

Solução final

Dado que temos um pouco de complexidade sobre as regras do sistema, podemos iniciar nossa solução, com um teste funcional, utilizando a técnica de tabela de decisão, do tipo de teste caixa preta, para organizar melhor algumas das regras que queremos testar.

Tipo	Variaveis	Partições	T1	T2	T3
Entrada	Cargo	Gerente	.		
		Especialista		.	
		Técnico			.
	Tempo de empresa	≥ 5 anos		.	.
		< 5 anos	.		
	regiões do país	Norte			
		Nordeste			
		Sul	.	.	.
Saída	Permite Abertura	Sim		X	
		Não	X		X
	Permite Acesso	Sim		X	
		Não	X		X
	Supervisionado?	Sim			X
		Não	X	X	

Desta forma, podemos montar alguns casos de teste:

- **Dado** que sou um GERENTE
- **E** tenho menos de 5 anos de experiência em usinas
- **Quando** eu tentar acessar o novo módulo
- **Então** o sistema não deve permitir minha entrada

- **Dado** que sou ESPECIALISTA
- **E** tenho exatos 5 anos de experiência em usinas
- **Quando** eu acessar o novo módulo
- **Então** o sistema deve autorizar minha entrada
- **E** eu serei autorizado a fazer abertura de comportas

- **Dado** que sou TÉCNICO
- **E** tenho mais de 5 anos de experiência em usinas
- **E** moro no Nordeste
- **Quando** eu acessar o novo módulo
- **Então** o sistema não deve permitir minha entrada
- **E** meu acesso deve ser supervisionado

Estratégia de teste -> **novο módulo de comportas, usina hidrelétrica X**

Este documento visa descrever como testamos o sistema, e qual o objetivo dos testes em cada nível do software.

1. Testes de Sistema

- a. **Usabilidade:** Os testes de usabilidade tem como objetivo observar e validar se todos os tipos de dispositivos descritos conseguem acessar as jornadas de forma correta, sem defeitos visuais, além de tentar garantir uma boa experiência para o usuário
 - i. **Como é feito?** Toda a tela desenhada pelo Designer, é testada a partir dos requisitos apresentados no texto, para entender se o sistema é acessível nas resoluções nativas dos dispositivos apresentados.
- b. **Caixa Preta:** Usado para entender o sistema com base nos requisitos apresentados, entendemos que teremos variações de valores, e regras complexas para compra e venda, por isso, definimos duas técnicas principais
 - i. **Tabela de decisão:** Dado a complexidade, para melhorar o entendimento e não encontrar discrepâncias nas regras, optamos pela tabela de decisão para organizar melhor nossos casos e partições a serem testadas.

2. Integração

- a. **Performance:** Como o sistema já possui uma base grande de usuários, e essa quantidade tem a possibilidade de aumentar, dado a implementação de um novo módulo, é importante entender com os testes de performance, se nossa estrutura atual e futura, consegue se adequar às novas expectativas do sistema.
- b. **Segurança:** Serve para entendermos e avaliarmos as diversas permissões que temos nos diferentes módulos do sistema
- c. **Integração de componentes:**
 - i. Integração de duas unidades, avaliando se estas funcionam corretamente, dado a nova implementação que estará sendo feita.

3. Unidade:

- a. Testes que visam garantir que toda nova unidade produzida, consiga ser testada de forma individual, é interessante que esse teste possa ser estático, dado a velocidade que os testes podem ser feitos, sem a necessidade do sistema estar em execução.