

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL  
ESCOLA POLITÉCNICA  
CURSO DE BACHARELADO EM ENGENHARIA DE SOFTWARE  
AGES - AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE SOFTWARE

MATEUS CAMPOS CAÇABUENA

**MEMORIAL DE ATUAÇÃO NA AGÊNCIA EXPERIMENTAL DE ENGENHARIA DE  
SOFTWARE – PERÍODO 2023/1  
AGES I, II**

Porto Alegre

2024

## RESUMO

Este trabalho retrata a minha atuação na Agência Experimental de Engenharia de Software (AGES), disciplina realizada quatro vezes durante o curso de bacharelado em Engenharia de Software da PUCRS. A AGES é o local onde ocorre grande troca de conhecimentos entre os alunos, assim como o aprendizado prático e integrador do curso, objetivando a imersão dos alunos no processo de desenvolvimento de software e o trabalho em equipe, similares ao mercado de trabalho. Em cada módulo, o aluno realiza diferentes papéis como tarefas de programação na primeira etapa, análise de dados da aplicação na segunda, arquitetura do software na terceira e, finalmente, a gestão do time na quarta. Será feita uma reflexão sobre minha evolução desde o início do curso até o momento atual, e o papel da AGES nessa evolução.

**PALAVRAS CHAVES:** AGES, Engenharia de Software, Aprendizado, Tarefas, Software, Desenvolvimento, Evolução.

## LISTA DE ILUSTRAÇÕES

<b>Figura 1: Time do projeto Veículos via Montadora .....</b>	<b>7</b>
<b>Figura 2: Tela de Extração do PDF .....</b>	<b>8</b>
<b>Figura 3: Descrição e Status da User Stories da Sprint 1 .....</b>	<b>12</b>
<b>Figura 4: Time do projeto ENSportive.....</b>	<b>18</b>
<b>Figura 5: Tela de Extração do PDF.....</b>	<b>19</b>

## **LISTA DE SIGLAS**

**AWS** – Amazon Web Services

**ABNT** – Associação Brasileira de Normas Técnicas

**AGES** – Agência Experimental de Engenharia de Software

**CEO** – Chief Executive Officer

**CSV** – Comma-Separated Values

**CSS** – Cascading Style Sheets

**ES** – Engenharia de Software

**HTML** – HyperText Markup Language

**JSON** – JavaScript Object Notation

**PDF** – Portable Document Format

**PUCRS** – Pontifícia Universidade Católica do Rio Grande do Sul

**US** – User Stories

**CRUD** – Create, Read, Update, Delete

**BD** – Banco de Dados

## SUMÁRIO

<b>1 – APRESENTAÇÃO DA TRAJETÓRIA DO ALUNO .....</b>	<b>5</b>
<b>2 – PROJETOS AGES I - “VEÍCULOS VIA MONTADORA” .....</b>	<b>6</b>
<b>2.1 Introdução.....</b>	<b>6</b>
<b>2.2 Desenvolvimento do Projeto .....</b>	<b>7</b>
<b>2.3 Atividades desempenhadas pelo aluno no projeto .....</b>	<b>10</b>
<b>2.4 CONCLUSÃO .....</b>	<b>15</b>
<b>3 – PROJETOS AGES II - “ENSportive – Estilo de Vida Esportivo” .....</b>	<b>17</b>
<b>3.1 Introdução.....</b>	<b>17</b>
<b>3.2 Desenvolvimento do Projeto .....</b>	<b>18</b>
<b>3.3 Atividades desempenhadas pelo aluno no projeto .....</b>	<b>21</b>
<b>REFERÊNCIAS.....</b>	<b>23</b>

## 1 – APRESENTAÇÃO DA TRAJETÓRIA DO ALUNO

Minha trajetória na área da computação se inicia na faculdade, com as cadeiras de Fundamentos de Programação e Programação Orientada à Objetos, cursadas como requisitos prévios no curso de ES, adquiri o conhecimento lógico necessário para poder atuar pela primeira vez na AGES.

Durante o primeiro semestre de 2023, realizei diversos cursos em plataformas de estudos de programação online e conquistei certificados. Como a formação em HTML com duração de 21 horas e o de React com duração de 34 horas. Estes cursos me ajudaram bastante a aprender como funciona o processo de desenvolvimento de um programa em diferentes tipos de linguagens e frameworks, já que na faculdade foi aprendido, até o momento, somente Java.

Não obstante, em maio deste mesmo semestre fui contratado como PJ na empresa SoftKuka para atuar como desenvolvedor *frontend*. Na SoftKuka, tive a oportunidade de adquirir conhecimento diário nas aplicações Web e foi onde eu descobri minha paixão por Engenharia de Software. Além disso, tive a experiência de participar de projetos com clientes externos, prazos a serem cumpridos, entre outras práticas que me adaptaram para o mercado de trabalho.

A proposta da AGES já era algo que via com grande apreço, permitindo uma experiência muito próxima do mercado de trabalho. Dito isso, minha primeira passagem na organização foi através do projeto Veículos Via Montadora. Tendo a oportunidade de obter o contato inicial de trabalhar com um time, fui posto à prova diversas vezes, algo que, como aluno iniciante na área, vi de forma muito positiva. Desenvolvi minhas soft-skills ao procurar ser comunicativo com minha equipe, tive contato com a linguagem Python no *backend* do projeto e apliquei meus conhecimentos de HTML no React, framework utilizado no projeto. Sou muito grato a AGES I por ter adquirido uma equipe que me impulsionou a estudar o curso de React para aplicar no projeto pois, coincidentemente, influenciou diretamente para minha capacidade de cumprir o desafio proposto para entrar no meu emprego.

## 2 – PROJETOS AGES I - “VEÍCULOS VIA MONTADORA”

Esta seção busca apresentar minha passagem como AGES I pelo projeto Veículos via Montadora. Aqui estão descritos os artefatos entregues, a atuação ao longo das sprints e os pontos de melhoria identificados no decorrer do projeto.

### 2.1 Introdução

O projeto Veículos via Montadora tem como objetivo desenvolver um sistema Web que pudesse auxiliar no cadastramento e atualização de informações dos produtos (carros) de uma maneira rápida e efetiva. Todos os dados vêm via PDF, logo, o desafio deste projeto é a organização e a estruturação de um banco de dados, tanto de captação iniciando com a leitura do PDF, como de entrega, exportando os dados para o sistema do cliente.

Inicialmente, o stakeholder registrado na documentação do projeto chamava-se Genaro Passos. Porém, na primeira reunião com o cliente, quem nos foi apresentado chama-se Leonardo Cunha, cujo explicou-nos que o cliente inicial do projeto, Genaro, havia sido desligado da empresa. Desta maneira, o nosso stakeholder oficial é o Leonardo, que nos introduziu a Sinosserra: empresa cliente que trabalha com consórcios de veículos. Nosso objetivo é facilitar o processo de extração de PDF que são enviados, tanto da Sinoscar, quanto do Tramonto, para o nosso cliente.

Posteriormente, foram introduzidos outros 2 *stakeholders* do projeto: Fabiano Longaray e Luana Lima. Ambos foram apresentados na segunda reunião que tivemos, o Fabiano para entender os termos técnicos e a Luana, CEO da Sinosserra e a usuária que iria utilizar o nosso programa na empresa.

A execução do projeto ocorreu no primeiro semestre de 2023, entre as datas 8 de março e 14 de junho, pelos estudantes de Engenharia de Software. Neste projeto, havia 8 AGES I, 4 AGES II, 3 AGES III e 3 AGES IV, totalizando 18 membros da equipe orientados pelo Prof. Daniel Antonio Callegari. A foto do time responsável pelo projeto pode ser vista na Figura 1:

**Figura 1: Time do projeto Veículos via Montadora**



Fonte: Wiki do projeto

## 2.2 Desenvolvimento do Projeto

Esta seção apresenta informações referentes ao desenvolvimento do projeto: localização do código-fonte, banco de dados, protótipos de tela desenvolvidos, arquitetura e tecnologias utilizadas.

### 2.2.1 Repositório do código Fonte do Projeto

O código-fonte do projeto encontra-se distribuído em dois repositórios, separados pelo *frontend* e *backend*:

- *Frontend*: <https://tools.ages.pucrs.br/veiculos-via-montadora/frontend>
- *Backend*: <https://tools.ages.pucrs.br/veiculos-via-montadora/backend>

Além disso, há a wiki do projeto que explica como instalar corretamente estes programas. Se encontra no presente link: <https://tools.ages.pucrs.br/veiculos-via-montadora/wiki/-/wikis/instalacao>.



### 2.2.2 Banco de Dados utilizado

Ao analisar a aplicação como um todo, definimos o banco de dados do projeto como não-relacional e foi escolhida a tecnologia MongoDB para ser utilizada. O link desta seção está disponível para acesso em: [https://tools.ages.pucrs.br/veiculos-via-montadora/wiki/-/wikis/banco\\_dados](https://tools.ages.pucrs.br/veiculos-via-montadora/wiki/-/wikis/banco_dados).

### 2.2.3 Arquitetura utilizada

Informações sobre o *frontend*, *backend*, Infraestrutura e orçamento AWS podem ser acessados na seção arquitetura da *wiki* em: <https://tools.ages.pucrs.br/veiculos-via-montadora/wiki/-/wikis/arquitetura>.

### 2.2.4 Protótipos das telas desenvolvidas

Buscamos seguir o mesmo padrão de aparência em todas as telas realizadas, a figura 2 apresenta um dos principais *mockups* que desenvolvi, junto de colegas da equipe.

**Figura 2: Tela de Extração do PDF**



Fonte: *Wiki* do projeto

Outros *mockups* desenvolvidos podem ser encontrados na página de *mockups* da *wiki*, localizado no link a seguir: <https://tools.ages.pucrs.br/veiculos-via-montadora/wiki/-/wikis/mockups>

## 2.2.5 Tecnologias Utilizadas

Para o lado do *frontend*, foi utilizado o React, biblioteca JavaScript de código aberto com foco em criar interfaces de usuário em páginas web que precisam ser atualizadas em tempo real. Ele é ideal para esse projeto, pois permite que atualize a interface de usuário de forma rápida e eficiente.

O desenvolvimento do código em React foi realizado com TypeScript: linguagem de programação de código aberto desenvolvida pela Microsoft. É um superconjunto sintático estrito de JavaScript e adiciona tipagem estática opcional à linguagem, além disso, sua estilização foi implementada através do Material UI/Design: linguagem de design desenvolvida pela Google.

Não obstante, preferimos usar o Swagger para utilizar as APIs necessárias no *frontend*: linguagem de descrição de interface para descrever APIs RESTful expressas usando JSON. É usado junto com um conjunto de ferramentas de software de código aberto para projetar, construir, documentar e usar serviços da Web RESTful.

Do lado do *backend*, optamos pelo Python: linguagem de programação de alto nível com sintaxe mais simplificada e próxima da linguagem humana, utilizada nas mais diversas aplicações, como desktop, web, servidores e ciência de dados. Para seu desenvolvimento, utilizamos o framework FastAPI, assim como Poetry e Pytest para gerenciamento de dependências e testes automatizados, respectivamente.

Para a montagem do banco de dados do projeto, foi realizado com o MongoDB: software de banco de dados NoSQL orientado a documentos livre, de código aberto e multiplataforma, escrito na linguagem C++.

No quesito infraestrutural do projeto, usamos tanto o Docker, quanto AWS. Docker é um conjunto de produtos de plataforma como serviço que usam virtualização de nível de sistema operacional para entregar software em pacotes chamados contêineres. Já a Amazon Web Services, também conhecido como AWS, é uma plataforma de serviços de computação em nuvem, que formam uma plataforma de computação na nuvem oferecida pela Amazon.com.

## 2.3 Atividades desempenhadas pelo aluno no projeto

Nesta seção, estão todas as atividades individuais feitas por mim durante o projeto, assim como as ferramentas utilizadas e contribuições. Estão separadas por *sprints*, que são períodos de mais ou menos 2 semanas que, em cada uma, são determinados objetivos e entregas para o final dela.

### 2.3.1 Sprint 0

Após a primeira reunião com o stakeholder, iniciou-se a *sprint* 0, que se consistia em compreender o projeto, escolher as melhores linguagens e frameworks, além de estudarmos estas tecnologias para conseguirmos desenvolver com êxito.

Compreendi que o principal desafio do projeto seria transformar o PDF que era lido por humanos para uma máquina. Não obstante, precisaria adaptar para arquivos que eram padronizados de duas maneiras diferentes, já que a Sinoscar e a Tramonto são empresas diferentes, possuem distintas maneiras de padronização de PDF.

A partir disto, juntamente com a equipe, realizei uma pesquisa que totalizou a procura de 8 ferramentas diferentes, a fim de encontrar uma que seja a ideal para uma leitura de PDF. Encontramos o Tabula, ferramenta para liberar tabelas de dados bloqueados em arquivos PDF. Assim, era extraído estes dados em uma planilha CSV ou Microsoft Excel.

Posteriormente, participei da primeira reunião com a equipe para discutirmos quais linguagens e frameworks usaríamos. Debates sobre as possibilidades entre Python e Java para o *backend* e, inicialmente, foi decidido que seria feito em Java, além do framework Spring Boot.

Em relação a tecnologias no *frontend*, havia a discussão entre React, Next e Vue, mas prontamente foi decidido que seria React pela votação da equipe, não foi gerada muita discussão.

Apesar de tudo ser novidade, entrei no projeto com a mentalidade de sair da zona de conforto e, por ter aprendido e desenvolvido Java nos últimos semestres da faculdade, não queria desenvolver novamente na linguagem.

Portanto, respaldei minha vontade para a equipe que gostaria de desenvolver o *frontend* deste projeto e me comprometi tanto a estudar e aprender, quanto a

contribuir positivamente no projeto. Felizmente, eu já havia realizado o curso de HTML e, por conta disso, já obtinha a mínima noção para aprender React. Posteriormente, enquanto meus colegas que já possuíam conhecimento montavam o Figma, intensifiquei meus estudos em CSS, além do React e Typescript para começar a *sprint* 1 podendo ter condições de programar.

Na última reunião desta *sprint*, houve uma discussão pela possível demora da execução do programa *backend* com o Java. Então, foi abordado a possibilidade de começarmos a usar 2 linguagens (Java e Python). Entretanto, percebemos que facilitaria muito se houvesse apenas Python, pois, além de Java não ser tão performático com *lambda* em uma comparação, Python é a melhor linguagem para extração de PDF graças ao Pandas, sem contar na complexidade a mais que seria se houvesse que aprender a fazer em 2 linguagens diferentes.

*Hoje em dia vejo que um grande problema causado nas sprints posteriores foi causada por um erro nesta sprint: indecisão em relação ao Back-End. O escopo do projeto estava confuso, por mais que eu soubesse que seria usado o Python e deveria estudar o Pandas, não sabia se ele realmente seria usado.*

*Neste ponto, acredito que pequei na minha habilidade de comunicação, por mais que eu fosse do frontend nesse começo, gostaria de entender o que aconteceria no backend para futuramente participar dele também.*

### 2.3.2 Sprint 1

Com o início da segunda *sprint* do projeto, participei da apresentação da *User Stories*, juntamente com os *Mockups* produzidos no Figma na reunião com o stakeholder Fabiano. Ele gostou muito da apresentação, surpreendeu-se positivamente com o resultado e principalmente com a estilização que foi proposta. Por termos recebido um *feedback* tão positivo, tanto eu, quanto a equipe, achamos que estávamos no caminho certo.

Finalmente, foi decidido que usaríamos o Tabula-py para extrair os PDFs, de acordo com os AGES III, ele seria mais “inteligente” para o tipo de PDF da Sinosserra.

Não obstante, foram definidas as *squads* da equipe e, em consenso da equipe, a função determinada para minha *squad* era começar o desenvolvimento do *frontend*.

Graças aos estudos feitos durante a primeira sprint, participei ativamente e ajudei a implementar a tela para baixar o arquivo PDF, além da opção de escolha de grupo de veículos e análise deles.

Esta sprint, devido ao mau planejamento dos gerentes do projeto da equipe (AGES IV), não houve muitas sessões de desenvolvimento do código e, por conta disso, minha *squad* foi a única que se reuniu para programar. As consequências disso foram grandes: na apresentação do projeto, não conseguimos alcançar os objetivos propostos no começo da Sprint e isso preocupou os stakeholders, ligando um sinal de alerta para toda a equipe.

**Figura 3: Descrição e Status da User Stories da Sprint 1**

### Sprint 1

- Começo: 29/03/2023
- Fim: 12/04/2023
- Apresentação para os Stakeholders: **10/04/2023**

US	Descrição	Status
US01	Tela de Envio de PDFs	<b>Não Aceito</b>
US03	Tela de Extração de Dados	<b>Não Aceito</b>
US04	Tela de Conclusão da Extração	<b>Não Aceito</b>

Fonte: Wiki do projeto

*Por mais que esta sprint tenha sido um desastre, eu fiquei muito feliz em ter participado, pela primeira vez, do desenvolvimento de um projeto com responsabilidades e colegas de equipe. Trabalhar utilizando os métodos que estudei nos últimos 2 semestres me inspirou a dar mais valor aos ensinamentos das cadeiras do curso. Ressaltei este ponto no discurso final do projeto com a equipe e repito: foi um momento muito especial para mim e creio que nessa sprint eu tive a certeza de que gosto de ser um desenvolvedor de software.*

### 2.3.3 Sprint 2

Por conta do fracasso das entregas estipuladas na *sprint* 1 para o *stakeholder*, foi reorganizado a *User Stories* e as *tasks* para cada equipe. Continuei no mesmo *squad*, porém, nesta etapa foi determinado que os AGES I são os responsáveis de desenvolver as *tasks*. Portanto, fiquei responsável de entregar duas tarefas:

- Montadora do PDF – Identificar qual montadora o PDF pertence, tanto pelo nome do arquivo, quanto pela possibilidade de o usuário selecionar.
- Preparação para extração - Com o recebimento do PDF pelo *frontend*, selecionar que tipo de Arquivo (MEV, Jeep, Outro) para selecionar o tópico de algoritmos de cada tipo.

Graças as intensivas tentativas do Arthur Ibarra (AGES III), descobrimos que não há padronização em nenhum dos 20 PDFs que nos foi disponibilizado da Chevrolet (MEV), ou seja, não tem como fazer um sistema para isso. Em uma decisão coletiva, optamos por descartar o MEV do escopo do projeto.

Durante o andamento da *sprint* me envolvi em uma discussão com o meu *squad*: tanto o Luiz (AGES IV), quanto o Kevin (AGES III), se reuniram com o Lucas Susin (AGES III de outro *squad*) sem a presença de nenhum AGES I para desenvolver a *task* da Montadora do PDF. Eu questionei o porquê da realização da tarefa em que eu era o responsável, sem a minha presença ou a de qualquer outro AGES I para aprender, sendo nós os responsáveis pelo desenvolvimento. Ambos se defenderam alegando que era aceitável o meu ponto, porém, era necessário codificar quando houvesse disponibilidade. Foi proposto por mim, na reunião seguinte, termos sessões de codificação todas as quintas e em um dia do fim de semana.

*Observando este atrito hoje em dia, fico orgulhoso do meu posicionamento. Pensei muito se deveria discutir com os AGES veteranos, estava inseguro em abrir o debate de terem feito algo errado. Contudo, o respaldo da Carolina e do Gabriel (outros AGES I do squad) me fizeram crer que o meu ponto de argumentação estava correto.*

*De fato, estava. Meus colegas concordaram comigo e a proposta gerou uma nítida melhora no rendimento da equipe depois do combinado das reuniões.*

### 2.3.4. Sprint 3

Durante a terceira *sprint* o time resolveu se dividir em equipes ainda menores, a fim de acelerarmos o processo de desenvolvimento das *tasks* devido ao problema encontrado no planejamento: os AGES IV separaram as tarefas de cada *squad* 5 dias depois do início da *sprint*, nos gerando uma grande quantia de afazeres em um curto período.

Finalmente, as *tasks* em que fiquei responsável eram:

- Criar o componente *input* de dados
- Integrar a estrutura base com os inputs
- Criar o botão de salvar

Eu senti uma grande insegurança de programar nesta *sprint*, vendo o programa encorpado, não me senti confiante de desenvolver. Pensei que era incapaz e que não possuía o conhecimento suficiente, podendo implicar erros no código. Por mais que a *sprint* tenha sido um sucesso na entrega, acredito que tenha sido a pior em questão de desempenho individual.

O ápice desta *sprint* ocorreu em apenas 1 dia: dos 18 integrantes da equipe, 10 participaram de uma ligação que aconteceu durante a madrugada do dia de apresentação ao *stakeholder*. Nesta reunião, desenvolvemos todas as *tasks* que faltavam e, por mais que tenha dado certo, foi uma das piores experiências que passei na AGES.

*A “cultura do herói” que o professor Callegari enfatizou no início do projeto para não fazermos, acabou sendo feita nessa sprint. Por mais desgastante que tenha sido dormir em ligação implementando funcionalidades, foi nesta reunião que me aproximei de muitos membros da equipe e o próprio professor percebeu que a equipe tinha amadurecido muito com este fato.*

*Acredito que eu poderia ter pedido o auxílio de algum colega para vencer minha insegurança de desenvolver nesta sprint. Pequei na comunicação e, assistindo o Felipe (AGES I) desenvolver enquanto todos os outros colegas auxiliavam ele, me encorajou novamente a programar.*

### 2.3.5. Sprint 4

Na última *sprint* o projeto entrou em fase final de desenvolvimento, com o time focado em finalizar o máximo possível para a última entrega da aplicação. Levando as lições aprendidas de planejamento da quarta *sprint*, as *tasks* foram determinadas com antecedência para termos maior período de desenvolvimento.

Em conjunto, o time decidiu acabar com as *squads*. Pois, devido aproximação do grupo na reunião da última *sprint*, percebemos a união do time e a separação entre subequipes só prejudicaria o processo.

A *task* que escolhi realizar nesta última etapa era a exportação de dados do PDF para JSON e CSV, além de realizar a integração. Como eu nunca fiz a integração do *frontend* com o *backend* de um projeto, tive várias dúvidas de como implementar, porém, em nenhum momento senti a insegurança de desenvolver, como na última *sprint*.

Consegui implementar a *task* perfeitamente, desenvolvendo com os meus conhecimentos adquiridos no decorrer do processo e, as dúvidas pertinentes, eram solucionadas com auxílio do Felipe (AGES I). Felipe foi uma peça extremamente importante para o meu desenvolvimento como programador neste projeto.

Ademais, as outras *tasks* foram feitas continuamente pelos outros integrar e, infelizmente, não conseguimos implementar o funcionamento do botão de salvar a tempo.

Durante a apresentação, foi apontado dúvidas e detalhes finais para serem acertados antes da entrega final. Dentre elas, cometi um erro ao escolher o ícone de exportação de dados para JSON ou CSV que confundiu o *stakeholder*.

## 2.4 CONCLUSÃO

Minha primeira passagem pela AGES através do projeto Veículos via Montadora foi diferente de qualquer experiência que tive até então, o que resultou em diversos aprendizados valiosos. Pude ter uma experiência mais próxima ao que devo encontrar no mercado de trabalho com uma equipe multidisciplinar.



Apesar de eu ter facilidade em me comunicar com as pessoas, tinha receio se isso valeria para o lado profissional. Neste projeto, pude transformar esta facilidade em *Soft Skills* que contribuíram muito com o meu crescimento profissional, visto que, foi fundamental para eu conquistar o meu primeiro emprego.

Ademais, acredito que meu desempenho técnico foi insuficiente comparado ao que desejava ter realizado. Porém, meu conhecimento adquirido neste período para realização das tarefas superou minhas expectativas. Antes da AGES, obtinha conhecimento de como implementar apenas alguns componentes de um site, como um texto ou link. Agora, graças ao meu estudo e prática nas *sprints*, tenho a capacidade de desenvolver um site inteiro estilizado conforme demanda. Este é o meu maior orgulho deste projeto.

No começo do projeto, apesar de estar nervoso por não saber se realmente desfrutaria da experiência, entrei entusiasmado pela oportunidade de obter uma amostra do que é o mercado de trabalho. Entretanto, nas *sprints* 2 e 3, percebi que deixei de priorizar o andamento do projeto e passei a me preocupar muito mais com as cadeiras adjacentes da faculdade. Acredito que este feito tenha sido consequência da data de entrega de trabalhos e provas coincidirem nas semanas em que se passaram estas *sprints*. Me desconectei muito do projeto e perdi um processo que obtinha muita curiosidade de descobrir: a transição do começo para o acabamento de um projeto. Tirarei esta lição como aprendizado para não cometer novamente quando atuarei como AGES II, III e, IV.

A presença de *stakeholders* reais, que esperam um projeto entregue conforme desejado, mudou minha forma de pensar em relação aos afazeres profissionais e acadêmicos. A partir de agora, darei muito valor a todos os ensinamentos da faculdade que, ocasionalmente, achava que seriam inúteis para mim. Em resumo, todas as cadeiras cursadas até o momento desta AGES obtinham algum conhecimento que foram utilizados e isso me tocou bastante.

Como conclusão final, afirmo que aprendi a me expressar melhor em uma equipe profissional e ser mais resiliente frente aos desafios de um projeto de software. Ademais, reconheço um grande avanço tanto em *soft-skills* como *hard-skills* justaposto ao começo do projeto. Porém, creio que ainda tenho um grande caminho pela frente antes de me tornar um bom Engenheiro de Software.

### **3 – PROJETOS AGES II - “ENSportive – Estilo de Vida Esportivo”**

Esta seção busca apresentar minha passagem como AGES II pelo projeto ENSportive – Estilo de Vida Esportivo. Aqui estão descritos os artefatos entregues, a atuação ao longo das sprints e os pontos de melhoria identificados no decorrer do projeto.

#### **3.1 Introdução**

O projeto ENSportive tem como objetivo desenvolver um programa que inicialmente seria um aplicativo, porém, após conversar e alinhar as ideias com o stakeholder, foi decidido que seria um sistema web. Com o objetivo de ensinar esportes que envolvam raquetes dentro da metodologia Ensportive, o papel designado para nossa equipe era de facilitar o processo da Fernanda e dos professores de administração das aulas, assim como de ajudá-los a trazer mais pessoas para suas aulas em busca de uma vida mais saudável.

O grande desafio do projeto é a implementação do calendário: um componente similar ao Google Calendar, em que será possível marcar/desmarcar aula e administrar o conteúdo de cada aula, além de ser o mais intuitivo possível, assim como o exemplo trazido é.

O stakeholder do projeto chama-se Fernanda Ens, que nos introduziu a ENSportive: escola de raquetes que auxilia pessoas que querem praticar uma atividade física que tenha diversão e aprendizado, bem como auxiliar no emagrecimento e saúde.

A execução do projeto ocorreu no primeiro semestre de 2024, entre as datas 6 de março e 12 de junho, pelos estudantes de Engenharia de Software. Neste projeto, havia 5 AGES I, 6 AGES II, 3 AGES III e 2 AGES IV, totalizando 16 membros da equipe orientados pelo Prof. Rafael Chanin. A foto do time responsável pelo projeto pode ser vista na Figura 4:

**Figura 4: Time do projeto ENSportive**



Fonte: Wiki do projeto

## 3.2 Desenvolvimento do Projeto

Esta seção apresenta informações referentes ao desenvolvimento do projeto: localização do código-fonte, banco de dados, protótipos de tela desenvolvidos, arquitetura e tecnologias utilizadas.

### 3.2.1 Repositório do código Fonte do Projeto

O código-fonte do projeto encontra-se distribuído em dois repositórios, separados pelo *frontend* e *backend*:

- *Frontend*: <https://tools.ages.pucrs.br/ensportive/ensportive-frontend>
- *Backend*: <https://tools.ages.pucrs.br/ensportive/ensportive-backend>

Além disso, há a wiki do projeto que explica sobre todas as tecnologias usadas no projeto, além do que foi implementado. Se encontra no presente link: <https://tools.ages.pucrs.br/ensportive/ensportive-wiki/-/wikis/home>.

### 3.2.2 Banco de Dados utilizado

Ao analisar a aplicação como um todo, definimos o banco de dados do projeto como um modelo relacional e foi escolhida a tecnologia PostgreSQL para ser utilizada. O link desta seção está disponível para acesso em: [https://tools.ages.pucrs.br/ensportive/ensportive-wiki/-/wikis/banco\\_dados](https://tools.ages.pucrs.br/ensportive/ensportive-wiki/-/wikis/banco_dados).

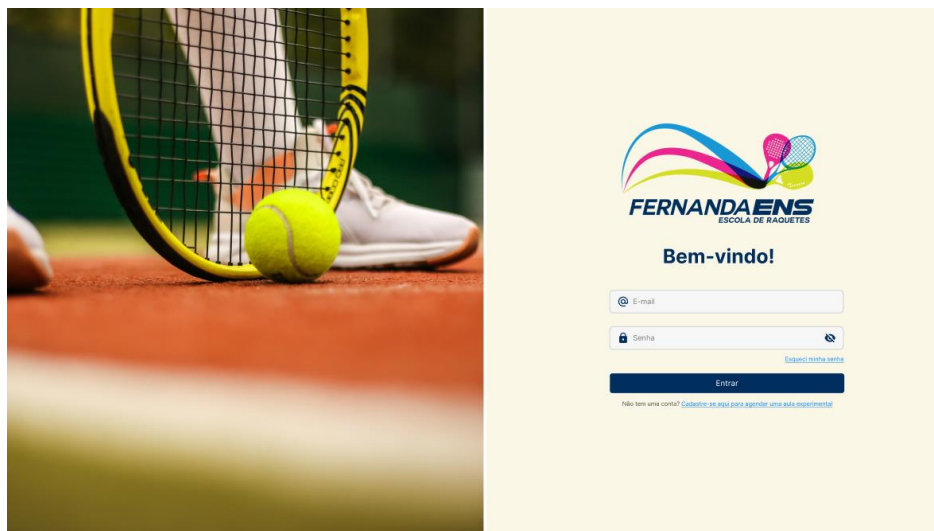
### 3.2.3 Arquitetura utilizada

Informações sobre o *frontend*, *backend*, Infraestrutura e orçamento AWS podem ser acessados na seção arquitetura da *wiki* em: <https://tools.ages.pucrs.br/ensportive/ensportive-wiki/-/wikis/arquitetura>

### 3.2.4 Protótipos das telas desenvolvidas

Buscamos seguir o mesmo padrão de aparência em todas as telas realizadas, a figura 5 apresenta um dos principais *mockups* que desenvolvi, junto de colegas da equipe.

Figura 5: Tela de Extração do PDF



Fonte: Wiki do projeto

Outros *mockups* desenvolvidos podem ser encontrados na página de *mockups* da *wiki*, localizado no link a seguir: <https://tools.ages.pucrs.br/ensportive/ensportive-wiki/-/wikis/mockups>

### 3.2.5 Tecnologias Utilizadas

Para o lado do frontend, foi utilizado o Angular, um framework de desenvolvimento mantido pelo Google e uma comunidade de desenvolvedores. Ele é projetado para criar aplicativos da web de página única (SPA) escaláveis e robustos. É o ideal para criar interfaces de usuário de forma rápida e eficiente, aproveitando seu sistema de componentes reutilizáveis e sua arquitetura baseada em módulos.

O desenvolvimento do código em Angular foi realizado com TypeScript: linguagem de programação de código aberto desenvolvida pela Microsoft. É um superconjunto sintático estrito de JavaScript e adiciona tipagem estática opcional à linguagem, além disso, sua estilização foi implementada através do Material UI/Design: linguagem de design desenvolvida pela Google.

Do lado do backend, optamos pelo Java com a ferramenta Spring Boot: um framework de desenvolvimento de aplicações Java que oferece uma abordagem rápida e simplificada para criar aplicativos robustos e escaláveis. Com foco na produtividade, o Spring Boot elimina a necessidade de configuração manual, oferecendo padrões e convenções predefinidos. Ele é ideal para o desenvolvimento de aplicativos da web que requerem atualizações em tempo real.

Para a montagem do banco de dados do projeto, foi usado o PostgreSQL: um sistema de gerenciamento de banco de dados relacional de código aberto, conhecido por sua confiabilidade, robustez e recursos avançados. Com sua capacidade de lidar eficientemente com atualizações em tempo real, o PostgreSQL é uma escolha sólida para armazenar e gerenciar dados em aplicações web e móveis que exigem uma experiência de usuário dinâmica e responsiva.

No quesito infraestrutural do projeto, usamos tanto o Docker, quanto AWS. Docker é um conjunto de produtos de plataforma como serviço que usam virtualização de nível de sistema operacional para entregar software em pacotes chamados contêineres. Já a Amazon Web Services, também conhecido como AWS, é uma plataforma de serviços de computação em nuvem, que formam uma plataforma de computação na nuvem oferecida pela Amazon.com.

### 3.3 Atividades desempenhadas pelo aluno no projeto

Nesta seção, estão todas as atividades individuais feitas por mim durante o projeto, assim como as ferramentas utilizadas e contribuições. Estão separadas por *sprints*, que são períodos de mais ou menos 2 semanas que, em cada uma, são determinados objetivos e entregas para o final dela.

#### 3.3.1 Sprint 0

Após a primeira reunião com o stakeholder, iniciou-se a *sprint* 0, que se consistia em compreender o projeto, escolher as melhores linguagens e frameworks, além de estudarmos estas tecnologias para conseguirmos desenvolver com êxito.

Compreendi através das duas reuniões a proposta do stakeholder, juntamente com o maior desafio do projeto: um sistema de aulas em esportes de raquetes, focada em alunos agendarem e cancelarem suas aulas através do calendário, com um administrador que controla tudo isso.

Decidimos as tecnologias do projeto: Java no *backend*, Angular no *frontend* e PostgreSQL no Banco de Dados. Tive muitas discordâncias com os Ages III sobre a tecnologia tanto para o backend, quanto para o frontend. No backend, defendi a ideia de usar NodeJS, pois seria uma grande oportunidade para todos aprenderem o javascript, uma linguagem muito usada no mercado de trabalho. Discordaram de mim e escolheram o Java, com o argumento de equilibrar o número de colegas que aprenderão com os que já sabem e podem ajudar. No frontend, defendi o uso de React no projeto, usando este mesmo argumento: eu possuía conhecimento e poderia ajudar todos que não conheciam, que representava a grande maioria da equipe. Novamente, discordaram e escolheram Angular, buscando segurança no programa e, divergindo nos argumentos, a aventura de aprendermos todos juntos Angular, pois ninguém no projeto obtinha conhecimento. Tivemos dificuldades da escolha da tecnologia para o banco de dados, apesar de querermos muito a experiência de um Banco de Dados com MongoDB, o diagrama de entidades retratou que o banco de dados relacional é o que melhor se encaixa para o projeto.

Minha contribuição nesta *sprint* foi baseado no Figma e no Banco de Dados: padronizei os estilos e cores para todos os integrantes que contribuíram no Figma

possuírem uma base, além de contribuir na página de login e introduzir a Landing Page. Também ajudei o Henrique Juchem, ages I do projeto a continuar a implementação da Landing Page, foi a primeira vez que exerci minha função de ages II, ajudando um colega que possuía menos conhecimento. Além disso, fiz o diagrama de entidades no Banco de Dados, identificando as relações entre os elementos que contribuiu na escolha de ser um modelo relacional.

### 3.3.2 Sprint 1

Com o início da segunda *sprint* do projeto, participei ativamente da escolha do PostgreSQL como tecnologia do banco de dados. Não consegui participar da modelagem em si, havia um colega que estava muito decidido do que queria fazer e, consequentemente, não estava aceitando ideias contrárias. Eu estava com muita vontade de fazer o banco de dados do projeto e isso me deixou bastante frustrado, mas serviu de aprendizado de como lidar com diferentes tipos de personalidades em um projeto.

Assim como na ages I, a equipe foi dividida em *squads* e eu fui o único integrante da minha *squad* que possuía interesse no *backend*. Felizmente, com os conhecimentos obtidos no *frontend* realizado na ages I, tive capacidade de ajudar meus colegas no *frontend*. Desta vez, senti dificuldades de explicar corretamente o código e acredito que a responsabilidade de ajudar os colegas como ages II contribui muito para a comunicação, também é importante saber explicar para transmitir o conhecimento e notei nesta *sprint* como estou em constante melhora em relação a isso.

Não obstante, com ajuda do Pedro Carlucci e da Laura Caetano implementei a lógica de criação, leitura, edição e exclusão (CRUD) de professores no *backend*. Tive muita dificuldade de fazer esta tarefa que é simples por conta da minha inexperiência, mas ao mesmo tempo em que fui ajudado pelos ages III, aprendi como funciona e me sinto muito animado para implementar mais códigos no *backend*.

## REFERÊNCIAS

AGES. **Veículos via Montadora**. Disponível em:

<<https://tools.ages.pucrs.br/veiculos-via-montadora>>. Acesso em: 13 jun. 2023.

AMAZON. **AWS**. Disponível em <<https://aws.amazon.com/>>. Acesso em 14 jun. 2023.

ANGULAR. **Angular**. Disponível em: <<https://angular.io/>>. Acesso em 10 abr. 2024.

DOCKER. **Docker**. Disponível em <<https://www.docker.com/>>. Acesso em 14 jun. 2023.

JAVA. **Java**. Disponível em <<https://www.java.com/>>. Acesso em 10 abr. 2024.

MATERIAL UI/DESIGN. **Material UI**. Disponível em <<https://material-ui.com/>>. Acesso em 14 jun. 2023

MONGODB. **MongoDB**. Disponível em <<https://www.mongodb.com/>>. Acesso em 14 jun. 2023.

POSTGRESQL. **PostgreSQL**. Disponível em: <<https://www.postgresql.org/>>. Acesso em 10 abr. 2024.

PYTHON. **Python**. Disponível em: <<https://www.python.org/>>. Acesso em 14 jun. 2023.

REACT. **React**. Disponível em: <<https://reactjs.org/>>. Acesso em 14 jun. 2023.

SPRING BOOT. **Spring Boot**. Disponível em <<https://spring.io/>>. Acesso em 10 abr. 2024.



SWAGGER. **Swagger**. Disponível em <<https://swagger.io/>>. Acesso em 14 jun. 2023.

TYPESCRIPT. **TypeScript**. Disponível em <<https://www.typescriptlang.org/>>. Acesso em 14 jun. 2023.