A grayscale photograph of a person wearing large headphones, looking intently at a laptop screen. The person's hand is resting on their chin in a thoughtful pose. The background is blurred, showing what appears to be a workshop or office environment. A white rectangular border frames the central text.

# ENGENHARIA DE SOFTWARE

Tema: "Desenvolvimento de aplicativos  
computacionais"

# Engenharia de Software

## Ian Sommerville

Este autor dividiu em 4 partes a Engenharia de Software:

- Parte 1 – Introdução à engenharia de software
- Parte 2 – Confiança e proteção
- Parte 3 – Engenharia de software avançada
- Parte 4 – Gerenciamento de software



# Parte 1 – Introdução à engenharia de software

## Capítulo 1 – Introdução.

### Capítulo 2 – Processos de software.

### Capítulo 3 – Desenvolvimento ágil.

### Capítulo 4 – Engenharia de requisitos.

- Requisitos funcionais e não funcionais
- O documento de requisitos de software
- Especificação de requisitos
- Processos de engenharia de requisitos
- Elicitação e análise de requisitos
- Validação de requisitos
- Gerenciamento de requisitos

## Capítulo 5 – Modelagem de sistemas.

### Capítulo 6 – Projeto de arquitetura.

### Capítulo 7 – Projeto e implementação.

### Capítulo 8 – Testes de software.

### Capítulo 9 – Evolução de software.

- Processos de evolução
- Dinâmica da evolução de programas
- Manutenção de software
- Gerenciamento de sistemas legados

# Parte 2 – Confiança e proteção

## Capítulo 10 – Sistemas sociotécnicos.

### • Capítulo 11 – Confiança e proteção.

### • • Capítulo 12 – Especificação de cor

### • • • Capítulo 13 – Engenharia de co

### • • • Capítulo 14 – Engenharia de

- • • Gerenciamento de riscos de prot
- • • Projeto para proteção
- • • Sobrevivência de sistemas

### Capítulo 15 – Garantia de confiança e proteção.

- Análise estática
- Testes de confiabilidade
- Testes de proteção
- Garantia de processo
- Casos de segurança e confiança

# Parte 3 – Engenharia de software avançada

## Capítulo 16 – Reúso de software.

## • [ Capítulo 17 – Engenharia de software

## • [ • Capítulo 18 – Engenharia de software

## • [ • Capítulo 19 – Arquitetura de software

## • [ • Capítulo 20 – Software de sistemas

- Projeto de sistemas embutidos
- Padrões de arquitetura
- Análise de timing
- Sistemas operacionais de tempo real

## Capítulo 21 – Engenharia de software orientada a aspectos.

- Separação de interesses
- Aspectos, pontos de junção e pontos de corte
- Engenharia de software com aspectos

# Parte 4 – Gerenciamento de software

## Capítulo 22 – Gerenciamento de projetos.

- Capítulo 23 – Planejamento de projeto
- Capítulo 24 – Gerenciamento de qualidade
- Capítulo 25 – Gerenciamento de configuração
- Capítulo 26 – Melhoria de processos
  - O processo de melhoria de processos
  - Medição de processos
  - Análise de processos
  - Mudança de processos
  - Framework CMMI de melhorias de processos



# Engenharia de software: uma abordagem profissional Pressman, Roger S.

Este autor dividiu em 6 partes a Engenharia de Software:

- Introdução: Software e Engenharia de Software
- Parte 1 - O processo de software
- Parte 2 - Modelagem
- Parte 3 - Qualidade e Segurança
- Parte 4 - Gerenciamento de projetos de
- Parte 5 - Tópicos Avançados



# Introdução: Software e engenharia de software

## **Introdução: Software e engenharia de software.**

- Definição de software
- Domínios de aplicação de software
- Software legado
- Definição da disciplina
- O processo de software
- A metodologia do processo
- Atividades de apoio
- Adaptação do processo
- prática da engenharia de software
- A essência da prática
- Princípios gerais
- Como tudo começa
- Resumo Problemas e pontos a ponderar



# Parte 1 - O processo de software - modelos de processo

## AGILIDADE E PROCESSO

### UM MOMENTO

- Definição
- Identificação
- Avaliação
- Modelos
- O modelo
- Modelo c
- Modelo c
- Modelo c
- Produto c
- Problema
- O que é agilidade?
- Agilidade e o custo das
- O que é processo ágil?
- Princípios da agilidade
- A política do desenvolvimento
- Equipes e artefatos
- Reunião de planejamento
- Reunião diária do Scrum
- Reunião de revisão c
- Retrospectiva do sprint
- Outros frameworks
- O framework XP
- Kanban
- DevOps
- Problemas e pontos

### MODELO DE PROCESSO RECOMENDADO

- Definição dos requisitos
- Projeto de arquitetura preliminar
- Estimativa de recursos

## ASPECTOS HUMANOS DA ENGENHARIA DE SOFTWARE

- Características de um engenheiro de software
- A psicologia da engenharia de software
- A equipe de software
- Estruturas de equipe
- O impacto das mídias sociais
- Equipes globais
- Problemas e pontos a ponderar

# Parte 2 - Modelagem

## MODELAGEM DE REQUISITOS: UMA ABORDAGEM RECOMENDADA -Parte1

## • MODELAGEM CONCEITOS DE PROJETO - Parte1. RECOMENDADA –Parte2

## PROJETO DE ARQUITETURA: UMA ABORDAGEM RECOMENDADA – Parte1.

## PROJETO DE ARQUITETURA: UMA ABORDAGEM RECOMENDADA – Parte2.

- Arc
  - O c
  - Por
  - Des
  - Dec
  - Agi
  - Est
  - Um
  - Pad
  - Org
  - Cor
  - Dec
- Projeto de arquitetura
  - Representação do sistema no contexto10.6.2 Definição de arquétipos
  - Refinamento da arquitetura em componentes
  - Descrição das instâncias do sistema
  - Avaliação das alternativas de projeto de arquitetura
  - Revisões da arquitetura
  - Revisão de arquitetura baseada em padrões
  - Verificação de conformidade da arquitetura
  - Problemas e pontos a ponderar

# Par PROJETO BASEADO EM PADRÕES

- PRO • Padrões de projeto
- Tipos de padrões
- Os • Frameworks
- Cc • Descrição de padrões
- Cc • Aprendizado de máquina e descoberta de padrões
- Ci • Projeto de software baseado em padrões
- Pr • Contexto do projeto baseado em padrões
- Li • Pense em termos de padrões
- Ar • Tarefas de projeto
- Af • Construção de uma tabela para organização de padrões
- Pi • Erros comuns de projeto
- Pr • Padrões de arquitetura
- Pr • Padrões de projeto de componentes
- Pr • Antipadrões
- Pr • Padrões de projeto de interfaces do usuário
- Pr • Padrões de projeto de mobilidade
- Me • Problemas e pontos a ponderar
- Me
- Problemas e pontos a ponderar

## GAR TI **TESTE DE SOFTWARE – TESTES ESPECIALIZADOS PARA MOBILIDADE**

- ENC**
- Diretrizes para testes móveis
  - As estratégias de teste
  - Questões de teste da experiência do usuário
  - Teste de gestos, Entrada por teclado virtual
  - Entrada e reconhecimento de voz
  - Alertas e condições extraordinárias
  - Teste de aplicações para Web
  - As estratégias de teste para a Web
  - Teste de conteúdo, Teste de interface
  - Testes de navegação
  - Internacionalização, Teste de segurança
  - Teste de desempenho, Teste em tempo real
  - Testes para sistemas de inteligência artificial (IA)
  - Testes estáticos e dinâmicos
  - Teste baseado em modelo , Teste de ambientes virtuais
  - Teste de usabilidade, Teste de acessibilidade
  - Teste de jogabilidade
  - Teste da documentação e dos recursos de ajuda
  - Problemas e pontos a ponderar

# Parte 4 - Gerenciamento de projetos

## C C I C G] UMA ESTRATÉGIA PARA SUPORTE DE SOFTWARE

- • • • • Suporte de software, Manutenção de software
- • • • • Tipos de manutenção, Tarefas de manutenção
- • • • • Engenharia reversa, Suporte proativo de software
- • • • • Uso de análise de software
- • • • • O papel das mídias sociais
- • • • • Custo do suporte, Refatoração
- • • • • Refatoração de dados, Refatoração de código
- • • • • Refatoração da arquitetura
- • • • • Evolução de software , Análise de inventário
- • • • • Reestruturação dos documentos
- • • • • Engenharia reversa , Refatoração de código
- • • • • Refatoração de dados , Engenharia direta
- • • • • Problemas e pontos a ponderar
- •

# Parte 5 Tópicos avançados

## **M TENDÊNCIAS EMERGENTES NA ENGENHARIA DE SOFTWARE**

- • Evolução da tecnologia
- • A engenharia de software como disciplina
- • Observação de tendências na engenharia de software
- • Identificação das “tendências leves”
- • Gestão da complexidade , Software aberto
- • Requisitos emergentes, O mix de talentos , Blocos básicos de software
- • Mudança na percepção de “valor”, Código aberto , Rumos da tecnologia
- • Tendências de processo , O grande desafio
- • Desenvolvimento colaborativo
- • Engenharia de requisitos29.5.5 Desenvolvimento de software dirigido por modelo
- • Engenharia de software baseada em busca
- • Desenvolvimento guiado por teste
- • Tendências relacionadas a ferramentas
- • Problemas e pontos a ponderar





# Meu Primeiro Software

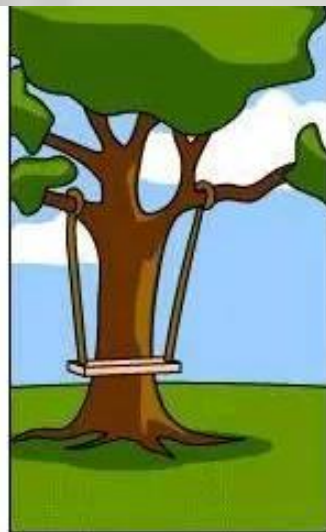
- Ufa !!!! Teoria que não acaba !!!

No próximo SLIDES em uma imagem resumiremos toda essa teoria !!!

Esta  
imagem  
é bem  
antiga  
mas muito  
atual



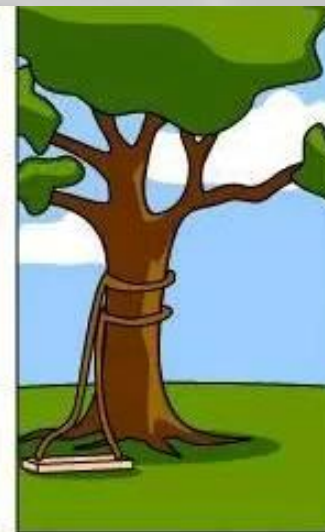
Como o cliente  
explicou...



Como o lider de projeto  
entendeu...



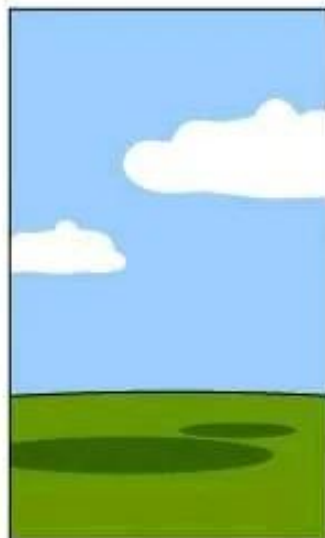
Como o analista  
projetou...



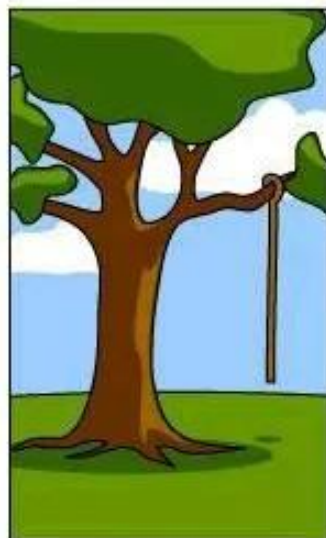
Como o programador  
construiu...



Como o Consultor de  
Negócios descreveu...



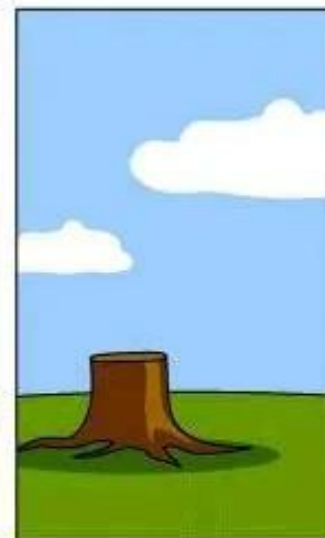
Como o projeto foi  
documentado...



Que funcionalidades  
foram instaladas...



Como o cliente foi  
cobrado...



Como foi mantido...



O que o cliente  
realmente queria...





# Hands On

- Vamos a Prática
- Em nossa Situação de Aprendizagem –SA01 Faremos um software em poucas linhas que faz o carregamento e a leitura de arquivos , dando assim início em nosso projeto



# Resumo da AS-01

- **CONTEXTUALIZAÇÃO:**
- A Empresa IANES - Call Center S/A é basicamente uma empresa que aluga espaços e pessoal para atendimento aos clientes de outras empresas.
- Por exemplo a empresa Faber Castell S/A tem o seu centro de atendimento instalado dentro da IANES - Call Center S/A.
- A IANES além de fornecer toda a infraestrutura para a Faber Castell S/A como computadores, telefones, internet e funcionários.
- Resolveu ter um diferencial na sua proposta de atendimento que é um software para controlar os chamados/ligações, fornecendo relatórios estatísticos e usando o Big Data gerado para resolver problemas de negócios que você não conseguiria resolver antes.
- A IANES vendo uma oportunidade de atender seus clientes resolveu criar o software para resolver esse problema. Qualquer falha, mínima que seja, no sistema pode resultar em um grande prejuízo.
- Preocupada com os indicadores que podem ser gerados com esse software a IANES iniciou este projeto.



# Progresso do curso

## Etapa 1

Usaremos a  
Linguagem  
Python e uma  
biblioteca :  
STREAMLIT

## Etapa 2

Instalar o Python  
Instalar o  
Streamlit  
Ou acessar o site:  
[https://streamlit.  
io/](https://streamlit.io/)

## Etapa 3

Instalar uma IDE  
para  
programação em  
Python

## Etapa 4

Codificar em  
Python

## Etapa 5

“Publicar” ou  
colocar em  
produção o

LINK da Aplicação: <https://senaispaula00-henrihva.streamlit.app/>

# Código da aplicação em Python

[https://github.com/hvieirasenai/SenaiSP\\_Aula00](https://github.com/hvieirasenai/SenaiSP_Aula00)

[LINK da Aplicação: https://senaispaula00-henrihva.streamlit.app/](https://senaispaula00-henrihva.streamlit.app/)





# OBRIGADO!

Henri Vieira Alves

Email: [henrihva@hotmail.com](mailto:henrihva@hotmail.com)

Os alunos motivam-se, envolvem-se e comprometem-se quando têm clareza da razão, da utilidade e do significado do que estão aprendendo.

