

# Melhoria no Processo de Teste Funcional no Desenvolvimento de Um Software

Pedro Antônio C. Sousa<sup>1</sup>

<sup>1</sup>Centro Universitário de Anápolis – Unievangélica  
Caixa Postal 122 e 901 – 75.083-515 – Anápolis – GO – Brazil

pedroacs14@gmail.com

**Abstract.** *This technical report will have as main subject to talk about the importance of the functional test in a software development that involves the quality of the delivery to the customer, how this test is treated in the FTT (Turing Technologies Factory), that possible tools could help in this specific test and finally treat an improvement that can be implemented in it so that it can help in the tests that will be done later.*

**Keywords:** *Software Quality, Functional Test, FTT, Verification and Validation.*

**Resumo.** *Este relatório técnico vai ter como assunto principal falar sobre a importância do teste funcional em um desenvolvimento de um software que envolve a qualidade da entrega para o cliente, como é tratado esse teste na FTT(Fábrica de Tecnologias Turing), que possíveis ferramentas poderiam ajudar neste teste específico e por fim tratar uma melhoria que poderá ser implementada nela para que possa ajudar nos teste que serão feitos posteriormente.*

**Keywords:** *Qualidade de software, Teste Funcional, FTT, Verificação e Validação.*

## 1. Introdução

Teste de software é a parte o desenvolvimento do software que verifica e valida se ele está de acordo com o que foi proposto a ser feito para o cliente. De acordo com Kalinowski, a Verificação é uma atividade, a qual envolve a análise de um sistema para certificar se este atende aos requisitos funcionais e não funcionais. Já a Validação, é a certificação de que o sistema atende as necessidades e expectativas do cliente. O processo de Validação e Verificação, não são processos separados e independentes.

Com essa definição podemos identificar que teremos requisitos funcionais, então com isso devemos fazer testes que precisam de estratégias e técnicas para que possa fazer os testes funcionais envolvidos naquele caso de uso ou história de usuário que foi definido para aquele requisito.

Para esse relatório técnico terá como objetivo geral falar sobre a importância do teste funcional e comparar como é feito na FTT (Fábrica de tecnologias Turing), e com os objetivos específicos falar sobre o que seria o teste, como é tratado a qualidade de software no desenvolvimento de um software e falar sobre algumas ferramentas que auxiliam nos testes funcionais.

E como finalidade deste relatório trazer algum resultado a FTT (Fábrica de tecnologias Turing) para a melhoria dos testes que são feitos na parte do funcional e com isso chega a questão, Por que os testes funcionais beneficiam na qualidade do produto e traz alguns resultados que ajudam antes da entrega para o cliente?

## **2. Referencial Teórico**

### **2.1 Teste de Software**

De acordo com o Sommerville, o teste é destinado a mostrar que um programa faz o que é proposto a fazer e para descobrir os defeitos do programa antes do uso. Quando se testa o software, o programa é executado usando dados fictícios. Os resultados do teste são verificados à procura de erros, anomalias ou informações sobre os atributos não funcionais do programa.

O processo de teste tem dois objetivos distintos:

1. Demonstrar ao desenvolvedor e ao cliente que o software atende a seus requisitos. Para softwares customizados, isso significa que deve haver pelo menos um teste para cada requisito do documento de requisitos. Para softwares genéricos, isso significa que deve haver testes para todas as características do sistema, além de suas combinações, que serão incorporadas ao release do produto.

2. Descobrir situações em que o software se comporta de maneira incorreta, indesejável ou de forma diferente das especificações. Essas são consequências de defeitos de software. O teste de defeitos preocupa-se com a eliminação de comportamentos indesejáveis do sistema, tais como panes, interações indesejáveis com outros sistemas, processamentos incorretos e corrupção de dados.

### **2.2 Qualidade de software**

De acordo com o Pressman, Conformidade a requisitos funcionais e de desempenho explicitamente declarados, a padrões de desenvolvimento claramente documentados e características implícitas que são esperadas de todo software profissionalmente desenvolvido.

O Sommerville destaca alguns problemas, a avaliação da qualidade de software é um processo subjetivo, em que a equipe de gerenciamento de qualidade precisa usar seu julgamento para decidir se foi alcançado um nível aceitável de qualidade. Ela precisa considerar se o software é adequado para sua finalidade ou não. Trata-se de responder a perguntas sobre as características do sistema. Por exemplo:

1. Durante o processo de desenvolvimento os padrões de programação e documentação foram seguidos?

2. O software foi devidamente testado?

3. O software é suficientemente confiável para ser colocado em uso?

4. O desempenho do software é aceitável para uso normal?

5. O software é útil?

6. O software é bem estruturado e compreensível?

## 2.3 Validação de software

De acordo com o Sommerville, Validação de software ou, mais genericamente, verificação e validação (V&V), tem a intenção de mostrar que um software se adequa a suas especificações ao mesmo tempo que satisfaz as especificações do cliente do sistema. Teste de programa, em que o sistema é executado com dados de testes simulados, é a principal técnica de validação. A validação também pode envolver processos de verificação, como inspeções e revisões, em cada estágio do processo de software, desde a definição dos requisitos de usuários até o desenvolvimento do programa. Devido à predominância dos testes, a maior parte dos custos de validação incorre durante e após a implementação.

Barry Boehm, pioneiro da engenharia de software, expressou sucintamente a diferença entre validação e verificação (BOEHM, 1979):

Validação: estamos construindo o produto certo?

Verificação: estamos construindo o produto da maneira certa?

## 4. Abordagem Proposta / Estudo na FTT

Atualmente na FTT (Fábrica de tecnologias Turing) tem um processo que se baseia em fazer casos de testes que são colocados na ferramenta TestLink, ela tem um foco que envolve desde realizar os casos de teste como nos exemplo da Figura 1, executar eles na própria ferramenta como no exemplo da Figura 2 e gerar relatórios que demonstra se passou não passou na execução como no exemplo da Figura 3.

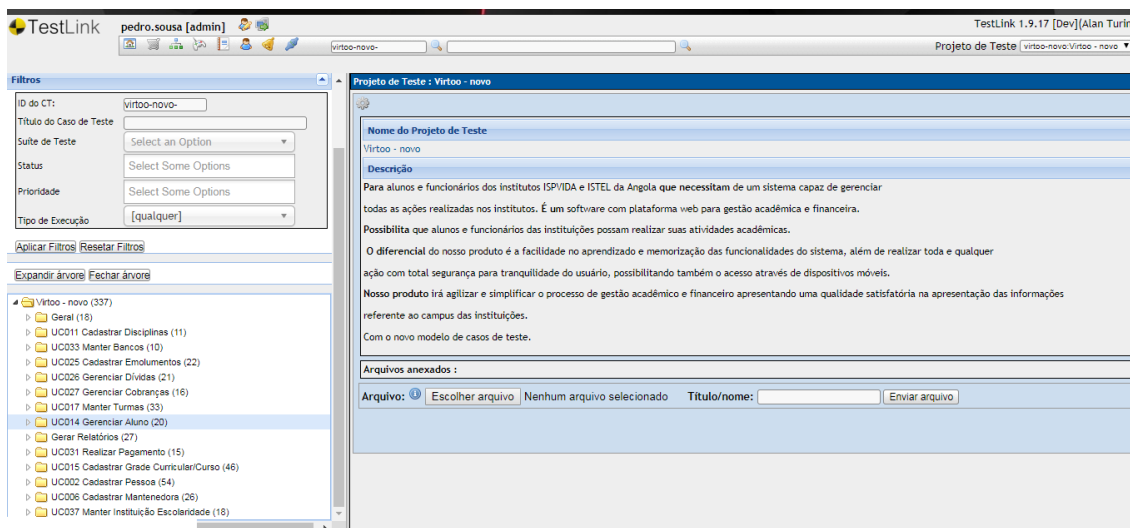


Figura 1. Casos de testes

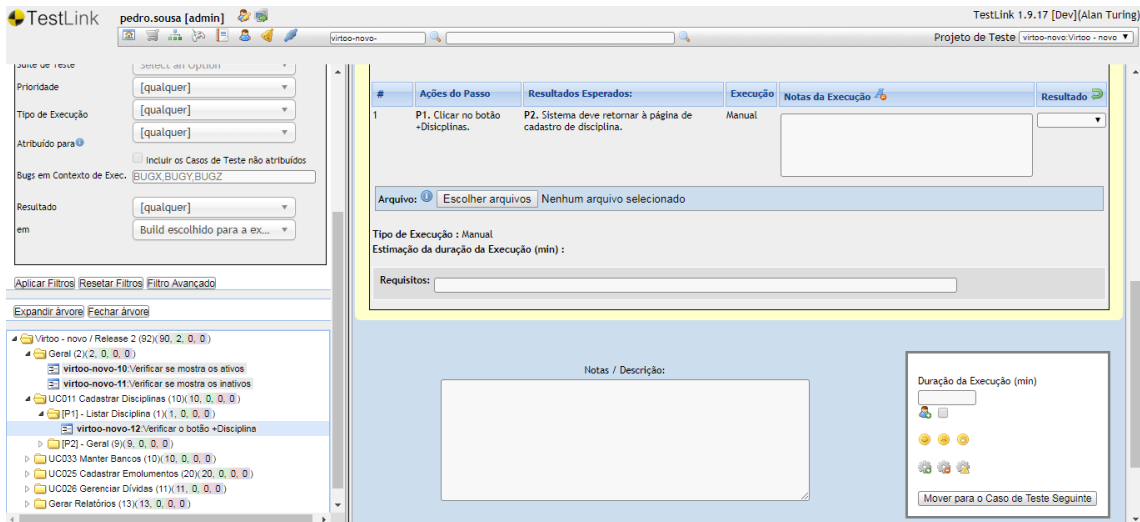


Figura 2. Execução dos casos de teste

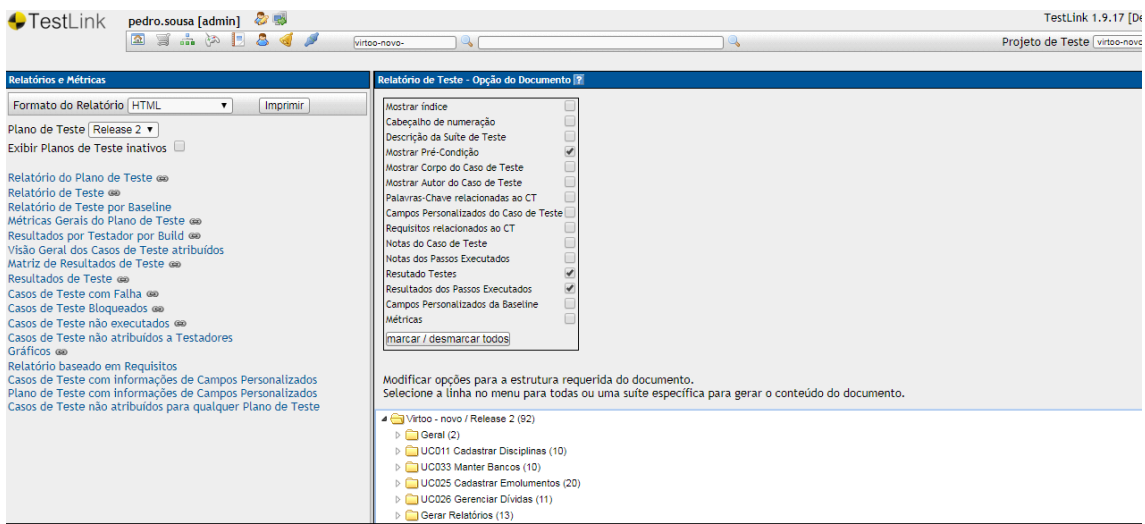


Figura 3. Relatório da execução

Depois que é feita essa parte, se executa o projeto e se baseia nos casos de testes para verificar se está de acordo com a documentação do requisito e finaliza ele para subir para a *branch master*, que fica o projeto aonde o cliente tem acesso. Se precisar executar o mesmo requisito quando for o teste novamente iria perder tempo, então com isso deveria ser usada alguma ferramenta de automação que envolva salvar em scripts os passos que foram feitos, para que depois se precisar só executar eles somente.

Como sugestão utilizar a extensão Katalon Recorder que é uma extensão do navegador que grava cada passo a passo e salva em formato HTML, que é um formato que beneficia todos, pois é simples e não é tão pesado para salvar e exportar para outras pessoas que queiram executar também.

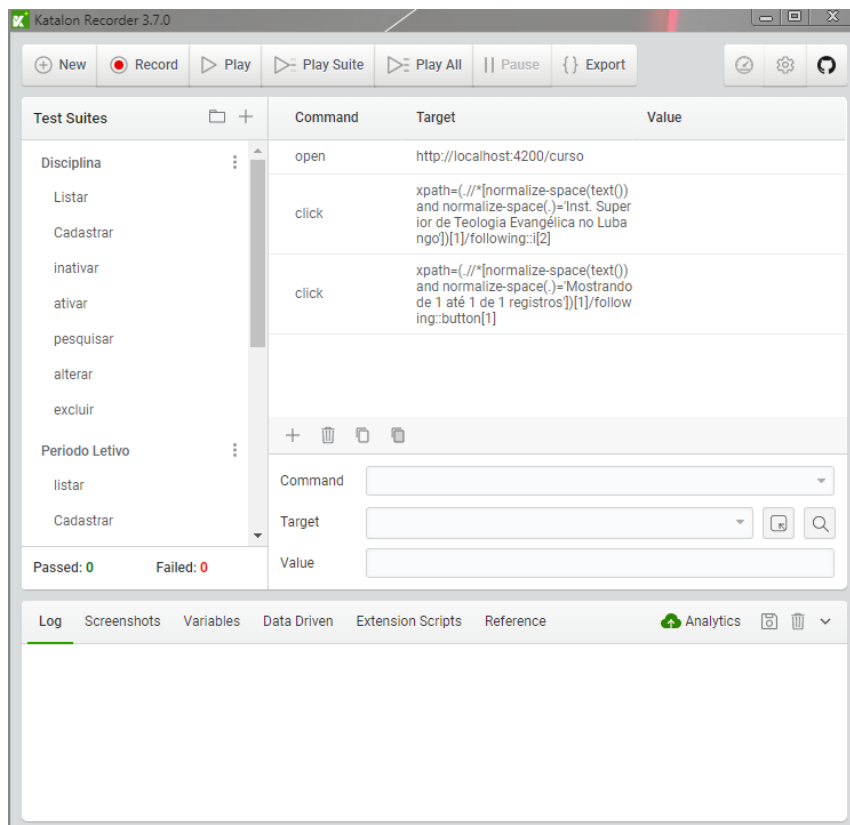


Figura 4. Katalon Recorder

Como segue na figura 4, ele faz a parte de gravar e de executar em fases ou somente um, se ele quiser apenas fazer o cadastro ou todas as funcionalidades juntar cadastrar, alterar, excluir, entre outras.

Para os benefícios de poder fazer o uso dele em requisitos que precisa de outros requisitos para poder gerenciar ele, pois tem dependência de outro. Com essa extensão é possível já fazer os cadastros prévios que ajudam na execução do teste naquele requisito que tem dependência.

## 5. Considerações Finais

Conclui-se que o Katalon Recorder ajudará na hora de precisa fazer os testes funcionais, com o intuito de beneficiar a execução destes possíveis testes que virão. A melhoria que trás com ele faz com o tempo que é gasto na inserção dos mesmos dados, sejam mais rápidos que o normal e com isso trazendo benefícios para o processo de teste funcional.

No contexto da FTT, em um futuro próximo pode-se utilizar os scripts que são gerados por *HTML*, para fazer teste ponta a ponta. Pois quando são realizados os testes no Katalon são gerados conjuntos de instruções para cada caso de teste que poderiam ser usados para agilizar o teste.

## **Referencias Bibliográficas**

KALINOWSKI, M., 2008. Introdução à Inspeção de Software. Engenharia de Software Magazine, Edição 1.

Pressman, Roger s. Engenharia de Software – 1995.

Sommerville, Ian. Engenharia de Software - 8ª Edição 2007.