

Teste a partir de Casos de Uso

Profa. Ellen Francine

***(slides parcialmente
elaborados pelo
Prof. Marcos L. Chaim)***

Teste de Software

- Análise **dinâmica** do produto de software.
 - Processo de executar o software de modo controlado, observando seu comportamento em relação aos requisitos especificados.
- Processo de executar um programa com a intenção de **encontrar erros**.
 - O teste bem sucedido é aquele que consegue determinar casos de teste que resultem na **falha** do programa sendo testado.

Teste de Software

- Como testar um software?
 - **Técnicas e critérios**: fornecem maneiras sistemáticas para se conduzir a atividade de teste.
 - Auxiliam na seleção, no projeto dos casos de teste.
 - Funcional, estrutural, baseada em erros.
 - Que **casos de teste** utilizar?
 - Entrada + Saída Esperada
 - Um bom caso de teste tem alta probabilidade de revelar um erro ainda não descoberto.

Técnicas de Teste

- Funcional
 - Critérios
 - Particionamento em Classes de Equivalência
 - Análise do Valor-Limite
 - ...

Particionamento em Classes de Equivalência

- Divide os domínios de entrada e saída do programa em classes de dados.
 - Em princípio, todos os elementos de uma classe devem se comportar de maneira equivalente.
 - **Classes de equivalência.**
- Os casos de teste são derivados a partir das classes de equivalência.

Particionamento em Classes de Equivalência

- Passos

1. Identifique as classes de equivalência.
2. Atribua um número único a cada classe de equivalência identificada.
3. Até que todas as classes de equivalência **válidas** tenham sido cobertas, escreva um novo caso de teste incluindo o maior número possível de classes **válidas** que ainda não foram cobertas.
4. Até que todas as classes de equivalência **inválidas** tenham sido cobertas, escreva um caso de teste para cada uma, e somente uma, classe **inválida** não coberta.

Determinação das Classes de Equivalência

- Se uma entrada especifica uma faixa, então uma classe válida e duas inválidas devem ser selecionadas.
 - Condição: $0 < x < 10$
 - Classe válida: $0 < x < 10$
 - Classe inválida: $x \leq 0$
 - Classe inválida: $x \geq 10$

Determinação das Classes de Equivalência

- Se a entrada especifica um número de valores, então uma classe válida e duas inválidas devem ser identificadas.
 - Condição: imóvel pode possuir de um a seis proprietários
 - Classe válida:
 - de um a seis proprietários
 - Classe inválida:
 - nenhum proprietário
 - mais de 6 proprietários

Determinação das Classes de Equivalência

- Se a entrada especifica um conjunto de valores, e suspeita-se que eles são tratados de maneira diferente, então deve ser identificada uma classe válida para cada valor e uma única classe inválida.
 - Condição: veículo deve ser: ônibus, caminhão, táxi, veículo de passeio ou motocicleta.
 - Classes válidas: ônibus, caminhão, táxi, veículo de passeio e motocicleta.
 - Classe inválida: trailer.

Determinação das Classes de Equivalência

- Se a entrada especifica uma determinada situação, devem ser identificadas uma classe válida e uma classe inválida.
 - Condição: primeiro caractere de um identificador deve ser uma letra.
 - Classe válida: primeiro caractere igual a letra.
 - Classe inválida: primeiro caractere diferente de letra.

Determinação das Classes de Equivalência

- Se uma entrada especifica uma condição booleana, então uma classe válida e uma inválida devem ser selecionadas.
 - Condição: os valores de entrada são inteiros positivos.
 - Classe válida: valor de entrada > 0
 - Classe inválida: valor de entrada ≤ 0

Particionamento em Classes de Equivalência

- Passos

1. Identifique as classes de equivalência.
2. Atribua um número único a cada classe de equivalência identificada.
3. Até que todas as classes de equivalência **válidas** tenham sido cobertas, escreva um novo caso de teste incluindo o maior número possível de classes **válidas** que ainda não foram cobertas.
4. Até que todas as classes de equivalência **inválidas** tenham sido cobertas, escreva um caso de teste para cada uma, e somente uma, classe **inválida** não coberta.

Particionamento em Classes de Equivalência

- Especificação do programa *Identifier*:

O programa deve determinar se um identificador é válido ou não em *Silly Pascal* (uma estranha variante do Pascal). Um identificador válido deve começar com uma letra e conter apenas letras ou dígitos. Além disso, deve ter no mínimo 1 caractere e no máximo 6 caracteres de comprimento.

- Exemplo:

abc12 (válido)
cont*1 (inválido)

1soma (inválido)
a123456 (inválido)

Particionamento em Classes de Equivalência

- Classes de equivalência

Condições de Entrada	Classes Válidas	Classes Inválidas
Tamanho t do identificador	$1 \leq t \leq 6$ (1)	$t > 6$ (2) $t < 1$ (3)
Primeiro caractere c é uma letra	Sim (4)	Não (5)
Só contém caracteres válidos	Sim (6)	Não (7)

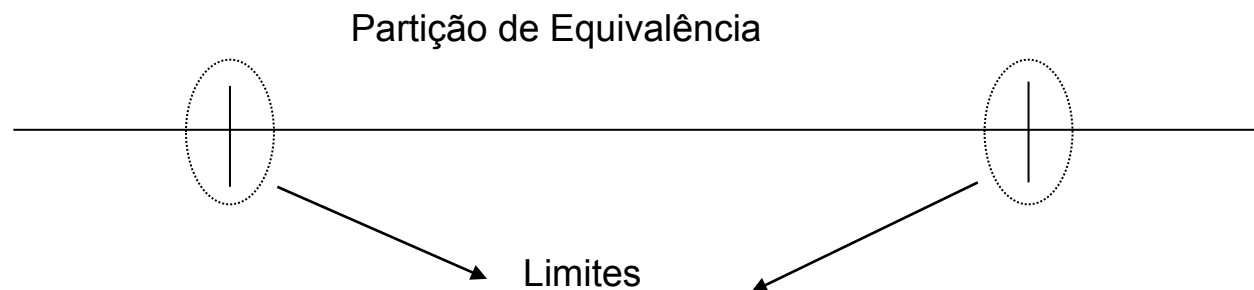
- Exemplo de Conjunto de Casos de Teste

$T_0 = \{(a1, \text{Válido}), (2B3, \text{Inválido}), (Z-12, \text{Inválido}), (A1b2C3d, \text{Inválido}), (\text{vazio}, \text{Inválido})\}$

(1, 4, 6) (5) (7) (2) (3)

Análise do Valor Limite

- Complementa o Particionamento de Equivalência.
 - Fonte propícia a erros – os **limites** de uma classe ou partição de equivalência.



Determinação dos Limites

- Se uma entrada especifica uma faixa limitada pelos valores x e y , então casos de teste devem ser projetados com os valores x e y e também com valores imediatamente inferiores e superiores a x e y .
 - Se a faixa de entrada é $-1.0 < t < 1.0$, devem ser criados casos de teste com valores de t igual a -1.0 , 1.0 , -1.1 , 1.1 .

Determinação dos Limites

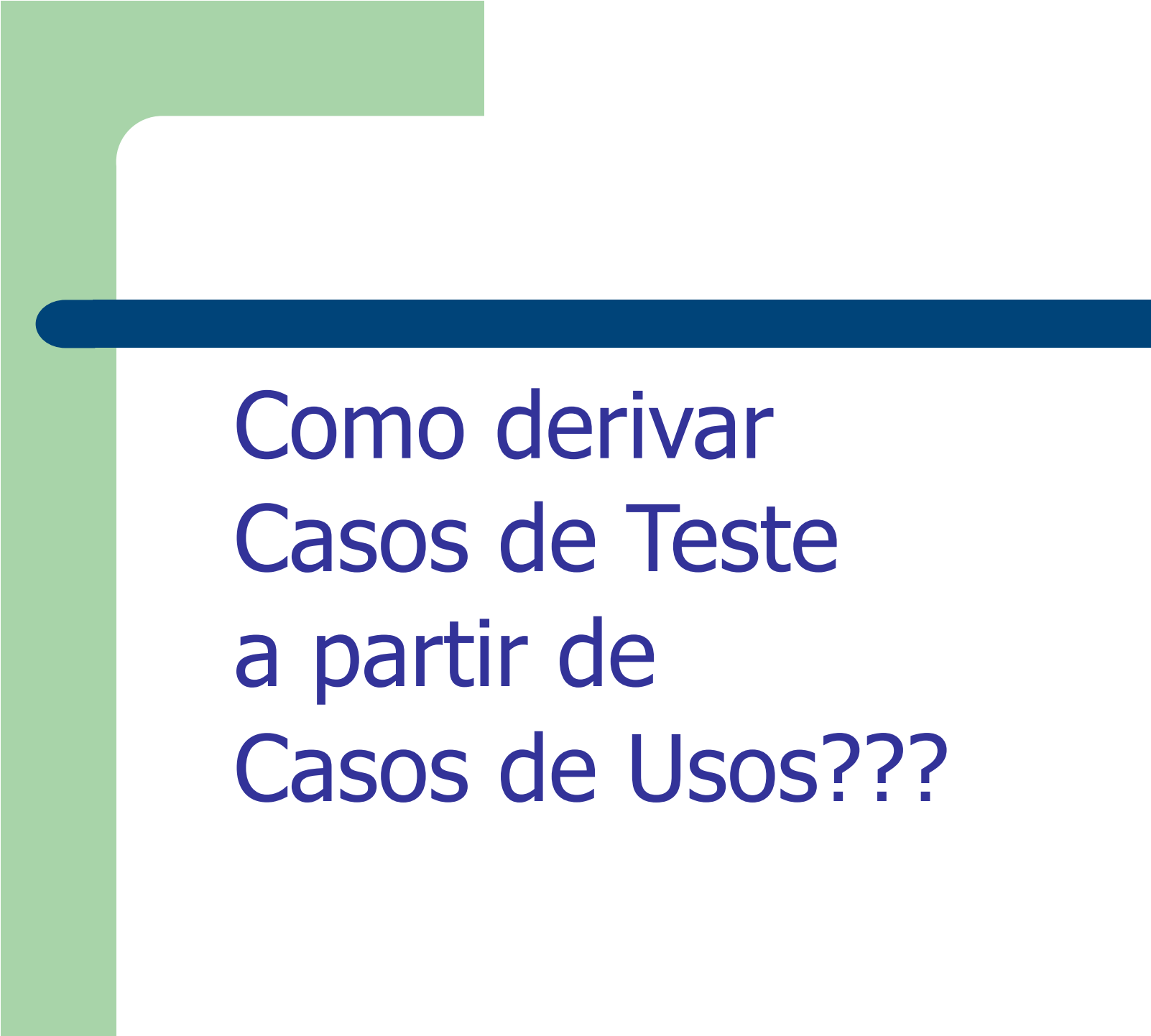
- Se uma entrada especifica um número de valores, então...
 - Casos de teste devem ser derivados para se exercitar o número máximo e o número mínimo de valores.
 - Valores imediatamente acima e abaixo do número máximo e mínimo também devem ser exercitados por casos de teste.
 - Se um arquivo de entrada pode conter 1 a 25 registros, escreva casos de teste com 0, 1, 25 e 26 registros.

Análise do Valor Limite

- Limites

Condições de Entrada	Classes Válidas	Classes Inválidas
Tamanho t do identificador	$1 \leq t \leq 6$ (1)	$t > 6$ $t < 1$ (2) (3)
Primeiro caractere c é uma letra	Sim (4)	Não (5)
Só contém caracteres válidos	Sim (6)	Não (7)

- Tamanho do identificador.
 - Os valores 0, 1, 6 e 7.



Como derivar
Casos de Teste
a partir de
Casos de Usos???

Casos de Teste e Casos de Uso

- Para derivar casos de teste a partir de casos de uso, utilizam-se os **cenários** desenvolvidos para os casos de uso com técnicas de **teste funcionais**.
 - Técnicas Funcionais:
 - Particionamento em classes de equivalência.
 - Análise do valor limite.

Sacar Dinheiro do Caixa Eletrônico

- **Caso de Uso:** Sacar Dinheiro
- **Ator principal:** Cliente do *BancoXYZ*
- **Tipo:** primário, essencial
- **Descrição:** Cliente cadastrado no banco fornece senhas no caixa eletrônico e saca dinheiro caso tenha saldo na conta.

Sacar Dinheiro do Caixa Eletrônico

- **Pré-condições:** Cliente possui conta no *BancoXYZ*; senhas de acesso cadastradas.
- **Pós-condições:** Saldo devidamente atualizado.
- **Regra de negócios:** Valores de saque devem ser múltiplos de 5, sem centavos.

Sacar Dinheiro do Caixa Eletrônico

- **Sequência típica de eventos:**
 1. Cliente insere o cartão no caixa eletrônico e introduz primeira senha.
 2. Sistema aprova conta descrita no cartão e a senha introduzida.
 3. Cliente seleciona valor a ser sacado.
 4. Sistema verifica que há saldo na conta e solicita segunda senha.
 5. Cliente fornece segunda senha.
 6. Sistema aprova senha e fornece o valor solicitado.

Sacar Dinheiro do Caixa Eletrônico

- **Sequências alternativas**
 - A1 – Linha 1: Primeira senha incorreta. Emitir mensagem de erro.
 - A2 – Linha 5: Segunda senha incorreta. Emitir mensagem de erro.
 - A3 – Linha 4: Falta de saldo. Emitir mensagem de erro.

Casos de Teste e Casos de Uso

■ Passos

1. Identificar as condições de entrada do caso de uso.
2. Identificar os cenários.
3. Para cada cenário desenvolver casos de teste variando as classes de equivalência e os valores limites, quando possível.
4. Adicionar valores para os casos de teste.

Casos de Teste e Casos de Uso

■ Passos

1. Identificar as condições de entrada do caso de uso.
2. Identificar os cenários.
3. Para cada cenário desenvolver casos de teste variando as classes de equivalência e os valores limites, quando possível.
4. Adicionar valores para os casos de teste.

Casos de Teste e Casos de Uso

- **Condições de entrada** para o caso de uso Sacar Dinheiro do Caixa Eletrônico:
 - **Conta:** válida e inválida.
 - **Senha 1:** válida e inválida.
 - **Senha 2:** válida e inválida.
 - **Saque:** $< \text{saldo}$; $= \text{saldo}$; e $> \text{saldo}$.
 - **Saldo:** > 0 ; < 0 e $= 0$.

Casos de Teste e Casos de Uso

■ Passos

1. Identificar as condições de entrada do caso de uso.
2. Identificar os cenários.
3. Para cada cenário desenvolver casos de teste variando as classes de equivalência e os valores limites, quando possível.
4. Adicionar valores para os casos de teste.

Casos de Teste e Casos de Uso

- **Cenários** para o caso de uso Sacar Dinheiro do Caixa Eletrônico:
 - **Sequência típica de eventos**, *happy day scenario*:
 - Saque bem sucedido.
 - **Sequências alternativas**:
 - Primeira senha incorreta.
 - Segunda senha incorreta.
 - Falta de saldo na conta.

Sacar Dinheiro do Caixa Eletrônico

- **Sequência alternativa A1**
Primeira senha incorreta:
 - 1.1 Cliente insere o cartão no caixa eletrônico e introduz primeira senha.
 - 1.2 Sistema aprova conta descrita no cartão, mas a senha introduzida é incorreta.
 - 1.3 Sistema emite mensagem de erro “senha caixa eletrônico incorreta”.
 - 1.4 Volta para o passo 1 da sequência típica.

Sacar Dinheiro do Caixa Eletrônico

- **Sequência alternativa A2**
Segunda senha incorreta:
 - 2.1 Cliente insere o cartão no caixa eletrônico e introduz primeira senha.
 - 2.2 Sistema aprova conta descrita no cartão e a senha introduzida.
 - 2.3 Cliente seleciona valor a ser sacado.
 - 2.4 Sistema verifica que há saldo na conta e solicita segunda senha.
 - 2.5 Cliente fornece segunda senha.
 - 2.6 Senha fornecida é incorreta e sistema emite mensagem de erro “senha cartão incorreta”.
 - 2.7 Volta para o passo 4 da sequência típica.

Sacar Dinheiro do Caixa Eletrônico

■ Sequência alternativa A3

Falta de saldo na conta:

- 3.1 Cliente insere o cartão no caixa eletrônico e introduz primeira senha.
- 3.2 Sistema aprova conta descrita no cartão e a senha introduzida.
- 3.3 Cliente seleciona valor a ser sacado.
- 3.4 Sistema verifica que não há saldo na conta e emite mensagem de erro “falta de saldo para realizar saque”.
- 3.5 Volta para o passo 1 da sequência típica.

Casos de Teste e Casos de Uso

■ Passos

1. Identificar as condições de entrada do caso de uso.
2. Identificar os cenários.
3. Para cada cenário desenvolver casos de teste variando as classes de equivalência e os valores limites, quando possível.
4. Adicionar valores para os casos de teste.

Projeto de Casos de Teste

Caso Teste	Cenário	Conta	Senha 1	Senha 2	Saldo	Saque	Saída Esperada
1	Seqüência típica – Saque bem sucedido	Válida	Válida	Válida	Saldo > 0	Saque < Saldo	Dinheiro entregue
2		Válida	Válida	Válida	Saldo > 0	Saque = Saldo	Dinheiro entregue
3		Válida	Válida	Válida	Saldo > 0	Saque < Saldo	Dinheiro entregue
4	Seqüência A1 – Senha 1 incorreta	Válida	Inválida	---	---	---	Mensagem – Senha do caixa incorreta
5	Seqüência A2 – Senha 2 incorreta	Válida	Válida	Inválida	Saldo > 0	Saque < Saldo	Mensagem – Senha do cartão incorreta

Projeto de Casos de Teste

Caso de Teste	Cenário	Conta	Senha 1	Senha 2	Saldo	Saque	Saída Esperada
6	Seqüência A3 – Falta de saldo na conta	Válida	Válida	Válida	Saldo > 0	Saque > Saldo	Mensagem – Falta de saldo para saque
7		Válida	Válida	Válida	Saldo > 0	Saque > Saldo	Mensagem – Falta de saldo para saque
8		Válida	Válida	Válida	Saldo = 0	---	Mensagem – Falta de saldo para saque
9		Válida	Válida	Válida	Saldo < 0	---	Mensagem – Falta de saldo para saque

Projeto de Casos de Teste

Caso de Teste	Cenário	Conta	Senha 1	Senha 2	Saldo	Saque	Saída Esperada
6	Seqüência A3 – Falta de saldo na conta	Válida	Válida	Válida	Saldo > 0	Saque > Saldo	Mensagem – Falta de saldo para saque
7		Válida	Válida	Válida	Saldo > 0	Saque > Saldo	Mensagem – Falta de saldo para saque
8		Válida	Válida	Válida	Saldo = 0	---	Mensagem – Falta de saldo para saque
9		Válida	Válida	Válida	Saldo < 0	---	Mensagem – Falta de saldo para saque
10	Seqüência alternativa ausente – conta inválida	Inválida	---	---	---	---	Mensagem – conta inválida

Casos de Teste e Casos de Uso

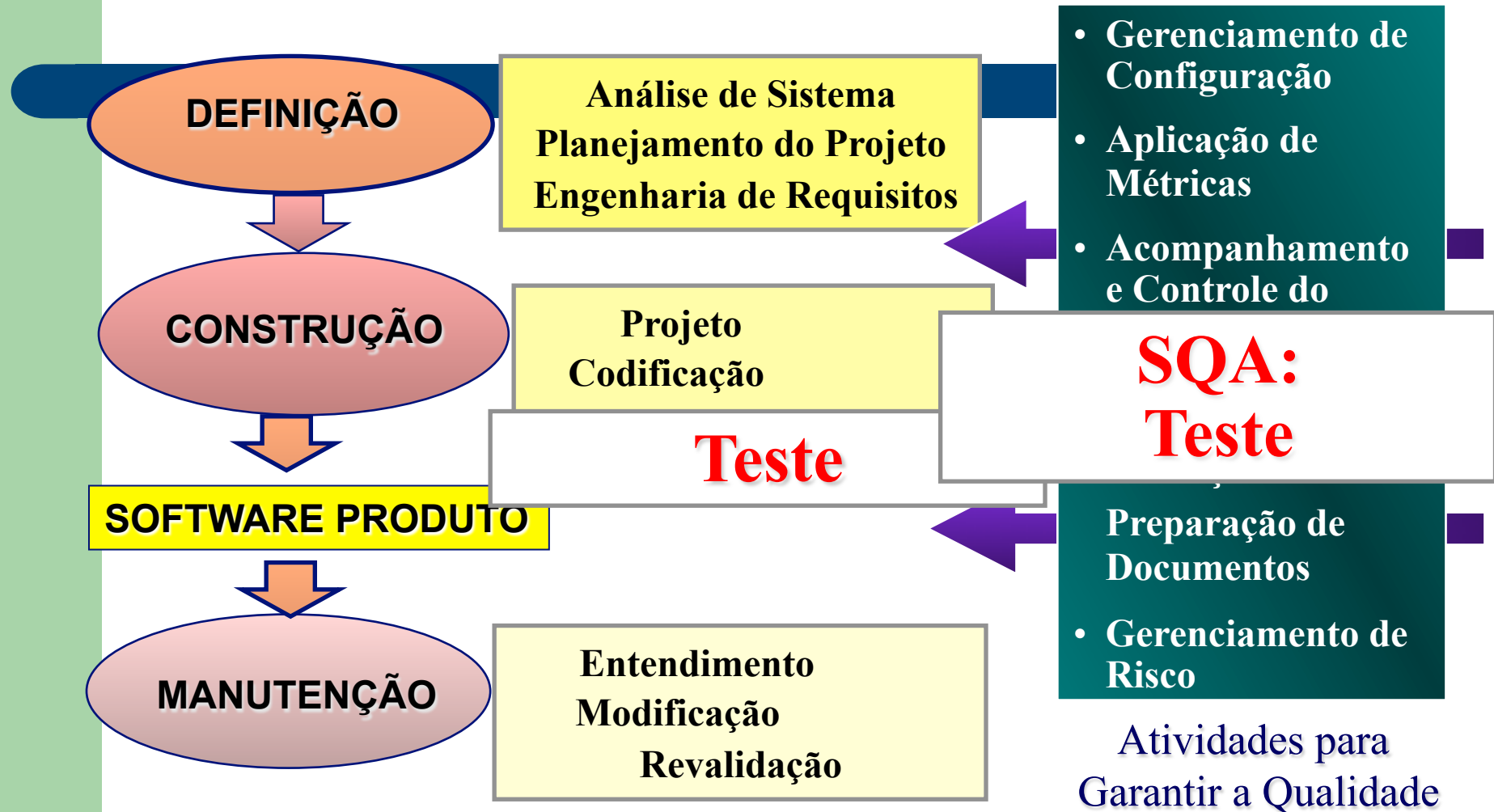
■ Passos

1. Identificar as condições de entrada do caso de uso.
2. Identificar os cenários.
3. Para cada cenário desenvolver casos de teste variando as classes de equivalência e os valores limites, quando possível.
4. Adicionar valores para os casos de teste.

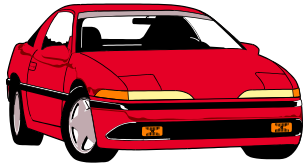
Casos de Teste

Caso de Teste	Cenário	Conta	Senha 1	Senha 2	Saldo	Saque	Saída Esperada
1	Seqüência típica – Saque bem sucedido	Válida (1511-15)	Válida (151101)	Válida (010877)	Saldo > 0 (100)	Saque < Saldo (50)	Dinheiro entregue
2		Válida (1511-15)	Válida (151101)	Válida (010877)	Saldo > 0 (100)	Saque = Saldo (100)	Dinheiro entregue
3		Válida (1511-15)	Válida (151101)	Válida (010877)	Saldo > 0 (100)	Saque < Saldo (99)	Dinheiro entregue
4	Seqüência A1 – Senha 1 incorreta	Válida (1511-15)	Inválida (151111)	---	---	---	Mensagem – Senha do caixa incorreta
5	Seqüência A2 – Senha 2 incorreta	Válida (1511-15)	Válida (151101)	Inválida (010878)	Saldo > 0 (100)	Saque < Saldo (50)	Mensagem – Senha do cartão incorreta

Processo de Software



Exercício



AlugaCar

- **AlugaCar** é uma empresa de aluguel de carros que deseja automatizar o seu sistema de reservas de carros.

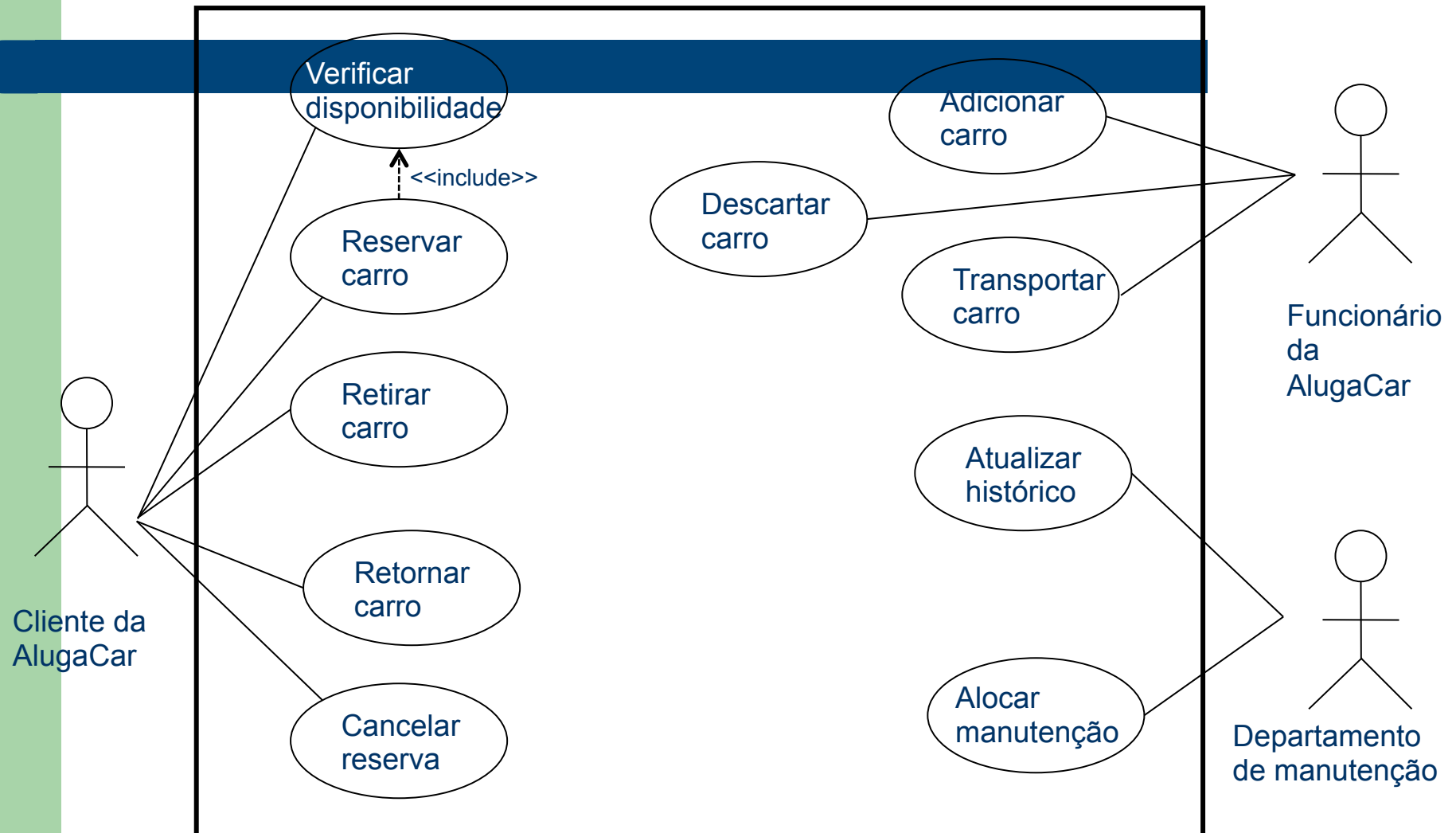
Atores identificados

- Cliente:
 - Indivíduo que deseja alugar um carro da AlugaCar.
- Funcionário da Agência da AlugaCar:
 - Empregado da AlugaCar que utiliza o sistema de reserva de carros.
- Departamento de Manutenção:
 - Departamento responsável pela manutenção preventiva dos carros, pela realização da manutenção e pela atualização do histórico de manutenções dos carros.

Casos de uso identificados

- Reservar carro
- Verificar disponibilidade
- Retirar carro
- Retornar carro
- Cancelar reserva
- Adicionar carro
- Transportar carro para agência
- Alocar manutenção de carro
- Atualizar histórico de revisões
- Descartar carro
- ...

Sistema de reservas da AlugaCar



Reservar carro

- **Nome:** Reservar carro
- **Descrição:** este caso de uso permite ao cliente reservar um carro de sua preferência em uma determinada agência para datas específicas.

Reservar carro

■ Sequência Típica:

1. Cliente solicita reserva de carro e fornece as seguintes informações:
 - agências de retirada e devolução do veículo
 - datas de retirada e devolução do veículo
 - tipo do veículo (classe, marca, modelo)
 - opcionais.
2. Sistema verifica a disponibilidade e informa os carros disponíveis.
3. Cliente seleciona carro.
4. Sistema determina diária do carro.
5. Cliente confirma reserva.
6. Sistema gera dados da reserva (no. da reserva, datas, carro, diárias etc.) para impressão.

Reservar carro

- **Sequência alternativa A1 – tipo de carro não está disponível:**

- 1.1 Cliente solicita reserva de carro e fornece informações sobre o carro desejado.
- 1.2 Sistema verifica a disponibilidade e emite a mensagem de erro “tipo carro desejado não está disponível”.
- 1.3 Volta para o passo 1 da sequência típica.

Reservar carro

- **Sequência alternativa A2 – carro solicitado não está disponível na data de retirada:**
 - 2.1 Cliente solicita reserva de carro e fornece informações sobre o carro desejado.
 - 2.2 Sistema verifica a disponibilidade e emite a mensagem de erro “carro solicitado não está disponível na data desejada para retirada”.
 - 2.3 Volta para o passo 1 da sequência típica.

Reservar carro

- **Sequência alternativa A3 – opcionais desejados não disponíveis:**

- 3.1 Cliente solicita reserva de carro e fornece informações sobre o carro desejado.
- 3.2 Sistema verifica a disponibilidade e emite a mensagem de erro “opcionais desejados não disponíveis”.
- 3.3 Volta para o passo 1 da sequência típica.

Reservar carro

- **Pré-condições:**
 - Cliente cadastrado no sistema.
- **Pós-condições:**
 - Reserva efetuada.
- **Regra de negócios:**
 - Tarifa calculada segundo as regras definidas pela AlugaCar.