UNIFACS – UNIVERSIDADE SALVADOR

BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

**PLANO DE TESTE**

Projeto de cadastro de clientes (Gestão e Qualidade de Software)

Salvador, 2024

**PLANO DE TESTE**

Projeto de cadastro de clientes (Gestão e Qualidade de Software)

Felipe Checcucci Ribeiro Rangel – 12722211197

Luiz Fernando Azevedo Barreto – 12723211455

Samuel Moura da Silva – 12722129904

Matheus Lima Lopes – 12723119077

Wesley Cristian Carvalho Dantas – 1272311443

Davi de Sena Barbosa Silva – 12722130689

Alisson Rayan Santos de Souza – 1272314418

Ludmila Silva Santos – 12724131626

Salvador, 2024

**SUMÁRIO**

**1. Introdução...............................................................................................................4**

**2. Testes Unitários......................................................................................................5**

**2.1 Teste de cadastrar cliente.........................................................................5**

**2.2 Teste de consultar cliente.........................................................................5**

**2.3 Teste de editar cliente...............................................................................6**

**2.4 Teste de listar clientes..............................................................................6**

**2.5 Teste de excluir cliente ............................................................................6**

**2.6 Teste de excluir cliente não existente.....................................................7**

**2.7 Teste de consultar cliente não existente.................................................7**

**2.8 Teste de editar cliente não existente.......................................................7**

**2.9 Teste de listar clientes vazio....................................................................8**

**3. Testes de Integração.............................................................................................9**

**3.1 Teste de integração de cadastro de cliente............................................9**

**4. Requisitos para teste...........................................................................................10**

**4.1 Teste de cadastrar cliente.......................................................................10**

**4.2 Teste de consultar cliente existente......................................................11**

**4.3 Teste de consultar cliente não existente...............................................11**

**4.4 Teste de listar clientes............................................................................11**

**4.5 Teste de integração de Cadastro de Cliente.........................................11**

**5. Métodos de Classes.............................................................................................13**

1. **INTRODUÇÃO**

Este documento descreve os testes realizados no software de cadastro de clientes desenvolvido pela equipe. O propósito dos testes é garantir que o software cumpra os requisitos funcionais e de qualidade estabelecidos, ao mesmo tempo em que detecta e soluciona eventuais defeitos ou questões de desempenho.

Foi utilizado o Maven para o gerenciamento de dependências e automatizar as builds do projeto. O software de cadastro de clientes permite aos usuários cadastrar, consultar,listar e excluir clientes. Para garantir a qualidade e confiabilidade do software, foram realizados testes unitários e testes de integração.

1. **TESTES UNITÁRIOS**

Os testes unitários foram desenvolvidos para verificar o comportamento individual de unidades de código do sistema, como métodos e classes, de forma isolada e simular dependências. Utilizamos a biblioteca JUnit para criar e executar os testes unitáriosde unidades isoladas e Mockito para simular dependências e isolar funcionalidades específicas.

**2.1 Teste de cadastrar cliente**

Este teste verifica se o método CadastrarCliente do programa adiciona corretamente um novo cliente ao sistema.

**2.2 Teste de consultar cliente**

Este teste verifica se o método ConsultarCliente do programa retorna corretamente os detalhes de um cliente existente no sistema.

**2.3 Teste de editar cliente**

Este teste verifica se o método EditarCliente do programa lida corretamente com a edião de dados de um cliente que está cadastrado no sistema.

**2.4 Teste de listar clientes**

Este teste verifica se o método listarClientes do programa retorna corretamente uma lista de todos os clientes cadastrados no sistema.

* 1. **Teste de Excluir cliente**

Este teste verifica se o método ExcluirCliente do programa realiza corretamente a remoção do cliente selecionado do sistema.

**2.6 Teste de Excluir cliente não existente**

Este teste verifica se o método ExcluirCliente verifica corretamente se o cliente está cadastrado no programa para fazer a remoção.

* 1. **Teste de consultar cliente não existente**

Este teste verifica se o método ConsultarCliente do programa retorna corretamente os detalhes de um cliente existente no sistema.

* 1. **Teste editar cliente não existente**

Este teste verifica se o método EditarCliente do programa lida corretamente com a edição de dados de um cliente que está cadastrado no sistema.

* 1. **Teste de listar clientes vazio**

Este teste verifica ListarClientes busca corretamente a quantidade de clientes cadastrados e mostra em número a quantidades de clientes cadastrados no programa.

1. **TESTES DE INTEGRAÇÃO**

Os testes de integração foram realizados para verificar a interação entre diferentes componentes do sistema, incluindo a entrada e saída de dados pelo console.

**3.1 Teste de integração de cadastro de cliente**

Este teste simula a interação do usuário com o sistema, verificando se o programa cadastra corretamente um cliente quando o usuário fornece os dados solicitados.

Como conclusão, podemos afirmar que os testes implementados asseguram que o software de cadastro de clientes funcione de acordo com as expectativas, cumprindo os critérios de qualidade e facilidade de uso estabelecidos. Qualquer problema identificado durante os testes foi prontamente corrigido pela equipe de desenvolvimento, assegurando a entrega de um produto de excelência aos usuários.

1. **REQUISITOS PARA TESTE**

**4.1 Teste de cadastrar cliente**

Objetivo: Verificar se o método cadastrarCliente do programa adiciona corretamente um novo cliente ao sistema.

Passos:

* Criar um objeto Cliente com dados válidos
* Chamar o método cadastrarCliente com o objeto Cliente criado.

Critérios de aceitação: O cliente deve ser adicionado com sucesso ao sistema.

**4.2 Teste de consultar cliente**

Objetivo: Verificar se o método consultarCliente do programa retorna corretamente os detalhes de um cliente existente no sistema.

Passos:

* Simular a consulta de um cliente existente no sistema.

Critérios de aceitação: Os detalhes do cliente consultado devem ser exibidos corretamente.

**4.3 Teste de consultar cliente não existente**

Objetivo: Verificar se o método consultarCliente do programa lida corretamente com a consulta de um cliente que não está cadastrado no sistema.

Passos:

* Simular a consulta de um cliente não existente no sistema.

Critérios de aceitação: O programa deve exibir uma mensagem informando que o cliente não foi encontrado.

**4.4 Teste de listar clientes**

Objetivo: Verificar se o método listarClientes do programa retorna corretamente uma lista de todos os clientes cadastrados no sistema.

Passos:

* Simular a listagem de clientes cadastrados no sistema.

Critérios de aceitação: A lista de clientes cadastrados deve ser exibida corretamente.

**4.5 Teste de integração de Cadastro de Cliente**

Objetivo: Verificar se o programa cadastra corretamente um cliente quando o usuário fornece os dados solicitados.

Passos:

* Simular a entrada de dados do usuário para cadastrar um cliente.

Critérios de aceitação: O cliente deve ser cadastrado com sucesso e o programa deve exibir uma mensagem de confirmação.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classe** | **Método** | **Descrição** | **Automatizado** | **Estágio do Teste** |
| ProgramaTesteUnitario | setUp() | Configuração inicial para os testes. | **Sim** | Pré-condição |
| ProgramaTesteUnitario | testCadastrarCli ente() | Testa o método  cadastrarCliente do Programa. | **Sim** | Teste unitário |
| ProgramaTesteUnitario | testConsultarClie nte() | Testa o método consultarCliente do Programa. | **Sim** | Teste unitário |
| ProgramaTesteUnitario | testEditarCliente(  ) | Testa o método EditarCliente do Programa. | **Sim** | Teste unitário |
| ProgramaTesteUnitario | testExcluirCliente(  ) | Testa o método ExcluirCliente do  Programa. | **Sim** | Teste unitário |
| ProgramaTesteUnitario | testConsultarCliente(  ) | Testa o método ConsultarCliente do Programa. | **sim** | Teste unitário |
| ProgramaTesteUnitario | testListarClientes(  ) | Testa o método ListarClientes do Programa. | **sim** | Teste unitário |
| ProgramaTesteUnitario | testConsultarClienteInexistente(  ) | Testa o método ConsultarClientes do Programa para um  cliente não existente. | **sim** | Teste unitário |
| ProgramaTesteInt egracao | setUp() | Configuração inicial  para os testes de integração. | **Sim** | Pré-condição |
| ProgramaTesteInt egracao | testCRUD() | Faz um teste geral do codigo-fonte | **Sim** | Teste de Integração |