



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIAS E TECNOLOGIA DO
AMAZONAS
CAMPUS MANAUS CENTRO
DEPARTAMENTO ACADÊMICO DE INFORMAÇÃO E COMUNICAÇÃO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

MAURÍCIO DA SILVA SOARES

Sistema Web de Monitoramento Para Controle Da Glicose.

**MANAUS – AM
2022**



**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIAS E TECNOLOGIA DO
AMAZONAS
CAMPUS MANAUS CENTRO
DEPARTAMENTO ACADÊMICO DE INFORMAÇÃO E COMUNICAÇÃO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

MAURÍCIO DA SILVA SOARES

Sistema Web de Monitoramento Para Controle Da Glicose.

Trabalho de Conclusão de Curso
apresentado à banca examinadora do
Curso Superior de Graduação em Análise
e Desenvolvimento de Sistemas do
Instituto Federal do Amazonas Campus
Manaus Centro como requisito para o
cumprimento na disciplina TCC II .

Orientador(a): Dr. Renildo Viana Azevedo

**MANAUS – AM
2022**

Biblioteca do IFAM – Campus Manaus Centro

S676s Soares, Maurício da Silva.
Sistema Web de monitoramento para controle da glicose / Maurício da
Silva Soares. – Manaus, 2022.
72 p. : il. color.

Trabalho de Conclusão de Curso (Tecnologia em Análise e
Desenvolvimento de Sistema) – Instituto Federal de Educação, Ciência e
Tecnologia do Amazonas, *Campus Manaus Centro*, 2022.
Orientador: Prof. Dr. Renildo Viana Azevedo.

1. Desenvolvimento de sistema. 2. Sensor de glicose. 3. LoRaWan. I.
Azevedo, Renildo Viana. (Orient.) II. Instituto Federal de Educação,
Ciência e Tecnologia do Amazonas III. Título.

CDD 005.3

Sistema Web de Monitoramento Para Controle Da Glicose

Trabalho de conclusão de curso submetido ao Departamento Acadêmico De Informação e Comunicação para a obtenção do Grau de Tecnólogo em Análise e Desenvolvimento De Sistemas.

Orientador:

Prof. Dr. Renildo Viana Azevedo
Orientador Acadêmico
Instituto Federal de Educação,
Ciência e Tecnologia do Amazonas – IFAM

Banca Examinadora:

Prof. Miguel Bonafe Barbosa
Primeiro Examinador
Instituto Federal de Educação,
Ciência e Tecnologia do Amazonas – IFAM

Prof. Msc. Jorlene de Souza Marques
Segundo Examinador
Instituto Federal de Educação,
Ciência e Tecnologia do Amazonas – IFAM

RESUMO:

O aumento da expectativa de vida da população, inúmeras ações tornam-se necessárias para garantir ao indivíduo a qualidade de vida adequada. Com a aglomeração de pessoas, crescente urbanização e recursos gradativamente mal utilizados, as Cidades Inteligentes, apoiadas fortemente na Internet das Coisas, surgem como um ponto para a otimização de recursos e bem-estar social. Alguns estudos mostram que um nível de glicose controlada fornece benefícios clínicos para pacientes críticos. No entanto, requer leituras de glicose a cada poucas horas em uma cansativa operação. Nos últimos anos, surgiram os primeiros sistemas comerciais de monitoramento contínuo de glicose para uso ambulatorial. Contudo, há uma série de limitações que dificultam seu uso. Neste contexto, a fim de propiciar uma tecnologia assistiva, esta pesquisa objetiva a criação de um sistema web para a medição da glicose, por meio de um dispositivo de monitoramento, dispositivo se conectará a um servidor web que fará a interface entre ele e o computador, *tablet* ou *smartphone*. No presente estudo, será implementado um sistema de captação e monitoramento de glicose. Para isso, um monitor de glicose tipo Holter disponível comercialmente será usado. Um Bracelete será conectado à glicose monitor, baixando os dados coletados, para que leituras possam ser obtidas. Apresento uma arquitetura de sistemas de custo acessível para cidades inteligentes empregando a tecnologia *LoRaWan* para comunicação dos dispositivos coletores de dados. Os resultados mostraram que a proposta da metodologia é uma aplicação promissora.

Palavras-Chave: Sensor de glicose, IOT, *LoRaWan*.

ABSTRACT:

The increase in the population's life expectancy, actions become necessary to guarantee the individual an adequate quality of life. With the agglomeration of people, growing urbanization and gradually misused resources, such as Smart Cities, strongly supported by the Internet of Things, emerge as a counterpoint to the optimization of resources and social well-being. Some studies show that an isolated glucose level benefits clinical criteria for patients. However, it requires glucose readings every hour and hour in a grueling operation. In recent years, the first commercial continuous glucose monitoring systems for outpatient use have emerged. However, there are a number of limitations that make it difficult to use. In this context, in order to provide an assistive technology, this research aims to create a web system for the use of glucose, through a monitoring device, the device will connect to a web server that will make an interface between it and the computer, tablet or smartphone. In the present study, a glucose uptake and monitoring system will be implemented. For this, a commercially available Holter glucose monitor will be used. A wristband will be connected to the glucose monitor, downloading the collected data so that readings can be appropriate. I present an affordable systems architecture for smart cities employing LoRaWan technology to communicate data collector devices. The harmful results that the proposed methodology is a promising application.

Keywords: Glucose sensor, IOT, LoRaWan.

Sumário

INTRODUÇÃO	6
Problematização.....	8
Justificativa	9
Objetivos.	10
Objetivo Geral	10
Objetivos Específicos	10
Metodologia.....	11
CAPÍTULO 1 CONCEITOS E PROCESSOS ESTUDADOS	14
1.1 Monitoramento Contínuo Da Saúde	14
1.2 Dispositivos Físicos Que Monitoram E Coletam Dados	18
1.3 Assistência À Vida No Ambiente (AAL).....	20
1.4 <i>Lorawan</i>	23
1.5 Trabalhos Relacionados.....	27
1.5.1 Avaliação Da Tecnologia Lora Lpwan Para Saúde E Bem-Estar Remotos Monitoramento Internos.	27
1.5.2 Monitoramento De Saúde Baseado Em IOT Via Lorawan	27
1.5.3 Cidades Inteligentes: Um Estudo De Caso Em Monitoramento E Gestão De Resíduos.....	29
1.5.4 Lorawan Para Monitoramento De Turbina Eólica: Protótipo E Implantação Prática.....	30
CAPÍTULO 2 TECNOLOGIAS CONSIDERADAS PARA O DESENVOLVIMENTO	30
2.1 JavaScript.....	30
2.2 Arquitetura Rest.....	31
2.3 Api Rest	33
2.4 Node.js.....	34
2.5 Formato Json.....	35
2.6 PostgreSQL	35
2.7 Express.....	36
2.8 React.	37

CAPÍTULO 3 ANÁLISE E MODELAGEM DO SOFTWARE.....	38
3.1 Descrição Do Sistema.....	38
3.2 Casos De Uso	38
3.2.1 Descrição De Casos De Uso	39
3.3 Implementação Do Sistema	50
3.3.1 Arquitetura Do Sistema	51
CONSIDERAÇÕES FINAIS	57
Resultados Alcançados.....	57
Trabalhos Futuros.	58
REFERÊNCIAS.....	59
Anexo 1 Telas do sistema	66
Anexo 2 Site usado para criação de dados	70

INTRODUÇÃO

Um dos grandes desafios na área de saúde é a identificação precoce de doenças. A diabetes não tratada a tempo pode gerar consequências desabilitadoras. É causa de degeneração ocular com consequências irreversíveis em pessoas com idade produtiva, podendo chegar a cegueira total (ABRAMOFF et. al., 2008).

Segundo estimativas do Ministério da Saúde, 35% da população brasileira acima de 40 anos possuem algum tipo de Doenças Crônicas Não Transmissíveis, a diabetes cerca de 11% na mesma faixa etária é portadora da doença, em torno de 5,5 milhões pessoas (MINISTÉRIO DA SAÚDE, 2011). Tecnologias vem sendo desenvolvidas para apoiar as atividades médicas de controle e prevenção dessas doenças (ISTEPANIAN et al., 2011).

Para incentivar a promoção de saúde e empoderamento do paciente devem ser desenvolvidas a seguintes ações: instrumentalizar o paciente, dando a ele condições de acessar informações sobre sua saúde; habilitar de modo que possa fazer uso dessas informações; empoderar para que transforme essas informações em conhecimento em seu próprio benefício e engajar através de ações que o transformem em um parceiro na promoção de sua própria saúde e do meio em que convive (FERGUSON, 2000).

Os avanços na medicina têm ampliado a expectativa de vida da população. De acordo com as Nações Unidas, projeções indicam que haverá mais pessoas com 60 anos ou mais que adolescentes e jovens entre 10 e 24 anos (2,1 bilhões contra 2,0 bilhões) (UNITEDNATIONS, 2017).

Van den Berghe (2004) mostrou que manter a glicemia em uma faixa normal bastante restrita, é benéfico para os pacientes diabéticos e o mesmo procedimento poderia reduzir a mortalidade entre os pacientes críticos de uma Unidade de Terapia Intensiva (UTI). Há evidências também de benefícios clínicos para alguns grupos de pacientes específicos, como os diabéticos cardiopatas. O controle restrito de glicose, reduz a mortalidade entre pacientes diabéticos após um infarto agudo de miocárdio (MALMBERG, 1997).

O monitoramento de um ambiente por meio de sensores e tomadas de decisões de acordo com as mudanças nesse ambiente e voltadas ao bem-estar do ser humano, torna-se o ambiente em algo inteligente (RAMOS, 2007). Os chamados

ambientes inteligentes, no âmbito da computação senciente, permitem a interação inteligente e natural entre o indivíduo e o ambiente físico (COOK e DAS, 2007).

Uma das suas aplicações é justamente o desenvolvimento de sistemas de Assistência À Vida No Ambiente (*Ambient Assisted Living* - AAL). O AAL tem por objetivo principal possibilitar que pessoas com algum tipo de limitação motora e/ou sensorial, sejam auxiliadas em sua rotina diária, propiciando um estilo de vida independente e seguro o mais duradouro possível, dentro de seus ambientes pessoais (RASHIDI e MIHAILIDIS, 2013).

Atualmente, já é um consenso que os sistemas AAL devem ser entidades dinâmicas que possibilitem adições de novos sensores, interpretação de novos comportamentos e execução de novas regras conforme o perfil de cada pessoa presente em uma determinada situação e contexto (ACAMPORA, COOK, RASHIDI e VASILAKOS, 2013).

Nas Cidades Inteligentes, a coleta de informações automatizadas do ambiente em tempo real combinada com sistemas de inteligência ajudam a visualizar e descobrir padrões promovendo melhores decisões. Tem-se como exemplo desse sistema um semáforo, pois ele altera seu ciclo automaticamente devido as informações do tráfego atual e gera uma otimização de vários recursos, como de combustível e de tempo(SOUZA,2019).

A Internet das Coisas impulsionou a criação de muitos dispositivos especializados, como pedômetros *Fitbit*, *Apple Watch*, *myZeo sleep*, *Oura ring* e *Nike +* e *Jawbone UP* rastreadores de *fitness* (SWAN, 2013; LABS, 2015). A maioria dos dispositivos usa sensor de qualidade do ar, coração sensor de taxa, eletrodos biomédicos, pressão arterial tecnologias de medição, GPS e acelerômetro (LABS, 2015).

Como resultado, muitos aplicativos que coletam, relatam e respondem às informações do próprio corpo do usuário já foram desenvolvido. As pessoas medem e entendem mais sobre seus qualidade do sono, saúde, desempenho de exercícios e muitos outros Atividades. Consequentemente, as pessoas estão deixando de ser passivo para participantes ativos no diagnóstico e cuidados preventivos (HIRSCH, 2015).

De mais a mais, os padrões de conexões adotados, como fibra ótica e 4G, promovem altas taxas de transmissão atendendo as demandas de grandes volumes de dados e de baixa latência em uma cidade inteligente. No entanto, toda essa

complexidade requer equipamentos mais complexos e de maior custo, esse fato torna-se o limitador para a aplicação em larga escala devido ao seu investimento financeiro inicial(SOUZA,2019).

Ademais, compreende-se que a aplicação dos conceitos de Cidades Inteligentes é muito dependente das empresas de Tecnologias da Informação e Comunicação, TICs, para a coleta, a comunicação e a análise dos dados(SOUZA,2019). Desse modo, plataformas como da IBM© tem forte apelo a seus produtos e serviços. Entretanto, um modelo genérico de Cidades Inteligentes é impossível (REDE BRASILEIRA, 2017), mesmo com padrões tecnológicos e interoperabilidade dos dispositivos e sistemas, pois cada cidade é única, tendo em vista que possui seus problemas e suas peculiaridades. Assim, estratégias nacionais de cidade inteligente falham, já que não capitalizam os recursos locais e ignoram as necessidades e as prioridades regionais (ANGELIDOU, 2014).

PROBLEMATIZAÇÃO

As novas tecnologias de comunicação que surgiram nos últimos anos trazem novos modelos de negócios ao setor de telecomunicação e promovem novas soluções que buscam atender as demandas de Internet das Coisas, Devido às dificuldades dos métodos convencionais, novos métodos de medição de glicemia têm sido estudados nas últimas décadas. As pesquisas levaram à criação de alguns sensores comercialmente disponíveis de glicose, como o *MiniMed CGMS (Medtronic Diabetes)*, *GlucoWatch (Cygnus)* e *GlucoDay (A. Menarini)*. Entretanto, há ainda diversas limitações a serem estudadas, como a acurácia, a geração de leituras em tempo real, a aplicabilidade em UTI e a integração a sistemas de informação. Dessa forma, o problema principal da pesquisa é captar e monitorar informações sobre a leitura de glicose, através de apenas um ponto com acesso à internet, seja possível a coleta de dados de sensores em tempo real, sem dependência de uma estrutura complexa de comunicações e que permita o acesso dessas informações através de aplicações *web*.

JUSTIFICATIVA

O sistema de monitoramento de glicose aqui desenvolvido propõe integrar-se à vida digital do usuário, participando da sua rotina em conjunto com outros dispositivos (como *smartphones*, computadores e *tablets*), trazendo mais tranquilidade no cuidado de suas saúdes. Pode vir a ser utilizado futuramente, também, por pessoas ou instituições que necessitem de uma solução para monitorar ambientes críticos de suas instalações internas, como por exemplo um *data center*, e ser monitorado pela própria rede, de modo a, por exemplo, acionar alarmes ou travas automaticamente em caso de atividades anormais. Dentre a aplicação dos ambientes inteligentes, destacam-se aquelas com foco no cuidado com a saúde das pessoas, como monitoramento constante, terapias e reabilitação, bem-estar emocional e persuasivo, hospitais inteligentes e assistência à vida no ambiente. Esta última chamada de *Ambient Assisted Living* (AAL), onde o ambiente inteligente auxilia pessoas com limitações em suas atividades cotidianas, na maioria dos casos pessoas que residem sozinhas, proporcionando independência e qualidade de vida em seus lares (RASHIDI e MIHAILIDIS, 2013). Com isso, a arquitetura proposta juntamente com os conceitos de dados abertos servem como catalisadores para novas aplicações e pesquisas que necessitem de dados em tempo real do ambiente da cidade. Além disso, incentiva o desenvolvimento e a produção de sensores e de coletores específicos na indústria local para necessidades da própria região, criando assim um ecossistema de empreendedorismo e de inovação. A Figura 1 representa a ideia desta comunicação. Em qualquer área que possua uma cobertura da rede *LORAWAN*, é possível que o paciente seja monitorado por uma equipe médica e que seu histórico seja registrado para consultas futuras.

Figura 1 – Uso do sistema.



Fonte: Autor

OBJETIVOS

OBJETIVO GERAL

Criar um sistema *web* para a medição da glicose, por meio de um dispositivo de monitoramento. O dispositivo se conectará a um servidor *web* que fará a interface entre ele e o computador, *tablet* ou *smartphone*. O sistema desenvolvido *web* visa a simplicidade e a praticidade, não sendo necessária nenhuma configuração, de modo que o usuário já possa utilizar o sistema de maneira intuitiva assim que o adquirir.

OBJETIVOS ESPECÍFICOS

Os objetivos específicos consistem em:

- Estudar as ferramentas computacionais necessárias para a construção do sistema.
- Modelar as funcionalidades do sistema de recepção e gerenciamento dados fornecidos por dispositivo de medição dos níveis de glicose.

- Produzir uma aplicação web para gerenciamento e controle de dados fornecidos por dispositivos de medição de níveis de glicose para controle do diabetes.

METODOLOGIA

Para a resolução dos problemas encontrados primeiramente foram levantados alguns requisitos para a criação de um novo sistema e como o mesmo poderia ser utilizado no dia-a-dia, em sequência foi projetado uma idéia de solução, e que por final foi executado e aplicado na organização.

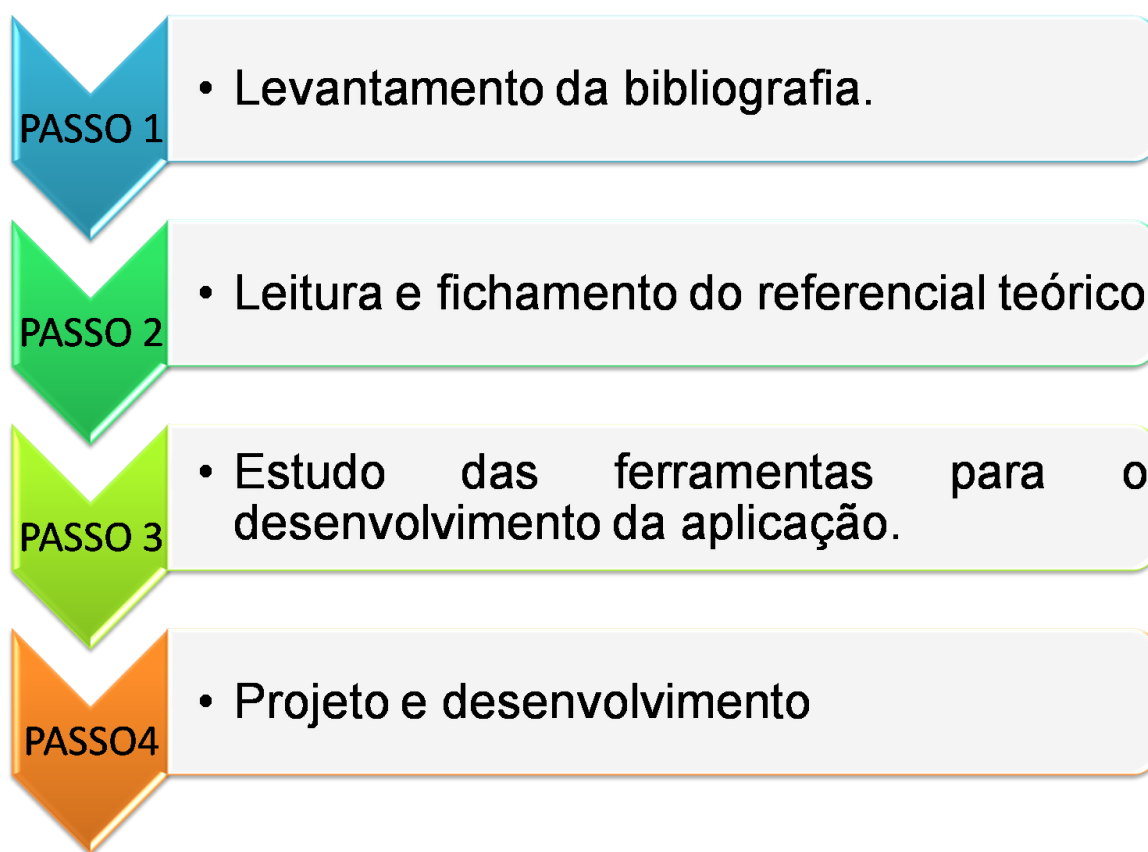
O presente trabalho, do ponto de vista da natureza pode ser classificado como uma pesquisa exploratória aplicada. Do ponto de vista técnico esse trabalho pode ser classificado como um estudo de caso.

As metodologias utilizadas consistem em coletar e analisar informações sobre medição da glicose, a fim de estudar aspectos relacionados ao tema de gerenciamento de documentos de especificação de software.

Os métodos utilizados para realizar uma pesquisa estão diretamente ligados ao problema a ser estudado e a escolha da metodologia de maneira direta dos objetivos do projeto. (CRESWELL, 2010). Esta pesquisa tem como objetivo a proposta de um sistema para monitoramento da glicose e uma proposta de integração com serviços de saúde de forma a atender o tratamento de pacientes em determinados cenários como, por exemplo, a existência de uma pré condição ou histórico de glicose elevada em um paciente que passa a maior parte de seu tempo não assistido ou que se encontra em uma zona remota dos centros urbanos. A ideia principal deste sistema é a integração com serviços de saúde já existentes para que o atendimento seja mais eficiente.

Assim, a Figura 2 apresenta a metodologia desta pesquisa, destacando os passos para cumprimento do objetivo proposto

Figura 2 – Passos tomados para o desenvolvimento deste projeto



Fonte: Autor.

Quadro 1 - Cronograma das Atividades

CRONOGRAMA DO PROJETO												Legenda
												F - Finalizado
												P - Programado
	Lista de Atividades		Agosto	Setembro	Outubro	Novembro	Dezembro					
1	Revisão bibliográfica	P										
		F										
2	Reunião com o orientador	P										
		F										
3	Escrita do TCCII	P										
		F										
4	Modelagem de Banco de dados	P										
		F										
5	Desenvolvimento da Aplicação	P										
		F										
6	Implantação da aplicação	P										
		F										
7	Testes e validação	P										
		F										
8	Elaboração da Apresentação do TCCII	P										
		F										
9	Revisão Geral do TCCII	P										
		F										
10	Preparação para a defesa do TCC II	P										
		F										
11	Apresentação da defesa do TCC II	P										
		F										
12	Entrega do Sistema	P										
		F										

Fonte: Autor

CAPÍTULO 1 CONCEITOS E PROCESSOS ESTUDADOS

Este capítulo apresenta os conceitos estudados, as ferramentas citadas anteriormente como possíveis soluções para o problema, que foram consideradas para o desenvolvimento do trabalho.

1.1 MONITORAMENTO CONTÍNUO DA SAÚDE

A comissão europeia *Information Society Technologies Programme Advisory Group* (ISTAG) apresentou o conceito de ambientes inteligentes, em inglês *Ambient Intelligence* (Aml), em 2001. Devido ao fato do grande número de tecnologias envolvidas na concepção de ambientes inteligentes, existem vários trabalhos relacionados ao tema e que foram produzidos anteriormente à própria formalização do termo (ISTAG, 2001; 2002).

O próprio termo *Internet of Things* (IoT), que se referencia a uma das tecnologias fundamentais para os ambientes inteligentes, foi criado em 1999 por Kevin Ashton, cofundador do *Auto-ID Center do Massachusetts Institute of Technology* (MIT). Ashton afirmou que a ideia original da IoT previa a conexão de todos os objetos físicos à Internet, com capacidade de capturar informações por meio de identificação por radiofrequência (RFID) e tecnologias de sensoriamento que permitia observar, identificar e compreender o mundo independentemente das pessoas e suas limitações de tempo, atenção e precisão (ASHTON, 2009).

Outros conceitos possuem algumas áreas que se sobrepõem aos ambientes inteligentes, como a computação ubíqua, computação pervasiva e a computação senciente (RAMOS, 2007). O conceito de computação ubíqua indica que há acesso a dispositivos computacionais em qualquer lugar, de forma integrada e coerente (WEISER, 1991).

Outro conceito visto comumente como sinônimo da computação ubíqua é a computação pervasiva. De acordo com Teresa Dillon, computação ubíqua é melhor considerada como estruturas subjacentes, sistemas, redes e dispositivos que são –invisíveis e estão em todos lugares, permitindo que novos dispositivos e ferramentas se conectem de forma transparente (*Plug-and-Play*), enquanto a computação pervasiva está relacionada a ferramentas se conectem de todas as partes físicas de nossas vidas, como o telefone celular, relógios inteligentes e sensores vestíveis (RAMOS, 2007).

O termo senciente, quando se refere a coisas vivas, está relacionado à capacidade de representar e entender o ambiente por meio das informações coletadas pelos canais sensoriais. Estas características inspiram a computação que se refere à capacidade de monitorar o ambiente, aprendendo com as informações coletadas e agindo de acordo com as necessidades momentâneas deste ambiente (SIMÃO, RENAUX, LINHARES e STADZISZ, 2014).

Na computação senciente, a ideia é prover sistemas que sejam adaptativos conforme os indivíduos presentes no ambiente, como, por exemplo, um sistema que controla um asilo onde os idosos têm suas particularidades e necessidades. Contudo, o sistema também deve ter a capacidade de se adaptar a um mesmo indivíduo uma vez que a condição de saúde pode ser alterada de um dia para o outro, e também ser extensível para suportar o ingresso de novos indivíduos no ambiente com suas próprias necessidades e limitações (OLIVEIRA, 2016; OLIVEIRA, et al., 2018).

A funcionalidade de um ambiente inteligente pode ser descrita como um ciclo composto por uma etapa de monitoramento do estado do ambiente, seguido pelo seu processamento para alcançar um objetivo específico ou antecipar respostas a possíveis ações, até a atuação sobre o ambiente para alterar seu estado. O monitoramento ocorre por meio da coleta de dados sensoriais, gerados por dispositivos distribuídos em uma rede de sensores que fornece informações para o sistema tomar decisões de como alterar o ambiente (COOK e DAS, 2007).

Os componentes computacionais de um ambiente inteligente agem de forma distribuída e colaborativa em um determinado contexto (LOKE, 2006) como, por exemplo, sensores que monitoram uma determinada parte do ambiente e, de forma colaborativa, analisam e geram ações no ambiente monitorado (POTDAR et al., 2009).

Várias técnicas de Inteligência Artificial (IA) podem ser utilizadas em ambientes inteligentes, tais como representação do conhecimento, aprendizado de máquina, inteligência computacional, planejamento, reconhecimento de fala, linguagem natural, visão computacional, robótica e sistemas multiagentes. Essas técnicas de IA colaboram para uma análise mais complexa e uma ação mais precisa no ambiente (RAMOS, 2007).

Os ambientes inteligentes representam a visão de futuro da computação inteligente em que ambientes apoiam as pessoas que habitam neles. Nesse novo

paradigma computacional, os dados de entradas e saídas convencionais não existem mais; ao invés disso, os sensores e processadores estão integrados nos objetos do dia a dia, trabalhando juntos em harmonia para apoiar as pessoas. Ao confiar em várias técnicas de inteligência artificial, a Aml promete a interpretação bem-sucedida da riqueza de informações contextuais obtidas desses sensores incorporados e adaptará o ambiente às necessidades do usuário de maneira transparente e antecipada (ACAMPORA, COOK, RASHIDI e VASILAKOS, 2013).

Em relação à infraestrutura, os ambientes inteligentes voltados para cuidados com a saúde podem utilizar redes chamadas BAN (*Body Area Networks*) onde vários sensores são anexados nas roupas, no corpo ou até mesmo implantados sob a pele, permitindo um monitoramento constante das condições de saúde do indivíduo; e WMSN (*Wireless Mesh Sensor Network*) composto por sensores de ambiente que coletam vários tipos de dados para deduzir atividades dos habitantes e antecipar suas necessidades, aumentando o conforto e qualidade de vida. Nas redes WMSN, cada nó, além de capturar e distribuir seus próprios dados, serve como um repetidor para os outros nós, ou seja, cada sensor colabora para a propagação de dados na rede (ACAMPORA, COOK, RASHIDI e VASILAKOS, 2013).

O desenvolvimento de sistemas voltados para ambientes inteligentes pode ser composto da utilização de alguns algoritmos e métodos, como o reconhecimento de atividades, identificação de padrões comportamentais, detecção de anomalias, planejamento das atividades diárias, apoio à decisão, e anonimato e preservação da privacidade. E as aplicações desses sistemas podem ocorrer no monitoramento contínuo de saúde, monitoramento comportamental, monitoramento para detecção de emergências, assistência na autonomia no domicílio, terapia e reabilitação, aplicações persuasivas de bem-estar, bem-estar emocional ou hospitais inteligentes (ACAMPORA, COOK, RASHIDI e VASILAKOS, 2013). A Tabela 1 apresenta um resumo dos tipos de aplicações dos ambientes inteligentes com foco na saúde, suas descrições e metodologias utilizadas.

Um dos problemas gerados por sistemas adaptativos é a definição da dependência entre as alterações do ambiente, por exemplo a climatização de um ambiente pode não interessar às regras dos demais ambientes. Katsiri propõe uma solução para lidar com os diferentes níveis de fatos e regras do ambiente chamada de Eventos Abstratos (*Abstract Events*), onde a ideia principal é que fatos primitivos, como a mudança de uma pessoa de lugar, devem afetar apenas regras primitivas

como aquelas que desejam saber se a pessoa mudou de lugar. Contudo, regras primitivas podem gerar algo chamado de fatos abstratos que quando gerados criam um evento abstrato, que pode ser utilizado de várias formas como notificar regras de outros ambientes que tem interesse neste evento (KATSIRI, 2005).

Tabela 1 – Aplicação de ambientes inteligentes no cuidado da saúde.

Tipo de Aplicação	Descrição	Medologia
Monitoramento contínuo da saúde	Uso de sensores para monitorar as condições fisiológicas do usuário (temperatura, batimentos cardíacos, etc.)	Reconhecimento de atividade
Monitoramento comportamental	Uso de sensores para monitorar o comportamento humano (uso da TV, tempo de sono, etc.)	Reconhecimento de atividade
Deteção de emergências	Uso de sensores para deteção de quedas e ferimentos.	Reconhecimento de atividade
Assistência na autonomia	Criação de um ambiente que apóie pacientes e idosos durante suas atividades diárias.	Apoio à decisão
Terapia e Reabilitação	Apoio às pessoas que necessitam serviço de reabilitação, por meios remotos ou autônomos.	Reconhecimento de atividade e apoio à decisão
Bem-estar persuasivo	Sistemas que visam mudar atitudes pessoais com o objetivo de motivá-la a um estilo de vida mais saudável.	Reconhecimento de atividade e apoio à decisão
Bem-estar emocional	Sistemas baseados em percepções neurológicas e psicológicas para analisar emoções e melhorar o bem-estar.	Reconhecimento de atividade
Hospitais inteligentes	Melhorar a comunicação entre as partes interessadas do	Apoio à decisão

hospital, por meio de
tecnologias ubíquas.

Fonte: Adaptado de ACAMPORA, COOK, RASHIDI e VASILAKOS, 2013.

1.2 DISPOSITIVOS FÍSICOS QUE MONITORAM E COLETAM DADOS

O principal aspecto da abordagem da IoT é o alto impacto que ela possui em vários tópicos do cotidiano e no comportamento das pessoas. Do ponto de vista de um usuário particular, os efeitos mais óbvios da introdução da IoT são visíveis nos campos do trabalho e no ambiente doméstico. Nesse contexto, a domótica, a vida assistida, *e-health* e a aprendizagem aprimorada são apenas alguns exemplos de possíveis cenários de aplicação nos quais o novo paradigma terá um papel de liderança no futuro próximo. Da mesma forma, do ponto de vista dos usuários de negócios, as consequências mais aparentes serão igualmente visíveis em campos como automação e manufatura industrial, logística, gerenciamento de negócios, processos, transporte inteligente de pessoas e mercadorias (ATZORI, 2010).

Para Whitmore (2015), o conceito principal de Internet das Coisas são objetos equipados com identificação, sensoriamento, acesso à rede e capacidade de processamento permitindo a comunicação com outros dispositivos através da Internet. Essa definição principal não é nova, por anos a tecnologia de RFID e rede de sensores estão sendo utilizadas pela indústria e manufatura para rastreamento de estoque.

Já segundo Yan et al., a IoT fornece uma integração de vários sensores e objetos que podem se comunicar diretamente uns com os outros sem intervenção humana. Sendo dispositivos físicos que monitoram e coletam vários tipos de dados de máquinas ou do ambiente, inclusive da vida humana (YAN, ZHANG, VASILAKOS, 2014).

Um dos aspectos cruciais que deve ser considerado em projeto de IoT são as redes de telecomunicações usadas para transmitir as informações dos sensores. A tecnologia escolhida deve ser adequada, principalmente no consumo de energia e, estando ciente de que o tráfego de dados não é alto, a demanda de taxa de transmissão também não é alta, na ordem de kbps. Essas redes são conhecidas como *LPWAN*, *Wide Area Networks Low Power* e, atualmente, são implementadas

através de soluções como *LoRaWan*, *SigFox*, *NB-IoT* e *LTE-M* (GARCIA e KLEINSCHMIDT, 2017).

Ademais, as redes celulares evoluíram muito nos últimos 15 anos, desde a rede 2G na tecnologia GPRS com taxas de dados de 9,6 *kbps*, depois *EDGE* com 547 *kbps*, passando pela *HSPA+* com 168 *Mbps* e, por fim, chegando na 4G com a tecnologia *LTE* podendo chegar a 300 *Mbps*. Essa evolução permitiu o consumo de conteúdo móvel com alta taxa de transmissão, alavancando os dispositivos móveis como *smarthphone*. No entanto, essa solução voltada para o contexto de IoT acaba sendo superdimensionada na questão de taxa de transmissão e na complexidade (LAURIDSEN et al., 2017), assim como no custo de manter um dispositivo conectado, esse ponto que até chega a inibir quando considerado milhões de dispositivos. A complexidade desses protocolos de alta taxa de transmissão gera maior necessidade de processamento e energéticas (DIAS, 2016). Para resolver essas necessidades está sendo introduzida a tecnologia *LTE-M*, que é uma variação voltada para IoT, com taxa de transmissão de até 1 *Mbps*, utilizando a mesma estrutura das rede celular, simplificando assim a implantação, mas acaba mantendo o custo elevado para cada dispositivo (GARCIA, 2017).

O *NB-IoT* está sendo padronizado pela 3GPP e tem o intuito de resolver o problema do consumo de energia e conexões de longo alcance com taxas de transmissão de 170 *kbps* para *downlink* e 250 *kbps* de *uplink*, além de simplificar a implantação, pois é um protocolo que utiliza a infraestrutura das operadoras de telefonia para prover o serviço aos usuários (DIAS, 2016).

Já o padrão *SIGFOX* surge como uma alternativa à tecnologia celular, para dispositivos de baixo custo, com capacidade de processamento reduzido e transmissão em longas distâncias. Ele possui um custo por mensagem enviada, mas é muito baixo em comparação à rede celular. Mesmo operando em faixas de frequência *ISM*, a rede *SIGFOX* é fechada, sendo necessário uma concessionária distribuir o sinal na região. No Brasil, a rede já está em processo de implantação pela concessionária WND (WND, 2018). O padrão provê mensagens de 12 *bytes* por vez, sendo possível o envio de 120 mensagens por dias, que permite utilização em vários dispositivos que necessitam de pouca banda, assim, dispositivos que utilizam simples baterias AA podem ter autonomia por até 10 anos, segundo o fabricante (WND, 2018).

Por sua vez, o *LoRaWan* se destaca sobre os padrões anteriores por não ser vinculado com a infraestrutura atual de rede celular e não depender de um operador exclusivo da tecnologia. Dessa forma, ele permite projetos customizados e aderente as necessidades locais, proporcionando também menor custo de manutenção. Em contrapartida, apresenta complexidade maior para implantação, pois é necessário instalar toda a infraestrutura de comunicação. O padrão *LoRaWan* terá maior detalhamento em sessão posterior.

1.3 ASSISTÊNCIA À VIDA NO AMBIENTE – *AMBIENT ASSISTED LIVING (AAL)*

Os sistemas AAL, quando desenvolvidos como ambientes inteligentes, podem ser usados na prevenção, tratamento e melhora das condições de saúde dos pacientes. Sua aplicação pode ocorrer em atividades de monitoramento das condições de saúde, notificações e controle no uso de medicamentos, detecção de quedas, monitoramento de segurança, auxílio nas tarefas cotidianas, facilidade na comunicação do idoso com familiares e enfermeiros, e até mesmo na geração e acompanhamento de um diário para indivíduos com demência (RASHIDI e MIHAILIDIS, 2013).

Um dos pontos fundamentais na concepção de sistemas AAL está relacionado ao design da aplicação uma vez que a maioria dos usuários teriam alguma limitação, e.g. são idosos com dificuldades visuais e na coordenação motora. É importante projetar uma interface com botões e letras grandes e com o uso de figuras intuitivas, de forma que seja a mais simples e amigável possível para se operar (RASHIDI e MIHAILIDIS, 2013).

Os sensores vestíveis, que monitoram as condições de saúde do usuário, não devem criar nenhum tipo de desconforto ou limitação nos movimentos. Uma alternativa é a utilização de dispositivos que o idoso já esteja familiarizado; por exemplo, celulares, *tablets* ou relógios e jóias inteligentes. Não obstante, é evidente a resistência da maioria dos idosos em relação à tecnologia, sendo necessária uma preocupação em desenvolver soluções que exerçam seu papel com a menor dependência de ações diretas desses usuários (RASHIDI e MIHAILIDIS, 2013).

Evidentemente, o sistema AAL jamais deve substituir completamente a necessidade de cuidado humano, pois, nesse caso, poderia resultar inclusive em isolamento do usuário (RASHIDI e MIHAILIDIS, 2013).

A Tabela 2 mostra um resumo de vários projetos que se destinam à vida assistida. Por exemplo, o projeto CASAS da *Washington State University* que fornece um ambiente assistencial não invasivo para pacientes com demência. O projeto *-Aging in Place* da Universidade do *Missouri*, visa fornecer um modelo de cuidados de longo prazo para idosos em termos de saúde e de suporte. O *Elite Care* é uma instalação de vida assistida equipada com sensores para monitorar indicadores como tempo na cama, peso corporal e inquietação do sono. O projeto *Aware Home*, da *Georgia Tech*, emprega uma variedade de sensores, como sensores de piso inteligente, além de robôs assistentes para monitorar e ajudar idosos (RASHIDI e MIHAILIDIS, 2013).

Tabela 2 – Projetos para ambientes inteligentes assistidos.

Projeto	Instituição
<i>Aging In Place</i>	<i>U. of Missouri</i>
<i>Aware Home</i>	<i>Geogria Tech</i>
<i>CareLab</i>	<i>Germany</i>
<i>CareNet (MIDAS)</i>	<i>U. of Wales, UK</i>
<i>CASAS</i>	<i>Washington State U.</i>
<i>DOMUS</i>	<i>U. of Sherbrooke</i>
<i>Elite Care</i>	<i>OHSU</i>
<i>ENABLE</i>	<i>Netherlands</i>
<i>Gator</i>	<i>UF</i>
<i>HIS</i>	<i>Grenoble U., France</i>
<i>MavHome</i>	<i>U. of Texas at Arlington</i>
<i>Millennium Home</i>	<i>Brunel U.</i>
<i>ProSAFE</i>	<i>LAAS, France</i>
<i>SELF</i>	<i>ETL, Japan</i>
<i>Smart Medical Home</i>	<i>Rochester U.</i>
<i>Ubiquitous Home</i>	<i>UCG, Japan</i>
<i>WTH</i>	<i>JMITI, Japan</i>

Fonte: Adaptado de RASHIDI e MIHAILIDIS, 2013.

Alguns ambientes assistidos podem ser compostos por robôs que auxiliam na independência de pessoas idosas e com limitações. Neste cenário temos a casa robótica inteligente -*Intelligent Sweet Home*®, desenvolvido em KAIST na Korea, que possui o controle de cama inteligente, cadeira de rodas inteligente e guincho robótico utilizado para transferência fácil do usuário entre a cama e a cadeira de rodas (KWANG-HYUN et al., 2007).

Uma aplicação comum dos sistemas AAL é o monitoramento das atividades diárias dos idosos, com o controle da duração, frequência e sensação de bem-estar do usuário após cada atividade. Este monitoramento pode auxiliar na identificação de anomalias na rotina diária e em sugestões de atividades que proporcionam bem-estar individual (HAIGH, et al., 2004; SEBESTYEN, STOICA e HANGAN, 2016; SURYADEVARA e MUKHOPADHYAY, 2014).

O monitoramento constante do padrão de movimentação do idoso em sua residência, e da leitura de sinais como temperatura do corpo e número de batimentos cardíacos por minuto, possibilita a detecção de episódios de Transtorno Depressivo Maior (TDM), chamado também de depressão. A rápida identificação da doença permite o envio de notificações à familiares, cuidadores ou médicos, que podem tomar medidas antecipada no tratamento da doença (EDWING, FERRUZCA e IVAN, 2015).

Em alguns cenários específicos, sistemas AAL podem monitorar apenas o ambiente, sem considerar o monitoramento constante do indivíduo nele inserido, para situações especiais ou tratamento/prevenção de doenças específicas, como soluções que monitoram a condição do ar, analisando a temperatura, umidade, concentração de partículas, níveis de monóxido de carbono e de outros gases. A partir da análise da qualidade do ar, é possível o envio de alertas para cuidadores de idosos para os casos de residências que possuem pessoas com histórico de crise asmática (SILVA, et al., 2015).

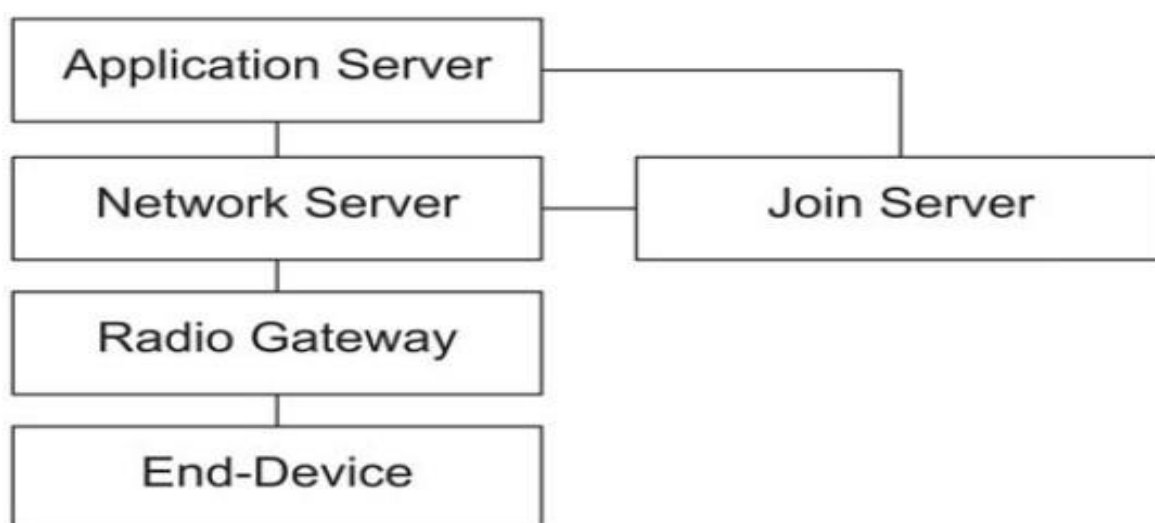
Em virtude do grande volume de dados de sensores trafegado nos sistemas AAL, e da necessidade de poder de processamento para desempenhar tarefas de análise dos dados, algumas soluções propõem que este processamento seja feito em ambiente *cloud* (nuvem), por permitir uma forma fácil e rápida de escalonamento

de recursos computacionais. Contudo essa mudança do local de processamento implica em uma necessidade de comunicação constante e ininterrupta com nuvem, pois caso contrário o sistema pode ficar sem uma resposta do retorno do processamento (FORKAN, KHALIL e TARI, 2014).

1.4 LORAWAN

Por definição *LoRa* é a tecnologia de rádio frequência enquanto *LoRaWan* é o protocolo de rede, sendo uma *Low Power Wide Area Network*, *LPWAN* que consiste em uma arquitetura de rede voltada ao baixo consumo de energia, baixa taxa de dados e longo alcance, pontos chave para IoT, provendo comunicação segura bidirecional, mobilidade e serviço de localização. Para mais, utiliza a topologia estrela, com um *gateway* como ponte entre os *end-devices* e o *Network Server*, esse que por sua vez realiza o roteamento dos pacotes de cada dispositivo para o servidor de aplicação (ALLIANCE, 2018). Outro ponto do *LoRaWan* é que ele é aberto, isto é, permite que qualquer empresa produza dispositivos, entretanto, eles devem ser homologados pela *LoRa Alliance* (ALLIANCE, 2018), já que o transmissor *LoRa* possui patentes pela *SemTech*. A especificação ainda prevê *Join Server* que gerencia a autenticação dos dispositivos na rede, o modelo de referência de rede pode ser observado na Figura 3.

Figura 3. Modelo de Referencia de Rede LoRa.



Fonte: Adaptado de ALLIANCE, 2018.

Os *End-Devices* podem ser sensores ou atuadores que são conectados sem fio na rede *LoRaWan* através do Radio Gateway. A transmissão utiliza o conceito de

chirp spread spectrum de modulação que provê resistência e robustez contra interferência e efeito Doppler, sendo esse o grande diferencial da tecnologia *LoRa* (ADELANTADO et al.,2016). Os demais parâmetros de comunicação são:

- ❖ *Bandwidth* (BW): podendo variar de 125 kHz e 500 kHz no continente americano (ADELANTADO et al.,2016).
- ❖ *Spreading Factor* (SF): podendo variar de 7 a 12, definindo o tempo e a taxa de transmissão que a informação será transmitida. Quanto menor o SF menor será o tempo de transmissão e maior será a taxa, assim abrange um alcance menor. Já quanto maior o SF maior será o tempo de transmissão e menor será a taxa de transmissão, contudo ele abrange um alcance maior (ADELANTADO et al.,2016).
- ❖ *TXPower*: indica a potencia do sinal, no padrão americano varia de 10 dBm a 30 dBm (ALLIANCE, 2018).

A combinação *bandwidth* e *Spreading Factor* proporcionam taxa de transmissão entre de 0.3 kb/s a 27 kb/s (ADELANTADO et al.,2016), ainda há a parametrização de *Adaptative Data Rate*, *ADR*, que permite que o *SF* e o *TXPower* do *end-device* seja ajustada para otimizar o consumo de energia. Isso na prática permite que na mesma área de abrangência de um *gateway* tenha diferentes taxas de dados, pois os dispositivos localizados próximos do *gateway* podem utilizar *SF* menor, assim economiza-se energia.

Os dispositivos *LoRa* são categorizados em classes para utilização adequada em cada projeto, sempre voltado para o menor consumo de energia:

- ❖ Classe A: dispositivos bidirecionais, mas com possibilidade de receber mensagens (*downlink*) restritas logo após o envio (*uplink*). Nessa forma de operação reduz o consumo de energia, pois durante o resto do tempo o módulo *LoRa* estará desligado.
- ❖ Classe B: dispositivos bidirecionais, mas tem possibilidade de receber mensagem com tarefas sincronizadas de tempo em tempo. Ao enviar uma mensagem pode ser criada uma tarefa para sincronizar o próximo horário em que o dispositivo estará ouvindo.
- ❖ Classe C: dispositivo bidirecional, mas com a possibilidade de recebimento de mensagens a quase qualquer momento.

Na Tabela 3 resume-se a diferença das classes de dispositivos *LoRaWan* (GARCIA, 2017).

Tabela 3. Comparativo entre classes de dispositivos LoRaWan.

Classe	A	B	C
Mensagens em Multicast	Não	Sim	Sim
Latência	Alta	Baixo	Não
Dispositivo inicia a comunicação	Sim	Sim	Sim
Servidor inicia a comunicação	Não	Sim, com intervalo	Sim, constante
Consumo de Energia	Baixo	Moderado	Alto

Fonte: Adaptado de SCHIOPOIU, 2018.

A utilização de faixas de frequência de rádio é limitada pelos governos locais dos países para definir regras, organização e evitar problemas de interferências entre os equipamentos de uso crítico. Portanto, há um acordo internacional que define algumas frequências sem necessidade de regulamentação para incentivar o desenvolvimento industrial, científico e médico, esse acordo é denominado *ISM band*, industrial, *scientific and medical*. No Brasil o órgão regulador de telecomunicação é a Agência Nacional de Telecomunicação, ANATEL, dentre suas responsabilidades está as definições das faixas de frequência que necessitam de cadastramento para operação, logo, segundo a resolução 506 da Anatel, no capítulo II, define as faixas de frequência restritas, que não necessitam de cadastramento junto ao órgão: 902-907,5 MHz, 915-928 MHz, 2.400-2.483,5 MHz, 5.725-5.875 MHz e 24,00-24,25 GHz (RESOLUÇÃO nº 506, 2018).

A instrução da *LoRa Alliance* para o continente americano é utilizar a faixa de frequência 902 a 928 MHz, como pode ser observado as faixas entre 907,5 a 915 MHz não podem ser utilizadas, assim, ao utilizar dispositivos do padrão americano deve-se tomar o cuidado de inutilizar esses canais via software na aplicação. Sabe-se, contudo, que essa limitação já ocorre em outros países que padronizaram a distribuição das faixas de frequência de tal modo que não sejam inutilizados canais, como é o caso do padrão australiano. Entretanto, ainda não há um padrão brasileiro definido (ALLIANCE, 2018).

Os pacotes ao chegarem no *Gateway* são encaminhados sem nenhum processamento ou interpretação ao *Network Server*, esse que é o centro da topologia estrela, tendo como funções:

- ❖ . Verificação de endereço do *End-Device*
- ❖ Autenticação dos *frames*
- ❖ Gerenciar o *Adaptive Data Rate*, *ADR*
- ❖ Responder a todas requisições dos *End-Device*
- ❖ Encaminhar ao *Application Servers* todas as mensagens de *uplink*
- ❖ Agendar e encaminhar ao *End-Device* todas as mensagens de *downlink* dos *Application Servers*.
- ❖ Encaminhar para o *Join Server* as requisições de autenticação dos *End-Devices*.

Um dos pontos críticos da infraestrutura para IoT é a segurança, dessa forma o protocolo *LoRaWan* possui um mecanismo criptografia baseado no padrão AES, e possui dois métodos de autenticação na rede. No método ABP, *Activation by Personalization*, o *End-Device* recebe os identificadores no processo de fabricação ou codificação, sendo únicos na rede local, assim não é possível transportar este *End-Device* para outra rede *LoRaWan*, sendo necessário alterar esses identificadores no código do dispositivo. Os parâmetros definidos nesse método são:

- ❖ *Device Address (DevAddr)*: Identificador único do dispositivo com 32 bits. Presente em cada data *frame*.
- ❖ *Network Session Key (NwkSKey)*: Identificador único na camada rede com 128-bit AES de encriptação, garantindo a integridade de mensagens entre o *end node* e o *Network Server*.
- ❖ *Application Session Key (AppSKey)*: Identificador único na camada de aplicação com 128-bit AES de encriptação, usado para criptografia das mensagens na aplicação.

No caso do método *Over-the-Air (OTAA)*, o *End-Device* possui um identificador único global *DevEUI*, através deste envia-se uma requisição de *join*, sendo essa não criptografada, caso aceito na rede o *DevAddr* é gerado e enviado para o dispositivo que irá utilizar estes para troca das próximas mensagens criptografadas (ALLIANCE, 2018). O *Join Server* gerencia a autenticação apenas neste método.

No *Application Server* são tratadas as mensagens, *payloads*, dos dispositivos para o usuário final sendo descriptografada utilizando o *AppSKey*, é apenas nessa camada que as informações geradas pelos dispositivos são de fato acessadas.

1.5 TRABALHOS RELACIONADOS

1.5.1 Avaliação da tecnologia LoRa LPWAN para saúde e bem-estar remotos Monitoramento internos.

O trabalho de Juha Petajajarvi et al. (2017) relata os experimentos utilizando dispositivos comerciais da tecnologia LoRaWan no campus da Universidade de Oulu na Finlândia, aplicando diferentes SF, Bandwidth e TXPower. No caso desse estudo, foi utilizado uma potencia de transmissão de 14 dBm e SF = 12, sendo possível cobrir toda a área do campus, 570 x 320 metros, com uma media de 96,7 % de sucesso de transmissão.

Sao relatados testes tanto no exterior quanto no interior de construcoes entre 55 e 370 metros de distancia, utilizando *spreading factors* de 7 a 10 avaliado o percentual de pacotes com erro e o RSSI (*Received Signal Strength*) em uma tabela e grafico em forma de mapa de calor. Ao final e realizado um breve relato do consumo de energia com essa variacao de *spreading factors*, concluindo que o consumo e maior devido ao tempo que o transmissor permanece ativo em SF maiores.

Sendo um dos primeiros trabalhos práticos da utilização da tecnologia *LoRa em ambientes* internos, conclui-se que e possível a transmissão *indoor em 300 metros mesmo tendo* perdas e atrasos. Além disso, e viável e possui potencial a utilização da tecnologia para monitoramento do bem-estar e controle de sinais não críticos, como atividades físicas, localização, monitoramento de animais de estimação e gerenciamento pessoal, etc.

1.5.2 Monitoramento de saúde baseado em IoT via LoRaWAN.

Os autores Mdhaffar, Afef, et al.(2017) da University of Sousse, Tunisia, apresentam um novo coletor de baixo custo para dados médicos utilizando *LoRaWan para transmissão dos* dados de pressão sanguínea, glicose e

temperatura. O denominado *IOT4HC* é baseado em dispositivos de maior custo que utilizam outras tecnologias de transmissão. Entretanto, nesse protótipo simplifica o dispositivo permitindo que pessoas com problemas de saúde específicos e que estão localizadas em áreas rurais ou sem acesso a internet não precisem se deslocar para coleta de dados preventivos.

O projeto abrange desde o sensor até o acesso web pelos usuários e médicos, porém, a pesquisa abordada no artigo é limitada ao dispositivo e se divide em dois subgrupos. O primeiro levantou a cobertura da área do *gateway LoRaWan*, que na área urbana teve resultados de até 1,89 Km em uma direção e 0,73 km em uma região de alta densidade urbana; contudo, na área rural alcançou 33 km² utilizando uma antena de 3 dBi de ganho.

Já o segundo grupo focou no consumo energético do dispositivo utilizando *LoRaWan* comparando com a tecnologia GPRS, o qual mesmo em *idle mode* o dispositivo equipado com GPRS consome aproximadamente 20mA, enquanto com *LoRa* consome aproximadamente 4.7 mA, e, utilizando uma estimativa de consumo, com uma *power bank* de 2200 mA, teria uma autonomia de até 10.875 dias.

Com um alcance na área rural de 33 km² e como em experimentos mostrou 10 vezes menor consumo energético que a transmissão por GPRS, o equipamento se mostra interessante para áreas remotas. No entanto, o não envio contínuo de dados não permite a utilização da solução para pacientes críticos, assim uma solução alternativa deveria ser implementada para esses casos.

1.5.3 Cidades inteligentes: um estudo de caso em monitoramento e gestão de resíduos.

O trabalho de Castro Lundin, Ozkil e Schuldt-Jensen (2017) relata a construção de um sistema de monitoramento de lixeiras na Technical University of Denmark, abordando o gerenciamento e a otimização do controle de resíduos como um dos principais itens para cidades inteligentes. Porém, as soluções comerciais tornam inviável a aplicação em larga escala devido ao preço muito elevado.

Na pesquisa é descrito a construção de todo o sistema de forma resumida, os dispositivos utilizam sensores ultra-sônicos para determinar o nível que a lixeira se encontra, os equipamentos para a montagem do gateway e as tecnologias e os protocolos para a implementação do *backend* e *frontend*.

Ainda são analisados os dados coletados durante a fase de teste, nos quais são verificados que algumas lixeiras eram preenchidas antes do recolhimento previsto, devido a sua localização próxima a uma lanchonete, enquanto outras permaneciam com o nível baixo. Esse comportamento já era previsto pelos coletores de lixo devido à experiência e o conhecimento do local, assim não foram aplicados predição nos resultados obtidos. Entretanto, o autor argumenta que no caso da indisponibilidade dos atuais coletores, os substitutos não teriam esse conhecimento e o ambiente ficaria à mercê do acúmulo de lixo bem como qualquer outro evento inesperado também causaria problemas. Logo, a implantação de um sistema de monitoramento inteligente permite que se utilize uma abordagem pró-ativa e não reativa.

Conclui-se que o sistema inteligente, mesmo sendo simples e barato, mostra sua importância para melhorar os serviços oferecidos aos cidadãos, e que mesmo não substituindo o conhecimento dos coletores, melhora a reação em algumas situações atípicas. A questão do custo por lixeira também foi respeitada durante o estudo, não ultrapassando 100 dólares, sendo esse um dos principais fatores para a implantação desse sistema em larga escala.

1.5.4 *LoRaWAN* para monitoramento de turbina eólica: protótipo e implantação prática.

Os autores Mikhaylov, Moiz, et al.(2018) da Universidade de Oulu, na Finlândia, apresentam um protótipo para monitoramento de turbina eólicas utilizando a tecnologia *LoRaWan*. Abordando como as redes de energia elétrica tornaram-se o principal sistema da civilização atual.

Foi desenvolvido dispositivo de coleta próprio em conjunto com gateway comercial *MultiTech*, foi documentado a utilização de *network server* privado, destacou-se ainda a implementação da aplicação em formato de painel para acompanhamento das informações.

A coleta foi realizada durante nove meses, registrando cerca de meio milhão de dados. Ao fim, afirma que a tecnologia *LoRaWan* prove soluções simples e eficiente em custos, permitindo o monitoramento não crítico no contexto de redes elétricas, mas com ressalvas por o artigo não aborda o aspecto da escalabilidade.

CAPÍTULO 2 TECNOLOGIAS CONSIDERADAS PARA O DESENVOLVIMENTO

Este capítulo apresenta os processos e conceitos que foram estudados e as tecnologias que foram consideradas para o desenvolvimento do trabalho.

2.1 JavaScript

Apesar do nome, JavaScript não tem relação alguma com a plataforma Java. Enquanto Javascript é uma plataforma *client-side*, Java é uma linguagem *server-side*. Javascript é uma linguagem baseada em objetos, ou seja, sua tecnologia tenta tratar as aplicações como um conjunto de objetos, ou partes individuais, que executam tarefas distintas.(MORRISON,2008)

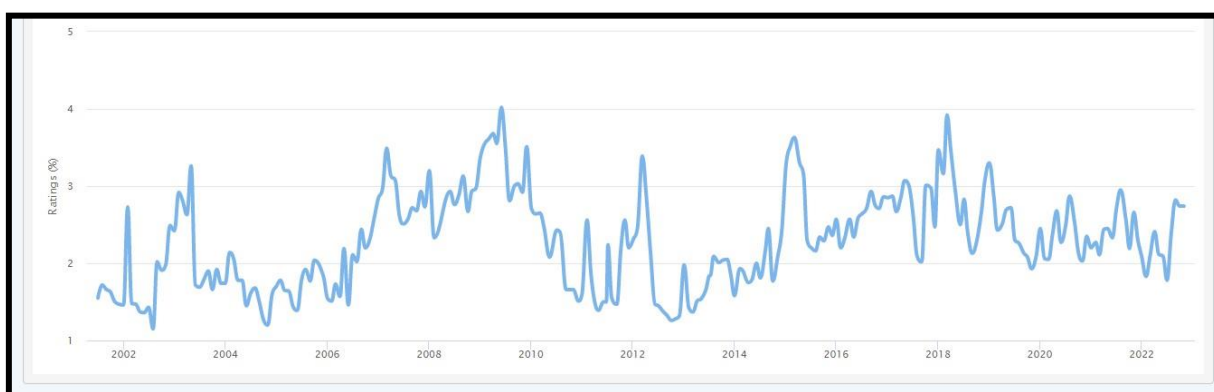
Criada no ano de 1995 pelo programador Brendan Eich, até então funcionário da NetScape Communications Corporation, Javascript foi inicialmente projetada para rodar nos navegadores NetScape, como proposta de oferecer aos desenvolvedores web solução para tornar as páginas web mais dinâmicas para tornar o uso mais agradável. Após um ano depois que foi lançada, a empresa Microsoft fez porte da linguagem, influenciando assim para que ela se tornasse uma das tecnologias mais importantes e mais utilizadas por desenvolvedores web. (MORRISON,2008)

A linguagem de programação Javascript possui várias características chave da plataforma. Entre elas temos as seguintes características:

- **CÓDIGO NO HTML:** O código fonte de Javascript pode ser incluso dentro do próprio código HTML, pois existe uma Tag que serve somente para isso, separar a marcação do Javascript dentro do seu código. (MORRISON,2008)
- **LINGUAGEM INTERPRETADA:** Javascript é uma linguagem interpretada, ou também chamada de linguagem de script, ou seja, os comandos de Javascript são executados um por um pelo interpretador, dispensando assim a necessidade de copilar o código, como acontece em outras linguagens de programação. (MORRISON,2008)
- **ORIENTADA A OBJETO:** o *Javascript* é uma linguagem orientada a objeto, ou seja, ela trata seu programa por partes individuais, objetos, e não sequencialmente. Ela fica agrupando os objetos similares em classe. (MORRISON,2008)

- SINTAXE: A sintaxe desta linguagem é similar a sintaxe de C, C++ e Java, sendo um facilitador para desenvolvedores que já tem alguma noção nessas outras tecnologias(MORRISON,2008).
- DIRIGIDA POR EVENTOS: Umas das mais importantes características do Javascript, a linguagem sempre separa em eventos comandos importantes do código, como por exemplo clicar em um botão ou preencher o campo de um formulário. (MORRISON,2008)
- INDEPEDENCIA: Javascript é uma linguagem independente de plataforma, isso porque os comandos de seu código são processados pelo navegador do usuário, ou seja, o código pode ser processado por um navegador instalado no Windows tanto como um navegador instalado no Linux. (MORRISON,2008)

Figura 4 - Evolução do JavaScript entre 2002 e 2022



Fonte: (TIOBE, 2022)

O gráfico da figura 4 deixa visível a evolução da popularidade do Javascript, em relação ao uso da linguagem, de 2002 até 2022. As características desta linguagem que justificam essa evolução são várias: sintaxe, suporte, tipagem, e outras.

2.2 Arquitetura Rest

O modelo arquitetural de Transferência de Estado Representacional (REST) para sistemas de hipermídia distribuídos, desenvolvido por Fielding (2000), fornece um conjunto de restrições arquiteturais que, quando aplicadas como um todo, enfatizam a escalabilidade de interações de componentes, generalidade de interfaces, implantação independente de componentes e componentes

intermediários para reduzir a latência de interação, reforçar a segurança e encapsular sistemas legados.

Conforme Fielding (2000) definiu em seu trabalho, as restrições do modelo *REST* incluem:

1. O modelo Arquitetural Cliente-Servidor, que separa as preocupações da interface do usuário das preocupações com armazenamento de dados, melhoramos a portabilidade da interface do usuário em várias plataformas e aprimoramos a escalabilidade simplificando os componentes do servidor. Essa separação permite que os componentes evoluem de forma independente.
2. A comunicação *stateless* entre Cliente e Servidor, cada solicitação do cliente para o servidor deve conter todas as informações necessárias para entender a solicitação e não pode aproveitar qualquer contexto armazenado no servidor. O estado da sessão é, portanto, mantido inteiramente no cliente.
3. Cache, os dados em uma resposta a uma solicitação sejam rotulados implícita ou explicitamente como armazenáveis em cache ou não. Se uma resposta puder ser armazenada em cache, o cache do cliente terá o direito de reutilizar esses dados de resposta para solicitações equivalentes posteriores.
4. Interface Uniforme entre componentes, essa restrição é dividida em quatro: identificação de recursos; manipulação de recursos através de representações; mensagens auto-descritivas; e, hipermídia como o motor do estado de aplicativo.
5. Sistema em camadas, permite que uma arquitetura seja composta de camadas hierárquicas, restringindo o comportamento do componente, de forma que cada componente não possa ver além da camada imediata com a qual eles estão interagindo.
6. Código sob demanda, a funcionalidade do cliente seja estendida baixando e executando o código na forma de scripts. Isso simplifica os clientes reduzindo o número de recursos necessários para serem pré-implementados melhorando a extensibilidade do sistema.

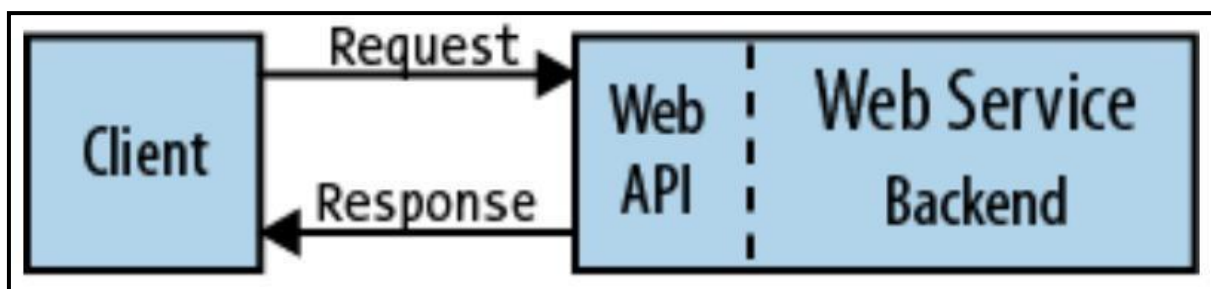
Essa tecnologia foi considerada para esse trabalho por ser um modelo arquitetural de sistemas *WEB* que responde as requisições de forma mais rápida,

devida a comunicação *stateless* entre cliente e servidor, e a identificação clara dos recursos, tornando o desenvolvimento e a manutenção da aplicação mais simples.

2.3 API Rest

Os *Web services* são servidores criados para uso específico que atendem às necessidades de um site ou de qualquer outro aplicativo. Programas que rodam no lado do cliente usam interfaces de programação de aplicativos (APIs) para se comunicar com serviços da *web*. De um modo geral, uma API expõe um conjunto de dados e funções para facilitar as interações entre os programas de computador e permitir que eles troquem informações. A figura 5 mostra como uma API da Web é a face de um serviço da Web, ouvindo diretamente e respondendo a solicitações de clientes. (MASSE, 2011).

Figura 5 - API WEB



Fonte: (MASSE, 2011).

Ter uma API REST, isto é, que implementa os princípios da arquitetura REST, torna um serviço da Web "RESTful". As APIs REST usam URIs (Uniform Resource Identifiers) para endereçar recursos. (MASSE, 2011, tradução do próprio autor)

Um dos princípios de design básico da REST estabelece um mapeamento individual entre as operações criar, ler, atualizar e excluir (CRUD) e os métodos HTTP. De acordo com este mapeamento: Para criar um recurso no servidor, use POST; Para recuperar um recurso, use GET.; Para alterar o estado de um recurso ou atualizá-lo, use PUT; Para remover ou excluir um recurso, use DELETE. (IBM, 2015).

Essa tecnologia foi considerada para esse trabalho pois, como implementa o modelo REST, possibilita uma identificação clara dos recursos, e fácil acesso a eles através dos métodos HTTP.

2.4 NodeJS

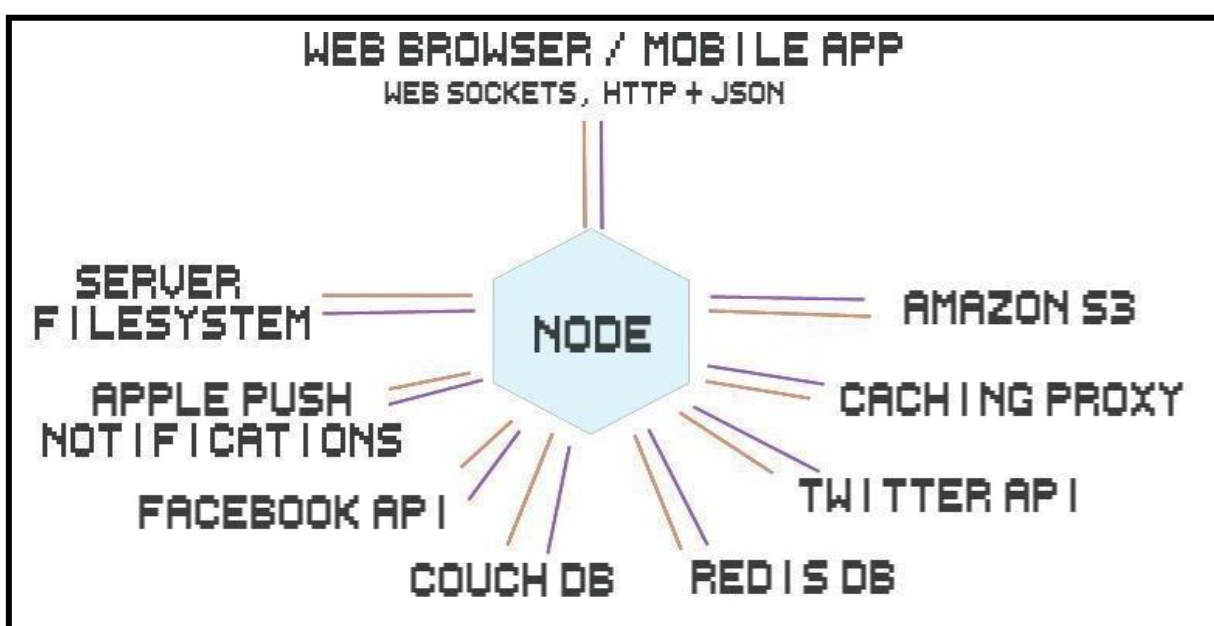
Baseado em redes utilizando Javascript e V8 Javascript Engine, o *Node.js* é uma plataforma para desenvolvimento server-side, onde dá para se criar uma grande variedade de projetos web utilizando o código Javascript. (KROL,2015).

Seu principal objetivo é fornecer de maneira fácil, um método para construção de aplicações de rede escaláveis. Além de resolver a questão de gerenciamento de memória que ocorre com outras plataformas, como Java e PHP, porque invés de criar uma nova tarefa para cada conexão, o *Node.js* dispara um evento executando dentro da engine de processos dele mesmo. Ou seja, ele impede que seja criado uma fila de tarefas. Este método permite afirmar que *Node.js* não dará nunca *deadlock*, pois não há bloqueios permitidos. (PEREIRA,2015).

Apesar de ser um servidor web, não se pode comparar o *Node.js* com base no *Apache* ou *Tomcat* (servidores webs comuns utilizados até o momento), pois enquanto estes servidores são *ready-to-intall*, servidores que instalam as aplicações imediatamente, o *Node.js* tem o conceito de módulos que podem ser adicionados no seu núcleo. (MOREIRA,2016)

Node.js também trabalha com o conceito de Orientação a Evento, onde há um laço de repetição e diversos pequenos tratadores de eventos, fazendo assim que o servidor nunca deixe ocorrer *overlock*.(PEREIRA,2015).

Figura 6 – Ilustração: Diagrama do Servidor NodeJs



Fonte: Pereira (2015)

2.5 Formato JSON

JavaScript Object Notation(JSON) é uma formatação leve de troca de dados. JSON é em formato texto e completamente independente de linguagem e é constituído de duas estruturas: Uma coleção de pares nome/valor; e uma lista ordenada de valores. (JSON, 2019) Esse formato é largamente utilizado para recuperar dados do servidor para uma aplicação front end por ser mais leve que o XML e ter uma sintaxe simples. Na figura 7, é mostrado como representar uma lista de objetos com seus atributos e valores.

Figura 7 - Representação de uma Lista de Objetos em JSON

```
[
  {
    "description": "quarter",
    "mode": "REQUIRED",
    "name": "qtr",
    "type": "STRING"
  },
  {
    "description": "sales representative",
    "mode": "NULLABLE",
    "name": "rep",
    "type": "STRING"
  },
  {
    "description": "total sales",
    "mode": "NULLABLE",
    "name": "sales",
    "type": "INTEGER"
  }
]
```

Fonte: (-Como carregar dados JSON a partir do Cloud Storage | BigQuery|, 2021)

2.6 PostgreSQL

PostgreSQL é um poderoso sistema de banco de dados objeto-relacional de código-fonte aberto que usa e estende a linguagem SQL combinada com muitos recursos que armazenam e dimensionam com segurança as cargas de trabalho de dados mais complicadas. (PostgreSQL, 2022)

Mesmo sendo um dos bancos de dados mais utilizados em todo mundo, *PostgreSQL* continua em constante desenvolvimento, com atualizações frequentes (TEIXEIRA, 2017).

Uma das funcionalidades mais relevantes do *PostgreSQL* para esse trabalho é a capacidade de armazenar objetos do tipo *JSON* e pesquisar dentro do *JSON*

armazenado com uma sintaxe simples de manipulação. A figura 8 apresenta uma consulta SQL onde a sintaxe para manipular valores *JSON* é utilizada. Nesse exemplo, a coluna álbum que pertence a tabela *records* é responsável por guardar um objeto *JSON*. Para acessar o valor do atributo nome do objeto armazenada utiliza-se a sintaxe `-->>||`.

Figura 8 - Consulta SQL de informações armazenadas em formato JSON

```
SELECT
    album-->>'Name' AS album,
    publicationdate AS date
FROM records
WHERE
    artist LIKE '%Hendrix%' AND
    publicationdate > '2010/1/1' AND
    Album-->'Track 2'-->>'Name' LIKE '%Nept%'
```

Fonte:(EnterpriseDB, 2021)

2.7 Express

A página oficial do *Express* o descreve como "um *framework NodeJs* minimalista e flexível, provendo um conjunto de recursos robustos para a construção de aplicações web" (EXPRESS, 2022). Foi lançado em novembro de 2010, desenvolvido inicialmente por TJ Holowaychuk, é atualmente mantido pela *StrongLoop* (EXPRESS, 2022).

Segundo Brown (2014), uma das características mais atraentes do *framework Express* é sua natureza minimalista. Ao contrário da maioria dos *frameworks* em que são acoplados vários módulos que muitas vezes não são utilizados, o *Express* utiliza a ideia de —menos é mais||, oferecendo um ambiente enxuto, porém com a capacidade de adicionar módulos para necessidades específicas (BROWN, 2014).

O *framework* possui implementações para a manipulação de rotas, *cache*, *cookies* e outros recursos que podem ser adicionados de acordo com a necessidade (HANH, 2016). Este *framework* vem sendo muito utilizado devido à facilidade de uso e a possibilidade de criação de servidores *web* de forma rápida. Sendo minimalista,

muitos desenvolvedores criaram pacotes compatíveis para resolver dificuldades encontrado no desenvolvimento de alguma aplicação *web* caso não seja possível apenas com o *Express*, como o auxílio ao utilizar parâmetros por URL¹⁸, dados por POST¹⁹ e sessões (EXPRESS, 2022).

2.8 React

O *ReactJS* é uma biblioteca *Front-end* baseada na linguagem *JavaScript*, seu principal objetivo é permitir o desenvolvimento interfaces baseadas em componentes para aplicações *web*. O *ReactJS*, é uma biblioteca *JavaScript* voltada para a construção de interfaces *web*. Esta biblioteca pode ser utilizada tanto para o desenvolvimento de aplicações *web* quanto para aplicações *mobile*. O *React* é capaz de manipular elementos visuais de forma que se possa criar componentes reutilizáveis. (LIMA, 2017).

É uma tecnologia que se esforça para fornecer velocidade, simplicidade e escalabilidade. Algumas de suas características mais notáveis são JSX, componentes com estado, modelo de objeto de documento virtual. (KHUAT, 2018)

Atualmente, em sua documentação *React* se define como: —O *React* é uma biblioteca *JavaScript* declarativa, eficiente e flexível para criar interfaces com o usuário. Ele permite compor *UIs* complexas a partir de pequenos e isolados códigos chamados `__componentes__`. (REACTJS, 2022). Mais de 9 mil empresas, incluindo Facebook, Uber, e Netflix, utilizam o *framework* segundo o site *Stackshare* (2022).

CAPÍTULO 3 ANÁLISE E MODELAGEM DO SOFTWARE

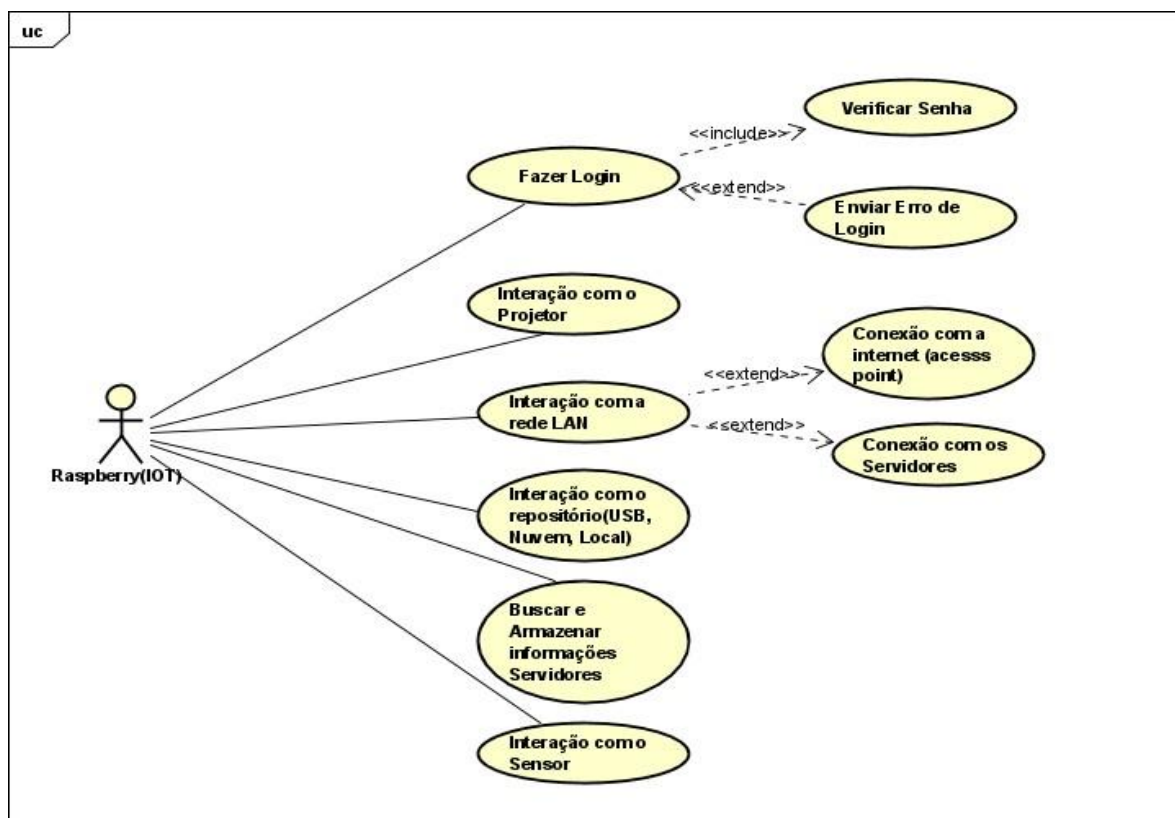
3.1 DESCRIÇÃO DO SISTEMA

O sistema permitirá o monitoramento da glicose em tempo real, através de um monitor de glicose tipo Holter disponível comercialmente. Um Bracelete é conectado à glicose monitor, para que leituras em tempo real serem obtidas, emprega-se microcomputadores Raspberry Pi, que são responsáveis pela comunicação com os sensores e atuadores físicos do ambiente, e por controlar notificações de forma distribuída entre os vários controladores presentes em uma mesma região. O paciente realiza seu credenciamento no sistema, onde o mesmo pode realizar a medição da glicose, consultar medições anteriores, enviar medição e pode gerar um relatório.

3.2 DIAGRAMAS DE CASOS DE USO

Para melhorar a análise e o entendimento dos diagramas, os casos de uso serão divididos por cenário.

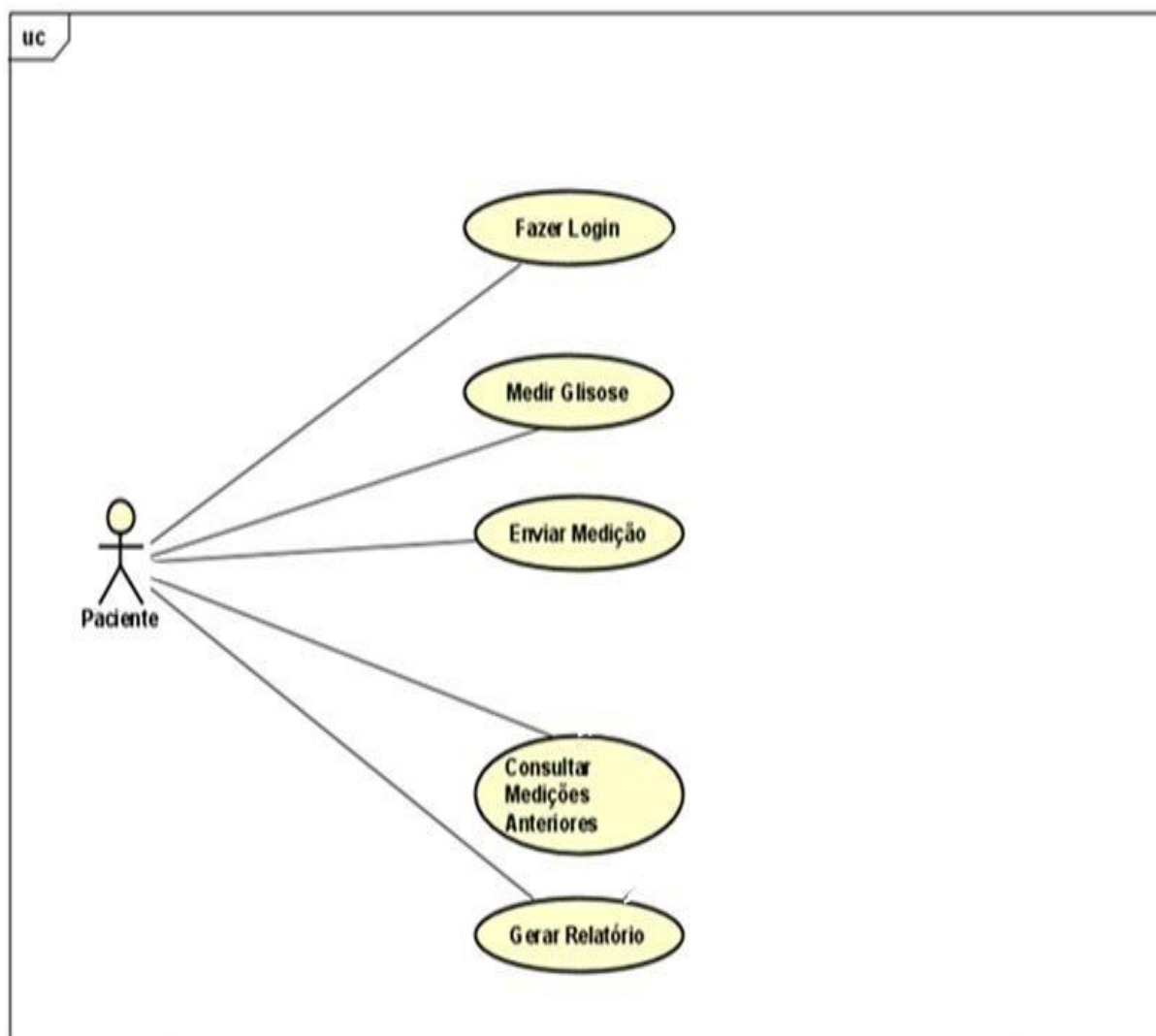
FIGURA 9. Diagrama Caso de Uso – IoT



Fonte: Autor.

FIGURA 10.

Diagrama Caso de Uso – Sistema.



Fonte: Autor.

3.2.1 DESCRIÇÃO DE CASOS DE USO

Esta seção possui a descrição dos casos de uso citados no diagrama anterior. Os casos de uso são identificados por um código, que é formado pelas letras UC (Use Case) seguido por uma identificação numérica, sendo também compostas pelas seguintes partes: Código, Caso de Uso, Descrição, Ator, Pré-Condições, Pós-Condições, Cenário Principal, Cenário Alternativo, Esboço de tela.

Tabela 3 – Caso de uso detalhado: Fazer Login.

ID: UC 001	
Caso de Uso:	Fazer Login.
Descrição :	Validar os dados de usuário e senha para que o paciente tenha acesso ao sistema.
Ator:	Paciente.
Pré-Condições:	O paciente deve possuir cadastro no sistema.
Pós-Condições:	O paciente logado no sistema.
Cenário Principal:	1. O ator deve entrar no website do sistema. 2. O ator deve escrever o seu usuário e senha nos devidos campos. 3. O sistema realizará a validação dos dados do usuário. 4. O ator será direcionado para a pagina de menu do sistema.
Cenário Alternativo:	2a. Paciente Cancela fazer login. 1. Sistema retorna a tela 1 do sistema. 3a. Campos não informados para realizar login. 1. Sistema emite mensagem "Informe um dos campos para fazer login". 2. Paciente retoma ao passo 3. 3b. Campos inválidos. 1. Sistema emite uma mensagem "Dados de usuário invalido" (sinalizando o campo digitado). 2. Paciente volta ao passo 3.

Esboço de Tela:

The image is a wireframe of a web browser window. The title bar at the top reads "Monitoramento do Controle de Glicose". The address bar shows the URL "https://mcg.com". The main content area has a white background and contains the following elements: a large heading "SEJA BEM VINDO!", a subtext "Para continuar, digite os dados abaixo", a user input field with a person icon and the label "Usuário", a password input field with a padlock icon and the label "Senha", a blue "ENTRAR" button, and a link "Esqueci minha senha". The browser window has standard navigation icons (back, forward, stop, home) and a search icon in the top left, and a close icon in the top right.

Monitoramento do Controle de Glicose

https://mcg.com

SEJA BEM VINDO!

Para continuar, digite os dados abaixo

 Usuário

 Senha

ENTRAR

[Esqueci minha senha](#)

Fonte: Autor.

Tabela 4 – Caso de uso detalhado: Medir Glicose.

ID: UC 002	
Caso de Uso:	Medir Glicose
Descrição :	Medir a glicose do paciente.
Ator:	Paciente.
Pré-Condições:	Usuário Autenticado no sistema.
Pós-Condições:	Uma instância de paciente ser incluída no banco de dados.
Cenário Principal:	1. Paciente seleciona a opção Medir Glicose. 2. Sistema mostra Tela 03. 3. Paciente informa os valores nos [Campos de preenchimento] e aperta o botão Medir. 4. Sistema valida os valores informados. 5. Sistema armazena os valores no banco de dados e mostra mensagem: "Medição Salvo com Sucesso."
Cenário Alternativo:	2a. Paciente Cancela a medição. 1. Sistema retorna a tela principal do sistema. 3a. Campos não informados na medição. 1. Sistema emite mensagem "Informe um dos campos para medir glicose". 2. Paciente retoma ao passo 3. 3b. Campos inválidos. 1. Sistema emite uma mensagem "Dados de medição inválido" (sinalizando o campo digitado). 2. Paciente volta ao passo 3.

Esboço de Tela:

The image is a wireframe of a web browser window. The title bar at the top reads "Monitoramento do Controle de Glicose". The address bar contains the URL "https://mcg.com". The main content area has a heading "Medir Glicose" followed by the instruction "Para continuar, digite os dados abaixo". Below this, there is a date selection field with a calendar icon and the label "Data". Next to it is a dropdown menu with the text "Opção 1" and a downward arrow. At the bottom of the form are two buttons: a blue "Medir" button and a white "Cancelar" button. A circular arrow icon is located in the bottom left corner of the main content area. The browser window has a standard toolbar with back, forward, and home icons.

Fonte: Autor.

Tabela 5 – Caso de uso detalhado: Enviar Medição.

ID: UC 003	
Caso de Uso:	Enviar medição.
Descrição:	Enviar medição de glicose do paciente.
Ator:	Paciente.
Pré-Condições:	Usuário Autenticado no sistema e existir medição de glicose realizada.
Pós-Condições:	Uma instância de paciente ser enviada a um email.
Cenário Principal:	1. Paciente seleciona a opção Enviar Medição. 2. Sistema mostra Tela 04. 3. Paciente informa os valores nos [Campos de preenchimento] e aperta o botão Enviar. 4. Sistema valida os valores informados. 5. Sistema envia o email com as medições e mostra mensagem: "Medição Enviada com Sucesso."
Cenário Alternativo:	2a. Paciente Cancela o envio. 1. Sistema retorna a tela principal do sistema. 3a. Campos não informados para envio. 1. Sistema emite mensagem "Informe um dos campos para enviar medição de glicose". 2. Paciente retoma ao passo 3. 3b. Campos inválidos. 1. Sistema emite uma mensagem "Dados de envio inválido" (sinalizando o campo digitado). 2. Paciente volta ao passo 3.

Esboço de Tela:

The wireframe depicts a web browser window with a title bar labeled "Monitoramento do Controle de Glicose". The address bar contains the URL "https://mcg.com". The main content area features a heading "Enviar Medição" followed by the instruction "Para continuar, digite os dados abaixo". Below this, there is a date selection field represented by a calendar icon and an empty text input. This is followed by a dropdown menu currently showing "Opção 1". At the bottom of the form are two buttons: a blue "Enviar" button and a white "Cancelar" button. A circular back button is located in the bottom-left corner of the page area. The browser's status bar at the bottom is empty.

Fonte: Autor.

Tabela 6 – Caso de uso detalhado: Consultar Medições Anteriores.

ID: UC 004	
Caso de Uso:	Consultar Medições Anteriores.
Descrição:	Consultar medições de glicose anteriores do paciente.
Ator:	Paciente.
Pré-Condições:	Usuário Autenticado no sistema e existir medição de glicose realizada.
Pós-Condições:	Não se aplica.
Cenário Principal:	1. Paciente seleciona a opção Consultar Medições Anteriores. 2. Sistema mostra Tela 06. 3. Paciente informa os valores nos [Campos de preenchimento] e aperta o botão Consultar. 4. Sistema valida os valores informados. 5. Sistema exibe os valores das medições realizadas existentes no banco de dados.
Cenário Alternativo:	2a. Paciente Cancela a consulta. 1. Sistema retorna a tela principal do sistema. 3a. Campos não informados para o agendamento. 1. Sistema emite mensagem “Informe um dos campos para consultar medição de glicose”. 2. Paciente retoma ao passo 3. 3b. Campos inválidos. 1. Sistema emite uma mensagem “Dados de consulta inválido” (sinalizando o campo digitado). 2. Paciente volta ao passo 3.

Esboço de Tela:

The wireframe depicts a web browser window titled "Monitoramento do Controle de Diabetes". The address bar shows "https://mcd.com". The main content area features the heading "Consultar Medições" followed by the instruction "Para continuar, digite os dados abaixo". Below this, there is a date selection field with a calendar icon and the label "Data", followed by a dropdown menu labeled "Opção 1". At the bottom of the form are two buttons: "Consultar" (highlighted in blue) and "Cancelar". A circular arrow icon is positioned on the left side of the form area. The browser's status bar at the bottom right contains two slanted lines.

Fonte: Autor.

Tabela 7 – Caso de uso detalhado: Gerar Relatório.

ID: UC 005	
Caso de Uso:	Gerar relatório.
Descrição:	Gerar relatório das medições de glicose do paciente.
Ator:	Paciente.
Pré-Condições:	Usuário Autenticado no sistema e existir medição de glicose realizada.
Pós-Condições:	O sistema irá gerar relatórios em Dashboard.
Cenário Principal:	1. Paciente seleciona a opção Gerar relatório. 2. Sistema mostra Tela 07. 3. Paciente informa os valores nos [Campos de preenchimento] e aperta o botão Gerar. 4. Sistema valida os valores informados. 5. O sistema gerar um relatório em Dashboard e mostra mensagem: "Relatório gerado com Sucesso."
Cenário Alternativo:	2a. Paciente Cancela o gerar relatório. 1. Sistema retorna a tela principal do sistema. 3a. Campos não informados para gerar relatório. 1. Sistema emite mensagem "Informe um dos campos para gerar relatório da medição de glicose". 2. Paciente retoma ao passo 3. 3b. Campos inválidos. 1. Sistema emite uma mensagem "Dados para gerar relatório inválido" (sinalizando o campo digitado). 2. Paciente volta ao passo 3.

Esboço de Tela:

The wireframe depicts a web browser window with a title bar reading "Monitoramento do Controle de Glicose". The address bar contains "https://mcg.com" and a search icon. The main content area features a heading "Gerar Relatório" followed by the instruction "Para continuar, digite os dados abaixo". Below this, there is a date selection field with a calendar icon and the label "Data", followed by a dropdown menu labeled "Opção 1". At the bottom of the form are two buttons: "Gerar" (highlighted in blue) and "Cancelar". A circular arrow icon is positioned on the left side of the form area. The browser window includes standard navigation icons (back, forward, stop, home) and a status bar at the bottom right.

Fonte: Autor.

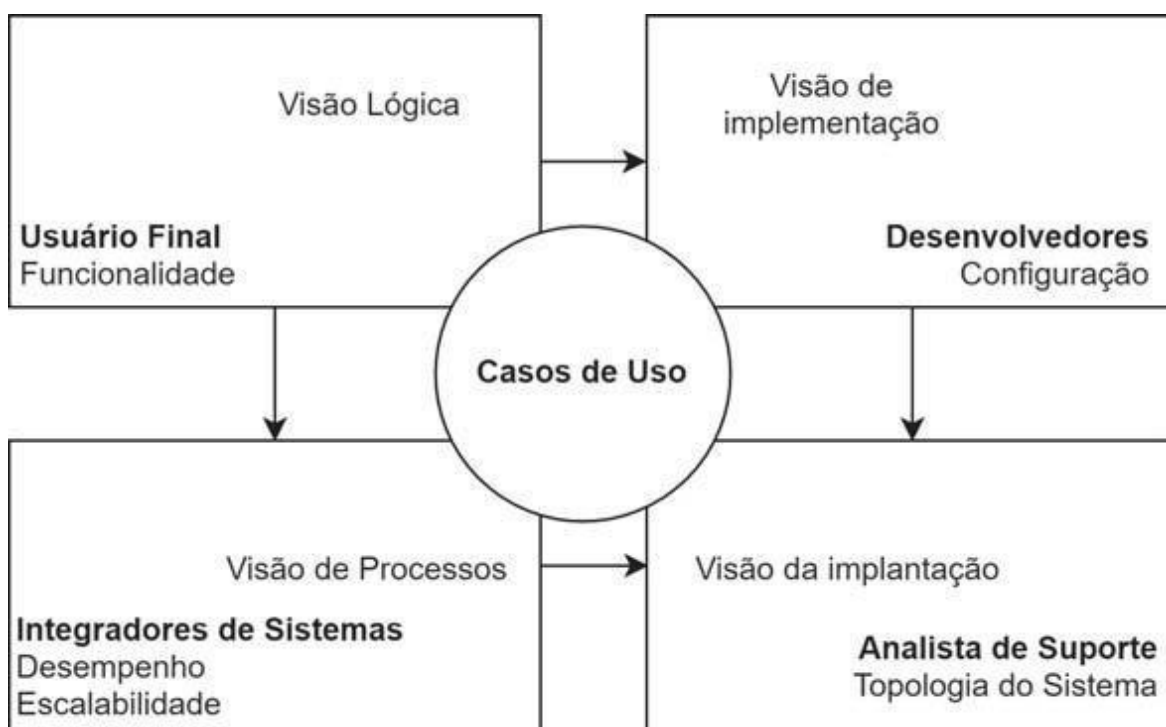
3.3 Implementação Do Sistema

Esse item descreve o fluxo das atividades para definições da arquitetura de sistema, a definição da arquitetura tem a abrangência dos protocolos de comunicação, armazenamento dos dados e aplicação de consulta web, serão mostrados os diagrama de classes, diagrama de atividades, diagrama de componentes, e diagrama de implantação para que se entenda a concepção idealizada durante este projeto.

3.3.1 Arquitetura Do Sistema

A arquitetura de sistema pode ser apresentada de diferentes formas, segundo Kruchten (1995), o modelo de visão 4 + 1 é bastante genérico. Portanto, outras notações, ferramentas e métodos de design podem ser usados, especialmente para as decomposições lógicas e de processo. A Figura 11 é uma visão geral do modelo 4+1, na qual são apresentadas a visão lógica, de processos, de implementação e de implantação.

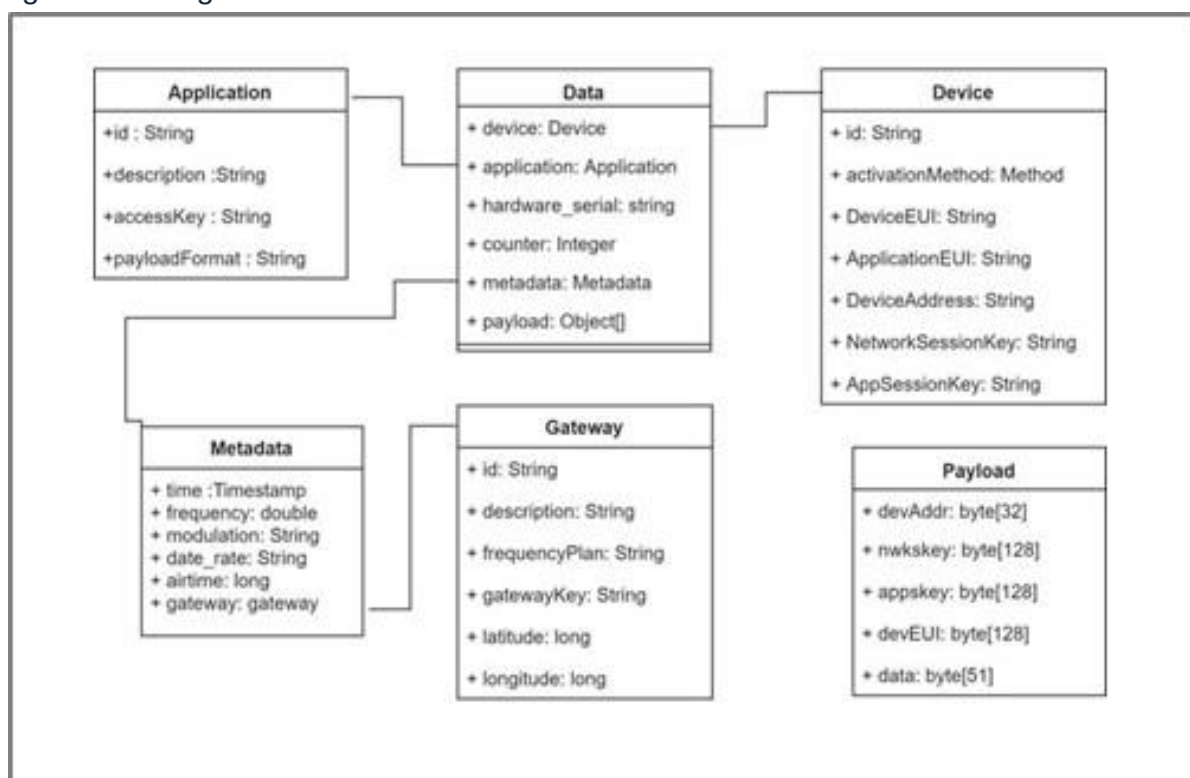
Figura 11. Visão 4+1 descrevendo a arquitetura de sistemas com base no uso de múltiplas visualizações simultâneas.



Fonte: Adaptado Kruchten (1995).

A visão lógica da arquitetura define as abstrações das partes de um sistema. Além de ser usado para modelar do que o sistema é composto e como as partes interagem umas com as outras. Os tipos de diagramas UML que geralmente retratam essa visão são: classes, objetos, máquina de estado e diagramas de interação (MILES e HAMILTON, 2006). Para representar essa visão utilizou-se o diagrama de classes, Figura 12, que demonstra a interação dos objetos desde o dispositivo até a aplicação.

Figura 12. Diagrama de Classes



A classe *PacketLoraWan* representa os dados gerados nos dispositivos, contendo os identificadores no padrão *LoRaWan* (*devAddr*, *nwkskey*, *appskey* e *devEUI*) e um vetor de *bytes*, *data*, que são as informações coletadas pelo sensores no padrão *CayenneLPP*.

A *Application* é a classe que representa a aplicação na plataforma *The Things Network*, contendo o *id* como identificador único, *description* descreve a aplicação, *payloadFormat* formato do pacote enviado pelo dispositivo, podendo ser *custom* ou

no padrão *Cayenne LPP*. O atributo *accessKey* é a chave de acesso gerado pela plataforma para acesso aos dados da aplicação através dos métodos de integração.

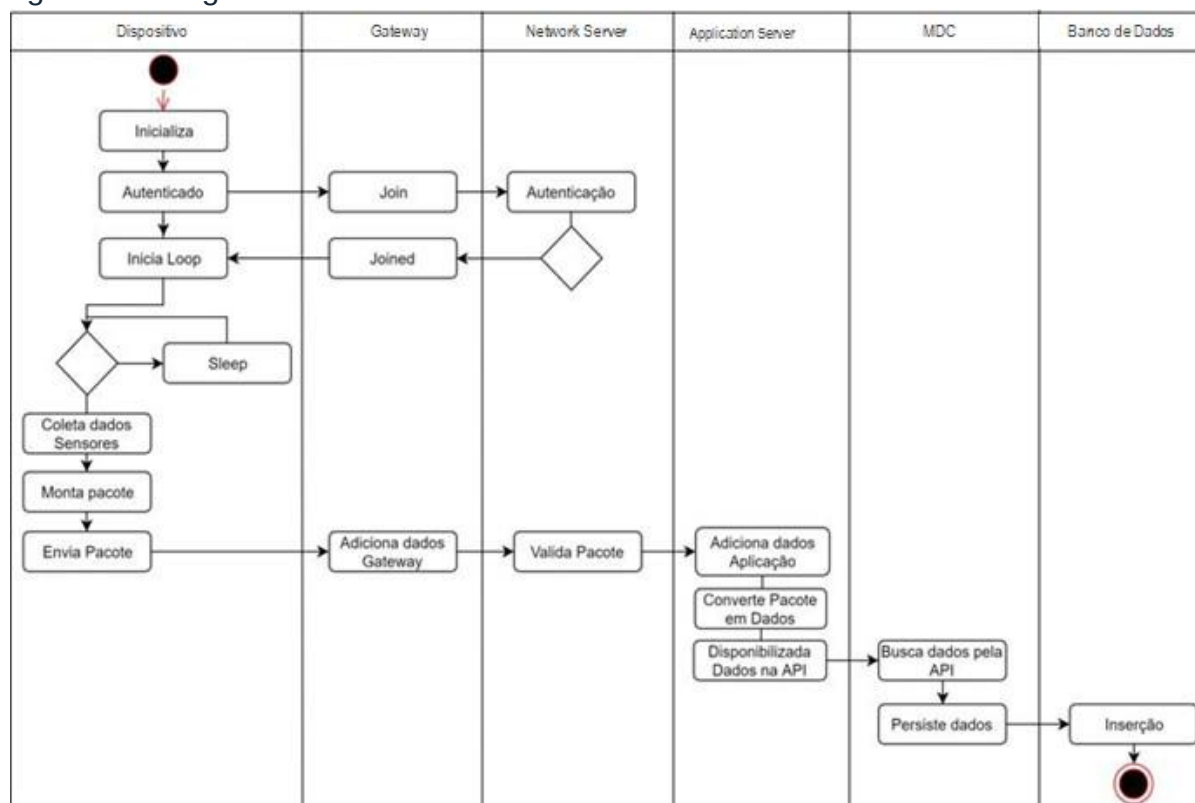
Já a classe *Device* representa o dispositivo cadastrado na plataforma *The Things Network*, possui os atributos *id* como identificador único, *activationMethod* podendo ser ABP ou OTAA, e os demais atributos para autenticação do dispositivo: *DeviceEUI*, *ApplicationEUI*, *DeviceAddress*, *NetWirkSessionKey*, *AppSessionKey*.

O *Gateway* é definido pelos atributos *id* como identificador único, *description* que descreve o *gateway*, *frequencyPlan* que no caso é 915 Mhz, *gatewayKey* define a chave de segurança que deve ser configurado no equipamento físico, e, por fim, latitude e longitude que armazenam a posição geográfica do *gateway*.

As informações, ao trafegarem pela rede *LoRaWan* até a aplicação, são convertidas na estrutura representada pelas classes *Data* que possuem o relacionamento com o *device* e *application*. O atributo *counter* é o número de pacotes enviados pelo dispositivo, *payload* representado por um vetor de objetos, esses que já são dados convertidos do vetor de bytes da classe *Payload*. Por fim, a classe *MetaData* possui dados referentes à transmissão como *time* com a data e hora do *gateway* no momento que recebeu o pacote, *frequency* com a frequência que foi transmitido o pacote, *data_rate* define o o *Spreading Factor* e *bandwidth* utilizados na transmissão e, finalmente, o *airtime* que é o tempo total do envio do pacote.

A Visão do Processo descreve as atividades dentro do sistema e é particularmente útil para visualizar o que deve acontecer no *software*, normalmente é representado por diagramas de atividades (MILES e HAMILTON, 2006). O diagrama de atividade da coleta dos dados pelos dispositivos pode ser observado na Figura 13. A atividade começa com a inicialização do dispositivo, logo após é realizada a autenticação na rede *LoRaWan* enviando um pacote do tipo *join* com os parâmetros necessários.

Figura 13. Diagrama de Atividades coleta de dados.



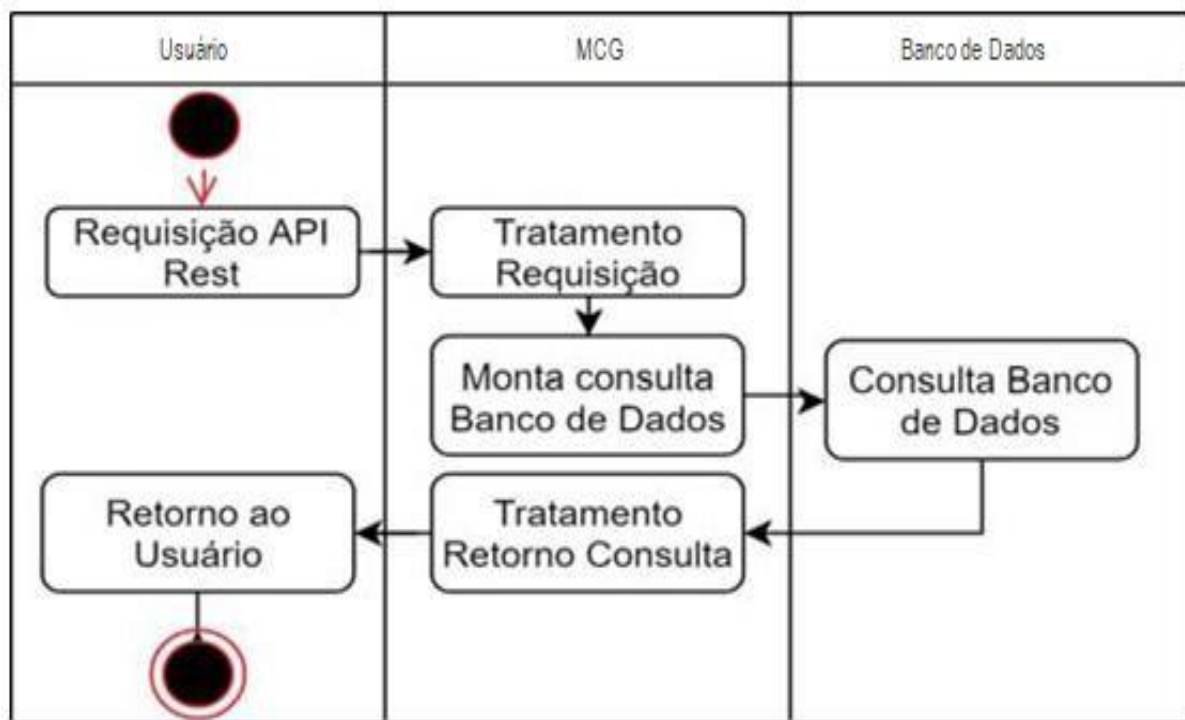
Esse pacote é encaminhado pelo *gateway* para o *Network Server* que valida se esta aplicação e dispositivos são válidos, caso verdadeiro, retorna o pacote *joined* com a confirmação da autenticação. Em seguida, o dispositivo entra no loop principal no qual se verifica se está no ciclo para envio de um novo pacote, caso negativo, entra em *sleep* até atingir esse tempo determinado. Ao ser atingido é realizada a leitura dos dados dos sensores, criado um pacote no padrão determinado e enviado esse pacote pela rede *LoRaWan*. Ao ser recebido pelo gateway este adiciona alguns dados referentes à performance da rede, como *Data Rate*, *SNR* e *RSSI*.

Ao ser recebido pelo *Network Server*, são validadas as informações de autenticação do pacote e ele é encaminhado para o *application server*, onde são adicionados os dados da aplicação e convertidos os dados brutos dos sensores para objetos JSON. Por fim, o pacote é capturado por API pelo MCG(Monitoramento do Controle de Glicose) que persiste no banco de dados.

Já no diagrama de atividades de consulta dos dados, Figura 14, inicia-se com a requisição por *API Rest* do usuário recebida pelo MCG(Monitoramento do Controle de Glicose) que realiza o tratamento dessa requisição, montando a consulta no padrão do banco de dados e, em seguida, envia a requisição de consulta ao banco

de dados. Ao receber o retorno, o MCG(Monitoramento do Controle de Glicose) ajusta o documento *JSON* para retorno via *HTTP* da solicitação *REST*. Para a consulta não é necessária nenhuma interação com o *Network Server* ou *gateway* da estrutura *LoRaWan*, os dados são consultados diretamente no banco de dados.

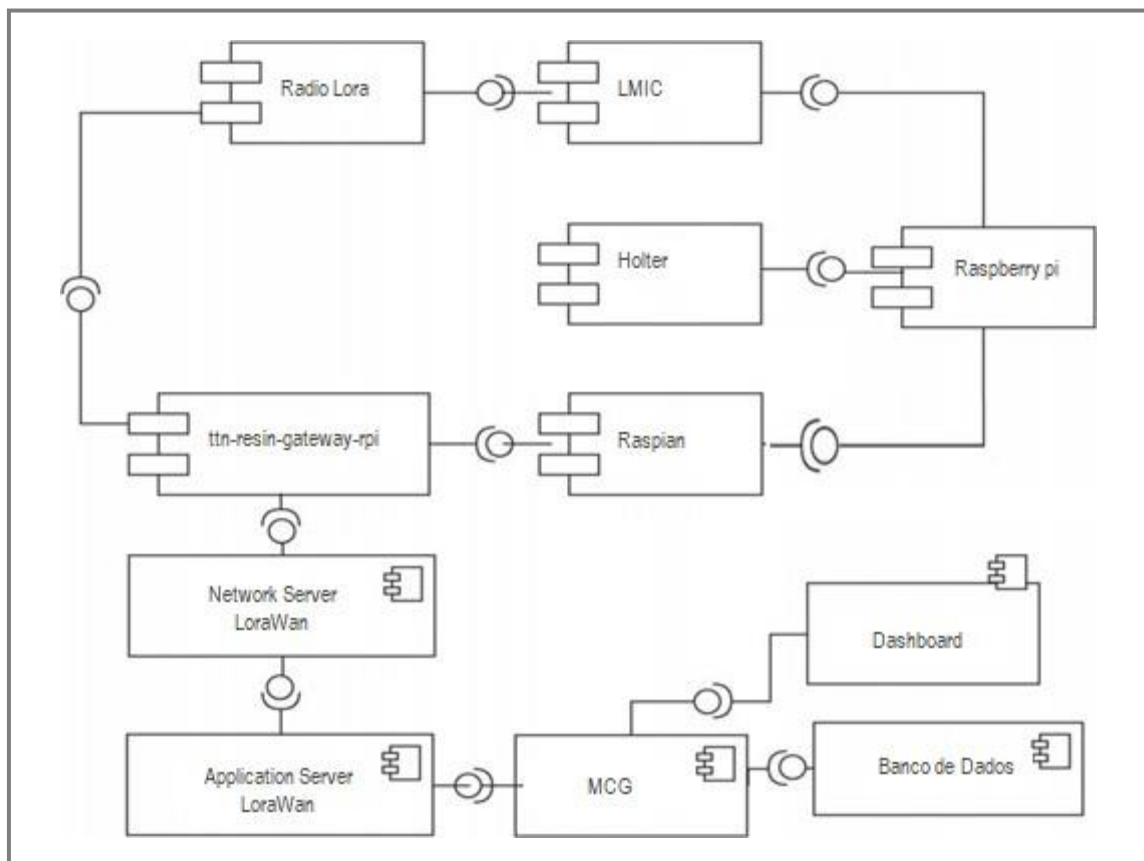
Figura 14. Diagrama de Atividades Consulta.



A visão do desenvolvimento descreve como as partes do sistema são organizadas em módulos e componentes, sendo muito útil para gerenciar as camadas dentro da arquitetura. Essa visão normalmente contém diagramas de pacote e componente. A Figura 15 apresenta o diagrama de componentes da arquitetura.

A dependência dos componentes começa pelo *Raspberry pi*, que provê a base para interligação do *software* e *hardware* do dispositivo. As bibliotecas do sensor *Holter* são acessadas pelo *Raspberry pi*. A biblioteca *LMIC* gerencia a operação do hardware do rádio *Lora* e o envio dos pacotes no padrão *LoRaWan*.

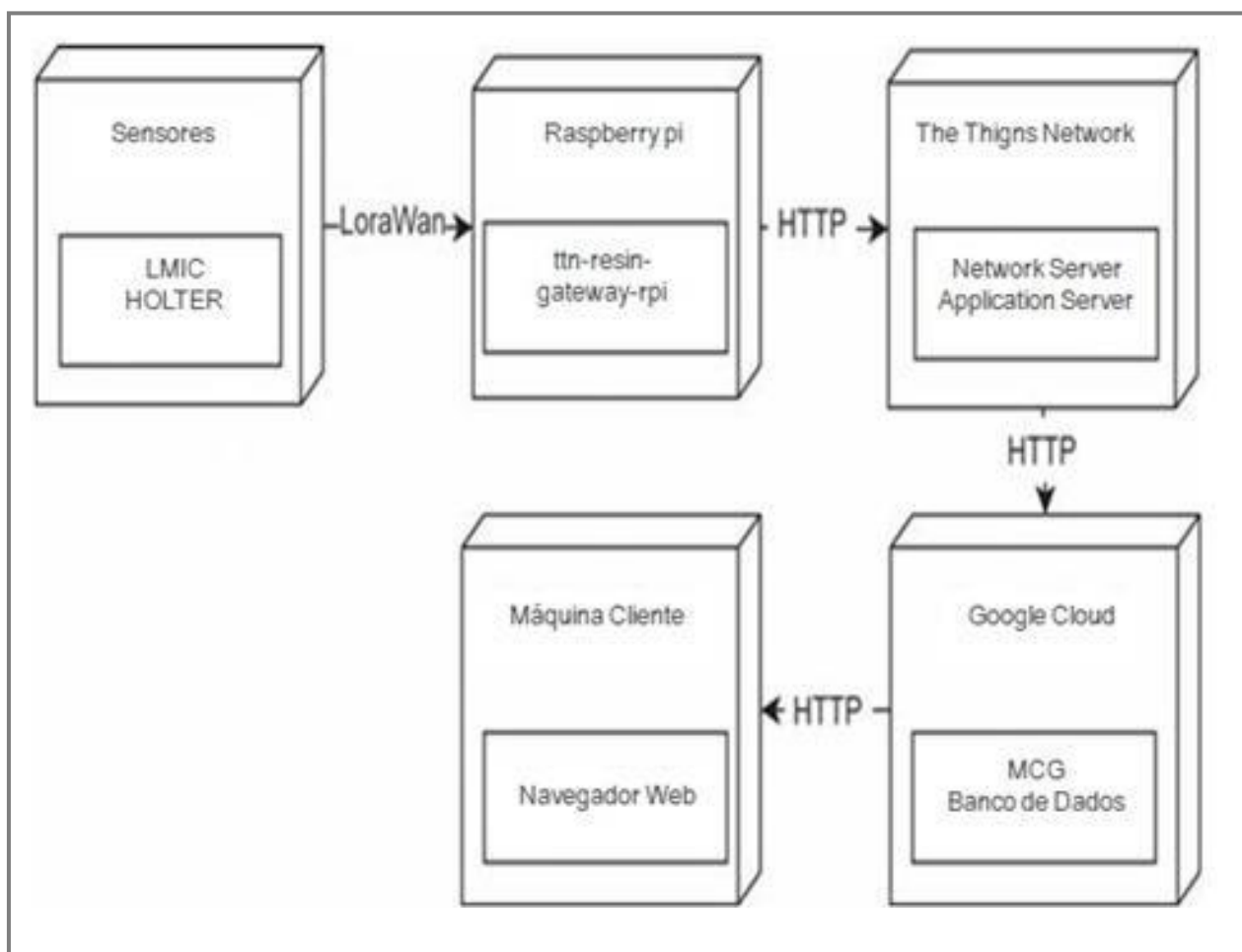
Figura 15. Diagrama de Componentes



O componente *ttn-resin-gateway-rpi* é responsável pela operação do hardware do gateway e também pelo encaminhamento dos pacotes *LoRaWan* para o Network Server. O *ttn-resin-gateway-rpi* utiliza o sistema operacional *raspian*, que é executado sobre uma placa *Raspberry pi*. Já o componente Network Server é acessado pelo Application Server dentro da plataforma *The Things Network*. O MCG(Monitoramento do Controle de Glicose) acessa o *Application Server* através de *API* de integração capturando as informações, salvando no banco de dados e disponibilizando os dados no *dashboard* aos usuários.

Uma visão física da arquitetura descreve como o design do sistema, conforme descrito nas três visões anteriores, é transformado em entidades do mundo real (MILES e HAMILTON, 2006). Os diagramas nesta exibição mostram como as partes abstratas são mapeadas na implementação final. Os diagramas de implantação podem ser observados na Figura 16, na qual estão definidos os hardwares necessários para a execução da arquitetura.

Figura 16. Diagrama de Implantação



Para execução da arquitetura são necessários alguns hardwares, iniciando pelos dispositivos que são os sensores *Holter* e o microcomputador *Raspberry pi*, pois têm compatibilidade com a biblioteca LMIC. Para o *gateway* é necessário uma placa *Raspberry pi*. O *Network Server* e *Application Server* executam sobre a plataforma *The Things Networks*, assim o hardware necessário é abstraído para a plataforma. Na execução do MCG e banco de dados é utilizado uma máquina virtual na plataforma do *Google Cloud*. Para consulta do *Dashboard* pelo cliente é necessário um computador ou *smartphone* com capacidade de executar um navegador *web*.

CONSIDERAÇÕES FINAIS

Os objetivos deste trabalho foram atingidos e até mesmo superados. Foi iniciada a construção de um sistema útil e agradável às pessoas que irão utilizá-lo, segundo requisitos levantados inicialmente.

Foi desenvolvido e disponibilizado um sistema para informatizar a atividade de medição da glicose em pacientes com Diabetes, que contribuiu para a manutenção da saúde pública através da informatização de uma atividade importante para o controle da doença.

Inicialmente, foram apresentadas a fundamentação teórico-metodológica para o desenvolvimento, os sistemas similares e as tecnologias utilizadas. Foram descritos os principais requisitos do sistema, a sua modelagem e o desenvolvimento da aplicação. E por fim foram apresentados os resultados do sistema.

Ao implementar o sistema, a experiência adquirida foi enorme, como o aprendizado dia a dia da linguagem *JavaScript*, parte visual de implementação das telas e conceitos de banco de dados, principalmente nos casos de uso mais complexos.

Foram utilizadas novas tecnologias de desenvolvimento como *Node* e *Express*, *frameworks JavaScript* que otimizou o desenvolvimento do *backend* da aplicação e o *React*, que é um framework para o desenvolvimento no *frontend*.

No desenvolvimento do *frontend* da aplicação, a maior dificuldade foi de manter os componentes visuais independentes entre si para facilitar a reutilização de código. No lado do *backend* da aplicação, a maior dificuldade foi realizar o *upload* dos níveis para a simulação do sistema e salva-los na base de dados.

Resultados Alcançados

Dentre os resultados obtidos podem ser destacados:

- Foram identificados as tecnologias e os componentes necessários para o desenvolvimento do software, que foram citados na seção Fundamentação Teórica.
- Foi desenvolvido o software que permite a informatização do monitoramento da atividade de medição da glicose;

- O código fonte do software foi disponibilizado no repositório, GITHUB, para que outras pessoas possam estudar e também contribuir para evolução do produto.

Trabalhos Futuros

Dentre os possíveis trabalhos futuros para este sistema, podemos citar:

- Implementação da funcionalidade de agendar a medição para obter os dados coletados desse paciente.
- Implementação de um controle de usuário para possibilitar um controle de acesso das funcionalidades disponíveis no sistema;
- Optimização da funcionalidade Gerar relatório, para que o relatório possa ser acessado por outros usuários.

REFERÊNCIAS

Abramoff, M.; Niemeijer, M.; Suttorp-Schulten, M.; Viergerver, M.; Russell, S.; Van Ginneken, B. Evaluation of a system for automatic detection of diabetic retinopathy from color fundus photographs in a large population of patients with diabetes. In: Diabetes care, 2008. (document), volume 31(2), p. 193-198.

ACAMPORA, G.; COOK, D. J.; RASHIDI, P.; VASILAKOS, A. V. A Survey on Ambient Intelligence in Health Care. Proceedings of the IEEE. Institute of Electrical and Electronics Engineers, v. 101(12), p. 2470–2494, 2013. DOI 10.1109/JPROC.2013.2262913.

ADELANTADO, F. et al. Understanding the limits of lorawan. arXiv preprint arXiv:1607.08011, 2016.

ALLIANCE, L. LoRaWAN™ Specification. 2018. **Alliance** Disponível em: <<https://lora-alliance.org/resource-hub/lorawantm-specification-v10>>. Acesso em: 06 jul. 2021.

ALLIANCE, L. LoRaWAN™ Regional Parameters. 2018. **Alliance** Disponível em: <<https://lora-alliance.org/resource-hub/lorawantm-regional-parameters-v10>>. Acesso em: 16 jul. 2021.

ANGELIDOU, M. Smart city policies: A spatial approach. Cities, Elsevier, v. 41, p. S3–S11, 2014.

ASHTON, K. That —Internet of Things|| Thing. RFID Journal, p. 4986, 2009.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. Computer networks, Elsevier, v. 54, n. 15, p. 2787–2805, 2010.

BROWN, Ethan. **Web Development with Node & Express: Leveraging the Javascript Stack**. 1.ed. O'Reilly Media Inc., 2014.

Como carregar dados JSON a partir do Cloud Storage | BigQuery. Disponível em: <<https://cloud.google.com/bigquery/docs/loading-data-cloud-storage-json?hl=pt-br>>. Acesso em: 01 nov. 2021.

COOK, D. J.; DAS, S. K. How Smart are our Environments? An Updated Look at the State of the Art. Journal of Pervasive and Mobile Computing, v. 3, p. 53-73, 2007.

CRESWELL, J. W. Projeto de pesquisa: Métodos qualitativo, quantitativo e misto. 3. ed. Porto Alegre: Artmed, 2010.

DIAS, R. R. de F. Internet das Coisas Sem Mistérios: Uma nova inteligência para os negócios. Sao Paulo: Netpress Books, 2016. 120 p.

EDWING, A.; FERRUZCA, M.; IVAN G. Detection of Episodes of Major Depression in Older Adults Through Physiological Markers and Movement Patterns Case Study. 2015 DOI 10.1109/ICHI.2015.48.

ENTERPRISE. 2021. **EnterpriseDB.** Disponível em: <<https://www.enterprisedb.com/nosql-code-samples>>. Acesso em: 01 nov. 2021.

EXPRESS. Express: Node.js web application framework. Disponível em: <<https://expressjs.com/>>. Acesso em: 20 mar. 2022.

FERGUSON, T., Online patient-helpers and physicians working together: A new partnership for high quality health care. British Medical Journal; International edition, Vol. 321, Nov. 2000.

FIELDING, R. T. REST : Architectural Styles and the Design of Network-based Software Architectures. **Doctoral dissertation, University of California**, 2000.

FORKAN, A.; KHALIL, I.; TARI, Z. CoCaMAAL: A Cloud-Oriented Context-Aware Middleware in Ambient Assisted Living. Future Generation Computer Systems, v. 35, p. 114–127, 2014. DOI 10.1016/j.future.2013.07.009.

GARCIA, P. S. R.; KLEINSCHMIDT, J. H. Tecnologias emergentes de conectividade na iot: Estudo de redes lpwan. XXXV SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES E PROCESSAMENTO DE SINAIS-SBrT2017, p. 1009–1013, 2017.

HAHN, Evan. **Express in Action: Writing, building, and testing Node.js applications.** 1.ed. Manning Publications, 2016.

HAIGH, K.; KIFF, L.; MYERS, J.; GURALNIK, V.; GEIB, C.; PHELPS, J.; WAGNER, T. The Independent Lifestyle Assistant™ (I.L.S.A.): AI Lessons Learned. P. 852-857, 2004.

HIRSCH, L. Wearable Tech, CES2015, and the Quantified Self of Healthcare. 2015. **Healthcare success** Disponível em

<<http://www.healthcaresuccess.com/blog/ces2015-quantified-self.html>> Acesso em: 21 jul. 2021.

HUMANAS, R. B. de Cidades Inteligentes e. BRASIL 2030 : INDICADORES BRASILEIROS DE CIDADES INTELIGENTES E HUMANAS. **Rede brasileira** 2017. Disponível em: <<http://redebrasileira.com/indicadores>>. Acesso em: 02 jul. 2021.

IBM DEVELOPER. 2015. **IBM** Disponível em: <<http://www.ibm.com/developerworks/br/library/ws-restful/index.html>>. Acesso em: 01 nov. 2021.

ISTAG. Scenarios for Ambient Intelligence in 2010. European Commission Report, 2001.

ISTAG. Strategic Orientations & Priorities for IST in FP6. European Commission Report, 2002.

ISTEPANIAN, RSH, PHILIP, NY, SUNGOOR, A, HU, S (2011) The potential of Internet of m-health Things m-IoT for noninvasive glucose level sensing. In , Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, pp.5264-5266.

JSON. 2021. **JSON**. Disponível em: <<https://www.json.org/json-pt.html>>. Acesso em: 01 nov. 2021.

KATSIRI, E. —Middleware Support for Context-Awareness in Distributed Sensor Driven Systems. II Ph.D. dissertation, University of Cambridge, 2005.

KATSIRI, E.; BACON, J.; MYCROFT, A. Linking Temporal First-Order Logic and Hidden Markov Models with Abstract Events. International Journal on Artificial Intelligence Tools, v. 19, p. 857-893, 2010.

KATSIRI, E.; MYCROFT, A.; BACON, J. Language and Model-based Abstractions for Estimating Energy Expenditure in Pervasive Healthcare. Book Chapter - Reasoning in Event-based distributed systems, Springer series on Studies in Comp. Int., 2010.

KROL, Mithun Satheesh; D'MELLO, Bruno Joseph; KROL, Jason. **Web Development with MongoDB and NodeJs**. 2a edição. Packt Publishing, 2015.

KRUCHTEN, P. B. The 4+ 1 view model of architecture. *IEEE software*, IEEE, v. 12, n. 6, p.42–50, 1995.

KHUAT, T. Developing a frontend application using ReactJS and Redux. Dissertação (Degree Programme in Business Information Technology Bachelor's)— Laurea University of Applied Sciences, Leppävaara, 2018. Disponível em: <https://www.theseus.fi/bitstream/handle/10024/150837/Tung_Khuat_1301747_T_hesis.pdf?sequence=1>. Acesso em: 01 out. 2022.

KWANG-HYUN, P.; ZEUNGNAM, B.; JU-JANG, L.; BYUNG, K.; JONG-TAE, L.; JINOH, K.; HEYOUNG, L.; DIMITAR, S.; DAEJIN, K.; JIN-WOO, J.; JUN-HYEONG, D.; KAP-HO, S.; CHONG, K.; WON-GYU, S.; WOO-JUN, L. Robotic Smart House to Assist People with Movement Disabilities. *Auton. Robots*. v. 22. p. 183-198, 2007. DOI 10.1007/s10514-006-9012-9.

LABS, Q. S. Quantified Self Guide to self-tracking tools. **Quantified self** 2014. Disponível em: <<http://quantifiedself.com/guide/>>. Acesso em: 21 jul. 2021.

LAURIDSEN, M. et al. Coverage comparison of gprs, nb-iot, lora, and sigfox in a 7800 km² area. In: IEEE. 2017 IEEE 85th Vehicular Technology Conference (VTC Spring). New York, NY, USA: Institute of Electrical and Electronics Engineers, 2017. p. 1–5. ISBN 978-1-5090-5932-4.

LIMA, M. **O guia completo do React e o seu ecossistema**. [S. l.]: - Imasters, 2017. <Disponível em: <https://imasters.com.br/desenvolvimento/o-guia-completo-do-react-e-o-seu-ecossistema>>. Acesso em: 03 out. 2022.

LOKE, S. Context-Aware Pervasive Systems: Architectures for a New Breed of Applications. CRC Press, 2006.

LUNDIN, A. C.; OZKIL, A. G.; SCHULDT-JENSEN, J. Smart cities: A case study in waste monitoring and management. *Hawaii International Conference on System Sciences 2017*, p. 10, 01 2017.

MDHAFFAR, A. et al. Iot-based health monitoring via lorawan. *IEEE EUROCON 2017 -17th International Conference on Smart Technologies*, IEEE, v. 1, n. 1, p. 519–524, July 2017.

MALMBERG, K. (1997), —Prospective randomised study of intensive insulin treatment on long term survival after myocardial infarction in patients with diabetes mellitusll, *British Medical Journal*, v. 314, p. 1512.

MASSE, M. REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces. Sebastopol, CA: O'Reilly Media, Inc., 2011.

MIKHAYLOV, K. et al. Lora wan for wind turbine monitoring: Prototype and practical deployment. In: *2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*. [S.l.: s.n.], 2018. p. 1–6. ISSN 2157-023X.

MILES, R.; HAMILTON, K. *Learning UML 2.0*. 1. ed. Sebastopol , CA, USA: "O'Reilly Media, Inc.", 2006. 286 p.

MOREIRA, Rafael Henrique. **O que é Nodejs?**. Disponível em <<http://www.nodebr.com/o-que-e-node-js/>>. Acesso em 25 out. 2012.

MORRISON, Michael. **Use a Cabeça! JavaScript**. 1a edição. O'REILLY, 2008.

OLIVEIRA, R. N. Uso do Paradigma Orientado a Notificações em Sistemas Sencientes. Relatório da disciplina de Tópicos Especiais Em Ec: Paradigma Orientado A Notificações. CPGEI-PPGCA/UTFPR, Curitiba -PR, Brasil, 2016.

OLIVEIRA, R. N.; ROTH, V.; HENZEN, A. F.; SIMÃO, J. M.; WILLE, E. C. G.; NOHAMA, P. Notification Oriented Paradigm Applied to Ambient Assisted Living Tool. *IEEE Latin America Transactions*, 2018.

PostgreSQL, **PostgreSQL: The world's most advanced open source database**. Disponível em: <<https://www.postgresql.org/>>. Acesso em: 24 out. 2022.

PEREIRA, Jonathas Bitencourt; SOUZA, Marta Alves de; COSTA, Helder Rodrigues da. **Segurança da informação em ambientes corporativos**. Disponível em <http://revistapensar.com.br/tecnologia/pasta_upload/artigos/a29.pdf>. Acesso em 14 out. 2022.

PETAJAJARVI, J. et al. Evaluation of lora lpwan technology for indoor remote health and wellbeing monitoring. *International Journal of Wireless Information Networks*, Springer, v. 24, n. 2, p. 153–165, 2017.

Plano de Ações Estratégicas para o Enfrentamento das Doenças Crônicas Não Transmissíveis (DCNT) no Brasil 2011-2022. Brasília: Ministério da Saúde; 2011.

POTDAR, V.; SHARIF, A.; CHANG, E. -Wireless Sensor Networks: A Survey. *International Conference on Advanced Information Networking and Applications Workshops*, v. 38, p. 636–641, 2009.

RAMOS, C. Ambient Intelligence – A State of the Art from Artificial Intelligence Perspective. *Progress in Artificial Intelligence*, v. 18, p. 285-295, 2007.

RASHIDI, P.; MIHAILIDIS, A. A Survey on Ambient-Assisted Living Tools for Older Adults. *IEEE Journal of Biomedical and Health Informatics*, v. 17, n. 3, p. 579-590, 2013.

REACTJS. **React** – A JavaScript library for building user interfaces. Disponível em: <<https://reactjs.org/>>. Acesso em: 19 set. 2022.

RESOLUÇÃO n 506, de 1º de julho de 2008. 2008. **Anatel.gov** Disponível em: <<http://www.anatel.gov.br/legislacao/resolucoes/23-2008/104-resolucao-06>>. Acesso em: 14 jul. 2021.

SEBESTYEN, G.; STOICA, I.; HANGAN, A. Human Activity Recognition and Monitoring for Elderly People. p. 341-347, 2016. DOI 10.1109/ICCP.2016.7737171.

SIGFOX. Sigfox - The Global Communications Service Provider for the Internet of Things (IoT). 2018. **Sigfox** Disponível em: <<https://www.sigfox.com>>. Acesso em: 06 jul. 2021.

SILVA, M.; GONCALVES, A.; DANTAS, M.; VANELLI, B.; MANERICHI, G.; SANTOS, S.; FERRANDIN, M.; PINTO, A. Implementation of IoT for Monitoring Ambient Air in Ubiquitous AAL Environments. p. 158-161, 2015. DOI 10.1109/SBESC.2015.37.

SIMÃO, J. M.; RENAUX, D. P. B.; LINHARES, R. R.; STADZISZ, P. C. Evaluation of the Notification Oriented Paradigm Applied to Sentient Computing. In: 10th Workshop on Software Technologies for Future Embedded and Ubiquitous Systems (SEUS 2014) in 2014 IEEE 17th International Symposium on Object/Component-Oriented Real-Time Distributed Computing, 2014, Reno - Nevada – USA, 2014. v. 1555-0. p. 253-260. DOI: 10.1109/ISORC.2014.54.

SOUZA, F. **Uma arquitetura LPWAN de custo acessível para cidades inteligentes**. 2019. p.12. Dissertação (Mestrado em Computação Aplicada) – Instituto De Ciências Exatas E Geociências, Universidade de Passo Fundo. Passo fundo, 2019.

SURYADEVARA, N.; MUKHOPADHYAY, S.C. Determining Wellness through an Ambient Assisted Living Environment. *Intelligent Systems, IEEE*, v. 29, p. 30-37, 2014. DOI 10.1109/MIS.2014.16.

STACKSHARE. **Why developers like React**. Disponível em: <<https://stackshare.io/react>>. Acesso em: 23 out. 2022.

SWAN, M. The quantified self: fundamental disruption in big data science and biological discovery. *Big Data*, Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, v. 1, n. 2, p. 85–99, 2013.

TEIXEIRA, José Ricardo. **Introdução ao PostgreSQL**. Publicado em: 25 de Jul. 2017. Disponível em <http://www.devmedia.com.br/autor/jose-ricardo-teixeira/225178>>. Acessado em: 19 out. 2022.

UNITED NATIONS. World Population Ageing 2017. Department of Economic and Social Affairs, Population Division, 2017, ST/ESA/SER.A/408.

Van den Berghe, G. (2004), —How does blood glucose control with insulin save lives in intensive care?‖, The Journal of Clinical Investigation, v. 114, n. 9, p. 1187-95.

WEISER, M. The Computer for the Twenty-First Century. Scientific American, pp. 94-104, 1991.

WHITMORE, A.; AGARWAL, A.; XU, L. D. The internet of things—a survey of topics and trends. Information Systems Frontiers, Springer, v. 17, n. 2, p. 261–274, 2015.

WND. WND - Brasil - WND Group. 2018. **WND Group** Disponível em: <<https://www.wndgroup.io/brasil/>>. Acesso em: 06 jul. 2021.

YAN, Z.; ZHANG, P.; VASILAKOS, A. V. A survey on trust management for internet of things. Journal of network and computer applications, Elsevier, v. 42, p. 120–134, 2014.

ANEXO 1

Telas do sistema

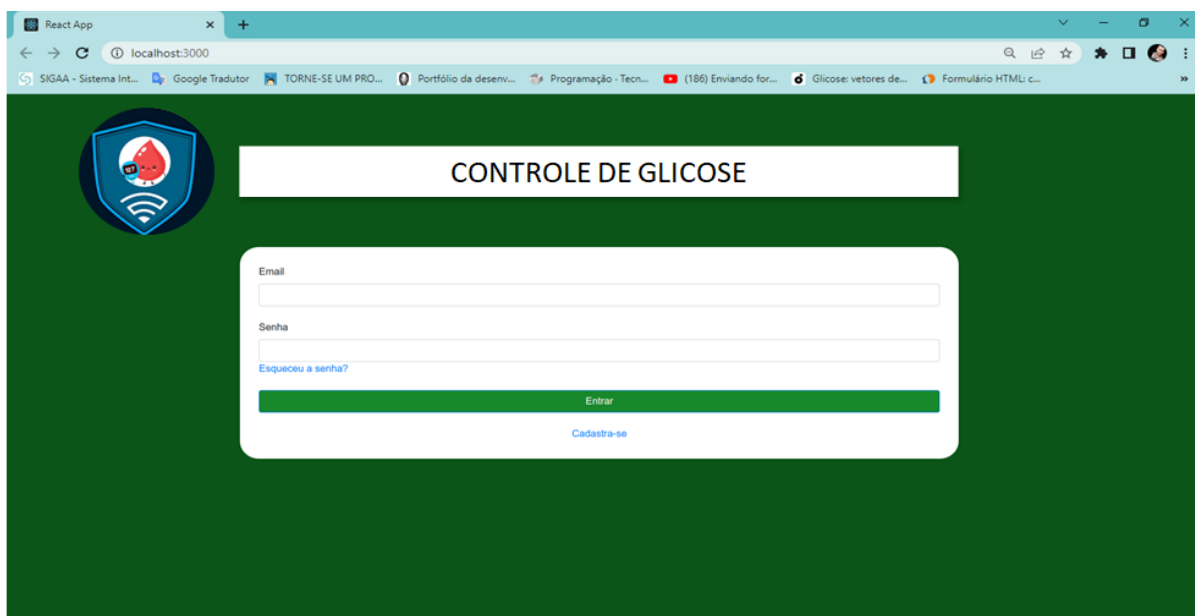
Figura 17 – Tela de *Login*.

Figura 18 – Tela Inicial.

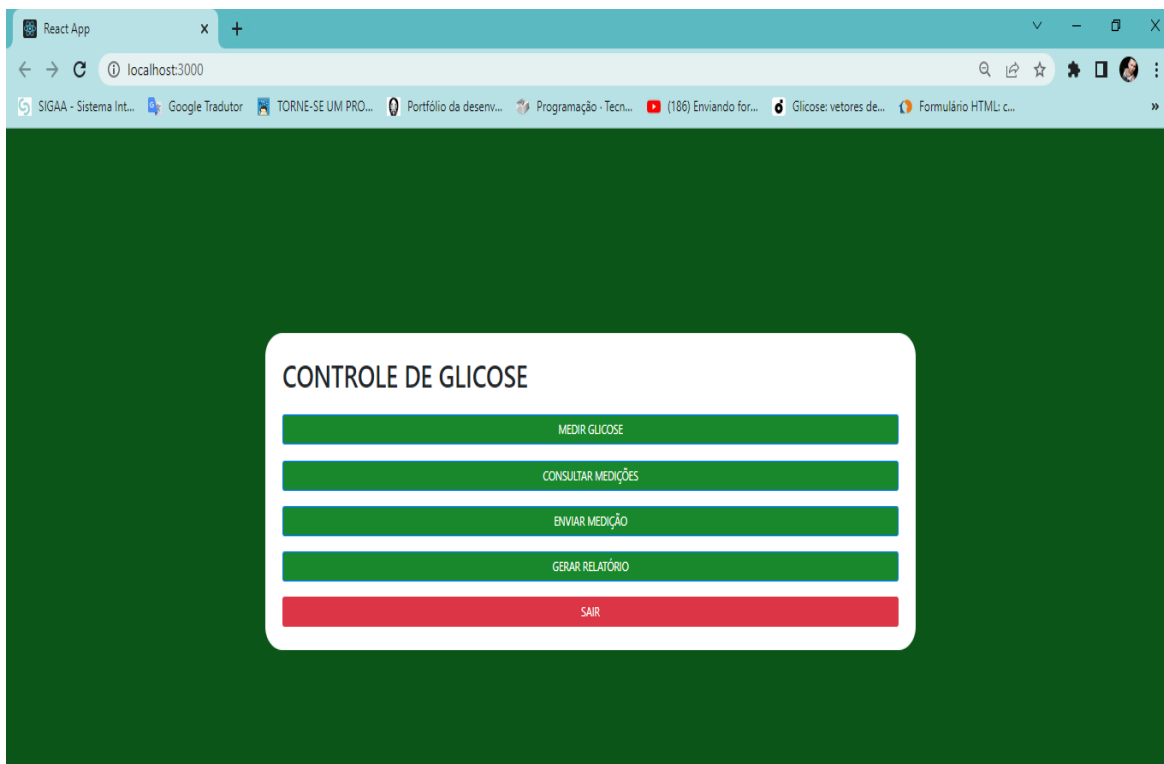
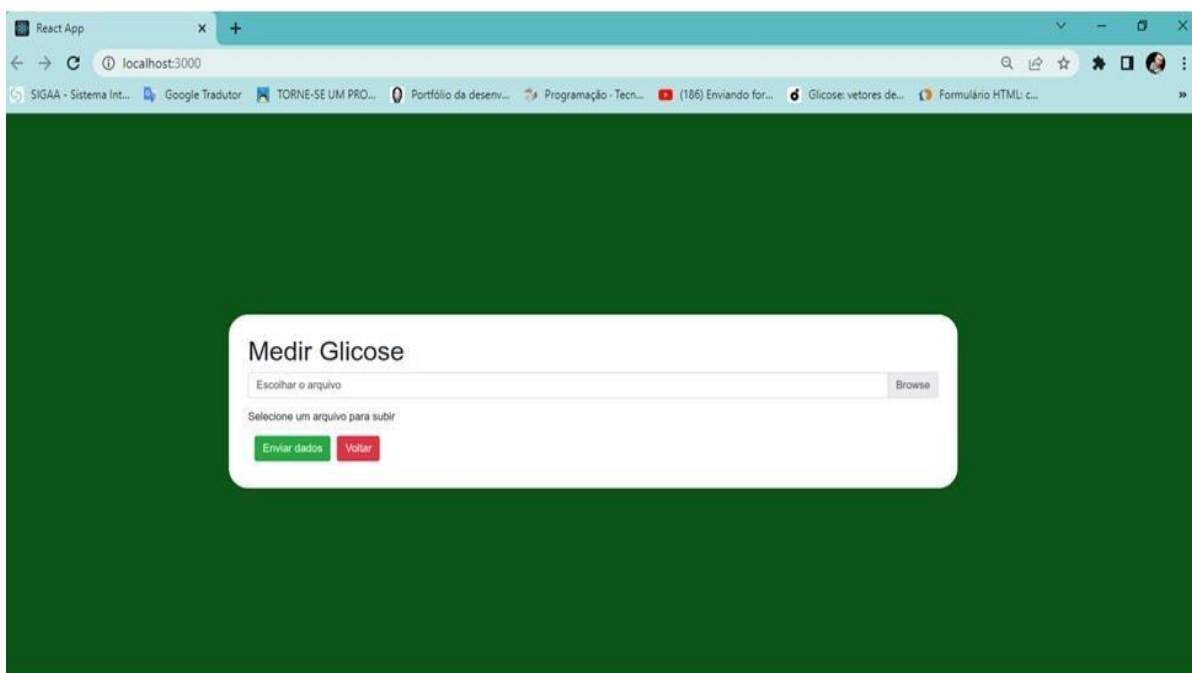


Figura 19 – Tela de Medir glicose.



React App

localhost:3000

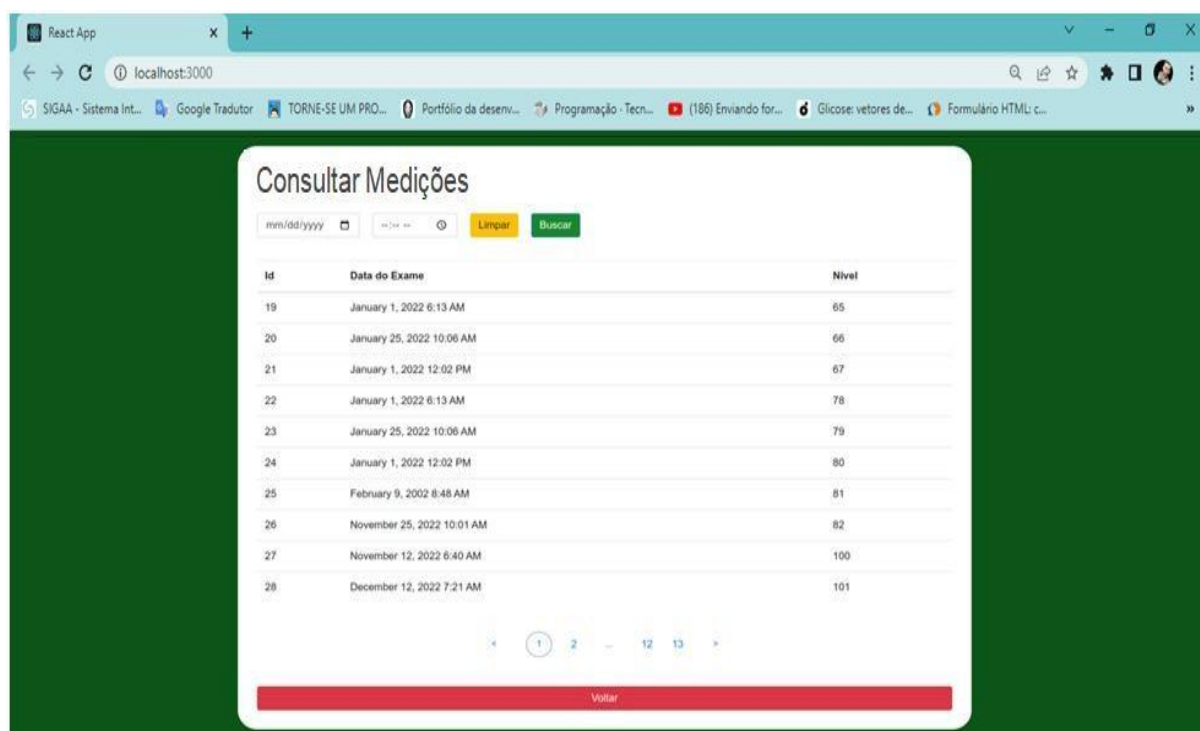
Medir Glicose

Escolhar o arquivo Browse

Selecione um arquivo para subir

Enviar dados Voltar

Figura 19 – Tela de consultar medições.



React App

localhost:3000

Consultar Medições

mm/dd/yyyy Limpar Buscar

Id	Data do Exame	Nível
19	January 1, 2022 6:13 AM	65
20	January 25, 2022 10:06 AM	66
21	January 1, 2022 12:02 PM	67
22	January 1, 2022 6:13 AM	78
23	January 25, 2022 10:06 AM	79
24	January 1, 2022 12:02 PM	80
25	February 9, 2002 8:48 AM	81
26	November 25, 2022 10:01 AM	82
27	November 12, 2022 6:40 AM	100
28	December 12, 2022 7:21 AM	101

Voltar

Figura 20 – Tela de Enviar Medição.

React App

localhost:3000

SIGAA - Sistema Int... Google Tradutor TORNE-SE UM PRO... Portfólio da desenv... Programação - Tec... (186) Enviando for... Glicose: vetores de... Formulário HTML: c...

Enviar Medição

Email

Enviar Voltar

Figura 21 – Tela de Gerar Relatório.

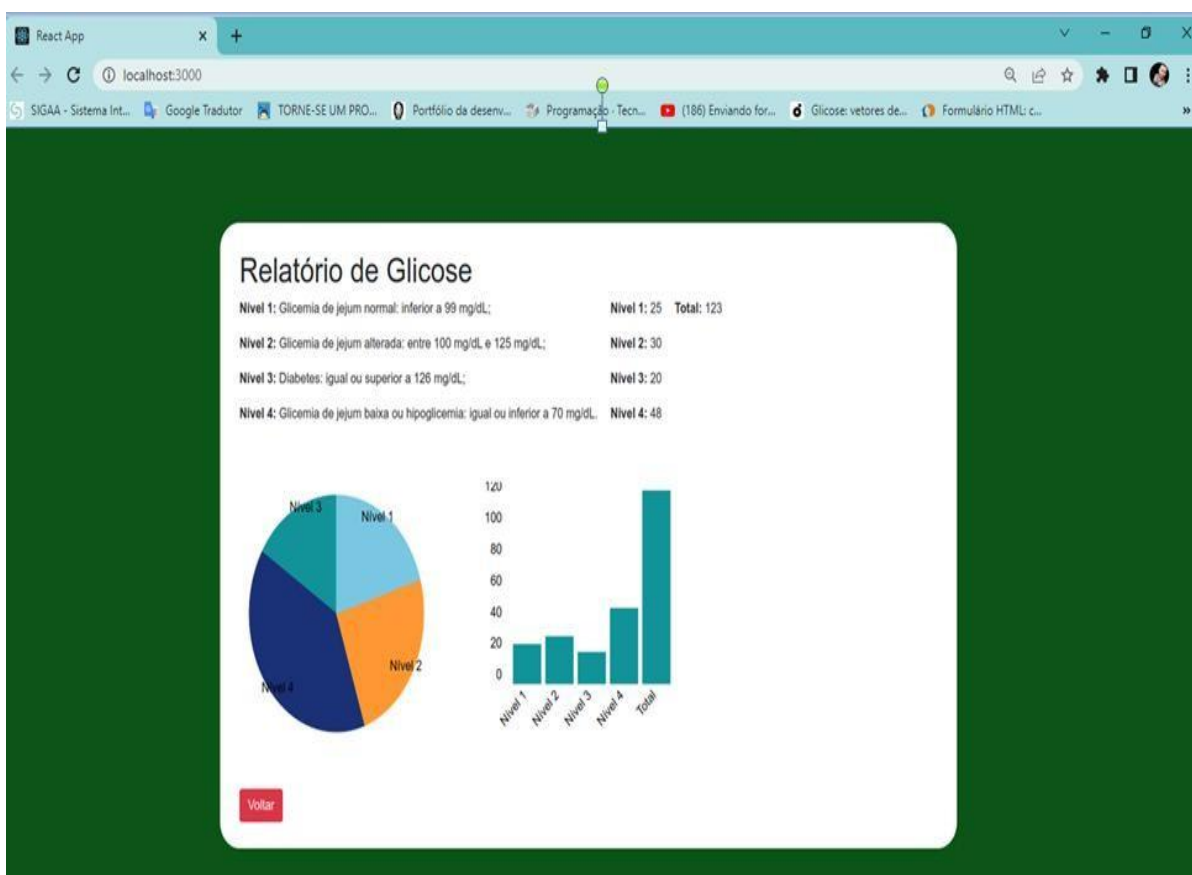
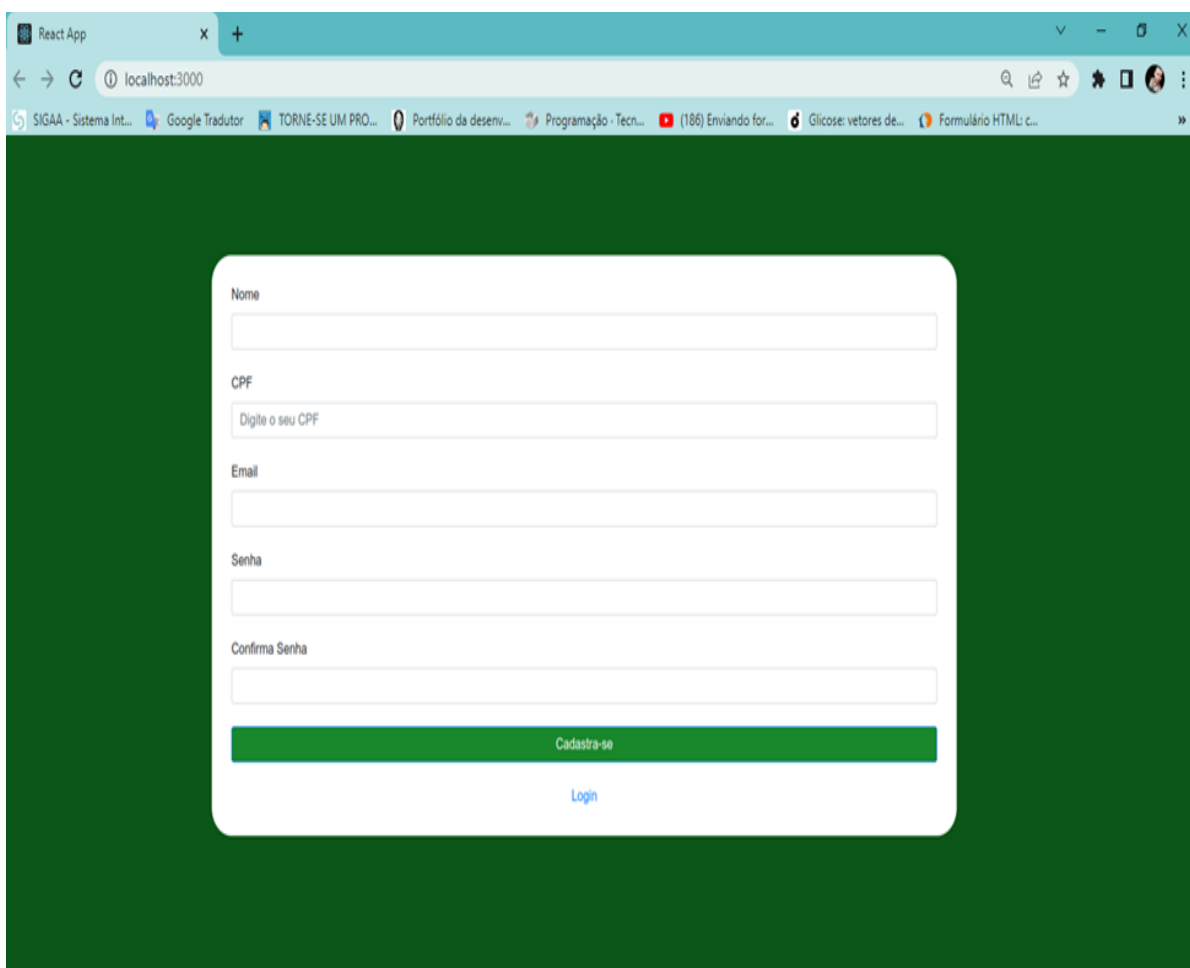


Figura 22 – Tela de Cadastro.



The image shows a web browser window with a single tab titled "React App". The address bar displays "localhost:3000". The browser's bookmark bar contains several items: "SIGAA - Sistema Int...", "Google Tradutor", "TORNE-SE UM PRO...", "Portfólio da desenv...", "Programação - Tecn...", "(186) Enviando for...", "Glicose: vetores de...", and "Formulário HTML: c...".

The main content area features a dark green background with a white registration form centered on it. The form includes the following elements:

- Nome**: A text input field.
- CPF**: A text input field with the placeholder text "Digite o seu CPF".
- Email**: A text input field.
- Senha**: A text input field.
- Confirma Senha**: A text input field.
- Cadastrar-se**: A prominent green button.
- Login**: A blue text link located below the "Cadastrar-se" button.

Anexo 2 Site usado para criação de dados

Figura 22 – Campos a serem preenchidos.

Mockaroo - níveis

mockaroo.com/schemas/454386

SIGAA - Sistema Int... Google Tradutor TORNE-SE UM PRO... Portfólio da desenv... Programação - Tecn... (186) Enviando for... Glicose: vetores de... Formulário HTML: c...

mockaroo ESQUEMAS¹ CONJUNTOS DE DADOS MOCK APIS CENÁRIOS PROJETOS

ATUALIZE AGORA

níveis

Nome do campo	Modelo	Opções
id	Número da linha	em branco: 0 %
data	Data hora	07/04/2022 para 12/15/2022 formato: dd-mm-aaaa em branco: 0 %
hora	Tempo	a partir de 6:00 AM para 12:59 PM formato: 24 horas em branco: 0 %
nivel	Número	min: 70 máximo: 400 decimais: 0 em branco: 0 %

ADICIONAR OUTRO CAMPO

Linhas: 1000 Formato: CSV Fim da linha: Unix (LF) Incluir: ☒ cabeçalho ☐ BOM

Anexar conjunto de dados: escolha um conjunto de dados...

BAIXAR DADOS VISUALIZAÇÃO ALTERAÇÕES SALVAS CRIAR API MAIS

Figura 23 – Resultados dos dados.

Mockaroo - níveis

mockaroo.com/schemas/454386

SIGAA - Sistema Int... Google Tradutor TORNE-SE UM PRO... Portfólio da desenv... Programação - Tecn... (186) Enviando for... Glicose: vetores de... Formulário HTML: c...

mockaroo ESQUEMAS¹ CONJUNTOS DE DADOS MOCK APIS CENÁRIOS PROJETOS

ATUALIZE AGORA

Visualização

MESA CRU

Eu iria	dados	hora	nivel
1	17-08-2022	12:25	259
2	25-11-2022	6:43	190
3	02-09-2022	10:19	169
4	21-08-2022	8:37	270
5	11-12-2022	11:19	362
6	10-06-2022	9:56	341
7	28-08-2022	7:00	355
8	28-10-2022	8:58	104
9	19-07-2022	11:34	173
10	12-09-2022	9:42	373

mostrando as primeiras 100 linhas

Linhas: 1000 BAIXAR DADOS PERTO

BAIXAR DADOS VISUALIZAÇÃO ALTERAÇÕES SALVAS CRIAR API MAIS

