CTRL Empréstimos: Sistema web para controle de empréstimos de materiais no IFSP Hortolândia

Maciel Sandes de Oliveira[1], Daniela Marques[1]

[1]Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – Campus Hortolândia – 13.183-250 – Hortolândia – SP – Brasil

maciel.sandes@aluno.ifsp.edu.br, marquesdaniela@ifsp.edu.br

Abstract. *Currently, the Education Support Coordination (CAE) of the Federal Institute of São Paulo - Hortolândia Campus carries out the loan of materials manually, recording checkouts and returns on paper. As an improvement proposal, it is suggested to develop a web system for managing borrowed materials. The system aims to automate and streamline the loan registration process. The development will use the Laravel framework, enabling the creation of the system as a Progressive Web App (PWA).*

Resumo. *Atualmente a Coordenadoria de Apoio ao Ensino (CAE) do Instituto Federal de São Paulo - Campus Hortolândia realiza os empréstimos de materiais de forma manual anotando retiradas e devoluções em papéis. Como proposta de melhoria, propõe-se a criação de um sistema web para controle dos materiais emprestados. O sistema visa automatizar e agilizar o processo de registro de empréstimo. Para o desenvolvimento foi utilizado o framework Laravel, que possibilitou a criação do sistema usando Progressive Web Apps (PWA).*

1. Introdução

O avanço tecnológico tem proporcionado resultados cada vez mais convenientes, como a possibilidade de utilizar *smartphones* para realizar compras, pesquisas, assistir a filmes, ouvir músicas, entre outras atividades. Gradualmente, os *smartphones* têm se tornado uma ferramenta indispensável em nossas tarefas diárias, inclusive no contexto acadêmico.

O Instituto Federal de São Paulo (IFSP), campus Hortolândia, por meio da Coordenadoria de Apoio ao Ensino (CAE), efetua frequentemente o empréstimo de equipamentos aos servidores e acadêmicos e o controle é feito em anotações no caderno. Para encontrar o material emprestado é necessário ficar lendo todas as anotações e isso demanda tempo. Dito isso, propõe-se usar a tecnologia e criar um sistema que possibilite controlar os empréstimos de materiais por meio de *smartphone* ou computador, desta forma, facilitando o trabalho dos servidores envolvidos.

Tecnologias de desenvolvimento *web*, como *Progressive Web Apps* (PWA), inicialmente apresentado pelo engenheiro da Google, Alex Russell, em junho de 2015, possibilitam a capacidade de oferecer um aplicativo *Web* com recursos pertencentes a aplicativos nativos executados via *browsers*. Segundo o Dr. Karpagam (Karpagam, 2017), *"PWA describes a collection of technologies, design concepts, and Web APIs that work in tandem to provide an app-like experience on the mobile web"*[1]. Sites desenvolvidos usando conceitos de PWA tornam a fase de desenvolvimento mais barata, já que não é necessário o desenvolvimento de um aplicativo nativo para a plataforma e, ainda, permite que o *site* seja adicionado aos aplicativos da plataforma, trabalhe de modo *offline* e forneça notificações de *p*ush (KARPAGAM, 2017). Frameworks como Laravel, Angular, React e Vue possibilitam uso dos conceitos de PWA para o desenvolvimento de aplicativos.

1Tradução nossa: "PWA descreve uma coleção de tecnologias, conceitos de design e APIs da Web que funcionam em conjunto para fornecer uma experiência semelhante a um aplicativo web móvel na web móvel".

O *framework* escolhido para este projeto foi o Laravel, por ser um produto ágil. Ele utiliza a linguagem de programação PHP e permite desenvolver sem se preocupar com detalhes menores. Seu desenvolvimento se iniciou em 2011, é licenciado sob a licença MIT e seu código fonte tem conquistado a comunidade de desenvolvedores ao redor do mundo, tornando a comunidade bastante ativa (Amaral, 2024).

Este trabalho está organizado com uma introdução na Seção 1; a Seção 2 descreve os trabalhos correlatos; a Seção 3 apresenta os materiais e métodos utilizados; a metodologia está descrita na Seção 4; a Seção 5 apresenta o desenvolvimento do trabalho; e a conclusão está descrita na Seção 6.

2. Trabalhos correlatos

Com o objetivo de obter informações sobre o fluxo para o gerenciamento de empréstimos, foi feita uma pesquisa e selecionados dois sistemas para se observar características que pudessem ser utilizadas no sistema. O primeiro sistema estudado foi BiblIntelligence (utilizado para o atendimento de bibliotecas comunitárias), possibilitando o controle de livros emprestados. O segundo, SIGEM (utilizado para gerenciamento de materiais de almoxarifado), possibilita o gerenciamento de materiais de almoxarifado.

2.1 BiblIntelligence

BiblIntelligence é um sistema *web* focado em atender bibliotecas comunitárias mantendo uma rastreabilidade de dados (Figura 1). O sistema utiliza a tecnologia JSF (Java *Server Page*) para gerar páginas dinamicamente em HTML. O sistema possui como proposta, módulos que permitem o gerenciamento de usuários, livros, empréstimos. Para o controle de empréstimos, o sistema possui módulos que permitem visualizar livros cadastrados, realizar registro de empréstimo e realizar aplicações de filtros. Como limitação, o sistema se limita apenas gerenciar o empréstimo de livros (Santos, Barion e Marques, 2019).

Figura 1. Tela do sistema BiblIntelligence

2.2 SIGEM

SIGEM é um sistema para o gerenciamento de materiais de almoxarifado desenvolvido em Java (Figura 2). O sistema possui módulos para controle de entrada e saída de materiais e emissão de relatórios de materiais emprestados. Como limitação o software foi desenvolvido para ser executado localmente em *desktop* e não possui a funcionalidade para acesso em dispositivos móveis (Oliveira, 2018).

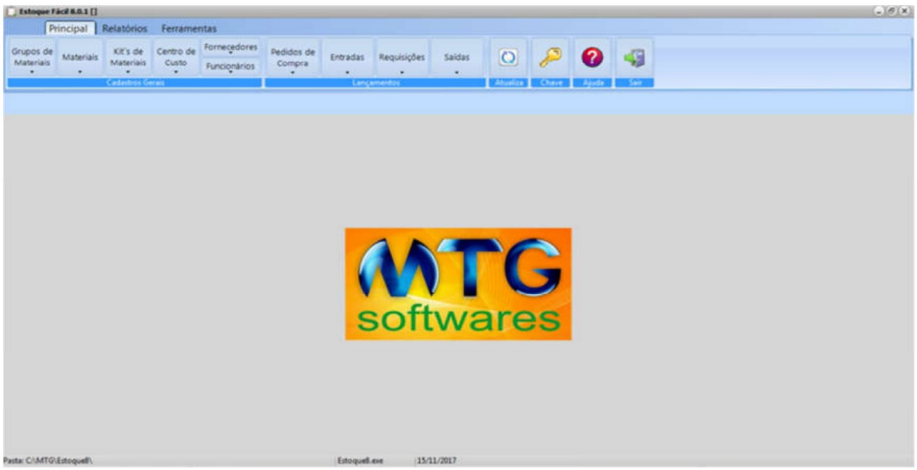

Figura 2. Tela do sistema SIGEM

3. Material e método

Esta seção apresentará os fundamentos para implementação do sistema proposto neste artigo.

3.1 Desenvolvimento Incremental

Este projeto utiliza o desenvolvimento Incremental (Pressman, 2006), por permitir que sejam feitas entregas com implementações rápidas de *software* útil ao cliente, mesmo sem todas as funcionalidades e, também, que ao longo do projeto possam ser feitas reflexões sobre o problema que está se tentando resolver.

Cada entrega inclui as funcionalidades de acordo com a priorização feita, permitindo que o cliente valide o sistema desde as etapas iniciais até a versão final (Oliveira; Bosnic, 2011).

3.2 Banco de dados

O banco de dados é uma coleção de dados relacionados que possuem algum significado de lógica inerente. São fatos que podem ser gravados e que possuem um significado implícito (Navathe, 2005).

Todo banco de dados relacional deve possuir elementos básicos como: campos, linhas, colunas e tabelas. Tabelas podem ser definidas como um conjunto de linhas, campos e colunas que possuem espaços reservados para armazenamento de dados. As colunas são registros de campos e linhas que podem ser agregados a outro conjunto de campos. Cada tabela do banco representará uma entidade ou formará um objeto em formato tabular (Carvalho, 2015).

3.3 *Progressive Web Apps* (PWA)

Progressive Web App (PWA) procura unificar o desenvolvimento *web* e móvel na procura de desenvolver um aplicativo com recursos pertencentes a aplicativos ativos. Basicamente, PWA é um *site* que pode ser adicionado à tela de aplicativos do celular e faz parecer um aplicativo nativo. Devido a isto é possível fazer a limpeza de *cache*, de espaço ocupado, forçar a parada ou desinstalação. Com isto cria-se a sensação de uma aplicação instalada no sistema operacional, contudo não é necessária a instalação de nenhum aplicativo (Fortunato; Bernardino, 2018).

Com PWA existe a possibilidade da aplicação ser executada em qualquer dispositivo devido ao fato de ser responsivo a diferentes tamanhos de telas e/ou orientação de telas (Karpagam, 2017).

3.4 Laravel

Laravel é um framework PHP, criado em 2011 e utilizado para o desenvolvimento web (Devmedia, 2022). Utiliza a arquitetura MVC (*model-view-control*) e tem como principal característica ajudar a desenvolver aplicações seguras e performáticas de forma rápida (Figura 3). Para isto, utiliza uma Engine para a criação de interfaces gráficas com *templates* chamados de Blade que trazem um conjunto de ferramentas para a criação de interfaces bonitas e funcionais.

Já para a comunicação com o banco de dados, o *framework* utiliza o ActiveRecord, também chamado de Eloquent ORM (*Object Relational Mapper*) que traz um conjunto de funcionalidades que facilitam a adição, atualização, busca e execução de registros.

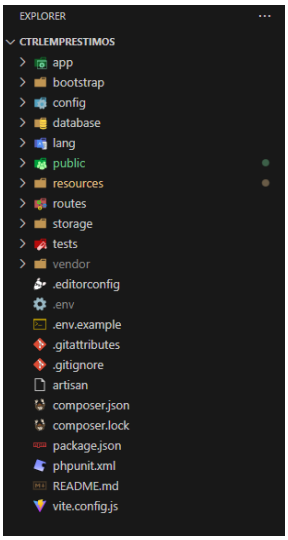

Figura 3. Estrutura do Laravel

3.4.1 Eloquent

O Eloquent é um ORM que facilita a implementação do Active Record Pattern, permitindo a interação simplificada com o banco de dados. Ele utiliza um modelo (*Model*) para criar um wrapper sobre a tabela do banco de dados à qual está associado. Cada instância de um modelo representa uma linha da tabela correspondente (Figura 4).

Além disso, o Eloquent oferece métodos acessores (*accessor methods*) que correspondem a cada célula de uma linha da tabela, proporcionando um acesso direto e intuitivo aos dados. (Eloquent ORM Prático, 2022).

```
public function up(): void
{
    Schema::create('password_reset_tokens', function (Blueprint $table) {
        $table->string('email')->primary();
        $table->string('token');
        $table->timestamp('created_at')->nullable();
    });
}
```

Figura 4. Método de criação de tabela

3.5.2 Blade

O Blade é um mecanismo de *template* simples e poderoso integrado ao Laravel, projetado para proporcionar maior eficiência na estruturação de códigos sem se limitar ao uso exclusivo

de PHP (Douglas; Marabesi, 2017). Os arquivos Blade são compilados para PHP e armazenados em cache, o que torna o carregamento mais rápido e reduz a sobrecarga na aplicação (Figura 5).

Os *templates* Blade utilizam a extensão ".blade.php" e, preferencialmente, devem ser armazenados no diretório *resources/views/*.

```
@extends('admin.layouts.default')

@section('title', 'Material')

@section('content')
    <!--Será mostrado uma mensagem informando que material foi salvo-->
    @if (session('msg'))
        <div class="alert alert-success">
            <p class="msg"> {{ session('msg') }}</p>
        <div>
    @endif

    <div class="container-fluid border p-3 mt-10">
        <div class="container-fluid bg-light border p-4 rounded">
            <h1>Materiais</h1>
        </div>
    </div>
```

Figura 5. Blade em arquivo HTML

3.6 Roteamento

Rotas de acordo com (Douglas; Marabesi, 2017) são teoricamente, itinerários que nos direcionam a qualquer localidade; e fazendo a associação a aplicações *web*, as rotas são os endereços ou propriamente os caminhos de acessibilidade por pessoas ou sistemas aos dados da aplicação desenvolvida.

Diferentemente de outros *frameworks*, o Laravel oferece um mecanismo de registro de rotas para a aplicação que é organizado, fácil de usar e flexível às necessidades do projeto. As rotas são definidas no arquivo routes.php, que processa as requisições HTTP enviadas pelo usuário por meio de URLs (Figura 6). Além disso, o Laravel permite o uso dos principais verbos HTTP na definição de rotas: GET, POST, PUT, PATCH e DELETE.

```
Route::get('/', function () {
    return view('welcome');
});
```

Figura 6. Exemplo mostrando como é um roteamento

3.7 Sistemas de autenticação

O *framework* Laravel de acordo com (Douglas; Marabesi, 2017), possui um mecanismo pré-implementado para gerenciar a autenticação de usuários, facilitada pelo uso de "*guards*" e "*providers*" e que cria todos os componentes vitais do projeto, provendo funcionalidades como controle de *login* de usuários, *logout* de usuários, cadastro de usuários, recuperação de senha de usuários e controle de sessão de usuários autenticados.

3.8 Validações

O Laravel fornece uma série de possibilidade de validação mais simples, fáceis e eficazes de dados oriundos de requisições de HTTP (Figura 7), além de prover métodos prontos que permite a análise de um dado, distinguindo o seu tipo (exemplo: arquivo, informação de e-mail, dados de informação do Excel etc.).

```
//Verifica se requisitos do formulario foram atendimentos antes de salvar no banco de dados
$rules = [
        'firstname' => 'required|string|max:255|min:3',
        'lastname' => 'required|string|max:255',
```

Figura 7. Exemplo mostrando como verificar requisito de senha

3.8 Bootstrap

O Bootstrap é um *framework front-end* que fornece estruturas de CSS para criar interfaces de usuários responsáveis e compatíveis com dispositivos móveis e *desktops* (Figura 8). Foi desenvolvido originalmente para o Twitter por um grupo de desenvolvedores liderados por Mark Otto e Jacob Thornton e é um dos projetos de estrutura de *front-end* mais populares do mundo. Geralmente ele é distribuído usado o site oficial, por npm ou CDN (Lima, 2024).

```
1   <div class="container">
2     <h1>My First Bootstrap Page</h1>
3     <p>This is some text.</p>
4   </div>
```

Figura 8. CDN de configuração do Bootstrap

3.9 Visual Studio Code

O Visual Studio Code é um editor de código gratuito desenvolvimento pela Microsoft, destinado ao desenvolvimento de aplicações *web* (Introdução ao visual Studio Code, 2022). O aplicativo é multiplataforma (disponível para Windows, Mac OS e Linux) e atende uma grama enorme de projetos. Através de extensões o programa dá suporte linguagem PHP, Java, Node.js, Python, Ruby, C++. Praticamente qualquer linguagem de programa.

4. Metodologia

Para o desenvolvimento do sistema desse artigo, foi escolhida a metodologia incremental. Por permitir sua implementação em pequenas partes. Como incremento inicial (zero) foi considerado a fase para determinar o escopo do projeto, cuja técnica de levantamento de requisitos seria conversar e conhecer o processo feito pela CAE para empréstimos de materiais. Devido uma pandemia com a necessidade de distanciamento social, não seria possível uma conversa presencial, para tal foi enviado um formulário de pesquisa via Google Forms.

Após este primeiro levantamento foi conversado com a Coordenadoria de Tecnologia de Informação (CTI) do campus para entender como funcionava a implementação de QrCode que estava sendo implementada no GLPI. A partir desses levantamentos foram identificados os requisitos iniciais do sistema.

Para cada funcionalidade do sistema criou-se um incremento, detalhando, implementando, testando os requisitos referente ao incremento, depois o teste e integração do código aos incrementos já existentes.

5. Desenvolvimento do trabalho

Esta seção apresenta todo o desenvolvimento do trabalho desde seu levantamento de requisitos até o código testado por incrementos.

5.1 Escopo do projeto

No incremento zero foram levantados os requisitos do sistema por meio de questionários enviados a CAE e reunião com a CTI.

5.1.1 Levantamento dos requisitos do sistema

Para o levantamento de requisitos, como não seria possível uma entrevista ou uma reunião com os servidores da CAE presencialmente devido a pandemia, foi elaborado um questionário com participação voluntária para preenchimento online. O objetivo desse questionário foi

compreender os problemas existentes, levantar o processo utilizado atualmente e compreender o negócio.

Ao envio deste questionário a CAE contava com 5 servidores. Obtivemos respostas somente de 2 dos servidores do setor. Apesar de não representar a opinião da maioria, consideramos essas informações para este trabalho.

O questionário proposto e enviado aos servidores é apresentado na Tabela 1.

Tabela 1. Questionário enviado aos servidores

1. Que tipo de material é emprestado pela CAE? Notebook / Projetores Adaptador (Tomada, VGA, HMDI) e outro
2. Todos os materiais emprestados pela CAE são patrimoniados? Sim / Não
3. Quem pode emprestar os materiais na CAE? Servidores (docentes e técnicos-administrativo) / aluno / outro:
4. Poderia descrever como funciona o processo de empréstimo de equipamentos?
5. Que tipo de dado é utilizado para controlar o que é emprestado? Patrimônio /nome do solicitante / matrícula / data de retirada/data de entrega / horário da retirada / horário de entrega.
6. Existe algum tipo de relatório ou informação gerada a partir dos dados coletados dos empréstimos de materiais realizados?
7. Qual a maior dificuldade encontrada pela CAE no gerenciamento desses materiais?
8. Você acredita que um processo automatizado poderia auxiliar a CAE? Por exemplo, a leitura de um código de equipamento e coleta de dados de dia e hora e informações do servidor.
9. Você teria alguma sugestão no processo que poderia facilitar a execução das suas atividades? Descreva sua sugestão.
10. Esse espaço é aberto para sugestões ou outras informações que julgar útil.

5.1.2 Resultado da aplicação do questionário

Em relação a primeira questão (Figura 9), referente ao material que pode ser emprestado pela CAE. O resultado foi que ambos os respondentes disseram que emprestam notebook, projetores e adaptadores e somente um foi mais detalhista em informar que emprestam extensão, caixa de som e microfone. Todos os dados citados foram conferidos e de fato podem ser emprestados pelos servidores.

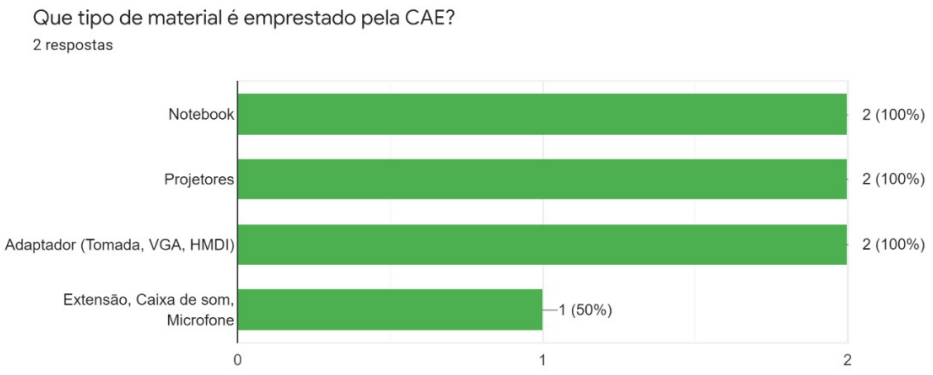

Figura 9. Tipos de material emprestado

Em relação a segunda questão (Figura 10), referente se todos os materiais emprestados pela CAE são patrimoniados. O resultado foi que um respondeu que nem todos os materiais emprestados são patrimoniados e outro respondente foi mais detalhista, respondeu que extensores e adaptadores não são patrimoniados.

Figure 10. Materiais emprestados possuem patrimônio

Em relação a terceira questão (Figura 11), referente a quem pode pegar materiais emprestados no CAE. O resultado foi que ambos os respondentes disseram que apenas servidores podem pegar materiais emprestados. Porém um dos respondentes também respondeu que alunos podem pegar materiais com autorização do professor.

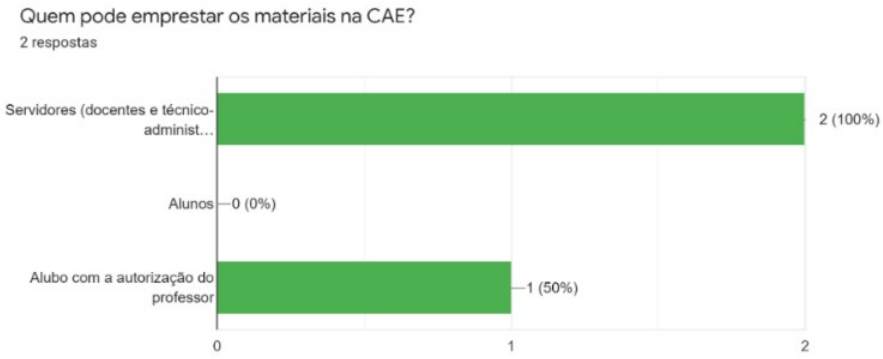

Figura 11. Pessoas que podem solicitar materiais

Em relação a quarta questão (Figura 12), referente a descrição do processo de empréstimo realizado pela CAE. As respostas dos respondentes foram:

- 1º Respondente: O professor solicita o material a ser emprestado, é marcado em um caderno quem retirou, o número do patrimônio (se houver), o horário de retirada e após o uso é marcado o horário que o material foi devolvido.
- 2º Respondente: É anotado o número do patrimônio, a data, quem retirou e a devolução.

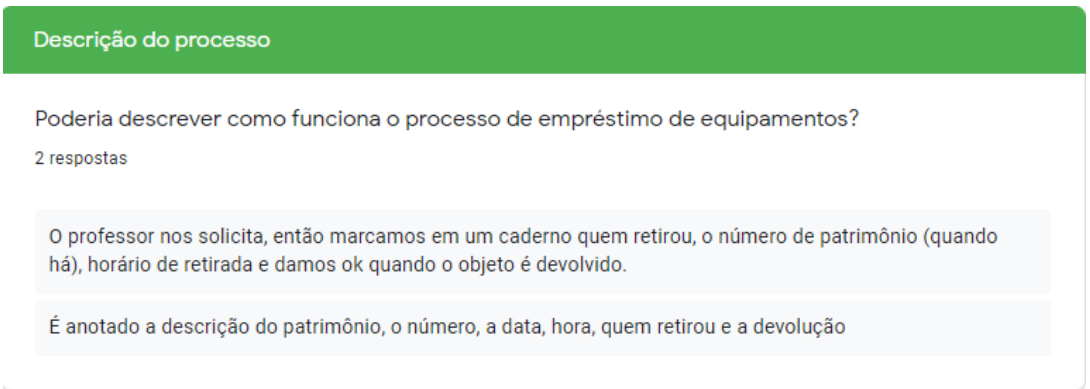

Figura 12. Descrição do processo de empréstimo

Em relação a quinta questão (Figura 13), referente ao tipo de dados utilizado para controlar o material emprestado pela CAE. O resultado dos dois respondentes foi que é utilizado o patrimônio, o nome do solicitante, data da retirada/data entrega, horário da retira/horário de entrega. Porém no caso do número do patrimônio, foi informado apenas por um dos respondentes que é utilizado.

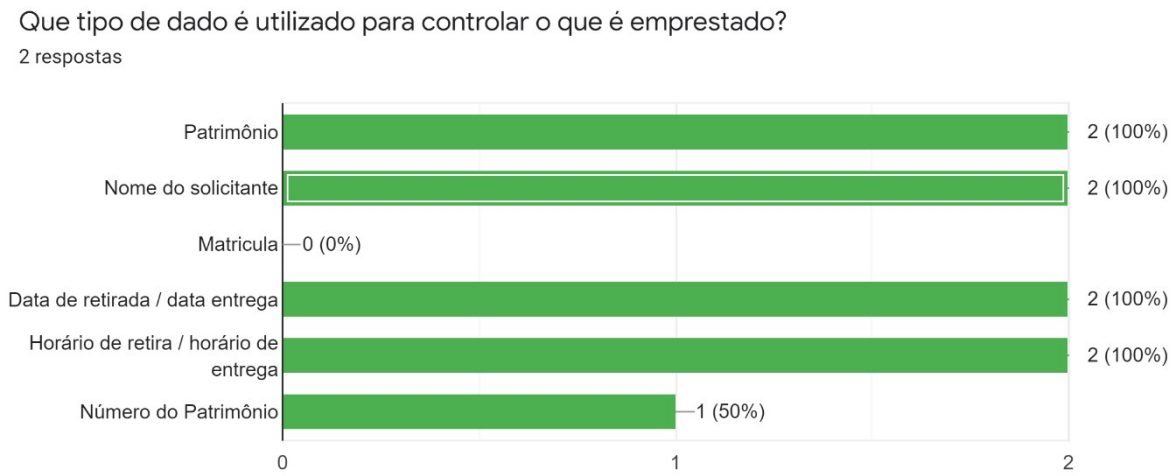

Figure 13. Dados utilizados para controlar empréstimos

Em relação à sexta questão, referente se existe algum tipo de relatório utilizado a partir de dados coletados dos materiais emprestados pela CAE. O resultado de ambos os respondentes, foi que Sim.

Em relação à sétima questão (Figura 14), referente às dificuldades encontradas no gerenciamento de matérias pela CAE. As respostas dos dois respondentes foram:

- 1º Respondente: Localizar empréstimos antigos.
- 2º Respondente: Algumas vezes um professor solicita um número de matérias que acaba excedendo a quantidade que a CAE dispõe. Assim quem chega antes acaba pegando materiais e falta para o outro. Muitas vezes os servidores da CAE não sabem por quanto tempo um professor que pegou o material, irá utilizar o material e acaba ficando difícil gerenciar os materiais de forma justa para todos.

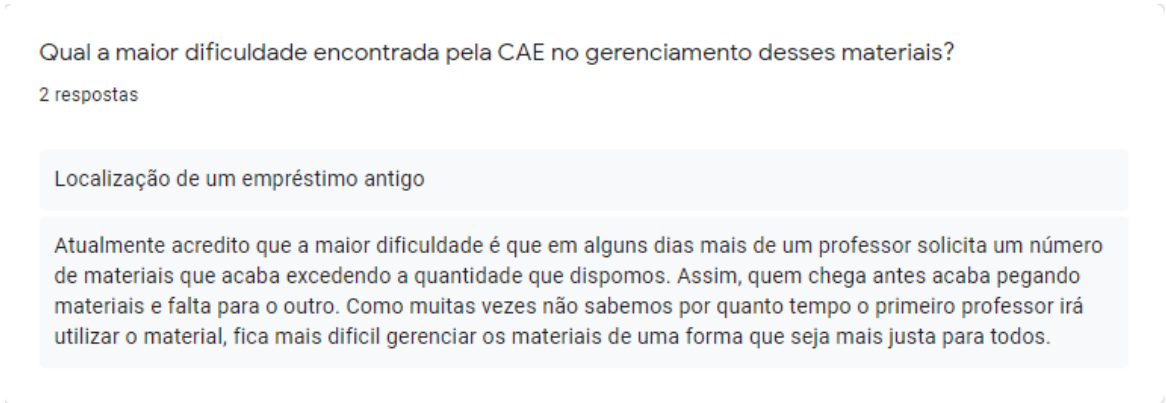

Figure 14. Dificuldades encontradas para gerenciamento de materiais

Em relação à oitava questão (Figura 15), se o respondente acredita que um processo automatizado poderia ajudar a CAE. As respostas passadas pelos dois respondentes foram:

- 1º Respondente: Sim pode ajudar.

- 2º Respondente: Acredito que sim, principalmente quando acontece de termos que procurar com quem está determinado equipamento.

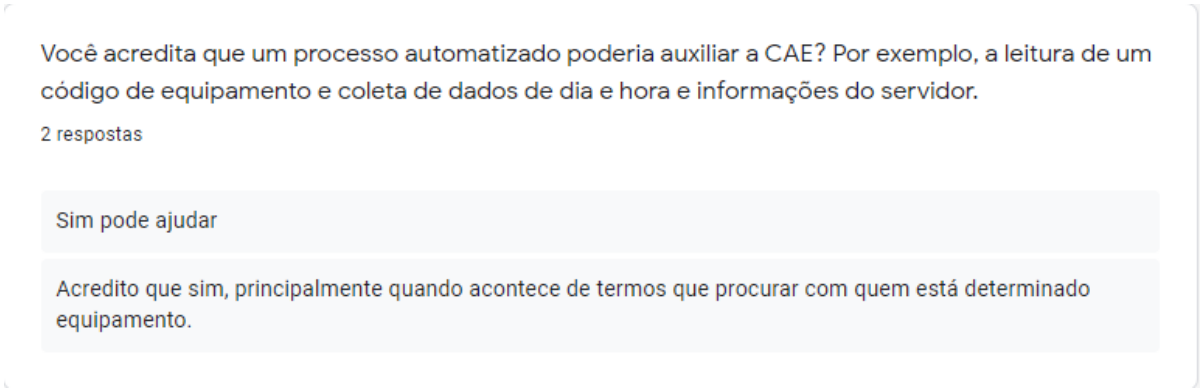

Figura 15. Resultado da pesquisa em relação ao processo automatizado

Em relação a nona questão (Figura 16), se o respondente teria alguma sugestão que pudesse facilitar a execução de suas atividades na CAE. A resposta informada por um dos respondentes, foi que se o software pudesse auxiliar com o agendamento de equipamentos, seria ótimo. Alguns professores utilizam notebooks que possuem programas específicos e solicitam que seja reservado para um dia e horário da semana. Atualmente não existe um software que controle agendamentos, correndo o risco de se esquecer e de emprestar para outro professor.

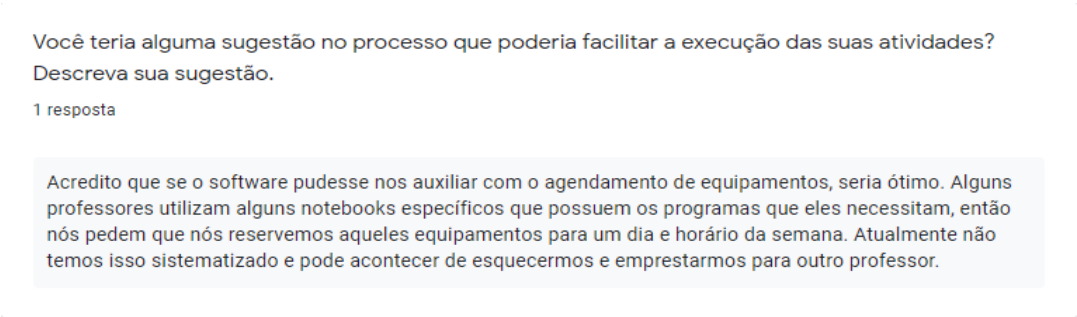

Figura 16. Dados utilizados para controlar empréstimos

A décima questão foi um espaço aberto deixado a sugestões ou informações. Neste caso não houve nenhuma resposta.

Para se obter informações sobre critérios internos utilizados pelo IFSP para implantação de sistemas em produção, foi agendada uma reunião com um servidor do CTI que estava desenvolvendo um *plugin* para o GLPI que possibilitaria a leitura de QRCode pelo sistema. O objetivo seria permitir que a partir de qualquer celular, quando um servidor necessitasse abrir chamado ao CTI, o mesmo entraria no sistema pelo celular, apontaria a câmera do celular para que fosse efetuado a leitura do QRCode com as informações do equipamento e efetuaria o descritivo do problema que gostaria que fosse verificado. Isto agilizaria o processo de abertura de chamado e facilitaria o processo de identificação do equipamento pelo CTI.

Para que o sistema fosse implementado dentro do IFSP, seria necessário que o sistema fosse desenvolvido como um *plugin* para uso dentro do GLPI. Para este tipo de implementação seriam necessárias mais horas de desenvolvimento, foi optado por um sistema independente.

5.2 Análise dos requisitos

Após a atividade de levantamento de requisitos, foi feito um diagrama de atividades (Figura 17) para demonstrar o fluxo de negócio entre os solicitantes que pode ser qualquer servidor ou aluno e os servidores da CAE.

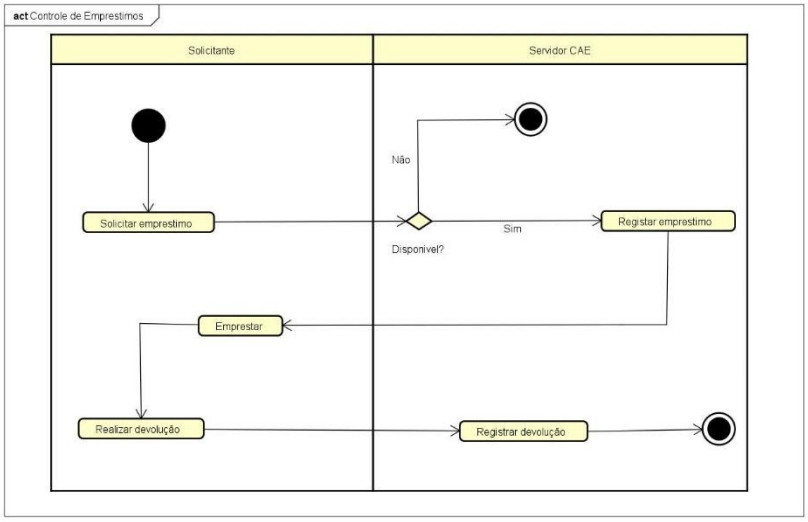

Figura 17. Diagrama de atividade do Controle de Empréstimo

Baseado no levantamento de requisitos e no fluxo de negócio, elaborou-se a Tabela 2 com a descrição dos requisitos funcionais do sistema. Todos os requisitos foram analisados como críticos ao sistema.

Tabela 2. Requisitos funcionais identificados no sistema

Id	*Descrição*	*Crítico (S/N)*
RF01	Controle de usuários. Criar, visualizar, editar e excluir usuários e perfis.	S
RF02	Controle de materiais. Criar, visualizar, editar e excluir materiais	S
RF03	Registrar empréstimos. Registrar empréstimo ou devolução de materiais.	S

Com a análise do Diagrama de Atividade e a identificação dos requisitos funcionais foi elaborado o Diagrama de Caso Uso, com objetivo de compreender melhor os requisitos de forma a entender as interações entre o sistema e os atores externos ao seu ambiente (Figura 18). O usuário principal do sistema é a conta administrador, contudo, o sistema ainda possibilita o cadastro de outros usuários individuais. Tanto o usuário Administrador, como os outros usuários cadastrados no sistema, pode realizar o gerenciamento de usuários, materiais e o registro de empréstimos de materiais.

Os requisitos não funcionais do sistema estão descritos na Tabela 3.

Tabela 3. Requisitos não funcionais identificados no sistema

Id	*Descrição*	*Crítico (S/N)*
RNF01	O sistema deverá ser executado através de navegadores de dispositivo móvel ou desktop	S
RNF02	O sistema deverá identificar o usuário que fizer acesso ao sistema com e-mail e uma senha.	S
RNF03	O sistema deverá verificar se o usuário tem permissão de fazer determinada alteração	S
RNF04	O sistema deverá permitir ao usuário, que recupere sua senha de acesso caso necessite	S

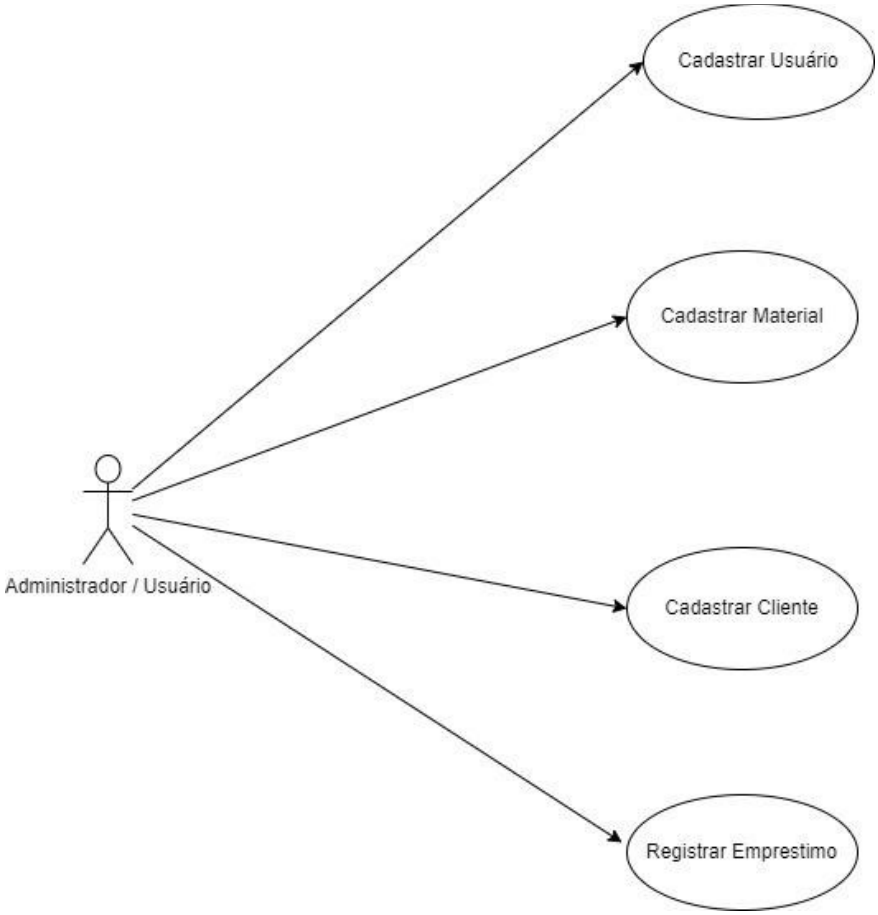

Figura 18. Diagrama de caso de uso

5.2.1 Definição do banco de dados

A partir das funcionalidades identificadas no Diagrama de Caso de Uso, foi possível elaborar o modelo conceitual do banco de dados. Na Figura 19 é possível verificar as entidades clientes e Materiais que contém os dados referentes a cada entidade e os dados do empréstimo e devolução que são identificados na relação dessas duas entidades. A Figura 20, apresenta a entidade de usuários e a entidade regras de usuários apresenta a entidade responsável pelas permissões dos usuários.

Figura 19. Diagrama Entidade-Relacionamento

5.3 Arquitetura do sistema

Como descrito neste artigo, o Laravel se utiliza da arquitetura MVC, visando benefícios atribuído pelo padrão de projeto de camada de serviços e repositórios (Figura 21).

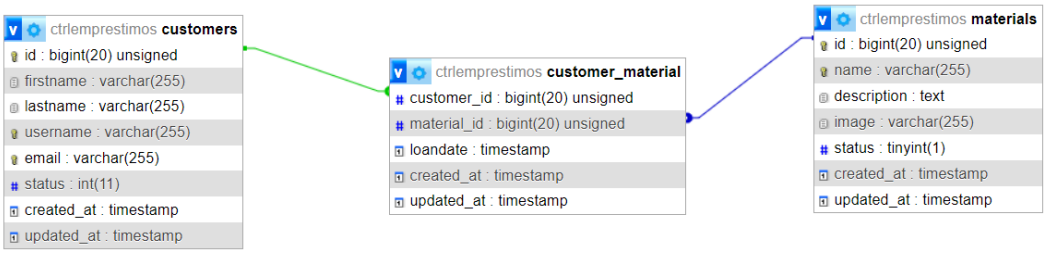

Figura 20. Diagrama Entidade-Relacionamento

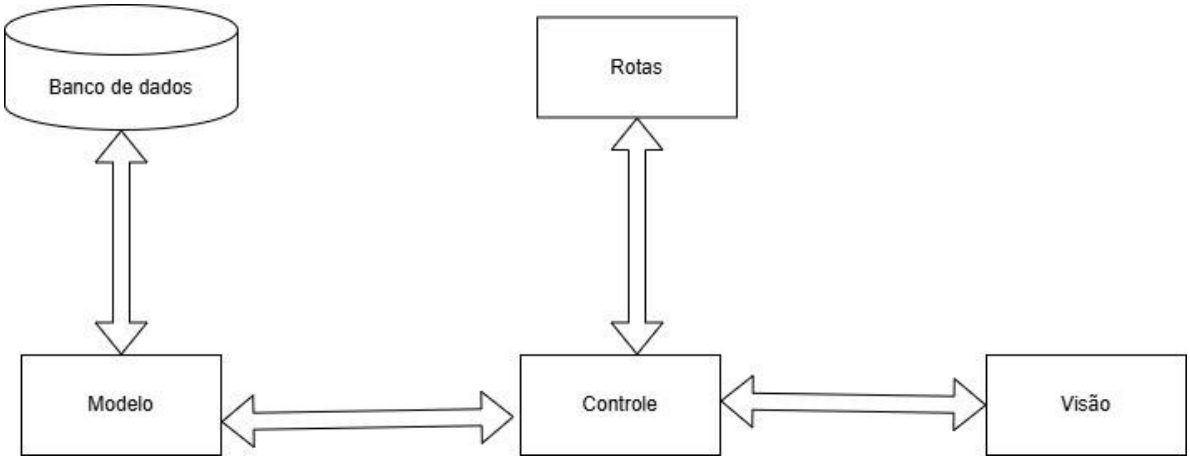

Figura 21. Reprodução da arquitetura MVC implementada

5.4 Implementação do sistema

Após análise do sistema, optou-se por criar a página inicial do sistema. A Figura 22 mostra a primeira versão da tela inicial do sistema. Para um esboço inicial foi criado uma página simples um menu vertical sem estilização. A Figura 23 mostra a versão final, a evolução da página inicial com as opções de menu principal estilizado e responsivo usando o *framework* Bootstrap. A Figura 24, mostra a página se adaptando a uma tela menor.

Figura 22. Tela do sistema de controle de Empréstimos

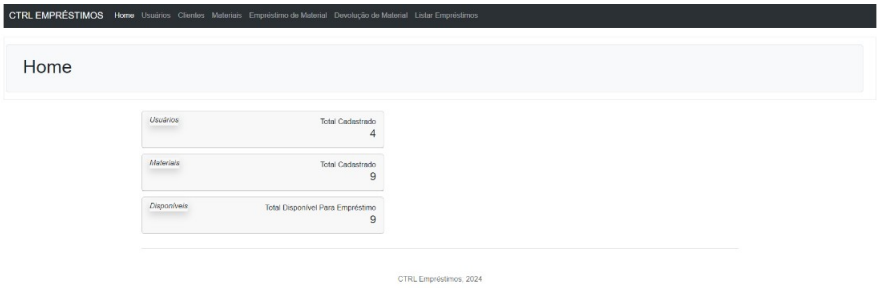

Figura 23. Segunda Final da tela do sistema

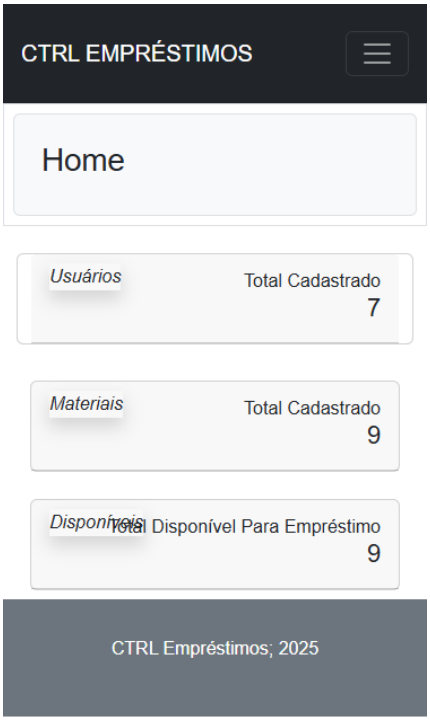

Figura 24. Tela são adaptáveis ao tamanho da tela do dispositivo

5.4.1 Incremento 1 - Cadastrar usuários

O primeiro Incremento foi referente ao caso de uso Cadastrar Usuário. A página Novo Usuário, possui um formulário que precisa ser preenchido para que um usuário serja cadastrado no sistema. Informações como nome, sobrenome, username, e-mail, senha e repetir senha (Figura 25). Ao clicar em Salvar o sistema verificará que os requisitos de cada campo foram adequadamente preenchidos e salvará as informações no banco de dados.

5.4.2 Listar usuário

Ao acessar a página principal de Usuários, é mostrado todos os usuários cadastrados no sistema (Figura 26). A página contém uma tabela com informações dos usuários e organizados por nome completo, username, e-mail e o status. A página ainda contém botões para: Novo Usuário (direciona para a página de cadastro de novo usuário), remover (remove o usuário), editar (edita as informações do usuário) e Situação (muda o estado atual do usuário).

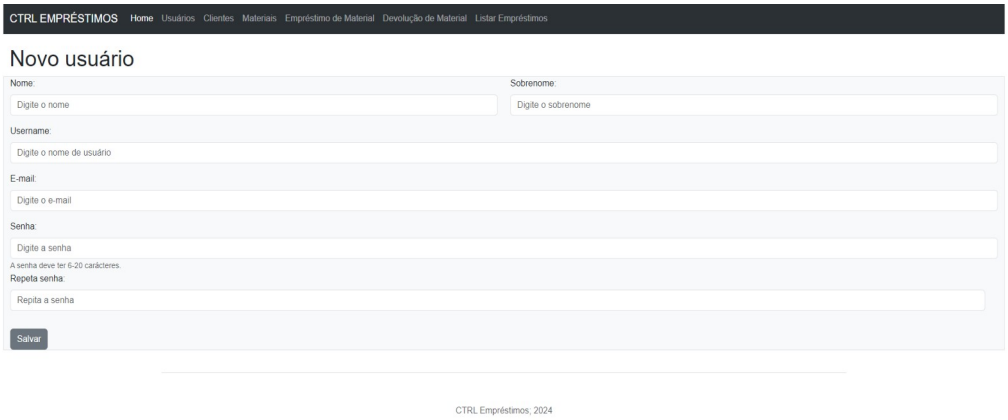

Figura 25. Cadastrar usuário

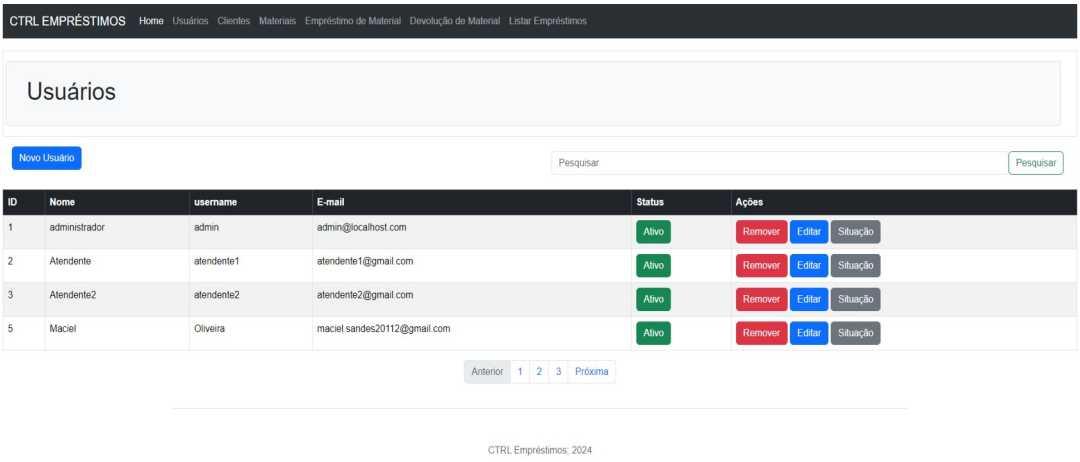

Figura 26. Lista usuários

5.4.3 Edição de usuário

Caso ocorra a necessidade de editar alguma informação de determinado usuário, poderá editar as informações do usuário, clicando sobre o botão editar e será direcionado a página de edição de usuário (Figura 27). Ao clicar no botão salvar, o sistema fará validações em todos os campos antes do envio ao banco de dados.

5.4.4 Exclusão do usuário

Caso seja necessário realizar a exclusão de um usuário do sistema, foi criado o botão Excluir que fará a remoção do usuário no banco de dados (Figura 28). Contudo, antes de efetuar a remoção, o sistema solicitará uma confirmação para evitar que ocorram exclusões erroneamente.

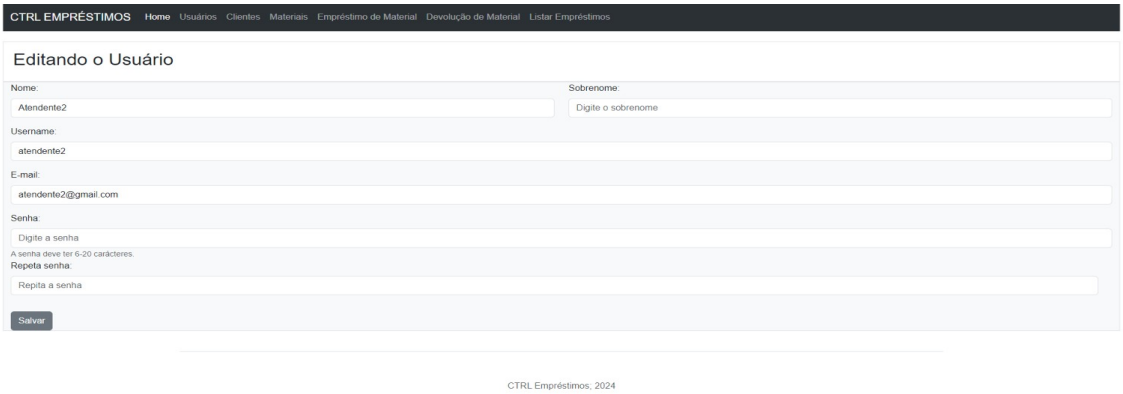

Figura 27. Edição de usuário

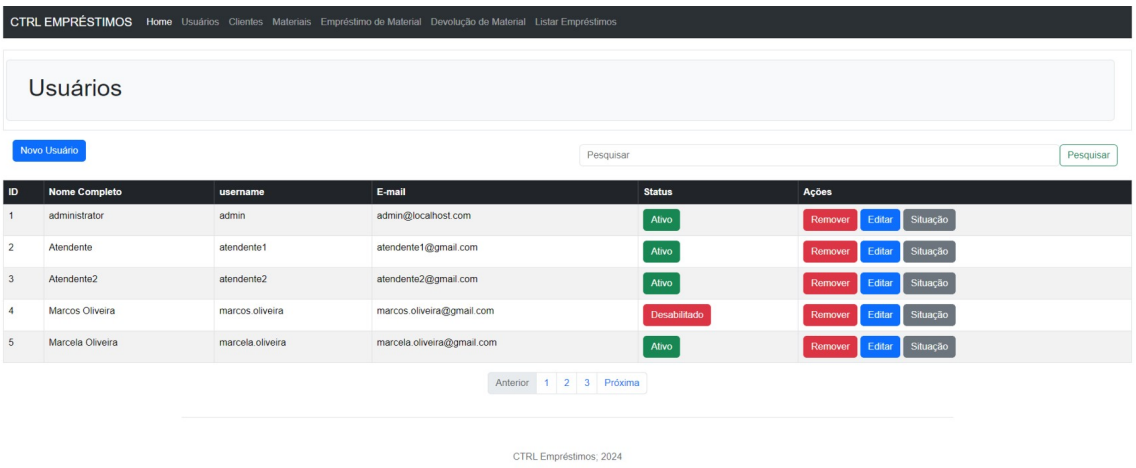

Figura 28. Edição de usuário

5.4.5 Status de usuário

Foi criado um botão que permite alterar a situação de um usuário (Figura 29). Ao clicar em Situação, o estado do usuário será alterado para ativado ou desativado.

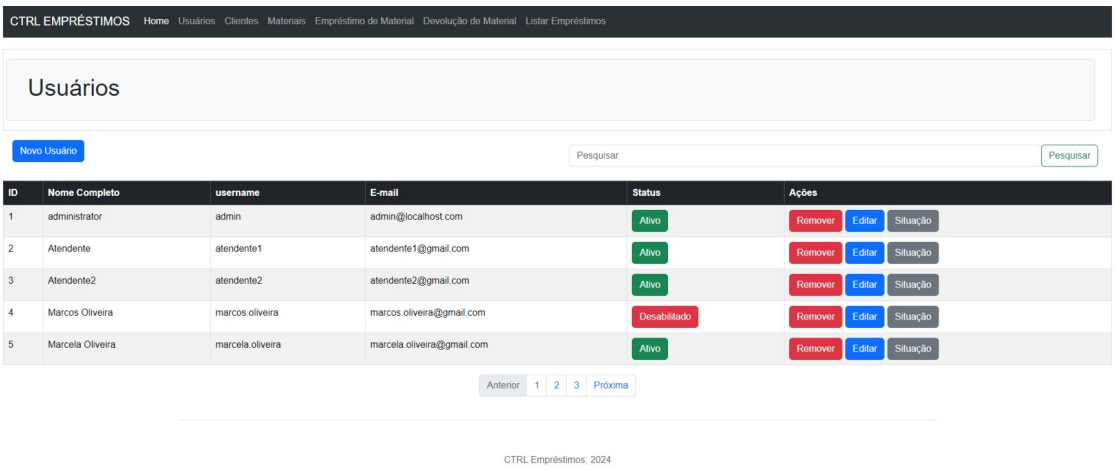

Figura 29. Edição de usuário

5.5 Incremento 2 - Cadastrar Clientes

A funcionalidade referente ao caso de uso Cadastrar Clientes possui o mesmo mecanismo de funcionamento descrito anteriormente para Cadastrar Usuário, como cadastro, edição e remoção de cliente. A Figura 30 apresenta a tela principal de Clientes.

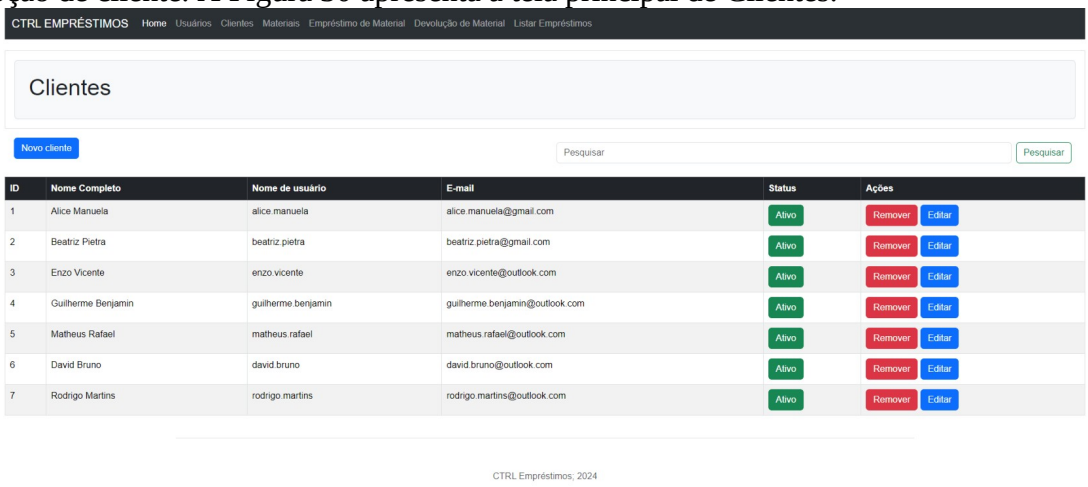

Figura 30. Lista de clientes

5.6 Incremento 3 - Cadastrar Materiais

A funcionalidade referente ao caso de uso Cadastrar Materiais possui o mesmo mecanismo de funcionamento descrito em Cadastrar Usuário, como cadastro, edição, e remoção de material. A funcionalidade é acessada por qualquer ator com permissão e edição. A Figura 31 apresenta a tela de Listagem de Materiais.

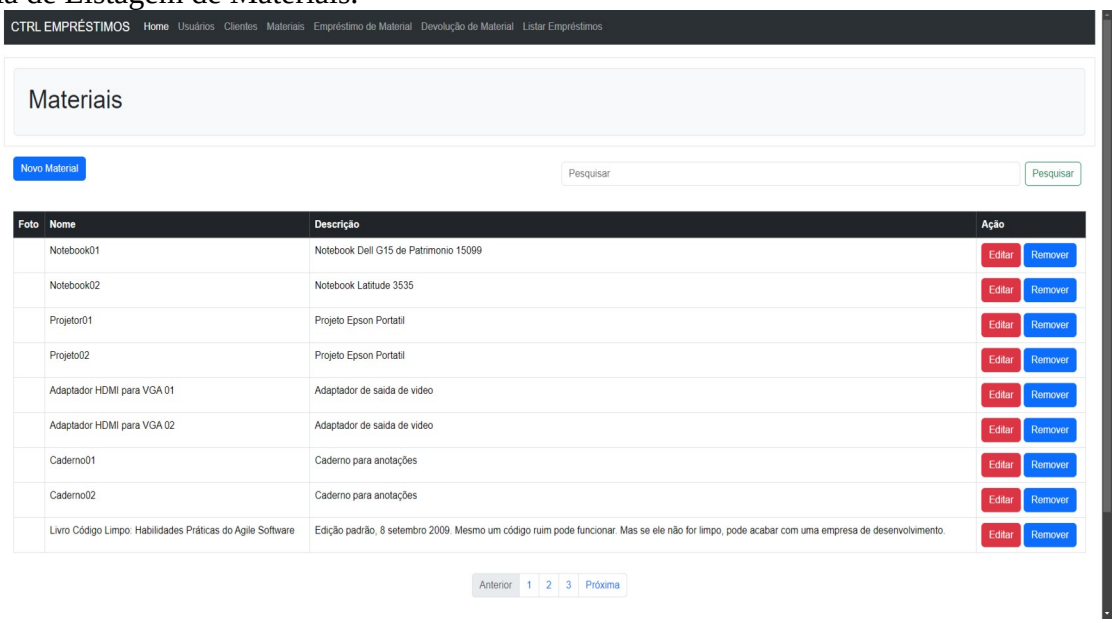

Figura 31. Lista de Materiais

5.7 Implantação Registro de empréstimo

Para o controle de materiais emprestados, foi criada uma página (Figura 32) onde o usuário deverá selecionar o material e o nome da pessoa que será efetuado o empréstimo. Também foi criada uma página onde permitirá que o usuário registre a devolução do material no sistema (Figura 33). O sistema também possui uma página lista os materiais emprestados.

Figura 32. Registro de empréstimo/Devolução

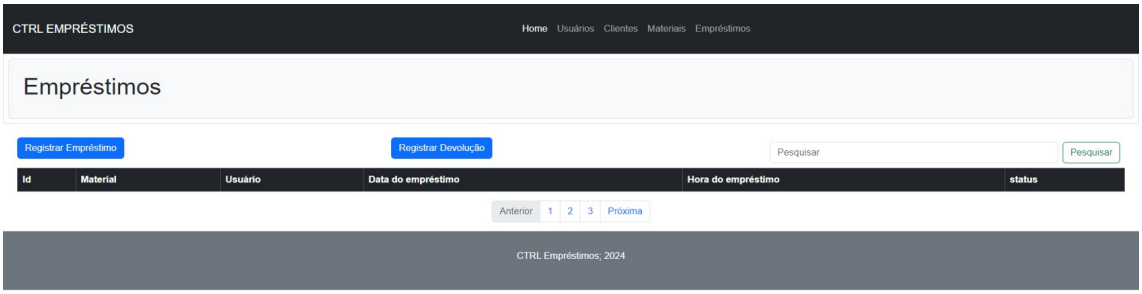

Figura 33. Listar materiais emprestados

5.8 Validação

O Laravel possui um sistema de validações em *Requests*, que permite desenvolver *script* que possibilitam a validações dos dados inseridos pelos usuários no sistema. Na página de Cadastro de Usuário, foi criado um script, que verificar se os campos como nome, sobrenome, nome de usuário, e-mail foram preenchidos (Figura 34), antes do envio ao banco de dados.

Caso o usuário venha a cometer algum erro ao inserir algum dado que esteja fora das regras da aplicação, uma mensagem será exibida no topo da página alertando o mesmo sobre o problema e solicitando realizando a correção.

```
//Verifica se requisitos do formulario foram atendimentos antes de salvar no banco de dados
$rules = [
        'firstname' => 'required|string|max:255|min:3',
        'lastname' => 'required|string|max:255',
        'username' => 'required|string|max:325|min:3',
        'email' => [ 'required', 'email', 'unique:users' ],
        'password' => [ 'required', 'min:6', 'confirmed'],
        'password_confirmation' => ['required', 'min:6' ],
    ];
```

Figura 34. Validação do campo de formulário de usuário

6. Conclusão

O objetivo deste trabalho foi desenvolver um sistema que permitisse automatizar o processo de empréstimo de materiais utilizado na CAE, abandonando assim anotações em papel. O sistema foi desenvolvido não só com a proposta de atender à demanda local, como também pensando em atender a empréstimo de qualquer tipo material. Como contribuição a comunidade, o seu código foi desenvolvido com software livre e disponibilizado no repositório no Github[2]. O fato de se usar Laravel torna o sistema compatível com os principais navegadores *web* e possibilita seu uso em computador ou disponíveis móveis. Com o estudo dos trabalhos correlatos foram aprendidos conceitos que possibilitaram o gerenciamento de usuários, clientes, materiais e empréstimos.

Uma das dificuldades encontradas durante as fases do projeto foi no módulo para registrar o empréstimo no sistema, inclusive para criar a estrutura do banco de dados, e como visualizar estas informações.

Para o desenvolvimento desse sistema, foram utilizados conceitos e tecnologias aprendidos no curso de Análise e Desenvolvimento de Sistema, como: Desenvolvimento *Web*, Programação Orientada a Objetos, Banco de Dados I, Análise Orientada a Objetos e Engenharia de Software. Ainda durante o desenvolvimento deste trabalho, foi necessário o aprendizado do framework Laravel.

Como trabalho futuro, é possível desenvolver as demais funcionalidades previstas no diagrama de caso de uso, tais como: página de login, criar o gerenciamento de permissões de usuários cadastrados, a leitura de material através de QRCode para agilizar o registro do material, o agendamento como reserva para uma data futura, melhoria na usabilidade da página de empréstimo de materiais alterando para não ser por seleção os campos e paginação das telas quando se tem um grande volume de dados. Outra possível integração, seria com o sistema BiblIntelligence via API, possibilitando que livros cadastrados dentro do sistema pudessem ser gerenciados pelo sistema.

2 https://github.com/macielsandes/ctrlemprestimos

Referências

ABREU, E. M.; DIA, S. A.; DALCORNO, L. C.; LANINI, F. D.; ALBARELLO, A. B. Uso de software livre para gestão do serviço de atendimento ao usuário de TI no INMETRO. Escola de Administração Fazendária (ESAF). s.d.

AMARAL, T. Disponível em: https://blog.configr.com/por-que-laravel-se-tornou-uma-das-melhores-opcoes-para-quem-trabalha-com-aplicacoes-em-php/. Acesso em: out. 2024.

AMINUDIN. Perancangan Sistema Repositori Tugas Akhir Menggunakan Progressive Web App (PWA). Techno.COM, v. 18, p. 154-165, maio 2019.

CARVALHO, V. MySQL: Comece com o principal banco de dados open source do mercado. In: CARVALHO, V. MySQL: Comece com o principal banco de dados open source do mercado. Casa do Código, 2015.

DEVMEDIA. Laravel tutorial. Disponível em: https://www.devmedia.com.br/laravel-tutorial/33173. Acesso em: 04 jul. 2022.

DOUGLAS, M.; MARABESI, M. Aprendendo Laravel: O framework PHP dos artesãos da web. Novatec Editora, 2017.

ELOQUENT ORM PRÁTICO. Introdução. Medium. Disponível em: https://medium.com/laraveltips/eloquent-avan%C3%A7ado-introdu%C3%A7%C3%A3o-d0515fe0f3c6. Acesso em: 04 jul. 2022.

FORTUNATO, D.; BERNARDINO, J. Progressive Web Apps: uma alternativa às Apps. Instituto Superior de Engenharia de Coimbra. s.d.

GLPI. Gestionnaire Libre de Parc Informatique. Disponível em: https://glpi-project.org/. Acesso em: 09 mai. 2020.

INTRODUÇÃO AO VISUAL STUDIO CODE. Devmedia. Disponível em: https://www.devmedia.com.br/introducao-ao-visual-studio-code/34418. Acesso em: 06 jul. 2022.

KARPAGAM, D. V. Performance enhancement of webpage using Progressive Web App features. 4 mar. 2017. p. 97-103.

LIMA, G. Bootstrap: O que é, Documentação, como e quando usar. Alura. Disponível em: https://www.alura.com.br/artigos/bootstrap?srsltid=AfmBOopAGv5MFpvpTKbNTE3rJHy3iAxnnTCu5VzyquNMKV3rB5t8QOh. Acesso em: 24 out. 2024.

MINDINVENTORY. Best Progressive Web Apps Frameworks. 2020. Disponível em: https://www.mindinventory.com/blog/best-progressive-web-apps-frameworks/. Acesso em: 23 jul. 2020.

NAVATHE, E. Sistemas de Banco de Dados. 2005.

OLIVEIRA, J. S. de. SIGEM – Sistema de Gerenciamento de Equipamentos e Materiais. 2018. Trabalho de Curso (Bacharelado em Sistemas de Informação) -- Campus Universitário de Castanhal, Universidade Federal do Pará, Castanhal, 2018. Disponível em: http://bdm.ufpa.br/jspui/handle/prefix/1075. Acesso em: 23 fev 2025.

OLIVEIRA, K.; BOSNIC, I. Engenharia de Software. 9. ed. Pearson Prentice Hall, 2011.

ORM: OBJECT RELATIONAL MAPPER. Devmedia. Disponível em: https://www.devmedia.com.br/orm-object-relational-mapper/19056. Acesso em: 11 jul. 2022.

PHP NET. Introdução. Disponível em: https://www.php.net/manual/pt_BR/intro-whatis.php. Acesso em: 04 jul. 2022.

PRESSMAN, R. S. Engenharia de Software. 2006.

WHAT IS JAVASCRIPT. MDN Web Docs. Disponível em: https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/First_steps/What_is_JavaScript. Acesso em: 05 jul. 2022.

Cópia de documento digital impresso por Michele Barion (1958699) em 6 de Março de 2025 às 15:06.

Documento Digitalizado Público

Anexo I Entrega do TCC final (Maciel S de Oliveira)

Assunto: Anexo I Entrega do TCC final (Maciel S de Oliveira)
Assinado por: Daniela Marques
Tipo do Documento: Projeto
Situação: Finalizado
Nível de Acesso: Público
Tipo do Conferência: Documento Digital

Documento assinado eletronicamente por:

- **Daniela Marques**, **PROFESSOR ENS BASICO TECN TECNOLOGICO**, em 26/02/2025 22:28:11.

Este documento foi armazenado no SUAP em 26/02/2025. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse https://suap.ifsp.edu.br/verificar-documento-externo/ e forneça os dados abaixo:

Código Verificador: 1952758
Código de Autenticação: f9edab03cf