

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA
CATARINA – CAMPUS FLORIANÓPOLIS**

DEPARTAMENTO ACADÊMICO DE METAL-MECÂNICA

BACHARELADO EM ENGENHARIA MECATRÔNICA

MARINA PINTO DE SOUSA MAGALHÃES DARDENNE

DESENVOLVIMENTO DO MEEPI: porta remédios inteligente

FLORIANÓPOLIS, 2021.

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA
CATARINA – CAMPUS FLORIANÓPOLIS**

DEPARTAMENTO ACADÊMICO DE METAL-MECÂNICA

BACHARELADO EM ENGENHARIA MECATRÔNICA

MARINA PINTO DE SOUSA MAGALHÃES DARDENNE

DESENVOLVIMENTO DO MEEPI: porta remédios inteligente

Trabalho de Conclusão de Curso submetido ao Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina como parte dos requisitos para obtenção do título de Engenheira Mecatrônica.

Orientadora:

Profa. Dra. Cynthia Beatriz Scheffer Dutra

FLORIANÓPOLIS, 2021.

FICHA DE IDENTIFICAÇÃO

Ficha de identificação da obra elaborada pelo autor.

Dardenne, Marina
Desenvolvimento do Meepi : porta remédios inteligente
/ Marina Dardenne ; orientação de Cynthia Beatriz Scheffer
Dutra. - Florianópolis, SC, 2021.
101 p.

Trabalho de Conclusão de Curso (TCC) - Instituto Federal
de Santa Catarina, Câmpus Florianópolis. Bacharelado
em Engenharia Mecatrônica. Departamento
Acadêmico de Metal Mecânica.
Inclui Referências.

1. Porta remédios. 2. MVP. 3. Internet das Coisas.
4. Polifarmácia. 5. Telemedicina. I. Beatriz Scheffer
Dutra, Cynthia. II. Instituto Federal de Santa Catarina.
Departamento Acadêmico de Metal Mecânica. III.
Título.

DESENVOLVIMENTO DO MEEPI: PORTA REMÉDIOS INTELIGENTE

MARINA PINTO DE SOUSA MAGALHÃES DARDENNE

Este trabalho foi julgado adequado para obtenção do Título de Engenheira Mecatrônica e aprovado na sua forma final pela banca examinadora do Curso de Engenharia Mecatrônica do Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina.

Florianópolis, 20 de abril, 2021

Banca Examinadora:

CYNTHIA BEATRIZ SCHEFFER
DUTRA:01756197946
2021.05.13 15:20:24 -03'00'

Cynthia Beatriz Scheffer Dutra, Dra. Eng.



Documento assinado digitalmente
Adriano Regis
Data: 13/05/2021 15:32:41-0300
CPF: 039.297.539-44

Adriano Regis, Me.



Documento assinado digitalmente
Gregory Chagas da Costa Gomes
Data: 13/05/2021 19:42:13-0300
CPF: 073.008.969-01

Gregory Chagas da Costa Gomes, Me. Eng.



MINISTÉRIO DA EDUCAÇÃO

SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE SANTA CATARINA
CAMPUS FLORIANÓPOLIS

DECLARAÇÃO DE FINALIZAÇÃO DE TRABALHO DE CURSO

Declaro que o estudante **Marina Pinto de Sousa Magalhães Dardenne**, matrícula nº 1610004183, do Curso de Engenharia Mecatrônica, defendeu o trabalho intitulado **DESENVOLVIMENTO DO MEEPI: porta remédios inteligente**, o qual está apto a fazer parte do banco de dados da Biblioteca Hercílio Luz do Instituto Federal de Santa Catarina, Campus Florianópolis.

Florianópolis, 30 de abril de 2021.

CYNTHIA BEATRIZ SCHEFFER
DUTRA:01756197946
2021.05.13 15:21:31 -03'00'

Prof. Orientador do TCC: Cynthia Beatriz Scheffer Dutra

AGRADECIMENTOS

Gostaria de agradecer a minha mãe, Hilda Pinto, e ao meu pai, Marco Dardenne, por todo o apoio dado por mim até então. Vocês são minha fortaleza.

Também gostaria de aproveitar a oportunidade para agradecer a todos meus parentes, amigos e professores que colaboraram com a minha jornada até o presente momento. Durante a vida encontrei diversas pessoas que me ajudaram ou me inspiraram e a elas eu também agradeço. Logo, um muito obrigada especial para Teresa Cristina Amâncio, Edivan Amâncio, Jéssica Amâncio, Lucas Amâncio, Rita de Cássia Nascimento, Eduardo Nascimento, Monique Moura, Carlos Cesar Marinho, Arthur Kretzer, Augusto Aguiar, Gabriela Mota, Isa Coutinho, Gabriel Dal Ponte e muitas outras pessoas que não foram citadas, mas que também estão no meu coração.

Agradeço também todo o corpo de professores da Engenharia Mecatrônica do Instituto Federal de Santa Catarina pela dedicação e pela paciência. Em especial, meus agradecimentos a minha orientadora, Cynthia Dutra, e a banca examinadora.

RESUMO

A partir da visualização do contexto do mercado farmacêutico e do envelhecimento da população mundial, é objetivo deste trabalho de conclusão de curso desenvolver um protótipo de porta remédios inteligente, de modo a ajudar pessoas, em sua maioria idosos, que consomem múltiplos medicamentos a seguir o tratamento de maneira adequada. Para isso, teve-se em mente a definição de Produto Mínimo Viável (MVP), presente nos conceitos da *startup* enxuta. Além disso, de modo geral, realizou-se uma pesquisa de mercado, definição das funcionalidades para o MVP e análise de soluções para o protótipo em questão. Na construção desse porta medicamento, nomeado de Meepi, foi feito o projeto de eletrônico de uma placa, assim como foi efetuada a sua fabricação. Também foi escopo do presente trabalho a modelagem tridimensional de um reservatório para os medicamentos, o qual foi fabricado por meio de impressão 3D. Na parte de programação do Meepi, utilizou-se se ferramentas como IBM *Cloud* e Node-RED, além de desenvolver a programação na ESP32. Ao final do trabalho se analisou se o protótipo desenvolvido estava apto para ser considerado um MVP, no qual conclui-se que seriam ainda necessários ajustes para que fosse possível a testagem com um potencial cliente. De modo geral, os objetivos do presente trabalho foram atingidos uma vez que foi possível a montagem de um protótipo funcional e a realização de uma pesquisa de mercado voltada para esse potencial produto.

Palavras-chave: Porta remédios, MVP, Internet das Coisas, Polifarmácia, Telemedicina

ABSTRACT

From the observation of the pharmaceutical business context and the world's aging population, the goal of this thesis is the development of a smart pillbox aiming to help people, mainly elders who take several medications, to follow the treatment correctly. Due to that, this thesis had the concept of Minimum Viable Product (MVP) from the lean startup in mind. Following this concept, a market research, a functionalities definition, and a solutions analysis were made before the hardware and software development. In the development of this pillbox, called Meepi, a PCB project was made and fabricated. Still inside the scope of this project was a three-dimensional model of the pillbox which was later 3D printed. Regarding Meepi's software development, tools such as IBM Cloud and Node-RED were applied, while the hardware coding was done in a ESP32 board. At the end of this work, despite all the functionalities having been accomplished, it was concluded that adjustments were still required to consider Meepi as an MVP. In conclusion, the main goals of this thesis were achieved, since a functional prototype was built, and the market research was done to make this project a potential product.

Keywords: Pill box, MVP, Internet of Things, Polypharmacy, Telemedicine

LISTA DE FIGURAS

Figura 1 – Ciclo Construir-Medir-Aprender.....	17
Figura 2 – Resultados da pergunta nº8 da pesquisa de mercado	21
Figura 3 – Resultados da pergunta nº9 da pesquisa de mercado	21
Figura 4 – Resultados da pergunta nº10 da pesquisa de mercado	22
Figura 5 – Resultados da pergunta nº12 da pesquisa de mercado	22
Figura 6 – Número de medicamentos consumidos por Faixa Etária.....	25
Figura 7 – Faixa Etária e Pontualidade na ingestão de medicamentos.....	26
Figura 8 – Dúvida referente a ingestão por faixa etária	26
Figura 9 – Exemplo de organizador de pílulas.....	27
Figura 10 – Organizador de pílulas Océane.....	28
Figura 11 – Dispensador de medicamentos Automático LiveFine	29
Figura 12 – Pillgo.....	30
Figura 13 – Arduino UNO R3	40
Figura 14 – ESP32.....	41
Figura 15 – Raspberry Pi Zero W.....	41
Figura 16 – Pinagem da ESP32	44
Figura 17 – Esquemático do Meepi	45
Figura 18 – Módulo Carregador de Bateria Lítio TP4056	46
Figura 19 – Conversor <i>boost</i> DC Ajustável.....	47
Figura 20 – DS1307	48
Figura 21 – Encoder Decoder KY-040 Rotacional	49
Figura 22 – Layout da placa eletrônica do Meepi.....	50
Figura 23 – Placa controladora do Meepi.....	50
Figura 24 – Vista da parte do circuito	51
Figura 25 – Vista explodida do Meepi	51

Figura 26 – Versão Final do Meepi.....	52
Figura 27 – Impressões 3D.....	52
Figura 28 – Central de dispositivos do Internet of Things Platform.....	55
Figura 29 – Central de chaves de API	55
Figura 30 – Node-RED	57
Figura 31 – Flow Cadastros Usuários/Tutor	58
Figura 32 – UI do Cadastro de Usuário.....	59
Figura 33 – UI do Cadastro do Tutor	59
Figura 34 – UI Cadastro de Medicamentos	61
Figura 35 – Flow de Cadastro Medicamentos	61
Figura 36 – UI Dashboard.....	63
Figura 37 – Parte do Flow Dashboard	64
Figura 38 – Parte 1/3 do Flow Comunicação	64
Figura 39 – Parte 2/3 do Flow Comunicação	65
Figura 40 – Parte 3/3 do Flow Comunicação	66
Figura 41 – Fluxograma Meepi Parte 1/7	68
Figura 42 – Fluxograma Meepi Parte 2/7	69
Figura 43 – Fluxograma Meepi Parte 3/7	70
Figura 44 – Fluxograma Meepi Parte 4/7	71
Figura 45 – Fluxograma Meepi Parte 5/7	72
Figura 46 – Fluxograma Meepi Parte 6/7	73
Figura 47 – Fluxograma Meepi Parte 7/7	74
Figura 48 – Caixa Meepi Montada (vista superior).....	75
Figura 49 – Caixa Meepi Montada (vista isométrica).....	76
Figura 50 – Início da conversa no Telegram com responsável e com o usuário.....	77
Figura 51 – Dashboard após programação dos remédios	78

Figura 52 – Dashboard após o primeiro horário do medicamento 1	78
Figura 53 – Dashboard após primeiro horário do medicamento 2.....	79
Figura 54 – Mensagens recebidas no Telegram do tutor após primeiro horário do medicamento 1 e do medicamento 2.....	79
Figura 55 – Mensagens recebidas no Telegram do usuário após primeiro horário do medicamento 1 e do medicamento 2.....	80
Figura 56 – Mensagem para o usuário e para o tutor referente ao reabastecimento do Meepi.....	81
Figura 57 – Caixa Meepi em funcionamento	81

LISTA DE QUADROS

Quadro 1 – Opções para armazenamento de medicamento.....	34
Quadro 2 – Opções para notificar momento da ingestão.....	35
Quadro 3 – Opções para informar a localização do remédio a ser ingerido	36
Quadro 4 – Opções para permitir a programação dos horários.....	36
Quadro 5 – Opções para detectar se o usuário ingeriu o medicamento.....	37
Quadro 6 – Opções para notificação de terceiros.....	37

LISTA DE TABELAS

Tabela 1 – Respostas referentes ao tipo de medicamento	24
Tabela 2 – Média de respostas para questões 4, 5 e 13 do formulário	24
Tabela 3 – Métodos utilizados para lembrar a ingestão de remédios	25
Tabela 4 – Resumo informações organizador de pílulas Océane	28
Tabela 5 – Resumo informações dispensador automático LiveFine	29
Tabela 6 – Resumo de informações Pillgo	31
Tabela 7 – Lista de soluções para o protótipo	38
Tabela 8 – Resumo referente a preço, a dimensões e a conectividade	42
Tabela 9 – Componentes eletrônicos principais para o projeto	43
Tabela 10 – Conexões do TP4056	47
Tabela 11 – Conexões do <i>Step Up</i>	47
Tabela 12 – Lista de bibliotecas e funções.....	68

SUMÁRIO

1	INTRODUÇÃO	16
1.1	Objetivo principal	17
1.2	Objetivos específicos	17
2	METODOLOGIA.....	17
3	HIPÓTESES DE PROBLEMA	19
3.1	Pesquisa na literatura científica	19
3.2	Pesquisa <i>on-line</i> aplicada	20
4	ESTUDO DE MERCADO	22
4.1	Contexto de mercado.....	23
4.2	Cliente	24
4.3	Pesquisa de concorrentes	27
4.3.1	Organizadores de pílulas.....	27
4.3.2	LiveFine – Dispensador Automático de Medicamentos.....	28
4.3.3	Pillgo.....	30
5	PROPOSTA DE SOLUÇÃO	31
5.1	Objetivos da proposta de solução	31
5.2	Requisitos do Produto Mínimo Viável (MVP).....	32
5.3	Funcionalidades do MVP.....	33
5.4	Análise de soluções tecnológicas.....	33
6	DESENVOLVIMENTO DO PROTÓTIPO.....	39
6.1	Desenvolvimento eletrônico.....	39
6.1.1	Escolha de <i>hardware</i> central.....	40
6.1.2	Projeto eletrônico.....	43
6.2	Desenvolvimento mecânico	51
6.3	Programação.....	53

6.3.1	IBM Cloud e Node-RED	53
6.3.1.1	Configuração do Internet of Things Platform	55
6.3.1.2	Programação no Node-RED	57
6.3.2	ESP32	67
6.4	Integração do porta medicamentos	75
7	TESTE DO MEEPI.....	76
8	CONCLUSÃO E TRABALHOS FUTUROS	82
	REFERÊNCIAS	84
	APÊNDICE A – Questionário do Google Forms.....	87
	APÊNDICE B – <i>Dashboard</i> durante os testes.....	92
	APÊNDICE C – Mensagens recebidas no Telegram do usuário (testes).....	95
	APÊNDICE D – Mensagens recebidas no Telegram do tutor (testes)	98

1 INTRODUÇÃO

De acordo com projeções realizadas pelo censo de 2010 do IBGE, Instituto Brasileiro de Geografia e Estatística, há a possibilidade de que em 2043 aproximadamente um quarto da população brasileira será de idosos (PERISSÉ; MARLI, 2019). Diante disso, com o envelhecimento da população mundial, torna-se crucial o investimento em tecnologias que permitem aumentar a qualidade de vida do idoso, ou seja, pessoas acima de 60 anos.

Sabe-se que a idade é um fator que colabora para o início de tratamentos farmacológicos e que existem indícios que mostram que a chance de usar algum medicamento aumenta desde a quarta década de vida, não se limitando apenas a faixa etária a partir dos 60 anos (SOUSA et al., 2011). De acordo com Maahs e Serrate (2005, p.925), “estima-se que 23% da população brasileira consomem 60% da produção nacional de medicamentos, principalmente as pessoas acima de 60 anos”.

Diante desse cenário, o projeto em questão apresenta o objetivo de desenvolver um porta remédios inteligente, voltado para o público idoso. Dessa maneira, no presente trabalho são feitas pesquisas na literatura científica, em reportagens, em relatórios, entre outras fontes, de modo a entender melhor o cenário do consumo de medicamentos pela sociedade. De modo adicional, elaborou-se uma pesquisa *on-line*, de modo a investigar aspectos importantes para o estudo de mercado.

Com a realização dos estudos de mercado, é feita a análise de soluções e de funcionalidades destinadas para o protótipo do porta remédios, o qual foi denominado Meepi. Depois disso, inicia-se a etapa de desenvolvimento, a qual inclui o projeto, a aquisição e a fabricação do *hardware* da caixa, a modelagem tridimensional do Meepi, impressão 3D e programação da interface e do próprio *hardware* baseado na ESP32. Com essas etapas concluídas, o próximo passo é a realização dos testes e da avaliação do desempenho do porta medicamentos desenvolvido.

É importante destacar que as atividades desenvolvidas no presente trabalho foram baseadas em múltiplas metodologias, mas destaca-se o conceito de Mínimo Produto Viável (MVP), além do método iterativo incremental utilizado durante o processo de programação.

1.1 Objetivo principal

Desenvolvimento de um protótipo de um porta remédios inteligente voltado ao público idoso, visando a aplicação de conhecimentos em eletrônica, em mecânica e em programação.

1.2 Objetivos específicos

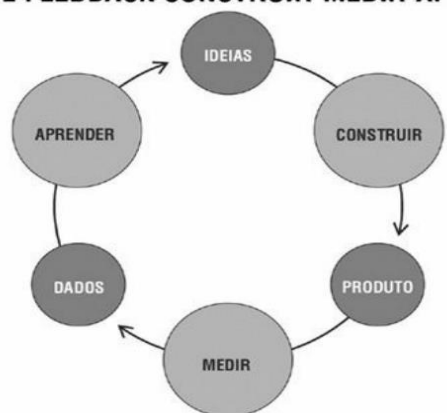
- Desenvolver uma solução que seja acessível a todas as faixas etárias, principalmente ao idoso;
- Pesquisar tecnologias existentes no mercado e tendências;
- Desenvolver conceito da solução;
- Especificar componentes necessários para o protótipo;
- Desenvolver o protótipo;
- Realizar testes para validação da solução;
- Elaborar documentação do projeto.

2 METODOLOGIA

Para o projeto em questão serão utilizadas metodologias tradicionais de desenvolvimento mescladas com princípios trazidos pelo startup enxuta.

Figura 1 – Ciclo Construir-Medir-Aprender

CICLO DE FEEDBACK CONSTRUIR-MEDIR-APRENDER



Minimizar o tempo TOTAL através do ciclo

Fonte: Ries (2012)

Em relação a maneira mais tradicional, o trabalho apresenta influências do modelo PRODIP de Back et al. (2008). O modelo PRODIP é o método para desenvolvimento de produto ensinado no curso de Engenharia Mecatrônica. Ele é

composto de 4 fases: projeto informacional, projeto conceitual, projeto preliminar e projeto detalhado.

Já em relação aos fundamentos do startup enxuta, para montagem do plano de trabalho a autora se baseou no ciclo de “Construir-Medir-Aprender” (Figura 1). Esse ciclo é explorado no livro *Startup Enxuta*, no qual o autor destaca a importância de construir um MVP (Mínimo Produto Viável) para validar uma ideia. O MVP é uma versão do produto que “permite uma volta completa do ciclo construir-medir-aprender, com o mínimo de esforço e o menor tempo de desenvolvimento. O produto mínimo viável carece de diversos recursos que podem se provar necessários mais tarde”. (RIES, 2012, p.58). Além disso, o autor ainda destaca:

Ao contrário do desenvolvimento de produto tradicional, que, em geral, envolve um período de incubação longo e ponderado e aspira à perfeição do produto, o objetivo do MVP é começar o processo de aprendizagem, não terminá-lo. Diferentemente de um protótipo ou teste de conceito, um MVP é projetado não só para responder a perguntas técnicas ou de design do produto. Seu objetivo é testar hipóteses fundamentais do negócio. (RIES, 2012, p.70).

A segunda etapa do ciclo da Figura 1 seria testar esse protótipo com o público-alvo, com objetivo de medir seu desempenho. A partir dos dados obtidos por essa segunda etapa, procura-se aprender, e dessa maneira, verificar o que funcionou ou não na sua ideia. Ao final desse ciclo, de acordo com Ries (2012), o empreendedor apresenta duas escolhas: mudar a ideia original, o que ele chama de pivotar, ou perseverar na ideia.

Com esse processo, o objetivo é que o empreendedor desenvolva um produto que se aproxime o máximo possível da necessidade do cliente, além de permitir um aprendizado validado para tornar determina ideia inovadora em um produto sustentável.

Diante dos conceitos explorados, este trabalho visará o desenvolvimento da etapa de construção da ideia, conforme apresentado no ciclo da Figura 1. Ao final do trabalho será avaliado se o protótipo desenvolvido está adequado para ser um MVP da ideia proposta e, dessa maneira, estar apto para avançar nas demais etapas do ciclo.

Por fim, no desenvolvimento do *software*, será empregada a abordagem do método iterativo incremental. O modelo iterativo corresponde à ideia de melhorar pouco a pouco, ou seja, refinar o sistema. A essência do sistema não é alterada, mas

o seu detalhe vai aumentando em iterações sucessivas. Por sua vez, o modelo incremental corresponde à ideia de aumentar gradativamente a esfera do sistema, ou seja, alargar o sistema em sucessivos incrementos (SILVA; VIDEIRA, 2008).

3 HIPÓTESES DE PROBLEMA

Para o desenvolvimento do presente Trabalho de Conclusão foram formuladas as seguintes hipóteses de problema:

- Hipótese 01: as pessoas, em especial idosos, costumam esquecer de tomar medicamentos ou não os ingerem nos horários prescritos;
- Hipótese 02: as pessoas, em especial idosos, costumam ficar na dúvida se já ingeriram ou não determinado medicamento;
- Hipótese 03: pessoas que moram distantes de familiares com mais idade apresentam dificuldade para acompanhar o tratamento farmacológico.

Após a formulação das hipóteses de problema, em um primeiro momento, foi realizada uma pesquisa na literatura científica para melhor compreensão do cenário relacionado a adesão medicamentosa por parte do idoso, ou seja, entendimento das razões que colaboram com o seguimento do tratamento conforme prescrito pelo profissional da saúde. Em seguida, foi formulada e aplicada uma pesquisa com o público em geral, cujos objetivos são validar as hipóteses de problema e complementar a pesquisa na literatura.

A pesquisa *on-line* foi realizada por meio da ferramenta gratuita de formulários do Google. Esse método foi escolhido devido às limitações trazidas pela pandemia provocada pelo novo coronavírus.

3.1 Pesquisa na literatura científica

Com a pesquisa na literatura científica, observou-se que com o aumento da idade, maior é a possibilidade de incidência de múltiplas patologias (MAAHS; SERRATE, 2005), o que resulta em um consumo de diferentes medicamentos e, por consequência, a polimedicação. Segundo Maahs e Serrate (2005, p.925) “há evidências de que os idosos americanos, vivendo na comunidade, tomam em média de 2,7 a 4,2 medicamentos prescritos ou não prescritos”.

Como normalmente essas patologias são do tipo crônica não transmissível, como a hipertensão, o paciente utiliza medicamentos de maneira contínua e por um grande período. Contudo, em muitos casos ocorre a não adesão ou não aderência ao tratamento, o que pode comprometer a eficácia da medicação. De acordo com ROCHA et al. (2008), a Organização Mundial da Saúde (OMS) define a adesão ou a aderência à prescrição médica como o nível em que o comportamento do paciente segue as recomendações realizadas pelo seu médico.

Esse não cumprimento do esquema posológico, que se refere à dose do fármaco, à frequência de administração e à duração do tratamento prescrito, apresenta várias razões, como: “consumo elevado e uso prolongado, efeitos colaterais, desaparecimento dos sintomas, desconhecimento sobre os medicamentos, alto custo das medicações, falta de motivação, analfabetismo e distúrbios de memória.” (CINTRA; GUARIENTO; MIYASAKI, 2010, p.508). Além disso, conforme destacado por ROCHA et al. (2008, p.705), “a exposição a múltiplos medicamentos expõe o idoso a um tratamento mais complexo, exigindo maior atenção, memória e organização diante dos horários de administração dos fármacos”.

Por fim, outra questão observada nos estudos de Cintra, Guariento e Miyazaki (2010) e Aiolfi et al. (2015) é a influência positiva do apoio das pessoas, sejam elas familiares ou não, para adesão do idoso ao tratamento farmacológico. Também, na pesquisa realizada por Cintra, Guariento e Miyazaki (2010) ainda foi mostrado que idosos que moravam sozinhos tinham três vezes mais chances de não aderência ao tratamento.

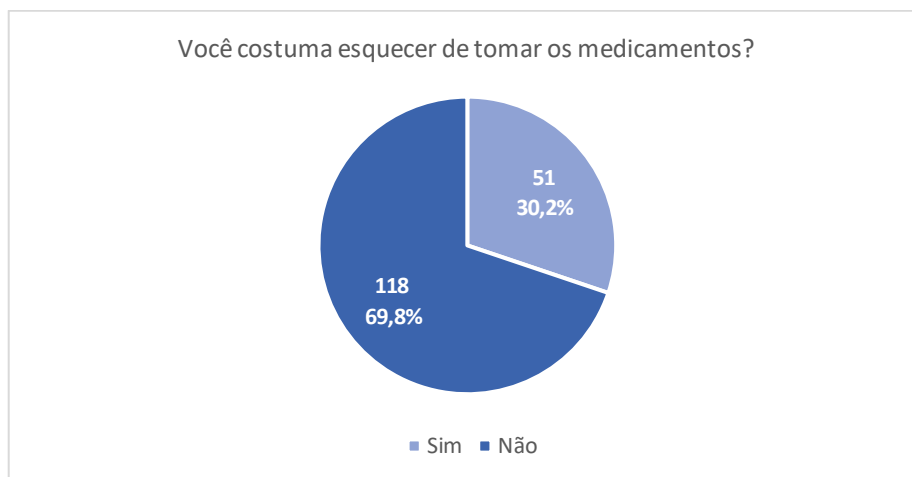
3.2 Pesquisa *on-line* aplicada

Após a pesquisa científica, foi feita e aplicada uma pesquisa de mercado através do *Google Forms*. O formulário (Apêndice A) contava com 13 perguntas, dentre as quais três estavam voltadas para validação de hipóteses de problema. Apenas 28 pessoas com mais de 60 anos responderam à pesquisa.

A primeira questão referente as conjecturas de problema, a questão 8 do formulário, está relacionada a hipótese 01. Conforme pode ser observado na Figura 2, 69,8% dos 169 respondentes alegaram não esquecer de tomar seus medicamentos. No entanto, ainda há uma porcentagem relevante de pessoas que costumam esquecer, a qual o presente produto poderia ser destinado. Já a questão

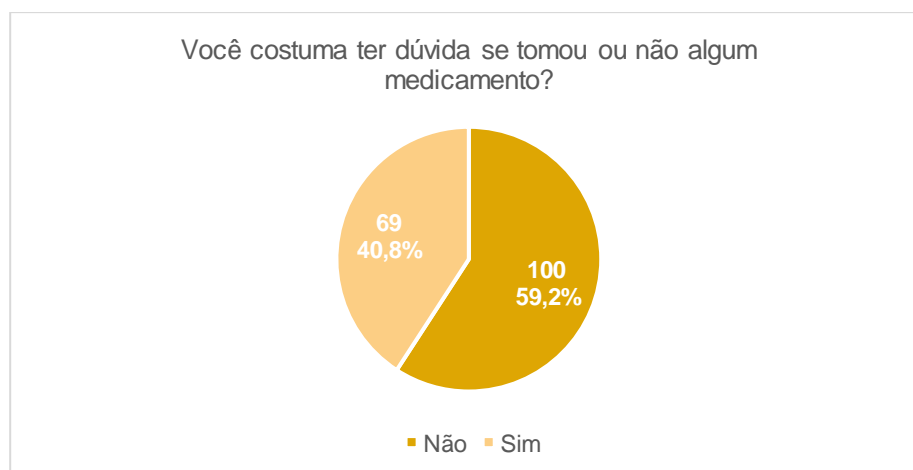
seguinte, questão 9, está relacionada a hipótese 02. Observa-se nas respostas que a maioria costuma não ficar na dúvida, contudo ainda 40,8% ainda apresentam esse problema.

Figura 2 – Resultados da pergunta nº8 da pesquisa de mercado



Fonte: Elaboração própria (2020).

Figura 3 – Resultados da pergunta nº9 da pesquisa de mercado



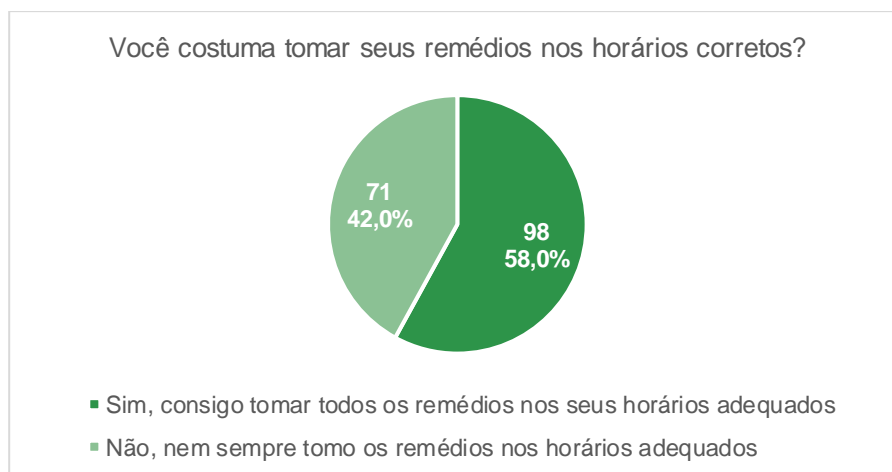
Fonte: Elaboração própria (2020).

Por fim, a última pergunta relacionada as hipóteses de problema é a questão 10, a qual aborda o aspecto do cumprimento dos horários prescritos. Pode-se ver que nos resultados presentes na Figura 4 é observado que a maioria costuma tomar seus remédios de maneira pontual.

Ao avaliar os resultados obtidos nos questionamentos mostrados, verifica-se que as hipóteses de problema levantadas não contemplam a maioria dos respondentes. Contudo, apesar de não ser um problema geral, na questão 12 do formulário, 53,3% responderam que investiriam em uma solução para o problema,

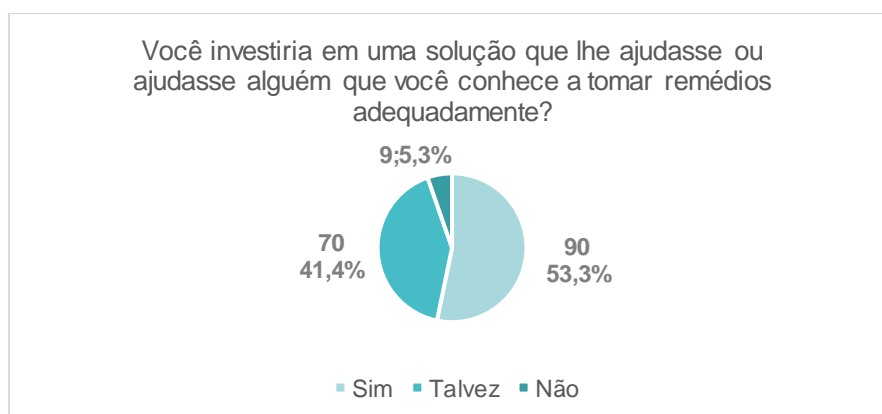
conforme pode ser observado na Figura 5. Vale destacar que 41,4% informaram que talvez investiria na solução, o que pode ser considerado um mercado em potencial também. Caso nessa pergunta ocorresse a maior frequência da resposta “Não”, poderia se dizer que a solução possivelmente não atenderia uma demanda do mercado.

Figura 4 – Resultados da pergunta nº10 da pesquisa de mercado



Fonte: Elaboração própria (2020).

Figura 5 – Resultados da pergunta nº12 da pesquisa de mercado



Fonte: Elaboração própria (2020).

4 ESTUDO DE MERCADO

O presente capítulo se dedica a realizar um estudo do mercado. Diante disso, buscou-se compreender o contexto de mercado no qual esse potencial produto está inserido e o cliente a qual seria destinado. Em seguida, foi realizada uma pesquisa de correntes.

4.1 Contexto de mercado

Em 2018, o gasto global em medicamentos chegou a 1,2 trilhão de dólares (AITKEN et al., 2019). De acordo com o guia publicado pela Associação da Indústria Farmacêutica de Pesquisa (2019), Interfarma, só o mercado farmacêutico brasileiro em 2018 movimentou cerca de US\$ 39 bilhões, onde foram comercializadas 238 milhões de doses de medicamentos. Todo esse cenário coloca o país entre os 10 maiores mercados de medicamentos do mundo e com projeções de que até 2023 se torne o sexto maior mercado (AITKEN et al., 2019).

Junto ao panorama do mercado farmacêutico brasileiro e mundial, para o presente trabalho é interessante observar, também, o processo de envelhecimento da população que está ocorrendo globalmente. De acordo com o censo de 2010 do IBGE - Instituto Brasileiro de Geografia e Estatística - a população com 60 anos ou mais no Brasil era de aproximadamente 20,6 milhões (PERISSÉ; MARLI, 2019). Existem projeções do mesmo órgão que mostram a possibilidade de que em 2043 aproximadamente um quarto da população brasileira será de idosos (PERISSÉ; MARLI, 2019).

De maneira complementar, só nos Estados Unidos há uma média de 10.000 pessoas que completam 65 anos todos os dias (WITTERMYER; JASON TOWNS, 2020). Além disso, há estudos que informam que a contribuição econômica de americanos acima de 50 anos, foi de 8,3 trilhões de dólares em 2018 e ainda se estima que essa contribuição aumentará para 28,2 trilhões em 2050, ao levar em consideração que os millenials (nascidos em 1981 a 1996) e a geração Z (nascidos em 1997 a 2012) completarão 50 anos a partir de 2031 e de 2047, respectivamente (AARP, 2019; KENNETH TERRELL, 2019).

Essa mudança demográfica colabora para o maior do consumo de medicamentos, uma vez que o avanço da idade está relacionado ao aumento da prevalência de doenças crônicas não transmissíveis, o que também expõe a população idosa a múltiplos medicamentos (AIOLFI et al., 2015). Segundo Maahs e Serrate (2005, p.925) “há evidências de que os idosos americanos, vivendo na comunidade, tomam em média de 2,7 a 4,2 medicamentos prescritos ou não prescritos”.

Diante de todas os pontos explorados, ao considerar o público principal que tal produto está destinado, o contexto de transição demográfica que o mundo se encontra e as perspectivas econômicas em relação a esse grupo, pode-se afirmar que o produto em questão se insere em um contexto favorável e promissor.

4.2 Cliente

A partir da pesquisa da literatura, pode-se observar que os idosos – ou seja, pessoas com 60 anos ou mais (MINISTÉRIO DA SAÚDE, 2013) – seriam o principal público-alvo do produto, visto que essa parcela da população é a que mais consome múltiplos medicamentos por um grande tempo. Contudo, de modo a complementar o estudo até então, foi aplicada uma pesquisa *online*, já abordada no capítulo anterior, com o objetivo de conhecer mais sobre o mercado em potencial e descobrir novas oportunidades. A união dessas informações pode trazer maior precisão na hora de definir as principais características dos consumidores do presente produto e, por consequência, também definir aspectos técnicos importantes do produto em si, os quais não são suficientemente definidos pela pesquisa na literatura.

Dos 169 respondentes, 122 afirmaram consumir medicamentos diariamente. Diante disso, as análises a partir de então serão focadas para melhor entendimento desse grupo, o qual apresenta maior potencial de investimento no porta remédios. Isso se dá ao fato que a premissa pétrea do produto desenvolvido é ajudar pessoas que apresentam esse padrão de consumo.

Tabela 1 – Respostas referentes ao tipo de medicamento

Tipo de medicamento	Respostas
Comprimido	111
Cápsula	63
Líquido	12
Efervescente	2
Outros	4

Fonte: Elaboração própria (2020).

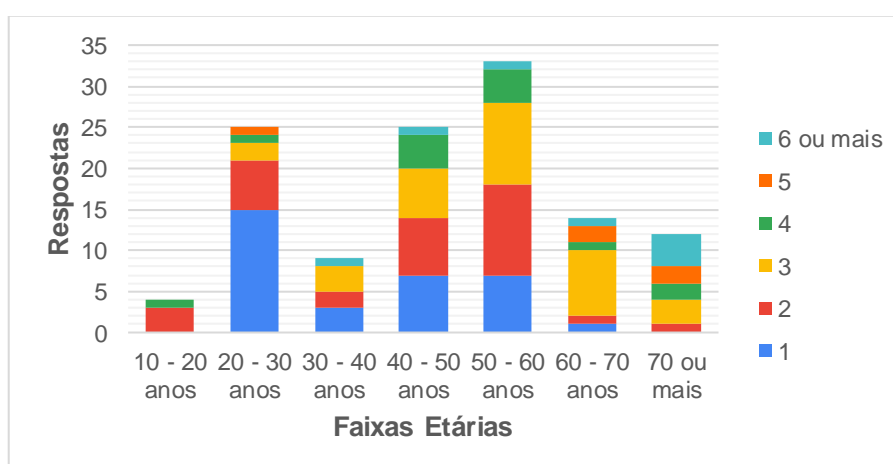
Tabela 2 – Média de respostas para questões 4, 5 e 13 do formulário

Média das questões 4, 5 e 13	Resultado
Média do número vezes ao dia que se está tomando algum medicamento (vezes)	1,7
Média do número comprimidos ou capsulas consumidos diariamente (unidade)	3,0
Média de investimento no produto (R\$)	109,02

Fonte: Elaboração própria (2020).

A partir da observação da Tabela 1, devido à grande incidência de consumo de comprimidos e cápsulas em relação aos demais, pode-se definir que o produto em questão deve ser voltado para um público que consome esses tipos de medicamento. Além disso, conforme visto nos resultados da Tabela 2, é esperada uma média de 3 remédios diários e uma frequência de 1,7 vezes ao dia de consumo de algum medicamento. De modo complementar as informações anteriores, pode-se observar na Figura 6 que a incidência de 3 ou mais medicamentos começa a crescer a partir da faixa etária de “40 - 50 anos”.

Figura 6 – Número de medicamentos consumidos por Faixa Etária



Fonte: Elaboração própria (2020).

Sobre a média de investimento, é importante destacar que não será foco do presente trabalho de conclusão de curso atender essa demanda. É objetivo deste projeto montar um produto mínimo viável de modo a apenas validar a solução, sem necessariamente respeitar os requisitos de preço. Tal informação foi recolhida para se ter uma base em caso de trabalhos e melhorias futuras do projeto.

Tabela 3 – Métodos utilizados para lembrar a ingestão de remédios

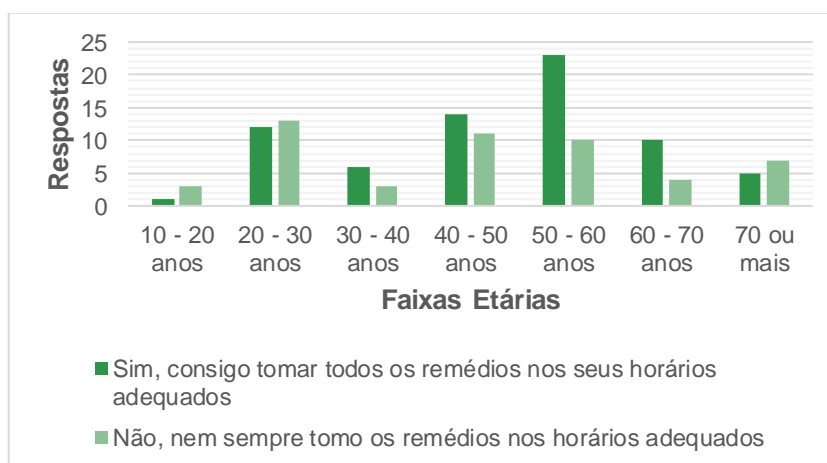
Método	Respostas (Geral)	Respostas (Idosos)
Memória/força do hábito/Nenhum	81	15
Despertador de celular	21	4
Porta remédios	16	7
Aplicativo	9	0
Um familiar/amigo/profissional administra os horários dos meus medicamentos	5	2
Lembretes em papel	3	2
Outros	7	3

Fonte: Elaboração própria (2020).

Na pesquisa também se buscou verificar quais métodos são utilizados atualmente pelas pessoas para lembrar a ingestão de medicamentos, conforme visto na Tabela 3. Vale ressaltar que na terceira aba da tabela foram considerados idosos

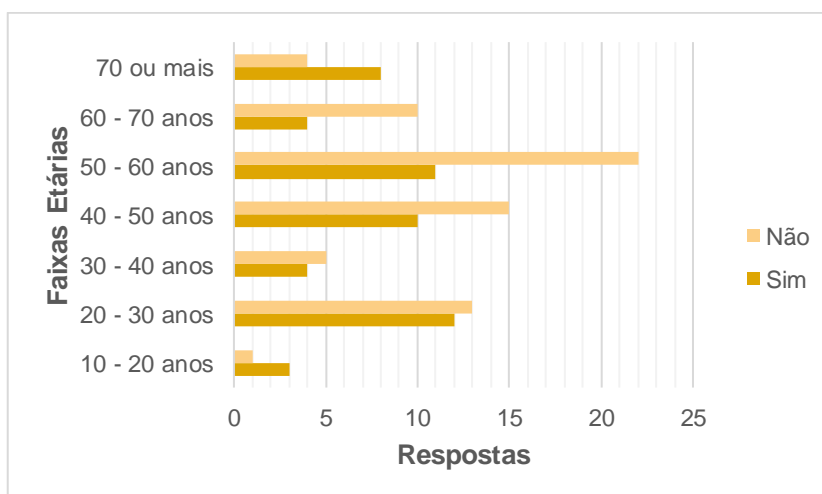
todos que assinalaram “60 – 70 anos” e “70 ou mais”, de acordo com o definido no estatuto do idoso (MINISTÉRIO DA SAÚDE, 2013). Além disso, o grupo de respostas “Geral” inclui todos os respondentes que consomem medicamentos diariamente. Na tabela observa-se que o método “Memória/força do hábito/Nenhum” é o mais frequente. Em ambos os grupos, os métodos “Porta remédios” e “Despertador de celular” também ganham destaque, o que é bom considerando a natureza do produto desenvolvido, uma vez que o porta remédios apresentará características desses dois recursos.

Figura 7 – Faixa Etária e Pontualidade na ingestão de medicamentos



Fonte: Elaboração própria (2020).

Figura 8 – Dúvida referente a ingestão por faixa etária



Fonte: Elaboração própria (2020).

Em paralelo a Tabela 3, é interessante verificar a Figura 7 e a Figura 8, nas quais se observa que as únicas faixas etárias que apresentam uma menor pontualidade e uma maior dúvida referente a ingestão do medicamento é a de “70

anos ou mais” e a de “10 – 20 anos”. Contudo, por ter sido apenas 04 respondentes de 10 - 20 anos, é difícil realizar uma conclusão assertiva sobre os comportamentos de consumo apresentados. Dessa maneira, considerando que o porta medicamentos buscará auxiliar o usuário nesses dois pontos comentados, a faixa etária de 70 anos ou mais deve ser considerada como foco principal do produto desenvolvido.

Em conclusão, diante de todas as informações estudadas até então, pode-se definir como público-alvo: homem/mulher a partir de 70 anos, com um padrão de consumo médio de 3 medicamentos na forma de comprimido e de capsula, distribuídos em média 2 horários no dia. Tal público-alvo apresenta dificuldades em relação a pontualidade de ingestão e a confirmação de ingestão do medicamento, problemas que para resolução ele está disposto a investir uma média de R\$109,02.

4.3 Pesquisa de concorrentes

O propósito desta busca é verificar os principais produtos existente no mercado na linha do projeto em desenvolvimento. A partir disso, será possível observar características cruciais para o produto em desenvolvimento, assim como diferenciais que podem ser implementados em relação aos equipamentos pesquisados.

4.3.1 Organizadores de pílulas

Os organizadores de pílulas, também chamados de porta medicamentos, consistem em uma caixa separadora, na maioria das vezes, com alguma indicação temporal, como dia da semana, mês, turno do dia, entre outras opções. Esses tipos de porta comprimidos podem ser encontrados de diferentes tamanhos e formas. Exemplos de porta remédios podem ser encontrados na Figura 9 e Figura 10.

Figura 9 – Exemplo de organizador de pílulas



Fonte: Magazine Luiza (2021)

Tais caixas não apresentam tecnologias para avisar ou para acompanhar a ingestão dos medicamento ingeridos. Além disso, por ser encontrado também de diferentes tamanhos, formas e qualidades sua faixa de preço também é variada. Na Tabela 4, pode-se encontrar outras informações pertinentes sobre o produto da Figura 10. Em resumo, pode-se destacar como vantagens desses produtos a faixa variada de preços e a facilidade de uso.

Figura 10 – Organizador de pílulas Océane



Fonte: Loja Océane (2020)

Tabela 4 – Resumo informações organizador de pílulas Océane

Dimensões do produto	1 x 24 x 21,5 cm - 0,21g
Número de compartimentos	28
Alimentado por pilha ou bateria?	Não
Conectividade	Nenhuma
Aplicativo ou site?	Não
Preço	R\$ 74,90

Fonte: Loja Océane (2020)

4.3.2 LiveFine – Dispensador Automático de Medicamentos

Este porta remédios é uma solução intermediária em termos de tecnologia. O dispensador da LiveFine (Figura 11) permite programar até 6 alarmes por dia, os quais podem ser configurados por meio dos três botões centrais. No horário determinado, a caixa irá girar automaticamente, permitindo o acesso apenas do

compartimento onde está localizado o medicamento, além de emitir um alarme sonoro e luminoso (led vermelho).

Figura 11 – Dispensador de medicamentos Automático LiveFine



Fonte: Amazon (2021)

Tabela 5 – Resumo informações dispensador automático LiveFine

Dimensões do produto	20.07 x 9.91 x 20.07 cm - 340,19g
Número de compartimentos	28
Alimentado por pilha ou bateria?	Sim, pilha
Conectividade	Nenhuma
Aplicativo ou site?	Não
Preço	U\$ 99.99 (~R\$ 522,94) Cotação utilizada: 1 U\$ vale R\$ 5,23

Fonte: Amazon (2021)

É importante destacar que a presente solução não envia avisos para familiares ou registra se o remédio foi ingerido, uma vez que conforme observado na Tabela 5, na qual se encontra um resumo dos principais pontos a serem vistos no produto, não existe conectividade via *bluetooth* ou via internet nesse dispositivo, além da ausência de aplicativo ou site. Ainda na Tabela 5, é possível verificar outros pontos importantes, como as dimensões, o número de compartimentos disponíveis e tipo de alimentação apresentada.

Por fim, outra funcionalidade interessante desse produto é que o usuário ou o responsável pela caixa pode trancar a dispensador de medicamentos por meio de uma chave, o que bloqueia o acesso a todos os comprimidos, exceto do compartimento que se encontra os remédios a serem ingeridos em determinado horário.

4.3.3 Pillgo

Pillgo é um porta medicamentos mais avançado tecnologicamente de todos explorados na presente pesquisa. Em termos de *software*, esse equipamento apresenta um aplicativo que monitora se o paciente consumiu o medicamento e apresenta relatórios mensais, diários e anuais. Há também a opção de avisar familiares caso o paciente não tome o medicamento. Além disso, monitora e registra sinais vitais do usuário, permite tirar foto da etiqueta do medicamento para extração das informações do medicamento prescrito, entre outras funções. O aplicativo também gera notificações em caso de os remédios cadastrados apresentem risco de interação medicamentosa ou reação alérgica.

Figura 12 – Pillgo



Fonte: Pillgo (2020)

O *hardware* apresenta iluminação lateral, a qual serve para alertar o momento de consumo do medicamento. Além disso, o acesso ao medicamento é feito pelo giro manual da caixa, permitindo o acesso de um compartimento por vez. Caso usuário gire para o lado incorreto, a iluminação alerta que aquele não é o sentido correto para acessar o medicamento. Por fim, o dispositivo é portátil, apresentando bateria recarregável. De modo complementar as informações já citadas, pode-se

observar na Tabela 6 um resumo com os principais pontos a serem observados no produto em questão.

Tabela 6 – Resumo de informações Pillgo

Dimensões do produto	Diâmetro: 93 mm / Altura: 32,6mm / 200g
Número de compartimentos	7
Alimentado por pilha ou bateria?	Sim, bateria
Conectividade	Bluetooth 4.0 Low Energy
Aplicativo ou site?	Sim, aplicativo
Preço	HK\$199 (~R\$ 140,00) Cotação utilizada: 1 HK\$ vale R\$ 0,68

Fonte: Pillgo (2020)

5 PROPOSTA DE SOLUÇÃO

Neste capítulo será definido os principais pontos referentes a solução de problema, tais quais: objetivo geral do produto, requisitos, soluções, especificações, entre outros.

5.1 Objetivos da proposta de solução

Diante do observado no estudo de mercado, com o crescimento contínuo da população idosa no Brasil e no mundo, é de extrema importância o desenvolvimento de tecnologias voltadas ou aplicáveis a esse grupo, de modo a auxiliar as famílias, cuidadores e casas de repouso, além de, se possível, promover uma maior independência do idoso.

Nesse sentido, o porta comprimidos inteligente apresenta o propósito de colaborar com o aumento da aderência de pacientes, na sua maioria com 70 anos ou mais, que consomem diversos medicamentos continuamente ou por um longo tempo determinado. Caso se mostre funcional, a utilização desse equipamento pode ocasionar resultados positivos no tratamento das comorbidades enfrentadas, uma vez o cumprimento do esquema posológico colabora diretamente na eficácia terapêutica da medicação (SOUSA *et al.*, 2011).

Além disso, é objetivo do produto a ser desenvolvido tornar visível aos profissionais da saúde todos os remédios consumidos pelo paciente, ação que pode ajudar na detecção de interações medicamentosas e, portanto, melhoramento das

abordagens de tratamento realizadas. Vale destacar, também, que por meio do porta remédio inteligente se espera conseguir averiguar se o usuário está ingerindo os medicamentos nos seus horários adequados, tornando visível essa informação para o próprio idoso, família, médicos e farmacêuticos. Com isso, espera-se permitir aos idosos que moram sozinhos um acompanhamento a distância.

Com a pesquisa de mercado foi possível verificar o cenário no qual o produto em questão se encontra, definição do público-alvo e visualização das principais funcionalidades oferecidas por produtos similares. Pode-se observar que esses objetivos explorados estão alinhados principalmente pela proposta apresentada pela empresa Pillgo.

Por fim, diante dos objetivos apresentados pelo MVP em questão, foi escolhido o nome de “Meepi” para proposta de solução em desenvolvimento.

5.2 Requisitos do Produto Mínimo Viável (MVP)

A partir de todos os estudos realizados até então, pode-se definir como requisitos do presente produto:

- **Uso simplificado:** o MVP deve ser fácil de usar para todos os públicos, em especial para o grupo de destaque obtido na pesquisa de mercado. Dessa maneira, o usuário não deve ter dificuldades para programar os horários dos medicamentos, assim como os ativar/desativar outras funcionalidades disponíveis;
- **Acessibilidade:** De acordo dados reunidos pela Organização Brasileira de Oftalmologia (OTTAIANO et al., 2019), existem cerca de 216,6 milhões de pessoas no mundo com deficiência visual moderada ou grave, 185,5 milhões de pessoas com comprometimento visual leve e 36 milhões de pessoas cegas, das quais 82% apresentam mais de 50 anos. Além disso, segundo estimativas da Organização Mundial da Saúde (2018), existem 466 milhões de pessoas no mundo com problemas de audição, com destaque para a estatística de que um terço das pessoas acima de 65 anos são afetadas pela perda da audição. Diante dos dados apresentados e do público a qual o produto se destina, é importante

que tal MVP apresente estratégias que permita o acesso de pessoas com visão e/ou audição comprometida;

- **Transparência:** o porta remédio deve permitir a visualização dos dados pelas partes interessadas, logo apresentar as informações pretendidas de maneira clara e objetiva, de modo a ajudar o usuário a ingerir os medicamentos adequadamente;
- **Conectividade:** os dados de ingestão dos medicamentos e horários devem estar disponíveis em tempo real para as partes interessadas;
- **Portabilidade:** é interessante que seja possível se locomover com o dispositivo. Logo, de maneira ideal, ele deve apresentar dimensões e uma eletrônica adequada para tal.

5.3 Funcionalidades do MVP

De acordo com o definido como objetivos da solução, o produto em questão apresentará como funcionalidades:

- Armazenar medicamentos;
- Notificar de maneira sinestésica (visual e sonora) o momento de ingestão do(s) medicamento(s);
- Informar para o usuário a localização no porta medicamentos do remédio a ser ingerido;
- Permitir a programação dos horários do medicamentos pelo usuário e/ou partes interessadas;
- Detectar se o usuário ingeriu o medicamento ou acessou a caixa;
- Notificar terceiros sobre a ingestão do(s) medicamento(s).

5.4 Análise de soluções tecnológicas

Neste tópico são detalhadas as possibilidades de soluções tecnológicas para cada funcionalidade definida para o porta comprimidos. Para melhor ilustrar essas informações, foram feitos quadros contendo a funcionalidade a que se remetem, os nomes das soluções e imagens para exemplificação de cada uma.

O primeiro ponto a se analisar é referente a funcionalidade de armazenamento de medicamentos, presente no Quadro 1. Sabe-se que existem diversas formas possíveis para guardar e permitir o acesso aos remédios na caixa, contudo foram listadas as principais alternativas observadas no mercado, no caso a

opção de caixa rotatória manual, de caixa retangular e de caixa rotatória automática. É importante destacar que todas as propostas são de caixas voltadas para o armazenamento de remédios sólidos (comprimidos e capsulas), uma vez que na pesquisa de mercado foi mostrado uma incidência maior no consumo desse tipo de medicamento.

A principal referência de caixa rotatória manual encontrada no Quadro 1 é a do produto Pillgo. A vantagem dessa configuração é que o usuário acessa apenas um compartimento por vez, não expondo por muito tempo todos os medicamentos ao ar. Essa característica é também encontrada na opção 3, dispensador de medicamentos LiveFine, na qual a rotação é feita de maneira automática. Ao colocar um giro automático, necessita-se acrescentar um motor, o que adiciona mais um componente ao projeto e, portanto, gera um impacto no custo do MVP.

Quadro 1 – Opções para armazenamento de medicamento

Funcionalidade	Opção 01	Opção 02	Opção 03
Armazenamento de medicamento	 <p>Caixa rotatória manual</p>	 <p>Caixa retangular</p>	 <p>Caixa rotatória automática</p>

Fonte: Elaboração própria (2021).

Em relação a opção 02 para funcionalidade de armazenamento de medicamentos (Quadro 1), a caixa retangular é que o formato retangular pode facilitar no momento da organização dos componentes eletrônicos. No entanto, pode-se citar como desvantagem a questão que ao abrir a caixa todos os remédios estarão expostos ao ar. Além disso, é mais complexo implantar um monitoramento de acesso dos compartimentos, o que pode ser facilmente realizado com uma caixa do tipo rotatória.

No Quadro 2, é possível verificar as opções de soluções para a funcionalidade de notificar o momento de ingestão do medicamento de maneira visual e de maneira sonora. É importante que ocorra a notificação de maneira sinestésica, de modo que, caso a caixa não esteja no campo de visão do usuário, seja possível saber o momento de ingerir a medicação. Além disso, apresentar essas duas formas também colabora para acessibilidade do produto.

Na opção 01 referente ao alerta visual presente no Quadro 2, a fita de LEDs lateral se mostra uma solução interessante, uma vez que esse componente é maleável, ou seja, pode ser compatível com diversos formatos de caixa. Além disso, tal equipamento permite uma iluminação que é possível ser vista de diferentes ângulos. Já a iluminação por LED unitário, opção 02, apesar da maior facilidade de implementação, não é tão visível de diferentes ângulos. Por fim, a opção 03 apresenta a desvantagem de ter tamanho e forma definida, o que restringe no momento da modelagem da caixa. Em termos de implementação eletrônica é semelhante a fita de LEDs.

Quadro 2 – Opções para notificar momento da ingestão

Funcionalidade	Opção 01	Opção 02	Opção 03
Notificar momento de ingestão de medicamento (alerta visual)	 <p>Iluminação lateral por fita de LEDs</p>	 <p>Iluminação por LED unitário</p>	 <p>Iluminação do anel de LEDs</p>
Notificar momento de ingestão de medicamento (alerta sonoro)	 <p>Buzzer</p>	 <p>Motor de Vibração 1027</p>	 <p>Alto-falante</p>

Fonte: Elaboração própria (2021).

Sobre as opções para funcionalidade de notificar momento de ingestão do medicamento por alerta sonoro, dentre as opções mostradas no Quadro 2, o motor de vibração se mostra como solução interessante em termos de acessibilidade, visto que a vibração, além de gerar um pequeno som, adiciona mais um sentido no momento da notificação: o tato. Contudo, como o som gerado pela vibração não é tão alto, torna-se válido o uso de um outro dispositivo complementar. Para isso, o *buzzer*, opção 01, se torna interessante pela facilidade de implementação e seu baixo custo.

No Quadro 3, que engloba as alternativas para a funcionalidade de informar para o usuário a localização no porta medicamentos do remédio a ser ingerido, foram listadas como propostas a opção de LCD, a de indicação luminosa em cada compartimento e a de assistente de voz. Considerando o requisito de acessibilidade

do produto, é importante que a informação sobre o remédio a ser ingerido chegue de maneira visual e sonora. Dessa forma, a aplicação da opção 03 se apresenta como solução para informar o paciente de maneira sonora. No entanto, para implementar um assistente de voz, há necessidade de maior complexidade em termos de *software* e de *hardware*, portanto não entrará no escopo do projeto no momento.




Quadro 3 – Opções para informar a localização do remédio a ser ingerido

Funcionalidade	Opção 01	Opção 02	Opção 03
Informar para o usuário a localização no porta medicamentos do remédio a ser ingerido	 LCD	 Indicação luminosa em cada compartimento	 Assistente de voz

Fonte: Elaboração própria (2021).

Em relação às alternativas de notificação de modo visual presentes no Quadro 3, no caso a de indicação luminosa em cada compartimento e a de assistente de voz, o principal diferencial em relação a opção 01 e a opção 02, é que com o uso do LCD seria possível escrever na tela o nome do medicamento a ser ingerido. Já no caso da alternativa de indicação luminosa, só estaria indicado o compartimento onde aquele remédio se encontra. É importante destacar que dentre as três alternativas de solução citadas, a indicação luminosa em cada compartimento seria a com menor custo de implementação.

Quadro 4 – Opções para permitir a programação dos horários

Funcionalidade	Opção 01	Opção 02	Opção 03
Permitir a programação dos horários do medicamentos pelo usuário e/ou partes interessadas	 Site	 Aplicativo próprio	 Botões na própria caixa

Fonte: Elaboração própria (2021).

A respeito das maneiras de permitir a programação dos horários pelo usuário e/ou partes interessadas, funcionalidade abordada no Quadro 4, a opção ideal seria via aplicativo próprio. No entanto, devido ao tempo dedicado para o desenvolvimento do TCC e ao foco existente na parte do *hardware*, o desenvolvimento de um aplicativo dedicado não entrará no escopo do protótipo. A utilização de botões

na própria caixa, opção 03 do Quadro 4, apresenta a vantagem de não ser necessário usar um intermediário para programar os horários, como um celular ou computador. Todavia, adiciona pelo menos três botões ao projeto eletrônico e exige a implantação de um LCD em conjunto.

Para a funcionalidade de detectar se o usuário ingeriu o medicamento (Quadro 5), apresentou-se as seguintes opções: fim de curso mecânico, LDR, *Encoder* rotativo e Botão. A opção 04 se mostra como a mais simples das soluções listadas, uma vez o usuário apenas pressionaria um botão na caixa para indicar que já ingeriu o medicamento. No entanto, só o botão pode não ser muito efetivo, visto que a pessoa pode clicar no botão para desativar o alarme, mas não tomar o remédio naquele momento.

Quadro 5 – Opções para detectar se o usuário ingeriu o medicamento

Funcionalidade	Opção 01	Opção 02	Opção 03	Opção 04
Detectar se o usuário ingeriu o medicamento ou acessou a caixa	 Fim de curso mecânico	 LDR	 Encoder Rotativo	 Botão

Fonte: Elaboração própria (2021).

Por conta disso, pode-se utilizar um monitoramento em paralelo, de modo a fazer uma dupla validação. No caso de uma caixa rotativa, o *encoder*, ou seja, a opção 03 do Quadro 5, é a solução mais adequada dentre as listadas, por meio do qual seria possível verificar se o usuário acessou determinado compartimento. Em uma caixa retangular, o fim de curso mecânico pode ser instalado para detectar a abertura e o fechamento da caixa. De modo semelhante funcionaria o LDR.

Quadro 6 – Opções para notificação de terceiros

Funcionalidade	Opção 01	Opção 02	Opção 03
Notificar terceiros sobre a ingestão do(s) medicamento(s)	 Via Telegram	 Via aplicativo próprio	 Via e-mail

Fonte: Elaboração própria (2021).

Para a funcionalidade de enviar notificações a terceiros presente no Quadro 6, apresenta-se como soluções viáveis e não excludentes notificações via e-mail e via *Telegram*¹. Contudo, por estar se visando a construção de um MVP e, portanto, não apresentar necessidade do e-mail para validar a ideia, prioriza-se a aplicação como o *Telegram*. Já a opção 02 se pode descartar devido a justificativas anteriormente apresentadas. Além disso, para o usuário do porta medicamentos e para terceiros cadastrados, por meio do *Telegram* é possível enviar notificações para ingestão do remédio no horário adequado, o que dispensa no momento a necessidade de um aplicativo próprio para realizar a mesma função.

A partir de todas as considerações realizadas, foi montada a Tabela 7 com a síntese de todas as soluções escolhidas para cada funcionalidade.

Tabela 7 – Lista de soluções para o protótipo

Funcionalidade	Solução
Armazenamento de medicamentos	Caixa rotatória
Informar para o usuário a localização no porta medicamentos do remédio a ser ingerido	Indicação luminosa em cada compartimento
Notificar momento de ingestão de medicamento (alerta visual)	Fita de LED
Notificar momento de ingestão de medicamento (alerta sonoro)	Buzzer e motor de vibração
Permitir a programação dos horários do medicamentos pelo usuário e/ou partes interessadas	Site
Detectar se o usuário ingeriu o medicamento ou acessou a caixa	Botão e encoder
Notificar terceiros sobre a ingestão dos medicamentos	Telegram

Fonte: Elaboração própria (2021).

A partir da análise dos prós e contras, decidiu-se pela utilização de uma caixa rotatória, a qual implica na escolha o *encoder* para detectar se o usuário ingeriu o medicamento ou acessou a caixa de fato. Com esse sensor seria possível verificar se o usuário girou a caixa até chegar no compartimento onde o medicamento a ser ingerido naquele momento se encontra.

Ao realizar a escolha por uma caixa rotatória, a opção de utilização do LCD, exigiria um grande espaço na caixa destinado para tal, o que impactaria no tamanho do MVP. Considerando também que ocorrerá notificação via celular por meio do

¹ *Telegram* é um serviço de troca de mensagens instantâneas. Esse aplicativo está disponível para celular, *tablet* e computador.

Telegram e que haverá um dashboard disponível para usuário onde ele poderá consultar os remédios pelo celular ou pelo computador, o LCD se mostra como redundante. Vale ressaltar também que na Tabela 3 – Métodos utilizados para lembrar a ingestão de remédios, boa parte do grupo dos idosos assinalou que utiliza “Memória/Força do hábito/Nenhum” como método, ou seja, aqueles que ingerem medicamentos diariamente já sabem qual remédio deve ser ingerido em cada horário. Diante disso, foi escolhida a opção de indicação luminosa em cada compartimento, a qual se mostra mais simples de implantar no quesito eletrônico, exige pouco espaço e com menor custo.

6 DESENVOLVIMENTO DO PROTÓTIPO

No presente capítulo se encontra o detalhamento técnico referente ao desenvolvimento eletrônico, mecânico e relacionado a programação do produto em questão.

6.1 Desenvolvimento eletrônico

Com base nas decisões tomadas na etapa anterior do trabalho em questão, partiu-se para etapa de desenvolvimento eletrônico do MVP. Diante disso, foram realizados os seguintes passos:

1. Análise de opções e seleção do embarcado para controle no MVP;
2. Montagem da lista de componentes e busca de folha de dados;
3. Testes com cada item adquirido individualmente;
4. Associação dos componentes do projeto com as portas adequadas do embarcado;
5. Construção do esquemático e do layout da placa;
6. Fabricação da PCI (Placa de Circuito Impresso) e soldagem de componentes;
7. Testes com a placa fabricada;
8. Realização de correções na PCI, se necessário.

Nos tópicos seguintes serão descritas as etapas citadas, com exceção das etapas 3, 4, 7 e 8.

6.1.1 Escolha de *hardware* central

A primeira etapa realizada no desenvolvimento eletrônico do MVP foi a análise de qual embarcado seria utilizado para controlar os sensores e os botões do projeto em questão. Apenas com essa decisão seria possível avançar para as etapas subsequentes, visto que a escolha de componentes também está relacionada às características apresentadas por esse elemento central, como a tensão de operação.

Dentre as opções de embarcados existentes no mercado para executar tais funções, somente entraram na análise o Arduino Uno, a ESP32 e a Raspberry Pi Zero W, visto que esses dispositivos eram apresentados pelo IFSC e pela graduanda. Além disso, foram definidos como critérios de seleção o custo de cada equipamento e a compatibilidade com os requisitos do protótipo.

Figura 13 – Arduino UNO R3



Fonte: Arduino.cc (2021).

O Arduino Uno (Figura 13) é uma placa baseada no microcontrolador ATmega328P. Tal componente apresenta tensão de operação de 5V, tensão de entrada recomendada de 7-12V, além de 14 pinos com entradas/saídas digitais (GPIO, *General Purpose Input/Output*), dos quais 6 podem apresentar saídas com modulação por largura de pulso, e de 6 pinos analógicos (ARDUINO.CC, 2021). De acordo com informações no site da Arduino.cc (2021), todas essas portas estão distribuídas em uma placa com 68,6 mm de comprimento e 53,4 mm de largura. Por ser um equipamento de código aberto, existe uma grande documentação associada a esse dispositivo.

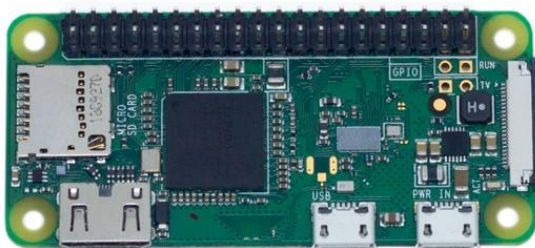
A segunda opção destacada é a ESP-WROOM-32 ou também chamada de apenas ESP32 (Figura 14). Essa placa microcontrolada apresenta chip híbrido de *wifi* e *bluetooth*, um baixo consumo de energia e um design robusto, ou seja, capaz de funcionar de maneira confiável em ambientes industriais e em uma ampla faixa de temperaturas (ESPRESSIF SYSTEMS, 2020). Esse equipamento apresenta 30 pinos, os quais muitos apresentam mais de uma função. De modo geral, apresenta 25 pinos de GPIOs e 18 pinos com conversores analógicos-digitais (CIRCUITS4YOU, 2018). É interessante destacar que a ESP32 apresenta a possibilidade de utilização de *Pulse Width Modulation* (PWM) ou modulação por largura de pulso em todos os seus pinos de saída. Outra característica é que esse placa opera com tensão de 3,3V e pode ser alimentada com até 9V.

Figura 14 – ESP32



Fonte: FILIPEFLOP (2021a).

Figura 15 – Raspberry Pi Zero W



Fonte: FILIPEFLOP (2021b).

Por fim, a terceira alternativa apresentada é a *Raspberry Pi Zero W*, a qual pode ser observada na Figura 15. Esse dispositivo possui *wifi-fi* e *Bluetooth*

integrados, característica também apresentada pela ESP32. Além disso, é composta por um processador Broadcom BCM2835 single-core de 1 GHz e apresenta 512 MB de memória RAM (FILIPEFLOP, 2021b). Essas características a permitem rodar distribuições do Linux, como o Raspbian e o Ubuntu. Outras características importantes são que a *Raspberry Pi Zero* apresenta 40 GPIOs, deve ser alimentada por uma fonte de 5V e seus pinos operam com tensão de 3,3V.

Tabela 8 – Resumo referente a preço, a dimensões e a conectividade

Equipamento		Preço	Dimensões	Conectividade
Arduino Uno	R\$	72,90	68,6 x 53,4 mm	Não
ESP32	R\$	79,90	52 x 28 x 5 mm	Sim
Raspberry Pi Zero	R\$	179,90	65 x 30 x 10 mm	Sim

Fonte: FILIPEFLOP (2021c, 2021b, 2021a)

De modo a complementar as informações já exploradas, montou-se a Tabela 8 com informações referentes ao preço de cada placa analisada, as dimensões e a conectividade. Essas informações foram destacadas para auxiliar na escolha desse elemento central para o projeto, a qual levou em conta não só as características técnicas, como também o impacto no custo do protótipo e os requisitos de portabilidade e conectividade, definidos anteriormente.

É importante destacar que o protótipo em questão será uma solução dentro do contexto de *Internet of Things* (IoT) ou Internet das coisas em português. Esse conceito se refere a tendência atual de conectar dispositivos a internet (RED HAT, 2021), os quais podem ser de artefatos industriais a itens menos comuns, como um porta medicamentos.

Ao verificar a primeira linha da Tabela 8, pode-se concluir que o Arduino Uno é descartado por não apresentar chip de *wi-fi* integrado. O porta comprimidos receberá informações vindas pela rede sem fio e enviará informações por esse meio, logo é imprescindível ter essa funcionalidade. No mercado existe opções de módulos e de *shields wi-fi* para utilização com o Arduino, porém, isso geraria um impacto no custo do MVP.

Já na terceira linha da Tabela 8, vê-se as características referentes a *Raspberry Pi Zero W*. Um ponto positivo é que essa placa apresenta *wi-fi* integrado, ao contrário do Arduino Uno. Vale destacar também que dentre as três opções, é esse dispositivo que tem as características técnicas mais avançadas. Contudo, diante das outras duas alternativas, esse equipamento apresenta o maior custo e as maiores

dimensões e, portanto, também é desconsiderado como escolha. É importante ressaltar que apesar das características técnicas de destaque, o protótipo não apresentará um grau de complexidade tão elevado, de modo que a *Raspberry Pi Zero* poderia ser subaproveitada.

Por fim, na segunda linha da Tabela 8, observa-se as características de preço, de dimensões e de conectividade da ESP32. Considerando que esse dispositivo já apresenta a possibilidade de conexão sem fio, a ESP2 se mostra como melhor opção relacionada ao preço, uma vez que não seria necessária a compra de um módulo adicional para apresentar essa função. Além disso, tal placa possui as menores dimensões em comparação com as outras, o que impacta no requisito de portabilidade, uma vez que uma placa pequena implica um projeto eletrônico menor e que por fim gera um projeto de caixa com dimensões mais reduzidas, permitindo, dessa maneira, ser comportada mais facilmente em bolsas, em maletas, entre outros.

Diante de todos os argumentos explorados, a ESP32 foi escolhida como solução para controlar o sistema a ser desenvolvido.

6.1.2 Projeto eletrônico

Tabela 9 – Componentes eletrônicos principais para o projeto

N.º	Componente	Funcionalidade
1	Bateria de íon-Lítio 18650 2600mAh	Alimentação do circuito
2	Buzzer Ativo 5V	Alerta sonoro
3	Chave <i>Push Button</i> N/A PSB-11B com 2 Terminais	Validação de consumo do medicamento
4	Conversor <i>Boost</i> DC Ajustável <i>Step Up</i>	Fornecimento da tensão adequada para o circuito
5	Encoder Decoder KY-040 Rotacional	Validação de consumo do medicamento
6	Fita de LED RGB WS2812 5050	Alerta visual
7	LED RGB WS2812B 5050 Endereçável	Indicação visual do compartimento
8	Módulo Carregador de Bateria de Lítio TP4056	Gerenciamento de carga da bateria
9	Motor de Vibração 1027	Alerta sonoro
10	<i>Real Time Clock</i> (RTC) DS1307	Monitoramento dos horários

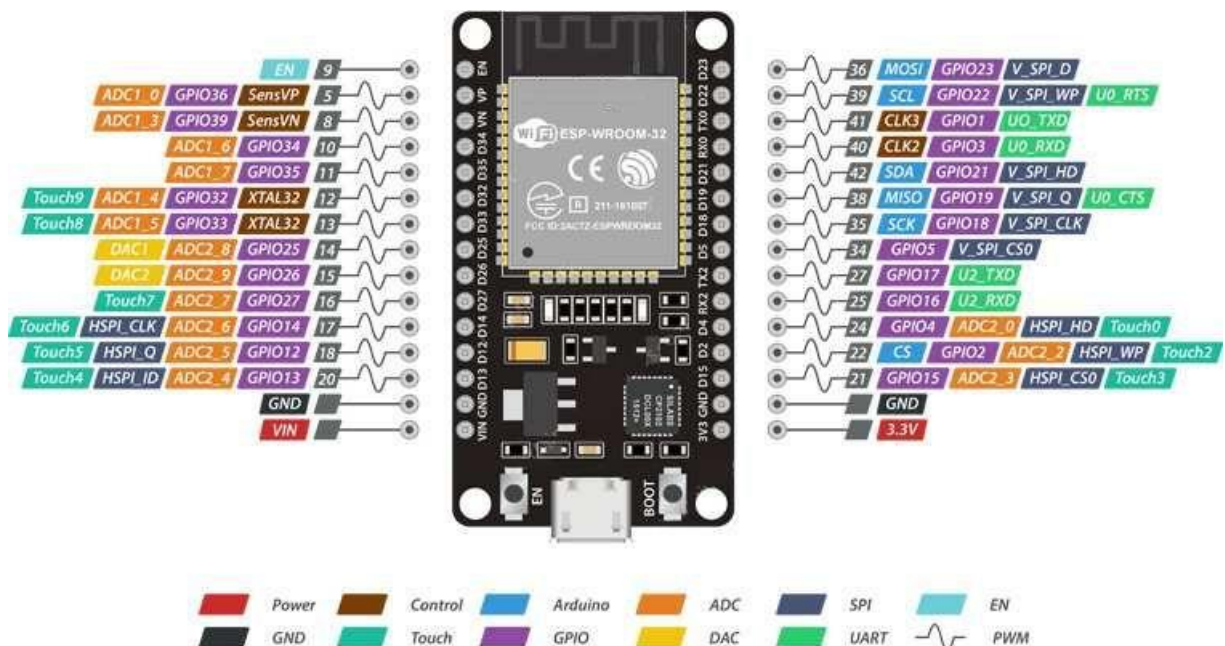
Fonte: Elaboração Própria (2021)

Com a escolha do embarcado principal, partiu-se para especificação dos componentes, cuja maioria já havia sido previamente escolhida na análise de soluções do capítulo anterior. Diante disso, procurou-se selecionar itens eletrônicos comerciais de baixo custo compatíveis com a ESP32. Na Tabela 9, é possível verificar os componentes a serem utilizados no projeto e a suas respectivas funções. É importante destacar que ao decorrer do desenvolvimento eletrônico certamente serão utilizados

resistores, barras de pinos, de *t-blocks*, transistores, entre outros. Contudo, esses tipos de elementos não serão considerados na Tabela 9, mas serão posteriormente especificados.

Após a escolha dos componentes presentes na Tabela 9, foi necessário a verificação das funções apresentadas por cada pino da ESP32, as quais podem ser observadas na Figura 16. A partir dessa atividade, foi realizada a associação de cada elemento com as portas da ESP32, levando-se em conta, também, as exigências de cada componente escolhido, se existente. Após essa etapa, foi realizada a montagem do esquemático no software adequado para tal.

Figura 16 – Pinagem da ESP32



ESP32 Dev. Board Pinout

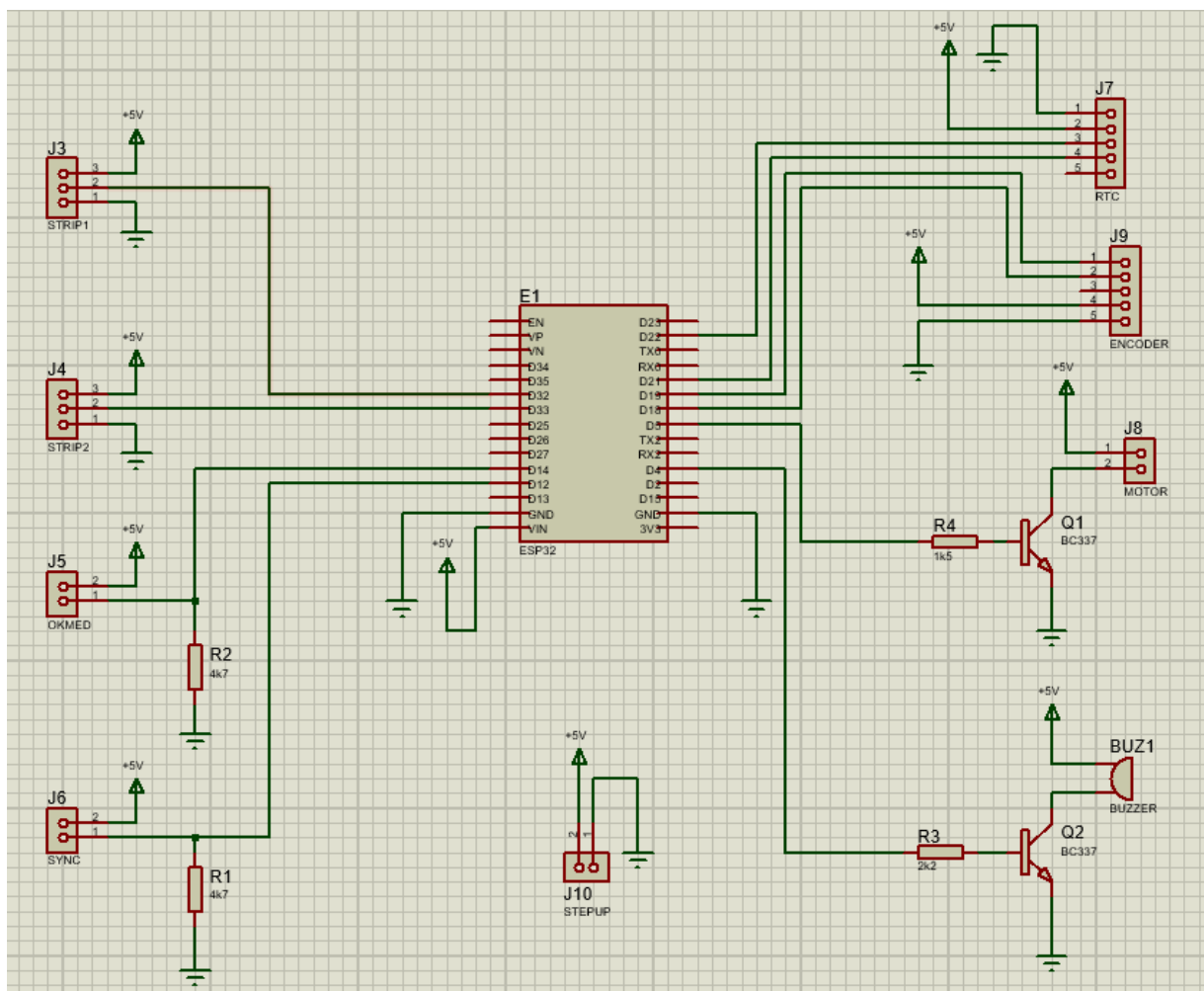
Fonte: Circuits4you (2018).

Conforme pode ser observado na Figura 16, a maior parte das portas da ESP32 apresentam múltiplas funções. É possível observar que existem 18 portas com conversão Analógico-Digital (legenda em laranja de nome “ADC”), 10 pinos de GPIO com sensor capacitivo (legenda em turquesa com nome “Touch”), 03 interfaces UART (legenda em verde claro), 03 interfaces SPI (legenda em azul marinho), 01 interface I2C ou *Inter-Integrated Circuit* (legenda em azul claro com nome de SCL e SDA), 02 canais com conversor Digital-Analógico (legenda em amarelo com nome “DAC”), 16 pinos de PWM, entre outras funções.

Para o projeto em questão, exceto o RTC (elemento 10, da Tabela 9), todos os outros periféricos não necessitam de uma porta específica da ESP32. Esse componente em específico necessita estar nas portas que permitem comunicação I2C, ou seja, os GPIOs de número 22 e 21. O *Encoder Decoder* KY-040 Rotacional, visto na posição 5 da Tabela 9, necessita estar em portas que apresentam PWM, dessa maneira qualquer pino de GPIO atenderia essa demanda.

A partir dessas considerações e dos periféricos listados na Tabela 9, montou-se o esquemático eletrônico no Proteus, *software* que permite a modelagem de esquemáticos de circuitos eletrônicos e a montagem do *layout* de placas para fabricação. Na Figura 17 pode ser observado o circuito montado para o Meepi.

Figura 17 – Esquemático do Meepi



Fonte: Elaboração Própria (2021).

Ao centro da Figura 17 se observa uma representação da ESP32, com código E1. No canto inferior esquerdo da mesma figura, é possível ver dois elementos com código J5 e J6 e com legendas de OKMED e SYNC, respectivamente. Esses

componentes são *t-blocks* onde serão conectados o botão de “*sync*”, cuja função é enviar a solicitação para o servidor na nuvem para atualizar as informações presentes na ESP32, e o botão de “*okmed*”, cuja função é ser utilizado pelo usuário quando for ingerir a medicação. Em ambos foi feita uma configuração de *pull-down*, o que deixa o estado dos botões normalmente em 0, ocorrendo a troca para 1 quando pressionado.

No canto superior esquerdo da Figura 17, visualiza-se os elementos J3, com legenda de STRIP1, e J4, com legenda de STRIP2, nos quais serão conectados a fita de LED RGB WS2812 5050 (elemento 6, Tabela 9) e os 7 LED RGB WS2812B 5050 endereçáveis em série (elemento 7, Tabela 9). Para conexão de ambos não é obrigatório a inserção de nenhum componente eletrônico como intermediário, sendo apenas mandatória a alimentação com 5V. O pino número 2 está conectado ao pino de “DIN” das duas fitas, o qual é o responsável por carregar a informação do padrão a ser ligado os LEDs enviado pelo código presente na ESP32.

Na parte inferior central do esquemático apresentado na Figura 17, é possível verificar a conexão com *t-block* J10, com legenda STEPUP. Nessa porta será conectado o elemento 8 da Tabela 9, o qual estará conectado ao componente TP4056, representado na tabela pelo número 4.

Figura 18 – Módulo Carregador de Bateria Lítio TP4056



Fonte: FILIPEFLOP (2021d).

Como um dos requisitos do protótipo era a portabilidade, a utilização de uma sistema de bateria era importante. Para isso foram selecionados os elementos de número 1, 4 e 8 da Tabela 9, os quais trabalham juntos para permitir esse requisito. O módulo TP4056 (Figura 18) é responsável por permitir o recarregamento da bateria 18650 por meio da sua entrada USB mini, sem a necessidade de retirá-la do circuito.

Esse módulo fornece uma corrente máxima de 1A, dessa maneira, considerando um consumo de 1A por hora pela ESP32, essa carga duraria por 2,6 horas.

Todavia, a bateria escolhida fornece uma tensão de 3,7V, valor com o qual não seria possível alimentar a ESP32, que exige uma alimentação de 5V no seu pino. Diante disso, utilizou-se o conversor *boost* DC ajustável *step up* (Figura 19), para elevar a tensão fornecida pela bateria para 5V. Esse elemento pode apresentar uma saída entre 2 - 48V (FILIPEFLOP, 2021e), valor que é ajustado por meio um *trimpot* que permite a regulagem da tensão para 5V.

Figura 19 – Conversor *boost* DC Ajustável



Fonte: FILIPEFLOP (2021e).

Na Tabela 10 e na Tabela 11 é possível verificar as conexões as serem realizadas entre essas a bateria, TP4056 e o conversor *boost* DC *step up*.

Tabela 10 – Conexões do TP4056

Portas do TP4056	Conexões
B+	Polo positivo da bateria
B-	Polo negativo da bateria
OUT-	VIN- do <i>Step Up</i>
OUT+	VIN+ do <i>Step Up</i>
USB Mini	Na fonte de alimentação

Fonte: Elaboração Própria (2021).

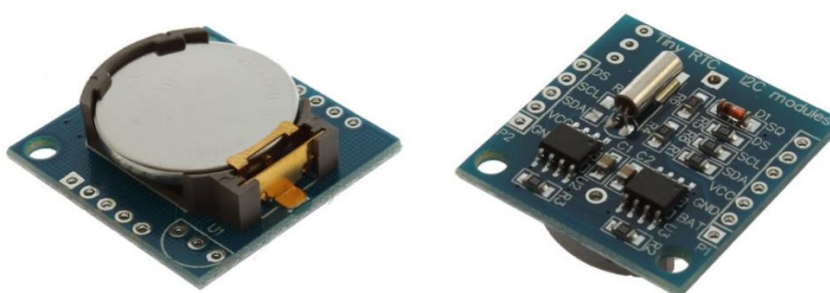
Tabela 11 – Conexões do *Step Up*

Portas do <i>Step Up</i>	Conexões
VIN-	OUT- do TP4056
VIN+	OUT+ do TP4056
VOUT-	Gnd ESP32 / J10-1
VOUT+	Vin da ESP32 / J10-2

Fonte: Elaboração Própria (2021).

Para realizar o alerta sonoro, foram escolhidos o *buzzer* e o motor de vibração, conforme observado na Tabela 9. No esquemático da Figura 17, é possível vê-los na parte inferior direita, com código J8 e BUZ1, respectivamente. Ambos os componentes estão sendo alimentados com 5V, contudo, para que sejam chaveados de maneira adequada, é necessária a utilização de transistores. Diante disso, foi escolhido um BC337 e a partir do ganho apresentado, dimensionou-se os resistores da base.

Figura 20 – DS1307



Fonte: FILIPEFLOP (2014).

Na parte superior direita do esquemático do Meepe, presente na Figura 17, observa-se a legenda RTC (cód. J7). Na Figura 20, pode-se observar a vista da parte superior e da parte inferior do RTC DS1307, o qual será conectado nos pinos do elemento J7. É importante mencionar que no projeto eletrônico foi incluído esse componente para permitir a aferição da data e da hora mesmo sem a presença de internet.

Para funcionamento correto do *Real Time Clock* em questão, basta utilizar uma das portas de SCL, SDA, VCC e GND presentes na sua placa. Diante disso, para o protótipo, será utilizado lado do DS1307 com a menor quantidade de pinos, uma vez que as portas de DS, BAT e SQ são destinadas a recursos opcionais e que não serão utilizados. Vale a pena destacar que nesse módulo existe um circuito capaz de detectar falhas de energia e, quando isso ocorre, é automaticamente acionada a bateria existente no próprio módulo para alimentá-lo (FILIPEFLOP, 2014). Isso permite a contagem do tempo mesmo na ausência de alimentação externa.

Conforme citado anteriormente, é mandatório para RTC estar conectado as portas de I2C da ESP32. Dessa maneira se vê no esquemático a conexão dos pinos

3 e 4 nas portas D22 (SCL) e D21 (SDA) da ESP32, respectivamente. Tais pinos serão conectados as portas SDA e SCL do DS1307, visualizadas na Figura 20. O pino 5 seria destinado ao pino DS do RTC, contudo como não foi usado o sensor de temperatura, não foi feita nenhuma conexão com a ESP32.

Figura 21 – Encoder Decoder KY-040 Rotacional



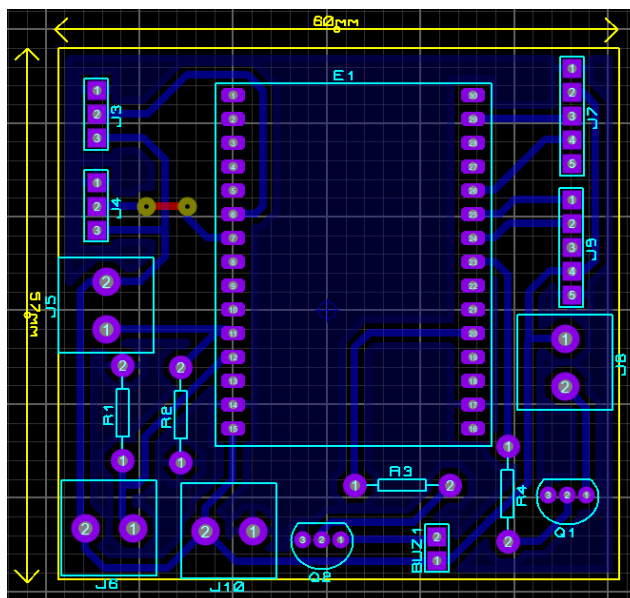
Fonte: Arduino e Cia (2015)

Por fim, na Figura 21 se pode observar o *encoder* o qual será conectado no elemento J9, localizado no canto superior direito, do esquemático da Figura 17. Esse dispositivo realiza a conversão de movimentos rotativos em impulsos elétricos de onda quadrada (ARDUINO E CIA, 2015). O módulo utilizado apresenta portas de GND, +, SW, DT e CLK. As duas primeiras são destinadas a alimentação do módulo, a terceira é para caso de uso do botão que existe embutido ao *encoder*, a quarta e a quinta porta é dedicada para conexão no microcontrolador, elemento que receberá os impulsos de onda quadrada. Como não será usado o botão do *encoder*, a porta 3 do elemento de código 3 não apresenta conexões com a ESP32.

Após a realização do esquemático, a próxima etapa foi a montagem do *layout* da placa para fabricação. Essa atividade também foi realizada no *software* onde foi feito o esquemático, ou seja, o Proteus. Na Figura 22 pode se observar o resultado.

A placa foi montada para ser uma PCI de face simples, o que significa dizer que só existe cobre em um dos lados. Todos os componentes escolhidos são PTH (*Pin Through Hole*), ou seja, apresentam terminais a serem inseridos em furos. No projeto da placa para fabricação foi feita uma malha de aterramento, de modo que todos os terras ficassem em comum. Isso ajuda a colocar todos os componentes na mesma referência, além de facilitar o roteamento das outras trilhas do esquemático. Vale destacar também que se optou pela não inclusão dos módulos diretamente na PCI, uma vez que isso permite uma maior liberdade no momento de se realizar a fixação desses componentes, além de resultar em uma PCI com dimensões menores.

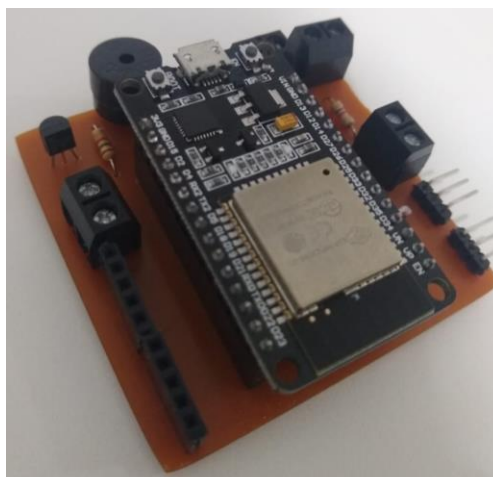
Figura 22 – Layout da placa eletrônica do Meepi



Fonte: Elaboração Própria (2021).

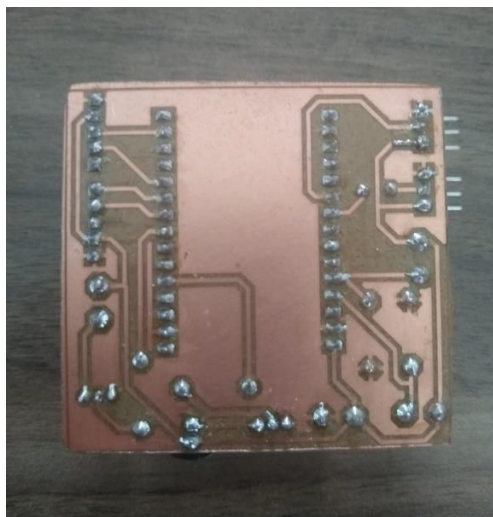
As distâncias relacionadas às trilhas e às ilhas, assim como os seus tamanhos, foram adaptadas de modo a permitir a fabricação da placa em uma CNC laser. Essa máquina executou a usinagem do circuito em uma placa de fenolite. Em seguida, em uma furadeira de bancada, foram realizados os furos para inserção dos componentes. Já a soldagem dos componentes foi realizada de maneira manual, com um ferro de solda e estanho. Antes da soldagem, foi aplicado um verniz para evitar oxidação do cobre. Na Figura 23 se pode verificar a placa do Meepi com todos os componentes soldados e a ESP32 encaixada nos pinos fêmea. Na Figura 24, tem-se uma imagem mostrando o lado do cobre da placa, onde se encontram os circuitos impressos.

Figura 23 – Placa controladora do Meepi



Fonte: Elaboração Própria (2021).

Figura 24 – Vista da parte do circuito

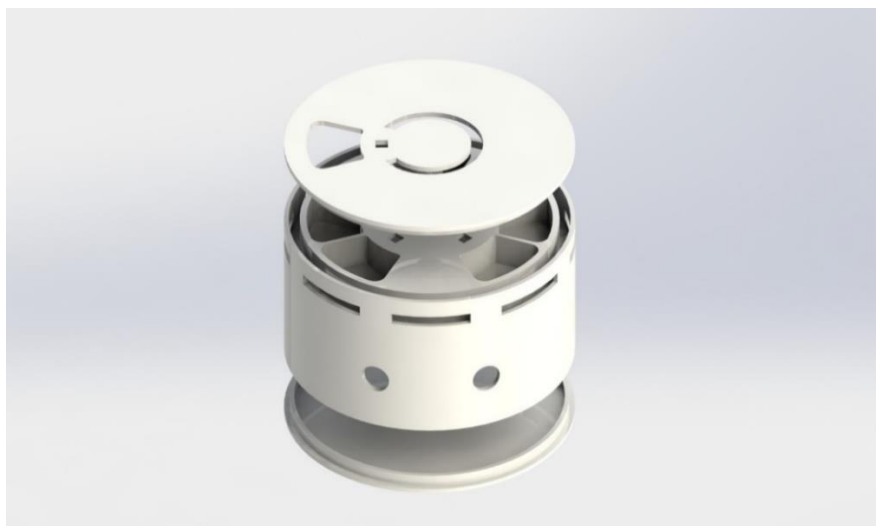


Fonte: Elaboração Própria (2021).

6.2 Desenvolvimento mecânico

Com a parte eletrônica concluída, partiu-se para a modelagem 3D do porta medicamentos. No que se refere ao projeto mecânico, o objetivo era realizar a modelagem tridimensional do Meepi, a qual fosse possível imprimir em uma impressora 3D. Para isso, foram feitos desenhos no *SolidWorks* 2019.

Figura 25 – Vista explodida do Meepi

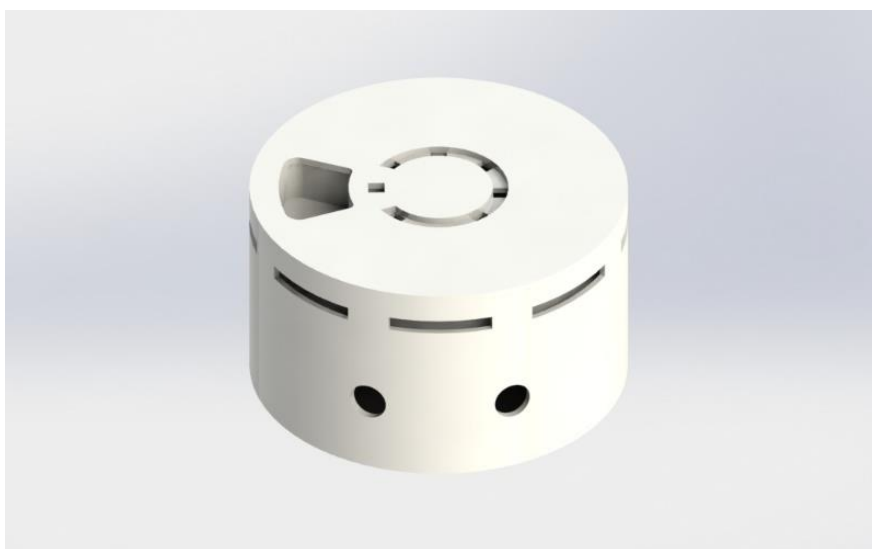


Fonte: Elaboração Própria (2021).

Na Figura 26 é possível verificar uma renderização da versão final do Meepi. Já a partir da Figura 25, é possível verificar uma vista explodida da peça, na qual pode-se observar que a montagem é composta de três peças: tampa superior, base e tampa inferior. A tampa superior apresenta função de segurar os medicamentos em seus compartimentos, além de permitir acesso ao conteúdo

presente em cada compartimento, por meio do seu rasgo, a partir do seu giro. Nessa peça presente nas Figura 25 e Figura 26 também é possível observar um corte em forma de um círculo, cuja função é permitir a visualização dos LEDs que serão ligados de acordo com a localização do remédio a ser ingerido pelo usuário.

Figura 26 – Versão Final do Meepi



Fonte: Elaboração Própria (2021).

A base, elemento central da vista explodida da Figura 25, é o local onde os remédios são armazenados, assim como a parte eletrônica. É importante destacar que nessa peça existe 8 compartimentos, 7 para colocar os medicamentos e 1 que é fechado. Esse compartimento fechado se configura como a posição inicial para o furo da tampa superior giratória, além de permitir fechar a caixa. Observa-se na Figura 25 e na Figura 26 que na base há também cortes retangulares nas laterais, os quais foram feitos para permitirem a visualização da fita de LED. Além disso, pode se verificar a presença de dois furos, onde são fixados os dois botões do protótipo. Por fim, tem-se a tampa inferior, cuja função é apenas isolar a parte eletrônica.

Figura 27 – Impressões 3D



Fonte: Elaboração Própria (2021).

Para se atingir essa versão final presente na Figura 26 foram feitas três impressões 3D, duas das quais podem ser observadas na Figura 27. Ao final o Meepi ficou com 130 mm de diâmetro e 74,5mm de altura.

6.3 Programação

Neste capítulo serão descritas as atividades relacionadas à programação. Para o desenvolvimento do porta remédios inteligente, foram utilizadas duas ferramentas: o Node-RED, onde foi feita a interface gráfica do protótipo e aplicação com o *Telegram*, e a interface de desenvolvimento do Arduino, onde foi programada a ESP32.

6.3.1 IBM Cloud e Node-RED

No capítulo 5.4, discutiu-se as principais alternativas de solução para cada uma das funcionalidades a serem implantadas no MVP. Em relação à funcionalidade de permitir a programação dos horários dos medicamentos pelo usuário e/ou partes interessadas, foi decidido que seria realizado um site para tal atividade. Já relacionado a funcionalidade de notificação a terceiros, optou-se por utilizar *Telegram*. Contudo, para concretizar tais atividades, seria necessário domínio de tecnologias relacionadas a desenvolvimento *web*, infraestrutura de rede, banco de dados, entre outras, além de um grande período de desenvolvimento.

No entanto, conforme citado anteriormente, um MVP deve ser feito da maneira mais rápida possível, com o propósito de se iniciar logo a etapa de medição e aprendizado. Após essas etapas finais do ciclo, seria possível verificar quais soluções implementadas funcionaram bem, quais necessitariam ser melhoradas ou adicionadas na próxima versão do produto. Considerando esse aspecto, a criação de um site geraria um custo de tempo e, no final, existiria a possibilidade de ser uma solução não desejada pelo público-alvo. Diante disso, uma opção encontrada com o menor custo de tempo e de recursos, foi a utilização do Node-RED como ferramenta para desenvolvimento.

O Node-RED é uma ferramenta de programação baseada em *flows*, ou chamados de fluxogramas, originalmente desenvolvida pelo time de tecnologias emergentes da IBM e atualmente parte da JS *Foundation* (OPEN JS FOUNDATION, 2021). Nas atividades realizadas por meio desse tipo de programação existem “caixas-pretas”, que no caso do recurso utilizado se chamam *nodes*, as quais realizam

alguma função específica. Para utilizar esse *nodes*, é necessário o mínimo de programação possível.

Além disso, o Node-RED é uma ferramenta baseada em Node.js (OPEN JS FOUNDATION, 2021). Normalmente, para se utilizar esse recurso, é destinada uma porta de uma máquina local e por meio do navegador se acessa a interface onde são criados os *flows* de programação. Nessa interface existe *nodes* que já vem como padrão no Node-RED, assim como há um repositório aberto, alimentado principalmente pela comunidade, onde existem diversos *nodes* disponíveis para *download*.

Como o site do Meepi deve permitir a programação e visualização das informações referentes à rotina de ingestão de medicamentos por terceiros, buscou-se uma maneira de disponibilizar essa interface a ser desenvolvida no Node-RED para qualquer um com acesso à *internet*. Diante disso, verificou-se que a melhor estratégia seria a implementação do Node-RED em nuvem, atividade que poderia ser realizada de maneira rápida e gratuita por meio da IBM *Cloud*.

O IBM Cloud fornece uma plataforma de nuvem de pilha completa e pública com vários produtos no catálogo, para computação, armazenamento, rede, soluções de desenvolvedor de ponta a ponta para desenvolvimento de app, teste e implementação, serviços de gerenciamento de segurança, bancos de dados tradicionais e de *software* livre e serviços nativos da nuvem.(IBM, 2021).

Diante das funções apresentadas pela IBM *Cloud*, utilizou-se dois produtos disponibilizados no catálogo da empresa: o *Internet of Things Plataforma* e o Node-RED App. O *Internet of Things Plataforma* apresenta a função de conectar dispositivos *wi-fi* reais ou simulados no IBM *Cloud* (WALICKI; IBM, 2018). Ambas as ferramentas foram usadas de maneira integrada e a sua configuração inicial foi conforme tutorial disponibilizado pela própria IBM. Durante essa etapa inicial, foi cadastrado um endereço por onde é possível acesso ao Node-RED no navegador de qualquer computador com *internet*.

É importante comentar que existem outras opções além do IBM *Cloud* para montar a infraestrutura em nuvem, como o *Amazon Web Services* (AWS) e o *Microsoft Azure IoT*. Contudo, foi visto que o serviço apresentado pelo IBM *Cloud* já contava com todos os módulos necessários para utilização do Node-RED e de dispositivos IoT, portanto se mostrou a opção mais completa.

6.3.1.1 Configuração do Internet of Things Platform

Após a configuração da infraestrutura em nuvem e a instalação dos recursos necessários, buscou-se realizar a configuração do *Internet of Things Platform*. Essa etapa é necessária para gerar chaves de API, senhas, IDs (identificações), entre outras informações necessárias viabilizar a comunicação da ESP32 com a nuvem da IBM, assim como com o Node-RED.

Nesse serviço da IBM é possível conectar até 500 dispositivos de maneira gratuita. Como no presente trabalho somente será realizado um protótipo, registrou-se um dispositivo chamado “meepi001”, conforme observado na Figura 28. Ao criar o dispositivo, também é gerado o ID da organização, o qual está omitido na Figura 28, e uma chave de autenticação.

Figura 28 – Central de dispositivos do Internet of Things Platform

The screenshot shows the 'Procurar Dispositivos' (Search Devices) page in the IBM Watson IoT Platform. The page includes a search bar, a 'Todos os dispositivos' button, and a table of devices. The table has the following columns: ID do dispositivo, Status, Tipo de dispositivo, ID de classe, Data de inclusão, and Localização descritiva. Two devices are listed:

ID do dispositivo	Status	Tipo de dispositivo	ID de classe	Data de inclusão	Localização descritiva
meepi001	Desconectado	ESP32	Dispositivo	21 de Fev de 2021 às 11:02	
nodered	Desconectado	nodered	Dispositivo	13 de Mar de 2021 às 17:29	

Fonte: Elaboração Própria (2021)

Figura 29 – Central de chaves de API

The screenshot shows the 'Procurar chaves API' (Search API Keys) page in the IBM Watson IoT Platform. The page includes a search bar and a table of API keys. The table has the following columns: Chave, Descrição, Função, and Expira em. Four API keys are listed:

Chave	Descrição	Função	Expira em
-bbus7r3alo	teste-meepi	Aplicativo de dispositivo	-
-xekjiv6sq	nodered	Aplicativo confiável de backend	-
-xwbogc14nh	Bound to Bluemix Application	Aplicativo padrão	-
-yrlplvszg2f	Bound to Bluemix Application	Aplicativo padrão	-

Fonte: Elaboração Própria (2021)

Ainda na plataforma de *Internet of Things da IBM*, gerou-se duas chaves de API: uma com a função de “aplicativo de dispositivo” destinada a ESP32 e outra voltada para o Node-RED, logo criada na categoria de “aplicativo confiável para *backend*”, as quais podem ser observadas na Figura 29. A primeira chave de API criada, na Figura 29 com a descrição de “teste-meepi”, apresenta o objetivo de gerar uma senha de autenticação para a ESP32, e a segunda, identificada por “nodered”, visa criar uma outra chave, para viabilizar a comunicação entre a nuvem e o dispositivo.

Com tais atividades são finalizadas as etapas de configuração no *Internet of Things Platform*. Em resumo, as principais saídas desse processo relacionadas ao Node-RED foram o ID da organização, a chave de API “nodered” e a senha de autenticação vinculada a essa chave. Já no caso da ESP32, os dados cruciais resultantes desse processo foram: o ID do dispositivo, ID da organização, chave de autenticação “teste-meepi” e a senha vinculada a ela.

As informações resultantes do *Internet of Things Platform* relacionadas ao Node-RED são destinadas a programação dos *nodes* de MQTT. O MQTT (*Message Queue Telemetry Transport*) é um protocolo de comunicação comumente utilizado em aplicações de IoT (*Internet of Things*), por ser leve, assíncrono, bidirecional, adequado para aplicação em redes com alta latência e largura de banda limitada, além de boa compatibilidade para implementação em *hardware* de dispositivo altamente restringido (YUAN; IBM, 2017).

O MQTT apresenta dois tipos de entidades na rede: o *broker* e o cliente. O cliente ao se conectar ao *broker* pode se inscrever em tópicos. Outros clientes conectados a esse *broker* podem publicar nesses tópicos, cuja mensagem vai ser recebida por todos os inscritos. O *broker* apresenta a função de intermediar essa comunicação entre os clientes, encaminhando as mensagens publicadas para as partes adequadas. No caso da aplicação em desenvolvimento, o *Internet of Things Platform* é o *broker* do sistema e as credenciais geradas serão utilizadas para permitir essa conexão dos clientes com o *broker*. No sistema em desenvolvimento só há dois clientes, os quais são a ESP32, ou seja, o porta remédios, e o próprio Node-RED.

Com o término da configuração dessa plataforma, iniciou-se a criação de *flows* no Node-RED, com o objetivo de gerar as páginas onde serão efetuados os

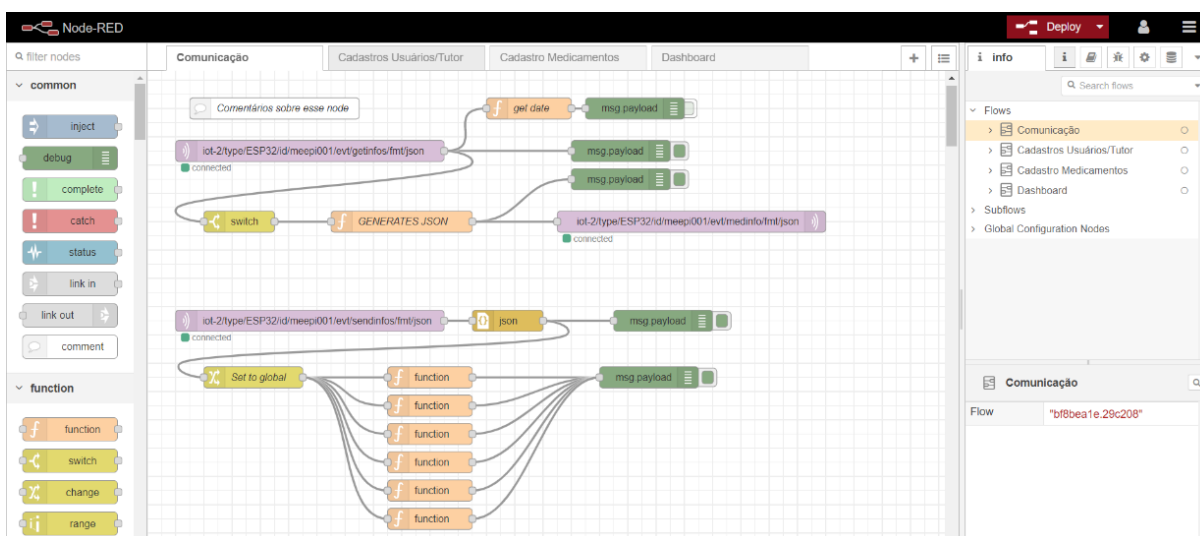
cadastros das informações dos remédios, acompanhamento de ingestão, entre outras atividades.

6.3.1.2 Programação no Node-RED

Para acessar o Node-RED, inseriu-se o endereço criado na etapa de configuração do IBM Cloud no navegador. Ao entrar no endereço, é solicitado um usuário e senha para ter acesso a parte de programação visual do Node-RED. Após permitido o acesso, a página exibida na Figura 30 é carregada.

Na imagem da Figura 30 se pode visualizar o Node-RED com fluxogramas criados para a aplicação em desenvolvimento. Na porção lateral esquerda da mesma figura, é possível se verificar alguns dos *nodes* disponíveis. Para utilizá-los, basta arrastá-los e soltá-los na região quadriculada, onde na Figura 30 já pode se observar a presença de *nodes*. Na porção lateral direita, há um painel de controle, onde é possível navegar em janelas de informação, de ajuda, de *debug*, de configuração, de contexto e de *dashboard*. Para entrar em cada uma das janelas, é necessário selecionar um dos ícones presentes na parte superior do painel de controle. Todas as vezes que forem feitas alterações nos diagramas, deve-se clicar no botão de “*Deploy*”, em vermelho, localizado acima do painel de controle mostrado na Figura 30, para que sejam salvas as alterações.

Figura 30 – Node-RED



Fonte: Elaboração Própria (2021)

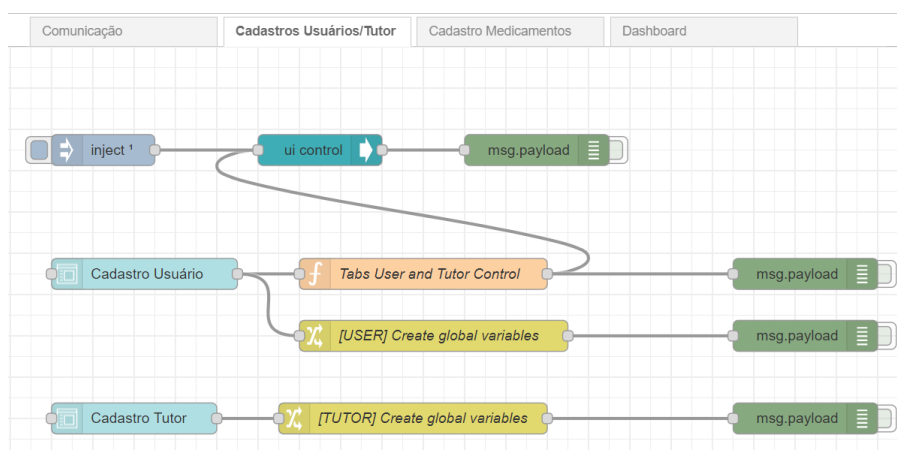
Para facilitar a organização, distribuiu-se a programação visual em abas ou então chamadas de *flows* no Node-RED. Na parte superior da Figura 30 se pode ver as abas de Comunicação, de Cadastros Usuários/Tutor, de Cadastro Medicamentos

e de Dashboard. A aba nomeada de Comunicação apresenta os fluxos relacionados ao protocolo MQTT e a comunicação via *Telegram* com o usuário e o tutor. A aba denominada de “Cadastro Usuários/Tutor” apresenta programações relacionadas a interface para cadastro do usuário do Meepi e do seu tutor, se necessário. Já a seção de “Cadastro Medicamentos” é destinada aos fluxos de programação da interface para cadastro dos medicamentos. Por fim, a aba “*Dashboard*” é relacionada a exibição dos dados recebidos e enviados para o porta medicamentos, de modo que seja possível ao usuário e ao tutor acompanharem tais informações. É importante destacar que os conteúdos dos fluxogramas de cada seção estão sendo executados em paralelo.

Ao se obter caixa de remédios Meepi, o primeiro passo é acessar o site do sistema, com o objetivo de cadastrar o usuário, o tutor (se escolhido) e os medicamentos. Dessa maneira, de modo a permitir uma melhor compreensão da programação desenvolvida, será iniciada a explicação pela aba de Cadastro Usuários/Tutor, seguido da exibição de como ficou a interface de usuário (UI) resultante. Para acesso do *site* do Meepi, o cliente deve escrever `meepi.mybluemix.net/ui/` no navegador.

Na Figura 31 se pode visualizar os *nodes* utilizados para programação das interfaces observadas na Figura 32 e na Figura 33. Na Figura 32 se vê a página que deve ser preenchida com as informações do usuário da caixa Meepi. Caso seja selecionada a opção “Deseja colocar responsável?”, na barra lateral mostrada na Figura 33 aparecerá a aba de “Cadastro Tutor”, permitindo o acesso para inserção das informações solicitadas. Portanto, ao iniciar o *site*, a página de “Cadastro Tutor” deve ser omitida, aparecendo apenas quando for adequada.

Figura 31 – Flow Cadastros Usuários/Tutor



Fonte: Elaboração Própria (2021)

Diante disso, na Figura 31 se observa um bloco de *inject*, cuja função é inserir uma mensagem de maneira manual ou em intervalos regulares. No caso da Figura 31, esse *node* envia um comando, toda vez que for realizado um *deploy*, para que se omita a aba da interface de “Cadastro Tutor”. Essa informação é lida e executada pelo bloco subsequente, o qual apresenta a função de controlar a interface de usuário do Node-RED. O bloco em verde é um bloco de *debug*, o qual permite ao programador visualizar a mensagem que está chegando naquela etapa do processo. Esse bloco não é obrigatório e pode ser utilizado conforme a necessidade do programador.

Figura 32 – UI do Cadastro de Usuário

Cadastro Usuario

Cadastro Usuário

Primeiro Nome
Marina

Celular (com DDD) *
00000000000000

E-mail *
email@email.com

Notificações via Telegram?

Deseja colocar responsável?

ENVIAR CANCELAR

Fonte: Elaboração Própria (2021)

Figura 33 – UI do Cadastro do Tutor

Cadastro Tutor

Cadastro Tutor

Primeiro nome do responsável *

Celular (com DDD) *

E-mail *

Notificações via Telegram para o responsável?

ENVIAR CANCELAR

Menu lateral: Cadastro Usuario, Cadastro Tutor, Medicamentos, Dashboard

Fonte: Elaboração Própria (2021)

Para geração da interface observada na Figura 32 e na Figura 33, utilizou-se os dois blocos em azul claro presentes na Figura 31, chamados de “Cadastro Usuário” e de “Cadastro Tutor”. Esses nodes são do tipo *form* e componentes da biblioteca *dashboard*, a qual apresenta blocos destinados a montagem de interfaces. Esse tipo caixa permite criar um formulário, os quais podem ser observados em ambas as figuras.

Ao clicar no botão de enviar na interface do cadastro de usuário da Figura 32, as informações preenchidas passam para as caixas conectadas ao bloco de *form*. Essas informações vão na forma de um *JavaScript Object Notation*² (JSON), para o *node* subsequente. O bloco seguinte, em laranja, é um *node* do tipo *function* (ou função), o qual permite que o programador desenvolva um código em *javascript* dentro dele.

No caso do bloco *function* conectado ao “Cadastro Usuário” da Figura 31, foi desenvolvido um código onde é visto se o usuário inseriu que deseja cadastrar responsável. Se sim, envia um comando para o bloco *ui control* para que ele mostre a aba da interface “Cadastro Tutor”. Já o bloco “[USER] Create global variables” é do tipo *change*, o qual tem a função de configurar, mudar, deletar ou substituir as propriedades de uma determinada mensagem para o contexto global ou para o contexto do daqueles fluxogramas presentes em determinada seção.

No Node-RED, as variáveis globais podem ser utilizadas em qualquer um dos *flows* criados e as variáveis no contexto do *flow* somente podem ser utilizadas na própria aba do Node-RED onde elas foram criadas. Contudo, os dados obtidos nesses fluxogramas serão utilizados na seção *Dashboard*, logo necessitam ser elevadas para um contexto global. Diante disso, o bloco *change* configura cada variável dentro do objeto *javascript* e a salva dentro de uma determinada variável global. O bloco de “[TUTOR] Create global variables” faz a mesma atividade, só que com as informações do tutor.

Ao passar pelo bloco *change*, na janela de contexto do painel de controle se pode verificar a variável criada e o valor armazenado nela. Para não ser necessário utilizar banco de dados no presente trabalho, escolheu-se salvar todas as informações pertinentes em variáveis globais. Assim, essas dados ficam salvos até o Node-RED ser reiniciado.

Por meio da aba de navegação lateral mostrada na interface da Figura 33, o próximo passo é acessar a página de Medicamentos para executar o cadastro das informações relacionadas. Na Figura 34, é possível verificar a página destinada para tal. Nessa parte do site, primeiramente, deve-se escolher o número de medicamentos

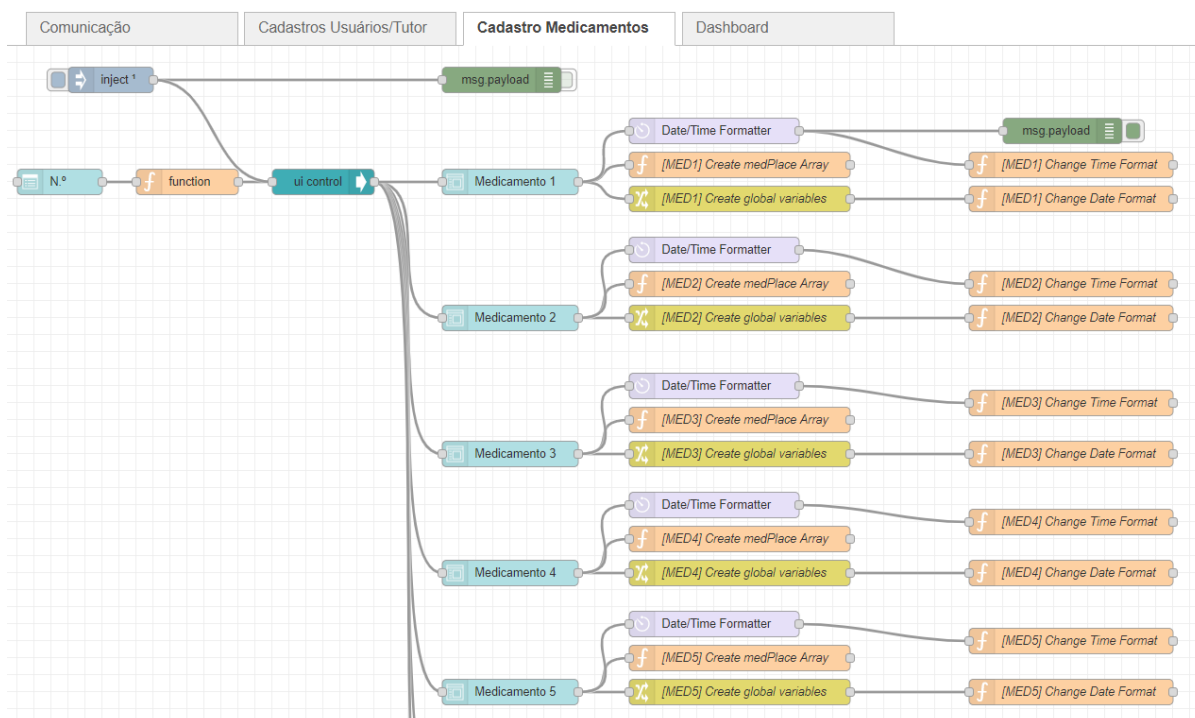
² Também conhecido por JSON ou Notação de Objeto *JavaScript*, em português, é um formato compacto e aberto para compartilhamento de dados.

que se deseja cadastrar. No exemplo mostrado na Figura 34 foram escolhidos 3 e, atualmente, existe a possibilidade de cadastrar até 7 medicamentos. Ao escolher a quantidade, aparece ao lado os formulários para cadastro.

Figura 34 – UI Cadastro de Medicamentos

Fonte: Elaboração Própria (2021)

Figura 35 – Flow de Cadastro Medicamentos



Fonte: Elaboração Própria (2021)

O *flow* da Figura 35 é referente a página presente na Figura 34. O bloco de *inject* apresenta a função de mandar o comando para o bloco de *ui control* esconder os formulários de 1 a 7 quando for realizado o *deploy* do Node-RED. O bloco chamado

de “N.” é da biblioteca *dashboard* e na interface ele é responsável por gerar a lista suspensa para seleção do número de medicamentos que se deseja cadastrar. Ao se escolher o número, esse valor segue para o bloco de função, o qual apresenta uma várias lógicas SE, que dependendo do número, tal *node* envia para o *ui control* a mensagem para omitir ou para mostrar formulários de medicamentos, os quais na imagem então nomeados por Medicamentos 1, Medicamentos 2, Medicamentos 3 e assim sucessivamente.

Depois de preencher com as informações dos medicamentos e pressionar os botões de enviar da interface, as mensagens contendo essas informações de cada remédio na forma de um objeto *javascript* é passada para os blocos de *change* e de *fuction*. Através dos *nodes* de *change*, os valores contidos nas mensagens são armazenados em variáveis globais. Já o *node* de *Date/Time Formatter*, em lilás claro na Figura 35, é para transformar a hora de saída do formulário para o fuso horário do Brasil.

Já as caixas de função como nome de “*Create medPlace Array*”, observadas na Figura 35, recebem também os objetos *javascript*, nos quais também existem as informações referentes ao compartimento do Meepi que esse medicamento foi armazenado. Essa informação, inicialmente, é apresentada em variáveis separadas e em formato booleano. O bloco de função em questão é responsável por transformar esses valores de *true* ou *false* em 1 e 0, respectivamente, e armazená-los na posição correspondente de um vetor. Por exemplo, se o usuário selecionou que o remédio 1 está no compartimento 1 do Meepi, essa função vai formar um vetor que possuirá o valor [1, 0, 0, 0, 0, 0, 0]. Com a criação desses vetores, eles são configurados como variáveis globais também dentro do próprio *node* de função.

Ao preencher a data e a hora nos formulários da Figura 34 são geradas duas *strings* com esses dados para cada medicamento. Com base nisso, os blocos que sucedem os *nodes* de *change* (em amarelo) e os de *Date/Time Formater* presentes na Figura 35, tem a função de partir esses textos, transformar em números inteiros e inserir em um vetor de data e um vetor de hora. Após a criação desses vetores, eles são configurados como variáveis globais também dentro do bloco de função. Todo esse processamento de informações do formulário foi feito com o propósito de facilitar o envio do dados dos medicamentos para a ESP32.

Após o cadastro de medicamentos, é possível ver as informações cadastradas na página de *Dashboard* presente na Figura 36. Na Figura 37, é possível visualizar parte do *flow* que gerou essa página.

No diagrama de blocos da Figura 37, há diversos *nodes* de *inject* que inserem o valor de variáveis globais a cada 5 segundos em um bloco de entrada de texto. Nesse fluxograma mostrado também foram configurados dois botões para apagar as informações referentes aos medicamentos, ao usuário e ao tutor. As informações de último horário consumido e de próximo horário dependem da ESP32 e, portanto, exibem a mensagem de “Aguardando horário” na Figura 36, enquanto não são recebidos esses dados. A informação da última atualização da caixa, vista na Figura 36, também é atualizada ao ser pressionado no botão de sincronização da caixa Meepi.

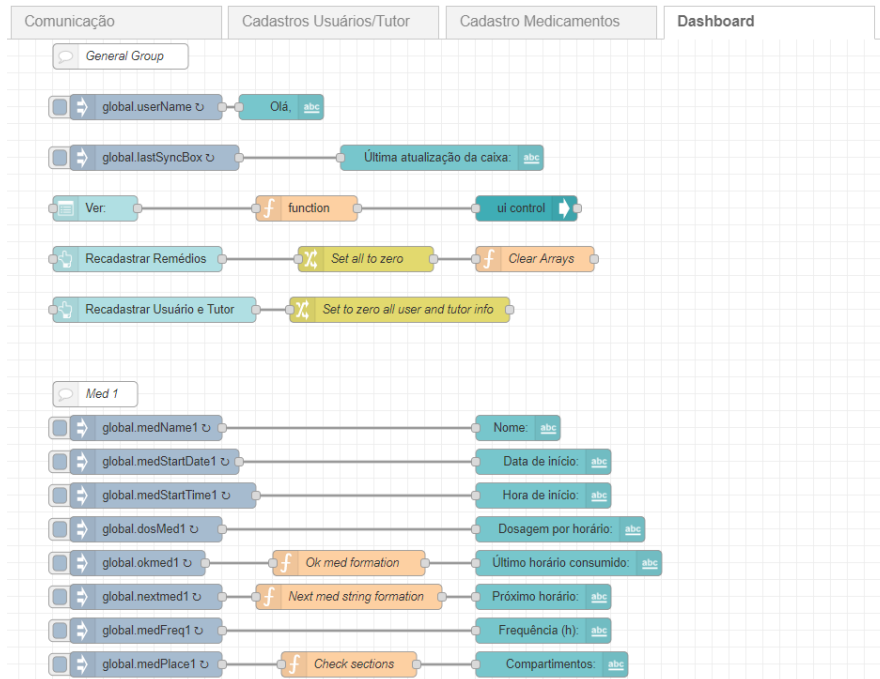
Figura 36 – UI Dashboard



Fonte: Elaboração Própria (2021)

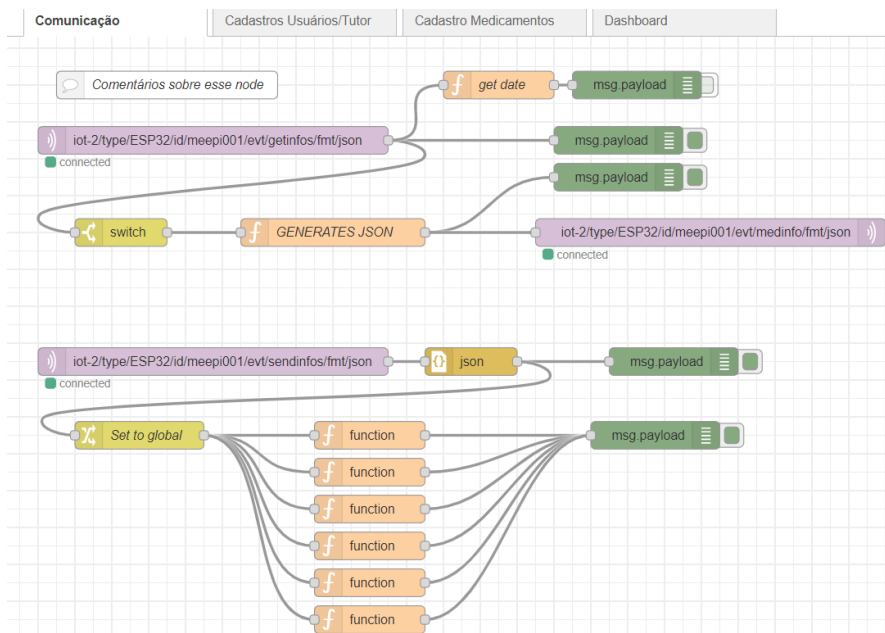
Por fim, no *flow* de Comunicação, exibido na Figura 38, pode-se observar a utilização de *nodes* referentes a comunicação MQTT (blocos em lilás). Dentro desses *nodes* são usadas as informações geradas pelo *Internet of Things Platform* para configuração do *broker* MQTT. O texto que pode ser visto no primeiro bloco é o tópico que o Node-RED está escutando. Por esse tópico, a ESP32 enviará a solicitação para sincronização do porta medicamentos. Ao receber essa solicitação, o bloco de função verifica o horário e salva essa informação em uma variável global, a qual será exibida na abaixo do texto “Última atualização da caixa:” da página de dashboard exibida na Figura 36.

Figura 37 – Parte do Flow Dashboard



Fonte: Elaboração Própria (2021)

Figura 38 – Parte 1/3 do Flow Comunicação



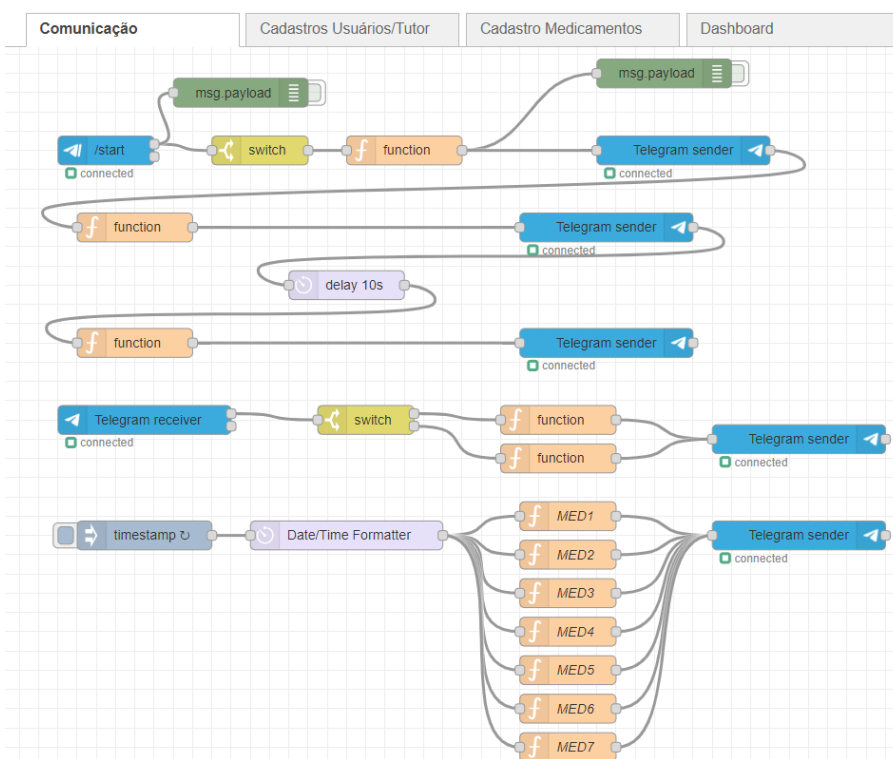
Fonte: Elaboração Própria (2021)

Na Figura 38 se visualiza que o primeiro node MQTT está conectado a um bloco chamado de *switch*. Esse bloco verifica se a mensagem recebida é um JSON contendo a mensagem “*get*”. Se for, é acionado o bloco “*GENERATES JSON*” no qual foi escrito um JSON com todas as variáveis globais pertinentes de serem enviadas

para ESP32, ou seja, a caixa do Meepi. Esse JSON é enviado no tópico presente no *node* MQTT conectado a esse bloco de função.

Logo abaixo do primeiro conjunto de blocos da Figura 38, há um outro *node* MQTT, o qual está escutando as mensagens do tópico “*iot-2/type/ESP32/id/meepi001/evt/sendinfos/fmt/json*”. Nesse tópico, a ESP32 publica as informações adquiridas pelos botões e sensores da caixa Meepi. Esses dados na forma de *string* JSON são recebidas pelo MQTT que em seguida envia para o *node* seguinte conectado a ele, responsável por converter essa mensagem em um objeto javascript. Em seguida, essas variáveis presentes no objeto são salvas em variáveis globais. Os blocos de *function* que sucedem são destinados a atualizar os dados referentes ao próximo e ao último horário de consumo dos medicamentos presente na *Dashboard* da Figura 36.

Figura 39 – Parte 2/3 do Flow Comunicação



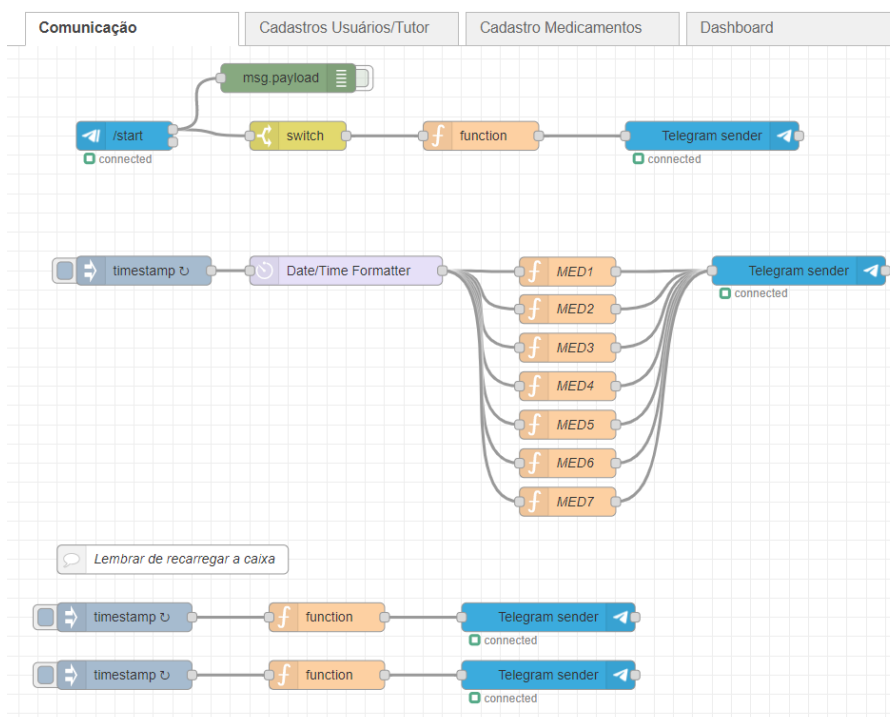
Fonte: Elaboração Própria (2021)

A parte 2 do *flow* de comunicação, presente na Figura 39, é referente a programação relacionada as mensagens do *Telegram* enviadas ao usuário da caixa Meepi. Pelo próprio aplicativo do *Telegram* foi registrado um robô. Por meio desse cadastro, foram geradas credenciais para robô, as quais foram inseridas nos *nodes* referentes a esse serviço. Ao iniciar uma conversa com o @MeepiBot no *Telegram*, o

comando de “/start” é enviado. A partir de então, se inicia o fluxo de mensagens para o usuário. De modo geral, são enviadas mensagens de boas-vindas e notificações nos horários dos medicamentos.

Na parte 3 do *flow* de Comunicação (Figura 40), observa-se um diagrama de blocos referente as mensagens que são enviadas ao tutor, o qual ao iniciar a conversa com o robô, começa o fluxo de notificações. Decidiu-se por enviar notificações para o responsável nos horários que o usuário necessita ingerir alguma medicação. Caso ele queira conferir se o usuário consumiu ou não determinado medicamento, o tutor deve acessar a Dashboard encontrada no site do Meepi. Por fim, os dois últimos diagramas que são iguais e estão presentes na Figura 40 apresentam o objetivo de enviar notificações em um horário e em dia definido para o usuário e para o tutor, com o objetivo de lembrar ambos de reabastecer a caixa Meepi com medicamentos, se necessário.

Figura 40 – Parte 3/3 do Flow Comunicação



Fonte: Elaboração Própria (2021)

Com isso se finaliza a parte referente ao Node-RED. Todos os *flows* e códigos desenvolvidos no presente trabalho estão disponíveis em repositório do GitHub³, caso seja necessária a análise mais aprofundada.

³ Disponível em: <<https://github.com/marinadardenne/meepi-mvp-codes>>. Acesso em: 27 mar. 2021

6.3.2 ESP32

A programação da ESP32, a qual é responsável por controlar a caixa Meepi, foi realizada na interface de desenvolvimento do Arduino. A linguagem utilizada nesse processo é a mesma usada nas placas Arduino, a qual apresenta influências majoritariamente do C++.

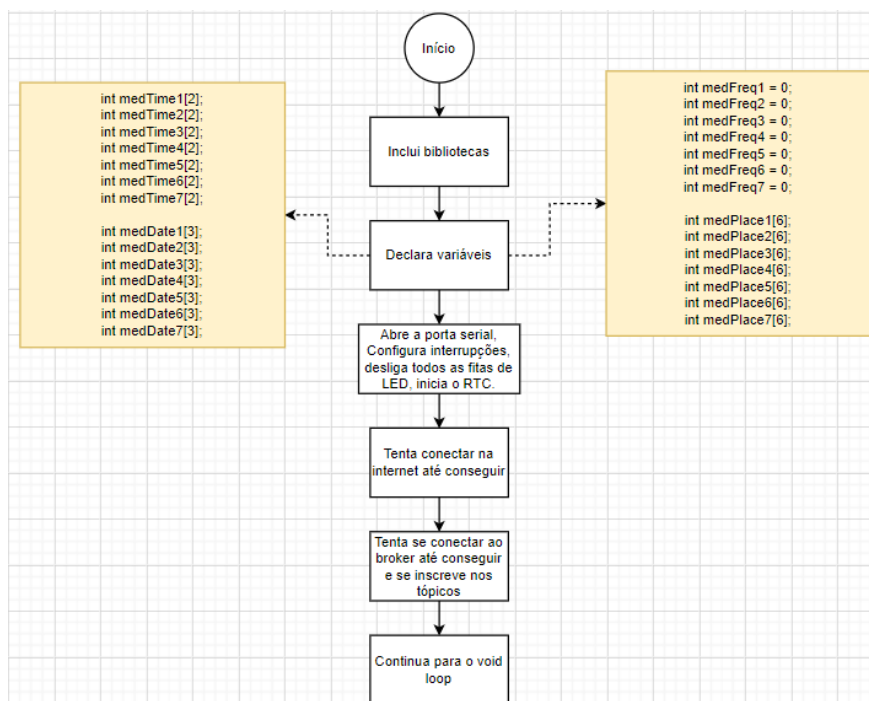
Para construir o código se utilizou do método iterativo incremental. Durante o processo de implementação, procurou-se utilizar todos os componentes eletrônicos propostos, com o objetivo de validar determinado trecho do código desenvolvido. Diante disso, foram realizados testes com diversos programas, muitos dos quais utilizavam o *encoder*. Contudo, não foi possível a sua implementação na versão final, uma vez que o dispositivo não apresentava a precisão necessária para a presente aplicação, portanto, optou-se por retirar esse elemento do projeto. Além disso, foi encontrada dificuldade para conciliar a necessidade de leitura por interrupção do *encoder* com as outras lógicas envolvidas no programa.

Considerando a retirada do *encoder*, a lógica de utilização do Meepi ficou baseada no botão “okmed”. Em resumo, para utilização do porta medicamentos, é necessário, em um primeiro momento, o usuário pressionar o botão de sincronização, para mandar uma solicitação para nuvem enviar as informações do medicamento. Após isso, os horários que a caixa despertará estarão armazenados na ESP32. Quando chegar a data e hora de ingestão do(s) medicamento(s), o Meepi iniciará os alertas sonoros e os visuais. Antes de tomar o(s) remédio(s), o usuário deve pressionar o botão de “okmed”, o que para os alertas sonoros da caixa. Logo em seguida, depois de tomar todos os medicamentos daquele determinado horário, o usuário deve clicar novamente no mesmo botão, o qual ocasionará o desligamento definitivo de todos os alertas e da iluminação dos compartimentos.

De modo a explicar as partes principais do código desenvolvido, serão utilizados fluxogramas feitos no Draw.io, o qual é um editor *online* integrado ao *Google Drive* que permite a criação de fluxogramas, desenhos e gráficos. Caso seja necessária análise aprofundada, a programação na íntegra desenvolvida está disponibilizada em repositório aberto do GitHub, juntamente com os *flows* do Node-RED.

Na Figura 41, verifica-se a primeira parte do fluxograma da caixa Meepi. Logo no início são incluídas as bibliotecas utilizadas no decorrer do código. Na Tabela 12, pode-se observar as bibliotecas usadas e as suas funções.

Figura 41 – Fluxograma Meepi Parte 1/7



Fonte: Elaboração própria (2021).

Tabela 12 – Lista de bibliotecas e funções

Biblioteca	Função
Adafruit_NeoPixel.h	Controle das fitas de LED
PubSubClient.h	Uso do protocolo MQTT
WiFi.h	Viabilizar a conexão com a <i>internet</i>
RTCLib.h	Utilização do RTC DS1307
Wire.h	Comunicação I2C
ArduinoJson.h	Criar e processar JSONs
Esp32MQTTClient.h	Uso do protocolo MQTT

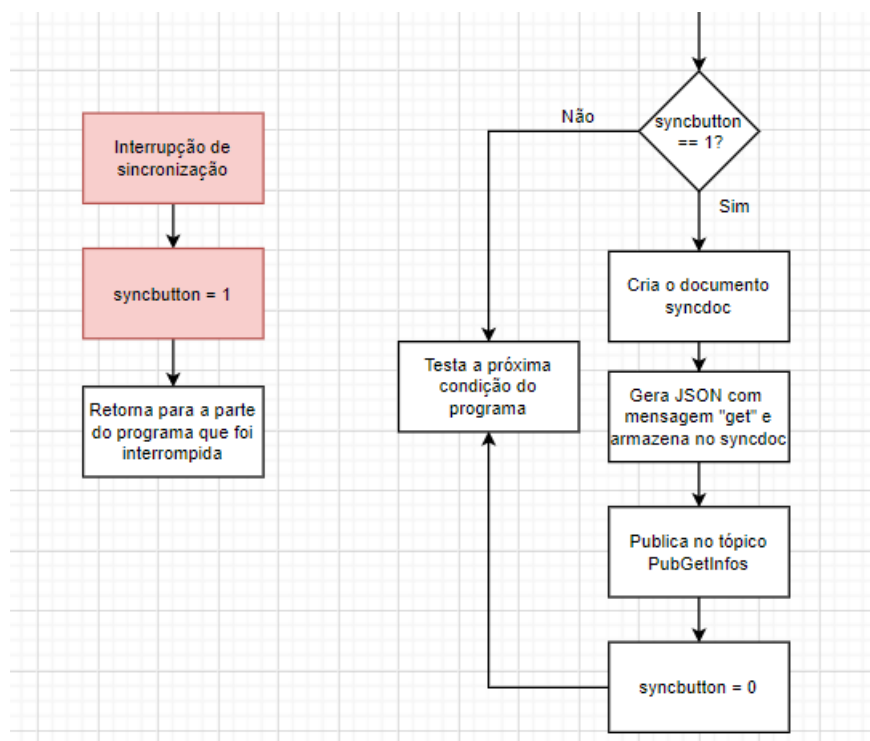
Fonte: Elaboração própria (2021).

Após a inclusão das bibliotecas, ocorre a declaração das variáveis. Nos quadrados em amarelo localizados na esquerda e na direita do bloco de “Declara variáveis” da Figura 41, é possível ver algumas das variáveis principais, onde são armazenadas as informações vindas do JSON gerado no Node-RED. No JSON recebido, há os vetores de data e hora, os quais são armazenados nas variáveis com prefixo “*medTime*” e “*medDate*” ao chegarem na ESP32. As variáveis de “*medPlace*”, que também são recebidas do IBM *Cloud*, são referentes a localização do remédio

nos compartimentos do Meepi e o “*medFreq*”, a frequência em horas que aquele determinado medicamento é ingerido.

Em seguida, no fluxograma presente na Figura 41, ocorrem as atividades padrões do *void setup*⁴, no qual é declarado as entradas e as saídas, iniciada a porta serial, realizada a configuração das interrupções aos pinos dos botões, iniciado o RTC e desligado as fitas de LED (os LEDs dos compartimentos também serão considerados uma fita de LED por funcionarem da mesma maneira que a fita de iluminação externa do Meepi). No projeto em questão, o botão de “*sync*” (sincronização) e o de “*okmed*” são utilizados com interrupções do tipo *RISING*, ou seja, quando o pino muda de estado de 0 para 1, ocorre a interrupção. Além disso, observa-se outras atividades presente no *void setup*: chamar a função para conexão com a *internet* e a função que conecta via MQTT com *IBM Cloud*. Após essas atividades iniciais, segue-se para o laço principal do programa, o *void loop*.

Figura 42 – Fluxograma Meepi Parte 2/7



Fonte: Elaboração própria (2021)

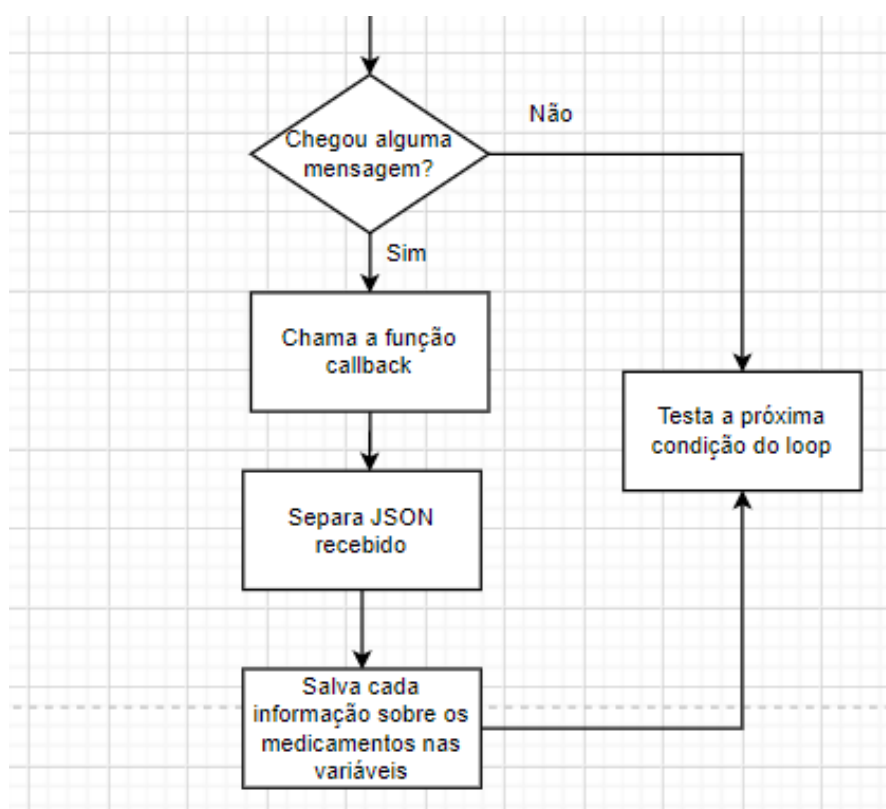
Conforme citado, a primeira atividade a ser realizada pelo usuário é pressionar o botão de sincronização. Ao fazer isso, no programa ocorre uma interrupção. Isso significa dizer que a ESP32 interrompe a atividade que estava sendo

⁴ Laço do programa que só é executado uma vez.

executada para atribuir o valor de 1 a variável “*syncbutton*”, o que pode ser observado no lado esquerdo da Figura 42. Após a execução dessa rotina, o programa retorna para parte do programa que foi interrompida.

Ao dar continuidade no laço principal do programa, observa-se que o pressionamento do botão de “*sync*” torna verdadeira a condição presente no lado direito da Figura 42. Dessa maneira, é criado um documento JSON, o qual conterá a mensagem “*get*”. Essa mensagem é publicada no tópico “*pubGetInfos*” e desencadeia o envio das informações das medicações pelo Node-RED. Por fim, atribui-se o valor de zero a variável “*syncbutton*”.

Figura 43 – Fluxograma Meepi Parte 3/7



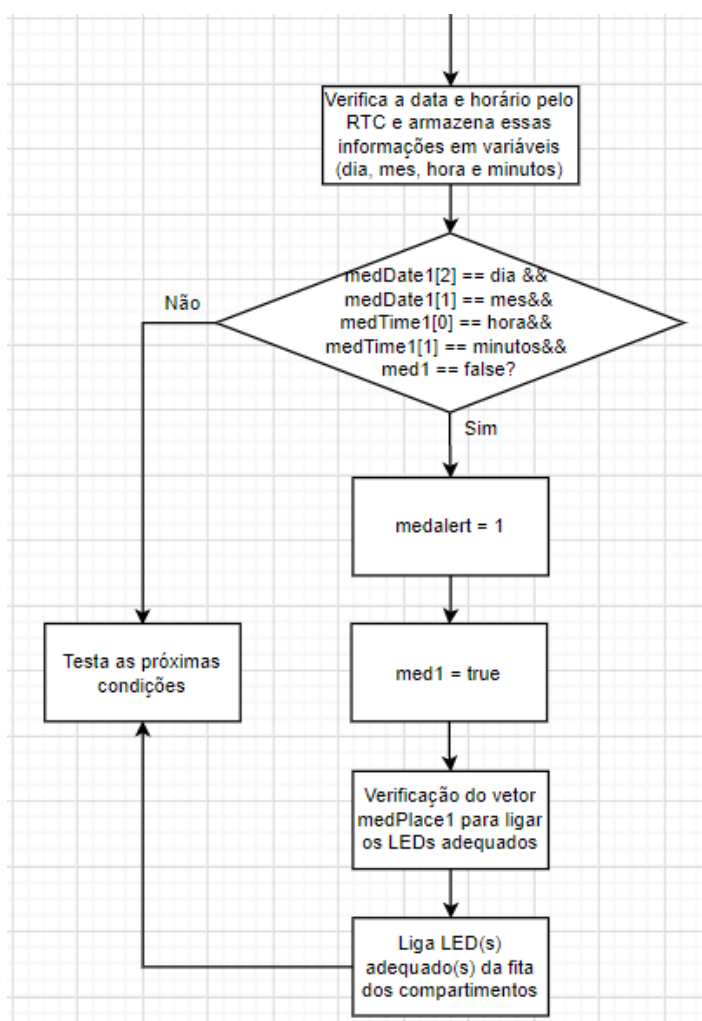
Fonte: Elaboração própria (2021)

Como foi solicitado pelo Meepi o envio das informações, a condição presente na Figura 43 será verdadeira. Essa atividade verifica se chegou alguma mensagem no tópico “*lisMedInfo*”, o único tópico no qual a ESP32 está inscrita. Caso a condição seja respeitada, chama-se a função de *callback*, a qual conterá as atividades de separar o JSON recebido e salvar cada um dos dados nas variáveis destacadas na Figura 41. Vale relembrar que no Node-RED é construído esse JSON com as informações dos medicamentos preenchidos no site.

Com as informações dos medicamentos salvas no Meepe, o usuário deverá aguardar o horário que a caixa o alertará. Diante disso, no *void loop* existe uma atividade dedicada a verificar se esse determinado horário chegou para cada um dos medicamentos cadastrados, a qual pode ser observada na Figura 44.

A cada laço do programa, é verificada a data e o horário. Essas informações são comparadas com o horário e data cadastrados pelo usuário, conforme pode ser visto na condição SE presente na Figura 44. Vale destacar que a mesma condição existe para os medicamentos de 2 a 7. Desse modo, após a condição referente ao remédio 1, testa-se para o remédio 2, 3, 4 e assim por diante.

Figura 44 – Fluxograma Meepe Parte 4/7

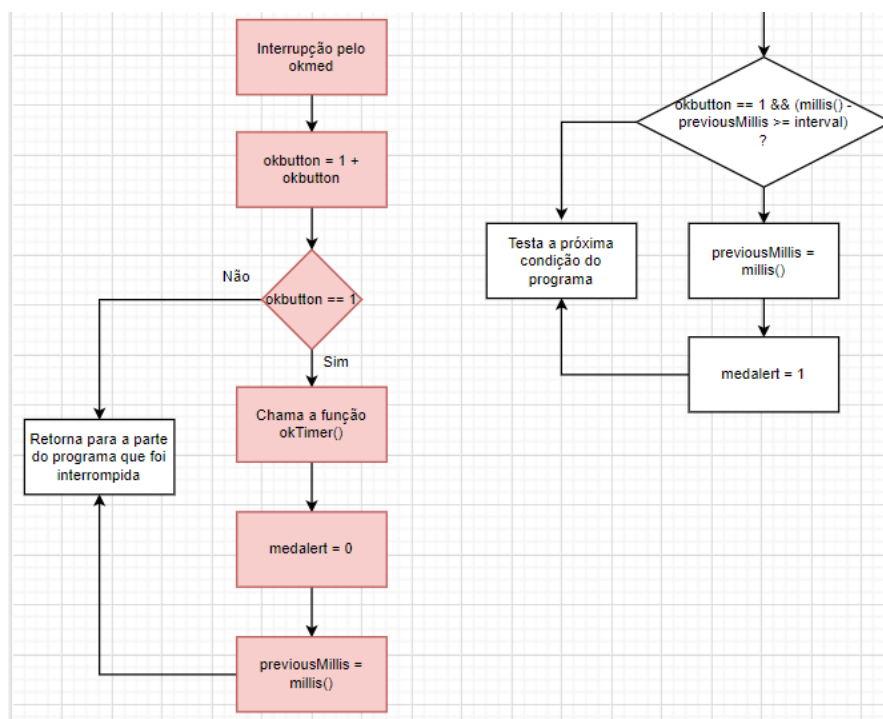


Fonte: Elaboração própria (2021)

Na Figura 44, observa-se que se a condição para o medicamento 1 for verdadeira, a variável “*medalert*” receberá o valor de 1, o que tornará verdadeira uma condição que executará a rotina de alertas. Em seguida, atribui-se o valor *true* para variável “*med1*”, verifica-se em quais compartimentos do Meepe se encontra o

medicamento daquele horário por meio do vetor “*medPlace1*” e, por fim, liga-se o(s) LED(s) da fita dos compartimentos adequada. A rotina executada para caso seja horário de algum outro medicamento é idêntica à mostrada no exemplo da figura em questão, alterando-se as variáveis apenas para numeração do remédio avaliado.

Figura 45 – Fluxograma Meepi Parte 5/7



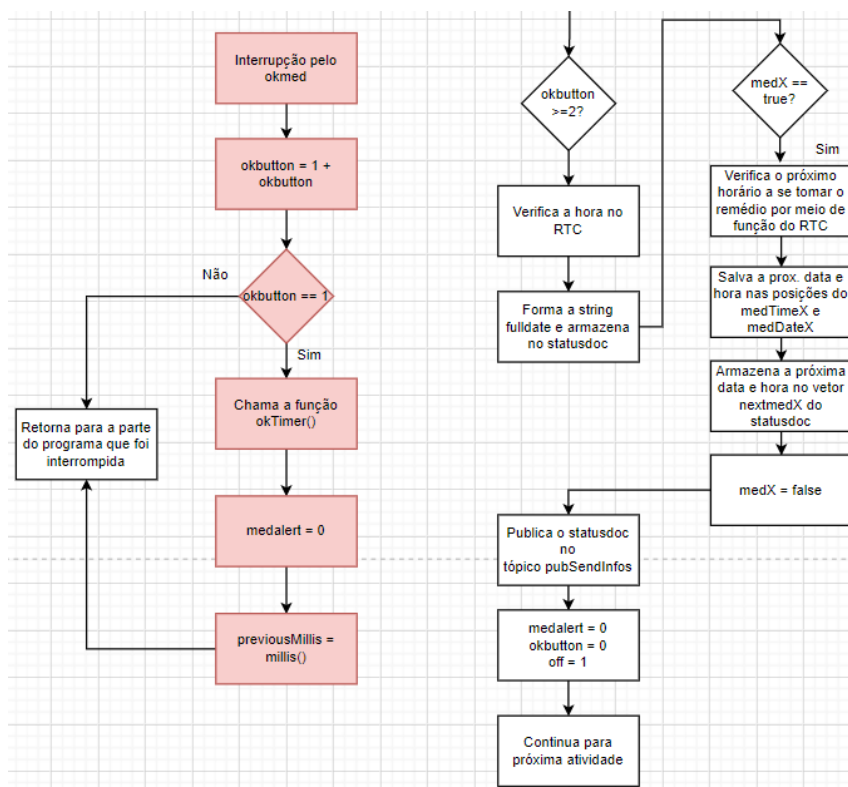
Fonte: Elaboração própria (2021)

Se chegou o horário de determinado medicamento, os alertas são ativados e, portanto, o usuário irá tomar o medicamento. Antes de fazer essa ação, ele deverá pressionar o botão de “*okmed*” pela primeira vez. Ao pressionar tal botão, é gerada uma interrupção, a qual é responsável por adicionar o valor de 1 a variável “*okbutton*”, o que pode ser observado no lado esquerdo da Figura 45. Se o valor de “*okbutton*” for igual a 1, chama-se a função “*okTimer()*”, na qual será atribuído o valor de 0 a variável “*medalert*” e será armazenado um valor de tempo na variável “*previousMillis*”. A função “*millis()*” verifica o tempo, em milissegundos, que a ESP32 está ligada. Além disso, ao atribuir o valor de 0 a “*medalert*”, a partir da verificação da condição relacionada a essa variável, são desligados o *buzzer*, o motor de vibração e a fita de LED externa para de oscilar a cor.

No laço principal do programa, existe uma condição diretamente relacionada a essa interrupção, a qual se pode verificar no lado direito da Figura 45. Nela, além de ver se “*okbutton*” é igual a 1, é avaliado se o tempo que a ESP32 está

ligada subtraído do tempo contido na variável “*previousMillis*” é maior ou igual a determinado intervalo. Se isso for verdadeiro, atualiza-se o a valor presente em “*previousMillis*” e atribui o valor de 1 para variável “*medalert*”, de modo a posteriormente voltar a ativar os alertas sonoros e visuais. Isso foi feito para evitar que a pessoa esqueça de clicar no botão de “*okmed*” pela segunda vez.

Figura 46 – Fluxograma Meepi Parte 6/7



Fonte: Elaboração própria (2021)

Após pressionar o botão pela primeira vez, o usuário do Meepi deve navegar pela caixa e tomar seus medicamentos. Com essa etapa concluída, deve-se pressionar pela segunda vez o botão “*okmed*”. Ao realizar tal ação, é chamada novamente a interrupção motivada por esse botão, a qual pode ser observada também no lado esquerdo da Figura 46. Por meio dela, incrementa-se o valor de 1 a variável “*okbutton*” novamente.

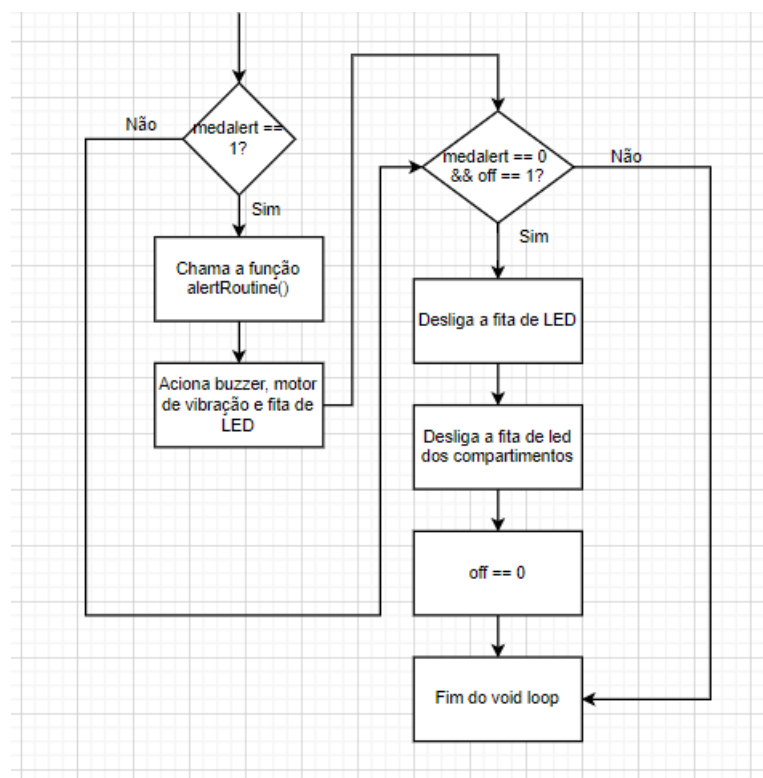
Já no lado direito da Figura 46, observa-se a condição presente *void loop* relacionada a interrupção anterior, a qual verifica se o valor de “*okbutton*” é maior ou igual a 2. Considerando tal condição verdadeira, verifica-se o horário naquele momento pelo RTC DS1307 e forma-se a *string* “*fulldate*”, a qual é um texto que contém o dia, mês, ano, hora e minutos que se entrou nessa condição. Essa *string* é armazenada no “*statusdoc*”, documento JSON que será enviado a nuvem no final do

processo. Tal documento JSON foi criado para armazenar a informação do horário que determinado medicamento foi ingerido e o dado do próximo horário a ser tomado o medicamento.

Em seguida, o próximo passo a ser realizado é verificar qual variável com prefixo “med” é verdadeira. A checagem de qual variável com prefixo “med” é verdadeira ocorre de 1 a 7, ou seja, condição “medX” igual verdadeiro ocorre 7 vezes. Contudo, para simplificar no fluxograma mostrado na Figura 46, foi representada como “medX”, onde X pode ser entendido como 1, 2, 3, 4, 5, 6 e 7.

Caso um ou mais remédios tenham sido ingeridos naquele momento, calcula-se o próximo horário a ser tomado o remédio, por meio de uma função da biblioteca “*RTClib.h*” e por meio do valor contido na variável “*medFreq*” do número correspondente, conforme pode ser observado na Figura 46. Considerando “*med1*” como verdadeiro, por exemplo, a próxima atividade seria atualizar os vetores “*medTime1*” e “*medDate1*” com os horários da próxima vez que o remédio 1 será consumido. Essa informação também é salva no vetor “*nextmed1*” do “*statusdoc*”.

Figura 47 – Fluxograma Meepi Parte 7/7



Fonte: Elaboração própria (2021)

Após verificar se foi o “*med7*” consumido naquele determinado horário, ocorre a publicação do “*statusdoc*” no “*pubSendInfos*” (Figura 46). Após esse

acontecimento, atribui-se o valor de zero para “*medalert*” e para “*okbutton*”. Já para variável “*off*” é atribuído o valor de 1. Essa variável “*off*” está relacionada à condição que ocasionará o desligamento de todos os LEDs do sistema, o que pode ser visto na Figura 47. A condição relacionada ao “*medalert*” também pode ser visualizada na Figura 47. Essas condições são as últimas testadas no programa e, após o seu término, é executado o *void loop* novamente.

6.4 Integração do porta medicamentos

Após o desenvolvimento do código da ESP32, procurou-se realizar a integração de todas as partes desenvolvidas nesse projeto na caixa impressa 3D. A inserção dos componentes eletrônicos na caixa foi difícil, uma vez que não foram feitas fixações para inseri-los. Na Figura 48 e na Figura 49, é possível verificar o resultado dessa integração.

Figura 48 – Caixa Meepi Montada (vista superior)



Fonte: Elaboração própria (2021).

Além disso, a bateria escolhida funcionou adequadamente com o porta medicamentos, contudo, é interessante a adição de uma chave para ligá-la e desligá-

la. Uma outra questão que se mostrou importante é ter um retorno visual da situação de carga da bateria, a qual atualmente existe no módulo TP5046, mas não é visível externamente a caixa.

Figura 49 – Caixa Meepi Montada (vista isométrica)



Fonte: Elaboração própria (2021).

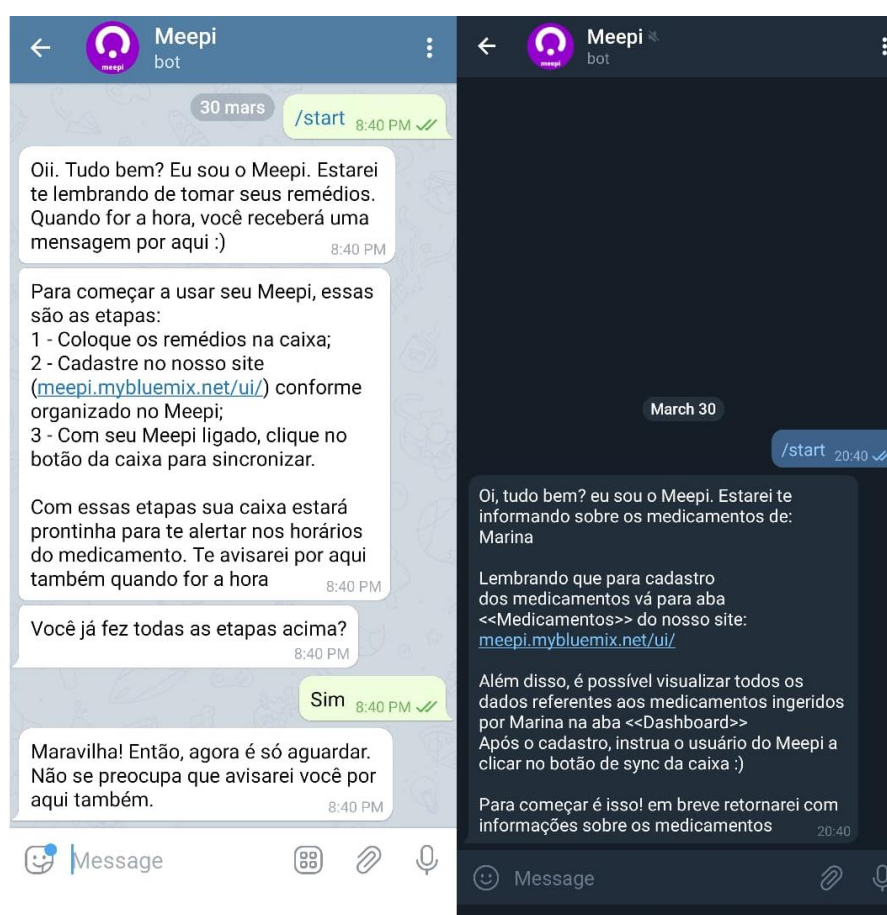
7 TESTE DO MEEPI

O presente teste foi realizado com a intenção de verificar o funcionamento do Meepi, de modo a que se avaliasse as oportunidades de melhoria e o desempenho do produto. Assim, será possível de definir se o protótipo está apto a ser considerado um MVP e, portanto, avançar nas próximas etapas do ciclo de Construir-Medir-Aprender. É importante destacar que esses testes foram realizados pela própria autora do projeto e, durante o a sua realização, o porta medicamentos estava conectado na USB de um computador para que fosse possível o acompanhamento das atividades via porta serial.

Para começar os testes, buscou-se cadastrar as informações do usuário e do responsável. Em seguida, foi iniciada a conversa no celular do usuário da caixa com o @MeepiBot no *Telegram*. A mesma ação foi realizada no celular do tutor.

As conversas ocorreram conforme o programado no Node-RED, as quais podem ser observadas na Figura 50. É importante destacar que o acesso ao @MeepiBot se deu por meio de um *link*, informado após a criação do robô. A princípio, a dinâmica de conversas com o robô apenas funciona com o celular desse usuário da caixa e desse responsável em específico, visto que foram cadastradas as identificações relacionadas a apenas essas duas conversas nos *nodes* do *Telegram* no Node-RED.

Figura 50 – Início da conversa no Telegram com responsável e com o usuário



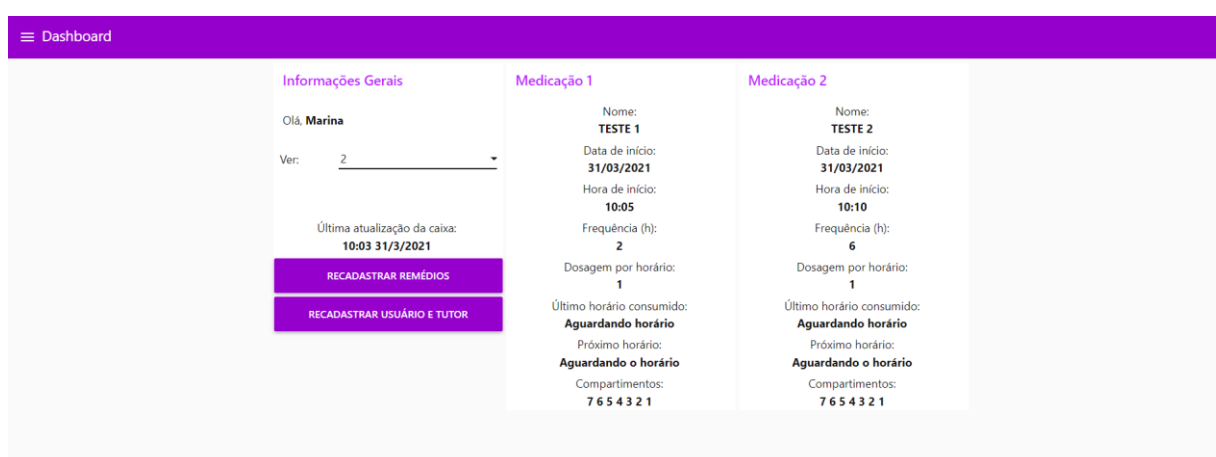
Fonte: Elaboração própria (2021).

Após o início das conversas com o robô, buscou-se realizar os testes para validar o funcionamento da caixa. Diante disso, programou-se no site do Meepi dois medicamentos, resultado que pode ser observado na aba de *dashboards* mostrado na Figura 51.

A *dashboard* se comportou também conforme o programado no Node-RED. Acima do botão de "RECADASTRAR REMÉDIOS" da Figura 51, observa-se o último horário no qual a caixa Meepi foi atualizada, ou seja, recebida a solicitação para

sincronização dos seus dados. Já na parte de medicação 1 e de medicação 2, vê-se o nome do medicamento, a data de início, a hora de início, a frequência da medicação, a dosagem por horário e os compartimentos conforme o cadastrado. Até a ocorrência do primeiro alarme na caixa, ficará exibindo a mensagem de “Aguardando o horário” abaixo de “Último horário consumido” e de “Próximo horário”, observada também na Figura 51. Na Figura 52 e na Figura 53 se pode ver a atualização do dashboard após a caixa Meepi ter enviado os dados, ou seja, após o usuário ter supostamente ingerido os medicamentos nos respectivos horários.

Figura 51 – Dashboard após programação dos remédios



Fonte: Elaboração própria (2021).

Figura 52 – Dashboard após o primeiro horário do medicamento 1



Fonte: Elaboração própria (2021).

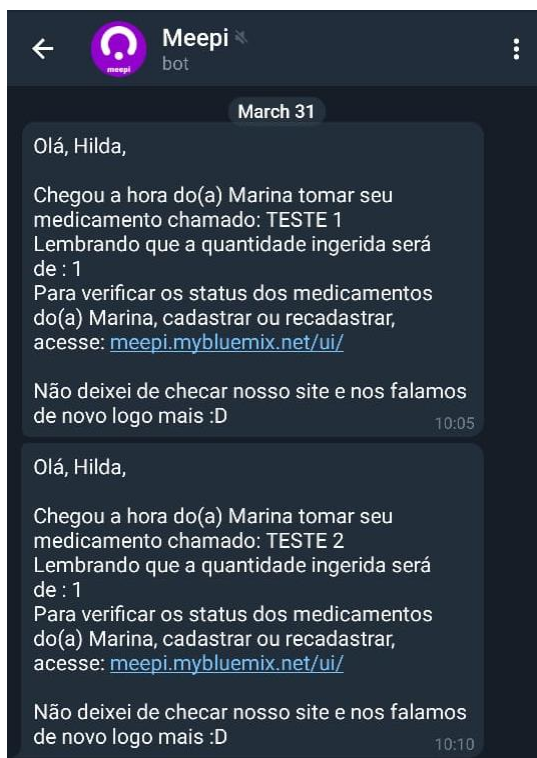
É importante destacar que na Figura 52 e que na Figura 53 as próximas datas e horários dos medicamentos a serem ingeridos estão de acordo com as frequências de repetição cadastradas.

Figura 53 – Dashboard após primeiro horário do medicamento 2

Informações Gerais	Medicação 1	Medicação 2
Olá, Marina	Nome: TESTE 1	Nome: TESTE 2
Ver: 2	Data de início: 31/03/2021	Data de início: 31/03/2021
	Hora de início: 10:05	Hora de início: 10:10
Última atualização da caixa: 10:03 31/3/2021	Frequência (h): 2	Frequência (h): 6
RECADASTRAR REMÉDIOS	Dosagem por horário: 1	Dosagem por horário: 1
RECADASTRAR USUÁRIO E TUTOR	Último horário consumido: 31/3/2021 10:5	Último horário consumido: 31/3/2021 10:11
	Próximo horário: 31/3/2021 12:5	Próximo horário: 31/3/2021 16:10
	Compartimentos: 7 6 5 4 3 2 1	Compartimentos: 7 6 5 4 3 2 1

Fonte: Elaboração própria (2021).

Figura 54 – Mensagens recebidas no Telegram do tutor após primeiro horário do medicamento 1 e do medicamento 2



Fonte: Elaboração própria (2021).

As mensagens no *Telegram* do responsável e do usuário também foram recebidas conforme o programado, visto que foram recebidas nos dois horários desejados. Na Figura 54 se pode observar as mensagens recebidas as 10 horas e 5 minutos e as 10 horas e 10 minutos no celular do responsável cadastrado. Já na Figura 55, pode-se visualizar, lado a lado, as mensagens recebidas nos mesmos horários pelo usuário da caixa. Em ambas, os dados referente aos medicamentos

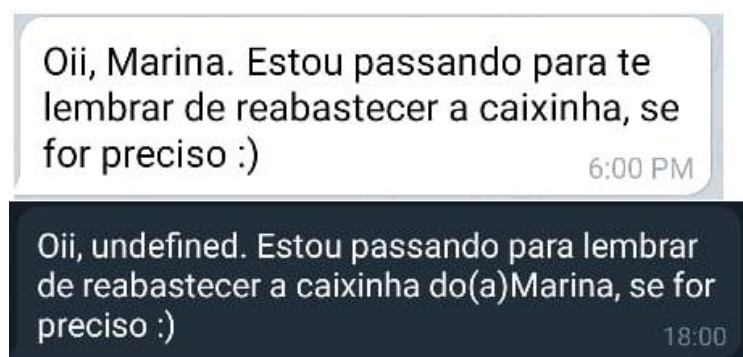
também foram mostrados conforme o programado, ou seja, com as informações iguais as que foram cadastradas, o que pode ser visto nas *dashboards* até então mostradas.

Figura 55 – Mensagens recebidas no Telegram do usuário após primeiro horário do medicamento 1 e do medicamento 2



Fonte: Elaboração própria (2021).

Figura 56 – Mensagem para o usuário e para o tutor referente ao reabastecimento do Meepi



Fonte: Elaboração própria (2021).

Conforme informado na programação do Node-RED, também foram organizadas notificações para lembrar o usuário e o responsável por reabastecer a caixinha. Essa mensagens foram programadas para serem enviadas em horários específicos e em dias específicos. Na Figura 56 é possível observar essas mensagens. Na parte superior se observa a mensagem para o usuário e na parte

inferior para o tutor. Pode-se verificar Figura 56 que houve um problema na captura do nome do tutor, o qual era para aparece no lugar de *undefined*. Contudo, essa questão foi posteriormente corrigida e estava ocorrendo devido ao nome da variável que contém o nome do responsável estar escrita incorretamente nesse script do Node-RED.

Além dos alertas no Telegram e das atualizações da *dashboard*, a caixa Meepi também apresentou os alertas sonoros e visuais adequadamente, a qual se pode observar na Figura 57. Nessa figura, todos os LEDs aparecem ligados visto que foi indicado no cadastro da medicação que ela estaria localizada em todos os compartimentos. Foi observado nos testes subsequentes que o primeiro LED do primeiro compartimento por vezes não ligava. Esse erro pode estar relacionado a mal contato que pode existir na conexão ou a uma presença de um ruído na hora de ligar o primeiro LED. É importante destacar que apenas esse LED em específico não ligava adequadamente.

Figura 57 – Caixa Meepi em funcionamento



Fonte: Elaboração própria (2021).

Os testes foram executados das 10h05min até as 22h10min. Durante esse período, a caixa Meepi tocou 7 vezes para medicação 1 e 3 vezes para a medicação 2. Em todas as repetições os horários foram atualizados na *dashboard* de maneira correta e o celular do responsável e do tutor receberam as notificações pertinentes. As anomalias notadas foram citadas anteriormente, as quais estão relacionadas aos

LEDs dos compartimentos e a ausência do nome do responsável na mensagem para reabastecimento da caixinha.

No Apêndice B se pode encontrar as imagens da *dashboard* além das mostradas no presente capítulo, referente aos outros 8 horários em que houve atualizações. Já as imagens referente ao *Telegram* podem ser observadas no Apêndice C e no D.

8 CONCLUSÃO E TRABALHOS FUTUROS

Acredita-se que os objetivos específicos anteriormente definidos foram atingidos no presente trabalho de conclusão de curso, uma vez que foi realizada uma pesquisa de mercado para entender melhor o contexto que esse potencial produto estaria inserido, foi desenvolvido o conceito do produto e, por fim, foi possível a montagem de um protótipo acessível destinado ao público idoso. Por consequência, o objetivo principal também foi atingido, visto que foram aplicados conhecimentos de eletrônica, mecânica e programação no desenvolvimento deste trabalho.

Apesar das funcionalidades definidas no capítulo 5.3 terem sido atendidas, acredita-se que para o protótipo em questão ser testado com o público, ou seja, ser considerado um MVP de fato, ainda são necessários alguns ajustes, como uma melhor fixação dos componentes eletrônicos dentro da caixa, melhoramento da estrutura em relação ao sistema de bateria e uma implementação de um sistema de *back-up* dos dados na caixinha em caso de perda de energia ou finalização da carga na bateria.

Atualmente, se o porta-medicamentos Meepi é reiniciado ou perde a alimentação, os dados armazenados nas variáveis iniciam no seu valor padrão também. Isso implica na necessidade de o usuário ir novamente ao site recadastrar todos os medicamentos novamente, para assim sincronizar na caixa. Para solucionar esse problema de maneira que não exija intervenções na parte eletrônica, pode-se salvar os dados na EEPROM da própria ESP32 e resgatá-los quando for necessário e desejado.

É importante ressaltar que após diversos testes, o motor de vibração utilizado parou de funcionar. Ao final do projeto se observou que na realidade o motor em questão estava sendo alimentado com uma tensão acima do necessário, o que colaborou com a diminuição da sua vida útil. Diante disso, uma possibilidade de

melhoria seria a mudança no projeto eletrônico da alimentação do motor de vibração para porta de 3,3V da ESP32. Outra opção seria a escolha de um outro modelo de componente no mercado compatível com a tensão atualmente utilizada no projeto, no caso, 5V.

Um aspecto interessante que não foi implementado também e que se configura como uma oportunidade de trabalho futuro, é a integração do Meepi com assistentes de voz, como IBM Watson, Assistente da Google ou Alexa da Amazon.

REFERÊNCIAS

AARP. The Longevity Economy Outlook. p. 31, dez. 2019.

AIOLFI, C. R. et al. Adesão ao uso de medicamentos entre idosos hipertensos. **Revista Brasileira de Geriatria e Gerontologia**, v. 18, n. 2, p. 397–404, 2015.

AITKEN, M. et al. The Global Use of Medicine in 2019 and Outlook to 2023. **IQVIA Institute for Human Data Science**, n. January, p. 1–56, 2019.

AMAZON. **LiveFine Automatic Pill Dispenser, 28-Day Electronic Medication Organizer with Alarm Reminders, Flashing Light and Safety Lock - Dispenses Prescriptions Up to 6 Times Per Day - Clear Lid**. Disponível em: <<https://www.amazon.com/LiveFine-Automatic-Dispenser-Electronic-Medication/dp/B00FL620B0>>. Acesso em: 15 mar. 2021.

ARDUINO.CC. **ARDUINO UNO REV3**. Disponível em: <<https://store.arduino.cc/usa/arduino-uno-rev3>>. Acesso em: 10 mar. 2021.

ARDUINO E CIA. **Como utilizar um encoder rotativo com Arduino**. Disponível em: <<https://www.arduinoocia.com.br/como-usar-encoder-rotativo-ky-040-arduino/>>. Acesso em: 23 mar. 2021.

BACK, N. et al. **Projeto Integrado de Produtos - Planejamento, Concepção e Modelagem**. 1. ed. Rio de Janeiro: Manole, 2008.

CINTRA, F. A.; GUARIENTO, M. E.; MIYASAKI, L. A. Adesão medicamentosa em idosos em seguimento ambulatorial. **Ciência & Saúde Coletiva**, v. 15, n. suppl 3, p. 3507–3515, 2010.

CIRCUITS4YOU. **ESP32 DevKit ESP32-WROOM GPIO Pinout**. Disponível em: <<https://circuits4you.com/2018/12/31/esp32-devkit-esp32-wroom-gpio-pinout/>>. Acesso em: 16 mar. 2021.

ESPRESSIF SYSTEMS. **ESP32**. Disponível em: <<https://www.espressif.com/en/products/socs/esp32>>. Acesso em: 16 mar. 2021.

FILIFEFLOP. **Relógio com o módulo RTC DS1307**. Disponível em: <<https://www.filipeflop.com/blog/relogio-rtc-ds1307-arduino/>>. Acesso em: 22 mar. 2021.

FILIFEFLOP. **Módulo WiFi ESP32 Bluetooth**. Disponível em: <<https://www.filipeflop.com/produto/modulo-wifi-esp32-bluetooth/>>. Acesso em: 16 mar. 2021a.

FILIFEFLOP. **Raspberry Pi Zero W**. Disponível em: <<https://www.filipeflop.com/produto/raspberry-pi-zero-w-anatel-com-header/>>. Acesso em: 16 mar. 2021b.

FILIFEFLOP. **Placa Uno R3 + Cabo USB para Arduino**. Disponível em: <<https://www.filipeflop.com/produto/placa-uno-r3-cabo-usb-para-arduino/>>. Acesso em: 16 mar. 2021c.

FILIFEFLOP. **Módulo Carregador de Bateria de Lítio TP4056**. Disponível em: <<https://www.filipeflop.com/produto/modulo-carregador-de-bateria-de-litio-tp4056/>>. Acesso em: 17 mar. 2021d.

FILIFEFLOP. **Conversor Boost DC Ajustável Step Up**. Disponível em: <<https://www.filipeflop.com/produto/conversor-boost-dc-step-up/>>. Acesso em: 17 mar. 2021e.

IBM. **O que é a plataforma IBM Cloud?** Disponível em: <<https://cloud.ibm.com/docs/overview?topic=overview-what-is-platform&locale=pt-BR>>. Acesso em: 25 mar. 2021.

INTERFARMA. Guia 2019. p. 1–38, 2019.

KENNETH TERRELL. **Americans 50 and Older are Growing Economic Powerhouse**. Disponível em: <<https://www.aarp.org/politics-society/advocacy/info-2019/older-americans-economic-impact-growth.html>>. Acesso em: 8 dez. 2020.

LOJA OCÉANE. **Porta Comprimidos em Estojo**. Disponível em: <<https://www.oceane.com.br/pill-organizer-estojo-porta-comprimidos/>>. Acesso em: 8 dez. 2020.

MAAHS, L.; SERRATE, S. Uso de medicamentos por idosos em região do sul do Brasil. **Revista de Saúde Pública**, v. 39, n. 6, 2005.

MAGAZINE LUIZA. **Porta Comprimidos Semanal Diário - Kit Girl**. Disponível em: <<https://www.magazineluiza.com.br/porta-comprimidos-semanal-diario-kit-girl/p/jec35k8bhe/cp/porc/>>. Acesso em: 15 mar. 2021.

MINISTÉRIO DA SAÚDE. **Estatuto do Idoso**. 3. ed. Brasília: Ministério da Saúde, 2013.

OPEN JS FOUNDATION. **About : Node-RED**. Disponível em: <<https://nodered.org/about/>>. Acesso em: 24 mar. 2021.

ORGANIZAÇÃO MUNDIAL DA SAÚDE. **WHO global estimates on prevalence of hearing loss**. Disponível em: <<https://www.who.int/deafness/estimates/en/>>. Acesso em: 14 jan. 2021.

OTTAIANO, J. A. A. et al. **As Condições de Saúde Ocular no Brasil 2019**. 1. ed. São Paulo: Conselho Brasileiro de Oftalmologia, 2019.

PERISSÉ, C.; MARLI, M. Caminhos para uma melhor idade. **Retratos**, v. 16, 2019.

PILLGO. **Pillgo Smart Pillbox**. Disponível em: <<https://www.pillgo.co/en/pillgo-smart-pillbox/>>. Acesso em: 8 dez. 2020.

RED HAT. **O que é Internet das Coisas (IoT)?** Disponível em: <<https://www.redhat.com/pt-br/topics/internet-of-things/what-is-iot>>. Acesso em: 22 abr. 2021.

RIES, E. **A Startup enxuta: como os empreendedores atuais utilizam a inovação**

contínua para criar empresas extremamente bem-sucedidas. São Paulo: Lua de Papel, 2012. v. 1

ROCHA, C. H. et al. Adesão à prescrição médica em idosos de Porto Alegre, RS. **Ciência & Saúde Coletiva**, v. 13, n. SUPPL. 0, p. 703–710, abr. 2008.

SILVA, A. M. R. .; VIDEIRA, C. A. E. **UML, Metodologias e Ferramentas CASE**. 2. ed. Lisboa: Centro Atlântico, 2008.

SOUSA, S. et al. Polimedicação em doentes idosos: Adesão à terapêutica. **Revista Portuguesa de Clínica Geral**, v. 27, n. 2, p. 176–182, 2011.

WALICKI, J.; IBM. **Create an Internet of Things Platform Starter application**. Disponível em: <<https://developer.ibm.com/technologies/iot/tutorials/how-to-create-an-internet-of-things-platform-starter-application/>>. Acesso em: 17 mar. 2021.

WITTERMYER, R.; JASON TOWNS. **Why America Needs More Innovation in Eldercare**. Disponível em: <<https://www.inc.com/renee-wittemyer-jason-towns/why-america-needs-more-innovation-in-eldercare.html>>. Acesso em: 13 set. 2020.

YUAN, M.; IBM. **Conhecendo o MQTT**. Disponível em: <<https://developer.ibm.com/br/technologies/iot/articles/iot-mqtt-why-good-for-iot/>>. Acesso em: 25 mar. 2021.

APÊNDICE A – Questionário do Google Forms

Responda esta pesquisa e ajude uma universitária :)

Olá pessoal! Meu nome é Marina, estudo Engenharia Mecatrônica no Instituto Federal de Santa Catarina (IFSC) e estou fazendo esta pesquisa para validar algumas hipóteses para o meu Trabalho de Conclusão de Curso. Gostaria de entender como é o seu hábito de consumo de medicamentos, pois estou desenvolvendo uma solução na área. Vale ressaltar que os dados adquiridos neste formulário poderão ser utilizados no desenvolvimento do meu trabalho.

Se você lembrou de alguém quando viu este formulário, pode compartilhar com essa pessoa também. Todas as respostas são bem vindas, principalmente daqueles que consomem medicamentos com frequência.

Ah e se você tem algum familiar que não sabe mexer muito bem no celular ou no computador, aproveita: liga para ele e responda este questionário com as respostas dele também.

Agradeço imensamente!

***Obrigatório**

1. 1. Qual sua faixa de idade? *

Observação sobre as alternativas: De 10 até 20 anos. 21 até 30 anos / 31 até 40 anos/ De 41 até 50 anos/ De 51 até 60 anos/ De 61 até 70 anos e 71 anos ou mais.

Marcar apenas uma oval.

- 10 - 20 anos
- 20 - 30 anos
- 30 - 40 anos
- 40 - 50 anos
- 50 - 60 anos
- 60 - 70 anos
- 70 ou mais

2. 2. Você toma algum medicamento diariamente? *

Marcar apenas uma oval.

Sim

Não

3. 3. Quantos remédios diferentes você ingere diariamente? *

Ex: Por dia eu tomo 2 remédios, o remédio x e o remédio y.

Marcar apenas uma oval.

0

1

2

3

4

5

6 ou mais

4. 4. Quantas vezes ao dia você toma algum medicamento? *

Ex: Em dois horários do dia estou tomando algum dos meus medicamento. Tomo 2 remédios, um deles 8h da manhã e o outro as 20h, logo 2 vezes ao dia estou tomando algum medicamento.

Marcar apenas uma oval.

0

1 vez

2 vezes

3 vezes

4 ou mais

5. 5. Quantos comprimidos (ou cápsulas) você toma por dia no total? *

Marcar apenas uma oval.

- 0
 1
 2
 3
 4
 5
 6 ou mais

6. 6. Quais tipos de medicamento você toma diariamente? *

Você pode selecionar mais de uma opção para essa pergunta

Marque todas que se aplicam.

- Comprimido
 Cápsula
 Líquido
 Efervescente
 Nenhum

Outro: _____

7. 7. Quais as razões do consumo desses medicamentos continuamente? *

Você pode selecionar mais de uma opção para essa pergunta

Marque todas que se aplicam.

- Diabetes
- Hipertensão
- Reumatismo/Artrite
- Tratamento de enxaqueca
- Memória
- Câncer
- Suplementação alimentar
- Depressão, ansiedade, pânico, entre outras razões semelhantes
- Contracepção
- Não realizo uso contínuo de medicamentos

Outro: _____

8. 8. Você costuma esquecer de tomar os medicamentos? *

Marcar apenas uma oval.

- Sim
- Não

9. 9. Você costuma ter dúvida se tomou ou não algum medicamento? *

Marcar apenas uma oval.

- Sim
- Não

10. 10. Você costuma tomar seus remédios nos horários corretos? *

Marcar apenas uma oval.

- Sim, consigo tomar todos os remédios nos seus horários adequados
- Não, nem sempre tomo os remédios nos horários adequados

11. 11. Qual método você utiliza para lembrar a ingestão de um determinado remédio? *

Você pode selecionar mais de uma opção para essa pergunta

Marque todas que se aplicam.

- despertador de celular
- aplicativo
- porta remédios
- lembretes em papel
- um familiar/amigo/profissional administra os horários dos meus medicamentos
- memória/força do hábito/Nenhum

Outro: _____

12. 12. Você investiria em uma solução que lhe ajudasse ou ajudasse alguém que você conhece a tomar remédios adequadamente? *

Marcar apenas uma oval.

- Sim
- Não
- Talvez

13. 13. Qual faixa de preço que você investiria em uma solução tecnológica que ajudaria você ou alguém que você conhece a tomar remédios de maneira adequada? *

Marcar apenas uma oval.

- R\$ 50,00 - 100,00
- R\$ 100,00 - 150,00
- R\$ 150,00 - 200,00
- R\$ 200,00 - 250,00
- R\$ 250,00 - 300,00
- Acima de 300,00

APÊNDICE B – *Dashboard* durante os testes

Horário das 12:05 – Medicamento 01

Dashboard

Informações Gerais	Medicação 1	Medicação 2
Olá, Marina	Nome: TESTE 1	Nome: TESTE 2
Ver: 2	Data de início: 31/03/2021	Data de início: 31/03/2021
Última atualização da caixa: 10:03 31/3/2021	Hora de início: 10:05	Hora de início: 10:10
RECADASTRAR REMÉDIOS	Frequência (h): 2	Frequência (h): 6
RECADASTRAR USUÁRIO E TUTOR	Dosagem por horário: 1	Dosagem por horário: 1
	Último horário consumido: 31/3/2021 12:6	Último horário consumido: 31/3/2021 10:11
	Próximo horário: 31/3/2021 14:5	Próximo horário: 31/3/2021 16:10
	Compartimentos: 7 6 5 4 3 2 1	Compartimentos: 7 6 5 4 3 2 1

Horário das 14:05 – Medicamento 01

Dashboard

Informações Gerais	Medicação 1	Medicação 2
Olá, Marina	Nome: TESTE 1	Nome: TESTE 2
Ver: 2	Data de início: 31/03/2021	Data de início: 31/03/2021
Última atualização da caixa: 10:03 31/3/2021	Hora de início: 10:05	Hora de início: 10:10
RECADASTRAR REMÉDIOS	Frequência (h): 2	Frequência (h): 6
RECADASTRAR USUÁRIO E TUTOR	Dosagem por horário: 1	Dosagem por horário: 1
	Último horário consumido: 31/3/2021 14:6	Último horário consumido: 31/3/2021 10:11
	Próximo horário: 31/3/2021 16:5	Próximo horário: 31/3/2021 16:10
	Compartimentos: 7 6 5 4 3 2 1	Compartimentos: 7 6 5 4 3 2 1

Horário das 16:05 – Medicamento 01

Dashboard

Informações Gerais	Medicação 1	Medicação 2
Olá, Marina	Nome: TESTE 1	Nome: TESTE 2
Ver: 2	Data de início: 31/03/2021	Data de início: 31/03/2021
Última atualização da caixa: 10:03 31/3/2021	Hora de início: 10:05	Hora de início: 10:10
RECADASTRAR REMÉDIOS	Frequência (h): 2	Frequência (h): 6
RECADASTRAR USUÁRIO E TUTOR	Dosagem por horário: 1	Dosagem por horário: 1
	Último horário consumido: 31/3/2021 16:5	Último horário consumido: 31/3/2021 10:11
	Próximo horário: 31/3/2021 18:5	Próximo horário: 31/3/2021 16:10
	Compartimentos: 7 6 5 4 3 2 1	Compartimentos: 7 6 5 4 3 2 1

Horário das 16:10 – Medicamento 02

Dashboard

Informações Gerais	Medicação 1	Medicação 2
<p>Olá, Marina</p> <p>Ver: <input type="text" value="2"/></p> <p>Última atualização da caixa: 10:03 31/3/2021</p> <p style="text-align: center;">RECADASTRAR REMÉDIOS</p> <p style="text-align: center;">RECADASTRAR USUÁRIO E TUTOR</p>	<p>Nome: TESTE 1</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:05</p> <p>Frequência (h): 2</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 16:5</p> <p>Próximo horário: 31/3/2021 18:5</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>	<p>Nome: TESTE 2</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:10</p> <p>Frequência (h): 6</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 16:10</p> <p>Próximo horário: 31/3/2021 22:10</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>

Horário das 18:05 – Medicamento 01

Dashboard

Informações Gerais	Medicação 1	Medicação 2
<p>Olá, Marina</p> <p>Ver: <input type="text" value="2"/></p> <p>Última atualização da caixa: 10:03 31/3/2021</p> <p style="text-align: center;">RECADASTRAR REMÉDIOS</p> <p style="text-align: center;">RECADASTRAR USUÁRIO E TUTOR</p>	<p>Nome: TESTE 1</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:05</p> <p>Frequência (h): 2</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 18:6</p> <p>Próximo horário: 31/3/2021 20:5</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>	<p>Nome: TESTE 2</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:10</p> <p>Frequência (h): 6</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 16:10</p> <p>Próximo horário: 31/3/2021 22:10</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>

Horário das 20:05 – Medicamento 01

Dashboard

Informações Gerais	Medicação 1	Medicação 2
<p>Olá, Marina</p> <p>Ver: <input type="text" value="2"/></p> <p>Última atualização da caixa: 10:03 31/3/2021</p> <p style="text-align: center;">RECADASTRAR REMÉDIOS</p> <p style="text-align: center;">RECADASTRAR USUÁRIO E TUTOR</p>	<p>Nome: TESTE 1</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:05</p> <p>Frequência (h): 2</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 20:6</p> <p>Próximo horário: 31/3/2021 22:5</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>	<p>Nome: TESTE 2</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:10</p> <p>Frequência (h): 6</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 16:10</p> <p>Próximo horário: 31/3/2021 22:10</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>

Horário das 22:05 – Medicamento 01

Dashboard

Informações Gerais	Medicação 1	Medicação 2
<p>Olá, Marina</p> <p>Ver: <input type="text" value="2"/></p> <p>Última atualização da caixa: 10:03 31/3/2021</p> <p style="text-align: center;">RECADASTRAR REMÉDIOS</p> <p style="text-align: center;">RECADASTRAR USUÁRIO E TUTOR</p>	<p>Nome: TESTE 1</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:05</p> <p>Frequência (h): 2</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 22:6</p> <p>Próximo horário: 1/4/2021 0:5</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>	<p>Nome: TESTE 2</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:10</p> <p>Frequência (h): 6</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 16:10</p> <p>Próximo horário: 31/3/2021 22:10</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>

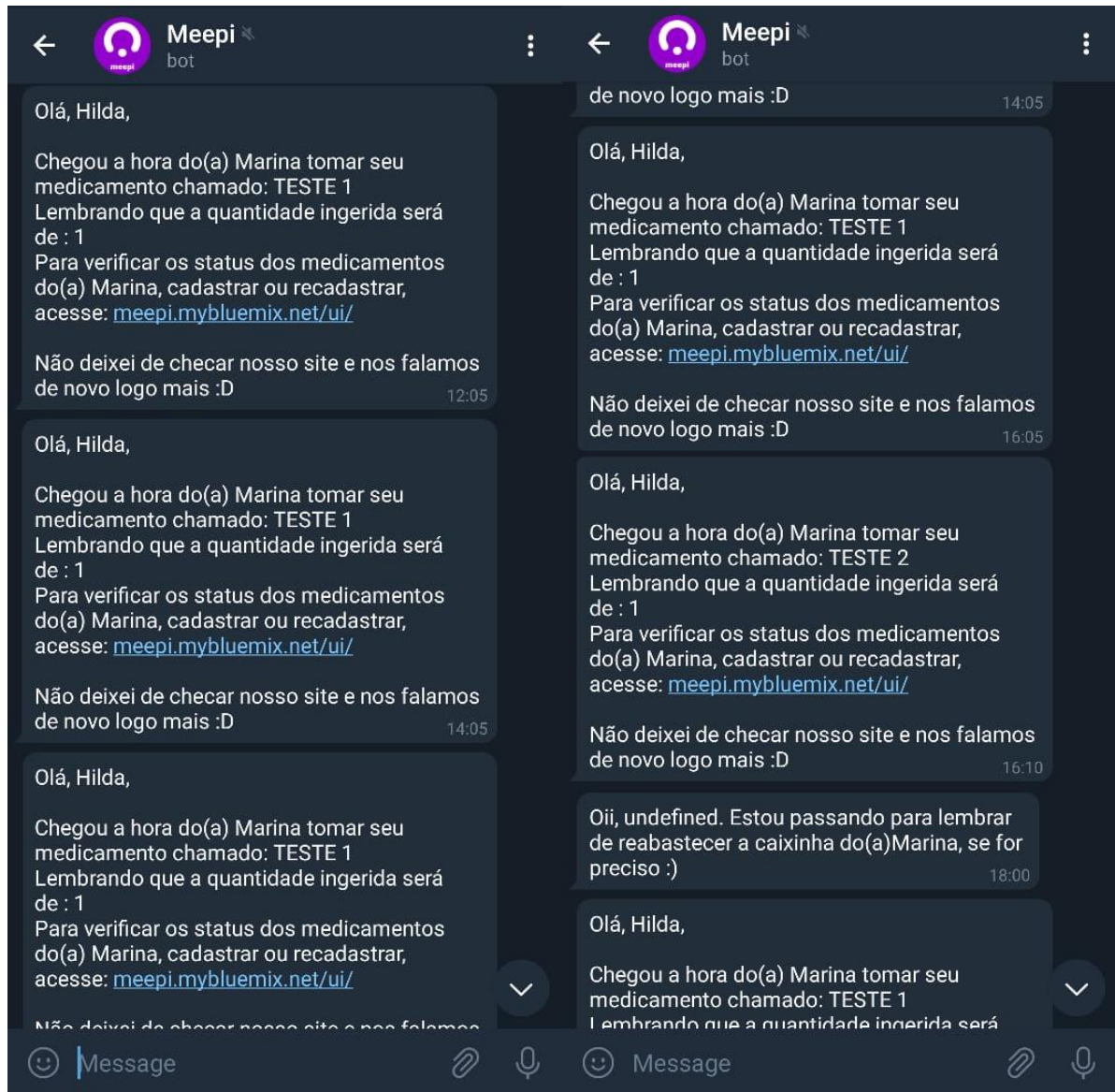
Horário das 22:10 – Medicamento 02

Dashboard

Informações Gerais	Medicação 1	Medicação 2
<p>Olá, Marina</p> <p>Ver: <input type="text" value="2"/></p> <p>Última atualização da caixa: 10:03 31/3/2021</p> <p style="text-align: center;">RECADASTRAR REMÉDIOS</p> <p style="text-align: center;">RECADASTRAR USUÁRIO E TUTOR</p>	<p>Nome: TESTE 1</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:05</p> <p>Frequência (h): 2</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 22:6</p> <p>Próximo horário: 1/4/2021 0:5</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>	<p>Nome: TESTE 2</p> <p>Data de início: 31/03/2021</p> <p>Hora de início: 10:10</p> <p>Frequência (h): 6</p> <p>Dosagem por horário: 1</p> <p>Último horário consumido: 31/3/2021 22:11</p> <p>Próximo horário: 1/4/2021 4:10</p> <p>Compartimentos: 7 6 5 4 3 2 1</p>

APÊNDICE C – Mensagens recebidas no Telegram do usuário (testes)

Notificações das 12:05, 14:05, 16:05 e 18:00



Notificações das 18:05, 20:05 e 22:05

The image shows a screenshot of a WhatsApp chat interface with a bot named 'Meepi'. The chat is displayed in a grid of three rows, each representing a notification. Each notification message is identical in structure and content, with the only difference being the time stamp at the bottom right of the message bubble.

Notification 1 (18:05):

Olá, Hilda,
Chegou a hora do(a) Marina tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade ingerida será de : 1
Para verificar os status dos medicamentos do(a) Marina, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/
Não deixei de checar nosso site e nos falamos de novo logo mais :D

Notification 2 (20:05):

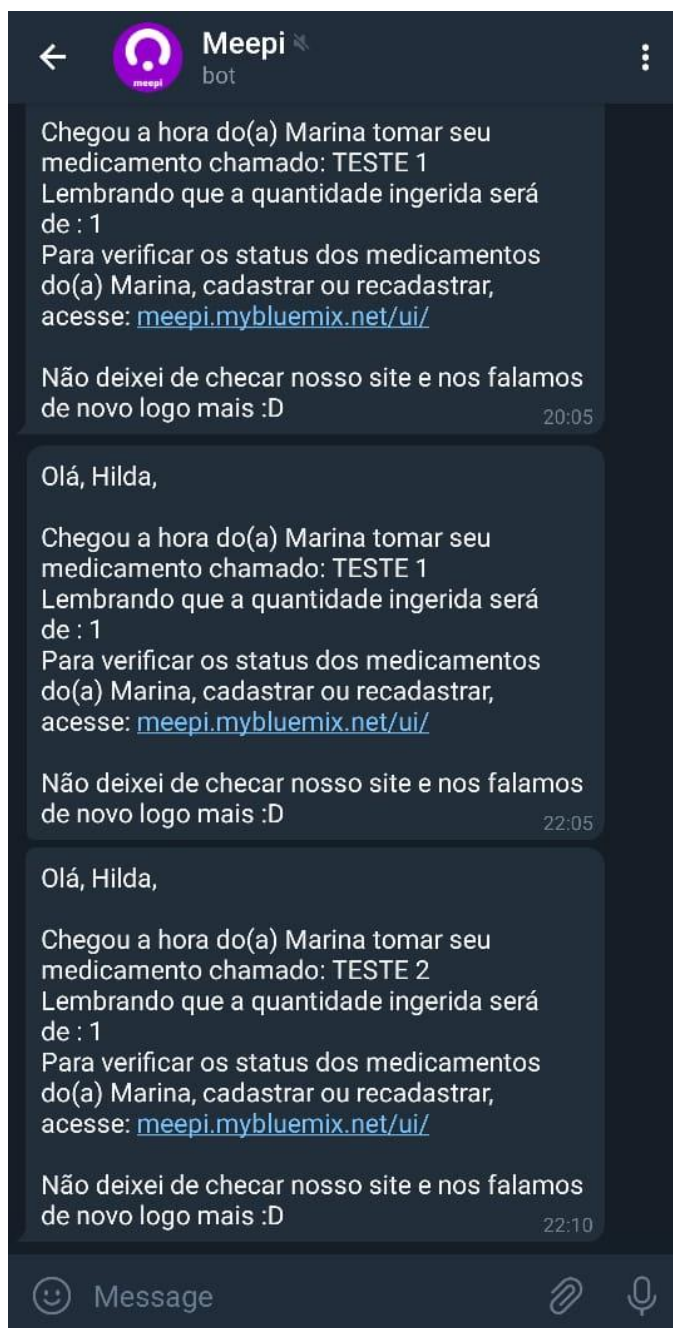
Olá, Hilda,
Chegou a hora do(a) Marina tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade ingerida será de : 1
Para verificar os status dos medicamentos do(a) Marina, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/
Não deixei de checar nosso site e nos falamos de novo logo mais :D



Notification 3 (22:05):

Olá, Hilda,
Chegou a hora do(a) Marina tomar seu medicamento chamado: TESTE 2
Lembrando que a quantidade ingerida será de : 1
Para verificar os status dos medicamentos do(a) Marina, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/
Não deixei de checar nosso site e nos falamos de novo logo mais :D

The chat interface includes a back arrow, the bot's profile picture and name 'Meepi bot', and a message input field at the bottom with a 'Message' placeholder and icons for emojis, attachments, and voice recording.

Notificação das 22:10



←  **Meepi** 
bot

Chegou a hora do(a) Marina tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade ingerida será de : 1
Para verificar os status dos medicamentos do(a) Marina, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

Não deixei de checar nosso site e nos falamos de novo logo mais :D 20:05

Olá, Hilda,




Chegou a hora do(a) Marina tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade ingerida será de : 1
Para verificar os status dos medicamentos do(a) Marina, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

Não deixei de checar nosso site e nos falamos de novo logo mais :D 22:05

Olá, Hilda,

Chegou a hora do(a) Marina tomar seu medicamento chamado: TESTE 2
Lembrando que a quantidade ingerida será de : 1
Para verificar os status dos medicamentos do(a) Marina, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

Não deixei de checar nosso site e nos falamos de novo logo mais :D 22:10

 Message  

APÊNDICE D – Mensagens recebidas no Telegram do tutor (testes)

Notificações das 12:05 e 14:05 (2:05 PM)



Notificações das 16:05 (4:05 PM), 16:10 (4:10 PM) e 18:00 (6:00 PM)

The image shows a WhatsApp chat interface with the 'Meepi bot'. The chat is split into two panels. The left panel shows messages from the bot at 4:05 PM and 6:00 PM. The right panel shows messages from the bot at 4:10 PM and 6:00 PM, plus a user response at 6:00 PM.

4:05 PM (Left Panel):

Olá, Marina,
 Chegou a hora de tomar seu medicamento chamado: TESTE 1
 Lembrando que a quantidade que você deve tomar é de: 1
 E seu medicamento de agora está localizado no(s) compartimento(s): 7 6 5 4 3 2 1
 Não deixe para depois, tome seu(s) remédio(s) agora mesmo. Não se esqueça de apertar no botão <<ok>> antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário
 Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

4:10 PM (Right Panel):

Olá, Marina,
 Chegou a hora de tomar seu medicamento chamado: TESTE 2
 Lembrando que a quantidade que você deve tomar é de: 1
 E seu medicamento de agora está localizado no(s) compartimento(s): 7 6 5 4 3 2 1
 Não deixe para depois, tome seu(s) remédio(s) agora mesmo. Não se esqueça de apertar no botão <<ok>> antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário
 Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

6:00 PM (Right Panel):

Oii, Marina. Estou passando para te lembrar de reabastecer a caixinha, se for preciso :)

6:00 PM (Left Panel):

Olá, Marina,
 Chegou a hora de tomar seu medicamento chamado: TESTE 1

The bottom of the chat shows the standard WhatsApp input area with a 'Message' label and icons for emojis, attachments, voice recording, and a microphone.

Notificações das 18:05 (6:05 PM) e 20:05 (8:05 PM)

WhatsApp chat interface with the Meepi bot. The chat header shows the Meepi bot profile. The messages are as follows:

Message 1 (6:05 PM):

Olá, Marina,

Chegou a hora de tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade que você deve tomar é de: 1
E seu medicamento de agora está localizado no(s) compartimento(s): 7 6 5 4 3 2 1

Não deixe para depois, tome seu(s) remédio(s) agora mesmo. Não se esqueça de apertar no botão <<ok>> antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário

Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

Message 2 (6:05 PM):

antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário

Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

Message 3 (8:05 PM):

Olá, Marina,

Chegou a hora de tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade que você deve tomar é de: 1
E seu medicamento de agora está localizado no(s) compartimento(s): 7 6 5 4 3 2 1

Não deixe para depois, tome seu(s) remédio(s) agora mesmo. Não se esqueça de apertar no botão <<ok>> antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário

Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

The bottom of the chat shows the standard WhatsApp input area with a 'Message' label and icons for emojis, attachments, voice recording, and a microphone.

Notificações das 22:05 (10:05 PM) e 22:10 (10:05 PM)

The image shows two screenshots of a WhatsApp chat with a bot named 'Meepi'. The chat interface includes a back arrow, the bot's profile picture and name, and a three-dot menu. The messages are white bubbles on a blue background. The first screenshot shows a reminder for 'TESTE 1' medication, including instructions on when to take it and a link to check medication status. The second screenshot shows a reminder for 'TESTE 2' medication with similar instructions and a link. The bottom of the image shows the WhatsApp input area with a 'Message' placeholder and various icons for emojis, attachments, voice recording, and voice messages.

WhatsApp chat with Meepi bot. The chat contains two messages, each a medication reminder.

Message 1 (10:05 PM):

Olá, Marina,

Chegou a hora de tomar seu medicamento chamado: TESTE 1
Lembrando que a quantidade que você deve tomar é de: 1
E seu medicamento de agora está localizado no(s) compartimento(s): 7 6 5 4 3 2 1

Não deixe para depois, tome seu(s) remédio(s) agora mesmo. Não se esqueça de apertar no botão <<ok>> antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário

Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

Message 2 (10:05 PM):

Olá, Marina,

Chegou a hora de tomar seu medicamento chamado: TESTE 2
Lembrando que a quantidade que você deve tomar é de: 1
E seu medicamento de agora está localizado no(s) compartimento(s): 7 6 5 4 3 2 1

Não deixe para depois, tome seu(s) remédio(s) agora mesmo. Não se esqueça de apertar no botão <<ok>> antes de tomar qualquer remédio e depois de tomar todos os seus medicamentos deste horário

Para verificar os status dos medicamentos, cadastrar ou recadastrar, acesse: meepi.mybluemix.net/ui/

WhatsApp input area: Message, [Grid icon], [Paperclip icon], [Microphone icon], [Voice message icon], Message, [Grid icon], [Paperclip icon], [Voice message icon]