

PÓS-GRADUAÇÃO EM DESENVOLVIMENTO FULL STACK

MEMORIAL

KLEBER SILVA DE ARAUJO

Orientador: Prof. Alexandre Agustini

2025

SUMÁRIO

1. TRAJETÓRIA PESSOAL.....	3
1.1. História de Vida	3
1.2. Área(s) Selecionada(s).....	4
2. ORGANIZAÇÃO DO PORTFÓLIO.....	5
2.1. Fundamentos de Computação e Algoritmos	5
2.2. Programação Orientada a Objetos	5
2.3. DevOps.....	5
2.4. Banco de Dados NoSQL	6
2.5. MicroServiços.....	6
2.6. Programação para Web	6
2.7. Client-Side	7
2.8. Server-Side	7
2.9. WebServices.....	7
3. LIÇÕES APRENDIDAS DURANTE A FORMAÇÃO.....	9
4. COMPETÊNCIAS E EXPECTATIVAS	10
REFERÊNCIAS.....	12

1. TRAJETÓRIA PESSOAL

Primeiramente, gostaria de expressar minha profunda gratidão à minha esposa, Patricia Ribeiro de Araujo, pelo seu amor incondicional, paciência e apoio contínuo ao longo desta jornada. Sem sua compreensão e encorajamento, este sonho não teria se tornado realidade.

Aos meus filhos Nigel Ribeiro de Araujo e Liam Ribeiro de Araujo, vocês são a minha inspiração diária. Seu entusiasmo e alegria de viver me motivaram a continuar, mesmo nos momentos mais difíceis. Este sucesso é tanto meu quanto de vocês.

Minha trajetória pessoal é apresentada em duas partes: (1.1) minha história de vida e (1.2) as áreas selecionadas. Em minha história de vida detalho minha vida pregressa a realização deste curso, bem como os motivos que me trouxeram até ele e as contribuições técnicas e comportamentais que a realização deste curso me ofereceu. Ao tratar sobre minha(s) área(s) de interesse, darei enfoque à algumas tecnologias que me chamaram a atenção e que entendo serem importantes na construção de minha vida profissional.

1.1. História de Vida

Minha trajetória na área de Tecnologia da Informação (TI) começou em 2012, quando ingressei no curso de Análise e Desenvolvimento de Sistemas na Universidade Federal do Paraná (UFPR). Desde o início, fiquei fascinado pelas possibilidades da área e rapidamente consegui um estágio em programação focado no sistema SAP, utilizando a linguagem proprietária ABAP (Advanced Business Application Programming), desenvolvida pela própria SAP.

Desde então, concentrei meu desenvolvimento profissional nessa área. Trabalhando com sistema ERP (Enterprise Resource Planning) da SAP, especializei-me tanto na parte técnica do desenvolvimento quanto nos processos relacionados, como Finanças, Cadeia de Suprimentos, Gerenciamento de Materiais e Recursos Humanos. Ao mesmo tempo, continuei estudando outras linguagens de programação, como JavaScript, Node.js, C# e C++, para criar pequenos sites, aplicações do dia a dia e jogos digitais.

Com o tempo, percebi que as tecnologias estavam avançando rapidamente e eu não conseguia acompanhar esse ritmo. Isso me motivou a realizar uma pós-graduação em desenvolvimento Full Stack, com o objetivo de me atualizar sobre as práticas mais atuais no ciclo de desenvolvimento de aplicações web em cloud. Além disso, busquei certificações em Google Cloud e Amazon Web Services.

Esse curso me proporcionou uma atualização significativa em minha base de conhecimento, reforçando aprendizados anteriores e incorporando conteúdos novos que transformaram minha visão sobre o mundo profissional.

1.2. Área(s) Selecionada(s)

Ao iniciar este curso, minha principal pretensão era aprimorar meu conhecimento técnico em desenvolvimento web, especialmente nas linguagens de programação JavaScript, TypeScript e Node.js, além de obter uma compreensão mais profunda de todo o ciclo de DevOps e CI/CD.

No entanto, fui surpreendido positivamente ao me deparar com as disciplinas de Design Thinking, Inovação, Lean Startup e Mentalidade Ágil. Essas áreas, que anteriormente eram um tanto negligenciadas em minha formação como desenvolvedor, proporcionaram uma nova perspectiva e me levaram a repensar o desenvolvimento de sistemas como produtos e serviços dinâmicos e em constante evolução.

Dessa forma, este trabalho de conclusão de curso, que vai além do simples desenvolvimento de um sistema, tem como objetivo criar um produto viável e robusto, voltado para a utilização comercial. A experiência adquirida nas disciplinas mencionadas me permitiu integrar conceitos de inovação e agilidade ao processo de desenvolvimento, garantindo que o resultado final não apenas atenda a requisitos técnicos, mas também ofereça valor real aos usuários e ao mercado.

2. ORGANIZAÇÃO DO PORTFÓLIO

Nesta seção apresentarei um conjunto de soluções elaboradas durante a realização deste curso e que foram fundamentais no entendimento do desenvolvimento Full Stack.

As soluções são apresentadas considerando a linha do tempo de realização das disciplinas:

2.1. Fundamentos de Computação e Algoritmos

Listado no repositório do GitHub [devfullstack/FundamentosDeComputacaoEAlgoritmos](https://github.com/devfullstack/FundamentosDeComputacaoEAlgoritmos) at main · kleber-s-araujo/devfullstack este conjunto de solução aborda alguns dos conceitos básicos de programação utilizando a linguagem de programação JavaScript em tela via HTML. Nos exemplos 01, 02 e 03 é abordado a realização de leitura de campos em tela e em seguida aplicamos uma pequena lógica/matemática para posterior exibição do resultado em tela. Já para o exercício 04 são abordados os conceitos de básicos de programação e manipulação de dados como declaração de variáveis, condições e loops de repetição.

2.2. Programação Orientada a Objetos

Listado no repositório do GitHub [devfullstack/ProgramacaoOrientadaAOobjetos](https://github.com/devfullstack/ProgramacaoOrientadaAOobjetos) at main · kleber-s-araujo/devfullstack, este conjunto de solução detalha um pouco mais sobre as bases de programação na linguagem JavaScript, neste repositório é possível entender a utilização e os conceitos das variáveis e arrays, bem como a utilização de classes, funções, objetos, estruturas JSON, exceptions e assincronicidade.

2.3. DevOps

Listado no repositório do GitHub [devfullstack/DevOps](https://github.com/devfullstack/DevOps) at main · kleber-s-araujo/devfullstack, esta aplicação de conversão de temperatura Celsius/Fahrenheit foi utilizada como base de uma implementação de pipeline

de CI e a utilização do Docker como empacotador da aplicação. Este repositório também contém toda informação desde a concepção da aplicação até a execução dos testes e deploy automatizados.

2.4. Banco de Dados NoSQL

Listado no repositório do GitHub [devfullstack/NoSQL at main · kleber-s-araujo/devfullstack](https://github.com/devfullstack/NoSQL_at_main_kleber-s-araujo/devfullstack), este repositório detalha a utilização e o melhor cenário a ser aplicado de alguns banco de dados NoSQL como Cassandra, Redis e Neo4j. Podemos verificar, por exemplo, que o Redis por ser um banco de dados em memória, tem o acesso muito mais rápido, sendo ideal para aplicações com muitos acessos diários, por exemplo a busca de livros na Amazon. Já o Neo4J, é um banco de dados baseado em grafos, sendo muito performático na resolução de problemas de performance em relacionamentos. Podemos verificar as demos de bancos disponibilizadas para consultas de teste do Neo4j online, utilizando os comandos deste repositório.

2.5. MicroServiços

Listado no repositório [devfullstack/Microservices at main · kleber-s-araujo/devfullstack](https://github.com/devfullstack/Microservices_at_main_kleber-s-araujo/devfullstack), esta solução exemplificada pelo professor em aula utiliza o protocolo AMQP para implementação de um Message Broker com o RabbitMQ, TypeScript e Docker. Nesta solução temos duas implementações de serviços: Produtor e Consumidor, sendo produtor o serviço que enviará as mensagens para o Broker e o consumidor que fará a leitura destas mensagens. Estas mensagens podem ser lidas de duas formas: Fila ou por Tópicos. A solução é executada via terminal e pode ser executada diversas vezes.

2.6. Programação para Web

Listado no repositório [devfullstack/ProgramacaoWeb at main · kleber-s-araujo/devfullstack](https://github.com/devfullstack/ProgramacaoWeb_at_main_kleber-s-araujo/devfullstack), este conjunto de soluções aborda alguns conceitos do desenvolvimento web frontend voltados às interações com a aplicação, como por exemplo: animações, alterações entre temas escuro e claro, contraste, como trabalhar com layouts responsivos (abordagem de design que visa criar

páginas que se adaptam a diferentes tamanhos de tela e dispositivos) e outros componentes.

Também temos exemplos de Single Page Application (SPA), Multi-Page Application (MPA) e Progressive Web App (PWA), este por sua vez combina as características de SPA e MPA, podendo também ser instalado no dispositivo do usuário.

2.7. Client-Side

Listado no repositório [devfullstack/ClientSide at main · kleber-s-araujo/devfullstack](https://github.com/kleber-s-araujo/devfullstack), este conjunto de soluções aborda alguns conceitos de arquitetura cliente-side como a utilização de componentes React, Eventos, Forms, utilização de Hooks `useState`, `useEffect` e `useReducer`, além exemplificar a utilização de frameworks como o NextJS e Vite. Também podemos observar um exemplo de arquitetura clean de um repositório público utilizando NextJS e TypeScript.

2.8. Server-Side

Listado no repositório [devfullstack/ServerSide at main · kleber-s-araujo/devfullstack](https://github.com/kleber-s-araujo/devfullstack), este conjunto de soluções aborda os conceitos principais do desenvolvimento de aplicações server side. Nestas soluções de exemplo, podemos observar como inicializar servidores utilizando as bibliotecas HTTP e Express, a partir desse ponto de partida, as soluções demonstram a aplicação do padrão de programação MVC (Model-View-Controller) para rotear e manipular as solicitações. Esse padrão facilita o redirecionamento das requisições para classes controller que serão responsáveis pelo processamento da requisição ou acesso ao banco de dados.

2.9. WebServices

No repositório [devfullstack/WebServices at main · kleber-s-araujo/devfullstack](https://github.com/kleber-s-araujo/devfullstack), encontra-se um conjunto de soluções que explora as principais abordagens para trabalhar com web services. Nestas soluções práticas, utilizamos alguns exemplos de SOAP (Simple Object Access Protocol) e REST (Representational State Transfer), permitindo uma comparação clara e didática entre essas duas tecnologias. Ao longo

das soluções apresentadas, destacamos não apenas as características distintas de cada abordagem, mas também as melhores práticas para sua aplicação, proporcionando uma compreensão mais profunda e coesa sobre o desenvolvimento e a integração de serviços web.

Também é exemplificado a utilização dos serviços da Amazon Web Services que executa uma API Lambda para manipulação de tabelas no DynamoDB.

3. LIÇÕES APRENDIDAS DURANTE A FORMAÇÃO

Durante a realização do curso de pós-graduação em Desenvolvimento Full Stack, fui exposto a muitos conteúdos que acrescentaram na minha visão sobre o mundo de TI para além das habilidades técnicas. Matérias como Design Thinking, Lean Startup e MVP, Inovação e Empreendedorismo por exemplo, me mostraram que o desenvolvimento de uma boa aplicação de sucesso vai muito além de ser apenas um bom programador com boas habilidades analíticas. Desta forma, algumas lições aprendidas durante a formação que considero muito importantes foram: Entendimento de diferentes perspectivas visando encontrar uma solução que satisfaça todos os envolvidos de uma forma colaborativa, gestão de tempo e priorização de tarefas com base em diversos fatores como jornada do usuário, brainstorm de funcionalidades e revisões técnicas entre outras coisas que foram abordados na matéria de Mentalidade Ágil, resiliência para adaptar-se rapidamente às mudanças e buscar soluções criativas.

Claro, habilidades técnicas são sim muito importantes, e entregar um produto de qualidade não é possível sem elas, durante o curso pude compreender melhor algumas das melhores práticas de programação e como aplicá-las com as tecnologias atuais do mercado. A utilização de frameworks como o React, Vite e NextJS facilitam muito a vida do programador automatizando algumas tarefas básicas e padronizando a forma como a aplicação vai ser desenvolvida.

4. COMPETÊNCIAS E EXPECTATIVAS

Ao longo do curso, passei por várias etapas de crescimento pessoal e profissional. A seguir elaboro sobre algumas percepções e experiências até este momento:

No início da trajetória, meus conhecimentos sobre o desenvolvimento de aplicações web end-to-end eram básicos e incompletos, mesmo sabendo como desenvolver websites, não tinha a confiança necessária em como realizar deploy de aplicações maiores ou mesmo a utilização de pipelines de CI/CD ou a utilização de Containers Docker e Kubernetes.

Ao longo do curso, as disciplinas abordadas desempenharam um papel fundamental no preenchimento das lacunas de conhecimento e na consolidação do aprendizado. Esse processo contínuo contribuiu significativamente para o meu desenvolvimento técnico, permitindo-me dominar linguagens essenciais como JavaScript, TypeScript e NodeJS. Além disso, adquiri uma compreensão aprofundada sobre o funcionamento de frameworks populares como React e Vite, bem como sobre a criação e implementação de arquiteturas de microsserviços, que se mostraram cruciais em minha formação.

Com a aquisição dessas novas habilidades e a maturidade técnica, a implementação do projeto final, que simula um ambiente produtivo real, tornou-se uma tarefa mais acessível. A posse do conhecimento necessário para conduzir todo o desenvolvimento de ponta a ponta facilitou uma entrega mais fluida e direta, possibilitando que eu aplicasse de forma prática tudo o que havia aprendido ao longo do curso. Dessa forma, pude transformar ideias em soluções concretas, preparando-me de maneira eficaz para qualquer desafio.

O contínuo aprimoramento técnico é uma necessidade para qualquer profissional de TI, todos os dias novas ferramentas surgem no mercado ou se atualizam. O ritmo acelerado das inovações exige que os desenvolvedores estejam sempre aprendendo e se adaptando. Além disso, a prática constante por meio de projetos pessoais e colaborativos, contribui não apenas para o desenvolvimento de

habilidades técnicas, mas também para a construção de um portfólio robusto que pode ser um diferencial na hora de buscar oportunidades.

Reconheço o curso como essencial na minha formação, aulas bem estruturadas com dois professores não apenas fornecem conhecimento técnico, mas também ajudam a desenvolver habilidades críticas de diferentes perspectivas. Portanto, o objetivo de atuar como desenvolvedor Full Stack vai além da simples execução de tarefas, trata-se de uma jornada contínua de aprendizado e aprimoramento que pode levar a uma carreira gratificante e impactante.

REFERÊNCIAS

Tecnologia em Análise e Desenvolvimento de Sistemas. Disponível em: <<http://www.sept.ufpr.br/portal/analisesistemas/>>.

JavaScript. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>>.

MICROSOFT. TypeScript - JavaScript that scales. Disponível em: <<https://www.typescriptlang.org/>>.

NODE.JS. Node.js. Disponível em: <<https://nodejs.org>>.

SAP Brasil | Software e Soluções Inteligentes na Nuvem. Disponível em: <https://www.sap.com/brazil/index.html?url_id=auto_hp_redirect_brazil>.

Evolution of ABAP. Disponível em: <https://learning.sap.com/learning-journeys/acquire-core-abap-skills/understanding-the-basics-of-abap_c0e5346f-a136-4b9f-a167-9031eca12932>.

GITHUB. Build software better, together. Disponível em: <<https://github.com>>.

React – Uma biblioteca JavaScript para criar interfaces de usuário. Disponível em: <<https://pt-br.legacy.reactjs.org/>>.

Vite. Disponível em: <<https://vite.dev/>>.

VERCEL. Next.js by Vercel - The React Framework. Disponível em: <<https://nextjs.org/>>.

Cloud Computing Services. Disponível em: <https://cloud.google.com/?_gl=1>.

Serviços de computação em nuvem - Amazon Web Services (AWS). Disponível em: <<https://aws.amazon.com/pt/>>.

Banco de dados de chave-valor NoSQL rápido – Amazon DynamoDB – Amazon Web Services. Disponível em: <https://aws.amazon.com/pt/dynamodb/?nc2=type_a>. Acesso em: 8 fev. 2025.

Amazon Lambda - AWS. Disponível em: <<https://aws.amazon.com/pt/lambda/>>.

RABBITMQ. Messaging that just works — RabbitMQ. Disponível em: <<https://www.rabbitmq.com/>>.

