



"Antes de imprimir pense em sua responsabilidade e compromisso com o **MEIO AMBIENTE.**"

Engenharia de Software

Práticas de Engenharia de Software



Givanaldo Rocha de Souza

givanaldo.rocha@ifrn.edu.br

<http://docente.ifrn.edu.br/givanaldorochoa>

Material original gentilmente cedido pelo professor Fábio Procópio



Como resolver problemas?

□ No livro *How to solve it* (1945), foi apresentada a essência da solução de problema, a qual pode ser aplicada à Engenharia de Software:



- ✓ Entenda o problema (comunicação e análise)
- ✓ Planeje uma solução (modelagem e projeto de software)
- ✓ Execute o plano (geração de código)
- ✓ Examine o resultado quanto à precisão (teste e qualidade)

Práticas de Eng. de Soft.

- Como resolver problema
- Práticas de Comunicação
- Princípios
- Práticas de Planejamento
- Princípios
- Práticas de Modelagem
- Princípios de Análise
- Princípios de Projeto
- Práticas de Construção
- Princíp.de Codificação
- Princípios de Teste
- Caixa Branca e Preta
- Práticas de Implantação
- Princípios
- Referências



Como resolver problemas?

Entenda o problema

- Quem tem interesse na solução do problema?
- Quais dados, funções, características e comportamento são necessários para resolver o problema?
- É possível representar problemas menores para facilitar a compreensão?
- O problema pode ser representado graficamente?

Planeje a solução

- Já viu problemas parecidos?
- Já resolveu algum problema parecido?
- É possível subdividir os problemas?
- É possível definir um modelo que possa ser implementado?

Práticas de Eng. de Soft.

- Como resolver problema
- Práticas de Comunicação
- Princípios
- Práticas de Planejamento
- Princípios
- Práticas de Modelagem
- Princípios de Análise
- Princípios de Projeto
- Práticas de Construção
- Princíp.de Codificação
- Princípios de Teste
- Caixa Branca e Preta
- Práticas de Implantação
- Princípios
- Referências



Como resolver problemas?

Execute o plano

- A solução está de acordo com o plano?
- Cada componente da solução está correto?

Examine o resultado

- Foi elaborada uma estratégia de teste?
- O software foi avaliado de acordo com os requisitos?

Práticas de Eng. de Soft.

- Como resolver problema
- Práticas de Comunicação
- Princípios
- Práticas de Planejamento
- Princípios
- Práticas de Modelagem
- Princípios de Análise
- Princípios de Projeto
- Práticas de Construção
- Princíp.de Codificação
- Princípios de Teste
- Caixa Branca e Preta
- Práticas de Implantação
- Princípios
- Referências

Práticas de Comunicação



❑ Antes de tudo, os requisitos são coletados por meio de uma atividade de comunicação chamada **levantamento de requisitos**.

❑ A comunicação para o entendimento de um problema, normalmente, é difícil.

❑ A comunicação é considerada uma das atividades mais desafiadoras encontrada por um engenheiro de software.



Práticas de Eng. de Soft.

➔ Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Comunicação: princípios

❑ **Princípio 1:** **Escute**

- ❑ Concentre-se nas respostas do interlocutor;
- ❑ Peça esclarecimentos quando algo estiver obscuro;
- ❑ Evite interrupções constantes;
- ❑ Evite sacudir a cabeça, virar os olhos;
- ❑ Evite influenciar a resposta do interlocutor.

❑ **Princípio 2:** **Prepare-se antes de se comunicar**

- ❑ Gaste tempo para entender o problema;
- ❑ Se necessário, pesquise para entender o jargão do negócio.

❑ **Princípio 3:** **Alguém deve facilitar a atividade**

- ❑ Toda reunião deve ter um facilitador para:
 - ❑ Conduzir a conversa para que seja sempre produtiva;
 - ❑ Mediar possíveis conflitos;
 - ❑ Garantir que os outros princípios sejam seguidos.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
➔ Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Comunicação: princípios

- ❑ **Princípio 4: Comunicação face a face é melhor**
 - ❑ Mas a apresentação de uma outra forma de comunicação é sempre bem-vinda. Por exemplo, um documento ou um desenho.
- ❑ **Princípio 5: Faça anotações e documente as decisões**
 - ❑ Alguém que participe da comunicação deve anotar todos os pontos e decisões importantes.
- ❑ **Princípio 6: Busque colaboração**
 - ❑ Procure fazer com que os membros da equipe façam colaborações, mesmo que pequenas, pois elas aumentam a confiança entre os membros.
- ❑ **Princípio 7: Conserve-se enfocado, modularize sua discussão**
 - ❑ O facilitador deve manter a conversa modular, abandonando um tópico apenas depois que tiver sido resolvido.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
➔ Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Comunicação: princípios

- ❑ **Princípio 8:** Se algo não está claro, desenhe uma figura
 - ❑ Quando as palavras não conseguem deixar algo bem esclarecido, normalmente, um desenho o faz.

- ❑ **Princípio 9:** Quando concordar com algo, prossiga; quando não concordar, prossiga; se algo não está claro e não pode ser esclarecido no momento, prossiga
 - ❑ Prosseguir, às vezes, é a melhor forma de conseguir agilidade na comunicação.

- ❑ **Princípio 10:** Negociação não é um concurso ou um jogo. Funciona melhor quando ambas as partes ganham
 - ❑ Muitas vezes, engenheiros de software e clientes precisam negociar e a negociação exige compromisso de todos.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
➔ Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Práticas de Planejamento

- ❑ Permitem à equipe de software definir um roteiro durante a execução da meta estratégica e dos objetivos táticos;
- ❑ A falta de planejamento é algo que pode levar um projeto ao caos;
- ❑ Todos os membros da equipe de software devem participar do planejamento.



Práticas de Eng. de Soft.

- Como resolver problema
- Práticas de Comunicação
- Princípios
- ➔ Práticas de Planejamento
- Princípios
- Práticas de Modelagem
- Princípios de Análise
- Princípios de Projeto
- Práticas de Construção
- Princíp.de Codificação
- Princípios de Teste
- Caixa Branca e Preta
- Práticas de Implantação
- Princípios
- Referências

Prof. Gustavo Henrique



Planejamento: princípios

- ❑ **Princípio 1:** Entenda o escopo do projeto
 - ❑ O escopo fornece à equipe de software um roteiro a seguir.
- ❑ **Princípio 2:** Envolver o cliente na atividade de planejamento
 - ❑ O engenheiro de software precisa negociar com o cliente ordem de entrega, prazos e outros itens do projeto.
- ❑ **Princípio 3:** Reconheça que o planejamento é iterativo
 - ❑ Quando o trabalho se inicia, provavelmente, haverá modificações;
 - ❑ Modelos de processos iterativos e incrementais determinam replanejamento, baseado em *feedbacks*.
- ❑ **Princípio 4:** Estime com base no que se sabe
 - ❑ Estimar fornece uma base para se calcular o esforço, o custo e a duração das tarefas.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princípios de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências




Planejamento: princípios

- ❑ **Princípio 5:** Considere riscos à medida que se define o plano
 - ❑ O plano do projeto deve ser ajustado para acomodar a probabilidade de um risco vir a ocorrer

- ❑ **Princípio 6:** Seja realista
 - ❑ Omissões, ambiguidades e modificações são comuns;
 - ❑ Existem falhas na comunicação entre as pessoas;
 - ❑ As pessoas não trabalham durante todo o dia.

- ❑ **Princípio 7:** Ajuste a granularidade à medida que o plano é definido
 - ❑ Granularidade fina – fornece detalhes significativos de tarefas incrementadas em um curto espaço de tempo;
 - ❑ Granularidade grossa – fornece detalhes significativos de tarefas incrementadas em um longo espaço de tempo.

Práticas de Eng. de Soft.



Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Planejamento: princípios

- ❑ **Princípio 8:** Defina como a qualidade será garantida
 - ❑ Se revisões técnicas formais vierem a ser usadas, elas devem ser definidas no cronograma;
 - ❑ Se a programação em pares vier a ser usada, isso deve ser definido dentro do plano.

- ❑ **Princípio 9:** Descreva como as modificações serão acomodadas
 - ❑ O planejamento pode ser comprometido por modificações;
 - ❑ A equipe deve identificar como as modificações devem ser acomodadas durante o projeto.

- ❑ **Princípio 10:** Acompanhe o plano com frequência e faça ajustes quando necessário
 - ❑ Acompanhe o progresso do projeto diariamente;
 - ❑ Identifique pontos em que o trabalho programado não está de acordo com o trabalho real conduzido.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Práticas de Modelagem

- ❑ Os modelos precisam representar os requisitos em diferentes níveis de abstração (do usuário e do técnico);
- ❑ Na Engenharia de Software, existem duas classes de modelos:
 - ❑ Modelo de análise e;
 - ❑ Modelo de projeto.
- ❑ O **Modelo de Análise** representa os requisitos do cliente sob três domínios: domínio da informação, domínio funcional e domínio comportamental;
- ❑ O **Modelo de Projeto** representa características do software que ajudam os profissionais a construir: arquitetura, interface do usuário e componentes.



Práticas de Eng. de Soft.

- Como resolver problema
- Práticas de Comunicação
- Princípios
- Práticas de Planejamento
- Princípios
- ➔ Práticas de Modelagem
- Princípios de Análise
- Princípios de Projeto
- Práticas de Construção
- Princíp.de Codificação
- Princípios de Teste
- Caixa Branca e Preta
- Práticas de Implantação
- Princípios
- Referências



Princípios de Análise

- ❑ **Princípio 1:** O domínio de informação de um problema precisa ser representado e entendido
 - ❑ O domínio da informação abrange os dados de entrada/saída do sistema e os depósitos de dados.
- ❑ **Princípio 2:** As funções a serem desenvolvidas pelo software devem ser definidas
 - ❑ A descrição das funções pode ser de forma genérica ou detalhada.
- ❑ **Princípio 3:** O comportamento do software precisa ser representado
 - ❑ O comportamento se dá por meio das suas interações com o ambiente externo.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
➔ Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Análise

- ❑ **Princípio 4:** Os modelos que mostram informação, função e comportamento devem ser particionados de um modo que revele detalhes em forma de camadas
 - ❑ Dividir um problema complexo em subproblemas de modo a torná-los mais fáceis de se compreender.

- ❑ **Princípio 5:** A tarefa de análise deve ir da informação essencial até os detalhes de implementação
 - ❑ Um problema é descrito sem considerar como a solução será implementada;
 - ❑ Já os detalhes da implementação indicam como a solução será implementada.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Projeto

- ❑ **Princípio 1:** O projeto deve estar relacionado ao modelo de análise
 - ❑ O modelo de projeto produz uma arquitetura com vários subsistemas e projetos em nível de componentes.
- ❑ **Princípio 2:** Sempre considere a arquitetura do sistema a ser construído
 - ❑ A arquitetura é o esqueleto do sistema que será construído e afeta:
 - ❑ Interfaces;
 - ❑ Estruturas de dados;
 - ❑ Fluxo de controle;
 - ❑ Forma como o software poderá ser testado e;
 - ❑ Outros
- ❑ **Princípio 3:** O projeto de dados é tão importante quanto o projeto de funções de processamento
 - ❑ Um projeto de dados bem estruturado simplifica o fluxo do programa, facilita a implementação de componentes e otimiza o processamento global.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Projeto

- ❑ **Princípio 4:** As interfaces (internas e externas) precisam ser projetadas cuidadosamente
 - ❑ Interfaces bem projetadas tornam a integração mais fácil e ajuda o testador durante a validação.
- ❑ **Princípio 5:** O projeto de interface do usuário deve estar sintonizado com as necessidades do usuário final. Mas, em cada caso, ele deve enfatizar a facilidade de uso
 - ❑ Mesmo que uma arquitetura esteja bem projetada, se o projeto de interface for pobre, geralmente, o usuário tem a percepção de que o sistema é ruim.
- ❑ **Princípio 6:** O projeto em nível de componente deve ser funcionalmente independente
 - ❑ Um componente deve focar apenas uma função ou subfunção.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Projeto

- ❑ **Princípio 7:** Os componentes devem ser fracamente acoplados uns aos outros e ao ambiente externo
 - ❑ À medida que o acoplamento aumenta, a probabilidade de propagação de erros também aumenta.
- ❑ **Princípio 8:** Representações de projeto (modelos) devem ser facilmente compreensíveis
 - ❑ Se o projeto for difícil de entender, ele não serve como um meio de comunicação.
- ❑ **Princípio 9:** O projeto deve ser desenvolvido iterativamente. A cada iteração o projetista deve focar sempre a simplicidade
 - ❑ As primeiras iterações refinam o projeto e corrigem erros;
 - ❑ Mas, as últimas iterações devem procurar tornar o projeto o mais simples possível.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Práticas de Construção

- ❑ Contemplam um conjunto de tarefas de codificação e de teste no software que será entregue ao usuário;
- ❑ A codificação pode ser:
 - ❑ criação direta de código-fonte;
 - ❑ geração automática de código executável.
- ❑ O foco inicial do teste é nos componentes, também chamado de **teste unitário**.
- ❑ Outros tipos de teste são:
 - ❑ teste de integração – é realizado enquanto o sistema está sendo construído;
 - ❑ teste de validação – avalia se o sistema completo atende os requisitos do cliente;
 - ❑ teste de aceitação – é conduzido pelo cliente a fim de utilizar todas as funções necessárias.



Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
→ Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Codificação

❑ Princípios de preparação: Antes de escrever uma linha de código certifique-se de:

- ❑ entender o problema que está tentando resolver;
- ❑ entender os princípios e conceitos básicos do projeto;
- ❑ escolher uma linguagem de programação que satisfaça às necessidades do software a ser construído e do ambiente em que ele vai operar;
- ❑ selecionar um ambiente de programação que forneça ferramentas para facilitar o trabalho;
- ❑ criar um conjunto de testes unitários que serão aplicados.



Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências





Princípios de Codificação

❑ Princípios de codificação: Quando começar a escrever o código, certificar-se de:

- ❑ usar programação estruturada nos algoritmos;
- ❑ escolher estruturas de dados que atendam às necessidades do projeto;
- ❑ entender a arquitetura do software e criar interfaces consistentes com ela;
- ❑ a lógica condicional deve ser tão simples quanto possível;
- ❑ criar ciclos aninhados que sejam fáceis de testar;
- ❑ definir nomes significativos para as variáveis;
- ❑ escrever códigos auto documentados;
- ❑ códigos devem ser identados.



Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princípios de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Codificação

- ❑ **Princípios de validação:** Depois de completar o primeiro passo de codificação, certificar-se de:
 - ❑ inspecionar o código adequado;
 - ❑ corrigir testes unitários e corrigir os erros descobertos;
 - ❑ Refatorar o código.



Práticas de Eng. de Soft.

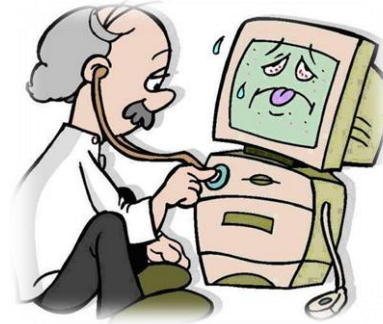
Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princípios de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Teste

- ❑ O livro *The Art of Software Testing* (1979) comenta sobre testes:
 - ❑ Teste é um processo de execução de um programa com a finalidade de encontrar erros;
 - ❑ Um bom caso de teste é aquele que tem alta probabilidade de encontrar um erro ainda não descoberto;
 - ❑ Um teste bem sucedido é aquele que descobre um erro ainda não descoberto.

- ❑ **Princípio 1:** Todos os testes devem estar relacionados aos requisitos do cliente
 - ❑ Do ponto de vista do cliente, um erro severo é aquele no qual o programa não atende aos seus requisitos.



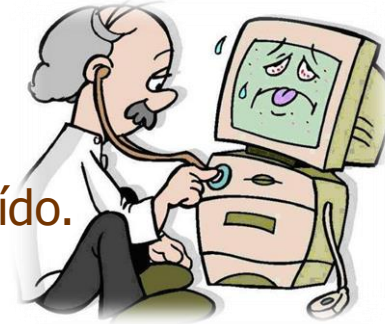
Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Teste

- ❑ **Princípio 2:** Os testes devem ser planejados muito antes de serem iniciados
 - ❑ O planejamento de testes pode começar assim que o modelo de análise for concluído.
- ❑ **Princípio 3:** O princípio de Pareto se aplica ao teste de software
 - ❑ 80% de todos os erros descobertos durante o teste estarão, provavelmente, relacionados a 20% de todos os componentes do programa.



Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Princípios de Teste

- ❑ **Princípio 4:** O teste deve começar no “varejo” e progredir até o “atacado”
 - ❑ Deve-se procurar identificar erros não apenas em um componente, mas em um conjunto deles.



- ❑ **Princípio 5:** Testes exaustivos não são possíveis
 - ❑ Dependendo da complexidade de um software, é impossível realizar todas as combinações de teste. Mas, é possível, planejar os testes de modo a atender todos os componentes do software.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências

Testes de Caixa Branca e Caixa Preta



Teste de caixa-branca (ou teste de caixa de vidro)

- Garante que todos os caminhos independentes de um módulo tenham sido exercitados pelo menos uma vez;
- Exercita todas as decisões lógicas do lado Verdadeiro e Falso;
- Exercita as estruturas de dados internas para garantir a sua validade.



Teste de caixa-preta (ou teste comportamental)

- Focaliza os requisitos funcionais do software;
- Abordagem complementar aos testes de caixa branca;
- Tenta encontrar os seguintes erros de:
 - funções incorretas ou omitidas;
 - interface;
 - estrutura de dados ou acesso ao BD externo;
 - comportamento ou desempenho;
 - iniciação e término.



Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências



Práticas de Implantação

- ❑ Os modelos modernos de processo de software são evolutivos e, assim, a implantação ocorre mais de uma vez;
- ❑ Cada ciclo de *feedback*, depois que o software é entregue, fornece à equipe diretrizes importantes que resultam em modificações no software;
- ❑ A entrega de um incremento representa um marco importante para qualquer projeto de software.



Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
➔ Práticas de Implantação
Princípios
Referências



Implantação: princípios

- ❑ **Princípio 1:** As expectativas do cliente quanto ao software devem ser geridas
 - ❑ Frequentemente, o cliente espera mais do que a equipe prometeu entregar e a frustração é imediata.

- ❑ **Princípio 2:** Um pacote completo de entrega deve ser montado e testado
 - ❑ Um CD, DVD ou outra mídia deve ser entregue ao cliente e rigorosamente testado.

- ❑ **Princípio 3:** Um regime de suporte deve ser estabelecido antes de o software ser entregue
 - ❑ Planejamento do suporte;
 - ❑ Preparação de material.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências





Implantação: princípios

- ❑ **Princípio 4:** Materiais institucionais adequados devem ser fornecidos aos usuários finais
 - ❑ Treinamento adequado;
 - ❑ Descrever as diferenças do incremento de software em relação à versão anterior.

- ❑ **Princípio 5:** Software defeituoso deve ser corrigido primeiro e, depois, entregue
 - ❑ *"Os clientes esquecerão que você entregou um produto de alta qualidade alguns dias depois, mas eles nunca esquecerão os problemas que um produto de baixa qualidade lhes causou", ditado no negócio de software.*

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação
Princípios
Práticas de Planejamento
Princípios
Práticas de Modelagem
Princípios de Análise
Princípios de Projeto
Práticas de Construção
Princíp.de Codificação
Princípios de Teste
Caixa Branca e Preta
Práticas de Implantação
Princípios
Referências





Referências

PRESSMAN, Roger S. **Engenharia de Software**. Mc Graw Hill, 6 ed, Porto Alegre, 2010.

SOMMERVILLE, Ian. **Engenharia de Software**. Pearson, 9 ed, São Paulo, 2011.

Práticas de Eng. de Soft.

Como resolver problema
Práticas de Comunicação

Princípios

Práticas de Planejamento

Princípios

Práticas de Modelagem

Princípios de Análise

Princípios de Projeto

Práticas de Construção

Princíp.de Codificação

Princípios de Teste

Caixa Branca e Preta

Práticas de Implantação

Princípios

➔ Referências