



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA
PARAÍBA CAMPUS MONTEIRO
COORDENAÇÃO DO CURSO TECNÓLOGO EM ANÁLISE E
DESENVOLVIMENTO DE SISTEMAS**

FILIFE DE OLIVEIRA BARBOSA

**RELATÓRIO DE ESTÁGIO NA ÁREA DE QUALIDADE EFETUADA
NA EMPRESA KLOK TECH**

MONTEIRO - PB

2023

FILIFE DE OLIVEIRA BARBOSA

**RELATÓRIO DE ESTÁGIO NA ÁREA DE QUALIDADE EFETUADA
NA EMPRESA KLOK TECH**

Relatório de estágio, apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba Campus Monteiro, como pré-requisito para a Disciplina de Trabalho de Conclusão de Curso (TCC), sob orientação do Prof. Dr. Tiago Brasileiro Araújo.

MONTEIRO, PB

2023

Dados Internacionais de Catalogação na Publicação – CIP
Bibliotecária responsável Porcina Formiga dos Santos Salgado CRB15/204
IFPB Campus Monteiro.

B238r Barbosa, Filipe de Oliveira.

Relatório de estágio na área de qualidade efetuada na Empresa
KLOK TECK / Filipe de Oliveira Barbosa – Monteiro-PB. 2023.
31fls. : il.

Relatório (Curso Superior de Tecnologia em Análise
e Desenvolvimento de Sistemas) - Instituto Federal de Educação,
Ciência e Tecnologia da Paraíba - IFPB campus, Monteiro.

Orientador: Prof. Dr. Tiago Brasileiro Araújo.

1. Software - Qualidade 2. Testes – Manuais – Contrato – Carga -
Integração 3. Empresa KLOK TECK I. Título.

CDU 004.453

FILIPE DE OLIVEIRA BARBOSA

**RELATÓRIO DE ESTÁGIO NA ÁREA DE QUALIDADE EFETUADA NA
EMPRESA KLOK TECH**

Relatório de estágio, apresentado ao Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba Campus Monteiro, como pré-requisito para a Disciplina de Trabalho de Conclusão de Curso (TCC).

Aprovado em 30 de Agosto de 2023.

BANCA EXAMINADORA

Tiago Brasileiro Araújo

Prof. Dr. Tiago Brasileiro Araújo (Orientador - IFPB)

Wagner de Oliveira Santos

Prof. Wagner de Oliveira Santos (Examinador - IFPB)



Documento assinado digitalmente

WANDERLEY ALMEIDA DE MELO JUNIOR

Data: 10/10/2023 10:12:01-0300

Verifique em <https://validar.iti.gov.br>

Prof. Wanderley Almeida de Melo Junior (Examinador - IFPB)

AGRADECIMENTOS

Desejo agradecer primeiramente a minha mãe, por sempre incentivar ao estudo e auto aprendizado. Ao professor Dr. Cleyton Caetano de Souza, por me apresentar ao curso em uma de suas apresentações a alunos do ensino médio, e despertar minha curiosidade sobre a área da programação.

Agradeço ao instituto federal da Paraíba e aos colegas e amigos que fiz durante o decorrer do curso de análise e desenvolvimento de sistemas, compartilhando conhecimentos e vivências, agradeço também ao professor Dr. Tiago Brasileiro Araújo por me apresentar a prática e o conceito de testes e qualidade de software.

E por fim, agradeço a Klok Tech e toda a equipe por me receber de braços abertos e me disponibilizar a oportunidade de aprender e crescer junto com eles. Em especial ao responsável por aprovar e supervisionar o discente no cargo de estagiário em qualidade de software e em todo o decorrer do seu estágio, Jânio Paz Amorim Júnior.

RESUMO

O objetivo da documentação presente aqui é por sua vez relatar o convívio, conhecimentos obtidos e atividades praticadas pelo discente, aluno do curso tecnólogo de análise e desenvolvimento de sistemas em seu cargo de estágio como QA (Quality Assurance), das atividades praticadas entre elas, testes manuais utilizando Postman e Dbeaver, testes de contrato com Pact, testes de integração com MockMvc e finalmente testes de carga com Jmeter, todas experienciadas e aqui descritas. O estágio com aproximadamente duração de 10 meses com o propósito de praticar o que foi aprendido ao decorrer do curso, por fim o relatório destaca a importância do estágio e da oportunidade adquirida para aplicar e desenvolver conhecimentos teóricos e práticos, fortalecendo o futuro profissional ao experienciar desafios que agregaram em seu desenvolvimento profissional ao longo desse período.

Palavras-chave: quality assurance; qualidade de software; testes manuais; testes de contrato; testes de carga; testes de integração; mockmvc; postman; jmeter; pact; estágio; análise e desenvolvimento de sistemas.

ABSTRACT

The objective of the documentation present here is, in turn, to report the coexistence, knowledge obtained and activities practiced by the student, student of the technologist course of analysis and systems development in his internship position as QA (Quality Assurance), of the activities practiced between them, manual tests using Postman and Dbeaver, contract tests with Pact, integration tests with MockMvc and finally load tests with Jmeter, all experienced and described here. The internship lasts approximately 10 months with the purpose of practicing what was learned during the course, finally the report highlights the importance of the internship and the opportunity acquired to apply and develop theoretical and practical knowledge, strengthening the professional future by experiencing challenges that added to their professional development throughout this period.

Keywords: quality assurance; software quality; manual Tests; contract tests; load tests; integration tests; mockmvc; postman; jmeter; pact; Internship; analysis and systems development.

LISTA DE ILUSTRAÇÃO

Figura 1 – Logo da marca da Klok Tech-----	13
Figura 2 – Abstração da organização da empresa-----	14
Figura 3 – demonstração de um quadro feito no Jira-----	17
Figura 4 – Interface do Dbeaver-----	18
Figura 5 – Registro de caso de teste-----	22
Figura 6 – Exemplo de requisição HTTP no Postman-----	24
Figura 7 – Exemplo de busca no banco usando Dbeaver-----	24
Figura 8 – Exemplo método consumidor simplificado-----	25
Figura 9 – Contrato exemplo em Json-----	26
Figura 10 – tela de configuração de requisições do Jmeter-----	27
Figura 11 – Exemplo da preparação da requisição no Jmeter-----	28
Figura 12 – Exemplo estruturação e teste de integração com MockMVC-----	29

LISTA DE ABREVIATURA

ADS	Quality Assurance/Garantia de qualidade
API	Application Programming Interface
BDD	Behavior Driven Development
DEV	Desenvolvedor(es)
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
DAC	Desenvolvimento de aplicações corporativas
IFPB	Instituto Federal da Paraíba
PO	Product Owner
QA	Quality Assurance/Garantia de qualidade
RH	Recursos humanos
RPE	Retail Payment Ecosystem
URL	Uniform Resource Locator

SUMÁRIO

1. INTRODUÇÃO	10
2. OBJETIVOS.....	12
2.1 Objetivo geral	12
2.2 Objetivo específicos.....	12
3. A EMPRESA.....	13
3.1 Descrição da empresa.....	13
3.2 Organização da empresa	14
4. EMBASAMENTO TEÓRICO	15
4.1 Metodologias ágeis (Scrum e Kanban)	15
4.2 Discord.....	16
4.3 Jira.....	16
4.4 Behavior Driven Development (BDD)	17
4.5 Docker	18
4.6 Postman	18
4.7 Dbeaver.....	18
4.8 Pact	19
4.9 MockMVC	19
4.10 Jmeter	19
4.11 sMockin e localStack.....	20
5. RELATO DE ESTÁGIO.....	20
5.1 Inserção na empresa.....	20
5.2 Registro de testes	22
5.3 Testes manuais.....	23
5.4 Testes de contrato	25
5.5 Testes de carga	26
5.6 Testes de integração.....	28
5.7 Testes manuais de interface	29
6. CONCLUSÃO	30
REFERÊNCIAS.....	31

1. INTRODUÇÃO

Entre as áreas da tecnologia, a profissão de QA - Quality Assurance ou em português Garantia de Qualidade é focada na qualidade do produto final entregue ao cliente, a área tem como objetivo garantir o comportamento esperado, encontrar possíveis falhas e apontar melhorias, segundo Roger S. Pressman e Bruce R. Marxim (2002) “[...] se uma equipe de software buscar a qualidade em todas as atividades de engenharia de software, a quantidade de retrabalho será reduzida. Isso resulta em custos menores [...] se o produto atende às especificações, ele apresenta qualidade.”

Todo o processo garante a satisfação do agente que solicita os serviços da empresa, estabelecendo assim uma relação de confiança sólida. Isso não apenas a torna mais confiável no mercado, mas também reforça a sua reputação, contribuindo para um crescente reconhecimento da empresa em meio ao competitivo mercado de tecnologia.

Os testes ajudam a reduzir os riscos de falhas que podem prejudicar em várias medidas o produto da empresa, assegurando assim sua qualidade e confiabilidade e fortalecendo a satisfação do cliente. Vale ressaltar que entre os testes existem alguns métodos, esses métodos são separados em dois, costumeiramente chamados de métodos caixa branca e caixa preta.

O método caixa branca consiste em ter conhecimento do algoritmo e lógica do serviço ou programa que está sendo testado, em contrapartida, o caixa preta é a falta de conhecimento sobre o produto, como uma venda, fazendo o QA ter uma experiência similar a um usuário comum que usaria o serviço posteriormente, segundo MYERS (1979) testar não deve ser tratada como uma tarefa simples, é necessário que a pessoa responsável pelos testes tenha experiência e

seja suficientemente treinada nas técnicas e ferramentas diversas que estão disponíveis, portanto os demais testes são baseados na experiência e criatividade do responsável por testar.

A área também é abordada ao longo do curso, em destaque a disciplina de Técnicas de Teste, matéria que faz a introdução e demonstra as técnicas e formas para testar e avaliar um software, além de apresentar o conceito e o processo de testes, destaca a importância de não apenas cumprir os requisitos solicitados, mas ter a certeza que o comportamento do software está conforme o planejado, desde casos de fluxo feliz, até os casos de exceção.

No decorrer deste relatório será descrito todas as atividades realizadas durante o tempo de estágio na empresa Klok Tech, relacionadas a área de Qualidade de software, entre essas atividades podem ser citadas antecipadamente testes automatizados, testes de carga, testes de contrato e testes manuais, processos e ferramentas que foram utilizados e absorvidos pelo aluno, sendo descritas todas as funções e desafios que o discente teve até o momento do encerramento do estágio descrito neste documento, além de ressaltar partes fundamentais do comportamento e organização da empresa citada.

2. OBJETIVOS

2.1 Objetivo geral

Como objetivos do estágio, em geral, a obtenção de conhecimento de ferramentas e técnicas para realização de diversos tipos de testes, utilizando e aprimorando o conhecimento adquirido ao longo do curso de Análise e Desenvolvimento de Sistemas, convivência empresarial e participação do dia a dia da empresa.

2.2 Objetivo específicos

- Efetuar o registro de cenários de testes utilizando a metodologia BDD.
- Analisar a excelência de programas através da realização de testes manuais utilizando as ferramentas Postman e Dbeaver visando avaliar sua qualidade e comportamento.
- Realização de testes de contrato utilizando Pact.
- Examinar, por meio de testes de carga, o desempenho de programas e replicar possíveis erros utilizando a ferramenta Jmeter.
- Desenvolver cenários de teste de integração utilizando a ajuda da ferramenta de testes MockMVC.
- Testes manuais em Front-End desenvolvida em Angular

3. A EMPRESA

3.1 Descrição da empresa

Figura 1 – Logo da marca da Klok Tech



Fonte: Klok Tech (2023).

<https://br.linkedin.com/company/klok-tech>

A empresa Klok Tech¹ (Figura 1), é uma startup que teve sua fundação no primeiro trimestre do ano de 2021 durante a atuação em alta do covid-19, tendo sua sede localizada no endereço Club CoWorking – Faria Lima Av. Pres. Juscelino Kubitschek, 1327 – Itaim Bibi, São Paulo, com o objetivo de disponibilizar uma melhor solução tecnológica de gestão de finanças, atuando como hub (concentrador de informações) e conectando seguradoras aos clientes varejistas, agregando valor à empresa, auxiliando na solução de meios de pagamento, concentração das informações de segurados e seus seguros e assistências fornecidas pelos clientes tomadoras do serviço da Klok.

Considerando sua criação durante o período da pandemia e isolamento social, foi optado por um modelo Home Office, evitando contato físico e amontoamento de pessoas no mesmo ambiente a fim de preservar a saúde dos membros da equipe. Essa escolha proporciona que a empresa contrate pessoas por todo país e beneficiando os seus funcionários com maior flexibilidade, privacidade, conforto e qualidade de vida, além de diminuir drasticamente qualquer gasto com estrutura física que poderia ter

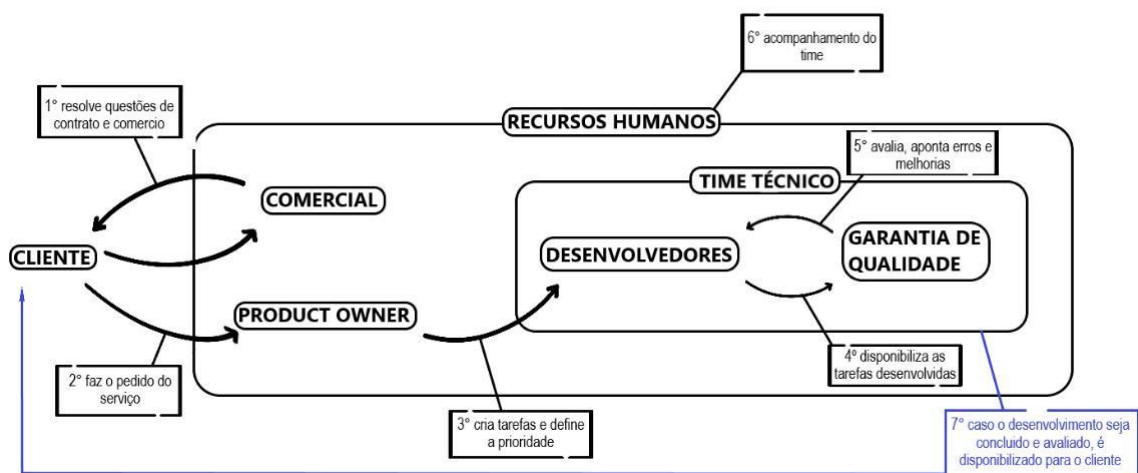
¹Informações adicionais sobre a Klok Tech em: <https://www.klok.tech>

seguinto o modelo presencial. Atualmente a empresa conta com quase 20 funcionários diversificados em mais de um estado, operando em dezenas de varejistas nacionais, que concentram suas informações na Klok Tech que tem comunicação com diversas processadoras de pagamento entre elas podemos citar a RPE² como exemplo.

3.2 Organização da empresa

A Klok tem uma estrutura de organização que boa parte das empresas de software adotam quando usam a metodologia ágil Scrum (Figura 2). Algumas partes da empresa estão entre comercial, product owner (PO) e o time técnico, e dentro desse setor técnico estão as equipes de desenvolvimento e a de qualidade, essas partes compartilham informações e trabalham em conjunto para atingir as metas sejam elas em benefício do cliente ou da própria empresa.

Figura 2 – Abstração da organização da empresa



Fonte: Própria do autor (2023).

Vale destacar que cada peça dessa estrutura é fundamental, e deve ser detalhada mais profundamente, em relação aos recursos humanos (RH), ela tem o dever de acompanhar o relacionamento dos funcionários como em quaisquer outros cenários, e esse comportamento se mantém mesmo a empresa tendo seu funcionamento de forma remota,

2 Informações adicionais sobre a RPE em: <https://rpe.tech>

entre as atividades que podem ser citadas estão a de recrutamento, administração de pessoas, avaliação de desempenho e velar pelas relações trabalhistas.

A equipe comercial tem a preocupação de buscar parceiros e propagar a reputação da empresa para possíveis clientes interessados, para os clientes cativados existe um requerimento para o PO, que abstrai as solicitações do cliente e transfere esses pedidos em formato de tarefas que são tocadas pelos desenvolvedores posteriormente. Sobre o time de QA que é foco principal neste relatório, quando os desenvolvedores finalizam as tarefas estimadas, elas são repassadas para serem validadas.

4. EMBASAMENTO TEÓRICO

Essa seção destaca algumas tecnologias, ferramentas e métodos que foram utilizadas durante o tempo do discente enquanto estagiário, e para melhor compreensão futura, sendo mencionados apenas de forma resumida e objetiva.

4.1 Metodologias ágeis (Scrum e Kanban)

Em geral a Klok adota a metodologia ágil Scrum, que no curso de ADS também foi abordada na disciplina de DAC, que segundo Ken Schwaber e Jeff Sutherland “Um framework dentro do qual pessoas podem tratar e resolver problemas complexos e adaptativos, enquanto produtiva e criativamente entregam produtos com o mais alto valor possível”, essa metodologia foi adotada por ser simples e eficaz tornando mais ágil o processo interno da empresa.

O Scrum tem características que envolvem times, eventos e artefatos, sobre o time, além da equipe de desenvolvimento, o PO é responsável por aumentar o valor do produto e trabalho da equipe técnica, além disso em declaração de Luiz Duarte “É ele que faz a ponte entre o Time Scrum e o mercado/clientes, sendo a sua palavra soberana neste

assunto”, em adição a presença de um Scrum Master sendo ela a pessoa que monitora se as práticas do Scrum estão sendo realizadas.

Sobre os eventos são marcados por reuniões diárias, planejamentos e revisões que acontecem ao decorrer da Sprint, proporcionando uma abordagem estruturada que maximiza a eficiência do processo. Essa abordagem é fundamental, visto que a Sprint é uma “janela” de tempo na qual as tarefas são realizadas, permitindo um controle mais preciso e adaptativo do desenvolvimento.

E por último os artefatos compostos por backlogs de produto e Sprint, que são as divisões das tarefas que ocorrem durante o planejamento da próxima Sprint, e todas essas características são adotadas pela empresa.

O Kanban é o método que a Klok adota de gerenciamento de fluxo de trabalho, nesse método são criados cartões com as descrições e ao decorrer da sua maturidade vão sendo movidas por colunas que apontam o andamento da produção da tarefa até chegarem na etapa de concluídas.

4.2 Discord

A Klok utiliza como meio de comunicação principal o aplicativo de voz discord³, é um aplicativo gratuito originalmente feito para comunidades de jogos, mas com o tempo começou a ser usado para outros setores, como organização de empresa e criação de comunidades e grupos que têm o mesmo interesse de forma totalmente online, nele também é possível criar servidores e criar diferentes canais de comunicação.

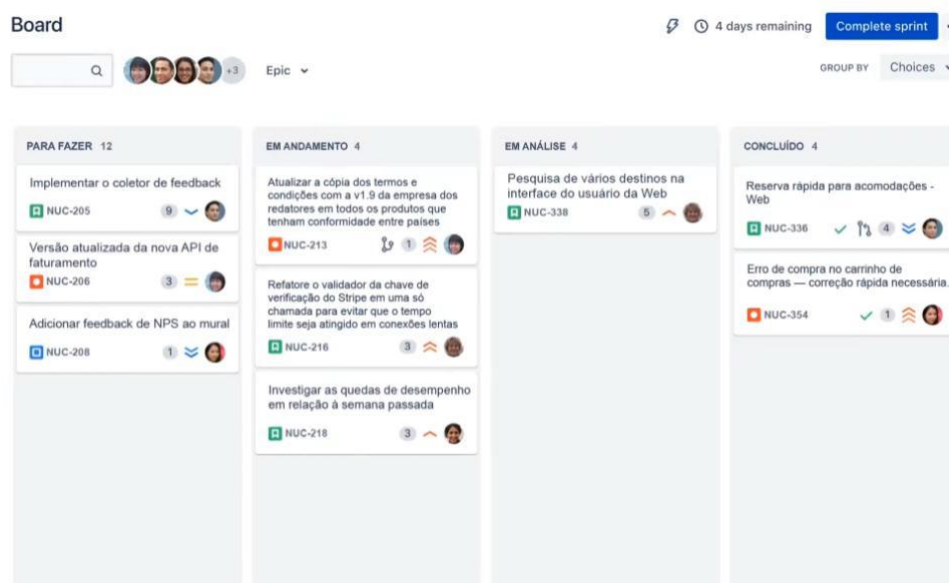
4.3 Jira

O Jira⁴ desenvolvida pela Atlassian, essa ferramenta ajuda a monitorar as tarefas de um projeto, é flexível e com diversos recursos personalizáveis, sendo possível criar etapas de trabalho, checklists, realizar controle de tempo entre outras formas de uso (Figura 3).

3 Informações adicionais sobre o Discord em: <https://discord.com/>

4 Informações adicionais sobre o Jira em: <https://www.atlassian.com/br/software/jira>

Figura 3 – demonstração de um quadro feito no Jira



Fonte: Atlassian Jira.

<https://www.atlassian.com/br/software/jira>

4.4 Behavior Driven Development (BDD)

O BDD é conhecido como uma técnica utilizada para documentação, desempenhando um papel crucial na clareza das especificações por meio da aplicação de normas de vocabulário bem definidas, com o objetivo de minimizar a dificuldade de comunicação e promover uma compreensão mais precisa entre as equipes envolvidas.

Tais normas quando seguidas envolvem regras de negócios e até linguagem de programação, para o entendimento amplo do time, de acordo com Wilker (2019) pode-se começar por onde for mais confortável, seja um código, método, classe ou um módulo específico menos complexo, mas também pode escalar e ser bem útil para situações mais complexas. Isso torna a prática de utilizar o BDD algo mais confortável no momento de descrever tarefas e testes.

4.5 Docker

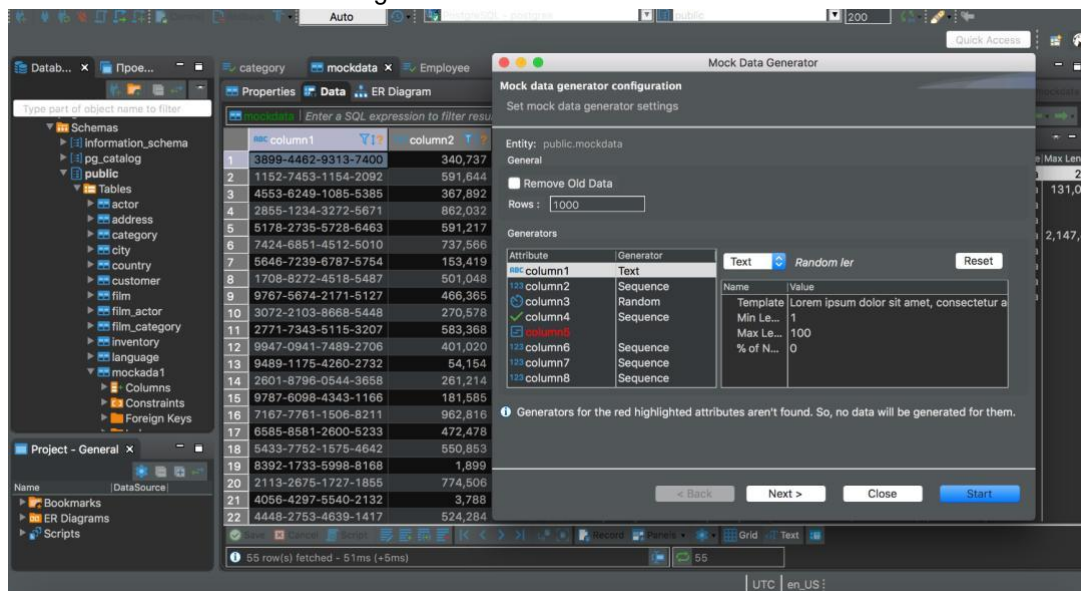
Docker é uma ferramenta em forma de software que é utilizado para inserir aplicativos em contêineres virtuais, dessa forma é possível utilizar várias aplicações sem precisar fazer configurações internas nas máquinas, isso facilita e torna viável que um mesmo projeto e/ou serviços rode em vários ambientes diferentes.

4.6 Postman

Utilizada como uma API client, ou seja, usada de apoio para os desenvolvedores consultarem APIs para teste ou que podem estar em fase de construção, essa ferramenta facilita a criação e compartilhamento das solicitações HTTP e HTTPS apresentando diversas formas de leitura das respostas.

4.7 Dbeaver

Figura 4 – Interface do Dbeaver⁵



Fonte: Site oficial do Dbeaver

<https://dbeaver.io>

5 Informações adicionais sobre o Dbeaver em: <https://dbeaver.io>

Esse software como mostrado na Figura 4, é utilizado como uma ferramenta de administração de banco de dados relacionais, com ele é possível manipular vários tipos de bancos de dados, como Postgres, MySql, entre outros, é bastante utilizada por ter uma interface amigável e mais intuitiva.

4.8 Pact

Pact⁶ é utilizado para realizar testes de contrato, que em resumo são verificações entre as mensagens compartilhadas entre aplicativos tornando essa verificação uma espécie de contrato entre o provedor e o consumidor, segundo “Este contrato garante que, se suas regras forem seguidas, toda comunicação entre as partes ocorrerá conforme o esperado” (APARECIDO, 2021), utilizando esse tipo de teste é possível validar se ambas as partes estão cumprindo com o que foi documentado.

4.9 MockMVC

Faz parte do pacote do Spring test, que acompanha o framework de desenvolvimento Spring⁷, o MockMVC é utilizado para testes de integração, usado para fazer simulação do comportamento usando em requisições HTTP para diferentes tipos de controladores e verificar as respostas.

Essa ferramenta tem características poderosas que ajudam nos testes, além das simulações com requisições e respostas, consegue isolar testes, pode ser usada em conjunto com mais ferramentas de teste com Junit, configuração flexível e fácil e principalmente uma ótima integração com o Spring, Pode também ser utilizada em conjunto com servidores para desempenhar o papel de lidar com as solicitações.

4.10 Jmeter

⁶ Informações adicionais sobre o Pact em: <https://docs.pact.io>

⁷ Informações adicionais sobre o Spring em: <https://spring.io>

Usamos o Jmeter⁸ para executar testes de carga e testes de estresse, com ele conseguimos fazer testes com uma grande quantidade de requisições com configurações diferentes para um servidor ou API, essas configurações podem ser personalizadas com quantidade de usuários e requisições em determinado tempo, com isso podemos analisar desempenho ou replicar erros, a ferramenta também disponibiliza relatórios e métricas visuais para facilitar o acompanhamento dos testes.

4.11 sMockin e localStack

O sMockin⁹ é uma ferramenta que é usada para simular respostas HTTP, podem ser configuradas e reutilizadas, com ele é possível replicar casos de exceção ou analisar qualquer tipo de resposta que a aplicação fornece a partir de uma requisição. LocalStack usamos para simular serviços da AWS, mais frequentemente mensagens SQS, que é um serviço de mensagem da Amazon, com essa ferramenta podemos criar filas para tornar os testes mais eficientes e ágeis.

5. RELATO DE ESTÁGIO

5.1 Inserção na empresa

O primeiro contato do discente com a Klok, se deu através de uma vaga de estágio compartilhada pelos meios de comunicação da coordenação do curso de ADS.

Na vaga divulgada existiam diversos pontos que eram interessantes e descreviam a oportunidade de conhecer mais sobre a área de qualidade que é fundamental para diversas empresas no mercado atual.

O processo seletivo se iniciou pelo envio de currículos para o RH da empresa, aos quais os mais atrativos para o interesse da empresa eram selecionados para uma entrevista em grupo.

⁸ Informações adicionais sobre o Jmeter em: <https://jmeter.apache.org>

⁹ Informações adicionais sobre o sMockin em: <https://www.smockin.com>

O discente após ter sido selecionado, teve participação na entrevista para expor sua forma de se expressar em grupo, comportamento e comunicação, além de ser avaliar através de perguntas os conhecimentos e experiências dos candidatos com as tecnologias citadas na descrição da vaga.

No dia seguinte foi enviado um desafio técnico para os participantes, o desafio tinha como objetivo simular um usuário buscando e escolhendo produtos para compra, validando todos os possíveis retornos, na documentação fornecida para a prática tinham dicas de ferramentas que poderiam ser utilizadas, entre os recursos indicados tinham diversos tipos de linguagens como Java, Python e JavaScript e frameworks de testes como Selenium, Junit e Cucumber

O desafio consistia em criar testes automatizados em um site e-commerce. Ao final do tempo estimado para a entrega dos desafios alguns pontos foram avaliados, como os casos de testes, documentação e estrutura, ao final o discente obteve destaque entre os demais participantes conseguindo uma reunião para o oferecimento da vaga de estágio, que foi aceita.

A data estimada para o início do estágio foi no dia 11 de julho de 2022, em que o discente foi recebido pelo líder do time de qualidade da Klok Tech, Jânio Paz Amorim Júnior, que foi responsável por mostrar os serviços e funcionamento da empresa e como iria ser o decorrer do estágio, horários e reuniões, além de vir a se tornar seu supervisor.

O aluno foi apresentado aos demais integrantes no decorrer da daily (reunião diária do método Scrum para plano de ação), normalmente realizada na plataforma do discord, após foi demonstrado ferramentas e frameworks, como exemplo utilizar os serviços e configurar os ambientes no docker, utilizando filas simuladas no localstack e simular retornos pelo smoking, também foi explicado como as tarefas eram entregues pela plataforma do Jira e sua movimentação com o método Kanban, isso para preparar o discente para a inicialização de suas funções na empresa.

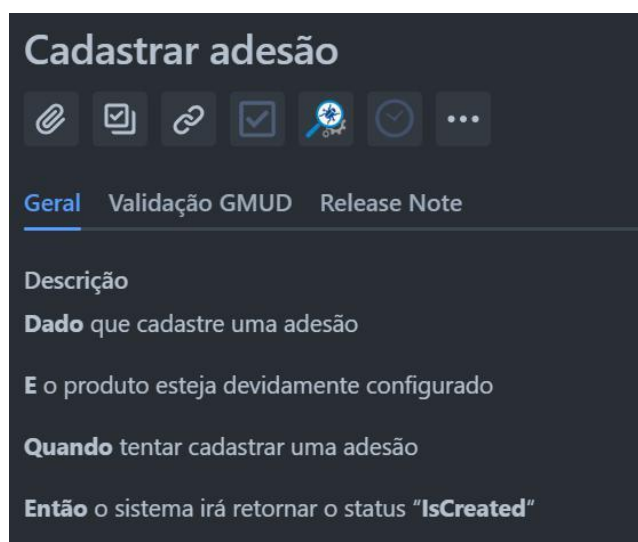
5.2 Registro de testes

Após ter conhecimento de como os serviços e a base de dados da empresa funcionavam, foi dada a tarefa de criar alguns casos de testes e documentá-los, para criar esses casos era necessário que todos fossem feitos utilizando BDD, uma técnica de documentação, que tem como característica mais marcante utilização de palavras específicas, essas palavras são Dado, E, Quando e Então.

A documentação dos testes é algo fundamental para a realização deles, com uma boa documentação é possível estimar quais ferramentas serão utilizadas para os testes, além de ter uma maior noção do fluxo que o produto ou serviço que está sendo testado pode fazer, com isso podemos apontar seus comportamentos, restrições e qualquer outra característica que a aplicação possa apresentar e se já era ou não esperado através do entendimento ao qual foi documentado.

Abaixo está um exemplo de como a documentação é realizada juntamente com a utilização da plataforma do Jira (Figura 5).

Figura 5 – Registro de caso de teste



Fonte: Própria do autor (2022)

As documentações que o discente realizou eram separadas em algumas sendo de testes existentes já nos serviços da empresa, e outros de futuros

possíveis casos de teste, ao final, o supervisor avaliava o que foi registrado, e apontava possíveis melhorias e correções.

5.3 Testes manuais

Testes manuais tem uma importância muito alta quando se fala no básico de qualidade de software, em sua maior parte é nesses cenários que a criatividade do QA trabalha para a criar diversos tipos de cenários de testes, "Os testes manuais de software são como a arte de desvendar os segredos ocultos do código, explorando cada funcionalidade em busca de falhas que escapariam à atenção das máquinas." (RAFAELA e JOÃO), isso torna muito mais desafiador testar os comportamentos sejam eles previsíveis ou não.

Segundo (BERNER, WEBER e KELLER, 2005) "verifica-se que os testes automatizados são bastante úteis para substituição de tarefas repetitivas, no entanto novos defeitos são em geral descobertos com a utilização de testes manuais", isso significa que os testes manuais podem captar particularidades escondidas nos fluxos de cada serviço sem utilizar ferramentas de automatização.

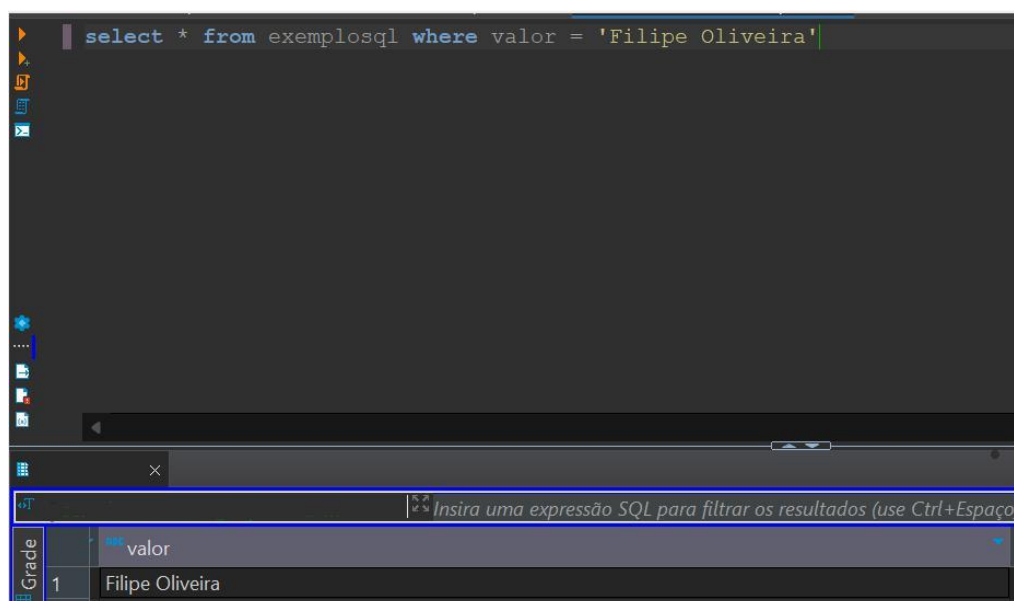
A introdução do discente aos testes manuais da empresa foi dada pelo supervisor, apresentando algumas tarefas que tinham a necessidade de serem testadas manualmente, esses testes em sua maior parte tinham a necessidade de validação seja ela visual do banco ou dos resultados de requisições via APIs(conexões com serviços e/ou soluções de software), para as validações foram usadas principalmente o Postman e o Dbeaver. Utilizando o postman para as requisições como na Figura 6, era validado a resposta que era retornada e como os resultados ficavam no banco de dados que era possível visualizar através do Dbeaver (Figura 7).

Figura 6 – Exemplo de requisição HTTP no Postman



Fonte: Própria do autor (2023)

Figura 7 – Exemplo de busca no banco usando Dbeaver



Fonte: Própria do autor (2023)

Para validar se o fluxo estava correto, era necessário acompanhar uma descrição que o PO deixava atrelada a demanda abstraída do cliente, e que os DEVs utilizavam para poder desenvolver, com isso o QA tinha a função de garantir que as solicitações feitas foram atendidas e se houve impacto em demais funcionalidades que fogem do escopo da tarefa, caso seja percebida alguma correção, também é função do responsável pelos testes apontar os erros e validar as possíveis correções.

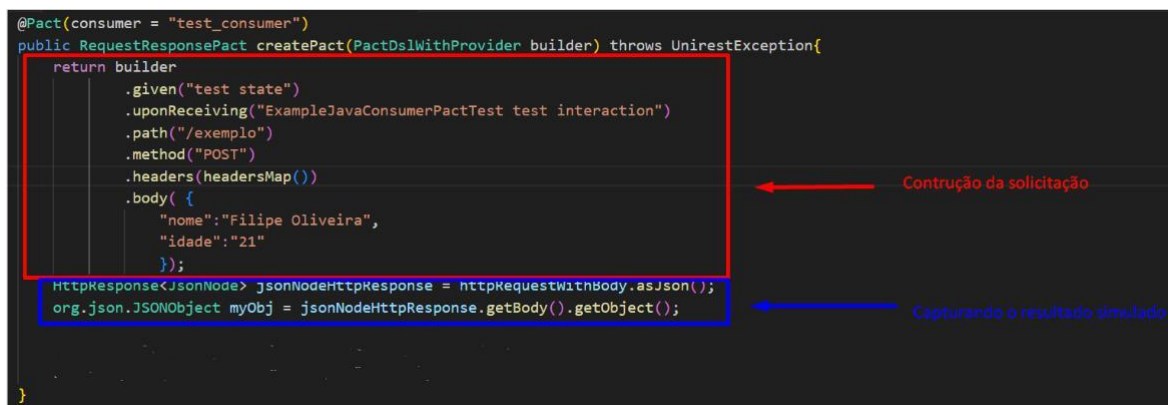
5.4 Testes de contrato

No decorrer das funções do aluno foi apresentada uma nova tarefa, de estudar e tentar implementar um exemplo utilizando os serviços da empresa, e o framework de testes de contrato nomeado de Pact. Podemos considerar também que “Os testes de contrato (ou CDCT-Consumer-Driven Contract Testing) são uma ótima alternativa aos testes end-to-end. Além de serem mais rápidos e menos propensos a erros” (EDUARDO, 2020), por firmarem um acordo entre um serviço que consumidor e outro provedor, onde ambas as partes precisam cumprir com o solicitado entre elas.

Ao final dos estudos o discente desenvolveu um protótipo utilizando um provedor simulado, semelhante a Figura 8, em que era analisado a requisição fornecida pelo consumidor e a resposta devolvida pelo provedor por meio de um resultado já configurada, com isso tanto a requisição HTTP quanto a resposta são verificadas no contrato (Figura 9).

Figura 8 – Exemplo método consumidor simplificado

```
@Pact(consumer = "test_consumer")
public RequestResponsePact createPact(PactDslWithProvider builder) throws UnirestException{
    return builder
        .given("test state")
        .uponReceiving("ExampleJavaConsumerPactTest test interaction")
        .path("/exemplo")
        .method("POST")
        .headers(headersMap())
        .body( {
            "nome": "Filipe Oliveira",
            "idade": "21"
        });
    HttpResponse<JsonNode> jsonResponse = httpRequestWithBody.asJson();
    org.json.JSONObject myObj = jsonResponse.getBody().getObject();
}
```



Fonte: Própria do autor (2023)

Figura 9 – Contrato exemplo em Json



Fonte: Própria do autor (2023)

Ao final foi repassado para o supervisor para avaliar o teste efetuado pelo discente, apontar melhorias, correções e a possibilidade de implementar os testes de contrato futuramente nas interações da empresa.

5.5 Testes de carga

Os testes de carga servem para mensurar a eficiência do serviço, eles ajudam a analisar possíveis problemas de desempenho e estabilidade em situações que deixam a aplicação sob estresse, portanto é importante que todas as novas demandas não prejudiquem a capacidade do sistema em ambiente de produção.

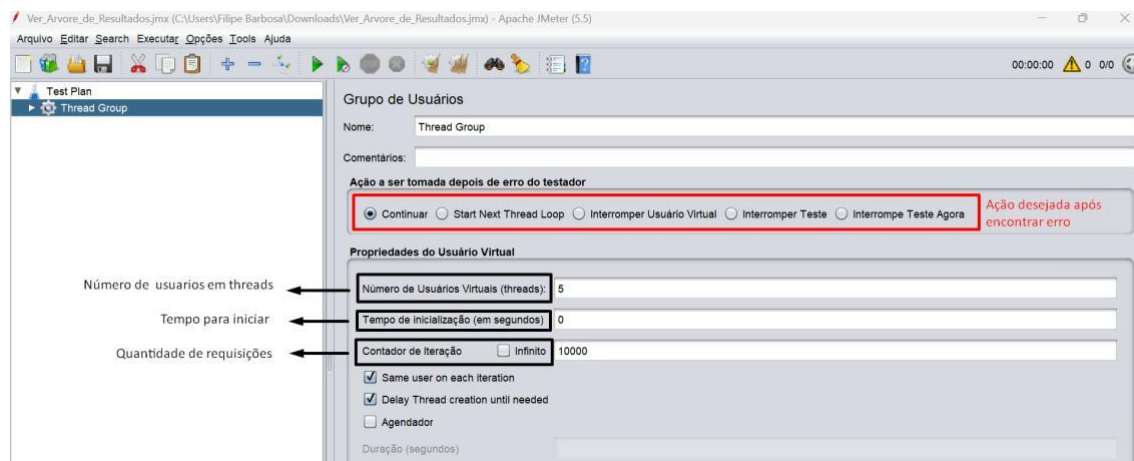
Vale frisar que os testes são feitos em ambientes mais próximas ao realmente utilizados pelos clientes, para descobrir possíveis gargalos entre outros problemas, portanto podemos dizer que "Testes de carga são fundamentais para avaliar o desempenho de sistemas em nuvem, permitindo que sejam identificados gargalos e otimizações necessárias para garantir uma experiência confiável aos usuários." (Ranganathan & Buyya, 2010, p. 1590), principalmente em datas comemorativas como natal e ano novo em que o nível de utilização da maioria dos serviços são

potencializados, em busca de ter maior segurança perante maior quantidade de solicitações são necessários esses casos de teste de carga.

Algumas das tarefas que eram fornecidas pelo PO tinham a necessidade de simular uma quantidade considerável de interações simultâneas para checar o desempenho da solução fornecida pelo DEV, para isso foi apresentado ao discente uma ferramenta específica para testes de carga chamada Jmeter.

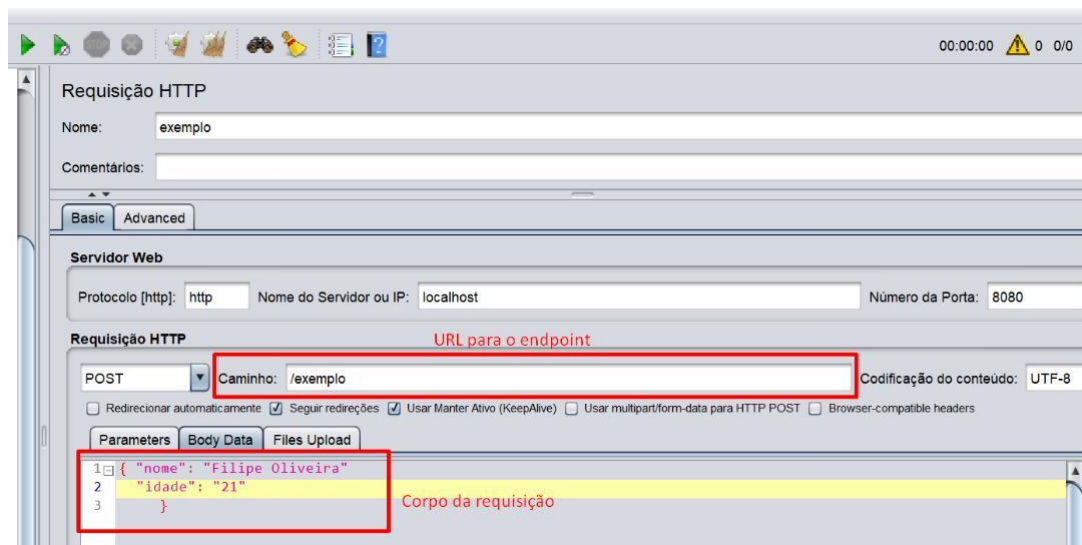
O Jmeter por ser um software eficiente e com diversas funcionalidades foi optado pelo time para ser utilizado para os testes de carga, os testes consistem em pré-determinar um número de usuários e uma quantidade de requisições a serem feitas em um certo período, a Figura 10 e 11 mostram um exemplo de como é a configuração da ferramenta.

Figura 10 – tela de configuração de requisições do Jmeter



Fonte: Própria do autor (2023)

Figura 11 – Exemplo da preparação da requisição no Jmeter



Fonte: Própria do autor (2023)

Após os testes, caso a quantidade de requisições fosse considerada interessante e coerente para a validação da solução e claramente não afetasse a estabilidade do serviço, a tarefa era considerada apta para subir em ambiente de produção.

Vale ressaltar que os testes executados são para determinar o quanto o serviço suporta e caso erros ocorram, que não interrompam a execução contínua da aplicação, esse objetivo de testar os limites de um serviço é característica dos testes de estresse, que não serão abordados aqui.

5.6 Testes de integração

Durante o tempo do discente no estágio, umas das atividades mais importante foram sem dúvidas os testes de integração, sendo estes testes verificando o comportamento entre as interações de duas ou mais unidades que foram desenvolvidas e que agora precisam se comunicar corretamente.

O objetivo desses testes é encontrar possíveis problemas após essas junções entre as diferentes partes do sistema, segundo Pressman (2016) “Os testes de integração revelam os muitos pontos fracos dos grupos de desenvolvimento de software que não formaram uma equipe coesa e consistente”, esses tipos de teste são muito importantes para definir se a

equipe está unida e coordenada, além de prevenir possíveis erros entre a comunicação entre os próprios serviços.

Baseado em suas experiências foi solicitado ao discente para realizar testes de integração utilizando MockMvc e a linguagem Java, com esse framework, foi possível para as atividades montar as requisições HTTP e checar suas respostas para validação como na Figura 12.

Figura 12 – Exemplo estruturação e teste de integração com MockMvc

```

    @Autowired
    protected MockMvc mockMvc;

    protected Exemplo exemplo;

    @Autowired
    protected ObjectMapper objectMapper;

    @BeforeAll
    public void setup() throws Exception {
        this.exemplo = new Exemplo("Filipe", -21);
    }

    @Test
    public void exemplo() throws Exception {
        mockMvc.perform(post("localhost:9999/url/exemplo")
            .header("Authorization")
            .contentType("application/json")
            .content(objectMapper.writeValueAsString(this.exemplo)))
            .andExpect(status().isCreated())
            .andExpect(jsonPath("$.data.nome").value("Filipe"));
    }

```

Atributos necessários para o teste

Setup para resetar cada teste

Criação do corpo da requisição

Verificação do esperado

Fonte: Própria do autor (2023)

5.7 Testes manuais de interface

Por último, com a chegada de um protótipo de front-end desenvolvido em Angular¹⁰ na empresa, o discente, em conjunto com o restante do time de QA, juntos iniciaram a tarefa de explorar a nova interface. Assim como os testes manuais descritos anteriormente, os testes realizados no front não tiveram a utilização de automatização, porém durante os testes, foi possível detectar problemas visuais e de design que, após serem reportados e retrabalhados, não apenas melhoraram a estética, mas também serviram para tornar a experiência do cliente final mais agradável e funcional.

10 Informações adicionais sobre o Angular em: <https://angular.io/>

6. CONCLUSÃO

Posso afirmar que a experiência proporcionada pela Klok Tech foi imensamente evolutiva para o discente, onde foi possível colocar em prática o conhecimento adquirido ao longo do curso de ADS, fazendo com que todas as ferramentas e métodos ensinados ao longo do estágio fossem assimilados, além de aperfeiçoar as técnicas e o conhecimento do mesmo, que sem dúvida agregou para o andamento do aluno durante a etapa final do curso.

Essa oportunidade trouxe consigo uma grande carga de contato real com o mercado de trabalho, que sem dúvida é algo para ser bastante valorizado, esse tipo de contato é de diversas formas engrandecedor, por trabalhar não apenas os aspectos técnicos, mas também acompanhar o cotidiano de uma empresa, participar dos eventos e trabalhar com pessoas de diversas áreas e lugares diversos, vale ressaltar que boa parte da boa impressão passada ao discente se deu pelo ótimo recebimento do discente na empresa Klok tech, em especial ao supervisor Jânio Paz Amorim Júnior, responsável por disseminar seu conhecimento no mercado.

Sobre as atividades desenvolvidas, o discente se sentiu desafiado, mas muito motivado para a obtenção de novos conhecimentos na área de QA, sobre os diferentes tipos de testes com testes manuais, testes de integração, testes de contrato e testes de carga, juntamente com o sentimento de agregação em cumprir suas tarefas e atingir as metas da empresa juntamente com a equipe.

Por último o discente conseguiu atingir maturidade suficiente dentro do ambiente em que estava, graças ao seu esforço e dedicação atingindo assim em menos de 10 meses, em meados do final do mês de março de 2023, a efetivação como QA Júnior na Klok Tech, e o desejo de continuar tendo um bom desempenho, e avançar em novos conhecimentos agora definitivamente no mercado de trabalho.

REFERÊNCIAS

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de software**: uma abordagem profissional. [S. l.: s. n.], 2016. cap. 3, p. 633-635.

DUARTE, Luiz. **Scrum e Métodos Ágeis**: Um Guia Prático, LuizTools, [S. l.: s. n.], 2016

PAULA, WILKER HUDSON. QUALIDADE DE SOFTWARE E DESENVOLVIMENTO DIRIGIDO POR COMPORTAMENTO: **BDD: UM ESTUDO DE CASO**, [s. l.], 2019.

RODRIGUES, Charles aparecido. **Testes de contrato – O que, quando e por quê?**. [S. l.], 7 dez. 2021. Disponível em: <https://www.zup.com.br/blog/testes-de-contrato>. Acesso em: 15 jul. 2023.

MYERS, G.F. **The Art of Software Testing**. 1a edição, John Wiley and Sons, 1979

"IEEE **Standard for Software Test Documentation**," in *IEEE Std 829-1998*, vol., no., pp.1-64, 16 Dec. 1998, doi: 10.1109/IEEESTD.1998.88820.

GERAS, Adam. **Leading Manual Test Efforts with Agile Methods**. Agile 2008 Conference, 2008.

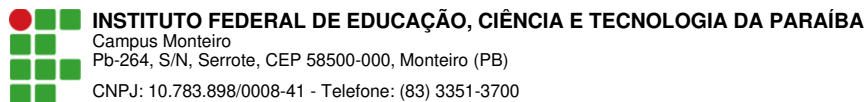
BERNER, Stefan; WEBER, Roland; Keller, Rudolf; **Observations and Lessons Learned from Automated Testing**, Proceedings of International Conference on Software Engineering, 2005.

MOLINA, Rafaela; SILVA, João. **Manual vs. Automated Testing**: A Comparative Study in Software Testing. In: 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC). IEEE, 2018. p. 707-712. DOI: 10.1109/COMPSAC.2018.00096.

CELESTE, E. P. **Introdução aos Testes de Contrato com Pact**. [S. l.], 21 de abril. 2020 Disponível em:

<https://medium.com/beyondtest/introdu%C3%A7%C3%A3o-aos-testes-de-contrato-com-pact-2b7e7a63ef0> . Acesso em: 26 jul. 2023.

RANGANATHAN, C.; BUYYA, R. **Performance Metrics and Benchmarks for Cloud Computing**. Concurrency and Computation: Practice and Experience, v. 22, n. 13, p. 1586-1611, 2010.



Documento Digitalizado Ostensivo (Público)

relatório de estágio

Assunto: relatório de estágio
Assinado por: Filipe Barbosa
Tipo do Documento: Relatório
Situação: Finalizado
Nível de Acesso: Ostensivo (Público)
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- **Filipe de Oliveira Barbosa, ALUNO (202015020013) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - MONTEIRO**, em 19/10/2023 23:12:57.

Este documento foi armazenado no SUAP em 19/10/2023. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 975323
Código de Autenticação: bc87c6e6c0

