

CENTRO UNIVERSITÁRIO UNIFACVEST
CURSO CIÊNCIA DA COMPUTAÇÃO
ISAQUE MASCARELLO

SAFEBOX IoT

**LAGES
2021**

ISAQUE MASCARELLO

SAFEBOX IoT

Trabalho de Conclusão do Curso de Ciência da Computação, submetido ao Centro Universitário UNIFACVEST, em cumprimento aos requisitos para obtenção do título de bacharelado em Ciência da Computação.

Orientador: Prof. Me. Márcio José Sembay

Coorientador: Prof. Jean Carlos Macedo

Coorientador: Prof. Willen Leolatto Carneiro

**LAGES
2021**

ISAQUE MASCARELLO

SAFEBOX IoT

Trabalho de Conclusão do Curso de Ciência da Computação, submetido ao Centro Universitário UNIFACVEST, em cumprimento aos requisitos para obtenção do título de bacharelado em Ciência da Computação.

Orientador: Prof. Me. Márcio José Sembay

Coorientador: Prof. Jean Carlos Macedo

Coorientador: Prof. Willen Leolatto Carneiro

Lages, SC ____ / ____ / ____ . Nota _____

Prof. Me. Márcio José Sembay
Coordenador do Curso

LAGES
2021

SafeBox IoT - SISTEMA DE GERENCIAMENTO DE REPOSITÓRIOS COMPARTILHADOS

RESUMO

O presente trabalho possui o objetivo de apresentar o desenvolvimento do projeto intitulado como SafeBox IoT, que por sua vez, oferece uma solução tecnológica para gerenciamento *on-line* de acesso remoto a repositórios compartilhados; com foco voltado principalmente para o mercado corporativo de segurança de valores e logístico. Este visa proporcionar dinamicidade, proteção remota aos bens compartilhados e registrar relatórios informativos de utilização dos mesmos. Projeto este composto de um protótipo *IoT* e plataforma de gerenciamento *on-line*. No desenvolvimento, utiliza-se materiais bibliográficos de acordo com as naturezas de pesquisa exploratória e descritiva conforme selecionadas previamente.

Palavras-chave: internet das coisas, desenvolvimento web, repositórios conectados.

SafeBox IoT - SISTEMA DE GERENCIAMENTO DE REPOSITÓRIOS COMPARTILHADOS

ABSTRACT

This work aims to present the development of the project entitled SafeBox IoT. Which, in turn, offers a technological solution for online management of remote access to shared repositories; focusing mainly on the corporate security and logistics market. It aims to provide dynamic, remote protection to shared assets and record informative reports on their use. This project consists of an IoT prototype and online management platform. In development, bibliographic materials are used according to the nature of exploratory and descriptive research as previously selected.

Keywords: internet of things, web development, connected repositories.

ILUSTRAÇÕES

Ilustração 1: Framework Visão Global do Sistema	21
Ilustração 2: Diagrama de Estrutura e Relacionamento de Dados (schema)	21
Ilustração 3: Diagrama de Estrutura e Relacionamento de Dados (data)	22
Ilustração 4: Diagrama de Arquitetura do Microcontrolador	23
Ilustração 5: Diagrama de Pinagem do Microcontrolador	23
Ilustração 6: Representação da aba de cadastro (client-side)	26
Ilustração 7: Representação da página de login (client-side)	26
Ilustração 8: Botão para o formulário de inclusão de dispositivos	27
Ilustração 9: Representação do formulário de inclusão de dispositivos	27

LISTA DE SIGLAS

CGNAT - *Carrier Grade Network Address Translation*

CSS3 - *Cascading Style Sheets Version 3*

DDoS - *Distributed Denial of Service*

GB - *Gigabyte*

GHz - *Gigahertz*

HTML5 - *HyperText Markup Language Version 5*

HTTP - *HyperText Transfer Protocol*

HTTPS - *HyperText Transfer Protocol Secure*

I2C - *Inter-Integrated Circuit*

IoT - *Internet of Things*

IPv6 - *Internet Protocol Version 6*

JSON - *JavaScript Object Notation*

LGPD - *Lei Geral de Proteção de Dados Pessoais (nº 13.709/2018)*

MB - *Megabyte*

MHz - *Megahertz*

NoSQL - *Not Only Structured Query Language*

RAM - *Random Access Memory*

SSL - *Secure Sockets Layer*

UI - *User Interface*

UX - *User Experience*

W3C - *World Wide Web Consortium*

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Justificativa	10
1.2 Importância	11
2 OBJETIVOS	12
2.1 Objetivo Geral	12
2.2 Objetivos Específicos	12
3 METODOLOGIA	13
3.1 Tipo de Pesquisa	13
3.2 Método de Pesquisa	13
3.3 Técnica de Análise de Dados	13
4 REVISÃO DE LITERATURA	14
4.1 Tecnologias Utilizadas	14
4.2 IoT (Internet das Coisas)	14
4.3 Segurança da Informação	15
4.4 Estrutura de Website	16
4.5 Normas	16
5 CRONOGRAMA	17
6 PROJETO	18
6.1 Requisitos de Hardware	18
6.1.1 Servidor	18
6.1.2 Cliente	18
6.1.3 Protótipo	18
6.2 Requisitos de Software	19
6.2.1 Servidor	19
6.2.2 Cliente	20
6.2.3 Protótipo	20

6.3 Modelagem e Diagramas	20
6.3.1 Conceito Geral	20
6.3.2 Estrutura e Relacionamento de Dados	21
6.3.3 Protótipo	22
6.4 Desenvolvimento	24
7 SISTEMA	25
7.1 Software Embarcado	25
7.1.1 Configurar Rede Wi-Fi (SSID e Senha)	25
7.1.2 Inserir Servidor de Controle	25
7.1.3 Inserir Chave Criptográfica	25
7.2 Softwares de Gerenciamento do Usuário - Front-end (Web Page e App)	25
7.2.1 Aba de Cadastro	25
7.2.2 Aba de Login	26
7.2.3 Inclusão de Dispositivos	27
8 CONSIDERAÇÕES FINAIS	28
REFERÊNCIAS	29

1 INTRODUÇÃO

A tecnologia está mudando rapidamente a maneira como interagimos com o mundo à nossa volta. Em relação aos dispositivos conectados na internet, ou internet das coisas como ficou popularizado, já tornou-se realidade do dia a dia em diversas formas; integradas em assistentes virtuais, televisores, sistemas de vigilância residencial, relógios (se é que podem ser chamados de relógios) como tantos outros. Estes dispositivos estão ganhando cada vez mais espaço à medida que evoluímos, em múltiplos objetos que nos rodeiam diariamente.

A internet das coisas de acordo com Magrani (2018) “pode ser entendido como um ambiente de objetos físicos interconectados com a internet por meio de sensores pequenos e embutidos, criando um ecossistema de computação onipresente (ubíqua), voltado para a facilitação do cotidiano das pessoas, introduzindo soluções funcionais nos processos do dia a dia”. Obviamente que em determinadas ocasiões haverá impedimentos em suas aplicações de forma plena e satisfatória. Casos estes como falta sinal, baterias e até recentemente alguns dispositivos sofrem problemas relacionados a portas de comunicações, quando estes trabalham conectados a CGNATs¹ de provedores de acesso (alguns casos contornáveis com a utilização do protocolo IPv6²). Mesmo com eventuais problemas de implementação ou utilização, é inegável a contribuição que a maioria destas soluções nos proporcionam.

Com toda esta expansão massiva de dispositivos conectados nos últimos anos, somado os recursos e facilidades agregados por estes e outras revoluções tecnológicas que vêm acontecendo, acredito haver outras áreas conservadoras e tradicionais, onde não foram exploradas, ou cogitadas a inclusão de recursos inovadores e facilitadores que a internet e os sistemas embarcados possibilitam para seus usuários.

1.1 Justificativa

Acredito que a necessidade é dinâmica e pode ser criada, ou seja, muito do essencial hoje em dia, há algumas décadas atrás não imaginávamos que existiriam. Assim sendo, no presente trabalho proponho uma solução (sistema) para auxiliar em repositórios de armazenamentos compartilhados.

1. *Carrier Grade Network Address Translation* é uma estrutura intermediária entre sua rede doméstica e a internet. Ou seja, várias redes domésticas acabam compartilhando um mesmo endereço público.
2. Versão 6 do protocolo de endereçamentos da internet.

Valendo-se do poder e das possibilidades que a internet das coisas nos permitem, somado ao conhecimento compartilhado pelos mestres e mentores neste curso de bacharelado em ciência da computação, tenho plena convicção de que o projeto apresentado neste trabalho de conclusão de curso, agregará experiência e conhecimentos extras no meu histórico acadêmico.

1.2 Importância

Havendo coerência nas inovações sempre é importante valorizá-las, principalmente quando esta apresenta um potencial revolucionário ou um legado para futuras criações, onde a mesma agregue benefícios, correções e recursos adicionais, qual facilite ou entregue mais opções aos usuários, melhorando a experiência proporcionada pelos produtos e serviços aos seus utilizadores.

Observa-se como exemplo esta pandemia da COVID-19³ que estende-se desde 2020 até o presente momento (2021), onde os indivíduos foram compelidos a uma série de restrições necessárias para prevenção a sua própria saúde e de terceiros, mudando radicalmente a forma com que exercem suas atividades rotineiras e compromissos. Tendo na tecnologia e inovação uma saída importantíssima para manutenção de seus afazeres, em alguns casos vêm com melhorias significativas, indicando serem mudanças definitivas e não apenas paliativas como de princípio, ou na pior das hipóteses deixando seu legado.

Além disso, a evolução humana depende de criações e inovações constantes, seja ela suprimindo necessidades ou disponibilizando novas opções.

3. Coronavírus 2019, ano em que a doença foi identificada pela primeira vez. Trata-se de uma doença infecciosa respiratória, causada pelo coronavírus SARS-CoV-2.

2 OBJETIVOS

2.1 Objetivo Geral

O presente trabalho possui como finalidade principal, apresentar um protótipo funcional que auxilie de maneira remota, o controle de acesso para repositórios de armazenamentos compartilhados.

Utilizarei como base para este projeto os cofres corporativos, de maneira a exemplificar o sistema proposto. Mas, o mesmo poderá ser implementado de diversas outras maneiras; como guarda-volumes, armazéns de depósitos dentre outros.

2.2 Objetivos Específicos

O SafeBox IoT é um sistema que apresenta-se via página web e aplicativo, permitindo aos usuários por meio destes, controlar remotamente o dispositivo protótipo.

Detalhes específicos:

- Proporcionar aval remoto e controle centralizado de dispositivos;
- Disponibilizar aplicação com plataforma de gerenciamento ao usuário;
- Servidor web para controle de requisições aos dispositivos;
- Dispositivo protótipo.

3 METODOLOGIA

3.1 Tipo de Pesquisa

Como natureza da pesquisa os tipos de pesquisa exploratória e descritiva de maneira aplicada, relaciona-se de melhor maneira com o projeto apresentado. Por este motivo, ficou eleito ambos como natureza de pesquisa do presente trabalho.

O tipo de pesquisa exploratória segundo Gil (2007), “tem a finalidade de ampliar o conhecimento a respeito de um determinado fenômeno. Segundo o autor, esse tipo de pesquisa, aparentemente simples, explora a realidade buscando maior conhecimento, para depois planejar uma pesquisa descritiva”.

3.2 Método de Pesquisa

O método de pesquisa adotado será documental e bibliográfico. Onde utiliza-se de fontes de dados primários e secundários, permitindo ao pesquisador uma cobertura mais ampla, e facilitando quando o problema de pesquisa requer dados dispersos.

3.3 Técnica de Análise de Dados

A técnica de análise de dados será qualitativa. De acordo com Gil (2007), os estudos qualitativos têm como preocupação básica o mundo empírico, tornando a pesquisa qualitativa de forma descritiva, pois se preocupa em descrever os fenômenos por meio dos significados que o ambiente manifesta.

4 REVISÃO DE LITERATURA

4.1 Tecnologias Utilizadas

O mundo do desenvolvimento *web* está em evidência atualmente. São inúmeras as tecnologias que estão na órbita da *web*, o que torna difícil, hoje em dia, encontrar alguma solução que não faça parte dela.

Segundo a empresa Akamai (2020), plataforma de armazenamento em nuvem responsável por 30% do tráfego online mundial, registrou um aumento de 112% no uso de internet no Brasil durante a pandemia da COVID-19. A empresa ressalta que esta alta foi causada devido mais pessoas passaram a usar a internet para trabalhar, estudar, fazer compras e se divertir.

Obviamente não seria diferente com este projeto. As tecnologias, normas e protocolos que servirão de suporte para o desenvolvimento do protótipo funcional a que este Trabalho de Conclusão de Curso baseia-se, são voltados para soluções *web*. Entre eles o HTML5, CSS3, JavaScript (*client-side*⁴ e *server-side*⁵), banco de dados NoSQL, criptografia dos dados sensíveis e protocolos de comunicação HTTP ou HTTPS.

4.2 IoT (Internet das Coisas)

Na atual realidade, se analisarmos o quanto estamos imersos num mundo digital e conectado. Servidos de assistentes virtuais, produtos *smart*⁶ e embarcados que anos atrás não imaginávamos que existiriam; ou seja, é inevitável que a internet, e em especial os produtos IoT (*Internet of Things*), nos auxiliam de maneira mais inteligente e até mesmo personalizado em relação aos dispositivos anteriores.

Talvez a mais impactante e pervasiva dessas tecnologias digitais seja a internet das coisas, objeto de atenção prioritária de governos e da iniciativa privada pelo mundo inteiro. Com certeza, o leitor já deve ter se deparado com o exemplo batido da “geladeira do futuro”, que vai detectar que o leite acabou e o incluir como item na lista de compras da semana. Bem, esqueça isso. Internet das coisas é muito mais que uma geladeira conectada. É a progressiva automatização de setores inteiros da economia e da vida social com base na comunicação máquina-máquina: logística, agricultura, transporte de pessoas, saúde, produção industrial e muitos outros. [...] (MARTINHÃO, 2018, p.15)

4. Parte da aplicação onde é executada no dispositivo do cliente, também conhecida como *front-end*.

5. Aplicação do servidor, onde é executado um *web service*, também conhecido como *back-end*.
6. Produto ou serviço inteligente.

Segundo Doneda (2018), sugere-se que a internet das coisas além de poderosa e já presente em nosso cotidiano até mesmo de forma oculta “[...] podendo estar mesmo em apetrechos que a princípio pareçam insuspeitos de qualquer sofisticação tecnológica”, está em franca expansão em muitos setores. Inclusive, o mesmo relata sobre haver debate e interesse do governo federal em regulamentar o IoT no país.

Não há dúvidas de que a internet das coisas está espalhando-se rapidamente nos mais diversos segmentos, e que naturalmente possui mais áreas para serem exploradas. Porém, devemos atentar para sua implementação de forma segura e coerente.

4.3 Segurança da Informação

Devido ao crescente número de informações geradas diariamente e em constante acúmulo das mesmas na nuvem, faz-se obrigatório a preocupação e questionamentos sobre segurança de dados. Não seria diferente com dispositivos IoT, onde já houve relatos de ataques massivos utilizando-se da vulnerabilidade destes.

Um dos ataques mais famosos e provavelmente dos mais importantes até o momento envolvendo dispositivos IoT, ocorreu em 2016 e ficou conhecido como *botnet*⁷ Mirai. O ataque foi contra diversos *sites* e provedores dos EUA e Europa, onde os mesmos acabaram ficando temporariamente fora do ar. Os dispositivos infectados pelo *malware*⁸ foram direcionados para efetuar requisições em massa aos servidores alvos. Os ataques deste tipo são conhecidos pelo acrônimo *DDoS*⁹ (*Distributed Denial of Service*).

Segundo publicação da Kaspersky Lab, empresa global de cibersegurança, o Brasil é o segundo país com mais ataques a dispositivos IoT no mundo, ficando atrás apenas da China no período de 2019.

[...] ocorreram 105 milhões de ataques contra dispositivos IoT vindos de 276 mil endereços exclusivos IP nos primeiros seis meses do ano em todo o mundo. Esse número é cerca de nove vezes maior do que o do primeiro semestre de 2018, período em que foram identificados cerca de 12 milhões de ataques originados em 69.000 endereços IP. O levantamento, que foi feito com ajuda de honeypots da empresa (redes de dispositivos e aplicativos virtuais conectados à internet), mostrou ainda os países mais atacados e o Brasil concentra 19% deles. Essas conclusões fazem parte do relatório ‘IoT: uma história de malware’. (KASPERSKY, 2019)

7. Referência a robôs em rede, ou seja, dispositivos conectados à *internet* executando tarefas combinadas.
8. Programa malicioso, destinado a infiltrar-se em um sistema alheio de forma ilícita, causando danos.
9. Ataque de negação de serviço, este tenta tornar um sistema indisponível a seus utilizadores.

Neste tema de segurança, somado a criticidade do projeto aqui apresentado. Adotou-se como método de criptografia para senhas e autenticações de usuários o BCrypt. A opção por este método ocorreu devido seu desempenho notável dentre outros métodos atuais. Criado por Niels Provos e David Mazières, o BCrypt foi apresentado na conferência da USENIX em 1999, este método de criptografia utiliza *hash* baseado em Blowfish.

4.4 Estrutura de Website

A estrutura da página *web client-side*, naturalmente será construída utilizando o que há de atual e trivial no momento; ou seja, o famigerado HTML5 e CSS3, também contando com funcionalidades de interatividade desenvolvidas baseada em JavaScript.

Visando um possível aplicativo *WebView*¹⁰, o mesmo será desde o primeiro momento confeccionado de maneira responsiva. Deste modo poderá adequar-se a uma gama variada de tamanhos e proporções de telas; tanto para computadores e *laptops*, quanto para celulares, tablets dentre outros.

4.5 Normas

Como as boas práticas no mundo da programação e desenvolvimento de aplicações são requisitos essenciais, nada mais justo que aplicá-las em todas as etapas deste projeto. Sendo assim, enfatizo o cumprimento das normas e padrões proposto pela W3C (*World Wide Web Consortium*), para maior compatibilidade, estabilidade e fluidez no uso da aplicação *web* e a LGPD (Lei Geral de Proteção de Dados Pessoais, nº 13.709, de 14 de agosto de 2018), no quesito de sigilo e privacidade das informações.

10. Trata-se de um componente em sistemas Android, que funciona de maneira similar a um navegador, porém, permitindo portar uma página web de maneira dedicada.

5 CRONOGRAMA

O desenvolvimento e estudos relacionados ao projeto proposto no presente trabalho, evoluem de acordo com o cronograma a seguir.

Quadro 1: Cronograma

Atividade / Prazo	Jul	Ago	Set	Out	Nov	Dez
Revisão de literatura (complementar)						
Especificação e ajustes do projeto						
Desenvolvimento dos sistemas <i>client-side</i> e <i>server-side</i>						
Desenvolvimento do protótipo funcional						
Testes do sistema						
Entrega do TCC						
Defesa do TCC						

Fonte: Elaboração do Autor (2021)

6 PROJETO

6.1 Requisitos de Hardware

Devido ser um sistema distribuído, no quesito *hardware* este necessita ser dividido em três partes. Sendo elas; servidor, onde hospeda a aplicação *web* que servirá como uma ferramenta de gestão dos dispositivos; cliente, onde o usuário executará os comandos necessários para consultas e interações com o(s) dispositivo(s); por fim, o próprio dispositivo (protótipo), qual enviará informações e receberá comandos de seus usuários através do servidor.

6.1.1 Servidor

Para teste local de funcionamento da aplicação, suportando até cinco usuários (clientes) conectados. Uma máquina com capacidade mínima de processamento com 2 núcleos de 2 GHz e cache de 2 MB, memória RAM de 4 GB e 5 GB de armazenamento em disco disponível (dedicados no momento do teste apenas à aplicação e suas dependências). Poderá obter resultados satisfatórios, para efeitos de testes e conhecimento do sistema.

Porém, em uma aplicação real sugere-se a utilização de um serviço especializado de hospedagem e computação em nuvem, onde tenha a possibilidade da alteração de capacidade alocada ao sistema, de acordo com a quantidade de usuários e a base de dados.

6.1.2 Cliente

Para o *hardware* do cliente, independentemente de teste ou utilização real não haverá alteração nos requisitos, sendo necessário apenas um dispositivo capaz de executar um navegador com suporte a HTML5. Os mais indicados são; computadores, *tablets* e *smartphones*, além obviamente de uma conexão estável com a internet.

6.1.3 Protótipo

O protótipo construído para exemplificação do conceito do presente projeto, utiliza como base um microcontrolador desenvolvido pela Espressif Systems Co. Ltd., da série ESP32, linha WROOM, embarcado em um kit de desenvolvimento DEVKIT-V1. O mesmo

conta com um processador Xtensa® LX6 Dual-Core 32-bit de 240 MHz, memórias *flash* de 4 MB e RAM 520 KBytes.

O protótipo também utiliza um leitor biométrico modelo FPM10A DY50, um teclado matricial de membrana 4x3, um micro servo Tower Pro SG90, um *display* de *oled* I2C com resolução 128x64 *pixels*, além de periféricos eletrônicos como resistores, diodos e transistores.

6.2 Requisitos de Software

Igualmente como os requisitos de *hardware*, os requisitos de *software* também ficaram divididos da mesma forma. Para melhor detalhamento das exigências do sistema, abaixo estão separadamente especificados.

6.2.1 Servidor

Pensando em flexibilizar a escolha do gestor do sistema quanto aos sistemas operacionais para o servidor de aplicação, desde o início do desenvolvimento do projeto, optou-se por utilizar e basear-se em ferramentas que proporcionam ao sistema a possibilidade de ser executado em múltiplas plataformas, além da preferência por ferramentas de código aberto.

O sistema *server-side* foi desenvolvido em JavaScript utilizando como interpretador o Node.JS. Ferramenta esta, que possui compatibilidade com os mais populares sistemas operacionais para servidores da atualidade.

Já o banco de dados, a escolha foi por um orientado a documentos, devido aos poucos relacionamentos de dados necessários para o funcionamento do sistema, onde nesta situação possui desempenho superior aos relacionais. A seleção foi na mesma linha da implementação anterior, com uma ferramenta desenvolvida para múltiplas plataformas e código aberto. O escolhido foi o MongoDB, um banco de dados NoSQL com um desempenho notável e muito popular atualmente, o que facilita na questão de suporte e manutenções.

Quanto a segurança das informações trafegadas entre cliente e servidor. O projeto conta com um certificado SSL, instalado na ferramenta de *proxy* reverso NGINX. Esta ferramenta facilita também na hipótese de escalabilidade do sistema, fornecendo gerenciamento de servidores de aplicações redundantes, permitindo o balanceamento de carga e evitando eventuais gargalos no sistema.

Não menos importante; parte dos requisitos de *software* são as bibliotecas de dependências do Node.JS. Sendo, o Express para os serviços HTTP, o EJS para renderização, Mongoose para manipulação do banco de dados, Socket.IO para comunicação auxiliar e o BCrypt para a criptografia das senhas armazenadas no banco de dados.

6.2.2 Cliente

Os requerimentos de *software* para o cliente, são os mais básicos possíveis; muito provavelmente já estejam disponíveis antes da adesão ao sistema. A exigência é um navegador com suporte a HTML5, preferível que seja um Opera ou Chrome atualizado. Havendo também a possibilidade de uma aplicação WebView, portando esta página de maneira mais agradável e fluida para o usuário.

6.2.3 Protótipo

Por sua vez, o dispositivo do ponto de vista do cliente, repete-se as exigências, sendo um navegador para efetuar a pré-configuração do *web service*; por exemplo, selecionar o endereço do servidor a qual ele se comunicará e inserir a chave de segurança.

Já do ponto de vista do desenvolvimento, faz se necessário o *framework* ESP-IDF ou Arduino IDE, previamente configurado para reconhecer o microcontrolador ESP32, como requisito para efetuar o *upload* do *sketch* para o dispositivo.

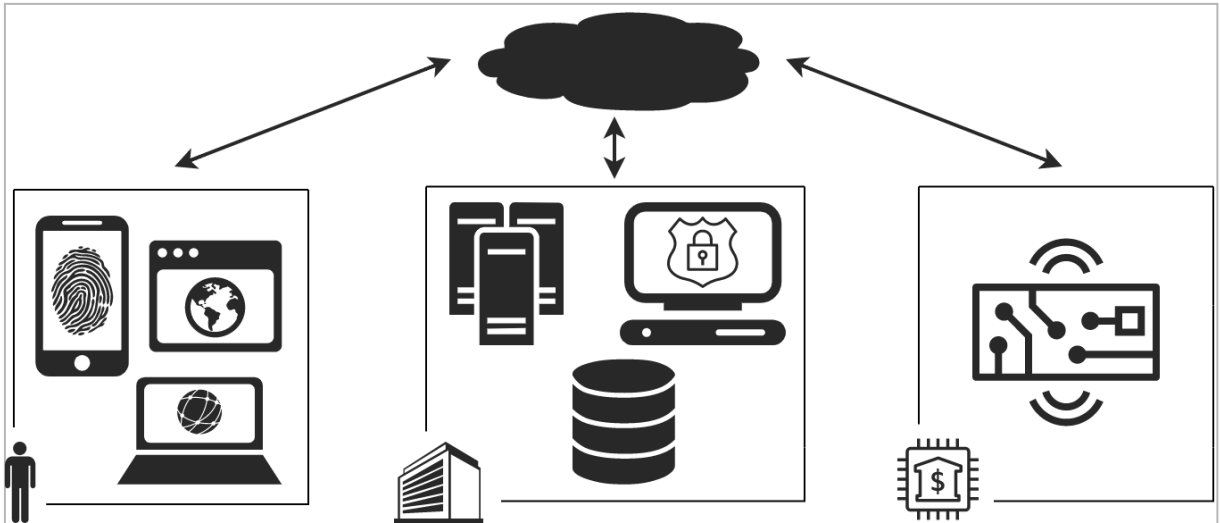
6.3 Modelagem e Diagramas

Em forma de facilitar a ilustração do conceito deste projeto, adotou-se a representação por meio de *frameworks* e diagramas, a qual apresenta desde uma visão global e genérica do sistema, até partes específicas e estruturais do mesmo.

6.3.1 Conceito Geral

A primeira imagem é um *framework*, onde auxilia na ilustração do sistema como um todo; além de exibir como os componentes do projeto relacionam-se, exemplificando a lógica de funcionamento do sistema proposto.

Ilustração 1: Framework Visão Global do Sistema

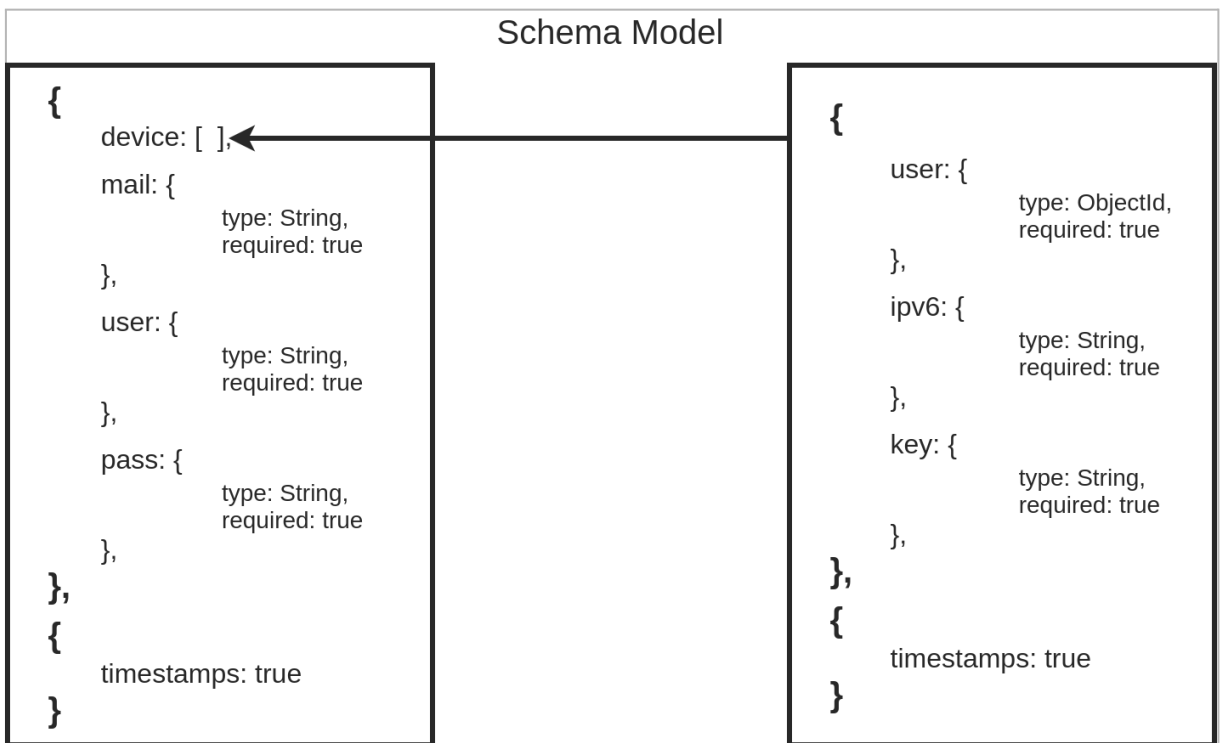


Fonte: Elaboração do Autor (2021)

6.3.2 Estrutura e Relacionamento de Dados

O diagrama abaixo foi elaborado com o propósito de ilustrar o esquema implementado, como estrutura e relacionamento de dados. Observação: este é NoSQL.

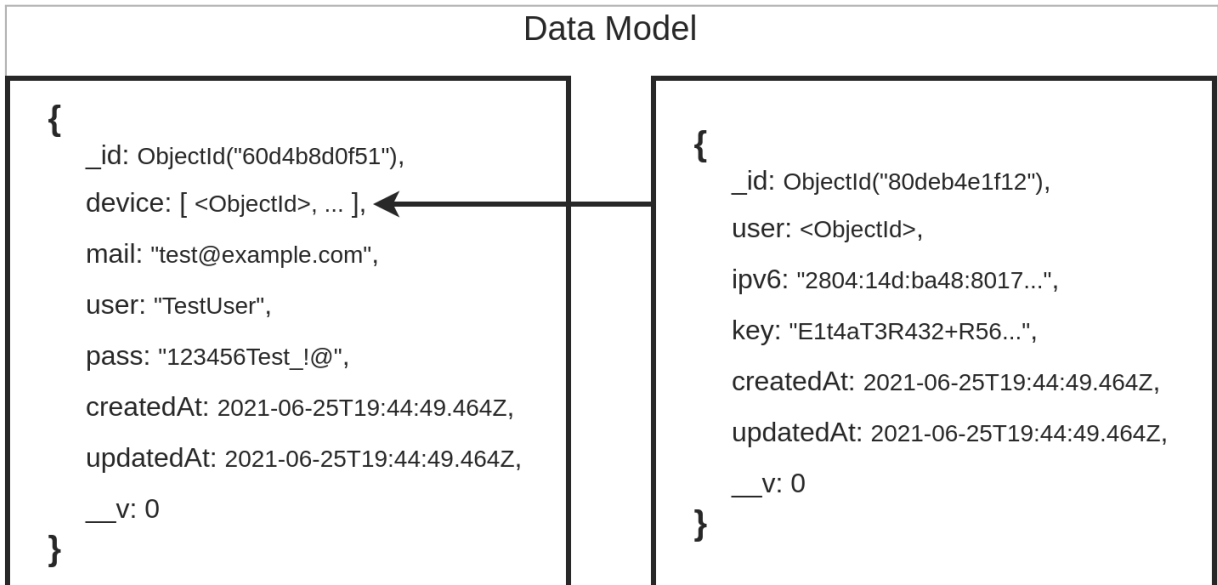
Ilustração 2: Diagrama de Estrutura e Relacionamento de Dados (schema)



Fonte: Elaboração do Autor (2021)

O diagrama abaixo representa uma estrutura de documento no formato JSON; como os dados serão armazenados no banco de dados. Este é um exemplo do modelo de dados, que foi utilizado durante os testes para seleção do banco de dados deste projeto.

Ilustração 3: Diagrama de Estrutura e Relacionamento de Dados (data)



Fonte: Elaboração do Autor (2021)

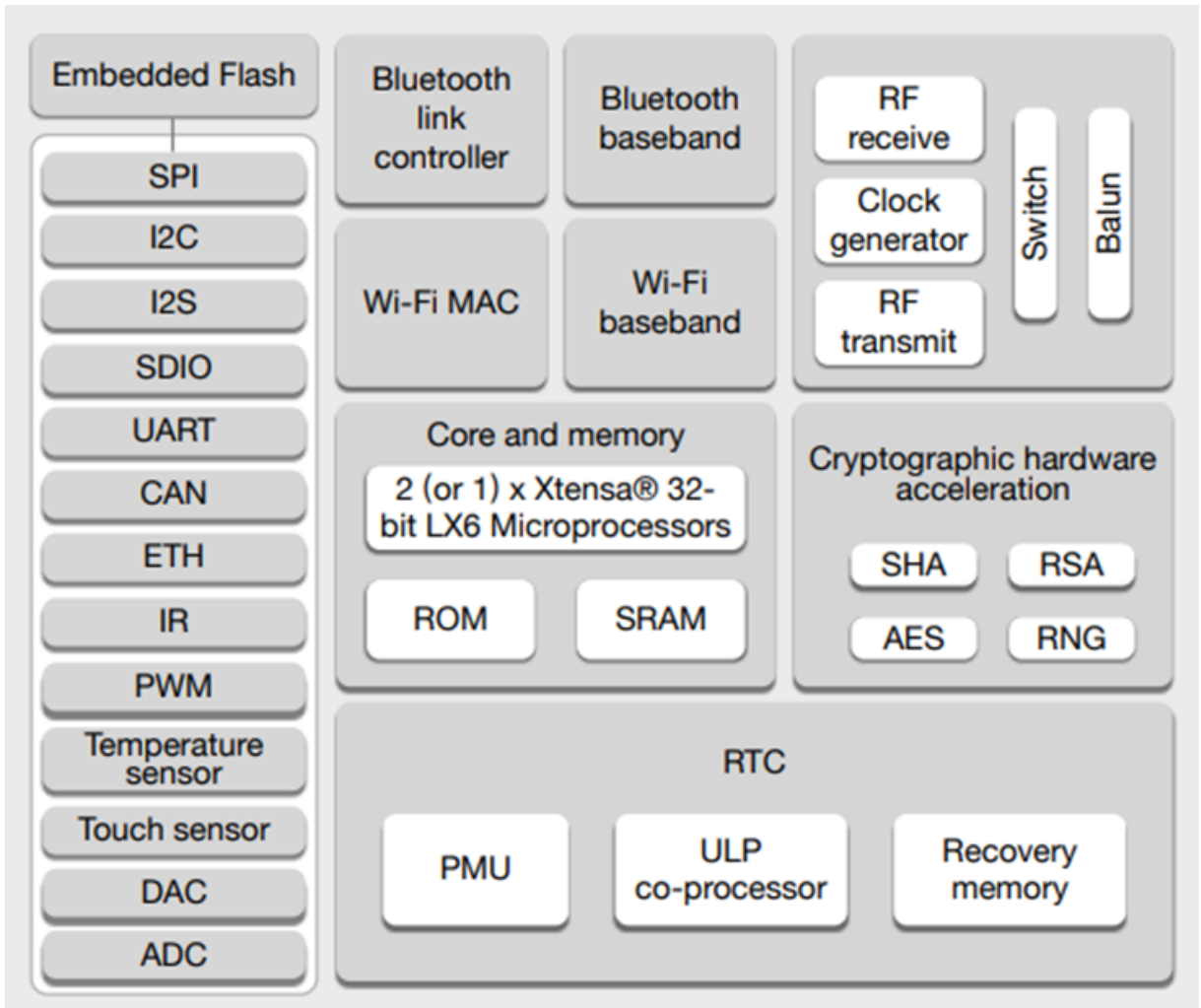
6.3.3 Protótipo

Os próximos diagramas são referentes a arquitetura do microcontrolador utilizado no protótipo do presente projeto. Toda função lógica do dispositivo é processada e gerenciada por este microcontrolador representado a seguir.

Segundo Fernando Koyanagi, especialista em *hardwares* de IoT e inteligência artificial, em publicações constantes no seu blog de tecnologias e desenvolvimento de soluções IoT, faz menções a quantidade de recursos proporcionados pelo microcontrolador ESP32, e sua vasta compatibilidade com os mais diversos periféricos eletrônicos e de automação.

Esse diagrama expõe que o ESP32 tem dual core, tem uma área do chip que controla o WiFi, e outra que controla o Bluetooth. Também tem aceleração de hardware para criptografia, que permite a conexão com LoRa, rede de longa distância que permite uma conexão até 15km, isso com uso de uma antena. Ainda observamos o clock generator, real time clock, e outros pontos que tratam, por exemplo, de PWM, ADC, DAC, UART, SDIO, SPI, entre outros, que fazem esse dispositivo bastante completo e funcional. (KOYANAGI, 2018)

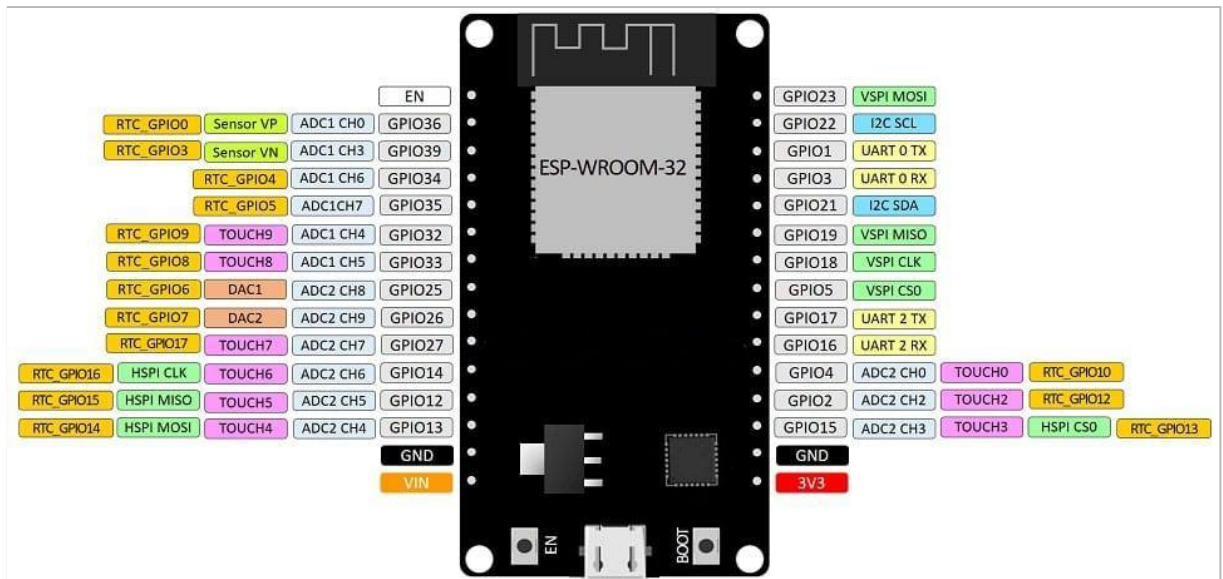
Ilustração 4: Diagrama de Arquitetura do Microcontrolador



Fonte: Fernando Koyanagi (2018)

Abaixo o diagrama de pinagem do microcontrolador ESP32.

Ilustração 5: Diagrama de Pinagem do Microcontrolador



Fonte: Fernando Koyanagi (2018)

6.4 Desenvolvimento

Na etapa de desenvolvimento do sistema e protótipo, para um resultado final satisfatório e harmônico com as práticas exigidas pelos docentes deste curso, todas as implementações e codificações do projeto passará por validações de desempenho, semântica e sintática, através de ferramentas de validação de organizações padronizadoras e normatizadoras, como, o World Wide Web Consortium (W3C) e Google Developers.

A primeira etapa do desenvolvimento será voltada para construção e implementação do *server-side*. Levando em consideração a estrutura e segurança de dados, protocolos de comunicação com desempenho adequado ao projeto, sistema de autenticação de usuários seguro e codificação estruturada em módulos, voltada para a facilitação das manutenções.

Posteriormente a implementação do *client-side*, com todas as validações voltadas para o desempenho, fluidez e compatibilidade da página *web*. A ideia é desenvolver uma página dentro das diretrizes de UX/UI sendo ela enxuta, objetiva e fluida, proporcionando uma experiência agradável ao usuário.

Por último, mas igualmente importante; o desenvolvimento do protótipo funcional. Onde o mesmo será desenvolvido sobre um microcontrolador ESP-WROOM-32; este já apresentado nos tópicos anteriores. Para o dispositivo, a ideia é desenvolver um *web service*, a qual o microcontrolador hospedará, funcionando como uma plataforma para pré-configuração do dispositivo e comunicação com servidor de gerenciamento.

7 SISTEMA

7.1 Software Embarcado

Acessando a página web diretamente do dispositivo, poderá efetuar-se a configuração para conexão com a plataforma de serviços. Abaixo encontram-se especificados os dados necessários para o funcionamento do mesmo.

7.1.1 Configurar Rede Wi-Fi (SSID e Senha)

Cada dispositivo possui um módulo de rede sem fio, onde haverá de ser conectado com a internet. Insira o SSID e senha de seu roteador nos respectivos campos para conexão com o dispositivo.

7.1.2 Inserir Servidor de Controle

Neste campo será inserido o endereço do domínio da plataforma de serviços. Desta maneira estabelecendo conexão entre o dispositivo e o servidor de serviços remoto.

7.1.3 Inserir Chave Criptográfica

Para garantir uma conexão segura do dispositivo e servidor, o usuário deverá inserir ou gerar uma chave criptográfica aleatória para o mesmo. A qual deverá importar para seu cadastro na plataforma, possibilitando desfrutar plenamente dos serviços remoto.

7.2 Softwares de Gerenciamento do Usuário - Front-end (Web Page e App)

O site visa auxiliar o usuário de forma dinâmica e simples no gerenciamento dos seus dispositivos de armazenamento seguro. Com seu visual enxuto e intuitivo, proporciona uma experiência de sofisticação e segurança para seus clientes.

7.2.1 Aba de Cadastro

O cadastramento de usuário é por meio de um formulário simples, requisitando apenas um e-mail válido (caso houver necessidade de recuperação de usuário e senha), nome de usuário e senha, conforme ilustrado na imagem abaixo.

Ilustração 6: Representação da aba de cadastro (client-side)

Ilustração da aba de cadastro (client-side). O formulário contém:

- Botões "Login" e "Cadastro" no topo.
- Campos de entrada para "Insira seu e-mail", "Insira seu usuário" e "Insira sua senha".
- Botão "Cadastrar" no rodapé.

Fonte: Elaboração do Autor (2021)

7.2.2 Aba de Login

Similar a tela de cadastramento de usuário, a tela de login solicita apenas o nome de usuário e senha para autenticação, conforme ilustrado na imagem a seguir.

Ilustração 7: Representação da página de login (client-side)

Ilustração da página de login (client-side). A interface apresenta:

- Logo "SafeBox IoT" e uma imagem de um cofre.
- Botões "Login" e "Cadastro" no topo.
- Campos de entrada para "Insira seu usuário" e "Insira sua senha".
- Botão "Entrar" no rodapé.
- Copyright "© SafeBox Corporation" no rodapé.

Um círculo de destaque amplifica a seção de login, mostrando os campos de usuário e senha, o botão "Entrar" e os botões "Login" e "Cadastro" no topo.

Fonte: Elaboração do Autor (2021)

7.2.3 Inclusão de Dispositivos

A inclusão dos dispositivos será por meio de um formulário simples, que abrirá clicando em um botão posicionado no canto inferior direito da página, conforme ilustrado nas seguintes imagens.

Ilustração 8: Botão para o formulário de inclusão de dispositivos



Fonte: Elaboração do Autor (2021)

Formulário para cadastramento dos dispositivos:

Ilustração 9: Representação do formulário de inclusão de dispositivos

Um formulário de inclusão de dispositivos apresentado em uma janela modal com uma borda cinza escura e um botão de fechar 'x' no canto superior direito. O formulário contém três campos de entrada de texto empilhados verticalmente, cada um com um rótulo cinza claro: 'Insira um identificador', 'Insira o IPv6' e 'Insira sua chave'. Abaixo dos campos, há um botão de ação arredondado com o texto 'Cadastrar' em cinza escuro.

Fonte: Elaboração do Autor (2021)

8 CONSIDERAÇÕES FINAIS

Diante destas restrições ocasionadas recentemente devido a pandemia, evidenciou-se a necessidade das tecnologias em auxiliar nas nossas atividades rotineiras, apresentando-nos uma segunda opção de realizá-las remotamente. Em especial ressalta-se a internet que proporcionou às diversas plataformas a garantir continuidade em boa parte das nossas atividades exercidas.

O SafeBox IoT além de ser uma solução inovadora dentro de seu nicho, vem atender da mesma maneira esta segunda opção a qual encontra-se restrita de acessos físicos pelos mais diversos motivos; mas, que ao mesmo tempo não abre mão do controle de seus bens.

Tendo em vista que diante de um novo problema, o melhor caminho para resolução é a inovação.

REFERÊNCIAS

MAGRANI, Eduardo. **A internet das coisas**. Rio de Janeiro, RJ: FGV Editora, 2018. Disponível em: <https://bibliotecadigital.fgv.br/dspace/bitstream/handle/10438/23898/A%20internet%20das%20coisas.pdf>. Acesso em: 01 abr. 2021.

MCEWEN, Adrian; CASSIMALLY, Hakim. **Designing the Internet of Things**. West Sussex, United Kingdom: John Wiley And Sons, Ltd, 2014. Disponível em: https://madsg.com/wp-content/uploads/2015/12/Designing_the_Internet_of_Things.pdf. Acesso em: 01 abr. 2021.

THIBODEAU, Patrick. **Explained: The ABCs of the Internet of Things**. 2014. Disponível em: <https://www.computerworld.com/article/2488872/emerging-technology-explained-the-abcs-of-the-internet-of-things.html>. Acesso em: 01 abr. 2021.

PEREIRA, Rômero Ricardo de Sousa. **Primeiros passos no mundo da Internet das Coisas**. 2016. Disponível em: <https://www.devmedia.com.br/primeiros-passos-no-mundo-da-internet-das-coisas/34102>. Acesso em: 01 abr. 2021.

FERREIRA, Elcio; EIS, Diego. **HTML5 Curso W3C Escritório Brasil**. 2013. Disponível em: <https://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>. Acesso em: 01 abr. 2021.

DANIEL, Glaucio. **Criptografia em Node JS com Bcrypt**. 2020. Disponível em: <https://hcode.com.br/blog/criptografia-em-node-js-com-a-lib-bcrypt>. Acesso em: 01 abr. 2021.

RABELO, Eduardo. **Node.js — Usando WebSockets**. 2020. Disponível em: <https://oieduardorabelo.medium.com/node-js-usando-websockets-5d642456d1f3>. Acesso em: 01 abr. 2021.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa**. 4ª Edição. São Paulo: Atlas, 2007.

KOYANAGI, Fernando. **ESP32: Detalhes internos e pinagem.** 2018. Disponível em: <https://www.fernandok.com/2018/03/esp32-detalhes-internos-e-pinagem.html>. Acesso em: 01 jul. 2021.