FACULDADE DE TECNOLOGIA DE SÃO PAULO

FÉLIX SILVA DE CARVALHO

Desenvolvimento de um Sistema de Simulação para Automação Corporativa

SÃO PAULO
2020

FACULDADE DE TECNOLOGIA DE SÃO PAULO

FÉLIX SILVA DE CARVALHO

Desenvolvimento de um Sistema de Simulação para Automação Corporativa

Monografia apresentada como requisito à obtenção do título de Tecnólogo em Microeletrônica, Curso de Tecnologia em Microeletrônica, Departamento de Sistemas Eletrônicos, Faculdade de Tecnologia de São Paulo.

Orientador: Roberto Katsuhiro Yamamoto

SÃO PAULO
2020

Dedico esse trabalho ao meu Senhor e Deus, a quem sou eterno devedor de minha vida, de minhas habilidades e de tudo o que produzo.

AGRADECIMENTOS

Agradeço a minha mãe pois sem a sua dedicação incansável eu jamais chegaria aqui.

A minha esposa Marcerlúcia pelo suporte durante todo o período de estudo.

Aos meus filhos Adriel e João Marcos que me inspiram e me incentivam todos os dias, mesmo quando sou alcançado pelo desânimo e pela vontade de desistir. O seu abraço me dava força para o próximo passo.

À ETEC Gildo Marçal Bezerra Brandão, na pessoa de seus professores, local onde nasceu a ideia desse trabalho e onde fui sempre incentivado a galgar patamares maiores.

À ETEC do Jaraguá que me permitiu o aprimoramento desse trabalho juntamente com apoio e amizade de seus professores, da coordenação e da direção.

À minha amiga Elisangela Rodrigues da Silva, que muito me ajudou nas primeiras revisões desse trabalho.

Ao amigo Jonas dos Santos Silva, que caminha comigo nessa empreitado boa parte do tempo, com seu apoio e suporte.

Ao Flávio Volpato que fez parte do grupo de TCC na ETEC de Jaraguá, pela sua paciência e apoio.

Ao prof. Dr. Roberto Katsuhiro Yamamoto, pela atenciosa e precisa orientação e pela confiança depositada nesse projeto.

A tantas outras pessoas que, de uma forma ou de outra, apoiaram, incentivaram e colaboraram para a realização desse trabalho e que por ser traído pela minha memória, foram omitidos aqui.

"Felix, qui potuit rerum cognoscere Causas"
(Feliz aquele que pode conhecer as causas das coisas).

VIRGÍLIO

RESUMO

Em qualquer tipo de projeto que está sendo desenvolvido, pode-se entrar na chamada divergência de perspectiva, que é a diferença de visão entre aquilo que o cliente conseguiu explicar, o que o desenvolvedor do projeto conseguiu entender e o que o cliente realmente quer. Num projeto de automação isso não é diferente. Portanto, o objetivo desse trabalho foi, usando conceitos encontrados na arquitetura, como a maquete eletrônica, juntamente com recursos advindos da indústria 4.0 e no mundo IoT como sensores e atuadores inteligentes, verificar a viabilidade do desenvolvimento de projetos de automação empresarial que permitissem ao cliente ver e interagir com o projeto antes mesmo do seu início. Para que esse objetivo fosse alcançado foi empregado a metodologia exploratória hipotético-dedutivo, já que o uso desses conceitos juntos é um tema pouco investigado. Usando uma abordagem direta através da construção de um protótipo, que contou com uma central de comandos, onde se encontra os servidores e onde os dados são processados, uma maquete eletrônica dentro de um ambiente de jogo, que permite ao cliente navegar por seu imóvel e interagir no mundo virtual através do acendimento e controle de luminosidade de lâmpadas, controle de velocidade de ventiladores, abertura de portas com controle de acesso e ver o resultado dessa interação no mundo virtual, ou seja, dentro do ambiente de jogo e no mundo real através de duas maletas, sendo que a maleta de controle permite a interação efetiva com o sistema através de botões, *slider* e leitor RFID, e outra maleta de potência, onde é possível verificar a funcionamento real de lâmpadas e ventiladores. Portanto, foi constatado que é perfeitamente factível essa abordagem, pois o protótipo, apesar de se utilizar de inúmeras ferramentas, o que lhe conferiram alguma complexibilidade em seu desenvolvimento, se mostrou viável, operacional e portátil, apresentando baixo custo para o desenvolvimento de projetos de automação. Há ainda uma caminhada para que o projeto alcance a sua maturidade através de possíveis otimizações e inclusões de novas funcionalidades, mas o fato é que esse projeto lança luz numa área com alta demanda, mas que ainda não se utiliza de todo o potencial que os novos paradigmas tecnológicos oferecem atualmente.

Palavras-chave: Maquete eletrônica, IoT, Autodesk Revit, Autodesk 3DS Max, Unreal Engine, MQTT, Industria 4.0.

ABSTRAT

In any type of project that is being developed, one can enter the so-called divergence of perspective, which is the difference in vision between what the client was able to explain, what the project developer was able to understand and what the client really wants. In an automation project this is no different. Therefore, the objective of this work was, using concepts found in architecture, such as the electronic model, together with resources from industry 4.0 and the IoT world such as sensors and intelligent actuators, to verify the feasibility of developing business automation projects that would allow the customer see and interact with the project even before it starts. In order for this objective to be achieved, the hypothetical-deductive exploratory methodology was employed, since the use of these concepts together is a topic that has been little investigated. Using a direct approach through the construction of a prototype, which had a command center, where the servers are located and where the data is processed, an electronic model within a game environment that allows the customer to browse his property and interact in the virtual world through the lighting and control of lamp luminosity, fan speed control, opening doors with access control and seeing the result of this interaction in the virtual world, that is, within the game environment and in the real world through two suitcases, the control suitcase allowing effective interaction with the system through buttons, slide and RFID reader and another power suitcase where it is possible to check the actual functioning of lamps and coolers. Therefore, we found that this approach is perfectly feasible, because the prototype, despite using numerous tools, which gave it some complexity in its development, proved to be viable, operational and portable, presenting low cost for the development of automation projects. . There is still a journey for the project to reach its maturity through possible optimizations and the inclusion of new features, but the fact is that this project sheds light in an area with high demand, but that still does not use all the potential that new technological paradigms currently offer.

Keywords: Electronic model, IoT, Autodesk Revit, Autodesk 3DS Max, Unreal Engine, MQTT, Industry 4.0.

LISTA DE FIGURAS

Figura 1 - Estrutura do VEoT 24
Figura 2 - Resultado do App X-Campus 24
Figura 3 - Exemplo de um auditório onde se pode usar automação corporativa..... 27
Figura 4 – Comparação entre a Indústria Tradicional e a Indústria 4.0 30
Figura 5 - Exemplo de uma Maquete Eletrônica 32
Figura 6 - Raspberry Pi 3 Modelo B. 35
Figura 7 - Família ESP8266 36
Figura 8 - Módulo ESP32 36
Figura 9 - Módulo Sensor Touch Capacitivo I2C MPR121 37
Figura 10 - Funcionamento do Módulo Touch 38
Figura 11 - Interface Gráfica do Node-RED 39
Figura 12 - Programação através de nós 40
Figura 13 - Tela do Revit 41
Figura 14 - Interface do 3DS Max 42
Figura 15 - Interface do Unreal Engine 4 com um projeto de arquitetura. 42
Figura 16 - Blueprint Visual Scripting 42
Figura 17 – Supervisório feito pelo editor vis 45
Figura 18 - Foto da Igreja Presbiteriana da Esperança 48
Figura 19 - Interações 50
Figura 20 - Editor do Unreal 52
Figura 21 - Editor do Unity 53
Figura 22 - Planta da baixa da igreja 56
Figura 23 - Modelagem produzida no Revit 59
Figura 24 - Modelagem da Porta no Revit, no 3DS Max e um foto da porta real 60
Figura 25 - Modelagem Final em 3DS Max 61
Figura 26 - Mini roteador. 63
Figura 27 - Exemplo de interface no ambiente real 66
Figura 28 - Exemplo de Interface Node-RED para celular 67
Figura 29 - Exemplo de interface do ioBroker 68
Figura 30 - Interface para smartphone e no Unreal. 69
Figura 31 - Interface da maleta de comando 73
Figura 32 - PCI da maleta de comando 74

Figura 33 - Antes de depois da construção da PCI 75
Figura 34 - Pinos do ESP32 75
Figura 35 - Sistemas na PCI 77
Figura 36 - Esquema elétrico da PCI 78
Figura 37 - PCI 79
Figura 38 - Imagem 3D da PCI 80
Figura 39 - Configuração para execução de vídeos no Adobe Reader – Menu 100
Figura 40 - Configuração para execução de vídeos no Adobe Reader – Tela de opção 101

LISTA DE VÍDEOS

Vídeo 1 - Exemplo de VEoT com realidade aumentada. 33
Vídeo 2 - Estrutura da Instalação Elétrica de uma Sala..... 58
Vídeo 3 - Modelagem 3D no ambiente do Unreal Engine. 62
Vídeo 4 - Testes iniciais da parte eletrônica. 74
Vídeo 5 - Controle Dinâmico de Interface. 82
Vídeo 6 - Vídeo da primeira versão do projeto 113

LISTA DE TABELAS

Tabela 1 – Adaptadores do ioBroker. 43
Tabela 2 - Dimensionamento das Trilhas da PCI 74
Tabela 3 - Definição dos pinos GPIO do ESP32 76

LISTA DE ABREVIATURAS E SIGLAS

AAL	*Ambient Assisted Living* (Ambiente de Vida Assistida)
BIM	Building Information Modeling (Modelagem de Informações da Construção)
CSS	*Cascading Style Sheets* (Folha de Estilo em Cascatas)
EDA	*Electronic Design Automation* (Projeto de Circuitos Integrados)
ERP	*Entreprise Resource Planning* (Planejamento dos Recursos da Empresa)
GPIO	*General Purpose Input/Output* (Entrada e Saida para Propósito Gerais, ou Portas Programáveis de Entrada e Saída)
HIoT	*Human Internet of Things* (Internet das Coisas humana)
HTML	*Hypertext Markup Language* (Linguagem de Marcação de Hipertexto
HTTP	*Hypertext Transfer Protocol* (Protocolo de Transferência de Hipertexto)
I^2C	*Inter-Integrated Circuit* (Circuito Inter-Integrado)
IHM	Interface Homem-Máquina
IIoT	*Industry Internet of Things* (Internet das Coisas Industrial)
IoE	*Internet of Everything* (Internet de Tudo)
IoT	*Internet of Things* (internet das coisas)
IP	*Internet Procotol* (Protocolo de Internt)
LED	*Light-Emitting Diode* (Diodo de emissão de luz
M2M	*Machine to Machine* (Comunicação Direta Entre Máquinas)
MEMS	*Micro-Electro-Machanical System* (Sistemas Microletromecânicos)
MQTT	*Message Queuing Telemetry Transport (*Transporte de Telemetria de Enfileiramento de Mensagens)
NVE	*Networked Virtual Environment* (Ambiente Virtual Através de Rede)
PLC	*Programmable Logic Controller* (Controlador Lógico Programável)
PS4	*PlayStation 4*
PWM	*Pulse-Width Modulation* (Modulação por Largura de Pulso)
REST	*Representational State Transfer* (Transferência de Estado Representacional)
RFID	*Radio Frequency Identification* (Identificação por radiofrequênia)
SCADA	*Supervisory Control and Data Acquisition* (Sistemas de Supervisão e Aquisição de Dados)
SoC	*System on a Chip* (Sistema em um chip)
TCP	*Transmission Control Protocol* (Protocolo de Controle de Transmissão)
TI	Tecnologia da Informação

UART	*Universal Asynchrounous Receiver/Transmiter* (Receptor/Transmissor Universal Assíncrono)
ULP	*Ultra Low Power coprocessor* (Coprocessador ultra baixo consumo)
VEoT	*Virtual Environment of Thing* (Ambiente Virtual das Coisas)
VR	Realidade Virtual
WoT	*Web of Things* (web das coisas)

SUMÁRIO

1 INTRODUÇÃO 16

1.1 Contextualização 16

1.2 Justificativa 17

1.3 Problema de pesquisa 18

1.4 Objetivo Geral 18

1.5 Objetivos Específicos 19

1.6 Metodologia de trabalho 19

1.7 Estrutura dos capítulos 19

2 REFERENCIAL TEÓRICO E REVISÃO BIBLIOGRÁFICA 21

2.1 Referencial Teórico 21

2.1.1 Realidade Virtual 21

2.1.2 BIM/IoT 22

2.1.3 VR, IoT, IIoT, HIoT, *Smart Cities* 23

2.1.4 VEoT 23

2.1.5 IoT para superar o gargalo da informação 25

2.2 Revisão Bibliográfica 25

2.2.1 Automação Corporativa 25

2.2.2 Tecnologias 26

2.2.3 Hardwares 34

2.2.4 Softwares 37

2.2.5 Protocolo 45

2.2.6 Demais objetos usados 46

3 DETALHAMENTO DO PROJETO 47

3.1 Definição do Protótipo 47

3.2 Funcionamento 47

3.3 Arquitetura 49

3.4 Tecnologias 50

3.4.1 Ambiente real 50

3.4.2 Ambiente de Virtualização 51

3.4.3 Servidor de Aplicativos 52

3.4.4 Comunicação 53

3.4.5 Processamento das informações 54
3.4.6 Interface para smartphone/tablet 54
3.4.7 Microcontrolador 54
3.4.8 Base de dados 55
4 DESENVOLVIMENTO DO PROTÓTIPO 56
4.1 Desenvolvimento do Ambiente Virtual 56
4.1.1 Autocad 56
4.1.2 Revit 57
4.1.3 3DS Max 58
4.1.4 Unreal Engine 60
4.2 Desenvolvimento da Central de Comando 61
4.2.1 Instalações 62
4.2.2 Criação das Interfaces 65
4.3 Maletas 70
4.3.1 Maleta de Comando 70
4.3.2 Maleta com os atuadores 71
4.4 Programação 77
4.4.1 Unreal 77
4.4.2 Maletas (ESP8266 e ESP32) 78
4.4.3 Central de Comando (Node-RED) 81
4.5 Testes do Sistema 82
4.5.1 Unreal 83
4.5.2 Maletas 83
4.5.3 Interfaces 83
4.5.4 Teste Geral 84
5 CONCLUSÕES 85
5.1 Trabalhos futuros 85
5.2 Material publicado e exposto 87
6 REFERÊNCIAS 88
7 APÊNDICES 100
APÊNDICE A – INSTRUÇÕES PARA EXECUTAR OS VÍDEOS. 100
APÊNDICE B – LISTA DE CHECAGEM DO SISTEMA. 102
APÊNDICE C – ARTIGO PUBLICADO NO BOLETIM TÉCNICO DA FATEC-SP EM OUTUBRO DE 2018. 110

APÊNDICE D - TRABALHO EXPOSTO NO 20º SIMPÓSIO DE INICIAÇÃO CIENTÍFICA E TECNOLÓGICA DA FATEC-SP - SICT REALIZADO EM 18 DE OUTUBRO DE 2018. 111
APÊNDICE E - TRABALHO DE CONCLUSÃO DE CURSO NA ETEC JARAGUÁ - 2º SEMESTRE 2019. 112
APÊNDICE F - VERSÃO 1.0 DO PROJETO VIRTUS – VÍDEO APRESENTAÇÃO DO TCC NA ETEC GILDO MARÇAL BEZERRA BRANDÃO EM DEZEMBRO DE 2016 113

1 INTRODUÇÃO

1.1 Contextualização

Imagine o seguinte cenário: uma mulher em sua cozinha super moderna com sua geladeira de última geração, e, ao pegar a última caixinha de creme de leite, a geladeira emite uma ordem de compra ao supermercado em que a família é cadastrada, solicitando um novo contingente de creme de leite juntamente com alguns outros itens que já estão em suas últimas unidades. O supermercado que dispõe de uma logística automatizada, separa de forma autônoma os produtos solicitados, faz o débito na conta bancária da família e solicita um transporte que faz a entrega na casa. Todas essas ações foram acompanhadas pela mulher do início dessa cena através de uma rede social. Como aponta Santos (2018, p. 14), isso parece um filme de ficção científica, mas parece que coisas assim estão cada dia mais próximas da realidade. Pelo menos as tecnologias implicadas já existem.

Já a indústria 4.0 é uma simbiose entre os conceitos trazidos pela internet das coisas (*Internet of Things* – IoT), a eletrônica e a tecnologia da informação, juntamente com os avanços na comunicação, integração e pela interconexão gerada por aplicativos cada vez mais eficientes e sensores cada vez menores (Dedding, 2018, p. 14) trazendo possibilidades que sequer eram imaginadas a poucas décadas atrás.

Voltando a atenção para o *modus operandi* do projeto de automação, seja ela residencial ou empresarial, tem-se inicialmente uma empresa que cria animações para dar uma ideia dos recursos de automação oferecidos (Discabos, 2017). Uma vez que o cliente se interesse pela proposta, recebe a visita dos representantes da empresa que vão verificar as possibilidades de implementação na empresa do potencial cliente e, então, oferecem a esse uma solução, geralmente de forma verbal com indicações dos locais e as respectivas soluções que melhor se adequam àquela situação. E aqui, em algumas ocasiões inicia-se com as potenciais dificuldades. É muito difícil um detalhamento preciso do que é possível ser feito, primeiro porque quem vende às vezes, está vendo o ambiente pela primeira vez e depende apenas do chamado “olho clínico” para encaixar a solução que ele conhece em um ambiente que ele não conhece. O cliente, por sua vez, não conhece o produto e devido a também estar sendo apresentado pela primeira vez, fica apenas com o seu primário entendimento da solução e com o sonho vendido pelo vendedor. No momento do fechamento, têm-se um vendedor que não conhece o ambiente, o cliente que não conhece o produto e um escopo superficial. Quando se tem em vista projetos pequenos de baixa complexidade, a falta de informações

precisas tanto do vendedor como também do cliente pode ter baixo impacto na execução e entrega do projeto. Mas, conforme aumenta o projeto sendo incluídos mais itens dentro da automação, a interligação entre a eletrotécnica, a automação e a tecnologia da informação (TI), as chances de um projeto que vai ter um custo muito maior, um prazo de execução não cumprido e um cliente decepcionado por não ter sido o projeto entregue exatamente como ele comprou, passam a ser relevantes.

É justamente nesse hiato de informação e expectativas que o presente projeto pretende atuar. Hoje

> ... as tecnologias digitais, como IoT, *big data*, *analytics*, *cloud computing*, robótica avançada e colaborativa, novos materiais (como fibras de carbono e grafeno), manufatura aditiva e híbrida, realidade aumentada e realidade virtual, o ciberespaço, sensores em rede, entre outras, combinadas com a automação industrial, conectando mundo real e virtual, proporcionam a melhoria da produtividade pela otimização de processos e novos modelos de negócios. (Dedding, 2018, p. 14)

Diante disso, como aproveitar essas novas tecnologias para dar ao cliente não a ideia, mas a sensação do projeto, melhorando exponencialmente a percepção do cliente em relação ao que está sendo vendido, além de organizar as informações de forma precisa e orgânica para, de fato, mensurar a complexidade e antever potenciais problemas que somente apareceriam, às vezes, nas etapas mais avançadas do projeto, impondo adendos que geram custos extras, alargamento de prazo de entrega e minando a eficiência e credibilidade do projeto?

1.2 Justificativa

O presente trabalho procura alinhar-se às práticas mais modernas no que se refere à confecção de projetos, uma vez que faz uso de tecnologias como BIM (*Building Information Modeling*), que possibilita não somente uma primeira modelagem em 3D, mas também permite um dimensionamento quantitativo mais preciso dos materiais usados e permite a integração com outros projetos, como estruturais, elétricos e hidráulicos. Também faz uso de conceitos usados pela IoT, como o protocolo MQTT (*Message Queuing Telemetry Transport*) que permite um sistema de transição de dados de forma rápida, leve, segura e eficiente. Há também o uso do conceito de maquete eletrônica imersa em um ambiente que originalmente

foi desenvolvido para jogos eletrônicos, mas que tem sido nos últimos anos, usado pela arquitetura para dar um ar de realidade a seus projetos.

Com o uso dessas tecnologias, é possível construir um projeto de automação e segurança que também anexa o projeto elétrico onde é possível antecipar problemas que não são facilmente vistos em projetos tradicionais, melhorar a definição de escopo do projeto, já que o projeto pode ser visto pelo cliente em um formato muito próximo ao que será quando o mesmo estiver implementado. Uma interação com o projeto como se esse já estivesse totalmente executado, permitindo revisões e ações que nos projetos tradicionais só poderiam ser feitos com a execução do projeto já em andamento.

1.3 Problema de pesquisa

Com o avanço cada vez mais rápido de novas tecnologias as metodologias aplicadas para o desenvolvimento de projetos em várias áreas precisam ser constantemente revisadas e quando possível, verificando sempre o custo-benefício dessas novas implementações devem ser atualizadas visando maior eficiência e menores custos. Diante desse constante desafio, o problema de pesquisa que o presente trabalho levanta é como aplicar os conceitos do IoT e da Indústria 4.0 no desenvolvimento de projetos de automação empresarial, já que esses dois paradigmas impõem novas formas de abordagens daquilo que já se tem o domínio abrindo uma infinidade de novas possibilidades.

1.4 Objetivo Geral

Através da aplicação de conceitos trazidos pelo IoT e pela Indústria 4.0, o objetivo do presente trabalho é a construção de um protótipo que permita verificar a viabilidade no desenvolvimento de projetos de automação empresarial, através da inter-relação entre o mundo virtual, que será feito através de um ambiente de jogos digitais, com o mundo real, através de atuadores e controladores, de forma que a ação no mundo real obtenha respostas tanto no mundo físico como no mundo virtual e a ação no ambiente de jogo obtenha respostas em ambas as esferas.

1.5 Objetivos Específicos

- Implementação da modelagem 3D do prédio que será a base da automação;
- Importação da modelagem em um ambiente de virtualização para implementação das animações;
- Construção de uma central de processamento para comandar os acionamentos;
- Construção de uma maleta de demonstração para acionamento e outra para potência com luzes e ventiladores;
- Construção de um aplicativo mobile para o controle através do ambiente virtual, e *smartphones*.

1.6 Metodologia de trabalho

A metodologia de trabalho usada nesse projeto é de natureza exploratória (Moretti, 2018), já que o uso de maquete eletrônica aliada a passeios virtuais está mais ligado à arquitetura e as áreas da eletrotécnica, automação e controle como também a área de segurança ainda são temas com pouca exploração nesse contexto. Em uma pesquisa preliminar não foi encontrado nenhum material que aborde de forma específica o conceito da maquete virtual como ferramenta para a confecção de projetos voltados para as áreas acima citadas.

Os procedimentos terão uma abordagem direta utilizando a construção de um protótipo, optando pela metodologia hipotético-dedutivo (Arruda, 2017), onde se parte da premissa da viabilidade do uso dessas tecnologias ligadas ao IoT e indústria 4.0 como ferramenta para a confecção de projetos de automação e segurança passando pela construção de um protótipo que tem como objetivo a comprovação e a veracidade da hipótese (Baxter, 2000, p. 243).

1.7 Estrutura dos capítulos

Sendo o objetivo principal desse projeto a construção de um protótipo que servirá de ferramenta para a confecção de projetos de automação e segurança, contará primeiramente com o capítulo sobre implementação da modelagem em 3D do prédio que será a base para o projeto de automação. Começará com uma modelagem dentro do conceito BIM e, em um

segundo momento, será usado um aplicativo de modelagem em 3D para que sejam trabalhados os detalhes da construção com mais precisão e detalhes.

Seguir-se-á um capítulo onde será mostrado o processo de importação dessa modelagem em 3D para uma Engine Gráfica (Kleina, 2011) onde será possível a realização do "passeio virtual" e as animações que permitirão abrir e fechar portas, acender e apagar luzes e ligar e desligar os ventiladores.

Com a parte gráfica concluída, o próximo capítulo se ocupará com a construção de uma central de processamento, que receberá as instruções de vários dispositivos, fará o processamento e enviará comandos para execução de ações dentro do sistema como um todo. Aqui contempla a configuração dos servidores que serão usados pelo protótipo.

A operacionalidade da central de processamento permitirá a confecção das maletas que comporão a parte física do projeto (juntamente com a central de processamento). Haverá uma maleta de demonstração que terá a função de controle onde será possível o acionamento das portas, lâmpadas e ventiladores, com interruptores e *slider*. Em uma segunda maleta de demonstração terá como função a demonstração física dos dispositivos (porta, lâmpadas e ventiladores).

Com a parte virtual e física finalizadas, a atenção do projeto se voltará para o controle *mobile* que será utilizado tanto através de *smartphones* como dentro do ambiente virtual e terá como base um aplicativo usado para automação residencial e empresarial.

As impressões finais juntamente com os direcionamentos para a continuação da pesquisa e as próximas etapas de desenvolvimentos comporão o último capítulo. Serão analisadas a possível viabilidade do protótipo procurando descrever os pontos positivos e aqueles que ainda merecem atenção no sentido de buscar soluções mais eficientes para o aprimoramento de um futuro produto.

2 REFERENCIAL TEÓRICO E REVISÃO BIBLIOGRÁFICA

2.1 Referencial Teórico

A proposta deste trabalho com relação ao referencial teórico (Centro de Ciências Exatas e Sociais Aplicadas, 2015) é, conforme o objetivo geral, trabalhar com os conceitos encontrados no IoT e Indústria 4.0 para a construção de um protótipo para verificar a viabilidade de elaboração de projetos de automação empresarial fazendo uso da interoperação entre o mundo real e a realidade virtual usando, para isso, dispositivos físicos como interruptores *dimmers*, lâmpadas, ventiladores, entre outros, cuja atuação e o resultado dessa atuação possam estar no mundo real como também na realidade virtual.

Em pesquisas foram encontrados alguns trabalhos que transitam dentro de alguns conceitos e aspectos que este trabalho se propõe a percorrer, mas nenhum trabalho pesquisado traz em seu bojo o mesmo objetivo que esse projeto toma para si. De qualquer forma, esses trabalhos lançam luz sobre o caminho escolhido e ajudam a refinar os conceitos e objetivos delineados no início desse projeto.

2.1.1 Realidade Virtual

O primeiro trabalho consultado foi o do sr. Freitas (2016) que, em sua tese de doutorado, utiliza-se de ferramentas disponibilizadas pela realidade virtual para simular o efeito radioativo de uma instalação de contenção de rejeitos radioativos sobre as pessoas que ali trabalham no manejo desse material. Foi desenvolvido um ambiente virtual tanto no Unreal como no Unity3D para dimensionar a radiação emitida por toneis que tem a capacidade de conter boa parte da radiação, sobre pessoas que estão nesse ambiente em diversas situações, como por exemplo, a várias distâncias, através de paredes, enquanto trabalham com empilhadeiras, etc.

Esse trabalho apresenta, então, o uso da realidade virtual como simulação para uma situação real onde se pode aferir algumas informações de forma mais simples e no caso específico desse projeto, com menos riscos à saúde do pesquisador. Ele trata de alguns assuntos que são relevantes a presente pesquisa tal como, além do próprio uso da realidade virtual, a definição de diversos autores sobre o que é realidade virtual, os tipos de interação que o usuário pode ter dentro de um ambiente virtual, os tipos de navegação, os tipos de ambientes virtuais e comentários, de forma breve, sobre o que é uma *engine*, ou motor

gráfico. Todos esses assuntos, foi de grande auxílio na delimitação desse projeto assim como prestou significativa ajuda para definir os conceitos que são correlatos em ambos os trabalhos.

Um ponto que chama a atenção é quando o autor aborda o conceito de dispositivo de entrada e de saída. Nesse ponto, os projetos convergem brevemente, mas o presente trabalho avança na importação de conceitos comuns na eletrotécnica e na automação para essa interação entre o mundo real e os seus dispositivos de entrada e saída comuns aos jogos eletrônicos. Nesse ponto, haverá acionadores e atuadores que participação dessa interação.

2.1.2 BIM/IoT

Outro trabalho consultado foi de Machado & Ruschel (2017) que teve por objetivo, através de Revisão Sistemática de Literatura (RSL), encontrar pesquisas que trabalhassem com a interação entre paradigma BIM e o paradigma IoT. Para isso, foi feita uma busca nas bases de dados SCOPUS, Compendex, ASCE *Library* e *Web of, Science*, no período de 2000-2016 (Machado & Ruschel, 2017, p. 143). A conclusão é que "abordagens relativas à integração BIM/IoT, com ênfase em energia, estão em pleno desenvolvimento e indicam soluções e conhecimento em evolução para a incorporação de informações dinâmicas no modelo BIM, associadas aos comportamentos dos objetos reais correspondentes" (Machado & Ruschel, 2017, p. 149).

Esse documento colabora como base do presente trabalho no sentido que já há várias pesquisas que se propõe verificar a usabilidade de tecnologias que trabalham de forma a interoperar o mundo real com o mundo virtual. Embora o enfoque do artigo ora analisado se restrinja ao paradigma BIM junto com o IoT, em um recorte que não faz parte essencial desse presente projeto, que é a tecnologia BIM como provedor de informações para monitoramento e análise, algo muito presente na Industria 4.0, a pesquisa mostra uma carência na pesquisa da junção de tecnologias visando um primeiro estágio no ciclo de vida de uma construção ou de um *retrofit*, ou seja, a fase de projeto onde a virtualização e a IoT podem dar uma substancial contribuição, assim como municia o presente trabalho com informações sobre BIM e IoT numa perspectiva diferente da que está contemplada no presente trabalho.

2.1.3 VR, IoT, IIoT, HIoT, *Smart Cities*

No artigo do autor Vaccari (2015), observando o crescimento da virtualização, propõe-se a dar uma visão geral nos termos que estão sendo usados no paradigma da realidade virtual e em sua conexão com o mundo real através do conceito da internet das coisas e de outras tecnologias, como a IIoT (internet das coisas industrial), a HIoT (internet das coisas humana - voltada para usuários domésticos) que tem surgido dessa inter-relação indo até o conceito de cidades inteligentes. Também aborda os motores de jogos que podem criar ambientes virtuais tanto para a realidade virtual como para a realidade aumentada que foram originalmente desenvolvidas para jogos, mas graças a IoT, tem sido usado para uma infinidade de coisas, como no exemplo já citado das cidades inteligentes. O autor traz inclusive dois exemplos de cidades inteligentes e como o seu uso tem sido eficiente na resolução de problemas reais dessas cidades.

2.1.4 VEoT

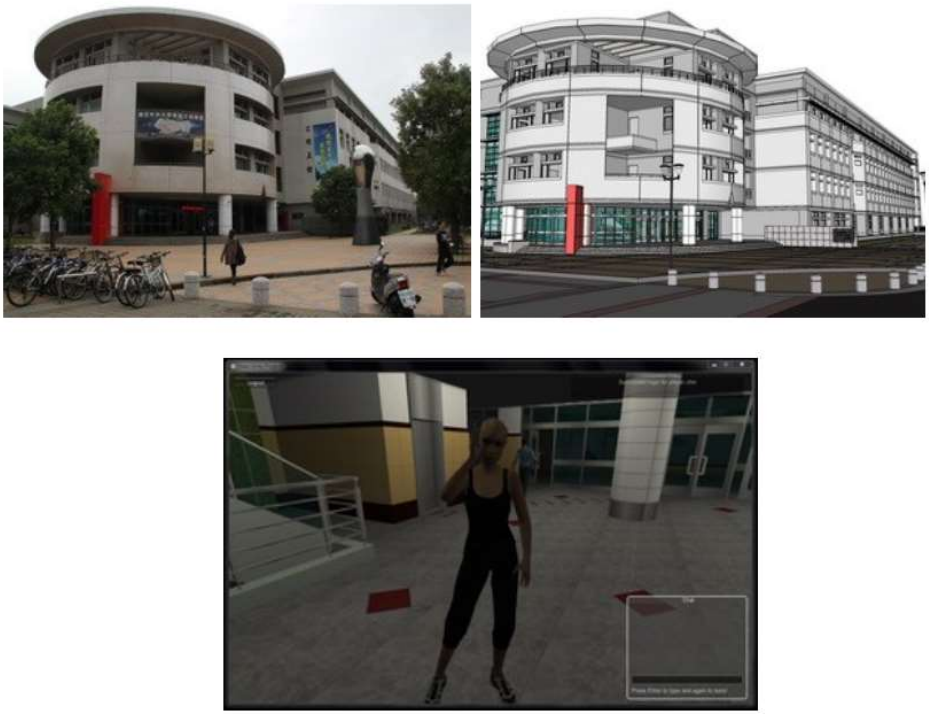
No artigo dos autores Wu, Chou e Jiang (2014) através do uso do paradigma usado na junção do IoT com a plataforma web, que deu origem a WoT (*Web of Things* - web das coisas) - que teve por objetivo usar os protocolos de comunicação como padronização para realizar a interoperação entre o mundo real e a interface baseada na web - para criar um novo paradigma, a VEoT (*Virtual Environment of Things* - ambiente virtual das coisas) que permite a interoperação entre o mundo físico (real) e o mundo virtual ou realidade virtual, através da junção do IoT com o chamado NVE (*Networked Virtual Environment* - ambiente virtual através de rede), um ambiente virtual onde pessoas assumem uma identidade virtual (avatares) e podem interagir um com os outros. Para viabilizar o *framework*, os autores usaram a arquitetura REST (*Representational State Transfer*) do HTTP (*Hypertext Transfer Protocol*) para montar a comunicação entre as duas realidades, conforme a Figura 1 e fizeram a validação através de uma aplicação chamada X-Campus que tinha por objetivo guiar visitantes através de um edifício da *National Central University* (Figura 2). A aplicação funcionou perfeitamente comprovando a perfeita viabilidade do *framework* VEoT.

Através dessa aplicação os objetos físicos, como lâmpadas e sensores (temperatura, humidade e luminosidade) puderam ser acionados e ter seus valores adquiridos através da aplicação. Ao fim da experiência os autores se propuseram a ir adiante em suas pesquisas para

Figura 1 - Estrutura do VEoT

Fonte: (Wu, Chou, & Jiang, 2014)

Figura 2 - Resultado do App X-Campus



Fonte: (Wu, Chou, & Jiang, 2014)

verificarem a escalabilidade do VEoT e uma forma de fazê-lo por meio da comunicação M2M, ou seja, comunicação máquina-máquina. Partindo dos conceitos lançados por esse documento e aplicando o princípio proposto pelo trabalho dos autores desse artigo, procurar-se-á dar as respostas levantadas com relação à escalabilidade e com relação ao uso da comunicação M2M.

2.1.5 IoT para superar o gargalo da informação

Nesse pequeno texto, Ashton (2009) reflete sobre a questão de que apesar da quantidade de informações que se tem dentro dos computadores (incluindo a internet), aproximadamente, 50 petabytes (na época em que o artigo foi escrito), o computador depende do ser humano para adquirir essas informações. Por outro lado, o ser humano não é um meio eficiente para adquirir informações. Nesse gargalo, entra a IoT com seus sensores e seu modo de transportar informações para, a partir daí, dar aos computadores um meio de obter a informação de modo focado, rápido e preciso. Nessa perspectiva, não se está preparado para o que virá e sequer se tem ideia das futuras possibilidades.

2.2 Revisão Bibliográfica

Nessa parte do trabalho, será utilizada a revisão bibliográfica para apresentar os principais elementos das tecnologias que serão usadas no projeto (Centro de Ciências Exatas e Sociais Aplicadas, 2015, pp. 7-8). A base para essa pesquisa foram a biblioteca digital SciELO (www.scielo.org) e a ferramenta de buscas do Google direcionado para pesquisa, o Google Acadêmico (scholar.google.com.br), além de pesquisas no Google padrão. Para essa apresentação, foi feita uma classificação para uma melhor apresentação desses conceitos: *hardwares*, *softwares*, paradigmas, linguagens de programação e outros que não permitiram ser agrupados em uma classificação específica.

2.2.1 Automação Corporativa

Automação Corporativa é um conceito relativamente novo e surge graças aos avanços tecnológicos nas áreas da eletrônica, internet e tecnologia da informação (TI), contando também com a renovação do ambiente de trabalho corporativo trazida pela arquitetura

contemporânea, sendo muito requisitado por consumidores de projetos comerciais (Neocontrol, s.d.) que querem ou precisam impactar seus próprios clientes apresentando-lhe um ambiente tecnológico moderno por um lado (Foneplan, 2015) e primam por economia de energia, otimização de processo, que consequentemente gera redução de custos e lhe permite estar de mãos dadas com a sustentabilidade (Fórum da Construção, 2018) como por exemplo, *coworkings* (escritórios compartilhados com planos de aluguel temporário), auditórios e salões de até 150 m^2, como na Figura 3, e escritórios com espaços compartilhados.

Mas, afinal o que é Automação Corporativa? O site Foneplan dá a seguinte definição:

> Automação corporativa é o controle automático dos recursos tecnológicos do ambiente a fim de melhorá-lo, facilitando a manipulação dos componentes do ambiente corporativo no dia a dia. Para isso são usados sistemas inteligentes capazes de controlar esses recursos (Foneplan, 2015).

Ou seja, é um sistema de controle e automação que se utilizada de tecnologias como IoT com uma central *plug and play* que possibilita o ajuste de cargas elétricas em equipamentos que estão dentro de sua área de atuação, sejam lâmpadas, ventiladores ou ar-condicionado, controle de acesso a portas, telas e projetores, entre outros (Neocontrol, s.d.).

2.2.2 Tecnologias

2.2.2.1 Internet das Coisas (IoT)

Já em 2009, a IoT recebia uma grande atenção e recursos já eram destinados para o seu estudo para uma melhor compreensão e a verificação de sua aplicabilidade pela União Europeia (Casagras: an EU Framework 7 Projects, 2009, p. 4). Assiste-se a indústria ser carregada por um *tsunami* de dados que são capturados e transmitidos por diversos dispositivos microcontrolados, sensores, atuadores se interligando a rede de computadores em diversas áreas. Segundo a empresa SAS, nesse ano, 2020, “o mercado da Internet das Coisas Industrial (IoT) movimentará US$ 151 bilhões, crescendo a uma taxa anual composta de 8%. E, segundo a IDC, o universo de informação digital criada e distribuída por diferentes objetos inteligentes no mundo deverá representar 44 trilhões de gigabytes” (SAS, 2019).

Figura 3 - Exemplo de um auditório onde se pode usar automação corporativa.

Fonte: (Foneplan, 2015)

Cunhado em 1999 (Ashton, 2009), a IoT enfrenta muita dificuldade quando o assunto é a conceituação do termo de forma universal (Cossini, 2015, p. 7) e é possível que ela permaneça como uma realidade intangível ainda por algum tempo, mas quando se olha para a perspectiva de cada vez mais tecnologias sendo combinadas, pode-se antever que nos próximos 5 a 15 anos uma modificação drástica de como a sociedade contemporânea funciona (Casagras: an EU Framework 7 Projects, 2009, p. 10).

Mas como definir a chamada “Internet das Coisas”?

Sinclair (2018, p. 15) define a IoT como “apenas uma evolução da internet. Nada mais, nada menos”. Mas, como explicar essa evolução? Quais os itens que a compõe? Talvez uma tentativa de compreensão seja olhar para o IoT como um conjunto de camadas de tecnologias (Machado & Ruschel, 2017, p. 147). Tem-se na primeira camada os objetos inteligentes (*smart things*) (Doneda, 2018, p. 11) que através de sensores, atuadores e microcontroladores tornam objetos simples (Alluring Stories, 2018) em objetos capazes de obter, processar, transmitir e atuar de forma autônoma, ou serem controlados remotamente

através de uma infraestrutura de rede (Thakur, 2016, p. 4) caminhando assim para uma classe de sistemas cyber-físicos que atingem casas (que já podem responder a comandos de voz semelhantes ao Siri do iOS, Alexa do Amazon ou ainda o Cortana do Windows), carros, edifícios e até cidades tornando-os todos inteligentes.

A segunda camada que é a capacidade de identificação única que torna um objeto inteligente único, mas com alta capacidade de cooperar (Casagras: an EU Framework 7 Projects, 2009, p. 10) e interagir com outros objetos (Thakur, 2016, p. 4).

Na terceira camada, tem-se a comunicação propriamente dita, onde se encontra a tecnologia M2M (*machine-to-machine* – comunicação direta entre máquinas) (Doneda, 2018, p. 11), a conexão entre objetos físicos e sistemas computacionais (Thakur, 2016, p. 4) e evoluindo para mundos virtuais com alto grau de captura e transferência de dados (Casagras: an EU Framework 7 Projects, 2009, p. 10). Tudo ligado à internet formando assim uma infraestrutura de rede global, onde desde microssensores podem conviver com bilhares de outros dispositivos com plena capacidade de interoperação e alto grau de captura, transferência e análise de dados, formando os dados a quarta camada, que podem ser adquiridos em tempo real, processados por itens com eletrônica embarcada e microprocessados gerando quase que instantaneamente respostas que podem atuar de forma precisa em qualquer lugar do mundo.

A quinta camada contém as inúmeras tecnologias que são interligadas tais como *big data*, *learning machine*, inteligência artificial, realidades virtual e aumentada gerando o que se encontra na sexta camada que é a evolução que todos esses componentes tem gerado e que estão muito longe de serem consolidados, já que a cada dia mais tecnologias são introduzidas, milhões de dispositivos são produzidos para alimentar esse paradigma e as informações coletadas aumentam de forma exponencial, fazendo-se necessária a criação de novos métodos de análise e o aprimoramento dos já existentes.

Outro ponto muito importante relacionado à IoT é na questão da aquisição de dados. Todos os dados disponíveis na internet, pelos menos até 2011 eram de aproximadamente 50 petabytes (Super Interressante, 2011), ou seja, o volume de informações que foi produzido na história de toda a humidade e colocado em formato digital (Ashton, 2009), sendo que um petabyte é 1024 terabyte ou 10^{15} em decimal ou 2^{50} em informação ou armazenamento computacional. E para os computadores terem essa quantidade de dados armazenada, houve a necessidade do ser humano inserir esses dados, seja através da digitação, gravação ou tirando fotos. Então, é possível dizer que o ser humano é quem roteia as informações do mundo real

para o mundo computacional. O problema é que as pessoas têm grande limitação de tempo, atenção e precisão e agilidade, na transmissão de dados tornando-se, assim, uma espécie de gargalo nessa tarefa (Ashton, 2009). A IoT, com os seus sensores microcontrolados, vence com muita facilidade as limitações inerentes dos seres humanos gerando uma quantidade exponencialmente muito maior de dados. A IoT pode ser uma solução para as deficiências humanas na aquisição de dados.

Diante da grandeza do paradigma da IoT, ele tem se subdivido quando adentra áreas mais específicas dentro do mundo computacional. Tem-se a IIoT e a HIoT que são respectivamente Internet das Coisas para indústria (*Industry Internet of Thing*) e Internet das Coisas para humanos (*Human Internet of Thing*) (Vaccari, 2015, p. 2). Tem-se ainda a WoT, ou Web das coisas (*Web of Things*) que junta o IoT com a plataforma WEB e a VEoT ou, Ambiente virtual das coisas (*Virtual Environment of Things*), quando há a junção do IoT com a realidade virtual (Wu, Chou, & Jiang, 2014, p. 1) e a IoE (Lal Das, Beisswenger, Mangalam, Yuce, & Lukac, 2002, p. 9), que é a internet de tudo (*Internet of Everthing*) que seria a evolução natural da IoT (Heebra, 2015).

2.2.2.2 Indústria 4.0

O conceito de Indústria 4.0 não é algo recente. Na verdade, a indústria vem ano após ano investindo em aprimorar os seus processos para diminuir o seus custos, melhorar o seu relacionamento com o cliente, tirar os gargalos na produção (Santos S. , 2018, p. 11), mas com o desenvolvimento de novas tecnologias, foi possível avançar de forma acentuada em inovações que culminaram em um novo paradigma na indústria mundial. “Ainda precisa-se compreender de forma mais abrangente a velocidade e a amplitude dessa nova revolução” (Schwab, 2019). A ideia propriamente tida da Indústria 4.0 nasce em Hannover, Alemanha em 2011, com o objetivo justamente de se servir do aumento computacional e da conectividade exponencial. (Suárez, Salazar, Nava, & Hernández, 2019, p. 8), mas vai muito além disso, como por exemplo:

> Computação em nuvem (e plataformas em nuvem), *big data* (análise avançada de dados, lagos de dados, inteligência de borda) com inteligência artificial (relacionada), análise de dados na borda de redes *Mobile*, tecnologias de comunicação de dados / rede, mudanças no nível de, entre outros, IHM e SCADA, sistemas de execução de manufatura, planejamento de recursos empresariais (ERP, tornando-se i-ERP), controladores lógicos programáveis (PLC), sensores e

> atuadores, MEMS [...] e modelos inovadores de troca de dados desempenham um papel fundamental (Santos S. , 2018, pp. 14-15).

Dessa forma, a mudança estrutural da indústria tradicional, como se conhece para a Indústria 4.0, é uma mudança completa de paradigma, como se vê na Figura 4, trazendo muito mais eficiência, precisão, agilidade, transferência, mesmo que parcial, de autonomia e decisões autônomas para sistemas e máquinas ciberfísicas, alavancando sistemas de informação (Santos S. , 2018, p. 14). Nessa dinâmica de fusão de tecnologias, onde uma dá base para a construção e ampliação de outras acabam por fundir os mundos físico, digital e biológico (Schwab, 2019).

Figura 4 – Comparação entre a Indústria Tradicional e a Indústria 4.0

Fonte: (Gerbert, et al., 2015)

2.2.2.3 BIM (Building Information Modeling)

BIM, ou Modelagem da Informação da Construção "é um dos desenvolvimentos mais promissores nas indústrias de arquitetura, engenharia e construção" (EASTMAN, TEICHOLS, SACKS, & LISTON, 2010, p. 1), já que vem como uma substituição do processo de planejamento linear e segmentado baseado em abstrações 2D tradicionais que tem como consequência o retrabalho, o desperdício com baixa eficiência e alto custo na realização

do empreendimento (MENEZES, VIANA, PEREIRA JR, & PALHARES, 2012) para um sistema que visa convergir todos os projetos de arquitetura, estrutura, mecânica, infraestrutura, construção, elétrica e mais recentemente automação (DZAMBAZOVA, KRYGIEL, & DEMCHAK, 2009, p. 4), trazendo detalhes precisos que ficam disponíveis para todos os que estão envolvidos no projeto em todo o ciclo de vida da construção facilitando as rápidas tomadas de decisões que esse tipo de projeto geralmente exige.

Com essa tecnologia, pode-se, antes de qualquer gasto, construir um edifício virtual com grande precisão que contém toda a geometria e os dados relevantes para que sejam feitas correções, adequações e mudanças mais profundas baseadas em problemas que podem ser visualizados virtualmente (EASTMAN, TEICHOLS, SACKS, & LISTON, 2010, p. 1).

2.2.2.4 Realidade Virtual

Hoje, com o poder de processamento dos computadores e suas placas gráficas, com o cinema criando ou copiando mundos inteiros que arrebatam e faz, mesmo que de forma não racional, ter a certeza que de tudo aquilo é tão real, com robôs sendo controlados a milhares de quilômetros com grande precisão entre outros exemplos possíveis, poder-se-ia dizer que se vive, ao menos quando se está em contato com essas tecnologias, que se está vivendo em um mundo virtualizado (TORI & KIRNER, 2006, p. 16). Mas, como se pode definir realidade virtual? Parte-se da ideia que uma simulação na qual um computador recria um mundo tal qual o real, tentando copiá-lo em tantos detalhes quanto for possível (Freitas, 2016, p. 31). Em outras palavras, pode-se dizer que “realidade virtual é um ambiente fictício com o qual você pode se relacionar de maneira intuitiva, exatamente como se relacionaria com o mundo real” (GOMES, 2019, p. 377). Mas, para que seja de fato realidade virtual, é essencial a interatividade em tempo real do usuário. Ele precisa ter a sensação de instantaneidade de seus comandos no mundo real dentro desse mundo virtual (BURDEA & COIFFET, 2003, p. 2) (TORI & KIRNER, 2006, p. 13).

A experiência proporcionada pela realidade virtual pode ser dividida em função de como o usuário se percebe diante do sistema virtual. Essa experiência pode ser imersiva, quando o usuário é transportado para dentro dessa realidade através da captura de parte de seus sentidos, como com o uso de óculos feitos especialmente para essas práticas. Outra possibilidade é da experiência não imersiva, quando o usuário tem contato com o mundo virtual, mas de forma parcial, através de uma janela, como um monitor ou um projetor (TORI

& KIRNER, 2006, p. 8). O presente projeto seguirá a experiência não-imersiva sendo o contato do usuário com o mundo virtual através da projeção do ambiente.

O ambiente virtual em si, pode ser definido como “um conjunto de geometria, a relação espacial da geometria com o usuário ou usuários e a alteração na geometria em resposta às ações do usuário e com o tempo.” (HALE & STANNEY, 2018, p. 260), ou seja, uma representação 3D de um ambiente que pode simular ou não um ambiente real e onde é possível se navegar e interagir, em tempo real (ZAGOLO & PEREIRA, 2008, p. 151). E um dos modos possíveis para a criação desses ambientes é usando o conceito de maquete eletrônica (Figura 5) que, utilizando-se de desenhos 2D pode ser transformado em ilustrações muito próximas a realidade (SERAFIM, s.d.), representando um projeto arquitetônico e/ou urbanístico (LIMA, 2016), onde se tem a simulação de espaços internos e externos que diminuem substancialmente as chances de erro na execução desse projeto (totalCAD, 2018).

Figura 5 - Exemplo de uma Maquete Eletrônica

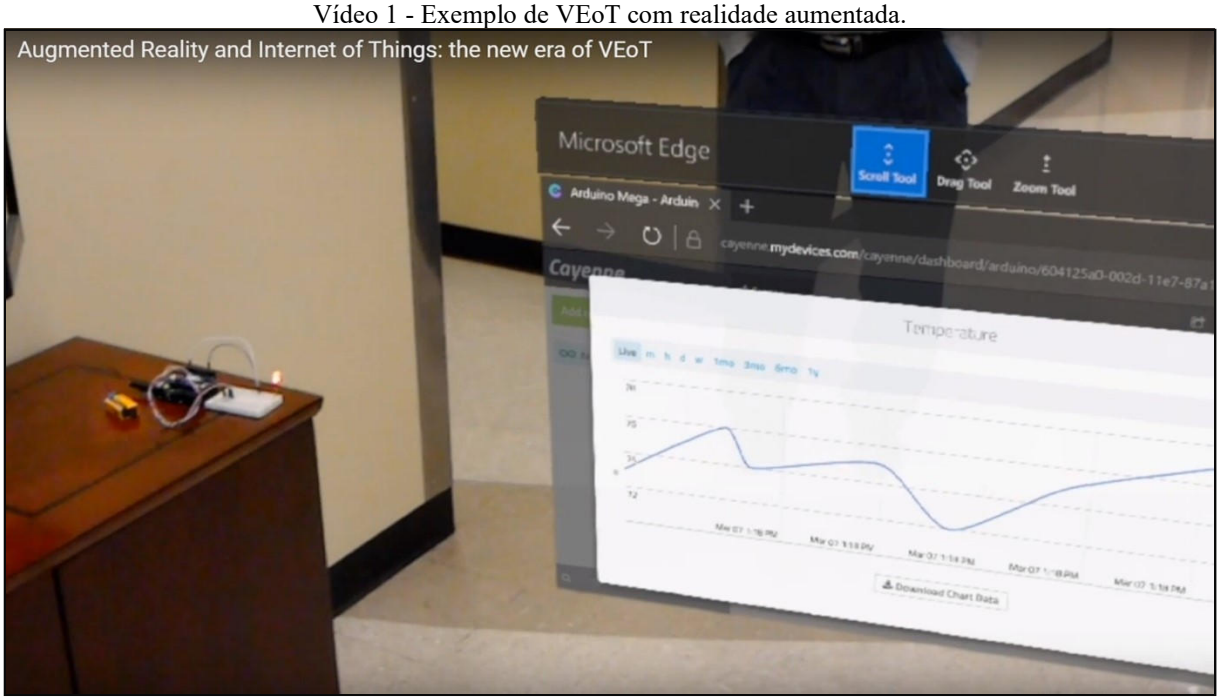
Fonte: (AMARAL, s.d.)

Um aspecto importante na realidade virtual diz respeito à interação entre o usuário e o ambiente virtual e isso é proporcionada através dos dispositivos de entrada e saída (VINCE, 1998, p. 13), uma “central de comando” que dá a interação do usuário um jeito mais próximo do dia a dia humano e menos mecânico/cibernético (GOMES, 2019, p. 377). Freitas (2016, p. 35) citando Pimentel, afirma que há duas categorias de dispositivos de entrada, a saber, o de interação e o de trajetória ou de navegação. Como dispositivos de interação pode-se citar as

luvas de dados e sensores de entrada biológicos (TORI & KIRNER, 2006, p. 48) enquanto como exemplo de dispositivo de navegação pode ser apresentado “rastreadores eletrônicos, ópticos, ultrassônicos, eletromagnéticos e inerciais” (SENA, 2013, p. 52).

Este trabalho trata de uma abordagem pouco usual referente à realidade virtual, que é a junção dela com o IoT, tendo como resultado o VEoT (*Virtual Environment of Things*, ou Ambiente Virtual das Coisas). O conceito foi abordado no Referencial Teórico (2.1.4 - VEoT) e citado quando foi discutido sobre a tecnologia IoT (2.2.2.1 - Internet das Coisas (IoT), por isso, aqui basta acrescentar que além da realidade virtual esse conceito também pode ser trabalhado com a realidade aumentada, como mostra um vídeo, de onde foi tirada o Vídeo 1 (clique sobre a imagem para rodar o vídeo. Caso não rode acesse o apêndice A).

Vídeo 1 - Exemplo de VEoT com realidade aumentada.



Fonte: (KOUTITAS, 2017)

2.2.2.5 RFID

A identificação por rádio frequência (RFID) é uma tecnologia de comunicação sem fio, onde dados (que são usados para a identificação de um determinado produto) (CHETOUANE, 2015, p. 382) são armazenados em um dispositivo, conhecido como *transponder* que pode ser checado usando-se campos eletromagnéticos (FINKENZELLER, 2010) entre as etiquetas com *chip* RFID (*tag*) e leitores RFID sendo intermediada por um *software* que faz a interpretação dos dados obtidos. À distância (dependendo do circuito

eletrônico) é possível identificar diferentes códigos em produtos similares com diversos níveis de identificação permitindo rastreabilidade e visibilidade em um processo de produção e/ou logística (CHETOUANE, 2015, p. 382).

Uma etiqueta RFID pode operar de forma ativa, quando a mesma, por ser alimentada por uma fonte consegue enviar os dados para o seu respectivo leitor e também pode operar de forma passiva, quando uma corrente é induzida pelo leitor para que os dados que estão na etiqueta possam ser lidos (NASSAR & VIEIRA, 2014).

2.2.3 Hardwares

2.2.3.1 Raspberry Pi

O Raspberry Pi nasceu de um desejo de disponibilizar poder computacional de baixo custo que fosse acessível ao maior número de pessoas (Oliveira, 2017, pp. 50-51), principalmente crianças, que poderiam, então, aprender como os computadores funcionam, ter acesso ao mundo da eletrônica e ter acesso ao mundo da programação (Souza, Carvalho, Andrade, & Silva, 2016, p. 20). O projeto é de código aberto equipado com processadores ARM,

> [...] na versão 3, tem um processador de 1,2 GHz de 64 bits, incluindo, ainda, WiFi, Bluetooth, um gigabyte de memória RAM, quatro portas USB, 40 pinos de entrada e saída, saída HDMI e diversas outras interfaces, como câmera e memória externa. Tem, ainda, um processador gráfico capaz de codificar vídeo H.264 full HD em tempo real (Oliveira, 2017, p. 51).

O Raspberry Pi é uma pequena placa (85,6 mm x 53,98 mm x 17mm) pesando 45g (Dennis, 2013, p. 10). Abaixo, na Figura 6, vê-se um exemplar do mini computador.

Em alguns projetos relacionados à IoT, como é o caso do presente, tem-se uma configuração de rede heterogênea e a inclusão do Raspberry Pi como um elemento centralizador dentre vários outros dispositivos, com aumento exponencial da disponibilidade de recursos sem que haja um aumento significativo no custo da rede (Oliveira, 2017, p. 50)

2.2.3.2 ESP8266

O ESP8266 é um módulo projetado pela Espressif Systems. É um sistema embarcado de alto desempenho e integração WiFi de dimensões e consumo diminutos (Silva, Oliveira,

Mendes, & Sales, 2017, p. 2). Por ter pinos de GPIO (*General Purpose Input Output* – Entrada/Saída para uso geral[1]), é possível conectar sensores e atuadores ao microcontrolador para recepção de informações e envio de comandos e por esses pinos também é possível a conexão com outros sistemas através de protocolos de comunicação como I2C, SPI e UART. Por ser pequeno e ter a habilidade tanto de se conectar a um roteador como de servir de ponto de acesso a outros dispositivos, é possível usá-lo para conectar outros dispositivos para o envio e recepção de dados. (Thakur, 2016, pp. 4-5). Há vários modelos de ESP8266, como mostrado na Figura 7 variando de um para outro a quantidade de pinos e funções disponíveis. O módulo que será usado no projeto é o NodeMCU 12-E que contém um conector mini-USB e 11 portas GPIO.

Figura 6 - Raspberry Pi 3 Modelo B.

Fonte: (Raspberry Pi Foundation, 2012)

2.2.3.3 ESP32

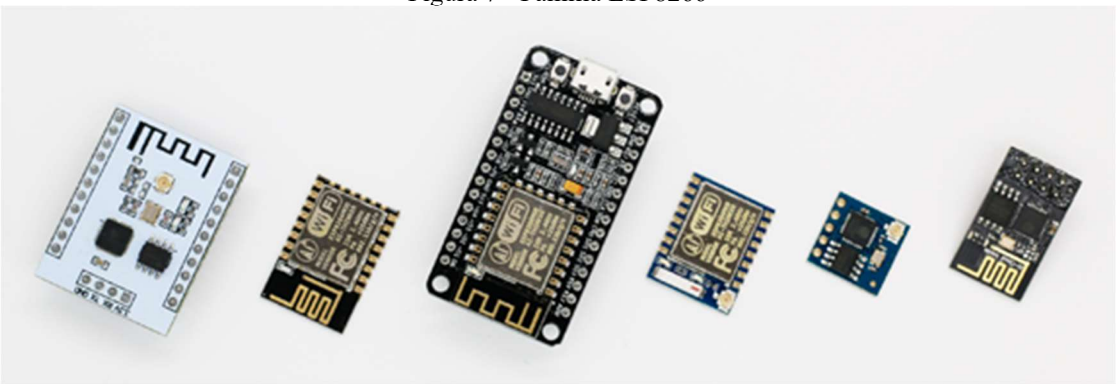
O ESP32 é o sucessor do ESP8266 que trouxe novos recursos e melhores que tornou o SoC (*System on a Chip* – sistema em um chip) muito mais robusto e expandiu o seu uso para outras áreas dentro dos sistemas embarcados como por exemplo, dispositivos *weareble* como os *SmartWatches* (Athos Electronics, s.d.). Entre as melhorias pode-se citar

> a memória SRAM de 520 KB, memória flash de 16 MB, dois núcleos de processamento em arquitetura Tensilica LX6 capaz de operar em 240 MHz, Bluetooth híbrido (clássico e BLE), tensão operacional de 2,3V á 3,6V, conector externo antena IPEX/PCB, redução de problemas com instabilidade, quantidade maior de portas programáveis, múltiplos sensores, cristal oscilador 32KHZ e criptografia com chave RSA de 4096 bits, ou seja, flexibilidade operacional e maior segurança na construção de sistemas automatizados (Curto Circuito, s.d.).

[1] GPIO é um sinal digital controlado por software (The Linux Kernel Archives, s.d.)

Além de possuir um processador principal com dois núcleos, o ESP32 ainda conta com um processador secundário ULP (*Ultra Low Power coprocessor* – Coprocessador ultra baixo consumo) de 8MHz que tem um consumo de 150 μA podendo ser usado até em *Deep Sleep* (processo de hibernação com baixíssimo consumo de energia) (Morais, 2017).

Figura 7 - Família ESP8266



Fonte: (THOMSEN, 2016)

Outra vantagem do ESP32 sobre o ESP8266 é a possibilidade do uso do FreeRTOS um RTOS (Real-Time Operating System – Sistema operacional de tempo real) consolidado no mundo dos embaraçados (Bertoleti, 2019).

Figura 8 - Módulo ESP32

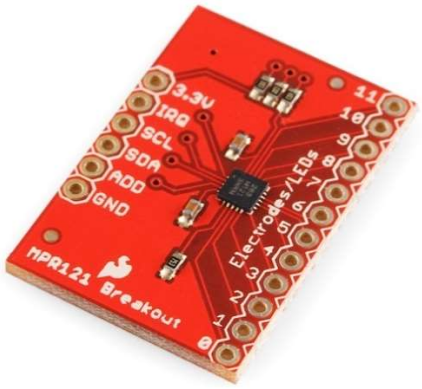
Fonte: (Allan, 2018)

O SoC ESP32 é apresentado em dois formatos: em forma de chip (conforme a Figura 8) e em módulo (Kurniawan, 2019). No presente projeto optou-se pelo chip por um melhor acabamento e menor espaço no PCI.

2.2.3.4 Módulo Touch (MPR121)

O módulo sensor *touch* capacitivo I2C MPR121 (conforme a Figura 9) utiliza o princípio da capacitância utilizando o eletrodo, um meio dielétrico que funciona como isolante, como verniz, vidro ou outro polímero, como o acrílico e ao toque de um dedo (ou da pele) há a inserção de uma capacitância adicional que indica que houve um toque (STEVAN JUNIOR & FARINELLI, 2019, p. 32) conforme mostrado na Figura 10. O módulo conta com uma interface I2C e pode controlar até doze eletrodos individuas sendo que 8 dos pinos apresentam função multifuncional com controle de LEDs e GPIO, operando com tensão entre 1.171V e 3.6V (Freescale Semiconductor, 2010).

Figura 9 - Módulo Sensor Touch Capacitivo I2C MPR121



Fonte: (Laboratório de Garagem, 2012)

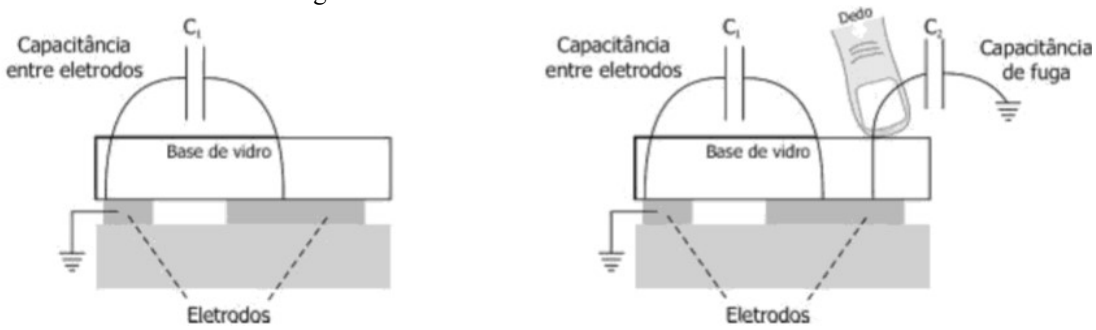
2.2.4 Softwares

2.2.4.1 Node-RED

Segundo o site "O Node-RED é uma ferramenta de programação para conectar dispositivos de *hardware*, APIs e serviços *online* de maneiras novas e interessantes" (OpenJS Foundation, s.d.). Ela permite reunir vários protocolos de comunicação, no qual o MQTT é nativo, juntamente com uma interface gráfica (Figura 11), permitindo o seu uso tanto por usuários experientes como aqueles que têm pouca vivência em programação (SZCRYS, 2020). Isso se deve ao fato de o Node-RED usar nós (*nodes*), como apresentado na Figura 12, que podem ser arrastados e colocados na tela e depois ligados por fios estabelecendo o fluxo

que a programação seguirá. Também é possível usar outras linguagens como o próprio Javascripts, que é a base dessa ferramenta, ou Python e Swift (OLIVEIRA, 2018).

Figura 10 - Funcionamento do Módulo Touch



Fonte: (STEVAN JUNIOR & FARINELLI, 2019, p. 32)

2.2.4.2 Autodesk Revit

Revit é uma aplicativo da Autodesk que trabalha dentro do conceito BIM, sendo ele um sistema de modelagem e documentação que consegue armazenar em um único lugar dados referente ao design, desenhos, cronogramas necessários para um projeto de construção (CHANG, 2018, p. 6). O grande trunfo do Revit ao se amoldar ao paradigma BIM é que ele permite não apenas fazer a modelagem 3D de uma forma relativamente intuitiva, em detalhes e permitindo tratar cada item do desenho com algo que tem a sua independência ao mesmo tempo em que se relaciona com o restante do cenário. Isso é feito através da parametrização onde cada parede, janela, porta, mesa, etc. tem um *id* e um conjunto de dados (como fabricante, medidas, composição, quantidade de material, etc.) que podem ser consolidados, contabilizados e até ser levantado um orçamento total do projeto. Além disso, como em materiais elétricos, é possível configurar as características elétricas e pontos de conexão de cada objeto (CHANG, 2018, p. 6). A Figura 13 mostra a interface do Revit.

2.2.4.3 Autodesk 3DS Max

O aplicativo 3DS Max é um programa de modelagem que aceita renderizar imagens e animações que é usado para a criação de mundos digitais para jogos, cenas fotorealísticas para a visualização de projetos arquitetônicos e ambientes virtuais para maquetes eletrônicas, animações, vinhetas, comerciais de TV (ALMEIDA, 2007, p. 5), visualizações forenses, médicas e científicas, *web design* (MATOSSIAN, 2001) e até a criação de simulações de fluidos (Autodesk, s.d.). Uma outra caractérista encontrada no 3DS Max é o ecossistema

montado pela Autodesk, em que projetos e objetos podem ser importados e exportados entre o 3DS Max e o Revit, proporcionando assim agilidade e precisão no desenvolvimento de projetos. Na Figura 14 tem-se uma visão geral da interface do 3DS Max.

Figura 11 - Interface Gráfica do Node-RED

Fonte: (SZCRYS, 2020)

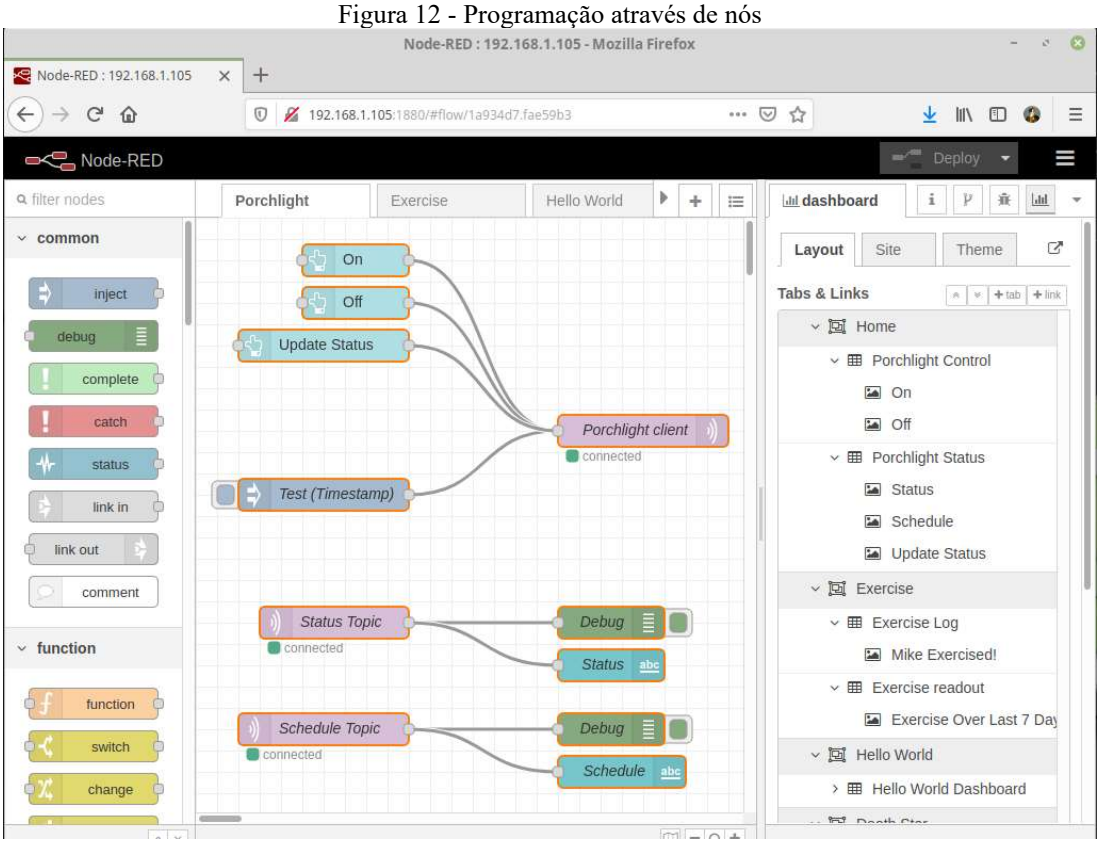
2.2.4.4 Unreal Engine 4

Segundo Tori & Kirner (2006, p. 203), o mundo dos jogos eletrônicos e o mundo acadêmico têm, a cada dia, estreitado relações quando se pensa nos jogos como ferramenta de pesquisa. E esse relacionamento com os jogos não é, nem de longe, algo exclusivo do mundo científico. O cinema, assim como as forças armadas, também já está nessa parceria há algum tempo. E nesse campo o Unreal Engine 4 é um dos parceiros mais populares.

O Unreal Engine, que agora está na versão 4.24, é um kit de desenvolvimento, que inicialmente era voltado para a produção de jogos eletrônicos criados pela empresa Epic Games. Teve seu início em 1998 com os conhecidos jogos de tiro em primeira pessoa com o nome que viria ser o nome da *engine*, “Unreal” (SANDERS, 2017, p. ix) e produz jogos para plataformas Microsoft Windows, Linux, Mac OS, em computadores pessoais, Android e iOS, em plataforma *mobile*, PS4, XBOX ONE, Nintendo Switch em consoles de vídeo game,

Steam e a própria Epic Games, em plataformas de jogos *online*, HTML 5, Flash e Javascript, em navegadores e Playstation VR, Oculus, Samsung Gear VR, Steam VR em plataforma de jogos *online* voltados para a realidade virtual (Unreal Engine, s.d.).

Figura 12 - Programação através de nós



Fonte: (SZCRYS, 2020)

Quando se fala em *engine*, ou motor de jogo, Tonéis (2017) dá a seguinte definição:

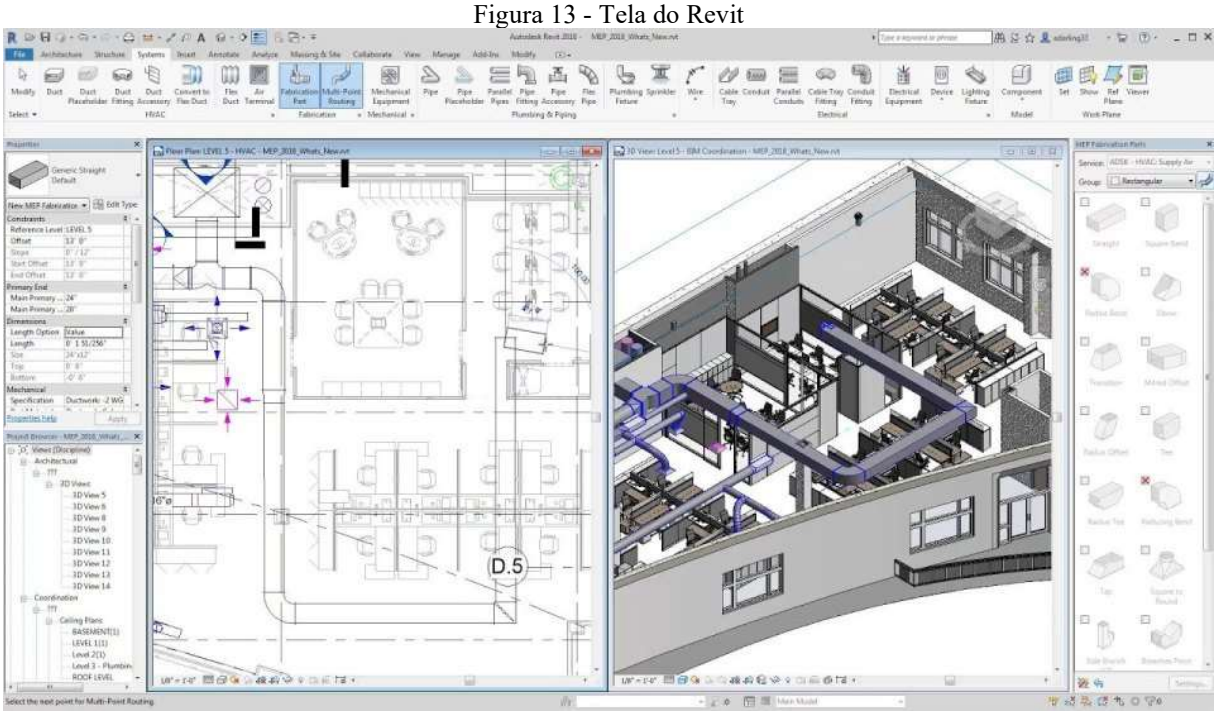
> *Software* de gerenciamento do núcleo usado para o desenvolvimento de jogos. Tem como propósito isolar as funcionalidades genéricas mais complexas nos *games*, como a física, iluminação e comportamento computacional, permitindo que o mesmo "motor" seja usado como base em vários títulos diferentes.

É nesse motor gráfico, como também é conhecido, que em *background*, uma maciça quantidade de cálculos é feita para, em tempo real, entregar todos os recursos disponíveis ao desenvolvedor (LEE, 2016, p. 6) tais como renderização, comunicação em rede, som e música, inteligência artificial, colisão, simulação física, entrada do usuário e multimídia (TORI & KIRNER, 2006, p. 214).

Com todos esses recursos que foram sendo desenvolvidos ao longo de sua história, o Unreal Engine coleciona títulos que fazem sucesso no universo *gamer*. A sua popularidade é

tamanha que entrou no Guiness World Book como “a *engine* de videogame de maior sucesso” (Guiness World Book, s.d.). Entre os melhores títulos recentes de jogos eletrônicos feitos com o Unreal Engine encontra-se o “Kingdom Hearts 3”, “LawBreakers”, “Sea of Thieves”, “Shadow Complex” e “Mass Effect” (Guilherme, 2019).

Figura 13 - Tela do Revit

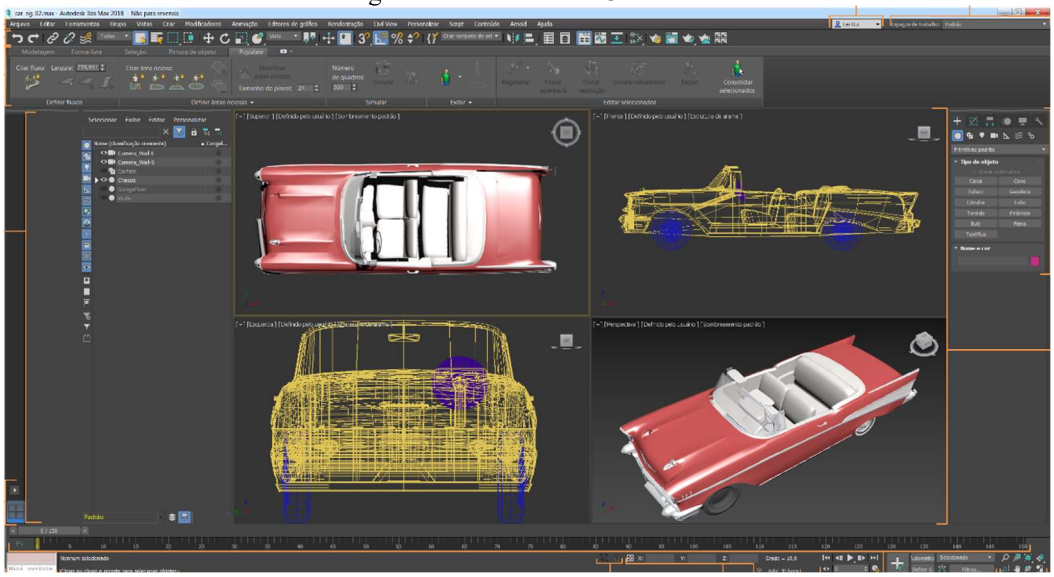


Fonte: (TAGLIANI, 2018)

Como já foi dito anteriormente, as *engines*, além do seu foco em jogos eletrônicos, tem feito parceiras com outros setores e entre eles está a engenharia e arquitetura, como mostrado na Figura 15, onde o uso do Unreal Engine resulta em projetos dinâmicos e interativos com a criação de rápida de protótipos completos, equipados com simulações e visualizações com qualidade fotorrealísticas (VENDRAMINI, s.d.).

Uma última informação em relação ao Unreal é que a partir de fevereiro de 2015 a Epic tornou a ferramenta gratuita em sua totalidade incluindo as futuras atualizações para todos os usuários. O único porém, é que a empresa pede 5% de *royalties* por trimestre quando as vendas brutas ultrapassarem o valor de US$ 3.000 (SWEENEY, 2015).

Figura 14 - Interface do 3DS Max

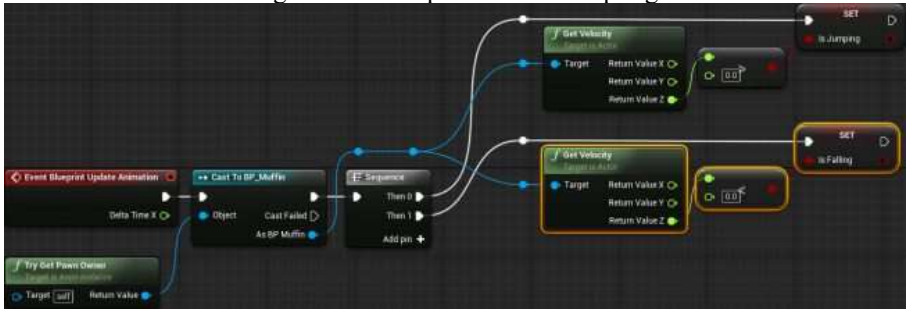


Fonte: (AutoDesk Help, 2018)

Figura 15 - Interface do Unreal Engine 4 com um projeto de arquitetura.

Fonte: (Papo de Arquiteto, s.d.)

Figura 16 - Blueprint Visual Scripting



Fonte: (TRAN, 2017)

2.2.4.5 IOBroker

Lançado em 2014, a plataforma integrada iobroker tem como principal função a intermediação de vários protocolos de comunicação voltados para a automação predial, *smart metering*[2], *Ambient Assisted Living*[3], processos de automação, registro e visualização de *logs*. Devido a essas características ele pode substituir ou estender a funcionalidade de sistemas de automação tais como FHEM, OpenHAB ou o Thing System (HEK, s.d.).

A estrutura do ioBroker funcionando de forma modular com os chamados adaptadores que representa cada sistema, protocolo, interface, etc. que funcionam de forma independente, mas que podem trocar dados e trabalhar de forma relacional. Todo gerenciado por uma interface intuitiva onde é possível iniciar, pausar e parar cada adaptador e trabalhar com mais de uma instância desse adaptador (HEINLE, 2017). Segue a Tabela 1 com alguns dos adaptadores nativos do ioBroker:

Tabela 1 – Adaptadores do ioBroker.

adaptador	Função
descoberta	Localiza dispositivos na rede local e sugere adaptadores
FHEM	Plataforma de automação comum
HomeMatic	Conexão ao sistema Homematic amplamente utilizado
iCal	Lê arquivos de calendário no formato ».ics«
Afluxo-DB	Log de estados em um banco de dados influx
JavaScript / Node.js	Executa seus próprios scripts Javascript
KNX	Acesso ao barramento KNX
Modbus	Conexão Modbus, escravo ou mestre
MPD	Controle do daemon do music player
MQTT	Comunicação com o broker MQTT
Mysensors	Projeto Hw / Sw-DIY para sensores / atuadores IoT acessíveis
Netatmo	Acesso a dispositivos domésticos inteligentes Netatmo

2 *Smart metering* é um dispositivo que registra o consumo de energia elétrica e comunica essa informação ao fornecedor de energia elétrica, além de também permitir a análise do consumo em tempo real tanto pelo consumidor como pela distribuidora de energia elétrica (RIVERA, 2018).

3 *Ambient Assisted Living* ou AAL é uma subcategoria da inteligência ambiental, o qual concebe o uso de técnicas, processos e tecnologias para permitir que os idosos vivam de forma independente o maior tempo possível, sem comportamentos intrusivos (SINGH, KROPF, HANKE, & HOLZINGER, 2017, p. 256).

Open HAB	Nova interface para o Open HAB
Osram Lightify	Produtos de iluminação inteligente, semelhantes ao Philips Hue
OWFS	Acesso ao barramento de 1 fio através do sistema de arquivos de um fio
Philips Hue Bridge	Comunicação com a Philips Hue Bridge
bala impulso	mensageiro App
RWE Smarthome	Integração de dispositivos domésticos inteligentes RWE
Cenas	Crie e ative cenas no IO-Broker
medidores inteligentes	Interrogatório de medidores de eletricidade via protocolo SML ou D0
Sonos	Consultar e controlar jogadores do Sonos
telegrama	Comunicação com o aplicativo Telegram
TR 064	Controle / consulta de caixas AVM Fritz
VIS	Ferramenta de visualização para suas próprias interfaces de usuário
Zwave	Conexão à rede de rádio Z-Wave (pacote Openzwave)

Fonte: (HEINLE, 2017)

Entre as principais características do ioBroker, listada por Hek (s.d.) estão:

- Desenvolvido completamente em JavaScript
- Integração com ioBroker acontece através dos chamados adaptadores, os quais permitem conectividade com outros sistemas que estão rodando na rede.

Um exclusivo editor de painel de controle (supervisório) chamado “vis”. Na

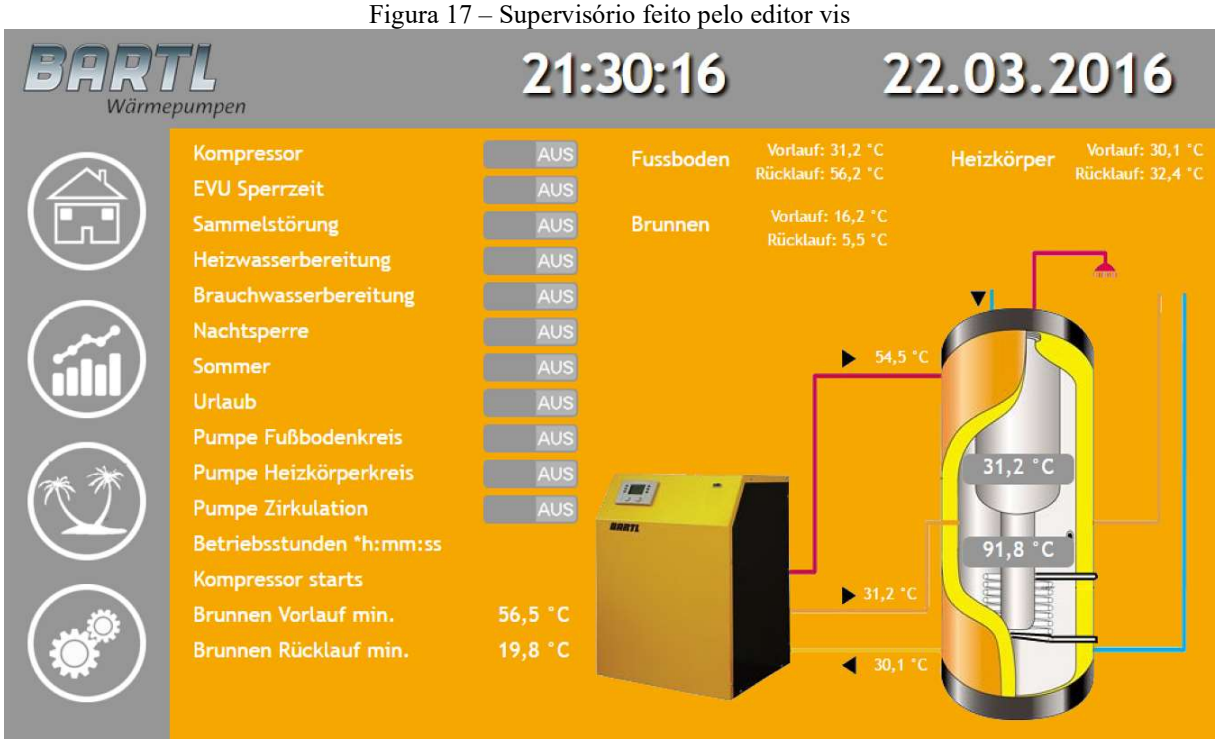
- Figura 17 encontra-se um exemplo de um painel de controle criado pelo “vis”.
- Pode lidar com o registro de eventos em muitos bancos de dados diferentes, como MS-SQL, MySQL, PostreSQL, InfluxDB, SQLite e apenas arquivos de texto simples.
- Integração nativa com o Node-RED com suporte a centenas de dispositivos e protocolos.
- Cenas e scripts em JavaScript
- Arquitetura extremamente modular com possibilidade de rodar módulos em diferentes computadores, mini computadores, NAS e tudo o que suporta Node.js.
- Suporte uso simples e fácil do protocolo MQTT.

- ioBroker pode rodar em qualquer sistema operacional que suporte Node.js como ARM, x86, Windows, Linux, OSX. O recomendado é o baseado em x86 ou ARM tais como RaspberryPi 2/3, BananaPi ou Cubietruck rodando o Linux Debian como sistema operacional.

- O sistema cria um novo processo Node.js para cada adaptador instanciado, portanto memória RAM torna-se um fator limitante. Um simples adaptador de impressão digital consome entre 10 e 60 MB.

- Grande e amigável comunidade em fórum.iobroker.net

Figura 17 – Supervisório feito pelo editor vis



Fonte: (HEK, s.d.)

2.2.5 Protocolo

2.2.5.1 MQTT

O protocolo MQTT (Message Queuing Telemetry Transport - Transporte de Telemetria de Enfileiramento de Mensagens) é um protocolo criado pela IBM na década de 90, com foco em aquisição de dados, mas evolui para conectividade M2M (Machine-to-Machine – Máquina à Máquina) e IoT, ou seja, é um protocolo para troca de mensagens entre máquinas e em 2014 se tornou um padrão aberto OASIS (COPPEN, 2018), passando a dar

suporte a várias linguagens de programação como Java, C, C#, Javascript, Python, entre outras (BARROS, 2015).

É um protocolo leve para trabalhar com redes baseadas na arquitetura de publicação-inscrição (HILLAR, 2017, p. 9). Também é simples com uma biblioteca para o cliente MQTT de baixo consumo de banda e processamento (CHEN, GUPTA, LAMPKIN, ROBERTSON, & SUBRAHMANYAM, 2014, p. xi) e pode trabalhar de forma assíncrona, ou seja, desconecta o emissor do receptor espaço-temporalmente abrindo a possibilidade de trabalhar de forma escalável mesmo em redes não confiáveis (Altus, 2018).

No seu funcionamento também pode-se notar simplicidade. As mensagens são identificadas através dos chamados tópicos que lembram o endereçamento da internet, com vários níveis separados por uma "/" como, por exemplo, em área/ID_da_area/sensor/ID_do_sensor/temperatura. A mensagem é enviada ao *broker* que atualiza a informação em seu banco de dados e envia essa informação para todos os dispositivos que estiverem inscritos naquele tópico. *Broker*, nessa estrutura, é o servidor do MQTT que tem sob a sua responsabilidade gerenciar as publicações, as autenticações e o controle das sessões que são iniciadas toda vez que um dispositivo se conecta a ele (BARROS, 2015). A segurança é provida tanto na encriptação dos dados que trafegam pela rede TCP/IP como através da autenticação que cada cliente pode ser obrigado a fazer para ter acesso *broker* utilizando-se um nome de usuário e senha certificados pelo padrão X.509. É um protocolo consistente, ágil, que demonstra uma preocupação com a segurança, altamente escalável, resistente a falhas e com uma excelente integração com outros sistemas (Carvalho, et al., 2018, p. 62).

2.2.6 Demais objetos usados

Há outras tecnologias que serão empregadas nesse projeto como *display* OLED, o aplicativo Autodesk AutoCAD, as linguagens de programação Javascript e C++ e o banco de dados MySQL, que são importantes, mas já são muito usados a um bom tempo, tendo farta documentação disponível. Outro fator que colaborou para a omissão de uma pesquisa sobre esses temas é a extensão do presente trabalho.

3 DETALHAMENTO DO PROJETO

3.1 Definição do Protótipo

Ao optar pela construção de um protótipo, é importante a definição do seu propósito, assim como o quanto esse protótipo refletirá um futuro produto final e quais serão os testes ao qual esse protótipo deve ser submetido, para avaliar se ele cumpre a função para o qual foi criado (Rachor, 2013, p. 7).

Para este trabalho de conclusão de curso, o propósito é verificar a possibilidade e a viabilidade de usar uma maquete virtual para auxiliar na projeção de um sistema de automação de iluminação e ventilação, assim como um controle de acesso às portas. Quanto a transformação do protótipo em um futuro produto, a expectativa é chegar o mais próximo possível, já que a viabilidade só poderia ser aferida mediante essa proximidade.

A avaliação da eficácia do protótipo será feita através da checagem das funcionalidades do sistema, onde cada item precisa ser testado e avaliado se cumpre o que está sendo proposto.

3.2 Funcionamento

O protótipo contará, primeiramente com uma modelagem em 3D de uma edificação que será inserida em um ambiente de jogo digital que possibilitará a locomoção nessa construção. O edifício escolhido é um templo religioso pertencente à Igreja Presbiteriana da Esperança, que está localizada na Rua Crispim do Amaral, 228, no bairro de Perus, noroeste da cidade de São Paulo, conforme a Figura 18. A edificação foi escolhida devido a facilidade de acesso, ser uma construção de dimensão razoável, mas de baixa complexidade para a implantação. Há um templo onde se pode projetar o acionamento das lâmpadas e dos ventiladores e onde o controle da potência das lâmpadas pode ser um recurso útil nas programações daquela comunidade religiosa, o salão de eventos também se mostra igualmente útil para a automação, principalmente se levar em conta o palco que ali existe. Além dessas duas construções, há um conjunto de salas que podem também servir para demonstração para a iluminação de ambientes menores e oferece a oportunidade de controle. seletivo de acesso a ambientes.

O projeto que está sendo desenvolvido poderia tomar como exemplo uma escola, ou algum prédio comercial, como um supermercado. Mas, além de haver uma complexidade maior, é provável que haja restrição de acesso por se tratar de lugares onde alguns ambientes são restritos ao público.

Figura 18 - Foto da Igreja Presbiteriana da Esperança

Fonte: Do próprio Autor

O protótipo contará com duas maletas de demonstração. A primeira maleta é a maleta de comando onde se encontra botões para seleção e acionamento das lâmpadas e dos ventiladores, além de um controle do tipo *slider* para controle da potência das lâmpadas e controle da velocidade dos ventiladores. Nessa maleta também estará alocado um leitor de RFID para liberação de abertura das portas dentro do ambiente virtual e da porta no ambiente físico. Haverá também *displays* para a indicação do estado das lâmpadas e ventiladores e para a visualização do *status* do *slider*.

Na segunda maleta estarão os dispositivos físicos que indicarão o funcionamento do sistema na vida real. Ela conta com sete lâmpadas LEDs dimerizáveis e com 7 ventiladores que mostrarão em tempo real a potência das lâmpadas controladas pelo sistema e o acionamento e a velocidade dos ventiladores. Contará também com uma pequena porta para se verificar o funcionamento do controle de acesso de porta do sistema.

A expectativa é que ao se acionar alguma lâmpada no ambiente de jogo, esta acenda ou apague nesse ambiente, mas haja a atuação em uma lâmpada de LED que está na maleta

dos atuadores e quando for feito o acionamento dessa mesma lâmpada na maleta de acionamento, a lâmpada de LED seja acionada na maleta dos atuadores e simultaneamente dentro do ambiente 3D.

Essa dinâmica permitiria um possível cliente ter o seu edifício modelado, dentro de um ambiente digital, onde ele pudesse percorrer todos os ambientes existentes e fazer o acionamento tanto da iluminação como da ventilação e verificar esse funcionamento tanto dentro do ambiente virtual como no mundo real. O sistema ainda permitiria que o mesmo cliente atuasse em botões e *slider* físico, podendo contemplar o acionamento tanto físico como virtual.

Dessa forma, o projeto permite ao cliente não somente ver como a automação funcionaria, mas também daria a ele a possibilidade de atuar nessa automação ainda na fase de projeto.

3.3 Arquitetura

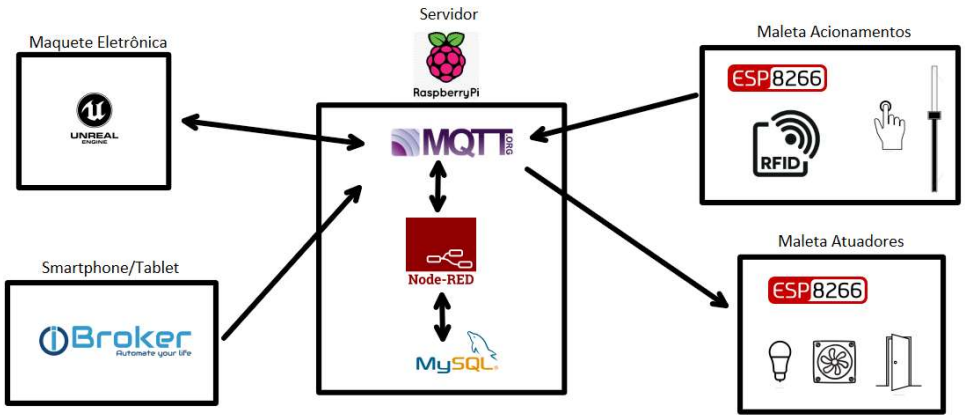
O sistema será composto de uma maquete eletrônica no ambiente de jogo digital chamado de Unreal Engine 4. Dentro desse ambiente será possível a exploração do ambiente e o acionamento virtual dos ventiladores, lâmpadas e portas de cada um dos ambientes disponíveis.

Como mencionado no tópico anterior, haverá duas maletas de demonstração. Uma para os acionamentos que contém os botões, o *slider* e o leitor de RFID e outra para os elementos que sofrerão a atuação do sistema no ambiente físico, composto pelos ventiladores, lâmpadas LEDs dimerizáveis e pela porta. Toda parte de controle será feita por uma placa eletrônica equipada por módulo ESP32, que também fará o acionamento e a leitura dos atuadores que permitem a comunicação WiFi. Os botões e o *slider* serão compostos por *touch* capacitivo e os status dos botões e do *slider* podem ser vistos pelos displays OLED.

O sistema também disporá de uma central que funcionará em um mini computador chamado de Raspberry Pi que terá função de servidor do protocolo MQTT, que se encarregará de intermediar a comunicação entre os itens do sistema, da central de processamento feita pelo software Node-RED, que processa todas as requisições, autentica e envia os comandos para que as atuações aconteçam. Terá também a base de dados para fazer a autenticação feita pelo sistema RFID. Acomodará ainda o software ioBroker, que será usado para o controle do sistema dentro da maquete eletrônica e feito através de tablet e

smartphones. A Figura 19 mostra de forma gráfica como se dá as interações no sistema. Como pode ser observado, a base das informações é feita pelo MQTT, que recebe informação do Unreal, da maleta de acionamento, processa essa informação através do Node-RED, devolve para o MQTT, que envia essas informações de volta para o Unreal e para a maleta de atuadores.

Figura 19 - Interações



Fonte: Do próprio autor

3.4 Tecnologias

O projeto que é objeto desse trabalho tem em sua estrutura diversas tecnologias (Centro de Ciências Exatas e Sociais Aplicadas, 2015, p. 8) que foram anexadas para o seu perfeito funcionamento. Para a escolha de cada uma dessas tecnologias foi usado, tanto quanto possível[4], critérios que otimizassem o tempo disponível, voltando-se para aquelas que tivessem a menor curva de aprendizagem e a disponibilidade de material para pesquisa.

3.4.1 Ambiente real

Antes de se voltar para as tecnologias usadas no presente projeto, é importante abordar a opção por um templo religioso. A automação empresarial acontece geralmente em

[4]A pesquisa se restringiu a duas ou três opções, pelo fator tempo para pesquisar outras opções, tendo-se já algum domínio na tecnologia escolhida, ou/e por falta de disponibilidade de material para pesquisa e ainda porque várias das opções dessas tecnologias precisam, para serem compreendidas de forma razoável, a instalação e o uso por algum tempo e, levando-se em conta a quantidade, as opções e o tempo disponível, em alguns casos não foi possível um estudo mais aprofundado para que, de todas as tecnologias disponíveis, a escolha recaísse na tecnologia mais eficiente e com um melhor custo/benefício, abrindo a possiblidade para uma pesquisa futura.

escritórios, *coworking*, auditórios e salões, como dito anteriormente (2.2.1 - Automação Corporativa), mas um templo religioso contém em sua estrutura muitos aspectos encontrados nos ambientes comerciais, como por exemplo, auditórios, salas de reunião, ambientes que funcionam como pequenos escritórios, palcos e no que diz respeito a iluminação e arrefecimento, as semelhanças podem ser ainda mais nítidas. Um ponto pelo qual esse trabalho se inclina para esse tipo de ambiente, o religioso, é que é um nicho muito pouco explorado no mundo da automação, mesmo com as semelhanças já destacadas.

Um outro fator foi a acessibilidade do imóvel usado como base para a modelagem. Houve fácil acesso as plantas do mesmo, assim como livre trânsito a todas as dependências do local o que permitiu obter um nível de detalhamento ímpar, fornecendo, dessa forma, uma maior precisão no ambiente 3D.

Além disso, o tamanho do imóvel (médio), juntamente com o conjunto de ambientes disponíveis fornecerão ao projeto várias possibilidades de automação, uma navegação pelo ambiente virtual que permita a exploração sem dispender tanto tempo, como no caso de uma construção muito grande, e a possibilidade de explorar-se um conjunto de soluções dando mais autenticidade ao projeto, já que se tem o imóvel completo no projeto.

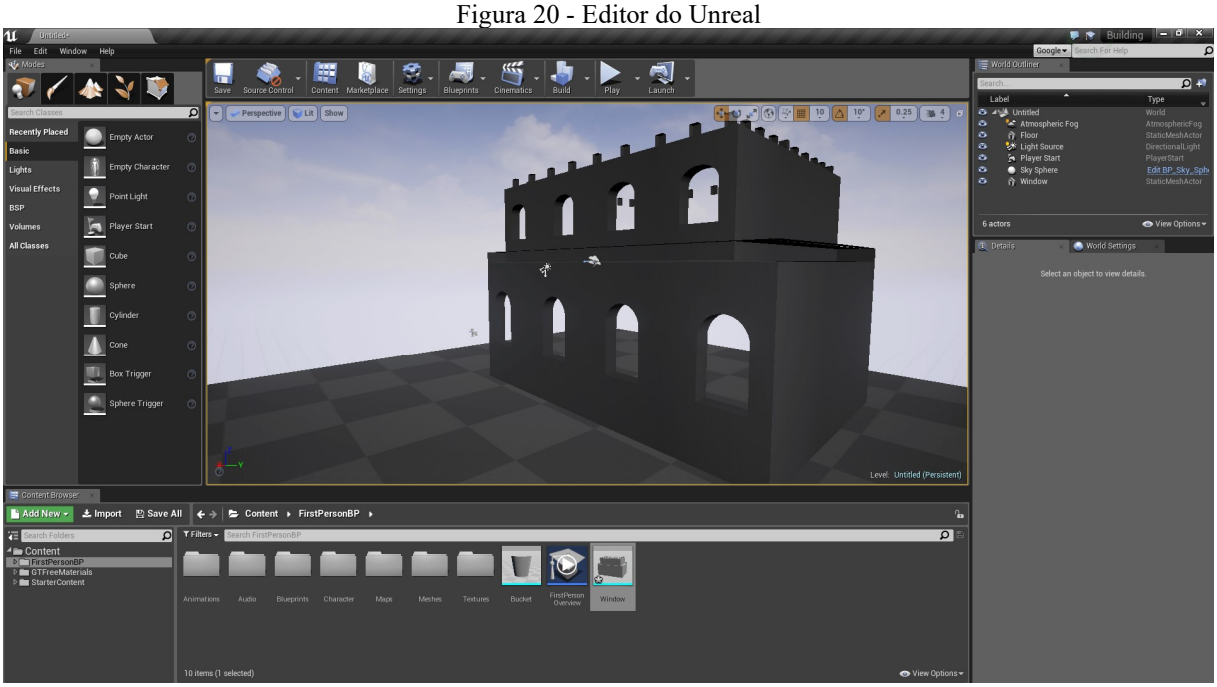
Além da iluminação e ventilação, pode-se explorar também o controle de acesso a vários ambientes diferentes, com distintos níveis de acesso e pode-se mostrar como a interface pode ser adaptada a cada ambiente.

3.4.2 Ambiente de Virtualização

No quesito ambiente de virtualização, a opção foi pelo motor gráfico Unreal Engine 4. Na pesquisa realizada, observou-se que a *engine Unity 3D* é mais frequentemente usada, mas quando o quesito é qualidade gráfica o Unreal consegue uma qualidade inigualável, chegando ao padrão de fotorrealismo onde, depois de usados vários recursos disponibilizados pela *engine*, para um observador desatento é difícil afirmar que aquele ambiente não é real (Soares, 2019). Um outro ponto que favorece o Unreal em detrimento da *Unity* é a linguagem de programação. A *Unity* trabalha com a linguagem C# da Microsoft, enquanto a Unreal, que originalmente trabalhava somente com a linguagem C++, oferece atualmente a linguagem de programação chamada Blueprint, que é baseada no C++, mas é uma linguagem gráfica que tem como primícias ser uma linguagem de fácil aprendizagem e com um paradigma orientado ao objeto que facilita muito a vida do desenvolvedor, possibilitando uma excelente

produtividade. Muitas das funções vêm em pequenos blocos que são facilmente ligados ao restante da programação. Abaixo encontra-se o editor da *Unreal* (Figura 20) e o da *Unity* (Figura 21).

Figura 20 - Editor do Unreal



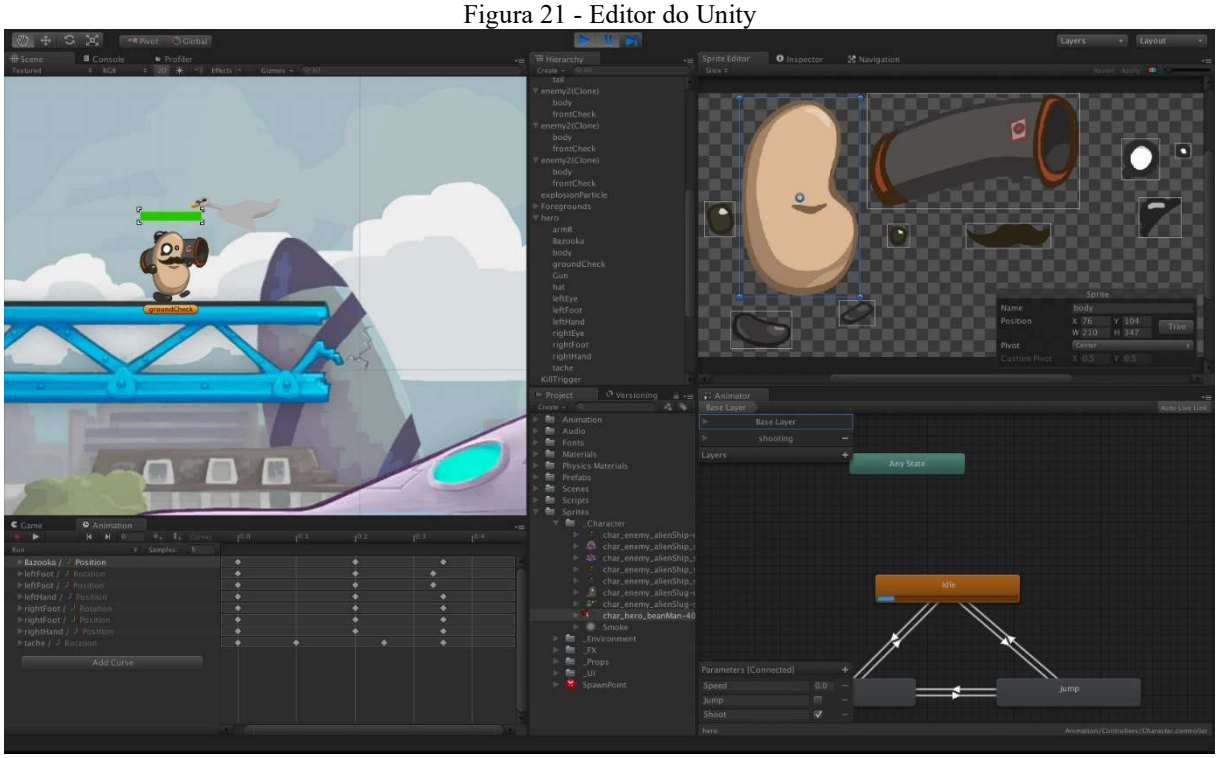
Fonte: (Soares, 2019)

3.4.3 Servidor de Aplicativos

Outro item que foi inserido no projeto foi a mini computador Raspberry Pi. Ele é responsável pela execução de todos os servidores: MQTT, Node-RED, MySQL e iobroker. Há outros mini computadores na atualidade, mas o Raspberry Pi continua tendo o melhor custo/benefício, um sistema operacional feito especialmente para ele que é constantemente atualizado baseado em Linux, que dá acesso a uma infinidade de recursos vindo do ecossistema do Linux Debian, uma das distribuições mais confiáveis do mundo Linux que coloca à disposição do usuário várias linguagens de programação, como Python, C++, Javascript, entre outras, e a possibilidade de instalar servidores como os citados acima. Com o seu tamanho reduzido (25 mm x 20 mm x 9 mm), baixo consumo de energia (em torno de 800mA com picos de 950mA com uso dos recursos moderado (EverPi, 2017)) é possível colocá-lo em uma pequena caixa e oculta-lo em qualquer parte do ambiente. As últimas versões do Raspberry já vêm com WiFi embutido, mas opta-se por usar um pequeno roteador

devido a limitação do Raspberry em ser usado como roteador para conexão de vários clientes simultâneos.

Figura 21 - Editor do Unity



Fonte: (Soares, 2019)

3.4.4 Comunicação

O MQTT entrou no projeto para tornar mais eficiente a troca de informações nos vários dispositivos. A tentativa anterior era composta pela comunicação via TCP entre o unreal e o Raspberry Pi, Socket.io entre o Raspberry e os *smartphones* e rádio frequência em 433Mhz entre o Raspberry e a parte eletrônica do projeto. Para manter todos sincronizados as informações sobre os status de todos os dispositivos (lâmpadas, ventiladores e portas) era usado o banco de dados MySQL que trabalhava com um *procedure*. Assim que algum comando era recebido pelo sistema, este transmitia uma *string* com uma variável de 256 caracteres que era decodifica em cada um dos dispositivos. Ao estudar o protocolo em uma iniciação científica (Carvalho, et al., 2018, p. 62) ele se mostrou um substituto único para as tecnologias usadas até então com um ganho substancial, não somente em eficiência, mas também em simplicidade, segurança, rapidez e versatilidade.

3.4.5 Processamento das informações

A opção pelo uso do Node-RED se deu inicialmente pela facilidade de comunicação com vários tipos de protocolos, que em um primeiro momento dava ao projeto versatilidade e ampliava as possibilidades nesse quesito, caso o MQTT apresentasse alguma limitação, ou se não fosse possível a comunicação com algum dos dispositivos usando o MQTT. Outro ponto que pendeu favoravelmente para o Node-RED é a programação em blocos, parecida com o Blueprint do Unreal que se mostrava mais simples com uma curva de aprendizado menor, baseada em uma linguagem com farto material disponível, que é o Javascript, o que permite uma programação hibrida, ora por blocos, ora procedural, além de já vir com o servidor instalado no Raspberry Pi.

3.4.6 Interface para smartphone/tablet

Como o projeto é composto por várias partes que precisam ser desenvolvidas, abdica-se da possibilidade de usar uma interface baseada em HTML/CSS, pois isso iria demandar tempo para o desenvolvimento das telas nessa linguagem, além de se optar por uma linguagem de programação *back-end* com um servidor web. Então a opção foi um aplicativo de automação residencial que “conversasse” com o MQTT, nativamente e que fosse simples para montar telas para interagir com o usuário. Dentre as opções como o openHAB, o iobroker, apesar de não ser tão popular e ter uma literatura restrita (inglês e alemão), se mostrou promissor devido a poder funcionar como cliente e servidor do MQTT. A configuração é relativamente fácil e também se mostra simples para a produção das telas de smartphones e tablets, além disso, há a possiblidade de instalação do servidor no Raspberry Pi.

3.4.7 Microcontrolador

A parte microprocessado será controlada pelos módulos de desenvolvimento NodeMCU, que é uma placa de desenvolvimento (HU Infitivo, 2019), equipado com o microcontrolador ESP8266 e um módulo ESP32, em detrimento do uso de algum tipo de arduino por um motivo simples, mas que faz toda a diferença: a presença de comunicação via WiFi através do protocolo TCP/IP (Bauermeister, 2018). Com a possibilidade de uso da

linguagem C++ voltada para arduino, a própria IDE da plataforma e, contando com uma grande quantidade de bibliotecas disponibilizadas, o seu uso é simples, mas com uma potencialidade extraordinária. O uso dos kits de desenvolvimento (ESP8266) e o módulo ESP32 permitiu a conexão do presente sistema à parte física ou eletrônica do projeto como os botões e *sliders* de acionamento e as lâmpadas LEDs e os ventiladores. Outra vantagem no uso desses módulos é o seu tamanho compacto, baixo consumo de energia com no máximo de 200 mA e com 13 GPIO disponíveis (Murta, 2018).

3.4.8 Base de dados

O uso de banco de dados é bem restrito e está ligado ao controle de acesso usado pelo RFID para que ao se passar o cartão, tanto o físico como o virtual, o sistema identifique o usuário e acesse quais são os ambientes que ele tem acesso. A escolha recaiu sobre o MySQL por ser uma base de dados de uso comum com farto material na internet, de fácil instalação no Raspberry Pi, e como o uso é de baixa intensidade, o consumo de recursos do sistema é equivalente.

4 DESENVOLVIMENTO DO PROTÓTIPO

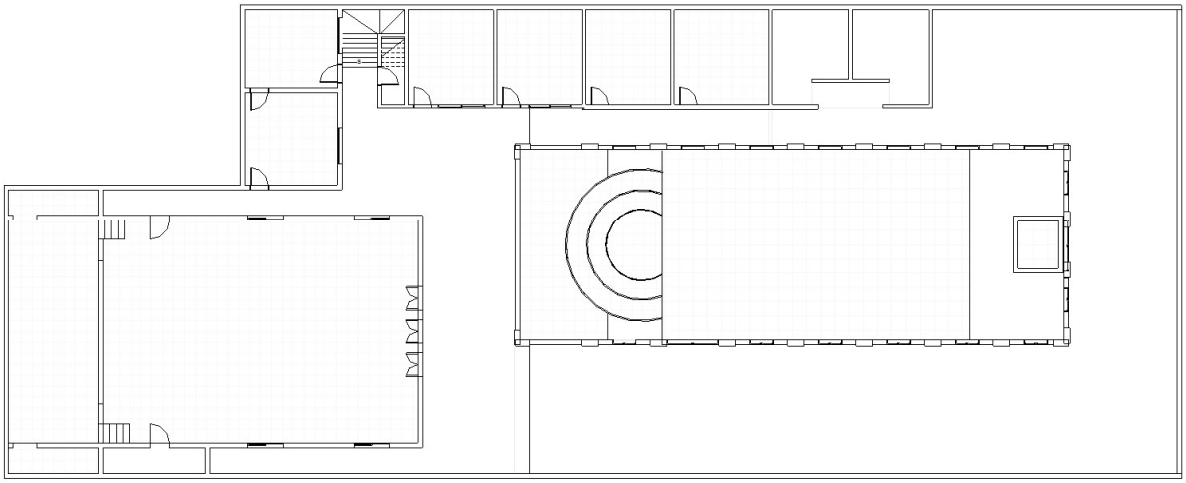
4.1 Desenvolvimento do Ambiente Virtual

O desenvolvimento do ambiente virtual procurará seguir os passos que seriam dados em um projeto real para poder-se analisar a viabilidade de um possível modelo de negócio onde, um cliente final ou uma empresa de automação corporativa, solicita um projeto virtual, mas com a possibilidade de interação com controles e atuadores reais que poderão interagir entre si e dar ao cliente uma imersão não total, mas com detalhes que permitam ao mesmo sentir como ficaria esse projeto de automação já implantado.

4.1.1 Autocad

Dentro dessa perspectiva, o primeiro passo é a obtenção das plantas do imóvel onde se deseja implantar o sistema de automação. Na Figura 22, encontra-se a planta que foi passada para o Autodesk AutoCAD. Isso ajuda a, desde o início, manter as proporções reais do projeto.

Figura 22 - Planta da baixa da igreja



Fonte: do próprio autor

A planta baixa feita em Autocad também é usado com base para o desenvolvimento do projeto em Revit que pode ser apenas importado ou pode ser vinculado para que, havendo

alguma alteração no Autocad, essa mudança se reflita no desenho em 2D que está no Revit (Netto, 2018).

Uma outra atividade que é necessária no desenvolvimento da modelagem, é ir ao local e tirar fotos de todos os detalhes da arquitetura do edifício, pois no desenvolvimento da modelagem, são as fotos que darão as particularidades de cada parte da edificação e ajudará a aproximar o modelo virtual do real. Se possível, é importante em algumas dessas fotos ter-se as medidas desses detalhes que ajudam na modelagem no Revit, mas principalmente na modelagem mais fina feita no 3DS Max. Esboços manuais com as medidas também podem ajudar.

4.1.2 Revit

O Revit é uma ferramenta que pode ser de extrema utilidade na montagem do modelo virtual quando o objetivo final é um projeto para automação. Isso porque, além da modelagem inicial que está relacionada a arquitetura, o mesmo também oferece a possibilidade do desenvolvimento do projeto elétrico, que se somando ao projeto arquitetônico permite, em uma apresentação à um cliente, verificar as implicações nas instalações de sua propriedade (Vídeo 2), não somente da visualização de toda a estrutura elétrica, como o fornecimento de uma lista de material e, se devidamente alimentado com os valores desses materiais, o fornecimento do orçamento dessa área do projeto.

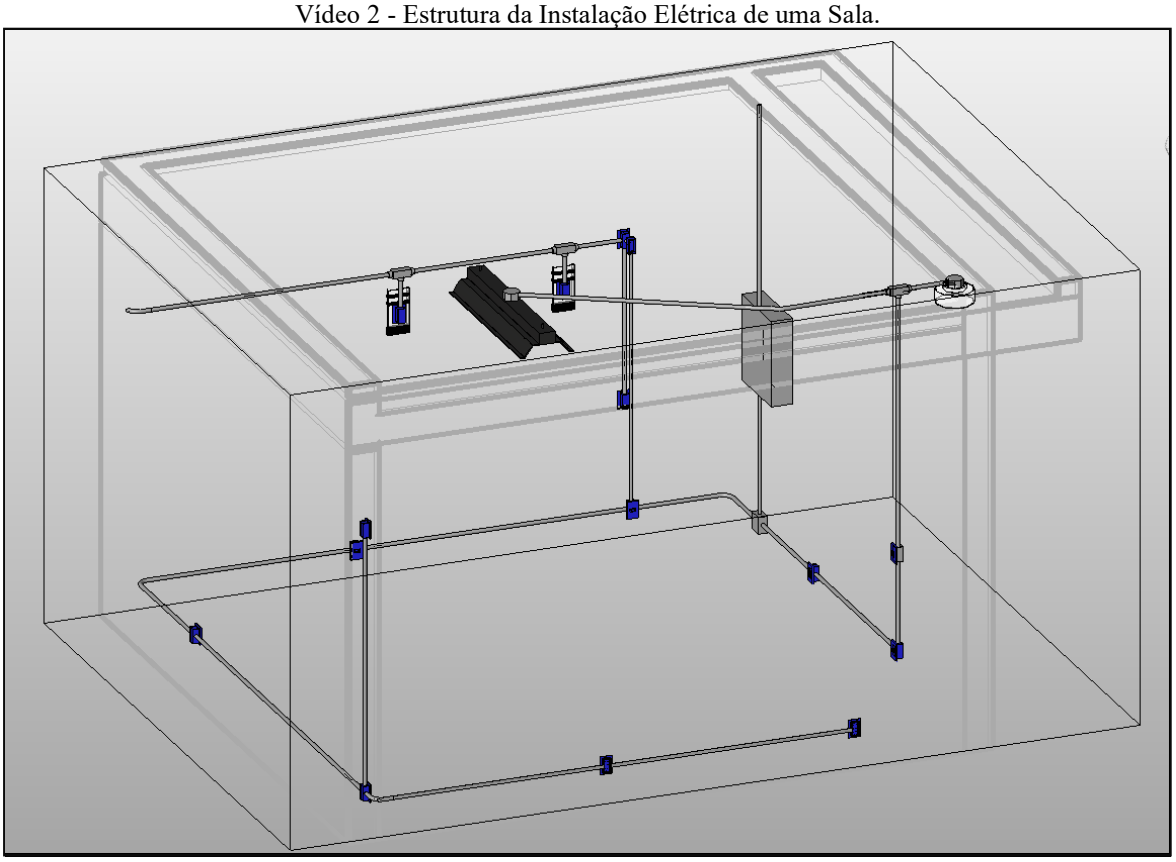
No presente projeto, como a intenção inicial era oferecer ao cliente um *retrofit* das instalações elétricas, a modelagem desenvolvida no Revit contém o projeto elétrico que seria oferecido para os líderes da igreja, pois para a automação da iluminação e da ventilação seriam necessárias algumas adequações na parte elétrica.

Outra vantagem que o Revit proporciona, mas que não foi contemplado nesse trabalho, já que local trabalha com o arrefecimento feito por ventiladores, é a possibilidade de cálculo da carga térmica dos ambientes de uma construção. Como se está lidando com um ambiente que comporta um grande número de pessoas, e que dependendo da taxa de ocupação essa carga térmica pode ter uma grande variação, é possível ter no projeto de automação um conjunto de valores pré-determinados para que essa variação seja usada pelo sistema para o controle automático do ar-condicionado.

Na Figura 23, encontra-se a modelagem do templo religioso usado nesse trabalho. A modelagem não tem muitos detalhes, mas mostra toda a parte estrutural da edificação,

servindo perfeitamente com a modelagem que será enviada para o ambiente de jogo, o *Unreal Engine*.

Vídeo 2 - Estrutura da Instalação Elétrica de uma Sala.



Fonte: do próprio autor

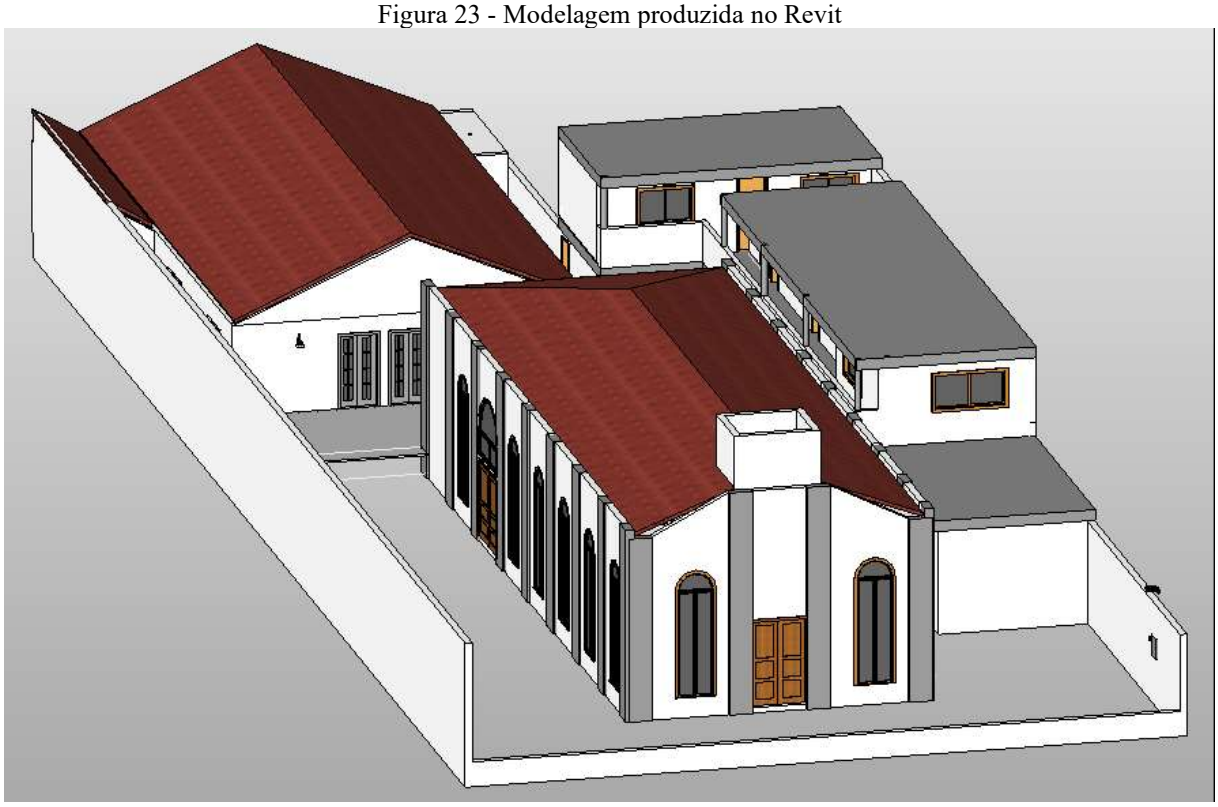
4.1.3 3DS Max

O Revit entrega a estrutura básica da modelagem, mas ele não traz entre as suas ferramentas algo que permita um detalhamento mais preciso da arquitetura do prédio objeto do projeto de automação. Para essa modelagem mais "fina" é usado o aplicativo 3DS Max que permite dar um nível maior de detalhamento.

O projeto feito no Revit então precisa ser exportado para o 3DS Max, onde todas as partes da modelagem que precisam ser refinadas são excluídas e refeitas, como é o caso, nessa edificação em específico, das portas, janelas, colunas, da fachada, e revestimento interno, entre outros. Isso é feito, pois entre as bibliotecas de objetos, dificilmente se consegue encontrar objetos que correspondam de forma exata ou muito próxima ao objeto real. Na Figura 24 é vista uma porta encontrada na biblioteca do Revit à esquerda, a porta modelada

(ainda sem a cor final que será colocada no Unreal) no centro e a foto da porta da frente do templo. Infelizmente a imagem não permite mostrar de fato o grau de semelhança entre a modelagem e o objeto real, mas percebe-se que, em relação à porta que foi encontrada no Revit, ela está mais próxima do real.

Figura 23 - Modelagem produzida no Revit



Fonte: do próprio autor

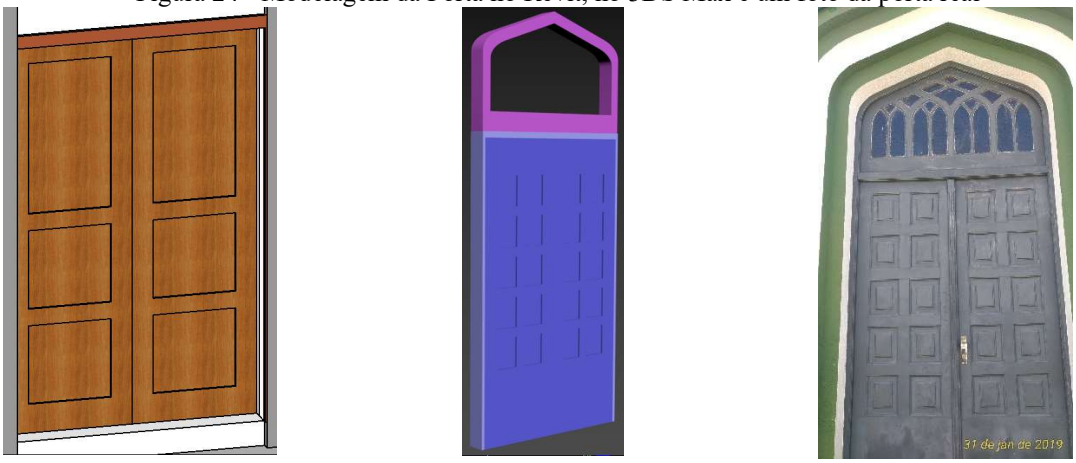
Nessa etapa do processo, também é necessário identificar as faces dos objetos que receberão mais de um material, colocando em cada um desses locais uma identificação, para que, por exemplo na pintura, cada parte possa receber um material/cor distinta. Um exemplo desse cuidado é feito com as paredes que precisam receber uma cor interna e uma cor externa.

O que se aproveita do Revit são os objetos que tem uma geometria simples, como as paredes, algumas colunas, o telhado, que geralmente não tem nenhum detalhamento específico que necessite ser destacado.

Depois de toda a modelagem dos objetos terem sido refeitas, é remontado o projeto e preparado para a exportação para o ambiente de jogo (*Unreal*). Isso pode ser feito através da conversão do projeto para o formado *.fbx que é maciçamente usado por aplicativos 3D como o próprio Revit e 3DS Max, Maia, Blender, entre outros. Na Figura 25 pode-se ver o projeto

finalizado nessa etapa da modelagem e pronto para ser exportado para a última etapa que é colocá-lo em um ambiente de jogo.

Figura 24 - Modelagem da Porta no Revit, no 3DS Max e um foto da porta real



Fonte: do próprio autor

4.1.4 Unreal Engine

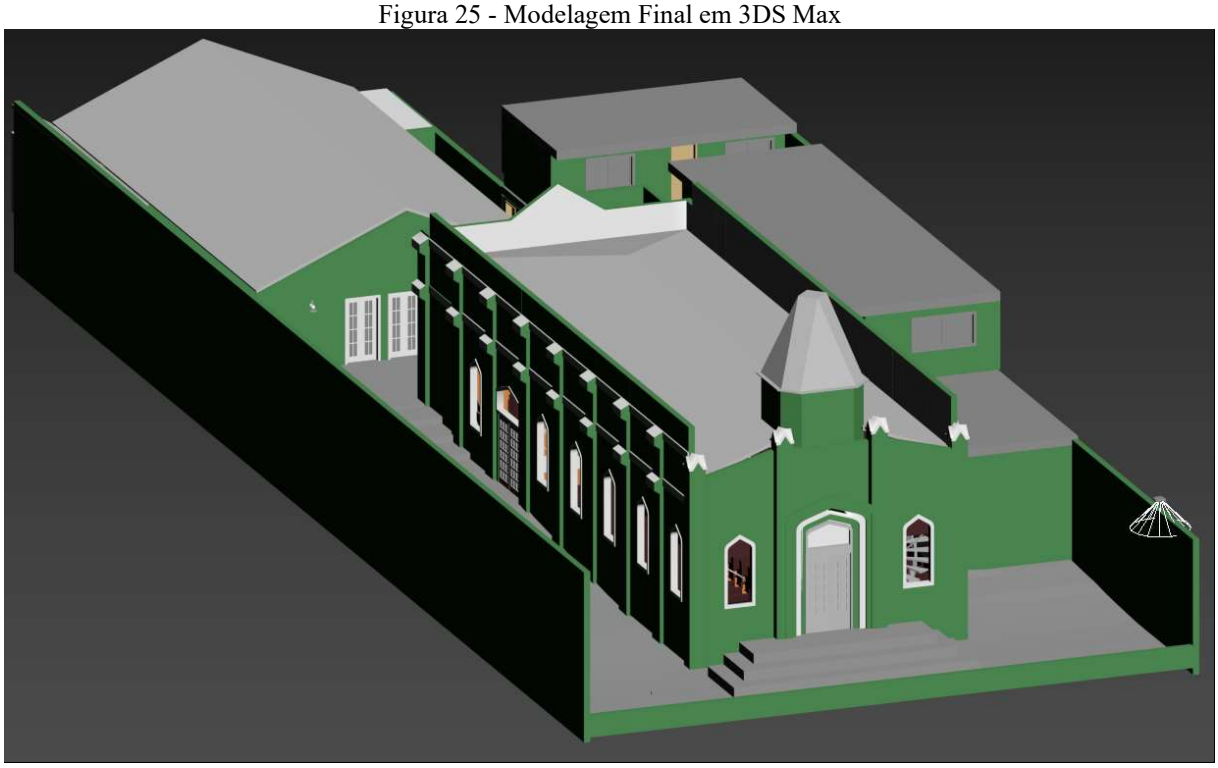
Nessa etapa, introduzida a modelagem no ambiente de jogo, é necessário implementar as animações que farão par com os dispositivos reais: a iluminação, a movimentação dos ventiladores e a abertura de porta. Na questão da iluminação, o *Unreal Engine* dispões de um conjunto de ferramentas muito bem elaborado que permite um nível de simulação muito sofisticado e preciso, como temperatura da cor da luz, controle da intensidade, opções em omnidirecional ou direcional, permitindo que o uso normal da iluminação dos ambientes do dia-a-dia sejam reproduzidos de forma autêntica sem maiores dificuldade.

Os ventiladores, assim como as portas têm a sua animação através da movimentação de algumas de suas partes. Por isso, é necessário fazer a modelagem dessas partes e importá-las de forma diferenciada e fazer a sua montagem usando também programação para que se tenha os seus movimentos definidos.

Também é nessa última etapa que os materiais dos objetos foram aplicados, tais como a tintura das paredes, os acabamentos em alumínio e em vidro, assim como a madeira e os pisos. É possível fazer essa aplicação tanto no Revit quanto no 3DS Max, mas a cada exportação, mesmo utilizando o Datasmith, percebe-se algum tipo de degradação fazendo-se necessário alguns retoques. Além disso, para se ter melhores resultados é necessário um conhecimento mais profundo em cada um dos softwares citados, algo que vai além do escopo proposto para este trabalho. E com as animações feitas e testadas, a etapa da modelagem é

encerrada. No Vídeo 3 é mostrado um passeio virtual no projeto para comparar com algumas fotografias reais do imóvel que foi usado como modelo para o protótipo do presente trabalho.

Figura 25 - Modelagem Final em 3DS Max



Fonte: o próprio autor

4.2 Desenvolvimento da Central de Comando

A central de comando, no presente projeto, tem a função de centralizar o status de cada dispositivo que será usado no protótipo, além de ficar responsável por todo o processamento que será realizado no sistema. Essa centralização permite colocar a complexidade da programação em um único lugar, possibilitando que esta seja feita uma única vez, independentemente da quantidade de controles, atuadores e dispositivos que sejam anexados ao sistema, trazendo assim flexibilidade na adição de futuros dispositivos, controles e atuadores. A centralização das informações ajuda também a manter todo o sistema integro, de forma assíncrona, facilitando que diferentes controles possam fazer parte do sistema, não tendo restrição no uso de linguagem de programação, podendo ser baseada em Web, mobile, eletrônico ou dentro do próprio jogo. Todos funcionam de forma sincronizada.

4.2.1 Instalações

O hardware da Central de Comando é constituído basicamente do mini computador Raspberry Pi em sua terceira geração, juntamente com um mini roteador wireless. O Raspberry Pi pode funcionar como um ponto de acesso, mas a disponibilidade para receber conexões wireless dos outros dispositivos é baixa, em torno de quatro ou cinco dispositivos, o que torna o seu uso nesse projeto inviável, já que toda a comunicação será feita através de rede sem fio. Por isso optou-se por roteador dedicado (conforme Figura 26) que, apesar de seu tamanho reduzido, tem recursos que vão muito além da necessidade do sistema que está aqui sendo desenvolvido.

Vídeo 3 - Modelagem 3D no ambiente do Unreal Engine.

Fonte: Do próprio autor

Esse hardware tem como missão abrigar todos os servidores que serão usados pelo sistema. Além do próprio sistema operacional, ele abrigará o servidor do MQTT (Broker), do Node-RED, do ioBroker, do banco de dados relacional MySQL.

4.2.1.1 Sistema Operacional

Apesar da possibilidade de trabalhar com o sistema operacional da Microsoft, que é desenvolvido especial para esse equipamento, voltado para IoT, o poder do Raspberry é melhor aproveitado no sistema Linux chamado Raspbian, uma distribuição Linux baseada em

Debian, conhecida por sua estabilidade e pela enorme quantidade de pacotes (programas) oferecidos (Brito, 2015). Nessa distribuição, já vem nativamente o servidor Node-RED, além de vários outros aplicativos, todos gratuitos, incluído alguns que em outros sistemas são pagos como o Aplicativo Matemática, desenvolvido pela Wolfram. Como o Raspbian é completo, além dos aplicativos já instalados, pode-se instalar qualquer aplicativo que esteja disponível no Debian, como será o caso do banco de dados MySQL.

Figura 26 - Mini roteador.

Fonte: Do próprio autor

4.2.1.2 Servidor ioBroker

O aplicativo ioBroker, que servirá nesse projeto como base do *Broker* do MQTT, atuará tanto como servidor como cliente no Node-RED, sendo possível iniciar, pausar e parar este servidor através da interface gráfica do ioBroker e também, será usado para desenhar a interface de controle, que será usado nos dispositivos móveis e também dentro do ambiente do jogo. Com este *modus operandis* consegue-se unificar a configuração e uso de diversas ferramentas (comunicação, programação e interface com o usuário) tornando o seu uso mais simples e intuitivo.

4.2.1.3 Servidor Node-RED

O Node-RED tem sob a sua responsabilidade receber as requisições de acionamentos e, processando requisições, devolver uma resposta ao solicitante. Essa intermediação é importante pois, como se pretende trabalhar com algum nível de segurança, ela permite que o sistema faça uma validação para saber se o solicitante tem acesso ao recurso que está solicitando. Isso acontece principalmente com relação a abertura das portas que serão liberadas por um cartão RFID, tanto real como virtual. Também, em algum momento, os

acionamentos das luzes e dos ventiladores ficarão restritos a algum grupo que seja especialmente designado para isso, para evitar que qualquer pessoa tenha acesso ao sistema e faça acionamentos não autorizados.

O controle dinâmico da interface, explicado com um pouco mais de detalhes mais adiante, também será feito pelo Node-RED, ou seja, ele fica com toda a parte de processamento e validação, permitindo que os outros dispositivos apenas enviem as informações e receba os comandos para executarem as atuações que lhes forem designadas. Então, é aqui que ficará a parte mais densa da programação que será feita através de blocos gráficos, mas que também permite que muita coisa seja feita em Javascript, dando fluidez ao desenvolvimento e legibilidade ao código.

4.2.1.4 Servidor MySQL

Com o uso do MQTT, as informações de status de todos os dispositivos ficam armazenadas no próprio *Broker*, o servidor do MQTT. Entretanto, é necessário uma base de dados, que em nesse projeto, tem a função de manter os dados dos usuários do sistema para fins de autenticação de acesso. É nele que será cadastrado os dados dos usuários com as suas respectivas permissões, assim como o acesso/restrição de cada ambiente. Apesar de não estar com previsão de implantação nesse estágio do projeto, o banco de dados ainda armazenará o conjunto de informações sobre as cenas que adequarão o ambiente para diversas ocasiões, tanto na abertura da igreja, onde o sistema, dependendo do usuário, do dia, e das programações, dará acesso aos ambientes necessários, tornando-os públicos, ao passo que manterá outros com acesso restrito, como também configurará o ambiente do templo e salão para ocasiões específicas, com mais ou menos luminosidade para tornar o ambiente mais propício para cada ocasião.

4.2.1.5 Configurações

Para que o sistema possa funcionar, é necessário a configuração do MQTT através da criação dos chamados tópicos, que são enviados por um cliente ao broker, e enviada para cada cliente que estiver inscrito naquele tópico. Cada dispositivo acionado, enviará através dos tópicos a sua solicitação e como foi dito acima, o Node-RED processará essa informação e devolverá para todos os dispositivos envolvidos a ação a ser executada. Então nesse momento é necessário configurar todas as interfaces (Unreal, ioBroker, Node-RED e a parte eletrônica)

como clientes MQTT e preparar cada ambiente para enviar e receber informações através desse protocolo.

4.2.2 Criação das Interfaces

Como está projetado, para que haja um único controle para atuar em todo o sistema, que pode ser replicado em vários dispositivos (web, móvel ou eletrônico), é importante que durante a navegação do edifício, o sistema identifique onde o usuário está, e lhe dê os controles daquele ambiente, e isso precisa ser feito de forma dinâmica. Esse controle, de receber o ambiente atual do usuário, e lhe entregar os controles desse ambiente, também é feito pelo Node-RED, que terá a função de reorganizar os controles na interface, em vista da atual posição do usuário. Por exemplo, quando o usuário estiver em frente a porta de uma das salas, ele terá os controles para abrir a porta daquela sala, assim como controlar a sua iluminação e o seu ventilador. Quando, saindo dali, for para umas das portas do templo, a interface do usuário adequará os controles para o templo, podendo o usuário abrir as suas portas, controlas as luzes e os ventiladores do templo.

4.2.2.1 Interfaces

Em um primeiro momento, houve a idealização de se construir uma interface para cada plataforma, no vislumbre de personalizar o máximo possível cada uma dessas interfaces e trabalhando o máximo para torná-las tão intuitivas e atraentes quanto possível. A ideia era no tablet desenvolver uma interface feita no próprio Unreal, aproveitando partes gráficas do próprio projeto, como alinhamento do design, pensando em usar essa tela como acionamento do sistema no ambiente real do templo e do salão, algo parecido com o ilustrado na Figura 27.

A interface para o *smartphone* poderia ser mais simples, podendo ser desenvolvido um aplicativo para android, ou fazer o uso da interface oferecida pelo Node-RED, conforme a Figura 28, que é muito funcional, ou ainda, ir na direção da interface do ioBroker, que também apresenta em seus exemplos, interfaces bem arrojadas, conforme se vê na Figura 29. Além disso, seria necessário o desenvolvimento de uma interface para cadastro e manutenção dos dados do usuário para o já referido controle de acesso, podendo fazer esse sistema em alguma linguagem da plataforma web, usando como *front-end* o próprio html/css juntamente com Javascript e *Jquery*, com o próprio Javascript como *back-end* ou fazer uso da linguagem de programação *Ruby on Rails*.

.

Figura 27 - Exemplo de interface no ambiente real

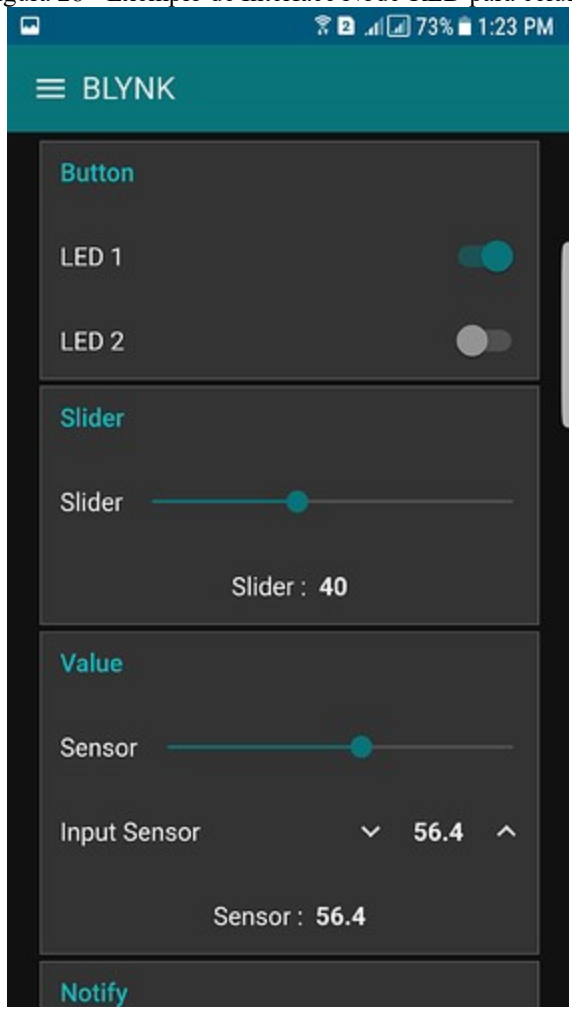
Fonte: (Mapa da Obra, 2018)

Nos testes iniciais para o desenvolvimento do projeto, foi usado, como experimentação, a interface do ioBroker, que devido a sua integração com o servidor e cliente do MQTT, se mostrou bem simples e fácil de ser implementada. No decorrer do processo, verificou-se algumas dificuldades de se implementar algumas de suas características, mas nada que comprometesse o seu uso.

Foi testado o uso da Unreal, para o desenvolvimento da interface que se mostrou realmente muito versátil no design, mas infelizmente, o plugin do MQTT não funcionou quando usado para produzir o aplicativo para android. Já no caso do Ruby on Rails, apesar de se mostrar uma linguagem de programação simples, existe uma curva de aprendizado que precisa de investimento de tempo, o que nesse momento é inviável. A interface do Node-RED, por sua vez, se mostrou com algumas limitações, que poderiam comprometer um design minimamente funcional, devido à complexidade que o sistema impõe para apresentar os controles de forma dinâmica, como está proposto. Diante disso, mesmo com as limitações encontradas, a opção para o desenvolvimento da interface recaiu sobre o ioBroker. É possível uma sensível melhoria no que foi produzido, mas o que até então foi implementado, se

mostrou funcional para a demonstração do conceito tratado e desenvolvido pelo presente projeto.

Figura 28 - Exemplo de Interface Node-RED para celular



Fonte: (Blynk.Community, 2018)

4.2.2.2 Interface para *smartphone*

O desenvolvimento da interface para *smartphone* foi uma evolução da interface que foi usado para testar as funcionalidades iniciais do projeto, e se mostrou plenamente adequada para o dinamismo dos controles relacionado aos diversos ambientes da construção. E durante os testes, foi constatado que o mesmo poderia ser usado para fazer os acionamentos dentro do ambiente virtual. Isso possibilitaria a diminuição da complexidade dos controles dentro do ambiente de jogo, que antes estava planejado para ser feito por meio do teclado, sendo necessário usar um número razoável de teclas, e acabava por criar um padrão único entre a interface usada dentro da maquete eletrônica (virtual) e o *smartphone* (real). Aproveitando é primeira uniformidade, o layout da maleta teve o seu design direcionado para o mesmo do

smartphone, criando assim uma identidade única nos controles da automação dentro do projeto. É possível ver na Figura 30, a interface que o projeto usa, tanto para os *smartphones*, como o personagem dentro do Unreal.

Figura 29 - Exemplo de interface do ioBroker

Fonte: (KLEEN, 2019)

4.2.2.3 Controle dinâmico

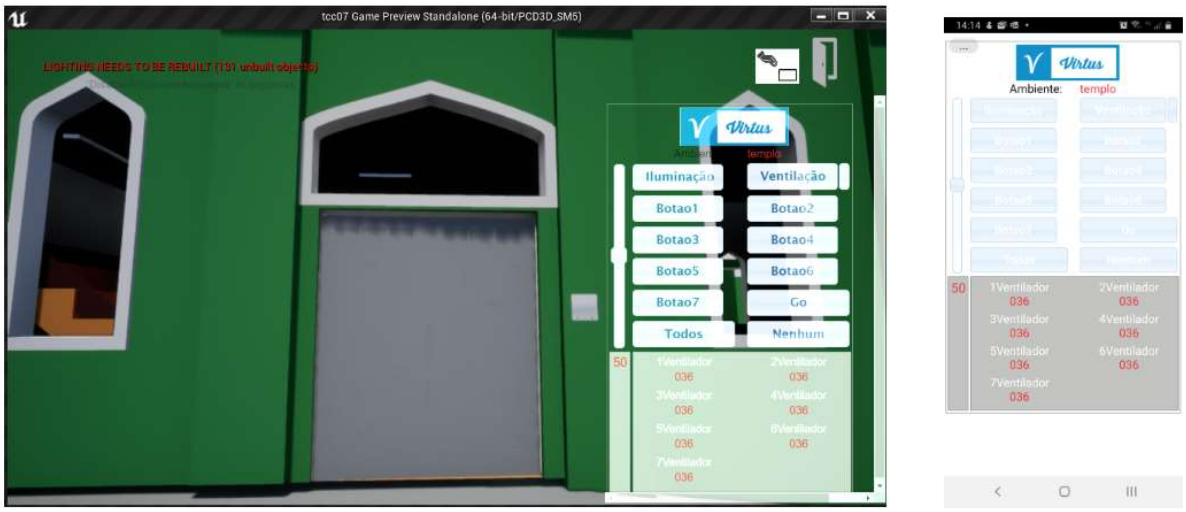
Um dos desafios no desenvolvimento deste projeto, foi gerenciar todos os controles de todos os ambientes em todas as plataformas. A dificuldade estava no fato de haver 10 ambientes diferentes com diferentes configurações, indo do templo, com o controle de 7 lâmpadas ou conjunto de lâmpadas, que permitisse não somente o ligá-las e desligá-las, mas também, o seu controle de potência, o controle dos 7 ventiladores incluindo também, a sua potência, e a abertura de 2 portas, até o menor ambiente, que seria o depósito de materiais de limpeza, que conta com apenas uma lâmpada. Criar um controle para cada um dos dispositivos, se mostrou inviável pela quantidade e diversidade de dispositivos, e limitaria de forma significativa as plataformas de controle, principalmente os controles físicos (botões e *sliders*).

A solução encontrada, foi criar um único conjunto de controles, que abrangesse todos os controles, desde o ambiente mais crítico quantitativamente, o templo com sete controles, e mesmo assim, abarcando apenas um tipo de controle, iluminação ou ventilação, colocando

uma chave seletora para direcionar esses controles para um ou outro tipo (conforme indicado na Figura 30), um único *slider* e um botão para execução de comandos.

O funcionamento se daria da seguinte forma: primeiro seria necessário selecionar se os controles estariam voltados para a iluminação ou para a ventilação através do seletor. Ao fazer a seleção todos os controles seriam direcionados para esse tipo de dispositivo. Em seguida, o usuário precisaria escolher qual(is) dispositivo(s) seria(m) acionado(s). Fazendo uso do *slider*, se escolheria a potência dos dispositivos e finalmente executaria as ações selecionadas através do botão de execução.

Figura 30 - Interface para smartphone e no Unreal.



Fonte: Do próprio autor

O ambiente, seria detectado pelo personagem virtual, ao se posicionar em frente a porta desse ambiente, e ao fazer isso, os controles da interface passariam a controlar os seus dispositivos, e mudariam a quantidade de controle, para se adequar a quantidades de dispositivos daquele ambiente.

Assim, centralizando todo o processamento dessas informações no Node-RED, foi possível fazer com que as interfaces das plataformas móveis, a parte eletrônica e a interface dentro do ambiente virtual, se comportassem como terminais, que apenas enviam requisições, e tanto o Unreal como os dispositivos físicos (lâmpadas de LED e ventiladores), recebessem os comandos para acionar os dispositivos do ambiente onde o personagem virtual se encontrasse.

Após a codificação e testes preliminares o conceito se mostrou funcional, permitindo o dinamismo necessário ao sistema, assim como também se mostrou versátil, pois permite que

qualquer sistema que trabalhe com MQTT possa ser introduzido de forma relativamente simples, com o mínimo de programação. Dessa forma, pode-se incluir, em um futuro próximo, interfaces feitas em *Android*, *Node-RED*, *LabView*, Elipse Scada (Damin, 2019), *openHAB* (Graeff, 2018), entre outros.

4.2.2.4 Interface para tablet

A interface para tablet foi designada para servir como controle real, tanto do ambiente no templo, como no salão, e por isso, eles não podem ser dinâmicos, como a interface para celular. Diante disso, foi desenvolvido uma interface, levando em conta que pessoas, em geral, farão uso dela. Então, ela precisa ser simples e intuitiva, com a capacidade de ser manejada por qualquer pessoa.

4.3 Maletas

Sendo um dos componentes essenciais do presente projeto, as maletas têm a função de interagir com o ambiente virtual, acionando os dispositivos virtuais, ao mesmo tempo que em que aciona os dispositivos reais, e ter os dispositivos reais acionados pelas interfaces digitais e/ou virtuais. Usando o ESP8266 e o ESP32 como microcontroladores, e o MQTT como protocolo de comunicação, é possível fazer de forma perfeita essa integração. Como há diversos controles e dispositivos, e com o intuito de mostrar as possibilidades e facilidades do uso do Wi-Fi em um potencial projeto real, serão empregadas duas maletas: uma para fazer os acionamentos, usando dois microcontroladores ESP8266, e outra onde serão instalados os atuadores, usando um microcontrolados ESP32.

4.3.1 Maleta de Comando

A maleta de comando contém sete botões para seleção de dispositivos, dois botões para seleção de tipo de dispositivo (iluminação e ventilação), um botão para fazer a seleção de todos os botões de dispositivos, um outro botão para a desseleção desses botões, e um botão para a execução do comando para os dispositivos selecionado, totalizando 12 botões. Tem também um *slider* para o controle da potência da iluminação e dos ventiladores dos dispositivos selecionados, o qual utiliza 10 eletrodos, onde cada eletrodo representa 10 porcento de potência, com exceção do primeiro, que representa a ausência de potência. Conta

ainda com um leitor de RFID para a identificação de usuário para a abertura de portas que contenham restrição de acesso. Todos os botões são *touch,* e o status de selecionado, com a respectiva potência de cada botão, será indicada através de um display de OLED. Como uma interface eletrônica, a sua função é basicamente enviar comandos para acender/apagar e controlar a suas respectivas potências, ligar/desligar e também controlar as respectivas potências dos ventiladores, e receber o código de cartões com *tag* RFID para, em caso de correspondência, abrir a porta no ambiente virtual e na maleta dos atuadores. Os controles serão ativados e disponibilizados de acordo com o número de dispositivos do ambiente, sendo esses indicados no display. No templo, por exemplo, todos os controles estarão habilitados e poderão receber valores de potência para serem enviados para o ambiente virtual e para a maleta com os atuadores. Já no caso das salas, apenas um controle de lâmpada e um controle de ventilador estará habilitado para o envio da potência desejada. Como todos os ambientes têm portas, o controle de acesso através do leitor RFID funcionará em todos os ambientes, mas será efetivo somente nos lugares em que houver restrição de acesso. A interface da maleta de comando é mostrada na Figura 31 assim como a placa que faz a ligação dos NodeMCU ESP8266 com os módulos *touch*, o leitor RFID e os *displays* é mostrada na Figura 32.

4.3.2 Maleta com os atuadores

A maleta de atuadores conta com sete ventiladores de 60 mm e com sete lâmpadas LEDs e uma porta miniaturizada com fechadura eletrônica. O acionamento desses dispositivos pode ser feito por qualquer uma das interfaces (eletrônica, no ambiente virtual, via navegador ou através de *smartphone* ou *tablet*) em paralelo com o ambiente em que o personagem virtual estiver. Para isso, há um sistema de controle de potência feito através de TRIACs para os ventiladores e driver com dimerização para as lâmpadas de LEDs. Então quando ele estiver no templo, todas as lâmpadas poderão ser acionadas, assim como os ventiladores. Se o personagem estiver em uma sala, que permiti o controle de apenas uma lâmpada e um ventilador, somente esses poderão ser controlados. Como foi dito, a porta poderá ser usada em todos os ambientes, independentemente de haver restrição ou não. No caso de restrição, somente com o cartão correto ela poderá ser aberta.

4.3.2.1 Placa de circuito impresso

Devido a quantidade de dispositivos e número de componentes para efetuar o controle desses dispositivos, a maleta com os atuadores, depois de montada começou a apresentar instabilidades no seu funcionamento. Como mostrado no Vídeo 4, há uma quantidade considerada de fios sendo que ali só havia 2 ventiladores e 1 lâmpada. Quando todos os dispositivos foram montados o seu funcionamento ficou debilitado e até mesmo intermitente. Devido à falta de equipamentos que pudesse fazer uma medição precisa dos sinais, o motivo para a dificuldade só pode ser estimado. Talvez, a mais provável causa dessas dificuldades seja o efeito eletromagnético gerado pelos cabos principalmente pela dificuldade de se isolar os cabos de sinal e de potência, além do uso de fontes de alimentação inadequadas, conforme mostrado na Figura 33, onde é mostrado o interior da maleta antes de depois da PCI. Segundo Cassiolato, em equipamentos eletrônicos, há a possibilidade de falhas na comunicação entre os dispositivos, pode-se ter sinais gerados sem explicação como em um relé que atua de forma aleatória. Isso pode acontecer devido ao fato de cabos se atuam como sistema de transmissão com linhas cruzadas e confusa, passando sinais dentre circuitos, sem falar na propagação de ondas eletromagnéticas que são propagadas por “antenas acidentais” que são formadas por ida e volta da corrente e dos materiais dielétricos ao seu redor (Cassiolato, 2020). Em equipamentos de automação, como é o caso do presente projeto, fazer o possível para minimizar desse tipo de ruído é vital, já que ele pode apresentar inúmeros problemas na transmissão e no recebimento de sinais.

Então, na tentativa de minimizar a instabilidade do sistema, foi feita a opção de se construir uma placa de circuito impresso. Esse foi um caminho para diminuir-se a quantidade de cabos dentro da maleta e uma forma um pouco mais eficiente de separar o caminho do sinal e da potência, além de diminuir, de forma considerável, o espaço ocupado pelos componentes e apresentá-los de uma forma mais organizada e consistente.

Para essa tarefa foi escolhido o aplicativo KiCAD que é “um conjunto de programas de código livre multiplataforma (Windows/Linux/Mac). (Lopes, 2016) sendo uma ferramenta EDA (Electronic Design Automation), que foi concebido inicialmente por Jean-Pierre Charras em 1992, e hoje é um gestor de projetos, onde se abrem ferramentas para edição de esquemáticos (eeschema), associação entre componentes e módulos ou footprints (cvpcb), edição da placa de circuito impresso (pcbnew), edição de arquivos gerber para a fabricação da PCI (gerbview), contando ainda com um visualizador 3D dos componentes e da placa resultante do projeto (Cavalcante, 2011).

A confecção da placa de circuito impresso foi iniciada pelo desenvolvimento do esquema elétrico como mostrado na Figura 36, sendo a etapa seguinte a confecção da placa em si, com a distribuição dos componentes e por último o roteamento das trilhas (Figura 37). Apresenta-se o projeto finalizado em 3D na Figura 38.

Figura 31 - Interface da maleta de comando

Fonte: Do próprio autor

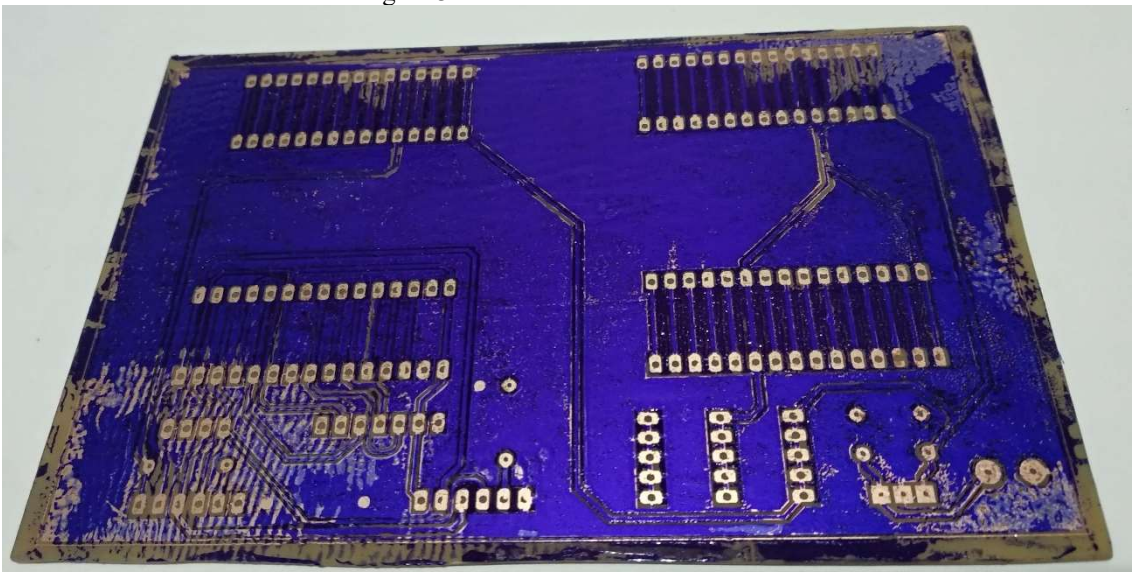
Quanto ao dimensionamento elétrico, essa placa funciona com 3 tensões: 127V para a alimentação das lâmpadas com um consumo aproximado de 3,22A. 12V para a alimentação dos ventiladores e fechadura eletrônica com um consumo estimado em 1,68A e 3,3V para a alimentação do ESP32, do relê e dos LEDs com um consumo calculado de 900 mA. Essas informações foram necessárias para o dimensionamento da fonte de alimentação, que nesse caso foi escolhida uma com a entrada em 127V e saída de 12V com o consumo máximo de 3A, já que o consumo das tensões 12V e 3,3V ficou em 2,58A. Com relação as trilhas da placa, segue a tabela com as respectivas medidas:

Tabela 2 - Dimensionamento das Trilhas da PCI

Tensão (V)	127	12	3,3
Corrente (A)	3,5	2	1
Mil (milionésimo de polegadas)	66,6	30,8	11,8
milímetros	1,69	0,781	0,3

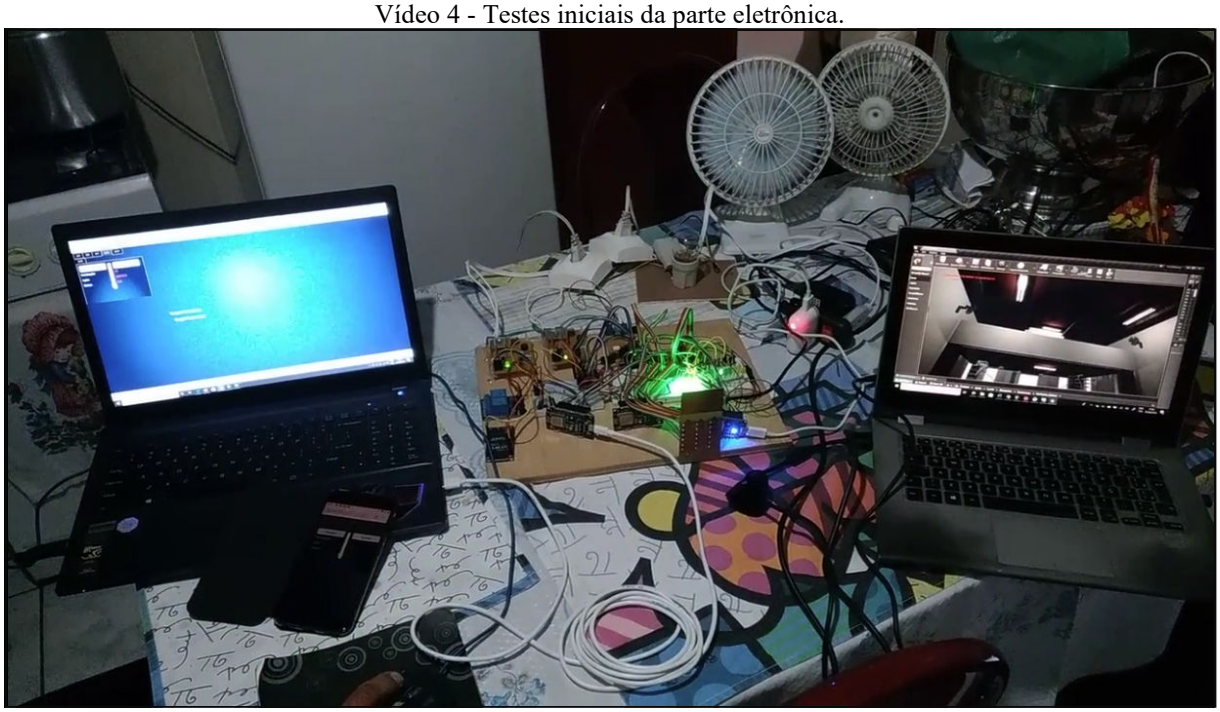
Fonte: Do próprio autor

Figura 32 - PCI da maleta de comando



Fonte: Do próprio autor

Vídeo 4 - Testes iniciais da parte eletrônica.

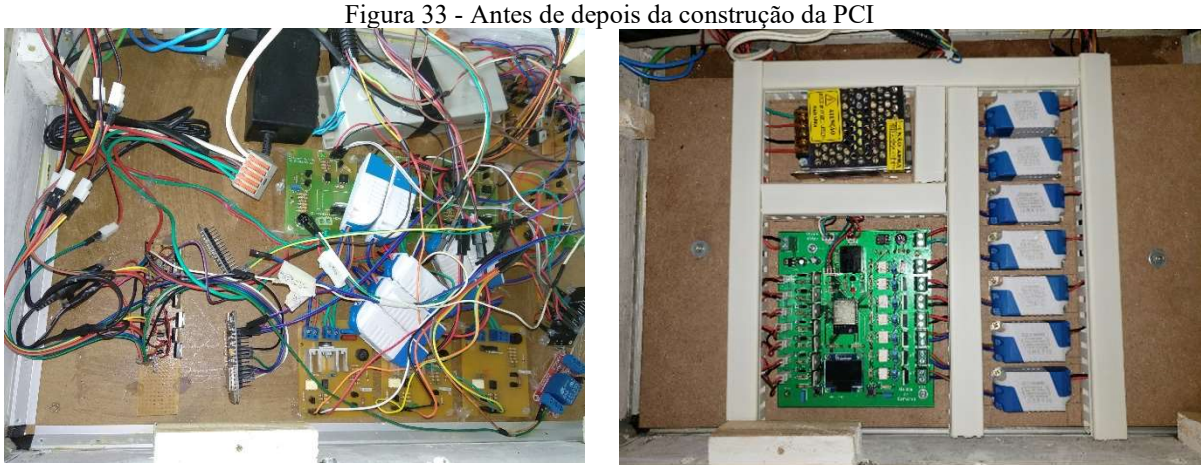


Fonte: Do próprio autor

4.3.2.1.1 Módulo de processamento e controle

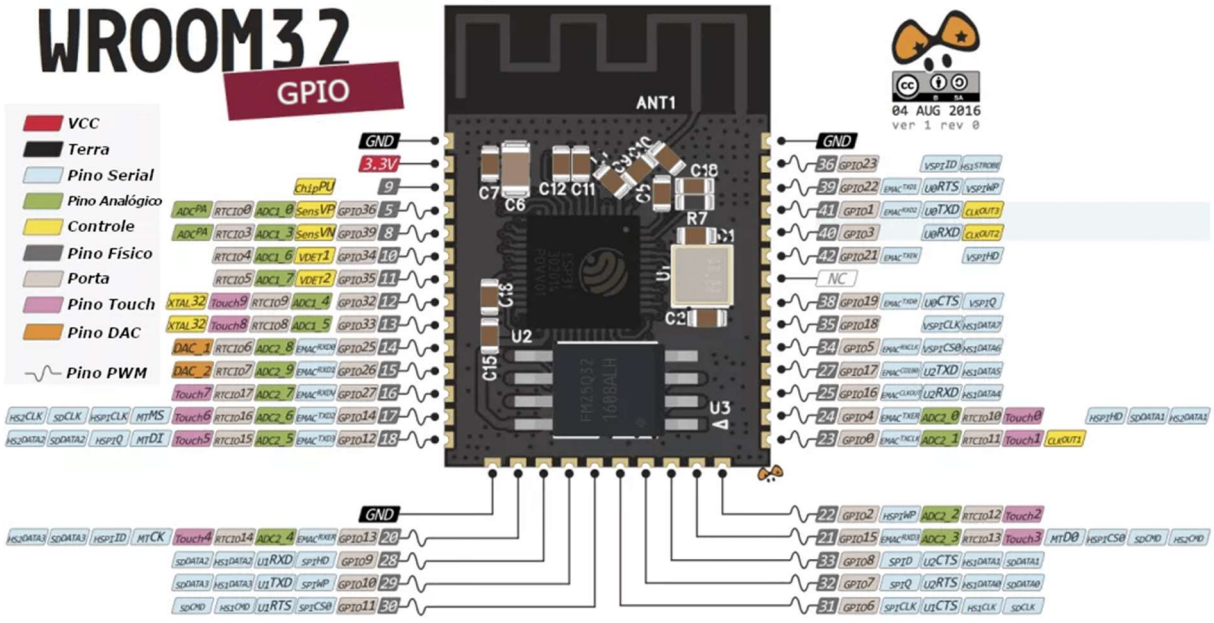
A parte de processamento e controle fica sob a responsabilidade do módulo ESP32 que devido aos seus 23 pinos de GPIO disponíveis conseguem, em um único módulo, controlar todos os ventiladores, lâmpadas e o controle e indicação para a porta. Na Figura 34 pode ser visto a numeração e descrição dos pinos GPIO do ESP32 e na Tabela 3 pode-se ver a direção e a descrição dos pinos usados para o presente projeto.

Figura 33 - Antes de depois da construção da PCI



Fonte: Do próprio autor

Figura 34 - Pinos do ESP32



Fonte: (Pinto, 2020)

4.3.2.1.2 Sistema Dimmer

Para o controle da potência luminosa das lâmpadas LEDs foi desenvolvido um módulo que, se utilizando da teoria de controle do tempo em 0V da rede elétrica, onde quanto maior foi o tempo em que a tensão permanece nessa tensão, menor, de forma proporcional é a intensidade da luz emitida, e quando a tensão 0V só permanece no momento em que a senoide altera de positivo para negativo e vice-versa, a intensidade luminosa mostra a sua totalidade (Straub, 2020).

Tabela 3 - Definição dos pinos GPIO do ESP32

GPIO	Direção	Descrição
1	OUTPUT	Serial TX/LED Verde
2	OUTPUT	Pulso para abrir porta
3	INPUT	Serial RX
4	OUTPUT	LED Vermelho
5	PWM	Controle do Ventilador 3
12	INPUT	Detecção de mudança de fase
13	INPUT	Botão para abrir a porta
14	OUTPUT	Controle da Lâmpada 1
15	INPUT	Fim de Curso Porta
16	PWM	Controle do Ventilador 1
17	PWM	Controle do Ventilador 2
18	PWM	Controle do Ventilador 4
19	PWM	Controle do Ventilador 5
21	I/O	I2C (SDA)
22	I/O	I2C (SCL)
23	OUTPUT	Controle da Lâmpada 7
25	OUTPUT	Controle da Lâmpada 4
26	OUTPUT	Controle da Lâmpada 3
27	OUTPUT	Controle da Lâmpada 2
32	OUTPUT	Controle da Lâmpada 6
33	OUTPUT	Controle da Lâmpada 5
34	PWM	Controle do Ventilador 7
35	PWM	Controle do Ventilador 6

Fonte: Do próprio autor

4.3.2.1.3 Sistema PWM

Diferente da corrente alternada, em que o controle da potência é feita através de TRIAC, fazendo uso do controle de tempo, em que a tensão é mantida em 0V dentro de um determinado período de tempo em cada senoide, o controle de potência na corrente contínua

se dá pelo controle ou modulação da largura de pulso (PWM – *Pulse-Width Modulation*), e é por meio dessa tecnologia que os ventiladores têm a sua potência controlada.

Os sistemas, juntamente com os módulos, são destacados na Figura 35.

Figura 35 - Sistemas na PCI

Fonte: Do próprio autor

4.4 Programação

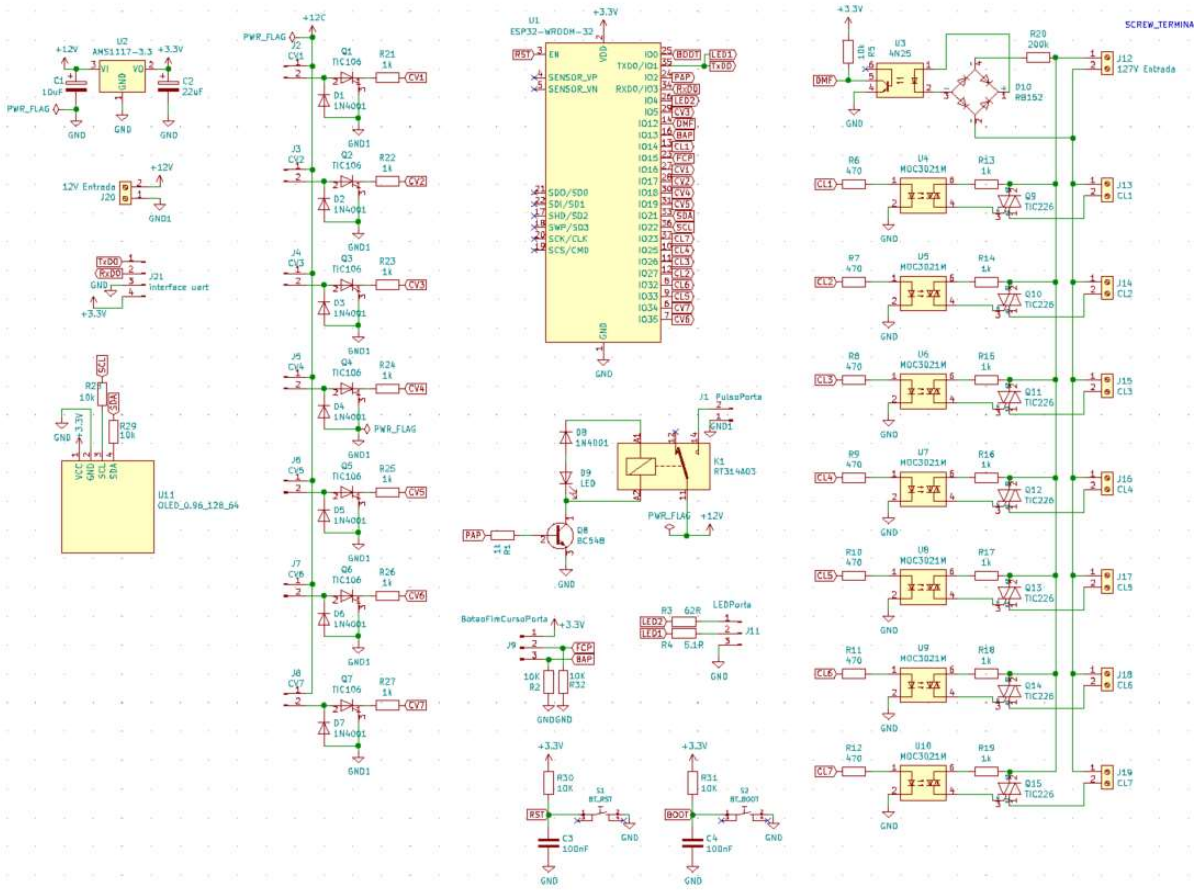
O desafio na programação foi evitar sempre que possível a complexidade. Como foi dito anteriormente, a parte e processamento das informações ficou por conta do Node-RED, mas foi necessário que os dados fossem empacotados e enviados para o servidor MQTT, e ao receber a resposta do Node-RED, era necessário que esses dados fossem interpretados e executassem a respectiva função. Então, em cada plataforma, a programação teve as suas próprias vicissitudes.

4.4.1 Unreal

No Unreal, a programação tinha sobre a sua responsabilidade o funcionamento das animações e a interpretação dos dados recebidos, para que ele acionasse uma determinada animação, abrindo ou fechando uma porta, acionando uma lâmpada ou um ventilador de cada um dos ambientes. Como a programação foi desenvolvida dentro de um ambiente gráfico,

como também foi feito no Node-RED, ao invés de programação usando a linguagem C++, houve um melhor rendimento na codificação, mas como há muitos ambientes e só no templos 12 lâmpadas e 7 ventiladores controlados por 7 acionamentos, o código ficou extenso e não foi possível fugir de forma eficiente da complexidade. Além disso, havia ainda o cartão RFID virtual que precisa atuar.

Figura 36 - Esquema elétrico da PCI



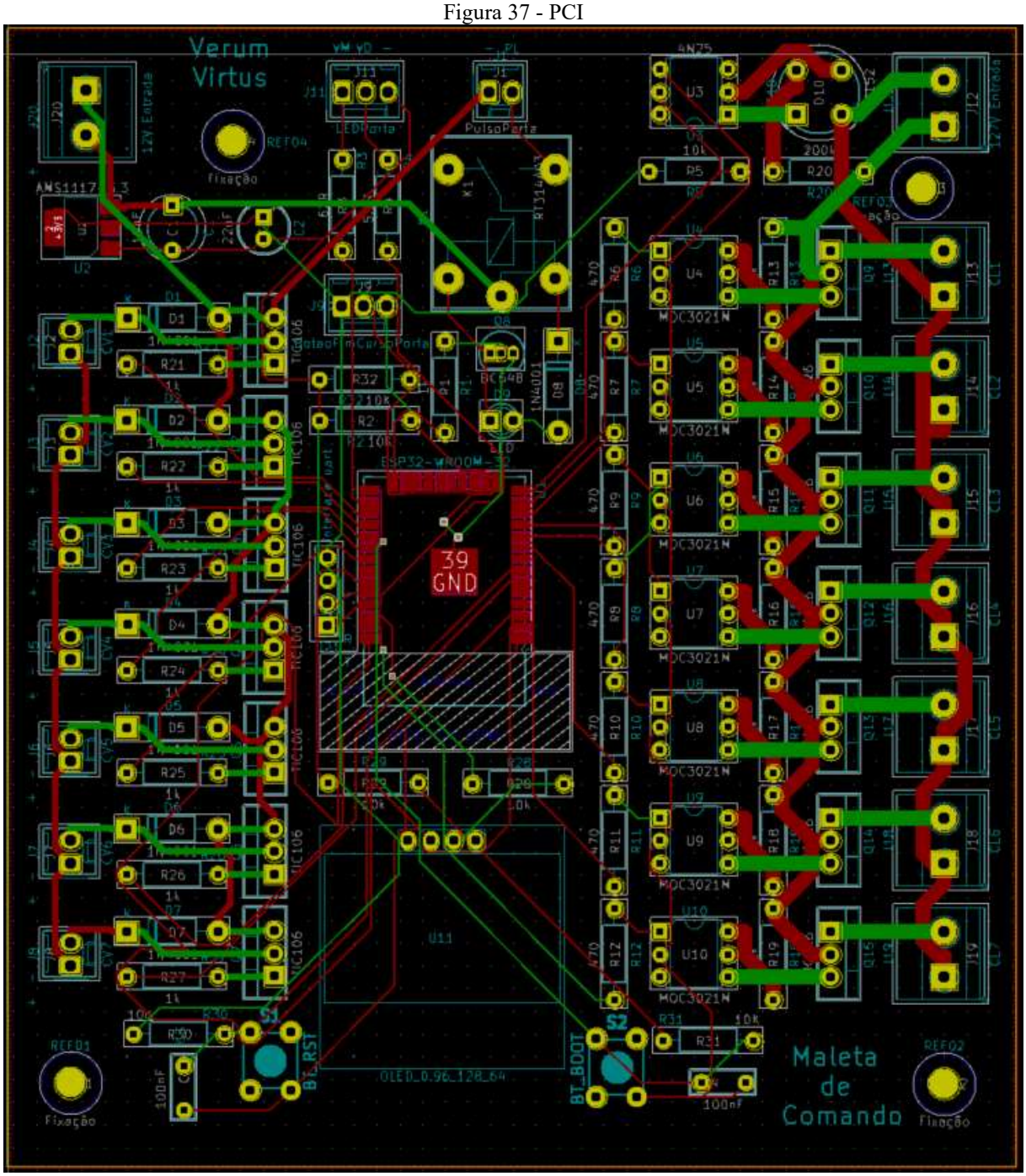
Fonte: Do próprio autor

4.4.2 Maletas (ESP8266 e ESP32)

Nas maletas, onde foram usados 2 ESP8266, a programação foi feita em Wiring, uma linguagem desenvolvida com base no Processing utilizando de abstrações de hardware, desenvolvida em C++ (Equipe Embarcados, 2020), te teve como principal objeto o controle dos sensores *touch* e RFID, que recebem as informações do usuário e envia através do MQTT para o servidor, que repassa para o Node-RED para que sejam processados e devolva-lhe as

respostas cabíveis. Nessa maleta de comando, os dados recebidos do servidor MQTT (*broker*) são apenas para a atualização dos indicadores que são feitos através dos displays OLED.

Figura 37 - PCI



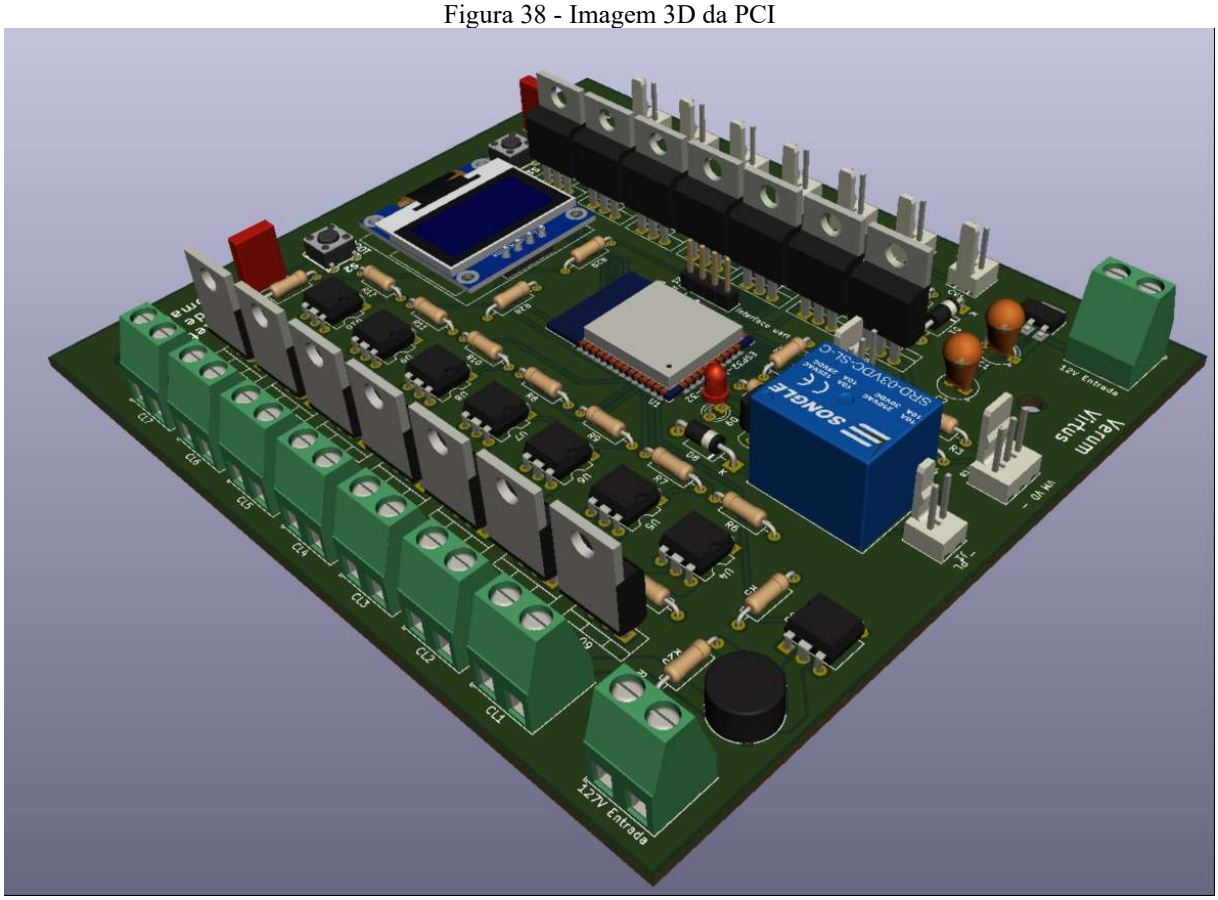
Fonte: Do próprio autor

Já na maleta de potência, a programação recebe dados do Broker e através do GPIO, transforma-os em sinais analógicos que informam as lâmpadas e cooler qual é a potência em que eles, de forma individual, devem operar. Um ESP32 controla as lâmpadas, os ventiladores

e a fechadura eletrônica, que, por sua vez, tem a sua abertura liberada pelo sistema através da leitura de cartões com *tags* RFID na maleta de comando e são abertas por um pequeno botão presente nessa maleta. Como o controle da potência é individualizado, a programação faz esse controle para cada subsistema de controle de potência.

Talvez a maior dificuldade na programação usada pelas maletas é o fato de haver uma grande quantidade de dispositivos, sejam de acionamento ou de controle, e aqui, para a otimização do sistema, foi construída uma biblioteca no arduino que contém os elementos em comum nos dois ESP8266 da maleta de comando e no ESP32 que está localizado na maleta de potência, como a configuração da rede, MQTT e a estrutura dos tópicos, dentro do conceito de reusabilidade de código (COSTA, et al., 2012, pp. 2-3), não somente diminuindo de forma considerável o tamanho do código, mas também diminuindo a complexidade do código nas manutenções e atualizações do mesmo.

Figura 38 - Imagem 3D da PCI



Fonte: Do próprio autor

4.4.3 Central de Comando (Node-RED)

Como foi dito anteriormente, o Node-RED fica responsável por toda a parte lógica do sistema. Usando o MQTT para receber e enviar as informações de forma assíncrona e, dependendo do que está sendo controlado, ele verifica se há alguma ação a ser executada. Diferente da programação em C++, Java, Python, dentro de tantas outras que são linguagens procedurais, ou seja, que são focadas em mudança de estado das variáveis, o Node-RED é uma linguagem orientada a eventos com o uso de blocos (TEDESCO, 2016), onde o fluxo da programação é iniciado com algum tipo de evento, seja uma interação do usuário, seja um dado recebido através de um dos muitos protocolos que a linguagem suporta nativamente. Então, todas as vezes que um dado é alterado no servidor MQTT, e esse mesmo dado é esperado na programação, é executado uma série de instruções que foram previamente arranjadas.

É também no Node-RED que foi desenvolvido o controle dinâmico de interface. Esse controle permite que todas as interfaces (eletrônicas ou através de software) sejam adaptadas de acordo com o ambiente que o avatar do usuário esteja, configurando a quantidade de controle para luzes e ventiladores para este local, bastando que este se posicione em frente à sua porta e que direcione o controle da mini porta, que está na maleta de potência para o presente ambiente, ou seja, ao passar o cartão RFID e apertar o botão que solicita a abertura da porta, se houver permissão, a mini porta se abre juntamente com a porta onde o avatar está. No Vídeo 5, é possível ver o autoajuste dos controles da interface que está no ambiente virtual conforme o personagem se posiciona em frente aos locais.

Esse recurso possibilitou que uma única interface pudesse ser usada para controlar os 13 ambientes que estão presentes no projeto de forma simples, transparente e natural para quem está no comando do avatar, diminuindo em muito, a complexidade, caso fosse necessário ter um controle individualizado para cada ambiente. Também possibilitou a padronização dos controles, facilitando a inserção de novas interfaces, já que estas precisam apenas enviar e receber comandos ficando a parte do processamento das informações inteiramente por conta do Node-RED. Com isso é possível de forma fácil e simples desenvolver uma interface em Android, ou via web, como *javascript* ou *Rubi*, além da possibilidade do incremento de controles físicos personalizados, dando ao cliente uma

interação ainda mais próxima do seu dia-a-dia, usando uma caixa de passagem com um interruptor digital juntamente com um *slider*.

4.5 Testes do Sistema

Os testes foram conduzidos com o avanço do desenvolvimento do projeto, e a cada não conformidade, era analisado a dificuldade encontrada e verificada a possibilidade de superá-la. Isso impôs diversas alterações no projeto, incrementando algumas novas tecnologias e investimentos. Algumas dificuldades, mesmo depois de várias tentativas, não foram devidamente superadas. Isso se deu por vários fatores, como a necessidade de um tempo considerável, o que se mostrou inviável devido ao limite que o presente projeto tem. Outro fator comum foi a questão de custo, já que os componentes e softwares nessas áreas que esse trabalho teve contato têm, em geral, um custo de certa relevância. Outro fator da qual se optou por não superar foi o incremento de novas tecnologia, que na maioria das vezes, apresentam uma curva de aprendizado acentuada e que, apesar de apresentar ganhos relevantes na qualidade do projeto, tiveram que ser deixadas de lado.

Vídeo 5 - Controle Dinâmico de Interface.

Fonte: Do próprio autor.

4.5.1 Unreal

No *Unreal*, os testes foram relativamente bem sucedidos, mas como há uma certa complexidade, tanto na adequação da modelagem, como nas animações e na programação, alguns detalhes, principalmente os que demandaria uma maior compreensão da ferramenta, não puderam ser plenamente corrigidos. Um exemplo que é possível citar, para ilustrar o ponto em questão, foi a seleção do leitor de RFID, que necessita de uma certa precisão para que o comando de usar o cartão RFID seja usado e algumas vezes não funciona com a simplicidade que o sistema necessita para ser operado por um possível cliente. Depois de algumas tentativas frustradas para melhorar o procedimento, ficou perceptível que o melhor caminho seria o desenvolvimento de outro mecanismo para essa ação. Mas isso demanda a pesquisa, o aprendizado e a adequação para o presente projeto.

4.5.2 Maletas

A maleta de comando não apresentou grandes problemas nos testes, mas conforme o projeto foi evoluindo foram necessárias constantes adequações à programação dos microcontroladores. Essa dificuldade se mostra presente sempre que há o desenvolvimento de um sistema com diversos componentes onde a correção de um dado componente exige a atualização de todo o restante. Mas, de todos os componentes do presente projeto, a maleta de comando mostrou-se um dos mais simples no quesito funcionalidade e atualização.

4.5.3 Interfaces

As interfaces, por terem como princípio apenas o envio de comando e a exibição de informação, se mostraram eficazes na operação do sistema como um todo. Mas, a comunicação feita pelo MQTT, devido à grande quantidade de dados circulando pelo sistema, em alguns momentos, apresentou alguns comportamentos indesejados, mas nada que comprometesse o funcionamento do projeto. Verificou-se a necessidade de uma análise mais profunda para descobrir melhor o seu funcionamento, para dessa forma, dar mais estabilidade ao sistema. Infelizmente essas informações não são fáceis de serem obtidas, já que na maior parte das fontes consultadas, as informações se mostraram um tanto superficiais e repetitivas, e nos mantenedores do protocolo a informação é muito técnica e exigem um conhecimento profundo em áreas que vão muito além do escopo desse trabalho.

4.5.4 Teste Geral

O teste geral mostrou que o sistema já adquiriu uma certa estabilidade e que a sua proposta básica é funcional sem grandes dificuldades. É possível fazer o passeio virtual pelo imóvel e fazer os acionamentos e aberturas de portas e é possível verificar o funcionamento do controle de acesso, tanto no ambiente real como no virtual, mas ainda há várias implementações e correções para que o projeto realmente se transforme em um produto. O núcleo do projeto funciona, mas para ser completamente funcional ainda há uma caminhada.

No apêndice B há uma lista de checagem que constam todos os acionamentos a serem testados para certificar o pleno funcionamento do sistema em uma situação que envolva um cliente real. Isso é feito não apenas para uma checagem final do sistema, mas a cada alteração importante que alguma parte do projeto sofra, ter-se condições de aferir-se que tudo permanece em pleno funcionamento. No presente trabalho, os testes não foram exaustivos por uma questão de escassez de tempo e porque o que se procura aqui é atestar a validade do conceito que foi proposto.

5 CONCLUSÕES

Apesar de se encontrar alguma complexidade na construção do protótipo devido a necessidade, em um primeiro momento (em um processo que precisa naturalmente de ser otimizado), do uso de várias tecnologias, linguagens de programação, conceitos da qual se teve apenas um primeiro contato e de ferramentas que precisa de um maior domínio para que possam ser usadas com mais eficiência (principalmente mais agilidade), este se mostrou viável e operacional, apresentando um baixo custo para a produção de novos projetos e com recursos cumulativos, ou seja, a cada projeto executado é possível acumular-se ferramentas, bibliotecas de programas e/ou modelagem para os projetos futuros, ou seja, há uma otimização natural do projeto que em pouco tempo pode alcançar uma maturidade e uma eficácia mostrando-se, dessa maneira, plenamente viável e, portanto, alcançando, de forma inequívoca, o objetivo desse trabalho.

Outra característica que merece menção é o fato de o produto final desse projeto ter sido construído buscando a sua mobilidade. Uma vez finalizado a virtualização do projeto de automação empresarial, com apenas duas maletas e o acesso a uma tomada de energia elétrica, é possível a demonstração para um possível cliente, de como o seu projeto se comportaria dentro de sua empresa permitindo, como foi mencionado ao longo deste trabalho, a verificação, atuação, visualização do resultado dessas atuações em praticamente qualquer escritório, sala de reunião ou outro ambiente em que o cliente tenha a possibilidade de receber a apresentação.

5.1 Trabalhos futuros

Em se tratando de conceitos tecnológicos, a amplitude do mesmo é vasta, e para cada tecnologia é necessário a dedicação de um tempo considerável, o que em um trabalho de conclusão de curso acaba impondo uma certa limitação naquilo que realmente pode ser feito em detrimento de todas as possibilidades que são abertas durante a caminhada. Outra limitação é, como foi afirmado acima a falta de domínio de ferramentas que foram usadas em várias etapas do projeto. É como conhecer uma melodia, mas não ter destreza suficiente para executá-la em um determinado instrumento musical.

Por isso, fica aqui registrado algumas sugestões para trabalhos futuros que incrementarão várias otimizações que, sendo implantadas impactarão nos quesitos tempo,

custo, qualidade e usabilidade, sem mencionar a possibilidade de expansão de funcionalidades que poderão ser implementadas no presente projeto.

A primeira sugestão está ligada ao fator custo e vai na direção da substituição dos *softwares* que, por motivo de ter uma curva de aprendizado menor, com ampla documentação e cursos disponíveis foram adotados na construção do protótipo. Esse é o caso do Autocad, Revit e 3DS Max, todos de propriedades da empresa Autodesk que podem ser substituídos por aplicativos livres, ou seja, de licença *open-source*. Como exemplo pode-se citar o aplicativo QCAD (Mustun, 2020) como um possível substituto para o Autocad, para o aplicativo Revit não foi encontrado um aplicativo com as suas características e com licenciamento *opens source*. Mas com a sua popularização é possível que em breve uma opção viável seja oferecida. Já para o aplicativo 3DS Max tem-se um candidato que merece a atenção, devido a sua maturidade e variedade de recursos: o Blender (Blender, s.d.).

Outra sugestão pertinente é otimização dos sistemas usados para diminuir a sua quantidade e tentar reduzir, tanto quando possível a sua complexidade. Dentro desta perspectiva, há por exemplo o estudo e a verificação da viabilidade da migração de toda a programação do Node-RED para o ioBroker que trabalham de forma similar, baseados ambos em Javascript. A dificuldade inicial é que o ioBroker é um software alemão com pouca documentação em inglês e português tendo o seu fórum basicamente em sua língua nativa.

A próxima sugestão envolve a parte física do projeto e vai na direção de um refinamento, tanto do conjunto de controle como de potência, para quem sabe, se aproximar, tanto quanto possível, ou da experiência do cliente, ou de uma proposta exequível para o cliente. Dentro dessa mesma linha, há a sugestão de módulos de controles separados da maleta que demostrem a facilidade de escalonamento do sistema.

Indo agora na direção de inclusão de novos recursos, a sugestão fica por conta da implementação do restante do conjunto de segurança, onde pode ser incluído um sistema de alarme com câmeras e central de monitoramento, um sistema de controle de acesso com cadastro de usuários e grupos de permissões com políticas de acesso individualizável e configurável, além de permissões temporárias de acesso a determinados locais, sendo tudo passível de rastreabilidade.

Outro recurso que pode ser implementado, dessa vez, dentro da maquete eletrônica é a simulação de tempo, que permitiria demostrar ao cliente um sistema automatizado de controle da potência das lâmpadas baseados na luminosidade externa, juntamente com outros parâmetros, como sensores de presença. Isso abriria a possibilidade de simulação de falta de

energia para o acionamento de nobreak/gerador de energia e estimativa de gasto de energia em um determinado período.

5.2 Material publicado e exposto

Esse trabalho é o resultado de um longo período de pesquisa, no qual, foram produzidos artigos e trabalhos que foram expostos e que de forma significativa, contribuíram para os resultados aqui apresentados. Os apêndices C e D estão relacionados ao estudo do protocolo MQTT, que permitiu verificar o seu funcionamento através de um pequeno protótipo, onde foi possível o dimensionamento do seu uso no presente trabalho.

Os apêndices E e F estão relacionados com as duas primeiras versões desse projeto, uma em 2016, onde a ideias nasceu e deu os seus primeiros passos, o projeto foi apresentado como trabalho de conclusão de curso de Eletrônica, onde apresentou inúmeras dificuldades, mas demostrou ser um conceito que poderia ser melhor explorado, e em 2019, com vários aprimoramentos o projeto foi apresentado novamente como trabalho de conclusão de curso de Eletrotécnica.

6 REFERÊNCIAS

Allan, A. (01 de 11 de 2018). *A Cheaper Single-Core ESP32 Module?* Acesso em 06 de Agosto de 2020, disponível em hackster.io: https://www.hackster.io/news/a-cheaper-single-core-esp32-module-53fa9c4143b5

Alluring Stories. (24 de Maio de 2018). *Amalgamation of IoT, Big Data and AR/VR: VEot*. Acesso em 08 de Janeiro de 2020, disponível em Alluring Stories: https://alluringstories.com/amalgamation-of-iot-big-data-and-ar-vr-veot/

ALMEIDA, M. (2007). *Desvendando o 3ds Max* (1ª ed.). São Paulo, SP, Brasil: Ed. Digerati Books.

Altus. (24 de Outubro de 2018). *Conheça o MQTT, protocolo mais utilizado em aplicações IoT*. Acesso em 24 de Janeiro de 2020, disponível em Altus: https://www.altus.com.br/post/194/conheca-o-mqtt-2C-protocolo-mais-utilizado-em-aplicacoes-iot

AMÂNCIO, A. (19 de Junho de 2018). *O Protocolo KNX*. Acesso em 22 de Janeiro de 2020, disponível em Wire Engenharia - Blog Automação: http://wireengenharia.com.br/br/o-protocolo-knx/

AMARAL, L. (s.d.). *Maquete Eletrônica: Importância, vantagens e principais programas*. Acesso em 19 de 01 de 2020, disponível em Arquiteto Leandro Amaral: https://arquitetoleandroamaral.com/vantagens-maquete-eletronica/

Arruda, E. (12 de Abril de 2017). *Como fazer a metodologia de um projeto*. (J. L. Alaúzo, Editor) Acesso em 26 de Dezembro de 2019, disponível em Biblioteconomia Digital: https://www.biblioteconomiadigital.com.br/2010/07/como-fazer-metodologia-em-um-projeto.html

Ashton, K. (22 de Junho de 2009). That 'Internet of Things" Thing in the real world, things matter more than ideas. *RFiD Journal*. Acesso em 10 de Janeiro de 2020, disponível em https://www.rfidjournal.com/articles/view?4986

Athos Electronics. (s.d.). *ESP32 - Especificações e projetos*. Acesso em 23 de 11 de 2020, disponível em Athos Electronics: https://athoselectronics.com/esp32/

Autodesk. (s.d.). *3DS Max*. Acesso em 21 de Janeiro de 2020, disponível em Autodesk: https://www.autodesk.com.br/products/3ds-max/overview

AutoDesk Help. (12 de Junho de 2018). *3DS Max*. Acesso em 21 de Janeiro de 2020, disponível em Autodesk Knowledge Network: https://knowledge.autodesk.com/pt-

br/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2018/PTB/3DSMax-Basics/files/GUID-F8326C68-F2F9-47F7-AC1D-BA41D7825C7C-htm.html

BARROS, M. (15 de Junho de 2015). *MQTT - Protocolo para IoT*. Acesso em 24 de Janeiro de 2020, disponível em Embarcados: https://www.embarcados.com.br/mqtt-protocolos-para-iot/

Bauermeister, G. (22 de Março de 2018). *Guia do Usuário do ESP8266*. Acesso em 03 de Janeiro de 2020, disponível em Filipeflop: https://www.filipeflop.com/blog/guia-do-usuario-do-esp8266/

Baxter, M. R. (2000). *Projeto de Produto: guia prático para design de novos produtos* (2ª ed.). (I. Iida, Trad.) São Paulo, SP, Brasil: Ed. Blucher. Acesso em 26 de Dezembro de 2019

Bertoleti, P. (2019). *Projetos com ESP32 e LoRa* (1ª ed.). São Paulo, SP, Brasil: CLUBE DE AUTORES. Acesso em 23 de Novembro de 2020

Blender. (s.d.). Acesso em 11 de Junho de 2020, disponível em Blender: https://www.blender.org/

Blynk.Community. (01 de Dezembro de 2018). *Controlling Blynk with Node-RED and vice-versa*. Acesso em 18 de Fevereiro de 2020, disponível em Blynk: https://community.blynk.cc/t/controlling-blynk-with-node-red-and-vice-versa-demo/31185

Brito, E. (24 de 06 de 2015). *Debian: distribuição oferece sistema completo e universal*. Acesso em 05 de 10 de 2020, disponível em techtudo: https://www.techtudo.com.br/tudo-sobre/debian.html

BURDEA, G. C., & COIFFET, P. (2003). *Virtual Reality Tecnology* (2ª ed.). Hoboken, New Jersey, Estados Unidos da America: John Wiley & Sons.

Carvalho, F. S., Castro, B. F., Sousa, L. M., Silva, P. F., Seves, R. G., & Limad, W. G. (Outubro de 2018). MQTT: Um protocolo para IoT. *Boletim Técnioca da Faculdade de Tecnologia de São Paulo*(Especial), 62. Acesso em 02 de Janeiro de 2020

Casagras: an EU Framework 7 Projects. (2009). *RFID and the Inclusive Model for the Internet of Things.* União Europeia, Casagras - Coodination and Support Action for Global RFID-Related Activities and Standardisation. União Européia: no prelo. Acesso em 14 de Janeiro de 2020, disponível em https://docbox.etsi.org/zArchive/TISPAN/Open/IoT/low%20resolution/www.rfidglobal.eu%20CASAGRAS%20IoT%20Final%20Report%20low%20resolution.pdf

Cassiolato, C. (30 de Janeiro de 2020). *EMI - Interferência Eletromagnética em instalações industriais e muito mais*. Acesso em 16 de 10 de 2020, disponível em smar - Technology Company: https://www.smar.com/brasil/artigo-tecnico/emi-interferencia-eletromagnetica-em-instalacoes-industriais-e-muito-mais

Cavalcante, A. (21 de 04 de 2011). *Ferramenta EDA open-source: Kicad*. Acesso em 16 de 10 de 2020, disponível em Jera: https://jera.com.br/blog/3146/desenvolvimento/ferramenta-eda-open-source-kicad

Centro de Ciências Exatas e Sociais Aplicadas. (20 de Maio de 2015). Produção do Trabalho de Conclusão de Curso em Engenharia de Computação. Bauru, São Paulo, Brasil. Acesso em 26 de Dezembro de 2019, disponível em Universidade Sagrado Coração: https://pt.slideshare.net/elainececiliagatto/orientaes-tcc-eng-comp

CHANG, L.-Y. (2018). *Revit MEP Step by Step 2019 Metric Edition* (1ª ed.). Seatle, Washington, EUA: Ed. Kindle.

CHEN, W.-J., GUPTA, R., LAMPKIN, V., ROBERTSON, D. M., & SUBRAHMANYAM, N. (2014). *Responsive Mobile User Experience Using MQTT and IBM MessageSight* (1ª ed.). Armonk, Nova York, EUA: Ed. IBM Redbook.

CHETOUANE, F. (2015). An Overview on RFID Technology Instruction and Application. pp. 382-387. Acesso em 24 de Janeiro de 2020, disponível em https://www.sciencedirect.com/science/article/pii/S240589631500350X

COPPEN, R. (24 de Maio de 2018). *OASIS Message Queuing Telemetry Transport (MQTT) TC*. Acesso em 24 de Janeiro de 2020, disponível em OASIS - Open standards. Open source: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=mqtt

Cossini, F. (2015). *Uma Ontologia dos Padrões da Internet das Coisas.* Instituto de Pesquisas Tecnológicas do Estado de São Paulo. São Paulo: no prelo. Acesso em 15 de Janeiro de 2020, disponível em https://www.ipt.br/pos_graduacao_ipt/solucoes/dissertacoes/787-uma_ontologia_dos_padroes_da_internet_das_coisas.htm

COSTA, C., DIAS, G., SANTOS, L., ANIELE, B., COSTA, A., & GUIMARÃES, D. (25 de Outubro de 2012). *Reusabilidade e portabilidade.* Artigo, Instituto Federal do Mato Grosso do Sul, Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas - TADS. Acesso em 10 de Março de 2020

Curto Circuito. (s.d.). *Conhecendo o ESP32*. Acesso em 23 de 11 de 2020, disponível em Curto Circuito: https://www.curtocircuito.com.br/blog/conhecendo-esp32

Damin, D. (25 de Março de 2019). *Aplicação exemplo: Driver MQTT em comunicação com Broker Mosquitto MQTT*. Acesso em 20 de Fevereiro de 2020, disponível em Elipse Knowledgebase: https://kb.elipse.com.br/aplicacao-exemplo-driver-mqtt-em-comunicacao-com-broker-mosquitto-mqtt/

Dedding, A. (2018). Apresentação - A Indutria 4.0 no Brasil. Em A. W. Lima, A. P. Cardoso Jr., B. Petroni, C. Couto, E. P. Silva, E. M. Moraes, . . . O. Vendrametto, J. B. Sacomano, R. F. Gonçalves, M. T. Silva, S. H. Bonilla, & W. C. Sátyro (Eds.), *Indústria 4.0 - Conceitos e Fundamentos. Apresentação* (1 ed., p. 182). São Paulo, São Paulo, Brasil: Ed. Bucker. Acesso em 27 de Dezembro de 2019

Dennis, A. K. (2013). *Raspberry Pi Home Automation with Arduino* (1ª ed.). Birmingham, Mumbai, Índia: Packt Publishing.

Discabos. (20 de Outubro de 2017). *Projeto de automação para sala de reunião de uma multinacional*. Acesso em 27 de Dezembro de 2019, disponível em Discabos: https://discabos.com.br/blog/exemplos-de-projetos/projeto-de-automacao-para-sala-de-reuniao-de-uma-multinacional

Doneda, D. (2018). Internet das Coisas. Em E. Magrani, *Internet das Coisas. Prefácio.* (1ª ed., p. 192). Rio de Janeiro, Rio de Janeiro, Brasil: FGV Editora. Acesso em 26 de Dezembro de 2019

DZAMBAZOVA, T., KRYGIEL, E., & DEMCHAK, G. (2009). *Introducing Revit Architecture 2010 - BIM for Begnners* (1ª ed.). (S. STAFFORD, Ed.) Hoboken, New Jersey, Estados Unidos da America: Wiley Publishing, Inc.

EASTMAN, C., TEICHOLS, P., SACKS, R., & LISTON, K. (2010). *BIM Handbook - A Guide to BIM for Owners, Managers, Designers, Engineers and Contractors* (1ª ed.). Hoboken, New Jersey, Estados Unídos da America: John Wiley & Sons.

Equipe Embarcados. (04 de Março de 2020). *Editorial: Linguagens de Programação para Sistemas Embarcados*, 2ª. Acesso em 10 de Março de 2020, disponível em Embarcados: https://www.embarcados.com.br/editorial-linguagens-para-sistemas-embarcados/

EverPi. (28 de Março de 2017). *Consumo de energia do Raspberry Pi parte 6 - Versão 3 Modelo B*. Acesso em 30 de Dezembro de 2019, disponível em EverPi - Tudo sobre o Raspberry Pi: http://blog.everpi.net/2017/03/raspberry-pi-3-consumo-de-energia.html

Fhem. (s.d.). *Fhem*. Acesso em 23 de Janeiro de 2020, disponível em Fhem: https://fhem.de/

FINKENZELLER, K. (2010). *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication* (3ª ed.). (D. MÜLLER, Trad.) Chichester, West Sussex, UK: Ed. Wiley.

Foneplan. (16 de Julho de 2015). *O que é Automação Corporativa e Porque Investir?* Acesso em 07 de Fevereiro de 2020, disponível em https://foneplan.com.br/blog/automacao-corporativa/o-que-e-automacao-corporativa-e-porque-investir/

Fórum dda Construção. (02 de Março de 2018). *Por que adotar a automação corporativa?* Acesso em 10 de Fevereiro de 2020, disponível em Instituto Brasileiro de Desenvolvimento da Arquitetura: http://www.forumdaconstrucao.com.br/conteudo.php?a=11&Cod=2124

Freescale Semiconductor. (01 de Setembro de 2010). Proximity Capacitive Touch Sensor Controller - Advanced Information. *Technical Data*. Petaling Jaya, Selangor, Malásia.

Freitas, V. G. (2016). *Realidade Virtual na Simulação de Procedimentos Operacionais em Depósitos de Rejeitos Radioativos.* Universidade Federal do Rio de Janeiro, Programa de Pós-Graduação em Engenharia Nuclear. Rio de Janeiro: no prelo. Acesso em 28 de Dezembro de 2019, disponível em http://www.con.ufrj.br/wp-content/uploads/2016/07/Tese-Victor-Gon%C3%A7alves-Gl%C3%B3ria-Freitas.pdf

Gerbert, P., Lorenz, M., Rubmann, M., Waldner, M., Justus, J., Engel, P., & Harnisch, M. (9 de Abril de 2015). *Industry 4.0: The futur of Productivity and Growth in Manufacturing Industries*. Acesso em 16 de Janeiro de 2020, disponível em BCG: https://www.bcg.com/publications/2015/engineered_products_project_business_industry_4_future_productivity_growth_manufacturing_industries.aspx

GOMES, J. (2019). *Cyberpreview: Cibernética aplicada à prevensão de erros e falhas* (2ª ed.). São Paulo, SP, Brasil: Ed. Nelpa.

Graeff, D. (16 de Dezembro de 2018). *MQTT Arrives in the Modern openHAB 2.x Architecture*. Fonte: Blog openHAB: https://www.openhab.org/blog/2018-12-16-mqtt-arrives-in-the-modern-openhab-2-x-architecture.html

Guilherme. (5 de Dezembro de 2019). *Jogos feitos na Unreal Engine: 5 Jogos de tirar o fôlego*. Acesso em 22 de Janeiro de 2020, disponível em Blog Danki Code: https://blog.dankicode.com/jogos-feitos-na-unreal-engine/

Guiness World Book. (s.d.). *Most Successful Videogame engine*. Acesso em 22 de Janeiro de 2020, disponível em Guiness World Book: https://www.guinnessworldrecords.com/world-records/most-successful-game-engine/#

HALE, K. S., & STANNEY, K. M. (Eds.). (2018). *Handbook of Virtual Environments: Design, Implementation, and Applications* (2ª ed.). Boca Raton, FLorida, Estados Unidos da América: CRC Press.

Heebra, A. (16 de Dezembro de 2015). *O futuro da Internet of Things será a Internet of Everything?* Acesso em 16 de Janeiro de 2020, disponível em ComputerWorld: https://computerworld.com.br/2015/12/16/o-futuro-da-internet-things-sera-internet-everything/

HEINLE, S. (Julho de 2017). Das Projekt IO-Broker macht Smart-Home-Anwendungen kompatibel. *Linux Magazin*. Acesso em 23 de Janeiro de 2020, disponível em https://www.linux-magazin.de/ausgaben/2017/07/io-broker/5/

HEINLE, S. (Abril de 2018). IO-Broker integriert diverse Smart-Home-Protokolle. *Linux Magazin*. Acesso em 23 de Janeiro de 2020, disponível em https://www.linux-magazin.de/ausgaben/2018/04/io-broker/

HEK. (s.d.). *ioBroker*. Acesso em 23 de Janeiro de 2020, disponível em MySensors: https://www.mysensors.org/controller/iobroker

HILLAR, G. C. (2017). *MQTT Essentials - A Lightweight IoT Protocol* (1ª ed.). Bimighan, UK: Ed. Packt Publishing.

HU Infitivo. (08 de 08 de 2019). O que é NodeMCU? *Artigo Eletrônico*. Brasilia, Distrito Federal, Brasil. Acesso em 05 de 10 de 2020, disponível em https://www.huinfinito.com.br/blog/artigos/o-que-e-nodemcu

KLEEN, S. (01 de Setembro de 2019). *ioBroker - Visualisierung erstellen im ioBroker*. Acesso em 18 de Fereveiro de 2020, disponível em Udemy: https://www.udemy.com/course/iobroker-visualisierung-erstellen/

Kleina, N. (22 de Março de 2011). *O que é engine ou motor gráfico?* Acesso em 27 de Dezembro de 2019, disponível em TecMundo: https://www.tecmundo.com.br/video-game-e-jogos/9263-o-que-e-engine-ou-motor-grafico-.htm

KOUTITAS, G. (Diretor). (2017). *Augmented Reality and Internet of Things: the new era of VEoT* [Filme Cinematográfico]. Acesso em 21 de 01 de 2020, disponível em https://www.youtube.com/watch?v=XHGtIbU-TBE

Kurniawan, A. (2019). *Internet of Things Projects with ESP32* (1ª ed.). Birmingham: Packt Publishing. Acesso em 23 de Novembro de 2020

Laboratório de Garagem. (10 de Agosto de 2012). *Tutorial: como utilizar a Breakout Sensor de toque capacitivo - MPR121 com Arduino*. Acesso em 24 de Fevereiro de 2020, disponível em Laboratório de Garagem: http://labdegaragem.com/profiles/blogs/tutorial-como-utilizar-a-breakout-sensor-de-toque-capacitivo

Lal Das, P., Beisswenger, S. C., Mangalam, S., Yuce, M. R., & Lukac, M. (2002). *Internet of things : the new government to business platform - a review of opportunities, practices, and challenges (English)* (1ª ed.). Washington, Estados Unidos da América: The World Bank Group. Acesso em 10 de Janeiro de 2020, disponível em http://documents.worldbank.org/curated/en/610081509689089303/Internet-of-things-the-new-government-to-business-platform-a-review-of-opportunities-practices-and-challenges

LEE, J. (2016). *Learning Unreal Engine Game Development* (1 ed.). Birmingham, West Midlands, UK: Ed. Packt Publishing.

LIMA, F. (2016). Maquete Eletrônica. Em F. BRAIDA, F. LIMA, J. FONSECA, V. MORAIS, & J. GASPAR (Ed.), *101 Conceitos de Arquitetura e Urbanismo na Era Digial* (p. 256). São Paulo, SP, Brasil: ProBooks.

Lopes, L. C. (10 de 06 de 2016). *Introdução ao KiCad*. Acesso em 16 de 10 de 2020, disponível em Embarcados: https://www.embarcados.com.br/introducao-ao-kicad/

Machado, F., & Ruschel, R. (10 de Novembro de 2017). A Integração de BIM e IOT com Ênfase em Energia na Fase de Operação e Manutenção da Edificação. *Ambiente Construido*(SBTIC/SIBRAGEC), 140-150. Acesso em 28 de Dezembro de 2019, disponível em https://www.researchgate.net/publication/321867585_A_ integracao _de_BIM_e_IoT_com_enfase_em_energia_na_fase_de_Operacao_e_Manutencao_da _edificacao/references

Mapa da Obra. (09 de Julho de 2018). *Automação Residêncial é uma Tendência?* Acesso em 18 de Fevereiro de 2020, disponível em Mapa da Obra: https://www.mapadaobra.com.br/inovacao/automacao-residencial/

MATOSSIAN, M. (2001). *3DS Max 4 para Windows* (1ª ed.). (A. D. MORAES, & C. B. DIAS, Trads.) Rio de Janeiro, RJ, Brasil: Ed. Campus.

MENEZES, A. M., VIANA, M. S., PEREIRA JR, M., & PALHARES, S. (2012). *O Impacto da Tecnologia BIM no Ensino de Projetos de Edificações. In: Congrego Brasileiro de*

Educação em Engenharia, 40. Faculdade de Engenharia e Arquitetura, Bélem -PA. Acesso em 16 de Janeiro de 2020, disponível em http://www.abenge.org.br/cobenge/arquivos/7/artigos/103929.pdf

Morais, J. (8 de Setembro de 2017). *Conhecendo o ESP32*. Acesso em 23 de Novembro de 2020, disponível em Vida de Silício: https://portal.vidadesilicio.com.br/conhecendo-o-esp32/

Moretti, I. (26 de Julho de 2018). *Metodologia de Pesquisa do TCC: conheça os tipos e veja como definir*. Acesso em 27 de Dezembro de 2019, disponível em Via Carreira: https://viacarreira.com/metodologia-de-pesquisa-do-tcc/

Murta, G. (1 de Março de 2018). *NodeMCU – ESP12: Guia completo – Introdução (Parte 1)*. Acesso em 03 de Janeiro de 2020, disponível em Blog Eletrogate: https://blog.eletrogate.com/nodemcu-esp12-introducao-1/

Mustun, A. (17 de Março de 2020). *QCAD.org*. Acesso em 11 de Junho de 2020, disponível em QCAD.org Open Source CAD: https://qcad.org/en/

NASSAR, V., & VIEIRA, M. H. (01 de Julho de 2014). A aplicação de RFID na logística: um estudo de caso do Sistema de Infraestrutura e Monitoramento de Cargas do Estado de Santa Catarina. *Revista Gestão & Produção*. Acesso em 24 de Janeiro de 2020, disponível em http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-530X2014000300006&lang=pt

Neocontrol. (s.d.). *https://www.neocontrol.com.br/news/automacao-corporativa-atrativa/*. Fonte: Neocontrole: https://www.neocontrol.com.br/news/automacao-corporativa-atrativa/

Netto, C. C. (2018). *Autodesk Revit Architecture 2018: Conceitos e Aplicações.* São Paulo, SP, Brasil: Ed. Saraiva.

OLIVEIRA, N. (13 de Fevereiro de 2018). *Por que Node-RED para IOT?* Acesso em 21 de Janeiro de 2020, disponível em Medium: https://medium.com/@netoolii/por-que-node-red-para-iot-41a4ab170c56

Oliveira, S. d. (2017). *Internet das Coisas com ESP8266, Arduino e Raspberry Pi* (1ª ed.). (R. Prates, Ed.) São Paulo, SP, Brasil: Ed. Novatec.

OpenJS Foundation. (s.d.). *Node-RED*. Acesso em 21 de 01 de 2020, disponível em Node-RED: https://nodered.org/

Papo de Arquiteto. (s.d.). *Curso Unreal para SketchUp, Revit e 3DS Max*. Acesso em 22 de Janeiro de 2020, disponível em Papo de Arquiteto Cursos: https://www.papodearquiteto.com.br/cursos/produto/curso-unreal-para-sketchup-revit-e-3ds-max/

Pinto, C. A. (22 de Março de 2020). *Referência de pinagem do ESP32: Quais pinos do GPIO você deve usar?* Acesso em 6 de Agosto de 2020, disponível em Cap Sistema: https://capsistema.com.br/index.php/2020/03/22/referencia-de-pinagem-do-esp32-quais-pinos-do-gpio-voce-deve-usar/

Rachor, E. (2013). *Desenvolvimento de Produto: Apresentação de um Método de Prototipagem de Hélices na Área de Geração de Energia Eólica.* Universidade Federal de Santa Maria, Centro de Tecnologia. Santa Maria, RS: (no prelo). Acesso em 26 de Dezembro de 2019, disponível em https://repositorio.ufsm.br/handle/1/2290

Raspberry Pi Foundation. (10 de Fevereiro de 2012). *Buy a Raspberry Pi 3 Model B*. Acesso em 14 de Janeiro de 2020, disponível em RaspberryPi: https://www.raspberrypi.org/products/raspberry-pi-3-model-b/

RIVERA, A. (29 de Junho de 2018). *Should Your Business Use Smart Metering?* Acesso em 23 de Janeiro de 2020, disponível em Business News Daily: https://www.businessnewsdaily.com/10925-what-is-smart-metering.html

SANDERS, A. (2017). *An Introduction to Unreal Engine 4* (1ª ed.). Boca Raton, Flórida, EUA: CRC Press.

Santos, S. (2018). *Industria 4.0: Saiba tudo sobre a Revolução das Máquinas* (1ª ed.). São Paulo, SP, Brasil: Publicação Independente.

Santos, S. R. (2018). *Introdução à IoT - Desvendando a internet das Coisas* (1ª ed.). São Paulo, São Paulo, Brasil: SS Trader Editor. Acesso em 26 de Dezembro de 2019

SAS. (17 de Junho de 2019). *Webinar On-Demand - Internet das coisas: coloque-a para trabalhar a favor da sua organização*. Acesso em 8 de Janeiro de 2020, disponível em SAS - The Power to Know: https://www.sas.com/pt_br/webinars/integrando-iot-na-sua-organizacao.html?utm_source=google&utm_medium=cpc&utm_campaign=iot-gen-cala&utm_content=3969870&gclid=CjwKCAiAmNbwBRBOEiwAqcwwpbWmY7JIR8HQ5FtNH4vslUVUlnbhWyca_7tiiXoSX_dnOM2EkmINmxoCJw4QAvD_BwE

Schwab, K. (2019). *A Quarta revolução Industrial* (1ª ed.). (J. L. Vieira, M. V. Micales, Eds., & D. M. Miranda, Trad.) São Paulo, SP, Brasil: Edipro.

SENA, D. C. (2013). *Simulação de Eventos Discretos Utilizando Realidade Virtual* (1ª ed.). São Paulo, SP, Brasil: Ed. Biblioteca24horas.

SERAFIM, G. (s.d.). *Maquete eletrônica*. Acesso em 20 de 01 de 2020, disponível em Maqueteria 3D: https://maqueteria3d.com.br/produto/maquete-eletronica/

Silva, R. G., Oliveira, C. S., Mendes, D. C., & Sales, A. M. (2017). *Sistema de Automação Residêncial de Baixo Custo Utilizando o ESP8266.* Instituto Federal de Educação, Ciência e Tecnologia do Ceará, Pró-Reitoria de Pesquisa e Inovação. Fortaleza: no prelo. Acesso em 14 de Janeiro de 2020, disponível em http://prpi.ifce.edu.br/nl/_lib/file/doc1243-Trabalho/PEVPI_RF.pdf

Sinclair, B. (2018). *IoT: Como usar a internet das coisas para alavancar seus negócios* (1ª ed.). (A. C. Serra, Trad.) São Paulo, São Paulo, Brasil: Ed. Autêntica Business. Acesso em 26 de Dezembro de 2019

SINGH, D., KROPF, J., HANKE, S., & HOLZINGER, A. (01 de Agosto de 2017). Ambient Assisted Living Technologies from the Perspectives of Older People and Professionals. *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pp. 255-266. doi:10.1007/978-3-319-66808-6_17

Soares, W. (30 de Maio de 2019). *Qual a melhor: Unity ou Unreal?* Acesso em 03 de Janeiro de 2020, disponível em Crie os seus jogos: https://www.crieseusjogos.com.br/qual-a-melhor-unity-ou-unreal/

Souza, A. R., Carvalho, F. S., Andrade, G. d., & Silva, J. S. (2016). *Automação de Igreja e Templos Religiosos.* Etec Gildo Marçal Bezerra Brandão, Eletrônica. São Paulo: No prelo.

STEVAN JUNIOR, S., & FARINELLI, F. (2019). *Domótica - Automação Residencial e Casas Inteligentes com Arduino e ESP8266* (1ª ed.). São Paulo: Ed. Érica.

Straub, M. G. (26 de Março de 2020). *Projeto ESP32 Dimmer - Automação para Controle da Luz*. Acesso em 24 de Agosto de 2020, disponível em Blog Usinainfo: https://www.usinainfo.com.br/blog/projeto-esp32-dimmer-automacao-para-controle-de-luz/

Suárez, J. C., Salazar, F. F., Nava, I. F., & Hernández, R. H. (30 de Janeiro de 2019). Industria 4.0 y Manufactura Digital: un Método de Diseño Aplicando Ingeniería Inversa. *Revista Ingeniería, 24*, 6-28. Acesso em 16 de Janeiro de 2020

Super Interressante. (31 de Outubro de 2011). *Quanto vale um petabyte?* Acesso em 16 de Janeiro de 2020, disponível em Super Interessante: https://super.abril.com.br/ciencia/quanto-vale-um-petabyte/

SWEENEY, T. (2 de Março de 2015). *If You Love Something, Set It Free*. Acesso em 22 de Janeiro de 2020, disponível em Unreal Engine: https://www.unrealengine.com/en-US/blog/ue4-is-free

SZCRYS, M. (15 de Janeiro de 2020). *AUTOMATE YOUR LIFE WITH NODE-RED (PLUS A DASH OF MQTT)*. Acesso em 21 de Janeiro de 2020, disponível em Hackaday: https://hackaday.com/2020/01/15/automate-your-life-with-node-red-plus-a-dash-of-mqtt/

TAGLIANI, S. (19 de Setembro de 2018). *Softwares para engenharia: saiba quais são as vantagens do uso do AutoDesk Revit*. Acesso em 21 de Janeiro de 2020, disponível em Engenharia 360: https://engenharia360.com/softwares-para-engenharia-autodesk-revit/

TEDESCO, K. (30 de Novembro de 2016). *Linguagens e paradigmas de Programação*. Acesso em 10 de Março de 2020, disponível em Treinaweb Blog: https://www.treinaweb.com.br/blog/linguagens-e-paradigmas-de-programacao/

Thakur, M. R. (2016). *Zero to Hero ESP8266* (1ª ed.). Seatle, Washington, Estados Unidos da America: Ed. Kindle.

The Linux Kernel Archives. (s.d.). *GPIO Interface*. Acesso em 24 de Fevereiro de 2020, disponível em The Linux Kernel Archives - Documentation: https://www.kernel.org/doc/Documentation/gpio/gpio.txt

THOMSEN, A. (11 de Abril de 2016). *Qual módulo ESP8266 comprar?* Acesso em 10 de 01 de 2020, disponível em FilipeFlop: https://www.filipeflop.com/blog/qual-modulo-esp8266-comprar/

TONÉIS, C. N. (2017). *Os Games na Sala de Aula: Games na Educação ou a Gamificação da Educação* (1ª ed.). Seatle, Washington, EUA: Ed. Kindle.

TORI, R., & KIRNER, C. (2006). Fundamentos de Realidade Virtual. Em *Fundamentos e Tecnologias de Realidade Virtual e Aumentada* (R. TORI, C. KIRNER, & R. SISCOUTTO, Trads., 1ª ed., pp. 2-21). Bélem, Pará: Sociedade Brasileira de Computação.

totalCAD. (10 de 01 de 2018). *Maquete eletrônica: 10 motivos para usar em apresentações de projetos*. Acesso em 20 de 01 de 2020, disponível em totalCad: Softwares

Técnicos: https://blog.totalcad.com.br/maquete-eletronica-10-motivos-para-usa-las-na-apresentacao-de-projetos/

TRAN, T. (21 de Setembro de 2017). *Unreal Engine 4 - Animation Tutorial*. Acesso em 22 de Janeiro de 2020, disponível em Unreal Engine Tutorials: https://www.raywenderlich.com/383-unreal-engine-4-animation-tutorial

Tumelero, N. (21 de Setembro de 2018). *Metodologia do TCC: como delimitar com lista de exemplos*. Acesso em 26 de Dezembro de 2019, disponível em Mettzer: https://blog.mettzer.com/metodologia-tcc/

Unreal Engine. (s.d.). *Overview*. Acesso em 22 de Janeiro de 2020, disponível em Unreal Engine: https://www.unrealengine.com/en-US/what-is-unreal-engine-4

Vaccari, A. (2015). *How Virtual Reality Meets the Industrial IoT.* Artígo Científico, Aalto University, School of Science, Helsinki, Finland. Acesso em 28 de Dezembro de 2019, disponível em https://www.semanticscholar.org/paper/How-Virtual-Reality-Meets-the-Industrial-IoT-Aalto/4af56a2a82f9683bb30b30711d613e2ca5dd7494#paper-header

VALCASARA, N. (2015). *Unreal Engine Game Development Blueprints* (1ª ed.). Birmangham, UK: Ed. Packt Publishing.

VENDRAMINI, R. M. (s.d.). *Unreal Engine 4 Básico Aplicado à Arquitetura*. Acesso em 22 de Janeiro de 2020, disponível em Cursos Construir: https://cursosconstruir.com.br/curso/unreal-engine-4-basico-aplicado-a-arquitetura/

VINCE, J. (1998). *Essential Virtual Reality fast: How To Understand The Techniques And Potential Of Virtual Reality* (1ª ed.). Verlag, Londres, Inglaterra: Springer.

Wu, J.-W., Chou, D.-W., & Jiang, J.-R. (01 de Setembro de 2014). The Virtual Environment of Things (VEoT): A Framework for Integrating Smart Things into Networked Virtual Environments. *ITHINGS '14*, pp. 456–459. Acesso em 09 de Janeiro de 2020, disponível em https://dl.acm.org/doi/abs/10.1109/iThings.2014.81

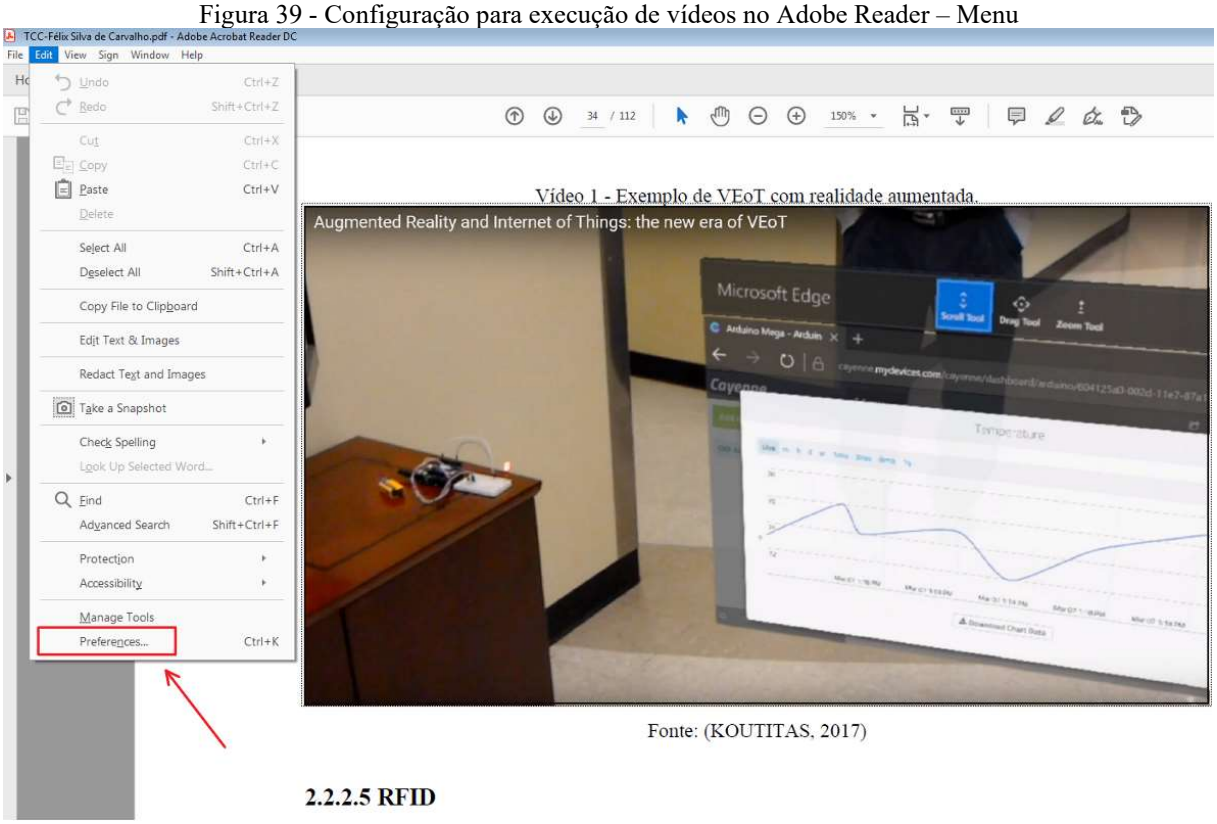
ZAGOLO, N., & PEREIRA, L. (2008). Ambientes Virtuais e Second Life. Em A. A. CARVALHO (Ed.), *Manual de Ferramentas da WEB 2.0 para Professores* (1ª ed., p. 240). Esplanada dos Ministérios, Brasília, Brasil: Ministério da Edução.

7 APÊNDICES

APÊNDICE A – INSTRUÇÕES PARA EXECUTAR OS VÍDEOS.

Os arquivos PDF atualmente suportam a incorporação de arquivos de mídias como áudio e vídeo, mas a execução dessas mídias só acontece em aplicativos para a leitura específica de arquivos PDF. Foram realizados testes bem sucedidos com o aplicativo Foxit Reader e o Adobe Reader. No caso do Adobe, há uma configuração chamada *Trust Option,* que estabelece a permissão ou não de execução de arquivos de vídeos, conforme as figuras abaixo. Para permitir essa execução, a opção que deve ser selecionada é a *prompt*, ou seja, o aplicativo vai pedir permissão para executar cada vídeo, e essa, por questões de segurança, é a mais recomendada. Há ainda a opção *Always*, que executa o vídeo sem nenhum questionamento, sendo uma opção menos segura e não recomendável.

Figura 39 - Configuração para execução de vídeos no Adobe Reader – Menu



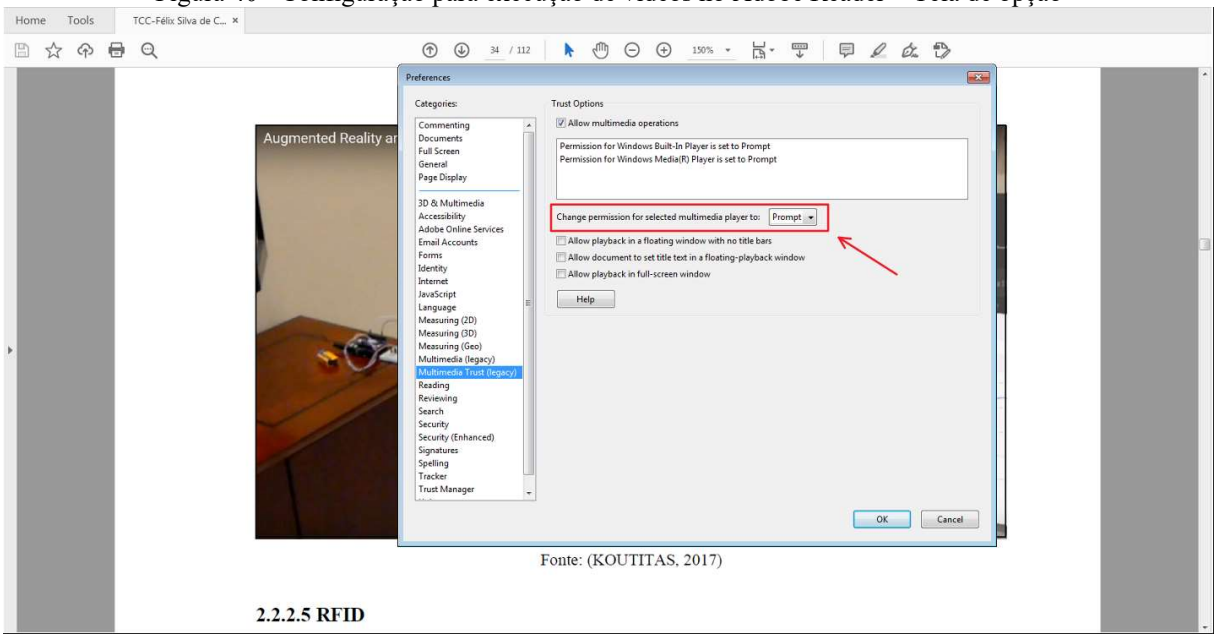
Fonte: (KOUTITAS, 2017)

Fonte: Do próprio autor

Há outros aplicativos que abrem arquivos em PDF, mas não são leitores nativos do formado, e por isso não conseguem acesso alguns recursos embutidos nesse tipo de arquivo,

entre eles a execução de vídeo. É o caso do Microsoft Word, dos navegadores de internet mais populares, entre outros. Nesses casos, os vídeos aparecem apenas como uma figura.

Figura 40 - Configuração para execução de vídeos no Adobe Reader – Tela de opção



Fonte: (KOUTITAS, 2017)

2.2.2.5 RFID

Fonte: Do próprio autor

APÊNDICE B – LISTA DE CHECAGEM DO SISTEMA.

				ioBroker (smartphone, unreal e web)	Maleta Comandos
templo	iluminação	Lâmpadas da Nave	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Lâmpadas do Púlpito	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 1	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 2	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 3	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 4	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 5	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
	Ventiladores	Ventilador 1	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador 2	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok

		Ventilador 3	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador 4	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador 5	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador 6	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador 7	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
	Portas	Porta da Frente	Acesso Inválido	ok	ok
			Acesso Correto: Não abrir a porta	ok	ok
			Acesso Correto: Abrir a porta	ok	ok
			Fechar a porta internamente	ok	ok
		Porta Lateral	Acesso Inválido	ok	ok
			Acesso Correto: Não abrir a porta	ok	ok
			Acesso Correto: Abrir a porta	ok	ok
			Fechar a porta internamente	ok	ok

Salão	iluminação	Lâmpadas da Nave	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 1	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok

			acionamento a 0%	ok	ok
		Refletor de Palco 2	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 3	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 4	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Refletor de Palco 5	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
	Ventiladores	Ventilador 1	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador 2	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador 3	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador 4	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
	Portas	Porta da Frente	Acesso Inválido	ok	ok
			Acesso Correto: Não abrir a porta	ok	ok
			Acesso Correto: Abrir a porta	ok	ok
			Fechar a porta internamente	ok	ok
		Porta Lateral	Acesso Inválido		

			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		

Salas	Sala 1	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Sala 2	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Sala 3	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		

			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Sala 4	Lâmpada	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Ventilador	acionamento a 100%	ok	ok
			acionamento a 70%	ok	ok
			acionamento a 30%	ok	ok
			acionamento a 0%	ok	ok
		Porta	Acesso Inválido	ok	ok
			Acesso Correto: Não abrir a porta	ok	ok
			Acesso Correto: Abrir a porta	ok	ok
			Fechar a porta internamente	ok	ok
	Sala 5	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Sala 6	Lâmpada	acionamento a 100%		
			acionamento a 70%		

			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Sala 7	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Conselho	Luminária 1	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Luminária 2	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador 1	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador 2	acionamento a 100%		

			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Secretaria	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Cozinha	Lâmpada	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Ventilador	acionamento a 100%		
			acionamento a 70%		
			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		
	Zeladoria	Lâmpada	acionamento a 100%		
			acionamento a 70%		

			acionamento a 30%		
			acionamento a 0%		
		Porta	Acesso Inválido		
			Acesso Correto: Não abrir a porta		
			Acesso Correto: Abrir a porta		
			Fechar a porta internamente		

APÊNDICE C – ARTIGO PUBLICADO NO BOLETIM TÉCNICO DA FATEC-SP EM OUTUBRO DE 2018.

O PROTOCOLO MQTT E SUA APLICAÇÃO

Felix S. Carvalho[1], Beatriz F. Castro[2], Lyliana M. S. de Sousa[3], Paulo R. F. X. da Silva[4], Rodrigo G. Seves[5], Prof. M. Sc. Waltson G. Neto de Limad[6]
[1,2,3,4,5,6] Faculdade de Tecnologia de São Paulo - FATEC-SP
[6]Escola Politécnica da USP
felix.carvalho@fatec.sp.gpv.br, wlimad@usp.br

1. Introdução

Com a conectividade trazida pela era digital, novos conceitos surgiram. Alguns já existentes ganharam nova relevância, tendo o seu uso otimizado, dando a abertura para novos padrões. Esse parece ser o caminho do protocolo MQTT, que apesar de não ser um protocolo de comunicação tão recente, ganhou novo brilho com o surgimento da Indústria 4.0 e com a popularização do IoT (Internet of Thing – internet das coisas).

Este projeto possui o objetivo de testar a viabilidade do uso do protocolo MQTT (Message Queue Telemetry Transport) através de um sistema de automação, que obtém dados através de um sensor de temperatura e humidade e depois de processá-los, atua em dispositivos de controle por uma rede sem fio. Para tanto, foi criado um protótipo que recebe esses dados e aciona dois ventiladores à medida que a temperatura atinge um nível pré-estabelecido.

2. Materiais e Método.

Os principais materiais usados na montagem do protótipo são: um minicomputador Raspberry PI 3, três módulos ESP8266-01, um sensor DHT-11 (temperatura e humidade), dois módulos relê e dois ventiladores.

O Raspberry tem função de servidor do protótipo MQTT e também conterá o supervisório do sistema, que utiliza o aplicativo Node-RED e pode ser acessado por qualquer navegador web (PC, smartphone, tablet) desde que estejam na mesma rede do sistema. O sensor DHT-11 é conectado a um dos módulos ESP8266-01, que fica responsável pela aquisição dos dados e o envio dos mesmos ao servidor. Uma vez que o Raspberry recebe esses dados, atualiza essa informação nos outros dos módulos ESP8266-01 que verificam, de forma independente, se a temperatura está dentro da condição de acionamento pré-estabelecida, acionando e desacionando cada um dos dois ventiladores, ligados através de sinais enviados aos módulos relés.

Figura 1- Supervisório do Sistema

3. Resultados

O sistema se mostrou consistente, ágil, seguro e escalonável. Nota-se a consistência no modo de publicação/inscrição para com os dados transmitidos ao e do servidor, são enviados de forma assíncrona para todos os dispositivos que forem previamente inscritos no servidor, com a segurança de que os dados foram transmitidos de forma bem-sucedida para os inscritos. Ágil pela instantaneidade da recepção dos dados requisitados pelo sensor, exibidos pelo supervisório e pela rápida resposta de acionamento dos ventiladores. Seguro, tendo-se vários níveis de autenticação e a possibilidade de encriptação dos dados que são transmitidos entre o servidor e os clientes, que enviam e recebem esses dados. Escalonável, devido a facilidade de se acrescentar novos sensores e novos atuadores sem a necessidade de alteração no servidor. Mudando apenas poucos parâmetros no módulo ESP8266-01, pode-se fazê-lo de forma modular semelhante à tecnologia "plug in play".

4. Conclusões

Portanto, concluímos que, ao desenvolvermos e analisarmos os resultados obtidos do projeto, foram comprovadas as qualidades do Protocolo de Comunicação. Em análise à desenvoltura obtida no final do projeto, pôde-se notar a sua eficiência e facilidade de utilização. Esse estudo pode ser ampliado para a escala real com ventiladores ou ar-condicionados.

Agradecimentos

À instituição Faculdade de Tecnologia de São Paulo, pela realização das medidas ou empréstimo de equipamentos. Ao Professor Doutor Waltson Limad pela colaboração e orientação.

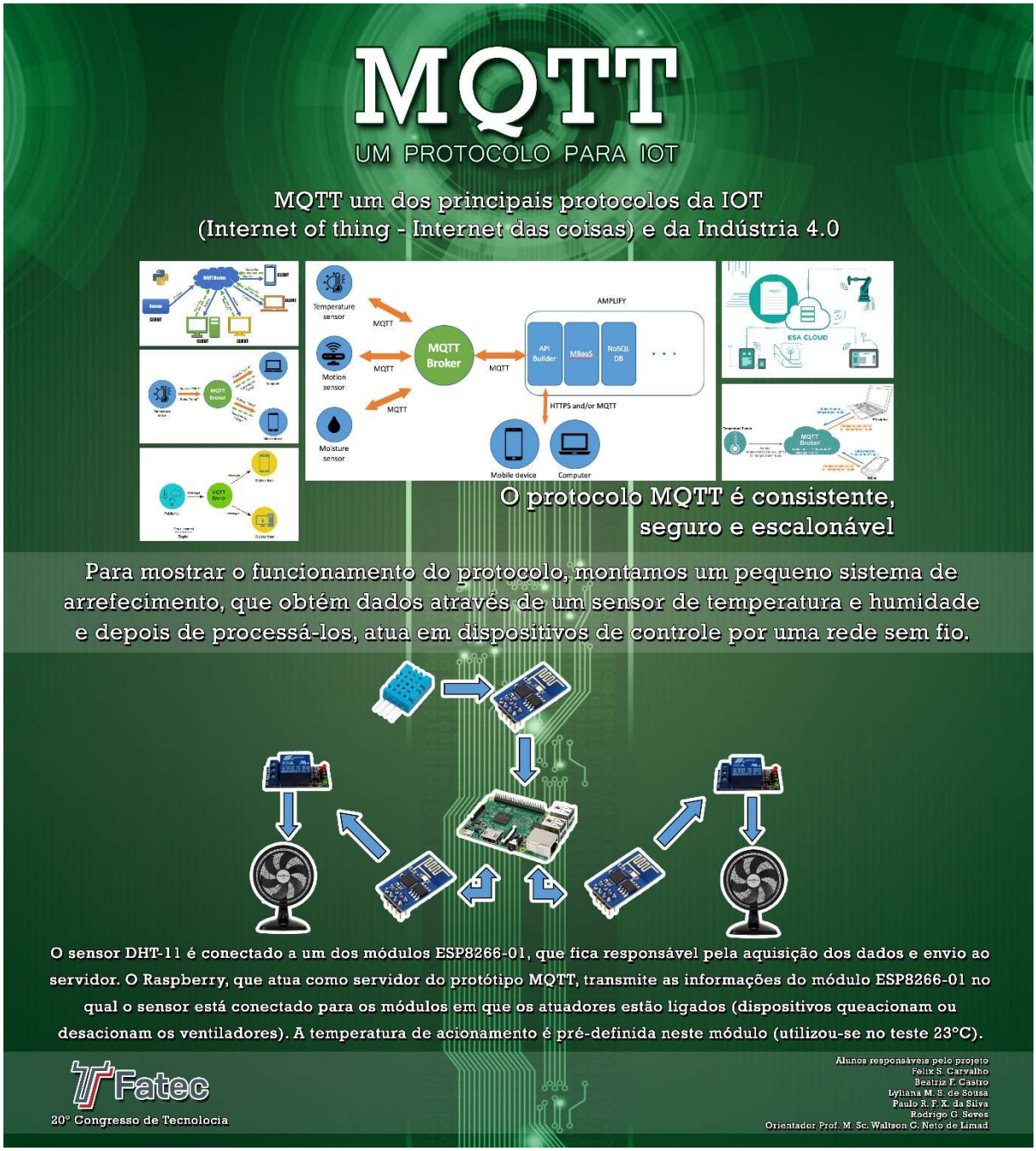
Referências Bibliográficas

[1] Hillar, G. C. (2017). ***MQTT Essentials - A Lightweight IoT Protocol***. Birmingham: Packt Publishing .

[2] Zem, I. R. (2015). **Estudo dos Protocolos De Comunicação MQTT e COAP para Aplicações Machine-To-Machine e Internet das Coisas**. *Revista Tecnológica da Fatec Americana*, 1-24.

[1] Aluno da Iniciação Científica do CNPq

APÊNDICE D - TRABALHO EXPOSTO NO 20º SIMPÓSIO DE INICIAÇÃO CIENTÍFICA E TECNOLÓGICA DA FATEC-SP - SICT REALIZADO EM 18 DE OUTUBRO DE 2018.



APÊNDICE E - TRABALHO DE CONCLUSÃO DE CURSO NA ETEC JARAGUÁ - 2º SEMESTRE 2019.

VIRTUS

SISTEMA DE SIMULAÇÃO PARA AUTOMAÇÃO CORPORATIVA

Carvalho, Félix ; Volpato, Flávio; Santos, Jonas.

Introdução

Diante de um cenário de novas tecnologias e a expansão do mercado de automação no Brasil. O projeto trata de um ambiente de simulação virtual, baseado nos já existentes para projetos de arquitetura.

Objetivos

Criar um protótipo de simulação para automação corporotiva interativo e com visualização em um ambiente virtual.

Metodologia

Utilizou-se duas maletas, uma para comando e outra para potência, Raspberry para instalação dos aplicativos, dispositivos móveis para controle à distância e um notebook para visualização da simulação.

Resultados

Baseando-se em testes realizados, o projeto permite o usuário atuar através de dispositivos móveis, ambiente virtual ou pulsadores locais. De modo a visualizar e simular a futura automação dos ambientes do estabelecimento.

Considerações finais

O trabalho permitiu a união de diversas tecnologias e, além da visualização da simulação, o protótipo pode ser utilizado para antever falhas e gerar economia.

Referências

DEMCHAK, Greg; KRYGIEL, Eddy; DZAMBAZOVA, Tatjana. **Introducting Revit architecture 2009**: BIM for beginners. Indianápolis: Sybex, 2009. 498 p.

OLIVEIRA, Sérgio de. **Internet das Coisas com ESP8266, Arduino e Raspberry PI**. São Paulo: Novatec, 2017. 240 p.

Orientador(es)

MENDES, Jean

Etec Jaraguá São Paulo — Centro Paula Souza — GOVERNO DO ESTADO SÃO PAULO

APÊNDICE F - VERSÃO 1.0 DO PROJETO VIRTUS – VÍDEO DA APRESENTAÇÃO DO TCC NA ETEC GILDO MARÇAL BEZERRA BRANDÃO EM DEZEMBRO DE 2016

Vídeo 6 - Vídeo da primeira versão do projeto

Fonte: Do próprio autor