos frisos, custo, tempo de vida esperado e peso e outros.

- 1) Quais são as características que são importantes para você decidir se deve ou não comprar um pneu para o seu carro?



- 2) Quais poderiam ser relevantes para uma equipe de engenheiros está desenvolvendo simulação de desempenho de novo projeto de pneus antiderrapantes para carros de corrida – Stock Car?



- 3) Quais são importantes para alguém que usa para construir balanços para crianças?



Referências bibliográficas

Berzins, V., Gray, M., and Naumann, D. May 1986. Abstraction-Based Software Development. Communications of the ACM vol. 29(5), p. 403.

BOOCH, G. Object-oriented analysis and design with applications. 3ª.ed. Addison-Wesley, 2007.

Dahl, O., Dijkstra, E., and Hoare, C. A. R. 1972. Structured Programming. London, England: Academic Press, p. 83.

LAUDON, J., LAUDON, K. Sistemas de Informação Gerenciais, 9ª ed. São Paulo: Peterson, 2011.

Shaw, M. 1989. Larger Scale Systems Require Higher-Level Abstractions. Proceedings of the Fifth International Workshop on Software Specification and Design. IEEE Computer Society.