

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba  
*Campus* Campina Grande  
Coordenação do Curso Superior de Bacharelado em Engenharia de  
Computação

## **Concepção e Desenvolvimento do Sistema Interno Integrado de Chaves (SIIC)**

Lucas Dantas de Araújo  
Pedro Henrique Guerra de Abreu

Orientador: Prof. Danyllo Wagner Albuquerque, DSc.

Campina Grande, Dezembro de 2023

®Lucas Dantas de Araújo

®Pedro Henrique Guerra de Abreu

A658c

Araújo, Lucas Dantas de

Concepção e desenvolvimento do sistema interno integrado de chaves (SIIC) / Lucas Dantas de Araújo, Pedro Henrique Guerra de Abreu. - Campina Grande, 2023.

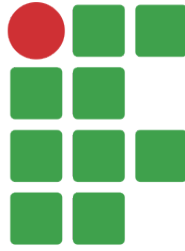
53f. : il.

Trabalho de Conclusão de Curso (Curso Superior de Bacharelado em Engenharia de Computação) - Instituto Federal da Paraíba, 2023.

Orientador: Prof. Dsc. Danyllo Wagner Albuquerque.

1. Engenharia de computação 2. Gestão - Processos internos 3. Desenvolvimento de sistema web I. Abreu, Pedro Henrique Guerra de II. Albuquerque, Danyllo Wagner III. Título.

CDU 004.41



Instituto Federal de Educação, Ciência e Tecnologia da Paraíba  
*Campus* Campina Grande  
Coordenação do Curso Superior de Bacharelado em Engenharia de  
Computação

## **Concepção e Desenvolvimento do Sistema Interno Integrado de Chaves (SIIC)**

Lucas Dantas de Araújo  
Pedro Henrique Guerra de Abreu

Monografia apresentada à Coordenação do  
Curso Superior de Bacharelado em Engenharia  
de Computação do IFPB - *Campus* Campina  
Grande, como requisito parcial para conclusão  
do curso de Bacharelado em Engenharia de  
Computação.

Orientador: Prof. Danyllo Wagner Albuquerque, DSc.

Campina Grande, Dezembro de 2023

# Concepção e Desenvolvimento do Sistema Interno Integrado de Chaves (SIIC)

**Lucas Dantas de Araújo**  
**Pedro Henrique Guerra de Abreu**

---

Prof. Danyllo Wagner Albuquerque, DSc.  
Orientador

---

Prof. Iana Daya Cavalcante Facundo Passos  
Membro da Banca

---

Prof. Igor Barbosa da Costa  
Membro da Banca

Campina Grande, Paraíba, Brasil  
Dezembro/2023

"Então, ele me disse: A minha graça te basta, porque o poder se aperfeiçoa na fraqueza. De boa vontade, pois, me gloriarei nas fraquezas, para que sobre mim repouse o poder de Cristo."  
(II Coríntios 12.9)

# Agradecimentos

## Agradecimentos Lucas Dantas de Araújo

Agradeço primeiramente a Deus, certamente quem mais me ajudou e tem ajudado em toda minha vida, diariamente e também nessa etapa.

Também agradeço a toda minha família que me apoiou e me incentivou a chegar até aqui. Por todo apoio e ajuda constante. Aos meus pais Ana Keylla e Franciclaudio por apoiar e incentivar nos estudos e na vida, a minha irmã, Ana Larissa, por todo companheirismo e companhia. E a minha namorada, Maria Vitória, que tem me ajudado e apoiado nesse final de curso.

Agradeço a Pedro Henrique por pegar junto nesse projeto, pelo auxílio nesse trabalho, mas também por todo apoio durante o curso, pela paciência, ajuda e companhia nas atividades, trabalhos e conversas.

Agradeço a todos os meus amigos e colegas que pelo esforço, pelas conversas e pelo exemplo que me motivaram e contribuíram para a concretização dessa fase.

Assim, também, aos professores e servidores do IFPB que trouxeram o conhecimento e auxílio fundamentais para a formação. Também ao professor Danyllo por ter nos passado essa missão de desenvolver o projeto deste trabalho e por toda ajuda nesse processo.

A todos que, mesmo não citados nesse texto, estão na memória nesse momento ou que mesmo que não lembrados agora, foram necessários para a conclusão deste trabalho, para a formação e para minha vida, para que eu conseguisse chegar nesse momento.

## **Agradecimentos Pedro Henrique Guerra de Abreu**

Em primeiro lugar, agradeço a Deus, por ter me dado forças e permitido que chegasse até aqui.

A minha família. Em especial a minha mãe Divaniedja, por sempre ter feito de tudo para que fosse possível concluir mais uma etapa em minha vida. Aos meus irmãos que sempre me motivaram e ajudaram nos momentos difíceis

Aos meus amigos, tanto os que estão presentes na rotina como os que estão distante mas mesmo assim se fizeram presente em diversos momentos. Agradeço ao meu amigo Lucas que foi fundamental para a realização desse e de outros trabalhos em equipe durante todo o curso, além do companheirismo. Aos amigos do grupo Hélix, que foram companhias em muitas noites, madrugadas de jogos e em outros momentos de diversão.

Aos professores, em especial ao Professor Danyllo por ter aceitado ser nosso orientador, além de toda ajuda e empenho no desenvolvimento desse trabalho.

E por fim ao IFPB, pela oportunidade e suporte durante esses 5 anos.

# Resumo

O eficiente registro e armazenamento diário de informações em um instituto desempenham um papel essencial na organização e eficácia de suas operações. Embora essa prática seja comum e inicialmente pareça ter impacto indireto no funcionamento da instituição, a gestão eficaz dessas informações é de extrema importância. Essa abordagem visa evitar conflitos nos horários de funcionamento de vários ambientes, prevenindo atrasos e adversidades que possam afetar negativamente as atividades realizadas. O registro cuidadoso e o armazenamento de dados cotidianos contribuem para o fluxo harmonioso das operações e a otimização do desempenho geral do instituto. Assim, o propósito deste trabalho é projetar e desenvolver uma plataforma que auxilie todas as partes envolvidas no registro e na utilização das salas e laboratórios. Utilizando tecnologias web e adotando uma abordagem ágil para o desenvolvimento, concebemos uma plataforma com funcionalidades voltadas ao processo de registro e gerenciamento de chaves no IFPB (Instituto Federal da Paraíba). Em um teste beta da plataforma, aproximadamente 10 usuários avaliaram as funcionalidades e a usabilidade do sistema, concluindo que sua utilização minimiza as problemáticas do atual formato de registro da instituição. Como conclusão, o desenvolvimento e implementação bem-sucedidos da plataforma de registro e gerenciamento de chaves no IFPB revelam sua eficácia em mitigar desafios operacionais, destacando-se como uma solução inovadora e promissora para otimizar a eficiência institucional e aprimorar a experiência dos usuários.

**Palavras-chave:** Registro de informações, Gerenciamento de salas e laboratórios, Plataforma web, IFPB, Desenvolvimento ágil, Teste beta.



# Abstract

The efficient daily recording and storage of information in an institute play a crucial role in its organization and operational effectiveness. Although this practice may seem commonplace and initially have an indirect impact on the institution's functioning, the effective management of this information is of paramount importance. This approach aims to prevent conflicts in the operating schedules of various environments, thereby preventing delays and adversities that could negatively impact the conducted activities. Careful recording and storage of daily data contribute to the smooth flow of operations and the overall optimization of the institute's performance. Therefore, the purpose of this work is to design and develop a platform that assists all parties involved in the registration and utilization of rooms and laboratories. By employing web technologies and adopting an agile development approach, we have conceived a platform with functionalities focused on the process of key registration and management at IFPB. In a beta test of the platform, approximately 10 users evaluated its features and usability, concluding that its use minimizes the challenges of the institution's current registration format. In conclusion, the successful development and implementation of the key registration and management platform at IFPB reveal its effectiveness in mitigating operational challenges, standing out as an innovative and promising solution to optimize institutional efficiency and enhance user experience.

**Keywords:** Information registration, Room and laboratory management, Web platform, IFPB, Agile development, Beta testing.

# Sumário

<b>Lista de Abreviaturas</b>	<b>xii</b>
<b>Lista de Figuras</b>	<b>xiii</b>
<b>Lista de Tabelas</b>	<b>xiv</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Problemática . . . . .	1
1.2 Objetivos . . . . .	2
1.3 Metodologia . . . . .	2
1.4 Justificativa e Relevância . . . . .	3
1.5 Estrutura do Trabalho . . . . .	4
<b>2 Fundamentação Teórica</b>	<b>5</b>
2.1 Tecnologias para Construção de Sistemas <i>Web</i> . . . . .	5
2.1.1 Tecnologias cliente . . . . .	6
2.1.2 Tecnologias servidor . . . . .	7
2.1.3 Banco de Dados . . . . .	7
2.2 Processo de Desenvolvimento . . . . .	7
2.2.1 Scrum . . . . .	8
2.2.2 Kanban . . . . .	9
2.3 Projeto Arquitetural . . . . .	9
<b>3 Trabalhos Relacionados</b>	<b>11</b>
<b>4 Processo de Desenvolvimento do SIIC</b>	<b>13</b>
4.1 Análise e Definição dos Requisitos . . . . .	13
4.2 Escolha das Tecnologias . . . . .	13
4.3 Processo de Desenvolvimento . . . . .	16
4.4 Validação e Testes . . . . .	16
4.5 Implantação . . . . .	18
<b>5 O Sistema Interno Integrado de Chaves</b>	<b>19</b>
5.1 A Plataforma SIIC . . . . .	19
5.1.1 Modelagem do Sistema . . . . .	19
5.1.2 Funcionalidades Públicas . . . . .	20
5.1.3 Funcionalidades do Usuário administrador . . . . .	20
5.1.4 Funcionalidades do Usuário Comum . . . . .	27

<b>6</b>	<b>Validação do SIIC</b>	<b>30</b>
6.1	Projeto do Questionário . . . . .	30
6.2	Definição da amostra e coleta de dados . . . . .	31
6.3	Resultados do Survey de Validação . . . . .	31
6.3.1	Caracterização da Amostra . . . . .	31
6.3.2	Resultados da Validação . . . . .	32
<b>7</b>	<b>Ameaças à Validade</b>	<b>36</b>
7.1	Validade de Construção . . . . .	36
7.2	Validade Interna . . . . .	36
7.3	Validade Externa . . . . .	37
7.4	Validade de Conclusão . . . . .	37
<b>8</b>	<b>Conclusões Finais e Trabalhos Futuros</b>	<b>38</b>
	<b>Referências Bibliográficas</b>	<b>40</b>

# Lista de Abreviaturas

IFPB	Instituto Federal da Paraíba
SIIC	Sistema Interno Integrado de Chaves
HTML	<i>HyperText Markup Language</i> (Linguagem de Marcação de HiperTexto)
CSS	<i>Cascading Style Sheets</i> (Folhas de Estilo em Cascata)
PHP	<i>Hypertext Preprocessor</i> (Processador de Hipertexto)
SGBD	Sistema de Gerenciamento de Banco de Dados
SQL	<i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
MVC	<i>Model-View-Controller</i> (Arquitetura Modelo-Visão-Controle)
IFBA	Instituto Federal da Bahia
SGC	Sistema de Gerenciamento de Chaves
UFC	Universidade Federal do Ceará

# Lista de Figuras

1.1	Etapas do processo de desenvolvimento. . . . .	3
2.1	Funcionamento do Scrum <sup>1</sup> . . . . .	9
2.2	Arquitetura MVC <sup>2</sup> . . . . .	10
2.3	Fonte: UsandoPy (2023) . . . . .	10
6.1	Caracterização da amostra. . . . .	32
6.2	Caracterização da ocupação. . . . .	32

# Lista de Tabelas

4.1	Lista de requisitos funcionais . . . . .	14
4.2	Lista de requisitos não-funcionais . . . . .	14
4.3	Cronograma de execução do projeto . . . . .	17
6.1	Seções do questionário do SIIC. . . . .	30

# Capítulo 1

## Introdução

Considerando o cenário contemporâneo, é incontestável que a informatização de processos transcende a condição de mero recurso complementar, tornando-se uma peça fundamental na estrutura das atividades cotidianas. Neste contexto, a adoção generalizada da tecnologia é evidenciada pela crescente necessidade de compreender e integrar soluções digitais, sendo que setores em diversas áreas de trabalho passaram por uma transformação significativa, seja por meio de aplicativos tradicionais ou sistemas móveis e baseados em plataformas *web*.

A evolução da computação corporativa e institucional reflete uma busca constante por maior agilidade e confiabilidade. O intuito é disponibilizar informações de maneira ampla ao público, garantindo simultaneamente a confiabilidade e utilidade desses dados. Diante desse panorama, torna-se imperativo o desenvolvimento contínuo de novos mecanismos de acesso e controle de informações, capazes de acompanhar a dinâmica da sociedade moderna e atender às demandas de um público cada vez mais conectado e exigente.

Nesse sentido, a presente pesquisa visa explorar e contribuir para esse ambiente em constante transformação, concentrando-se na automatização do registro de chaves no IFPB. Este projeto se insere em um contexto mais amplo de modernização e eficiência, alinhando-se com a necessidade latente de aprimorar processos institucionais por meio da incorporação estratégica de tecnologias avançadas. Ao abordar a relevância desse tema, busca-se não apenas atender às demandas presentes, mas também antecipar as necessidades futuras em um cenário marcado pela constante evolução tecnológica.

### 1.1 Problemática

A busca contínua por históricos de registros e os métodos tradicionais de registro representam atividades recorrentes no cotidiano de professores e servidores no IFPB. Contudo, esse procedimento, muitas vezes repetitivo, acaba gerando dificuldades no gerenciamento ágil e eficiente, no rastreamento de chaves, na busca por históricos específicos e na segurança dessas informações, que estão suscetíveis a danos ou perdas.

Com o avanço tecnológico constante e a rápida transformação digital, especialmente impulsionado

nada pelo contexto pandêmico recente, torna-se evidente a oportunidade de reimaginar e otimizar processos que antes eram conduzidos manualmente. A percepção é de que a aplicação de recursos tecnológicos disponíveis em nossa rotina pode oferecer soluções que não apenas aprimoram a qualidade do trabalho realizado, mas também proporcionam ganhos em agilidade, eliminando problemas inerentes à mão de obra repetitiva e estressante.

A redundância nas atividades manuais torna-se mais evidente em um cenário em que a tecnologia pode desempenhar um papel significativo na simplificação e eficiência operacional. A automação do registro de chaves, nesse contexto, surge como uma resposta estratégica para superar as limitações impostas por métodos convencionais, propondo uma abordagem inovadora e eficaz para lidar com desafios operacionais, ao mesmo tempo em que promove um ambiente mais ágil e adaptado às demandas contemporâneas.

Ao confrontar as dificuldades inerentes aos processos manuais, surge a oportunidade de não apenas modernizar as práticas institucionais, mas também de elevar a experiência dos usuários, promovendo a eficiência, a segurança e a conformidade com as expectativas de um ambiente acadêmico cada vez mais digital e dinâmico.

## 1.2 Objetivos

Tendo em vista a problemática apresentada, pretende-se desenvolver o Sistema Interno Integrado de Chaves (SIIC), um sistema *web* que deve atuar na automatização do atual registro de chaves, para facilitação dos serviços relacionados ao mesmo. Assim, o objetivo do presente trabalho é descrever a experiência na concepção e desenvolvimento deste sistema que visa facilitar o trabalho de professores e servidores em relação ao formato atual do empréstimo de chaves.

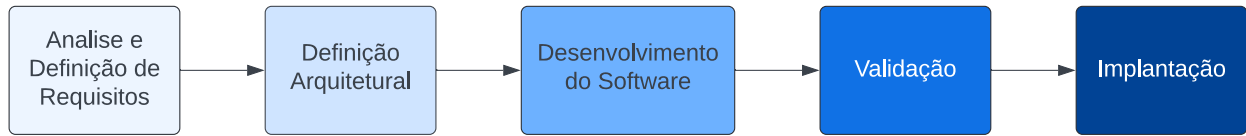
A fim de atender o objetivo geral descrito anteriormente, é necessária a observação dos seguintes objetivos específicos:

- Desenvolver uma aplicação Web para funcionários do IFPB;
- Permitir que os usuários acompanhem a disponibilidade dos espaços;
- Implementar um sistema que possibilite os registros de empréstimos de salas/chaves;
- Fácil acesso às informações;
- Permitir que o usuário notifique um problema;
- Validar os requisitos com as principais personas;

## 1.3 Metodologia

Este trabalho tem como finalidade dar continuidade a um projeto já existente que consiste em construir uma plataforma que pretende digitalizar um sistema de registro de chaves, a configuração desse trabalho será dividido nas etapas, conforme descrito na Figura 1.1.





**Figura 1.1:** *Etapas do processo de desenvolvimento.*

A **primeira etapa** do desenvolvimento consiste na análise do projeto existente e definição de requisitos ainda necessários do sistema. Os requisitos definem o que o sistema deve fazer em termos de funcionalidades e estabelecem como o sistema deve se comportar e suas restrições.

A **segunda etapa** consiste na definição da arquitetura do sistema. Desse modo, nessa etapa serão projetados em alto nível as definições dos módulos do sistema, bem como suas respectivas interações e os principais modelos de dados.

A **terceira etapa** consiste no desenvolvimento da aplicação, essa etapa envolve criação de banco de dados, criação de modelos, classes, funções, estruturas relacionadas ao *Backend* e *Frontend* entre outros. Além disso, a metodologia a ser utilizada no desenvolvimento é uma decisão importante uma vez que procuram estabelecer um processo disciplinado, com o objetivo de o tornar mais previsível e eficiente possível a realização as tarefas relacionadas a cada funcionalidade. (Awad, 2005).

A **quarta etapa** abrange verificar e documentar o sistema e o meio no qual ele atua com a intenção estabelecer maior garantia ao que já foi desenvolvido atendendo aos requisitos já definidos (Conexorama, 2018). Inicialmente, será realizada uma validação dos requisitos do sistema, verificando se de fato as funcionalidades do sistema cumprem com o que foi planejado em sua definição (Conexorama, 2018).

A **quinta etapa** foi definida como implantação, segundo (FOKUS Mercury Computer Systems, 2003), o processo de implantação de *software* é iniciado quando o *software* já está desenvolvido, empacotado e é transferido para o servidor que proporcionará o acesso aos demais serviços oferecidos.

## 1.4 Justificativa e Relevância

A automação do registro de chaves proposta no IFPB apresenta uma justificativa substancial baseada na necessidade de modernizar os processos institucionais. A sobrecarga de trabalho manual associada aos métodos tradicionais de registro impõe desafios consideráveis à eficiência operacional do IFPB, consumindo recursos valiosos que poderiam ser direcionados a tarefas estratégicas e analíticas. O SIIC surge como resposta a essa demanda, visando não apenas otimizar

a carga de trabalho, mas também proporcionar uma transição eficaz em direção a práticas mais contemporâneas e eficientes.

Além de aliviar a carga operacional, o SIIC responde às crescentes exigências de segurança da informação. A automação não só reduz o risco de erros humanos no registro de dados, mas também fortalece os controles de acesso, contribuindo para a integridade e confiabilidade das informações. Nesse cenário, a relevância do SIIC transcende a simplificação de processos; ele se torna um pilar fundamental para a proteção dos dados institucionais, atendendo não apenas às demandas presentes, mas também preparando o IFPB para os desafios futuros em um ambiente digital cada vez mais complexo.

Além disso, a escolha de tecnologias modernas e a abordagem ágil no desenvolvimento do SIIC não são apenas escolhas metodológicas, mas estratégias deliberadas para promover a inovação e a adaptação contínua. Ao alinhar-se com as melhores práticas de desenvolvimento de software, o SIIC não só assegura a eficácia imediata, mas também estabelece uma base flexível para futuras melhorias e atualizações. Esta abordagem não só contribui para a modernização do IFPB, mas também para sua capacidade de se manter ágil e relevante em um cenário educacional em constante evolução.

## 1.5 Estrutura do Trabalho

Os capítulos restantes que compõem este documento estão organizados da seguinte forma:

- **Capítulo 2: Fundamentação Teórica.** Apresentam-se definições gerais dos temas abordados neste documento.
- **Capítulo 3: Trabalhos Relacionados.** Discutem-se os principais trabalhos relacionados ao presente estudo.
- **Capítulo 4: Configuração De Estudo.** Apresenta a configuração metodológica bem como os passos realizados para a construção desse estudo.
- **Capítulo 5: SIIC.** Apresenta-se os resultados obtidos ao final do processo metodológico.
- **Capítulo 6: Validação.** Apresenta-se os métodos utilizados para validação da plataforma.
- **Capítulo 7: Ameaças a Validade.** Apresenta-se as principais ameaças identificadas.
- **Capítulo 8: Considerações Finais.** Apresentam-se as principais conclusões e desdobramentos com a realização deste trabalho.

# Capítulo 2

## Fundamentação Teórica

A digitalização e automatização dos processos é um fenômeno cada vez mais recorrente. Facilidade de obtenção de dados e celeridade nas realização de tarefas são benefícios que, assim como neste projeto, levam empresas e instituições a transformarem seus processos. Atualmente, a sociedade passa por um processo de intensificação da virtualização de tarefas corriqueiras e importantes, como transações bancárias, processos judiciais, consultas médicas, entre outras, tornando cada vez mais a realidade concreta e virtual interdependentes. (Sousa; Neto; Pessoa, 2023)

Outro fator que tem promovido a implantação de *software* nos mais variados setores e lugares da sociedade, é o avanço no acesso a internet. em 2021, o número de domicílios com acesso à internet no Brasil chegou a 90,0%, segundo dados da Pesquisa Nacional por Amostra de Domicílios. Em termos absolutos, são 65,6 milhões de domicílios conectados, 5,8 milhões a mais do que em 2019. (BRASIL, 2022)

Além do avanço do acesso a internet de qualidade, também temos o aperfeiçoamento dos sistemas *web*. Sendo assim, a virtualização do serviço público é de extrema importância para que haja um atendimento mais rápido e com respeito à dignidade da pessoa humana. (Sousa; Neto; Pessoa, 2023)

Assim, neste capítulo são introduzidos os principais conceitos necessários ao entendimento desse trabalho de conclusão de curso. Inicialmente, há de se exibir os conceitos relacionados ao desenvolvimento *web* e as tecnologias utilizadas (Seção 2.1), em seguida será mostrado o processo de desenvolvimento (Seção 2.2). Finalmente, ocorre a apresentação das especificidades acerca de projetos arquiteturais (Seção 2.3)

### 2.1 Tecnologias para Construção de Sistemas *Web*

A área da tecnologia dedicada à criação de *software*, *websites*, aplicativos e outras ferramentas que compõem a Internet é conhecida como Desenvolvimento *Web*. Para que essa vertente do desenvolvimento seja bem-sucedida, é crucial a utilização de várias tecnologias para viabilizar a conexão e a comunicação entre dispositivos. Isso inclui a interação entre o cliente ( quem faz uma solicitação) e o servidor (quem fornece respostas a uma requisição). As tecnologias empregadas

no desenvolvimento *web* podem ser divididas entre aquelas que operam no dispositivo do cliente e as que operam nos servidores. A conectividade entre esses elementos é habilitada pelas diversas camadas de rede que interligam ambos.

Para construção destes sistemas há uma grande variedade de ferramentas e tecnologias, que podem ser definidas de acordo com as necessidades do projeto e preferenciais do desenvolvedor. Dentre elas, utilizamos o *framework* Laravel para php, css, *bootstrapp*.

Nesta seção, serão apresentadas as tecnologias-chave utilizadas na construção de diversas interfaces para interação com os usuários (Seção 2.1.1), bem como aquelas aplicadas na implementação das regras de negócio de um sistema (Seção 2.1.2). Além disso, serão discutidos os aspectos específicos envolvidos na criação da camada de dados (Seção 2.1.3).

### 2.1.1 Tecnologias cliente

No âmbito deste trabalho, definiu-se as tecnologias do lado do cliente como as linguagens e *frameworks* que contribuem para a construção das páginas web. Conforme essa definição, essas tecnologias são executadas exclusivamente no navegador da máquina do usuário (Miletto; Bertagnolli, 2014). Também definiu-se esta parte da aplicação como *frontend*, sendo os exemplos mais significativos incluem HTML, CSS e JavaScript. Além delas também utilizamos no projeto o php.

O HTML (*HyperText Markup Language*) é a linguagem de marcação padrão utilizada para criar páginas da web e aplicativos web (Enterprise, 2018). Seus elementos constituem os blocos fundamentais das páginas, representando texto formatado, imagens, entradas de formulário e outras estruturas. Quando um navegador faz uma solicitação a uma URL (*Uniform Resource Locator*), seja para obter uma página ou um aplicativo, o primeiro conteúdo retornado é um documento HTML. Esse documento HTML pode fazer referência ou incluir informações adicionais sobre a aparência e o layout por meio do CSS, ou sobre o comportamento por meio do JavaScript.

O CSS (*Cascading Style Sheets*) é utilizado para controlar a aparência e o layout dos elementos HTML (Silva, 2008). Os estilos CSS podem ser aplicados diretamente a um elemento HTML, definidos separadamente na mesma página ou incluídos em um arquivo independente referenciado pela página. Os estilos são aplicados em cascata, seguindo as regras de seleção de elementos HTML. Por exemplo, um estilo pode ser aplicado a todo o documento, mas pode ser substituído por um estilo específico para um elemento em particular.

O JavaScript é uma linguagem de programação dinâmica e interpretada, padronizada pela especificação da linguagem ECMAScript (Silva, 2010). É a linguagem de programação da *web*. Similar ao CSS, o JavaScript pode ser incorporado como atributos em elementos HTML, como blocos de código em uma página ou em arquivos independentes. Ao lidar com JavaScript em um aplicativo *web*, algumas tarefas comuns incluem: (i) selecionar um elemento HTML e recuperar/atualizar seu valor; (ii) consultar dados de uma API *web*; (iii) enviar comandos para uma API *web* e lidar com os resultados das chamadas de retorno; e (iv) realizar validações.

### 2.1.2 Tecnologias servidor

A aplicação de tecnologias servidor possibilita a operação do sistema. é a parte também definida como *backend* da aplicação. Em resumo, estas tecnologias cuidam das engrenagens de uma aplicação, criando estruturas de códigos para que as funções do sistema sejam executadas. No que segue, está descrito de modo sucinto as principais tecnologias e ferramentas empregadas no presente trabalho.

O Laravel<sup>1</sup> é um *framework* de código-aberto para linguagem PHP que foi criado com o intuito de facilitar o desenvolvimento de sistemas *web* padronizados em MVC (do inglês, *model-view-controller*). Para tal, possui diversas formas de acesso a bancos de dados relacionais, uma linguagem simplificada, um sistema modular com gerenciador de dependências dedicado, entre outras características que auxiliam, tanto no desenvolvimento, como também na manutenção de sistemas *web* (Filitto, 2021)..

O PHP (do inglês, *Hypertext Preprocessor*) é uma linguagem de programação utilizada no desenvolvimento de sistemas *web*, a qual possui um diferencial de facilitar a criação de páginas com conteúdos dinâmicos, além de poder ser instruída a ser usada na execução de ações um pouco mais complexas, através do uso de códigos *scripts* (Melo, 2021).

O PHP é uma linguagem de criação de *scripts*, que pode ser incorporada em HTML, ou utilizada como um binário independente. (Converse; Park, 2003). Desta maneira, principalmente com a utilização do framework Laravel, o php é utilizado na construção da aplicação tanto do lado do cliente, como também no lado do servidor

### 2.1.3 Banco de Dados

Na ciência computacional, banco de dados é definido como uma coleção estruturada de dados. Os dados armazenados em um banco de dados são organizados de forma a permitir agilidade na busca e na recuperação por um computador, ou seja, não há nada além de uma simples coleção de itens. (Manovich, 2015)

O MySQL é um servidor e gerenciador de banco de dados (SGBD) relacional, de licença dupla (sendo uma delas de software livre), projetado inicialmente para trabalhar com aplicações de pequeno e médio portes, mas hoje atendendo à aplicações de grande porte e com mais vantagens do que seus concorrentes (Milani, 2006)

## 2.2 Processo de Desenvolvimento

O processo de desenvolvimento é composto por um conjunto de atividades e resultados necessários para construir aplicativos de alta qualidade. Ele oferece a base para o controle gerencial do projeto, definindo o contexto em que os métodos técnicos são aplicados, os produtos de trabalho

---

<sup>1</sup><<https://laravel.com/>>

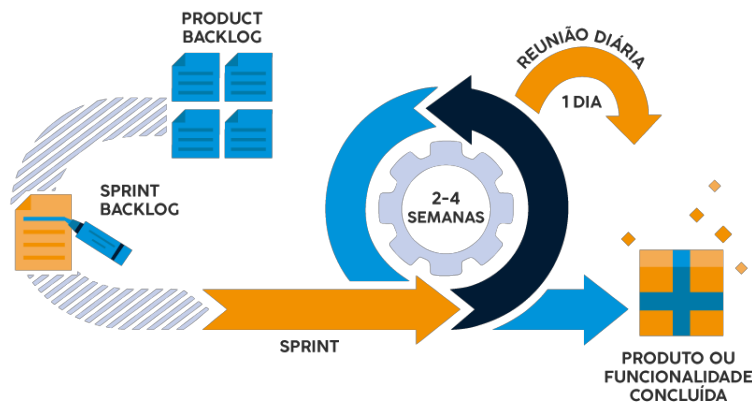
são gerados e os marcos são estabelecidos para garantir a qualidade e gerenciar as modificações (Pressman, 2006).

Existem duas metodologias principais para a gestão das atividades de desenvolvimento de software: a abordagem cascata e a abordagem ágil (Pressman, 2006). O modelo cascata é uma abordagem linear, composta por uma sequência de eventos (como análise de requisitos, definição de arquitetura, desenvolvimento, entre outros) que conduzem aos resultados. Em projetos que utilizam o Modelo Cascata, cada evento representa uma fase distinta do desenvolvimento de software, e cada fase geralmente deve ser concluída antes que a próxima possa começar (Pressman, 2006). Entretanto, essa abordagem pode ter dificuldades em lidar com requisitos complexos e alterações tardias, o que pode levar à insatisfação do cliente.

Por outro lado, o modelo Ágil é uma abordagem iterativa que enfatiza a entrega rápida de funcionalidades completas. No modelo Ágil, o tempo é dividido em iterações chamadas "*sprints*". Cada sprint tem uma duração definida e é planejado com uma lista de entregas prioritárias (??). As entregas são priorizadas com base no valor para o negócio, conforme determinado pelo cliente. Caso nem todas as entregas planejadas para um *sprint* possam ser concluídas, o trabalho é reavaliado e as informações são usadas para o planejamento futuro. À medida que as tarefas são concluídas, elas são revisadas e avaliadas pela equipe do projeto e pelo cliente por meio de compilações diárias e demonstrações de final de *sprint* (Schwaber; Sutherland, 2011).

### 2.2.1 Scrum

Uma metodologia ágil amplamente adotada é o Scrum, que é um *framework* para o gerenciamento de projetos. Ele simplifica os processos e mantém registros visíveis do progresso das etapas, facilitando o entendimento do estágio atual do projeto, do que foi concluído e do que ainda precisa ser feito (Bhavsar; Shah; Gopalan, 2020). O Scrum é baseado em papéis, como o *Product Owner*, que é responsável pelo desenvolvimento de uma lista de objetivos chamada "*Product Backlog*". Esses objetivos são distribuídos para a equipe, formando a "*Product Backlog*" que direciona as *sprints*. O ciclo de trabalho inclui reuniões como a *Sprint Planning*, *Daily Scrums* para alinhar o progresso e a *Sprint Retrospective* para avaliar o projeto e planejar a próxima *sprint* (Schwaber; Sutherland, 2011). A figura a seguir ilustra esse processo:



**Figura 2.1:** *Funcionamento do Scrum* <sup>2</sup>.

Essa metodologia, com seu foco na entrega de valor ao cliente e na colaboração constante, torna-se uma abordagem eficaz para o desenvolvimento de *software* em ambientes dinâmicos e em constante mudança.

### 2.2.2 Kanban

Outra metodologia ágil bastante utilizada é o *Kanban*. Ela se baseia em um quadro que divide as tarefas em colunas *TO DO* (a fazer), *IN PROGRESS* (fazendo) e *DONE* (concluído). Todos os requisitos são colocados na primeira coluna do quadro e cada membro da equipe fica responsável por implementá-los e entregá-los. Cada tarefa tem sua descrição, prioridade e seu prazo de entrega. Assim como no *Scrum*, as tarefas prioritárias devem ser executadas primeiro. A ideia central do *Kanban* é o foco na produtividade. Cada membro não pode executar mais de uma tarefa por vez. Seu método é tão simples que pode ser personalizado de inúmeras maneiras para atender as demandas de uma empresa.

O ponto central do *Kanban* é aumentar a produtividade e eficiência. Ele enfatiza que cada membro da equipe deve focar em uma tarefa por vez, evitando sobrecarga e facilitando a concentração. A simplicidade do método *Kanban* é notável, e ele pode ser personalizado de várias maneiras para atender às demandas específicas de cada empresa ou projeto. A abordagem *Kanban* é altamente flexível e permite que as equipes adaptem suas práticas de acordo com suas necessidades, tornando-a uma opção valiosa para ambientes com requisitos variados e em constante mudança.

## 2.3 Projeto Arquitetural

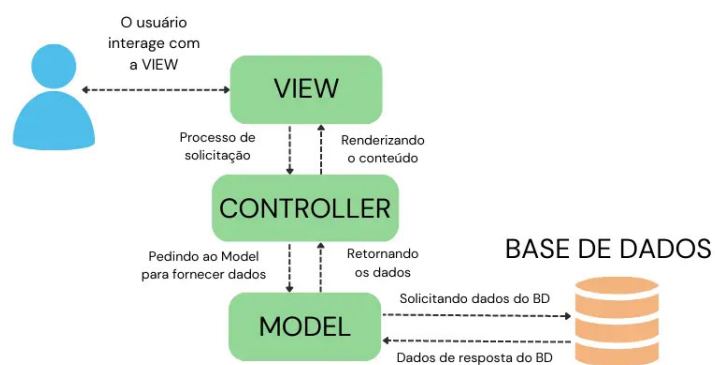
A arquitetura de software de um programa ou sistema computacional é a estrutura ou estruturas do sistema, que abrange os componentes de software, as propriedades externamente visíveis desses componentes e as relações entre eles (PRESSMAN, 1995)

Uma boa arquitetura de software auxilia na produção de um código mais organizado e de fácil manutenção. O Laravel utiliza, em sua estrutura base, a arquitetura MVC (Model, View e Controller), a qual foi utilizada na implementação deste projeto.

#### Arquitetura MVC

Basicamente, a arquitetura MVC é um padrão de arquitetura de software que divide a aplicação em três camadas: manipulação dos dados (model); interação do usuário (view); camada de controle (controller). (Silva, 2021) Esta divisão pode ser observada na figura 2.2 que nos mostra quais componentes constituem essa arquitetura e como funciona a comunicação entre elas

**Figura 2.2:** *Arquitetura MVC* <sup>3</sup>.



**Figura 2.3:** *Fonte: UsandoPy (2023)*

O principal objetivo do MVC é separar dados lógicos (Model) da interface do usuário (View) e as ações da aplicação (Controller). A ideia é deixar que uma mensagem do banco de dados seja acessada e visualizada através de várias interfaces. (Barbosa; al, 2017)



# Capítulo 3

## Trabalhos Relacionados

A análise de sistemas de gerenciamento de chaves físicas para salas e laboratórios revela uma série de iniciativas que compartilham desafios semelhantes aos enfrentados pelo Sistema de Automação de Registro de Chaves (SIIC). A seguir, está destacado alguns estudos notáveis que contribuem para o entendimento do contexto específico abordado por este trabalho.

Uma solução bastante relevante a ser considerada é o "Organize Rooms"(Mayron, 2022) da Universidade Federal do Ceará (UFC), um sistema online desenvolvido por Matheus Mayron Lima, servidor técnico-administrativo da UFC, para facilitar o gerenciamento de salas de aula no Centro de Humanidades. O sistema permite que chefes de departamento e coordenadores de cursos solicitem reservas de salas por períodos que variam de um dia a um semestre. Após avaliação, os pedidos podem ser aprovados, rejeitados ou receber indicação de outra sala. O software também fornece informações sobre as condições físicas das salas, como ar-condicionado e projetor, auxiliando na gestão de infraestrutura.

Outra solução é a plataforma desenvolvida, denominada "PyControlRoom"(HORNICHE, ) é uma aplicação web administrativa educacional criada para otimizar o gerenciamento de salas de aula em instituições educacionais. Utilizando a linguagem de programação Python, juntamente com o *framework* Django, a plataforma oferece funcionalidades como cadastro de usuários (professores, alunos, funcionários), cadastro de salas, alocação e consulta de reservas, além de outras operações relacionadas à gestão acadêmica.

O "Sistema de Gerenciamento de Chaves (SGC)"(Freitas, 2023), desenvolvido no IFBA, é um aplicativo que busca informatizar e otimizar o processo de gerenciamento de chaves. O aplicativo proposto visa substituir o processo manual atualmente utilizado, oferecendo uma solução automatizada e eficiente que reduza as dificuldades de controle, perdas de chaves e os custos associados a essas questões.

Ao contextualizar esses sistemas de gerenciamento de chaves físicas, é possível notar tanto semelhanças quanto distinções em relação ao SIIC. Embora cada projeto aborde o controle de chaves físicas de maneira específica, compartilham a mesma preocupação com a segurança e rastreabilidade das chaves em ambientes sensíveis, refletindo a importância de soluções eficazes e confiáveis para o gerenciamento de acesso em laboratórios e salas de alta segurança.

**Diferencial do SIIC:** Este sistema se destaca ao integrar não apenas a automação do controle de chaves, mas também um registro digital abrangente que transcende a mera gestão física. Sua abordagem inovadora foca não apenas na segurança física das chaves, mas também na precisão e confiabilidade do registro, oferecendo uma solução mais completa para os desafios enfrentados no contexto institucional do IFPB. Enquanto os sistemas mencionados se concentram predominantemente na gestão física, o SIIC busca otimizar processos manuais, aprimorar a eficiência operacional e garantir a integridade das informações em um ambiente acadêmico dinâmico e em constante evolução.

# Capítulo 4

## Processo de Desenvolvimento do SIIC

Neste capítulo são descritas todas as etapas metodológicas do presente trabalho. Para guiar os esforços do trabalho foram seguidas algumas etapas.

Na etapa 1 (Seção 4.1) foi realizada a identificação e análise de requisitos com vistas a obter as principais premissas, restrições e funcionalidades da aplicação. Na etapa 2 (Seção 4.2) teve a escolha das tecnologias utilizadas para o desenvolvimento do sistema SIIC. Na etapa 3 (Seção 4.3) foi realizado o processo de desenvolvimento visando atender os requisitos bem como a arquitetura, e posteriormente, a etapa 4 (Seção 4.4) foi executada pela realização do processo de teste mais abrangente bem como a validação do sistema SIIC junto ao cliente demandante. Finalmente, na etapa 5 (Seção 4.5), o sistema SIIC foi colocado em produção em uma infraestrutura baseada em computação na nuvem. Cada uma dessas etapas é descrita em detalhes nas próximas seções.

### 4.1 Análise e Definição dos Requisitos

Visando levantar os requisitos que são necessários para a construção do sistema SIIC, foi identificado uma série de técnicas de levantamento de requisitos que podem ser importantes. As principais técnicas utilizadas foram entrevista e prototipagem. A técnica de entrevista pelo qual seu próprio nome já faz referência é quando o desenvolvedor abre espaço para o cliente expor suas ideias e expectativas no que diz respeito ao que este espera do sistema. No caso do sistema SIIC, os dados foram coletados com alguns professores do IFPB do *campus* Campina Grande, através de reuniões sistemáticas, sendo possível fazer o alinhamento e a definição dos requisitos.

Após a definição de uma lista preliminar de requisitos funcionais e não-funcionais (vide Tabelas 4.1 e 4.2), utilizou-se da técnica de prototipagem a fim de explorar e validar os principais aspectos críticos dos requisitos e minimizar os riscos na construção do sistema.

### 4.2 Escolha das Tecnologias

Com o esclarecimento das funcionalidades e fluxos do sistema SIIC, a escolha das tecnologias para seu desenvolvimento do sistema é um fator importante para o sucesso do projeto.

**Tabela 4.1:** *Lista de requisitos funcionais*

ID	REQUISITOS FUNCIONAIS
RF01	Adicionar, editar e remover usuários
RF02	Adicionar, editar e remover chaves
RF03	Adicionar, editar e remover salas e laboratórios
RF04	Adicionar, editar e remover materiais da sala
RF05	O sistema deverá armazenar e informar sobre problemas em uma sala ou laboratório
RF06	Confirmar a resolução de um problema em alguma sala ou laboratório
RF07	Gerar relatórios com base no nome, período e/ou sala
RF08	Listar chaves disponíveis e que já foram reservadas

**Tabela 4.2:** *Lista de requisitos não-funcionais*

ID	REQUISITOS NÃO-FUNCIONAIS
RNF01	O sistema deverá ser implementado em PHP com <i>framework</i> Laravel
RNF02	O sistema deverá ser seguro
RNF03	O sistema deverá ter uma interface amigável, de fácil utilização e responsiva
RNF04	O sistema deverá ter desempenho aceitável
RNF05	O sistema deverá ter alta disponibilidade

O desenvolvimento de sistemas pode ser um processo difícil e arriscado, já que projetar produto de *software* não é apenas ter uma interface do usuário conveniente, mas também garantir que o produto seja estável, seguro e fácil de manter, sendo também necessário garantir que ele atenda aos requisitos funcionais e não-funcionais.

Diante disso, durante o planejamento do SIIC foram selecionadas tecnologias que visavam atender da melhor forma possível todas as necessidades do sistema, considerando a facilidade de desenvolvimento, estabilidade, segurança e manutibilidade.

**Tecnologias Cliente.** As tecnologias cliente designadas foram o HTML5, CSS3, *JavaScript* e o Blade.

O HTML5 é comumente usado com outras tecnologias, como o CSS3 e o JavaScript, e por conta dessa facilidade é possível desenvolver uma aplicação mais completa e robusta sem tantas dificuldades. Além disso, também possui a vantagem de funcionar bem em todos os navegadores *web*, e ser uma ferramenta de fácil aprendizado com grande quantidade de material de estudo de qualidade.

CSS3 tem o importante papel de melhorar o visual das páginas do SIIC. Ele pode ser rapidamente implementado e facilmente modificado, e possui a vantagem de ser aplicado em diversas páginas simultâneas, gerando economia de tempo e reuso de código. Com o CSS3 pudemos também organizar layouts para adaptar os blocos para diferentes tamanhos e resoluções de telas, sendo possível criar regras para mudar as configurações das telas de acordo com o dispositivo que

estiver acessando o SIIC. Além de todas essas facilidades e benefícios, o CSS3 garante que o sistema funcionará perfeitamente em qualquer navegador *web*, eliminando a necessidade de adaptar o sistema para diferentes *browsers*.

O *JavaScript* (JS) foi a terceira tecnologia cliente escolhida para uso no desenvolvimento do SIIC. Com ele, o desenvolvimento pôde ser mais ágil, haja vista que o JS é uma tecnologia simples de ser aprendida, com uma literatura vasta, sendo também compatível com diversas plataformas, *frameworks* e *browsers*. Com a utilização do JS, foi possível controlar o comportamento do navegador e tornar o site mais dinâmico. Os códigos desenvolvidos em *JavaScript* puderam ser executados naturalmente em conjunto aos códigos HTML e CSS.

Todas as tecnologias cliente do SIIC foram implementadas segundo a ferramenta *Blade*, que é o *template engine* do *framework Laravel*, é simples de usar e extremamente poderoso. Ele possui uma sintaxe expressiva que ajuda na legibilidade das *views*, além de ter uma curva de aprendizado rápida.

**Tecnologias Servidor.** A linguagem PHP foi escolhida como tecnologia servidor mais relevante na implementação do sistema SIIC. Esta linguagem se tornou uma das linguagens de programação mais usadas no mundo. As vantagens que ela proporciona atendem às necessidades tanto de usuários iniciantes quanto experientes em programação e desenvolvimento para internet. Esta linguagem é de fácil aprendizagem; possui alto desempenho; é de código aberto; é multiplataforma; possui compatibilidade com diversos bancos de dados; possui compatibilidade com serviços de hospedagens. Tais diferenciais foram preponderantes para a escolha dessa linguagem.

O MVC foi escolhido para o desenvolvimento do SIIC pois ele é um padrão de arquitetura de *software* que melhora a conexão entre as camadas de dados, lógica de negócio e interação com usuário. Através da sua divisão em três camadas interconectadas, o processo de desenvolvimento se torna mais simples e dinâmico. Nesse sentido, após uma busca sistemática a respeito de qual *framework* MVC da linguagem PHP seria eleito para o desenvolvimento do SIIC, chegou-se à escolha do *Laravel*. O *Laravel* é um dos *framework* mais populares da atualidade, possui uma equipe de desenvolvedores ativa, além de ter uma gigantesca comunidade e bastante aceitação no mercado. Este *framework* traz consigo a perspectiva de tornar o sistema mais seguro, protegendo os códigos e os dados dos usuários, atendendo justamente a uma das nossas preocupações para com o sistema. A partir dele, também, desenvolvemos a autenticação do sistema, por meio de ferramentas e técnicas do próprio *framework*, o qual possui a capacidade de simplificar a restrição de acesso apenas a conteúdos não autorizados.

**Camada de dados.** Para criação do modelo conceitual foi utilizado a ferramenta BRModelo<sup>1</sup> enquanto que para geração dos modelos lógicos foi utilizada a ferramenta *workbench*<sup>2</sup>. Tais modelos foram criados de modo iterativo e, mediante discussão entre os autores do presente trabalho e um especialista em banco de dados, chegou-se a uma versão final de ambos os modelos. A ferramenta MariaDB<sup>3</sup> foi eleita como sistema para gerenciamento de banco de dados. O MariaDB

---

<sup>1</sup><<http://www.sis4.com/brModelo/>>

<sup>2</sup><<https://mariadb.com/products/skysql/docs/connect/clients/mysql-workbench/>>

<sup>3</sup><<https://mariadb.com/>>

é um dos bancos de dados mais conhecidos do mundo, criado pelos mesmos desenvolvedores do MySQL, que mantiveram a estrutura de código aberto. Sua principal característica é a rapidez, escalabilidade e robustez de suas ferramentas, *plugins* e, claro, capacidade de armazenamento.

Outro motivo, que justifica a escolha das ferramentas citadas, é que o projeto aqui desenvolvido foi feito utilizando como base um código anteriormente produzido por um estudante do IFPB para esse mesmo propósito, gerenciar as reservas de chaves das salas do instituto.

### 4.3 Processo de Desenvolvimento

Para o andamento do processo de desenvolvimento do projeto, foi utilizada a metodologia *Scrum* para que houvesse uma maior agilidade dentro dos horários dos membros autores deste trabalho. Nesse aspecto, o projeto foi dividido em *releases*, *sprints* e *tasks* a serem cumpridas dentro de um determinado tempo pré-estabelecido. Assim, o processo de desenvolvimento do SIIC se deu em 10 *sprints*, conforme demonstrado na Tabela 4.3, no qual possui suas determinadas *tasks* a serem realizadas em um período de tempo estabelecido.

É importante mencionar que a divisão das tarefas foi feita de maneira a se adequar com a complexidade da *sprint*. Nesse aspecto, as reuniões diárias foram feitas entre a equipe de desenvolvimento e pelo menos uma reunião semanal com o orientador para acompanhamento do projeto.

Um dos momentos desafiadores desse da implementação desse projeto foi a refatoração do código utilizado como base, a reescrita de funções e reorganização das classes e das regras de negocio exigiram um trabalho maior do que era esperado para essa etapa do desenvolvimento. O que foi útil para motivar a produção de um código de fácil manutenção.

O código do projeto está disponível em: "<https://github.com/lucasdantas2014/siic/>"<sup>4</sup>

### 4.4 Validação e Testes

No processo de validação, identificamos oportunidades para aperfeiçoar alguns fluxos do sistema bem como sua forma de operação e funcionamento. Para tal, utilizamos tanto os protótipos de alta fidelidade, como também a própria aplicação. E, para acrescentar credibilidade à validação, em meio à uma pesquisa realizada com possíveis futuros usuários do sistema (mais detalhes no Capítulo 6), apresentamos brevemente o SIIC (Capítulo 5) e suas principais funcionalidades, e pedimos opiniões a respeito das funcionalidades demonstradas, bem como possíveis novas funções não abordadas. Desse modo, tivemos um acompanhamento da eficiência e aceitação do sistema antes de estar disponível para os usuários finais, além de termos um parâmetro visual de *design* aprovado para ser a base de desenvolvimento para o *front-end*.

---

<sup>4</sup><https://github.com/lucasdantas2014/siic/>

**Tabela 4.3:** Cronograma de execução do projeto

<b>Sprint</b>	<b>Tarefa</b>	<b>Data Início</b>	<b>Data Fim</b>	<b>DoD</b>
1	Análise e definição dos requisitos	01/03/2023	08/03/2023	Reuniões com a equipe para o alinhamento das funcionalidades do sistema.
2	Draft protótipo de baixa fidelidade	08/03/2023	15/03/2023	Fluxo das principais telas e funcionalidades do SIIC em uma ferramenta de modelagem.
3	Análise do projeto	15/03/2023	22/03/2023	Análise do projeto que já foi iniciado pro outro aluno.
4	Configuração do ambiente	22/03/2023	29/03/2023	Criação do repositório da aplicação; Servidor e banco de dados;
5	Refatoração e ajustes	29/03/2023	05/04/2023	Refatoração de funcionalidades e telas já existentes do projeto.
6	Fluxos do usuário Comum	05/04/2023	03/05/2023	Login e Tela do professor; Visualizar reservas; Relatar problema; Visualizar problemas; Registrar problema resolvido.
7	Fluxos do usuário Admin	10/05/2023	24/05/2023	Login e tela do admin; Visualizar resevas; Visualizar salas; Visualizar chaves; Fazer devolução; Geração de relatórios;
8	Ajustes	24/05/2023	31/05/2023	Ajustes de telas e código.
9	Validação dos fluxos	31/05/2022	07/06/2023	Checagem ao atendimento dos requisitos inicialmente propostos para o sistema.
10	Testes	07/06/2023	14/06/2023	Utilização do sistema por parte dos alunos desenvolvedores, professor da disciplina e apresentação do sistema para o professor da disciplina de PEC.

## 4.5 Implantação

A implantação consiste em colocar o sistema em produção, é o momento em que o sistema passará a ser usado pelos usuários finais. Para realização desta etapa se faz fundamental a adoção prévia de uma infraestrutura a ser seguida durante o processo de desenvolvimento do projeto, de modo que, sejam minimizados os possíveis e plausíveis problemas que possam vir a ocorrer.

Posteriormente, também se faz necessária a definição de um processo de *deploy* a ser seguido pelos integrantes da equipe de desenvolvimento, principalmente pela motivação de haver um controle de versionamento dos códigos construídos. Para o desenvolvimento do SIIC, o GIT foi a ferramenta escolhida para controle de versão do sistema, sendo utilizado também o seu aplicativo para *desktop* (i.e., GIT Bash). Esta tecnologia permite manter um histórico de todos os pontos de alterações dos projetos ou arquivos, permitindo também que cada pessoa trabalhe em uma versão diferente do mesmo projeto, impedindo que, ao realizar uma alteração, altere a versão que outra pessoa está trabalhando. Dessa forma, caso haja algum erro, este não irá refletir na versão de outra pessoa e, se precisar, é só voltar para uma versão anterior em que o projeto estava funcionando perfeitamente.

Finalmente, o SIIC será hospedado no provedor de hospedagem *web* Hostinger<sup>5</sup>. Esta plataforma foi escolhida em virtude de ser um provedor de serviços de computação em nuvem de relevância em diversos países. Adicionalmente, foi identificado que sua base de serviços possuía excelente custo benefício para o presente sistema *web*. É importante destacar que após o término do processo de implantação da plataforma, buscou-se monitorar diversos indicadores disponibilizados através da infraestrutura Hostinger. Dados como tempo de resposta, latência de banco de dados, tempo entre requisições foram identificados e monitorados a fim de assegurar a correta operação do sistema SIIC. A plataforma desenvolvida está disponível em: "<https://dantas.tech/siic/painel>"<sup>6</sup>

Durante o processo para disponibilizar a aplicação na *internet* foram encontrados alguns desafios. Uma das situações, que foi preciso contornar, está relacionada aos custos de de hospedagem de *sites* de muitas plataformas, o preços dos planos disponíveis em muitas plataformas não condiziam com o orçamento existente para o projeto. Além disso, outro desafio encontrado foi o tempo de aprendizado para utilizar a plataforma de hospedagem escolhida, pois a equipe só possuía experiência com outras soluções de hospedagem.

---

<sup>5</sup><<https://www.hostinger.com.br/>>

<sup>6</sup><<https://dantas.tech/siic/painel>>



# Capítulo 5

## O Sistema Interno Integrado de Chaves

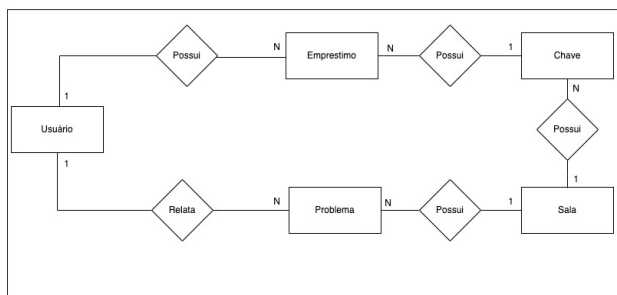
Este capítulo descreve os principais resultados obtidos a partir da realização do presente estudo. Inicialmente, é descrito as principais funcionalidades da plataforma SIIC de acordo com todas as atividades do processo de desenvolvimento (Seção 5.1).

### 5.1 A Plataforma SIIC

Esta seção tem por objetivo demonstrar os principais artefatos de software desenvolvidos de acordo com a metodologia. Em cada seção a seguir, estão descritos os fluxos principais da plataforma SIIC, com foco especial na camada de *frontend*, para ilustrar ao leitor a forma de operação e as principais interações que o usuário pode realizar. É descrito a área inicial da plataforma (Seção 5.1.1), as funcionalidades para usuários administradores (Seção 5.1.2) e as funcionalidades para usuários comuns, como professores e técnicos (Seção 5.1.3).

#### 5.1.1 Modelagem do Sistema

O sistema contém 5 entidades principais: Usuário, Chave, Sala, Pedido e Problema. Através da comunicação e da relação entre essas entidades, o sistema realiza a seu objetivo. Assim, um usuário pode possuir vários pedidos, esses pedidos contém uma chave que pertence a uma sala. O usuário também pode ter uma lista de problemas relatadas, que por sua vez, possuem uma sala nas quais foram identificados estes problemas. Na figura abaixo, podemos observar o modelo Entidade Relacionamento que ilustra estas relações.



## 5.1.2 Funcionalidades Públicas

A seguir, apresentaremos as principais seções da página de entrada da plataforma SIIC. Ao acessar o SIIC, o usuário encontra a tela de login, onde é necessário inserir suas credenciais para entrar no sistema.

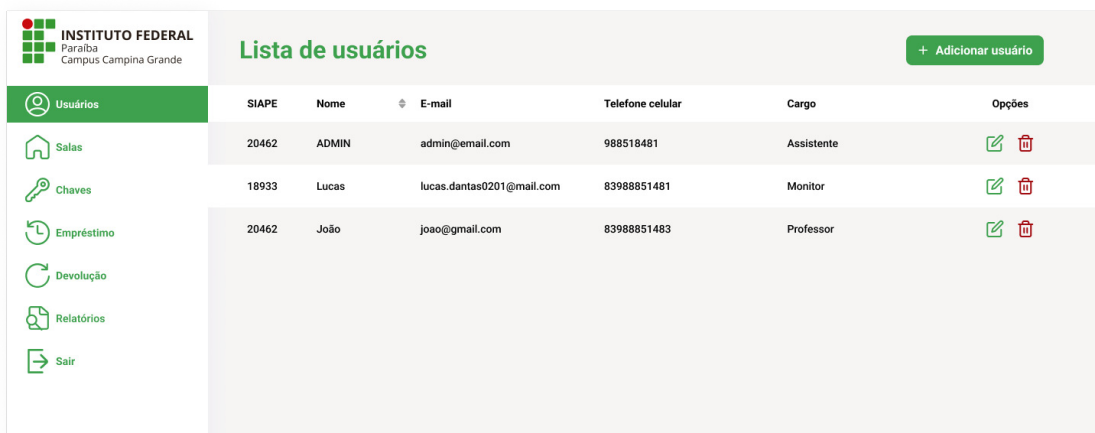


A tela de login, intitulada "Entrar", possui um fundo verde escuro com uma borda branca arredondada. No topo, o título "Entrar" é exibido em uma fonte grande e preta. Abaixo dele, o texto "Insira seu n° SIAPE e senha" orienta o usuário. Há dois campos de entrada: "SIAPE" e "Senha", ambos com bordas cinzas e ícones de lupa para busca. Um botão verde com o texto "Entrar" em branco está posicionado na base do formulário.







Após o login, a página inicial é direcionada de acordo com o tipo de usuário, sendo este um administrador ou um usuário comum.

## 5.1.3 Funcionalidades do Usuário administrador

O administrador tem acesso a todas as funcionalidades do sistema, desempenhando um papel crucial no controle e acompanhamento das atividades. O menu lateral oferece opções abrangentes para gerenciamento de usuários, salas, chaves, empréstimos, devoluções e relatórios.



A interface de administração de usuários apresenta o logotipo do Instituto Federal Paraíba no canto superior esquerdo. O título "Lista de usuários" está no topo central, acompanhado de um botão "+ Adicionar usuário" no canto superior direito. Um menu lateral à esquerda contém ícones para "Usuários", "Salas", "Chaves", "Empréstimo", "Devolução", "Relatórios" e "Sair".

SIAPE	Nome	E-mail	Telefone celular	Cargo	Opções
20462	ADMIN	admin@email.com	988518481	Assistente	 
18933	Lucas	lucas.dantas0201@mail.com	83988851481	Monitor	 
20462	João	joao@gmail.com	83988851483	Professor	 

Com o administrador possuindo acesso a todas as funcionalidades, também vêm consigo as responsabilidades de se ter controle sobre o sistema. O administrador é responsável pelo cadastro, alteração e remoção de usuários do sistema, assim cumprindo com o primeiro requisito funcional estabelecido que consistiu em adicionar, editar e remover usuários. A adição de um novo usuário é realizada por meio do preenchimento dos dados necessários em uma tela dedicada.

Ao clicar no botão "Adicionar Usuário", o administrador vê uma aba com espaços para inserir os dados do novo usuário.

The screenshot displays the 'Adicionar usuário' (Add user) form within the SIIC system. The interface features a green sidebar on the left with navigation icons for 'Usuários', 'Salas', 'Chaves', 'Empréstimo', 'Devolução', 'Relatórios', and 'Sair'. The main content area is titled 'Adicionar usuário' and contains several input fields: 'Nome do usuário', 'SIAPE do usuário', 'Senha do usuário' (with a toggle for visibility), 'Telefone celular do usuário', and 'E-mail do usuário'. Below these are three dropdown menus for 'Setor do usuário', 'Cargo do usuário', and 'Tipo do usuário', each with the placeholder text 'Selecione uma opção'. A green 'Cadastrar' button is positioned at the bottom of the form. The footer of the page reads 'Instituto Federal da Paraíba - Campus Campina Grande - Direitos'.

O administrador também vai ser responsável pelo gerenciamento das salas. A administração de salas envolve o cadastro, verificação de dados e gerenciamento da disponibilidade. O administrador pode adicionar uma nova sala, fornecendo as informações pertinentes. Essa funcionalidade se refere ao terceiro requisito funcional do sistema que é adicionar, editar e remover salas e laboratórios.

Nome	Categoria	Descrição	Status	Opções
Sala 1	Padrão	Sala 1 do bloco de aulas	Disponível	
Sala 2	Padrão	Sala 2 corredor central	Indisponível	

Ao clicar no botão de "Adicionar sala", o administrador ira ver uma aba com os espaços em que ele informa os dados da sala e ao clicar no botão "Cadastrar" a sala será adicionado ao sistema.

**Adicionar sala**

Nome da sala

Categoria do laboratório  
Selecione uma opção ▼

Descrição

Cadastrar

Em conformidade com o requisito funcional 2, adicionar, editar e remover chaves, a seção de chaves permite ao administrador gerenciar as informações relacionadas a esses elementos, de maneira semelhante às funcionalidades anteriores.

Chave	Sala	Descrição	Status	Opções
LABINFO2	Laboratório de informática 2	Assistente	Indisponível	
SALA01	Sala 1	Monitor	Indisponível	

**Adicionar chave**

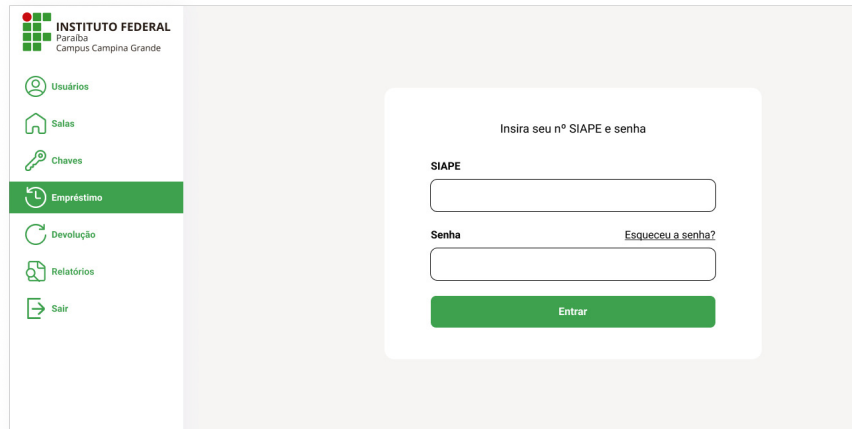
Nome da chave

Sala  
Selecione uma opção

Descrição

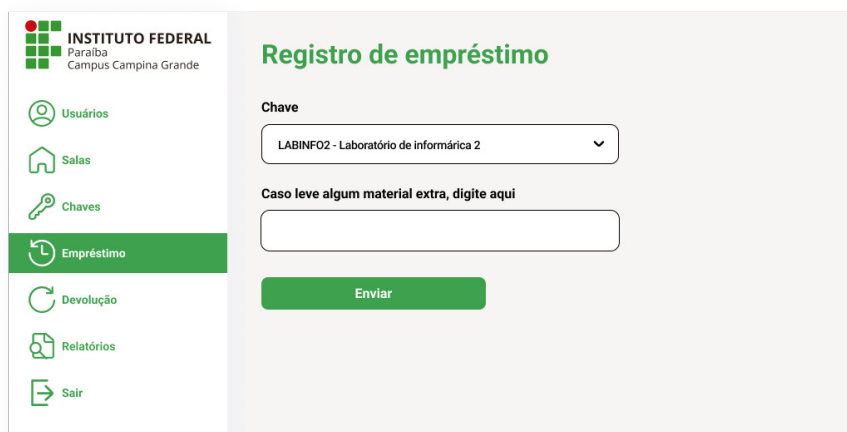
Cadastrar

Na seção de "Empréstimos" do SIIC, o administrador assume a responsabilidade pelo controle e registro das reservas de salas ou laboratórios. Essa funcionalidade é projetada para oferecer aos usuários a capacidade de agendar e utilizar esses espaços de maneira eficiente. Cumprindo, assim, o requisito funcional 05.

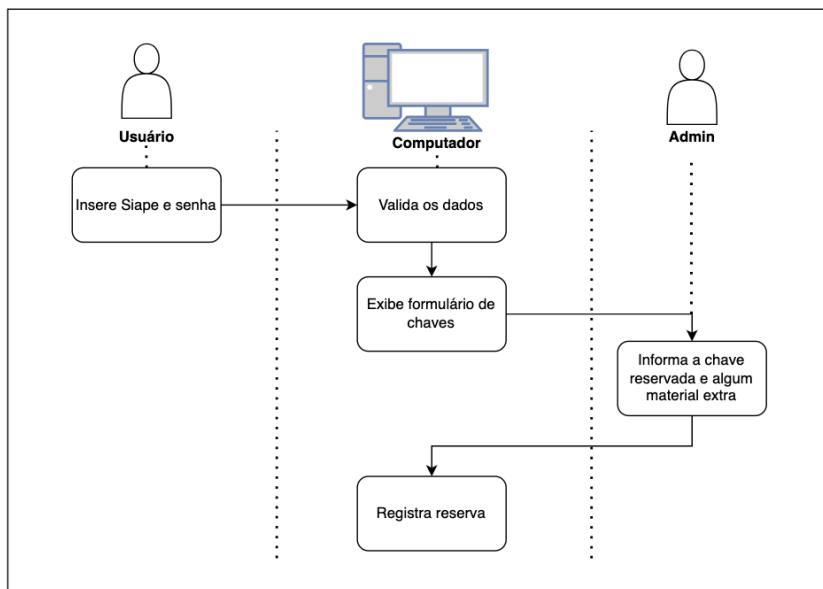


The screenshot shows the login interface of the SIIC platform. On the left is a vertical navigation menu with the following items: 'Usuários', 'Salas', 'Chaves', 'Empréstimo' (highlighted in green), 'Devolução', 'Relatórios', and 'Sair'. The main content area is titled 'Insira seu nº SIAPE e senha'. It contains two input fields: 'SIAPE' and 'Senha'. To the right of the 'Senha' field is a link that says 'Esqueceu a senha?'. Below the input fields is a green button labeled 'Entrar'.

Ao clicar no botão "Entrar", o administrador é redirecionado para uma tela onde pode informar a chave e, se necessário, adicionar detalhes adicionais, como materiais extras. Ao clicar em "Enviar", o empréstimo é registrado no sistema, contribuindo para uma gestão organizada e transparente dos recursos institucionais.

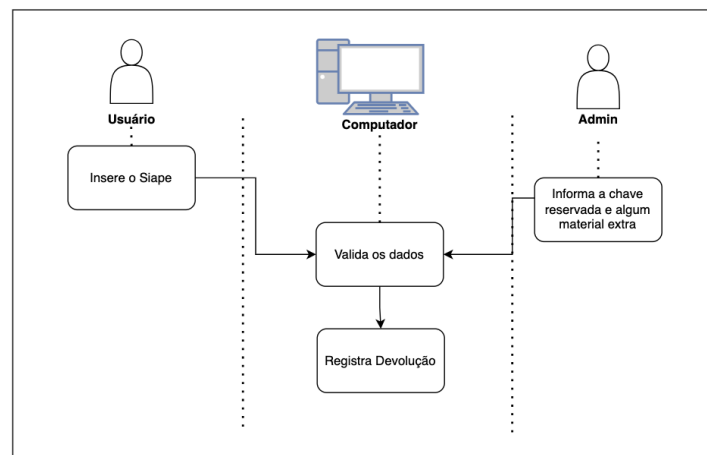


The screenshot shows the 'Registro de empréstimo' (Borrowing Record) page. The left navigation menu is identical to the previous screenshot, with 'Empréstimo' highlighted. The main content area has the title 'Registro de empréstimo' in green. It features a 'Chave' (Key) dropdown menu with the selected value 'LABINFO2 - Laboratório de informática 2'. Below this is a text input field with the placeholder 'Caso leve algum material extra, digite aqui'. At the bottom is a green button labeled 'Enviar'.



O diagrama acima ilustra o processo de devolução, no qual o usuário insere seus dados e o administrador informa a chave e os materiais que o usuário deseja reservar. Toda essa operação pode ser feita com a utilização de apenas um computador de forma que o usuário tenha um periférico, por exemplo um teclado, para inserir seus dados e o administrador possa selecionar a chave e materiais desejados.

Na área de "Devoluções" do SIIC, o administrador desempenha um papel fundamental no registro e resolução de problemas relacionados aos espaços reservados. Ao realizar a devolução, o administrador tem a oportunidade de relatar problemas encontrados, contribuindo para melhorias futuras na experiência dos usuários. Esse processo não apenas simplifica a devolução, mas também fortalece a comunicação entre os usuários e a administração, garantindo um ambiente institucional mais eficaz e adaptado às necessidades dos usuários.



Na imagem acima, está ilustrado o processo de devolução onde o usuário informa o sua matrícula e o administrador informa a chave que será devolvida, então o sistema registra a devolução.

Cumprindo o requisito funcional 07, gerar relatórios com base no nome, período e/ou sala) a funcionalidade de "Relatórios" permite ao administrador filtrar informações de maneira eficaz, melhorando a qualidade e agilidade na busca por dados específicos relacionados ao instituto.



**INSTITUTO FEDERAL**  
Paraíba  
Campus Campina Grande

### Gerar relatórios

**Filtrar pela data de empréstimo**

Data início:

Data fim:

**Filtrar pela data de devolução**

Data início:

Data fim:

**Filtrar pela chave**

Todas as Chaves

**Filtrar por categoria de sala**

Todas as Categorias

**Filtrar por usuário**

SIAPE

SIAPE do usuário

Enviar

### 5.1.4 Funcionalidades do Usuário Comum

Para os usuários comuns, a tela de login é semelhante à do administrador. Após o login, o usuário tem acesso a funcionalidades específicas relacionadas às suas atividades.

## Entrar

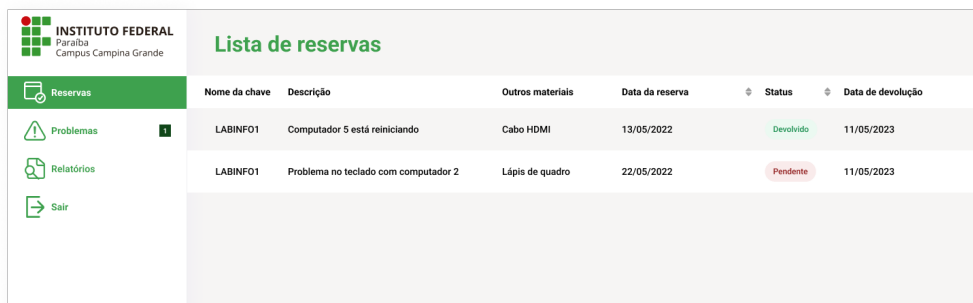
Insira seu n° SIAPE e senha

SIAPE

Senha

Entrar

Dentro da plataforma SIIC, os usuários comuns desfrutam da facilidade de notificar problemas e consultar histórico de reservas de salas e laboratórios de forma direta no sistema. No entanto, vale ressaltar que a capacidade de efetuar reservas deve ser restrita aos administradores. O que está de acordo com o requisito funcional 06: O sistema deverá armazenar e informar sobre problemas em uma sala ou laboratório.



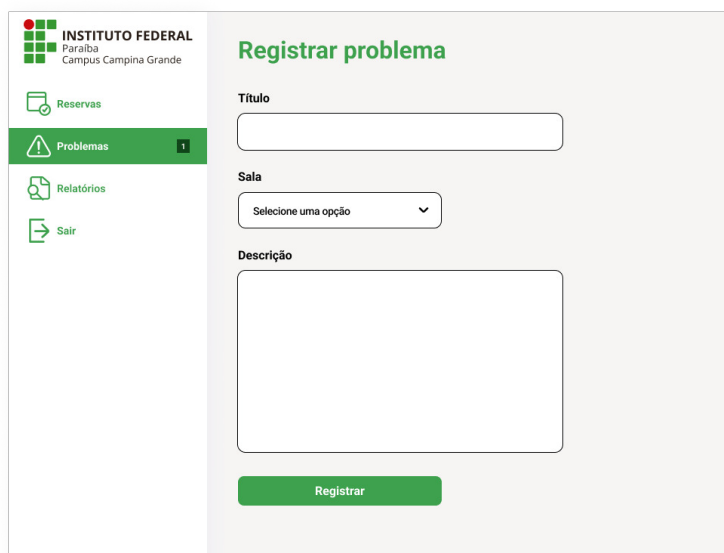
Nome da chave	Descrição	Outros materiais	Data da reserva	Status	Data de devolução
LABINFO1	Computador 5 está reiniciando	Cabo HDMI	13/05/2022	Devolvido	11/05/2023
LABINFO1	Problema no teclado com computador 2	Lápis de quadro	22/05/2022	Pendente	11/05/2023

Os usuários comuns podem contribuir notificando problemas ou informando que um problema existente já foi resolvido.



Nome da chave	Descrição	Sala	Resolvido em	Data de registro	Status	Opções
LABINFO1	Computador 5 está reiniciando	Sala 1	13/05/2022	11/05/2023	Resolvido	Resolver
LABINFO1	Problema no teclado com computador 2	Sala 1	22/05/2022	11/05/2023	Pendente	Resolver

Ao clicar no botão "Registrar Novo Problema", os usuários são redirecionados para uma tela onde podem fornecer detalhes sobre questões encontradas, tornando o processo de resolução mais ágil e eficiente.



**Registrar problema**

**Titulo**

**Sala**

Selecione uma opção

**Descrição**

Registrar

A funcionalidade de "Relatórios" no SIIC oferece aos usuários comuns a capacidade de gerar relatórios sem depender da disponibilidade do administrador. Isso elimina a necessidade de aguardar a geração de relatórios, proporcionando aos usuários comuns acesso imediato a informações relevantes. Essa abordagem visa agilizar a obtenção de dados, contribuindo para uma gestão mais eficiente e informada dos recursos do IFPB.

**INSTITUTO FEDERAL**  
Paraíba  
Campus Campina Grande

Reservas

Problemas **1**

**Relatórios**

Sair

## Gerar relatórios

**Filtrar pela data de empréstimo**

Data início:

Data fim:

**Filtrar pela data de devolução**

Data início:

Data fim:

**Filtrar pela chave**

Todas as Chaves

# Capítulo 6

## Validação do SIIC

Esta seção aborda as etapas realizadas para avaliar e validar a plataforma SIIC (Sistema Integrado de Registro de Chaves) no Instituto Federal da Paraíba (IFPB) com foco no uso exclusivo de servidores e professores. A atual abordagem de registro manual de chaves no IFPB impulsionou a necessidade de desenvolver uma solução tecnológica mais eficiente. Nas subseções a seguir, é descrito o projeto do questionário (Seção 6.1), a definição da amostra de participantes (6.2), e os resultados obtidos a partir do levantamento de avaliação (Seção 6.3).

### 6.1 Projeto do Questionário

Foi elaborado um questionário para coletar dados usando a ferramenta *Google Forms*, com perguntas específicas organizadas para fornecer suporte às questões de pesquisa definidas neste estudo. A Tabela 6.1 mostra as seções do questionário do SIIC.

**Tabela 6.1:** Seções do questionário do SIIC.

Seção	Tópico	Descrição
0	Consentimento e participação	Obter o consentimento explícito do respondente em relação à participação na pesquisa.
1	Perfil do participante	Coletar informações pessoais como idade, gênero, formação acadêmica, ocupação e a qual grupo (servidor, professor) pertence.
2	Uso atual do sistema de registro de chaves	Avaliar o conhecimento e uso do sistema atual de registro de chaves no IFPB.
3	Usabilidade do SIIC	Apresentar o SIIC através de telas e funcionalidades, e coletar opiniões sobre a usabilidade do sistema.
4	Comentários e sugestões	Coletar sugestões dos usuários para melhorias futuras do SIIC.

O questionário incluiu perguntas subjetivas e objetivas para coletar dados quantitativos e qualitativos relacionados à experiência dos usuários com o sistema de registro de chaves atual do IFPB e suas expectativas em relação ao SIIC.

## 6.2 Definição da amostra e coleta de dados

Os participantes da pesquisa foram definidos como servidores, colaboradores, professores e alunos do IFPB, os principais usuários do sistema de registro de chaves manual. Todas as respostas foram coletadas e organizadas automaticamente pela ferramenta de análise de respostas do *Google Forms*. Antes de responder ao questionário, os participantes concordaram com um "Termo de Consentimento Livre e Esclarecido".

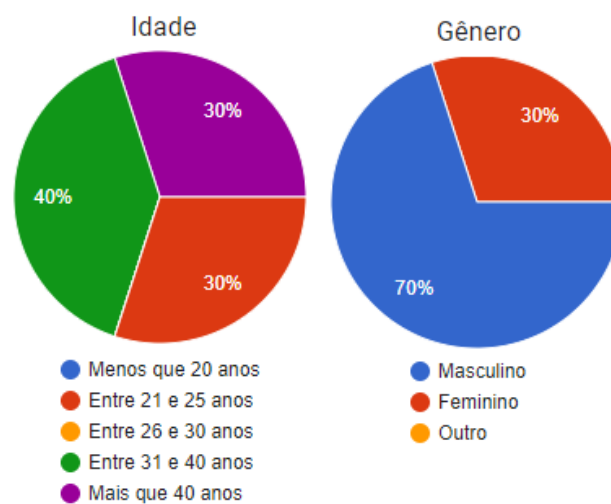
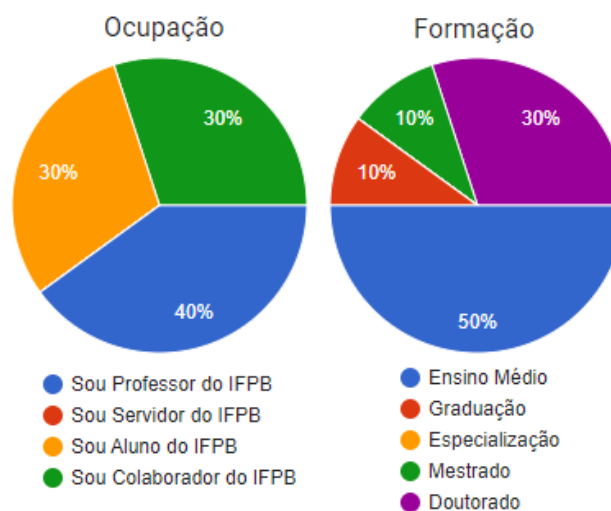
## 6.3 Resultados do Survey de Validação

Esta pesquisa foi realizada durante o mês de Novembro de 2023. Onde foram realizadas algumas entrevistas com usuários futuros onde 10 participantes responderam ao questionário.

### 6.3.1 Caracterização da Amostra

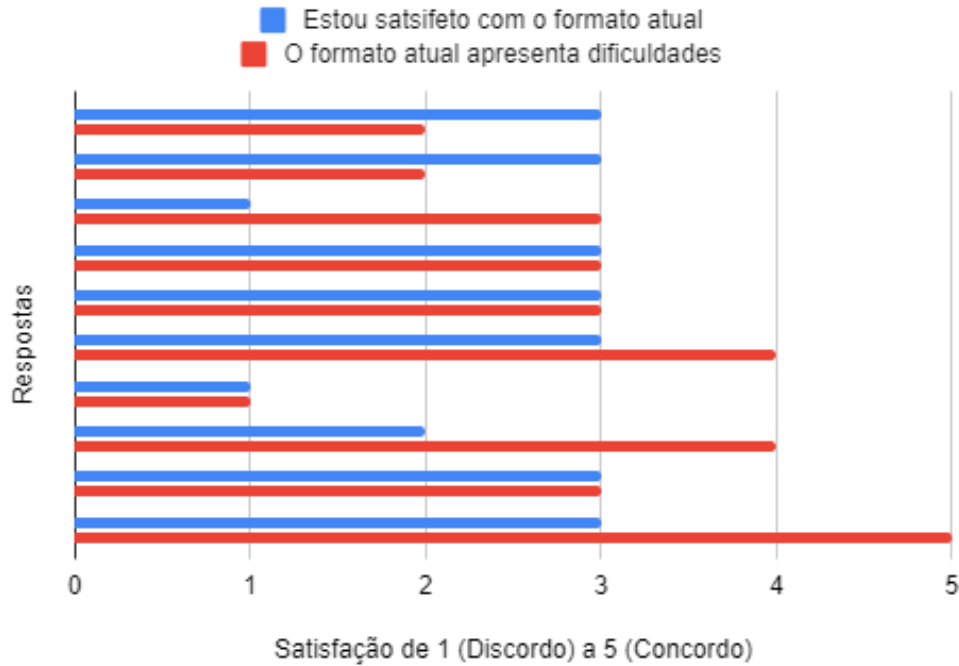
**Caracterização da amostra.** As respostas da Seção 1 do questionário, forneceram dados para constituição do perfil dos participantes. A respeito da idade, 3 declararam ter entre 21 e 25 anos (30%), 3 mais que 40 anos, (30%), 4 entre 31 e 40 anos(40%). Com relação ao gênero, 7 pessoas afirmaram ser do gênero masculino (90%), enquanto que 3 afirmaram ser do gênero feminino (30%).

A maioria destes respondentes possuem formação até o Ensino Médio 5 (50%), enquanto que 3 possuem Doutorado (30%), 1 têm Mestrado (10%) e possui graduação (10%). Ainda nesta seção de caracterização do perfil do entrevistado, foi perguntado sobre qual grupo este fazia parte. 3 são alunos do IFPB (30%), 3 são colaboradores (30%) e 4 são professores (40%) (Figura 6.1 e Figura 6.2).

**Figura 6.1:** Caracterização da amostra.**Figura 6.2:** Caracterização da ocupação.

### 6.3.2 Resultados da Validação

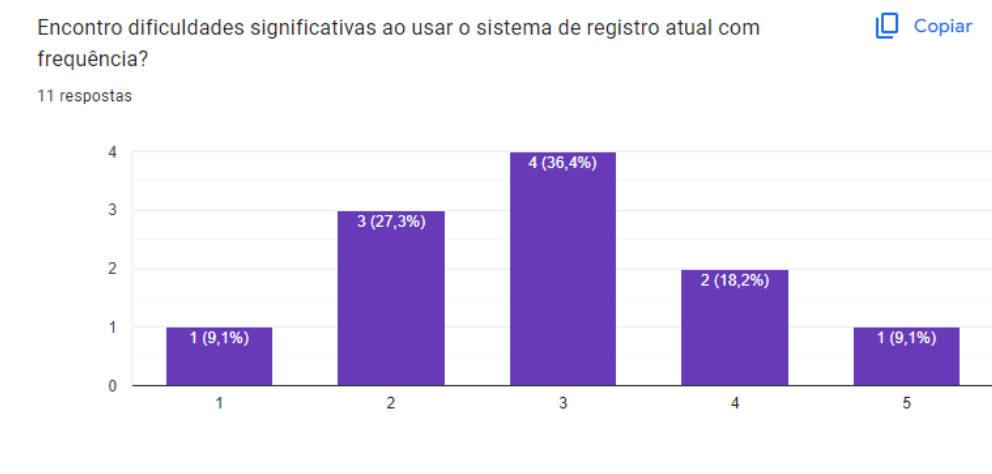
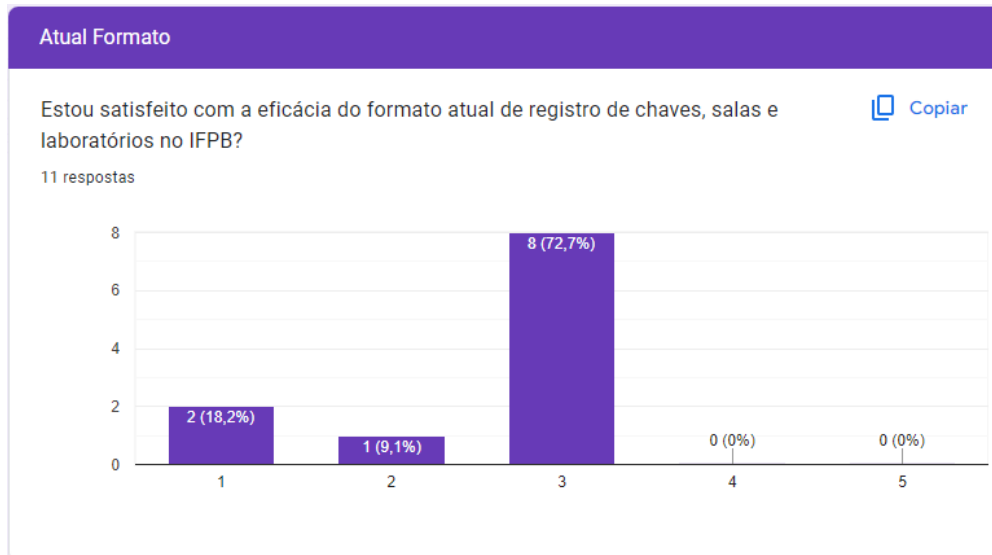
Após a análise das percepções dos usuários em relação ao sistema atual, identificaram-se questões relacionadas à dificuldade de buscar informações, falta de praticidade, perda de tempo e necessidade de deslocamento.



A seguir, está apresentado de forma concisa o SIIC, exibindo telas correspondentes aos perfis dos entrevistados e suas principais funcionalidades. Posteriormente, são solicitadas opiniões sobre a usabilidade do sistema.

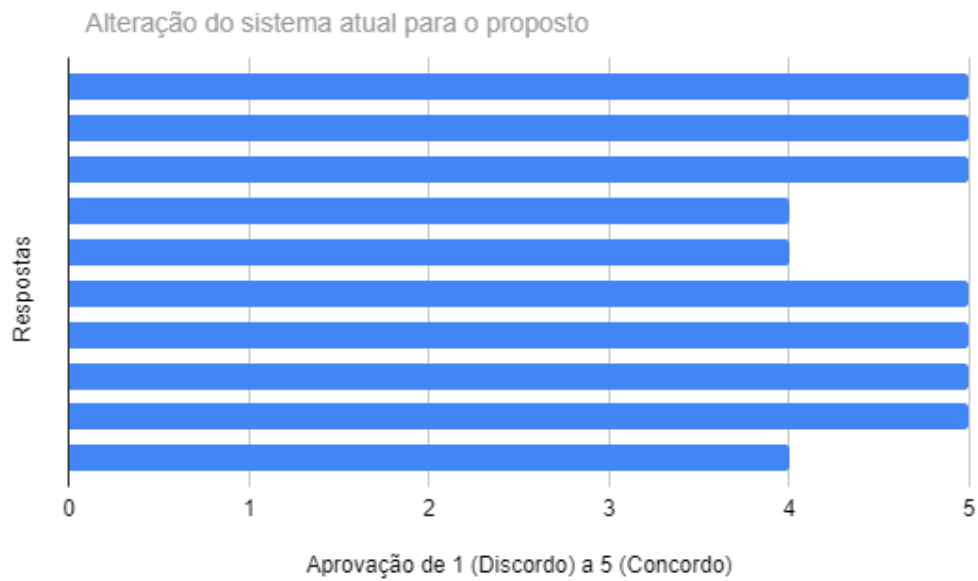
**Usuário Comum:** Durante a avaliação, este perfil explorou diversos *templates*, incluindo a tela de login, a página de visualização de reservas, a página de relato de problemas e a interface para geração de relatórios. Os usuários comuns expressaram a visão de que o SIIC poderia aprimorar e agilizar os processos de reserva, resolução/notificação de problemas, bem como oferecer uma forma mais eficiente de busca por informações.

**Usuário Administrador:** Este perfil examinou diversos *templates*, desde a tela de login até as páginas de visualização e gerenciamento de usuários, reservas, salas, chaves, além das funcionalidades de busca e geração de relatórios. Os administradores destacaram a potencial utilidade da plataforma, especialmente em relação a desafios enfrentados, como autorizações para reservas, não devolução de chaves e busca por informações.



Também foram questionados sobre a satisfação tanto com o registro atual quanto em comparação com a plataforma apresentada, e foi de forte aceitação a possibilidade da implementação do sistema para melhora do desenvolvimento de tarefas relacionados a esse controle no dia a dia.





**Sugestões de Melhorias:** Com base nas respostas coletadas, os entrevistados ofereceram sugestões para aprimorar a usabilidade da plataforma. As principais recomendações incluíram a adição de recursos para agendamento de salas, uma experiência de login mais ágil e a integração com o memorando de um aluno, que serviria como a autorização para a reserva de uma sala. Todas as sugestões serão consideradas para possíveis atualizações futuras do sistema.

# Capítulo 7

## Ameaças à Validade

Este capítulo descreve as principais ameaças identificadas bem como algumas ações adotadas com intuito de mitigar os seus efeitos. No que segue, utilizamos o arcabouço de classificação proposto por Wohlin (Wohlin et al., 2012) para expor as ameaças bem como as ações visando minimizar seus efeitos.

### 7.1 Validade de Construção

*Bugs de Software:* A presença de bugs ou falhas de programação durante a construção do SIIC pode resultar em comportamentos inesperados ou na não execução adequada das funções do sistema. Para minimizar os efeitos dessa ameaça, adotou-se metodologias de desenvolvimento ágil, com testes contínuos durante todas as fases. Adicionalmente, estabeleceu-se um protocolo de revisão de código e realizou-se testes de usuário beta para identificar e corrigir bugs antes da implementação completa.

*Problemas de Integração:* Dificuldades na integração do SIIC com outros sistemas existentes no IFPB podem resultar em conflitos de dados, perda de informações ou interrupções nos fluxos de trabalho. Para minimizar os efeitos dessa ameaça, realizou-se testes extensivos de integração com os sistemas existentes do IFPB antes da implementação completa do SIIC. Ainda, estabeleceu-se uma equipe de suporte dedicada para resolver rapidamente quaisquer problemas de interoperabilidade que possam surgir.

### 7.2 Validade Interna

*Resistência à Mudança:* Os usuários finais podem resistir à adoção do SIIC devido à familiaridade com os métodos tradicionais de registro de chaves. A resistência à mudança pode impactar a eficácia do sistema, levando a uma baixa taxa de adoção e subutilização. Para minimizar o efeito dessa ameaça, realizou-se sessões de conscientização antes da implementação, destacando os benefícios do SIIC para os usuários finais. Desse modo, buscou-se envolvê-los nas fases iniciais do desenvolvimento para incorporar feedback e tornar o sistema mais adaptado às necessidades

específicas.

*Falta de Engajamento do Usuário:* Se os usuários finais não estiverem engajados durante as fases de treinamento e desenvolvimento, a compreensão inadequada do SIIC pode levar a erros de uso e, conseqüentemente, à subutilização do sistema. Para minimizar os efeitos dessa ameaça, criou-se programas de treinamento interativos e personalizados, adaptados às diferentes necessidades e níveis de familiaridade dos usuários. Ainda, estabeleceu-se canais de comunicação eficazes para fornecer suporte contínuo e esclarecer dúvidas.

### 7.3 Validade Externa

*Atualizações Não Planejadas:* Mudanças não previstas nos sistemas operacionais ou nas infraestruturas de rede externas podem causar incompatibilidades e interrupções na funcionalidade do SIIC, prejudicando sua operação eficiente. Para minimizar os efeitos dessa ameaça, manteve-se uma equipe de monitoramento para rastrear atualizações em sistemas externos relevantes. Ainda, testou-se a compatibilidade do SIIC com versões mais recentes de software e estabeleceu-se protocolos para rápida adaptação a mudanças inesperadas.

*Segurança Cibernética:* Ataques cibernéticos externos representam uma ameaça à integridade e segurança do SIIC. A violação de dados ou a manipulação do sistema por parte de terceiros mal-intencionados podem comprometer a confidencialidade das informações. Para minimizar os efeitos dessa ameaça, implementou-se medidas robustas de segurança cibernética, como firewalls, criptografia e autenticação multifatorial. Ainda, conduziu-se auditorias de segurança regulares e promoveu-se a conscientização sobre práticas seguras entre os usuários.

### 7.4 Validade de Conclusão

*Amostras Não-Representativas:* A coleta de dados pode ser comprometida se as amostras não forem representativas da diversidade de situações reais enfrentadas pelo IFPB. Por exemplo, a concentração em determinados departamentos ou áreas pode resultar em generalizações inadequadas sobre a eficácia do SIIC. Para minimizar o efeito dessa ameaça adotou-se uma estratégia de amostragem estratificada, garantindo que as amostras representem adequadamente diferentes departamentos, níveis hierárquicos e tipos de espaços.

*Viés de Seleção:* Pode haver um viés na seleção de participantes para validar o SIIC, com indivíduos mais propensos a aceitar ou adotar novas tecnologias participando de maneira desproporcional. Isso poderia distorcer os resultados, sugerindo uma aceitação mais ampla do sistema do que seria verdadeiro em toda a instituição. Para minimizar o efeito, implementou-se um processo de seleção aleatória para a participação na validação do SIIC, minimizando a influência de fatores pessoais na escolha dos participantes. Desse modo, garantiu-se a participação voluntária e incentivou-se uma variedade de perfis a se envolverem.

# Capítulo 8

## Conclusões Finais e Trabalhos Futuros

A presente pesquisa concentrou-se na concepção, desenvolvimento e implementação da plataforma SIIC, visando aprimorar os processos de registro de chaves. Entre os objetivos específicos delineados, destaca-se a criação de um sistema integrado para o gerenciamento de chaves e seus respectivos espaços, fundamentado em uma análise abrangente dos processos internos e nas necessidades dos usuários envolvidos. Ao longo do desenvolvimento da plataforma, diversas etapas foram executadas, desde a análise de requisitos até a definição de arquitetura, desenvolvimento, validação e implantação. A colaboração estreita com os principais interessados, como professores e funcionários administrativos, foi crucial para assegurar a adequação da plataforma às demandas reais e específicas do IFPB.

Através da realização de um *survey*, foi obtido um *feedback* valioso de 10 participantes que avaliaram as funcionalidades e a usabilidade da plataforma. Os resultados revelaram desafios significativos enfrentados por alunos, professores e responsáveis pelo gerenciamento nos procedimentos atuais de gestão e busca de informações. Emerge da análise que a implementação de um sistema *web*, como proposto pelo SIIC, apresenta um potencial considerável para mitigar efetivamente esses desafios identificados na pesquisa. Entre as contribuições deste trabalho de conclusão de curso podem ser citados:

1. *Análise Abrangente de Processos Internos*: A pesquisa envolveu uma análise abrangente dos processos internos relacionados ao registro de chaves, identificando lacunas e oportunidades de melhoria. Isso não apenas contribuiu para a eficácia imediata da plataforma SIIC, mas também oferece insights valiosos para futuras otimizações de processos no IFPB;
2. *Colaboração Efetiva com Stakeholders*: A colaboração estreita com os principais interessados, incluindo professores e funcionários administrativos, foi uma contribuição crucial. Essa abordagem assegurou que a plataforma SIIC fosse alinhada às reais necessidades e demandas dos usuários, promovendo a aceitação e eficácia do sistema;
3. *Desenvolvimento da Plataforma SIIC*: A pesquisa resultou na concepção, desenvolvimento e implementação da plataforma SIIC, uma solução inovadora para o registro e gerenciamento


de chaves no âmbito do IFPB. Essa contribuição direta atende a uma necessidade específica da instituição, proporcionando eficiência e modernização aos processos.

Quanto ao futuro, é planejado operar plenamente o sistema para aprimorar o registro e gerenciamento de chaves no IFPB. Além disso, há o pensamento de expandir a plataforma, adicionando novos recursos para melhorar continuamente a experiência dos usuários e atender às necessidades em constante mudança da instituição. Essa evolução será guiada pelo *feedback* constante dos usuários e pela busca ativa por inovações tecnológicas. O objetivo é criar uma solução flexível e adaptável que esteja alinhada não apenas com os desafios atuais, mas também com o crescimento futuro do IFPB.

# Referências Bibliográficas

- Awad, M. A. A comparison between agile and traditional software development methodologies , this report is submitted as partial fulfillment of the requirements for the honours. In: *The University of Western Australia*. 2005. 3
- Barbosa, A. M. dos S.; al et. Website empresarial com arquitetura mvc utilizando ddd e angularjs. *RE3C - Revista Eletrônica Científica de Ciência da Computação*, 2017. 10
- Bhavsar, K.; Shah, V.; Gopalan, S. Scrum: An agile process reengineering in software engineering. *International Journal of Innovative Technology and Exploring Engineering*, v. 9, n. 3, p. 840–848, 2020. 8
- Conexorama. *Validação de Softwares, o guia definitivo – Parte I: como começar um processo de validação*. 2018. Disponível em: <https://vistosistemas.com.br/validacao-de-softwares/> Acesso em: Outubro de 2023. 3
- Converse, T.; Park, J. *PHP a B iblia.* : Editora Campus, 2003. 7
- Enterprise, J. *HTML, PHP, dan MySQL untuk Pemula.* : Elex Media Komputindo, 2018. 6
- Filitto, D. *O que é o Laravel*. 2021. <https://dfilitto.com.br/desenvolvimento/desenvolvimento-web/php/o-que-e-o-laravel/> Acesso em: 22/07/2023. 7
- FOKUS Mercury Computer Systems, R. C. F. Specification for deployment and configuration of component-based distributed applications. *Proposal to the OMG MARS RFP: Deployment and Configuration of Component-based Distributed Applications Joint Revised Submission*, p. 160, 2003. 3
- Freitas, P. B. *Projeto e desenvolvimento de um sistema de gerenciamento de chaves (SGC) físicas convencionais para instituições de ensino*. 2023. [Http://www.repositorio.ifba.edu.br/jspui/handle/123456789/445](http://www.repositorio.ifba.edu.br/jspui/handle/123456789/445). Acesso em : novembro. 2023. 11
- HORNICHE, M. C. J. *Gerenciamento de Salas de Aula*. [/www.atenaeditora.com.br/catalogo/post/gerenciamento-de-salas-de-aula-pycontrolroom-desenvolvimento-web-com-framework-django](http://www.atenaeditora.com.br/catalogo/post/gerenciamento-de-salas-de-aula-pycontrolroom-desenvolvimento-web-com-framework-django). Acesso em: novembro. 2023. 11
- Manovich, L. O banco de dados. *Revista Eco-Pós*, 2015. 7
- Mayron, M. *Sistema on-line de gerenciamento de salas de aula criado na UFC tem registro expedido pelo INPI*. 2022. <https://www.ufc.br/noticias/noticias-de-2022/16582-sistema-on-line-de-gerenciamento-de-salas-de-aula-criado-na-ufc-tem-registro-expedido-pelo-inpi>. Acesso em: novembro. 2023. 11

- Melo, D. *O que é PHP? [Guia para Iniciantes]*. 2021. <https://www.terra.com.br/noticias/tecnologia/o-que-e-php-guia-para-iniciantes,9257bda9e03fb4e57594a0da92e3d7268bfj4vzy.html>: :text=OAcesso em: 21/07/2023. 7
- Milani, A. *MySQL - Guia do Programador*. : novatec, 2006. 7
- Miletto, E.; Bertagnolli, S. de C. *Desenvolvimento de Software II: Introdução ao Desenvolvimento Web com HTML, CSS, JavaScript e PHP - Eixo: Informação e Comunicação - Série Tekne*. Bookman Editora, 2014. (Tekne). ISBN 9788582601969. Available on: <<https://books.google.com.br/books?id=lcLFAwAAQBAJ>>. 6
- PRESSMAN, R. *Engenharia de software*. : Makron Books, 1995. 9
- Pressman, R. *Engenharia de software*. McGraw-Hill, 2006. ISBN 9788586804571. Available on: <<https://books.google.com.br/books?id=MNM6AgAACAAJ>>. 8
- Schwaber, K.; Sutherland, J. The scrum guide. *Scrum Alliance*, v. 21, n. 19, p. 1, 2011. 8
- Silva, G. *O que é arquitetura MVC?* 2021. <https://coodesh.com/blog/dicionario/o-que-e-arquitetura-mvc/>: :text=BasicamenteAcesso em: 24/08/2023. 10
- Silva, M. S. *Criando sites com HTML: sites de alta qualidade com HTML e CSS*. : Novatec Editora, 2008. 6
- Silva, M. S. *JavaScript-Guia do Programador: Guia completo das funcionalidades de linguagem JavaScript*. : Novatec Editora, 2010. 6
- Sousa, A. F. da S.; Neto, P. M. da C.; Pessoa, A. N. L. de S. Inclusão digital: Os obstáculos a serem enfrentados na busca pela dignidade na sociedade conectada. *RECIMA21 - Revista Científica Multidisciplinar*, 2023. 5
- Wohlin, C. et al. *Experimentation in software engineering*. : Springer Science & Business Media, 2012. 36

	<b>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA</b>
	Campus Campina Grande
	R. Tranquílino Coelho Lemos, 671, Dinamérica, CEP 58432-300, Campina Grande (PB)
	CNPJ: 10.783.898/0003-37 - Telefone: (83) 2102.6200

## Documento Digitalizado Ostensivo (Público)

### TCC

<b>Assunto:</b>	TCC
<b>Assinado por:</b>	Lucas Dantas
<b>Tipo do Documento:</b>	Anexo
<b>Situação:</b>	Finalizado
<b>Nível de Acesso:</b>	Ostensivo (Público)
<b>Tipo do Conferência:</b>	Cópia Simples

Documento assinado eletronicamente por:

- Lucas Dantas de Araújo, ALUNO (201911250023) DE BACHARELADO EM ENGENHARIA DE COMPUTAÇÃO - CAMPINA GRANDE, em 16/02/2024 09:15:13.

Este documento foi armazenado no SUAP em 16/02/2024. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1082974

Código de Autenticação: f9e9bba659

