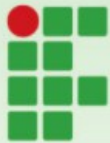


INSTITUTO FEDERAL
Santa Catarina

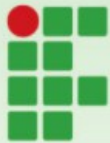


Introdução à Engenharia de Software

Prof. Edjandir Corrêa Costa
edjandir.costa@ifsc.edu.br



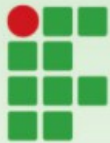
- **Introdução**
 - Crise do Software
 - Engenharia de Software
 - Termos básicos
- **Ciclo de vida**
 - Conceito
 - Fases e etapas



- **Crise do Software**

- O que foi ?

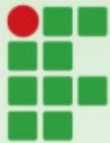
- Ganhou importância, principalmente, em razão da má qualidade dos softwares produzidos até a década de 70
 - Muitos deles eram entregues fora do prazo e com algumas poucas funcionalidades
 - Muitas funcionalidades não funcionavam como deveriam



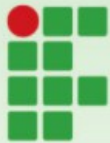
- **Crise do Software...**

- Para Dijkstra, 1972:

“A maior causa da crise do software é que as máquinas tornaram-se várias ordens de magnitude mais potentes! ... Enquanto não haviam máquinas, programar não era um problema... Agora que temos computadores gigantescos, programar tornou-se um problema gigantesco”



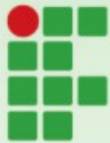
- **Crise do Software...**
 - A situação era agravada por um outro motivo: os primeiros programadores não possuíam ferramentas como dispomos hoje. A tarefa poderia ser comparada a erguer prédios empilhando mais e mais tijolos.



- **Crise do Software...**

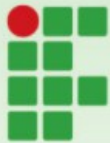
- Principais consequências/impactos:

- Necessidade de criar e adotar melhores métodos e ferramentas para produzir software
 - Criação de uma disciplina denominada de:
 - Engenharia de Software
 - A primeira vez em que se utilizou o termo “Engenharia de Software” foi em uma conferência com esse nome, realizada em 1968, na Alemanha.

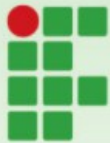


- **Engenharia de Software**

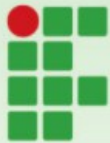
- É uma disciplina que se preocupa com todos os aspectos da produção de software de qualidade.
 - Ela usa três elementos fundamentais:
 - Métodos (fornecem a técnica: orientação a objetos)
 - Ferramentas (fornecem o suporte para os métodos - *CASE: Computer-Aided Software Engineering*)
 - Procedimentos (ações que unem métodos e ferramentas).



- **Software Engineering Body Of Knowledge - SWEBOK**
 - Em 2004, a Eng. de Software passou por um estudo para definir suas fronteiras (assuntos de maior importância);
 - Esse estudo foi conduzido pela IEEE e chama-se de SWEBOK (Corpo de Conhecimento de Eng. de Software);
 - O objetivo é concentrar mais esforço e organizar melhor o conhecimento sobre Eng. de Software.



- **Software Engineering Body Of Knowledge - SWEBOK**
 - Requisitos (Requirements) de Software
 - Projeto (Design) de Software
 - Construção (Construction) de Software
 - Teste (Testing) de Software
 - Manutenção (Maintenance) de software
 - Gerência de Configuração de Software
 - Gerência de Engenharia de Software
 - Processos de Engenharia de Software
 - Ferramentas e Métodos de Engenharia de Software
 - Qualidade (Quality) de Software



- **Termos básicos:**

- **Análise**

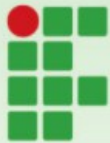
- Diz respeito ao conjunto de atividades necessárias para se descobrir **o que fazer**

- **Projeto**

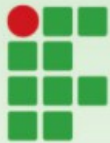
- É um termo que está relacionado ao **como fazer** o que se descobriu na análise

- **Requisito**

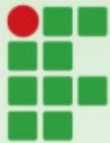
- É tudo aquilo que um cliente deseja que um software possua



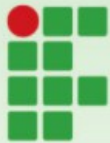
- **Mas, um software, é construído de que forma?**
 - Será que existe um passo a passo, uma receita de bolo?



- **Ciclo de vida (padrão-básico) é composto por uma série de fases ou etapas:**
 - Levantamento de requisitos
 - Análise
 - Projeto
 - Implementação
 - Testes
 - Implantação
 - Manutenção

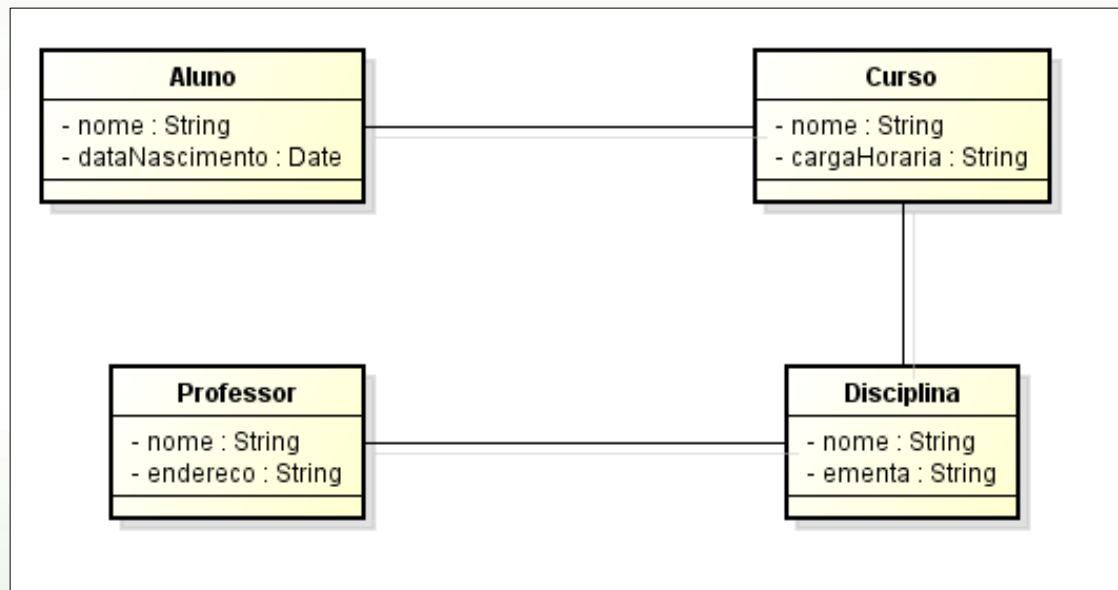


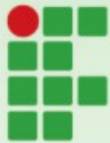
- **Etapa: Levantamento de Requisitos**
 - É uma etapa inicial, destinada ao conhecimento das requisitos (funcionalidades - operações) e das restrições que o software deve ter e/ou respeitar.
 - Nela aplicam-se algumas técnicas, como:
 - Questionário, Entrevista, Reunião, Observação, análise de documentos ...
 - O principal resultado constitui uma relação de requisitos.



- **Etapa: Análise**

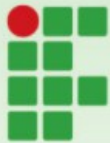
- O principal resultado desta etapa é a produção de um modelo que permita representar o problema a ser resolvido.





- **Etapa: Projeto ou *Design***

- É nesta etapa que os requisitos são transformados em software
- Nela, ainda, definem-se:
 - Linguagem de programação a ser usada
 - Qual plataforma de hardware o software deverá rodar
 - Como deverão ser as interfaces
 - Esquema de banco de dados
 - ...

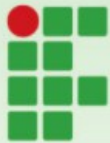


- **Etapa: Implementação**

- Aqui, com base no projeto, faz-se a programação ou codificação, ou seja, o projeto guia a codificação evitando que se perca o foco do trabalho ou se façam coisas desnecessárias

- **Etapa: Teste**

- São as ações que permitem verificar se um software atende ou não sua especificação (requisitos)
- É preciso testar com várias entradas e com várias situações que podem ocorrer no dia a dia do cliente

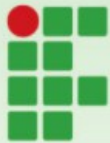


- **Etapa: Implantação**

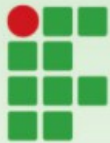
- Instalar o software na máquina do cliente, na rede do cliente, no ambiente do cliente
- Dar um suporte inicial e se necessário um treinamento

- **Etapa: Manutenção**

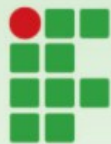
- Pode ser:
 - Corretiva: problemas ou erros no próprio software
 - Adaptativa: um novo imposto por exemplo (software refletir)
- Inicia com a instalação e só termina quando não houver mais usuários usando o software



- **Em relação as fases ou etapas do ciclo de vida**
 - Dependendo da metodologia que se usa a sequência de execução delas pode mudar
 - Algumas etapas podem ser aglutinadas ou agrupadas
 - Em alguns casos, algumas fases pode ter maior importância que outras
 - Em geral, as metodologias variam quanto a forma e a ordem, mas os conceitos aqui apresentados fazem parte e são usados no desenvolvimento de software minimamente complexo.



- **Tudo o que foi visto pretende evitar que:**
 - Software entregue esteja incorreto
 - Custos desnecessários;
 - Re-trabalhos;
 - Cronograma seja ultrapassado;
 - Insatisfação do cliente;
 - ...



Como o cliente explicou...



Como o líder de projeto entendeu...



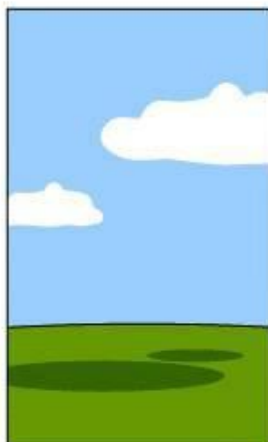
Como o analista projetou...



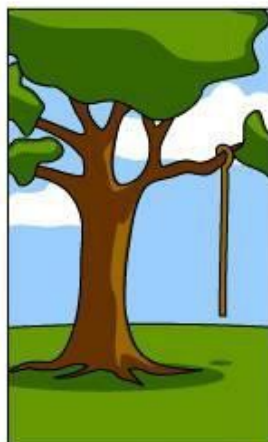
Como o programador construiu...



Como o Consultor de Negócios descreveu...



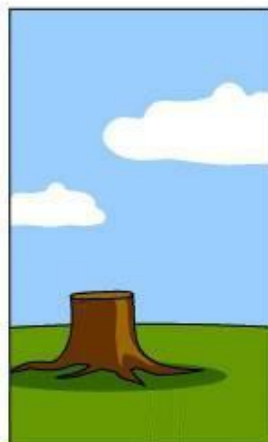
Como o projeto foi documentado...



Que funcionalidades foram instaladas...



Como o cliente foi cobrado...



Como foi mantido...



O que o cliente realmente queria...