



UEPB

**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS CAMPINA GRANDE
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO COMPUTAÇÃO
CURSO DE CIÊNCIA DA COMPUTAÇÃO**

VIVIANE SILVA HONORATO

**A IMPORTÂNCIA DOS TESTES MANUAIS E AUTOMATIZADOS EM SISTEMAS
CRÍTICOS PERANTE UM CENÁRIO PANDÊMICO E REMOTO**

**CAMPINA GRANDE
2022**

VIVIANE SILVA HONORATO

**A IMPORTÂNCIA DOS TESTES MANUAIS E AUTOMATIZADOS EM SISTEMAS
CRÍTICOS PERANTE UM CENÁRIO PANDÊMICO E REMOTO**

Trabalho de Conclusão de Curso (Artigo) apresentado ao Departamento do Curso de Ciência da Computação da Universidade Estadual da Paraíba, como requisito parcial à obtenção do título de Bacharelado em Ciência da Computação.

Orientador: Prof. Dr. Wellington Candeia de Araújo.

**CAMPINA GRANDE
2022**

É expressamente proibido a comercialização deste documento, tanto na forma impressa como eletrônica. Sua reprodução total ou parcial é permitida exclusivamente para fins acadêmicos e científicos, desde que na reprodução figure a identificação do autor, título, instituição e ano do trabalho.

H774i Honorato, Viviane Silva.

A importância dos testes manuais e automatizados em sistemas críticos perante um cenário pandêmico e remoto [manuscrito] / Viviane Silva Honorato. - 2022.

21 p.

Digitado.

Trabalho de Conclusão de Curso (Graduação em Computação) - Universidade Estadual da Paraíba, Centro de Ciências e Tecnologia, 2022.

"Orientação : Prof. Dr. Wellington Candeia de Araújo , Coordenação do Curso de Computação - CCT."

1. Análise documental. 2. Pandemia. 3. Homologação de software. I. Título

21. ed. CDD 005.1

VIVIANE SILVA HONORATO

A IMPORTÂNCIA DOS TESTES MANUAIS E AUTOMATIZADOS EM SISTEMAS
CRÍTICOS PERANTE UM CENÁRIO PANDÊMICO E REMOTO

Trabalho de Conclusão de Curso (Artigo)
apresentado ao Departamento do Curso
de Ciência da Computação da
Universidade Estadual da Paraíba, como
requisito parcial à obtenção do título de
Bacharelado em Ciência da Computação.

Aprovada em 01 de Agosto de 2022.

BANCA EXAMINADORA



Prof. Dr. Wellington Candeia de Araújo (Orientador)
Universidade Estadual da Paraíba (UEPB)



Prof. Me. Francisco Anderson Mariano da Silva
Universidade Estadual da Paraíba (UEPB)



Prof. Graduado Vinícius Reuteman Feitoza Alves de Andrade
Universidade Estadual da Paraíba (UEPB)

SUMÁRIO

| | | |
|----------------|--|-----------|
| 1 | INTRODUÇÃO | 5 |
| 2 | TESTES EM SOFTWARE | 5 |
| 2.1 | Testes Automatizados | 7 |
| 2.1.1 | <i>Framework para Testes Automatizados</i> | 8 |
| 2.1.2 | <i>Tipos de Teste Automatizados</i> | 8 |
| 2.1.2.1 | <i>Teste de Unidade</i> | 8 |
| 2.1.2.2 | <i>Teste de Integração</i> | 9 |
| 2.1.2.3 | <i>Teste de Sistema</i> | 10 |
| 2.2 | Testes Manuais | 11 |
| 2.2.2 | <i>Tipo de Teste Manual</i> | 11 |
| 2.2.2.1 | <i>Teste AD-HOC</i> | 11 |
| 2.2.2.2 | <i>Teste Exploratório</i> | 12 |
| 2.2.2.3 | <i>Teste de Usabilidade</i> | 12 |
| 3 | SISTEMA CRÍTICO EM CENÁRIO PANDÊMICO E REMOTO | 12 |
| 3.1 | Sistema Crítico | 12 |
| 3.2 | Cenário Pandêmico | 13 |
| 3.3 | Cenário Remoto | 14 |
| 4 | METODOLOGIA | 14 |
| 5 | ANÁLISE | 15 |
| 6 | CONSIDERAÇÕES FINAIS | 18 |
| | REFERÊNCIAS | 20 |

A IMPORTÂNCIA DOS TESTES MANUAIS E AUTOMATIZADOS EM SISTEMAS CRÍTICOS PERANTE UM CENÁRIO PANDÊMICO E REMOTO

THE IMPORTANCE OF MANUAL AND AUTOMATED TESTS IN CRITICAL SYSTEMS IN A PANDEMIC AND REMOTE SCENARIO

Viviane Silva Honorato¹

RESUMO

O propósito deste artigo é ressaltar a importância existente no processo dos testes dentro de sistemas críticos em cenário que contém característica remota e pandêmica. O objetivo é a demonstração dos artefatos de teste relacionados aos cenários propostos, por meio análise vinculada às evidências que trarão ao leitor a adição de conhecimento relacionado ao tema. A metodologia utilizada foi a análise documental, conquistando como referências artigos correspondentes ao tema, episódios sócio-políticos registrados em conteúdo jornalístico, além de livros que obtêm conteúdo relevante aos testes. Como resultado, temos a percepção da indispensável presença de aspectos técnicos para esse tipo de sistemas críticos em cenários pandêmicos e remotos, com a adição de pontos relevantes tais como ambientes de homologação e produção. Com isso, a conclusão se deu entre as somatórias de resultantes dos assuntos abordados, além da verificação destes artefatos ao longo do artigo compreendendo melhor o tema abordado. Adicionando assim, uma somatória na visão referente ao tema de forma relevante.

Palavras-chave: Testes. Análise Documental. Pandemia. Homologação.

ABSTRACT

The goal of this article is to highlight the importance of the many kinds of testing in a scenario that contains a remote and pandemic characteristic. The objective is the demonstration of test artifacts related in some scenarios, through analyzes together with the evidence that will bring the reader additional knowledge about the theme. The methodology used was document analysis, capturing some reference articles corresponding to the theme, socio-political episodes recorded in journalistic content, as well the books that test relevant content. It is apparent that it happens also for technical aspects and for social analysis, because relevant points such as homologation and production environments are present in this study. With this, the conclusion was reached between the sums of results of the topics, viewing the need for tests in the face of the social situation addressed. Adding a summation in the view referring to the theme in a relevant way. The conclusion was made through the sum of the results of topics in this article, as well verification of the situation studied here with a better understanding of the topic. Adding a new view on the theme.

Keywords: Test. Document Analysis. Pandemic. Homologation.

¹ Graduanda em Ciência da Computação pela Universidade Estadual da Paraíba.

viviane.honorato@aluno.uepb.edu.br

Orientador: Prof. Dr. Wellington Candeia de Araújo

1 INTRODUÇÃO

Com a crescente da tecnologia, a demanda por *softwares* que contém requisitos funcionais importantes para a empresa e o cliente tem obtido um progresso considerável, tendo em vista inclusive ações que eram rotineiras e presenciais tornaram-se possíveis de serem realizadas remotamente. Com isso, surgiu a necessidade de sistemas serem cada vez mais seguros para que o cliente obtenha a realização de suas tarefas importantes concluídas com sucesso e segurança.

A partir desse pensamento, surgiu a necessidade de fazer verificações precisas nos sistemas desenvolvidos para obtenção das validações necessárias para o seu ideal funcionamento. Com isso, podemos afunilar e observar que os sistemas que necessitam de uma maior rigidez são justamente os sistemas que obtêm título de crítico.

Avaliando essa situação, o objetivo para a construção deste artigo tem como pontos cruciais o estudo dos acontecimentos que ocorreram no século XXI, mais precisamente a pandemia como também a maior utilização de sistemas que detém atividades que normalmente eram realizadas de forma presencial e física, passando assim para a forma remota. Além disso, este estudo busca ampliar o olhar da importância dos testes sejam estes manuais ou automatizados, em sistemas críticos perante algum cenário que ocorra socialmente, sendo abordado como estratégia o auxílio do ambiente remoto.

Somado isso, à adição de conhecimento sobre os assuntos de teste dentro de ambientes como de homologação e produção também serão tratados neste artigo com a intenção de tornar o cenário o mais próximo possível da realidade do desenvolvedor. Com isso, ao término desse trabalho a finalidade é a uma soma de artefatos que terão ao leitor tanto a visão social quanto a visão técnica do impacto e da importância desses testes em sistemas críticos no cenário pandêmico e remoto, ou seja, o resultado que esse estudo visa trazer vem por meio do acréscimo de conhecimento que o leitor irá ter em relação ao tema, compreendendo melhor o ambiente estudado.

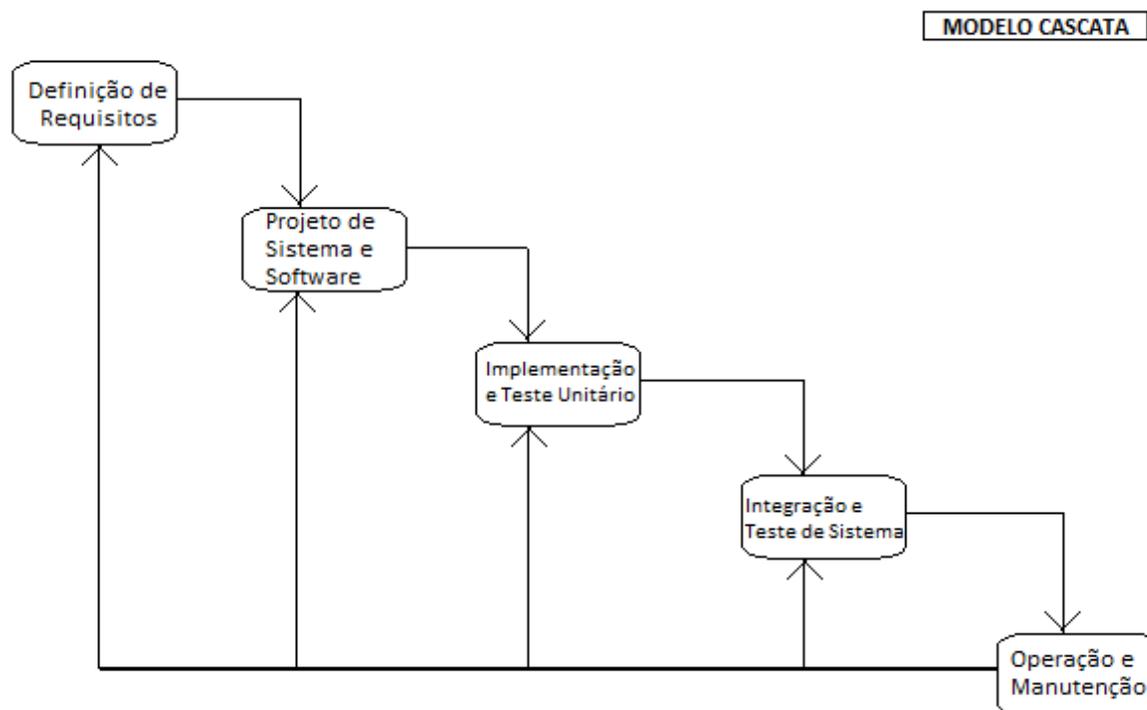
2 TESTES EM SOFTWARE

O *software* é um conjunto de instruções que têm finalidades relacionadas a ações visando a solução de algum impasse, seja este particular do usuário, financeiro com impacto crítico ou até mesmo referente a sociedade no âmbito público, entre diversas outras possibilidades.

A iniciação do desenvolvimento do *software* acontece além do desenvolvimento do código implementado em linguagens de programação, sendo necessário um planejamento em equipe mais detalhado, contando com requisitos do sistema a partir da necessidade do cliente, modelagem do sistema, projeção do sistema e a partir desse momento em diante existirá o desenvolvimento do sistema relacionado aos códigos, seguido dos testes que irão ocorrer no *software*, além da devida manutenção a longo prazo.

É possível observar que fatores citados acima são válidos em vários tipos de modelo para desenvolvimento de *software*, inclusive para Sommerville (2007) onde o modelo cascata exemplificado na Figura 1 faz referência ao seu conteúdo.

Figura 1 – Modelo em Cascata



Fonte: Adaptado de Sommerville, (2007).

Para exercer o seu propósito em funcionamento solicitado pelo cliente, em sua documentação existem as especificações necessárias para a construção do sistema, entre elas estão os "requisitos funcionais" que tem como objetivo detalhar ações necessárias para que o sistema funcione corretamente diante ao que o usuário demanda. Por meio do desenvolvimento do *software* perante o planejamento e análise dos desenvolvedores e engenheiros, é de extrema importância verificar se esses requisitos foram atendidos com o objetivo do produto final ter a usabilidade e eficácia indispensável que o cliente deseja com a adição da qualidade requerida (CUNHA, 2022).

Com isso tornou-se fundamental criar uma maneira que fosse possível fazer a validação e verificação do sucesso do *software* e suas funções, para que isso ocorra se faz essencial a detecção de defeitos, erros e falhas existentes em determinadas operações, sendo estes de fácil, média e difícil localização. Por esse motivo, o teste de *software* é imprescindível para a realização da aceitação do sistema.

Segundo Sommerville (2007, p. 144), o teste de *software* é a maneira de visualizar se o sistema está correspondendo a necessidade que o usuário tem, ou seja, se o sistema faz o que está proposto a fazer e que, além disso, essas validações e verificações são feitas com dados fictícios. Mas o que difere a validação da verificação? De acordo com Boehm (1979), a validação é fruto do questionamento se o que estamos construindo é o produto certo, enquanto a verificação questiona se estamos construindo o produto da maneira certa.

Desse modo, a estratégia utilizada neste tipo de teste é basicamente verificar os fluxos pertencentes a cada requisito funcional ou até mesmo unidades, sendo que esses procedimentos podem ser automatizados por um *framework* controlando o sistema e suas execuções ou manuais onde o desenvolvedor faz uma exploração

no *software* de maneira manual para detectar algum tipo de *bug* ou confirmar o sucesso da aplicação, neste último a experiência é uma importante habilidade que conduz o modo em que o teste será realizado.

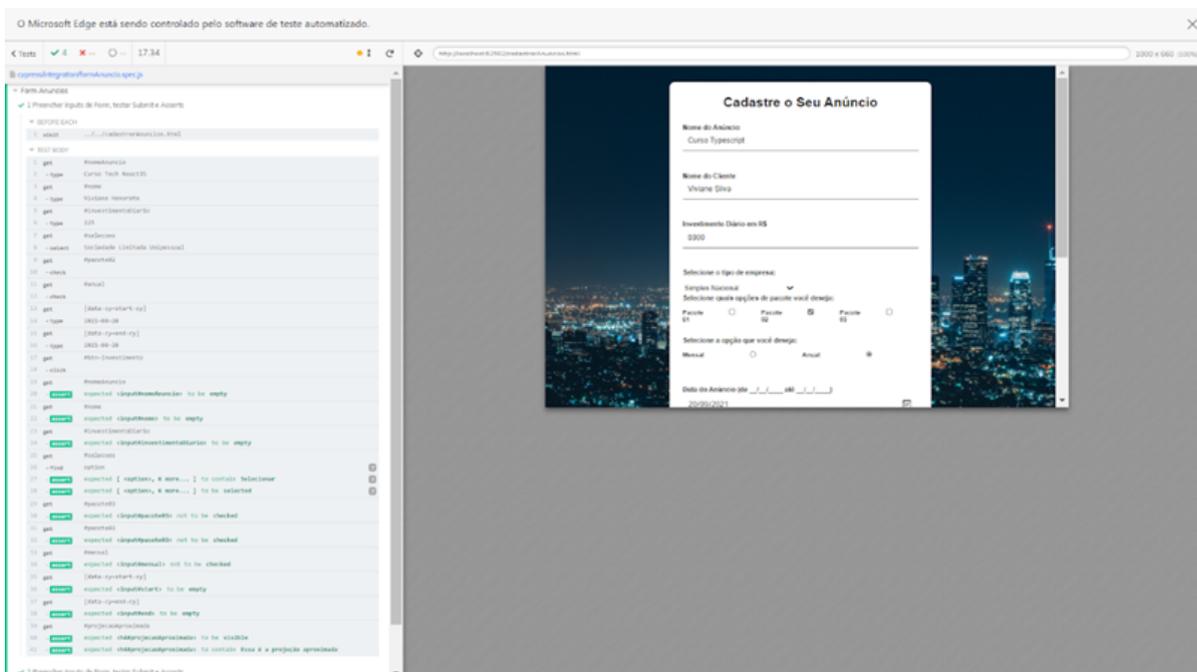
2.1 Testes Automatizados

Os testes automatizados foram desenvolvidos para melhor assertividade e performance da validação dos requisitos funcionais de um sistema. São produzidos com o auxílio de alguma ferramenta - normalmente um *framework*, existindo a possibilidade de visualizar a cobertura do código em relação ao teste, localizar onde o teste foi realizado com sucesso e onde o teste falhou.

Ademais, o desempenho efetuado pelo teste irá depender da qualidade em que o código realizado pelo desenvolvedor foi construído. Tendo em vista que este tipo de teste pode ser inserido tanto na área de *backend* conhecida por conter diversas unidades, funções, rotas e outros, sendo esses atuantes por traz da aplicação, como também na parte *frontend* que tem como princípio ser visualizada de forma direta e visual pelo cliente.

Além disso, como dito anteriormente, existem ferramentas que auxiliam esse tipo de teste e são ferramentas que foram construídas por desenvolvedores com o objetivo de ajudar a progredir em relação aos testes de forma confiável e validando os requisitos necessários. Na Figura 2, é possível observar o *framework* para testes automatizados denominado *Cypress* com o resultado dos testes realizados em uma página que contém *HTML*, *CSS* e *JavaScript*:

Figura 2 – Framework Cypress após a realização de Teste Automatizado



Fonte: Elaborada pelo autor, (2021).

Como resultado da realização do teste automatizado visualizado na Figura 2, é observável que todas as situações testadas adquiriram sucesso, obtendo assim, a validação do funcionamento da página para as funções requisitadas. Ou seja, inicialmente foi definido um cenário que seria utilizado para validar os requisitos,

logo após foram realizados os testes das ações requeridas e por fim, verificado se o retorno foi o esperado, sendo este o sucesso, porém caso contrário, o retorno seria um defeito que se tornou uma falha na aplicação para a equipe analisar e ajustar.

Apesar do sucesso da realização do teste, isso não simboliza que a aplicação não tenha defeitos, somente significa que para as situações testadas, o teste foi realizado, concluído e verificado com êxito. Resultando assim, que para aquele cenário o teste foi satisfatório e cumpriu com a assertividade necessária da ação em questão, sendo ideal pensar em outros cenários testáveis para a cobertura de teste ser mais abrangente. No próximo subtópico, é possível observar a quantidade de *frameworks* existentes para esse auxílio ao testador.

2.1.1 Framework para Testes Automatizados

Visando facilitar o desenvolvimento e produção de programas de *software*, foram criados trechos de código - *framework* - que tem como objetivo solucionar problemas que podem se repetir de forma frequente em várias situações no cotidiano do desenvolvedor.

Outrossim, o *framework* está presente no auxílio aos testes que podem ser realizados de maneira automatizada em um sistema. Existindo assim como objetivo, o benefício ao desenvolvedor em criar testes de forma mais precisa para que a aplicação possa ser considerada segura dentro dos cenários verificados.

Na pesquisa de *frameworks* que terão maior destaque no ano de 2022 criada por Hamilton, podemos observar também a variedade desse tipo de ferramenta no mercado para atender a maior quantidade de tipos de *software* possíveis, além de auxiliar o desenvolvedor e a segurança da aplicação (HAMILTON, 2022).

Figura 3 – Ferramentas de Teste de Automatização



Fonte: Hamilton, (2022), com adaptações.

Por mais que esta lista seja consideravelmente extensa, no mercado de *frameworks* para teste existem ainda mais desenvolvimentos para auxiliar esse tipo de prática, tendo em vista que os programas e ferramentas criadas para essa ajuda ao desenvolvedor só tendem a aumentar conforme a tecnologia evolui.

2.1.2 Tipos de Teste Automatizados

A abrangência em relação aos testes automatizados é um fator extenso a se explorar. Com isso, iremos abordar alguns tipos de testes existentes que são de extrema relevância para o desenvolvimento e funcionamento de *softwares*.

2.1.2.1 Teste de Unidade

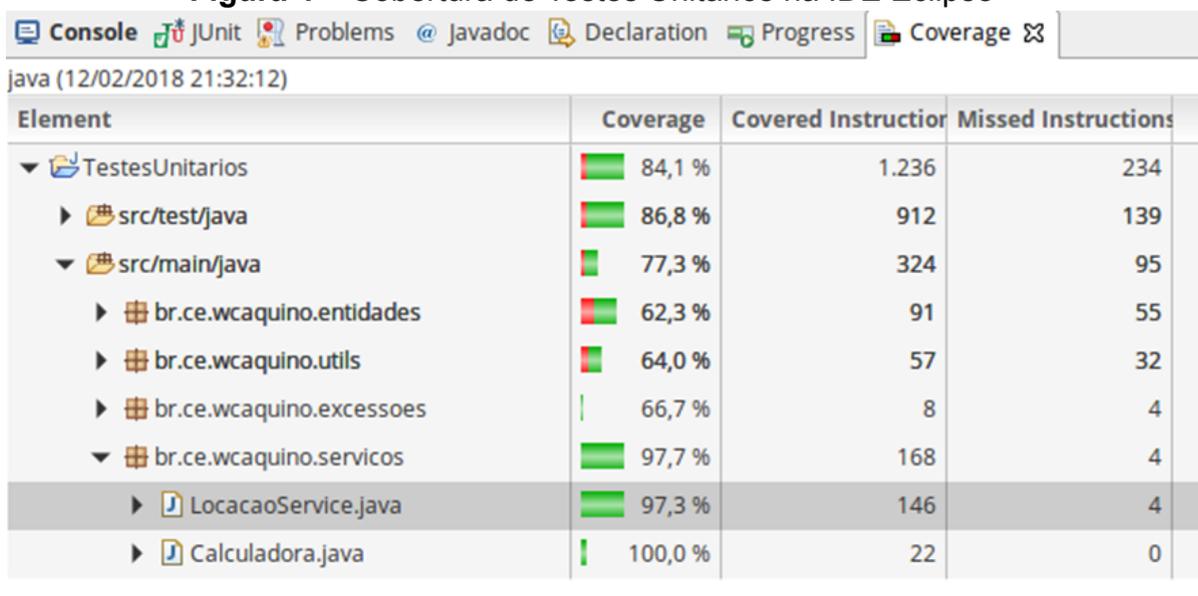
O teste de unidade é um tipo de teste automatizado com o objetivo de testar as unidades de trechos de códigos tais como: propriedades, parâmetros, métodos, classes referentes à implementação, onde essa decisão de cenário irá ser tomada pela equipe de desenvolvedores perante a análise do projeto. O objetivo é verificar a assertividade do funcionamento esperado para o script e não necessariamente a falha destes.

Normalmente é desenvolvido por um programador, que cria a classe de teste para executar e verificar se as unidades daquela classe estão correspondendo de forma satisfatória, caso contrário o sistema irá gerar a saída com informações mais precisas possíveis de onde está a falha e em algumas situações por qual motivo a falha ocorreu. Além disso, quando executado, a resposta em alguns casos vem em forma de porcentagem, demonstrando que a partir de uma certa porcentagem o teste passou e está validado, porém abaixo de certa porcentagem o teste não passou e está invalidado, necessitando de uma análise mais precisa sobre o impasse, tudo isso ocorre dentro do ambiente escolhido para ser testado.

Vamos imaginar o cenário: após fazer um código de um determinado projeto, o desenvolvedor reflete sobre as regras de negócio, com isso observa que existe um trecho de código que corresponde a regra de negócio de extrema importância ao projeto, com isso resolve testar determinado método e suas respectivas propriedades existentes no código, esse tipo de teste satisfaz a necessidade do programador em relação a essa situação que necessita passar por essa verificação e validação, onde essa ação irá trazer a resposta se o trecho de código está correspondendo de maneira correta ou não.

No *software* para desenvolvimento de programas denominado Eclipse, após a realização da saída dos testes unitários há a possibilidade de visualizar a cobertura dos testes como demonstrado na Figura 4.

Figura 4 – Cobertura de Testes Unitários na IDE Eclipse



| Element | Coverage | Covered Instructior | Missed Instructions |
|--------------------------|----------|---------------------|---------------------|
| TestesUnitarios | 84,1 % | 1.236 | 234 |
| src/test/java | 86,8 % | 912 | 139 |
| src/main/java | 77,3 % | 324 | 95 |
| br.ce.wcaquino.entidades | 62,3 % | 91 | 55 |
| br.ce.wcaquino.utils | 64,0 % | 57 | 32 |
| br.ce.wcaquino.excessoes | 66,7 % | 8 | 4 |
| br.ce.wcaquino.servicos | 97,7 % | 168 | 4 |
| LocacaoService.java | 97,3 % | 146 | 4 |
| Calculadora.java | 100,0 % | 22 | 0 |

Fonte: Ribeiro, (2022).

2.1.2.2 Teste de Integração

De acordo com o dicionário Michaelis (2022), integração tem como significado: Condição de construir um todo pela adição ou combinação de partes ou

elementos. Ou seja, o teste de integração é um tipo de teste automatizado que tem como objetivo unir as unidades existentes no projeto, testando-as de forma adjacente.

Em um projeto existem as documentações e os relatórios, onde nesses tipos de documentos estão alocadas regras, requisitos funcionais, modelagens do sistema, projeção do sistema que irão ser observados e validados durante e após o teste está sendo realizado e finalizado. Por meio de situações reais que acontecem no cotidiano de equipes desenvolvedoras de teste, é possível refletir em um cenário onde testes de unidade já foram realizados e o próximo passo possível seria a junção das ideias de cada teste de unidade para testá-los em conjunto, verificando se ao unir as propriedades, os métodos e as classes, o sistema corresponde ao esperado.

Ademais, essa seria uma forma complementar ao teste do tópico anterior. Com isso, podemos concluir pelo pensamento de que o desenvolvedor pode fazer primeiro o teste de unidade e de forma complementar pode realizar o teste de integração para realizar um teste com uma abrangência maior em relação à área testada no projeto.

2.1.2.3 Teste de Sistema

O teste de sistema visa realizar um tipo de teste com um cenário maior que o teste de unidade e de integração realizam, ou seja, fluxos que acabam ocorrendo por todo sistema. Com isso, é indispensável que boa parte do sistema já esteja em fase de conclusão ou na sua versão final, isso é perceptível olhando pelo parâmetro de que o caminho a ser percorrido para o teste, deve estar com todas suas lacunas preenchidas.

Ademais é possível observar que esse tipo de teste é considerado mais complexo que os testes automatizados citados nos tópicos anteriores, tendo em vista que, como o objetivo é ter uma área maior de cobertura, torna-se necessário uma organização de quais cenários irão ser verificados inicialmente, qual será a ordem do fluxo para que a sua cobertura seja eficaz e totalmente testada, quais fluxos podem estar interligados e quais não podem.

Observando esse ponto de vista, é perceptível que todos os caminhos testáveis existentes no produto devem ser organizados e entre as técnicas de organização, temos também mapas mentais de caso de teste, como ilustrado na Figura 5.

Figura 5 – Mapa Mental de Casos de Teste



Fonte: Elaborada pelo autor, (2021).

O mapa mental de casos de teste detém cenários para serem realizados testes, os separando em situações individuais para melhor visualização e avaliação. Isso faz com que o desenvolvedor tenha mais clareza de quais pontos e caminhos devem ser seguidos. Na Figura 5, é perceptível que há um sistema cujo nome é “Sistema do Controle de Obras e Autores”, existindo duas entidades principais: Autores e Obras. Dentro dessas entidades, há situações que se tornaram cenários para realização de testes verificando se a funcionalidade da execução de suas funções está correta ou se há algum tipo de falha. Esse tipo de técnica pode ajudar muito em relação aos Testes de Sistema, tendo em vista que é possível enxergar o sistema como um todo e obter um maior controle do que está sendo testado.

Além disso, a modelagem dos casos de teste é outro viés importante para a realização desse processo, tendo em vista que irá ser exemplificado quais serão os passos e resultados esperados para a realização dos testes. Após os passos indicados e necessários, a execução do teste de sistema pode ser realizada com maior segurança de que está sendo compreendido o caminho a ser seguido, como a saída desse caminho deve responder e se está válido ou não, caso não esteja, o desenvolvedor irá investigar qual o impasse está ocorrendo, se é direcionado ao teste criado ou se o problema está relacionado ao código fonte, para assim poder corrigir e testar novamente.

Dessa forma, é conclusivo que o teste de sistema é indispensável para levar o produto do cliente para a área de produção, onde os usuários poderão usufruir do sistema criado com maior segurança de que os requisitos funcionais e regras de negócio estão sendo atendidas de forma eficaz.

2.2 Testes Manuais

Os Testes Manuais vão para outro domínio da área dos testes, é feito manualmente por um testador. Ou seja, um ser humano faz a exploração do sistema abordando todas as ações possíveis com a intenção de encontrar defeitos, erros e falhas ou validar que o requisito funcional funciona perfeitamente como deveria funcionar.

Esse é um tipo de teste válido e indispensável para um sistema, pois é como se o testador fosse o próprio usuário usufruindo da aplicação. Obtendo pensamentos e opiniões que talvez com o teste automatizado não teria a perspicácia de ter. Em outras palavras, o testador irá agir como se fosse o usuário real, explorando o sistema tanto indo por fluxos corretos, quanto indo por outro tipo de caminho que possa levar o produto a corresponder de uma maneira que não seja a ideal.

A pessoa desenvolvedora que normalmente faz esse tipo de teste, tem uma visão mais ampla dessa ação, muitas vezes e de forma ideal, com uma experiência já pré-existente em testes. Podendo assim, acessar o *software* como se fosse o usuário, mas com a lucidez de um testador. De forma reflexiva, podemos observar que o teste manual se faz tão necessário quanto o teste automatizado, sendo assim, em um produto que exige segurança em seus requisitos e regras de negócio essas ações tornam-se um complementar ao outro.

2.2.2 Tipo de Teste Manual

2.2.2.1 Teste AD-HOC

O teste *AD-HOC* é um tipo de teste manual que tem como objetivo validar requisitos que existem no sistema, porém neste teste não há um roteiro ou direção. O testador irá percorrer todos os caminhos que acredita cobrir a necessidade dos cenários existentes pelo interior da sua intuição e experiência. A partir do pensamento citado em *SWEBOK - Software Engineering Body of Knowledge por Bourque* (2014), é possível concluir que o *AD-HOC* é um dos tipos de teste manuais mais utilizados, porém no olhar de alguns desenvolvedores é o tipo de teste que pode ser inconsistente, já que não existe um fluxo correto a seguir, não sendo guiado por casos de teste ou visualizações de requisitos funcionais e regras de negócio, por exemplo.

2.2.2.2 Teste Exploratório

O teste exploratório é um tipo de teste manual que tem como objetivo fazer a validação de uma determinada hipótese existente no sistema, o que difere este do teste *AD-HOC*. São realizados por um analista de qualidade com experiência pelo motivo de ter o conhecimento de explorar o ambiente com perspicácia, tendo em vista que por ser um teste manual algo pode passar despercebido. Por meio do estudo Conhecimento e Teste Exploratório: um modelo de captação e Execução, podemos concluir que o teste exploratório tem ferramentas essenciais para o sucesso do teste, tais como:

O teste exploratório tem como meta utilizar os recursos disponíveis, habilidades e conhecimento da melhor maneira possível, a fim de encontrar o maior número de erros críticos dentro do tempo disponível. É baseado na habilidade e conhecimento do testador sobre o processo e técnicas de teste. (BRAGA, PRETZ, p. 6).

2.2.2.3 Teste de Usabilidade

O teste de usabilidade é um tipo de teste manual, com o objetivo de ter conhecimento se o sistema tem a usabilidade que corresponde a necessidade do usuário de forma intuitiva, ou seja, se o sistema na visão do cliente, além de corresponder às suas necessidades, também é fácil de utilizar. Nesse tipo de teste, pode-se utilizar algum tipo de padrão existente, como por exemplo o ISO 9241-11 documentado em uma nova versão pela Associação Brasileira de Normas Técnicas (2011), que tem como foco a interação homem-computador, centrando em princípios básicos para que isso ocorra com êxito, tais como: Eficácia, Eficiência, Satisfação.

3 SISTEMA CRÍTICO EM CENÁRIO PANDÊMICO E REMOTO

3.1 Sistema Crítico

Sistema Crítico é todo sistema que existe a possibilidade de perdas de grande relevância, caso exista instabilidades decorrentes de falhas. Os danos que ocorrerem podem ter impacto do tipo pessoal, físico, além do dano financeiro e por esse motivo o cuidado em relação a estabilidade do seu funcionamento deve ser monitorado para verificar se sua funcionalidade é satisfatória e segura.

O objetivo desse tipo de sistema é realizar operações que tenham impacto relevante sobre alguma situação que é comumente realizada na vida de cidadãos. Podendo ser uma transação bancária, um controle de aparelhos que auxiliam os

sensores de segurança de um carro, o armazenamento de dados pessoais cruciais de um sistema cadastral ou até mesmo o controle de um avião com alta capacidade de passageiros, por exemplo.

Esse tipo de sistema, tem como fatores básicos a integridade, a confiabilidade e a disponibilidade, tendo em vista que as duas primeiras se tornam indispensáveis para esse tipo de serviço, podemos concluir isso pelos tipos existentes de sistemas críticos:

- **Segurança:** uma falha em um sistema desse tipo pode causar danos referentes à vida do ser humano, além da possibilidade de impactar a natureza.
- **Missão:** uma falha em um sistema desse tipo pode causar danos referentes a operações que auxiliam em serviços cruciais para a execução de tarefas em uma empresa, por exemplo.
- **Negócio:** uma falha em um sistema desse tipo pode causar danos financeiros consideráveis que irão impactar o cliente, além de consequentemente impactar a empresa que oferece o serviço relacionado ao negócio.

Com isso, é possível observar que a atenção necessária para esse tipo de sistema é considerada alta, tendo em vista que o desenvolvimento tem que estar adequado ao projeto e suas operações essenciais, além da necessidade em o produto ser seguro, sendo algo que possa ser validado pelo meio de testes de *software* realizados exaustivamente para confirmar sua segurança.

3.2 Cenário Pandêmico

O Cenário Pandêmico é um fruto da pandemia, sendo essa uma situação preocupante relacionada à saúde das pessoas ocorrendo ao redor do mundo em uma mesma época. Como detalhado pela Fundação Oswaldo Cruz, a Organização Mundial da Saúde (2021) declara que:

A pandemia é a disseminação mundial de uma nova doença e o termo passa a ser usado quando uma epidemia, surto que afeta uma região, se espalha por diferentes continentes com transmissão sustentada de pessoa para pessoa. (OMS, 2021).

Com isso, a sociedade nesse tipo de ambiente tem a necessidade de tomar medidas de prevenção para a doença não atingir principalmente os grupos de risco, tais como: idosos, pessoas com doenças respiratórias e doenças que vem de âmbito genético, além da área infantil. Tendo em vista essa situação, um cenário pandêmico cria novas normas de convivência a serem seguidas para que não se torne algo ainda mais preocupante e alastrante.

Entre as normas existentes desse tipo de situação, podemos incluir a recomendação da não circulação frequente nas ruas e ambientes sociais externos, além da utilização de máscaras para proteção e não retenção do vírus, por exemplo.

Não somente o contexto social tem que se moldar ao novo estilo de vida que ocorre após esse tipo de circunstância, mas também a própria tecnologia. O mundo tecnológico tornou-se essencial para o cenário pandêmico, dando ainda mais força ao cenário remoto.

3.3 Cenário Remoto

O Cenário Remoto é o ambiente que tem acesso à tecnologia, como por exemplo a internet que traz a possibilidade de inserção no mundo remoto por conexões globais. Nesse local é possível trabalhar, obter entretenimento, fazer pagamentos e transferências, inclusive realizar consultas médicas de forma online e não presencial.

Com o ambiente pandêmico, foi possível visualizar uma crescente utilização de serviços remotos para a conclusão de atividades que anteriormente eram realizadas de forma presencial, como enfrentar a fila de um banco para pagar um boleto, consultar-se com o psicólogo, ter reunião com clientes, entre outros. Ainda há com certeza, quem usufrua disso presencialmente, mas com o cenário de pandemia o ambiente remoto tem obtido um espaço maior no dia a dia das pessoas.

É possível observar que tecnologias para atividades online cresceram em relação a outros anos anteriores. Por meio da pesquisa realizada pelo FIS -fornecedor líder de soluções tecnológicas para clientes - em parceria com o Ipsos, houve uma análise que trabalhou justamente na questão da compra remota realizada por um grupo de pessoas. Essa pesquisa concluiu que 52% das pessoas utilizaram formas de pagamento online, 49% adquiriram os produtos de interesse por aplicativos, além de 40% desses indivíduos pagarem a compra sem nenhum tipo de contato físico, ou seja, por meio de sites, *QR Codes*, sistemas como *Buy Now Pay Later*, entre outros.

4 METODOLOGIA

Para o desenvolvimento deste artigo, o estudo foi centrado na coleta de materiais e pesquisas, além das análises onde existem pontos e detalhamentos que foram levantados anteriormente por autores com direcionamento ao tema de softwares, consistindo na afirmação de experiência vivenciadas do ambiente de estudo e profissional, além do cenário de crescimento sócio político adquirido ao longo dos anos.

Com isso, é possível afirmar que o trabalho se encaixa na metodologia de análise documental, onde o objetivo é justamente coletar informações relacionadas ao tema através da percepção sobre o que deve ser elaborado, como descrito por Sá-Silva, Almeida e Guindani (2009, p. 5), "... um procedimento que se utiliza de métodos e técnicas para a apreensão, compreensão e análise de documentos dos mais variados tipos".

Isso se evidencia não somente por existir diversos conteúdos relacionados a área de testes e sua importância, como também no assunto abordado nessa pesquisa relacionado ao ambiente que tem como parâmetros novos acontecimentos sociopolíticos e econômicos.

A pesquisa ocorre por meio do detalhamento dos conceitos necessários para compreender a relevância da atuação de testes manuais e automatizados em cenários que além de serem considerados críticos, também têm o agravante de uma epidemia mundial. Para discorrer sobre o contexto do tema, a adição de conteúdos jornalísticos que contenham informações relevantes relacionadas a pesquisa também foi uma das abordagens utilizadas para chegar ao resultado final.

Ademais, a importância de fontes secundárias para esse tipo de método, foi essencial tendo em vista que autores tais como Sommerville (2007) com o livro Engenharia de *Software*, Braga e Pretz com o artigo Conhecimento e Teste

Exploratório: um modelo de captação e Execução, como também consultas realizadas no dicionário de língua portuguesa Michaelis e na documentação de padrões pela Associação Brasileira de Normas Técnicas enriqueceram as descrições de variáveis necessárias para a demonstração do tema abordado. Além de outras fontes significativas que não foram aqui descritas, mas estão na referência.

Em capítulos existentes nos materiais citados acima, como por exemplo no livro de Engenharia de *Software*, foi possível identificar grande parte dos conceitos relevantes para esse artigo, tendo em vista que, além de ser um livro conceituado e utilizado na área acadêmica, traz assuntos que vão além de somente testes, mas também sistemas críticos, requisitos funcionais, software em geral. Isso traz clareza para o aprendizado e entendimento para dissertar sobre o tema. Quando a fonte foram artigos, conteúdos jornalísticos e padrões pré-definidos por instituições, o objetivo foi obter uma quantidade e qualidade de conteúdos mais amplos, onde poderia ser possível visualizar resultados de métodos em situações reais.

A abordagem de leitura aconteceu pelo levantamento de pontos que foram considerados relevantes para a construção desse artigo, visualizando pontos conceituais importantes para a compreensão, além do olhar desses conceitos direcionados a situação social que esta pesquisa aborda. Com o escopo final de trazer uma soma de conceitos e ideias no resultado deste artigo.

Podemos concluir que a pesquisa qualitativa é o tipo de padrão que proporciona o modelo adequado para essa estrutura de trabalho, onde o parâmetro para medir esses resultados estarão presentes nas reflexões, conceitos, informações, além de ideias obtidas após a conclusão do artigo científico.

5 ANÁLISE

Na atual conjuntura sociopolítica e estrutural, é notável que durante o cenário pandêmico a expansão da utilização de sistemas críticos se tornou mais frequente nas camadas sociais. Em alguns casos de forma espontânea, porém em outros, o uso surgiu por alguma necessidade. Isso se evidencia não somente por percepções sensoriais sociais, mas também por acontecimentos que existiram no coletivo, como por exemplo, sobre o que vamos discorrer neste tópico.

Em virtude da confirmação do ambiente pandêmico que se instaurou em meados de janeiro de 2020 no Brasil, a presidência da república visualizou a necessidade da criação da Lei Nº 13.982, na data de 2 de abril de 2020, descrita no trecho abaixo pelo Instituto Brasileiro de Geografia e Estatística (2020):

Art. 2º Durante o período de 3 (três) meses, a contar da publicação desta Lei, será concedido auxílio emergencial no valor de R\$ 600,00 (seiscentos reais) mensais ao trabalhador que cumpra cumulativamente os seguintes requisitos:

I - seja maior de 18 (dezoito) anos de idade, salvo no caso de mães adolescentes; (Redação dada pela Lei nº 13.998, de 2020)

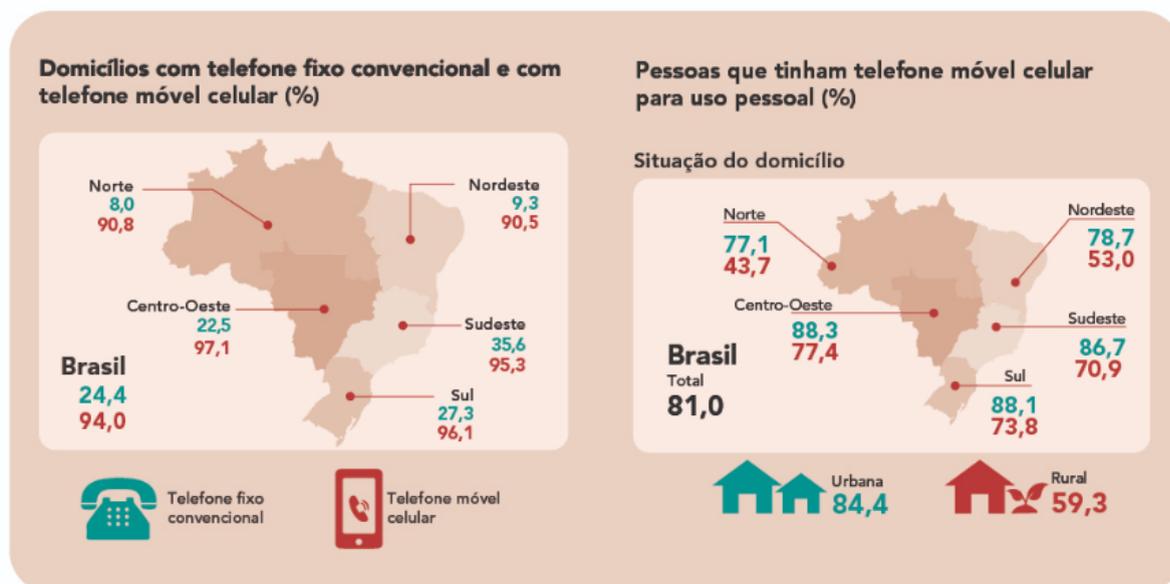
II - não tenha emprego formal ativo; (...). (Instituto Brasileiro de Geografia e Estatística, 2020).

Essa norma prevê o pagamento de valor monetário para que a sociedade diante do quadro social daquele momento, tivessem um auxílio para a quitação de suas dívidas, além da possibilidade de comprar itens alimentícios.

Como o cenário estava situado em distanciamento social, foi necessária uma estratégia para que essa contribuição chegasse ao maior número possível de

peças sem que existisse a necessidade do contato físico. Por meio de pesquisa realizada pelo IBGE no ano de 2019, pode-se observar a quantidade de aparelhos móveis no território brasileiro como demonstrado na Figura 6. Com isso a estratégia foi a criação de um aplicativo para realizar esse pagamento aos cidadãos, visando assim, o celular como meio essencial para o sucesso dessa tática. Necessitando medidas de prevenção para que a segurança no aplicativo não seja falha e os usuários consigam receber o pagamento do auxílio de maneira segura.

Figura 6 – Porcentagens da Pesquisa do IBGE relacionadas aos telefones móveis



Fonte: IBGE, (2019).

Como já dito anteriormente nesse artigo, essa situação se enquadra no Sistema Crítico de Negócio, tendo em vista que pagamentos serão realizados a milhares de cidadãos ocorrendo por meio de um aplicativo de *software* de nome instalado no aparelho móvel. Isso evidencia uma situação que ocorreu em diversos cenários, ou seja, o aumento da demanda de sistemas críticos em decorrência da pandemia tornou ainda mais necessárias as estratégias de segurança que irão tornar o *software* mais seguro para o uso social.

Dessa forma, uma tática viável para a obtenção da convicção de segurança em determinadas situações é o teste de *software*, que irá englobar alguns tipos de técnicas sejam estes manuais ou automatizadas, para a confirmação de que a atividade verificada está com o seu uso seguro.

Mas como podemos dar maior assertividade que em um sistema crítico o teste irá funcionar no ambiente remoto disponibilizado ao cliente? Essa pergunta pode ser respondida por conceitos denominados: ambiente de homologação e ambiente de produção. O conceito de cenário de homologação pode ser definido a partir da afirmação da empresa QASOLUTIONS (2020):

O objetivo da homologação de sistema é garantir que o projeto web obedeça às regras de negócio e que a experiência do usuário seja positiva. O serviço consiste em assumir o papel do usuário final, utilizando todas as funcionalidades a ele oferecidas com a intenção de descobrir erros e defeitos. Podemos atuar na fase de desenvolvimento ou em sistemas já

implementados, seja em módulos ou sistemas completos, identificando problemas e acompanhando correções. (QASOLUTIONS, 2020).

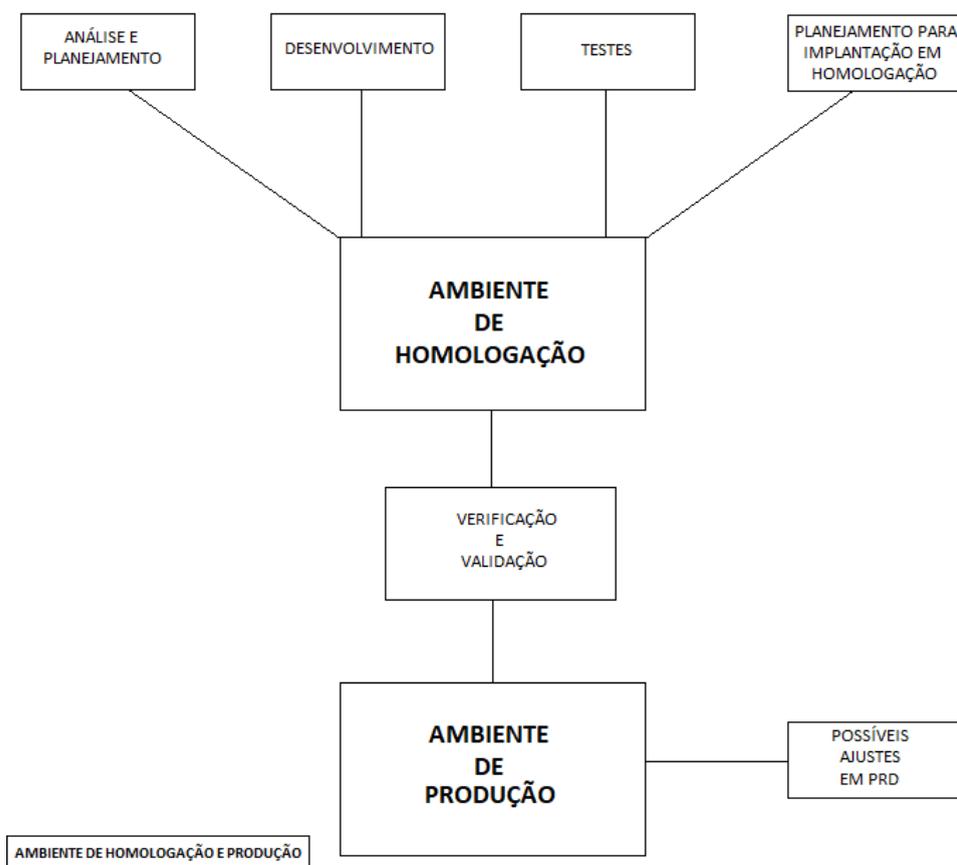
Em outras palavras, o ambiente de homologação é direcionado ao teste do desenvolvedor, antes que o desenvolvimento vá para o ambiente de produção. Ou seja, o ambiente de produção é aquele cujo usuário final pode usufruir.

Para todo sistema de *software* testável, é perceptível que há uma sequência em relação à análise feita a partir do caminho necessário para monitorar o comportamento esperado do teste, e com sistemas críticos não seria diferente. Após os testes de unidade, de integração que podem ser feitos diretamente no código fonte, há a necessidade de realizar testes no ambiente de homologação onde essas técnicas podem ser tanto automatizadas, quanto manuais.

Supondo que temos um sistema crítico que está em um cenário social pandêmico, visualizamos a necessidade em analisar e testar se o sistema após a finalização do desenvolvimento está correspondendo a meta de segurança requerida e planejada. É nesse momento que entra o ambiente de homologação, onde após a inserção do código desenvolvido pela *squad* e a finalização de testes relacionados ao código fonte, os cenários e fluxos irão ser testados como se estivessem em produção, mas o usuário final não terá acesso nesse momento. Isto faz com que, os desenvolvedores tenham um ambiente que seja equivalente ao ambiente de produção, onde este sim é utilizado pelo usuário final.

Ou seja, como exemplificado na Figura 7, primeiramente irá existir o desenvolvimento da equipe e testes que são possíveis realizar no código, por exemplo. Após isso, temos a introdução desse desenvolvimento no ambiente de homologação onde os testes são intensificados, obtendo uma precisão maior já que o ambiente de homologação é o equivalente ao de produção. Após a verificação e validação de que os testes estão correspondendo de maneira correta e o desenvolvimento dos fluxos e cenários estão trazendo segurança ao sistema, existirá nesse momento a transição do conteúdo testado em homologação para o ambiente de produção, onde o cliente e usuário final irão usufruir do sistema com maior segurança.

Figura 7 – Ambiente de Homologação e Produção



Fonte: Elaborada pelo autor, (2022).

A partir do momento em que os testes são considerados um sucesso no ambiente de homologação pela equipe, é assimilado que dentro dos fluxos e cenários testados, o sistema está correspondendo de maneira satisfatória, ou seja, está seguro dentro das situações monitoradas. Isso não significa que no sistema não exista algum tipo de falha, mas traz a segurança de que nas situações verificadas tudo está ocorrendo como o esperado e com segurança.

Esses conceitos valem tanto para sistemas não-críticos como para sistemas críticos, porém quando são críticos são necessários cuidados redobrados, já que outras questões como vida humana, danos financeiros, impacto à natureza, além de danos referentes a operações também entram em pauta. A maneira que podemos fortalecer esses cuidados vem da prática constante de testes, além de ambientes de testes, como o citado nesse tópico, o ambiente de homologação.

6 CONSIDERAÇÕES FINAIS

A existência de práticas essenciais para a segurança de sistemas críticos quando estão em um agravante social como a pandemia discutida nesse artigo, é de suma importância a realização de várias técnicas de testes, além de ambiente de homologação equivalente ao de produção, para que os desenvolvedores tenham uma visão mais clara do que irá ocorrer no sistema disponibilizado ao usuário. Visando assegurar que as situações testadas estão de acordo com a devida segurança que o sistema necessita.

Por meio de todos conceitos levantados, além de cenários exemplificados é possível afirmar que os objetivos do estudo foram atingidos, trazendo uma maior clareza e objetividade sobre o assunto tratado. Tendo em vista que foram utilizados artigos, normas, matérias jornalísticas, livros de engenharia de *software*, além de materiais com grande experiência em discussão de tecnologias.

Além disso, a possibilidade de continuidade desse estudo, seria o desenvolvimento de um *software* onde existam alguns cenários citados por aqui. Cenários que seriam relevantes para a construção e afirmação no modo prático como por exemplo a sequência de planejamento dos requisitos extraídos do cliente, o desenvolvimento da aplicação, os testes manuais e automatizados, a inserção destes no ambiente de homologação que trará a verificação e a validação para assim, poder prosseguir com o ambiente de produção. Sendo essa, a possível sugestão para continuidade deste trabalho.

É possível pontuar que alguns dos contextos abordados aqui são recentes, ou seja, as informações que ocorreram diariamente muitas vezes não foram documentadas de maneira correta. Porém, mesmo existindo esse ponto de atenção podemos concluir que as técnicas relacionadas aos modelos de testes são indispensáveis para sistemas críticos, tornando-se extremamente relevantes e necessárias, ainda mais perante ao contexto de utilização social que o sistema terá em adição à situação pandêmica.

REFERÊNCIAS

- ABNT. Associação Brasileira de Normas Técnicas. **Requisitos Ergonômicos para Trabalho de Escritórios com Computadores Parte 11 – Orientações sobre Usabilidade**. [S.n], 2011. Disponível em: http://www.inf.ufsc.br/~edla.ramos/ine5624/_Walter/Normas/Parte%2011/iso9241-11F2.pdf. Acesso em: 01 jun. 2022.
- BIO FIOCRUZ. **O QUE É UMA PANDEMIA**. [S.n], 2021. Disponível em: <https://www.bio.fiocruz.br/index.php/br/noticias/1763-o-que-e-uma-pandemia>. Acesso em: 18 mar. 2022.
- BOEHM, B.W., ABTS, C., BROWN, A.W., CHULANI, S., CLARK, B.K., HOROWITZ, E., MADACHY, R., REIFER, D., STEECE, B. **Software Cost Estimation with COCOMO II**. 1ª. Ed. Prentice Hall, 2000.
- BOURQUE, P., FAIRLEY, R. E. **Guide to the Software Engineering Body of Knowledge (Swebok(r))**. 3ª Ed. IEEE Computer Society Press, 2014.
- BRAGA, K., PRETZ, E. **Conhecimento e Teste Exploratório: um modelo de captação e Execução**. [S.d], 2022. Disponível em: <https://www.yumpu.com/pt/document/read/21959183/tc-online-feevale>. Acesso em: 01 jun. 2022.
- CUNHA, F. **requisitos funcionais e não funcionais: o que são?**. Mestres da Web, [S.d], 2022. Disponível em: <https://mestresdaweb.com.br/tecnologias/requisitos-funcionais-e-nao-funcionais-o-que-sao/#:~:text=O%20que%20s%C3%A3o%20requisitos%20funcionais,meio%20de%20fun%C3%A7%C3%B5es%20ou%20servi%C3%A7os>. Acesso em: 18 abr. 2022.
- GOV.BR, Governo do Brasil. Instituto Brasileiro de Geografia e Estatística. **USO DE INTERNET, TELEVISÃO E CELULAR NO BRASIL**. [S.n], 2020. Disponível em: <https://www.in.gov.br/en/web/dou/-/lei-n-13.982-de-2-de-abril-de-2020-250915958>. Acesso em: 10 jun. 2022.
- HAMILTON, T. **20 BEST Automation Testing Tools (May 2022 Update)**. [S.], 2022. Disponível em: <https://www.guru99.com/automated-testing-tools.html>. Acesso em: 01 jun. 2022.
- IBGE. Instituto Brasileiro de Geografia e Estatística. **USO DE INTERNET, TELEVISÃO E CELULAR NO BRASIL**. [S.n], 2019. Disponível em: <https://educa.ibge.gov.br/jovens/materias-especiais/20787-uso-de-internet-televisao-e-celular-no-brasil.html>. Acesso em: 10 jun. 2022.
- Michaelis Dicionário Brasileiro da Língua Portuguesa. **Integração**. [S.n], 2022. Disponível em: <https://michaelis.uol.com.br/busca?r=0&f=0&t=0&palavra=integração>. Acesso em: 01 jun. 2022.
- LACARTE, C. **Relatório Pace Pulse mostra que 49% dos consumidores compram via aplicativo e 40% fazem pagamentos sem contato; novos hábitos**

são convergentes à oferta de novas ferramentas e inovações tecnológicas, como Pix, que encerrou 2021 movimentando R\$ 622,4 milhões. Jornal do Sudoeste, 2022. Disponível em: <https://www.jornaldosudoeste.com/brasileiro-amplia-uso-de-pagamentos-digitais-diz-pesquisa-fis/>. Acesso em: 27 mar. 2022.

QASolutions. **TESTE e HOMOLOGAÇÃO.** [S.n], 2020. Disponível em: <https://www.qasolutions.com.br/teste-e-homologacao-de-sistemas#:~:text=O%20objetivo%20da%20homologa%C3%A7%C3%A3o%20de,de%20descobrir%20erros%20e%20defeitos.> Acesso em: 10 jun. 2022.

RIBEIRO, J. V. **Java Junit – Principais assertivas Asserts.** Joao Verissimo Ribeiro Programação Java, Spring, Php, Jquery, Css3, Mvc, CentOS. [S.], 2022. Disponível em: <https://joaoverissimoribeiro.wordpress.com/2018/02/11/java-junit-principais-assertivas-asserts/>. Acesso em: 01 jun. 2022.

SÁ-SILVA, J. R.; ALMEIDA, C. D.; GUINDANI, J. F. **Pesquisa documental:** pistas teóricas e metodológicas. Revista Brasileira de História e Ciências Sociais, São Leopoldo, RS, 2009.

SOMMERVILLE, I. **Engenharia de Software.** 8ª. Ed. PEARSON, 2007.