



CURSO DE SISTEMAS DE INFORMAÇÃO

IMPLEMENTAÇÃO DOS PRINCÍPIOS DA CULTURA DE DEVOPS NAS APLICAÇÕES DO PREVIPALMAS - ESTUDO DE CASO

CÉLIO GOMES GOUVÊA

Palmas

2023



CURSO DE SISTEMAS DE INFORMAÇÃO

IMPLEMENTAÇÃO DOS PRINCÍPIOS DA CULTURA DE DEVOPS NAS APLICAÇÕES DO PREVIPALMAS - ESTUDO DE CASO

CÉLIO GOMES GOUVÊA

Trabalho de Conclusão de Curso apresentado ao
Curso de Sistemas de Informação da Fundação
Universidade do Tocantins - UNITINS como parte
dos requisitos para a obtenção do grau de Bacharel
em Sistemas de Informação, sob a orientação do
Profº Especialista Mailson S. de Oliveira

Palmas

2023



CURSO DE SISTEMAS DE INFORMAÇÃO
IMPLEMENTAÇÃO DOS PRINCÍPIOS DA CULTURA DE DEVOPS
NAS APLICAÇÕES DO PREVIPALMAS - ESTUDO DE CASO

CÉLIO GOMES GOUVÊA

Trabalho de Conclusão de Curso apresentado ao
Curso de Sistemas de Informação da Fundação
Universidade do Tocantins - UNITINS como parte
dos requisitos para a obtenção do grau de Bacharel
em Sistemas de Informação, sob a orientação do
Profº Especialista Mailson S. de Oliveira

**Profº Especialista Mailson S. de
Oliveira**
Orientador

**Me. José Itamar Mendes de Sousa
Junior**
Co-orientador

Jocivan Suassone Alves
Convidado 1

Jeferson Moraes da Costa
Convidado 2

Palmas
2023

**Dados Internacionais de Catalogação na Publicação
(CIP) Sistema de Bibliotecas da Universidade Estadual
do Tocantins**

G633I

GOMES GOUVEA, CELIO
IMPLEMENTAÇÃO DOS PRINCÍPIOS DA
CULTURA DE "DEVOPS" NAS APLICAÇÕES DO
PREVIPALMAS:: UM ESTUDO DE CASO . CELIO
GOMES GOUVEA. - Palmas, TO, 2023

Monografia Graduação - Universidade Estadual do
Tocantins – Câmpus Universitário de Palmas - Curso de
Sistemas de Informação, 2023.

Orientador: MAÍLSON DE OLIVEIRA

1. FERRAMENTA DEVOPS. 2. SISTEMA DE
INFORMAÇÃO. 3. TECNOLOGIAS DA
INFORMAÇÃO. 4. INSTITUTO PRVIDÉNCIÁRIO
DO MUNICÍPIO DE PALMAS - TOCANTIS.

CDD 610.7

TODOS OS DIREITOS RESERVADOS – A reprodução total ou parcial, de qualquer forma ou por
qualquer meio deste documento é autorizado desde que citada a fonte. A violação dos direitos do
autor (Lei nº 9.610/98) é crime estabelecido pelo artigo 184 do Código Penal.

Elaborado pelo sistema de geração automática de ficha catalográfica da UNITINS com os
dados fornecidos pelo(a) autor(a).

**ATA DE DEFESA DE TRABALHO DE CONCLUSÃO DE CURSO DO CURSO DE SISTEMAS
DE INFORMAÇÃO DA FUNDAÇÃO UNIVERSIDADE ESTADUAL DO TOCANTINS - UNITINS**

Aos 16 dias do mês de **Junho de 2023**, reuniu-se na Fundação Universidade Estadual do Tocantins, Câmpus Palmas, Bloco B, às **11:00 horas**, sob a Coordenação do Professora **Mailson Santos de Oliveira**, a banca examinadora de Trabalho de Conclusão de Curso em Sistemas de Informação, composta pelos examinadores Professor **Mailson Santos de Oliveira** (Orientador), Professor **José Itamar M. de Souza Júnior** (Coorientador), Professor **Jocivan Suassone Alves** e Professor **Jeferson Morais da Costa**, para avaliação da defesa do trabalho intitulado **"Implementação dos Princípios da Cultura DevOps nas Aplicações do PREVIPALMAS - Estudo de Caso"** do acadêmico **Célio Gomes Gouvêa** como requisito para aprovação na disciplina Trabalho de Conclusão de Curso II (TCC II). Após exposição do trabalho realizado pelo acadêmico e arguição pelos Examinadores da banca, em conformidade com o disposto no Regulamento de Trabalho de Conclusão de Curso em Sistemas de Informação, a banca atribuiu a pontuação: 6,5.

Sendo, portanto, o Acadêmico: Aprovado Reprovado

Assinam esta Ata:

Professor Orientador:

Mailson S. Oliveira

Professor Coorientador:

José Itamar M. de Souza Júnior

Examinador:

Jocivan S. Alves

Examinador:

Jeferson M. da Costa

Mailson Santos de Oliveira

Presidente da Banca Examinadora

Coordenação do Curso de Sistemas de Informação

Este trabalho é dedicado aos meus familiares e amigos, pelo apoio incondicional.

Agradecimentos

Dedico esse trabalho aos meus familiares. Aos meus amigos. Por todo apoio incondicional com a minha jornada de formação no ensino superior público.

O desenvolvimento deste trabalho de conclusão de curso contou com a ajuda de diversas pessoas, dentre as quais agradeço: Aos professores orientadores que durante o período da execução deste projeto me acompanharam pontualmente, dando todo o suporte necessário para a elaboração e ajuste do projeto.

Aos professores do curso de Sistemas de Informação da Universidade Estadual do Tocantins - UNITINS pelos seus ensinamentos e permitiram que eu pudesse hoje concluir este trabalho com a excelência de conhecimento que foi repassado. Aos meus familiares que me auxiliaram durante o processo e na minha caminhada até o encerramento desse ciclo com todos os empecilhos que tive e sempre me animando e me motivando. Em especial ao meu filho e minha companheira que são o grande motivo para continuar e proporcionar um futuro digno e com conforto que merecem.

“O discípulo não está acima do seu mestre, mas todo aquele que for bem preparado será como o seu mestre, Lucas 6:40

Resumo

A dinâmica de DevOps, como ferramenta abrange um conjunto de boas práticas em versionamento de código, sendo processo para integração e entrega contínua das aplicações dentro da organização com a colaboração entre a equipe de analistas operantes no sistema. Com o objetivo de entregar soluções de forma ágil, segura e estável. Por tal conceito, este trabalho desenvolveu a implementação dos princípios e da cultura de DevOps e as utilizações das ferramentas propostas para a entrega de um cenário onde as micro-aplicações desenvolvidas junto ao setor de TI do Instituto de Previdência Social de Palmas, estejam. Associado paralelamente ao foco no desenvolvimento dos micro-aplicativos com a utilização dos mecanismos. Além de evidenciar os mecanismos que contribuem para o processo de construção do cenário como Docker, Jenkins, Gitlab. O projeto teve como produto diversas melhorias do serviço de deploy dos micro-aplicativos juntamente com a execução do serviço com automação até a utilização do usuário.

Palavras-chaves: DevOps, Ferramentas, Controle de versionamento, Integração e Entrega.

Abstract

The dynamics of DevOps, as a tool, encompass a set of best practices in code versioning, being a process for integration and continuous delivery of applications within the organization with collaboration between the team of analysts operating in the system. With the aim of delivering solutions in an agile, safe and stable way. For such a concept, this work developed the implementation of the principles and culture of DevOps and the uses of the proposed tools for the delivery of a scenario where the micro-applications developed together with the IT sector of the Instituto de Previdência Social de Palmas, are. Associated in parallel with the focus on the development of micro-applications with the use of mechanisms. In addition to highlighting the mechanisms that contribute to the process of building the scenario such as Docker, Jenkins, Gitlab. The project resulted in several improvements to the micro-application deployment service along with the execution of the service with automation until the user uses it.

Key-words: DevOps, Tools, Version Control, Integration and Delivery.

Lista de ilustrações

Figura 1 – DevOps	22
Figura 2 – Modelos de implementação em cluster	26
Figura 3 – Arquitetura de cluster para balanceamento de carga	28
Figura 4 – Ambiente Proposto de Desenvolvimento	35
Figura 5 – Fluxo de Requisitos	38
Figura 6 – Ciclo Teste de Validação	39
Figura 7 – Modelo de Projeto	42
Figura 8 – Versionamento de Código	43
Figura 9 – Conteiner Server 1	44
Figura 10 – Conteiner Server 2	45
Figura 11 – Conteiner Server 4	45
Figura 12 – Fluxo de Automação	47
Figura 13 – Divisão dos projetos	49
Figura 14 – Gitlab server	49
Figura 15 – Gitlab PREVIPALMAS	50
Figura 16 – Arquitetura do Projeto	51
Figura 17 – Repositorios	52
Figura 18 – Divisão dos trabalhos	52
Figura 19 – Cluster planejado para implementação	54
Figura 20 – Fila de projetos para automação	55
Figura 21 – Exemplo de comandos para automação	55
Figura 22 – Automação em tempo de execução	56
Figura 23 – Automação em tempo de execução	57

Lista de tabelas

Tabela 1 – Tabela de Ganhos	22
Tabela 2 – Sistema de noticias - tabela de noticia	40
Tabela 3 – Sistema de estoque - tabela de produtos	40
Tabela 4 – Sistema de estoque - tabela de pedidos	41

Lista de abreviaturas e siglas

AD - Active Directory

API - Interface de Programação de Aplicações

Back-end - Regras de Negócio e Gerenciamento de conectividade com o Banco de Dados

CI/CD - Integração Contínua e Entrega Contínua

CLI - Interface de Linha de Comando

Debug - processo voltado para identificar e remover qualquer tipo de erro existente no código-fonte de um programa

Deploy - Implantação de software

Front-end - Parte Visível de uma Aplicação

Java - linguagem de programação e um ambiente computacional

JS - Java Script

JVM - Java Virtual Machine

JWT - Json Web Token

Login - Conectar

Logout - Desconectar

POO - Programação Orientada a Objetos

RPPS - Regime Próprio de Previdência Social

SCVD - Sistema de Controle de Versionamento Distribuído

SGBD - Sistema de Gerenciamento de Banco de Dados

SO - Sistema Operacional

Staff - Pessoal

TI - Tecnologia da informação

UI - Interface do Usuário

Views - Visualizações

WAR - Arquivo Binario Web Java

PREVIPALMAS - Instituto de Previdência Social do Município Palmas

Sumário

1	INTRODUÇÃO	16
1.1	Objetivos	18
1.1.1	Objetivo Geral	18
1.1.2	Objetivos Específicos	18
1.2	Justificativa e Motivação	18
2	REFERENCIAL TEÓRICO	20
2.1	DevOps	20
2.1.1	CI/CD	23
2.1.1.1	Integração Contínua	24
2.1.1.2	Entrega Contínua	24
2.1.2	Teste contínuos	25
2.1.3	Cluster	26
2.1.3.1	Alta disponibilidade	27
2.1.3.2	Balanceamento de carga	27
2.2	Ferramentas	27
2.2.1	NodeJS	28
2.2.2	NestJS	29
2.2.3	ReactJS	30
2.2.4	Keycloak	30
2.2.5	PostgreSQL	31
2.2.6	Gitlab	31
2.2.7	Docker	32
2.2.8	Jenkins	32
2.2.9	HAProxy	33
3	METODOLOGIA	34
3.1	Levantamento de Ambiente	34
3.2	Cenário de equipe	36
4	IMPLEMENTAÇÃO DOS MÉTODOS	37
4.1	Etnografia e versões de teste	37
4.2	Codificação	39
4.3	Controle de versionamento	41
4.4	Autenticação e autorização	43
4.5	Microsserviços	44

4.6	Integração e Automação	44
5	RESULTADOS	48
5.1	Portal de Notícias	58
5.2	Sistema de Estoque	58
6	CONCLUSÃO	60
	REFERÊNCIAS	62

1 Introdução

A inserção de componentes e mudança de paradigmas mentais apresentam ganhos efetivos dentro das organizações de mercado globalizado, assim se tornam indispensáveis a utilização de ferramentas cujo a funcionalidade sustentam-se, a partir de um cenário que conte com as inovações tecnológicas e suas utilizações, de maneira que as mesmas possam interagir e se complementar no mercado mutável e competitivo dos sistemas da informação e suas tecnologias, afirma [Baylão e Oliveira \(2015\)](#). Portanto, é imprescindível pensar que:

A mudança, diz respeito a equipe não somente manter o desempenho já alcançado, mas melhorar-se continuamente, aprimorando as competências adquiridas e inovar constantemente, pois o mundo organizacional é dinâmico e as equipes precisam adaptar-se as novas situações que o ambiente lhe proporciona". Entende-se que mudanças são necessárias diante da mentalidade e implantação de mecanismo que estão em constante processos de mudanças para que os hábitos organizacionais possam ser alterados e as estruturas das mesmas tendo ganhos significativos e relevantes. ([SILVA et al., 2011](#), p. 12)

Dessa forma comprehende-se que as mudanças são necessárias diante da mentalidade e implantação de mecanismos que se alteram a todo tempo - processos de mudanças para que os hábitos organizacionais possam ser alterados e assim, as estruturas das mesmas apresentem ganhos significativos e relevantes. Logo, implantar ferramentas que auxiliem no processo de automação dos serviços das organizações, rompe a visão de organização anacrônica e deslancha mais produtividade e eficiência nas entregas de software. Por tal a forma de implantação dos princípios da cultura de DevOps, são diretrizes que nessa transição, potencializam boas práticas dentro das instituições.

Conforme diversos estudos, e um dos mais significantes, no qual descreve ([ARUNDEL; DOMINGUS, 2019](#)) o conceito de DevOps dentro das organizações é mal compreendido. Inicialmente pelo seu significado real, apontado apenas uma trindade em que os sujeitos sociais, inseridos no processo de trabalho e suas culturas tornam uma abstração da tríade, em uma ponta está o processo na outra ponta está a prática, e por segundo, percebido como última ponta, de ferramentas e tecnologias completando o conjunto. Dessa forma este estudo se concentrou em uma implementação desse tipo de ferramentas de automação, em processo de incorporação a estrutura do núcleo de tecnologia do Instituto de Previdência Social de Palmas - PREVIPALMAS, e também será agregado aos padrões e projetos e aos princípios da Cultura de DevOPS, executados, na cena atual deste órgão público, com ênfase em atingir a consolidação das boas práticas da cultura de Devops.

O autor ([MUNIZ et al., 2019](#)) , argumenta, que os pilares da cultura de DevOps, sua colaboração entre os indivíduos da organização torna-se fundamental, possibilitando

a afinidade para relações entre as equipes organizacionais, ferramentas que aceleram e impulsionam as transformações de mentalidade dos colaboradores dentro da companhia, em uma escala, que leva a definição do tamanho das aplicações que estão sendo produzidas.

Estabelecer a capacidade de resposta do serviço de Tecnologia da Informação enquanto automatizam o processo de entrega de software por meio de ferramentas apropriadas, permite mudanças nas áreas de infra estruturas dentro das organizações, isso incorpora um processo produtivo com inovação e agilidade. Acarretando ótimos resultados, isto é, uma eficácia dos instrumentos tornando-os céleres e eficientes na entrega com resposta dos envolvidos e na sua utilização, apontada pelo estudo ([RATO, 2017](#)).

Nesse contexto, com base na revisão de literatura realizada para este estudo, podemos apontar que ([MUNIZ et al., 2019](#)), ([ARUNDEL; DOMINGUS, 2019](#)) e ([SILVA et al., 2011](#)) existem ganhos relevantes que apontam a real mudança, implementações de novas ferramentas que possibilitem as equipes envolvidas, desempenho favorável e gradual benéfico dentro das organizações, de modo que também esses ganhos sejam sentidos pelas equipes em seus processos de trabalho, melhorando a qualidade de vida no ambiente organizacional. Em destaque com base no conceito de DevOps, as atividades em destaque são:

- Cultura
- Automação
- Medição
- Compartilhamento

Garcia ([GARCIA, 2020](#)) explica que as práticas em DevOps desenvolvidas associadamente com sistemas em microsserviços, realizando os testes automatizados, em contínuo de suporte de integração e com o controle de versionamento, acaba entregando sistemas através de funcionalidades mais rapidamente com uma qualidade relativamente maior, apresentando aos profissionais envolvidos uma elevação na execução dos processos a sua eficiência.

Incentivando a equipe por meio de estudos de métodos ágeis para que através de ferramentas a staff do setor comprehenda o processo da autonomia e da usabilidade dos intrumentos adotados, sintam-se preparada para uma implantação efetiva e consciente ficando bem ajustado pelo uso de boas práticas.

Segundo Bajdiuk e Meirelles ([BAJDIUK; MEIRELLES, 2020](#)), o uso completo dos pilares de suporte e o entendimento do papel de DevOps a transformação digital das organizações públicas no setor previdenciário, envolvem uma especificidade da administração

de tecnologia da informação, oferecendo um novo modelo de operação, agregando valor aos projetos.

1.1 Objetivos

Implementação dos princípios da cultura de DevOps no PREVIPALMAS. Inicialmente nas aplicações para as áreas financeira/contábil e comunicação do PREVIPALMAS, que propiciem maior agilidade nos processos de deploy.

1.1.1 Objetivo Geral

Implantar ferramenta de automação de processos de deploy a partir do controle de versionamentos das aplicações desenvolvidas junto ao setor de tecnologia para os serviços do PREVIPALMAS, com ganho de tempo na entrega dos softwares que serão utilizados nas atividades demandadas os servidores do instituto.

1.1.2 Objetivos Específicos

- Implantar sistema de versionamento
- Gerenciar compartilhamento com princípios da cultura de DevOps
- Integrar as ferramentas adotadas
- Aplicar a automação na entrega dos serviços

1.2 Justificativa e Motivação

O Instituto de Previdência Social de Palmas - PREVIPALMAS é um serviço destinado aos servidores do município para a concessão do benefícios que tratem das questões previdenciárias em nível de aposentadoria, sejam pelas plurais motivações, tempo de trabalho, tempo de trabalho e idade, aposentadoria por invalidez, pensões e etc., referente aos servidores e seus dependentes, conforme leis em vigências que regem este serviço público.

Os softwares que são utilizados pelo instituto são contratados para uso exclusivo da demanda de documentação e solicitações dos usuários, assim destacamos que esses softwares, não abrange todas as demandas da organização. O que provoca a introdução das ferramentas e dos princípios de DevOps em serviços essenciais desta instituição que, sobretudo, utilizam de mais tempo e esforço desse processo de trabalho dos servidores aos usuários, somado às muitas atividades de cunho manual.

Atualmente os softwares ofertados aos servidores do instituto em vigência são, a saber:

1. Contratados;

- a) Sistema de gestão de Regime Próprio de Previdência Social - RPPS de segurados e pensionistas;
- b) Sistema de agendamento médico para aposentadoria e ou pensão;
- c) Sistema de emissão de contracheque dos beneficiários;
- d) Portal da transparência.

2. Proprios;

- a) Sistema de monitoramento dos Equipamentos de TI;
- b) Sistemas de suporte PREVIPALMAS;

Todos esses sistemas que estão em total funcionamento dentro da estrutura são indispensáveis para o andamento das atividades. Todavia existe uma demanda reprimida para pequenos sistemas, e sua implantação tornaria mais exitosa a prestação de serviços para os usuários e melhor fator de eficiência na pauta das respostas aos processos (deferimentos) nas solicitações que chegam ao órgão. Essa implementação exige métodos de aplicações em deploy. Assim, quando pensamos em implantar essas ferramentas, pensamos na garantia de soluções propositivas, ágeis e criativas no desempenho dos serviços prestados pelo PREVIPALMAS.

A codificação e a sua devida automação junto aos serviços e rotinas junto ao setor de tecnologia da informação um grau maior de desenvolvimento e segurança nos deploy's e os seus testes das aplicações de forma autônoma e eficaz, com base em poucas esferas. De maneira a agregar conhecimento ao setor público, esse estudo pauta pela proposta de implantar ferramentas de desenvolvimento de software, já na estrutura existente em microsserviços e com os seus gerenciamentos, respeitando todas as suas etapas definidas, sem comprometer o andamento e/ou perda de agilidade na entrega das soluções a serem apresentadas por demanda do PREVIPALMAS.

2 Referencial Teórico

Destaca-se que neste capítulo, com base nas pesquisas bibliográficas realizadas em relação ao tema, far-se-á um relato das bases teóricas de componentes que são utilizados no cerne dos projetos, assim como, seus itens apresentados no mesmo, não pertencer na parte conclusiva, resultado final proposto, que possam corresponder à veracidade, aos componentes para execução de todas as etapas ilustradas e apontadas em síntese nas suas características para a escolha e a adoção da implementação. Quanto aos principais conceitos trataremos, a seguir:

2.1 DevOps

Pode-se dizer que a competitividade nos setores de tecnologia da informação, provoca a uma atualização quase que ininterrupta dos profissionais neste ramo. A saber não se trata de qualquer atualização, mas de uma atualização eficiente, posto que o primado da qualidade da técnica, via as novas tecnologias do mercado, agregam o que as empresas necessitam “produtividade”, além do que maiores são as chances de profissionais com esse perfil serem admitidos. Acrescenta-se que devido ao aumento da robustez das aplicações nas empresas, há uma exigência de profissionais cada vez mais aptos.

Dessa forma, para que as instituições e seus trabalhadores, tenham possibilidade de evolução constante, a partir das novas bases tecnológicas, o principal é começar estudos das metodologias aplicáveis, que tragam melhorias nos processos e que suportem o atendimento das demandas. No caso da metodologia DevOps, as empresas e seus colaboradores, tendem a se conscientizar de que diversas alterações podem se estabelecer com o objetivo de mais ganhos na inclusão da cultura DevOps.

No cenário da metodologia DevOps, tanto as empresas quanto os analistas que a utilizam devem ter conhecimento das alterações positivas que a sua inclusão acarreta. Além do que o investimento em metodologias ágeis, provoca uma maior integração entre as equipes, o item da cooperação é expandido em prol dos resultados. A saber os métodos ágeis, são observados como diretrizes de ponta no mercado de tecnologia da informação, e mais outras empresas passaram a utilizar em seus projetos, isto é, passou a ser uma devolutiva espontânea e mercado. Por todo esse contexto, as empresas se fixam em concentrar, mas valor aos seus usuários, entregando em tempo mais breve os projetos solicitados.

Essa eficácia é real no cotidiano, já que se acelera a produção de metodologias céleres, afirma ([BAJDIUK; MEIRELLES, 2020](#)). Um grande diferencial das metodologias ágeis é encadear o conceito de feedback, a todo tempo, já que a qualidade do software

cresce a cada etapa, de acordo com a comunicação com os stakeholders dos projetos, que são os clientes, analistas e gestores. O que é o avesso da metodologia tradicional, quando o sistema só se tornava aplicável 100% pronto. E nas metodologias ágeis o usuário pode realizar testes e aplicações a qualquer tempo da implantação e citar melhorias de acordo com seus objetivos.

Nesse sentido, destacamos que na perspectiva de aumento dos produtos, serviços, há associadamente uma célebre atualização dos sistemas de tecnologia, respostas práticas e eficientes. Assim a cultura de DevOps apresenta um protagonismo fulcral na metodologia ágil e deve ser implementada seguindo as escalas, potencializando o entrelaçar entre o desenvolvimento e operações de tecnologia da informação, sendo benéfica a automatização de tarefas, a mudanças de ambientes em ambientes escalonados, com progressão positiva dos indicadores de eficiência, assim como e minimização de gastos para as organizações, bem reflete ([BAYLÃO; OLIVEIRA, 2015](#)).

Desta forma a cultura DevOps torna-se um processo importante e que deve ser problematizado constantemente, em relação as melhorias criadas em sua “nuvem”, prática desenvolvida e projetada havendo mudanças nos códigos, soltando em automático para o ambiente de produção, após testes do time DevOps, o que cita Gill ([SATO, 2014](#)). A diretriz de integração ininterrupta, para evitar incompatibilidades e acelerar os testes unitários no sistema, o que leva a uma automatização rápida e permiti descobrir erros, ainda na etapa de implementação.

A lógica seria entender que estes reparos assertivos em tempo breve, evitam problemas no final da implantação. A saber quanto aos testes de aceitação, de acordo com a programação externa, estes são classificados pelo cliente, com o foco de recursos e funcionalidades do sistema, com base na coleta de informações que a equipe usou e testou. Os elementos criados podem ser aplicados e usados na etapa de teste padronizado, analisando as problemáticas e erros em produção, associando tal como os testes de IU (INTERFACE DO USUÁRIO, carga, integração, API (Application Programming Interface), como afirma ([SATO, 2014](#)).

Conforme o estudo a arte feito para esta pesquisa, percebemos que há inúmeras denominações para o conceito de “DevOps”. Aqui destacamos a designação que fazem ([SOUSA; TRIGO; VARAJÃO, 2019](#)). DevOps podem ser metodologias, abordagens, modelos de gestão e movimentos de profissionais da tecnologia da informação. O conceito para DevOps, conforme o estudo bibliográfico realizado, que mais nos chamou atenção, é o que o identifica como um instrumento de abordagem em que a equipe de desenvolvimento tem uma intervenção direta com o staff de operações da rede.

É importante dizer que a equipe de desenvolvimento agrupa pessoas altamente capacitadas na formatação do incremento backlog, modelando os releases no final de cada Sprint. Portanto, essas devem ser treinadas para a gestão em seu próprio trabalho, haja

vista que o Scrum o time de desenvolvimento é autogerenciável e multifuncional, assim sendo cada integrante apresenta a competência e habilidade para atuar no elemento do backlog do produto.

Há uma grande pluralidade de softwares disponíveis, ferramentas de cunho complexo e configuração, alguns com licença livre e outros acessíveis monetariamente. Vejamos a figura abaixo, em que cada ferramenta integra um ou mais processos de implantação e entrega do software.

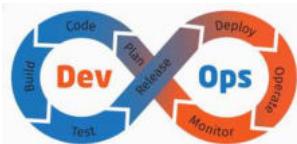


Figura 1 – DevOps
Fonte: ([DEVOPS, 2022](#))

Com base nessas informações, destacamos que todas as ferramentas, precisam ser muito bem articuladas, estudadas, anterior ao seu uso na produção. Conforme especialistas da área, há de se considerar o elemento de configuração. Fator que exige da equipe, conhecimento em relação aos fundamentos que compõem as metodologias ágeis e os elementos utilizados na empresa, refletem Simplilearn Solutions (2018).

Analisados no cerne dos estudos de ([FORSGREM; HUMBLE; KIM, 2018](#)) com adoção de conceitos de DevOps, que a formação nas organizações podem possibilitar o ganho de desempenho monstruoso dentro das equipes distintas quanto possam ter. Isso porque a implantação demonstra que a equipe está madura e organizada com foco no crescimento.

Itens	Tradicional	DevOps
Deploy	1	400x
Recuperação de tempo	1	170x
Falhas e Auteração	1	-80%

Tabela 1 – Tabela de Ganhos
([FORSGREM; HUMBLE; KIM, 2018](#))

Analisamos que esta diretiva de DevOps vem de inúmeros princípios, com controle, processo e aceitação da equipe, onde a staff consiga uma entrada de serviços autônoma, de baixo despedimento após a sua implantação. Mas que esse trabalho inicial acabe futuramente com um único clique de botão ou até mesmo sem. Como trata ([HUMBLE; FARLEY, 2014](#)) os processos de implantação ao serem realizados em scripts para uma automatização, todo o processo de ações a partir do desenvolvimento e técnica de deploy das aplicações, sofrem nenhum desperdício de esforço, mesmo na sua produção inicial.

2.1.1 CI/CD

A Integração Contínua e Entrega Contínua (CI/CD) traduzida do Continuous Integration/Continuous Delivery é uma prática fundamental no desenvolvimento de software, que visa melhorar a eficiência, qualidade e velocidade do processo de entrega de software. Os estudos de ([SATO, 2013](#)) traduzem os benefícios de se manter o processo para padronização de entrega de softwares, o que resulta em um fator de solução inicial e mitiga aspectos das soluções alongadas.

A CI/CD envolve a integração regular e automatizada do código-fonte em um repositório central, juntamente com a execução de testes automatizados e a entrega contínua de versões estáveis do software. Esse processo contínuo permite detectar e corrigir problemas de integração, erros de código e falhas de testes de forma mais rápida e eficiente. Além disso, possibilita que as equipes entreguem novas funcionalidades e atualizações aos usuários com maior agilidade.

Uma das principais características do CI/CD é a automação. Por meio de ferramentas e scripts automatizados, é possível realizar a compilação, testes, empacotamento e implantação do software de maneira consistente e confiável. Em ([SATO, 2013](#)), relata que reduz a dependência de processos manuais e minimiza erros dos profissionais, garantindo maior qualidade e estabilidade do software entregue.

Ressalta-se que a utilização do CI/CD acarreta muitos benefícios para as equipes de desenvolvimento em sistemas de informação. Em primeiro lugar, a detecção precoce de problemas permite uma rápida correção, evitando que erros se acumulem e se tornem mais complexos de resolver. Isso resulta em um software mais estável e seguro, com menos intercorrências e problemas de integração.

Outrossim, a entrega contínua possibilita uma resposta rápida às necessidades dos usuários e do mercado. As atualizações e melhorias podem ser entregues de modo criativo propositivo, permitindo que a equipe esteja mais alinhada com as demandas a serem respondidas de acordo com o perfil de prestação de serviços tecnológicos. Isso melhora a satisfação do cliente e a competitividade da organização.

Em ([MORAIS, 2015](#)), aponta que, neste tipo de abordagem, cria-se o benefício importante em relação a melhoria na colaboração e comunicação entre os membros da equipe. Além disso, a automação das tarefas repetitivas libera os desenvolvedores para se concentrarem em atividades mais criativas e de maior valor. A adoção do CI/CD requer um planejamento adequado e a criação de uma cultura organizacional voltada para a automação e colaboração.

A infraestrutura e as ferramentas necessárias devem ser implementadas e configuradas corretamente, e as equipes devem ser treinadas para trabalhar com os novos processos e práticas. Neste âmbito percebemos que os autores se complementam tanto ([MORAIS,](#)

(2015) e (SATO, 2013) , discutem a Integração Contínua e Entrega Contínua, como prática ímpar no desenvolvimento de software moderno. Ao automatizar o processo de integração, testes e entrega, assim os grupos de trabalho alteram positivamente a qualidade e eficácia de modo veloz da produção do software.

2.1.1.1 Integração Contínua

A Integração Contínua (CI) é uma prática essencial na formulação de software, que abrange a integração regular e automatizada do código-fonte em um repositório central. Essa integração contínua é acompanhada pela execução de testes automatizados, garantindo que o código esteja funcionando corretamente e mantendo a estabilidade do projeto. Segundo Sato (SATO, 2013) , um exemplo é a validação no processo de CI, enquanto execução de testes de unidade. Testes esses que processam a validação da funcionalidade correta de componentes individuais do código, garantindo que cada parte esteja operando conforme sua função. Esses testes são executados automaticamente, sempre que ocorre uma nova integração de código, permitindo a detecção precoce de problemas.

Em (MORAIS, 2015) diante desse tema relata que a integração contínua, também pode envolver outras etapas de validação, como análise estática de código, revisões de código por pares e análise de qualidade do código. Essas práticas adicionais ajudam a garantir que o código esteja bem estruturado, siga boas práticas de programação e seja de fácil manutenção. Nessa pesquisa percebemos que nos arranjos de CI, as validações são operadas automaticamente em associação com transformações no código-fonte, o que é possível ser provocado por commits no repositório central ou por um cronograma definido.

Essa abordagem garante que qualquer problema seja identificado rapidamente, permitindo que os desenvolvedores corrijam as falhas antes que elas se propaguem para o restante do sistema.

2.1.1.2 Entrega Contínua

Em relação ao quesito de entrega contínua das aplicações que demandam os serviços, se faz pertinente a presença de um grupo de profissionais da tecnologia da informação que a equipe, possam garantir os princípios básicos de controle, processo e aceitação, em prol de manter os objetivos. No caso de falha de um desses princípios, o projeto de implantação da cultura de DevOps está fadado ao erro. Quanto aos princípios, destacamos aqui os citados por (HUMBLE; FARLEY, 2014),), em consonância com uma integração contínua, vejamos:

- Controle de versionamento: esse passo determina que toda e qualquer alteração no código da aplicação deve estar sempre atualizada;

- Processo de automatização e compilação: ter sempre scripts de execução dentro de cada projeto, e não depende na IDE's ou outra ferramenta que necessitem de um maior poder computacional desnecessário.
- Aceitação da equipe: os princípios anteriores são mais dependentes deste, já que é a parte onde a equipe fará com que o código com suas alterações necessita da cooperação sem nenhuma quebra de exceções.

Com um comprometimento desses princípios, a entrega da automação dos serviços será realizada com facilidade, tanto de Dev quanto a de Ops. Os processos já criados podem facilitar e agilizar o ciclo das aplicações, acarretando inovações em relação a para a manutenção de códigos e ferramentas para alteração.

2.1.2 Teste contínuos

Entendemos que a definição de teste contínuo passa por aceitação de dedução das validações dos aspectos da aplicação, abrangendo princípios em diferentes etapas para que tal finalidade alcance a maturidade e valor referencial de uma aplicação consolidada para uso. Entretanto, testes contínuos não acabam após as suas validações de aplicação, e sim serão provocados constantemente, sabendo que a cada nova alteração da implementação de novas funcionalidades é necessário repetir o ciclo de testes para sua confirmação.

Critérios podem ser pertinentes conforme ([HUMBLE; FARLEY, 2014](#)), afirma que há critérios pertinentes, que devem interagir com a capacidade, desempenho, disponibilidade, segurança e a usabilidade. A questão dos testes tem associação diretamente ligada a requisitos de sucesso e assim determinando os critérios de aceitação. Importante dizer que a concessão da automação dos processos de construção das aplicações do PREVIPALMAS exige ruptura e regressões dos defeitos que prejudicam as funções preexistentes.

O foco em testes é uma forma de mostrar que a aplicação atende a seus critérios para cada requisito e tenha um benefício adicional. Destacamos que uma parte emblemática do desenvolvimento de um sistema é a refatoração, praticada pela equipe, não alterando o comportamento do programa e sim melhorando a sua organização / estrutura interna, “limpando” o código e reduzindo as chances de erros que ocorram em meio as mudanças efetivadas pelo código.

Esse processo deve funcionar de modo constante conforme a programação do sistema. Encadeando o processo, deve se efetivar os testes unitários no sistema de cada “estória” desenvolvida, que será incluída na versão nova de incremento do software. Posteriormente aos testes unitários, há um feedback instantâneo para os desenvolvedores.

Nesse sistema de testes é importante o conceito de Programação Extrema, uma programação em pares, uma dupla de profissionais em uma estação de trabalho tem o

objetivo de criar um código para uma “estória”, fornecendo diretrizes resolutivas para problemas em tempo real, proporcionando ações ágeis dos desenvolvedores, haja vista que o código é analisado algumas vezes pelo par à medida que o programa é criado, mantendo os desenvolvedores focados no problema.

2.1.3 Cluster

Para a adoção das boas práticas de DevOps, faz-se necessário também pensar como e em que momento as aplicações serão alocadas, tendo em mente a capacidade de armazenamento, memória entre outras máximas. A partir da abordagem do conceito institucional das organizações, apresenta-se soluções que atendem requisitos de ganhos computacionais, visando a utilização dos recursos dentro da mesma. Logo faz-se a construção de um pequeno cluster onde os elos são representados por cada máquina física ou virtual.

De acordo com os estudos de ([FERREIRA; SANTOS; ANTUNES, 2005](#)), os Cluster agregam uma conclusão de benefícios na implantação de serviços web e proxy de cluster de alta disponibilidade, compreendendo que por se usar softwares livres na sua funcionalidade, existe ganho na entrega das aplicações aos usuários finais.

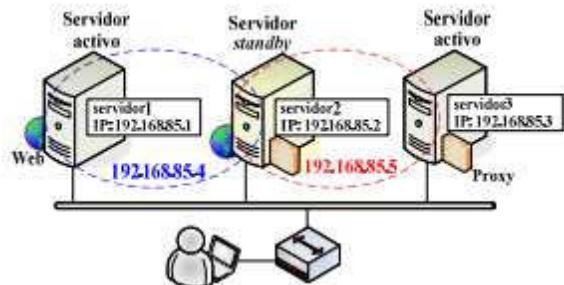


Figura 2 – Modelos de implementação em cluster
Fonte: ([FERREIRA; SANTOS; ANTUNES, 2005](#))

Os estudos de ([FERREIRA; SANTOS; ANTUNES, 2005](#)), na implantação de serviços em cluster, inferem que adquirem ganhos maiores com acréscimos de outras funcionalidades como平衡amento de cargas com o conceito de algoritmo “N+1”. Por “N” identificasse → nó principal, ou seja, o endereço de IP público, quanto a “+1” designa → nós dentro do cluster, representando que os mesmos tenham igual estrutura de configuração para o compartilhamento de responsabilidades. Observando a figura 3, é visível a implementação do serviço de computação no cenário atual, isto é, que usa máquinas de diferentes configurações em favor de atividades em conjunto, portanto a criação de determinada assistência para as aplicações.

Este movimento exige determinado balanceamento de carga, caso necessário, com diferentes pesos para cada nó. Isso implica colocar uma configuração única para o cluster,

mas com uma demanda de serviço diferente para cada ramificação da estrutura. Uma união de dois conceitos do modelo de cluster - alta disponibilidade e balanceamento de carga - torna a vantagem para os projetos em garantia de entrega contínua para os clientes finais das aplicações que estão sendo requisitadas.

2.1.3.1 Alta disponibilidade

No âmbito do contexto computacional, em que os clientes/usuários acabam estando fora dos serviços internos, a viabilização dos serviços tem que ter uma garantia de andamento das práticas, e com segurança, assim o autor ([BARCELAR, 2018](#)), afirma que dentro de um cluster deve haver replicações das aplicações dentro de todos os “nós” é única via de serviços. Onde a garantia de entrega dos serviços aos clientes finais sejam sempre atendidas em conformidade ao planejado.

Agrega a esse tema o autor Costa ([COSTA, 2009](#)) que fala da implementação dentro de ambientes GNU/Linux, ocorre certo favorecimento para a continuidade das soluções e, consequentemente, a sua eficácia na continuidade dos projetos que possam ser desenvolvidos. Onde há vantagens em clusters que optem por seguir esse caminho, com uma garantia de eficiência e de baixo custo com aplicações.

2.1.3.2 Balanceamento de carga

Em um sistema de replicação de aplicações entre os diversos “nós” do cluster, o balanceamento de cargas em alta disponibilidade é um método de conceito, havendo duas características contempladas, a figura 3 caracteriza, quando coloca-se um “nó”平衡ador, e a distribuição de cargas entre todos, é feita.

Assim, o balanceamento se torna diferencial aos projetos que estão sendo desenvolvidos para a alta disponibilidade pensando em diversas requisições simultaneamente. A pesquisa ([NETO; NOGUEIRA; SIVA, 2018](#)) informa plurais pontos positivos na implantação dos métodos de balanceamento de carga → estratégia quando executada. Isso define que o poder de um balanceamento consiga manter as atividades por uma maior disponibilidade, não afetando na entrega daquelas, ocorrendo de maneira natural ao cliente final.

2.2 Ferramentas

A disponibilidade de suporte e documentação que tenham uma comunidade ativa e engajada, que forneça suporte, respostas a dúvidas e atualizações regulares. Além disso, a existência de documentação clara e abrangente facilita o processo de aprendizado e solução de problemas durante o desenvolvimento e as integrações entre as ferramentas.



Figura 3 – Arquitetura de cluster para平衡amento de carga

Fonte: ([BARCELAR, 2018](#))

A eficiência e a produtividade oferecidas por uma ferramenta também são considerações importantes. Os desenvolvedores buscam ferramentas que possam acelerar o desenvolvimento, automatizar tarefas repetitivas e oferecer recursos avançados que facilitem o trabalho.

2.2.1 NodeJS

Na perspectiva de Nodejs, os autores ([SILVA; OLIVEIRA, 2020](#)), relatam que esta ferramenta é justificada em decorrência da linguagem utilizada JavaScript tanto no frontend quanto no backend, o que pode facilitar a criação de aplicações mais coesas e eficientes. Além disso, os autores ([GOMES; SOUZA, 2019](#)) falam que a capacidade do Nodejs de tratar processos assíncronos de modo eficiente em relação a outras tecnologias, tornando-se uma aplicação real, como por exemplo, aplicações de chat e jogos online.

Também Nodejs é possível criar aplicação sendo criado em JS com sintaxe extremamente simplificada de fácil compreensão para desenvolvedores de back-end e front-end que utilizam essa linguagem. A migração de trabalho com esse instrumento acaba tendo um ganho de performance e desempenho, pois a mesma linguagem permanecerá inalterada para desenvolver as aplicações. O Nodejs pela literatura da área do sistema de informação é percebido como uma plataforma de software que faz uso do motor V8 da plataforma Google, que é executado a partir do ECMAScript e WeAssembly.

Esse tipo de tecnologia permite que servidores como o Nodejs possa ser executado em diferentes sistemas operacionais e arquiteturas, passando a ser uma ferramenta segura e independente da plataforma de hospedagem. Ou seja, o Nodejs pode ser utilizado em diferentes ambientes sem a necessidade de alterações significativas, o que o torna uma opção atrativa para desenvolvedores que buscam uma solução flexível e escalável. Com isso, é possível desenvolver aplicações web robustas e escaláveis, além de outras soluções de software que podem se beneficiar dessa tecnologia. De acordo com a pesquisa, uma das

principais características do Nodejs é o seu arranjo baseado em eventos.

A saber, o modelo ([NODEJS, 2019](#)) eventos é uma estrutura amplamente usada para o desenvolvimento de sistemas distribuídos. Outrossim apresenta um loop de eventos – uma construção de tempo de execução, o que projeta um comportamento padronizado através de retornos de chamadas no começo de um script e no final, e o servidor é iniciado, com uma chamada de bloqueio.

Esse tipo de abordagem possibilita inúmeras vantagens para desenvolvedores, tais como a capacidade de lidar com muitas e simultâneas conexões, abrindo mão de um thread para cada conexão, o que eleva a escalabilidade e o desempenho do sistema. Além disso, o padrão de eventos permite que o código seja escrito de modo assíncrono, o que provoca a execução de operações de entrada e saída sem bloquear a thread principal, tornando o sistema mais responsivo e eficiente.

Dessa forma, o motor entrará no loop de eventos após executar o script de entrada e sairá do loop de eventos quando não houver mais retornos de chamada para executar. Esse comportamento é como uma porta giratória que é acionada a saída do usuário.

2.2.2 NestJS

Ademais, com a agilidade de criação da API, os NetJs passam a ser uma ferramenta de suporte robusto, ao lado do servidor com HTTP, fornecendo abstração direta da API, configurando liberdade com infinitos modos de terceiro para o seguir da produtividade. Temos também os ReactJS, que retratam uma biblioteca de encapsulados, para fins de criação de telas de interação com o usuário com linguagem baseada em JS.

O React é uma biblioteca de código que permite criar interfaces de usuário ativas e dinâmicas em aplicações web. É mantido pelas empresas Meta Open Source e por uma comunidade ativa de desenvolvedores individuais, tornando-o acessível e disponível para todos. Com uma sintaxe poderosa, o React é amplamente utilizado em sites populares como o NASA, ATT, IBM, Amazon entre outros, demonstrando sua versatilidade e confiabilidade, conforme ([NESTJS, 2017-2022](#)).

ReactJS faz com que a criação de interfaces interativas com o usuário seja uma tarefa fácil, para cada estado da aplicação, visto que se assemelha a interação humana em busca de informações, o React atualiza seus componentes e busca uma atualização eficiente em componentes específicos, na medida em que os dados mudam. Existe uma flexibilidade de linguagem adicionada a biblioteca ReactJS, favorecendo, esta ser uma poderosa ferramenta de desenvolvimento, de modo que as views declarativas façam com que seu código seja previsível e simples de debug e a sua reescrita em um código.

2.2.3 ReactJS

O ReactJS trata-se de uma biblioteca de encapsulados que é usada para a criação de telas de interação com usuário com linguagem baseado na em JS. O React é uma biblioteca de código aberto que permite criar interfaces de usuário reativas e dinâmicas em aplicações web. É mantido pelas empresas Meta Open Source e por uma comunidade ativa de desenvolvedores individuais, tornando-o acessível e disponível para todos. Com uma sintaxe poderosa, o React é amplamente utilizado em sites populares como o NASA, ATT, IBM, Amazon entre outros, demonstrando sua versatilidade e confiabilidade ([TEAM, 2023](#))

ReactJS faz com que a criação de interfaces interativas com o usuário seja uma tarefa fácil, a criação das views de maneira simples para cada estado da aplicação, visto que semelhante a interação da pessoas com busca de informações, o React atualizar seus componentes e buscar a atualização eficiente em componentes específicos, na medida em que os dados mudam.

Segundo ([REACT, 2019](#)), a flexibilidade da linguagem adicionada a biblioteca ReactJS torna-se uma poderosa ferramenta de desenvolvimento, de modo que as views declarativas façam com que seu código seja previsível e simples de debug e a sua reescrita em um código.

2.2.4 Keycloak

É uma autorização das aplicações de verificação de usuários, sem seus respectivos sistemas assíncrono, que fazem validação, de modo que, não haja necessidade de que cada aplicação tenha que criar seu próprio sistema de verificação. Mantida com uma ferramenta de código aberto pela empresa Red Hat Inc., essa solução é um utilidade para a implementação na prática do estudo de caso, com as funcionalidades sendo todas descomplicadas e uma liberdade de ajuste conforme exigências, que venha precisar em quesito de segurança das aplicações.

Em Keycloak, as verificações não precisam de criação de formulários login e ou de autorizações de sessões, sendo que os usuários navegam em muitas aplicações por meio de um único login, assim como por um só logout.

A propósito Keycloak é um dispositivo, que em sua implantação apresentam-se dois tipos de segurança, assim o administrador de sistemas pode optar por conexão com redes sociais, de servidores LDAP, servidor Active Directory já existentes, ou ainda podendo criar um novo tipo interno de autenticação. O Keycloak trabalha com padrão OpenID Connect ou SAML 2.0. Quando essa solução é implementada de forma centralizada, os microsserviços podem compartilhar informações de autenticação e autorização entre si, sem a necessidade de cada micro serviço implementar sua própria solução de gerenciamento de

acesso.

Isso significa que, quando novos microsserviços são adicionados ao projeto, automaticamente se integram à solução existente de gerenciamento de acesso, sem a exigência de criação da solução do zero. Isso economiza tempo e recursos, tornando o processo de desenvolvimento mais eficiente.

Um exemplo seria uma plataforma de e-commerce com vários microsserviços. O serviço de carrinho de compras, serviço de pagamento, serviço de gerenciamento de produtos, são exemplos. Se cada um desses serviços tivesse própria solução de autenticação e autorização, não seria necessário implementar uma nova solução sempre que um novo serviço fosse adicionado. Mas, se uma solução centralizada for implementada, cada novo serviço pode ser integrado à solução existente sem esforço adicional de desenvolvimento.

2.2.5 PostgreSQL

Setor de segurança de dados das aplicações, passa a ser uma ferramenta de escolha por ter maturidade e segurança com uma extensa documentação. Sendo uma SGBD com mais de 30 anos em operação no mercado, sua confiabilidade, robustez e desempenho são alcançados dentro deste grande dispositivo..

Seus recurso ([POSTGRESQL, 2022](#)) é usado como foco usar recursos que estejam associados aos desenvolvedores que criam os sistemas com segurança no SGBD ALOCADOS AO PADRÃO DE OPERAÇÕES DERIVADAS DO SQL onde os administradores de banco podem proteger a integridade dos dados e a criar ambientes tolerantes a falhas.

2.2.6 Gitlab

O Git é um sistema de controle de versão distribuído, o que significa que cada desenvolvedor possui uma cópia local completa do repositório, incluindo todo o histórico de alterações. Isso permite que os desenvolvedores trabalhem offline e sincronizem suas alterações posteriormente.

Além disso, o Git é uma solução open source, o que significa que é gratuito e pode ser instalado localmente em qualquer sistema operacional baseado em linux. Isso oferece maior controle e segurança sobre o repositório, pois todo o código é armazenado em um servidor local ou na nuvem privada do instituto.

No entanto, apesar de ser uma solução de instalação local, isso não significa que o repositório é público. É possível definir configurações de segurança e permissões de acesso para restringir o acesso apenas a contas privadas dentro do instituto, garantindo a privacidade e segurança dos dados do projeto.

Segundo ([BITBUCKET, 2022](#)) uma das maiores vantagens de usar o padrão

GIT são seus recursos de ramificação. Ao contrário dos sistemas de controle de versão centralizados, as suas ramificações do fluxo de desenvolvimento tornam-se baratas e fáceis de mesclar. Isso facilita a sucessão de trabalho para o desenvolvimento.

Além de possuir uma plataforma na internet, essa ferramenta tem a possibilidade de criação em uma instância própria para organizações que queiram que seus códigos permaneçam privados, sem a necessidade de aquisição licença para mantê-las, dando ainda maior confiabilidade no quesito de segurança aos projetos governamentais de maneira que atenda ao princípio da privacidade de código.

2.2.7 Docker

Exemplificando o Docker ([DOCKER, 2022](#)) a utilização dessa excelente ferramenta pode até eliminar grandiosas e cansativas tarefas de configuração repetitivas de servidores com mudanças, e assim, usando todo o ciclo de vida para um desenvolvimento de aplicações rápidas, fácil e portátil, acima de tudo com versões determinada pelo administrador de sistema tendo o servidor físico ou em nuvem.

Com as facilidades de alguns equipamentos e várias aplicações opostos com linguagens diferentes, ao mesmo tempo em que os recursos computacionais são díspares em uma ferramenta que agiliza o conceito de DevOps dentro das organizações.

O dispositivo inclui Interface do Usuário -UI, Interface de Linha de Comando - CLI, API com todos os sistemas de segurança que são projetados para desenvolvimento com foco no trabalho em conjunto com todo o ciclo de vida até o deploy das aplicações.

2.2.8 Jenkins

Com o conceito do ([JENKINS, 2022](#)), é um dispositivo de automação de serviços que em sua base existem centenas de plug-ins que fornecem para os administradores e gestores de infraestrutura, uma garantia de que o projeto pensado em realizar na automação nos servidores será bem atendido dentro da proposta inicial.

Ainda segundo as funcionalidades ([JENKINS, 2022](#)) essa ferramenta possui características excepcionais, que permitem a entrega contínua das aplicações com simples comandos e com um clicar de um botão dentro de uma plataforma web ou até mesmo a criação de mecanismos integrados ao controle de versionamento, onde pode se configurar basicamente toda a vida de realização até o seu deploy.

Dentro da prática de DevOps, a implementação é de maneira bem simplificada, bastando que o servidor alocado necessite de um JVM e do arquivo WAR do projeto. Isso em caso de opção do administrador dos sistemas, mas o mesmo pode ser implantado com um container docker com uma imagem oficial dele próprio fazendo todas as funcionalidades estejam prontas.

Entretanto, ainda sobre (JENKINS, 2022), esse dispositivo pode fazer trabalhos distribuídos, isso permite que administradores reflitam sobre escalar os serviços em vários nós em diferentes servidores e localidades, podendo ser gerenciados somente por uma única aplicação master e deixando o que em cada ponto de montagem das aplicações tenha sua execução efetiva.

2.2.9 HAProxy

O HAProxy, que atua como ferramenta intermediária entre o acesso das aplicações e os serviços web (workers) funciona como um Gateway de API em gerenciamento de roteamento, balanceamento de carga, com servidor de proteção contra sobrecarga, limitação de taxa e registro de métrica. No entanto, para serem bem sucedidas, as APIs também devem ser seguras, confiáveis e fornecer boa observabilidade para um gateway que possa ser uma solução econômica para adicionar esses recursos a qualquer API moderna.

Verificações ativas e passivas das soluções das organizações, o HAProxy é uma ferramenta que indica como seus servidores de aplicativos estão funcionando, instantaneamente removendo nós ruins do balanceamento de carga. A desvantagem é que não lhe dá uma rica sensação sobre o estado dos servidores, como carga da CPU, espaço livre em disco e seus servidores de API estão ativos. Assim deve se observar a integridade do servidor, no status crítico para saber quantos servidores estão passando as sondagens de verificação de integridade que o HAProxy envia.

O status de envio de ferramenta de balanceamento com a definição de ativo e inativo de cada servidor junto com o resultado de aprovação ou reprovação pela verificação de integridade dos servidores mais recente proporciona que o instrumento as valide. Assim, poderá ser usado como indicador, como métrica para quando mais de 25% dos seus servidores estiverem inativos.

3 Metodologia

Neste tópico apresentamos o modelo de implementação que a pesquisa estabelece como proposta, de acordo com os conhecimentos das ferramentas com propósito de resolver problemas que se apresentam nas demandas da instituição.

Nesse contexto, considerando a demanda central na implementação do portal de notícias e um sistema de estoque como um primeiro impulso para levantar o cenário de crescimento das atividades de desenvolvimento do setor de TI, alocado dentro do PREVIPALMAS. Levando em conta restrições físicas além da restrição de poucos profissionais disponíveis para executar ambos os projetos, é fundamental adotar uma estratégia eficaz para garantir o sucesso das iniciativas. Medida essa que tem a função de agregar os processos mencionados, já que é requerida essa melhora, constatadas nas solicitações para tais demandas, percebendo como uma construção eficiente e dinâmica.

A seleção da metodologia para a aplicação da estruturação da implantação dos princípios da cultura de DevOps nas dependências do PREVIPALMAS, onde o levantamento de ferramentas e modelo de estruturação passa por uma análise bibliográfica dos conceitos e métodos que farão parte deste projeto.

Outrossim, balizar elementos como infraestrutura e levantamentos de equipe para o desenvolvimento de maneira que possam atender todo e qualquer entendimento para a criação da estratégia, passa por uma análise qualitativa das informações com o levantamento de requisitos para o desenvolvimento bem como o levantamento dentro do análise bibliográfica e dos materiais necessários para a sua construção.

Quando interagimos com os recursos humanos, é importante organizar um planejamento tácito e zeloso, identificando as principais etapas e atividades envolvidas em cada projeto. Sendo uma abordagem sugerida juntos ao referencial de embasamento, assim dividir os projetos em fases distintas. Isso permite que a equipe se concentre em uma tarefa de cada vez, minimizando a sobrecarga e otimizando o tempo disponível

3.1 Levantamento de Ambiente

Após analizar e entender o cenário da instituição disponibiliza para o projeto e criada uma estratégia que evidencia uma visão total do ambiente almejado de trabalho conclusos ao final na sua entrega que se espera alcançarm, em que o pesquisador desenvolve / aplica a pesquisa. A figura 4 aponta os elementos essenciais que deveram integrar o contexto do estudo bem como ambiente proposto para o caso, como âmbitos propositivo e criativo.

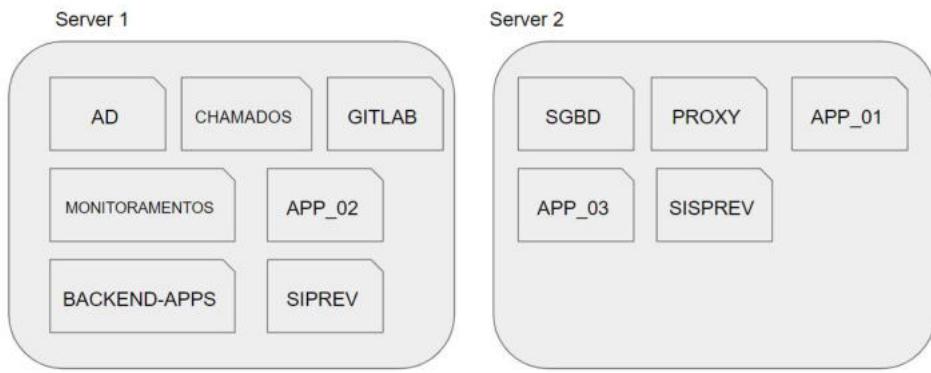


Figura 4 – Ambiente Proposto de Desenvolvimento
Fonte: Imagem do Autor

Referente a figura 4, aponta o cenário a qual está sendo proposto ao projeto, a partir de uma base de planejamento com o escopo inicial das demandas solicitadas a partir de uma infraestrutura para a aplicação dos conceitos e princípios que foram levantados:

- Server 1 - Dell

Processador: Xeon(R) E5-2630 v3 @ 2.40GHz

Memoria: 32GB

Hd: 10TB

Virtualização: VMware ESXi 6.7

1. AD: Instância de Active Directory (AD)
2. Chamados: Instância de sistema de chamados para suporte
3. GITLAB: Instância de sistema de versionamento de código
4. Monitoramentos: Instância de sistema de monitoramento
5. APP2: Instância de aplicações frontend
6. BACKEND-APPS APPS: Instância de aplicações de Backend
7. SIPREV: Instância de sistema previdenciário governamental

- Server 2 - HP

Processador: Xeon(R) E5-2665 0 @ 2.40GHz

Memoria: 32BG

Hd: 2TB

Virtualização: VMware ESXi 6.7

1. SGBD: Instância Sistema de gerenciamento de dados
2. PROXY: Instância de redirecionamento

3. APP1: Instância de aplicações frontend
4. SISPREV: Instância de sistema previdenciário não governamental
5. APP3: Instância de armazenamento de imagens

Em contexto apresentado do estado dos equipamento apresentados para esse projeto de estudo, o pesquisador estará utilizando os materiais disponibilizados pela a estrutura da instituição para a sua construção com base na leitura bibliografias levantadas em fase da oportunização de entrega futura do ambiente final proposto.

3.2 Cenário de equipe

Junto aos desenvolvedores, desempenham um papel de planejamento na validação da arquitetura dos sistemas, garantindo a qualidade do código e a eficiência das soluções desenvolvidas. Além de ([MUNIZ et al., 2019](#)) embasar que o processo que deve melhorar, educar e planejar sempre a continuada no âmbito dos integrantes da equipe, o momento em que conhecimentos teóricos e práticos são partilhados.

Assim membros novos dentro da equipe devem ser tratados discentes trazem energia renovada e vontade de aprender e contribuir para os projetos. Os desenvolvedores experientes têm a oportunidade de mentorear, construir habilidades e competências profissionais, transmitindo boas práticas de desenvolvimento de software. Construindo um ambiente de aprendizado colaborativo, ético, participativo e favorável para a viabilidade de diagnósticos eficazes diante das demandas requeridas pela instituição.

Apesar da equipe ser composta por um número limitado de desenvolvedores, a colaboração conforme ([RATO, 2017](#)), o compartilhamento de conhecimentos entre os membros permitem que os projetos avancem de forma eficiente. A liderança de alguns desenvolvedores aliada à dedicação e vontade da equipe resultam em um ambiente de trabalho produtivo e propício à criação de soluções de software de qualidade.

Outrossim, a documentação está em formato objetivo e abrangente o que garante aprendizados de resolutividade quanto ao uso das ferramentas levantadas e sua viabilidade integrativas. A eficiência e a produtividade ofertadas, decorrem destas ferramentas, que aceleram o desenvolvimento e o automatizar das tarefas repetitivas, oferecendo recursos em alto grau que favorecem ainda mais o processo de trabalho em desenvolvimento de sistemas.

4 Implementação dos métodos

A implementação de ferramentas do projeto consiste nas Gitlab, Jenkyns, Docker e as micro-aplicações dentro do PREVIPALMAS, utilizando todas as ferramentas citadas, onde estão focadas na entrega do projeto. A concepção deste projeto/pesquisa foi idealizada a partir da solicitação de desenvolvimentos de aplicações, para responder às operações dos analistas de maneira eficiente. No foco de melhoria na comunicação transparente, das atividades exercidas e no controle dos materiais utilizados dentro do instituto para o desenvolvimento dos seguintes softwares:

- Portal de notícias
- Sistema de controle de estoque

A partir de solicitação via chamado ao setor de TI do instituto que desenvolve os softwares conforme anexo I e II para as requisições vigentes foram levantados requisitos mínimos para sua sequência de implementação das aplicações propostas, pensando sempre na regra de negócios levantados. Havendo a entrega de forma rápida após as modificações feitas para as mesmas com suas respectivas validações pelo setor demandante.

4.1 Etnografia e versões de teste

Aplicou-se o padrão etnográfico e versões de teste. Com o objetivo de levantar os requisitos do processo de desenvolvimento de software, onde cada peculiaridade exposta pelo requerente seja validada com todas as equipes envolvidas. Diante da solicitação de criação do modelo software foi feita a opção do método de etnografia e versão de teste, que se encaixam no contexto qual é proposto o projeto, por não haver existência de um consenso dos preceitos do modelo, importante é observar a interação das pessoas em um cenário, percebendo como funciona a forma de trabalho.

Entre as opções de versão de sistema de teste, desenvolvimentos compartilhados, o grupo de trabalho tende a requerer uma maior coordenação à vista dos requisitos da construção das aplicações, para toda a equipe de TI do instituto. Assim verificando uma abordagem mais eficiente e menos onerosa para o início da implementação do projeto.(MATTOS, 2011), a etnografia é bastante utilizada por antropólogos. Consiste em analisar a conexão entre os indivíduos separadamente e sua relação com o grupo de estudos de observação. Aplicado em várias áreas de atuação da tecnologia da informação, é executável na compreensão da forma de trabalho de cada setor da organização, passando a ideia inicial dos elementos e caracterizações na iminência de resolver problemas levantados e

propondo soluções. Em (ROCHA; ECKERT, 2008) discutem que a referência do observador da pesquisa, precisa estar atrelada no laço temporal e espacial, levantando as análises, com cautela. O observador não pode alterar o fluxo das atividades exercidas, e assim conseguir uma maior fidelidade dos dados coletados.

Dentro desse pensamento, a etnografia é uma metodologia a basilar de levantamento de requisitos das atividades requeridas dos projetos solicitados, com as suas peculiaridades de implementação, com uma maior eficácia dos tipos de requisitos, tendo em vista que a equipe de desenvolvimento já faz parte da organização, pode-se avaliar naturalmente o fluxo de operação e extrair suas informações observando o ambiente constante.



Figura 5 – Fluxo de Requisitos

Fonte: Imagem do Autor

Com os requisitos levantados, começa a codificação e versionamento de testes de validação dos serviços. Nesse sentido, o estudo de (RIOS; FILHO, 2013), demonstra a necessidade de um processo de versionamento de teste ou, mais popularmente, teste de correções de erros, o qual direcionamos. Dentro dessa visão para uma equipe de pequeno porte de desenvolvimento acaba se tornando um excelente ganho de assertividade a aplicação solicitada.

Validações das versões, além de possibilitar o feedback para os setores demandantes dos serviços, quanto ao andamento e ainda mais enfático, leva a equipe de TI, para uma codificação com a garantia de que o projeto está correndo em conformidade ou não ao que foi levantado pelo método de análise de requisitos.

Em (SATO, 2013) destaca que nessa fase, após o levantamento - pesquisa de campo, é importante salientar que a cultura de DevOps tem que externar todas as boas práticas. Isso implica dizer que o processo de implantação da cultura juntamente com o arrolamento dos requisitos para análises, antes de seguir detalhadamente um processo ágil, deve ocorrer ciência de que, há um processo contínuo e que precisa ser implantado aos poucos na organização e todos os integrantes devem fazer parte dessa melhoria.

Esse processo é de fato contínuo e progressivo na entrega dos demais passos dos projetos e sempre haverá a possibilidade das modificações quando solicitadas utilizando os princípios da cultura de DevOps.

Incorporado ao ambiente de codificação e validação da prototipação do projeto, com base no estudos de (VALENTE, 2020), os critérios de codificação com métodos automáticos e com validação de funcionalidades já definidas, apesar de ter validações unitárias na vida real onde os usuários, via manuseio ao software que está sendo desenvolvido, possam agregar valor às propriedades da análise de requisitos, dando assim uma fidelidade ao processo e uma conformidade para a continuidade da ideia concebida e almejada com o resultado.

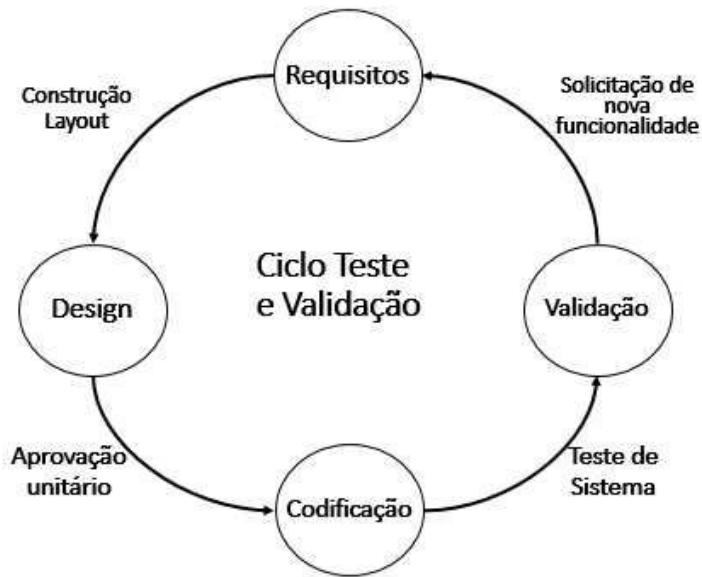


Figura 6 – Ciclo Teste de Validação

Fonte: Imagem do Autor

De maneira que os testes de validação, conforme figura nº6 da prototipação, demonstra que o resultado já está bem encaminhado para um desfecho, seja positivo das aplicações ou uma reestruturação da análise de requisitos com a ideia pré-concebida do projeto. Onde essa intersecção da análise de confirmação da continuidade do projeto deve ser feita pelo setor demandante, dando assim, posicionamento para um caminho que deve ser seguido ou projetado. Importante destacar a etapa de codificação.

4.2 Codificação

Com os levantamentos de requisitos aprovados pelo setor solicitante conforme anexo III e IV, inclusive o ciclo de validação já estabelecido, bem como a característica pronta para implementação e utilização dos seus usuários, passem a agregar funções, em decorrência do reconhecimento dos serviços do PREVIPALMAS, essas soluções as quais estão sendo solicitadas à equipe e pensando na resolução, por meio das análise da linguagem que deve ser utilizada, junto ferramentas as quais, provoque que os serviços fossem realizados com maior eficiência para uma equipe de pequeno porte.

No contexto do modelo conforme figura 7 os bancos de dados que estão sendo utilizados por serviços de sistemas autônomos também dependem de recursos ao mesmo nível. O SGBD. As tabelas 2 ,3 e 4 com os requisitos levantados inicialmente de cada aplicação, descrevem o cenário com os levantamentos de requisitos junto ao setor demandante.

As tabelas com os requisitos levantados inicialmente de cada aplicação estão expressas nas tabelas nº3 e na tabela nº4, descrevem o cenário com os levantamentos de requisitos junto ao setor demandante.

Noticia	Tipo
Id	Number
Titulo	String
Texto	Text
Image	String
Key Author	String
Key Editor	String
Key Publicador	String
Autorização Editor	Boolean
Autorização publicador	Boolean
Data de Criação	Date

Tabela 2 – Sistema de noticias - tabela de noticia

Fonte: Tabela do Autor

Produtos	Tipo
Id	Number
Nome	String
Quantidade	Number
Validade	Date
Criação	Date

Tabela 3 – Sistema de estoque - tabela de produtos

Fonte: Tabela do Autor

Uma vez dentro desse projeto que existe um instrumento que fará o armazenamento dos cargos, departamentos e dos indivíduos, constatou que não havia a necessidade dessa criação de tabela. Em teoria, será uma separação de campos onde só é armazenada a chave estrangeira dos indivíduos nos armazenamento da aplicação e a tabela com informações dos profissionais em outra aplicação com maior capacidade de funcionalidades.

Para criar com facilidade variações de ambiente de desenvolvimento junto à equipe, mantendo a configuração semelhante ao código final de produção. A codificação de aprovisionamento da cultura de DevOps é aplicada e replicada para colocar um analista localmente em uma linha de base de versionamento conhecida. Quando isso é feito, a cifragem da estrutura haverá a replicação do código, ainda sim, mantendo a necessidade de haver manual de execução e documentação interna sempre atualizados, assim o resultado são processos repetidos e sistemas confiáveis em todas as instâncias dos colaboradores.

Com objetivo de uma solução de API-Rest's é utilizada a ferramenta supramencionada o NestJS, tornando o instrumento tranquilo de ser operacionalizado com poucas linhas de comando, fazendo integração com banco de dados quanto com a integração front-end dinamicamente. O NestJS, nesse sentido, está sendo utilizado em três API como na figura 11 na camada 2, separadas pelo fato onde cada um fará um trabalho específico, assim aplicando conceito de micro serviço separadamente um do outro.

Pedidos	Tipo
Id	Int
Key Usuario	String
Key Setor	String
Quantidade	Number
Criação	Date
Status	Boolean
Entregue	Boolean
Descrição	String
Key Produtos	Int

Tabela 4 – Sistema de estoque - tabela de pedidos

Fonte: Tabela do Autor

Sistemas autônomos de back-end que fazem separadamente as atividades consumindo recursos em par com métodos serão, estão destacados na figura 7. Isso determina que as soluções precisam passar por reparo ou até mesmo interrupção do seu serviço, e ainda assim não acarretará prejuízo às demais aplicações. Para a sequência do modelo da figura 7 a solução com interação do usuário com a API pode ser utilizada facilmente entre o conjunto de comunicações, com as demais ferramentas sendo abordadas e isso mostra a versatilidade no desenvolvimento da parte que o projeto necessita.

Assim como no back-end o front-end seguirá o mesmo padrão de projeto com micro serviços com a separação dos programas e trabalhando individualmente em container isoladamente, tendo cada sistema um código gerenciado, não interferindo em qualquer uma das outras aplicações.

Back-end e front-end sendo implantados em um sistema de gerenciamento autônomos e conversando entre si por meio de páginas confiáveis com o auxílio do gerenciador de identidade o KEYCLOAK, criar atividades com seus respectivos conteúdos com uma roupagem significativa com o foco no princípio da cultura de DevOps aplicando aos microsserviços necessárias.

4.3 Controle de versionamento

Essa lógica opera com base no Controle de Versionamento, em que há uma exceção da parte da autorização e automatização de processos e a totalidade de implementação

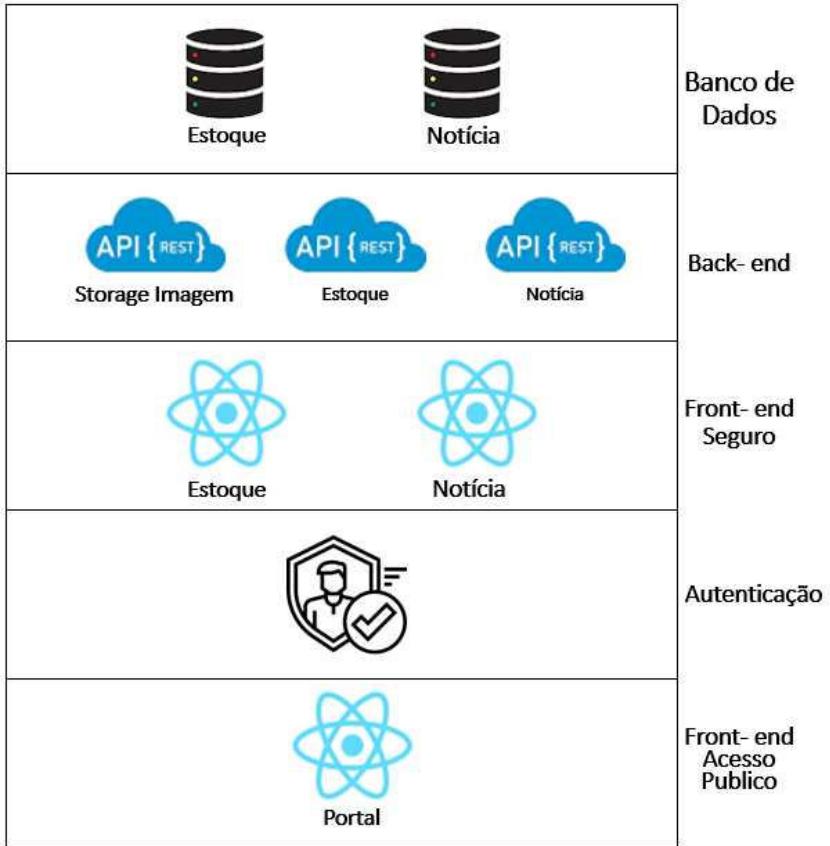


Figura 7 – Modelo de Projeto
Fonte: Imagem do Autor

é desenvolvida, sendo uma ferramenta funcional. As etapas adicionais exigem que o sistema de controle de versão do código esteja sempre atualizado, dando continuidade ao processo de projeto, em via de implementação pelo PREVIPALMAS, o que garantirá uma manutenção contínua e confiável. No âmbito do controle de versão, preocupa-se em aumentar a segurança e a velocidade de automatização das aplicações em desenvolvimento, a figura 8 reflete esse movimento.

Movimento que indica que ocorrendo o ingresso de um membro novo ou uma nova implementação de funcionalidade, somente com ação das medidas básicas dentro do sistema de controle de versão, o processo automatizado por execução de ramificação de uma ramificação do local, é acionado. Portanto, os analistas do sistema podem realizar seus próprios testes para a validação e verificação da estrutura de desenvolvimento aplicadas.

Após a análise do administrador as validações unitárias das ações feitas na manutenção ou acréscimo de funcionalidades, junções ocorrem, bem explica a figura 8, no cerne do ramo principal do projeto, logo futuramente a implementação se estabiliza, sendo monitorado seu desenvolvimento.

Posteriormente as validações unitárias, têm-se um controle o controle de versão aliado à implementação de boas práticas de DevOps, assim há o incremento da sequência

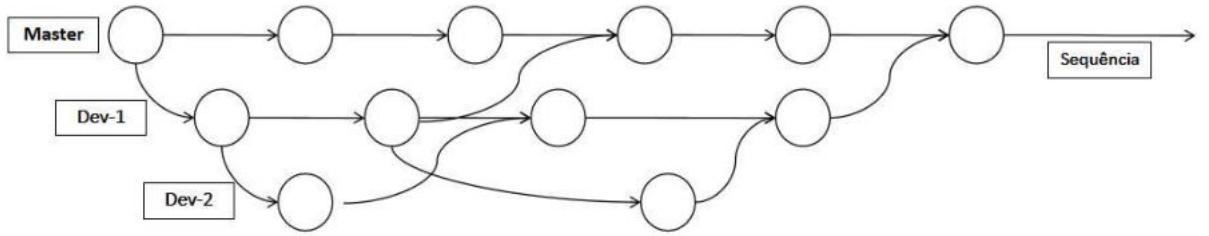


Figura 8 – Versionamento de Código
Fonte: Imagem do Autor

de automação do projeto e suas fases, que dependem que o controle de versão, a todo tempo siga sincronizado com a versão de teste.

Seguindo este percurso de pesquisa, com um viés de continuidade para a mesma, retornamos a ideia do pedido de alteração em novo regulamentos, novos mecanismos e até mudanças em diretrizes iniciais, de acordo com todo o fluxo das figuras já apontadas, o que impulsiona ao aspecto automatizar a criação do projeto, nas linhas de adaptações, em relação às demandas que aparecem junto ao de versionamento.

4.4 Autenticação e autorização

Nessa linha de análise a pesquisa infere que os mecanismos de autenticação e autorização, são instrumentos utilizados no projeto, bem representados na Figura 7 em nível 05 para camada 2, que impulsiona a verificação da legitimidade, criando um JWT, que autentica e autoriza a sessão no front-end e, portanto, as permissões no back-end.

Havendo uma verificação em modo simples, se extingue a precisão de solicitações individuais para um serviço, e a criação de dispositivos, para cada verificação, dando os direitos de confirmação, em um só eixo. Assim os analistas, no ramo de aplicativos, tendem a não se preocupar em criar e/ou implementar integração entre aplicativos individualmente.

O benefício esperado pela utilização desta ferramenta é atender as requisições de integração entre os sistemas autônomos via API e as troca de chaves de acesso via JWT que são totalmente integradas às micro-aplicações, tornando-se um mecanismo confiável, simples e intuitivo quando usado. Logo as abas que precisam de autorização podem ser tratadas com maior eficiência da disponibilidade de uso. Tal como exemplificado na figura 8, as camadas superiores do projeto não podem ser acessadas sem a devida autenticação, o que gera segurança, pois os usuários não poderão acessar certas funções às quais não têm permissão.

Neste parâmetro os Micro Serviços, descritos nos pontos da figura 7 as camadas superiores do projeto não podem ser acessadas sem a devida autenticação verificada em instrumentos previamente.

4.5 Micros serviços

Neste parâmetro os Micro Serviços, descritos nos pontos da figura 8, mostram fragmento da aplicação apontadas, nas figuras seguintes, por meio da ferramenta Docker, cada instância do modelo se tornará um container independente e isolado internamente → um sistema separado entre si, personalizado e com os recursos do dispositivo compartilhados. Para isso as aplicações podem ser criadas separadamente umas das outras sem grande robustez e/ou grande número de servidores para alocar serviços das aplicações.

Relatamos que mesmo havendo serviços isolados, que se comunicam conforme figura 7, podem interagir e reagir, ao comando e necessidades quando necessário das outras instâncias, mas com a garantia de que as necessidade de reparos ou imprevistos não impliquem na não continuidade da demanda do PREVIPALMAS, Vejamos a figuras 9, 10 e 11.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
bcbec11019cc	portal	"docker-entrypoint.s..."	2 days ago	Up 2 days
c7933ff09f40	estoque	"docker-entrypoint.s..."	12 days ago	Up 12 days
7a9900709871	noticia	"docker-entrypoint.s..."	3 weeks ago	Up 3 weeks
77ad813294d8	91bdbece70e4	"/bin/sh -c 'npm ins..."	3 weeks ago	Exited (1) 3 weeks ago
88bb19105ad3	1ce0d007202f	"/bin/sh -c 'npm ins..."	3 weeks ago	Exited (1) 3 weeks ago
Finished: SUCCESS				
PORTS		NAMES		
0.0.0.0:3000->3000/tcp, :::3000->3000/tcp		portal_previpalmas		
0.0.0.0:3002->3000/tcp, :::3002->3000/tcp		estoque_previpalmas		
0.0.0.0:3001->3000/tcp, :::3001->3000/tcp		noticia_previpalmas		
		nostalgic_hodgkin		
		wizardly_wing		

Figura 9 – Conteiner Server 1

Fonte: Imagem do Autor

4.6 Integração e Automação

Nesse sentido o Sistema Integração e Automação, é um sistema-alvo em fase de implementação, levando a fase final do projeto de integração e automação, que visualiza quaisquer anomalias ou gerencie falhas da parte principal que podem acarretar os ganhos positivos esperados ou mesmo os pontos negativos.

Do ponto de vista da integração será uma forma de conectar o código e todas as áreas de microsserviços, utilizando um formato conteinerizado como base na individualização de cada sistema. As cenas apresentadas estão longe do trabalho intensivo em mão-de-obra, trabalho intensivo, de acordo com a figura 7, “à quantidade de 14 Microsserviços” envolvidos

Figura 10 – Conteiner Server 2

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
4eb83560b541	portal	"docker-entrypoint.s..."	2 days ago	Up 2 days
60e1eddbc228	estoque	"docker-entrypoint.s..."	12 days ago	Up 12 days
328597ec8cbe	noticia	"docker-entrypoint.s..."	3 weeks ago	Up 3 weeks
2e06ff029335	6eb1158884cb	"/bin/sh -c 'npm ins..."	3 weeks ago	Exited (1) 3 weeks ago
5d77a7c8a629	86b10d72c7fb	"/bin/sh -c 'npm ins..."	3 weeks ago	Exited (1) 3 weeks ago
60445316a9bf	9f8e9c19fac2	"/bin/sh -c 'npm ins..."	3 weeks ago	Exited (1) 3 weeks ago
Finished: SUCCESS				
POROS			NAMES	
0.0.0.0:3000->3000/tcp, :::3000->3000/tcp			portal_previpalmas	
0.0.0.0:3002->3000/tcp, :::3002->3000/tcp			estoque_previpalmas	
0.0.0.0:3001->3000/tcp, :::3001->3000/tcp			noticia_previpalmas	
			nostalgic_dubinsky	
			infallible_colden	
			musing_clarke	

Fonte: Imagem do Autor

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
dfec37922d58	estoque	"docker-entrypoint.s..."	12 days ago	Up 12 days
afc4d72d3ebc	postgres:13.3	"docker-entrypoint.s..."	3 weeks ago	Up 3 weeks
7747057ea496	noticia	"docker-entrypoint.s..."	2 months ago	Up 2 months
2434ca56df6b	postgres	"docker-entrypoint.s..."	2 months ago	Up 2 months
068c5a4a756e	storage	"docker-entrypoint.s..."	3 months ago	Up 2 months
ef6a2fab3e94	bitnami/keycloak:12-debian-10	"/opt/bitnami/script..."	3 months ago	Up 2 months
db0d696ae406	bitnami/postgresql:11-debian-10	"/opt/bitnami/script..."	3 months ago	Up 2 months
STATUS	POROS		NAMES	
Up 12 days	0.0.0.0:3002->3000/tcp, :::3002->3000/tcp		estoque_previpalmas	
Up 3 weeks	0.0.0.0:5433->5432/tcp, :::5433->5432/tcp		estoque_db_1	
Up 2 months	0.0.0.0:3001->3000/tcp, :::3001->3000/tcp		noticia_previpalmas	
Up 2 months	0.0.0.0:5434->5432/tcp, :::5434->5432/tcp		noticia_db_1	
Up 2 months	0.0.0.0:3000->3000/tcp, :::3000->3000/tcp		storage_previpalmas	
Up 2 months	0.0.0.0:8090->8080/tcp, :::8090->8080/tcp		keycloak_keycloak_1	
Up 2 months	5432/tcp		keycloak_postgresql_1	

Figura 11 – Conteiner Server 4

Fonte: Imagem do Autor

diretamente neste projeto. Essencialmente, a integração dessas soluções, resolução de cinco camadas/níveis, conforme o projeto.

A saber a camada 1, infere que, existe o armazenamento dos dados, ocorrendo em duas instâncias separadamente, mesmo que sejam feitas por um mesmo banco de dados, essa individualidade de cada uma tira o que a automação possa trabalhar e manipular, separadamente suas interações.

Para a camada 2, representa-se a área interna das aplicações dos back-end, em que existem aplicações de API, sistematicamente são:

- Storage: A API é responsável pelo armazenamento das imagens as quais os projetos devem preservar. Essa solução traz uma redução da carga do banco de dados onde somente irá guardar o seu nome e o caminho e não o arquivo completo. Essa abordagem permite que as outras “abas” possam ser modificadas e reutilizadas, em seu padrão de automação com uma tranquilidade em perda desses arquivos;
- Estoque: API fará o gerenciamento dos produtos, os quais o PREVIPALMAS apresenta, ou se não deve incorporar aquisições pertinentes, e também terá a incumbência de gerir o pedido de entrega dos mesmos, para cada setor solicitante bem como todos os relatórios necessários que possam ser solicitados;
- Notícias: API tem como funcionalidade armazenar e manipular as notícias front-end do portal. Como existem regras específicas observadas junto ao levantamento de requisitos do setor demandante, a codificação exige uma atenção para os cargos e permissões.

Na camada 3, é um front-end seguro das API's, ou seja, dependendo de uma autorização e autenticação, dentro dos serviços e cada aspecto com a interação com os usuários e suas personalidades.

Em relação à camada 4, há uma autenticação e autorização das sessões, considerando que há permissões imprescindíveis para que os usuários possam acessar as suas interações tanto com front-end quanto suas respectivas API's, logo essa possui um papel determinante para o sucesso de gerir toda a estrutura desse projeto.

A camada 5, é a parte que se localiza o front-end público, e qualquer beneficiário que queira ter acesso às demais instâncias de interação com usuários autenticados ou não. Nessa camada os beneficiários do PREVIPALMAS terão acesso às informações relevantes, notícias diversas, quanto as pautas dos serviços que necessitem ser repassadas a elas como informativos, esclarecimentos ou quaisquer outros, a serem repassados de maneira prática e célere, e com o menor gasto de recurso físico ou computacional.

O que agrega a função do agente de automação, a identificação das possíveis alterações que vão e/ou estão sendo praticadas pela equipe, para a criação da nova imagem e deploy [9](#) aponta esses passos verticais, processos dinâmicos e ininterruptos até o deploy da aplicação com a execução do projeto expondo-o para a sua utilização.

Nesse sentido é que pensamos, problematizamos a importância hoje na utilização da ferramenta, que leva a automação de todos os serviços práticos, já que após a sua implantação inicial, se faz rapidamente toda a conta de conteinerização das aplicações separadamente, e com sua integração efetivamente concluída. Ainda neste cenário, as aplicações serão feitas as requisições de automação a partir de um único nó onde estará o agente de irá direcionar para qual das instâncias deverá ser feita a automação.

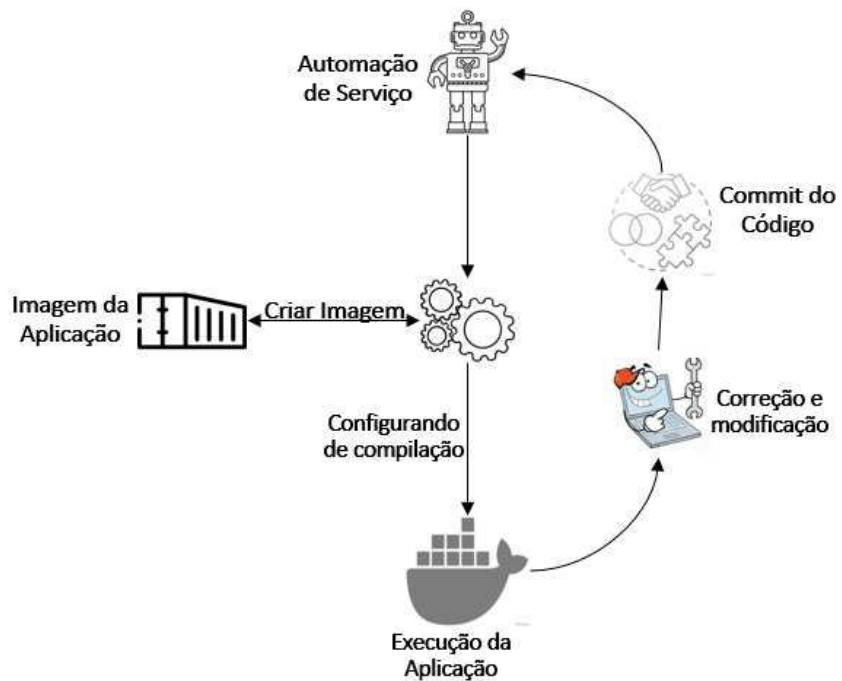


Figura 12 – Fluxo de Automação
Fonte: Imagem do Autor

Caso exista a necessidade de escalabilidade de novas aplicações, essas poderão ser feitas dessa mesma maneira, colocando os vários “nós” dos servidores disponibilizados ou até mesmo contratar um serviço em nuvem de hospedagem.

Ademais, isso é possível, porque a ferramenta de automação está em total capacidade de executar dentro do cenário proposto, a partir das funcionalidades bem definidas. Observando sempre que é informado o próximo “nó” para automação, deixando o serviço ainda mais rápido, mantendo a mesma característica é a mesma eficiência, mesmo que existisse somente um servidor.

5 Resultados

Os dados – aqui, estão detalhados, as análises e a integração das ferramentas, com base na figura 13, que mostra a divisão do serviço no desenvolvimento das aplicações, no âmbito da codificação em microsserviços e a entrega dos mesmos em funcionamento interno.

De acordo com o objetivo do projeto, para atender a demanda solicitada foi necessário a implantação de uma base, de aos moldes da figura 4. Constatamos que a apresentação dos mecanismos fundamentais para a instalação das ferramentas necessárias é um marco essencial para o sucesso de projetos de desenvolvimento em sistemas de informação, como esta pesquisa.

A partir da observação da figura 4 dentro do server-01, o sistema de chamados representou uma aplicação com a responsabilidade de gerenciar e organizar as solicitações que eventualmente serão feitas para a manutenção. Deixando sempre um histórico das necessidades solicitadas envolvendo cada aspecto, sendo aplicáveis para a gestão de desenvolvimento das aplicações, assim como utilizadas para tarefas de manutenção do dos equipamentos do instituto.

Outrossim pensando em atenuar atividades preventivas diante das atividades de desenvolvimento das aplicações requeridas “manutenção por imprevistos” e seu tempo de espera, projetou-se uma forma de monitoramento aos equipamentos e das instâncias das aplicações, por meio de levantamentos de indicadores. Por se tratar de mecanismo de suporte para que a equipe esteja focada no resultado esperado, o instrumento foi implantado juntamente com o intuito de atender as ocorrências que vivem os analistas mesmo com chamado em aberto, delegando assim um tempo a ser organizado ou mesmo antecipando alguma eventual solicitação. No cerne do objetivo do projeto, ficaram definidas as tarefas deixando cada um dos integrantes, responsáveis pelas codificações necessárias de cada atividade.

Com as divisões efetuadas, cada um dos projetos foi subdividido em três etapas:

1. Gerenciamento;
2. Portal de notícias;
3. Sistema de estoque

Consta de codificação separada entre front-end e o back-end, por se tratar de uma aplicação simples, pouco é necessário para a escrita do código, e também representa o sistema de estoque. Os Relatórios de gestão para o controle de estoque estão sempre em

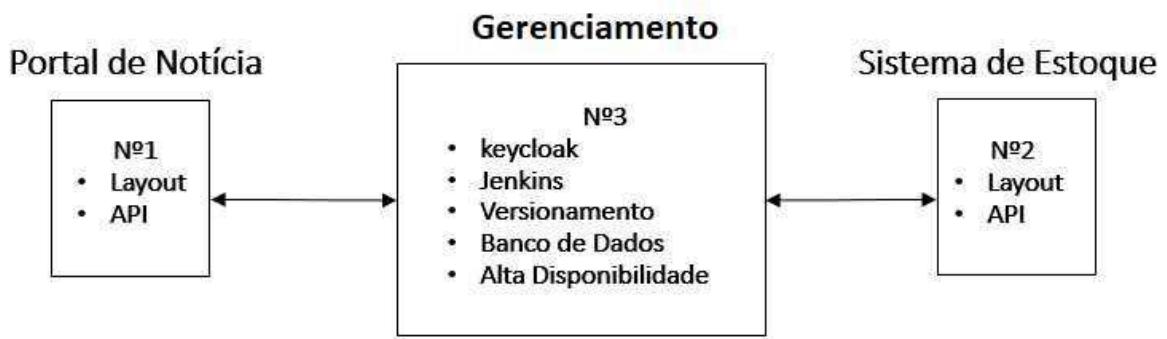


Figura 13 – Divisão dos projetos

Fonte: Imagem do Autor

desenvolvimento, assim a manutenção de correções do mesmo acaba tornando-se uma necessidade, para a implementação de mudanças constantes no código fonte.

Assim é importante se ater às solicitações externas de informações e valores. Posto que, sendo a necessidade de geração de relatórios para informar ao órgão de fiscalização e controle de quantidade dos produtos adquiridos estão em conformidade com o que está em estoque.

A figura 13 também aponta o gerenciamento de aplicações e suas automatizações centralizadas em pontos estratégicos e separadamente das tarefas dentro do modelo proposto e gerenciável para a equipe. De modo que o responsável por esse requisito acaba tornando-se o administrador do projeto e validando as informações para o deploy final das aplicações.

A pauta da colaboração aqui é descrita, no formato da integração dos itens de implantação do sistema de versionamento, conforme figura 14 isto é, uma integração entre os membros da equipe para que eles possam efetivamente compartilhar os códigos de forma mais controlada como se espera da cultura DevOps.



Figura 14 – Gitlab server

Fonte: Imagem do Autor

No intuito de apresentar a instância foi disponibilizada para a versão n. 13.12.3 conforme a figura 15 com os requisitos necessários para este projeto.

1. Características

Memória: 8Gb

HD: 256Gb

SO: Debian

Na instância de projeção do sistema de versionamento repositórios de código-fonte de cada projeto, foi gerenciado branches e fusões, revisões de código de forma eficiente com gerenciamento de tarefas que contribuem para uma melhor organização e comunicação no projeto.

Destaca-se que foram criados vários versionamentos iniciais para teste e a inicialização do compartilhamento do projeto, com o amadurecimento dos conceitos por parte da equipe, se desenvolveu a base e o fluxo de trabalho. Vejamos a figura 17.

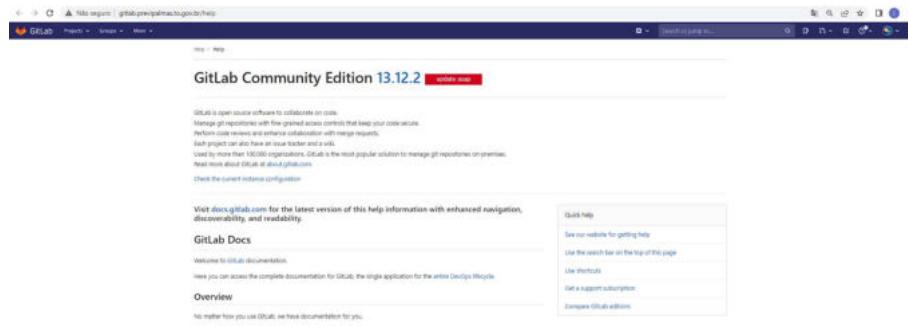


Figura 15 – Gitlab PREVIPALMAS

Fonte: Imagem do Autor

Conforme a figura 8, em cada vertente há um profissional para o gerenciamento, um para o desenvolvimento do portal de notícias do PREVIPALMAS e dois para o sistema de gerenciamento de estoque do PREVIPALMAS

Com as divisões de trabalho definidas, os responsáveis pelo controle de estoque seguem com o padrão de layout ajustado e conferido entre todos os integrantes. Assim o desenvolvimento das regras de negócios dentro do back-end e levantados com os requisitos junto ao setor demandante, tendo como sua validação da solução apresentada e aceita.

Para o panorama do portal de notícias o desenvolvimento do backend e frontend, o profissional ficou encarregado da criação do layout que é usado entre todas as micro aplicações de front-end e os demais integrantes tiveram a meta de desenvolver as regras do cadastro dos informativos, que serão apresentados pelo site oficial da instituição e no sistema de estoque.

Apresentando a documentação inicial do portal de notícias em pequeno detalhamento das funcionalidades, o portal consegue executar a sua implementação simplificada, em poucas funcionalidades de código. Contudo com peso real para as informações relevantes que serão disponibilizadas aos usuários do portal.

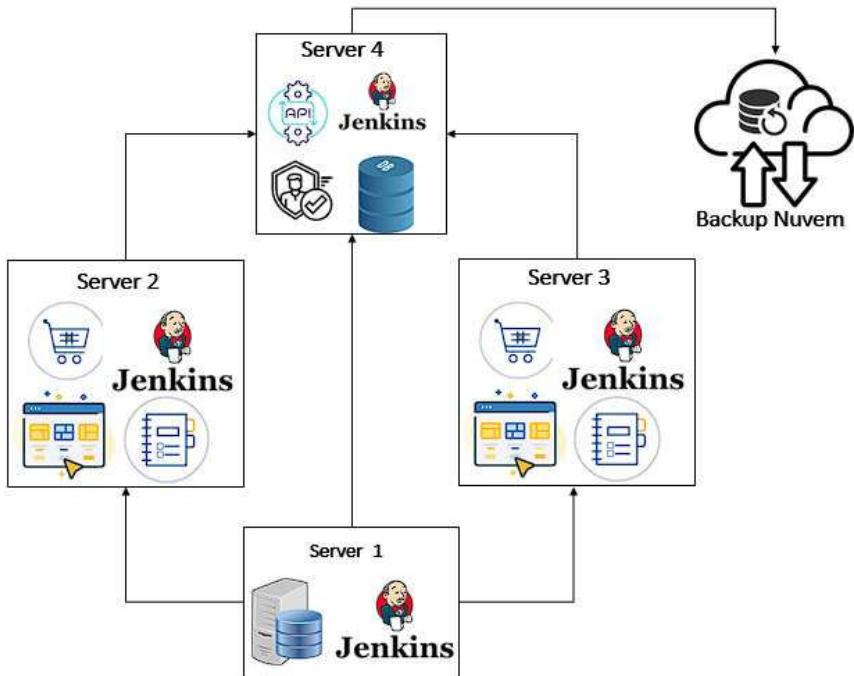


Figura 16 – Arquitetura do Projeto
Fonte: Imagem do Autor

Assim com o profissional que está a cargo do gerenciamento das aplicações ficará com um papel central no projeto, onde o mesmo gerencia tempo de entrega das aplicações, teste de validação, gerenciamento da ferramenta de CI/CD e o dispositivo de autenticação das aplicações. Ademais os os microsserviços, precisam uma comunicação de redirecionamento de rotas, as duas equipes têm que trabalhar e se comunicarem para definirem modelos de transição com rotas definidas e assim dando uma segurança de retorno das aplicações corretas.

Por isso pensou-se em aplicar a estratégia de “conversa / reunião de equipe”, para se estruturar elaboração de métodos de compartilhamento de código e construção da integração com as ferramentas. Elemento que deve garantir êxito em projetos dessa vertente, já que essa comunicação deve ocorrer por parte de todos os integrantes, lembrando que, o profissional que está à frente do gerenciamento acaba assumindo o papel fundamental para a integração das ferramentas externas do projeto que estão sendo implantadas.

Por tal comprehende-se que a colaboração dentro da cultura de DevOps está ligada a esse tópico onde os projetos sempre mantiveram a sua documentação dentro dos projetos de versionamento. Isso parte do princípio que a continuidade e andamento do projeto podem ser feitos pela equipe que aqui se encontra ou por outros profissionais que venham agregar a equipe. Vejamos a figura 18

A associação de das ferramentas do projeto, modelou a execução conforme a figura 18. Posto que cada micro serviço é um ambiente definido, na sua integração não pode haver falhas de comunicação, por parte da equipe ou na implementação dos conceitos.

A	Celio Gouvea / Auth_login	Maintainer	★ 0 Y 0 I 0 D 0
B	Celio Gouvea / backend_images_storage	Maintainer	★ 0 Y 0 I 0 D 0
B	Murilo Alves Melo / BACK_ESTOQUE	Maintainer	Em Funcionamento
B	Celio Gouvea / BACK_NOTICIA	Maintainer	Em Funcionamento
D	Celio Gouvea / Desafio	Maintainer	● ★ 0 Y 0 I 0 D 0
D	Celio Gouvea / DevOps - PREVIPALMAS	Maintainer	★ 0 Y 0 I 0 D 0
E	Celio Gouvea / Estoque	Maintainer	● ★ 0 Y 0 I 0 D 0
F	Celio Gouvea / Frontend_Noticias	Maintainer	★ 0 Y 0 I 0 D 0
F	Murilo Alves Melo / FrontEndPrevEstoque	Maintainer	★ 0 Y 0 I 0 D 0
F	Murilo Alves Melo / FRONT_ESTOQUE	Maintainer	Em Funcionamento
F	Celio Gouvea / FRONT_NOTICIA	Maintainer	Em Funcionamento
K	Celio Gouvea / keycloak	Maintainer	★ 0 Y 0 I 0 D 0
M	Celio Gouvea / myzap	Maintainer	● ★ 0 Y 0 I 0 D 0
P	Celio Gouvea / ponto	Maintainer	★ 0 Y 0 I 0 D 0
P	Ti / Ponto Eletronico	Developer	★ 0 Y 0 I 0 D 0

Figura 17 – Repositorios
Fonte: Imagem do Autor

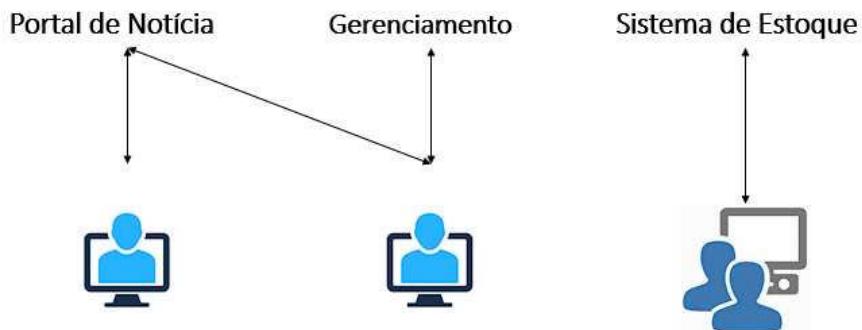


Figura 18 – Divisão dos trabalhos
Fonte: Imagem do Autor

Para o papel de unificação, o responsável da vertente da gerenciamento disponibilizou todas as ferramentas, para aplicação dos analistas. Esse compartilhamento das versões dos códigos fontes de desenvolvimento foi primordial para o seu sucesso do projeto.

Ressalta-se que no contexto na implantação, o projeto foi feito em quatro instâncias virtuais diferentes, distribuídas dentro de dois servidores físicos, em cada servidor com suas peculiaridades. O servidor 1 alocou a ramificação principal de automação com o jenkins.

Logo nessa parte, ocorre a aplicação, o serviço e o gerenciador de tarefas entre as demais instâncias das demais máquinas virtuais, bem como a automação com cada um deles tendo um "nó" para a execução. Conforme a figura 16, a partir da commit da codificação no sistema de versionamento é reconhecido e destinado ao servidor de responsabilidade que se encarrega de executar o deploy. Também dentro do servidor 1, existe a escalabilidade do projeto, que define qual o servidor com as devidas aplicações será retornado ao usuário.

Para o papel de unificação, o responsável da vertente da infraestrutura disponibilizou todas as ferramentas, para aplicação dos analistas. Esse compartilhamento das versões dos códigos fontes de desenvolvimento foi primordial para o seu sucesso do projeto.

Na construção do projeto do PREVIPALMAS, a implantação do sistema, prevê redirecionamento das rotas para os outros servidores e aplicações. Mesmo havendo a queda do servidor, o serviço ficará disponível. Os levantamentos de requisitos do projeto foram levantados no anexo III e anexo IV e registrados dentro do posicionamento de código. Cada projeto em específico apresenta layout com os levantamentos de requisitos para cada funcionalidade levantada, apresentada a devida solicitante e aprovada, com os detalhes da demanda posteriormente colocada em implementação junto do escopo.

As aplicações são exatamente os mesmos, como demonstrado na figura 16, respectivamente estão com as aplicações de front-end do portal de notícias, front-end de cadastro de notícias e o front-end do sistema de estoque. Além do Jenkins de ramificação onde recebem as informações do principal e montam os containeres de cada programa. E por fim, o servidor 4 hospeda os containeres de banco de dados das suas aplicações bem como das API de conexão com os bancos e das regras de negócios levantadas, além do gerenciamento das autorizações de funcionalidades pelo sistema KEYCLOAK. O servidor 4 tem como papel enviar para a unidade de backup externo uma cópia da última versão dos códigos fontes das aplicações, bem como o backup do banco de dados com as informações cadastradas.

No cenário proposto, seriam utilizados dois servidores para formar um cluster conforme figura n. 19. Cada servidor teria duas instâncias diferentes, totalizando quatro instâncias no ambiente. Essas instâncias poderiam ser configuradas para executar diferentes aplicações do projeto.

Uma das principais vantagens desse modelo é a capacidade de alta disponibilidade. Caso um dos servidores apresente falhas ou precise ser desligado para manutenção, as outras instâncias continuariam funcionando, garantindo a continuidade dos serviços. Isso reduziria significativamente o impacto de eventuais interrupções no sistema, aumentando a confiabilidade e a disponibilidade das aplicações.

Apesar de ainda não ter sido implementado conforme a figura n. 16, o planejamento para a aplicação do cluster demonstra uma abordagem proativa e estratégica para o

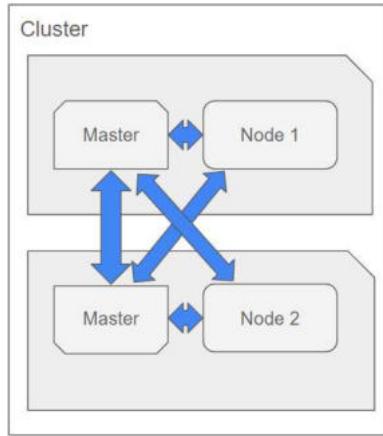


Figura 19 – Cluster planejado para implementação
Fonte: Imagem do Autor

gerenciamento de infraestrutura. A intenção de adotar essa solução no futuro reflete o desejo de aprimorar a infraestrutura e preparar o ambiente para o crescimento e as demandas futuras.

Ao apresentar os resultados dessa abordagem, é importante enfatizar a intenção de implementar o cluster no futuro, destacando os benefícios esperados, ainda analizando a figura n. 19 como maior disponibilidade, escalabilidade e resiliência. Além disso, é fundamental ressaltar o planejamento cuidadoso realizado, considerando aspectos como a capacidade dos servidores, a distribuição das instâncias e as necessidades específicas do projeto.

Em suma, a proposta de aplicação de um cluster em dois servidores com quatro instâncias representa uma abordagem estratégica para o aprimoramento da infraestrutura. Embora ainda não tenha sido executada, a intenção de implementar essa solução no futuro demonstra o compromisso em buscar uma arquitetura de alta disponibilidade e escalabilidade.

A parte final do projeto com todos os requisitos levantados e a codificação, sendo necessário o deploy dos projetos para as instâncias, o processo de automatização apresenta a sua funcionalidade, associada a sua demanda respectiva. O processo de automatização, em conjunto com as boas práticas de DevOps que estão sendo implantadas dentro do cenário do PREVIPALMAS, é demonstrado junto ao acompanhamento de listagem e implantação dos códigos-fonte ao mesmo. É benéfico e eficiente no ganho de produtividade.

Nesse sentido, quando um dos projetos exige a necessidade de modificação do escopo anterior, a automação do projeto recebe a informação do controle de versionamento, com as funções de automação estabelecidas dentro dos servidores, a ramificação principal do Jenkins master e repassa para as demais, para realizar o deploy.

Logo deixando a fileira de automação esse processo em tempo de execução automá-



Figura 20 – Fila de projetos para automação

Fonte: Imagem do Autor

tica exige não necessariamente que o profissional responsável pela implementação esteja verificado durante o tempo da ação. Dentro desse período de execução, os servidores e seu um conjunto de instruções predefinidas na configuração, conforme a figura 20, mostram comandos, que seguem uma ordem para que as ações da automação tornem eficientes e eficazes, e garantindo a segurança da entrega do deploy nos respectivos servidores.

```
docker rm -f portal_previpalmas
docker image rm portal
docker build . -t portal
docker run --name portal_previpalmas -td -p 3000:3000 portal
```

Figura 21 – Exemplo de comandos para automação

Fonte: Imagem do Autor

O primeiro comando conforme 21, representa a exclusão do container criado anteriormente nos servidores. A exclusão é necessária para que o futuro container assuma o mesmo nome e a mesma porta estabelecida. De maneira que se o mesmo for implantar a nova versão do projeto acontecerá conflito de nome e endereçamento. O segundo comando, representa a exclusão da versão ultrapassada do projeto, por uma questão de organização de versões anteriores dos projetos já computados pelo administrador do projeto, optou-se pela exclusão da variante do escopo. Deixando assim, sempre a pasta dos projetos limpa e com o mínimo de arquivos para não sobrecarregar a instância.

Na terceira linha de comando da figura 21, existe a criação do novo container com todas as dependências do projeto, conforme a versão atualizada do sistema que está sendo modificado e as novas estruturas. Assim sendo a parte do processo que mais demanda tempo

Estado do executor de builds		
	master	
1	Parado	
2	Parado	
	Server APP01	
1	PORTAL	#25
	Server APP02	
1	PORTAL2	#21
		FRONT_NOTICIA2
		PORTAL
		PORTAL2
		SGBD_START
		Status APP1
		Status APP2

Figura 22 – Automação em tempo de execução
Fonte: Imagem do Autor

Assim, deixando a fileira de automação, conforme a fig. n. 20 para a fig. n. 23, esse processo em tempo de execução automática exige não necessariamente que o profissional responsável pela implementação esteja verificado durante o tempo da ação.

Dentro desse período de execução o que acontece com os servidores, um conjunto de instruções predefinidas na configuração, conforme a fig. n. 21, esses comandos seguem uma ordem para que as ações da automação tornem eficientes e eficazes, e garantindo a segurança da entrega do deploy nos respectivos servidores.

O primeiro comando representa, conforme a fig. n. 21, a exclusão do container criado anteriormente nos servidores. A exclusão é necessária para que o futuro container assuma o mesmo nome e a mesma porta estabelecida. De maneira que se o mesmo for implantar a nova versão do projeto acontecerá conflito de nome e endereçamento.

O segundo comando da fig. n. 21 representa a exclusão da versão ultrapassada do projeto, por uma questão de organização de versões anteriores dos projetos já comitados pelo administrador do projeto, optou-se pela exclusão da variante do escopo. Deixando assim, sempre a pasta dos projetos limpa e com o mínimo de arquivos para não sobrecarregar os servidores.

Na terceira linha de comando da figura n. 21 e a criação do novo container com todas as dependências do projeto, conforme a versão atualizada do sistema que está sendo modificado e as novas estruturas. Assim sendo a parte do processo que mais demanda tempo. Por fim o último comando da figura 21, sugere para o deploy efetivamente a aplicação, de instrução e depois da criação do container, expõe a porta da aplicação junto ao sistema, e assim deixando com que a aplicação esteja visível e funcional. A figura 23 acaba se tornando a abstração de todos os comandos da figura 21 em que haja uma quebra de processo e ocasiona erros no deploy das aplicações.

Esse processo dentro do cenário do escopo do projeto, para cada aplicação mon-

tando todos os contêineres levará em média quatro minutos a partir da autorização do administrador no versionamento de código. Se tratando das métricas e validações, a qualidade funcional de um sistema e de seu processo utilizado para desenvolvê-lo, pode ser entendida como um conjunto de métricas consideradas críticas para o bom atendimento das necessidades do usuário final. Cada serviço necessário ou modificação no sistema representa um defeito funcional que precisa ser corrigido.

O número de correções funcionais não podem ser uma medida deste projeto, onde a automatização de cada projeto e o processo de inserção de novas funções são tidos em conta como aspecto geral do projeto e, se necessário, a solução é diretamente atualizada.

Estado do executor de builds		
	master	
1	Parado	
2	Parado	
	Server APP01	
1	<u>PORTAL</u>	#25
	Server APP02	
1	<u>PORTAL2</u>	#21
	FRONT_NOTICIA2	
	PORTAL	
	PORTAL2	
	SGBD_START	
	Status APP1	
	Status APP2	

Figura 23 – Automação em tempo de execução
Fonte: Imagem do Autor

Por fim o último comando do figura n [21](#) e para o deploy efetivamente da aplicação, essa instrução depois da criação do container expõe a porta da aplicação junto ao sistema, e assim deixando com que a aplicação esteja visível e funcional.

A figura n [23](#) acaba se tornando a abstração de todos os comandos de figura n [21](#) em que haja uma quebra de processo e ocasiona erros no deploy das aplicações. Esse processo dentro do cenário do escopo do projeto, para cada aplicação montando todos os contêineres levará em média quatro minutos a partir da autorização do administrado no versionamento de código.

Assim, recursos como os informados abaixo dão uma base da qualidade e validações das soluções propostas:

1. Facilidade de uso
2. Facilidade de reuso
3. Auto aprendizagem
4. Percepção favorável do usuário

A construção foi realizada com o código-fonte, inicialmente por meio do levantamento das necessidades dos departamentos que precisavam transmitir informações relacionadas a outras unidades da cidade de Palmas, que verificavam e controlavam o custo de materiais e informações junto ao portal de notícias.

Os elementos considerados como necessidades do usuário possuem perfis diferentes, pois cada usuário possui um grau de importância diferente, portanto não é calculado o mesmo grau de importância para suas necessidades de informação.

Por exemplo, em um ambiente competitivo, informações sobre qualidade de produtos e serviços podem ter um peso maior para os usuários, do que informações sobre uso ou satisfação. Portanto, precisamos determinar as taxas de preferência do usuário para vários requisitos do sistema de informação.

Como os itens são novos e pouco usados, não há métricas previamente definidas para mensurar seu uso, porque todos dentro do Instituto ainda estão se ajustando aos novos modelos, solicitando materiais ao sistema de estoque e postando novas mensagens no portal. Basta dizer que a aceitação desses dois sistemas e suas soluções totalmente integradas hoje não apresenta nenhuma solicitação de correções ou modificações, ou mesmo novos requisitos para implementar alterações no código-fonte.

5.1 Portal de Notícias

Conforme o anexo V o portal de notícias é uma forma fácil de apresentar notícias aos usuários previdenciários da PREVIPALMAS, notícias relevantes do dia a dia para os funcionários e agenda de reuniões do conselho.

As informações apresentadas no portal são especialmente elaboradas por profissionais responsáveis e serão verificadas pelos superiores antes da publicação. Isso garante que notícias ou informações relacionadas à PREVIPALMAS não sejam encaminhadas diretamente para a Internet.

Nesta verificação, apenas determinados profissionais estão autorizados a postar notícias no portal. Tais solicitações foram resolvidas conforme necessário na coleta de requisitos com o executivo responsável pelos requisitos do departamento de TI.

5.2 Sistema de Estoque

Já na aplicação, foi realizado um levantamento de necessidades diretamente com o Departamento Financeiro e Contábil da PREVIPALMAS, resultando em uma solução para as necessidades de desenvolvimento.

No desenvolvimento do controle de estoque, são necessárias múltiplas implemen-

tações conforme anexo V, sendo relatadas modificações e constantes exceções de dados, exigindo o uso múltiplo de sistemas automatizados. Ressalte-se que em nenhum momento tais modificações interferirão nas atividades em andamento, onde os produtos são registrados, enquanto outros setores estão pesquisando materiais de aplicação.

6 Conclusão

Os resultados apresentados, abrangem uma solução para atender os requisitos de funcionais para a entrega dos softwares sob demanda solicitados pelo departamento de TI onde foi aprovada por meio de convocação do comitê responsável. Assim, todas as ferramentas de investigação de provisionamento de requisitos levantados para a codificação das micro-aplicações, em um conjunto de trabalho disponibilizado para fazer os mecanismos funcionarem conjuntamente e manter a segurança dos serviços prestados, deixando em pleno funcionamento.

Investigar os sistemas que devem interessar à PREVIPALMAS, que não existem ou carecem de posterior implementação para que possam agregar valor aos serviços da instituição com maior eficiência e eficácia na produtividade das atividades diárias. As sugestões para o futuro desenvolvimento de software independente do instituto de pesquisa são as seguintes:

1. Orquestração dos serviço em Cluster
2. Implantar novas ferramentas ao ciclo DevOps
3. Implantar novos sistemas
 - a) Sistema de gestão de Regime Próprio de Previdência Social - RPPS de segurados e pensionistas;
 - b) Sistema de agendamento médico para aposentadoria e ou pensão;
 - c) Sistema de emissão de contracheque dos beneficiários;
 - d) Portal da transparência.
 - e) Sistema de ponto eletrônico;
 - f) Sistema de agendamento de disponibilidade do veículo institucional.

Atualmente, na situação financeira do Instituto, a continuidade dos contratos de prestação de serviços relacionados à manutenção de ferramentas acarreta custos bastante elevados. O que está sendo proposto não ignora as capacidades técnicas das empresas prestadoras de serviços, mas visa reduzir os custos econômicos de longo prazo para o PREVIPALMAS.

Fechamento do projeto proposto e dos sistemas que precisam ser implantados, recomendações para continuidade e melhoria de trabalhos futuros, recomendações para implantação dos sistemas acima e implementação de todos os conceitos da cultura DevOps para atingir os resultados listados abaixo:

1. Diminuição de riscos;
2. Melhor comunicação entre os setores da instiruição;
3. Soluções maior estabilidade e desempenho.

Assim, o final deste projeto é um precursor dos princípios da cultura DevOps e do cenário do PREVIPALMAS para que cada projeto futuro tenha uma mentalidade clara e concorde com os gestores da agência para mudar o estado atual dos departamentos em que estará envolvida as atividade.

Referências

- ARUNDEL, J.; DOMINGUS, J. *DevOps nativo de nuvem com Kubernetes*. [S.l.: s.n.], 2019.
- BAJDIUK, C. U.; MEIRELLES, F. D. S. *A CULTURA DEVOPS NA TRANSFORMAÇÃO DIGITAL ATRAVÉS DA AFFORDANCE ACTUALIZATION*. [s.n.], 2020. Disponível em: <<http://bibliotecadigital.fgv.br/ocs/index.php/ctd/ctd2020/paper/view/7643/0>>.
- BARCELAR, R. R. *Sistemas Distribuidos*. 2018. 15 p. Disponível em: <https://www.academia.edu/18485047/Aula_4_Supercomputa%C3%A7%C3%A3o>.
- BAYLÃO, A. L. D. S.; OLIVEIRA, V. M. D. *Impacto da evolução tecnológica na gestão empresarial*. [s.n.], 2015. Disponível em: <<https://www.aedb.br/seget/arquivos/artigos15/14922205.pdf>>. Acesso em: 2022.
- BITBUCKET. 2022. Disponível em: <<https://www.atlassian.com/br/git/tutorials/why-git#git-for-developers>>. Acesso em: 19 fev. 2022.
- COSTA, H. L. A. *Alta Disponibilidade e balanceamento de carga pra melhoria de sistemas computacionais críticos usnado software livre: um estudo de caso*. 2009. Disponível em: <<https://www.locus.ufv.br/bitstream/123456789/2594/1/texto%20completo.pdf>>. Acesso em: 20 jan. 2022.
- DEVOPS. 2022. Disponível em: <<https://www.objective.com.br/insights/testes-automatizados-na-implantacao-de-devops/>>. Acesso em: 19 fev. 2022.
- DOCKER. 2022. Disponível em: <<https://www.docker.com/>>. Acesso em: 19 fev. 2022.
- FERREIRA, F. et al. *Clusters de alta disponibilidade – uma abordagem Open Source*. 2005. Disponível em: <https://www.dcc.fc.up.pt/~mantunes/papers/Eng2005_1.pdf>. Acesso em: 17 mar. 2022.
- FORSGREM, N. et al. *ACCELERATE buidin and Scaling High Performing Tecnology Organizations*. 2018. 286 p.
- GARCIA, I. *Os desafios enfrentados e os benefícios conquistados com a cultura e as práticas DevOps*. [s.n.], 2020. Disponível em: <<https://app.uff.br/riuff/bitstream/handle/1/22493/Iuri-Garcia-TCC-final-secretaria%20%282%29.pdf?sequence=1&isAllowed=y>>.
- GOMES, A.; SOUZA, B. Desenvolvimento de software com node.js. *Revista de Tecnologia da Informação*, v. 10, n. 2, p. 43–56, 2019.
- HUMBLE, j.; FARLEY, D. *Entrega Continua*. 2014.
- JENKINS. 2022. Disponível em: <<https://www.jenkins.io/>>. Acesso em: 19 fev. 2022.
- MATTOS, C. L. G. d. *A abordagem etnográfica na investigação científica*. 2011. 37 p. Disponível em: <<https://books.scielo.org/id/8fcfr/pdf/mattos-9788578791902-03.pdf>>.

MORAIS, G. *Caixa de ferramentas DevOps*. CASA DO CODIGO, 2015. ISBN 978-85-5519-082-7. Disponível em: <<https://github.com/free-educa/books/blob/main/books/Caixa%20de%20Ferramentas%20DevOps%20Um%20guia%20para%20construcao%2C%20administracao%20e%20arquitetura%20de%20sistemas%20modernos%20-%20Casa%20do%20Codigo.pdf>>.

MUNIZ, A. et al. *Jornada DevOps: unindo cultura ágil, Lean e tecnologia para entrega de software com qualidade*. Brasport, 2019. ISBN 9788574529271. Disponível em: <<https://books.google.com.br/books?id=94qXDwAAQBAJ>>.

NESTJS. 2017–2022. Disponível em: <<https://docs.nestjs.com/>>. Acesso em: 10 out. 2021.

NETO, R. et al. Balanceamento de carga de rede: Alto desempenho e escalabilidade para serviços computacionais. 12 2018. Disponível em: <https://www.researchgate.net/publication/329708095_Balanceamento_de_Carga_de_Rede_Alto_Desempenho_e_Escalabilidade_para_Servicos_Computacionais>.

NODEJS. 2019. Disponível em: <<https://nodejs.org/en/>>. Acesso em: 01 out. 2019.

POSTGRESQL. 2022. Disponível em: <<https://www.postgresql.org/>>. Acesso em: 19 fev. 2022.

RATO, F. d. O. *DevOps em Sistemas de Informação*. [s.n.], 2017. Disponível em: <<https://run.unl.pt/bitstream/10362/35816/1/TGI0143.pdf>>. Acesso em: 2022.

REACT. 2019. Disponível em: <<https://pt-br.reactjs.org/>>. Acesso em: 01 out. 2019.

RIOS, E.; FILHO, T. M. *Teste de Software*. [s.n.], 2013. 17 p. Disponível em: <https://www.academia.edu/37989821/Teste_de_Software>.

ROCHA, A. L. C. d.; ECKERT, C. *Etnográfica: Saberes e Práticas*. 2008. 23 p. Disponível em: <<https://www.lume.ufrgs.br/bitstream/handle/10183/30176/000673630.pdf>>.

SATO, D. *Devops Na Pratica: ENTREGA DE SOFTWARE CONFIÁVEL E AUTOMATIZADA*. CASA DO CODIGO, 2013. ISBN 978-85-5519-082-7. Disponível em: <<https://books.google.com.br/books?id=nTgdvgAACAAJ>>.

SATO, D. *DevOps na prática: Entrega de software confiável e automatizada*. Casa do Código, 2014. ISBN 9788566250664. Disponível em: <<https://books.google.com.br/books?id=Cm2CCwAAQBAJ>>.

SILVA, A. d. S. d. et al. *Equipes de alta performance e o papel do líder como construtor de resultados*. [s.n.], 2011. Disponível em: <<http://pg.utfpr.edu.br/expout/2011/artigos/24.pdf>>. Acesso em: 2022.

SILVA, M. S.; OLIVEIRA, R. F. Node.js as an efficient platform for web development: A systematic literature review. *Journal of Systems and Software*, Elsevier, v. 162, p. 110435, 2020.

SOUZA, L. et al. *DevOps - fundamentos e perspectivas*. 2019.

TEAM, R. *Introducing React.dev*. 2023. <<https://pt-br.react.dev/blog/2023/03/16/introducing-react-dev>>. Acesso em 10 de Maio de 2023.

VALENTE, M. T. *Engenharia de Software Moderna*. [s.n.], 2020. Disponível em:
<https://engsoftmoderna.info/>.

ANEXO I

Chamado			
Chamado - ID 1100			
Data de abertura	02-02-2022 21:03	Por	FERNANDA LEAO
Última atualização	31-08-2022 17:29 por Murilo Alves Melo		
Tempo para atendimento		Tempo para solução	
Tempo interno para atendimento		Tempo interno para solução	
Data da solução	31-08-2022 17:29	Data de fechamento	31-08-2022 17:29
Tipo	Requisição	Categoria	Computador > Software
Status	Fechado	Origem da requisição	Helpdesk
Urgência	Média	Aprovação	Não está sujeita a aprovação
Impacto	Médio	Localização	Financeiro
Prioridade	Média		
Autor	Requerente	Observador	Atribuído para
	FERNANDA LEAO i	Celio Gomes Gouvea i	Murilo Alves Melo i
Título	SOLICITA DESENVOLVIMENTO DE SOFTWARE		
Boa tarde,			
Descrição *	Solicito o desenvolvimento de software de controle de estoque para utilização dos servidores do PREVIPALMAS. Grata		
Chamados relacionados			

Processando chamado 4**Histórico de ações :**

Foi entregue e concluído o sistema de estoque e já está sendo utilizado.
Aguardando feedback para futuras atualizações.

(L) 31-08-2022 17:29

Helpdesk



Murilo Alves Melo i

(L) 18-03-2022 19:22



FERNANDA LEAO i

Para criação de relatórios do estoque:

RELÓRIO 1 - DESCRIÇÃO DOS ITENS + QUANTITATIVO + VALIDADES

RELÓRIO2- PEDIDOSCOMITENS PORDIRETORIA/ASSESSORIA

RELÓRIO 3 - ENTRADA E SAÍDA... DESCRIÇÃO DO ITEM + QUANTIDADE DE ENTRADA + VALIDADE + QUANTIDADE DE SAÍDA

Helpdesk

(L) 21-02-2022 21:57



FERNANDA LEAO i

Boa tarde.

Como tenho reuniões dia 22 e 23 que podem durar a tarde inteira, solicito que a reunião seja organizada para dia 24/02 às 15h.

Grata.

Helpdesk

(L) 18-02-2022 16:58

17/05/2023, 16:31

GLPI - Chamados - 1100

O software de controle de estoque já está em fase de finalização e está pronto para uma apresentação, assim podemos discutir alguns pontos para o fechamento do projeto e para seguir para a apresentação para os servidores e começar uma fase de testes somente dos usuários do FINANCEIRO no momento.



Murilo Alves Melo i

Sugerimos a data do dia 22/02/2022 a partir das 16hrs.

Helpdesk

02-02-2022 21:03



FERNANDA LEAO i

SOLICITA DESENVOLVIMENTO DE SOFTWARE

Ticket # 1100 description

Boa tarde,

Solicito o desenvolvimento de software de controle de estoque para utilização dos servidores do PREVIPALMAS.

Grata

Estatísticas

Datas

02-02-2022 21:03 * Data de abertura

11-02-2022 18:35 ✓ Atribuição

31-08-2022 17:29 ■ Data de fechamento

31-08-2022 17:29 ✓ Data da solução

Tempos

Atribuição	8 dias 21 horas 32 minutos
Solução	209 dias 20 horas 26 minutos
Fechamento	209 dias 20 horas 26 minutos 24 segundos
Pendente	

Aprovações

Aprovações

Status de validação global

Esperando por uma validação

Estado

Um mínimo de validação é necessária 0%

Aprovação para o(s) chamado(s)

Nenhum item encontrado

Base de Conhecimento

Vincular uma entrada na base de conhecimento

..... i

Nenhuma entrada na base de conhecimento vinculada

Itens

Adiciona um item

Meus periféricos

Ou busca completa

Tipo	Entidade	Nome	Número de série	Número de inventário
------	----------	------	-----------------	----------------------

Custos

Custo	Duração do item	0 segundo
Nenhum item encontrado		

Projetos**Adicionar um projeto****Projeto****Tarefas do projeto****Adicionar uma tarefa ao projeto****Problemas****Adicionar um problema****Abrir um problema a partir deste chamado****Problema****Mudanças****Adicionar uma alteração****Criar uma mudança a partir deste chamado****Mudança****Histórico 14****Histórico**Exibir (número de itens)

De 1 para 14 de 14

ID	Data	Usuário	Campo	Atualizar
Mostrar filtros 				
14033	31-08-2022 17:29	Murilo Alves Melo (111)	Acompanhamento	Adiciona um item: Acompanhamento (1446)
14032	31-08-2022 17:29	Murilo Alves Melo (111)	Data da solução	Mudança de para 31-08-2022 17:29
14031	31-08-2022 17:29	Murilo Alves Melo (111)	Data de fechamento	Mudança de para 31-08-2022 17:29
14030	31-08-2022 17:29	Murilo Alves Melo (111)	Status	Mudança de Processando (planejado) para Fechado
12310	18-03-2022 19:22	FERNANDA LEAO (98)	Acompanhamento	Adiciona um item: Acompanhamento (1234)
11923	21-02-2022 21:57	FERNANDA LEAO (98)	Acompanhamento	Adiciona um item: Acompanhamento (1189)
11902	18-02-2022 16:58	Murilo Alves Melo (111)	Acompanhamento	Adiciona um item: Acompanhamento (1186)
11901	18-02-2022 16:58	Murilo Alves Melo (111)	Status	Mudança de Processando (atribuido) para Processando (planejado)
11784	11-02-2022 18:35	Celio Gomes Gouvea (88)	Status	Mudança de Novo para Processando (atribuido)
11783	11-02-2022 18:35	Celio Gomes Gouvea (88)	Usuário	Adicionar um relacionamento com um item: Murilo Alves Melo (111) (Técnico)
11782	11-02-2022 18:35	Celio Gomes Gouvea (88)	Leve em conta o tempo	Mudança de 0 segundo para 8 dias 21 horas 32 minutos
11781	11-02-2022 18:35	Celio Gomes Gouvea (88)	Usuário	Adicionar um relacionamento com um item: Celio Gomes Gouvea (88) (Observador)
11658	02-02-2022 21:03	FERNANDA		Adicionar o item

17/05/2023, 16:31

GLPI - Chamados - 1100

ID	Data	Usuário	Campo	Atualizar
Mostrar filtros 				
		LEAO (98)		
11657 02-02-2022 21:03 FERNANDA LEAO (98) Usuário Adicionar um relacionamento com um item: FERNANDA LEAO (98) (Requerente)				
ID	Data	Usuário	Campo	Atualizar

Histórico

Exibir (número de itens) 5000

De 1 para 14 de 14

ANEXO II

Chamado**Chamado - ID 1101**

Data de abertura	02-02-2022 21:04	Por	FERNANDA LEAO
Última atualização	17-05-2023 19:31 por Oziel Silva Ferreira		
Tempo para atendimento		Tempo para solução	
Tempo interno para atendimento		Tempo interno para solução	
Data da solução	17-05-2023 19:31	Data de fechamento	17-05-2023 19:31
Tipo	Requisição	Categoria	Computador > Software
Status	Fechado	Origem da requisição	Helpdesk
Urgência	Média	Aprovação	Não está sujeita a aprovação
Impacto	Médio	Localização	
Prioridade	Média		
Autor	Requerente	Observador	Atribuído para
 FERNANDA LEAO		 Celio Gomes Gouvea	 Celio Gomes Gouvea
Título	SOLICITA DESENVOLVIMENTO DE SOFTWARE		
BOA TARDE			
Descrição *	SOLICITO O DESENVOLVIMENTO DE SOFTWARE DO PORTAL DE NOTÍCIAS A SER UTILIZADO PELO PREVIPALMAS.		
GRATA			
Chamados relacionados			

Processando chamado 3**Histórico de ações :**

- (⌚ 17-05-2023 19:31) Processo finalizado!  Oziel Silva Ferreira
- (⌚ 16-05-2022 20:53) Haja vista que o prazo de entrega não foi cumprido, solicito que o servidor Celio desenvolva o layout do portal de notícias.
Grata  FERNANDA LEAO
- (⌚ 25-03-2022 18:37) REGISTRO PRAZO PARA ENTREGA DEFININO EM EM 18/03/2022 PARA O DIA 18/04/2022.  FERNANDA LEAO
- (⌚ 02-02-2022 21:04) **SOLICITA DESENVOLVIMENTO DE SOFTWARE**  BOA TARDE
SOLICITO O DESENVOLVIMENTO DE SOFTWARE DO PORTAL DE NOTÍCIAS A SER UTILIZADO PELO PREVIPALMAS.

GRATA

Estatísticas**Datas**

02-02-2022 21:04 Data de abertura

11-02-2022 18:34 Atribuição

17-05-2023 19:31 Data de fechamento

17-05-2023 19:31 Data da solução

Tempos

Atribuição 8 dias 21 horas 30 minutos

Solução 468 dias 22 horas 27 minutos

Fechamento 468 dias 22 horas 27 minutos 0 segundos

Pendente

Aprovações**Aprovações**

Status de validação global

Esperando por uma validação

Estado

Um mínimo de validação é necessária

0%

Aprovação para o(s) chamado(s)

Nenhum item encontrado

Base de Conhecimento

Vincular uma entrada na base de conhecimento

Nenhuma entrada na base de conhecimento vinculada

Itens

Adiciona um item

Meus periféricos -----

Ou busca completa -----

Tipo	Entidade	Nome	Número de série	Número de inventário
------	----------	------	-----------------	----------------------

Custos

Custo

Duração do item

0 segundo

Nenhum item encontrado

Projetos

Adicionar um projeto

Projeto**Tarefas do projeto**

Adicionar uma tarefa ao projeto

Problemas[Adicionar um problema](#)[Abrir um problema a partir deste chamado](#)**Problema****Mudanças**[Adicionar uma alteração](#)[Criar uma mudança a partir deste chamado](#)**Mudança****Histórico 17****Histórico**

De 1 para 17 de 17

ID	Data	Usuário	Campo	Atualizar
Mostrar filtros				
15899	17-05-2023 19:31	Oziel Silva Ferreira (95)	Acompanhamento	Adiciona um item: Acompanhamento (1669)
15898	17-05-2023 19:31	Oziel Silva Ferreira (95)	Data da solução	Mudança de para 17-05-2023 19:31
15897	17-05-2023 19:31	Oziel Silva Ferreira (95)	Data de fechamento	Mudança de para 17-05-2023 19:31
15896	17-05-2023 19:31	Oziel Silva Ferreira (95)	Status	Mudança de Processando (atribuído) para Fechado
13334	23-06-2022 20:37	Celio Gomes Gouvea (88)	Status	Mudança de Novo para Processando (atribuído)
13333	23-06-2022 20:37	Celio Gomes Gouvea (88)	Usuário	Adicionar um relacionamento com um item: Celio Gomes Gouvea (88) (Técnico)
13332	23-06-2022 20:37	Celio Gomes Gouvea (88)	Usuário	Excluir um link com um item: Oziel Silva Ferreira (95)
13331	23-06-2022 20:37	Celio Gomes Gouvea (88)	Última edição por	Mudança de 95673679149 (98) para 01430857196 (88)
13330	23-06-2022 20:37	Celio Gomes Gouvea (88)	Status	Mudança de Processando (atribuído) para Novo
12892	16-05-2022 20:53	FERNANDA LEAO (98)	Acompanhamento	Adiciona um item: Acompanhamento (1308)
12337	25-03-2022 18:37	FERNANDA LEAO (98)	Acompanhamento	Adiciona um item: Acompanhamento (1236)
11780	11-02-2022 18:34	Celio Gomes Gouvea (88)	Status	Mudança de Novo para Processando (atribuído)
11779	11-02-2022 18:34	Celio Gomes Gouvea (88)	Usuário	Adicionar um relacionamento com um item: Oziel Silva Ferreira (95) (Técnico)
11778	11-02-2022 18:34	Celio Gomes Gouvea (88)	Leve em conta o tempo	Mudança de 0 segundo para 8 dias 21 horas 30 minutos
11777	11-02-2022 18:34	Celio Gomes Gouvea (88)	Usuário	Adicionar um relacionamento com um item: Celio Gomes Gouvea (88) (Observador)
11660	02-02-2022 21:04	FERNANDA LEAO (98)		Adicionar o item
11659	02-02-2022 21:04	FERNANDA LEAO (98)	Usuário	Adicionar um relacionamento com um item: FERNANDA LEAO (98) (Requerente)
ID	Data	Usuário	Campo	Atualizar
Histórico				
Exibir (número de itens)		5000		
De 1 para 17 de 17				

ANEXO III



A gestão de estoque é um aspecto fundamental para empresas de todos os segmentos, pois a falta ou excesso de produtos pode impactar diretamente nas finanças e na reputação da empresa. Nesse contexto, a utilização de um sistema de gerenciamento de estoque se torna essencial para garantir uma gestão eficiente, organizada e controlada das mercadorias.

Com o aumento da concorrência no mercado e a crescente demanda dos consumidores, a gestão de estoque se torna ainda mais crítica para empresas que buscam se destacar e se manter competitivas. Além disso, a falta de controle sobre o estoque pode gerar prejuízos financeiros e impactar a entrega de produtos aos clientes, o que pode prejudicar a imagem da empresa e afetar a fidelidade dos consumidores.

Um sistema de gerenciamento de estoque permite que a empresa tenha um controle preciso e atualizado de todas as mercadorias disponíveis em seu estoque, bem como dos gastos relacionados à sua gestão. Com essa ferramenta, é possível verificar quais produtos estão em falta, quais estão em excesso, além de permitir o planejamento de compras futuras e a identificação de produtos com maior e menor giro.

Além disso, um sistema de gerenciamento de estoque permite que a empresa atribua responsabilidades e controle o acesso dos usuários ao sistema, garantindo uma gestão mais transparente e segura. Essa ferramenta também pode ser utilizada para gerar relatórios e gráficos que ajudam na análise de desempenho da empresa, facilitando a tomada de decisão e a identificação de áreas que precisam de melhorias.



Em resumo, a utilização de um sistema de gerenciamento de estoque se torna cada vez mais importante para empresas que buscam se manter competitivas no mercado e garantir a satisfação dos clientes. Com essa ferramenta, é possível ter um controle preciso e atualizado das mercadorias, dos gastos relacionados à gestão do estoque e da responsabilidade de cada usuário. Além disso, a geração de relatórios e análises de desempenho ajuda na tomada de decisão e no planejamento estratégico da empresa.

1. Requisitos funcionais:

- a) Registro de entrada e saída de materiais de escritório e materiais de limpeza no estoque.
- b) Armazenamento dos dados de fornecedores e fornecimento para cada material.
- c) Possibilidade de editar e excluir registros de entrada e saída de materiais.
- d) Alertas de estoque mínimo e máximo para cada tipo de material.
- e) Registro do responsável pelo uso de cada tipo de material.
- f) Geração de relatórios de estoque atualizado e de movimentação de materiais em determinado período.
- g) Visualização de relatórios de gastos com materiais por setor ou por responsável pelo uso.
- h) Integração com sistema financeiro para registro de gastos e recebimentos relacionados aos materiais.
- i) Controle de validade e descarte de materiais vencidos.
- j) Possibilidade de exportar relatórios em formatos como Excel ou PDF.

2. Requisitos não Funcionais:

JOSÉ



Segurança: o sistema deve garantir a segurança dos dados e informações do estoque, evitando o acesso não autorizado e prevenindo perda de informações. É importante que seja utilizado algum mecanismo de autenticação e autorização, além de backups frequentes.

Usabilidade: a interface do sistema deve ser intuitiva e fácil de usar, mesmo para usuários com pouca experiência em sistemas de gerenciamento de estoque. Isso inclui elementos como botões, menus e formulários bem projetados e organizados.

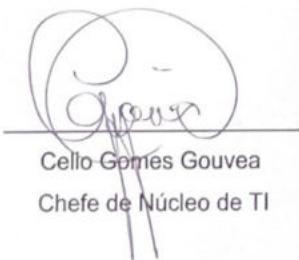
Performance: o sistema deve ter uma performance rápida e confiável para que os usuários possam acessar as informações do estoque rapidamente, sem perda de tempo. Também é importante garantir que o sistema possa lidar com grandes volumes de dados sem perder desempenho.

Disponibilidade: o sistema deve estar disponível para uso em tempo integral, com poucas ou nenhuma interrupção do serviço. Isso inclui a garantia de que o sistema estará disponível em casos de falha de energia ou de hardware.

Escalabilidade: o sistema deve ser capaz de lidar com um grande volume de usuários e dados sem comprometer o desempenho. Deve ser projetado para permitir a adição de novas funcionalidades e a expansão do armazenamento de dados.

Compatibilidade: o sistema deve ser compatível com diferentes dispositivos e plataformas, como computadores, smartphones e tablets, e também deve ser compatível com diferentes navegadores e sistemas operacionais.

Manutenção: o sistema deve ser projetado de forma modular e com uma documentação completa, para facilitar a manutenção e correção de eventuais problemas ou falhas. É importante também que seja possível realizar atualizações sem interrupção do serviço



Cello Gomes Gouveia
Chefe de Núcleo de TI



Fernanda Dias Machado Zerbini Leão
Diretora

ANEXO IV



O mundo atual está cada vez mais conectado e informações estão disponíveis em tempo real na internet. Em meio a essa realidade, os portais de notícias surgem como uma ferramenta fundamental para manter a população informada sobre os acontecimentos.

Com a evolução da tecnologia e a popularização da internet, a demanda por portais de notícias online aumentou consideravelmente. Isso se deve à facilidade de acesso, rapidez na divulgação de informações e a possibilidade de atualização em tempo real.

Além disso, os portais de notícias possibilitam a veiculação de diversos tipos de conteúdo, como textos, fotos, vídeos e áudios, o que enriquece a experiência do usuário e torna o processo de informação mais dinâmico e interessante.

Dessa forma, a criação de um sistema de portal de notícias surge como uma oportunidade para oferecer um produto de qualidade e relevância para os usuários. Com um design intuitivo e uma interface amigável, é possível criar uma experiência agradável para o usuário, mantendo-o atualizado e informado sobre os mais diversos assuntos do instituto.

Por fim, um sistema de portal de notícias também pode ter um papel importante na democracia, garantindo a livre circulação de informações e possibilitando o acesso a uma grande variedade de opiniões e pontos de vista sobre os mais diversos assuntos. Assim, a criação de um portal de notícias pode contribuir para a construção de uma sociedade mais informada, crítica e participativa.

1. Requisitos Funcionais

1.1 Sistema de Autenticação: O portal de notícias deve possuir um sistema de autenticação de usuários que permita a criação de contas de usuários



com e-mail, senha e dados pessoais básicos. Esse sistema deve ser capaz de realizar as seguintes funções:

- Criação de contas de usuários: O sistema deve permitir que os usuários possam criar suas próprias contas com informações básicas, como nome completo, e-mail e senha.
- Recuperação de senha: O sistema deve permitir que os usuários possam recuperar suas senhas em caso de esquecimento ou perda.
- Validação de e-mail: O sistema deve validar o endereço de e-mail fornecido pelos usuários para garantir que ele seja válido e para prevenir o registro de contas falsas ou spams.
- Atualização de perfil: O sistema deve permitir que os usuários possam atualizar suas informações de perfil, como nome, e-mail e senha.
- Remoção de conta: O sistema deve permitir que os usuários possam remover suas contas, caso desejem.

1.2 Sistema de Validação de Notícias: O portal de notícias deve contar com um sistema de validação de notícias para garantir a qualidade e veracidade das informações publicadas. Esse sistema deve ser capaz de realizar as seguintes funções:

- Dupla validação de notícias: Toda notícia publicada no portal deve passar por uma dupla validação antes de ser exibida ao público. A primeira validação será realizada por um editor ou revisor do portal, que avaliará a qualidade e relevância da notícia. A segunda validação será realizada por um especialista ou fonte confiável, que avaliará a veracidade e precisão das informações da notícia.
- Sistema de aprovação: O sistema deve permitir que os editores ou revisores do portal possam aprovar ou rejeitar uma notícia. Caso a notícia seja rejeitada, o sistema deve fornecer feedback ao autor da notícia, indicando os motivos da rejeição.

Two handwritten signatures are present in the bottom right corner. One signature is in blue ink and appears to begin with 'JOSÉ'. The other is in black ink and appears to be a stylized 'Q' or 'R'.



1.3 Publicação de Notícias: O editor do portal deve ter uma interface que permita a visualização das notícias submetidas pelos usuários em uma fila de aprovação, onde ele poderá aprovar ou rejeitar cada notícia. Esse sistema deve ser capaz de realizar as seguintes funções:

- Fila de aprovação: Todas as notícias submetidas pelos usuários devem ser apresentadas em uma fila de aprovação. O editor do portal deve ter acesso a essa fila e ser capaz de visualizar todas as notícias submetidas.
- Aprovação de notícias: O editor do portal deve ter a capacidade de aprovar ou rejeitar cada notícia submetida pelos usuários. Caso ele aprove uma notícia, ela será publicada no portal. Caso ele rejeite uma notícia, ela não será publicada e o usuário que a submeteu receberá uma notificação informando a rejeição.
- Edição de notícias: Caso o editor do portal considere necessário, ele deve ter a capacidade de editar as notícias submetidas pelos usuários antes de aprová-las para publicação. Essa função deve ser restrita apenas ao editor do portal e não deve estar disponível para outros usuários.
- Histórico de aprovação: O sistema deve manter um registro do histórico de aprovação de cada notícia, incluindo a data de submissão, a data de aprovação ou rejeição, e o nome do editor responsável pela aprovação ou rejeição.
- Notificações para os usuários: O sistema deve enviar notificações aos usuários informando sobre a aprovação ou rejeição de suas notícias. Essas notificações devem ser enviadas automaticamente pelo sistema e devem incluir um link para a notícia aprovada, caso ela tenha sido aprovada.

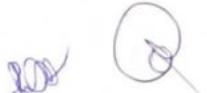
1.4 Notícias em destaque: O portal deve ter uma funcionalidade para destacar as principais notícias em uma área especial do portal. Para isso, o sistema deve ser capaz de realizar as seguintes funções:

- Seleção das notícias em destaque: O editor do portal deve ter a capacidade de selecionar as notícias que serão destacadas em uma área especial do portal. Essa seleção pode ser feita manualmente, ou pode ser baseada em critérios pré-definidos, como o número de visualizações ou o número de compartilhamentos.
- Exibição das notícias em destaque: As notícias em destaque devem ser exibidas em uma área especial do portal, que pode ser localizada na página principal ou em uma página específica. Essa área deve ser claramente identificada como uma área de destaque, e deve incluir um título, uma breve descrição e uma imagem para cada notícia destacada.
- Ordenação das notícias em destaque: As notícias em destaque devem ser ordenadas de acordo com critérios pré-definidos, como a data de publicação ou o número de visualizações. Essa ordenação pode ser configurada pelo editor do portal de acordo com suas preferências.
- Remoção das notícias em destaque: O editor do portal deve ter a capacidade de remover as notícias em destaque a qualquer momento, caso julgue necessário. Essa remoção pode ser feita manualmente, ou pode ser baseada em critérios pré-definidos, como a data de publicação.

2. Requisitos Não-Funcionais

2.1 Segurança: O sistema de autenticação deve ser seguro e criptografado para evitar ataques de hackers. Além disso, todas as notícias devem passar por uma revisão humana para evitar a propagação de notícias falsas ou inapropriadas. Para atender a esses requisitos, o sistema deve ser capaz de realizar as seguintes funções:

- Autenticação segura: O sistema de autenticação deve ser seguro e criptografado para proteger as informações pessoais dos usuários. Isso pode ser realizado utilizando técnicas como SSL/TLS e criptografia de senha.



Two handwritten signatures are present in the bottom right corner of the page. One signature is in blue ink and the other is in black ink, both enclosed in small circles.



- Revisão humana de notícias: Todas as notícias submetidas pelos usuários devem passar por uma revisão humana antes de serem publicadas no portal. Isso pode ser feito por um editor responsável pela revisão de conteúdo. O editor deve avaliar a veracidade e relevância da notícia, bem como garantir que ela esteja de acordo com as políticas editoriais do portal.
- Detecção de notícias falsas: O sistema deve ser capaz de detectar notícias falsas ou inapropriadas que possam ter sido submetidas pelos usuários. Por meio da revisão humana mencionada anteriormente.
- Controle de acesso: O sistema deve permitir que apenas usuários autorizados tenham acesso ao painel de administração do portal, onde poderão realizar funções como a revisão de notícias e a moderação de comentários.
- Monitoramento de atividade: O sistema deve ser capaz de monitorar a atividade dos usuários no portal e detectar comportamentos suspeitos, como tentativas de invasão ou de submissão de notícias falsas.
- Backup de dados: O sistema deve realizar backups regulares dos dados do portal, garantindo que as informações estejam seguras e protegidas em caso de falhas no sistema ou ataques de hackers.

2.2 Performance: O portal deve ter uma performance rápida para permitir a navegação dos usuários sem lentidão.

- A performance do portal de notícias diz respeito à velocidade e eficiência do site. Isso significa que o portal deve ser projetado e construído para oferecer um alto desempenho, incluindo tempos de carregamento rápidos e navegação fluida entre as páginas.
- Para atender a esse requisito, é importante considerar vários fatores, como a escolha da tecnologia de desenvolvimento, a otimização de imagens e arquivos de mídia, a escolha do servidor de hospedagem e o uso de cache de página. Além disso, é importante realizar testes de desempenho periódicos para garantir que o portal esteja funcionando de forma eficiente.



- Uma boa performance é importante para garantir a satisfação do usuário e reduzir a taxa de rejeição, mantendo os usuários engajados com o conteúdo do portal de notícias. Por isso, seja levado em consideração desde o início do processo de desenvolvimento e que a equipe responsável esteja comprometida em garantir uma experiência de usuário satisfatória.

2.3 Usabilidade: O portal deve ser intuitivo e fácil de usar para os usuários, com uma interface limpa e organizada.

- Isso significa que a interface do usuário deve ser limpa, organizada e fácil de entender, com botões e menus claramente identificados e dispostos em uma lógica coerente. Além disso, é importante garantir que o portal seja responsivo e possa ser acessado de qualquer dispositivo, como computadores, tablets e smartphones, sem prejudicar a usabilidade.
- Para atender a esse requisito, é importante considerar as melhores práticas de design de interface do usuário e realizar testes de usabilidade com usuários reais para identificar possíveis problemas e melhorias a serem implementadas. Também é importante fornecer feedback e suporte ao usuário, caso seja necessário.
- Um portal de notícias com uma boa usabilidade pode ajudar a aumentar a satisfação do usuário, a retenção e o engajamento, além de reduzir a taxa de rejeição e melhorar a imagem do portal perante o público. Por isso, é fundamental que o requisito funcional seja considerado desde o início do processo de desenvolvimento e integrado em todas as fases do projeto.

2.4 Escalabilidade: O sistema deve ser escalável para atender um grande número de usuários sem comprometer a performance.

- Isso significa que o sistema deve ser projetado e implementado de forma escalável, utilizando tecnologias e arquiteturas que permitam aumentar a capacidade do sistema de forma horizontal ou vertical, conforme a



demandas de usuários aumenta. É importante também considerar a utilização de recursos de computação em nuvem para permitir a escalabilidade automática e o gerenciamento eficiente de recursos.

- Ao garantir que o portal de notícias seja escalável, é possível evitar que o sistema fique sobrecarregado durante períodos de pico de tráfego, o que pode levar a erros, lentidão ou até mesmo falhas no sistema. Além disso, a escalabilidade permite que o portal possa crescer de forma sustentável, atendendo a uma base crescente de usuários sem comprometer a qualidade do serviço.
- Para atender a esse requisito, é importante considerar as melhores práticas de arquitetura de sistemas escaláveis e realizar testes de carga para identificar possíveis gargalos ou limitações no sistema. Também é importante monitorar a performance do sistema constantemente e realizar ajustes conforme necessário para garantir que a escalabilidade seja mantida.

• 3 - Requisitos de Conteúdo

3.1 Notícias Atualizadas: O portal deve publicar notícias atualizadas e relevantes para seus usuários.

- Para detalhar melhor o requisito funcional é importante considerar que o portal de notícias deve manter os usuários sempre informados sobre os acontecimentos mais recentes e relevantes. Para isso, é necessário que a equipe esteja constantemente em busca de novas informações e atualizando as notícias já publicadas.
- Além disso, o portal deve ter uma abrangência ampla em sua cobertura, abordando diferentes temas e assuntos de interesse público, para atender a diversidade de interesses dos usuários.
- Para garantir a qualidade da informação, é importante que as notícias sejam verificadas por fontes confiáveis e tenham uma linguagem clara e objetiva, evitando sensacionalismo ou distorções dos fatos.

100%



- Por fim, é essencial que o portal tenha uma política clara de ética e transparência, para garantir a credibilidade do veículo e a confiança dos usuários na informação fornecida.

3.2 Veracidade das informações: Todas as notícias publicadas no portal devem ser verificadas e possuir fontes confiáveis antes de serem publicadas.

- O requisito funcional é crucial para a credibilidade e confiança do portal de notícias. Para atender a esse requisito, é importante que a equipe editorial do portal tenha um processo rigoroso de verificação de fatos e fontes antes de publicar qualquer notícia.
- A verificação deve envolver a confirmação de informações com pelo menos duas fontes confiáveis e independentes, além de consultar especialistas no assunto, quando necessário. É importante que a equipe editorial esteja atenta a notícias falsas e desinformação, e seja capaz de identificá-las e excluí-las rapidamente do portal.

3.3 Variedade de conteúdo: O portal deve publicar notícias em diversas áreas, como política, economia, esportes, cultura, entre outras.

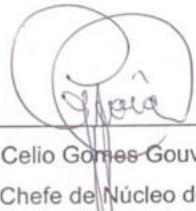
- Variedade de conteúdo, específica que o portal de notícias deve oferecer aos seus usuários uma ampla variedade de conteúdos noticiosos em diferentes áreas temáticas, de forma a atender a diferentes interesses e necessidades.
- Dessa forma, o portal deve contar com uma equipe de redatores e editores de conhecimento de forma a fornecer conteúdo variado e atualizado para seus usuários.
- Ademais, o portal deve ser capaz de apresentar o conteúdo de forma organizada e categorizada para facilitar a navegação e a pesquisa de notícias pelos usuários. Por exemplo, pode-se criar uma seção específica para cada área temática, com subcategorias para assuntos específicos dentro de cada área.

PPV
Q

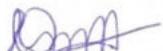


Por fim, para garantir a qualidade e a veracidade do conteúdo, é importante que a equipe de redação e edição do portal seja composta por profissionais qualificados e experientes em suas respectivas áreas de atuação. Também é necessário ter um processo de revisão de conteúdo eficiente e rigoroso para garantir a qualidade do material publicado.

Com base nesses requisitos, o portal de notícias com dupla validação poderá ser desenvolvido para atender às necessidades dos usuários e garantir a qualidade e veracidade das informações publicadas.



Celio Gomes Gouveia
Chefe de Núcleo de TI



Fernanda Dias Machado Zerbini Leão
Diretora

ANEXO V

Não seguro | previpalmas.to.gov.br



Notícias **Serviços** **Sobre**

Previpalmas suspende atendimento presencial e adota novas medidas de proteção aos servidores
Processos de solicitação de concessão e revisão de benefícios previdenciários devem ser enviados

Covid-19: Previpalmas adota medidas de prevenção e combate à pandemia do novo coronavírus
Medidas garantem mais segurança para servidores e cidadãos
Em virtude do número de casos da Covid-19 confirmados

Atendimento do Previpalmas é alterado e serviços podem ser solicitados virtualmente
Servidores podem acompanhar online os andamentos dos processos, consultar contracheques, informação de rendimentos, e simulações de aposentadoria

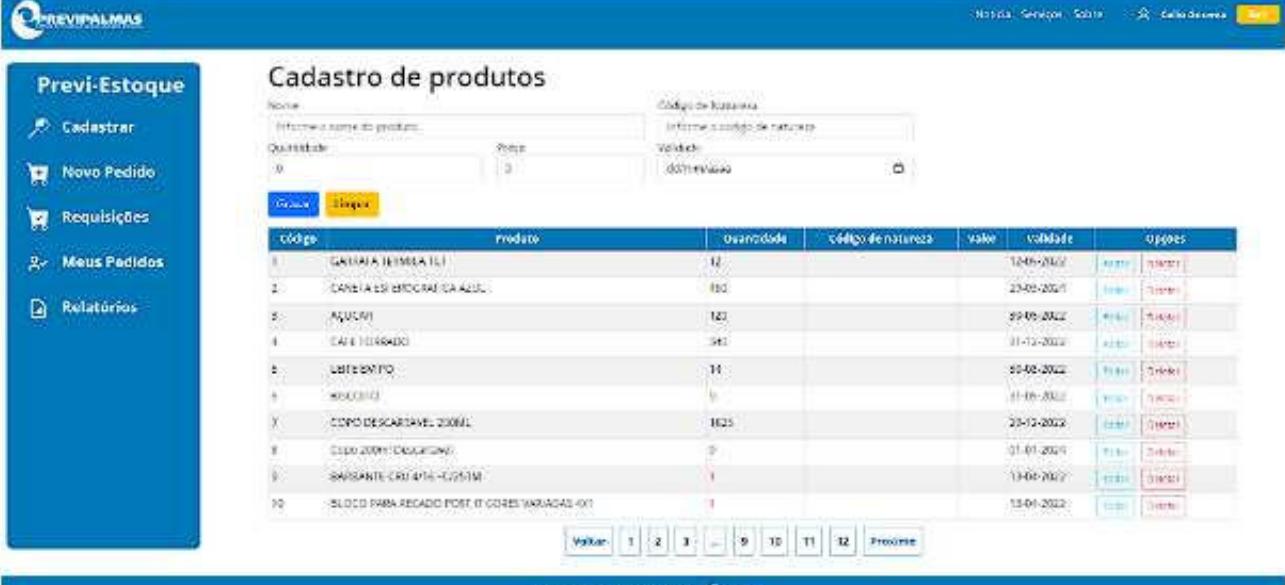
[Mais Notícias](#)

[Sisprev](#) [Portal do Segurado](#) [Portal da Transparéncia](#)

© GTI - PREVIPALMAS 2019 - 2023 | 

Portal de Notícias

Fonte: Imagem do Autor



Previ-Estoque

- [Cadastrar](#)
- [Novo Pedido](#)
- [Requisições](#)
- [Meus Pedidos](#)
- [Relatórios](#)

Cadastro de produtos

Nome	Código de Barras
Informe o nome do produto.	Informe o código de barras.
Quantidade	Válida
0	0000000000000000

Lista de Produtos

Código	Produto	Quantidade	Código de natureza	Validade	Opções
1	GARRAFA TERMICA 1L	12	12-05-2022	12-05-2022	Detalhar
2	CANETA A ESTOQUE PLASTICA AZUL	180	23-05-2022	23-05-2022	Detalhar
3	AGUARU	120	22-05-2022	22-05-2022	Detalhar
4	CADERNO BRANCO	960	31-12-2022	31-12-2022	Detalhar
5	LIVRO DE FOLHOS	11	20-05-2022	20-05-2022	Detalhar
6	MOUSE	0	31-05-2022	31-05-2022	Detalhar
7	CD/DVD DESCARTAVEL 200ML	1625	23-12-2022	23-12-2022	Detalhar
8	CD/DVD 200MB DIGITALIZADO	0	31-01-2023	31-01-2023	Detalhar
9	SABONETE CRISTAL C/250ML	1	13-06-2022	13-06-2022	Detalhar
10	BLOCO PARA RECADOS POST IT CORES VARIADAS 4X1	1	13-01-2022	13-01-2022	Detalhar

[Voltar](#) [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [11](#) [12](#) [Próximo](#)

© GTI - PREVIPALMAS 2019 - 2022 | 

Sistema de Estoque

Fonte: Imagem do Autor