

# Desenvolvimento de um sistema web para gerenciamento das qualificações dos servidores públicos da Secretaria de Estado da Justiça de Rondônia

Caio Vitor Lima Brito<sup>1</sup> Leandro Ferrarezi Valiante<sup>2</sup>

<sup>1</sup>Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO

caio.vitor@estudante.ifro.edu.br

**Resumo.** *A qualificação contínua é um mecanismo fundamental para o desenvolvimento e melhoria das competências profissionais, resultando em um melhor desempenho diante das funções laborais. Nesse contexto, a Escola Estadual de Serviços Penais (ESEP), responsável pela formação inicial e capacitação continuada dos servidores públicos da Secretaria de Estado da Justiça (SEJUS) de Rondônia, exerce esse papel por intermédio de cursos de capacitação e palestras regulares. Diante dessa demanda, este trabalho tem como objetivo criar um sistema web para auxiliar no gerenciamento das qualificações destes servidores, proporcionando maior praticidade e consistência. A saber, no desenvolvimento desta solução tecnológica, foi utilizado tecnologias e ferramentas atuais e amplamente utilizadas, combinadas com um entendimento sólido dos requisitos funcionais, validação da modelagem do sistema e uma abordagem de desenvolvimento modular, que possibilitaram implementar uma aplicação com interface que oferece uma experiência de uso mais fluída e agradável, além de um backend organizado e robusto. O produto final deste trabalho é o sistema web funcional, que promove um avanço significativo para otimizar a eficiência na gestão dos cursos aplicados pela ESEP.*

**Palavras-chave:** *Sistema web. Desenvolvimento. Gerenciamento. Tecnologias.*

## 1. Introdução

A Secretaria de Estado da Justiça (SEJUS) é um órgão do governo de Rondônia que teve a sua criação através do Artigo 1º da Lei Estadual Complementar nº 412, de 28 de dezembro de 2007<sup>1</sup>, sendo esta responsável pela coordenação e execução das políticas públicas relacionadas à área da Justiça no estado de Rondônia, a qual possui como principais atribuições a administração do sistema penitenciário do estado, elaboração e execução das políticas de administração penitenciária, administração orçamentária e financeira dos recursos destinados à Secretaria de Justiça, a execução penal e a reinserção social.

De acordo com o Portal da Transparência do Estado de Rondônia, no mês de julho de 2023, haviam um total de 1.005 servidores ativos na SEJUS, dentre estes psicólogos, médicos, enfermeiros, engenheiros civis, contadores, administradores, técnico em informática, policiais penais, entre outros. Vale ressaltar ainda que nesse total de servidores, 882 são policiais penais [RONDÔNIA, 2023a].

<sup>1</sup>[http://ditel.casacivil.ro.gov.br/COTEL/Livros/Files/LC412\\_compressed.pdf](http://ditel.casacivil.ro.gov.br/COTEL/Livros/Files/LC412_compressed.pdf). Acesso em 10/08/2023.

Visando a qualificação contínua dos profissionais da SEJUS, a Escola Estadual de Serviços Penais (ESEP), a qual é responsável pela formação inicial e capacitação continuada dos servidores da SEJUS, exerce esse papel por intermédio de cursos de capacitação e palestras regulares, via editais internos. Destaca-se que a contínua qualificação é um mecanismo fundamental para o desenvolvimento e melhoria das competências profissionais, o que resulta diretamente na melhoria do desempenho das atividades do ofício [Bomfim, 2012].

No decorrer do ano de 2022, a ESEP disponibilizou um total de 1.757 vagas, distribuídas entre 50 cursos de capacitação [RONDÔNIA, 2023b] para servidores da SEJUS. Diante dessa demanda, vê-se a importância de efetuar o controle e a gestão dos servidores inscritos nos cursos, bem como daqueles que efetivamente compareceram às atividades propostas. Tal controle teria por objetivo colaborar na emissão das certificações de conclusão ao término das formações, e quando necessário, inclusive para a emissão de segunda via de certificados mediante confirmação da participação do servidor.

Uma das formas de lidar com o problema exposto, pode ser o desenvolvimento de um sistema capaz de auxiliar na gestão da problemática abordada, o que de acordo com Kroenke [2017], o uso de sistemas de informação gerenciais viabiliza as empresas a alcançarem suas metas e objetivos. Seguindo essa afirmação, a aplicação possibilitaria a automação de fluxos para geração dos certificados, o facilitamento dos processos de cadastro de cursos, além da vinculação dos servidores envolvidos.

Por consequência da adesão e proveito da possível solução retratada acima, vislumbra-se a possibilidade da otimização de recursos humanos, pois vê-se que para desempenhar as funções mencionadas anteriormente seria necessário a participação de múltiplas pessoas, gerando assim outras demandas, como a coordenação efetiva da equipe, além dos custos envolvidos. A informatização da problemática apresentada, poderia viabilizar a integração interna de processos e áreas, o aumento da produtividade e a melhoria da qualidade dos processos [Albertin e Albertin, 2008], além da redução no quantitativo de pessoas para executar essas demandas de forma eficiente, resultando desta maneira na redução de custos e esforços envolvidos para gerir os cursos ofertados e na posterior emissão dos certificados.

Como exemplo de uma ferramenta que gerencia certificados pode-se citar o Sistema de Gestão de Certificados Eletrônicos<sup>2</sup> (SGCE), o qual é gratuito e viabiliza a criação, a gestão de eventos e cursos, incluindo a validação de certificados eletrônicos. Entretanto, o sistema desenvolvido e proposto por este trabalho possui algumas vantagens. Primeiramente, destaca-se a escolha de ferramentas atualizadas e amplamente utilizadas no cenário atual para o desenvolvimento de sistemas. Essa escolha não apenas reduz a probabilidade de obsolescência precoce, mas também facilita a manutenção contínua da aplicação, uma vez que tais ferramentas fazem parte do *know-how* dos desenvolvedores da SEJUS. Além disso, o sistema proposto oferece funcionalidades específicas que o SGCE não possui, como a capacidade de vincular servidores a cursos, permitir o cadastro e a manipulação de assinaturas, e possibilitar a emissão de relatórios tanto para os cursos cadastrados quanto para os servidores.

---

<sup>2</sup><https://softwarepublico.gov.br/social/sgce>. Acesso em 12/08/2023

## 2. Objetivos

O objetivo geral deste trabalho é desenvolver uma aplicação web com o intuito de melhorar a gestão e o controle de fluxos para cadastros de cursos, vinculação de servidores aos cursos e emissão de certificados, proporcionando um melhor desempenho diante das demandas para qualificação dos servidores da SEJUS.

### 2.1. Objetivos Específicos

Para atingir o objetivo geral, foram definidos os seguintes objetivos específicos:

- Realizar a coleta de dados para a definição das funcionalidades do sistema;
- Realizar a prototipagem das telas do sistema; e
- Desenvolver o sistema utilizando tecnologias de desenvolvimento para web.

O próximo capítulo comenta as definições de tecnologias e seus conceitos fundamentais.

## 3. Tecnologias, Conceitos e *Frameworks*

Com a intenção de entregar uma solução que utilize tecnologias e ferramentas atuais, e que são amplamente utilizadas, optou-se pela escolha das técnicas e tecnologias descritas neste capítulo.

### 3.1. Prototipagem

A prototipagem é compreendida como um mecanismo fundamental no processo de modelagem e desenvolvimento de projetos, permitindo transpor as ideias do mundo imaginário para o real, favorecendo a materialização, visualização, completo entendimento do projeto e validação das ideias. Adicionalmente, corroborando com tal argumentação, Theis et al. [2021] afirma que a prototipagem pode produzir informação e gerar conhecimento, além de favorecer a visualização e materialização das ideias com maior amplitude e fidelidade.

De forma complementar, segundo Soares e Resende [2017], a utilização da prototipagem permite inúmeras vantagens ao processo de desenvolvimento de software, viabilizando além da compreensão bem fundamentada das necessidades dos usuários, a redução de riscos relacionados às possíveis mudanças de requisitos, a qual pode afetar diretamente na interface do sistema, prejudicando a experiência do usuário final.

Visando compreender as necessidades dos usuários, validar as ideias e entregar um sistema que possua usabilidade, interface agradável e que seja capaz de gerar valor, o software escolhido para realizar a prototipagem do sistema é o Figma<sup>3</sup>, que de acordo com a empresa mantenedora, favorece a criação de protótipos realistas, com fluxos interativos, aumentando a eficiência na criação do design, além de permitir o trabalho colaborativo em tempo real. Outra vantagem da utilização deste software não está somente no eixo da criação de protótipos, mas também na disponibilidade de ferramentas em apoio ao desenvolvedor, como as especificações do *layout*, espaçamentos, tamanhos, bordas, e cores utilizadas na criação do design, facilitando no desenvolvimento do *front-end* do sistema.

---

<sup>3</sup><https://www.figma.com>. Acesso em 15/08/2023

### 3.2. *Hypertext Markup Language (HTML)*

De acordo com Silva [2018], HTML é uma linguagem para marcação de hipertexto desenvolvida em 1992 por Tim Berners-Lee, que tem exclusivamente por objetivo estruturar documentos web permitindo a comunicação com os navegadores, retirando de sua atribuição ou função a apresentação estilizada ou dinâmica do documento. Ainda seguindo a linha de raciocínio do autor, é crucial possuir conhecimento em HTML como requisito fundamental para a criação de páginas web. O autor destaca que a HTML possui seus próprios termos, palavras, regras de sintaxe e formatação, os quais devem ser rigorosamente observados para garantir um entendimento perfeito por parte do navegador [Silva, 2018].

Corroborando com a ideia disposta no parágrafo anterior, Andrei L. [2023b] expõe que um hipertexto é um texto utilizado para referenciar outros textos, ao passo que uma linguagem de marcação é constituída por uma série de marcações que indicam aos servidores da web o estilo e a estrutura de um documento. Ele também menciona que o HTML é um padrão oficial da internet, constituindo a maior parte dos aplicativos online e páginas web.

### 3.3. *Cascading Style Sheets (CSS)*

A linguagem CSS, que traduzindo para o português significa Folha de Estilo em Cascata, foi desenvolvida em 1996 pela *World Wide Web Consortium (W3C)* com o intuito de ficar responsável pela estilização e formatação dos elementos escritos em uma linguagem de marcação, ou seja, pela representação visual da página web, característica essa que demonstra uma divisão de responsabilidade e, conseqüentemente uma forte união entre CSS e HTML, sendo a HTML o alicerce de um site e o CSS responsável pela estética visual [Ariane G., 2022].

Adicionalmente, conforme destaca Silva [2018], a responsabilidade pela formatação visual de um documento HTML, incluindo aspectos como dimensões, posicionamentos, tamanhos e cores dos elementos, recai sobre o CSS. O autor também aborda as regras essenciais de sintaxe que devem ser seguidas para a correta estilização do documento, compreendendo seletor, declaração, propriedade e valor, além de informar sobre a existência de inúmeras propriedades CSS e seus respectivos valores para estilizar cada aspecto dos seletores.

Nesse sentido, e buscando reduzir o tempo no processo de desenvolvimento do sistema, com ênfase na qualidade e efetividade, foi utilizado o *framework* Bootstrap<sup>4</sup>, que provê estruturas prontas de CSS para o desenvolvimento de sites e aplicações responsivas que atendem todos dispositivos, evitando gasto desnecessário de tempo na escrita de diversas linhas de código [Bootstrap, 2023].

### 3.4. *JavaScript*

Sobre a sua origem, Flanagan [2012] aponta que o JavaScript foi criado pela Netscape (empresa de tecnologia que existiu na década de 90 e pioneira na criação de navegadores web, como o já descontinuado Netscape Navigator) durante os estágios iniciais da Web. Com a intenção de estabelecer um padrão para essa nova linguagem, a Netscape optou por

---

<sup>4</sup><https://getbootstrap.com/docs/5.0/getting-started/introduction/>. Acesso em 06/09/2023

unir forças com a *ECMA International* (Associação Europeia dos Fabricantes de Computadores). Essa colaboração levou à criação de um novo padrão de linguagem denominado *ECMAScript*. Embora *ECMAScript* seja o nome técnico oficial dessa tecnologia, o termo *JavaScript* é mais amplamente reconhecido e utilizado.

Para Flanagan [2012], JavaScript é a linguagem de programação Web amplamente utilizada na maioria dos sites modernos e todos os navegadores atuais — em *desktops*, consoles de jogos, *tablets* e telefones — incluem intérpretes JavaScript, tornando-a a linguagem de programação mais ubíqua da história.

Ainda segundo Flanagan [2012], JavaScript é uma linguagem de alto nível, dinâmica, interpretada e não tipada, que atende tanto ao paradigma de programação funcional quanto orientada a objetos (POO). Esta linguagem de programação tem sua sintaxe derivada do Java, mas vale um adendo, o JavaScript é completamente diferente da linguagem Java, pois o mesmo já deixou para trás suas raízes como linguagem de script, tornando-se uma linguagem de uso geral robusta e eficiente.

### 3.4.1. Biblioteca React

Em síntese, o React é uma biblioteca JavaScript de interface gráfica (GUI) de código aberto, amplamente usada para criar interfaces interativas, tornando-as mais agradáveis e eficientes, aprimorando a experiência do usuário. Ele organiza a construção de interfaces em partes reutilizáveis chamadas de componentes, simplificando o desenvolvimento e manutenção do código, além de melhorar a eficiência ao manipular somente partes específicas no momento de atualização do DOM (*Document Object Model*, Modelo de Documento do Objeto), o que resulta em um desempenho mais rápido, e oferece uma abordagem de programação mais previsível e compreensível ao descrever o comportamento da interface em diferentes estados [Meta Open Source, 2023].

Além disso, Andrei L. [2023c] aponta que o React faz uso do JSX (JavaScript Syntax Extension), que é uma extensão de sintaxe JavaScript usada na criação de elementos React, permitindo incorporar código HTML em objetos JavaScript e simplificar estruturas complexas com funções e expressões JavaScript. Além de facilitar a escrita de componentes React, o JSX ajuda a prevenir ataques XSS (*Cross-Site Scripting*)<sup>5</sup>, convertendo valores embutidos em strings no *React DOM*. Essa sintaxe destaca-se na criação de componentes especiais e aplicações de alto desempenho, simplificando a estrutura de codificação de sites e facilitando a renderização de várias funções. A abordagem do React em favor do JSX é motivada pela combinação eficiente de JavaScript e HTML, oferecendo uma extensão de sintaxe que simplificasse a vida dos desenvolvedores com uma configuração limpa e eficiente.

---

<sup>5</sup>O ataque *Cross-Site Scripting* é uma forma de ameaça cibernética que insere códigos maliciosos em páginas da web confiáveis. Este ataque afeta os usuários que navegam no site, pois o código malicioso tem a capacidade de acessar informações sensíveis armazenadas no navegador da vítima.

### 3.4.2. Inertia.js

O Inertia.js é uma biblioteca de roteamento baseada em JavaScript que foi projetado para trabalhar tanto com *frameworks backend*<sup>6</sup> quanto *frameworks* e biblioteca *frontend*<sup>7</sup>, a exemplo do Laravel e React, permitindo facilitar a criação de aplicações web sem as complexidades envolvidas no desenvolvimento de SPA (*Single Page Applications*) - tecnologia que permite a renderização das páginas de um sistema no lado do cliente, evitando a realização de várias requisições ao servidor - como a necessidade de criar e autenticar API RESTful<sup>8</sup> e gerenciamento de estado do lado do cliente. Demonstrando a busca em facilitar a navegação entre a aplicação sem precisar forçar o recarregamento completo das páginas, com isso deixando o sistema mais dinâmico e responsivo [Inertia, 2023].

Adicionalmente, Kinsta [2023] corrobora afirmando que o Inertia.js funciona em sincronia com o *backend*, através da interceptação do evento de um clique em um link enviando uma solicitação XHR (*XMLHttpRequest*) para o servidor, o qual reconhece a solicitação como sendo a do Inertia.js e retornado uma resposta no formato JSON (JavaScript Object Notation, Notação de objeto JavaScript) contendo o nome e os dados do componente JavaScript necessários para a nova página, o que resulta em uma transição suave entre páginas sem a necessidade de carregá-la por inteira novamente, fornecendo uma interação aprimorada, agradável e responsiva para o usuário.

### 3.5. PHP

A linguagem de programação PHP foi criada em 1994 por Rasmus Lerdorf, inicialmente para criar páginas dinâmicas, constituindo-se de scripts em linguagem C, e após passar por evolução, ganhou recursos como a interação com bancos de dados [Dall'Oglio, 2015]. Em 1995, seu código-fonte foi liberado e mais desenvolvedores se envolveram, levando a reescritas e crescimento. Em 1998, Andi Gutmans e Zeev Suraski deram início ao PHP 3, tendo como características a extensibilidade, sintaxe consistente, a possibilidade de conexão com vários banco de dados, novos protocolos, orientação a objetos e uma nova API, passando a atrair diversos desenvolvedores ao PHP e levando a alteração do nome para Hypertext Preprocessor. Estima-se que em 2015, o PHP já estava presente em mais de 80% dos servidores web, tornando-a a linguagem mais utilizada no desenvolvimento web [Dall'Oglio, 2015].

De acordo com a empresa mantenedora PHP GROUP, o PHP é uma linguagem de código aberto amplamente utilizada, reconhecida por sua versatilidade e desempenho ágil, podendo ser utilizado na maioria dos sistemas operacionais, além de possuir suporte à maior parte dos servidores web. Ela se destaca especialmente no desenvolvimento web, permitindo a criação dinâmica de conteúdo e interação com servidores. O aprendizado inicial do PHP é facilitado devido à sua sintaxe acessível, o que a torna uma excelente escolha para iniciantes. No entanto, essa linguagem também oferece recursos avançados, como suporte a programação orientada a objetos, manipulação de bancos de dados e inte-

---

<sup>6</sup>*Backend* é a parte de um sistema responsável por processar as requisições do usuário, gerenciar o banco de dados e entregar as informações solicitadas. Ele cuida da lógica, segurança e integração dos dados.

<sup>7</sup>*Frontend* é a parte de um sistema que o usuário interage diretamente, responsável pela interface visual de um site, aplicativo ou software.

<sup>8</sup>Interface de Programação de Aplicações que dois sistemas de computador utilizam para trocar informações de forma segura pela internet.

gração com uma variedade de protocolos. Portanto, tanto novatos quanto programadores experientes podem se beneficiar do PHP para criar desde sites simples até aplicações web complexas e robustas [PHP Group, 2023].

### **3.5.1. Framework Laravel**

O *framework* Laravel surgiu em 2011 com intuito de ser uma ferramenta mais completa e com sintaxe expressiva para a codificação em PHP, tendo por objetivo facilitar o desenvolvimento web, através da reunião dos melhores pacotes do ecossistema da linguagem, extensa biblioteca de documentação e ainda ofertar recursos considerados robustos, como a injeção de dependência completa, testes de integração e unidade, além de uma camada de abstração de banco de dados expressiva, fazendo com que o desenvolvedor foque mais na criação do que dos mínimos detalhes da codificação [CARDOSO, 2021].

Sobre a camada de abstração de banco de dados, de acordo com Erika Heidi [2023], o Eloquent é um ORM (Object Relational Mapper, mapeador objeto relacional) incluso por padrão na estrutura do Laravel, que atua como um facilitador para manipulação de dados no bancos de dados, permitindo que cada tabela do banco de dados tenha a sua representação através de modelos correspondentes no projeto. Isso significa que o Eloquent age como uma camada intermediária entre o aplicativo e o banco de dados, tornando mais simples a inserção, atualização e exclusão de registros, além de oferece uma interface simplificada para realizar consultas SQL complexas.

### **3.6. MySQL**

O MySQL é um Sistema de Gerenciamento de Banco de Dados (SGBD) de código aberto e gratuito, notável por ser um dos mais populares e amplamente empregado pela comunidade. Ele adota o modelo cliente/servidor, uma arquitetura que facilita a interação entre diferentes componentes de um sistema. A linguagem primária utilizada pelo MySQL é o SQL (*Structured Query Language*), que permite a realização de consultas e a manipulação de dados no banco de dados de maneira eficaz e intuitiva. Além de sua popularidade, o MySQL se destaca por sua segurança, flexibilidade, escalabilidade, desempenho otimizado e facilidade de uso, tornando-o uma escolha confiável para uma ampla gama de aplicações [Andrei L., 2023a].

Ademais, o site DB-Engines [2023] que é uma iniciativa para coletar e apresentar informações através de classificações mensais conforme o nível de popularidade dos SGBDs, corrobora informando que o MySQL ocupa a segunda posição como o mais amplamente utilizado na atualidade, porém a primeira colocação entre os SGBDs gratuitos. Isso se deve, além de sua ampla gama de funcionalidades, a sua robusta comunidade de suporte, o que facilita a resolução de problemas e a obtenção de assistência técnica quando necessário.

## **4. METODOLOGIA**

A metodologia adotada neste trabalho baseia-se em três fatores alinhados ao tema do artigo. O primeiro foca no propósito descritivo, analisando as características e o processo de gestão dos cursos pela ESEP, com ênfase em como a Tecnologia da Informação pode otimizar esses processos. O segundo fator aborda a coleta de dados qualitativa, visando

compreender as necessidades, opiniões e expectativas dos usuários para a implementação de funcionalidades no sistema. Por fim, o terceiro fator refere-se aos procedimentos e técnicas adotadas ao longo do projeto, os quais são descritos a seguir.

Durante todo o processo de desenvolvimento do sistema, que envolveu desde a compreensão do método utilizado para gerenciar os cursos ministrados pela ESEP, a coleta dos requisitos para o desenvolvimento das funcionalidades, as validações tanto da prototipagem quanto das partes desenvolvidas do sistema, até a entrega do produto final, optou-se pela abordagem da entrevista exploratória como método primário de coleta de dados. As entrevistas foram conduzidas com a participação ativa da diretora e dos auxiliares administrativos da ESEP, o Gerente e o Chefe de Desenvolvimento da Gerência de Tecnologia da Informação da SEJUS e o autor do presente trabalho. Esta seleção estratégica de participantes permitiu obter uma compreensão holística e aprofundada das necessidades e requisitos do sistema, alinhando-se de maneira eficaz com os objetivos do projeto.

Após compreender os requisitos e funcionalidades principais coletadas nas entrevistas, optou-se por empregar a técnica de prototipagem para materializar as informações adquiridas. Essa abordagem proporciona a criação de uma modelagem fidedigna do sistema, permitindo visualizar e validar as necessidades dos usuários de maneira eficaz. Além disso, ela favorece o completo entendimento, validação do projeto e a diminuição dos riscos em possíveis mudanças de requisitos. Desta forma, para o desenvolvimento da prototipagem de design, foi escolhido a ferramenta Figma, conforme descrito na Seção 3.1.

Com a aprovação da modelagem, ocorreu a seleção das tecnologias para o desenvolvimento da aplicação, as quais foram baseadas em cinco critérios fundamentais. O primeiro e o segundo critério foram a atualidade e a popularidade das ferramentas, respectivamente. O terceiro e o quarto critério enfocaram a disponibilidade de documentação abrangente e o suporte de uma comunidade robusta, visando facilitar a resolução de possíveis desafios. Isso culminou diretamente no último critério, que se concentrou na curva de aprendizagem a curto e médio prazo para garantir um desempenho eficaz com as ferramentas escolhidas.

Após a seleção das tecnologias e o planejamento para o desenvolvimento, o sistema foi dividido em módulos, seguindo uma ordem de prioridade. O objetivo foi concentrar os esforços no desenvolvimento dos principais escopos do sistema antes de abordar os escopos auxiliares. Cada módulo concluído passava por uma fase de testes de uso, com o intuito de identificar possíveis falhas e implementar as devidas correções. Uma vez concluídos os testes e as correções do módulo, procedia-se à apresentação para os responsáveis do sistema, visando obter a validação da parte finalizada ou identificar a necessidade de modificações.

Para a implementação de diversas funcionalidades no *software*, foram empregados componentes e bibliotecas provenientes tanto da comunidade JavaScript quanto do PHP. Essas ferramentas externas desempenham um papel crucial ao acelerar o processo de desenvolvimento. No entanto, é imperativo possuir um conhecimento prévio para utilizá-las de forma eficiente. A obtenção desse conhecimento foi alcançada por meio da leitura das documentações correspondentes. Além disso, a análise das documentações das di-

versas tecnologias utilizadas no trabalho se mostrou fundamental para o desenvolvimento desta aplicação, permitindo uma compreensão mais profunda das capacidades e integrações possíveis. Esta abordagem metodológica proporcionou um embasamento sólido para a tomada de decisões durante o processo de desenvolvimento, garantindo a eficiência e eficácia do sistema resultante.

## 5. DESENVOLVIMENTO

Após completar as fases iniciais do planejamento da aplicação, que incluíram a obtenção dos requisitos, bem como a modelagem do sistema e a sua validação, ocorreu a etapa de desenvolvimento. Nessa fase, o objetivo foi a construção tanto do *Front-end* quanto do *Back-end* da aplicação. Como resultado, as Seções 5.1, 5.2 e 5.3 demonstram de forma fidedigna ao que foi planejado, os módulos principais que foram requisitados e as funcionalidades exigidas que deveriam conter neles.

Nesta etapa, voltada para a construção da interface interativa da aplicação, optou-se pela combinação das ferramentas JavaScript, utilizando o ReactJS em conjunto com o Inertia. A escolha pelo ReactJS não apenas se justifica por ser uma biblioteca de interface gráfica de implementação simples, proporcionando uma experiência de usuário mais agradável, mas também pela sua capacidade de reutilização de componentes HTML em diferentes páginas da aplicação. A integração do Inertia amplia ainda mais esses benefícios, facilitando a comunicação eficiente entre o *frontend* e o *backend*, proporcionando também uma transição suave entre as páginas sem recarregamentos completos. Essa abordagem não só evita a duplicação de código, tornando o desenvolvimento mais eficiente, mas também simplifica a manutenção do sistema.

Ademais, para construir um *backend* robusto para a aplicação, foi decidido pelo uso do *framework* Laravel, uma escolha justificada por sua robustez, simplicidade e eficiência no desenvolvimento de aplicativos web. O Laravel, como um *framework* PHP moderno, oferece uma extensa variedade de ferramentas e recursos que aceleram a criação de aplicativos, desde o gerenciamento de rotas até o ORM Eloquent, que facilita a interação com bancos de dados MySQL. Sua sintaxe simples e expressiva, juntamente com a arquitetura MVC (*Model-View-Controller*), simplifica o código, promovendo boas práticas de desenvolvimento. A arquitetura MVC divide a aplicação em três componentes principais: o *Model*, responsável pela manipulação dos dados; a *View*, encarregada da apresentação; e o *Controller*, responsável pela lógica de controle. Essa separação clara de responsabilidades facilita a manutenção do código e promove uma estrutura organizada e escalável para o desenvolvimento de *software*.

### 5.1. Módulo de cursos

Nesse módulo foram requisitadas como funcionalidades principais, a possibilidade de realizar cadastros, edições e exclusões de cursos, a listagem de todos que foram cadastrados e a viabilidade de realizar filtros por características específicas.

Após o entendimento das funcionalidades, ocorreu a coleta das características específicas que os cursos deveriam ter, e como resultado, foi estabelecido os seguintes aspectos: o nome do curso, o ano que está sendo cadastrado, o período que estará ocorrendo, a carga horária, a modalidade, a situação atual e o modelo do certificado do curso.

A seguir, as Figuras 1 e 2 demonstram o resultado da interface alcançada na implementação do módulo do sistema requerido.

The screenshot shows the 'Novo Cadastro' (New Registration) form in the ESEP MASTER system. The form is titled 'CURSOS' and includes a 'Voltar' (Back) button. The form fields are as follows:

- Nome \***: Curso exemplo 2
- Ano \***: 2023
- Data de início \***: 05/12/2023
- Data de finalização \***: 01/12/2023
- Carga horária \***: 30
- Modalidade \***: Presencial
- Situação \***: Processo de inscrição
- Modelo de certificado**: Seleccione um modelo...

A red error message is displayed at the top right of the form, stating: "A data de finalização não pode ser menor que a data de início do curso." The 'Salvar' (Save) button is located at the bottom right of the form.

**Figura 1. Mensagem de erro ao tentar cadastrar um curso.**

A Figura 1, seguindo as características específicas exigidas, apresenta a visualização da página destinada para realização dos cadastros dos cursos, onde nesta tem o campo opcional (modelo do certificado) e os que devem ser preenchidos, os quais são representados por um símbolo asterisco vermelho, além de validações específicas a depender do campo. Além disso, a página evidencia a tentativa de um cadastro que não foi permitido devido a inserção de datas incorretas, o que mantém a integridade do sistema e do banco de dados.

Após o preenchimento correto dos campos e apertado o botão de salvar, o usuário é redirecionado para a página incumbida de listar, de forma dinâmica, todos os cursos registrados, permitindo-lhe visualizar tanto a mensagem de êxito quanto o curso cadastrado que foi anexado na tabela, conforme a Figura 2 demonstra. Nesta tela, visando atender aos requisitos, é possível observar a possibilidade de realizar pesquisas pelos cursos já cadastrados, bem como também permitir a manipulação de um curso específico através dos botões de ações que foram pensados para serem intuitivos e possuindo funcionalidades específicas, como o botão de editar com a cor azul, o de deletar com a cor vermelha e o de visualizar os detalhes do curso com a cor amarela, onde neste último é possível também vincular os servidores ao curso, como será mostrado na Seção 5.2.

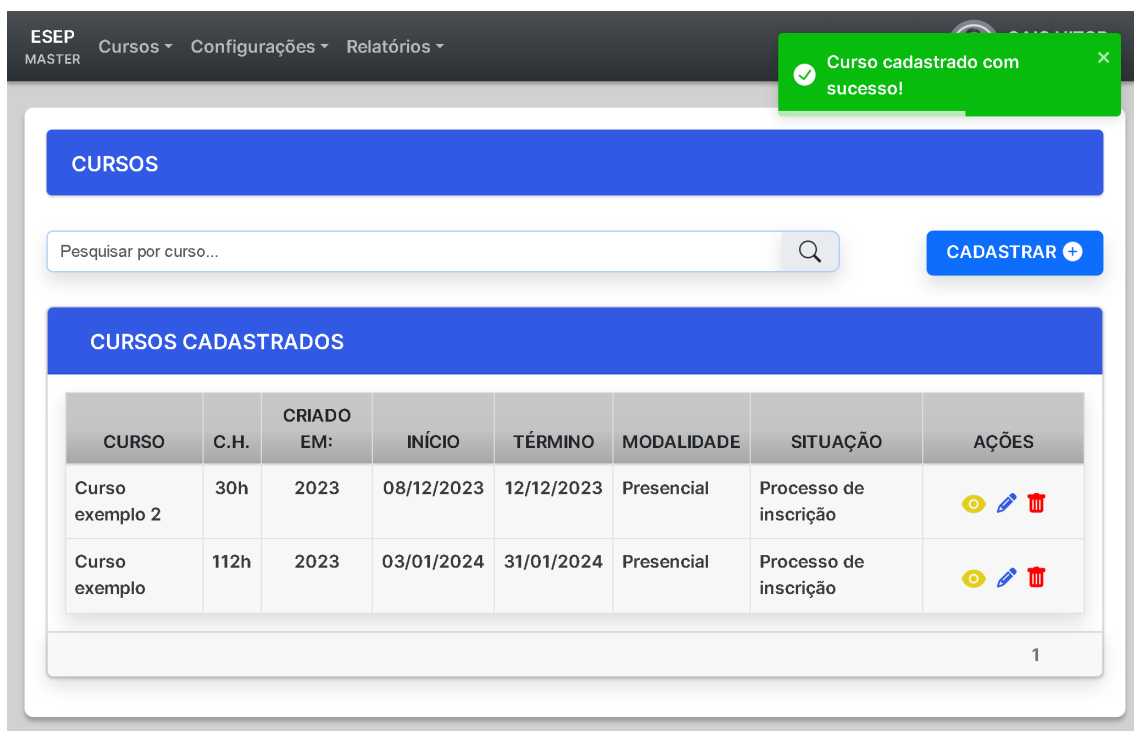


Figura 2. Mensagem de sucesso ao cadastrar um curso.

Para efetivar o cadastro, as informações enviadas pelo usuário passam pela função *store* no controlador do Laravel, responsável pelo módulo de curso, conforme a Figura 3 ilustra. Essa abordagem visa individualizar as responsabilidades de cada módulo do sistema, promovendo uma arquitetura mais coesa e facilitando a manutenção e expansão do software ao longo do tempo.

```

57 public function store(StoreUpdateRequest $request)
58 {
59     if($dtIn > $dtFm) {
60         return redirect()->route('cursos.create')
61         ->withErrors('A data de finalização não pode ser menor que a data de início do curso.');
```

Figura 3. Exemplo de código utilizado para cadastrar um curso.

A função *store*, como destacado na Figura 3, realiza importantes validações, ve-

rificando se a data de início é anterior à data de finalização do curso (linha 59) e se há correspondência da carga horária com o período estabelecido (linha 63). Quando todas as verificações são bem-sucedidas, os dados, incluindo o nome do curso, ano de criação, datas de início e término, modalidade, situação e modelo do certificado, são enviados ao banco de dados por meio do método *create* do *Eloquent*. Esse processo efetua o cadastro do curso (linhas 68 a 77) e redireciona para a lista de cursos (linha 78). No caso de falha em alguma validação ou no processamento dos dados, a função impede o cadastro e redireciona as mensagens de erro para a tela de cadastro (linhas 60, 64 e 80).

## 5.2. Módulo de servidor

Foi levantado o requisito de haver um mecanismo que possibilitasse vincular os servidores da SEJUS aos cursos, e após que ocorresse a vinculação destes a um curso específico, fosse mostrado todos os nomes que já estão associados e o quantitativo total de alunos. Adicionalmente, como informações específicas do aluno, deverá ser exibido a nota final, o total de falta e a data de emissão do certificado. Além do mais, como sugestão do autor deste trabalho aos futuros usuários do sistema, fosse adicionado a esse módulo, as funcionalidades que permitissem a visualização prévia do modelo do certificado e a posterior emissão de todos os certificados dos alunos.

A Figura 4 apresenta o resultado obtido da interface deste módulo, ilustrando os detalhes do curso selecionado pelo usuário na Figura 2. Além disso, a página exibe um botão de redirecionamento para a grade curricular, a opção de pesquisa por um servidor específico para adicionar como aluno do curso, a visualização na tabela dos membros vinculados e a capacidade de pré-visualizar ou emitir os certificados. Ao clicar nos botões “Prévia do certificado” ou “Emitir certificados”, uma mensagem de alerta é exibida, informando que o curso não possui um modelo de certificado vinculado, impedindo assim o usuário de visualizar o certificado.

The screenshot displays the 'ESEP MASTER' interface. At the top, there are navigation tabs for 'Cursos', 'Configurações', and 'Relatórios'. A yellow warning banner at the top right states: 'É necessário ter um modelo de certificado vinculado ao curso.' Below this, a blue 'Voltar' button is visible. The main content area shows details for 'Curso exemplo 2' (Presencial), including dates '08/12/2023 á 12/12/2023' and '30 horas'. A green 'Processo de inscrição' button and a blue 'Grade curricular' button are present. Below this, a section for 'Adicionar servidor ao curso' features a search input with 'CAIO VITOR LIMA BRITO' and an 'Adicionar' button. The 'MEMBROS DO CURSO' section shows 'INSCRITOS: 1' and a search input. Below the search input are buttons for 'Prévia do certificado' and 'Emitir certificados'. A table lists the member 'CAIO VITOR LIMA BRITO' with columns for 'NOME', 'EMISSÃO', 'FALTAS', 'NOTA', and 'AÇÕES'. The 'EMISSÃO' column shows 'NÃO EMITIDO', 'FALTAS' shows 'NÃO INFORMADO', and 'NOTA' shows 'NÃO INFORMADO'. The 'AÇÕES' column contains edit and delete icons. A page number '1' is centered at the bottom.



NOME	EMISSÃO	FALTAS	NOTA	AÇÕES
CAIO VITOR LIMA BRITO	NÃO EMITIDO	NÃO INFORMADO	NÃO INFORMADO	 

Figura 4. Tela de detalhes do curso e vinculação de servidores.

Após vincular um modelo de certificado ao curso e acionar o botão para visualizar o molde, uma nova guia será aberta no navegador, exibindo um PDF com o certificado. O mesmo ocorrerá ao acionar o botão de emissão, consoante mostra a Figura 5 à esquerda.



**Figura 5. Emissão e validação do certificado.**

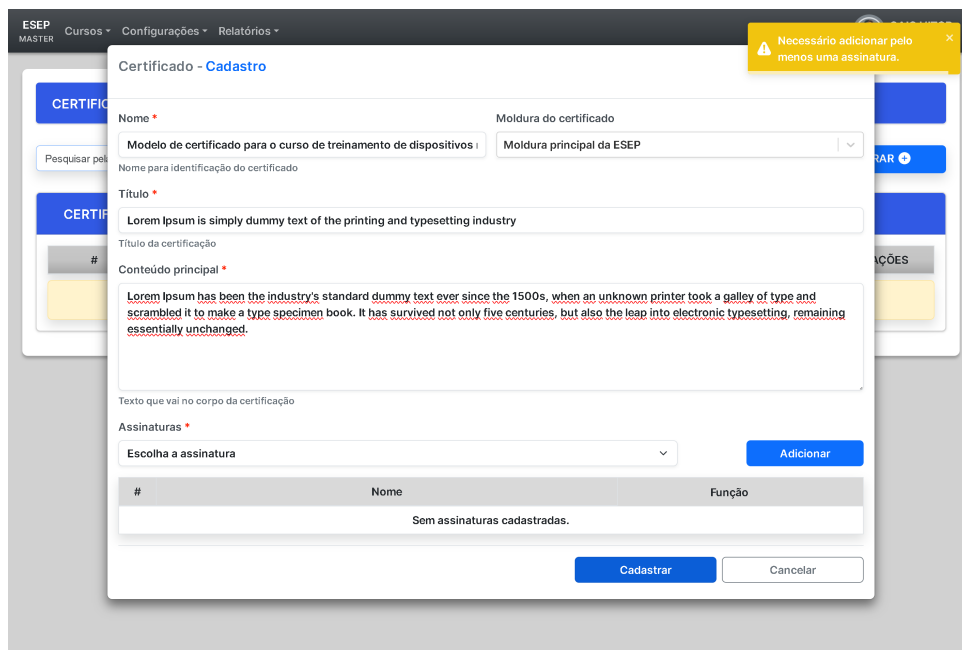
Nessa figura, é apresentado o certificado emitido para o aluno vinculado ao curso, incluindo um código QR para autenticidade, além de todos os elementos cadastrados para o modelo do certificado (consulte a Seção 5.3). Isso inclui o título acima do nome do servidor, o conteúdo principal do certificado e a assinatura. À direita da Figura 5, há uma página originada do escaneamento do código QR, onde são exibidos o nome do aluno, nome do curso, data de emissão do certificado e um campo indicando a validação do certificado.

### **5.3. Módulo dos certificados**

Para esse terceiro módulo, o objetivo principal exigido foi a capacidade do sistema viabilizar a criação de modelos de certificados que permitisse facilitar na reutilização dos modelos, sem precisar criar um específico para cada curso aplicado pela ESEP. Além disso, como funcionalidade complementar para esse módulo, foi solicitado um mecanismo que permitisse armazenar as assinaturas que serão utilizadas nos certificados.

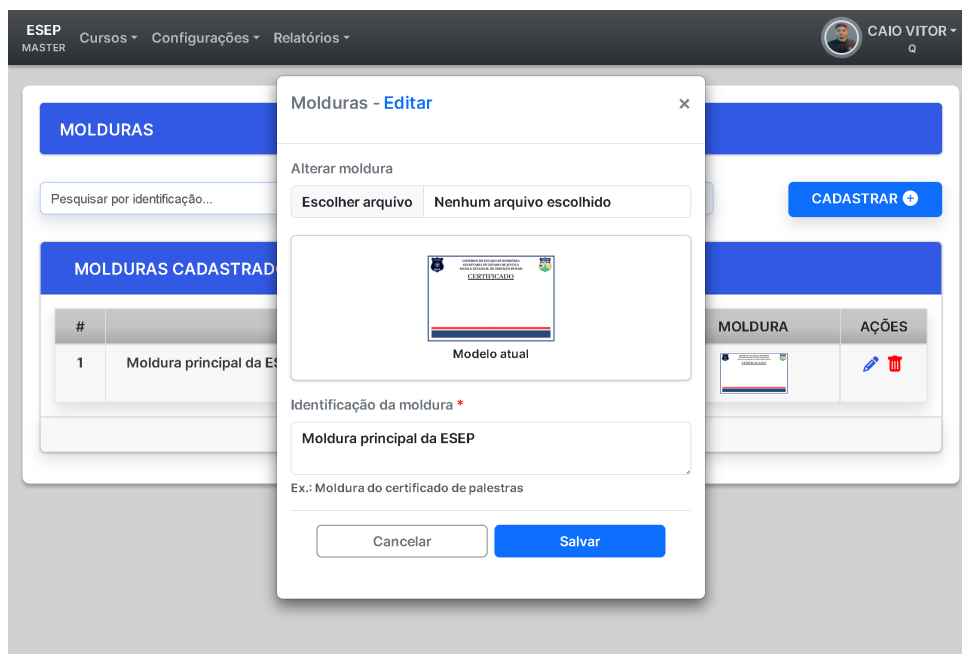
Desta forma, visando reduzir o acoplamento das funcionalidades requisitadas, o que, por consequência, favorecerá o aprendizado e a utilização do sistema pelos usuários, proporcionando-os maior flexibilidade de uso, além de facilitar a manutenção das partes do sistema e a implementação de novos requisitos, foi sugerido dividir esse módulo em três partes. Ficou estabelecido que uma das partes terá a responsabilidade exclusiva pelo cadastramento da estrutura do certificado, outra será destinada às molduras e, por último, uma terceira será dedicada exclusivamente às assinaturas.

Como atributos específicos da parte do sistema responsável pela estrutura dos certificados, ficou definido as seguintes características: identificação do modelo do certificado, o título, o conteúdo principal, a moldura que será utilizada, e as assinaturas que estarão presentes no certificado. Todas essas características específicas requisitadas, podem ser visualizadas na Figura 6, a qual mostra o resultado obtido da tela que permite a realização dos cadastros das estruturas dos certificados.



**Figura 6. Tela para o cadastro dos modelos de certificados.**

A Figura 6 exibe uma mensagem de alerta que informa ao usuário sobre a necessidade de incluir pelo menos uma assinatura na estrutura do certificado. Essa mensagem surge, de forma personalizada, quando o usuário tenta finalizar o cadastro com o formulário incompleto, destacando assim a obrigatoriedade dos campos. Essa exigência não apenas guia o usuário durante o preenchimento, mas também preserva a integridade do sistema, assegurando que informações essenciais sejam devidamente fornecidas.



**Figura 7. Janela modal para edição de uma moldura cadastrada.**

Para a parte responsável pelas molduras dos certificados, foram estabelecidos os seguintes requisitos: identificação da moldura, capacidade de anexar, armazenar e alterar a imagem da moldura. A Figura 7 permite visualizar a janela modal, sendo esta uma janela que se abre sobre o conteúdo da página, responsável pela edição de uma moldura. Nessa janela, apenas o campo de identificação é de preenchimento obrigatório, uma vez que a imagem da moldura já estará salva. Ao contrário para o cadastramento, que utiliza a mesma estrutura do formulário, porém todos os campos deverão ser preenchidos para realizar o efetivo cadastro da moldura. Após findado o cadastramento, a janela modal será fechada e a moldura será adicionada na tabela que possui todos os dados renderizados de forma dinâmica, juntamente com uma mensagem informando o êxito da ação realizada.

Na última repartição, dedicada às assinaturas, foram determinados os requisitos subsequentes: identificação e função do responsável pela assinatura, viabilidade de anexar, armazenar e manipular a imagem da assinatura, bem como a opção de deixar a assinatura ativa ou inativa. A Figura 8 demonstra a tela encarregada por permitir a edição e manipulação das assinaturas.

The screenshot shows a web application interface for editing signatures. At the top, there is a navigation bar with 'ESEP MASTER' and links for 'Cursos', 'Configurações', and 'Relatórios'. A user profile for 'CAIO VITOR' is visible in the top right. The main content area has a blue header 'ASSINATURA' and a 'Voltar' button. Below this is a section titled 'Editar assinatura' with the following fields: 'Nome' (Caio Vitor Lima Brito), 'Função' (Desenvolvedor de software), 'Assinatura' (Escolher arquivo, assinatura.png), and 'Ativar assinatura?' (Sim). A visual comparison shows the 'Assinatura atual' and the 'Assinatura selecionada'. A 'Redimensionar' button is located below the comparison. At the bottom right, there is an 'Atualizar' button.

**Figura 8. Tela para edição das assinaturas.**

A tela de edição de assinaturas, representada na Figura 8, foi implementada conforme os requisitos solicitados. Nela, estão disponíveis campos para preenchimento do nome e função do responsável, bem como a opção para estabelecer se a assinatura estará ativa ou inativa, um fator crucial para determinar sua disponibilidade para vinculação aos modelos de certificados (vide Figura 6). Além disso, a figura ilustra um campo de

entrada de dados para a alteração e seleção de um novo arquivo proveniente do computador do usuário, permitindo substituir a assinatura atual. Logo abaixo desse campo, são exibidas tanto a imagem da assinatura cadastrada quanto a imagem que substituirá a atual, permitindo a confirmação visual da escolha do usuário. O botão **redimensionar**, ao ser acionado, permite que o usuário faça manipulações na imagem, como o redimensionamento, implementado com a utilização da biblioteca `react-filerobot-image-editor`<sup>9</sup>, e a remoção do fundo branco.

#### 5.4. Módulo auxiliar

Nesse tópico, foi requisitado um módulo adicional para complementar o objetivo principal do sistema. Esse módulo deve possibilitar a apresentação de informações organizadas, precisas e resumidas de forma clara e compreensível por meio de relatórios gerados a partir de filtros aplicados com base em características específicas, tanto de cursos previamente cadastrados na aplicação quanto de servidores da SEJUS, favorecendo assim para tomadas de decisões.

Com o propósito de atender às demandas desse módulo e gerar os relatórios mencionados, foram estabelecidas especificidades para os filtros destinados aos cursos. O sistema deve ser capaz de viabilizar pesquisas por nome, situação atual, modalidade, data de início e de finalização do curso. A Figura 9 apresenta a implementação desta parte dedicada aos relatórios de cursos.

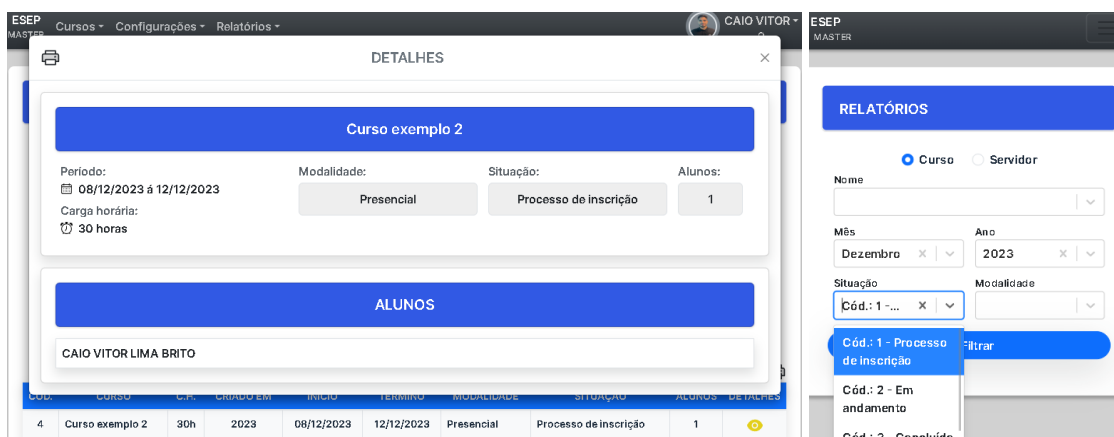


Figura 9. Relatório com os detalhes do curso e seleção de filtros para a consulta.

A direita da Figura 9, é exibido a página responsável por permitir realizar consultas por cursos. Nesta página, são disponibilizados cinco campos para seleção, permitindo que o usuário escolha combinações de características, resultando em uma consulta e resultado mais específico. Os dados apresentados na seleção dos campos de nome e ano são incrementados de forma dinâmica. Ou seja, a cada cadastro de um novo curso, o nome deste é adicionado à lista, e de maneira diferente, ocorrerá com o ano. O valor só será incrementado se ainda não exista no banco de dados, evitando a repetição de registros iguais.

Assim que o usuário realizar a consulta e o critério de busca corresponder com pelo menos um registro armazenado no banco de dados, uma tabela será exibida abaixo

<sup>9</sup><https://www.npmjs.com/package/react-filerobot-image-editor>. Acesso em 16/11/2023

dos filtros. Nessa tabela, as informações, como nome do curso, carga horária, data de criação, período de início e término das atividades, modalidade, situação atual e quantitativo de alunos vinculados, serão apresentadas de forma organizada. Além disso, um botão de cor amarela estará disponível. Ao ser acionado, esse botão abrirá uma janela modal. Dentro desta janela, além das informações já mencionadas, será exibido um ícone de impressora para a realização da impressão da janela. Ademais, o nome de todos os alunos matriculados no curso será apresentado, conforme ilustrado à esquerda da Figura 9.

## 6. CONSIDERAÇÕES FINAIS

Neste trabalho foram apresentados uma variedade de conceitos e tecnologias relacionados ao desenvolvimento da aplicação. A integração desses elementos, aliada à coleta de dados e à metodologia adotada, foi fundamental para o alcance bem-sucedido de todos os objetivos estabelecidos ao longo do processo de desenvolvimento da aplicação. Compreendendo desde o entendimento dos requisitos funcionais, prototipagem dos módulos, até a conclusão do desenvolvimento e a sua validação.

Adicionalmente, vale ressaltar que todas as ferramentas escolhidas para a implementação do sistema, foram cruciais para obter uma aplicação com interface interativa, proporcionando uma experiência de usuário mais fluída e agradável. Além de um *backend* robusto, com uma arquitetura que permite uma estrutura organizada e escalável para futuras implementações no software.

Após a implementação de todos os módulos que foram requisitados, resultou em um sistema capaz de viabilizar cadastros e manipulações de cursos, servidores, estrutura de certificados, moldura e assinaturas. Além disso, possibilita a geração de certificados para os participantes dos cursos e a emissão de relatórios atualizados.

Este sistema, destinado a auxiliar a ESEP no gerenciamento das qualificações dos servidores da SEJUS do estado de Rondônia, representa um avanço significativo para otimizar a eficiência nos fluxos de cadastros, vinculações de servidores e emissão de certificados, proporcionando maior praticidade dessa gestão e tornando-a mais consistente e aprimorada.

Durante a elaboração deste trabalho, surgiram oportunidades adicionais que não foram exploradas, uma vez que abordá-las tornaria o documento excessivamente extenso. Desta forma, sugiro, como trabalhos futuros, a implementação de uma biblioteca que contenha todos os materiais instrucionais que serão aplicados nos cursos, além da possibilidade de implementação de uma API de assinatura digital favorecendo para a validação das assinaturas.

## Referências

- Albertin, A. L. and Albertin, R. M. d. M. (2008). Benefícios do uso de tecnologia de informação para o desempenho empresarial. *Revista de Administração Pública*, 42:275–302.
- Andrei L. (2023a). O que É mysql? guia simples e direto para iniciantes? <https://www.hostinger.com.br/tutoriais/o-que-e-mysql> [Acesso em 23/10/2023].
- Andrei L. (2023b). O que é html: O guia definitivo para iniciantes. <https://www.hostinger.com.br/tutoriais/o-que-e-html-conceitos-basicos> [Acesso em 02/09/2023].

- Andrei L. (2023c). O que é react e como funciona? <https://www.hostinger.com.br/tutoriais/o-que-e-react-javascript> [Acesso em 24/08/2023].
- Ariane G. (2022). O que é css? guia básico para iniciantes. <https://www.hostinger.com.br/tutoriais/o-que-e-css-guia-basico-de-css> [Acesso em 04/09/2023].
- Bomfim, R. A. (2012). Competência profissional: uma revisão bibliográfica. *Revista Organização Sistêmica*, 1(1):46–63.
- Bootstrap (2023). Documentação bootstrap. <https://getbootstrap.com/docs/5.3/getting-started/introduction/> [Acesso em 06/09/2023].
- CARDOSO, L. (2021). *Frameworks Back End*. Editora Saraiva.
- Dall’Oglio, P. (2015). *PHP Programando com Orientação a Objetos 3ª Edição*. Novatec Editora.
- DB-Engines (2023). Db-engines ranking - popularity ranking of database management systems. <https://db-engines.com/en/ranking> [Acesso em 23/10/2023].
- Erika Heidi (2023). A practical introduction to laravel eloquent orm. <https://www.digitalocean.com/community/tutorial-series/a-practical-introduction-to-laravel-eloquent-orm> [Acesso em 27/09/2023].
- Flanagan, D. (2012). *JavaScript: o guia definitivo*. Bookman Editora.
- Inertia (2023). Inertia. <https://inertiajs.com/> [Acesso em 09/10/2023].
- Kinsta (2023). O guia definitivo para inertia.js. <https://kinsta.com/pt/base-de-conhecimento/inertia-js/#como-trabalhar-com-o-inertiajs> [Acesso em 14/10/2023].
- Kroenke, D. M. (2017). *Sistemas de informação gerenciais*. Saraiva Educação SA.
- Meta Open Source (2023). React. <https://react.dev/> [Acesso em 24/08/2023].
- PHP Group (2023). Php: Hypertext preprocessor. [https://www.php.net/manual/pt\\_BR/](https://www.php.net/manual/pt_BR/) [Acesso em 22/08/2023].
- RONDÔNIA (2023a). Portal da transparência do governo do estado de Rondônia. <https://transparencia.ro.gov.br/> [Acesso em 10/08/2023].
- RONDÔNIA (2023b). Secretaria de estado da justiça do governo do estado de Rondônia. <https://rondonia.ro.gov.br/sejus/> [Acesso em 10/08/2023].
- Silva, M. S. (2018). *Fundamentos de HTML5 e CSS3*. Novatec Editora.
- Soares, B. C. and Resende, R. S. (2017). Requisitos para utilização de prototipagem evolutiva nos processos de desenvolvimento de software para web. *Universidade Federal de Minas Gerais (UFMG)*.
- Theis, M. R., de Souza, G. G. A., Fialho, F. A. P., and Pereira, R. (2021). A importância da prototipagem no processo de design e suas relações como mídia do conhecimento. In *Anais do Congresso Internacional de Conhecimento e Inovação–ciki*, volume 1.