



# UNIVERSIDADE JOAQUIM CHISSANO

**CURSO:** ENGENHARIA EM TECNOLOGIAS  
E SISTEMAS DE INFORMAÇÃO

**DISCIPLINA:** ENGENHARIA DE SOFTWARE

Aula 01 – Introdução a engenharia de software

Eng. Momade Marcos. H. Abdul – momademha@gmail.com

Maputo, Agosto de 2022

# Conteúdos

- Apresentação do plano temático;
- Visão Geral de Engenharia de Software (Conceitos e contextualização)
  - Engenharia
  - Software
  - Engenharia de Software
- Principais problemas actuais (Crise de Software);
- Equipa de desenvolvimento de Software
- Mitos e Verdades sobre a engenharia de Software

# O plano da disciplina

- Introdução à Engenharia de Software
- Processos e modelos de processo de Software
- O padrão UML
- Engenharia de requisitos
- Arquitectura de Software
- Gestão de projectos
- Reengenharia
- Estimativas de custo
- Qualidade de Software e Certificação de qualidade
- Gestão de versões e configurações
- Projecto integrado da disciplina

# O que é software?

# O que é software?

- Invisível
- Complexo
- Mutável
- Em conformidade com o ambiente

# O que é software?

- Em termos práticos, o software é uma coleção de dados ou instruções que informam a um mecanismo como trabalhar. Ele, nada mais é do que um programa que você acessa no celular, tablet, PC, ou qualquer outro dispositivo eletrônico.
- O termo “software” foi criado apenas na década de 1940 como um trocadilho com o termo hardware. Trocando a parte “Hard” (rígido) para “Soft” (macio), o software seria tudo o que não está ligado à parte física de um dispositivo.

# Tipos de software?

- De maneira geral, baseado no objectivo, os softwares podem ser divididos em duas grandes categorias: "**softwares de sistema**" e "**softwares aplicativo**".
- Um **software de sistema** é responsável por gerir o comportamento do hardware de um dispositivo. Assim, ele oferece funcionalidades básicas exigidas pelos utilizadores, ou até mesmo por outros softwares. Alguns exemplos mais comuns:
  - **Sistemas operativos**, como Windows, macOS, Android ou iOS;
  - **Engines de jogos**, como Unity, Unreal e Cry Engine;
  - **Firmwares**, que fornecem controle de baixo nível para o hardware;
  - **Drivers**.

# Tipos de software?

- **Softwares aplicativo** são aqueles que permitem os utilizadores realizarem uma ou mais tarefas específicas. São, em sua maioria, os softwares que você utiliza diariamente no celular ou no PC, como redes sociais, navegadores, jogos e muitos outros.



# O que é engenharia?

# O que é engenharia?

- “A engenharia é a arte da aplicação dos princípios científicos, da experiência, do julgamento e do senso comum, para implementar ideias e ações em benefício da humanidade e da natureza” (COCIAN, 2009d, p. 16).
- A engenharia envolve o desenvolvimento de um produto técnico ou sistema que seja adequado para resolver uma questão específica, valendo-se, para isso, de técnicas de utilização de materiais que a natureza oferece com a energia para fazer as transformações requeridas. Pensando nisso, pode-se dizer que “A engenharia é a aplicação dos saberes científicos para criar algum elemento de valor a partir dos recursos naturais” (COCIAN, 2009d, p. 17).

# O que é engenharia?

## Os engenheiros são criativos?

- O significado da palavra “criar”, está relacionada diretamente com a engenharia. Esses profissionais criam e inovam. Não há criação e inovação sem criatividade.
- **O grupo dos engenheiros é** provavelmente **o** conjunto **mais criativo dos profissionais existentes**, olhando do ponto de vista das utilidades.
- Eles abstraem, sintetizam, resolvem problemas e inovam. Em outras palavras: permitem que se façam coisas novas e que as velhas funcionem melhor.
- Os profissionais das artes também são pessoas criativas, mas a sua criatividade se defronta com pensamentos e emoções. Já a dos engenheiros se defronta com normas técnicas, escassez de recursos e equações matemáticas. A criatividade da engenharia está diretamente relacionada com objetos e resultados. (Engenharia uma breve introdução, pag. 22)

# O que é engenharia de software?

# O que é engenharia de software?

- Engenharia de Software é uma área da Ciência da Computação voltada especificamente ao desenvolvimento de software;
- “É a aplicação dos princípios científicos, métodos, modelos, padrões e teorias que possibilitem gerir, planejar, modelar, projectar, implementar, medir, analisar, manter e aprimorar um sistema de software”.
- É uma disciplina de Engenharia relacionada com todos os aspectos da produção de *Software*.

# Principais problemas : a crise de software

- Os dispêndios com *Software* representam uma fracção significativa do Produto Interno Bruto (**PIB**) em todos os países desenvolvidos.
- A **crise do software** foi um termo utilizado nos anos 1970, quando a engenharia de software era praticamente inexistente. O termo expressava as dificuldades do desenvolvimento de software frente ao rápido crescimento da demanda por software, da complexidade dos problemas a serem resolvidos e da inexistência de técnicas estabelecidas para o desenvolvimento de sistemas que funcionassem adequadamente ou pudessem ser validados.

# Principais problemas : a crise de software

- As causas da crise do software estão ligadas a complexidade do processo de software e a relativa imaturidade da engenharia de software como profissão. A crise se manifesta de várias formas:
  - Projectos estourando o orçamento;
  - Projectos estourando o prazo;
  - Software de baixa qualidade;
  - Software muitas vezes não satisfaz os requisitos;
  - Projectos ingerenciáveis e código difícil de manter;
  - Sistemas desenvolvidos incorretamente a partir de especificações corretas;
  - Sistemas desenvolvidos correctamente a partir de especificações erradas ou incompletas.

# Principais problemas : a crise de software

- Corte deliberado do escopo do projeto, em razão do estouro do prazo ou da verba do projeto;



# Principais problemas : a crise de software

- A maior parte dos projectos continuam com estes problemas ainda na actualidade, assim pode se dizer que a crise continua vigente ainda na actualidade.
  - As soluções para a crise de software:
  - Análise econômica de sistemas de informação;
  - O uso de melhores técnicas, métodos e ferramentas;
  - Interesse do governo em treinamentos e educação;
  - A mudança de paradigma sobre o que é desenvolver software e como deveria ser feito;

# Equipa de Desenvolvimento de Software

- Gestor de projetos;
- Analista (de negócios, de requisitos, de sistemas);
- *Product owner* (Contacto com o cliente);
- Arquitecto de sistemas
- Designer
- Desenvolvedor/programador (front-end, back-end ou fullstack)
- Administrador de bases de dados - DBA

# Mitos e verdades sobre a engenharia de Software

- **Mito:** Se a equipe dispõe de um manual de padrões e procedimentos de desenvolvimento de software, então a equipe está apta a encaminhar bem o desenvolvimento.
- **Realidade:** É necessário que o que conste no dado manual reflita a prática de desenvolvimento de software e que esta prática seja verificada com frequência, para confirmar o uso do conhecimento.
- **Mito:** A equipe tem ferramentas de desenvolvimento de software de última geração, uma vez que eles dispõe de computadores de última geração.
- **Realidade:** Mais importante que ter um hardware de última geração é ter ferramentas para a automatização do desenvolvimento de software. As chamadas ferramentas *CASE (Computer-Aided Software Engineering)*.

# Mitos e verdades sobre a engenharia de Software

- **Mito:** Se o desenvolvimento do software estiver atrasado, basta aumentar a equipe para honrar o prazo de desenvolvimento.
- **Realidade:** Quanto mais pessoas pegarem “o barco andando”, pior. Por que os novos profissionais deverão ser treinados e isto será feito por membros da equipe, o que vai implicar em maiores atrasos no cronograma.
- **Mito:** Uma descrição breve e geral dos requisitos do software é o suficiente para iniciar o seu projeto. Maiores detalhes podem ser definidos posteriormente.
- **Realidade:** Este é um dos problemas que pode levar um projeto ao fracasso. O cliente deve ser questionado a fim de definir o mais precisamente possível os requisitos importantes para o software: funções, desempenho, interfaces, restrições de projeto e critérios de validação. Estes são pontos fundamentais para o sucesso do projeto.

# Mitos e verdades sobre a engenharia de Software

- **Mito:** Os requisitos de projeto mudam continuamente durante o seu desenvolvimento, mas isto não representa um problema, uma vez que o software é flexível e poderá suportar facilmente as alterações.
- **Realidade:** O software é mais flexível que a maioria dos produtos manufaturados, mas não existe software que suporte uma alteração significativa em seu escopo sem influenciar no custo e no prazo de entrega.
- **Mito:** Após a edição do programa e a sua colocação em funcionamento, o trabalho está terminado.
- **Realidade:** A realidade é justamente o contrário. Segundo estatísticas, após a implantação ocorrerão 50% a 70% do esforço do desenvolvimento de software (manutenção).

# Referencias bibliograficas

- Pressman, R.(2006). Engenharia de *Software*: Uma abordagem Profissional. 7ª edição. Editora Bookman.
- Sommerville, I.(2007). Engenharia de *Software* 8ª edição. Editora Pearson Education.
- <https://canaltech.com.br/software/o-que-e-software/>
- <https://statics-submarino.b2w.io/sherlock/books/firstChapter/129092356.pdf>