



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS MONTEIRO
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

FILIPE FARIAS CHAGAS

**RELATO DE EXPERIÊNCIA PROFISSIONAL NA ÁREA DE DESENVOLVIMENTO
DE SOFTWARE NA EMPRESA KLOK TECH.**

MONTEIRO - PB
2023

FILIFE FARIAS CHAGAS

**RELATO PROFISSIONAL NA ÁREA DE DESENVOLVIMENTO DE SOFTWARE
NA EMPRESA KLOK TECH.**

Relatório de Estágio apresentado à Coordenação de Estágio do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Campus Monteiro, como requisito parcial para a conclusão do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Me. Julierme Silva de Araújo

**MONTEIRO - PB
2023**

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

C433r Chagas, Filipe Farias.

Relato de experiência profissional na área de desenvolvimento de software na Empresa KLOK TECK / Filipe Farias Chagas – Monteiro-PB. 2023.

46fls. : 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. Me. Julierme Silva de Araújo.

1. Software - Desenvolvimento 2. Arquitetura – Microsserviços 3. Empresa KLOK TECK - Home Office I. Título.

CDU 004.453:004.273

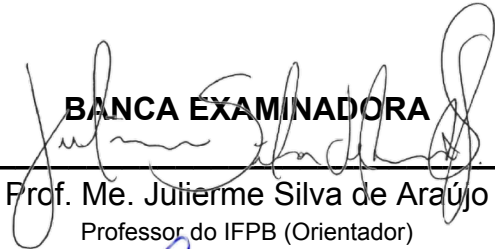
FILIFE FARIAS CHAGAS

RELATO DE EXPERIÊNCIA PROFISSIONAL NA ÁREA DE DESENVOLVIMENTO DE SOFTWARE NA EMPRESA KLOK TECH.

Relatório de Estágio apresentado à Coordenação de Estágio do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Campus Monteiro, como requisito parcial para a conclusão do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas.

Aprovado em 29 de agosto de 2023.

BANCA EXAMINADORA



Prof. Me. Julierme Silva de Araújo
Professor do IFPB (Orientador)



Prof. Esp. Wagner de Oliveira Santos
Professor do IFPB (Examinador)



Prof. Esp. Gilmar de Jesus Barros
Professor do IFPB (Examinador)

AGRADECIMENTOS

Agradeço primeiramente ao meu falecido irmão Fábio Farias por todo o suporte enquanto ainda presente, e também aos meus pais Flaviano Saraiva e Izabel Cristina, minha irmã Flávia Farias e meu sobrinho Christopher Theodore por todo o apoio nos momentos difíceis.

Aos meus amigos e colegas que participaram da minha jornada no curso e na vida, em especial Gabryel Alexandre, Pedro Eduardo, Lucas Gonçalves e Horácio Honorato pela presença insubstituível na minha vida.

Ao Instituto Federal da Paraíba Campus Monteiro e a todo o corpo de docentes, que colaboraram no crescimento pessoal e profissional através do curso de Análise e Desenvolvimento de Sistemas, principalmente aos docentes Roberto Ranniere e Cleyton Caetano.

A empresa Klok Tech que me deu a oportunidade profissional para poder demonstrar minhas capacidades e dedicação. Agradeço especialmente ao meu gestor e supervisor Fagner Silva de Lima, por sempre ser solícito e me auxiliar na minha primeira experiência profissional. Agradeço também a Jânio Paz de Amorim Júnior, por contribuir na evolução pessoal e profissional.

Também agradeço ao meu professor de Programação 1 e orientador do estágio, o Me. Julierme Silva de Araújo por ter feito a introdução do discente na área da tecnologia.

RESUMO

O presente relatório tem como objetivo apresentar as experiências profissionais e atividades realizadas pelo discente do curso de Análise e Desenvolvimento de Sistemas no Instituto Federal da Paraíba no Cargo de Desenvolvedor de Software durante o estágio supervisionado na empresa Klok Tech durante o período de 18 de maio de 2022 a 30 de novembro de 2022 sob a modalidade *Home Office*. Colocando em prática os fundamentos adquiridos durante o período da graduação e traçando um paralelo com as tarefas desenvolvidas, assim atuando efetivamente no análise, desenvolvimento e manutenção de aplicações Java com o framework Spring e Spring Boot, utilizando metodologias de desenvolvimento ágil como o Scrum e Kanban, e de sistemas de gerenciamento de bancos de dados como o PostgreSQL, e empregando os princípios da engenharia de software e padrões de projeto, como também atuando no desenvolvimento de aplicações Javascript com Node.js serverless para execuções na nuvem no ambiente da AWS.

Palavras-chave: estágio supervisionado; desenvolvimento de software; arquitetura de microsserviços; fullstack; metodologias ágeis; Klok Tech;

ABSTRACT

This report aims to present the professional experiences and the activities performed by the student of the Systems Analysis and Development course at the Federal Institute of Paraíba at the role of Software Developer during the supervised internship at Klok Tech during the period from May 18, 2022 to November 30, 2022 under the Home Office modality. Putting into practice the fundamentals acquired during the graduation period and drawing a parallel with the tasks developed, thus effectively acting in the analysis, development and maintenance of Java applications with the Spring and Spring Boot framework, using agile development methodologies such as Scrum and Kanban, and database management systems such as PostgreSQL, and employing software engineering principles and design patterns, as well as developing serverless Node.js applications for cloud execution in the AWS environment.

Palavras-chave: supervised internship; software development; microservices architecture; fullstack; agile methodologies; Klok Tech;

LISTA DE FIGURAS

Figura 1 - Logomarca da Klok Tech.....	13
Figura 2 - Fluxo do Scrum.....	22
Figura 3 - Fluxo do Kanban Klok.....	25
Figura 4 - Fluxo do Gitflow.....	28
Figura 5 - Arquitetura Multi-tenant.....	31
Figura 6 - Integração entre serviços da plataforma Klok.....	33
Figura 7 - Exemplo de dashboard de vendas.....	35
Figura 8 - Dashboard geral de vendas.....	36
Figura 9 - Diagrama de sequência do SMS Service.....	38

LISTA DE SIGLAS, ABREVIATURAS E ACRÔNIMOS

ADS	Análise e Desenvolvimento de Sistemas
API	Application Programming Interface
AWS	Amazon Web Services
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
IFPB	Instituto Federal da Paraíba
JVM	Java Virtual Machine
MR	Merge Request
OOP (POO)	Object-oriented programming (Programação Orientada a Objetos)
PM	Project Manager
PO	Product Owner
QA	Quality Assurance
REST	Representational State Transfer
RPE	Retail Payment Ecosystem
SFTP	Secure File Transfer Protocol
SM	Scrum Master
SMS	Short Message Service (Serviço de mensagens curtas)
SQL	Structured Query Language
TI	Tecnologia da Informação

SUMÁRIO

1 INTRODUÇÃO.....	10
2 OBJETIVOS.....	12
2.1 OBJETIVO GERAL.....	12
2.2 OBJETIVOS ESPECÍFICOS.....	12
3 RELATO DE ESTÁGIO.....	13
3.1 KLOK TECH.....	13
3.2 DADOS REFERENTES AO ESTÁGIO.....	14
3.3 PERFIL DA EMPRESA E PARTICIPAÇÃO NO MERCADO.....	15
3.4 DIVISÕES DA EMPRESA, OS SEUS PROCESSOS E O AMBIENTE.....	16
3.5 TAREFAS EXERCIDAS.....	18
3.5.1 Descrição dos projetos.....	19
3.5.1.1 Hub.....	19
3.5.1.2 Provider.....	20
3.5.1.3 Customer.....	20
3.5.1.4 Kear Web.....	21
3.5.1.5 Serviços secundários.....	21
3.5.2 Processos de desenvolvimento.....	21
3.5.2.1 Scrum.....	22
3.5.2.1.1 Sprint Daily.....	23
3.5.2.1.2 Sprint Planning.....	23
3.5.2.1.3 Sprint Retrospective.....	23
3.5.2.1.4 Scrum Master.....	24
3.5.2.2 Kanban.....	24
3.5.2.3 Ferramentas auxiliares.....	26
3.5.2.3.1 Discord.....	26
3.5.2.3.2 PlanItPoker.....	26
3.5.2.3.3 Jira.....	26
3.5.2.3.4 Git, Gitlab e GitFlow.....	27
3.5.3 Arquitetura da plataforma.....	29
3.5.3.1 Computação na nuvem e AWS.....	30
3.5.3.2 Arquitetura serverless e AWS Lambda.....	30
3.5.3.3 Arquitetura multitenancy.....	30
3.5.4 Configuração de produto.....	32
3.5.5 Code review.....	34
3.5.6 Kear Web - Dashboard Geral.....	35
3.5.7 Envio de SMS e SMS Service.....	37
4 CONCLUSÃO.....	39
REFERÊNCIAS.....	41
ANEXO A - Detalhes da Vaga.....	45
ANEXO B - Detalhes do Desafio.....	46

1 INTRODUÇÃO

O Instituto Federal da Paraíba (IFPB) oferece cursos para formação acadêmica, dentre essas formações, o de ensino superior. O objetivo de um curso de ensino superior é auxiliar na captação de conhecimentos e habilidades, com o intuito de dominar uma área de sua escolha para futuramente exercer uma profissão.

O curso ofertado de Análise e Desenvolvimento de Sistemas (ADS), tem o propósito de formar indivíduos capazes de analisar, desenvolver e manter projetos de software. E a partir disso focar em vários aspectos de um produto de software, desde a lógica de programação até a elaboração e construção de aplicações corporativas.

Entretanto a graduação em questão não oferece um real caso de experiência profissional, para isso o discente realizou um estágio supervisionado com o propósito de aplicar os conhecimentos aprendidos com o decorrer do curso, e aprimorar seus conhecimentos, além de buscar evoluir como profissional da área.

O estágio supervisionado foi feito na empresa Klok Tecnologia LTDA, mais conhecida como Klok Tech, uma empresa do âmbito de produtos financeiros que oferece a gestão de serviços financeiros, em principal, seguros de vida, residenciais, veículos automotivos, de equipamentos, assistência odontológica, entre outros.

A gestão desses produtos financeiros é feita através de uma API (Application Program Interface) REST (Representational State Transfer) de cadastro, processamento e gerenciamento de cada operação feita por um produto financeiro, essas operações sendo, adesões, cancelamentos, cobranças, estornos e pagamentos desses produtos. Como também de outros módulos de coleta de dados de um contratante de um produto, até o gerenciamento de um produto e suas integrações.

Na empresa todo o desenvolvimento é feito utilizando uma combinação de metodologias ágeis como o Scrum e o Kanban, e a utilização de boas práticas de programação como a de revisão de código, código limpo e arquitetura limpa, descritos no livros Clean Code e Clean Architecture (MARTIN, 2009, 2019).

Durante o estágio, o discente atuou na manutenção e refinamento dos projetos da empresa participando em principal da squad (equipe) de produto,

fazendo o desenvolvimento dos serviços Java e Node.js da empresa, e ocasionalmente na *squad* de *Onboarding*, tratando da configuração dos produtos financeiros e da coleta dos dados.

O documento apresentado está disposto em quatro seções, a primeira sendo a **introdução** que é a seção atual, a segunda seção são os **objetivos gerais e específicos do estágio** que trata dos objetivos do estágio e do discente, a terceira seção é o **relato de estágio** que aborda a empresa, processos, e atividades exercidas dentro do estágio em detalhe, e a quarta e última seção de **conclusão e agradecimentos**, todas estruturadas de modo a resguardar dados sigilosos da empresa de maneira que a compreensão das informações pelo leitor não sejam afetadas.

2 OBJETIVOS

2.1 OBJETIVO GERAL

Relatar as experiências profissionais do discente durante o estágio supervisionado na área de desenvolvimento e manutenção de software na empresa Klok Tech.

2.2 OBJETIVOS ESPECÍFICOS

- Apresentar brevemente a Klok Tech, seus processos, seus produtos e sua participação no mercado e a arquitetura da plataforma.
- Descrever as tarefas exercidas pelo discente durante o estágio supervisionado, de modo a relacionar os conhecimentos adquiridos durante o curso da graduação, com as práticas usadas no ambiente de trabalho e os conhecimentos assimilados durante o estágio.
- Trazer paralelos entre o conteúdo ensinado durante o período da graduação, e apresentar a evolução interna da empresa e do discente no decurso do estágio.
- Externar pontos positivos e adversidades sentidas pelo discente como estagiário.

3 RELATO DE ESTÁGIO

3.1 KLOK TECH

A empresa Klok Tech é uma empresa que atua na orquestração de integrações e serviços financeiros. Um serviço financeiro são processos feitos para facilitar ações como de transferências bancárias, pagamento de contas e folha de pagamento (JEHNIFFER, 2021), onde alguns exemplos de serviço financeiro são, bancos, casas de câmbio, e corretores. A Klok Tech foi fundada em março de 2021 e tem sua sede principal na Alameda Rio Negro, 503, Sala 2301, Barueri, São Paulo. O logotipo da empresa pode ser observado na ilustração ressaltada na Figura 1.

Figura 1 - Logomarca da Klok Tech.



Fonte: Klok Tech (2023)

Dado que a Klok Tech foi fundada durante a época da pandemia do COVID-19 (2019-ATUAL) a modalidade de trabalho *Home Office* foi adotada como padrão de funcionamento, de modo a flexibilizar a comunicação e trabalho do time e reduzir os custos com a infraestrutura da empresa, apesar da empresa disponibilizar um escritório para atividades presenciais caso desejado.

A empresa Klok Tech opera de maneira a fazer o papel da gestão de ativos de produtos financeiros, realizando as integrações de provedores de produtos financeiros com as principais processadoras de meios de pagamento como a RPE e a Dock (antiga Conductor), como também com qualquer outra empresa que processadora e provedor.

Dada o ambiente flexível da Klok, os meios de comunicação a distância são variados, seja o Discord para as cerimônias dos times de desenvolvimento, como a *daily*, a *planning* e a *retro*, o WhatsApp para comunicados gerais da empresa, e comunicações ágeis com parceiros, Gmail e Google Meet para conversas mais formais.

3.2 DADOS REFERENTES AO ESTÁGIO.

O discente teve acesso a vaga de estágio através do grupo de *e-mail* acadêmico do IFPB Campus Monteiro, a vaga foi disponibilizada no dia 28 de março de 2022, como descrito no ANEXO A, os requisitos para a seletiva de estágio na Klok foram experiência no *framework* Spring e Spring Boot, com a versão do Java 8 ou superior, conhecimentos em APIs Restful e SQL (Structured Query Language).

Para a admissão como estagiário na empresa, o discente passou por um processo interno de seleção, que consistiu em reuniões coletivas com outros candidatos para a apresentação da empresa e do discente, assim se apresentando ao Diretor de Tecnologia da empresa Jardell Antunes, como também por um desafio de programação para a avaliação do perfil do discente, os requisitos desse desafio estão descritos no ANEXO B, e que esses requisitos consistiram em dois serviços, o serviço X sendo uma API para o gerenciamento de adesões, cancelamentos, cobranças e pagamentos, e o serviço Y sendo para o processamento de cobranças, e que se comunicava com o serviço X através de mensageria utilizando RabbitMQ, ambos os serviços e o banco de dados sendo empacotados em containers Docker.

Após essa seletiva, o discente foi escolhido para um período de estágio supervisionado que teve início no dia 16 de maio de 2022 e finalizou efetivamente no dia 30 de novembro de 2022, contemplando a duração aproximada de 6 meses. Com a carga horária de 30 (trinta) horas semanais, o que totaliza uma carga horária total de 834 (oitocentos e trinta e quatro) horas ao fim do estágio. A carga horária em questão é adaptável o que não define um expediente diário, mas sim somente a carga horária necessária diariamente, apesar das reuniões como as *dailies*, *planning* e *retro* serem em um horário predeterminado.

O estágio teve início sob a mentoria e supervisão do desenvolvedor pleno Fagner Lima, porém durante todo o decorrer do estágio o time inteiro colaborou no crescimento profissional do discente. Durante os primeiros dois meses do estágio, o

discente passou por vários treinamentos com o supervisor para a adaptação a plataforma, esses treinamentos consistem em:

- Definição do mercado de serviços financeiros.
- Introdução a arquitetura de microsserviços e multi-tenant.
- Apresentação dos projetos hub, customer e provider.
- Exposição ao processo interno.
- Boas práticas de desenvolvimento.
- Configuração de ambiente de desenvolvimento.
- Infraestrutura, gerenciamento e manutenção da plataforma.
- Departamentos

Os treinamentos acima citados serão abordados posteriormente neste relatório.

As tarefas feitas pelo discente durante o percurso do estágio foram de nível apropriado para um estagiário, sendo empregadas atividades anteriormente vistas no âmbito acadêmico no percurso da graduação, como por exemplo, manutenção de código Java, criação de scripts SQL, entretanto, após um certo período e a pedido próprio, o discente requisitou tarefas de nível mais avançado para serem desenvolvidas, isso tendo o acompanhamento do time e em principal do supervisor, a fim de adquirir experiência além do que é ensinado no durante o curso.

Na extensão do estágio as demandas tiveram um efeito positivo no desenvolvimento do discente, dada a complexidade dessas tarefas e a assistência provida pelos integrantes da empresa, porém, apesar da escolha em buscar tarefas mais desafiadoras, o discente acabou por se sobrecarregar durante o estágio e impôs uma pressão própria apesar do suporte da empresa, algo que foi sendo superado a cada entrega, e com a acumulação de experiência do discente.

3.3 PERFIL DA EMPRESA E PARTICIPAÇÃO NO MERCADO.

A Klok é uma empresa de TI do ramo de soluções financeiras, sendo a mediadora e fazendo a conexão entre seguradores e provedores de produtos financeiros, sejam esses seguros financeiros, como seguro de vida, automobilístico, ou patrimonial, ou assistências odontológicas, de saúde, escolar (FIGUEIREDO, 2023), a redes varejistas, como supermercados, e lojas de departamentos (SAAB; GIMENEZ, 2000), e empresas de trâmite e processamento de meios de pagamento.

A propósito de elucidação da atuação no mercado onde a Klok se encaixaria, segue um exemplo:

A empresa A está interessada em disponibilizar novos produtos aos seus clientes, entretanto a empresa A não possui a infraestrutura, ou não deseja gerenciar toda essa operação, e então necessita terceirizar essa atividade para uma empresa B que disponibiliza esses serviços. Ainda assim é necessário o esforço da administração dos ativos, o que demandará esforço e custos adicionais à empresa A além do contrato com a empresa B e com os seus clientes.

Para fazer essa orquestração dessas operações, a Klok atua como uma fachada entre a empresa A com a empresa fornecedora dos produtos financeiros, a empresa B. De modo a oferecer a infraestrutura, manutenção operacional e o suporte técnico à empresa A, assim como uma comunicação uniforme, seja tecnologicamente como também mercadologicamente com a empresa B.

3.4 DIVISÕES DA EMPRESA, OS SEUS PROCESSOS E O AMBIENTE.

A Klok adota uma metodologia ágil baseada no Scrum, Kanban e Lean, com o intuito de evitar desperdícios, evolução dos processos, acompanhamento dos fluxos e o foco em resultados. Para isso é feito uso de ferramentas que auxiliam o gerenciamento das atividades, em principal o Jira, onde as tarefas estão dispostas num quadro Kanban e essas são determinadas e distribuídas pelos líderes do projeto, a *Product Owner* e o *Project Manager*, intitulados PO e PM respectivamente. O papel desses líderes é em principal realizar a comunicação com os clientes e provedores de serviços e definir as prioridades das tarefas de acordo com as necessidades de cada cliente, assim moldando o fluxo de desenvolvimento.

Cada tarefa passa pelas etapas abaixo durante o processo de desenvolvimento de software:

1. Criação, que pode ser originada de avaliação interna do PO, PM, desenvolvedores ou do time de qualidade.
2. Classificação e qualificação, que define a prioridade, urgência e/ou importância.
3. Seleção e caracterização, que determina se a tarefa estará na atual sprint e qual time e/ou subdivisão será responsável para tal.

4. Desenvolvimento, que é a escolha do desenvolvedor da tarefa, e o processo de desenvolvimento e teste unitário.
5. Review (ou revisão), que define o desenvolvedor responsável a analisar e evidenciar possíveis problemas ao desenvolvedor.
6. Testes, que determina um QA para testar a tarefa e definir se está de acordo com a qualidade esperada.
7. Homologação, no qual é realizada a comprovação de realização dos requisitos.
8. GMUD (ou Gestão de Mudanças), no qual é realizada a efetivação das mudanças desenvolvidas em um ambiente de produção.

A Klok Tech é dividida em *squads* que atuam em diferentes visões de um mesmo produto. O discente atuou em principal na *squad* de *produto* durante o curso do estágio, um dos times de desenvolvimento responsável pela evolução da plataforma e que é conduzido pela PO, também atuando na *squad* de *onboarding* que é responsável pela configuração e manutenção de produtos financeiros de clientes esse que é gerido pelo PM.

Outras divisões são a equipe de arquitetura, que atua em quesitos de arquitetura e infraestrutura da plataforma, como também são responsáveis pelo armazenamento de informações sigilosas ou confidenciais dos projetos, e a equipe de Qualidade de Software, responsáveis pelos testes das tarefas, validação de migrações de dados, e sustentação/ratificação de implantações.

Vendo pelo ponto de vista ambiente de trabalho, o estágio é uma ou a maneira de aplicar os conhecimentos adquiridos sobre processos de software e relação interpessoal num ambiente profissional, que no âmbito acadêmico são um tanto teóricos.

A filosofia e cultura interna da Klok Tech, e inclusive um dos motivos de sua função, é promover um ambiente saudável para seus integrantes, de modo a corrigir os erros graves e frequentes em outras e até grandes empresas do mercado da tecnologia, de modo a integrar os integrantes aos processos e no aperfeiçoamento desses processos, a comunicação descomplicada entre os membros, e também a melhoria contínua da equipe, de modo a quando um erro acontecer não será procurado o “culpado”, mas sim decompor a culpa sobre todas as partes, dado que isso aconteceu mesmo após todo o extenso processo de implantação. Para o

ambiente da Klok Tech a equipe é bastante prestativa, dada a característica de melhoria contínua da equipe, cada integrante pode solicitar auxílio em uma tarefa para qualquer integrante das equipes citadas anteriormente e obter assistência independente do cargo. Um membro da equipe se dispõe a dar apoio caso acredite que outro membro esteja precisando, havendo a mentalidade de ajuda recíproca para alcançar um objetivo mútuo.

3.5 TAREFAS EXERCIDAS.

As atividades presentes neste documento foram desenvolvidas pelo discente ao longo do período de 6 meses e 15 dias que atuou como estagiário em desenvolvimento de software, trabalhando na implementação de novas funcionalidades, manutenção da plataforma e configurações de produtos.

Durante as primeiras 5 semanas, cerca de 1 mês, o discente foi integrante da *squad* de *onboarding*, um *time* que trata da manutenção dos produtos financeiros gerenciados pela Klok, com o propósito de aprender a regra do negócio, como também se habituar com o ambiente de desenvolvimento/trabalho, esse treinamento foi supervisionado pelo desenvolvedor pleno Fagner Lima a fim de facilitar a evolução do discente, dentre essas atividades estão:

- Elaborar *scripts* para correção de informações na base;
- Ajustes nos leiautes de rotina de arquivo;
- Configuração de produtos financeiros;
- Inclusão de validações de dados.

Durante o curso do treinamento o discente foi apresentado aos processos de entrega preestabelecidos, essas entregas eram feitas através de Merge Requests (MR). Os MRs são criados pelo desenvolvedor de uma tarefa dentro da plataforma do GitLab no qual é feito o *push* de uma *branch* de um dos projetos da Klok associados à essa tarefa, nesse MR é disponibilizado todo o código escrito em comparação ao código do ambiente de produção, o código desenvolvido é então analisado por outro desenvolvedor com a finalidade de relatar falhas na lógica do código e expressar a opinião do desenvolvedor sobre diversas perspectivas do desenvolvimento.

3.5.1 Descrição dos projetos

Com o objetivo de auxiliar no entendimento das atividades desenvolvidas, um resumo breve dos projetos será feito para se tornar uma base para as informações dispostas na sequência.

3.5.1.1 Hub

O Hub é o serviço central que realiza a orquestração de todos os produtos financeiros e suas características, isso sendo feito de maneira genérica para maior adoção de produtos, provedores e clientes à plataforma. Esse serviço abstrai as lógicas de um produto financeiro e as subdivide em módulos.

O módulo de operação é responsável pela lógica e gerenciamento relacionadas a adesão, cancelamento, cobrança, pagamento e estorno, como também a notificação de um contratante através de um email e/ou SMS (Short Message Service), esse segundo que será melhor abordado posteriormente.

O módulo de capitalização é responsável pelo gerenciamento e lógica relacionadas de títulos de capitalização, baldes de sorteio, números da sorte e empresas capitalizadoras de produtos financeiros.

O módulo de coleta é responsável pelo gerenciamento e lógica da coleta de dados de um produto financeiro, definindo os campos, seus tipos, atributos, opções, tratando da validação dos dados recebidos e armazenados e o armazenamento das respostas de um campo de uma adesão de um produto.

O módulo do gerencial é responsável pela gerenciamento e lógica dos produtos financeiros, tratando de informações dos canais de venda dos produtos, sua categoria de serviço, processadora de cobranças, pagamentos e estornos (cliente), documentos atrelados a um produto (termos de contrato, informações adicionais), locais de origem de adesões (origem comercial), provedor do(s) produto(s), regra de negócio aplicada, metadados atrelados ao produto (parâmetros), configurações de integração e definições de rotinas de execução, em principal, rotinas de geração e leitura de arquivos.

O módulo de relatório é responsável por disponibilizar uma interface unificada para a consulta a dados das operações do banco através de um objeto que simula uma DQL, possibilitando a busca, listagem e agrupamento de informações, como por exemplo, o id (identificador), data de cadastro, status entre outros.

O módulo de rotina é responsável pelo histórico e detalhamento de execuções de rotina, armazenando informações sobre arquivos gerados em rotinas, status de sucesso ou erro durante a execução, como também informações sobre confirmação de “recebimento” de arquivos, horário de início e fim.

Cada módulo pode ser acessado através de uma API HTTP complacente aos princípios REST.

3.5.1.2 Provider

O provider é o serviço que abstrai a lógica da integração com os provedores de seguros financeiros através de conexões por APIs HTTP/REST ou SOAP, através da definição de estratégias que manipulam as operações de um seguro financeiro, em principal as operações de adesão, cancelamento e pagamento.

As informações relacionadas a uma operação, como os dados coletados de uma adesão e metadados de um produto financeiro são enviadas pelo *Hub* através de mensageria ao *Provider*, que interpreta as informações recebidas e encaminha solicitação de acordo com o provedor, operação e (opcionalmente) categoria de produto financeiro.

No provider cada integração pode ser configurada definindo as características de autenticação/autorização, interface/informações esperadas (DTO) para cada operação, e essa interface é montada pelas informações da requisição.

Ao final do processamento da operação pelo *Provider*, é enviado por mensageria uma confirmação de processamento da operação ao *Hub* que define se essa operação foi feita com êxito, se houve erros ou falhas.

O *provider* é responsável pela lógica de integrações feitas via conexões em APIs, porém há provedores que trabalham com o processamento de arquivos com as informações relacionadas às operações, nesses casos, a integração é feita diretamente no *Hub* com a configuração de rotinas de geração de arquivos.

3.5.1.3 Customer

O customer é o serviço que abstrai a lógica da integração com as processadoras de métodos de pagamento através de conexões por APIs HTTP/REST, através da definição de estratégias que manipulam as operações de um seguro financeiro, em principal as operações de cobrança e estorno.

As informações relacionadas a uma operação, como os dados coletados de uma adesão e metadados de um produto financeiro são enviadas pelo *Hub* através de mensageria ao *Customer* que interpreta as informações recebidas e encaminha solicitação de acordo com o cliente (processadora), operação e (opcionalmente) a modalidade de pagamento (A vista ou recorrente).

Assim como no provider, no customer cada integração pode ser configurada definindo as características de autenticação/autorização, interface/informações esperadas (DTO) para cada operação, e essa interface é montada pelas informações da requisição.

Ao final do processamento da operação pelo *Customer*, é enviado por mensageria uma confirmação de processamento da operação ao *Hub* que define se a operação foi feita com êxito, se houve erros, ou falhas.

3.5.1.4 Kear Web

O Kear Web é a aplicação WEB para o gerenciamento de operações através de uma interface gráfica (GUI), sem a necessidade de requisições manuais a API do Hub, como também disponibilizar uma interface em comum para interação com a plataforma para os clientes e/ou provedores.

Nele é possível consultar informações relacionadas a cada módulo, como também realizar o cadastramento, edição e ativação/inativação de componentes.

3.5.1.5 Serviços secundários

Outros projetos são utilizados para auxiliar na divisão da lógica de negócio.

O *Authenticator Service* encapsula a lógica de autenticação de usuários, nele é possível o cadastro e edição de usuários, login e autenticação por token.

O *Email Service* é responsável pela validação e envio de email.

O *SMS Service* é responsável pela validação e envio de email.

O *SFTP Client Service* é responsável pela configuração e sincronização de Secure File Transfer Protocol para integração via arquivo.

3.5.2 Processos de desenvolvimento

No âmbito do desenvolvimento de software corporativo profissional se faz necessário a aplicação de metodologias para o gerenciamento de tarefas e

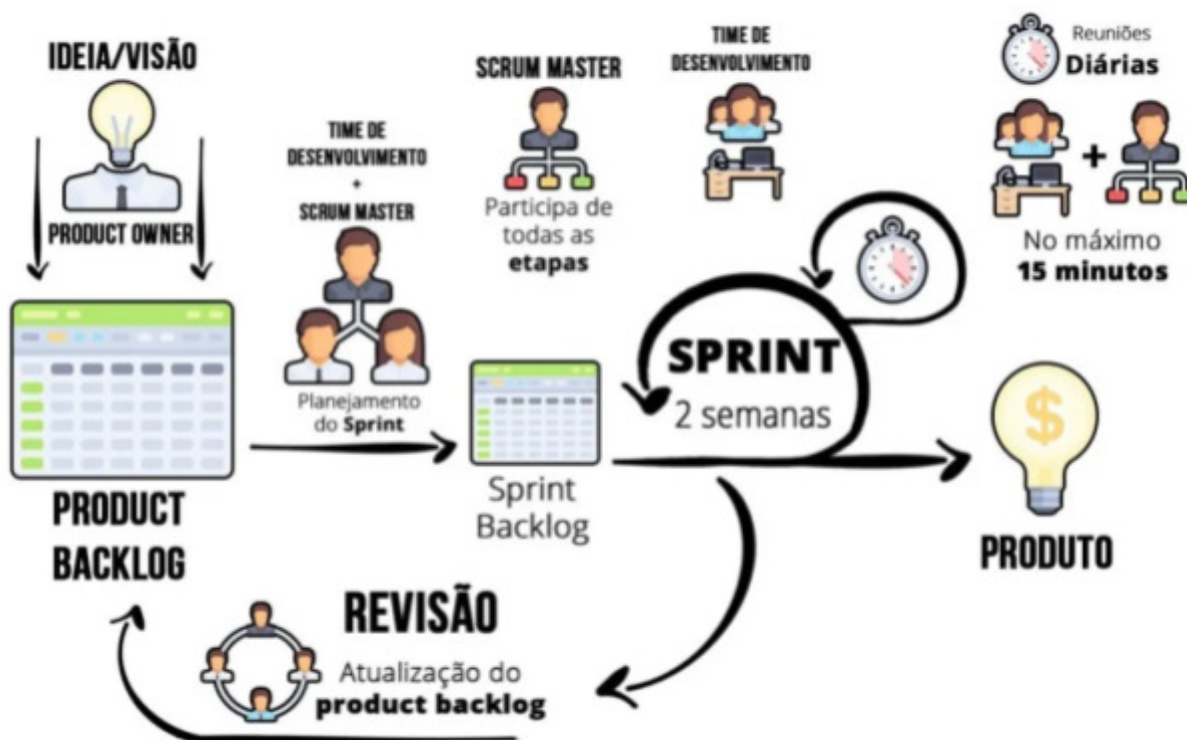
planejamento de recursos. Para tal a empresa utiliza o híbrido de três metodologias ágeis para realizar essa administração, esses sendo, o Scrum e o Kanban.

3.5.2.1 Scrum

“A metodologia Scrum é um framework para trabalhar com projetos complexos, ele faz parte das metodologias ágeis e é utilizado em principal para o desenvolvimento de software, e permite potencializar o trabalho em equipe, acompanhar a evolução do produto, sempre com foco na qualidade da produção e nos prazos estipulados” (ANDRADE, M. 2023). O Sprint Scrum é um período de tempo fixo, no qual uma versão incremental e usável de um produto é desenvolvida e esse período é utilizado como *timebox* das tarefas que estão no Sprint Backlog.

Ele foi implementado na empresa utilizando um ciclo/sprint de duas semanas, ou seja, 10 dias úteis como pode ser visto na Figura 2.

Figura 2 - Fluxo do Scrum.



Fonte: RIBEIRO, Andreza (2020)

O discente teve um contato prévio com essa metodologia de desenvolvimento de software durante as disciplinas de Processos de

Desenvolvimento, Análise e Projeto de Sistemas, Desenvolvimento de Aplicações Corporativas, Projeto 1 e Projeto 2, logo a adaptação ao processo foi facilitada, apesar do contexto de maior complexidade que é o desenvolvimento de software profissional.

3.5.2.1.1 Sprint Daily

A daily é uma reunião feita a cada dia durante uma Sprint, nela cada indivíduo aponta a situação atual das suas tarefas, relata impedimentos ou obstáculos para a continuação dessas tarefas, como também expõe o que foi planejado para o dia atual.

O objetivo dessa reunião é apresentar ao time o estado do andamento da Sprint até a entrega, como também alertar ao Product Owner sobre possíveis problemas para tal, e é feita num breve momento pela manhã a fim de evitar ao máximo o desperdício dos integrantes.

3.5.2.1.2 Sprint Planning

A Sprint Planning é o pontapé inicial da Sprint, nela é feito o planejamento e a definição de escopo de desenvolvimento, essa reunião é decomposta em duas outras reuniões nomeadas de Planning e a “Planning Técnica”, durante a Planning é realizado um debate entre os times de desenvolvimento e o time de qualidade de software com o propósito de designar a expectativa de tempo e complexidade em Story Points das tarefas para o Backlog da Sprint, cada squad possui sua planning.

A reunião da “Planning Técnica” é caracterizada pelo debate para a definição das abordagens a serem seguidas no decorrer do desenvolvimento das tarefas do backlog da Sprint, como também a definição de critérios de aceitação e avaliação pelo time de qualidade de software.

3.5.2.1.3 Sprint Retrospective

A Sprint Retrospective é uma reunião para analisar os resultados alcançados ao final da Sprint, cada integrante dê o seu feedback em relação a sprint, levantando os pontos positivos encontrados que devem permanecer, pontos negativos encontrados e ideias para combatê-los ou extingui-los, como também sugestões para aprimorar o processo.

3.5.2.1.4 Scrum Master

O papel de Scrum Master na empresa é feito de maneira rotatória, e tem a responsabilidade de facilitar as *dailies* e quaisquer impedimentos encontrados, registrar os *feedbacks* gerados a cada *Daily* ou *Retrospective*, e realizava a condução das cerimônias Scrum, durante duas *sprints* o discente atuou em tal papel.

3.5.2.2 Kanban

De acordo com Pereira (2023), o Kanban é uma metodologia ágil que permite o acompanhamento de tarefas e identificação/detecção de gargalos no fluxo de trabalho . O Kanban se baseia na filosofia Lean que busca evitar desperdícios, reduzir custos e aumentar a produtividade dentro de uma empresa, através da melhoria contínua e do adiantamento de demanda (TOTVS, 2023).

O Kanban permite a visualização do fluxo de desenvolvimento representando visualmente cards/cartões as tarefas a serem realizadas e suas descrições, essas tarefas se movimentam entre colunas que definem a etapa ou estado da tarefa, desde a sua criação até a entrega, isso que caracteriza o progresso dessa tarefa dentro do fluxo de trabalho.

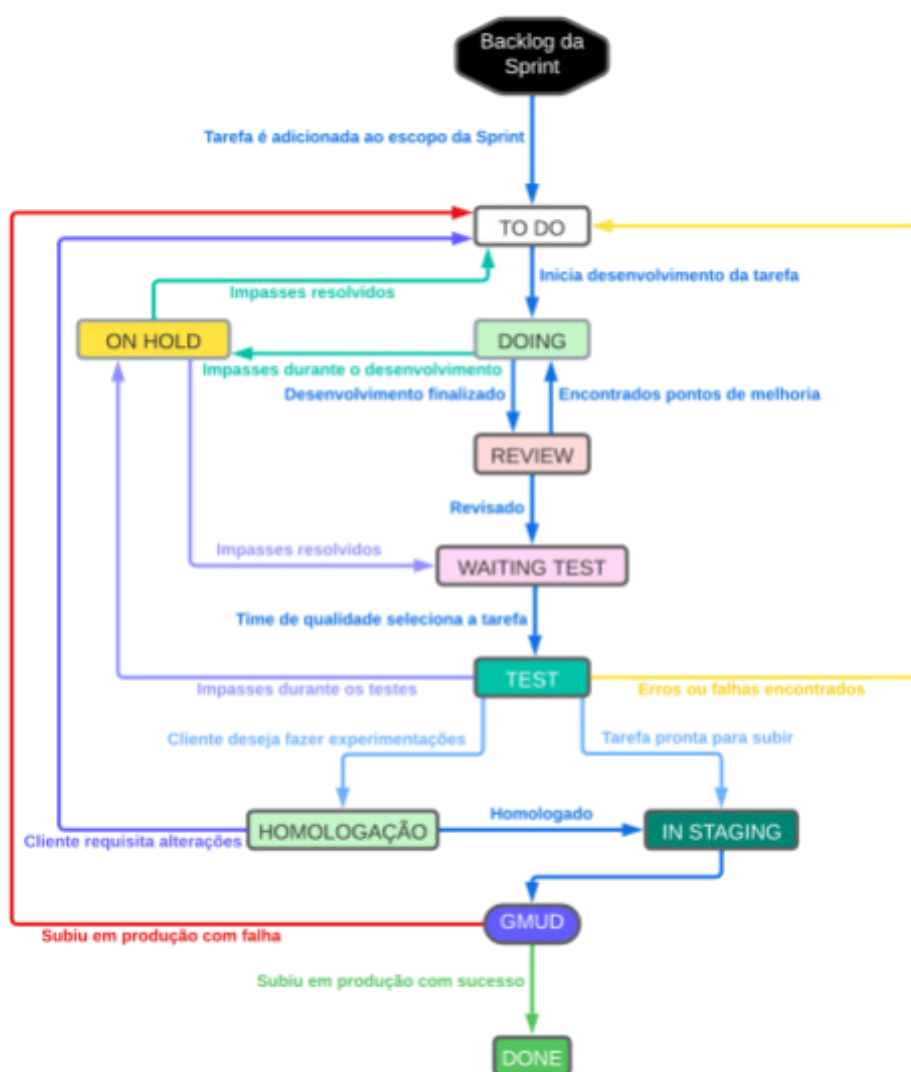
As etapas definidas para cada tarefa são:

- TO DO (A fazer) representa tarefas da sprint que ainda não foram tratadas.
- DOING (Fazendo) representam tarefas que estão em desenvolvimento.
- REVIEW (Revisão) representa tarefas que foram desenvolvidas e estão prontas para análise da solução.
- WAITING TEST (Aguardando testes) representa as tarefas que foram revisadas mas que não foram testadas ainda.
- TEST (Em teste) representa as tarefas que estão sendo testadas pelo time de qualidade.
- ON HOLD (Em espera) representa as tarefas que tiveram seu prosseguimento impedido por algum motivo interno e/ou externo.
- EM HOMOLOGAÇÃO representa as tarefas que passaram pelo fluxo até os testes e que está sendo feita a homologação com o cliente e/ou provedor para que possa ser validado se está de acordo com o planejado.

- IN STAGING (Pronta) representa as tarefas que estão prontas para subirem no ambiente de produção.
- DONE (Implantada) representa as tarefas que subiram em produção e são contabilizadas para as métricas de resultados da sprint.

Todo o fluxo de desenvolvimento é demonstrado de maneira ampla pelo diagrama na Figura 3 logo abaixo.

Figura 3 - Fluxo do Kanban Klok.



Fonte: Própria do autor (2023)

3.5.2.3 Ferramentas auxiliares

Para realizar o gerenciamento dessas práticas, algumas ferramentas auxiliares foram utilizadas.

3.5.2.3.1 Discord

O Discord é uma plataforma de comunicação online onde indivíduos podem se comunicar através de mensagens de texto e chamadas de áudio e vídeo. Essa plataforma é utilizada para a comunicação entre times e squads da empresa, como também para a realização das cerimônias do Scrum (daily, planning e retro), a fim de facilitar e centralizar a comunicação interna.

3.5.2.3.2 PlanItPoker

O PlanItPoker é uma plataforma de coleta de estimativas Scrum como a estimativa de tempo e a complexidade (Story Point), essa plataforma é utilizada durante a cerimônia da Sprint Planning onde após cada debate o integrante expõe sua opinião em relação a cada estimativa, e o time entra em consenso em relação a qual a estimativa da tarefa.

3.5.2.3.3 Jira

O Jira é a ferramenta utilizada na empresa para gerenciamento e monitoramento de projetos e squads, rastreamento de bugs e comunicação entre times (ATLASSIAN, 2023), e é um software que reúne todas as funcionalidades necessárias para o controle e organização de diferentes tipos de projetos, equipes e ambientes de desenvolvimento (Hörlle, 2022).

Esse gerenciamento é feito através de quadros Kanban, cada quadro personalizado para uma função dentro do fluxo interno de desenvolvimento, a fim de organizar atividades similares e separar as diferentes responsabilidades dentro desse fluxo.

No quadro “Klok Tech” são definidos o Product Backlog e Sprint Backlog da squad de Produto, no qual são desempenhadas as tarefas que envolvem a evolução da plataforma. Como integrações com processadoras de meios de pagamento, com

provedores de serviços financeiros, correções de bugs, aperfeiçoamentos nos serviços, o discente atuou principalmente nesse quadro e squad.

No quadro “Kear” são definidos o Product Backlog e Sprint Backlog da squad de Onboarding, essa squad tem o intuito de apresentar a plataforma para novos estagiários dado que possui um maior teor da lógica de negócio concentrado nele, nesse quadro são desempenhadas as tarefas relativas à manutenção de produtos financeiros. Como a configuração dos produtos financeiros, ajustes na coleta ou no gerencial no ambiente de produção, correções de adesões, cancelamentos, cobranças, estornos e pagamentos, o discente atuou nesse quadro e *squad*.

No quadro “Priorização Klok” são cadastrados erros, *bugs*, ou possíveis melhorias na plataforma encontradas por desenvolvedores, time de qualidade ou gerente de projetos, esse quadro tem o intuito de sistematizar a criação de tarefas, como também a manter o rastreamento dessas ideias ou problemas.

3.5.2.3.4 Git, Gitlab e GitFlow

O Git é um sistema de controle de versão distribuído gratuito e de código aberto usado para rastrear alterações em arquivos e código, como também reverter essas alterações para versões anteriores de uma maneira simples e rápida, e auxilia na colaboração entre pessoas permitindo modificações concorrentes (KRIGER, 2022).

O gerenciamento dos projetos é feito através de repositórios Git que são hospedados e versionados utilizando da plataforma GitLab. Essa plataforma auxilia nos fluxos de revisão de código, *Pipelines CI/CD* (Integração Contínua/Entrega Contínua) e implantação, uma *pipeline* consiste em etapas a serem executadas para a entrega de uma nova versão de um software. E no Gitlab é possível criar *Merge Requests*, que são uma maneira de solicitação de incorporação de código a outra *branch*, como também a visualização das alterações feitas ao código-fonte, isso auxilia o desenvolvedor na disponibilização do código alterado, e a cada nova versão do código de um *Merge Request* é executado a *pipeline CI/CD*, no qual cada etapa da pipeline consiste na avaliação dos critérios mínimos necessários para entrega, como a compilação, execução de testes, empacotamento de projeto e implantação.

Nessa mesma estrutura o revisor de uma tarefa pode realizar comentários O com o objetivo de identificar e corrigir possíveis erros, melhorar a legibilidade e a manutenção do código, e garantir que o código atenda aos requisitos do programa e empresa.

Para a organização do versionamento de código é utilizado o GitFlow que é um fluxo de trabalho de controle de versão que é projetado para projetos grandes e complexos (Pedroso, 2022).

Ele é baseado no conceito de usar diferentes ramificações (*branches*) para diferentes tarefas ou fases do desenvolvimento, essas sendo:

- *master*, esta é a *branch* principal do projeto e contém o código mais estável.
- *release*, esta é a *branch* de implantação e contém o código que será implantado em produção.
- *release/homologacao*, esta é a *branch* de desenvolvimento e contém o código que está sendo homologado pelo cliente.
- *develop*, esta é a *branch* de desenvolvimento e contém o código mais recente que está sendo desenvolvido.
- *feature*, são as *branches* usadas para desenvolver novas funcionalidades.
- *hotfix* e *bugfix*: são as *branches* usadas para corrigir bugs críticos.

De modo a complementar o rastreamento de mudanças e bugs, é definida uma convenção de nomenclatura de ramificações, no qual, o prefixo da ramificação é o tipo de *branch*, *feature* para novas funcionalidades, *hotfix* ou *bugfix* para correção de *bugs*, seguido pelo identificador da tarefa no *Jira*, como por exemplo, “hotfix/KLOK-1234”, um exemplo do Gitflow é apresentado abaixo (Figura 4).

Figura 4 - Fluxo do Gitflow



Fonte: Pedroso, Murillo Godoi (2020)

3.5.3 Arquitetura da plataforma

Os serviços e projetos da empresa seguem o padrão de arquitetura de microsserviços. Uma arquitetura de microsserviços é um modelo de desenvolvimento de software que consiste em uma coleção de serviços independentes que interagem entre si (GONÇALVES, 2020). Cada serviço é responsável por uma única funcionalidade e é implementado como uma unidade autônoma que pode ser desenvolvida, testada e implantada de forma independente.

Os serviços em uma arquitetura de microsserviços são geralmente organizados em torno de domínios de negócios ou funcionalidades. No caso da Klok, como citado anteriormente, o Hub encapsula a regra de negócio da lógica operacional de produtos financeiros, o Provider encapsula a regra de negócio de integrações com provedores de serviços com comunicação via API, o Customer encapsula a regra de negócio de integrações com processadoras de meio de pagamentos com comunicação via API, todos os serviços são desenvolvidos na linguagem Java uma linguagem simples, Orientado a Objetos (OOP), robusta, segura, arquitetura neutra, portátil, interpretado, alto desempenho, múltiplos threads, dinâmico (CURY, 2014), juntamente ao *framework* Spring com Spring Boot, o Spring é um ecossistema de desenvolvimento bastante conhecido de nível empresarial, para facilitar a criação de aplicações Java utilizando diversos módulos independentes que são executados na JVM (ANDRADE, A. 2021), o Spring Boot é uma ferramenta que facilita e agiliza o desenvolvimento de aplicativos da web e de microsserviços com o Spring Framework de recursos como, autoconfiguração, uma abordagem opinativa à configuração do Spring e bibliotecas de terceiros, e a facilidade de execução (LIMA, 2021).

Os serviços auxiliares como o Authenticator Service, Email Service e SMS Service, SFTP Client Service, Storage Service são desenvolvidos na linguagem Javascript com o Node.js.

A comunicação entre serviços é feita primariamente através de mensageria, “a mensageria é um conceito que define que sistemas distribuídos, possam se comunicar por meio de troca de mensagens (evento), sendo estas mensagens “gerenciadas” por um Message Broker (servidor/módulo de mensagens).” (BRITO, 2019) e é utilizado para desacoplar e facilitar a comunicação entre serviços, e evitar o bloqueio do processamento.

O Kear Web é a aplicação para interação com o Hub através de uma interface gráfica, é uma aplicação WEB desenvolvida em Angular 2 com Typescript dada que ele é um framework excelente é usado para construir aplicações web corporativas, especialmente desenvolvimento front-end, devido à estabilidade que oferece (Handa, 2021).

3.5.3.1 Computação na nuvem e AWS

Computação na nuvem é a oferta de serviços de computação sob-demanda através da internet (SAVARESE NETO, 2019), como por exemplo, hospedagem de aplicações, bancos de dados, armazenamento de arquivos, entre outros.

A AWS é a plataforma de computação na nuvem da Amazon, essa que disponibiliza os serviços no qual a Klok utiliza para compor a infraestrutura dos hardware e software.

O Amazon *Simple Queue Service* (SQS) é o *broker* de mensageria no qual é ele é responsável pelo armazenamento, gerenciamento e troca de mensagens entre serviços, cada ambiente de execução possui suas próprias filas de comunicação.

3.5.3.2 Arquitetura serverless e AWS Lambda

A computação sem servidor é uma abordagem de desenvolvimento que permite aos desenvolvedores criar e executar aplicativos sem ter que gerenciar servidores. Em vez disso, os fornecedores de nuvem gerenciam a infraestrutura subjacente e os desenvolvedores podem se concentrar em escrever código e criar aplicativos (SERRA, 2018).

O AWS Lambda é um serviço de computação sem servidor utilizado para criar aplicativos e serviços que respondem a eventos, como solicitações de API, atualizações de banco de dados ou uploads de arquivos.

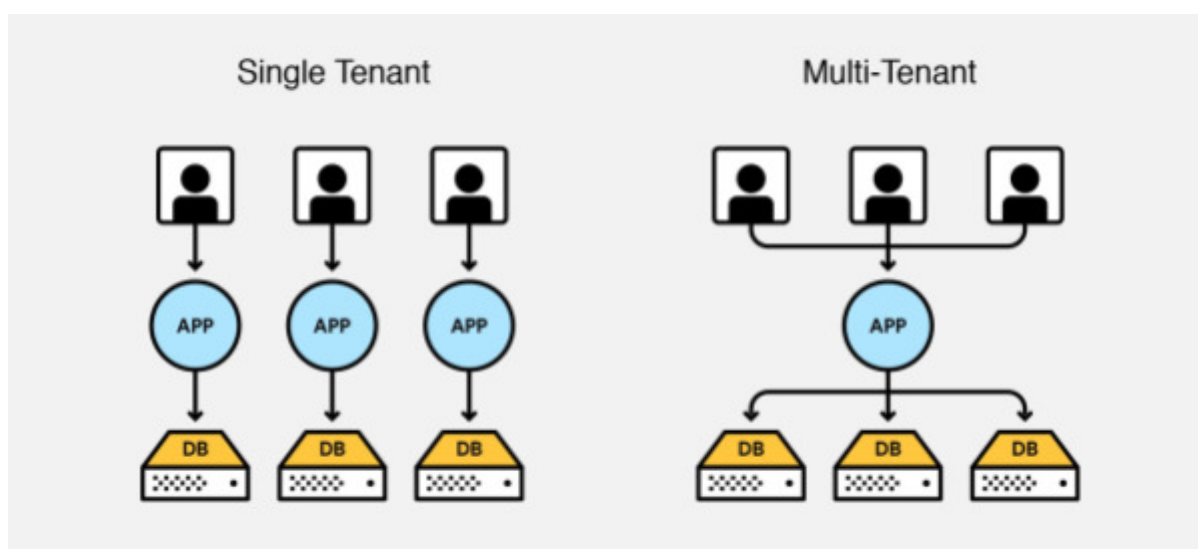
Os serviços auxiliares da plataforma são disponibilizados no AWS Lambda, como o Authenticator Service, Email Service e SMS Service.

3.5.3.3 Arquitetura multitenancy

A arquitetura *multitenancy*, ou multilocação é uma arquitetura em que é usada apenas uma instância de software para atender a vários grupos de usuários diferentes (RED HAT, 2020).

Essa arquitetura colabora para a redução dos custos de desenvolvimento e manutenção de software a longo prazo, dado que apenas uma única instância do software precisa ser criada e mantida, a diferença para a arquitetura padrão de software Single Tenant pode ser observada na Figura 5.

Figura 5 - Arquitetura Multi-tenant



Fonte: KUMAR, Pradeep (2023)

Na empresa isso é implementado em todos os serviços utilizando a infraestrutura da AWS de forma que cada cliente possua um *tenant*, e cada tenant possua sua própria base de dados. Cada tenant possui suas credenciais de acesso, e através delas são feitos os redirecionamentos para o *tenant* em questão.

Como os usuários acessam uma única aplicação e cada *tenant* possui uma base de dados, é utilizado o *Flyway* que é uma ferramenta de migração de banco de dados baseada em código aberto. Ele permite que os desenvolvedores gerenciem a evolução do banco de dados de maneira confiável, automatizada e incremental (DAROS, 2023).

Ele é utilizado para o acompanhamento das alterações feitas no banco de dados ao longo do tempo, e aplica essas alterações de forma consistente em diferentes ambientes (*tenants*).

3.5.4 Configuração de produto

A configuração do produto é a atividade realizada para a disponibilização e venda de produtos financeiros na plataforma da Klok, para tal é necessário a negociação entre a Klok, o cliente, o provedor de tal serviço, e a processadora de pagamentos, para a definição de categoria de produto financeiro, como seguro, assistência ou assinatura, os dados coletados juntamente com as informações que serão armazenadas e também sobre canais de venda, regras de negócio e títulos de capitalização.

Para tal é necessário realizar a configuração do produto que consiste em configurar e anexar a *coleta*, *gerencial* e *capitalização* no *tenant* do cliente.

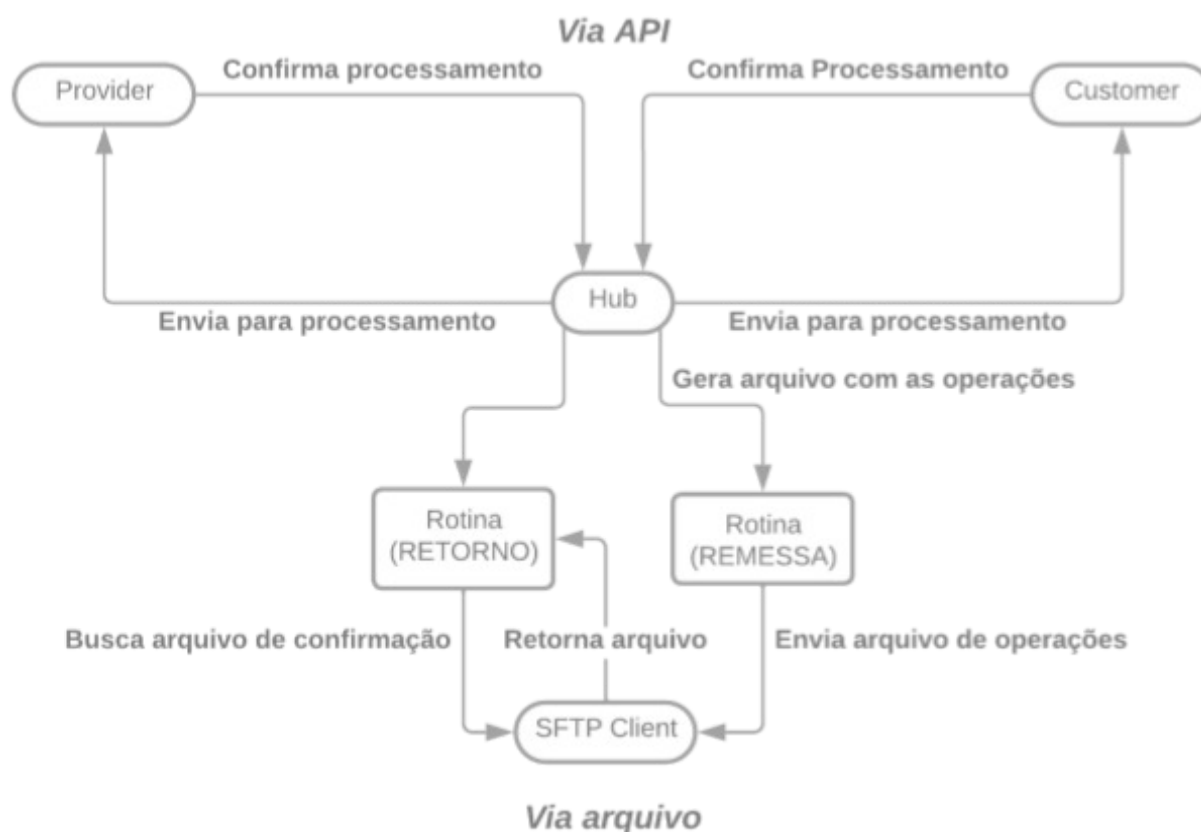
Na coleta são determinados os *campos* necessários para o cadastro de uma adesão, estes campos são utilizados no processamento dessa adesão pelo provedor. Estes campos representam dados referentes a uma adesão, e podem conter informações relacionadas ao contratante, item assegurado pelo produto financeiro, vendedor que comercializou o produto, origem comercial (em geral lojas). Cada campo define um conjunto de *atributos* que modelam os critérios para a aceitação de uma adesão, cada atributo corresponde uma característica do campo, como o tipo de dado esperado no campo (texto, numérico, decimal, data, entre outros), sua obrigatoriedade, tamanho e valor mínimo e máximo, intervalo mínimo e máximo de tempo de datas, entre outros atributos.

Um campo pode também ter opções, que são valores restritos de que o usuário pode inserir nos campos, por exemplo, *Solteiro*, *Casado*, *Separado*, *Divorciado*, *Viúvo*, num campo que representa o estado civil do contratante, ou, *Pai*, *Mãe*, *Irmão/Irmã*, *Filho/Filha*, num campo que representa o parentesco de um contratante dependente.

No gerencial são determinadas as informações relacionadas ao produto que determinam o seu fluxo de funcionamento dentro do Hub e outros serviços, a categoria, provedor, cliente (processadora de pagamento), se o produto é capitalizado, canais habilitados para cadastro, valor do produto, rotinas de processamento, documentos, parâmetros, entre outros. Os parâmetros definem as especificidades que o produto deve possuir, como se possui cadastro de dependentes, quantidade máxima de dependentes, chave da adesão, se dispara email ou sms na adesão ou cancelamento, tipo e momento de integração com

parceiro. No caso da configuração do parâmetro de integração são necessárias para definir o fluxo de cadastro e confirmação de operações entre a processadora, o provedor e a Klok. Operações com integração via API são enviadas para o Customer (Cobrança, Estorno ou Pagamento), ou para o Provider (Adesão, Cancelamento, Pagamento), já integrações via transferência de arquivo são feitas através do rotinas de remessa de operações, sendo configurado o leiaute do arquivo, horário de agendamento de execução, contexto das operações, e intervalo de tempo de cadastro, através da mesma estrutura de rotina é possível definir uma rotina de retorno de processamento pela integração em questão caso necessário, o processamento de operações é demonstrado pela ilustração abaixo (Figura 6).

Figura 6 - Integração entre serviços da plataforma Klok.



Fonte: Própria do autor (2023)

Na capitalização são determinadas as informações relacionadas a título de capitalização e empresas capitalizadoras caso o produto possua capitalização, nele é possível definir qual a série de capitalização e os números capitalizados, como

também é possível definir a quantidade de números capitalizados por operação (Adesão ou Cobrança) através de um parâmetro do produto.

Essas configurações são armazenados no projeto do *setup* em scripts SQL, esses scripts possuem uma convenção de nomenclatura baseada no padrão de nomenclatura do Flyway, com o prefixo “V”, em seguida um número sequencial do *script* dentro do *tenant*, logo depois o tipo operação DML (Data Manipulation Language), DDL (Data Definition Language), ou DQL (Data Query Language), sendo assim, operação de definição, manipulação ou consulta de dados, seguidos de uma descrição rápida do *script*.

O discente exerceu essa atividade esporadicamente durante o percurso do estágio, e é utilizada como “ferramenta” de adequação de estagiários à regra de negócio por tratar de fazer uma introdução de cada projeto/serviço, como também apresentar uma visão geral da arquitetura do software dos mesmos. A adaptação do discente em relação às metodologias, linguagens e tecnologias foi feita de maneira fluida dado que são fundamentos adquiridos durante a graduação.

3.5.5 Code review

O Code Review, ou revisão de código, é uma avaliação sistemática de código produzido seguindo uma série de parâmetros definidos (FELIPE, 2020). Essa prática é realizada com o intuito de aumentar a qualidade do código, melhorar a legibilidade, reduzir a probabilidade de bugs e propagar conhecimento entre os integrantes da equipe.

O Code Review é parte do fluxo de implantação da empresa, e alguns critérios de aceitação do código são:

- Espaçamento
- Indentação
- Nomenclatura
- Complexidade
- Conformidade
- Segurança
- Performance

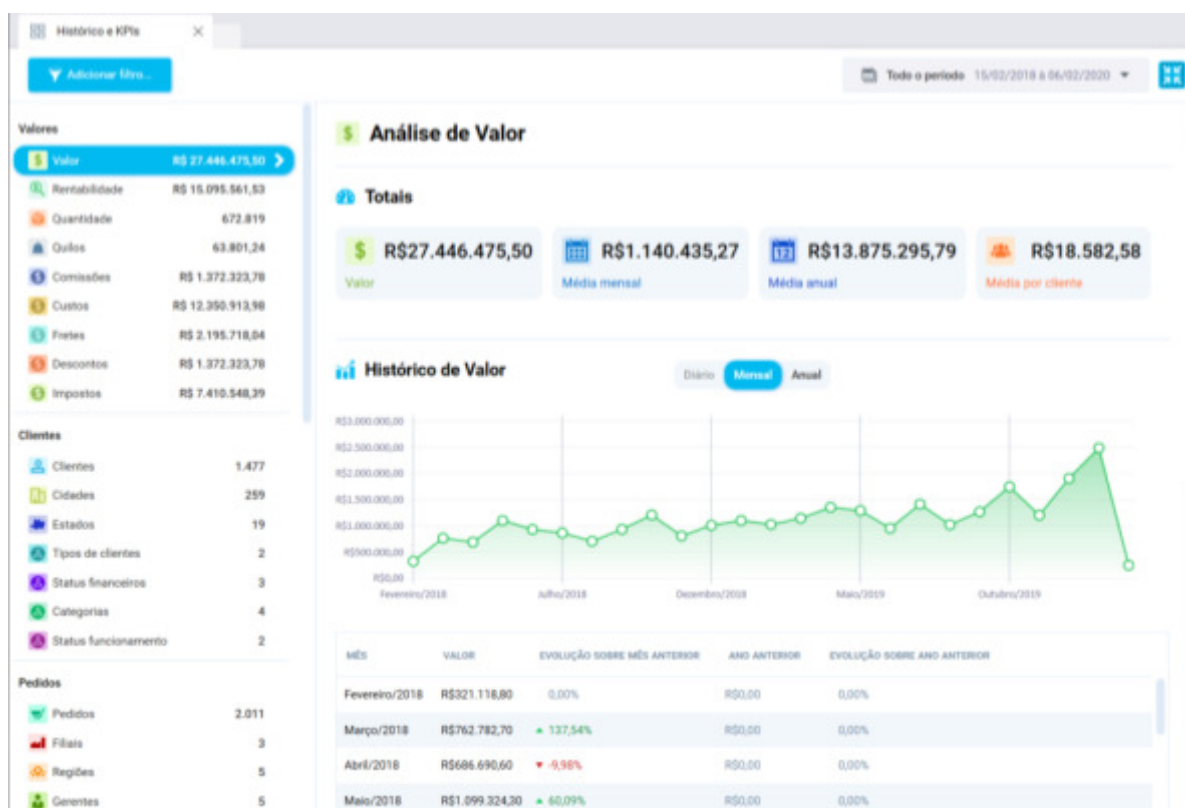
No fluxo de revisão de código, após a finalização do desenvolvimento o responsável passa a tarefa para a raia de REVIEW no Jira, e adiciona o link para o

Merge Request da tarefa no GitLab, ao finalizar a revisão o revisor pode atribuir novas melhorias a serem feitas pelo desenvolvedor e retornar a tarefa em TO DO, ou aprovar esse Merge Request e passar para WAITING TEST.

3.5.6 Kear Web - Dashboard Geral

Dashboard de vendas é um painel de controle que centraliza, organiza e apresenta dados comerciais relevantes. É uma ferramenta visual que fornece um panorama sobre o progresso da empresa em relação ao seu cumprimento de metas e objetivos (SILVA, 2022), Um exemplo de *dashboard* de vendas pode ser observado na imagem ressaltada na Figura 7 e Figura 8.

Figura 7 - Exemplo de dashboard de vendas.



Fonte: SOARES, Bruno (2020)

Essa visão é em especial importante para os clientes Klok por se tratarem em geral de empresas varejistas, como também de produtos que em sua grande maioria possuem mensalidade para efetivação e vigência do serviço.

Para atender essa demanda dos clientes de geração de relatórios, a Klok definiu uma abordagem a ser utilizada no Kear Web para que os clientes interessados possam ter acesso a essas informações sem ter a necessidade de desenvolvimento externo e reduzindo ao máximo a possibilidade de falha humana na geração de relatórios.

Com esse objetivo foi criado um endpoint no Hub para consulta de dados relacionados às operações através de uma interface simulando uma consulta DQL a base de dados, podendo realizar agrupamentos, buscar por valor de campo da coleta, propriedade das operações e/ou produto, uma prévia do produto final apresentada na figura abaixo (Figura 8).

Figura 8 - Dashboard geral de vendas.



Fonte: Própria do autor (2023)

Informações confidenciais filtradas

Para tal o discente ficou encarregado de desenvolver tal funcionalidade na aplicação, a biblioteca utilizada para a criação de gráficos no Angular foi a ApexCharts.

Tal demanda não deveria estar na responsabilidade de um estagiário dada a alta complexidade e exigências de qualidade e tempo, entretanto, foi de solicitação

própria do discente desenvolver uma tarefa de maior complexidade e obteve todo o apoio do supervisor que é o especialista no desenvolvimento front end da empresa.

3.5.7 Envio de SMS e SMS Service

Apesar do cliente ter a possibilidade de notificar os contratantes dos seus produtos financeiros através de correio eletrônico (*e-mail*), nem todo indivíduo possui esse meio de comunicação, uma mídia de maior abrangência é o de telefonia celular, e de acordo com Grandchamp (2022), esse sistema de mensageria curta é um excelente canal de comunicação para o setor varejista, o qual proporciona enormes benefícios para a fidelização e atração dos clientes e consequente o aumento de vendas do negócio.

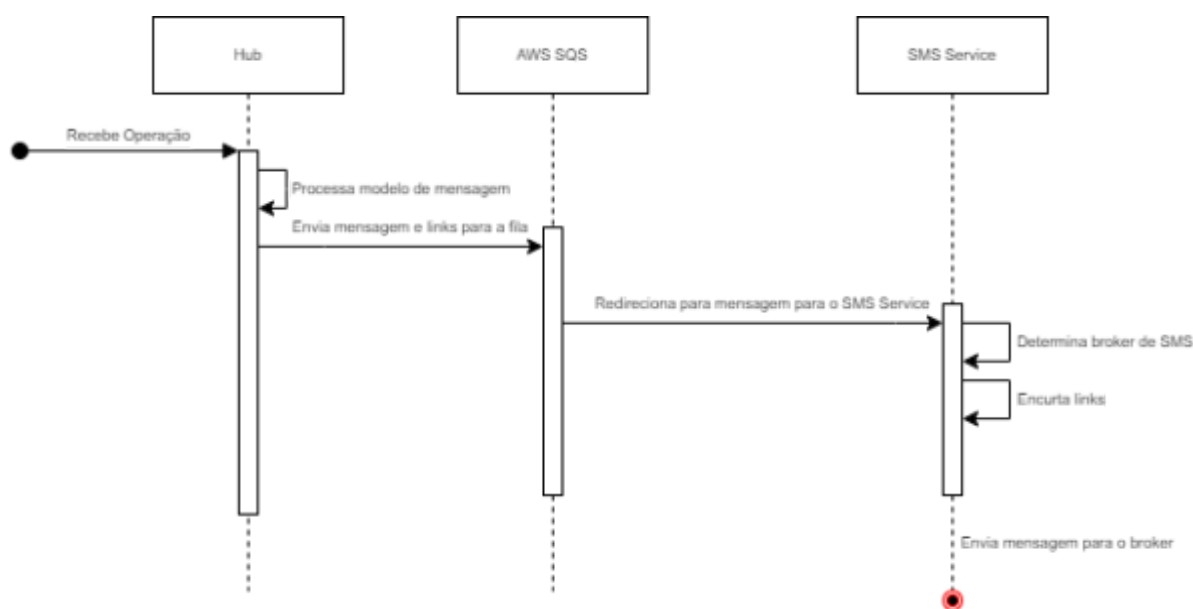
Sendo assim, para atender essa demanda a plataforma Klok também disponibiliza o disparo de SMS, similar ao disparo de email, sua configuração é feita através de um parâmetro que define a operação (adesão ou cancelamento) no qual será enviada a mensagem, o modelo da mensagem, os documentos e os links da web vinculados.

O cliente define e disponibiliza o modelo de texto da mensagem a ser enviado ao contratante, juntamente as demais informações de documentos e links atrelados, e o desenvolvedor realiza o mapeamento dessas informações para a mensagem final através da *Template Engine* Thymeleaf, como exibido no ANEXO C, um template engine é o software que pega um texto e faz a substituição de partes deste texto por alguma informação a ser definida, seja por um código, seja por um padrão que será determinado por alguma regra fixa (MANIERO, 2017). O gatilho para o disparo dessa mensagem é feito ao ser cadastrada essa operação relacionada, o hub processa o modelo de mensagem, e envia para a fila de entrada do serviço de envio de SMS.

O SMS Service é o projeto *serverless* desenvolvido pelo autor dentro da plataforma Klok que é utilizado como o intuito de prover uma interface em comum para envio de SMS entre os serviços da plataforma. Foi desenvolvido em Javascript com Node.js e hospedado na AWS Lambda, seguindo os padrões arquiteturais da empresa de multi-tenant e comunicação por mensageria.

O serviço disponibiliza o envio de mensagens para diferentes *brokers*, e possibilita o encurtamento de urls atreladas as mensagens de SMS, seguindo o fluxo abaixo (Figura 9).

Figura 9 - Diagrama de sequência do SMS Service.



Fonte: Própria do autor (2023)

O discente desenvolveu esse serviço durante o período de estágio, entretanto, a aplicação só veio a ser implantada algumas semanas após a finalização do estágio.

Essa demanda assim como o dashboard de vendas são tarefas que vão além da responsabilidade de um estagiário, o desenvolvimento dessa tarefa acabou causando alguns contratempos do discente nas disciplinas do curso, dada as exigências simultâneas de entrega nas matérias do período e no estágio, o que originou uma situação de *burnout* do discente, toda a situação foi se aliviando de acordo com as entregas eram feitas, e também com o auxílio do time da empresa, em principal do meu supervisor, como também do diretor de tecnologia, esses que faziam o acompanhamento de entregas e o ensinamento de boas práticas.

4 CONCLUSÃO

O discente durante o estágio supervisionado pôs em prática vários dos conteúdos aprendidos no decorrer da graduação, tendo em vista a abrangência dos conteúdos abordados na grade curricular do curso de ADS, que oferece uma base prática e teórica sólida para a iniciação profissional, como também adquiriu novos conhecimentos técnicos e comportamentais ideais para o ambiente profissional como a comunicação profissional, cooperatividade, gestão de tempo.

O estágio teve um efeito positivo significativo na formação pessoal e profissional do discente, apesar das adversidades que aconteceram ao percorrer do estágio que fizeram o discente se impor uma pressão imensa além das responsabilidades de um estagiário, essa experiência ainda assim favoreceu ao crescimento profissional, dado que o discente atuou ativamente no processo de desenvolvimento de *software* em um ambiente corporativo de TI, tendo contato e/ou aprendendo as tecnologias mais utilizadas pelo mercado, como também com as metodologias de desenvolvimento ágeis num âmbito profissional, e atuando diretamente no desenvolvimento de *software* empresarial. O discente obteve experiência prática das estruturas organizacionais e nas relações e comunicações interpessoais dentro do ecossistema da empresa.

O discente a princípio apresentou algumas dificuldades relacionadas às ferramentas e tecnologias utilizadas na empresa, como o *Docker*, *RabbitMQ*, *APIs SOAP* e *AWS*, por isso, deve-se destacar a importância do supervisor e mentor durante o estágio supervisionado Fagner Lima na evolução e progressão do autor acerca desses problemas e no percurso do mesmo dentro da empresa.

Pouco após o final do estágio, o discente foi efetivado como Desenvolvedor Júnior na empresa Klok Tech, trazendo à tona os conceitos organizacionais da empresa de *cooperação mútua* e a *valorização de esforços internos*, o que evidencia o destaque do discente durante o período de estágio, como também revela o ambiente saudável construído pelos integrantes.

Em suma é relevante salientar que conhecimentos abordados durante a graduação como processos de desenvolvimento, algoritmos, SQL, Java e Spring Boot, tiveram grande influência no desenvolvimento das tarefas que o discente foi responsável, como também, a dedicação contínua do discente no aperfeiçoamento

peçoal e profiſſional, ſeja t cnico ou comportamental, foram de maior import ncia para a ſua efetiva o.

REFERÊNCIAS

ANDRADE, Ana Paula de. O que é o Spring?. TreinaWeb, Verão 2021. Disponível em: <https://www.treinaweb.com.br/blog/o-que-e-o-spring>. Acesso em: 1 ago. 2023.

ANDRADE, Marcio Roberto. O que é a metodologia Scrum?. *In: Metodologia Scrum: Metodologia Scrum: O Que é, Métodos Ágeis e Guia Prático*. [S. l.], 7 jul. 2023. Disponível em: <https://blog.contaazul.com/metodologia-scrum>. Acesso em: 31 jul. 2023.

ATLASSIAN. Jira Software. <https://www.atlassian.com/br/software/jira>, 2023. Disponível em: <https://www.atlassian.com/br/software/jira>. Acesso em: 31 jul. 2023.

BRITO, Thiago. #01 – Mensageria. Medium: Thiago Brito, 30 dez. 2019. Disponível em: <https://medium.com/@devbrito91/mensageria-1330c6032049>. Acesso em: 1 ago. 2023.

CURY, Thiago. CARACTERÍSTICAS DA LINGUAGEM JAVA. [S. l.], 12 mar. 2014. Disponível em: <http://www.thiagocury.eti.br/disciplinas/javaavancado/caracteristicas-linguagem-java.php>. Acesso em: 31 jul. 2023.

DAROS, Geraldo. Flyway: o que é e começando a usar com Spring Boot. Blog Cod3r, 13 jul. 2023. Disponível em: <https://blog.cod3r.com.br/flyway-o-que-e-e-comecando-a-usar-com-spring-boot>. Acesso em: 1 ago. 2023.

DOCK TECH. Site Oficial - Dock Tech. [S. l.], 2023. Disponível em: <https://dock.tech>. Acesso em: 19 ago. 2023.

FELIPE, Luis. Code Review – Parte 1: O que é, quais seus objetivos e obstáculos. [S. l.], 30 dez. 2020. Disponível em: <https://www.luisdev.com.br/2020/12/30/code-review-parte-1-o-que-e-quais-seus-objetivos-e-obstaculos>. Acesso em: 1 ago. 2023.

FIGUEIREDO, Jennifer. Quais os principais tipos de seguro?. *In: FIGUEIREDO, Jennifer. Tipos de Seguros: entenda a diferença de cada um e o melhor para você*. [S. l.], 6 jul. 2023. Disponível em: <https://www.mobills.com.br/blog/seguros/tipos-de-seguros>. Acesso em: 31 jul. 2023.

GODOI PEDROSO, Murillo. Git Flow: entenda o que é, como e quando utilizar. Alura: Alura, 7 jun. 2022. Disponível em:

<https://www.alura.com.br/artigos/git-flow-o-que-e-como-quando-utilizar>. Acesso em: 31 jul. 2023.

GONÇALVES, Marcelo. Arquitetura de Microsserviços. Medium: Marcelo M. Gonçalves, 1 jun. 2020. Disponível em: <https://medium.com/@marcelomg21/arquitetura-de-microsserviços-bc38d03fbf64>. Acesso em: 31 jul. 2023.

GRANDCHAMP, Leonardo. Entenda a importância do SMS para o varejo. Jornal Contábil, 7 fev. 2022. Disponível em: <https://www.jornalcontabil.com.br/entenda-a-importancia-do-sms-para-o-varejo>. Acesso em: 1 ago. 2023.

HANDA, Udit. 7 razões para usar Angular para seus aplicativos da web em 2022. Cynoteck, 27 out. 2021. Disponível em: <https://cynoteck.com/pt/blog-post/reasons-to-use-angular-for-your-web-app>. Acesso em: 1 ago. 2023.

HÖRLLE, Wagner. Jira Software: tudo o que você precisa saber. CEP Tech: CEP Tech, 12 jan. 2022. Disponível em: <https://blog.csptecnologia.com/jira-software>. Acesso em: 31 jul. 2023.

JEHNIFFER, Jaíne. O que são serviços financeiros?. *In*: Serviços financeiros: o que são e tipos de produtos utilizados. ISardinha, 20 nov. 2021. Disponível em: <https://investidorsardinha.r7.com/aprender/servicos-financeiros/>. Acesso em: 31 jul. 2023.

KLOK TECH. Site Oficial - Klok Tech. [S. l.], 2023. Disponível em: <https://www.klok.tech>. Acesso em: 19 ago. 2023.

KRIGER, Brunno. O que é git: conceitos, principais comandos e quais as vantagens?. Kenzie: Kenzie, 18 out. 2022. Disponível em: <https://kenzie.com.br/blog/o-que-e-git>. Acesso em: 31 jul. 2023.

KUMAR, Pradeep. Multi-tenancy vs Single-tenancy software architecture. [S. l.], 1 fev. 2023. Disponível em: <https://dev.to/themodernpk/multi-tenancy-vs-single-tenancy-software-architecture-1nhk>. Acesso em: 19 ago. 2023.

LIMA, Cleyson. O que é o Spring Boot?. TreinaWeb, Verão 2021. Disponível em: <https://www.treinaweb.com.br/blog/o-que-e-o-spring-boot>. Acesso em: 1 ago. 2023.

MANIERO, Antonio. O que é motor de template?. Stack Overflow, 17 jul. 2017. Disponível em: <https://pt.stackoverflow.com/questions/220996/o-que-é-motor-de-template>. Acesso em: 20 ago. 2023.

PEREIRA, Mateus. Método kanban: o que é, como funciona e modelos de quadros. [S. l.]: Blog Runrun.it, 27 fev. 2023. Disponível em: <https://blog.runrun.it/kanban>. Acesso em: 31 jul. 2023.

RED HAT. Arquitetura multitenancy. [S. l.], 23 abr. 2020. Disponível em: <https://www.redhat.com/pt-br/topics/cloud-computing/what-is-multitenancy>. Acesso em: 1 ago. 2023.

RIBEIRO, Andreza. SCRUM: entenda a metodologia utilizada na Gestão de Projetos. [S. l.]: Engenharia360, 24 nov. 2020. Disponível em: <https://engenharia360.com/scrum-a-metodologia-utilizada-na-gestao-de-projetos>. Acesso em: 31 jul. 2023.

RPE. Site Oficial - Retail Payment Ecosystem. [S. l.], 2023. Disponível em: <https://www.rpe.tech>. Acesso em: 19 ago. 2023.

SAAB, W. G. L.; GIMENEZ, L. C. P. A segmentação do comércio varejista. 12. ed. Rio de Janeiro: Banco Nacional de Desenvolvimento Econômico e Social, Março 2000. 6 p. DOI <http://web.bndes.gov.br/bib/jspui/handle/1408/16042>. Disponível em: <http://web.bndes.gov.br/bib/jspui/handle/1408/16042>. Acesso em: 31 jul. 2023.

SAVARESE NETO, Eduardo. Computação em Nuvem: O que é, Como funciona e Importância. FIA, 11 fev. 2019. Disponível em: <https://fia.com.br/blog/computacao-em-nuvem>. Acesso em: 1 ago. 2023.

SERRA, Gabriel. O que é Arquitetura Serverless?. [S. l.], 6 set. 2018. Disponível em: <https://blog.tecnospeed.com.br/o-que-e-arquitetura-serverless>. Acesso em: 1 ago. 2023.

SILVA, Ludmila. O que é dashboard de vendas?. In: Dashboard de vendas: passo a passo para criar + lista de ferramentas. Plugu, 26 set. 2022. Disponível em: <https://pluga.co/blog/dashboard-de-vendas>. Acesso em: 1 ago. 2023.

TOTVS. Lean: conheça a metodologia, seus 5 princípios e como aplicar com eficácia. Totvs.com: TOTVS, 11 jan. 2023. Disponível em: <https://www.totvs.com/blog/negocios/metodologia-lean>. Acesso em: 31 jul. 2023.

SOARES, Bruno. Dashboard de vendas: Como utilizar. Salescope, 14 fev. 2020.
Disponível em:
<https://blog.salescope.com.br/dicas-de-uso/dashboard-de-vendas-como-utilizar>.
Acesso em: 1 ago. 2023.

ANEXO A - Detalhes da Vaga

Estágio em Desenvolvimento!

A Klok é uma plataforma de alta performance que atende grandes marcas e busca a excelência em tudo que faz.

Atividades

Trabalhar em demandas de desenvolvimento relacionadas à evolução e sustentação das nossas aplicações.

Requisitos

- Java 8+;
- Spring Framework / Spring Boot;
- RESTful;
- SQL;

Enviar currículo p/ stefanie.bentzville@finbooster.com.br assunto: Estágio Desenvolvimento

KLOK
TECH



ANEXO B - Detalhes do Desafio

README.md

Klok - Desafio Spring Boot

REQUISITOS:

- Implementar uma api capaz de gerenciar adesões (cadastrar; atualizar; e cancelar;) e agendar cobranças;
- Implementar uma api que recebe cobranças do serviço acima, via fila (RabbitMQ) e que recebe pagamentos das cobranças via HTTP/REST.
- Os serviços e bancos deverão subir através de containers Docker;
- Ambas apis deverão ser privadas, sendo necessário informar um token JWT para acessá-las;
- Testes unitários;

TECNOLOGIAS:

- Java 8+
- Spring Boot
- Spring Security
- Spring Data
- Spring MVC
- JWT
- RabbitMQ
- SQL Migration com Flyway
- ApiDoc com Swagger
- Postgres

CRITÉRIOS AVALIADOS:

- Funcionamento;
- Consistência do modelo / aderência ao mundo real;
- Qualidade do código;
- Inovação;
- Testes;

Documento Digitalizado Ostensivo (Público)

RELATO DE EXPERIÊNCIA PROFISSIONAL NA ÁREA DE DESENVOLVIMENTO DE SOFTWARE NA EMPRESA KLOK TECH

Assunto:	RELATO DE EXPERIÊNCIA PROFISSIONAL NA ÁREA DE DESENVOLVIMENTO DE SOFTWARE NA EMPRESA KLOK TECH
Assinado por:	Filipe Farias
Tipo do Documento:	Dissertação
Situação:	Finalizado
Nível de Acesso:	Ostensivo (Público)
Tipo do Conferência:	Cópia Simples

Documento assinado eletronicamente por:

- **Filipe Farias Chagas, ALUNO (201825020003) DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - MONTEIRO**, em 01/11/2023 14:11:58.

Este documento foi armazenado no SUAP em 01/11/2023. 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: 984971

Código de Autenticação: 196d68c234

