

Desenvolvimento de um protótipo básico de hardware e software para segurança física e controle de acesso em ambientes institucionais que contenham ativos e informações de valor

Development of a basic prototype of hardware and software for physical security and access control in institutional environments that contains valuable assets and information

Desarrollo de un prototipo básico de hardware y software para seguridad física y control de acceso en ambientes institucionales que almacenen activos e información de valor

Recebido: 31/12/2022 | Revisado: 26/01/2023 | Aceitado: 25/03/2023 | Publicado: 31/03/2023

Romeryto Vieira Lira

ORCID: <https://orcid.org/0000-0003-2567-0839>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil
E-mail: romeryto.lira@ifpb.edu.br

Ewerthon Dyego de Araújo Batista

ORCID: <https://orcid.org/0000-0003-4993-9900>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil
E-mail: ewerthon.batista@ifpb.edu.br

Livia Vitoria Do Nascimento Alves

ORCID: <https://orcid.org/0000-0002-0182-9027>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil
E-mail: livia.vitoria@academico.ifpb.edu.br

Alice dos Santos Araújo Lourenço

ORCID: <https://orcid.org/0000-0002-2769-0115>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil
E-mail: alice.araujo@academico.ifpb.edu.br

Gigliarly Marcelino Gonzaga

ORCID: <https://orcid.org/0000-0001-7771-0439>

Instituto Federal de Educação, Ciência e Tecnologia da Paraíba, Brasil
E-mail: gigliarly.marcelino@academico.ifpb.edu.br

Resumo

Introdução: A segurança de ambientes institucionais, que contenham bens de valor, muitas vezes é feita por sistemas de hardware e software proprietários de custo alto. Visando amenizar esse problema, o presente artigo propõe uma solução inicial baseada em desenvolvimento de software e utilização de hardware gratuitos, em sua maioria de código aberto, para que no futuro possa ser usada para proteger esses ambientes. Temos como objetivo propor um Protótipo Básico de Hardware e um Sistema Web utilizando tecnologias gratuitas, visando propor uma base de software e hardware para gerar segurança e controle de abertura de portas em ambientes. **Metodologia:** Neste projeto pesquisamos as ferramentas e tecnologias disponíveis para a criação do Protótipo e do Sistema Web. Foram definidas as ferramentas para desenvolvimento, criamos planos de trabalho para a modelagem, criação e desenvolvimento, e finalmente foi colocado em prática a criação da solução usando linguagem de programação. **Resultados:** Desenvolvemos um Protótipo Básico Hardware utilizando a plataforma Arduino juntamente com outros componentes acessórios de modo a ser possível a comunicação com o Software Web. Este foi inicialmente modelado utilizando diagrama de classes e posteriormente desenvolvido utilizando a framework Grails com linguagem de programação Groovy, além do MySQL para armazenamento dos dados. Este sistema contém serviços web desenvolvidos que podem ser acessados pelo protótipo. Com estes artefatos produzidos, pudemos criar uma solução básica que servirá de modelo inicial para implantação de um sistema de segurança de baixo custo e mais robusto no futuro.

Palavras-chave: Protótipo; Segurança; Hardware; Software; Programação.

Abstract

Introduction: The security of institutional environments, which contain valuable assets, is often done by expensive proprietary hardware and software systems. Aiming to alleviate this problem, this article proposes an initial solution based on software development and use of free hardware, mostly open source, so that in the future it can be used to

protect these environments. Our goal is to propose a Basic Hardware Prototype and a Web System using free technologies, aiming to propose a software and hardware base to generate security and door opening control in environments. Methodology: In this project, we researched the tools and technologies available for the creation of the Prototype and the Web System, the tools for development were defined, we created work plans for modeling, creation and development, and finally the work in the solution was put into practice using programming language. Results: We developed a Basic Hardware Prototype using the Arduino platform along with other accessory components in order to be able to communicate with the Web Software. This was initially modeled using a class diagram and later developed using the Grails framework with Groovy programming language, in addition to MySQL for data storage. This system contains developed web services that can be accessed by the prototype. With these artifacts produced, we were able to create a basic solution that will serve as an initial model for the implementation of a low-cost and more robust security system in the future.

Keywords: Prototype; Safety; Hardware; Software; Programming.

Resumen

Introducción: La seguridad de los ambientes institucionales, que contienen activos valiosos, a menudo se realiza mediante costosos sistemas de hardware y software propietarios. Con el objetivo de paliar este problema, este artículo propone una solución inicial basada en el desarrollo de software y el uso de hardware libre, en su mayoría de código abierto, para que en el futuro pueda ser utilizado para proteger estos entornos. Nuestro objetivo es proponer un Prototipo de Hardware Básico y un Sistema Web utilizando tecnologías libres, con el objetivo de proponer una base de software y hardware para generar seguridad y control de apertura de puertas en ambientes. Metodología: En este proyecto investigamos las herramientas y tecnologías disponibles para la creación del Prototipo y el Sistema Web, se definieron las herramientas para el desarrollo, creamos planes de trabajo para el modelado, creación y desarrollo y finalmente se puso en marcha la creación de la solución haciendo el uso del lenguaje de programación en la práctica. Resultados: Desarrollamos un Prototipo de Hardware Básico utilizando la plataforma Arduino junto con otros componentes accesorios para poder comunicarse con el Software Web. Este se modeló inicialmente utilizando un diagrama de clases y luego se desarrolló utilizando el framework Grails y el lenguaje de programación Groovy, además de MySQL para el almacenamiento de datos. Este sistema contiene servicios web desarrollados a los que puede acceder el prototipo. Con estos artefactos producidos, pudimos crear una solución básica que servirá como modelo inicial para la implementación de un sistema de seguridad más robusto y de bajo costo en el futuro.

Palabras clave: Prototipo; Seguridad; Hardware; Software; Programación.

1. Introdução

Atualmente, num contexto de mundo onde nós estamos nos prevenindo constantemente quanto a aspectos de segurança pessoal, física, de nossas informações, bens, ambientes, dos nossos dados informáticos existentes, tanto nos nossos dispositivos móveis quanto na internet e sua nuvem de dados, surgem desafios quanto às formas e metodologias para proteção e gerenciamento dos ativos de valor que utilizamos no nosso cotidiano, como bem falaram sobre o tema, Hutter (2019) e Fraga-Lamas et al. (2016).

Hoje vivemos grande parte do nosso dia em vários ambientes, principalmente nas escolas, universidades, trabalho, etc. onde em cada um desses lugares existem diversos ativos de valor, bens, móveis, dispositivos, computadores, e informações físicas ou digitais etc. que precisam ser protegidas.

Numa realidade onde a Segurança da Informação e a Segurança Física andam juntas, houve uma corrida para implementação e criação de sistemas de informação aliados a dispositivos de segurança física que promovessem proteção aos ambientes que abrigam bens valiosos, conforme abordado por Hutter (2019).

Também com o surgimento de dispositivos enquadrados no conceito de IOT (Internet das Coisas), conectados à rede mundial de computadores em 2006, as tecnologias ligadas à segurança física, interconectadas a softwares web, começaram a se integrar nessa nova classificação de itens tecnológicos, segundo Dachyar et al. (2019).

Nas empresas privadas e também do governo federal existem vários processos de licitações para adquirir dispositivos de proteção para vários órgãos, sejam para portas, janelas de ambientes que contém bens de valor, como laboratórios de

informática, escritórios, gabinetes etc. e também para o gerenciamento de quem entra e sai de cada local, determinando quem pode ou não transitar, de acordo com, do Brasil, (1993) e do Brasil, (2016).

Numa época onde a preocupação com a contenção de gastos e no melhoramento e a otimização dos gastos públicos toma conta dos debates públicos, Allas et al., (2018) ressalta a existência da preocupação em gastar de maneira precavida e ao mesmo tempo manter a qualidade dos dispositivos de controle de acesso e proteção de ambientes em vários setores públicos. Essa necessidade especialmente se nota nas Universidades e Institutos Federais, onde o bom uso e aplicação do dinheiro do povo é o objetivo para a oferta de educação, pública, gratuita e de qualidade, conforme citou Vasconcelos Lima e Raposo Pereira Feitosa (2014).

Soluções totalmente proprietárias para segurança física envolvendo hardware e software fechados, mesmo tendo uma boa eficácia para alguns problemas, são caras dependendo da empresa, já que no seu preço há embutido o lucro de vários atores na sua cadeia de produção, o que pode acarretar num gasto considerável para as instituições, como constatado por Sood et al., (2016). Também é custoso seguir à risca normas ISO como a 27001 e a 27002, ISO 27001. (2020) e ISO 27002. (2020).

Utilizar tanto *hardware* quanto *software livre* Thomsen (2020) e Software Livre. Projeto GNU. (2020) nas soluções computacionais, é uma tendência em vários governos. É tanto que o governo brasileiro intensificou o uso de software livre e de código aberto em vários setores da administração pública nos últimos anos, com o objetivo de economizar e evitar gastos desnecessários com tecnologias proprietárias.

Baseados nessa realidade, o projeto teve como objetivo criar um protótipo básico, baseado nos conceitos de IOT (Internet das coisas) para auxiliar na segurança física e no gerenciamento de acesso, que poderá no futuro ser instalado em porta(s) de laboratório(s) e/ou outros ambientes, utilizando o máximo de *hardware livre* (protótipo de hardware para segurança física), e uma solução de *software livre* web para se conectar ao hardware. A solução possibilitará uma base para o monitoramento desses dispositivos físicos para obtenção de informações relevantes como o estado de aberto ou fechado de uma porta e.g., ou em quais horários o ambiente foi utilizado, visando o menor custo possível, assim podendo ser instalado no âmbito institucional da administração pública, com o objetivo de proteger os ativos de valor.

Deste modo, neste projeto foi desenvolvido um Protótipo Físico Básico que serve de base para uma futura solução de monitoramento do estado de portas de ambientes que contenham objetos de valor, utilizando o máximo de hardware e software livre de baixo custo, e que seja conectado à internet. Também foi construído um Software Web que utilizou tecnologias livres, para receber e exibir os dados e solicitações do protótipo, visando assim, como solução conjunta, num futuro melhorar a segurança de ambientes com itens de baixo custo, podendo ser utilizado por instituições em benefício da proteção do patrimônio institucional.

2. Metodologia

O primeiro passo para execução deste projeto foi o *estudo e pesquisa de ferramentas e tecnologias disponíveis* para o desenvolvimento de soluções de segurança, priorizando o hardware e software livre, fazendo um apanhado do estado da arte das publicações sobre segurança física e informática, existentes na literatura, incluídos na introdução.

Depois foram *definidas as ferramentas de desenvolvimento*. Em seguida foi *criado um plano de trabalho* em duas frentes distintas, uma para o *desenvolvimento do Protótipo de Hardware* e outra para a *modelagem e o desenvolvimento do Software Web*. Nestas frentes, foram criados artefatos que possibilitaram tornar realidade nosso projeto. O trabalho no projeto seguiu muitos princípios da metodologia de gerenciamento de equipes SCRUM, Org, scrum. (2022).

A pesquisa e desenvolvimento foi feita por alunos do primeiro ano do curso técnico em Informática integrado ao ensino médio.

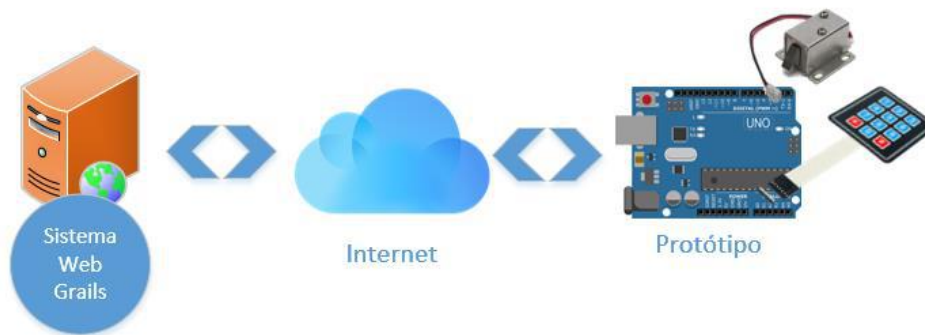
O projeto tem como primeiro objetivo criar um *Protótipo inicial de hardware* o qual possa verificar o status de abertura de uma porta, utilizando componentes de hardware livre. No nosso projeto utilizamos a plataforma Arduino, sensores e alguns componentes eletrônicos básicos. O protótipo deve poder se comunicar com serviços web e se conectar à Internet.

O segundo objetivo foi a criação de um Sistema web para cadastrar as informações do protótipo, e do local onde o mesmo poderá ser instalado e também poder receber chamadas do protótipo, oferecendo os serviços necessários para monitoramento de entrada e saída de um ambiente.

3. Resultados e Discussões

O projeto segue a arquitetura conforme o que está descrito na Figura 1:

Figura 1 - Arquitetura básica da solução.



Fonte: Autores (2022).

Conforme podemos visualizar na ilustração, a solução conta inicialmente com um Sistema Web desenvolvido no arcabouço de ferramentas de programação *Grails*, Grails framework. (2022), que utiliza a linguagem de programação *Groovy*, Groovy language. (2022), para deste modo poder armazenar em um banco de dados as informações recebidas do protótipo instalado na porta do ambiente. Este Sistema Web é acessado a partir da internet pelo Protótipo Básico de Hardware. Ele tem o objetivo de gerenciar a autenticação de usuários além de armazenar dados sobre esta solução física instalada, como *local*, *descrição do laboratório*, *estado da porta física do local*, e ainda deixamos aberta a possibilidade do armazenamento de informações para vários protótipos, visando uma expansão futura da solução. No desenvolvimento utilizamos a *IDE IntelliJ IDEA* IntelliJ IDEA. (2022), juntamente com o *plugin* de desenvolvimento para *Groovy* e *Grails*. A seguir apresentaremos as características das duas soluções, a do Sistema Web e a do sistema de hardware, Protótipo Básico.

3.1 Protótipo Básico

O protótipo foi feito utilizando algumas tecnologias, a seguir descrevemos cada uma delas e seu papel na estruturação do protótipo.

3.1.1 Arduino

O Arduino é uma plataforma eletrônica de hardware livre para prototipagem e criação de soluções de baixo custo, muito usado na automação residencial, segurança, agricultura, entre outros, segundo Zlatanov, Nikola. (2015). No projeto foi utilizado como base o Arduino Uno, que podemos visualizar na Figura 2, o qual é um pequeno chip/placa da família dos microcontroladores, onde é possível lidar com dados de entrada (que podem vir de sensores externos acoplados) processá-los e

gerar uma saída resultado, seja a partir dos *leds* da própria placa ou *leds* avulsos, assim como também o envio desses dados para outros sensores. No projeto, o Arduino concentrou a conexão da fechadura básica e dos sensores necessários para o desenvolvimento da solução.

Figura 2 - Arduino Uno.



Fonte: Arduino CC (2022).

3.1.2 Placa de rede com conexão RJ 45

Utilizamos uma placa de rede que tem a entrada para cabo com conector RJ45. O objetivo dela é fazer com que o Arduino possa se conectar à internet de maneira fácil e prática. A placa é conectada ao Arduino a partir de seus encaixes e conectores se tornando uma extensão do mesmo. Pode-se ver na Figura 3, que os encaixes da placa seguem o mesmo padrão possibilitando esta junção de maneira prática e fácil. Ao serem ligadas na energia e na rede, um endereço IP é atribuído para poder possibilitar que a comunicação seja via *intranet* ou *internet*.

Para a conexão, são feitas requisições aos serviços disponíveis no Sistema Web.

Figura 3 - Placa Ethernet com suporte a conexão RJ45.



Fonte: Arduino CC (2022).

3.1.3 Fechadura

Para nosso protótipo utilizamos uma fechadura eletromagnética básica de 12 volts que pode ser visualizada na Figura 4. A mesma tem dois fios, um para o polo positivo e outro para o negativo. Utilizamos uma fonte de roteador de 12v usada com a mesma voltagem para poder alimentar a fechadura, e controlamos a abertura e fechamento da mesma utilizando um relé básico de liga/desliga.

Figura 4 - Fechadura eletromagnética de 12V.



Fonte: Arduino CC (2022).

3.1.4 Teclado

Para nossa solução, foi utilizado um teclado simples para poder simular o envio dos dados de *login* e *senha* para testar o serviço de autenticação básica do sistema web. Na Figura 5, podemos visualizar este teclado que tem caracteres alfanuméricos e dois símbolos que podem ser digitados.

Figura 5 - Teclado.



Fonte: Arduino CC (2022).

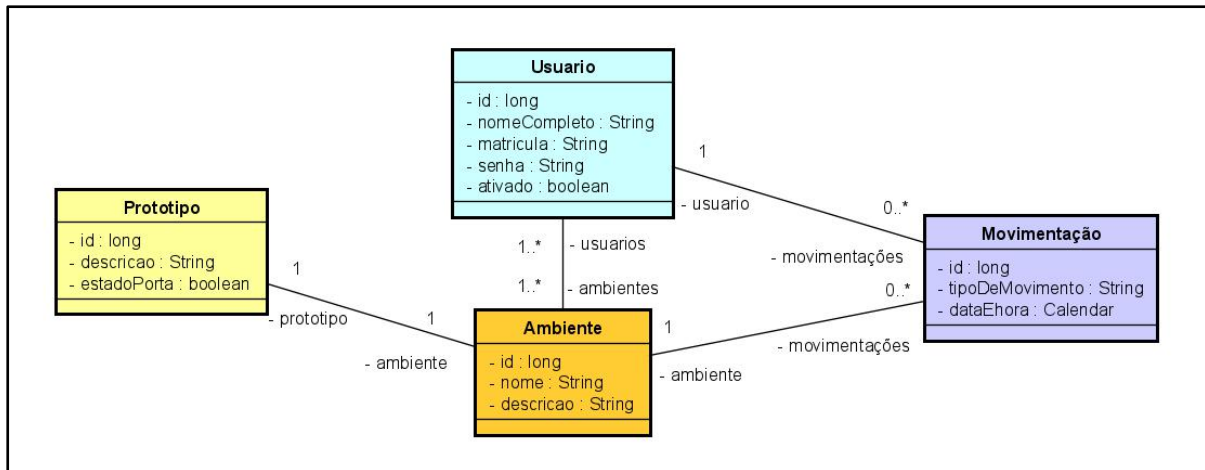
3.2 Sistema Web

O desenvolvimento foi feito em **Grails** utilizando a linguagem de programação Groovy. O código do programa para controle do protótipo inclui quatro classes (em programação, classe é uma maneira de definir o tipo de um dado em uma linguagem orientada a objetos): “*Usuario*”, “*Prototipo*”, “*Movimentacao*” e “*Ambiente*”.

3.2.1 Diagrama de Classes Básico e Sistema Web em Grails

Conforme podemos visualizar na Figura 6, de maneira simples, a classe *Usuario* é responsável por armazenar os dados de autenticação de *login* e *senha*. Já a classe *Protótipo*, é o ponto de partida inicial e representa logicamente o protótipo de hardware. Ela está associada à classe *Ambiente*, já que o protótipo básico de hardware quando estiver em produção, deve estar instalado em um ambiente físico, como um laboratório de informática por exemplo. Cada ambiente desses tem movimentações, de abertura e fechamento de porta, tal situação foi representada na classe *Movimentação* que deve registrar a data e hora de abertura e fechamento da porta feita por cada usuário. O suporte ao salvamento desta informação já foi desenvolvido em nossa solução inicial, porém o armazenamento destes dados ficou para a continuação que demos ao projeto em sua expansão.

Figura 6 - Diagrama de Classes.



Fonte: Autores (2022).

As classes foram implementadas em *Groovy* utilizando o Framework Grails, Grails framework. (2022), que permite a geração do Banco de dados e o básico da interface gráfica, a partir da descrição das classes. No Quadro 1 segue o exemplo da implementação da classe *Prototipo* com seus atributos.

Quadro 1 - Classe Protótipo.

	Prototipo.groovy
1	class Prototipo {
2	long id
3	String descricao
4	boolean estadoPorta
5	Ambiente ambiente
6	
7	static constraints = {
8	descricao blank: false, nullable: false
9	id blank: false, nullable: false
10	estadoPorta blank: false
11	ambiente nullable: true
12	}
13	}

Fonte: Autores (2022).

O Banco de Dados que utilizamos foi o *MySQL*. O esquema do mesmo é gerado automaticamente pelo Framework Grails, assim não é necessário escrever nenhuma linha de código *SQL*.

3.2.2 Serviços WEB disponíveis

Serviços Web de forma simples, são funcionalidades implementadas em linguagens de programação, utilizando métodos que ficam online na Internet de modo que outros sistemas de software e hardware possam se comunicar com os mesmos de modo a atender determinadas demandas do usuário, Rocher e Brown (2009). No Quadro 2 descrevemos de forma simples os *serviços web* disponíveis e suas funcionalidades.

Quadro 2 - Serviços Web disponíveis.

Serviço	Parâmetros	retorno	Descrição
<i>logon</i>	matrícula, senha	chave	Este serviço permite se autenticar no sistema utilizando a <i>matrícula</i> e <i>senha</i> . Em caso de sucesso, uma <i>chave</i> de acesso em forma de <i>String</i> é retornada, para ser utilizada nos outros serviços.
<i>estadoPorta</i>	código do protótipo, chave	verdadeiro ou falso	Este serviço permite saber o <i>estado</i> da porta, e é acessado passando o <i>código do protótipo</i> e a <i>chave de acesso</i> . Caso a porta esteja aberta, o estado retornado será <i>verdadeiro</i> , caso contrário, <i>falso</i> . Neste projeto trabalhamos apenas com um (1) protótipo.
<i>abrirFecharPorta</i>	código do protótipo, chave, estado	verdadeiro ou falso	Este serviço permite que o protótipo altere no sistema Web o <i>estado da porta</i> para <i>aberto</i> ou <i>fechado</i> modificando o parâmetro <i>estado</i> , refletindo assim neste sistema o estado do hardware. Este método também pode permitir o salvamento da <i>Movimentação</i> .

Fonte: Autores (2022).

No Quadro 3 demonstramos um exemplo de implementação do código da funcionalidade de autenticação no Sistema Web.

Quadro 3 - Método de autenticação de usuário e retorno de chave de autenticação ou falha.

	ArduinoController.groovy
1	<code>def logon() {</code>
2	<code>String usuariomat = params.matricula+""</code>
3	<code>String senha = params.senha+""</code>
4	<code>long codigoPrototipo = Long.parseLong(params.codprot+""</code>
5	<code>def String chaveDeAutenticacao = arduinoService.logon(usuariomat, senha, codigoPrototipo)</code>
6	<code>if(chaveDeAutenticacao!=null){</code>
7	<code>render contentType: "text/json", text: [{"chave":'+chaveDeAutenticacao+'}]'</code>
8	<code>}else{</code>
9	<code>render contentType: "text/json", text: [{"erro":'+FALHA+'}]'</code>
10	<code>}</code>
11	<code>}</code>
12	<code>}</code>

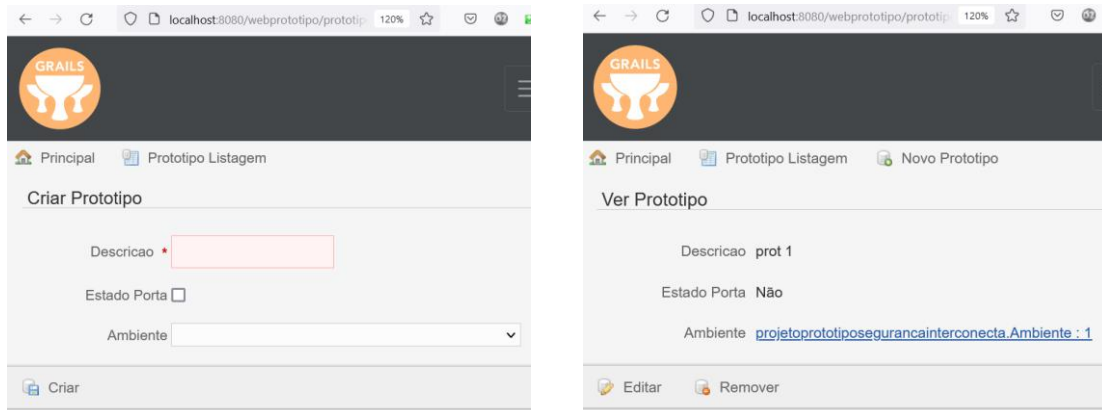
Fonte: Autores (2022).

Seguindo a mesma tecnologia, foram criados métodos para os dois outros serviços.

3.2.3 Interface Gráfica do Sistema Web

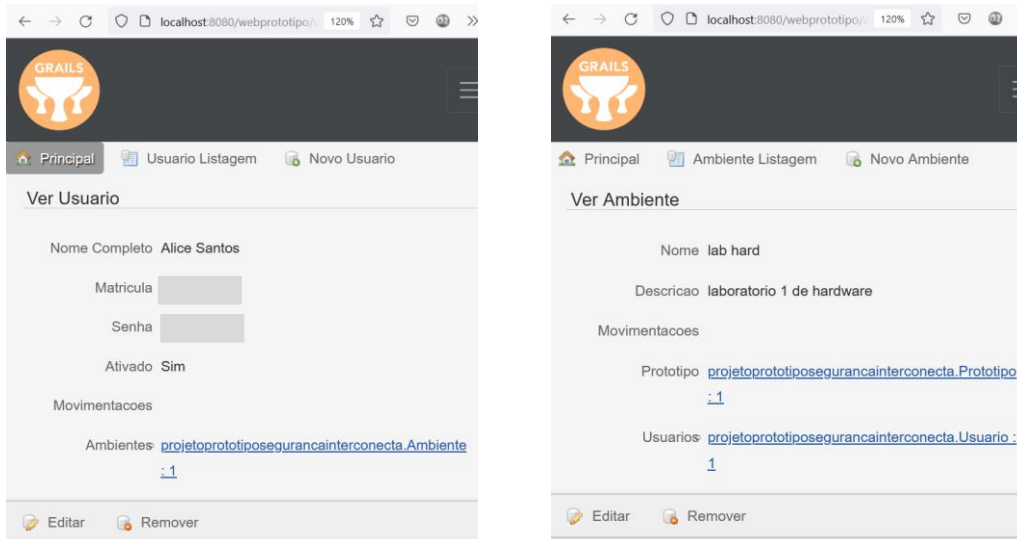
O Sistema Web contém uma interface gráfica para cadastro das informações do Protótipo, e dos objetos das demais classes, conforme podemos verificar na Figura 7 e na Figura 8. Para que o *Protótipo de Hardware* possa se comunicar com a parte Web, é necessário cadastrá-lo, juntamente com o ambiente e o usuário nas respectivas telas. Essas informações são salvas em um banco de dados. No nosso caso utilizamos MySQL. Cada classe do sistema tem telas de criação, edição, listagem e a funcionalidade de remoção de objetos cadastrados. A seguir mostraremos as imagens de algumas telas do Sistema.

Figura 7 - Criação e Listagem do Protótipo.



Fonte: Autores (2022).

Figura 8 - Visualização de Usuário e Ambiente.



Fonte: Autores (2022).

3.2.4 Comunicação do protótipo com o Sistema Web

Nesta etapa a comunicação foi feita a partir de requisições Web enviadas diretamente do Protótipo Básico em *Arduino*, passando pela *placa Ethernet RJ 45*, e finalmente sendo enviadas para o Sistema Web. No Quadro 4 e no Quadro 5, pode-se verificar os códigos desenvolvidos, para as funcionalidades principais.

Quadro 4 - Código de Abrir e fechar porta enviado ao Relé que controla a Fechadura.

	Trecho de Código no Arduino
1	<code>void abreOuFechaPorta(char comando){</code>
2	<code> if (Serial.available(>0){</code>
3	<code> comando = Serial.read();</code>
4	<code> if (comando == 'a') {</code>
5	<code> digitalWrite(2,LOW);</code>
6	<code> Serial.print("Porta aberta");</code>
7	<code> } else if(comando == 'f'){</code>
8	<code> digitalWrite(2,HIGH);</code>
9	<code> Serial.print("Porta fechada");</code>
10	<code> }</code>
11	<code> }</code>
12	<code>}</code>
13	<code>}</code>

Fonte: Autores (2022).

Quadro 5 - Código de chamada, a partir do Protótipo Básico, ao Sistema Web para fazer login.

	Trecho de Código de Chamada
1	<code>if (client.connect("192.168.1.5",8080) == 1){</code>
2	<code> Serial.println("Connected to server");</code>
3	<code> client.println("GET /webprototipo/arduino/logon?matricula=*****&senha=*****&codprot=1");</code>
4	<code> client.println("Host: 192.168.1.5:8080");</code>
5	<code> client.println();</code>
6	<code>}</code>
7	<code>else{</code>
8	<code> Serial.println("Connection to server failed");</code>
9	<code>}</code>

Fonte: Autores (2022).

Da mesma forma que o serviço *logon* é chamado, foram desenvolvidas as chamadas aos outros serviços. Assim foi possível a comunicação completa entre os dois mundos dessa solução básica inicial de nossa pesquisa, ou seja, o Protótipo Básico podendo se comunicar com o Sistema Web, possibilitando assim uma base para expansão do projeto para uma solução mais completa no futuro.

4. Considerações Finais

Neste projeto de pesquisa, os alunos tiveram a oportunidade de trabalhar com ferramentas novas com as quais nunca tiveram contato. Vale ressaltar que foi um grande desafio, para estudantes, do primeiro ano do ensino médio, ter contato com a programação. Eles tiveram todo o suporte dos professores orientadores assim como obtendo aprendizado de programação a partir das disciplinas presentes na grade curricular técnica. Eles tiveram uma boa facilidade de absorver a forma de trabalhar com a linguagem *Groovy*, devido a sua simplicidade.

Dessa maneira, utilizando tecnologias em sua maioria de código aberto e/ou gratuitas, foi possível criar uma solução inicial básica a partir do desenvolvimento de um Protótipo Básico de Hardware que consegue se comunicar com um Sistema Web utilizando serviços web, de modo que foi possível criarmos as bases de um sistema para proteção de ambientes institucionais que contém objetos e informações de valor, utilizando tecnologias de baixo custo, evitando soluções proprietárias de alto custo, Foxcrestsecurity (2017), de modo a diminuir ao máximo o uso de tecnologias proprietárias e caras.

Foi possível trabalhar com muitas tecnologias novas, tanto para programar um microcontrolador de hardware como o Arduino e seus acessórios conectados, de modo que os mesmos pudessem funcionar em conjunto. O mais interessante é que conseguiu-se fazer com que essa camada de hardware pudesse se comunicar com o Sistema Web, que também foi criado pela equipe de alunos e professores. Este último foi projetado e modelado utilizando técnicas de Engenharia de Software, a partir da criação de um diagrama de classes, para posteriormente a partir desse modelo, codificar as classes em *Groovy* para cada entidade prevista no diagrama de classes, respeitando seus atributos.

Sobre as limitações deste estudo, como é uma proposta mais básica, focamos no funcionamento apenas de um protótipo simples, e necessitamos, todavia, melhorar e refatorar os códigos da parte do hardware assim como prover uma melhor proteção para os dados que estão sendo enviados em rede.

Para os trabalhos futuros, pode-se melhorar todo o código de Protótipo de hardware, oferecer suporte ao monitoramento de mais de um protótipo a partir do Sistema Web, promover a montagem de uma pequena estrutura para acomodar os componentes físicos e possivelmente criar um aplicativo para dispositivos móveis para visualização dos protótipos instalados nos ambientes para ver se estão ativos ou inativos, de modo oferecer um panorama do monitoramento de vários locais simultaneamente.

Agradecimentos

O desenvolvimento deste artigo contou com o auxílio de algumas pessoas, dentre as quais agradecemos: ao IFPB (Instituto Federal de Ciência e Tecnologia da Paraíba) por todo o apoio financeiro destinado à pesquisa; aos nossos professores, coordenador e orientador, que nos acompanharam, incentivaram no desenvolvimento deste projeto de pesquisa.

Referências

- Allas, T., Dillon, R., & Gupta, V. (2018). A smarter approach to cost reduction in the public sector. *McKinsey & Company*. <https://www.mckinsey.com/industries/public-sector/our-insights/a-smarter-approach-to-cost-reduction-in-the-public-sector>
- Arduino CC (2022). Acessado em 15 de janeiro de 2023., <https://www.arduino.cc>
- Dachyar, M., Zagloel, T. Y. M., & Saragih, L. R. (2019). Knowledge growth and development: Internet of things (IoT) research, 2006–2018. *Heliyon*, 5(8), e02264. <https://doi.org/10.1016/j.heliyon.2019.e02264>
- do Brasil, G. (1993). Lei 8666. *DECRETO Nº 3.847, DE 21 DE JUNHO DE 1993 Lei de Licitações*. http://www.planalto.gov.br/ccivil_03/leis/18666cons.htm
- do Brasil, G. (2016). *DECRETO nº 8.638, de 15 de Janeiro de 2016. Institui a Política de Governança Digital no âmbito dos órgãos e das entidades da administração pública federal*. http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2016/decreto/d8638.htm
- Foxcrestsecurity (2017). *Proprietary Security System Equipment - Love it or Hate It?* FoxCrest Security. Acessado em 15 de janeiro de 2023. <https://foxcrestsecurity.com/proprietary-security-system>
- Fraga-Lamas, P., Fernández-Caramés, T., Suárez-Albela, M., Castedo, L., & González-López, M. (2016). A review on internet of things for defense and public safety. *Sensors*, 16(10), 1644. <https://doi.org/10.3390/s16101644>
- Grails framework. (2022). Acessado em 10 de dezembro de, 2022. <https://grails.org>
- Groovy language. (2022). The Apache Groovy programming language. Acessado em 10 de dezembro de, 2022. <https://groovy-lang.org>
- Groovy. (2007). In *Groovy Programming* (pp. 341–348). Elsevier. <http://dx.doi.org/10.1016/b978-012372507-3/50029-0>
- Hutter, D. (2019). Information Security Reading Room Physical Security and Why It. <https://www.sans.org/reading-room/whitepapers/physical/paper/37120>
- IntelliJ IDEA. (2022). An IDE called IntelliJ IDEA by JetBrains. <https://www.jetbrains.com/idea>.
- ISO 27001. (2020). Acessado em 07 de março de, 2020, <https://www.27001.pt>
- ISO 27002. (2020). Acessado em 07 de março de, 2020: https://en.wikipedia.org/wiki/ISO/IEC_27002
- ORG, scrum. (2022). *What is Scrum?* Scrum.Org. Acessado em 07 de março de, 2020 . <https://www.scrum.org/resources/what-is-scrum>
- Rocher, G., & Brown, J. (2009). Web services. In *The Definitive Guide to Grails* (pp. 449–485). Apress. http://dx.doi.org/10.1007/978-1-4302-0871-6_15

Software Livre. Projeto GNU. (2020). O que é o software livre? Free Software Foundation. <https://www.gnu.org/philosophy/free-sw.pt-br.html>

Sood, G., Shipra, & Soni, R. (2016). Comparative Study: Proprietary Software vs. Open Source Software. *International Journal of Innovative Research in Computer and Communication Engineering*, 4.

Thomsen, A. (2020). Open Hardware. Vamos falar de Open Hardware? *FilipeFlop*. Acessado em 07 de março de, 2020. <https://www.filipeflop.com/blog/open-hardware-livre>

Vasconcelos Lima, G., & Raposo Pereira Feitosa, G. (2014). Políticas públicas para o software livre, práticas colaborativas e os princípios da administração pública. *Direito e Novas Tecnologias Ii: Xxiii Congresso Nacional Do Conpedi.*, 23(1), 379–393.

Zlatanov, N. (2015). Arduino and Open Source Computer Hardware and Software. 10.13140/RG.2.1.1071.7849.