

Testes Funcionais - Evolução de Regras de Negócio

João Paulo Pereira de Medeiros¹

¹Instituto Metr pole Digital – Universidade Federal do Rio Grande do Norte (UFRN)
59.078.970 – Natal – RN – Brazil
joao.pereira.118@ufrn.edu.br

Abstract. *This paper addresses the impact of business rules updates into automated tests regarding UFRN’s approval use case in 2024.1.*

Resumo. *O presente trabalho descreve o impacto que evolu  es em regras de neg cio t m sobre casos de testes automatizados. Para tanto, usou-se como caso de uso, a regra para aprova  o por nota na UFRN em vigor a partir de 2024.1.*

1. Contextualiza  o

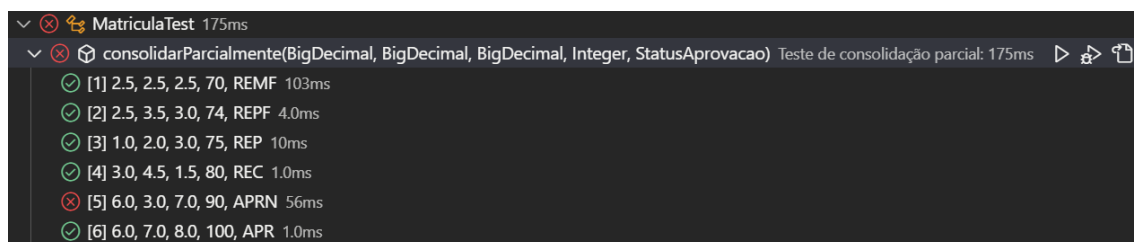
A resolu  o n  016/2023-CONSEPE, de 04 de julho de 2023, instaurou a nova regra para aprova  o por nota na UFRN em vigor a partir de 2024.1. O artigo 114 considera aprovado por nota o estudante com m dia parcial igual ou superior a 6,0 (seis) e com notas iguais ou superiores a 4,0 (quatro) em cada unidade.

Por essa raz o, houve a altera  o da regra de neg cio no c digo fonte do projeto “Matr cula” e, por conseguinte, dos valores para o teste parametriz vel. Com isso, tornou-se n tido o impacto que evolu  es em regras de neg cio possuem acerca dos casos de testes automatizados.

2. Implica  es em casos de testes automatizados

A primeira implica  o foi a quebra do caso de teste com status “APRN”, isto  , aprovado por nota (Figura 01) no arquivo MatriculaTest.java.

Figura 01– Execu  o inadequada do caso de teste



Fonte: o autor (2023)

Uma vez que a nota de uma das unidades era inferior a quatro e a m dia parcial resultava em um valor inferior a seis, o status exibido era “REC” (em recupera  o).

A segunda implica  o da mudan a da regra de neg cios para os testes foi a altera  o da quinta linha do @CsvSource (Figura 02).

Figura 02 – Mudança em CsvSource

```
@ParameterizedTest
@CsvSource({
    "2.5, 2.5, 2.5, 70, REMF",
    "2.5, 3.5, 3.0, 74, REPF",
    "1.0, 2.0, 3.0, 75, REP",
    "3.0, 4.5, 1.5, 80, REC",
    "6.0, 4.0, 8.0, 90, APRN",
    "6.0, 7.0, 8.0, 100, APR"
})
@DisplayName("Teste de consolidação parcial")
```

Fonte: o autor (2023)

Dessa forma, os critérios especificados foram atendidos e o teste realizou a asserção do status “APRN”, conforme deveria.

3. Implicações em código-fonte

A classe *Matricula.java* teve como alteração o método *estaAprovadoNota()*. Este passou a considerar diferentes *thresholds*. Além disso, foram adicionados *static final attributes* contendo os valores quatro e seis como *BigDecimal* no início do arquivo.

Figura 03 – Mudança em método de Matricula.java

```
/**
 * É considerado aprovado, quanto à avaliação do rendimento acadêmico, o estudante que tem média parcial igual ou superior a 6,0 (seis)
 * rendimento acadêmico igual ou superior a 4,0(quatro) em todas as unidades.
 * @return Boolean
 */
public Boolean estaAprovadoPorNota()
{
    return this.mediaParcial.compareTo(SEIS) >= 0 &&
        this.nota1.compareTo(QUATRO) >= 0 && this.nota2.compareTo(QUATRO) >= 0 && this.nota3.compareTo(QUATRO) >= 0;
}
```

Fonte: o autor (2023)

References

CONSEPE. *Resolução nº 0166/2023-CONSEPE, de 04 de julho de 2023*. UFRN. 2023.