

Complexidade Ciclomática

Critério de Mccabe

Legenda

- Condição Indesejada
- Classe Validação
- Classe Usuário
- Classe UsuarioDAOImpl

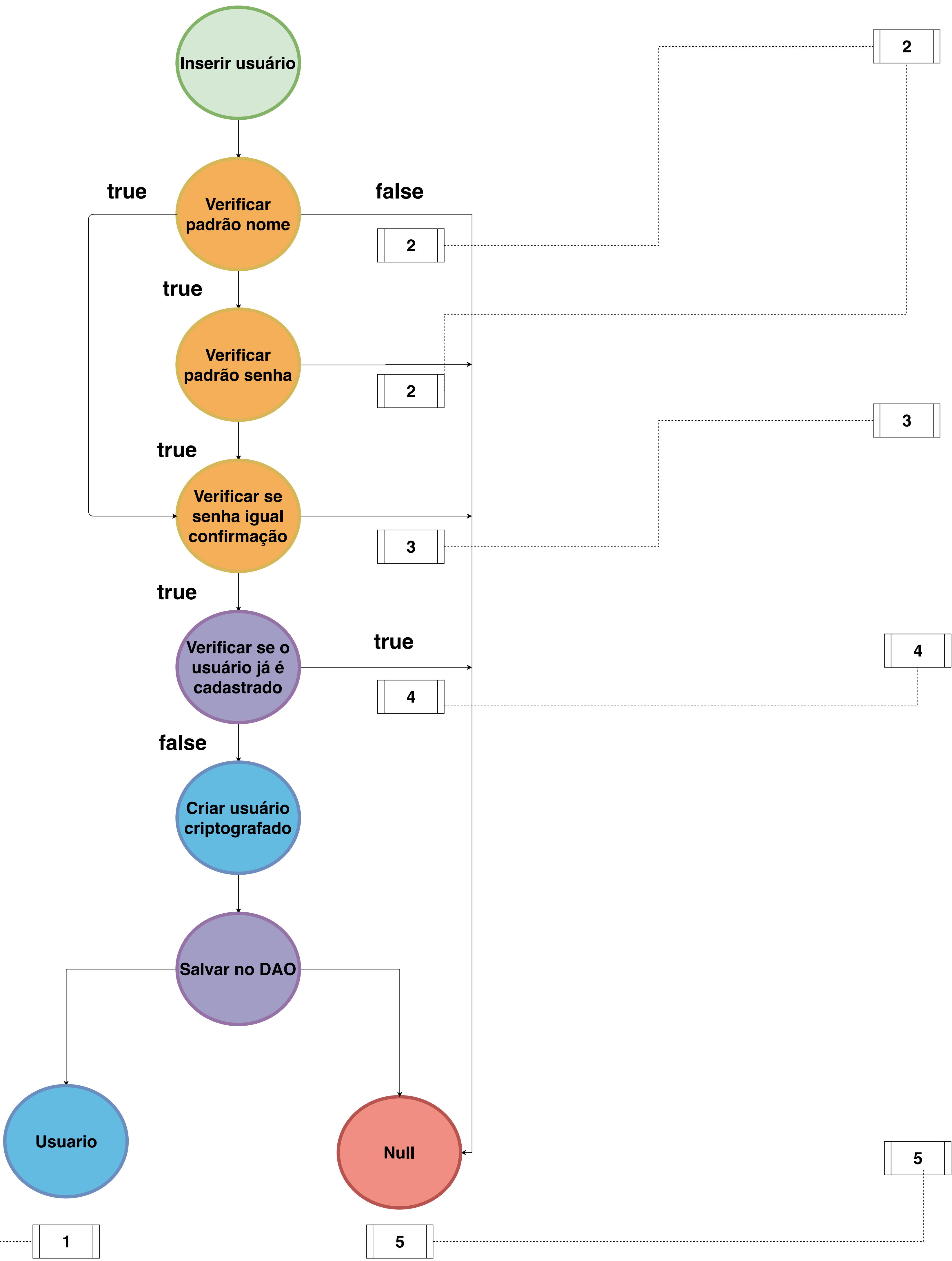
Cálculo

V - número de testes
E - número de ramos
N - número de nós

$V(G) = E - N + 2$

E = 12 ramos
N = 9 nós

V = 12 - 9 + 2
V = 5 testes



```
@Test
Run Test | Debug Test | ✓
public void should_insert_new_user(){
    //given
    when(daoMock.buscarPorNomeUsuario(isA(String.class))).thenReturn(null);
    Usuario u = new Usuario("ixodqr", "vhqkd", null);
    when(daoMock.inserir(isA(Usuario.class))).thenReturn(u);
    servCon.setDAO(daoMock);

    //when
    Usuario uActual = servCon.inserir("fulano", "senha", "senha", null);

    //then
    assertEquals(u, uActual);
    verify(daoMock).inserir(any(Usuario.class));
    verify(daoMock, never()).removerUsuario(any(Usuario.class));
}
```

```
@Test
Run Test | Debug Test | ✓
public void should_not_insert_if_user_already_exists(){
    //given
    Usuario u = new Usuario("ixodqr", "vhqkd", null);
    when(daoMock.buscarPorNomeUsuario(isA(String.class))).thenReturn(u);
    servCon.setDAO(daoMock);

    //when
    Usuario uActual = servCon.inserir(u);

    //then
    assertNull(uActual);
    verify(daoMock).buscarPorNomeUsuario(any(String.class));
}
```

```
@Test
Run Test | Debug Test | ✓
public void should_not_insert_user_if_password_differs(){
    //when
    Usuario uActual = servCon.inserir("anaMaria", "loveMe", "senha", null);

    //then
    assertNull(uActual);
    verify(daoMock).inserir(any(Usuario.class));
}
```

```
@Test
Run Test | Debug Test | ✓
public void should_not_insert_if_user_already_exists(){
    //given
    Usuario u = new Usuario("ixodqr", "vhqkd", null);
    when(daoMock.buscarPorNomeUsuario(isA(String.class))).thenReturn(u);
    servCon.setDAO(daoMock);

    //when
    Usuario uActual = servCon.inserir(u);

    //then
    assertNull(uActual);
    verify(daoMock).buscarPorNomeUsuario(any(String.class));
}
```

```
@Test
Run Test | Debug Test | ✓
public void should_not_insert_user_if_dao_fails(){
    //given
    Usuario u = new Usuario("ixodqr", "vhqkd", null);
    when(daoMock.buscarPorNomeUsuario(isA(String.class))).thenReturn(null);
    when(daoMock.inserir(isA(Usuario.class))).thenReturn(null);
    servCon.setDAO(daoMock);

    //when
    Usuario uActual = servCon.inserir(u);

    //then
    assertNull(uActual);
    verify(daoMock).inserir(any(Usuario.class));
    verify(daoMock, never()).removerUsuario(any(Usuario.class));
}
```