

**INSTITUTO
FEDERAL**

Goiás

Instituto Federal de Goiás

Campus Formosa

Análise e Desenvolvimento de Sistemas

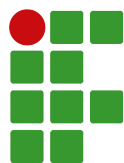
**IFGACCESS: SISTEMA WEB DE CONTROLE DE ACESSO UTILIZANDO RFID E O
MICROCONTROLADOR ESP8266**

DANIELE DOS SANTOS ARAÚJO

Trabalho de Conclusão de Curso

FORMOSA

2022



**INSTITUTO
FEDERAL**

Goiás

Instituto Federal de Goiás

Campus Formosa

Análise e Desenvolvimento de Sistemas

**IFGACCESS: SISTEMA WEB DE CONTROLE DE ACESSO
UTILIZANDO RFID E O MICROCONTROLADOR ESP8266**

Daniele dos Santos Araújo

Trabalho de Conclusão de Curso apresentado ao Departamento de Áreas Acadêmicas do Instituto Federal de Goiás campus Formosa, como requisito parcial para obtenção do grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Me. Mário Teixeira Lemes

FORMOSA

2022

Daniele dos Santos Araújo

IFGACCESS: SISTEMA WEB DE CONTROLE DE ACESSO UTILIZANDO RFID
E O MICROCONTROLADOR ESP8266/ Daniele dos Santos Araújo. – FORMOSA,
2022-

54 p.; 30 cm.

Orientador Me. Mário Teixeira Lemes

Trabalho de Conclusão de Curso – Instituto Federal de Goiás, 2022.

1. Plano Semestral de Trabalho Docente 2. IFG 3. Desenvolvimento de software 4.
Aplicação *Web* I. Orientador: Me. Mário Teixeira Lemes. II. Instituto Federal de Goiás.
IV. Título: IFGACCESS: SISTEMA WEB DE CONTROLE DE ACESSO UTILIZANDO
RFID E O MICROCONTROLADOR ESP8266

CDU 02:141:005.7



INSTITUTO FEDERAL
Goiás

MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE GOIÁS
CÂMPUS FORMOSA

ATA DA SESSÃO PÚBLICA DE APRESENTAÇÃO E DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO

Ao 15º dia do mês de dezembro do ano de dois mil e vinte e dois, às 15:00 horas, no Instituto Federal de Goiás, Câmpus Formosa, situado à Rua 64, Setor Expansão Parque Lago da cidade de Formosa, Estado de Goiás, foi realizada a sessão pública de apresentação e defesa do Trabalho de Conclusão de Curso da Graduanda **Daniele dos Santos Araújo** (matrícula 20161070130229) do curso de Tecnologia em Análise e Desenvolvimento de Sistemas, no segundo semestre letivo do ano de dois mil e vinte e dois. A banca foi composta pelos seguintes membros: Profº. Me. Mário Teixeira Lemes (IFG/Formosa), Profª Dra. Uyara Ferreira Silva (IFG/Formosa) e Profº Esp. Danilo Souza Almeida (IFTM/Paracatu). O Trabalho de Conclusão de Curso tem como título "**IFGAccess: Sistema Web de Controle de Acesso Utilizando RFID e o Microcontrolador ESP8266**", da área de Informática, sob orientação do Profº. Me. Mário Teixeira Lemes. Após apresentação do Trabalho de Conclusão de Curso, tendo sido a autora arguida pela Banca Examinadora, a nota obtida foi **9,6** pontos, sendo, portanto, aprovada com correções.

Encerra-se a presente sessão às 16 horas e 20 minutos. Eu, Profª. Dra. Uyara Ferreira Silva, dato e assino a presente ata que segue assinada por todos os membros da Banca e pela graduanda.

Profª Dra. Uyara Ferreira Silva

Profº Esp. Danilo Souza Almeida

Profº. Me. Mário Teixeira Lemes

Daniele dos Santos Araújo
(Graduando)

Documento assinado eletronicamente por:

- **Mário Teixeira Lemes**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 09/01/2023 10:30:01.
- **Danilo Souza Almeida**, Danilo Souza Almeida - Professor Efetivo EBTT - Iftm (10695891000282), em 15/12/2022 18:45:55.
- **Uyara Ferreira Silva**, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 15/12/2022 16:22:15.

Este documento foi emitido pelo SUAP em 15/12/2022. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifg.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 357051
Código de Autenticação: 4307513a87



Instituto Federal de Educação, Ciência e Tecnologia de Goiás
Rua 64 esquina com Rua 11, S/Nº, Expansão Parque Lago, Formosa / GO, CEP 73813-816
(61) 3642-9491 (ramal: 9491), (61) 3642-9493 (ramal: 9493)

**TERMO DE AUTORIZAÇÃO PARA DISPONIBILIZAÇÃO
NO REPOSITÓRIO DIGITAL DO IFG - ReDi IFG**

Com base no disposto na Lei Federal nº 9.610/98, AUTORIZO o Instituto Federal de Educação, Ciência e Tecnologia de Goiás, a disponibilizar gratuitamente o documento no Repositório Digital (ReDi IFG), sem ressarcimento de direitos autorais, conforme permissão assinada abaixo, em formato digital para fins de leitura, download e impressão, a título de divulgação da produção técnico-científica no IFG.

Identificação da Produção Técnico-Científica

- | | |
|--|---|
| <input type="checkbox"/> Tese | <input type="checkbox"/> Artigo Científico |
| <input type="checkbox"/> Dissertação | <input type="checkbox"/> Capítulo de Livro |
| <input type="checkbox"/> Monografia – Especialização | <input type="checkbox"/> Livro |
| <input type="checkbox"/> TCC - Graduação | <input type="checkbox"/> Trabalho Apresentado em Evento |
| <input type="checkbox"/> Produto Técnico e Educacional - Tipo: _____ | |

Nome Completo do Autor:

Matrícula:

Título do Trabalho:

Restrições de Acesso ao Documento

Documento confidencial: Não Sim, justifique: _____

Informe a data que poderá ser disponibilizado no ReDi/IFG: ___/___/___

O documento está sujeito a registro de patente? Sim Não

O documento pode vir a ser publicado como livro? Sim Não

DECLARAÇÃO DE DISTRIBUIÇÃO NÃO-EXCLUSIVA

O/A referido/a autor/a declara que:

- i. o documento é seu trabalho original, detém os direitos autorais da produção técnico-científica e não infringe os direitos de qualquer outra pessoa ou entidade;
- ii. obteve autorização de quaisquer materiais incluídos no documento do qual não detém os direitos de autor/a, para conceder ao Instituto Federal de Educação, Ciência e Tecnologia de Goiás os direitos requeridos e que este material cujos direitos autorais são de terceiros, estão claramente identificados e reconhecidos no texto ou conteúdo do documento entregue;
- iii. cumpriu quaisquer obrigações exigidas por contrato ou acordo, caso o documento entregue seja baseado em trabalho financiado ou apoiado por outra instituição que não o Instituto Federal de Educação, Ciência e Tecnologia de Goiás.

_____, ____/____/____.
Local Data

Assinatura do Autor e/ou Detentor dos Direitos Autorais

Dedico este trabalho ao projeto Caliandras Digitais, em especial a Uyara e a Laiane, por dedicarem tempo e esforço a incentivar a permanência de meninas na computação.

Não importa como você leve sua vida, sua inteligência o defenderá melhor que uma espada. Trate de mantê-la afiada.

—PATRICK ROTHFUSS

Resumo

Os avanços na microeletrônica são observados através do desenvolvimento de dispositivos computacionais miniaturizados e de baixo custo. Internet das Coisas (IoT) é o paradigma que permite que objetos inteligentes sejam conectados à Internet, potencializando o surgimento de novas aplicações. A popularização de IoT, aliado aos avanços tecnológicos nos processos de processamento, comunicação e identificação, permitem que sejam construídos sistemas automatizados que proporcionam comodidade, credibilidade e segurança aos usuários. Neste contexto, sistemas automatizados de controle de acesso podem fornecer confiabilidade e segurança ao serem utilizados como métodos de controle e registro de acesso de pessoas a ambientes de alto valor agregado. O controle de chaves para liberação dos laboratórios do IFG-*Campus* Formosa é realizado de forma manual, o que pode gerar problemas relacionados a segurança, tais como esquecimento de registro de entrada/saída, bem como a possibilidade de ações mal intencionadas por parte dos usuários. O objetivo deste trabalho é desenvolver um sistema *Web* de controle de acesso, denominado IFGAccess, através do uso da tecnologia de radio-frequência e dispositivos computacionais embarcados, e apresentar os resultados obtidos através do experimento simulado realizado.

Palavras-chave: Internet das Coisas, Sistema de Controle, Autenticação, Identificação por Radio-Frequência.

Abstract

Advances in microelectronics are observed through the development of miniaturized, low-cost computing devices. Internet of Things (IoT) is the paradigm that allows smart objects to be connected to the Internet, boosting the emergence of new applications. The popularization of IoT, coupled with technological advances in processing, communication and identification, allows automated systems to be built that provide convenience, credibility and security to users. In fact, automated access control systems can provide reliability and security by being used as methods of controlling and recording people's access to high-value-added environments. The control of keys for release of the laboratories of IFG-Campus Formosa is carried out manually, which can generate problems related to security, such as forgetting entry/exit record, as well as the possibility of malicious actions by the users. The main of this work is to develop IFGAccess, a Web access control system, through the use of radio frequency technology and embedded computing devices, and present the results achieved through the simulated experiment realized.

Keywords: Internet of Things, Control System, Authentication, Radio-Frequency Identification.

Lista de Figuras

2.1	NodeMCU v2.	19
2.2	Módulo relé atuador.	21
2.3	Tag RFID Programável 13,56Mhz	21
2.4	Organização Global de um site <i>Web</i> tradicional.	22
3.1	Visão Geral do Projeto	29
4.1	Gráfico do questionário sobre segurança do sistema atual	31
4.2	Gráfico do questionário sobre perda ou roubo	31
4.3	Visão Geral da Arquitetura do IFGAccess	32
4.4	Diagrama de Atividades do IFGAccess	34
4.5	Diagrama de Casos de Uso - IFGAccess	34
4.6	Diagrama de Entidade-Relacionamento do IFGAccess	35
4.7	Docker do projeto IFGAccess	35
4.8	Página de consulta de <i>tags</i> não cadastradas	36
4.9	Envio de requisição usando <i>Postman</i>	36
4.10	Página de consulta de <i>tags</i> cadastradas	36
4.11	Página de cadastro de usuário	36
4.12	Página de lista de usuário cadastrados	37
4.13	Página de editar dados de usuário	37
4.14	Página de lista de acessos	38
4.15	Desenho do protótipo	40
4.16	<i>Monitor Serial</i> - Conexão com a Internet	41
4.17	<i>Display</i> - Aproxime a <i>tag</i>	43
4.18	<i>Display</i> - Acesso negado	44
4.19	<i>Display</i> - Acesso liberado	44
4.20	<i>Display</i> - Aguarde <i>checkout</i>	45
4.21	<i>Display</i> - <i>Checkout</i>	45

Lista de Códigos

4.1	Código para validação da requisição <i>POST</i>	38
4.2	Código para verificação de <i>check-out</i> pendente	38
4.3	Código para verificação de <i>check-out</i> pendente	39
4.4	Código para conexão do NodeMCU ao enlace de comunicação Wi-Fi	40
4.5	<i>Loop</i> no NodeMCU para captura de <i>tags</i> RFID	41
4.6	Validação de <i>check-in</i> pendente de <i>check-out</i>	42

Lista de Tabelas

3.1	Relação de Questões Realizadas e Categoria	28
4.1	Relação de conexões dos dispositivos	40

Lista de Acrônimos

IFG	Instituto Federal de Educação, Ciência e Tecnologia de Goiás	26
UNIVASF	Universidade Federal do Vale do São Francisco	16
DAA	Departamento de Áreas Acadêmicas	15
URL	<i>Uniform Resource Locator</i>	22
HTML	<i>Hypertext Markup Language</i>	22
SQL	<i>Structured Query Language</i>	27
NFC	<i>Near Field Communication</i>	16
HTTP	<i>Hypertext Transfer Protocol</i>	22
IoT	<i>Internet of Things</i>	15
RFID	<i>Radio Frequency Identification</i>	16
Wi-Fi	<i>Wireless Fidelity</i>	19
WWW	<i>World Wide Web</i>	18
SGBD	Sistema Gerenciador de Banco de Dados	24
BD	Banco de Dados	28
UTFPR	Universidade Tecnológica Federal do Paraná	16
ESTG	Escola Superior de Tecnologia e Gestão	16
DER	Diagrama Entidade-Relacionamento	28
PHP	PHP: <i>Hypertext Preprocessor</i>	23
MCU	<i>Microcontroller Unit</i>	19
UML	<i>Unified Modeling Language</i>	17
LCD	<i>Liquid Crystal Display</i>	33
SQL	<i>Structured Query Language</i>	24
TCC	Trabalho de Conclusão de Curso	26
ID	Identificador	34
I2C	<i>Inter-Integrated Circuit</i>	39
PaaS	<i>Platform as a Service</i>	24

Sumário

1	Introdução	15
1.1	Objetivo Geral	17
1.2	Objetivos Específicos	17
1.3	Descrição dos Capítulos	17
2	Referencial Teórico	18
2.1	Internet das Coisas	18
2.2	Sistemas embarcados	18
2.2.1	Microcontroladores	19
2.2.2	NodeMCU	19
2.2.3	C++	20
2.2.4	Arduíno IDE	20
2.3	Sensores e Atuadores	20
2.3.1	RFID	20
2.4	Sistemas <i>Web</i>	22
2.4.1	PHP	23
2.5	Banco de Dados	23
2.5.1	Banco de dados relacional	23
2.5.2	SGBD	24
2.5.3	SQL	24
2.5.4	MySQL	24
2.6	Contêiner	24
2.6.1	Docker	25
2.7	UML	25
2.7.1	Diagrama de Casos de Uso	25
2.7.2	Diagrama de Atividades	25
3	Materiais e Método	26
3.1	Metodologia da Pesquisa	26
3.2	Construção dos artefatos da UML	27
3.3	Ferramentas utilizadas	27
3.4	Construção e Integrações	29
4	Resultados	30
4.1	Questionário	30
4.2	Visão Geral do Sistema	31

4.2.1	Diagrama de Atividades	33
4.2.2	Diagrama de Casos de Uso	33
4.2.3	Diagrama de Entidade-Relacionamento	33
4.3	Sistema <i>Web</i>	35
4.4	Montagem do protótipo	39
4.5	Comunicação entre as partes do sistema	39
5	Conclusão	46
5.1	Sugestão de Trabalhos Futuros	46
	Referências	48
	Apêndice	51
	A Questionário	52

1

Introdução

É notório o aumento da preocupação com a confiabilidade dos registros e processos de identificação. A tecnologia, por sua vez, vem com o intuito de proporcionar comodidade, credibilidade e facilidade com propostas de acesso remoto a dispositivos de automação, que proporcionam ao usuário o registro dos acessos, facilitando o processo de identificação (MEDEIROS et al., 2020). Dentro desse contexto da busca por soluções que proporcionem comodidade surgem dispositivos inteligentes no universo de Internet das Coisas, do inglês *Internet of Things* (IoT), tema recorrente em discussões sobre conectividade de dispositivos e aplicações.

O crescimento de IoT pode ser observado através do aumento da quantidade de objetos inteligentes conectados à Internet. Em 2019, as conexões IoT cresceram a uma taxa média anual de 14%, com tendência de se chegar a 1,3 bilhão em 2025, de acordo com ASSOCIATION et al. (2019). Ao conectar objetos com diferentes recursos a uma rede, potencializa-se o surgimento de novas aplicações. Em IoT, os objetos podem prover comunicação entre usuários e dispositivos. Com isto emerge uma nova gama de aplicações tais como coleta de dados de pacientes e monitoramento de idosos, sensoriamento de ambientes inóspitos e de difícil acesso, pedágios inteligentes (através de chips com identificação por radiofrequência, veículos conseguem passar e pagar, sem precisar parar), relógios inteligentes, entre outras (SUNDMAEKER et al., 2010).

Sistemas automatizados podem fornecer confiabilidade e segurança ao serem utilizados em tarefas cotidianas, como por exemplo o controle de acesso a ambientes. O controle de acesso aos laboratórios de informática do IFG é realizado de forma manual. Para obter acesso aos laboratórios de informática, basta preencher manualmente a folha de controle de chaves do Departamento de Áreas Acadêmicas (DAA) e retirar a chave. Essa prática possibilita que pessoas não autorizadas possam acessar livremente os ambientes de laboratório, comprometendo a segurança física e de seus ocupantes. Levando em consideração o alto valor agregado do material do laboratório, a citar: computadores, roteadores, *switches* e outros equipamentos, é fundamental o desenvolvimento de soluções que garantam segurança no acesso físico desses locais.

MAIA et al. (2019) desenvolveram um protótipo para realizar o controle automático de acesso de pessoas a Universidade Federal Rural do Semi-Árido (UFERSA)-*Campus* Mossoró.

Este protótipo utiliza RFID integrada a plataforma *open-source* de prototipagem eletrônica Arduino®. A validação de permissão no protótipo ocorre por meio do código fonte, onde permanece o registro dos identificadores das *tags* de maneira estática.

Conforme pesquisado em BRITO et al. (2019), o uso de RFID é benéfico para controle patrimonial. De acordo com o levantamento feito pelos autores, os principais pontos são redução de custo e tempo no uso de sistemas de identificação de bens públicos, através do uso das etiquetas RFID na Universidade Federal do Vale do São Francisco (UNIVASF). Os autores argumentam que o sistema automatizado contrapõem-se ao sistema de controle manual, naturalmente suscetível a falhas. Outro ponto destacado pelos autores é a não necessidade de mão de obra devido ao uso de uma solução automatizada.

Em seu estudo, LISBOA (2021) desenvolveu o uma proposta de controle de patrimônio utilizando tecnologia RFID, com o objetivo de automatizar a auditoria de movimentação de bens pertencentes a Universidade Tecnológica Federal do Paraná (UTFPR), Campus Ponta Grossa. O método proposto possui a vantagem de eliminar erros humanos em processos burocráticos. No cenário desenvolvido pelos autores, a movimentação do patrimônio é detectada através do uso de uma antena, conectada a um computador, que armazena o local e data da passagem do patrimônio. Para isso também foi desenvolvida uma aplicação *Web*, que tem a capacidade de tornar mais acessível o processo de verificação e controle do patrimônio da universidade.

Os autores em SILVEIRA et al. (2021) desenvolveram uma fechadura eletrônica para controle de acesso ao laboratório da Universidade Federal de Santa Catarina, utilizando um sistema que possibilita a utilização das carteirinhas de identificação como chave de acesso. Para armazenamento organizado de cada usuário cadastrado ou o registro de entradas e saídas, os autores utilizaram um banco de dados. Um *site* foi desenvolvido no intuito de fornecer rápida visualização do histórico de acessos realizados ao ambiente controlado.

PEIXOTO (2022) desenvolveu um protótipo de sistema portátil e de baixo custo para registro eletrônico da presença dos alunos na Escola Superior de Tecnologia e Gestão (ESTG), em Portugal. O sistema faz uso de *Near Field Communication* (NFC), tecnologia disponível em *smartphones*, *Radio Frequency Identification* (RFID), utilizado no tradicional cartão de estudante e da leitura de impressões digitais como métodos de identificação dos alunos.

Sabe-se que RFID é uma tecnologia que captura dados por meio de rádio frequência, tornando possível a identificação de objetos com dispositivos eletrônicos (*tags*). Dispositivos eletrônicos de baixo custo, tais como o NodeMCU, são capazes de processar e tratar esses sinais, bem como realizar o envio destas informações pela Internet devido sua comunicação Wi-Fi integrada. Desta forma, é possível construir um sistema computacional baseado na *Web* para efetivo controle de acesso aos laboratórios do IFG utilizando o NodeMCU e a tecnologia de comunicação RFID. Os acessos aos laboratórios podem ser identificados individualmente com informações fundamentais no contexto do IFG, tais como: matrícula e tempo de permanência. O registro automatizado do acesso aos laboratórios de informática do IFG - Campus Formosa possibilitará maior eficiência e confiabilidade quando comparado ao registro manual atualmente

utilizado como método de controle de acesso.

1.1 Objetivo Geral

Projetar e desenvolver um sistema *Web* de controle de acesso aos laboratórios de informática do IFG - Campus Formosa ou ambientes com alto valor agregado, utilizando tecnologia de RFID e dispositivos computacionais embarcados de baixo custo.

1.2 Objetivos Específicos

1. Modelar o sistema de informação com o auxílio da *Unified Modeling Language* (UML);
2. Projetar e construir um banco de dados relacional para cadastro de usuários e registro dos acessos;
3. Desenvolver uma página *Web* a nível de *front-end* para apresentação da listagem de acessos ao laboratório e para cadastro dos dados do usuário vinculado a *tag*;
4. Desenvolver uma página *Web* a nível de *back-end* com as funcionalidades básicas, tais como leitura, registro e verificação de permissão de acesso das *tags*.
5. Realizar a montagem e testagem física dos dispositivos, a fim de validar códigos criados para o NodeMCU, através da troca de mensagem pelo protocolo HTTP entre o protótipo e a página *Web*;

1.3 Descrição dos Capítulos

No Capítulo 2 são apresentados importantes conceitos teóricos relacionados ao contexto deste trabalho. No Capítulo 3 é explicado a metodologia utilizada para atingir os objetivos geral e específicos mostrados nesta Introdução. Por fim, no Capítulo 4 é exposto os artefatos de *software* criados no processo de análise e documentação do sistema IFGAccess, tais como fluxos de utilização, diagramas de atividades e casos de uso. Também é apresentado o processo de desenvolvimento e a utilização prática das tecnologias escolhidas, tais como os resultados referentes aos objetivos definidos. No Capítulo 5 são listadas as conclusões deste trabalho, bem como tópicos de pesquisa que permitem continuidade para possíveis trabalhos futuros.

2

Referencial Teórico

Nesse Capítulo é abordada toda a fundamentação teórica para o entendimento e desenvolvimento do trabalho desenvolvido. Aqui são apresentados brevemente conceitos fundamentais sobre a Internet das Coisas, sistemas *World Wide Web* (WWW), microcontroladores, tecnologia de radio frequência nos processos de identificação, bancos de dados e a estrutura em *software* do dispositivo desenvolvido ao longo da pesquisa aqui apresentada.

2.1 Internet das Coisas

IoT, considerada um novo paradigma tecnológico, pode ser considerada uma extensão da internet atual (MANCINI, 2017). Esta extensão possibilita que objetos que possuem capacidade computacional e de comunicação se conectem à Internet. Os autores argumentam que por meio dessa conexão é possível controlar remotamente objetos (coisas), tornando-os provedores de serviços. Os objetos passam a ser denominados de objetos inteligentes. As características de computação e comunicação dos objetos inteligentes permitem o registro contínuo de dados sobre o estado desses objetos durante seu funcionamento, permitindo que sejam realizadas respostas aos dados recebidos.

2.2 Sistemas embarcados

Segundo JIMÉNEZ; PALOMERA; COUVERTIER (2013), um sistema embarcado, do inglês *embedded system*, pode ser amplamente definido como um dispositivo que contém componentes de *hardware* e *software* fortemente acoplados para executar uma única função.

De acordo com os autores em DENARDIN; BARRIQUELLO (2019), sistemas em tempo real normalmente são sistemas embarcados e que isto significa que o sistema computacional é completamente encapsulado e dedicado ao dispositivo ou sistema que controla. Essa é a fundamental diferença entre sistemas embarcados e sistemas de propósito geral.

Por sua vez, POHL et al. (2012) definem que sistemas embarcados são microcontroladores conectados a sistemas completos por intermédio de sensores, atuadores, controles de operação e dispositivos de comunicação. Esses dispositivos interagem de diversas formas com o

ambiente e oferecem uma variedade de funções, por meio de *software* abrangentes. A maioria dos sistemas embarcados interage diretamente com os processos ou com o ambiente, tomando decisões em tempo real, com base em suas entradas.

2.2.1 Microcontroladores

BATES (2008) afirma que um computador ou controlador digital tem três elementos principais: (i) dispositivos de entrada e saída; (ii) processador e (iii) memória. Porém, ao contrário de um sistema microprocessado convencional (como um PC), que possui *chips* separados em uma placa de circuito impresso, o microcontrolador contém todos esses elementos em um único *chip*. Dessa forma, o microcontrolador, também conhecido como *Microcontroller Unit* (MCU), é essencialmente um computador em um *chip*. Segundo FIRAT; UĞURLU (2018), microcontroladores são muito usados em aplicações embarcadas.

2.2.2 NodeMCU

O NodeMCU é um kit de desenvolvimento com *firmware* de código aberto com o objetivo de facilitar a prototipagem de produtos com IoT através tanto da programação em linguagem Lua, quanto a linguagem nativa do também kit de desenvolvimento Arduíno. O kit de desenvolvimento apresentado na figura 2.1 é baseado na placa ESP8266 (NODEMCU, 2022). Já a placa ESP8266, dentre suas muitas características e recursos, permite a conectividade *Wireless Fidelity* (Wi-Fi) (Espressif Systems, 2022). Esse módulo pode ser facilmente programado utilizando Lua ou C++ junto a biblioteca do Arduíno, permitindo assim a conexão utilizando Wi-Fi 2.4 GhZ.

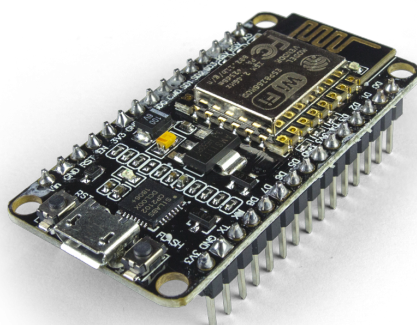


Figura 2.1: NodeMCU v2.

Fonte: (ROBOCORE, 2023a)

2.2.3 C++

A linguagem C++ é uma extensão da linguagem C, que adicionou ao C a capacidade de criar classes, usar tipos abstratos de dados e criar herança como meio de criar novos tipos de dados. Também foram adicionadas as capacidades de sobrecarga de operadores e funções, o que permite o uso de sintaxe mais intuitiva. C++ foi projetado para ser uma linguagem multi-paradigma, podendo ser usada como linguagem imperativa, orientada a objetos e genérica. Mas o principal objetivo de C++ sempre foi prover uma forma de codificação mais clara e abstrata dos sistemas (RICARTE, 2001).

2.2.4 Arduíno IDE

O Arduino IDE, do inglês *Integrated Development Environment*, é um *software* oficial introduzido por Arduino.cc, que é usado principalmente para editar, compilar e carregar o código em dispositivos Arduino (FEZARI; AL DAHOUD, 2018). A placa NodeMCU é compatível com essa plataforma, que contém ferramentas essenciais para a conexão do Arduíno para que esta seja reprogramada utilizando ferramentas de edição disponíveis na interface gráfica da IDE.

2.3 Sensores e Atuadores

Em seu dicionário de informática e Internet, SAWAYA (2002) afirma que um sensor é um dispositivo que converte as variáveis de um processo físico em dados que sejam interpretados e possam ser trabalhados pelo computador. Os dados coletados pelos sensores possibilitam que o microcontrolador tome decisões e realize ações que são realizadas por meio de atuadores, que são dispositivos eletrônicos que transformam pulsos elétricos em atividades mecânicas (BANZI; SHILOH, 2015).

Um sensor comumente utilizado em aplicações IoT é o leitor de RFID Mfrc522 da *NXP Semiconductors*, que será utilizado no projeto apresentado. De acordo com suas especificações técnicas o Mfrc522 realiza a conversão da energia elétrica recebida em seu polo de alimentação para um sinal eletromagnético que realiza a leitura de uma *tag* aproximada (SEMICONDUCTORS, 2018).

Neste projeto não serão usados atuadores, mas um exemplo de um atuador também comumente utilizado em aplicações IoT é o módulo relé atuador apresentado na figura 2.2, onde através de um pulso elétrico ele realiza a troca física no chaveamento de um interruptor (TECHNOLOGY, 2022).

2.3.1 RFID

O RFID surgiu com o proposito de identificar um objeto por meio de envio de ondas de radiofrequência. Teve seu inicio em 1940, com uso de *transponders*. Tais dispositivos foram



Figura 2.2: Módulo relé atuador.

Fonte: (ROBOCORE, 2023b)

utilizados nos aviões na Segunda Guerra Mundial, auxiliando a identificação de outros aviões que estivessem ao redor, evitando possíveis colisões e facilitando manobras e ataques aéreos.

O RFID é composto por duas partes: Um receptor RFID e uma *tag* RFID sendo um exemplo a *tag* apresentada na figura 2.3. Tanto o receptor quanto a *tag* podem ser ativos (contendo uma fonte própria de alimentação de energia), quanto passivos (alimentados por um leitor externo) (WEINSTEIN, 2005).



Figura 2.3: Tag RFID Programável 13,56Mhz

Fonte: (ROBOCORE, 2023c)

Ainda em seu trabalho, WEINSTEIN (2005) apresenta um comparativo entre os dois modelos de *tags*: O modelo do tipo ativo pode possuir um circuito mais complexo (acarretando em um valor de custo mais elevado), é comumente encontrada operando dentro de frequências mais altas (entre 455MHz, 2.45GHz e 5.8GHz) e podem se comunicar com um leitor em

distancias entre 20 a 100 metros. Já o modelo passivo geralmente possui menor custo, trabalha em frequências mais baixas (128KHz, 13.6MHz, 915MHz e 2.45 GHz) e pode se comunicar com seu leitor até uma distância de aproximadamente 9 metros.

2.4 Sistemas Web

A *Web*, que vem de *WWW*, é um sistema distribuído que disponibiliza o acesso a serviços e/ou documentos por meio de uma rede. Um documento, o mesmo que uma página *Web*, é formada por objetos e um objeto se refere a um único arquivo. A maior parte da *Web* é escrita em linguagem *Hypertext Markup Language (HTML)* que possibilita, dentro da sua estrutura, a referência de outros objetos, tais como: imagens e vídeos. (TANENBAUM; STEEN, 2008; KUROSE; ROSS, 2013).

A arquitetura comumente usada em sistemas *Web* é a cliente-servidor, exemplificada na Figura 2.4. Como pode ser observado na figura, o servidor disponibiliza acesso aos documentos que podem ser solicitados por clientes, enquanto o cliente oferece uma interface amigável que torna mais simples a utilização do serviço para o usuário final. O usuário acessa um *browser*, comumente conhecido como navegador *Web*, e pode fazer as requisições por meio de uma referência chamada de *Uniform Resource Locator (URL)*. A URL corresponde ao identificador uniforme de recurso, identificando o documento ou objeto unicamente na Internet. Já o uso do *browser* possibilita as interações do usuário com o serviço fornecido pelo servidor. (TANENBAUM; STEEN, 2008).

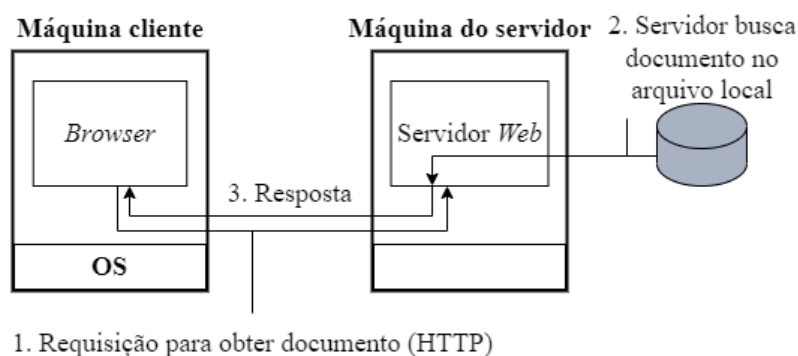


Figura 2.4: Organização Global de um site *Web* tradicional.

Fonte: (TANENBAUM; STEEN, 2008)

O protocolo de comunicação da arquitetura cliente-servidor é o *Hypertext Transfer Protocol (HTTP)*, que é responsável por definir como será feita a troca de mensagem entre o cliente e o servidor (KUROSE; ROSS, 2013). O HTTP fornece uma quantidade limitada de métodos, dos quais são os mais conhecidos o *GET* e o *POST*. Os métodos possibilitam que o cliente acesse ao documento desejado por meio da URL com o uso do método *GET* ou envie informações ao servidor com o uso do *POST* (COULOURIS et al., 2013).

2.4.1 PHP

O PHP: *Hypertext Preprocessor* (PHP) foi apresentado em sua primeira forma como PHP/FI em 1994 por Rasmus Lerdof, como um *script* na linguagem de programação C para acompanhamento do seu currículo *online*. Após recompilar o robusto conjunto de *scripts* e renomear para "*PHP Tools*", em 1995 Rasmus abriu o código fonte para o livre uso público (GROUP, 2023).

O PHP que nasceu com o intuito de possibilitar o pré-processamento de páginas HTML dinamicamente. Desta forma, consegue-se alterar o conteúdo de uma página antes de enviá-la para o navegador ou mesmo criar uma nova página a cada requisição, captura de entrada de dados e outras formas de interação que geram requisições ao servidor (BENTO, 2021).

2.5 Banco de Dados

Banco de dados é um sistema computadorizado de manutenção de registros. É um repositório ou recipiente para uma coleção de arquivos de dados computadorizados. Sua função é armazenar informações e permitir que os usuários busquem e atualizem essas informações. (DATE, 2004).

Também podem ser compreendidos como um conjunto de variáveis organizadas de modo a explicitar alguma possível correlação entre seus elementos que de outra forma poderiam ser inexistentes se inspecionadas isoladamente à parte do grande conjunto ao qual pertencem. É interessante notar que a própria terminologia da expressão implica que sem a devida análise exploratória não é possível de fato visualizar o Banco de Dados como um conjunto de informações significativas para um fim determinado pelo cientista de dados. Desse modo, existem diversas ferramentas capazes de realizar estudos descritivos de conjuntos de informações organizados em Bancos de Dados possibilitando assim uma maior abstração das informações coletadas durante o desenvolvimento de experimentos ou rotinas computacionais. No contexto de banco de dados, persistência é definida como a propriedade de acordo com a qual dados, uma vez que inseridos no banco de dados, permaneçam lá ("persistam"), até que sejam removidos em resposta a alguma solicitação explícita do usuário (DATE, 2015).

2.5.1 Banco de dados relacional

Modelo de entidade-relacionamento é um conjunto de convenções de acordo com as quais se desenha diagramas de entidade-relacionamento. Por sua vez, um diagrama de entidade-relacionamento é uma imagem destinada a explicar o *design* lógico ou conceitual de um determinado banco de dados em um nível de abstração no qual muitos detalhes são omitidos, em particular, detalhes dos tipos subjacentes de dados e quase todas as restrições de integridade. As principais restrições que não são omitidas são as chave-primárias e chaves estrangeiras (DATE, 2015).

Banco de dados relacional é definido como uma coleção de dados relacionados, onde por uma tabela e suas colunas é possível representar uma entidade do mundo real. Cada linha na tabela representa uma coleção de valores de dados relacionados de uma entidade em particular. Os nomes da tabela e de coluna são usados para ajudar a interpretar o significado dos valores em cada linha (ELMASRI et al., 2005).

2.5.2 SGBD

Sistema Gerenciador de Banco de Dados (SGBD) é um sistema responsável pela segurança e proteção dos dados de um banco. Uma das funcionalidades de um SGBD é garantir que somente usuários autorizados tenham acesso de leitura ou gravação em determinadas tabelas ou objetos. O grande diferencial de seu uso é tornar os dados independentes das aplicações (MILANI, 2006).

2.5.3 SQL

Structured Query Language (SQL), se traduz por Linguagem de Consulta Estruturada, oferece ao usuário uma linguagem de alto nível e declarativa. A linguagem SQL é uma linguagem de consulta para interagir com bancos de dados relacionais, sendo um padrão para os SGBD (ELMASRI et al., 2005).

2.5.4 MySQL

MySQL é um SGBD que usa um modelo cliente-servidor. Guarda informações em estruturas registradas, assemelha-se com planilhas porém tem maior capacidade de armazenamento, busca e relacionamento entre os dados (BENTO, 2021).

2.6 Contêiner

Contêineres são ferramentas para entrega de software dentro de aplicações que suportam o paradigma da computação em nuvem referente à *Platform as a Service* (PaaS), que se traduz para Plataforma como Serviço (PAHL, 2015).

De acordo com PAHL (2015), uma aplicação containerizada pode fornecer um tempo de execução de aplicações mais eficiente, escalável e eficaz, bem como a capacidade de desenvolver, testar e implantar aplicações em ambientes controlados e paralelizados.

Dentro do desenvolvimento de um *software*, a RED HAT (2022) relata que os *containers* podem ser utilizados para empacotar conjuntos de códigos do *software* que executam tarefas específicas, e que essas funcionalidades são chamadas de microsserviços.

2.6.1 Docker

Docker é um projeto de *software* livre que atua na automatização e orquestração de implantações de aplicativos como contêineres autossuficientes transportáveis, assim, podendo ser executados em nuvem ou localmente (TORRE; BILL WAGNER; MIKE ROUSOS, 2022).

2.7 UML

UML é uma linguagem amplamente utilizada para estrutura de projetos, bem como o *design* de interfaces de usuários. Ela facilita a abstração que corriqueiramente dificulta a construção dos modelos e estruturas, segundo FOWLER (2004) a construção intelectual e os projetos de *design* dessas interfaces passam por diferentes níveis de abstração, logo, faz-se necessário o uso de ferramenta que possibilitem e adéquem-se as especificações da relação homem-computador.

2.7.1 Diagrama de Casos de Uso

O diagrama de casos de uso especifica um conjunto de funcionalidades, através do elemento sintático “casos de uso”, e os elementos externos que interagem com o sistema, através do elemento sintático “ator” (SILVA, 2007). Assim relaciona dependência, generaliza e associa elementos sendo basicamente usados para visão estática do caso de uso de um sistema. Proporcionando suporte para o procedimento do sistema, em outros termos, os serviços externamente visíveis fornecidos no contexto do seu ambiente (SILVA; MARTINS; DINIZ, 2017).

2.7.2 Diagrama de Atividades

O diagrama de atividade é caracterizado pela execução das ações e as transições que são acionadas pela conclusão de outras ações e atividades (SILVA; MARTINS; DINIZ, 2017). A diferença básica entre os dois conceitos que descrevem comportamento e que a ação é atômica, admitindo particionamento, o que não se aplica a atividade, que pode ser detalhada em atividades e ações (SILVA, 2007).

3

Materiais e Método

Nesse Capítulo são abordadas as metodologias de pesquisa utilizadas e o arcabouço ferramental utilizado, especificando suas versões e integrações, para o desenvolvimento do IFGAccess, um sistema de controle de acesso aos laboratórios do Instituto Federal de Educação, Ciência e Tecnologia de Goiás (IFG) - Campus Formosa, por meio do uso de tecnologia por radiofrequência e dispositivos computacionais embarcados de baixo custo.

3.1 Metodologia da Pesquisa

O projeto de Trabalho de Conclusão de Curso (TCC) em questão baseia-se em uma pesquisa básica descritiva e exploratória, orientada para a resolução de um problema prático real. A problemática solucionada por este trabalho é relacionada aos empréstimos de chaves dos laboratórios de informática do IFG - Campus Formosa. Logo, trata-se de uma pesquisa com base teórica e em revisão bibliográfica, que busca realizar uma revisão a fim de gerar insumos para caso proposto uma solução por meio do uso de ferramentas e tecnologias relacionadas a IoT.

A pesquisa bibliográfica consiste na coleta de informações a partir de textos, livros, artigos e demais materiais de caráter acadêmico. As técnicas utilizadas foram procedimentos operacionais que servem de mediação prática para realização da pesquisa (SEVERINO, 2007).

Por meio da pesquisa bibliográfica foi possível adquirir base de conhecimento necessária para definição, prototipagem e construção do sistema IFGAccess. Para este fim, foram considerados trabalhos que tratam o problema de controle de acesso a ambientes com alto valor agregado. Foram consideradas as contribuições científicas dos últimos 4 (quatro) anos. Para levantamento das referências bibliográficas foram utilizadas as bases de dados acadêmicas, tais como: "SciELO", "Google Acadêmico" e o "Repositório Institucional da UFJF", e em menor número, *sites* de busca.

Durante as buscas foram utilizados refinadores de pesquisa *AND*, que busca assuntos relacionados aos termos inseridos, e *OR*, que realiza a busca um ou mais termos utilizados. Os termos de busca utilizadas foram "controle de acesso", "controle patrimonial", "internet das coisas", "RFID" e "NodeMCU".

3.2 Construção dos artefatos da UML

Para o entendimento do problema e construção da solução foi utilizada a técnica de aplicação de questionário para levantamento dos requisitos para atender as necessidades. A técnica consiste na aplicação de perguntas para as partes interessadas (*stakeholders*) do projeto. Em relação a construção deste questionário, as questões devem ser claras e neutras (de forma a não influenciar a resposta dos participantes). Em relação ao formato, as perguntas podem prover respostas limitadas (múltipla escolha, por exemplo) ou livres, onde a parte interessada pode detalhar melhor seu ponto de vista. De acordo com VAZQUEZ; SIMÕES (2016), após esta coleta de informações, o resultado do questionário serve como base para a análise de requisitos.

Com o auxílio da ferramenta *Google Forms*, foram criadas perguntas variadas de múltipla escolha, bem como perguntas abertas, de modo a entender o problema e validar a aceitação da proposta inicial do projeto. O formulário foi respondido pelos funcionários que compõem a parte administrativa do IFG-Campus Formosa, bem como professores que possuem acesso aos laboratórios.

As perguntas realizadas foram elaboradas e projetadas para suprir duas categorias, i) contextualização, de modo a entender a problemática e ii) validação da solução, para garantir que a solução proposta agrega valor as partes interessadas. A Tabela 3.1 mostra as questões que foram aplicadas e sua devida categorização. Veja que as perguntas Q1 a Q4 possuem intuito de contextualizar o problema de acesso atual aos laboratórios de informática, explorando a forma manual realizada para identificação e liberação de chaves. As perguntas Q5-Q6 são relacionadas a validação da solução tratada neste trabalho, ou seja, de que o acesso automatizado por meio de dispositivos de baixo custo podem melhorar a segurança e confiabilidade dos dados de acesso de ambientes com alto valor agregado.

Com apoio das respostas recebidas pelo questionário foi possível a criação dos seguintes artefatos: i) Diagrama de Casos de Uso; ii) Diagrama de Atividades e iii) Diagrama de Entidade-Relacionamento do sistema IFGAccess. Estes artefatos são apresentados em detalhes no Capítulo 4.

3.3 Ferramentas utilizadas

Para construção dos diagramas e fluxos do sistema foi utilizado o *software* de desenho gráfico Diagrams.net, em sua versão online, disponível em: app.diagrams.net. O diagrama de entidade-relacionamento, por sua vez foi modelado utilizando o *DBeaver*, em sua versão 21.2.3, um cliente *Structured Query Language* (SQL), que foi utilizado também para construir as tabelas e administrar os dados do IFGAccess.

Para a realização de testes, construção e compilação dos códigos desenvolvidos para o NodeMCU V1 na linguagem C++ foi utilizado o Ambiente de Desenvolvimento Integrado (IDE) Arduíno na versão 1.8.18. O protótipo do dispositivo embarcado e seu esquemático foi elaborado

através do *software* Fritzling, na versão 0.9.3.

O projeto do Banco de Dados (BD) se iniciou com a construção do Diagrama Entidade-Relacionamento (DER). Este diagrama é um tipo de fluxograma que explica como entidades (pessoas, objetos ou conceitos) se relacionam entre si dentro de um sistema. As tabelas do banco de dados foram criadas utilizando a linguagem SQL. O SGBD utilizado foi *MySQL* em sua versão 8.0.

Para desenvolver a página *Web*, a nível de *front-end* (interface da aplicação), foi utilizado o *Bootstrap*, na versão 3.3.7. O desenvolvimento do *back-end* (regra de negócio da aplicação), utilizado para recuperar dados recebidos pelo *front-end*, foi desenvolvido através da linguagem de programação PHP em sua versão 7.4. Para a testagem da página sem o usuário do dispositivo embarcado foi utilizado o Postman, na versão 10.5.2, uma plataforma que permite o envio de requisições HTTP.

Todo ambiente de desenvolvimento local foi configurado e executado pelo *Docker*, um conjunto de produtos de plataforma como serviço que permite a entrega de *software* em contêineres, na versão 4.6.1. O dispositivo computacional utilizado durante o desenvolvimento do projeto foi o Windows 11 X64 com processador AMD Ryzen 7 4800H e 8GB de RAM.

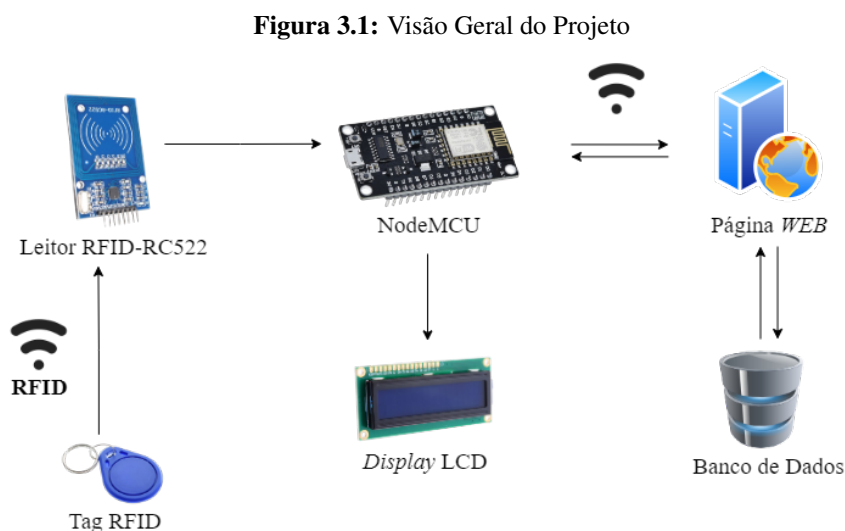
Tabela 3.1: Relação de Questões Realizadas e Categoria

Questões	Categoria
Q1 - Seu nome e cargo no IFG - <i>Campus Formosa</i>	Contextualização
Q2 - Quem tem acesso as chaves do laboratórios de informática do IFG - <i>Campus Formosa</i> ? - <i>Campus Formosa</i>	Contextualização
Q3 - Como é feito o controle de acessos ao laboratório? E como é identificado quem o acessa?	Contextualização
Q4 - A forma de controle de acesso atual é considerada uma forma segura de controle?	Contextualização
Q5 - Justifique sua resposta anterior	Contextualização
Q6 - Já houve relatos de perda ou roubo de equipamentos dos laboratórios de informática?	Validação da solução
Q7 - Um controle automatizado traria uma maior segurança no seu ponto de vista?	Validação da solução
Q8 - Há necessidade de geração de relatórios de acessos?	Validação da solução
Q9 - É necessário o conhecimento do horário de saída do laboratório?	Validação da solução
Q10 - Possui mais alguma informação que considera relevante ser informada?	Validação da solução

Finalmente, para versionamento e disponibilização do código desenvolvido para comunidade científica utilizamos o GitHub¹.

3.4 Construção e Integrações

Após a realização do estudo relacionado aos temas centrais do trabalho (tais como IoT, programação física, RFID, etc); tendo sido levantados os requisitos, que são base para os artefatos UML, e escolhidas as ferramentas necessárias para execução da pesquisa, foram executadas as seguintes fases: i) Virtualização do ambiente *Web* com a utilização do Docker; ii) Criação da página a nível de *front-end*; iii) Criação do banco de dados, para comunicação com o página *Web*, como mostra a Figura 3.1; iv) Desenvolvimento da lógica do *back-end* a nível inicial, com cadastro da *tag* e visualização dos acessos; v) Prototipação do dispositivo físico, exemplificado na Figura 3.1 pelo Leitor RFID, NodeMCU e *Display*; vi) Integração do protótipo com a página *Web*.



¹<https://github.com/felurye/ifgaccess>

4

Resultados

Neste Capítulo são descritos as funcionalidades e a visão completa do protótipo do IFGAccess. Primeiramente, mostramos resultados decorrentes da aplicação do questionário a possíveis *stakeholders*. Após, artefatos da UML foram criados para aprofundamento e elucidação do sistema. Também mostramos passos utilizados para o desenvolvimento, através da contextualização de ferramentas e integrações entre componentes de *software* e *hardware*.

4.1 Questionário

Com o objetivo de entender o fluxo do empréstimo de chaves dos laboratórios do IFG-Campus Formosa, primeiramente foi realizado um questionário para levantamento de informações pertinentes para desenvolvimento do sistema. O questionário foi aplicado a comunidade acadêmica do Campus que lida diariamente com o processo de empréstimo e uso dos laboratórios. Neste contexto, obtivemos respostas de servidores técnico-administrativos, professores e também daqueles que ocupam cargo de gestão administrativa. Ademais, servidores terceirizados e alunos autorizados por docentes para exercer atividades de vínculo acadêmica, como por exemplo, estágio, iniciação científica e/ou TCC, também foram consultados.

Basicamente, o controle de empréstimo de chaves que dão acesso aos laboratórios é baseado por uma lista de autorização (papel escrito) que se encontra junto às chaves no DAA. Normalmente, o fluxo do empréstimo das chaves é realizado por algum servidor técnico-administrativo que esteja presente na sala no momento. O acesso é controlado manualmente, sendo que a pessoa autorizada registra, por escrito, o horário de entrada e saída do laboratório.

O gráfico representado pela Figura 4.1 mostra que, quando perguntados se a forma de controle dos laboratórios é considerada segura, 100% dos entrevistados consideraram que não é seguro. Como justificativa, diversos argumentos foram apontados tais como a possibilidade de não haver nenhum servidor técnico-administrativo na sala no momento em que o empréstimo de chaves é requerido e a possibilidade de erros, esquecimentos e de agir propositalmente de forma maldosa.

Os entrevistados entendem que não existe possibilidade de 100% do tempo ter alguém disponível para liberação dos acessos. Nesse sentido, apontam que quando não há nenhum

servidor na sala, o acesso fica livre, o que torna o acesso aos ambientes vulnerável e suscetível a problemas relacionados a segurança. Também são apontados episódios de esquecimento de assinatura de entrada e saída dos laboratórios e da devolução das chaves. Este controle manual revela problemas de segurança no acesso aos laboratórios, não intencionais como acontece nos esquecimentos, ou intencionais tais como uso da falsificação ideológica ou o não correto preenchimento da lista de acessos.

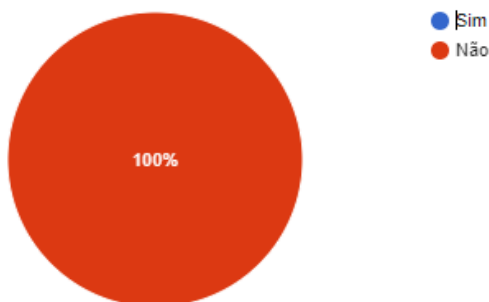


Figura 4.1: Gráfico do questionário sobre segurança do sistema atual

Quando questionados sobre relatos de perda e/ou roubo de objetos dos laboratórios, 50% dos participantes afirmam ter ocorrências sobre, enquanto os outros 50% dizem não ter conhecimento ou que não descartam a hipótese, como pode-se perceber no gráfico da Figura 4.2. Outro resultado interessante da aplicação deste questionário foi a necessidade de um controle automatizado de acessos. Todos os participantes indicaram que o controle automatizado pode melhorar a segurança dos laboratórios, uma vez que não é necessário a verificação manual para controle e liberação destes ambientes.

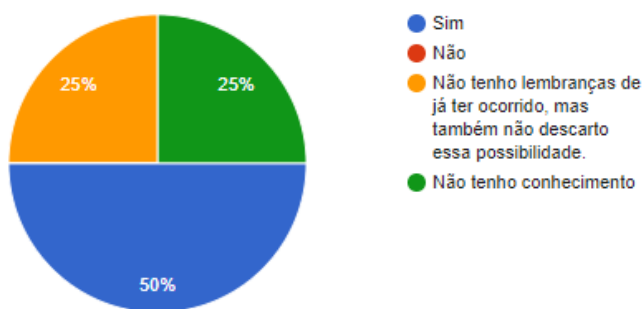


Figura 4.2: Gráfico do questionário sobre perda ou roubo

4.2 Visão Geral do Sistema

Dada a necessidade do controle de acesso aos laboratórios de informática do IFG, com foco na carência de um controle de acessos automatizado e confiável, o sistema foi projetado

para fornecer não-repúdio e integridade aos dados relacionados aos acessos dos laboratórios, tais como usuário requerente, hora de entrada e saída e tempo de permanência. O sistema consiste em: dado que um usuário tenha uma *tag* RFID ativa previamente cadastrada por um usuário administrador, ao aproximar esta *tag* do leitor o acesso é liberado. A identificação da *tag* é capturada pelo leitor RFID e enviada para a página *Web*. A partir deste processo de captura, é possível checar a existência deste dado no BD que retorna informações importantes relacionadas a validade e permissão de acesso. A partir deste momento, se um usuário é devidamente validado a partir de sua *tag* RFID, gera-se registro de acesso deste usuário, bem como data e horário do momento desta permissão. Caso a *tag* não esteja cadastrada, então é realizado o envio da *tag* para página *Web* possibilitando que a *tag* seja cadastrada.

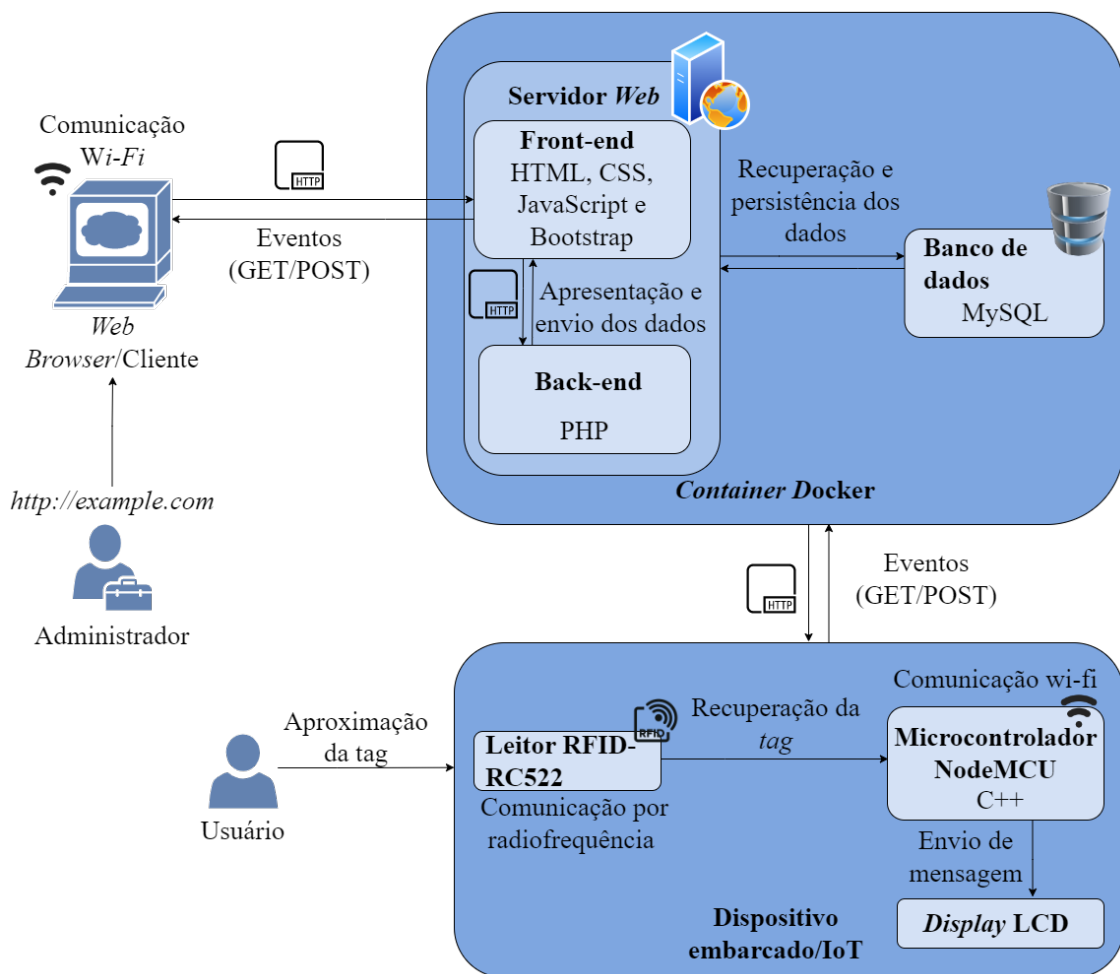


Figura 4.3: Visão Geral da Arquitetura do IFGAccess

O sistema de controle de acesso IFGAccess é composto pela página *Web* e um banco de dados que estão alocados em um contêiner *Docker*. Também fazem parte do sistema o conjunto de dispositivos IoT, como é mostrado na Figura 4.3. Os dispositivos IoT são compostos por sensores, atuadores, um microcontrolador e o módulo de comunicação Wi-Fi. A camada que faz interface com o usuário é responsável por intermediar com o dispositivo embarcado. Já a camada de sensores é responsável por capturar informações da *tag* RFID que serão utilizadas

pelo microcontrolador para tomada de decisões. Finalmente, a camada *Web* é responsável por persistir os dados junto ao banco de dados, recebendo informações do microcontrolador pelo módulo de comunicação Wi-Fi. Em posse dos dados coletados pelos atuadores e processados pelo microcontrolador, comandos são enviadas para atuadores que realizam determinadas ações, tais como: liberar o acesso ou exibir esses dados no *Display Liquid Crystal Display* (LCD).

O protocolo de comunicação utilizado para a troca de mensagens é o HTTP, o que possibilita o envio e recebimento de dados tanto entre o dispositivo IoT e a página *Web*. Essa comunicação ocorre por meio dos métodos *GET* e *POST*. O leitor RFID-RC522 é responsável por realizar a leitura das *tags* quando aproximadas. Cada *tag* RFID reproduz uma frequência única, o que possibilita sua identificação exclusiva. Após a leitura da *tag* e o processamento pelo dispositivo NodeMCU, é realizado o envio para a página *Web* que busca a informação de identificação da *tag* no banco de dados. Durante todo esse processo são enviadas mensagens de *log* para o *display* para que o usuário esteja ciente de todo processo. Caso a *tag* não esteja cadastrada, ou seja, não existe registro no banco de dados, a aplicação envia para a página *Web* o identificador da *tag*, o que possibilita que a *tag* seja associada a um usuário. O servidor é responsável também por registrar no banco de dados as informações de data do acesso.

4.2.1 Diagrama de Atividades

O diagrama de atividades mostra o fluxo do sistema e de tomada de decisões. Os círculos fechados do diagrama, exibidos na Figura 4.4, representam o início de um fluxo. Usuário tem como ponto de partida a ação de aproximar a *tag* RFID do leitor. Caso a *tag* esteja cadastrada no sistema, Dispositivo valida se há alguma entrada (*check-in*) pendente de saída (*check-out*). O registro da entrada (*check-in*) de alguma *tag* RFID só é possível se, e somente se, não houver nenhuma outra *tag* com o estado de *check-in* habilitado. No cenário em que a *tag* não possui cadastro junto ao banco de dados, Administrador pode realizar o cadastro de um usuário, vinculando-o a *tag* aproximada. O administrador também possui como ação visualizar a lista de acessos e a possibilidade de edição dos dados dos usuários cadastrados.

4.2.2 Diagrama de Casos de Uso

O diagrama de casos de uso descreve os atores e suas funções e ações dentro do sistema IFGAccess. No diagrama representado na Figura 4.5 é apresentado os 2 (dois) atores do sistema, usuário, que interage com o dispositivo IoT aproximando a *tag* RFID do leitor, que possibilita o acesso ou a realização do cadastro, e o administrador, que tem como ações cadastrar o usuário, vincular a *tag* ao cadastro e visualizar a lista de acessos ao laboratório por meio da página *Web*.

4.2.3 Diagrama de Entidade-Relacionamento

De modo a representar as entidades do sistema apresentamos o DER na Figura 4.6. As tabelas *users* e *access* contém os atributos dos usuários e acessos do sistema, respectivamente.

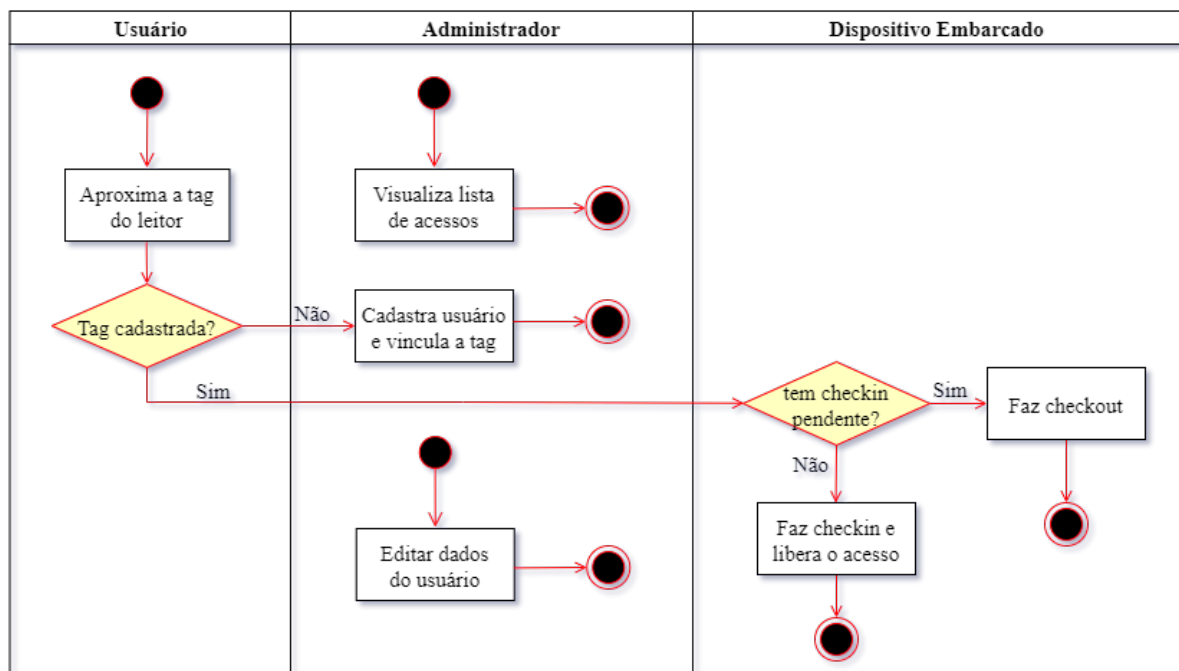


Figura 4.4: Diagrama de Atividades do IFGAccess

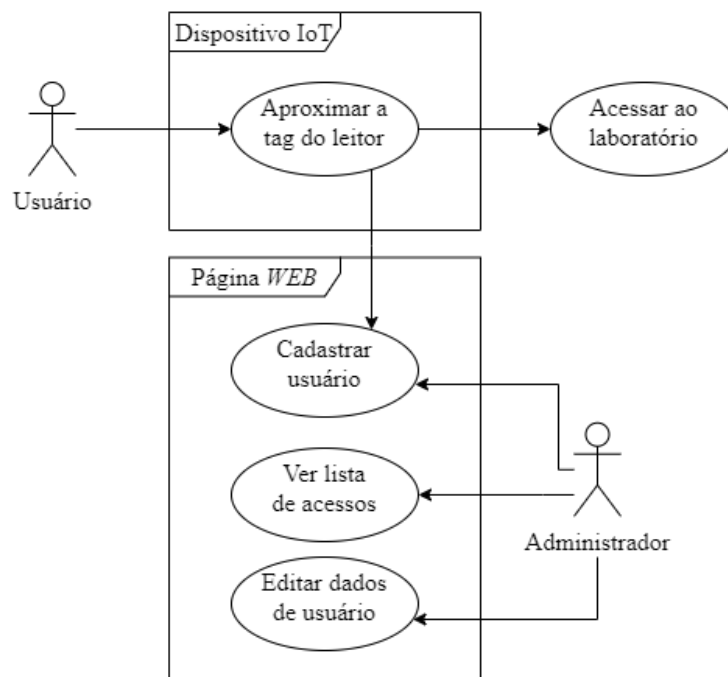


Figura 4.5: Diagrama de Casos de Uso - IFGAccess

Como pode-se notar na Figura, a tabela *users* registra dados do usuário, tais como matrícula, nome, e-mail, telefone e valor da *tag*. Esses dados são importantes no contexto de identificação. A cada novo registro de usuário é gerado um Identificador (ID) dinâmica. A tabela *access* se relaciona com a tabela de *users* por meio do ID da tabela *users* com a chave estrangeira em *access*. A tabela *access* também possui um ID gerada aleatoriamente, o número da sala acessada

bem como registros de data e hora de entrada e saída.

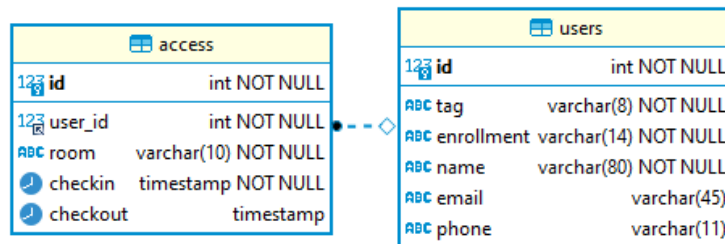


Figura 4.6: Diagrama de Entidade-Relacionamento do IFGAccess

4.3 Sistema Web

A construção da página *Web* iniciou-se com a containerização do ambiente de desenvolvimento, através do *Docker*, como pode ser observado na Figura 4.7. O projeto contém duas imagens: uma do *apache*, para que seja possível a interpretação da linguagem *PHP*, e a segunda com o banco de dados *MySQL*, utilizando para persistência dos dados.

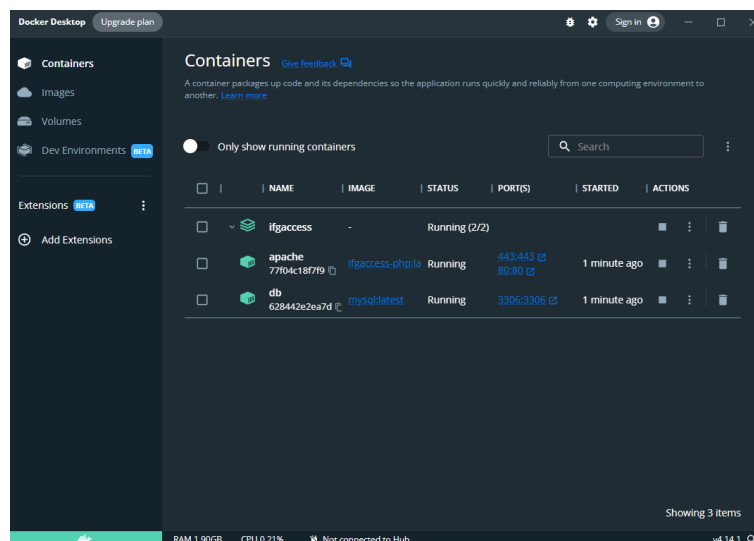


Figura 4.7: Docker do projeto IFGAccess

Com o ambiente configurado, então foi possível iniciar o desenvolvimento do sistema *Web*. A Figura 4.8 exibe a consulta de uma *tag* não cadastrada. Para realização dos testes sem o dispositivo físico embarcado criamos requisições por meio do *software* Postman, como é observado na Figura 4.9. Com o uso deste programa é possível passar o endereço e também o *body* de uma requisição que é capturado pela página *Web* pelo meio do método *POST* do *HTTP*. Também é possível visualizar na Figura 4.9 o retorno da página *Web*, como *Status Code* e *body*.

Nos cenários em que a *tag* é cadastrada, a página de consulta apresenta os dados da *tag*, como é mostrado na Figura 4.10. Já na página de cadastro, ao enviar uma requisição com o



Figura 4.8: Página de consulta de *tags* não cadastradas

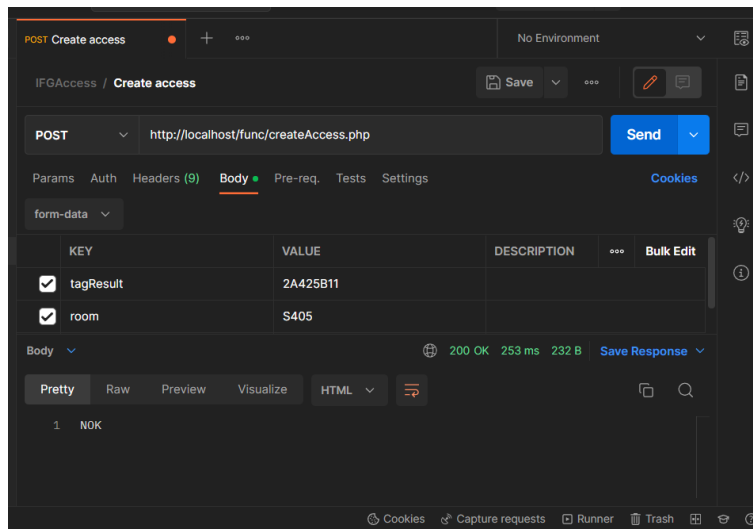


Figura 4.9: Envio de requisição usando *Postman*

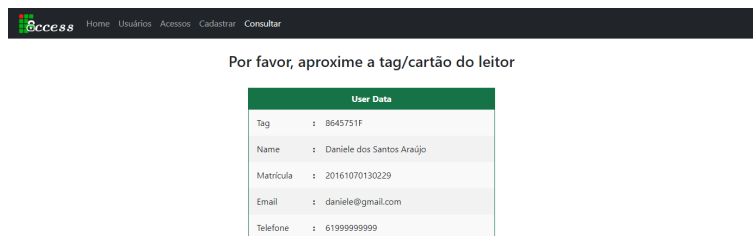


Figura 4.10: Página de consulta de *tags* cadastradas

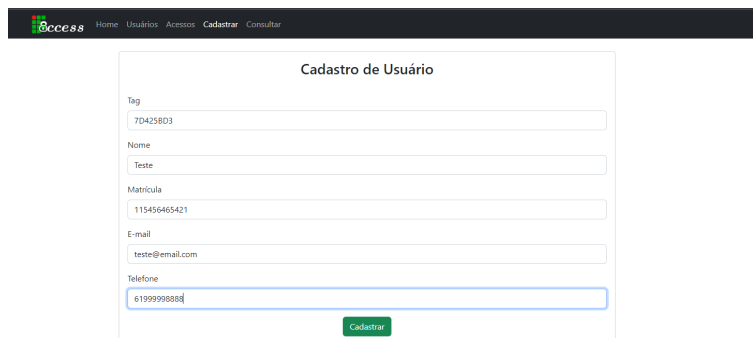
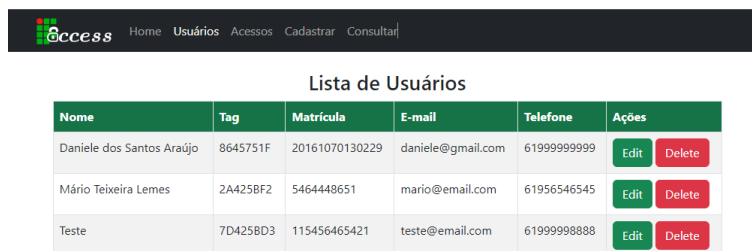


Figura 4.11: Página de cadastro de usuário

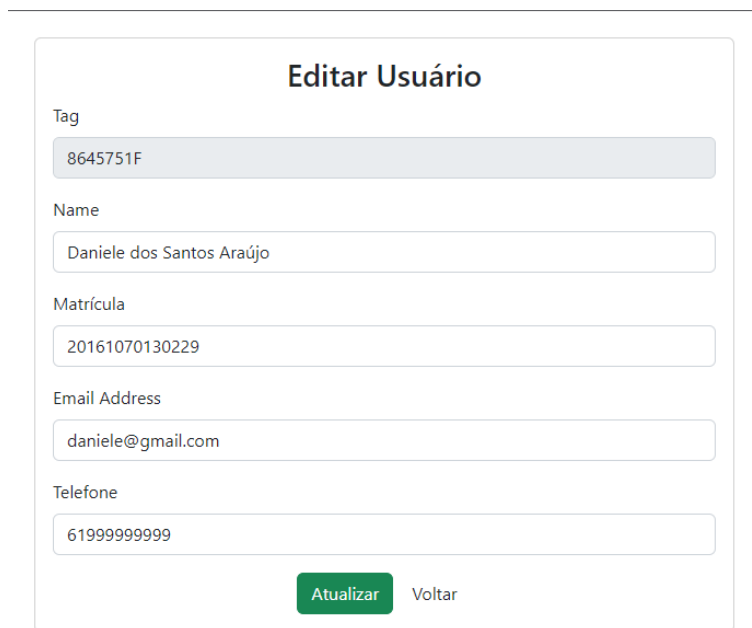
identificador da *tag* é possível realizar o cadastro da *tag*, informando o nome, matrícula, telefone e e-mail do usuário, como mostra a Figura 4.11.

A partir da página de listagem de usuários do sistema, na Figura 4.12 é possível realizar a edição dos dados do usuário, para casos de erros nos dados ou troca de e-mail ou telefone, por exemplo. A Figura 4.13 exibe a página que permite a atualização das informações após seleção de um usuário para edição.



Nome	Tag	Matrícula	E-mail	Telefone	Ações
Daniele dos Santos Araújo	8645751F	20161070130229	daniele@gmail.com	61999999999	Edit Delete
Mário Teixeira Lemes	2A425BF2	5464448651	mario@email.com	61956546545	Edit Delete
Teste	7D425BD3	115456465421	teste@email.com	61999998888	Edit Delete

Figura 4.12: Página de lista de usuário cadastrados



Editar Usuário

Tag
8645751F

Name
Daniele dos Santos Araújo

Matrícula
20161070130229

Email Address
daniele@gmail.com

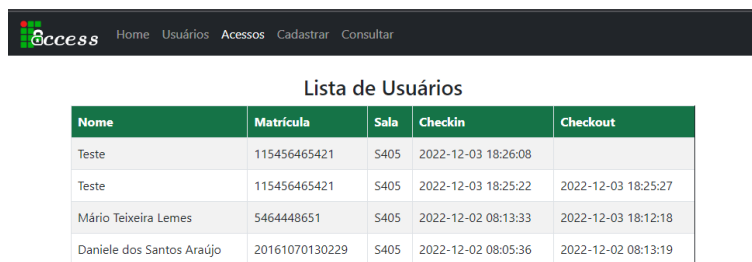
Telefone
61999999999

Atualizar Voltar

Figura 4.13: Página de editar dados de usuário

A página de acessos, Figura 4.14, lista os acessos através do monitoramento do usuário requisitante (nome e matrícula), identificação da sala e data e hora de *check-in* e *check-out*. Caso o usuário ainda não tenha realizado o *check-out*, a coluna *check-out* permanece vazia.

Quanto a lógica para validação de *tags* RFID, primeiramente a requisição *POST* recebida é verificada. Nessa etapa de verificação, é checado se o identificador da *tag* foi devidamente recebido, como pode-se perceber no Código 4.1. Caso a requisição *POST* for considerada válida, outras validações são executadas. Primeiro, verifica-se se a *tag* já possui cadastro junto ao banco de dados. Caso não exista registro para *tag*, é retornado o valor NOK no *body* da requisição.



Nome	Matrícula	Sala	Checkin	Checkout
Teste	115456465421	S405	2022-12-03 18:26:08	
Teste	115456465421	S405	2022-12-03 18:25:22	2022-12-03 18:25:27
Mário Teixeira Lemes	5464448651	S405	2022-12-02 08:13:33	2022-12-03 18:12:18
Daniele dos Santos Araújo	20161070130229	S405	2022-12-02 08:05:36	2022-12-02 08:13:19

Figura 4.14: Página de lista de acessos

```

1 $tag = null;
2 if (!empty($_POST['tagResult'])) {
3     $tag = $_REQUEST['tagResult'];
4     $room = $_REQUEST['room'];
5 }
6
7 $userResult = Database::query("SELECT * FROM users where tag = '$tag'
8     ");
9 if (empty($userResult)) {
10     echo "NOK";
11     return;
12 }

```

Código 4.1: Código para validação da requisição *POST*

Caso exista cadastro vinculado a *tag*, é validado se, vinculada a esta *tag*, existe algum procedimento de *check-in* em aberto. IFGAccess considera um *check-in* em aberto se não foi realizado o procedimento de *check-out*. Este comportamento inviabiliza que um novo *check-in* seja realizado por outro usuário. Dessa forma, com a existência de *check-out* a ser realizado, o sistema mostra a mensagem *"Wait for another user to checkout"*, como pode ser verificado no Código 4.2.

```

1 $userId = $userResult[0]['id'];
2
3 $checkinAnotherUserResult = Database::query("SELECT * FROM access
4     where room = '$room' AND user_id != '$userId' AND checkout is NULL
5     ORDER BY checkin DESC");
6
7 if (!empty($checkinAnotherUserResult)) {
8     echo "Wait for another user to checkout";
9     return;
10 }

```

Código 4.2: Código para verificação de *check-out* pendente

Se a *tag* possuir cadastro junto ao banco de dados e não existir nenhum usuário com *check-in* pendente de *check-out* é criado um novo registro de acesso, com identificação de usuário, sala, data e hora da entrada. A lógica desse fluxo de informação pode ser observada no Código 4.3.

```
1 $checkinResult = Database::query("SELECT * FROM access where user_id
   = '$userId' AND checkout is NULL ORDER BY checkin DESC");
2
3 if (empty($checkinResult)) {
4     echo "Checkin";
5
6     $dataCheckIn = [
7         'user_id' => $userId,
8         'checkin' => $dateNow,
9         'room' => $room,
10    ];
11
12    Database::create("access", $dataCheckIn);
13    return;
14 }
```

Código 4.3: Código para verificação de *check-out* pendente

4.4 Montagem do protótipo

A Figura 4.15 mostra as conexões realizadas entre NodeMCU, leitor RFID-RC522 e *Display* LCD. Para realização das conexões entre o *Display* e o NodeMCU utilizamos o módulo *Inter-Integrated Circuit* (I2C) para reduzir a quantidade de portas utilizadas. A Tabela 4.1 mapeia as portas e conexões entre os dispositivos. A biblioteca responsável pela leitura e gravação do *Display* LCD realiza monitoramento das portas do NodeMCU de forma nativa, sendo assim necessário apenas o mapeamento das portas D4 para o SSPIN (transmissão de dados) e D3 para o RSTPIN (*reset* de leitura do RFID) para a manipulação dos dados junto ao RFID-RC522.

4.5 Comunicação entre as partes do sistema

Para a comunicação entre o dispositivo embarcado e a página *Web* é requisito que NodeMCU esteja conectada à Internet para comunicação entre as partes do sistema através do Wi-Fi. O Código 4.4 mostra como é realizado a conexão a uma rede Wi-Fi. Já na Figura 4.16 é possível visualizar o fluxo de conexão por meio do *Monitor Serial* do Arduíno IDE.

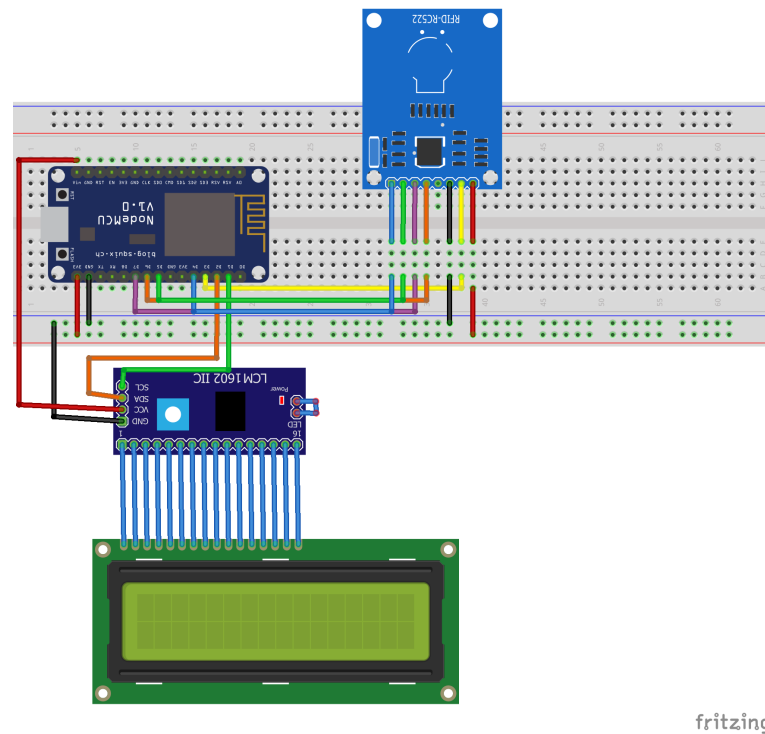


Figura 4.15: Desenho do protótipo

Tabela 4.1: Relação de conexões dos dispositivos

NodeMCU	Leitor Rfid Mfrc522	I2C
GND		GND
Vin		VCC
D2		SDA
D1		SCL
D4	SDA	
D5	SCK	
D7	MOSI	
D6	MISO	
GND	GND	
D3	RST	
3V3	3V3	

```

1 const char *ssid = "Nome da rede";
2 const char *password = "Senha da rede";
3
4 void setup()
5 {
6   Serial.begin(115200);
7   WiFi.begin(ssid, password);
8
9   Serial.print("Connecting");
10  while (WiFi.status() != WL_CONNECTED)

```

```
11 {
12     Serial.print(".");
13     delay(100);
14 }
15
16 Serial.print("Conectado a rede com sucesso: ");
17 Serial.println(ssid);
18 Serial.print("Endereço IP: ");
19 Serial.println(WiFi.localIP());
20 }
```

Código 4.4: Código para conexão do NodeMCU ao enlace de comunicação Wi-Fi

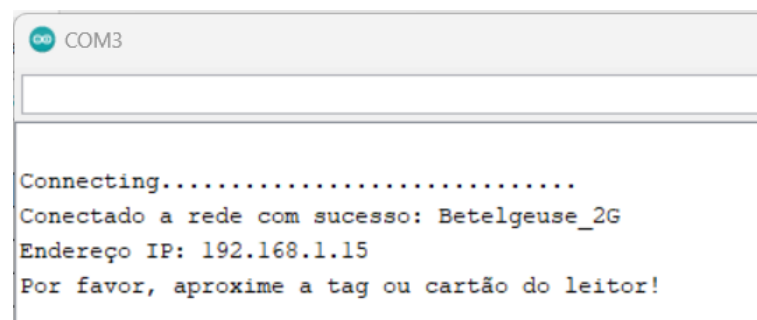


Figura 4.16: Monitor Serial - Conexão com a Internet

No Código 4.5 nota-se que o leitor RFID permanece ativo e aguardando (buscando) a aproximação de uma *tag* de identificação. Este processo é acompanhado pela mensagem "Aproxime a *tag* do leitor!", exibida no *Display*, como pode ser observado na Figura 4.17. Quando uma *tag* é aproximada, uma operação *POST* é executada para o código PHP validador de acessos, enviando no *body* a sala e a *tag*.

```
1 void loop()
2 {
3     readsuccess = getid();
4
5     if (readsuccess)
6     {
7         HTTPClient http;
8         String tagResult, postData;
9         tagResult = StrUID;
10
11         postData = "tagResult=" + tagResult + "&room=S405";
12         Serial.print("TAG RECEBIDA: " + tagResult + "\n");
13     }
```

```
14 http.begin(wifiClient, "http://192.168.1.19/func/createAccess.php
    ");
15 http.addHeader("Content-Type", "application/x-www-form-urlencoded
    ");
16
17 int httpCode = http.POST(postData);
18 String response = http.getString();
19
20 Serial.print("Status Code: ");
21 Serial.println(httpCode);
22 Serial.println("Response: " + response);
23
24 validateAccess(response, tagResult);
25
26 http.end();
27 delay(1000);
28
29 lcd.clear();
30 lcd.setCursor(0, 0);
31 lcd.print("Aproxime a tag");
32 lcd.setCursor(0, 1);
33 lcd.print("do leitor!");
34 }
35 }
```

Código 4.5: *Loop* no NodeMCU para captura de *tags* RFID

Após a validação das informações recebidas feita pelo sistema *Web*, o mesmo dispara as mensagens de retorno impressas no *Display* para notificação do usuário sobre o fluxo de comunicação. O Código 4.6 mostra a lógica de cada retorno: ao receber da página "*Checkin*", o texto da Figura 4.19 é exibido. Caso o retorno seja "*Checkout*", a mensagem impressa é a da Figura 4.21. Quando ainda existe um *checkout* pendente de outro usuário, isto é, outra *tag* ativa, é exibida mensagem semelhante à Figura 4.20. Caso contrário, em caso de acesso negado, é apresentado a mensagem da Figura 4.18.

```
1 void validateAccess(String response, String tag) {
2   lcd.clear();
3   lcd.setCursor(0, 0);
4   if (response == "Checkin") {
5     lcd.print("Acesso liberado!");
6   } else if (response == "Checkout") {
7     lcd.print("Checkout!!!");
```

```
8 }
9 else if (response == "Wait for another user to checkout") {
10     lcd.print("Aguarde checkout!");
11 } else {
12     lcd.print("Acesso negado!");
13 }
14 lcd.setCursor(0, 1);
15 lcd.print("TAG: " + tag);
16 delay(2000);
17 }
```

Código 4.6: Validação de *check-in* pendente de *check-out*

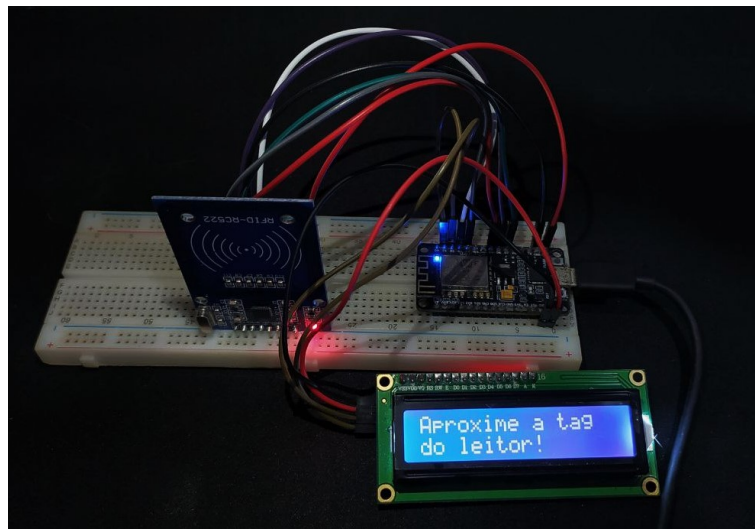


Figura 4.17: Display - Aproxime a tag

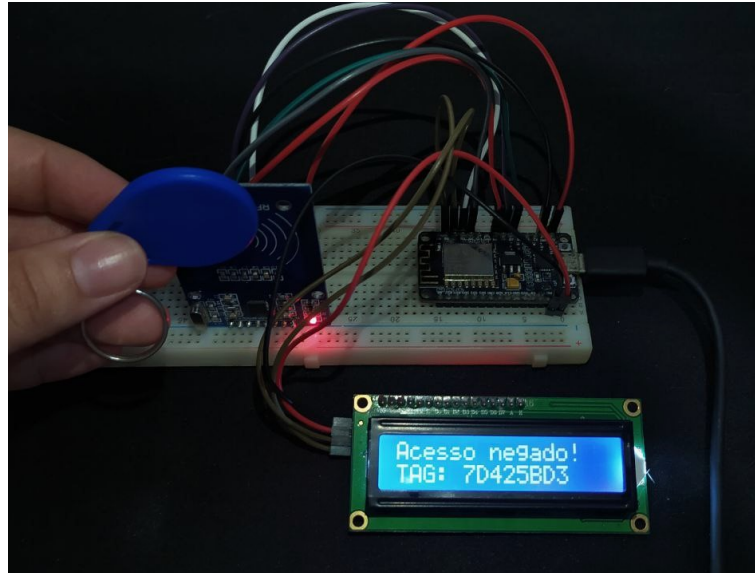


Figura 4.18: *Display - Acesso negado*

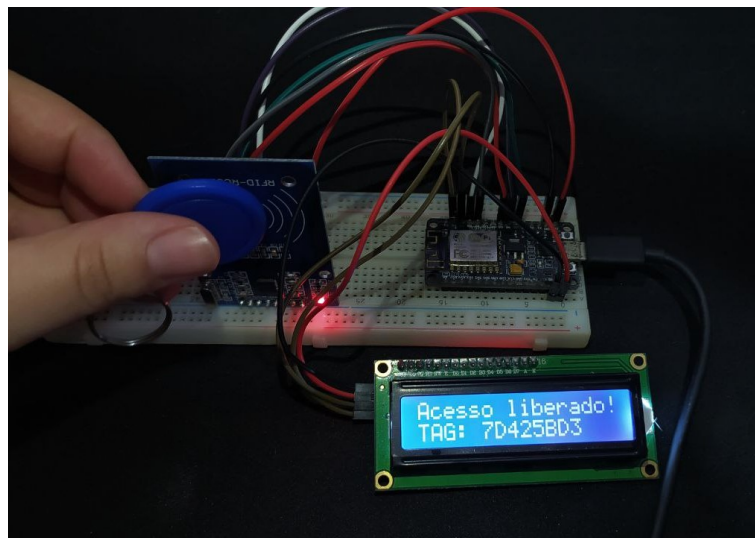


Figura 4.19: *Display - Acesso liberado*

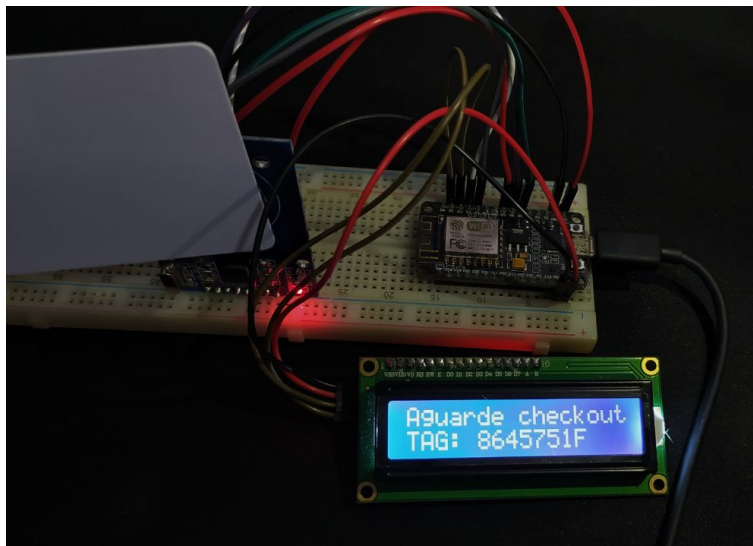


Figura 4.20: *Display - Aguarde checkout*

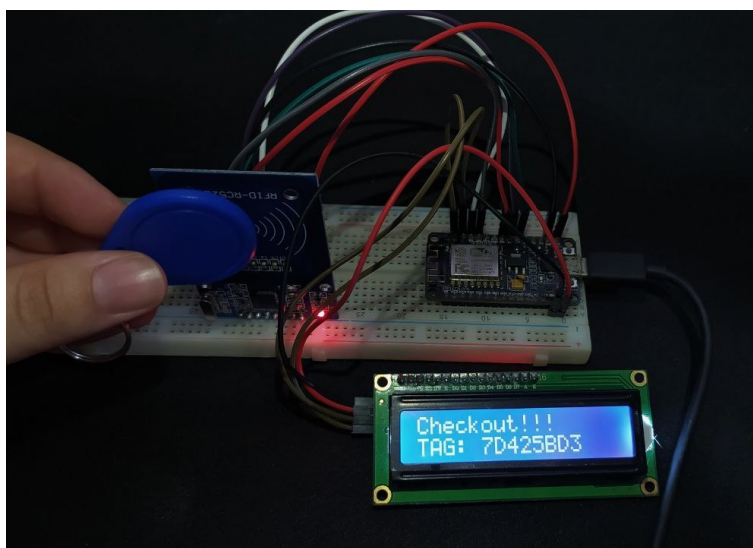


Figura 4.21: *Display - Checkout*

5

Conclusão

Objetos inteligentes oferecem uma série de recursos que possibilitam a construção de sistemas para monitoramento remoto, automação, controle de dispositivos, e podem ser usados para aumentar a eficiência, reduzir custos, melhorar a segurança e facilitar a vida das pessoas. Além disso, é possível criar aplicações e serviços para cada dispositivo inteligente, permitindo criação de redes de dispositivos inteligentes e conectá-los entre si para que possam trocar a dados e informações de forma eficiente e segura.

Este trabalho apresentou o desenvolvimento de um protótipo para o controle automatizado de acessos aos laboratórios de informática do IFG-*Campus* Formosa por meio da tecnologia de radio-frequência e dispositivos computacionais de baixo custo, no intuito de geração uma maior confiabilidade nos dados dos acessos. Para atingir este objetivo, uma página *Web* foi construída para realizar consulta e cadastro de *tags* RFID, listagem de usuários e acessos, além da edição de dados através de um ambiente containerizado.

O sistema embarcado, baseado no microcontrolador ESP8266 e integrado ao módulo RFID, mostraram-se adequados na integração com a página *Web*. A comunicação entre componentes de *software* e *hardware* se baseou no HTTP, o que permitiu o envio de *tags* lidas pelo leitor RFID e validação das ações com base nas respostas das requisições. Os resultados obtidos nos testes integrados entre essas partes do sistema se mostraram satisfatórios, cumprindo com o objetivo de registrar entrada e saída dos laboratórios de forma automatizada e garantir integridade e segurança física dos laboratórios.

5.1 Sugestão de Trabalhos Futuros

O leitor RFID-RC522 reduz a cobertura de *tags* que podem ser utilizadas na implementação e testes, uma vez que a identificação das *tags* utilizadas não podem ultrapassar 8 *bits*, correspondendo a modelos básicos disponíveis no mercado para fins de pesquisa e para elaboração de projetos de pequeno porte. Em uma nova versão e evolução do protótipo, pretende-se adicionar um módulo RFID mais robusto para que seja possível o cadastro de *tags* de maior tamanho e com maior poder de alcance. Ademais, pretende-se também adicionar um relé e uma trava elétrica solenoide aos componentes de *hardware* para efetiva liberação das portas dos

laboratórios. Sugere-se também como melhorias para a página *Web* um sistema de *login* que limite acessos as funcionalidades de cadastro e edição para o usuário administrador, assim como a inclusão da funcionalidade de geração de relatório de acessos. Juntamente com as melhorias ao sistema, propõe-se a apresentação da solução, com orçamento dos dispositivos, a chefia do IFG - *Campus* Formosa para a implementação e testagem do mesmo em um ambiente real.

Referências

- ASSOCIATION, G. et al. **The Mobile Economy Latin America**. [S.l.]: <https://www.gsma.com/mobileeconomy/latam>, 2019.
- BANZI, M.; SHILOH, M. **Primeiros Passos com o Arduino–2ª Edição**: a plataforma de prototipagem eletrônica open source. [S.l.]: Novatec Editora, 2015.
- BATES, M. P. **Programming 8-bit PIC microcontrollers in C**: with interactive hardware simulation. Burlington, MA: Newnes, 2008.
- BENTO, E. J. **Desenvolvimento web com PHP e MySQL**. [S.l.]: Editora Casa do Código, 2021.
- BRITO, C. V. d. S. P. et al. Etiquetas inteligentes na administração pública: análise da viabilidade no controle patrimonial da univasf. **ForScience**, [S.l.], v.7, n.2, 2019.
- COULOURIS, G. et al. **Sistemas Distribuídos**: conceitos e projeto. Porto Alegre: Bookman Editora, 2013.
- DATE, C. J. **Introdução a sistemas de bancos de dados**. [S.l.]: Elsevier Brasil, 2004.
- DATE, C. J. **The New Relational Database Dictionary**: terms, concepts, and examples. Sebastopol, CA: "O'Reilly Media, Inc.", 2015.
- DENARDIN, G. W.; BARRIQUELLO, C. H. **Sistemas operacionais de tempo real e sua aplicação em sistemas embarcados**. São Paulo, SP: Editora Blucher, 2019.
- ELMASRI, R. et al. **Sistemas de banco de dados**. , [S.l.], 2005.
- Espressif Systems. **ESP8266EX Datasheet**. 2022.
- FEZARI, M.; AL DAHOUD, A. Integrated development environment “IDE” for Arduino. **WSN applications**, [S.l.], p.1–12, 2018.
- FIRAT, Y.; UĞURLU, T. Automatic Garage Door System with Arduino For defined licence plates of cars. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND DATA PROCESSING (IDAP), 2018., Malatya, Turkey. **Anais...** [S.l.: s.n.], 2018. p.1–8.
- FOWLER, M. **UML distilled**: a brief guide to the standard object modeling language. [S.l.]: Addison-Wesley Professional, 2004.
- GROUP, T. P. **História do PHP**. Disponível em: https://www.php.net/manual/pt_BR/history.php.php. Acesso em: 08 jan. de 2023.
- JIMÉNEZ, M.; PALOMERA, R.; COUVERTIER, I. **Introduction to embedded systems**. New York, NY: Springer, 2013.
- KUROSE, J.; ROSS, K. **Redes de computadores e a internet**: uma abordagem top-down. São Paulo, SP: Pearson Universidades, 2013.
- LISBOA, R. A. C. C. B. **Aplicação de controle patrimonial utilizando tecnologia RFID**. 2021. B.S. thesis — Universidade Tecnológica Federal do Paraná.

MAIA, G. Q. et al. Protótipo de controle de acesso utilizando RFID para automatização da segurança interna da UFERSA - Campus Mossoró. , Mossoró, RN, 2019.

MANCINI, M. Internet das Coisas: história, conceitos, aplicações e desafios. **Project Management Institute - PMI**, [S.l.], v.27, 2017.

MEDEIROS, V. de et al. SISTEMA DE AUTOMAÇÃO, CONTROLE E REGISTRO DE ACESSO NO GRUPO DE PESQUISA GAMA. **Anais do Salão Internacional de Ensino, Pesquisa e Extensão**, [S.l.], v.11, n.2, 2020.

MILANI, A. **MySQL - Guia do Programador**. São Paulo, SP: Novatec Editora, 2006.

NODEMCU. **NodeMcu Connect Things EASY**. Disponível em: https://www.nodemcu.com/index_en.html; Acesso em: 09 de jul. 2022.

PAHL, C. Containerization and the paas cloud. **IEEE Cloud Computing**, [S.l.], v.2, n.3, p.24–31, 2015.

PEIXOTO, R. M. B. **Comunicações RFID para Identificação e Controle de Acessos**. 2022. Tese (Doutorado em Ciência da Computação) — .

POHL, K. et al. **Model-based engineering of embedded systems: the spes 2020 methodology**. Heidelberg: Springer, 2012.

RED HAT, I. **O que é containerização?** Disponível em: <https://www.redhat.com/pt-br/topics/cloud-native-apps/what-is-containerization?pfe-8ozr9lu56=soluç~oes-red-hat>. Acesso em: 04 dez. de 2022.

RICARTE, I. L. M. Programação orientada a objetos com c++. **Versão preliminar (apostila). Campinas: Universidade Estadual de Campinas**, [S.l.], 2001.

ROBOCORE. **nodeMCU v2**. Disponível em: <https://www.robocore.net/wifi/nodemcu-esp8266-12-v2?gclid=CjwKCAiA8OmdBhAgEiwAShr402L49ZUONit8MDRIkcK6LYodKUNlBTRfA7A1cYau92ASNYpBwE>. Acesso em: 07 jan. de 2023.

ROBOCORE. **Módulo Relé**. Disponível em: <https://www.robocore.net/atuador-rele/modulo-rele?gclid=CjwKCAiA8OmdBhAgEiwAShr407GnW2nhcFwpYqJBmoiYOvvh4DIt14GhFfkNdOabnN8CZ2TBwE>. Acesso em: 07 jan. de 2023.

ROBOCORE. **Tag RFID Programável 13,56Mhz**. Disponível em: <https://www.robocore.net/rfid/tag-rfid-programavel-mifare-chaveiro-13mhz?gclid=CjwKCAiA8OmdBhAgEiwAShr409KLF-F4fqzxyx2xbLZ3lqKXE3l2savHAjvvpbZalAejvApOBwE>. Acesso em: 07 jan. de 2023.

SAWAYA, M. **Dicionário De Informática & Internet: inglês-português**. São Paulo, SP: Nobel, 2002.

SEMICONDUCTORS, N. **MFRC522: standard performance mifare and ntag frontend**. 2016. [S.l.]: Rev, 2018.

SEVERINO, A. J. **Metodologia do Trabalho Científico**. São Paulo, SP: Cortez Editora, 2007.

SILVA, R. O. da; MARTINS, B. R.; DINIZ, W. G. A complexibilidade da UML e seus diagramas. **TECNOLOGIAS EM PROJEÇÃO**, Brasília, v.8, n.1, p.86–99, 2017.

SILVA, R. P. e. **UML 2 em Modelagem Orientada a Objetos**. Florianópolis: Visual Books, 2007.

SILVEIRA, D. W. et al. Desenvolvimento de uma fechadura eletrônica: um sistema de controle de acesso com registro em banco de dados e site de gerenciamento. , [S.l.], 2021.

SUNDMAEKER, H. et al. Vision and challenges for realising the Internet of Things. **Cluster of European research projects on the internet of things, European Commission**, [S.l.], v.3, n.3, p.34–36, 2010.

TANENBAUM, A. S.; STEEN, M. V. **Sistemas Distribuídos. Princípios e Paradigmas**. [S.l.]: Pearson, 2008.

TECHNOLOGY, H. **1 Channel 5V Optical Isolated Relay Module**. Disponível em: <https://handsontec.com/dataspecs/relay/1Ch-relay.pdf>. Acesso em: 04 dez. de 2022.

TORRE; BILL WAGNER; MIKE ROUSOS, C. de la. **The New Relational Database Dictionary: terms, concepts, and examples**. Redmond, Washignton: Microsoft Developer Division, .NET and Visual Studio product teams, 2022.

VAZQUEZ, C. E.; SIMÕES, G. S. **Engenharia de Requisitos: software orientado ao negócio**. [S.l.]: Brasport, 2016.

WEINSTEIN, R. RFID: a technical overview and its application to the enterprise. **IT professional**, [S.l.], v.7, n.3, p.27–33, 2005.

Apêndice

A

Questionário

Quem tem acesso as chaves do laboratórios de informática do IFG - Câmpus Formosa?

1. Docentes, técnicos administrativos, servidores terceirizados e alunos autorizados.
2. Docentes, técnicos administrativos e alunos autorizados por docentes
3. Teoricamente os servidores e alunos vinculados a alguma atividade acadêmica (Ex: estágio, TCC...)
4. Docentes, servidores técnico-administrativos e terceirizados do Departamento de Áreas Acadêmicas

Como é feito o controle de acessos ao laboratório? E como é identificado quem o acessa?

1. Há uma lista de autorização que fica ao lado do claviculário onde estão as chaves. Além dos servidores, que possuem livre acesso às chaves, os alunos que estão nesta lista também podem pegar as chaves. Normalmente eles se identificam a algum servidor que esteja na sala no momento.
2. Definido pelo departamento acadêmico (DAA)
3. O acesso é controlado manualmente. O servidor ou aluno deve retirar a chave do escriturário de chaves e anotar horário de entrada e saída dos laboratórios. A identificação de quem acessa o laboratório é realizada apenas com a assinatura simples.
4. O controle é feito através da data e horário da retirada e da devolução da chave do laboratório na sala da Coordenação de Apoio Administrativo ao DAA.

A forma de controle de acesso atual é considerada uma forma segura de controle?

1. Não

2. Não
3. Não
4. Não

Justifique sua resposta anterior.

1. Não é segura pois em muitos momentos, quando não há nenhum servidor na sala, o acesso às chaves fica praticamente livre.
2. As chaves são de fácil acesso a qualquer um e não há um controle rigoroso pelos docentes do acesso
3. O controle de acesso realizado não é seguro pois podem haver esquecimentos na assinatura da folha de retirada de chaves. Pode acontecer também a falsificação de identidade.
4. Infelizmente, a sala que contem o quadro de chaves é de livre acesso a qualquer pessoa no interior do DAA.

Já houve relatos de perda ou roubo de equipamentos dos laboratórios de informática?

1. Não tenho lembranças de já ter ocorrido, mas também não descarto essa possibilidade.
2. Sim
3. Sim
4. Não tenho conhecimento

Um controle automatizado traria uma maior segurança no seu ponto de vista?

1. Creio que sim!
2. Com certeza
3. Sim
4. Sim.

Há necessidade de geração de relatórios de acessos?

1. Talvez
2. Sim
3. Sim

4. Talvez

É necessário o conhecimento do horário de saída do laboratório?

1. Seria interessante essa informação também fazer parte do controle.
2. Sim
3. Sim, pois poderia ser usado para controle de frequência. Ex: se um estagiário necessita fazer 4 horas por dia. O horário de entrada e saída pode ser usado para controle de realização deste horário.
4. Sim.

Possui mais alguma informação que considera relevante ser informada?

1. Já pensou-se, como uma alternativa possível, que as chaves fiquem sob controle da recepção do Câmpus, e não na sala da Coordenação de Apoio Administrativo. Mas toda ideia de melhoria deste controle é muito bem vinda.
2. Identificação eletrônica do servidor ao emprestar chaves
3. Nada mais a declarar.
4. .