

UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

CAMPUS ALTO PARAOBEPA – UFSJ/CAP

ENGENHARIA DE TELECOMUNICAÇÕES

**RELATÓRIO Trabalho de Conclusão de Curso i**

Curso de Bacharelado em Engenharia de Telecomunicações

SAMUEL JULIÃO RAMOS

Ouro Branco – 2019



UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

CAMPUS ALTO PARAOBEPA – UFSJ/CAP

ENGENHARIA DE TELECOMUNICAÇÕES

**RELATÓRIO Trabalho de Conclusão de Curso i**

Relatório de Trabalho de Conclusão de Curso I, sendo um dos pré-requisitos para obtenção do grau de Bacharel em Engenharia de Telecomunicações.

**Discente:** Samuel Julião Ramos

**Orientador:** Prof. Dr. Fernando Augusto Teixeira

Ouro Branco – 2019

**Resumo**

O presente relatório tem como objetivo descrever o desenvolvimento da elaboração do Trabalho de Conclusão de Curso I (TCC I), do curso de Engenharia de Telecomunicações, da Universidade Federal de São João Del-Rei – UFSJ. O título do trabalho é Análise de Metodologias e Ferramentas de Testes de Softwares.

O trabalho tem como objetivo expor a importância dos testes de softwares dentro de um processo produtivo de uma aplicação e/ou sistema, demonstrar as metodologias e principais ferramentas utilizadas no mercado atual. E como trabalho futuro, será realizado a execução de uma análise, aplicando as ferramentas de testes em um estudo de caso, coletar os resultados e elaborar um comparativo entre as ferramentas utilizadas.

Sumário

**1 - INTRODUÇÃO …………………………………………………… 5**

**2 - VISÃO GERAL DE TESTES …………………………………… 6**

2.1 - TIPOS DE TESTES ………………………………………….... 6

2.2 - FERRAMENTAS DE TESTES ……………………………….. 7

2.2.1 - SELENIUM…………………………………………..... 7

2.2.2 - KATALON …………………………………………...... 8

2.2.3 - ROBOTIUM …………………………………………... 8

**3 - CRONOGRAMA …………………………………………………. 9**

**4 - CONCLUSÃO …………………………………………………..... 10**

**5 - REFERÊNCIAS…………………………………………………... 11**

**6 - ANEXO ................................................................................... 12**

**ANEXO A ................................................................................. 13**

1 INTRODUÇÃO

Nos dias de atuais, os softwares definem comportamentos dos quais nossa civilização depende, sistemas como roteadores de rede, financeiros*,* redes elétricas, sistemas de transporte, e sistemas essenciais de comunicações, comando e controle de serviços. Nas últimas duas décadas, a indústria de software tornou-se muito maior, está mais competitivo, e mais usuários. Software é um essencial componente de aplicações embutidos em aeronaves, sistemas de controle de tráfego, bem como, aparelhos domésticos, como televisores, micro-ondas, telefones celulares. [1]

Com base nesse cenário, pode se observar que os softwares tornaram se peça fundamental para o funcionamento de serviços de extrema importância para a sociedade e os usuários cada vez mais otimista, assumindo que os softwares nunca iram falhar. Com isso, a Engenharia de Software teve a necessidade de evoluir os processos de desenvolvimento e testes de softwares. Foram criados metodologias de desenvolvimento de software baseados em testes, como TDD (*Test Driven Development*) [4]. Tudo para que, os usuários continuem otimistas na utilização dos sistemas e dispositivos, sem a presença de erros e funcionando de maneira como esperado.

Com o objetivo de explorar e demonstrar a importância da realização e implementação de testes de softwares nos etapas de desenvolvimentos de sistemas e aplicações, buscando a melhoria na qualidade e na diminuição do retrabalho no processo de desenvolvimento. Esse presente trabalho irá fazer uma análise das principais ferramentas utilizadas no mercado atualmente, fazendo comparativos e aplicando - as em um teste de caso. E como contribuição o trabalho irá fornecer o conhecimento das principais ferramentas e metodologias que são baseadas o cenário de testes de softwares atualmente.

# VISÃO GERAL DE TESTES

Os testes podem ser destinados a verificar e validar propriedades diferentes. Casos de testes podem ser projetados para verificar se as especificações funcionais estão implementadas corretamente, o que é conhecido como testes de conformidade, testes de correção ou testes funcionais. No entanto, vários outros propriedades não-funcionais podem ser testadas, incluindo desempenho, confiabilidade e usabilidade, entre muitas outras. [2]

## TIPOS DE TESTES

Além das subdivisões dos testes por requisitos do softwares, como citadas acima, os testes são separados por duas metodologias em que cada requisito pode ser testado de maneira distintas. Essas metodologias são chamadas de teste caixa preta e teste caixa branca.

Teste caixa preta, tem por objetivo determinar defeitos na estrutura interna do produto, através do desenho de testes que exercitem suficientemente os possíveis caminhos de execução.

Teste caixa branca, tem por objetivo determinar se os requisitos foram total ou parcialmente satisfeitos pelo produto. Os testes de caixa preta não verificam como ocorre o processamento, mas apenas os resultados produzidos. [3]

## FERRAMENTAS DE TESTES

As ferramentas de testes foram criadas com o intuito de auxiliar no gerenciamento e execução de testes de softwares, a maioria das ferramentas de testes são utilizadas para automatizar e agilizar os processos de testes dentro de uma linha de produção de softwares.

Existem diversas ferramentas de testes disponíveis no mercado, na elaboração deste trabalho foram elencadas três ferramentas para serem utilizadas, com base na utilização no mercado atualmente, que foram o Selenium, Katalon e Robotium. Ambas ferramentas escolhidas, grandes empresas de desenvolvimento de software do mercado tem utilizado para implementação de testes automatizados.

2.2.1 SELENIUM



Figura 1 – Selenium

Selenium é uma ferramenta bastante popular no mercado. O Selenium IDE, é um add-on de navegador, permite gravar e reproduzir os testes em suas aplicações web. Selenium *WebDriver* ajuda a criar scripts de automação de testes mais complexos e avançados, suportando que os testadores escrevam seus scripts em diversas linguagens de programação como Java, Perl, JavaScript, PHP, Python, C#, Ruby e Groovy. [5]

2.2.2 KATALON



Figura 2 – Katalon

É uma solução para testes automatizados de sistemas Web e Mobile utilizadas por várias empresas no mercado. A ferramenta possui integração com os drivers dos navegadores mais conhecidos como Chrome, Firefox, Internet Explorer, e também permite o teste em tempo real com dispositivo mobile. Usos comuns do Katalon Studio são testes automatizados em aplicações web/mobile, testes de stress, testes de entrada e diversidade de valores. [6]

2.2.3 ROBOTIUM

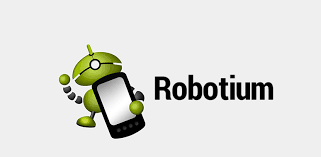


Figura 3 – Robotium

*Framework* para automação de testes para Android e compatível com aplicações nativas e híbridas. O Robotium facilita a gravação de testes robustos automáticos de UI (*User Interface*), baseados na metodologia de teste caixa preta. Voltado para testes de interface sua principal função consiste em simular procedimentos que geralmente são executadas por um analista de teste, buscando a otimização e agilidade na execução dos testes de interface. [7]

**3 CRONOGRAMA**

Como o intuito de programar e estabelecer datas para os próximos passos a serem realizados na elaboração do trabalho de conclusão de curso, foi criado uma tabela de cronograma a partir da lista de atividades a seguir.

1. Definir o estudo de caso para ser utilizado como prova de conceito para análise da ferramenta de teste;
2. Montar um formulário para realizar entrevista com especialistas da área de testes que estão atuando no mercado. Com o objetivo de buscar informações sobre o cenário atual do mercado de teste e definir uma ferramenta a partir da indicação dos próprios especialistas.
3. Realizar contato com os especialistas para combinar a entrevista;
4. Definir duas ferramentas de testes a serem utilizadas no estudo de caso;
5. Montar infraestrutura de teste, preparar todo o ambiente de teste em uma máquina;
6. Iniciar a atividade de implementação do teste no estudo de caso;
7. Definir métricas de comparação entre as ferramentas, realizar experimentos comparativos e coletar os dados a partir dos resultados;
8. Fechar a primeira versão do texto;
9. Revisar e entregar para a banca o texto e montar apresentação;
10. Apresentação do trabalho para a banca.

|  |  |  |  |
| --- | --- | --- | --- |
| **CRONOGRAMA** | | | |
| **Lista de Atividades** | **Previsão de Entrega** | **Lista de Atividades** | **Previsão de Entrega** |
| Definir estudo de caso | Julho | Implementação dos testes | Setembro |
| Montar formulário para entrevistas | Julho/Agosto | Definir métricas | Setembro |
| Estabelecer contato e realizar entrevistas | Agosto | Fechar primeira versão do texto | Outubro |
| Definir ferramentas de testes | Agosto | Revisar e entregar o texto. Montar apresentação | Outubro/Novembro |
| Montar infraestrutura | Agosto/Setembro | Apresentação do trabalho | Dezembro |

# 4 conclusão

Durante a elaboração desse relatório, foi definido e mostrado a importância de um dos principais conceitos que compõem a Engenharia de Software, que são os testes. Demonstrado as principais ferramentas de testes automatizados que estão sendo utilizados atualmente no mercado. Com o objetivo de ressaltar a importância da qualidade de emprega de um software para o usuário final. De forma a aumentar a confiabilidade nas etapas de desenvolvimento e diminuição de retrabalho no processos produtivos de softwares.

Os próximos passos para na elaboração do trabalho de conclusão de curso, foram listadas e montado uma tabela de cronograma com as datas de previsão de realização no item 3 desse presente relatório. Com isso, estabelecemos as metas a serem atingidas para a finalização do trabalho de a análise das ferramentas e metodologias de testes de softwares.

# REFERÊNCIAS

1 - AMMAN, Paul; OFFUTT, Jeff. **Introduction to Software Testing.** Cambridge University Press, 2017.

2 - BOURQUE, Pierre; FAIRLEY, Richard. **Swebok – Guide to the Software Engineering Body of Knowledge**, IEEE, 2014.

3 - FILHO, Wilson de Pádua P. **Engenharia de Software – Fundamentos, Métodos e Padrões,** LTC – Livros Técnicos e Científicos Editora, 2003.

4 – DOOLEY, John F. **Software Development, Design and Conding With Patterns, Debugging, Unit Testing and Refactoring,** Apress, 2017.

5 - **Selenium.** Disponível em: <https://www.seleniumhq.org/> . Acesso em: 10 de Junho de 2019.

6 - **Katalon.** Disponível em: <https://www.katalon.com/> . Acesso em: 10 de Junho de 2019.

7 - **Robotium.** Disponível em: [http://www.robotium.org](http://www.robotium.org/) . Acesso em: 10 de Junho de 2019.

**ANEXO**



UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL-REI

CAMPUS ALTO PARAOBEPA – UFSJ/CAP

ENGENHARIA DE TELECOMUNICAÇÕES

FOLHA DE AVALIAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO I

NOTA:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Prof. Dr. Fernando Augusto Teixeira** Universidade Federal de São João Del-Rei

Professor Orientador