

**Universidade de Salvador - Unifacs**  
**Ciências da Computação**

Pedro Henrique Nascimento Luz RA: 12722122396

Vitor Pio Vieira RA: 1272220376

André Neves dos Santos Braga RA: 12722126112

André Filipe Reis Santos RA: 1272212034

Adelirio Alexandre Dos Santos Neto RA:12722127337

Vitor Reis dos Santos - 12722124265

**TransSalvador**

Salvador - BA  
2024

Pedro Henrique Nascimento Luz RA: 12722122396

Vitor Pio Vieira RA: 1272220376

André Neves dos Santos Braga RA: 12722126112

André Filipe Reis Santos RA: 1272212034

Adelirio Alexandre Dos Santos Neto RA:12722127337

Vitor Reis dos Santos - 12722124265

### **TransSalvador**

Este documento tem como objetivo descrever os testes das Caixas branca e preta, além de apresentar as métricas.

Orientado pelo professor (a): Thiago

Salvador - BA  
2024

## **Sumário**

<b>Relatório de Teste de Caixa Branca: Check de Velocidade.....</b>	<b>4</b>
<b>Relatório de Teste de Caixa Preta: Check de Velocidade.....</b>	<b>5</b>
<b>Métricas.....</b>	<b>6</b>

## Relatório de Teste de Caixa Branca: Check de Velocidade

```
1 function checkSpeedAlert() {  
2   if (speed > 60) {  
3     alert('Excesso de Velocidade! Limite de 60 km/h.');4   } else {  
5     alert('Velocidade dentro do limite.');6   }  
7 }
```

No código acima, temos uma função `checkSpeedAlert()`, dentro dela há um `if` com duas condições, a primeira é se a velocidade do veículo é maior que 60, a segunda é se a velocidade é menor ou igual a 60. Para realizarmos o teste de Caixa Branca, criamos 2 condições.

### Caso de teste 1: Velocidade de 70 km/h

Neste caso, a função `checkSpeedAlert()` deve exibir o alerta "Excesso de Velocidade! Limite de 60 km/h."

- Condição de teste: velocidade de 70 km/h.
- Execução da função: A função `checkSpeedAlert()` é chamada com um valor maior que 60km/h.C
- Resultado: A função `alert()` é chamada com a mensagem "Excesso de Velocidade! Limite de 60 km/h."

### Caso de teste 2: Velocidade de 50 km/h

Neste caso, a função `checkSpeedAlert()` deve exibir o alerta "Velocidade dentro do limite."

- Condição de teste: velocidade de 50 km/h.
- Execução da função: A função `checkSpeedAlert()` é chamada com um valor menor que 60km/h.C
- Resultado: A função `alert()` é chamada com a mensagem "Velocidade dentro do limite."

Conclusão:

Então, para ambos os testes, a cobertura de código é de aproximadamente 42.9%

## Relatório de Teste de Caixa Preta: Check de Velocidade

```
function checkSpeedAlert() {  
  if (speed > 60) {  
    alert('Excesso de Velocidade! Limite de 60 km/h.');  } else {  
    alert('Velocidade dentro do limite.');  }  
}
```

### Metodologia de Teste de Caixa Preta

No teste de caixa preta, não consideramos a estrutura interna do código. Em vez disso, focamos nas entradas e saídas esperadas com base nos requisitos. Aqui, os requisitos indicam que a função deve alertar quando a velocidade excede 60 km/h e quando está dentro do limite.

### Casos de Teste

#### Caso de Teste 1: Velocidade de 70 km/h

- **Entrada:** 70 km/h
- **Resultado Esperado:** A função alert() deve exibir a mensagem "Excesso de Velocidade! Limite de 60 km/h."

#### Caso de Teste 2: Velocidade de 50 km/h

- **Entrada:** 50 km/h
- **Resultado Esperado:** A função alert() deve exibir a mensagem "Velocidade dentro do limite."

### Execução dos Testes

#### Caso de Teste 1: Velocidade de 70 km/h

- **Entrada:** 70 km/h
- **Resultado da Execução:** A função checkSpeedAlert() foi chamada com speed = 70. A função alert() exibiu a mensagem "Excesso de Velocidade! Limite de 60 km/h."
- **Resultado:** Passou
- 

#### Caso de Teste 2: Velocidade de 50 km/h

- **Entrada:** 50 km/h

- **Resultado da Execução:** A função `checkSpeedAlert()` foi chamada com `speed = 50`. A função `alert()` exibiu a mensagem "Velocidade dentro do limite."
- **Resultado:** Passou

## Conclusão

Com base nos testes realizados:

1. **Caso de Teste 1 (70 km/h):** Passou, pois a mensagem correta "Excesso de Velocidade! Limite de 60 km/h." foi exibida.
2. **Caso de Teste 2 (50 km/h):** Passou, pois a mensagem correta "Velocidade dentro do limite." foi exibida.

Ambos os casos de teste confirmam que a função `checkSpeedAlert()` está funcionando conforme os requisitos especificados. Portanto, a função se comporta corretamente para as entradas testadas.

## Métricas

As métricas são medidas derivadas ou combinadas para fornecer uma visão mais aprofundada sobre a qualidade do software. Algumas métricas importantes incluem:

1. **Cobertura de Testes:** Percentual de código que é coberto por testes automatizados.

Exemplo Fictício: A cobertura de testes atingiu 85%, indicando que a maioria dos módulos críticos do sistema estão sendo testados regularmente.

2. **Tempo Médio para Correção (MTTR):** Tempo médio necessário para corrigir um defeito.

Exemplo Fictício: O MTTR atual é de 48 horas, refletindo um tempo de resposta eficiente da equipe de desenvolvimento na correção de bugs.

3. **Índice de Manutenibilidade:** Avalia quão fácil é manter, corrigir e melhorar o software.

Exemplo Fictício: O índice de manutenibilidade está em 92%, sugerindo que o código é bem estruturado e documentado, facilitando a realização de futuras manutenções.

4. **Taxa de Defeitos:** Número de defeitos encontrados por unidade de tempo durante a fase de teste.

