

**FACULDADE DE TECNOLOGIA DE SÃO JOSÉ DOS CAMPOS
FATEC PROFESSOR JESSEN VIDAL**

João Victor Menezes Oliveira

**PORTFÓLIO DE APRENDIZAGEM DE
PROJETOS INTEGRADO (API) E
RuleAI**

São José dos Campos
2026

João Victor Menezes Oliveira

**PORTFÓLIO DE APRENDIZAGEM DE
PROJETOS INTEGRADO (API) E
RuleAI**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de São José dos Campos, como parte dos requisitos necessários para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Gerson da Penha Neto

São José dos Campos
2026

João Victor Menezes Oliveira

**PORTFÓLIO DE APRENDIZAGEM DE
PROJETOS INTEGRADO (API) E
RuleAI**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de São José dos Campos, como parte dos requisitos necessários para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Gerson da Penha Neto - FATEC

Professor, Fernando Masanori - FATEC

Professor, Emanuel Mineda – FATEC

Professor, Giuliano Bertoti - FATEC

15/06/2026

DATA DA APROVAÇÃO

AGRADECIMENTOS

Gostaria de expressar meus sinceros agradecimentos a todos os professores que contribuíram para minha formação ao longo desta jornada acadêmica. Cada ensinamento, orientação e apoio recebido foi fundamental para a construção do conhecimento adquirido, desde os conceitos iniciais da área de Tecnologia da Informação até os conteúdos mais complexos e aprofundados que fizeram parte da minha trajetória no curso.

Agradeço também à minha família, pelo apoio, incentivo e compreensão nos momentos de dificuldade, cansaço e incerteza. Sua presença foi essencial para que fosse possível superar os desafios encontrados durante esse percurso e continuar avançando mesmo diante dos obstáculos.

Aos amigos e colegas que estiveram presentes ao longo dessa caminhada, deixo meu reconhecimento pela parceria, pelas trocas de conhecimento, pelas experiências compartilhadas e pelo apoio nos momentos em que as dificuldades surgiram. Cada convivência contribuiu, de alguma forma, para meu crescimento pessoal, acadêmico e profissional.

Encerra-se, assim, um ciclo importante, marcado por aprendizados, desafios, conquistas, tropeços e amizades construídas ao longo do caminho. Levo comigo não apenas o conhecimento adquirido, mas também as experiências e os ensinamentos que contribuíram para a pessoa e o profissional que me torno a partir desta etapa.

RESUMO

Este trabalho apresenta a trajetória de desenvolvimento dos Projetos de Aprendizagem por Projeto Integrador realizados ao longo do curso de Análise e Desenvolvimento de Sistemas. O documento tem como objetivo descrever os projetos desenvolvidos em cada semestre, evidenciando o contexto das empresas parceiras, os problemas identificados, as soluções propostas, as tecnologias utilizadas e as contribuições realizadas durante a formação acadêmica. A metodologia adotada baseia-se na análise documental dos repositórios dos projetos, na descrição das funcionalidades implementadas e na reflexão sobre a aplicação de práticas ágeis, especialmente relacionadas ao Scrum, ao levantamento de requisitos, à organização de backlog, ao desenvolvimento incremental e à validação das entregas. Ao longo da graduação, o autor participou de projetos voltados ao ensino de Scrum, consulta a banco de dados por linguagem natural, gestão de pesquisas de clima organizacional, gerenciamento de projetos, endereçamento digital rural e criação de regras de negócio com inteligência artificial generativa. Os resultados demonstram a evolução progressiva das competências técnicas e profissionais, com aplicação de tecnologias como HTML, CSS, JavaScript, TypeScript, Java, Python, React, React Native, Vue.js, Node.js, Spring Boot, FastAPI, bancos de dados relacionais e não relacionais, Firebase, serviços em nuvem e ferramentas de inteligência artificial. Conclui-se que os projetos integradores contribuíram de forma significativa para a consolidação dos conhecimentos teóricos e práticos, para o desenvolvimento da autonomia, da comunicação, do trabalho em equipe e da capacidade de resolução de problemas, preparando o aluno para atuar em contextos reais do mercado de tecnologia.

Palavras-Chave: Projetos integradores; Desenvolvimento de software; Metodologias ágeis; Inteligência artificial; Formação profissional.

ABSTRACT

This work presents the development trajectory of the Project-Based Integrated Learning projects carried out throughout the Systems Analysis and Development course. The document aims to describe the projects developed in each semester, highlighting the context of the partner companies, the identified problems, the proposed solutions, the technologies used, and the contributions made during the academic education process. The methodology is based on documentary analysis of the project repositories, description of the implemented features, and reflection on the application of agile practices, especially those related to Scrum, requirements gathering, backlog organization, incremental development, and delivery validation. Throughout the course, the author participated in projects focused on teaching Scrum, querying databases through natural language, managing organizational climate surveys, managing projects, creating rural digital addressing solutions, and developing business rules with generative artificial intelligence. The results demonstrate the progressive evolution of technical and professional skills, with the application of technologies such as HTML, CSS, JavaScript, TypeScript, Java, Python, React, React Native, Vue.js, Node.js, Spring Boot, FastAPI, relational and non-relational databases, Firebase, cloud services, and artificial intelligence tools. It is concluded that the integrated projects contributed significantly to the consolidation of theoretical and practical knowledge, as well as to the development of autonomy, communication, teamwork, and problem-solving skills, preparing the student to work in real contexts of the technology market.

Keywords: Integrated projects; Software development; Agile methodologies; Artificial intelligence; Professional education.

SUMÁRIO

1. INTRODUÇÃO	8
2. PROJETOS	11
3. CONSIDERAÇÕES FINAIS.....	25
REFERÊNCIAS BIBLIOGRÁFICAS	28
APÊNDICE A - PORTIFÓLIO	29

1. INTRODUÇÃO

O curso de Análise e Desenvolvimento de Sistemas da Fatec São José dos Campos possui, em sua estrutura curricular, o desenvolvimento de Projetos Integradores denominados API — Aprendizagem por Projeto Integrador. Esses projetos são realizados ao longo dos semestres e têm como finalidade aproximar os conteúdos teóricos das disciplinas às demandas práticas do mercado, por meio da resolução de problemas reais apresentados por empresas parceiras ou por setores vinculados à própria instituição. Nesse contexto, o presente trabalho apresenta uma síntese da trajetória acadêmica construída ao longo dos seis semestres do curso, evidenciando os projetos desenvolvidos, os problemas enfrentados, as tecnologias utilizadas e os resultados obtidos.

Ao longo da graduação, foram desenvolvidas soluções com diferentes níveis de complexidade, abrangendo aplicações web, aplicações desktop, sistemas mobile, plataformas de gestão, dashboards, integrações com banco de dados, recursos de geolocalização e soluções baseadas em inteligência artificial. Cada projeto possibilitou a aplicação prática de conhecimentos relacionados a engenharia de software, banco de dados, desenvolvimento web, desenvolvimento mobile, experiência do usuário, metodologias ágeis, computação em nuvem e inteligência artificial. Dessa forma, os projetos de API contribuíram para a consolidação progressiva das competências técnicas e comportamentais necessárias à formação profissional na área de tecnologia.

A problematização central observada ao longo dos projetos esteve relacionada à necessidade de transformar demandas reais em soluções digitais funcionais, organizadas e utilizáveis. Em cada semestre, a equipe precisou compreender o problema apresentado pelo cliente, levantar requisitos, planejar entregas, definir prioridades, desenvolver funcionalidades, realizar testes e apresentar resultados. Esse processo evidenciou desafios recorrentes do desenvolvimento de software, como comunicação com stakeholders, clareza na definição de requisitos, organização do backlog, integração entre frontend e backend, modelagem de dados, autenticação de usuários, deploy de aplicações, manutenção do código e adaptação a mudanças durante o desenvolvimento.

A relevância dos projetos de API está diretamente associada à sua capacidade de integrar teoria e prática. Por meio dessas atividades, tornou-se possível aplicar conceitos estudados em sala de aula em situações próximas à realidade profissional, permitindo maior compreensão sobre o ciclo de vida de desenvolvimento de software. Além disso, os projetos favoreceram o desenvolvimento de habilidades relacionadas ao trabalho em equipe, à

comunicação, à gestão de tempo, à resolução de problemas e à tomada de decisão. A vivência com papéis do Scrum, como Scrum Master, Product Owner e membro do Development Team, também contribuiu para a compreensão das responsabilidades existentes em equipes ágeis.

No primeiro semestre, foi desenvolvido o projeto Pandora Box, uma aplicação web educativa voltada ao ensino da metodologia Scrum. A solução buscou facilitar o aprendizado de colaboradores sobre conceitos ágeis, utilizando recursos visuais e interativos. No segundo semestre, o projeto JPT — Já Pode Trancar consistiu em uma aplicação desktop capaz de converter comandos em linguagem natural em consultas SQL, permitindo que usuários sem conhecimento técnico pudessem acessar informações em banco de dados. No terceiro semestre, o projeto OAK-RH teve como proposta uma aplicação web para gestão de pesquisas de clima, cultura e feedback organizacional, com foco em Recursos Humanos.

No quarto semestre, desenvolveu-se o projeto Lumen, uma plataforma web para gestão de projetos, etapas, tarefas, usuários e responsabilidades, em parceria com a FAPG. No quinto semestre, o projeto GeoMaps, realizado em parceria com a Visiona, teve como objetivo criar uma solução mobile e web para endereçamento digital de propriedades rurais vinculadas ao Cadastro Ambiental Rural, utilizando recursos de geolocalização, mapas, rotas, Plus Code e portal administrativo. No sexto semestre, o projeto RuleAI, desenvolvido em parceria com a Dom Rock, propôs uma aplicação web com inteligência artificial generativa para criação, gerenciamento e controle de regras de negócio por meio de linguagem natural.

As tecnologias utilizadas ao longo da graduação variaram conforme os objetivos de cada projeto. Foram empregados recursos como HTML, CSS, JavaScript, TypeScript, Python, Java, React, React Native, Vue.js, Node.js, Spring Boot, FastAPI, Flask, PostgreSQL, MySQL, MongoDB, Firebase, JWT, LangChain, LangGraph, RAG, Ollama, HuggingFace, Groq, Figma, Git, GitHub e serviços em nuvem, como AWS. Essa diversidade tecnológica permitiu o contato com diferentes arquiteturas, linguagens, bancos de dados, ferramentas de prototipação, serviços de autenticação, integrações externas e modelos de implantação.

Os resultados obtidos demonstram a evolução gradual da formação acadêmica e profissional. Inicialmente, os projetos permitiram o contato com fundamentos de desenvolvimento web, prototipação e metodologias ágeis. Nos semestres seguintes, houve avanço para aplicações mais complexas, envolvendo integração com bancos de dados, autenticação, dashboards, sistemas administrativos, aplicações mobile, recursos de

geolocalização e inteligência artificial. Dessa forma, a trajetória construída por meio das APIs possibilitou o amadurecimento técnico, a ampliação da capacidade de análise e a compreensão mais ampla sobre o desenvolvimento de soluções digitais em contextos reais.

Assim, este trabalho tem como objetivo apresentar os projetos desenvolvidos ao longo dos seis semestres do curso de Análise e Desenvolvimento de Sistemas, destacando o contexto de cada API, a problemática abordada, a solução proposta, as tecnologias utilizadas, o papel desempenhado pelo aluno e os principais resultados obtidos. A organização do trabalho permite evidenciar a progressão das competências desenvolvidas durante a graduação e demonstrar a importância das práticas integradoras para a preparação profissional na área de tecnologia.

2. PROJETOS

Os projetos API correspondem às atividades desenvolvidas ao longo da graduação, organizadas cronologicamente por ano e semestre. Cada projeto integrador foi elaborado a partir de demandas reais propostas por docentes e empresas parceiras, com o objetivo de aplicar, na prática, os conhecimentos adquiridos no curso. As descrições apresentam o contexto de desenvolvimento, os objetivos, as soluções implementadas e as tecnologias utilizadas, podendo ser consultadas integralmente no Apêndice A deste documento. Nas subseções a seguir, apresenta-se uma síntese de cada projeto.

Tabela 1 – Lista de Projetos API

Título do Projeto	Link do GitHub
Pandora Box	https://github.com/jvictormo/adsapi1
JPT	https://github.com/jvictormo/adsapi2
OAK-RH	https://github.com/Phoenix-Team-Fatec/OAK-RH
Lumen	https://github.com/Phoenix-Team-Fatec/API-4
Geo Maps	https://github.com/Phoenix-Team-Fatec/geo-maps
RuleAI	https://github.com/Phoenix-Team-Fatec/API_6

Fonte: Elaborado por <João Victor Menezes Oliveira>, (2026).

2.1 ANO 1 – 1º semestre

O projeto foi desenvolvido no período de agosto de 2023 a novembro de 2023, durante o primeiro semestre do curso de Análise e Desenvolvimento de Sistemas. O projeto, intitulado Pandora Box, foi realizado em parceria com a empresa Antonio Egydio, cuja demanda estava relacionada à capacitação interna de novos colaboradores na metodologia ágil Scrum. O problema identificado consistia na necessidade recorrente de reapresentar os conceitos de Scrum sempre que novos grupos de funcionários eram contratados, o que tornava o processo repetitivo, custoso e dependente da disponibilidade de instrutores. Dessa forma, a proposta do projeto foi desenvolver uma solução educativa capaz de tornar o aprendizado sobre Scrum mais rápido, intuitivo, acessível e autônomo para os colaboradores da empresa. O repositório da equipe apresenta o objetivo do produto como uma página web voltada a explicar e ensinar, de forma intuitiva e didática, a aplicação do método ágil Scrum para integrantes de diferentes níveis operacionais.

No desenvolvimento do projeto, o autor assumiu o papel de Scrum Master, ficando responsável pela coordenação da equipe, comunicação entre os membros, organização das reuniões diárias, acompanhamento do burndown chart e apoio ao engajamento do grupo. A atuação também envolveu a preparação dos slides utilizados nas apresentações ao cliente, bem como a participação ativa na exposição do andamento da aplicação e dos dados atualizados do projeto. Essa função esteve alinhada à aplicação da metodologia Scrum, que organiza o trabalho por meio de papéis, eventos, artefatos e ciclos iterativos de desenvolvimento, chamados de Sprints. De acordo com Schwaber e Sutherland (2020), o Scrum é um framework voltado à geração de valor por meio de equipes organizadas, eventos definidos e entregas incrementais. O próprio repositório da equipe registra que o projeto foi realizado com uso da metodologia Scrum, com divisão de papéis, definição do MVP, organização do Product Backlog e desenvolvimento em Sprints incrementais.

A solução desenvolvida consistiu em uma página web educativa e interativa, destinada ao ensino da metodologia ágil Scrum. A plataforma apresenta conteúdos explicativos sobre o framework, organizados de forma visual e didática, intercalando textos, imagens e recursos de interação para reduzir a fadiga do usuário durante o aprendizado. O usuário interage com o sistema por meio da navegação entre os tópicos, da avaliação do conteúdo por página e da realização de autoavaliações relacionadas aos papéis do Scrum, como Scrum Master, Product Owner e Development Team. Entre as funcionalidades previstas no backlog do produto, destacaram-se a criação de um protótipo funcional, o desenvolvimento da aplicação web, a produção e inserção dos conteúdos sobre Scrum, a implementação de formulários PACER, a criação de avaliações sobre o conteúdo e a melhoria do design das páginas.

Além das atividades relacionadas à gestão ágil, o autor contribuiu com o design completo do website no Figma, tomando como base os requisitos do cliente e os objetivos do projeto. Também houve participação no desenvolvimento das principais telas, com estruturação em HTML e aplicação de CSS. Essas contribuições envolveram aspectos de usabilidade, organização visual, responsividade e clareza na apresentação dos conteúdos, buscando tornar o ambiente de aprendizagem mais acessível e atrativo. A proposta dialoga com os princípios do Scrum, especialmente no que se refere à organização do trabalho em equipe, à adaptação contínua e à entrega incremental de valor. Sutherland (2014) destaca que o Scrum favorece a produtividade, a colaboração e a melhoria contínua dos processos, aspectos observados na condução do projeto e na organização das entregas realizadas.

As tecnologias, linguagens e ferramentas utilizadas no projeto foram Figma, para elaboração dos wireframes e protótipos de interface; HTML, para estruturação das páginas;

CSS, para estilização e responsividade; JavaScript, para adição de interatividade; Flask, para criação e gerenciamento do servidor web; e Git/GitHub, para versionamento, documentação e controle do projeto. O repositório também indica predominância de Python, além de HTML e outros recursos complementares na composição do projeto.

2.2 ANO 1 – 2º semestre

O projeto foi desenvolvido no período de janeiro de 2024 a junho de 2024, durante o segundo semestre do curso de Análise e Desenvolvimento de Sistemas. O projeto, intitulado JPT, foi realizado em parceria com Giuliano Bertoti, no contexto de uma proposta voltada à aplicação prática de banco de dados, desenvolvimento de software e inteligência artificial. O problema identificado consistia na dificuldade de colaboradores consultarem informações em bancos de dados corporativos sem conhecimento técnico em programação ou SQL, o que gerava dependência do setor de TI, lentidão nos processos internos e desvio da equipe técnica para demandas repetitivas. Dessa forma, a contribuição pretendida foi democratizar o acesso às informações, permitindo que usuários não técnicos realizassem consultas por meio de linguagem natural. O repositório da equipe descreve o objetivo do projeto como a criação de uma aplicação desktop em Java capaz de enviar comandos do usuário a um modelo de linguagem local, interpretar esses comandos, convertê-los em SQL e retornar os dados obtidos na consulta.

No desenvolvimento do projeto, o autor assumiu o papel de Scrum Master, ficando responsável pela coordenação da equipe, definição de prioridades de desenvolvimento, manutenção da comunicação entre os integrantes e apoio à entrega dentro do cronograma proposto. A atuação também envolveu participação no planejamento da arquitetura da aplicação, modelagem do banco de dados, integração do LangChain com o modelo local executado no LM Studio e integração com o banco SQL. O repositório do Trabalho de Graduação também registra contribuições relacionadas à realização de testes de desempenho, correção de inconsistências no comportamento da IA, refatoração do código e otimização do tempo de resposta da aplicação.

A solução desenvolvida consistiu em uma aplicação desktop, criada em Java, voltada à realização de consultas em banco de dados a partir de comandos escritos em linguagem natural. O usuário interage com o sistema digitando uma solicitação em texto comum; em seguida, a aplicação encaminha o comando ao modelo de linguagem local, que interpreta a intenção do usuário e gera uma consulta SQL correspondente. Após a geração do comando SQL, a aplicação executa a consulta no banco de dados e apresenta os resultados de forma

organizada ao usuário. Com isso, colaboradores de diferentes áreas poderiam acessar informações com maior autonomia, sem necessidade de conhecimento prévio em SQL.

A organização do projeto seguiu a metodologia ágil Scrum, com divisão de papéis entre Scrum Master, Product Owner e Time de Desenvolvimento, além da organização das tarefas em Sprints incrementais. O repositório da equipe apresenta a definição do MVP, a estruturação do Product Backlog e a entrega progressiva de funcionalidades, como a construção do wireframe, a conversão de linguagem natural em SQL, a criação de um segundo banco de dados para consulta, a troca entre bancos de dados e a possibilidade de alternância entre modelos de IA. Essa organização evidencia a aplicação de práticas ágeis na priorização das entregas, no acompanhamento do progresso e na validação incremental da solução.

Entre as principais funcionalidades implementadas, destacaram-se a interface gráfica desktop, a transformação de linguagem natural em comandos SQL, a execução de consultas em banco de dados, o retorno dos dados da pesquisa ao usuário, a possibilidade de troca do banco de dados consultado e a possibilidade de troca entre modelos de IA. A implementação dessas funcionalidades buscou tornar a consulta a dados mais simples, acessível e eficiente, reduzindo a necessidade de intervenção direta de profissionais técnicos em demandas recorrentes.

As tecnologias, linguagens e ferramentas utilizadas no projeto foram Java, para o desenvolvimento da aplicação desktop; MySQL, para estruturação e consulta dos dados; LangChain, para integração entre o modelo de linguagem e o banco de dados; LM Studio, para execução e gerenciamento do modelo de IA local; Figma, para elaboração dos wireframes e protótipos de interface; e Git/GitHub, para versionamento, documentação e controle do projeto. O repositório principal da equipe indica predominância de Java no código-fonte, enquanto o documento do Trabalho de Graduação registra também o uso de Figma, LangChain, LM Studio, MySQL e Git como tecnologias e ferramentas associadas ao desenvolvimento.

A proposta relaciona-se diretamente aos conteúdos de Engenharia de Software, desenvolvimento em Java e banco de dados, uma vez que envolveu a criação de uma aplicação desktop, a organização de requisitos, a integração com modelo de linguagem, a construção de consultas SQL e a execução de operações sobre dados estruturados. Nesse contexto, Sommerville (2019) contribui para a fundamentação do projeto ao abordar processos de desenvolvimento, especificação, validação e evolução de sistemas de software.

Além disso, Deitel e Deitel (2017) são pertinentes à solução desenvolvida por tratarem da programação em Java, linguagem utilizada na construção da aplicação desktop e na implementação das funcionalidades principais do sistema. Assim, os conhecimentos relacionados à engenharia de software e à programação orientada a objetos apoiaram a construção de uma solução capaz de transformar comandos em linguagem natural em consultas ao banco de dados.

2.3 ANO 2 – 1º semestre

O projeto foi desenvolvido no período de agosto de 2024 a dezembro de 2024, durante o terceiro semestre do curso de Análise e Desenvolvimento de Sistemas. O projeto, intitulado OAK-RH, foi realizado em parceria com a empresa Youtan, atuante no desenvolvimento de soluções tecnológicas. A proposta teve como foco o desenvolvimento de um dashboard de feedback e pesquisa de clima e cultura, voltado ao apoio das rotinas de Recursos Humanos. O problema identificado estava relacionado à realização descentralizada e manual de pesquisas de clima organizacional, o que dificultava a consolidação dos dados, atrasava a análise dos resultados e prejudicava a visualização do desempenho de gestores, equipes e colaboradores. Dessa forma, a contribuição pretendida foi centralizar o processo de coleta, organização e visualização de feedbacks, permitindo que a equipe de RH acompanhasse o desenvolvimento dos colaboradores e identificasse áreas de melhoria com maior clareza e agilidade. O repositório da equipe descreve o projeto como uma aplicação destinada à criação de pesquisas personalizadas envolvendo categorias como habilidades técnicas, comportamento e satisfação, com o objetivo de gerar insights sobre o clima organizacional e a performance individual e coletiva.

No desenvolvimento do projeto, o autor atuou como membro do Development Team, sendo responsável pela implementação de funcionalidades relacionadas à personalização das pesquisas de clima. A atuação envolveu tanto o desenvolvimento de interfaces no frontend, seguindo o design definido pela equipe no início do projeto, quanto atividades no backend, incluindo a modelagem da funcionalidade no banco de dados e sua integração com a aplicação. Também foi realizada a integração e configuração do Firebase como serviço de autenticação, possibilitando login, cadastro de usuários e recuperação de senha por meio de envio de e-mail com link para redefinição. Essas contribuições foram registradas no documento do Trabalho de Graduação, que também indica o uso de React, TypeScript, Node.js, PostgreSQL, Firebase, Figma e Git no desenvolvimento da solução.

A solução desenvolvida consistiu em uma aplicação web para gestão de pesquisas de clima, cultura e feedback organizacional. O sistema permite que administradores gerenciem usuários, classifiquem perfis como líder, liderado ou ambos, cadastrem equipes, criem formulários personalizados e associem perguntas a diferentes tipos de avaliação, como autoavaliação, avaliação de liderança e avaliação de liderado. As perguntas podem ser configuradas em diferentes formatos, incluindo texto longo, escolha única e múltipla escolha, além de serem vinculadas a categorias como hard skills, soft skills, expectativas e satisfação. O usuário interage com a plataforma por meio de login com permissões específicas, acesso a formulários pendentes, resposta às avaliações e visualização de dashboards conforme seu nível de acesso.

Entre as principais funcionalidades, destacaram-se o cadastro e gerenciamento de usuários, a classificação por níveis de acesso, o cadastro e gerenciamento de equipes, a criação de formulários personalizados, a associação de perguntas a categorias e tipos de avaliação, a resposta de pesquisas por colaboradores, a visualização de dashboards pessoais, dashboards de liderados e dashboard geral para administradores, além da aplicação de filtros por data e da geração de relatórios em PDF. O repositório da equipe apresenta essas funcionalidades no backlog do produto, nos requisitos funcionais e no planejamento das Sprints, demonstrando a evolução incremental da aplicação ao longo do semestre.

Como a atuação ocorreu no papel de desenvolvedor, foram priorizados aspectos relacionados à construção de interfaces, implementação de funcionalidades, integração entre frontend, backend e banco de dados, organização das informações e melhoria da usabilidade. Durante o projeto, também houve necessidade de lidar com trechos de código desorganizados, especialmente na exibição das pesquisas no frontend, o que levou à prática de comunicação e colaboração com a equipe para refatoração de partes da aplicação, tornando o código mais claro e estruturado. Essa experiência reforçou a importância da manutenibilidade, da colaboração entre desenvolvedores e da organização do código no desenvolvimento de sistemas.

As tecnologias, linguagens e ferramentas utilizadas no projeto foram Figma, para criação de wireframes e protótipos de interface; React, para desenvolvimento da interface e dos componentes interativos; TypeScript e JavaScript, para implementação das funcionalidades; Node.js, para desenvolvimento do backend; PostgreSQL, para armazenamento e gerenciamento dos dados; Firebase, para autenticação de usuários; Visual Studio Code, como ambiente de desenvolvimento; e Git/GitHub, para versionamento,

documentação e controle do projeto. O repositório da equipe também indica predominância de TypeScript, seguido por CSS, JavaScript e HTML na composição do código-fonte.

A proposta relaciona-se diretamente aos conteúdos de Engenharia de Software, Banco de Dados, Desenvolvimento Web e programação com TypeScript, uma vez que envolveu levantamento de requisitos, modelagem de dados, construção de interfaces, definição de perfis de acesso, implementação de funcionalidades, autenticação de usuários e preocupação com usabilidade. Nesse sentido, Sommerville (2019) contribui para a fundamentação do projeto ao abordar práticas de especificação, projeto, desenvolvimento, validação e manutenção de sistemas. A utilização de TypeScript também se relaciona à obra de Adriano (2021), por tratar da melhoria de aplicações JavaScript por meio de tipagem, organização e maior segurança no desenvolvimento. Além disso, a construção das funcionalidades e a organização lógica da aplicação dialogam com os fundamentos apresentados por Nascimento Júnior (2017), especialmente no que se refere à estruturação de código e à abordagem prática da programação orientada a objetos.

2.4 ANO 2 – 2º semestre

O projeto foi desenvolvido no período de janeiro de 2025 a junho de 2025, durante o quarto semestre do curso de Análise e Desenvolvimento de Sistemas. O projeto, intitulado Lumen, foi realizado em parceria com a FAPG, Fundação de Apoio à Pesquisa e Gestão, organização relacionada ao apoio, gestão e acompanhamento de projetos. A proposta teve como objetivo desenvolver uma plataforma web para gestão de projetos da FAPG, permitindo maior organização das iniciativas, das equipes, dos responsáveis e dos orçamentos associados. O problema identificado estava relacionado à ausência de um sistema centralizado para gerenciamento de clientes, equipes, projetos e orçamentos, o que dificultava o controle das entregas, a comunicação entre os envolvidos e o acompanhamento das responsabilidades individuais. O repositório da equipe apresenta como objetivo o desenvolvimento de uma plataforma web para gestão de projetos da FAPG.

No desenvolvimento do projeto, o autor atuou como membro do Development Team, contribuindo para a implementação de funcionalidades centrais da aplicação. A atuação envolveu a construção da relação hierárquica entre projetos, etapas e tarefas, permitindo que cada projeto fosse estruturado em partes menores e acompanhado de forma mais organizada. Também foram implementados recursos de cadastro e autenticação de usuários via Firebase, incluindo login, criação de conta, recuperação de senha e atribuição de usuários a projetos. Além disso, foi desenvolvida uma função para associação de membros externos, de modo

que novos cadastros pudessem receber automaticamente seus respectivos projetos no painel principal. Essas contribuições são descritas no repositório do Trabalho de Graduação, que registra a atuação do autor como desenvolvedor no projeto.

A solução desenvolvida consistiu em uma plataforma web para gestão de projetos, com foco no cadastro, organização, acompanhamento e recuperação de informações relacionadas aos projetos da FAPG. O usuário interage com o sistema por meio de uma interface web, acessando a plataforma com credenciais seguras, cadastrando projetos, criando etapas e tarefas, associando responsáveis, acompanhando o progresso das atividades e consultando informações por diferentes critérios. O backlog do produto contemplou funcionalidades como login seguro, cadastro de projetos, listagem de projetos cadastrados, criação de etapas e tarefas, divisão de tarefas em subtarefas, associação de datas e responsáveis, filtros por área de atuação, edição e exclusão de projetos, atualização de status de tarefas e recuperação de projetos excluídos.

Entre os principais requisitos funcionais, destacaram-se o cadastro de projetos com informações como nome, descrição, área de atuação, responsáveis e status; a recuperação intuitiva dos dados dos projetos; a atualização e exclusão de registros; a visualização por área de atuação, responsáveis e status; e o acompanhamento do andamento das atividades. Também foram definidos requisitos não funcionais relacionados à usabilidade, segurança e privacidade dos dados, além da capacidade de converter linguagem natural em chamadas de funções, extrair parâmetros de mensagens e orquestrar chamadas de forma organizada, sem uso de APIs externas.

Como a atuação ocorreu no papel de desenvolvedor, foram priorizados aspectos de implementação, integração entre camadas, organização de dados, clareza do fluxo de navegação e construção de funcionalidades alinhadas às necessidades da empresa parceira. Durante o projeto, também foram enfrentadas inconsistências em trechos de código, o que exigiu colaboração com a equipe, pensamento crítico e alinhamento entre os membros para tornar as funcionalidades mais compreensíveis, reduzir conflitos e favorecer a manutenção futura da aplicação. Essa experiência evidenciou a importância da organização do código, da comunicação técnica e da padronização durante o desenvolvimento de software.

As tecnologias, linguagens e ferramentas utilizadas no projeto foram Figma, para prototipação de interface; GitHub, para versionamento; Visual Studio Code, como ambiente de desenvolvimento; PostgreSQL, para armazenamento e gerenciamento dos dados; Node.js, para execução e desenvolvimento do backend; JavaScript e TypeScript, para implementação

das funcionalidades; React, para desenvolvimento da interface web e dos componentes interativos; e Firebase, para autenticação de usuários. O repositório da equipe lista essas ferramentas e tecnologias como parte da construção do projeto.

A proposta relaciona-se diretamente aos conteúdos de Banco de Dados, Desenvolvimento Web, Engenharia de Software e padrões de projeto, uma vez que envolveu requisitos funcionais e não funcionais, organização de backlog, modelagem de dados, implementação de interfaces, controle de usuários, segurança, manutenção de código e acompanhamento de progresso. Nesse contexto, Gamma et al. (1993) contribuem para a fundamentação do projeto ao abordarem padrões de projeto como soluções reutilizáveis para problemas recorrentes no desenvolvimento orientado a objetos, aspecto relevante para a organização e manutenção de sistemas corporativos. Além disso, Date (2005) é pertinente ao projeto por tratar da teoria relacional de banco de dados, relacionada à estruturação, persistência e recuperação das informações utilizadas pela plataforma de gestão de projetos. Dessa forma, os conhecimentos de arquitetura, organização de dados e reutilização de soluções apoiaram a construção de uma aplicação mais estruturada e adequada às necessidades da empresa parceira.

2.5 ANO 3 – 1º semestre

O projeto foi desenvolvido no período de agosto de 2025 a dezembro de 2025, durante o quinto semestre do curso de Análise e Desenvolvimento de Sistemas. O projeto, intitulado GeoMaps, foi realizado em parceria com a Visiona, empresa atuante no setor aeroespacial e em soluções tecnológicas voltadas ao uso de dados geoespaciais. A proposta teve como objetivo desenvolver uma solução para endereçamento digital de propriedades rurais vinculadas ao CAR — Cadastro Ambiental Rural, permitindo que moradores, empresas e órgãos públicos pudessem localizar imóveis rurais com maior precisão. O problema identificado estava relacionado à ausência de endereços oficiais reconhecidos por aplicativos de navegação em diversas regiões rurais, o que dificultava entregas, acesso a serviços básicos, deslocamentos e localização de propriedades. Embora os imóveis possuam registro no CAR, o código é extenso e não pode ser utilizado diretamente para geração de rotas em ferramentas como Google Maps ou Waze. Dessa forma, a contribuição pretendida foi criar uma solução capaz de transformar informações georreferenciadas em um endereço digital utilizável, facilitando a navegação rural e o acesso às propriedades. O repositório da equipe define o objetivo do projeto como o desenvolvimento de um aplicativo móvel para

endereçamento digital de imóveis do CAR, roteirização rural, registro colaborativo de condições de vias e exibição de alertas.

No desenvolvimento do projeto, o autor assumiu o papel de Product Owner, sendo responsável pela documentação das funcionalidades, comunicação entre equipe e cliente, definição e refinamento de requisitos, organização das demandas e apoio à priorização das entregas. A atuação envolveu a tradução das necessidades da empresa parceira em requisitos compreensíveis para a equipe de desenvolvimento, utilizando descrições funcionais, exemplos práticos e referências visuais de aplicações conhecidas, como sistemas de navegação e mapas, com o objetivo de reduzir interpretações ambíguas e facilitar a execução das tarefas. Essa atuação esteve alinhada às práticas de metodologias ágeis, especialmente na organização do backlog, na definição de histórias de usuário e no acompanhamento das entregas ao longo das Sprints. O repositório da equipe registra o autor como Product Owner e apresenta um backlog estruturado com histórias de usuário, prioridades, estimativas, Sprints e requisitos funcionais associados.

A solução desenvolvida consistiu em uma aplicação mobile e web, composta por um aplicativo voltado ao usuário final e por um portal administrativo. No aplicativo mobile, o morador ou proprietário rural pode realizar cadastro, autenticar-se no sistema, visualizar propriedades associadas ao CPF, consultar os imóveis no mapa e atribuir um Plus Code ao ponto de acesso principal da propriedade. Esse ponto passa a funcionar como um endereço digital, permitindo que outros usuários tracem rotas até o local. A aplicação também contempla funcionalidades para registro colaborativo de condições de vias, visualização de ocorrências no mapa, alertas meteorológicos, notificações, rotas entre propriedades e suporte a uso offline com sincronização posterior.

Além do aplicativo, a solução incluiu um portal web administrativo, por meio do qual administradores podem gerenciar usuários, visualizar logs, revisar relatos de vias, moderar informações indevidas e acompanhar estatísticas, como número de imóveis endereçados e alertas ativos. O sistema também prevê controle de acesso por perfil, contemplando usuários públicos, proprietários e administradores. Usuários públicos podem acessar informações não sensíveis, como rotas e alertas gerais; proprietários podem acessar dados de seus próprios imóveis, cadastrar alertas e atualizar informações de endereçamento; e administradores podem gerenciar dados, moderação e registros sensíveis da plataforma.

Entre as principais funcionalidades previstas e desenvolvidas, destacaram-se a autenticação com JWT, a listagem de propriedades associadas ao CPF, a visualização de

imóveis no mapa, a criação e atualização de Plus Code, a geração de certificado digital de endereço, o histórico de alterações, o cadastro de condições de vias, a visualização de ocorrências georreferenciadas, o traçado de rotas com integração ao Google Maps, a exibição de alertas meteorológicos, o portal administrativo, a moderação de relatos, o registro de logs e a implantação do serviço em ambiente AWS. As entregas foram organizadas em três Sprints principais: a primeira voltada à autenticação, propriedades, Plus Code e base visual; a segunda voltada ao registro e visualização de vias, rotas, histórico e notificações; e a terceira voltada ao portal administrativo, moderação, preferências de notificação e sincronização offline.

Além das responsabilidades de Product Owner, o autor também contribuiu tecnicamente para o projeto. Foram implementadas funcionalidades relacionadas ao cadastro, autenticação e recuperação de senha com JWT, além da integração de rotas geográficas no frontend e no backend. Também foi realizado o deploy completo da API e do site na AWS, garantindo disponibilidade da aplicação e permitindo que o sistema pudesse ser acessado em ambiente externo. Essas contribuições evidenciam a atuação tanto na camada de gestão do produto quanto na implementação técnica da solução.

As tecnologias, linguagens e ferramentas utilizadas no projeto foram Figma, para criação de wireframes e protótipos de interface; MongoDB, para armazenamento e gerenciamento de dados; Python, para implementação de regras e rotas no backend; React Native, para desenvolvimento da interface mobile; TypeScript e JavaScript, para construção das funcionalidades do frontend; JWT, para autenticação e controle de acesso; GPS/Mapas, para recursos de geolocalização e navegação; AWS, para publicação e disponibilização da aplicação; e Git/GitHub, para versionamento, documentação e controle do projeto. Observa-se que o documento do Trabalho de Graduação menciona FastAPI como tecnologia de backend, enquanto o README do repositório da equipe lista Django entre as tecnologias de APIs e serviços. Para o texto final do TG, recomenda-se padronizar essa informação conforme a implementação efetivamente utilizada no código.

A proposta relaciona-se diretamente aos conteúdos de Desenvolvimento Mobile, Engenharia de Software, Banco de Dados, Geolocalização, Integração de APIs e Gestão Ágil de Projetos, uma vez que envolveu levantamento de requisitos, organização de backlog, definição de perfis de usuário, construção de interfaces, persistência de dados, autenticação, consumo de serviços de mapas, deploy em nuvem e validação incremental da solução. Nesse sentido, Griffiths (2019) contribui para a fundamentação do projeto ao abordar conceitos de desenvolvimento para Android, relevantes para a construção de aplicações móveis. Paul e

Nalwaya (2019) também se relacionam diretamente à solução, por tratarem do desenvolvimento mobile com React Native, tecnologia associada à criação da interface do aplicativo. Além disso, Sadalage e Fowler (2013) são pertinentes ao projeto por discutirem bancos de dados NoSQL e persistência poliglota, aspecto relacionado ao uso de MongoDB e à necessidade de armazenamento flexível de dados georreferenciados, usuários, propriedades, relatos e alertas.

2.6 ANO 3 – 2º semestre

O projeto foi desenvolvido no período de janeiro de 2026 a junho de 2026, durante o sexto semestre do curso de Análise e Desenvolvimento de Sistemas. O projeto, intitulado RuleAI, foi realizado em parceria com a empresa Dom Rock, organização atuante no desenvolvimento de soluções baseadas em dados, automação e inteligência artificial aplicada a processos corporativos. A proposta teve como objetivo o desenvolvimento de uma aplicação web com uso de IA Generativa para criação, gerenciamento e controle de regras de negócio, permitindo que gestores interagissem com o sistema por meio de linguagem natural e acompanhassem os resultados por indicadores e gráficos. O problema identificado estava relacionado à complexidade do gerenciamento de regras de comissionamento, uma vez que diferentes empresas possuem critérios próprios para cálculo de comissões, bonificações, afastamentos, desligamentos, regras comerciais e condições específicas de remuneração. Essa variação tornava o processo operacionalmente custoso, sujeito a erros e dependente de equipes dedicadas à manutenção manual das regras. O repositório da equipe descreve o objetivo do projeto como o desenvolvimento de uma aplicação web para criação e controle de regras de negócio com técnicas de IA Generativa, promovendo maior explicabilidade e eficiência nas decisões.

No desenvolvimento do projeto, o autor atuou como membro do Development Team, com foco principal na construção da camada de inteligência artificial. A atuação envolveu o desenvolvimento de um código-base capaz de aceitar intercorrências e bonificações para criação de novas regras de comissão, além da implementação de uma estrutura de RAG — Retrieval-Augmented Generation com base nesse código. Também houve participação no gerenciamento do fluxo da IA por meio de LangChain e LangGraph, bem como na criação de rotas em FastAPI para expor as funcionalidades da LLM ao backend principal do projeto. O documento do Trabalho de Graduação registra ainda a contribuição do autor na definição da lógica da solução e no auxílio à integração entre IA, backend e frontend, garantindo que os módulos fossem conectados conforme o planejamento da equipe.

A solução desenvolvida consistiu em uma aplicação web para criação e gerenciamento de regras de comissão com apoio de inteligência artificial. O usuário, em perfil de gerente, interage com a plataforma informando regras de negócio em linguagem natural, como condições de comissão, bonificação ou exceções operacionais. A partir desse texto, a IA interpreta a intenção da regra, identifica parâmetros relevantes e gera uma representação estruturada da regra, que pode ser visualizada em formato textual ou em código. A aplicação também permite listar regras vigentes, consultar histórico de alterações, restaurar versões anteriores, avaliar a qualidade das respostas geradas, acompanhar indicadores de desempenho e visualizar gráficos com os resultados das regras de negócio. Essas funcionalidades aparecem no backlog e nos requisitos funcionais do repositório da equipe, incluindo criação de regras em linguagem natural, listagem de regras vigentes, visualização clara do resultado, cálculo de comissões, restauração de versões anteriores, avaliação da resposta, histórico de alterações, monitoramento da aplicação e exibição de gráficos.

Como a atuação ocorreu no papel de desenvolvedor, foram priorizados aspectos de implementação técnica, integração entre serviços, organização do fluxo de IA, estruturação das rotas e criação de uma solução capaz de conectar linguagem natural, regras de negócio e processamento automatizado. A camada de IA foi construída com o objetivo de reduzir o esforço manual necessário para cadastrar ou alterar regras de comissão, tornando o processo mais ágil, flexível e menos suscetível a inconsistências. Também foram consideradas necessidades de explicabilidade, uma vez que o sistema deveria apresentar de forma clara o resultado gerado, permitir avaliação da resposta e exibir indicadores que auxiliassem o usuário na compreensão dos impactos das regras configuradas.

A arquitetura do projeto foi composta por uma interface web, um backend principal e um serviço específico para a LLM. A interface foi desenvolvida em Vue.js com TypeScript, enquanto o backend utilizou Java com Spring Boot. A camada de inteligência artificial foi implementada em Python, com exposição de rotas por meio do FastAPI. Para o gerenciamento e orquestração dos fluxos da LLM, foram utilizados LangChain e LangGraph. O projeto também mencionou o uso de MongoDB para armazenamento e gerenciamento dos dados, além de ferramentas e serviços como MLFlow, Ollama, HuggingFace, Groq e Git/GitHub para versionamento e organização do projeto. O repositório da equipe lista Spring/Java, LangChain, Vue.js, Python, MLFlow, Ollama, HuggingFace e Groq entre as tecnologias do projeto, enquanto o documento do Trabalho de Graduação também registra MongoDB, FastAPI, LangGraph, TypeScript, Vue e Git.

A proposta relaciona-se diretamente aos conteúdos de Inteligência Artificial, Engenharia de Software, Banco de Dados, Desenvolvimento Web, Arquitetura de Sistemas e Gestão Ágil de Projetos, uma vez que envolveu levantamento de requisitos, construção de aplicação web, integração entre frontend e backend, modelagem de regras de negócio, persistência de dados, uso de modelos de linguagem, implementação de RAG e organização incremental das entregas. Nesse contexto, Russell e Norvig (2021) contribuem para a fundamentação do projeto ao abordarem os conceitos de agentes inteligentes, representação de conhecimento, raciocínio e aplicações de inteligência artificial em sistemas computacionais. Além disso, a referência da Hugging Face (2026) sobre Sentence Transformers é pertinente ao projeto por relacionar-se ao uso de modelos de representação semântica de textos, embeddings e recuperação de informações, aspectos associados à construção de soluções baseadas em RAG e processamento de linguagem natural.

3. CONSIDERAÇÕES FINAIS

Ao longo da graduação em Análise e Desenvolvimento de Sistemas, os projetos integradores desenvolvidos em cada semestre possibilitaram a aplicação progressiva dos conhecimentos adquiridos nas disciplinas do curso. A trajetória acadêmica foi marcada pela participação em diferentes propostas, com níveis crescentes de complexidade, abrangendo desde aplicações educativas e sistemas desktop até plataformas web, aplicações mobile e soluções com inteligência artificial. Dessa forma, as atividades integradoras contribuíram para consolidar competências técnicas, ampliar a compreensão sobre o desenvolvimento de software e aproximar o processo formativo de demandas reais apresentadas por empresas parceiras.

No primeiro semestre, o projeto **Pandora Box** permitiu o contato inicial com o desenvolvimento de uma aplicação web educativa, voltada ao ensino da metodologia Scrum. A experiência favoreceu a compreensão sobre organização de equipes ágeis, estruturação de conteúdos, prototipação de interfaces e construção de páginas web. No segundo semestre, o projeto **JPT** ampliou esse contato ao propor uma aplicação desktop capaz de converter comandos em linguagem natural em consultas SQL, envolvendo conceitos de banco de dados, inteligência artificial e integração entre sistemas.

No terceiro semestre, o projeto **OAK-RH** possibilitou o desenvolvimento de uma aplicação web voltada à gestão de pesquisas de clima, cultura e feedback organizacional. Nesse projeto, foram exercitadas habilidades relacionadas à criação de interfaces, autenticação de usuários, modelagem de dados, integração entre frontend e backend e organização de funcionalidades voltadas à área de Recursos Humanos. No quarto semestre, o projeto **Lumen** aprofundou a experiência com sistemas de gestão, ao propor uma plataforma para acompanhamento de projetos, etapas, tarefas, usuários e responsabilidades, reforçando a importância da organização da informação, da segurança e da usabilidade em aplicações corporativas.

No quinto semestre, o projeto **GeoMaps**, desenvolvido em parceria com a Visiona, representou um avanço significativo em termos de complexidade técnica e responsabilidade sobre o produto. A solução envolveu o desenvolvimento de uma aplicação mobile e web para endereçamento digital de propriedades rurais vinculadas ao CAR, contemplando geolocalização, rotas, Plus Code, alertas, portal administrativo e deploy em ambiente de nuvem. A atuação como Product Owner permitiu o aprimoramento de competências relacionadas ao levantamento de requisitos, comunicação com cliente, organização do

backlog, priorização de entregas e tradução de necessidades de negócio em funcionalidades compreensíveis para a equipe de desenvolvimento.

No sexto semestre, o projeto **RuleAI**, desenvolvido em parceria com a Dom Rock, possibilitou o contato com tecnologias mais avançadas, especialmente relacionadas à inteligência artificial generativa, RAG, LangChain, LangGraph, FastAPI e integração entre serviços. A proposta de criação e gerenciamento de regras de negócio por meio de linguagem natural evidenciou a importância da arquitetura de sistemas, da integração entre frontend, backend e serviços de IA, bem como da explicabilidade e confiabilidade em soluções automatizadas. Esse projeto contribuiu diretamente para a consolidação de conhecimentos técnicos mais próximos das demandas atuais do mercado de tecnologia.

As atividades integradoras também contribuíram para o desenvolvimento de habilidades interpessoais e organizacionais. Ao longo dos projetos, foram exercitadas práticas de trabalho em equipe, comunicação, divisão de responsabilidades, organização de tarefas, acompanhamento de prazos, apresentação de resultados e colaboração na resolução de problemas. A utilização de metodologias ágeis, especialmente Scrum, favoreceu a compreensão sobre ciclos incrementais de desenvolvimento, definição de papéis, planejamento de Sprints, refinamento de requisitos e validação contínua das entregas.

Além dos aprendizados técnicos, os projetos permitiram enfrentar desafios recorrentes do desenvolvimento de software, como mudanças de requisitos, dificuldades de integração, inconsistências no código, necessidade de refatoração, limitações de ferramentas, organização de banco de dados, autenticação de usuários, deploy de aplicações e comunicação entre diferentes camadas do sistema. A superação desses desafios contribuiu para o amadurecimento acadêmico e profissional, pois exigiu análise crítica, pesquisa, adaptação, colaboração e tomada de decisão.

Dessa forma, os projetos desenvolvidos ao longo da graduação tiveram papel fundamental na formação do aluno, pois permitiram relacionar teoria e prática em contextos reais de aplicação. As experiências vivenciadas contribuíram para o desenvolvimento técnico, ao ampliar o domínio sobre linguagens, frameworks, bancos de dados, ferramentas de versionamento, serviços em nuvem e inteligência artificial; para o desenvolvimento acadêmico, ao fortalecer a capacidade de documentação, análise e apresentação de soluções; e para o desenvolvimento pessoal, ao estimular responsabilidade, autonomia, comunicação e trabalho colaborativo.

Conclui-se que as práticas integradoras foram essenciais para a preparação para o mercado de trabalho, pois proporcionaram uma vivência próxima à realidade profissional da

área de tecnologia. A participação em projetos com empresas parceiras, diferentes papéis no Scrum e múltiplas tecnologias permitiu compreender melhor o ciclo completo de desenvolvimento de software, desde a identificação do problema até a entrega de uma solução funcional. Assim, a trajetória construída ao longo dos seis semestres reforça a importância da aprendizagem contínua, da adaptação às novas tecnologias e da busca constante por evolução profissional.

REFERÊNCIAS BIBLIOGRÁFICAS

SCHWABER, Ken; SUTHERLAND, Jeff. The Scrum Guide: The Definitive Guide to Scrum: The Rules of the Game. [S.l.]: Scrum.org & Scrum Inc., 2020.

SUTHERLAND, Jeff. Scrum: A arte de fazer o dobro de trabalho na metade do tempo. Rio de Janeiro: Leya Brasil, 2014.

SOMMERVILLE, Ian. Engenharia de software. 10. ed. São Paulo: Pearson Universidades, 2019.

DEITEL, H. M.; DEITEL, P. J. Java: como programar. 10. ed. São Paulo: Pearson, 2017.

SOMMERVILLE, Ian. Engenharia de software. 10. ed. São Paulo: Pearson Universidades, 2019.

NASCIMENTO JR., O. S. Introdução à Orientação a Objetos com C++ e Python: Uma abordagem prática. São Paulo: Novatec, 2017.

ADRIANO, Thiago da Silva. Guia prático de TypeScript: melhore suas aplicações JavaScript. São Paulo: Casa do Código, 2021.

GAMMA, Erich et al. Design patterns: Abstraction and reuse of object-oriented design. In: European conference on object-oriented programming. Berlin, Heidelberg: Springer Berlin Heidelberg, 1993. p. 406-431.

DATE, Chris J. Database in depth: relational theory for practitioners. [S.L.]: O'Reilly Media, 2005.

GRIFFITHS, D. Use a Cabeça! Desenvolvendo Para Android. 2. ed. Rio de Janeiro: Alta Books, 2019.

PAUL, Akshat; NALWAYA, Abhishek. React Native for Mobile Development. [S.l.]: Apress, 2019.

SADALAGE, Pramod J.; FOWLER, Martin. NoSQL essencial: um guia conciso para o mundo emergente da persistência poliglota. São Paulo: Novatec, 2013.

RUSSELL, S.; NORVIG, P. Artificial Intelligence: A Modern Approach. 4. ed. [S.l.]: Pearson, 2021

HUGGING FACE. Sentence Transformers. Disponível em: <https://huggingface.co/sentence-transformers>. Acesso em: 01 mar. 2026.

APÊNDICE A - PORTIFÓLIO



1 Apresentação do Aluno



Nome: João Victor Menezes Oliveira

Idade: 21 anos

Curso: Análise e Desenvolvimento de Sistemas



Histórico Acadêmico

- **Ensino Médio:** Colégio Poliedro – São José dos Campos (*jan/2019 – dez/2021*)
 - **Cursinho:** Poliedro (*fev/2022 – ago/2022*)
 - **Faculdade:** PUC Minas – Coração Eucarístico (*Psicologia | ago/2022 – jan/2023*)
 - **Faculdade:** UNIP – São José dos Campos (*Análise e Desenvolvimento de Sistemas | fev/2023 – jun/2023*)
-



Motivação para Ingressar na FATEC

Desde pequeno, sempre fui fascinado por tecnologia. Adorava mexer no computador, explorar cada função e descobrir algo novo a cada dia. Para mim, parecia óbvio que meu caminho profissional seria na área de TI.

No terceiro ano do ensino médio, decidi apostar nesse sonho e me inscrevi para faculdades como UNICAMP e USP. Passei na primeira fase de ambas, mas uma conversa específica acabou mudando meus planos. Ao ouvir o relato de um profissional da área, visivelmente descontente com a própria carreira, senti um receio que até então não existia. Aquilo abalou minhas certezas e, tomado pela dúvida, desisti temporariamente da formação em

tecnologia e ingressei em um cursinho pré-vestibular.

Durante o cursinho, fiz um aconselhamento profissional. O resultado apontou a psicologia como uma opção promissora. Com essa indicação e uma bolsa de estudos que consegui na PUC Minas, decidi experimentar uma nova direção. No entanto, após um semestre de curso, percebi que aquela não era minha vocação. Foi então, no tempo livre, que fiz um curso gratuito de programação na internet e esse simples passo reacendeu totalmente minha paixão por **Tecnologia da Informação**, e me fez querer sair do curso em que eu estava para cursar algo na área de TI.

Minha intenção passou a ser ingressar na **FATEC**, mas como havia deixado a faculdade de psicologia em fevereiro, o prazo de inscrições já tinha se encerrado. Para não ficar parado, me matriculei na UNIP e cursei um semestre de Análise e Desenvolvimento de Sistemas enquanto aguardava o próximo vestibular da FATEC. Quando finalmente prestei a prova, fui aprovado e pude realizar a transferência.

Desde então, tenho mergulhado intensamente no universo da tecnologia. Cada aula, projeto e desafio só reafirma aquilo que, no fundo, sempre esteve claro: meu lugar é na área de TI.



Experiência Profissional

- **ValeJet** (*jun/2023*)

Estagiário em Suporte de TI. Responsável por corrigir cadastros incorretos no site da empresa e dar suporte técnico básico.

- **X-MDG** (*abr/2024 – out/2025*)

Estagiário em Desenvolvimento Full-Stack. Atividades principais:

- Criação do site da empresa utilizando **React**;
- Desenvolvimento da **API** com **MongoDB**;
- Implementação de recursos de **Inteligência Artificial**;
- Realização de **deploy** em servidor **Ubuntu**.

- **Liquide** (*mar/2026 – atualmente*)

Estagiário em Desenvolvimento Full-Stack. Atividades principais:

- Manutenção do site da empresa utilizando **React**;
 - Manutenção do **backend e banco de dados** da empresa utilizando **Supabase**;
 - Criação e implementação de um **chatbot** conversacional com base nas regras de negócio da empresa;
 - Realização de deploy em 2 ambientes utilizando **Heroku**.
-

Informações de Contato

- E-mail: jvictormo018@gmail.com
- GitHub: github.com/jvictormo
- LinkedIn: linkedin.com/in/jvictormo

Principais Conhecimentos

- Git
- MongoDB
- SQL
- React
- Vite
- Native
- Next
- TypeScript e JavaScript
- APIs RESTful
- Node.js
- Python

2. Projetos API

Identificador	Empresa Parceira	Descrição da Solução Desenvolvida	Nome	Link para Detalhes
API 1º Semestre (ago/2023)	Fatec-SJC	Página web interativa para ensino da metodologia ágil Scrum de forma autônoma.	Pandora Box	Ver Projeto
API 2º Semestre (jan/2024)	Fatec-SJC	Aplicação Java que traduz linguagem natural em SQL usando LangChain e LLM local.	JPT	Ver Projeto

API 3º Semestre (ago/2024)	Youtan	Plataforma web React para pesquisas de clima com visualização de resultados em tempo real.	OAK-RH	Ver Projeto
API 4º Semestre (jan/2025)	FAPG	Sistema de gestão de projetos e orçamentos com IA e autenticação Firebase.	Lumen	Ver Projeto
API 5º Semestre (ago/2025)	Visiona	Aplicativo e portal web para endereços digitais georreferenciados em áreas rurais.	Geo Maps	Ver Projeto
API 6º Semestre (jan/2026)	Dom Rock	Aplicativo e portal web para endereços digitais georreferenciados em áreas rurais.	RulesAi	Ver Projeto

Primeiro Semestre – ago/2023

Empresa Parceira: Antonio Egydio

Link do Repositório Git: github.com/jvictormo/Pandora-Box

Tecnologias Utilizadas:

- **Figma:** Elaboração dos wireframes e protótipos de interface
- **HTML:** Estruturação das páginas do site
- **CSS:** Estilização e responsividade das telas
- **JavaScript:** Adição de interatividade à página
- **Flask:** Criação e gerenciamento do servidor web
- **Git:** Versionamento e controle do projeto

Problema

Dentro da empresa, toda vez que um novo grupo de funcionários era contratado, havia a necessidade de reapresentar e ensinar a metodologia Scrum para essa nova equipe. Esse processo, além de repetitivo, exigia muito tempo dos instrutores e tornava-se custoso para a empresa. Era preciso encontrar uma forma de tornar o ensino mais rápido, intuitivo e acessível, permitindo que os colaboradores aprendessem de maneira autônoma e prática. Além disso, havia o desafio de manter a didática envolvente, de modo que os funcionários compreendessem não só a teoria, mas também a aplicação real da metodologia no ambiente corporativo. Essa situação causava retrabalho, baixa eficiência e perda de produtividade no treinamento inicial.

Solução

Com base nesse problema, foi desenvolvido uma página web educativa voltada ao ensino interativo da metodologia ágil Scrum. A plataforma apresenta explicações claras e exemplos práticos sobre o framework, utilizando uma interface intuitiva e atrativa.

O sistema conta com recursos que estimulam o aprendizado ativo, como avaliação por página, autoavaliação dos papéis Scrum (Scrum Master, Product Owner e Development Team) e uma navegação fluida entre os tópicos. Assim, o treinamento tornou-se mais dinâmico, acessível e eficiente, reduzindo o tempo gasto pela empresa na capacitação de novos colaboradores.

Contribuições Pessoais

Atuei como Scrum Master, sendo responsável pela coordenação da equipe e pela comunicação entre os membros. Realizei o design completo do website no Figma, baseando-me nos requisitos do cliente e nas metas do projeto, onde organizei e conduzi as daily meetings, mantive o controle do burndown chart e incentivei a participação e o engajamento da equipe. Também fiz os slides das apresentações para o cliente, e participei dela ativamente, mostrando os dados atualizados, e o estado da aplicação. Além disso, participei ativamente do desenvolvimento das principais telas, estruturando o HTML e aplicando o CSS. Além disso.

Hard Skills

- **Flask:** uso com auxílio
- **HTML:** consigo ensinar
- **CSS:** consigo ensinar
- **JavaScript:** consigo ensinar
- **Git:** uso com autonomia

Soft Skills

Durante o projeto, exercitei habilidades de **liderança**, **comunicação assertiva** e **resolução de conflitos**. Precisei mediar situações em que membros do grupo não cumpriram as tarefas propostas a eles. Para isso, tive um diálogo particular com eles, reforçando a importância da responsabilidade e da colaboração.

Desenvolvi também **resiliência**, **organização** e **empatia**, buscando manter a equipe motivada mesmo diante dessas dificuldades. Essas experiências foram fundamentais para meu crescimento pessoal e profissional.

Segundo Semestre – jan/2024

Empresa Parceira: Giuliano Bertoti

Link do Repositório Git: github.com/jvictormo/JPT

Tecnologias Utilizadas:

- **Figma:** Elaboração dos wireframes e protótipos de interface
- **Java:** Desenvolvimento da aplicação (frontend e backend)
- **LangChain:** Integração entre o modelo de linguagem e o banco de dados
- **LM Studio:** Execução e gerenciamento do modelo de IA local
- **Git:** Versionamento e controle do projeto

Problema

Diversos colaboradores da empresa precisavam consultar informações no banco de dados corporativo, mas muitos não possuíam conhecimentos em programação ou SQL. Isso gerava uma grande dependência do time de TI, tornando os processos mais lentos e desviando o foco da equipe técnica para tarefas repetitivas de baixo valor agregado.

Essa situação dificultava o acesso rápido a informações estratégicas e impactava diretamente a produtividade dos setores internos, tornando essencial uma solução que democratizasse o uso dos dados dentro da organização.

Solução

Foi desenvolvida uma aplicação desktop em Java, capaz de interpretar comandos em linguagem natural e traduzi-los automaticamente em consultas SQL. Com o auxílio de um modelo de linguagem local (LLM) integrado via LangChain, o sistema compreende o texto digitado pelo usuário, executa a consulta e exibe os resultados de forma clara e organizada. Dessa forma, colaboradores de diferentes áreas passaram a realizar consultas com autonomia, sem depender do setor de TI. A solução trouxe mais agilidade, reduziu gargalos internos e aumentou a eficiência no uso dos dados corporativos.

Contribuições Pessoais

Atuei como Scrum Master, sendo responsável pela coordenação da equipe e pela definição das prioridades de desenvolvimento. Contribuí no planejamento da arquitetura da aplicação, participei da modelagem do banco de dados e realizei a integração do LangChain com o modelo local (LM Studio) e com o banco SQL.

Além disso, conduzi testes de desempenho com base no tempo de resposta, corriji *bugs* como a falta de consistência da IA, e participei ativamente da refatoração do código, otimizando o tempo de resposta da aplicação. Trabalhei também para manter a comunicação fluida e garantir a entrega dentro do cronograma proposto pelo cliente.

Hard Skills

- **Java:** uso com auxílio
- **LangChain:** uso com auxílio
- **MySQL:** uso com autonomia
- **Git:** uso com autonomia

Soft Skills

Durante o projeto, utilizei **proatividade** para aprofundar meus conhecimentos em Java e LangChain, com o objetivo de solucionar o problema de lentidão da aplicação. Para isso, desenvolvi um prompt mais específico, reduzindo a margem de interpretação da IA e, consequentemente, diminuindo sua taxa de "alucinação". Essa atitude me ajudou a **desenvolver autonomia técnica**, aprimorar meu raciocínio lógico e fortalecer minha **capacidade de adaptação** diante de novos desafios.

Terceiro Semestre – ago/2024

Empresa Parceira: Youtan

Link do Repositório Git: github.com/Phoenix-Team-Fatec/OAK-RH

Tecnologias Utilizadas:

- **Figma:** Criação dos wireframes e protótipos de interface
- **PostgreSQL:** Armazenamento e gerenciamento dos dados
- **Node.js:** Desenvolvimento do servidor backend
- **TypeScript:** Implementação do backend e frontend
- **React:** Desenvolvimento da interface e componentes interativos
- **Firebase:** Autenticação de usuários
- **Git:** versionamento e controle do projeto

Problema

A empresa realizava pesquisas de clima para avaliar o ambiente organizacional, mas o processo era descentralizado e manual, dificultando a consolidação dos dados e atrasando a análise dos resultados.

Essa falta de integração gerava retrabalho e impedia uma visão precisa sobre o desempenho dos gestores, equipes e colaboradores, prejudicando a tomada de decisão e o engajamento interno.

Solução

Foi criada uma aplicação web em React que permite à empresa montar pesquisas de clima personalizadas, com perguntas dissertativas ou de múltipla escolha, categorizadas por temas. As respostas são armazenadas em um banco PostgreSQL, e os administradores podem visualizar os resultados em tempo real, com pontuações calculadas automaticamente.

Com isso, a plataforma trouxe centralização, agilidade e clareza no processo de avaliação interna, tornando a gestão de clima mais eficiente e interativa.

Contribuições Pessoais

Atuei como Desenvolvedor sendo responsável por implementar a personalização das pesquisas de clima, onde utilizei tanto do front, seguindo o design proposto pelo time no começo do projeto, e adaptando o necessario, quanto ao back, realizando a modelagem

dessa funcionalidade no banco de dados, e integrando ela. Além disso, também realizei a integração e configuração do FireBase utilizado apenas como serviço para realizar o login do usuário na aplicação, de forma a facilitar a segurança do site, tendo opções de cadastrar um usuário, reset de senha com um e-mail enviado contendo um link para modifica-la.

Hard Skills

- **PostgreSQL:** uso com autonomia
- **Node.js:** consigo ensinar
- **TypeScript:** consigo ensinar
- **React:** consigo ensinar
- **FireBase:** uso com autonomia
- **Git:** uso com autonomia

Soft Skills

Durante o projeto, enfrentei diversos trechos de código desorganizados, especialmente na exibição das pesquisas no frontend. Isso me levou a praticar **comunicação e colaboração** com a equipe para refatorarmos parte do código, tornando-o mais limpo e claro. Também aproveitei para reforçar a importância de um código bem estruturado e do trabalho em equipe, evitando que o problema se repetisse no futuro.

Quarto Semestre – jan/2025

Empresa Parceira: FAPG

Link do Repositório Git: github.com/Phoenix-Team-Fatec/Lumen

Tecnologias Utilizadas:

- **Figma:** Criação dos wireframes e protótipos de interface
- **PostgreSQL:** Armazenamento e gerenciamento dos dados
- **Node.js:** Desenvolvimento do servidor backend
- **TypeScript:** Implementação do backend e frontend
- **React:** Desenvolvimento da interface e componentes interativos
- **Firebase:** Autenticação de usuários
- **Inteligência Artificial (IA):** Chatbot interativo integrado à plataforma
- **Git:** Versionamento e controle do projeto

Problema

A empresa atuava com projetos terceirizados, mas não possuía um sistema centralizado para gerenciar clientes, equipes e orçamentos. Isso dificultava o controle das entregas e o acompanhamento de responsabilidades individuais.

Além disso, a falta de familiaridade dos colaboradores com ferramentas digitais tornava a adoção de novas soluções ainda mais complexa, afetando a eficiência e a comunicação entre as equipes.

Solução

Foi desenvolvida uma plataforma web em React que permite cadastrar e gerenciar projetos, associando-os a empresas, áreas e limites orçamentários. Dentro de cada projeto, é possível criar etapas e tarefas interligadas, com indicadores visuais de progresso e um chat de IA integrado para auxiliar na execução das atividades.

A solução trouxe organização, transparência e acessibilidade, modernizando o fluxo de trabalho da empresa e facilitando o acompanhamento de resultados.

Contribuições Pessoais

Atuei como Desenvolvedor, responsável por implementar a relação hierárquica entre projetos, etapas e tarefas, De forma direta, melhore a gramatica e a ortografia desse texto. Também implementei o sistema de cadastro e autenticação via FireBase, incluindo o recurso

de reset de senha e atribuição de usuários a projetos. Criei ainda uma função que permite associar membros externos, para que novos cadastros recebam automaticamente seus projetos no painel principal.

Hard Skills

- **PostgreSQL:** uso com autonomia
- **Node.js:** consigo ensinar
- **TypeScript:** consigo ensinar
- **React:** consigo ensinar
- **FireBase:** uso com autonomia
- **Git:** uso com autonomia

Soft Skills

Durante o projeto, enfrentei diversos códigos inconsistentes. A cada nova funcionalidade, eles se tornavam destoantes, gerando conflitos e exigindo explicações constantes para os outros membros da equipe. Por isso, apliquei **colaboração e pensamento crítico** para, junto ao time, alinhar os trechos conflitantes e sugerir que comentássemos o código para deixar as funcionalidades mais claras e evitar problemas futuros. Essa experiência reforçou minha **resiliência e foco em resultados**.

Quinto Semestre – ago/2025

Empresa Parceira: Visiona

Link do Repositório Git: github.com/Phoenix-Team-Fatec/Geo-Maps

Tecnologias Utilizadas:

- **Figma:** Criação dos wireframes e protótipos de interface
- **MongoDB:** Armazenamento e gerenciamento dos dados
- **FastAPI:** Desenvolvimento do servidor backend
- **Python:** Implementação das rotas e lógica do backend
- **TypeScript:** Desenvolvimento do frontend
- **React Native:** Criação da interface mobile
- **Git:** Versionamento e controle do projeto

Problema

Em regiões rurais, muitos moradores não possuem endereços oficiais reconhecidos por aplicativos de navegação, o que dificulta o acesso a serviços básicos e entregas. Embora cada terreno possua um registro no CAR (Cadastro Ambiental Rural), o código é extenso e não pode ser utilizado diretamente para gerar rotas em plataformas como Google Maps ou Waze.

Isso cria uma barreira de acessibilidade tanto para os moradores quanto para empresas e órgãos públicos que necessitam localizar essas propriedades.

Solução

Foi desenvolvido uma aplicação mobile e web em React Native e FastAPI, que permite ao morador registrar um ponto georreferenciado em seu terreno, vinculando-o ao CPF e ao código CAR. Assim, o local passa a funcionar como um endereço digital, permitindo que qualquer usuário trace rotas até ele.

Além do app de navegação, há um portal web administrativo, onde a empresa pode gerenciar alertas, condições meteorológicas e dados de cadastro, centralizando todas as informações do sistema.

Contribuições Pessoais

Atuei como Product Owner, sendo responsável pela documentação das funcionalidades e pela comunicação entre equipe e cliente. Descrevi os requisitos técnicos de forma clara,

auxiliando na priorização e na execução das entregas.

No desenvolvimento, implementei o cadastro, autenticação e recuperação de senha com JWT, além da integração das rotas geográficas no front e back-end. Também realizei o deploy completo da API e do site na AWS, garantindo estabilidade e disponibilidade do sistema.

Hard Skills

- **MongoDB:** consigo ensinar
- **FastAPI:** faço uso com ajuda
- **TypeScript:** consigo ensinar
- **React Native:** uso com autonomia
- **Python:** uso com autonomia
- **Git:** uso com autonomia

Soft Skills

Durante o projeto, precisei exercitar **adaptabilidade** e **organização** para ajustar continuamente a documentação conforme o avanço do time. Busquei formas mais eficientes de descrever as funcionalidades, utilizando exemplos de aplicações existentes e conhecidas para reduzir o tamanho das descrições e diminuir a margem para interpretações individuais. Otimizando assim a comunicação e garantindo clareza na execução das tarefas.



Sexto Semestre – jan/2026

Empresa Parceira: Dom Rock

Link do Repositório Git: github.com/Phoenix-Team-Fatec/API_6

Tecnologias Utilizadas:

- **MongoDB:** Armazenamento e gerenciamento dos dados
- **FastAPI:** Desenvolvimento do servidor da LLM
- **Python:** Implementação da LLM e da lógica das rotas relacionadas à IA
- **LangChain:** Gerenciamento da LLM em formato de agente
- **LangGraph:** Organização do fluxo do agente por meio de grafos
- **Java:** Implementação das rotas e da lógica do backend
- **Spring Boot:** Desenvolvimento do servidor backend
- **TypeScript:** Desenvolvimento do frontend
- **Vue:** Criação da interface web
- **Git:** Versionamento e controle do projeto



Problema

Diversas empresas possuem modelos próprios de comissionamento para seus funcionários. No entanto, devido à constante mudança de regras, afastamentos, desligamentos e diferentes critérios de remuneração, o cálculo de comissões se torna um processo complexo e difícil de gerenciar.

Além disso, a governança entre cargos, regras comerciais e condições específicas de bonificação aumenta a dificuldade de garantir que os valores sejam calculados corretamente. Isso gera a necessidade de um time dedicado ao cálculo e à distribuição das comissões, aumentando o esforço operacional e a margem de erro.



Solução

Foi desenvolvida uma aplicação web em Vue, utilizando um backend em Spring Boot e uma LLM em Python, exposta por meio de rotas em FastAPI. A solução permite que o usuário registre novas regras de comissão de forma rápida, visualizando em tempo real os impactos que essas alterações podem causar no sistema.

A aplicação funciona com o apoio de uma IA que interpreta o input do usuário, identifica a intenção da regra informada e gera um código com base nesse entendimento. Dessa forma,

o processo de criação e adaptação de regras de comissão se torna mais ágil, flexível e menos suscetível a erros manuais.

Contribuições Pessoais

Atuei como Desenvolvedor, sendo responsável principalmente pela parte da LLM. Desenvolvi um código-base capaz de aceitar intercorrências e bonificações para a criação de novas regras de comissão. Além disso, implementei um RAG com base nesse código e auxiliei no gerenciamento do fluxo da IA utilizando LangChain e LangGraph.

Também fui responsável por criar rotas em FastAPI para expor as funcionalidades da LLM ao backend do projeto. Durante o desenvolvimento, contribuí com ideias para a lógica da solução e auxiliei integrantes do grupo na integração entre IA, backend e frontend, garantindo que as partes do sistema fossem conectadas conforme o planejamento definido pela equipe.

Hard Skills

- **Python:** uso com autonomia
- **FastAPI:** uso com autonomia
- **LangChain:** uso com autonomia
- **LangGraph:** uso com autonomia
- **Git:** uso com autonomia

Soft Skills

Durante o projeto, precisei exercitar **proatividade**, **adaptabilidade** e **aprendizagem contínua** para pesquisar, avaliar e escolher as tecnologias que melhor se encaixavam na proposta da solução. Esse processo exigiu atenção ao *trade-off* entre complexidade, produtividade e integração com as demais partes do sistema.

Com o conhecimento obtido por meio de fóruns, inteligências artificiais, videoaulas e discussões com o time, auxiliei no desenvolvimento da arquitetura do projeto, contribuindo para a definição das ferramentas, fluxos e integrações utilizados na aplicação.