**Módulo 06 - Tipos de testes**

**Requisitos:**

De forma simples, trata-se do processo de compreensão e identificação das necessidades que o cliente espera ser solucionado pelo sistema que será desenvolvido, definindo o que o software vai fazer.

**Criação dos Cenários de Teste de Aceitação**

**Especificar:**

As funcionalidades do Sistemas

**Arquitetura:**

Criação dos Cenários de Teste de Sistema.

**Codificação:**

Testes de unidade

Testes de integração

* Testes de Sistema
* Testes de Aceitação
* Testes Alfa e Beta

**Criação do Plano de Sistema:**

**Ferramentas - Testes de Sistema**

**Boas Práticas (Não Técnicas)**

* A pessoa que especifica os cenários de teste de sistema não deve estar participando da codificação do sistema.
* A especificação e a documentação é a base para os cenários de testes.
* O'Que não não está especificado, não será validado.

**Faça uma boa seleção de cenários, considerando:**

* O tempo disponível para execução dos testes
* A maturidade dos testadores
* A relevância das funcionalidades a serem testadas.
* Ser pessimista! Cuidado para não ignorar cenários aparentemente inocentes.
* Todo software contém falhas.
* Registrar todas as falhas

**Boas Práticas ( Técnicas).**

* Não execute cenários roboticamente. Explore quando necessário.
* Achou uma falha? detalhe ao maximo!
* Uma descrição detalhada do erro ou falha encontrada.
* A falha foi corrigida? reteste! - Teste de regressão.
* Busque padrões ao reportar uma falha

**Ferramentas:**

1. Prog4mer
2. jira
3. Bugzilla
4. Trac
5. Etc..

**Técnicas:**

**Teste sempre os valores limites, grande parte dos erros estarão lá.**

**formulários:**

* Campos em branco
* Campos com valores inválidos
* Quantidade de caracteres
* E-mail inválido
* Mensagens de erro coerentes

**Cálculos:**

* Valores positivos
* Valores negativos
* Zero
* Intervalos fechados e abertos
* Acima do limite máximo e abaixo do limite mínimo.
* Pensar como o usuário final - para encontrar falhas.

**O Custo da Falha:**

* Iniciar e corrigir os erros no início do projeto.

**Mits do Teste de Software:**

* Testando vamos encontrar 100% dos defeitos.
* Teste de Software e um custo extra para o projeto.
* Qualquer pessoa é capaz de testar bem.
* Programadores não precisam testar.
* Os testes dos programadores são suficientes.
* Só podemos testar quando o sistema estiver 100% pronto.