

Introdução à Engenharia de *Software*

Sistema de informação

- Os dados têm, como origem, diversas fontes.
- O processamento de dados (transformação de dados em informação) é realizado por diversas partes que se relacionam em maior ou menor grau.
- A este conjunto de partes dá-se o nome de sistema de informação.

Sistema de informação

Um sistema de informação é responsável por:

- estruturar os dados;
- transformar os dados;
- distribuir a informação.

Sistemas de informação

- Os sistemas de informação são mais antigos que o computador.
- Então, o uso de sistemas de informação não está obrigatoriamente atrelado ao uso de *softwares*.

- Produzir *software* não é algo simples.
- O *software*, no fim das contas, é a tradução de um processo de negócio realizado pelo ser humano para o computador.

- Para dificultar mais ainda, esses processos de negócio mudam constantemente devido a diversos fatores:
 - políticos (internos e externos);
 - econômicos;
 - tecnológicos.

- O ser humano não começou a desenvolver *software* a tão pouco tempo.
- Entre acertos e falhas, descobriu processos e ferramentas que o auxiliam a diminuir as falhas no desenvolvimento de *software*.
- A este arcabouço teórico e prático dá-se o nome de **Engenharia de Software**.

Engenharia de *Software*

- A Engenharia de *Software* foi inicialmente proposta em 1968 em uma Conferência.
- O desenvolvimento de *software* era realizado de maneira bastante precária do ponto de vista da organização.
- Não havia um conjunto de boas práticas e organização.
- Isto levou à alta taxa de falha nos projetos de desenvolvimento de *software*, além da entrega de produtos com qualidade aquém do esperado.

- Para tentar resolver este problema, a Engenharia de *Software* surge como uma disciplina que estuda, de maneira sistemática e organizada, os seguintes aspectos do desenvolvimento de *software*:
 - atividades;
 - métodos;
 - ferramentas;
 - teorias;
 - técnicas

Engenharia de *Software*

- Um ponto importante a ser salientado é que a Engenharia de *Software* não estabelece uma resposta única para todos os problemas.
- Um *software* para *e-commerce* não deve ser desenvolvido da mesma maneira que um sistema embarcado para aviões.
- Apesar de haver características semelhantes (afinal, é tudo *software*), a natureza de cada *software*, seu contexto de uso, o tempo de desenvolvimento, seus custos associados, além de outros fatores são o que definirá as atividades, métodos, ferramentas e técnicas a serem utilizadas pela equipe de desenvolvimento.
- Em outras palavras, não há bala de prata.

Processo de *software* × Produto de *software*

- Outro ponto a ser observado é se o processo de negócio já existe ou não.
- Nos primórdios do desenvolvimento de *software*, os processos de negócio já existiam nas empresas e nos governos e eram traduzidos para *software*.
- Este tipo de atividade gera dependência da maturidade e a definição do processo de negócio.
- Mais do que isso, gera a necessidade de um processo de desenvolvimento de *software* bastante formal.

Processo de *software* × Produto de *software*

- Vem ganhando força nas duas últimas décadas o foco no produto de *software* e não no processo de *software*.
- A criação de *software* é não atrelada obrigatoriamente a necessidades já existentes, mas a oportunidades.
- As redes sociais são um exemplo disso. Elas surgiram por observação de uma oportunidade e não por um processo de negócio já existente.

Processo de *software* × Produto de *software*

- Produto de *software* não é algo novo ou revolucionário.
- *Softwares* de prateleira, como soluções para comércio ou *softwares* de escritório, já existem há muito tempo.
- Entretanto, esses *softwares* de prateleira e os *softwares* desta nova geração não podem ser colocados num mesmo grupo.

Referências

- Sommerville, Ian. Software Engineering - Global Edition. 10ed. 2016. Pearson Education.
- Sommerville, Ian. Engineering Software Products: An Introduction to Modern Software Engineering. 1ed. 2021. Pearson Education.