

Engenharia de Software

Aula1: Introdução

Dra. Ana Patrícia F. Magalhães Mascarenhas
anapatriciamagalhaes@gmail.com

PLANO DE AULA

► Objetivo

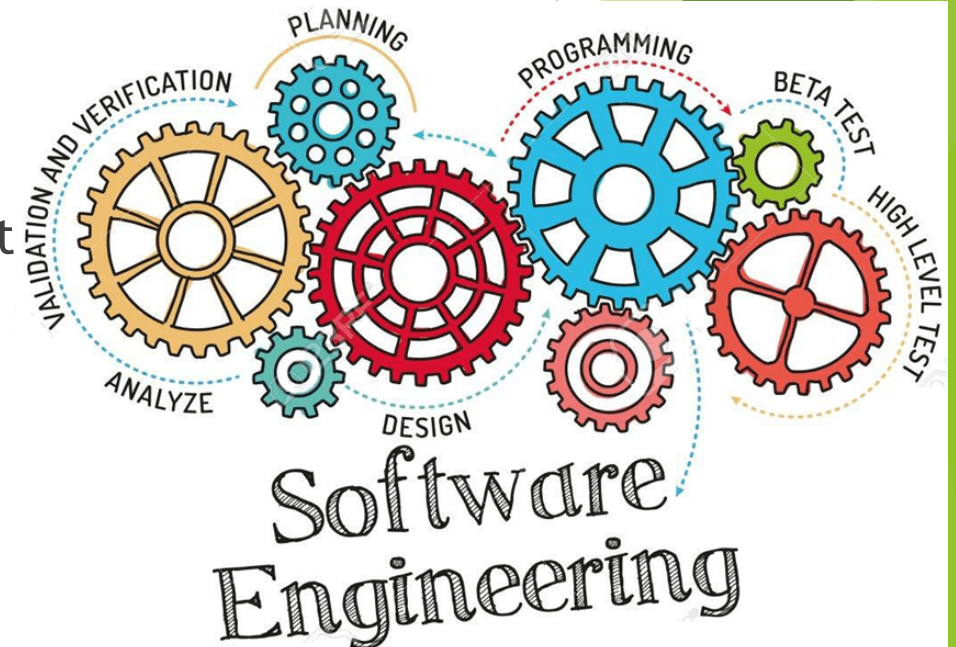
- Introduzir e conceituar engenharia de software e identificar a sua importância.
- Identificar os diferentes campos de aplicação de software e os sistemas legados
- Analisar a natureza mutante do software
- Entender a evolução da Engenharia de software e os desafios atuais
- Analisar algumas questões éticas e profissionais para engenheiros de software.

► Bibliografia básica

- PRESSMAN, R. , MAXIM, B. Engenharia de Software, **Capítulo 1**, 8th edição. AMGH, 01/2016
- SOMMERVILLE, I. Engenharia de Software. 9a edição. **Capítulo 1**. Pearson Addison Wesley. 2011.

Antes de começar...

- ▶ Qual o papel do software no mundo atual?
- ▶ Você já usou algum software que deu problema? Que tipo de problema?
- ▶ E construir um software, é complexo? Por que?
- ▶ Que problemas você pode enfrentar quando constrói um software?
- ▶ Por que estamos aqui falando de Engenharia de Software?



Introdução

Praticamente todos os países, hoje em dia, dependem de sistemas complexos baseados em computadores.

“Ideias e descobertas tecnológicas são as forças propulsoras do crescimento econômico”

The Wall Street journal

Introdução

- ▶ Produzir e manter software dentro de custos adequados é essencial para o funcionamento da economia nacional e internacional.
- ▶ Mas então... O que é Software?

O que é Software?

- ▶ “Software de computador é o produto que profissionais de software desenvolvem ao qual dão suporte no longo prazo”
(Roger Pressman)
 - ▶ Software \neq Programa de computador
- ▶ Software engloba:
 - ▶ Programa
 - ▶ Documentações (para técnicos e usuários)
 - ▶ Especificações
 - ▶ Configurações
 - ▶ Etc.

O que é Software?

- ▶ Software é abstrato e intangível
 - ▶ Não é limitado por materiais ou controlado por leis da física ou por processos de manufatura
 - ▶ Qual a implicação disto?
- ▶ Software se “desgasta”?
 - ▶ Software se *deteriora*!
- ▶ O software é desenvolvido ou passa por um processo de engenharia; não é fabricado no sentido clássico (artesanal)

Campos de aplicação de software

- ▶ Software de sistema - feitos para atender a outros programas (ex.: compiladores, componentes de um sistema operacional)
- ▶ Software de aplicação - solucionam uma necessidade específica de negócio
- ▶ Software de engenharia/científico - programas de cálculo em massa (auxiliam em áreas como astronomia, biologia molecular, meteorologia)
- ▶ Software embarcado - reside em um produto ou sistema (ex.: controle de painel de um forno de micro-ondas, funções digitais de um automóvel)
- ▶ Software para linha de produtos - projetado para prover capacidades específicas de utilização por muitos clientes diferentes (produtos de controle de inventário)
- ▶ Aplicações Web / Aplicativos móveis - contempla aplicações voltadas para navegadores e software residentes em dispositivos móveis
- ▶ Software de inteligência artificial - faz uso de algoritmos não numéricos para solucionar algoritmos complexos que não são passíveis de computação ou análise direta (sistemas especialistas, reconhecimento de padrões de imagem e voz)

Software legado

- ▶ Mais antigos, desenvolvidos já há algum tempo
- ▶ Continuamente modificado para se adequar às mudanças dos requisitos de negócio e as plataformas computacionais
- ▶ Muitos ainda dão suporte para funções de negócio que são vitais para as empresas
- ▶ As vezes tem baixa qualidade, projetos inexistentes, documentação deficiente e códigos difíceis de serem entendidos, alterações mal gerenciadas, casos de testes não documentados

Longevidade e criticidade para os negócios

Uma opção é a reengenharia, que veremos em aulas futuras

Natureza mutante do software

- ▶ WebApps
 - ▶ Inicialmente formados por páginas de conteúdos estáticos
 - ▶ Aplicações baseadas na web, com capacidade de processamento
 - ▶ Integração com banco de dados
 - ▶ APIs de acesso externo
 - ▶
- ▶ Aplicativos móveis
 - ▶ O termo aplicativo é usado para expressar software que reside em uma plataforma móvel
 - ▶ Acessa diretamente as características do hardware do dispositivo (ex. localizador GPS)
- ▶ Computação em nuvem
 - ▶ Compreende uma estrutura que permite a usuários utilizar de qualquer lugar dispositivos de computação para compartilhar recursos computacionais em grande escala
- ▶ Software em linha de produtos de software
 - ▶ Sistemas que pertencem a um segmento de mercado e compartilham recursos da Engenharia de Software (requisitos, arquitetura, padrões....) (estudaremos em aulas futuras).

Voltando ao início....

... o que é **Engenharia de Software**?

Engenharia de Software

- ▶ “A *engenharia de software* é um ramo da engenharia cujo foco é o *desenvolvimento* dentro de *custos adequados* de sistemas de software de *alta qualidade*.” Roger Pressman
- ▶ “Uma *disciplina da engenharia* relacionada com *todos* os *aspectos* da produção de software, desde os estágios iniciais de especificação do sistema até a sua manutenção, após entrar em operação.” Ian Sommerville
- ▶ “Aplicação de uma abordagem *sistemática, disciplinada e quantificável* no desenvolvimento, na operação e na manutenção de software.” IEEE

Engenharia de software

- ▶ Mas o que é Engenharia?
 - ▶ Engenharia = Uso de princípios científicos para uma atividade de projeto e construção
- ▶ Vamos usar princípios científicos para construir Software

Origem da Engenharia de Software

- ▶ Inicialmente proposto em 1968
 - ▶ Conferência da OTAN (Organização do Tratado do Atlântico Norte)
 - ▶ Objetivo de discutir a “*Crise de Software*”



O que a Crise de Software?

“Crise de Software” (Histórico)

- ▶ Introdução de novo hardware de computador baseado em circuitos integrados.
- ▶ Aplicações de computador até então não realizáveis passaram a ser viáveis.
- ▶ Software resultante era ordens de grandeza maior e mais complexo que sistemas anteriores de software.

“Crise de Software” (Histórico)

- ▶ Experiências iniciais na construção desses sistemas mostrou que o desenvolvimento informal de software não era suficiente.
- ▶ Projetos importantes apresentavam, algumas vezes, anos de atraso.

“Crise de Software” (Histórico)

- ▶ Consequência daquele cenário:
 - ▶ Custo superava previsões
 - ▶ Software não era confiável
 - ▶ Difícil de manter
 - ▶ Desempenho insatisfatório

“Crise de Software” (Histórico)

- ▶ Enquanto o custo de hardware caía o custo de software aumentava rapidamente
- ▶ Novas técnicas e métodos eram necessários para controlar a complexidade inerente aos grandes sistemas de software
- ▶ **Tais técnicas e métodos (de antigamente até hoje) compõem o corpo de conhecimento e práticas da ES**

De 1968 pra cá...

- ▶ Conseguimos melhorar o processo de construção de software
- ▶ Conhecemos uma variedade de métodos de especificação, projeto e implementação de software
- ▶ Conhecemos notações e ferramentas que reduzem o esforço necessário para a produção de sistemas
- ▶ Sabemos que não existe uma abordagem “ideal” para a engenharia de software

Analizando

- ▶ Sem software complexos:
 - ▶ não teríamos explorado o espaço,
 - ▶ não existiriam a internet e os sistemas de telecomunicações modernos,
 - ▶ meios de viagem seriam mais perigosos e caros,
 - ▶ diversos serviços essenciais para o progresso econômico custariam muito mais pela falta de informatização
- ▶ A ES tem contribuído muito.
- ▶ À medida que essa disciplina amadurece, sua contribuição no século XXI será ainda maior

Atualmente ...

- ▶ Novos desafios estão sempre aparecendo à frente do engenheiro de software
 - ▶ Interoperabilidade, múltiplas plataformas, segurança ...
- ▶ Esse cenário piora quando as empresas não aplicam as práticas de ES de forma efetiva.

Dinâmica

Desenhe uma árvore em um papel em no máximo 2 minutos

O que você considera um software de boa qualidade?

- ▶ Para te ajudar, pense que o software é um produto
- ▶ Se você fosse comprar um produto, ex. um aparelho de ar condicionado, o que você consideraria para realizar a sua compra?
 - ▶ O aparelho “gela” o ar?
 - ▶ Tem um bom design?
 - ▶ Tem assistência técnica?
 - ▶ O preço é competitivo?
 - ▶ Gasta pouca energia?
 - ▶ É silencioso?

Atributos de um bom software

- ▶ Refletem a qualidade do software
- ▶ Não estão diretamente associados ao que o software faz, fletem o comportamento durante o funcionamento, a estrutura e organização dos programas fonte e a documentação associada
- ▶ Dependem da aplicação. Ex. para um sistema bancário a segurança é essencial
- ▶ Dois atributos são destacados por Sommerville:
 - ▶ Manutenção e confiabilidade

O engenheiro de software

- ▶ Responsáveis por criar e manter o software
- ▶ O trabalho do Engenheiro de Software envolve responsabilidades mais amplas do que apenas aplicar técnicas
- ▶ O trabalho é realizado dentro de uma estrutura legal e social
- ▶ A Engenharia de Software é delimitada por leis locais, nacionais e internacionais
- ▶ O Engenheiro de Software deve se comportar de forma ética e moral

Reflexão

- ▶ Atualmente temos muito cuidado com a proteção de dados sejam eles de empresas ou pessoas
- ▶ Qual a relação entre proteção de dados e a profissão do engenheiro de software no que se refere a?
 - ▶ Acesso a dados pessoais dos cidadãos
 - ▶ Propriedade intelectual dos produtos de software
 - ▶ Confidencialidade (ex. funcionamento dos software)
 - ▶ Utilização dos computadores (vírus, acesso a arquivos, uso pessoal...)

Responsabilidade profissional ética

- ▶ **Confidencialidade:** Engenheiros de Software devem respeitar a confidencialidade de seus empregadores e clientes, tendo ou não assinado um acordo formal.
- ▶ **Competência:** Engenheiros de Software não devem enganar quanto ao seu nível de conhecimento, aceitando serviços que estejam fora do seu limite de competência;
- ▶ **Direitos de propriedade intelectual:** Os engenheiros devem estar cientes das leis locais que regulam o uso de propriedade intelectual, como patentes e direitos autorais. Devem ser cuidadosos para que a propriedade intelectual de empregadores e clientes seja protegida;
- ▶ **Má utilização de computadores:** Engenheiros de Software não devem empregar suas habilidades técnicas para o mau uso dos computadores de outras pessoas (ex. jogar, disseminar vírus)

Código de ética profissional da ES

Código de ética e de prática profissional da engenharia de software

Força tarefa da ACM/IEEE-CS sobre éticas e práticas profissionais da engenharia de software

Preâmbulo

A versão resumida do código apresenta as aspirações em um alto nível de abstração; as cláusulas que estão incluídas na versão integral dão exemplos e detalhes de como essas aspirações modificam o modo pelo qual agimos como profissionais da engenharia de software. Sem essas aspirações, os detalhes podem se tornar muito específicos e tediosos; sem os detalhes, as aspirações podem se tornar aparentemente importantes, mas vazias; juntos, aspirações e detalhes formam um código coeso.

Os engenheiros de software se comprometerão a fazer da análise, da especificação, do projeto, do desenvolvimento, dos testes e da manutenção de software uma profissão benéfica e respeitada. De acordo com seu compromisso com a saúde, a segurança e o bem-estar do público, os engenheiros de software deverão aderir aos seguintes princípios:

1. PÚBLICO – Os engenheiros de software agirão consistentemente com o interesse público.
2. CLIENTE E EMPREGADOR – Os engenheiros de software agirão de uma maneira que esteja em conformidade com os melhores interesses de seus clientes e empregadores e consistente com o interesse público.
3. PRODUTO – Os engenheiros de software deverão assegurar que seus produtos e as alterações a eles relacionadas cumpram o mais alto padrão profissional possível.
4. JULGAMENTO – Os engenheiros de software manterão integridade e independência em seu julgamento profissional.
5. GERENCIAMENTO – Os gerentes e os líderes de engenharia de software adotarão e promoverão uma abordagem ética para o gerenciamento do desenvolvimento e da manutenção do software.
6. PROFISSÃO – Os engenheiros de software fomentarão a integridade e a reputação da profissão, de modo consistente com o interesse público.
7. COLEGAS – Os engenheiros de software serão justos e darão apoio aos seus colegas.
8. PESSOAL – Os engenheiros de software participarão de aprendizado constante com relação à prática de sua profissão e promoverão uma abordagem ética dessa prática.

Atividade

Vamos analisar a demanda a seguir

Você é gerente de projetos de uma fábrica de software e foi alocado para gerenciar o projeto de desenvolvimento de um **sistema para informatização da matrícula da rede estadual de ensino**.

Atualmente a matrícula dos alunos da rede pública é bastante complicada, pois é realizada presencialmente em cada escola.

Essa prática gera muitas filas e problemas quando os alunos querem se transferir de uma escola para outra.

Imagine que temos muitas escolas e muitos alunos.

Precisamos diminuir as filas, atender ao máximo as demandas dos alunos em termo de escola/vaga

Atividade (2)

Proposta de solução

Sua equipe foi contratada para solucionar o problema da matrícula

O que poderia ser feito? Como a tecnologia pode ajudar?

Descreva em linhas gerais como seria um sistema para resolver esse problema, que funcionalidades ele teria, a plataforma (ex. web, mobile), quem utilizaria...

Atividade (3)

Antecipando os problemas

Que problemas você imagina que teria no desenvolvimento e implantação de um sistema como esse?