

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ (UTFPR)
CURSO DE ENGENHARIA DE COMPUTAÇÃO

CÁSSIO KEISUKE YAMAUCHI
GUILHERME BARBOZA GOMES, PAULO SÉRGIO ÁVILA JÚNIOR

VMG - *VENDING MACHINE A GRANEL*

OFICINA DE INTEGRAÇÃO 2 – RELATÓRIO FINAL

CURITIBA

2022

CÁSSIO KEISUKE YAMAUCHI
GUILHERME BARBOZA GOMES, PAULO SÉRGIO ÁVILA JÚNIOR

VMG - *VENDING MACHINE A GRANEL*

Relatório Final da disciplina Oficina de Integração 2, do curso de Engenharia de Computação, apresentado aos professores que ministram a mesma na Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção da aprovação na disciplina.

Orientador: Prof. Dr. César Manuel Vargas Benítez
Prof. Dr. Heitor S. Lopes

CURITIBA

2022

RESUMO

. VMG - *VENDING MACHINE A GRANEL*. 70 f. Oficina de Integração 2 – Relatório Final – Curso de Engenharia de Computação, UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ (UTFPR). Curitiba, 2022.

O projeto tratado neste documento consiste no desenvolvimento de uma *vending machine* que possa ser utilizada para auxiliar as vendas de produtos *a granel*. É possível usufruir do serviço da VMG com um aplicativo para celulares móveis, de maneira a ser conveniente e de fácil acesso. Para a construção de tal, foram realizados os projetos de hardware, software e mecânica, posteriormente integrando todas essas partes. Ao final, foi possível obter uma máquina que atendia os objetivos iniciais e suas especificações.

Palavras-chave: vending, machine, granel

ABSTRACT

. VMG - *VENDING MACHINE A GRANEL*. 70 f. Oficina de Integração 2 – Relatório Final – Curso de Engenharia de Computação, UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ (UTFPR). Curitiba, 2022.

The project object of this report consists in the development of a bulk vending machine designed for assisting in the selling of nuts. Paired with an app, it is possible to connect to the VMG via Bluetooth, offering three options of nuts: peanuts, cashew nuts and Brazil nuts. For its construction, the projects of the mechanical structure, hardware and software were realized and later integrated. The end result was a vending machine that could execute the tasks initially specified.

Keywords: vending, machine, bulk

LISTA DE FIGURAS

FIGURA 1	– GPIOs da ESP32	15
FIGURA 2	– Esquemático de uma célula de carga	17
FIGURA 3	– Diagrama de blocos do projeto	23
FIGURA 4	– Modelo 3D da VMG	24
FIGURA 5	– Diagrama de blocos do hardware	25
FIGURA 6	– Esquemático Inicial	27
FIGURA 7	– Esquemático Final	28
FIGURA 8	– PCB para o circuito da placa	29
FIGURA 9	– Disposição dos componentes na PCB	29
FIGURA 10	– Representação 3D da PCB	30
FIGURA 11	– Diagrama de casos de uso do projeto.	31
FIGURA 12	– Diagrama de sequências para o caso de uso UC001.	35
FIGURA 13	– Diagrama de sequências para o caso de uso UC002.	36
FIGURA 14	– Diagrama de estados e transições do firmware da máquina.	37
FIGURA 15	– Diagrama visual da tela de conexão à máquina.	37
FIGURA 16	– Diagrama visual da tela de adição de créditos.	38
FIGURA 17	– Diagrama visual da tela de escolha de produtos.	38
FIGURA 18	– Diagrama visual da tela de seleção de pesos.	39
FIGURA 19	– Diagrama de fluxo de telas do aplicativo.	39
FIGURA 20	– Planejamento dos pacotes Bluetooth.	40
FIGURA 21	– Placa perfurada na máquina	44
FIGURA 22	– Placa perfurada na máquina	45
FIGURA 23	– Placa perfurada na máquina	46
FIGURA 24	– Esquema de cores do cronograma	49
FIGURA 25	– Tarefas planejadas para o entregável 1	49
FIGURA 26	– Tarefas planejadas para o entregável 2	50
FIGURA 27	– Tarefas planejadas para o entregável 3	50
FIGURA 28	– Tarefas planejadas para o entregável 4 - parte 1	51
FIGURA 29	– Tarefas planejadas para o entregável 4 - parte 2	51
FIGURA 30	– Tarefas planejadas para o entregável 5	52
FIGURA 31	– Tarefas planejadas para o entregável final	52
FIGURA 32	– Gráfico de horas trabalhadas e estimadas semanalmente por integrante ...	53

LISTA DE TABELAS

TABELA 1	– Tabela de conexões	26
TABELA 2	– Diagrama de caso de uso 1: UC001	32
TABELA 3	– Especificação do caso de uso 1: UC1, Parte 1	33
TABELA 4	– Especificação do caso de uso 1: UC1, Parte 2	34
TABELA 5	– Tabela do orçamento inicial.	54
TABELA 6	– Tabela de custos de produção e pesquisa.	55

SUMÁRIO

1	INTRODUÇÃO	13
1.1	MOTIVAÇÃO	13
1.2	OBJETIVOS	13
1.2.1	Objetivo geral	13
1.2.2	Objetivos específicos	13
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	PLANEJAMENTO	14
2.2	ESP32	14
2.2.1	Módulo conversor e amplificador	15
2.3	TRANSMISSORES IR	16
2.3.1	LED emissor de infravermelho	16
2.3.2	Fototransistor receptor de infravermelho	16
2.4	MOTORES	16
2.4.1	Motor DC e drivers	16
2.4.2	Servomotores	17
2.4.3	Células de Carga	17
2.4.4	Android	18
2.4.4.1	Android Studio	18
2.4.4.2	Kotlin	18
2.4.5	Bluetooth	18
3	METODOLOGIA	20
3.1	VISÃO GERAL	20
3.2	PROJETO MECÂNICO	24
3.3	PROJETO DE HARDWARE	24
3.4	PROJETO DE SOFTWARE	31
3.5	INTEGRAÇÃO	41
4	EXPERIMENTOS E RESULTADOS	42
5	CRONOGRAMA E CUSTOS DO PROJETO	49
5.1	CRONOGRAMA	49
5.2	CUSTOS	53
6	CONCLUSÕES	56
6.1	CONCLUSÕES	56
6.2	TRABALHOS FUTUROS	56
	REFERÊNCIAS	58
	Apêndice A – DIAGRAMAS AUMENTADOS	59
	Apêndice B – DIAGRAMAS AUXILIARES E VERSÕES INICIAIS	62

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

Máquinas de vendas propiciam muita comodidade no dia-a-dia do homem moderno, oferecendo serviços rápidos e sempre disponíveis mesmo em horários de pouco movimento. Uma categoria de produtos pouco explorada pelas máquinas de venda atuais é a dos produtos alimentícios vendidos *a granel*, sendo eles grãos, castanhas e cereais. Existe, portanto, um mercado pouco explorado visando a venda de produtos *a granel* por meio de máquinas de venda automáticas.

1.2 OBJETIVOS

1.2.1 OBJETIVO GERAL

O objetivo da VMG é preencher a lacuna das máquinas de venda por peso atualmente presente no mercado, propiciando a clientes um meio de suprir suas necessidades quanto a produtos *a granel* com atendimento rápido e conveniente a qualquer hora do dia.

1.2.2 OBJETIVOS ESPECÍFICOS

O produto do projeto apresentado tem por objetivos específicos:

- Realizar a venda de produtos *a granel*
- Possibilitar a clientes obterem produtos fora de horários comerciais
- Atender clientes de maneira ágil e conveniente
- Necessitar de pouca manutenção

2 FUNDAMENTAÇÃO TEÓRICA

O desenvolvimento desse projeto fez o uso de conceitos de diversas áreas do conhecimento, vistos dentro e fora do ambiente escolar. Nas sessões seguintes serão detalhados os fundamentos teóricos que foram aplicados no decorrer do projeto.

2.1 PLANEJAMENTO

Um planejamento detalhado e preciso é um componente essencial de um projeto e um dos principais critérios que determinam seu sucesso ou fracasso. Por esse motivo, para a elaboração da VMG, foram utilizadas diversas ferramentas para auxiliar na comunicação e organização da equipe:

- Para a construção e *layout* do cronograma, foi decidido que seria utilizado o Google Docs.
- Para a construção e manutenção do blog, foi decidido que seria utilizado o Notion.
- Para a construção do diagrama de blocos, foi utilizado o site miro.com
- Para a construção dos diagramas de sequência, diagrama de estados e fluxograma, foi utilizado os softwares grátis de diagramas diagrams.net e draw.io
- Para a construção do planejamento da estrutura, foi utilizado o Blender
- Para a construção dos esquemáticos de hardware e design de pcb, foi utilizado o software livre para EDA, Kicad.

2.2 ESP32

A ESP32 é um microprocessador de alta performance e baixo custo utilizado em diversos projetos que requerem um módulo wi-fi ou bluetooth. O módulo NODEMCU ESP32 foi utilizado, o qual apresenta diversos recursos, como um processador Xtensa 32-bit LX6 Dual

Core, 520Kbytes de memória RAM, 36 Portas GPIO com diversas funções alternativas e um módulo Bluetooth BLE 4.2. Na Figura 1, há uma representação da pinagem do ESP32, com detalhamento dos GPIOs e suas funcionalidades padrões.

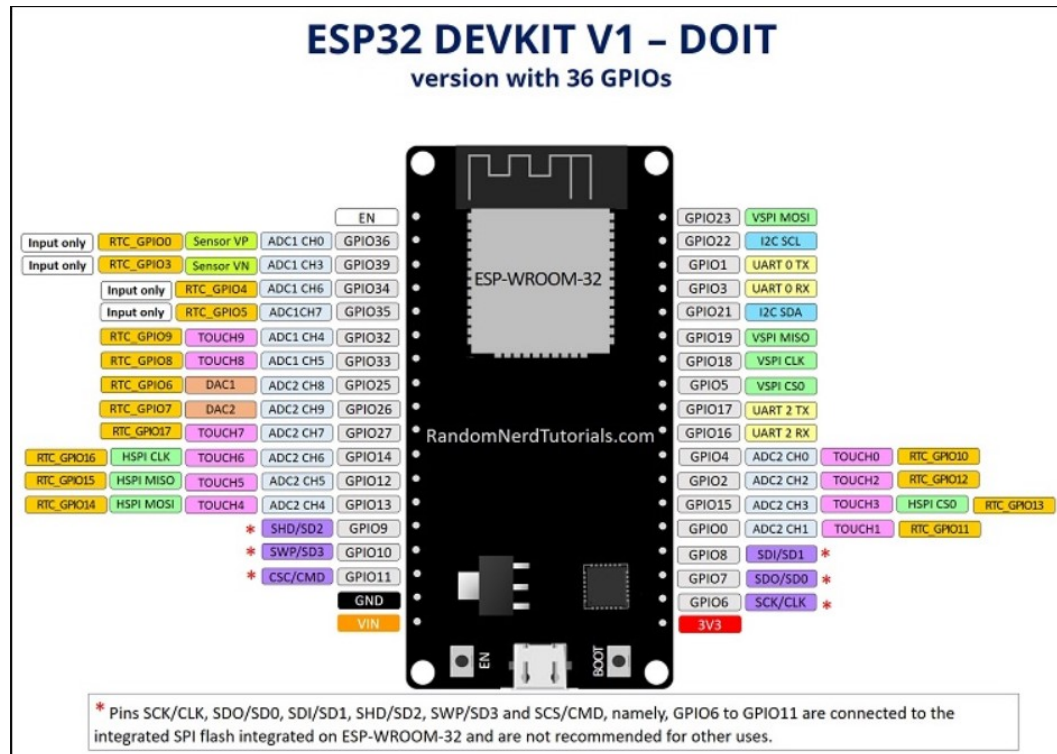


Figura 1: GPIOs da ESP32

2.2.1 MÓDULO CONVERSOR E AMPLIFICADOR

No projeto, foi utilizado um circuito conversor AD e amplificador para as células de carga. O tópico de amplificadores foi abordado na disciplina de Eletrônica Geral 1 e aprofundado na disciplina de Eletrônica Geral 2. Já o tópico de conversores foi abordado em circuitos digitais e aprofundado em microcontroladores.

Um amplificador consiste, de maneira simplificada, em um componente que irá amplificar um sinal elétrico recebido (CODELLOS, 2013). O quociente entre a tensão de saída e de entrada é denominado ganho. Ao amplificar o sinal, o amplificador pode ocasionar uma distorção harmônica no sinal de saída. Por exemplo, ao receber uma onda senoidal perfeita, a saída pode apresentar discrepâncias em relação a entrada. Quanto menor a distorção harmônica obtida, maior a qualidade do amplificador. Ao utilizar amplificadores em áudio, para a qualidade de som ser boa, a distorção harmônica deve ser inferior a 1% (CODELLOS, 2013).

Um conversor AD consiste em transformar o sinal analógico em um sinal digital para que seja possível tratar as alterações decorrentes das deformações nos sensores de carga. Assim,

é possível ler de maneira consideravelmente precisa o peso dos produtos sobre a balança, uma vez que o conversor utilizado possui uma precisão de 24 bits. (SEMICONDUCTOR,)

Para o presente projeto, foi utilizado o HX711. Esse circuito é comumente utilizado quando se é utilizado células de carga pela sua facilidade de uso e pela documentação já existente de bibliotecas públicas disponíveis no GitHub.

2.3 TRANSMISSORES IR

2.3.1 LED EMISSOR DE INFRAVERMELHO

LEDs emissores de infravermelho são LEDs capazes de emitir ondas na frequência do infravermelho. Utilizando esse tipo de onda, é possível identificar se certa frequência foi utilizada por um receptor para codificar sinais, como em TVs, ou simplesmente para identificar se existe algum objeto opaco entre o emissor e o receptor (detector de presença), como utilizado neste projeto.

2.3.2 FOTOTRANSISTOR RECEPTOR DE INFRAVERMELHO

Fototransistores receptores de infravermelho são transistores que ativam sua base quando recebem a frequência para determinado infravermelho configurado. Assim, quando existe alguma onda infravermelha suficiente forte que ativa sua base, o fototransistor permite a passagem de corrente.

2.4 MOTORES

2.4.1 MOTOR DC E DRIVERS

Motores DC e seu interfaceamento com microcontroladores foi um dos tópicos estudados na matéria de Sistemas Microcontrolados. Esses motores são atuadores, que transformam energia elétrica em energia mecânica. São cargas analógicas, ou seja, o torque e a velocidade variam conforme se varia a tensão e a corrente sobre ele (WIKIPEDIA, 2021). Para serem acionados e controlados por meio de um microcontrolador, há a necessidade de um driver de corrente, pois sua alta demanda de corrente e o fato de ser uma carga indutiva podem causar danos ao micro controlador.

Em nosso projeto, como o motor DC utilizado é um motor de vibração (simplesmente um motor DC com uma carga desbalanceada em seu eixo), o driver de corrente utilizado foi um

transistor PN2222a, já que não há a necessidade de rotacionar para ambos os lados.

2.4.2 SERVOMOTORES

Servomotores, estudados na disciplina Sistemas Microcontrolados, são motores DC realimentados em malha fechada para os quais o posicionamento depende de um sinal de entrada, como por exemplo a porcentagem de tempo em nível alto de um PWM em uma frequência específica (WIKIPEDIA, 2020).

2.4.3 CÉLULAS DE CARGA

Existem diversos tipos de células de carga. Desde do tipo canister, até os de pressão. Para este projeto, foram utilizados os sensores de carga do tipo viga de flexão, os quais são os mais utilizados. Esse tipo de sensor funciona por meio de pontes de Wheatstone. Seu funcionamento consiste em converter a carga que age sobre ela em sinais elétricos. O próprio sensor é colado em uma viga ou em um elemento estrutural que se deforma quando peso é aplicado. Na maioria dos casos, quatro sensores de deformação e tensão são usados para obter máxima sensibilidade e compensação de temperatura. Normalmente, dois dos sensores estão em tensão e dois em compressão, sendo conectados com os ajustes de compensação. Quando o peso é aplicado, a deformação mecânica altera a resistência elétrica dos medidores de forma proporcional à carga usada. (OMEGA,). Esse circuito pode ser visto na Figura 2.

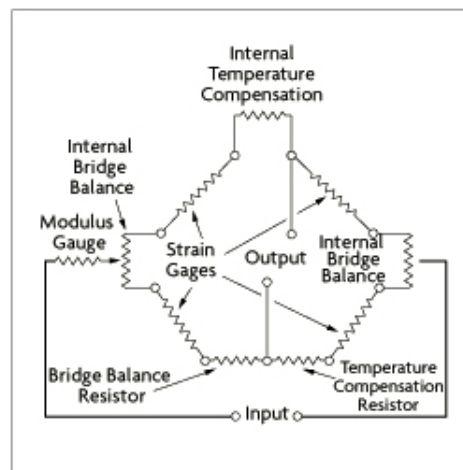


Figura 2: Esquemático de uma célula de carga

2.4.4 ANDROID

Neste projeto, o aplicativo da VMG foi feito para o sistema operacional móvel Android, utilizando da linguagem Kotlin. Esse sistema operacional é famosamente *open-source* e tem grande compatibilidade entre versões, propiciando uma ampla cobertura de dispositivos-alvo. Além disso, ao contrário do iOS, da Apple, desenvolver e publicar aplicativos para Android pode ser completamente gratuito, sem a necessidade de comprar licenças caras.

2.4.4.1 ANDROID STUDIO

Para o desenvolvimento do aplicativo foi utilizada a IDE Android Studio, uma ferramenta gratuita disponibilizada pela Google e especificamente projetada para auxiliar no desenvolvimento para plataformas Android.

O Android Studio contém diversas ferramentas úteis que auxiliam no desenvolvimento de aplicativos, como um emulador, debugger, editor gráfico de telas, verificação de gramática e sintaxe em tempo real, entre outros recursos. Anunciado em 2013, ele foi construído em cima da IDE IntelliJIDEA, outra IDE muito famosa utilizada para desenvolvimento em Java, e disponibilizado em 2014 para o público em geral.

2.4.4.2 KOTLIN

A linguagem Kotlin substituiu o Java em 2019 como linguagem oficial da Google para desenvolvimento em Android. Uma mistura de Python com Java, essa linguagem foi desenvolvida de modo a ser concisa, segura, expressiva, interoperacionável e multi-plataforma, atributos que rapidamente a levaram a ser uma das linguagens mais utilizadas para desenvolvimento de aplicativos no mundo atualmente. Fácil de aprender e muito potente, o Kotlin é visto como uma evolução do Java em diversos aspectos quando se trata de desenvolvimento para plataformas que utilizam da JVM (Java Virtual Machine).

2.4.5 BLUETOOTH

Visto que um dos requisitos da VMG era estabelecer uma conexão única e de pouca distância entre aplicativo e máquina, onde uma máquina pode se conectar a apenas um aplicativo e vice-versa, além de ser uma ferramenta relativamente simples, amplamente reconhecida e de fácil utilização, foi escolhido o Bluetooth como método de conexão no projeto.

O Bluetooth é um meio de comunicação de curto alcance que utiliza de ondas

de rádio para transmitir informações entre até 7 dispositivos próximos, dependendo do dispositivo. A comunicação é feita principalmente pelo processo de *pareamento*, onde dois dispositivos, cada um com seu próprio endereço de 48-bits, se reconhecem e permitem que uma conexão seja estabelecida, possibilitando que seja realizada a criptografia de dados transmitidos entre quaisquer dois dispositivos sem esforço qualquer por parte do usuário ou programador(GOOGLE, 2021)(WIKIPEDIA, 2022).

3 METODOLOGIA

3.1 VISÃO GERAL

Antes de iniciar o desenvolvimento do projeto foram definidos seus requisitos funcionais e não funcionais, de maneira a possibilitar que fosse feito um planejamento mais adequado ao projeto e de certificar que os objetivos da VMG seriam cumpridos extensivamente. Esses requisitos foram então divididos entre dois grupos: Requisitos da Máquina, que compõem os requisitos que dependem majoritariamente da estrutura mecânica, e Requisitos do Aplicativo, que compõem aqueles requisitos que podem ser cumpridos separadamente da VMG a partir de um aplicativo de celular. A seguir estão listados os Requisitos Funcionais, Não-Funcionais e Anti-Requisitos extensivamente.

Requisitos Funcionais da Máquina

- **RF01:** A máquina deve entregar os produtos escolhidos pelo cliente com o peso determinado pelo cliente.
- **RF02:** A máquina deve entregar todos os produtos em uma singular embalagem colocada no lugar correto pelo cliente.
- **RF03:** A máquina deve pesar o produto que sai das caixas, de maneira a saber o quanto está despejando.
- **RF04:** A máquina deve pesar o produto que entra na embalagem, de maneira a saber o quanto está sendo entregue ao cliente.
- **RF05:** A máquina deve detectar a presença de uma embalagem que possa receber os produtos, utilizando um sensor de proximidade.
- **RF06:** A máquina deve interromper o funcionamento e soltar um alarme visual caso seja detectada uma falha no sistema, baseando-se no sensor de proximidade da embalagem, na balança da embalagem e na balança da caixa.

- **RF07:** A máquina deve impedir o acesso à embalagem durante a dispensa de um produto.
- **RF08:** A máquina deve permitir ao usuário encaixar uma embalagem em uma posição específica para que os produtos sejam despejados no lugar correto.
- **RF09:** A máquina deverá concluir o processo em andamento no caso de queda de energia, utilizando-se de uma bateria interna.
- **RF10:** A máquina deve possuir um sistema de alimentação reserva em caso de queda de energia.
- **RF11:** A máquina deve ter um timer e um LED para indicar que o produto em determinada caixa deve ser trocada.

Requisitos Não Funcionais da Máquina

- **RN01:** A máquina não deve ter erro de pesagem dos produtos excedente a 10g.
- **RN02:** A máquina deve ter resolução de dispensa de produtos de 20g.
- **RN03:** A máquina deve ser alimentada por meio de fonte.
- **RN04:** A máquina deve vender os seguintes produtos: Castanha do Pará, Amendoim e Castanha de Caju.
- **RN05:** A máquina deverá sempre dispensar um peso pelo menos 10g maior do que o pedido pelo cliente, de modo a evitar que eventuais erros deixem o cliente insatisfeito.

Anti-Requisitos da Máquina

- **AR01:** A máquina não venderá produtos líquidos ou pastosos.
- **AR02:** A máquina não fechará as embalagens.
- **AR03:** A máquina não pode vender um produto caso não seja inserida uma embalagem na saída.

Requisitos Funcionais do Aplicativo

- **RF12:** O aplicativo deve permitir ao cliente escolher os produtos que deseja comprar.

- **RF13:** O aplicativo deve permitir ao cliente determinar o peso que deseja comprar de cada produto.
- **RF14:** O aplicativo deve permitir que o usuário realize as transições monetárias para adquirir o produto.
- **RF15:** O aplicativo deve bloquear a venda caso a massa requisitada seja maior que a massa total do produto na máquina.
- **RF16:** O aplicativo deve permitir aos usuários adicionar créditos com o uso de códigos promocionais.
- **RF17:** O aplicativo deve armazenar o nome e os créditos dos usuários em um banco de dados.
- **RF18:** O aplicativo deve armazenar e resgatar informações sobre os produtos a serem vendidos em um banco de dados.
- **RF19:** O aplicativo deve bloquear a venda de algum produto caso o timer dele indique que o produto deva ser trocado.
- **RF20:** O aplicativo deve conectar-se a uma singular máquina por vez.

Requisitos Não Funcionais do Aplicativo

- **RNF6:** O aplicativo deve ser de fácil utilização do usuário.
- **RNF7:** O aplicativo deve aceitar apenas pedidos com peso total na faixa de 50 a 500g.
- **RNF8:** O aplicativo deve funcionar em dispositivos Android.
- **RNF9:** O aplicativo deve apenas permitir vendas acima de 100g de cada item.
- **RNF10:** O aplicativo se comunicará com a máquina por meio de bluetooth.
- **RNF11:** As transações monetárias devem ocorrer antes da máquina emitir os produtos e depois do cliente escolher os produtos a serem emitidos.

Anti-Requisitos do Aplicativo

- **AR4:** O aplicativo não deve permitir que mais de um cliente se conecte a uma mesma máquina ao mesmo tempo.

Requisitos Não Funcionais Gerais

- **RNF12:** O banco de dados deve ser SQLite.
- **RNF13:** Os avisos de validade dos produtos, tanto da máquina quanto do aplicativo devem ser ativados após um mês.

Diagrama de Blocos

Para melhorar a visualização do funcionamento do projeto, foi feito um diagrama de blocos (Figura 3). Nele é possível observar como serão integrados os componentes da VMG: a máquina será controlada pela ESP32, que controla os motores de vibração e das portas e coleta informações dos sensores de peso para gerenciar o funcionamento do sistema como um todo. Ela se comunica com um aplicativo via Bluetooth, que é encarregado de servir como a interface cliente-máquina, coletando as informações dos pedidos e as enviando para a ESP32 de modo a permitir a máquina a receber pedidos dos clientes. A caixa ilustrada no diagrama em blocos é modularizada, de maneira que múltiplas caixas podem ser conectadas à máquina para possibilitar a venda de diversos produtos simultaneamente.

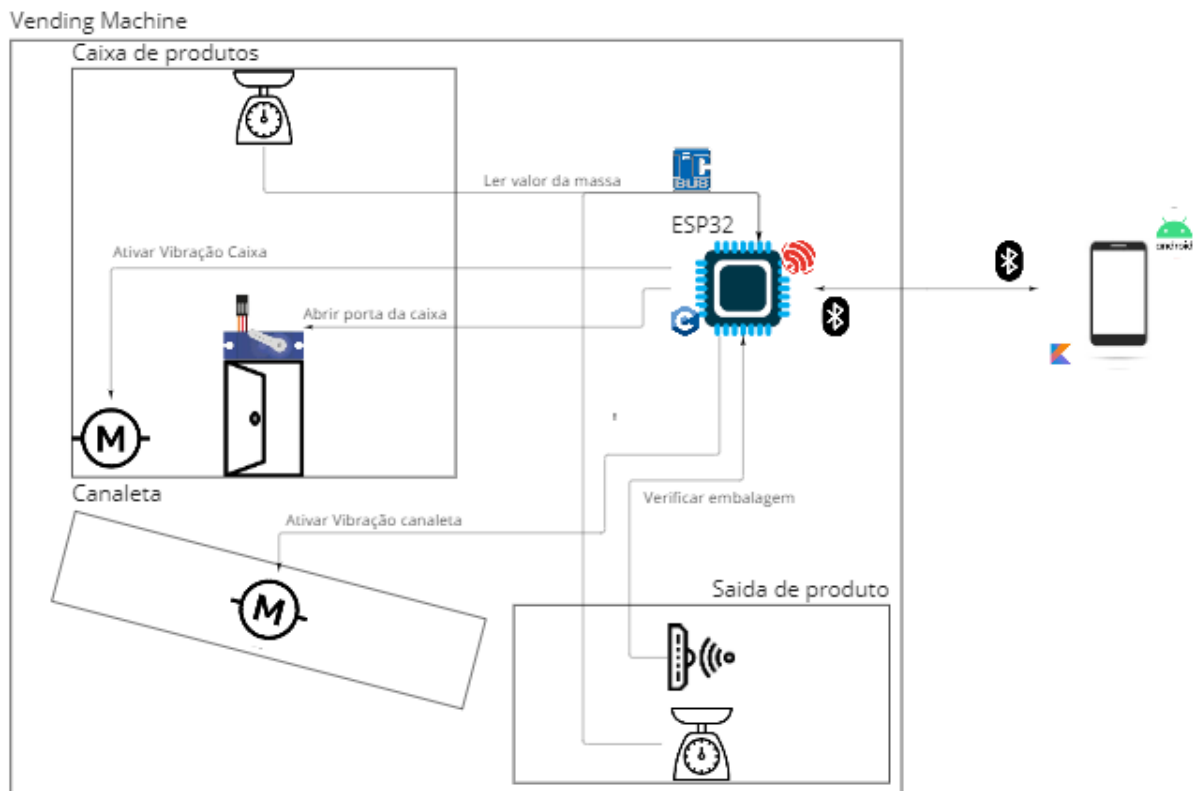


Figura 3: Diagrama de blocos do projeto

3.2 PROJETO MECÂNICO

Foi utilizado, como base mecânica uma estrutura feita de canos de PVC, madeira e papelão para a estrutura da VMG, pela sua capacidade de ser flexível para alterações, como aumentar a quantidade de produtos que podem ser vendidos.

Foi utilizado também para cada caixa uma estrutura composta por uma caixa organizadora modificada com uma rampa de madeira internamente para direcionar o produto para a saída da caixa e também com uma porta construída sob medida para a saída de produtos da caixa.

Foi feito um modelo em 3D do projeto inicialmente, como pode ser visualizado na figura 4, porém durante o desenvolvimento do projeto, foi necessário alterar o formato das canaletas por onde o produto seria direcionada a partir da saída da caixa ate a embalagem de entrega de produto.

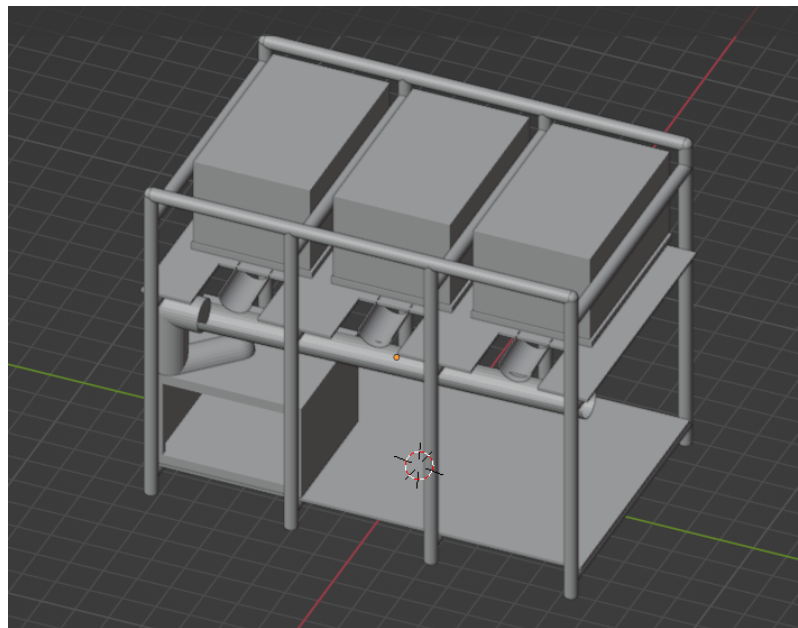


Figura 4: Modelo 3D da VMG

3.3 PROJETO DE HARDWARE

O circuito projetado e construído possui componentes específicos para cada funcionalidade da VMG:

- Conjuntos de 2 balanças de 1kg e 1 módulo conversor amplificador HX711 para fazer a aquisição do valor da massa da caixa de produto e da embalagem.

- Buzzer e LEDs para fazer a sinalização de algum defeito ou falta de produto em uma caixa.
- Servos motores SG-90 para fazer o controle de abertura das portas das caixas.
- LEDs emissores de infravermelho e fototransistores receptores de infravermelho para fazer a verificação da embalagem na região de entrega de produto.
- Motores de vibração e drivers para controlar os motores para melhorar o fluxo de saída de produto da caixa.

Os módulos e componentes utilizados no hardware do projeto são ilustrados na Figura 5.

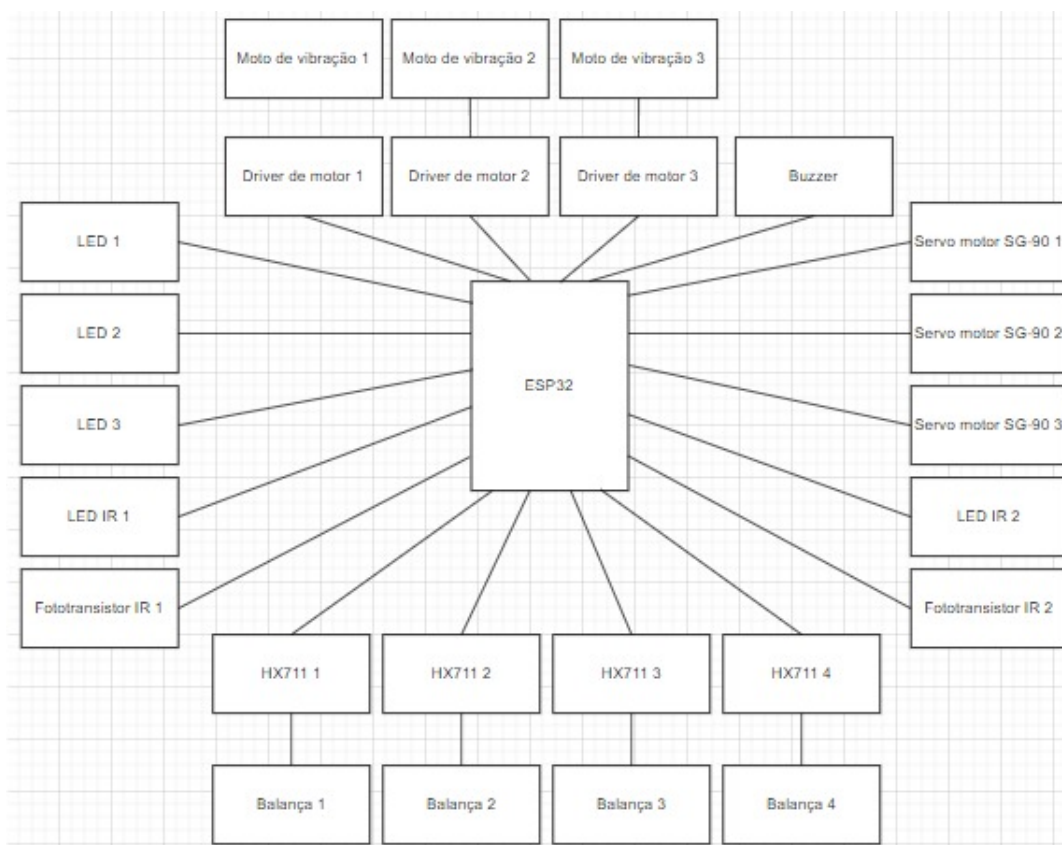


Figura 5: Diagrama de blocos do hardware

Foi utilizado duas formas de alimentação no projeto, a primeira foi uma fonte de 5V e 2A que alimenta os motores, servo motores, o microcontrolador, buzzer, LEDs IR e LEDs. Já a segunda alimentação partiu do regulador de tensão que existe no modulo da ESP32, que possui uma tensão de 3.3V e serve para alimentar os fototransistores e o HX711.

A Tabela 1 apresenta as conexões dos componentes no microcontrolador.

Tabela 1: Tabela de conexões

Sensor	Pino da ESP32
Data HX711 (caixa 1)	4 (GPIO34)
Data HX711 (caixa 2)	8 (GPIO25)
Data HX711 (caixa 3)	9 (GPIO26)
Data HX711 (caixa 4)	10 (GPIO27)
HX711 CLK	13 (GPIO13)
Fototransistor 1	3 (GPIO39)
Fototransistor 2	5 (GPIO35)
Atuador	Pino da ESP32
Motor Vibração (caixa 1)	30 (GPIO23)
Motor Vibração (caixa 2)	29 (GPIO22)
Motor Vibração (caixa 3)	26 (GPIO21)
Servo (caixa 1)	25 (GPIO17)
Servo (caixa 2)	24 (GPIO16)
Servo (caixa 3)	23 (GPIO4)
Outros	Pino da ESP32
LED (caixa 1)	25 (GPIO19)
LED (caixa 2)	24 (GPIO18)
LED (caixa 3)	23 (GPIO5)
LED (caixa 4)	22 (GPIO2)
Buzzer	21 (GPIO15)

Além disso, é possível verificar na Figura 6 o esquemático inicial produzido para o planejamento das conexões conforme o plano do projeto.

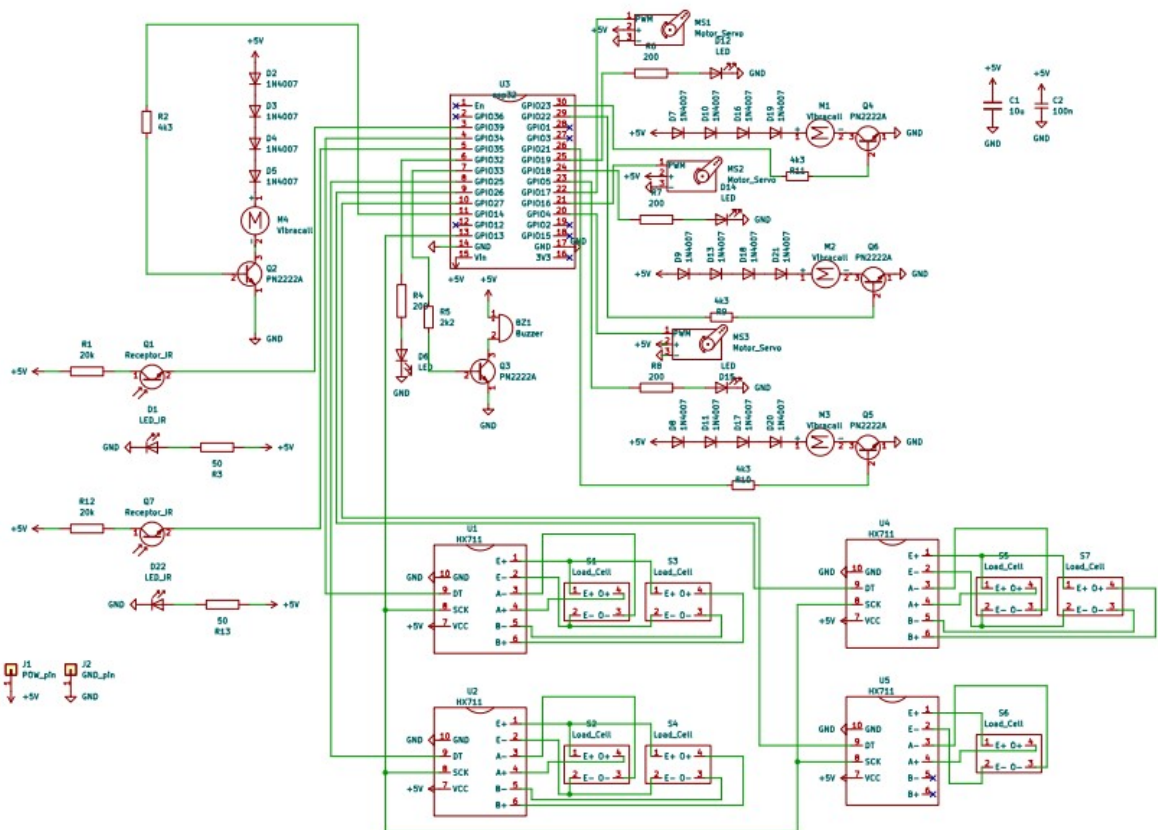


Figura 6: Esquemático Inicial

No entanto, após modificações providas de erros de design e replanejamento da estrutura, o esquemático utilizado para a última etapa do projeto é apresentada na figura 7. A diferença se deve principalmente à remoção de alguns componentes, assim como o uso de pinos fêmea para representar as conexões com os amplificadores HX711, uma vez que a conexão entre as células de carga e esses conversores não ocorre na placa, mas sim por meio de jumpers.

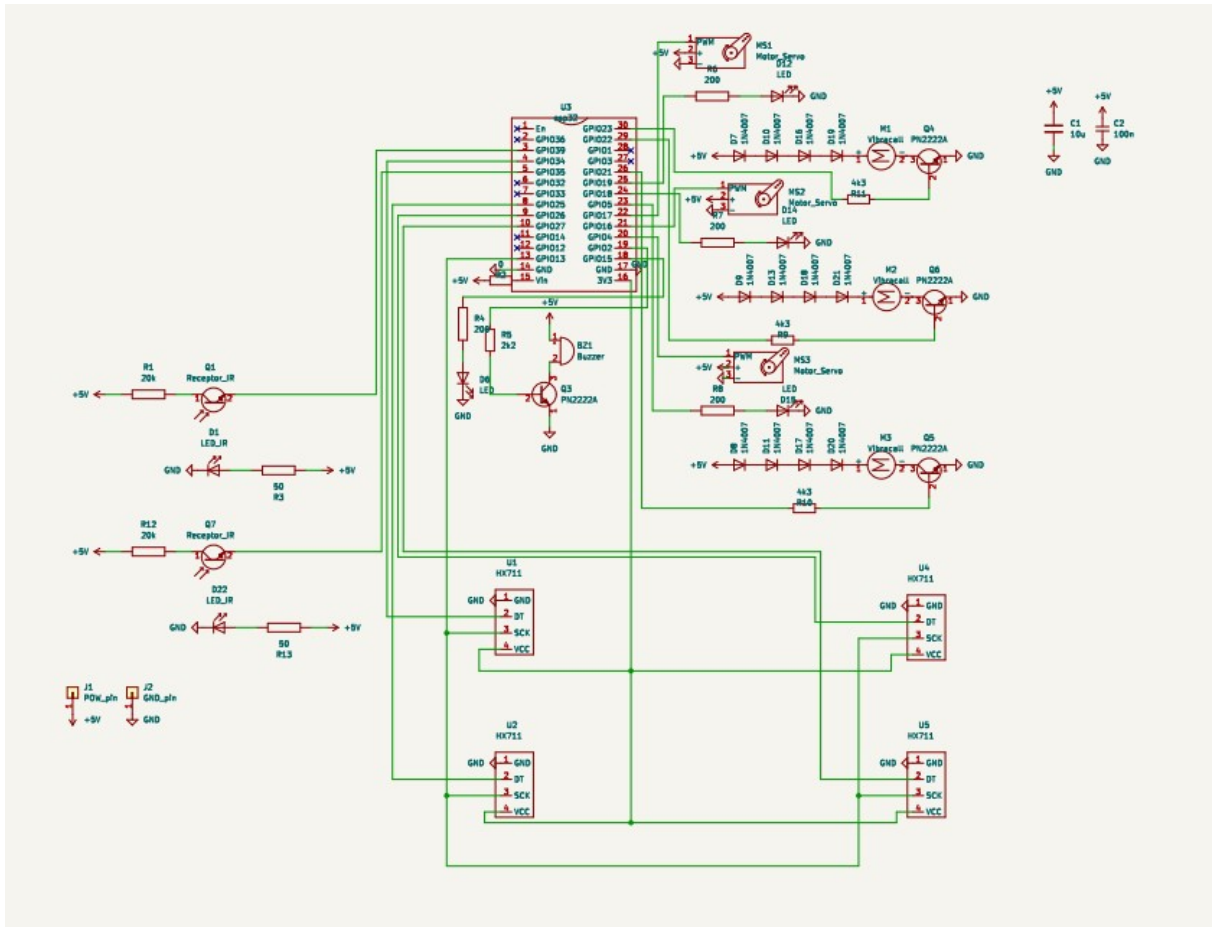


Figura 7: Esquemático Final

Na Figura 8 é possível verificar o design da pcb planejado, na Figura 9 é demonstrado a disposição dos componentes na pcb, e na Figura 10, é representado pela função do Kicad uma prévia de como seria a placa em 3D.

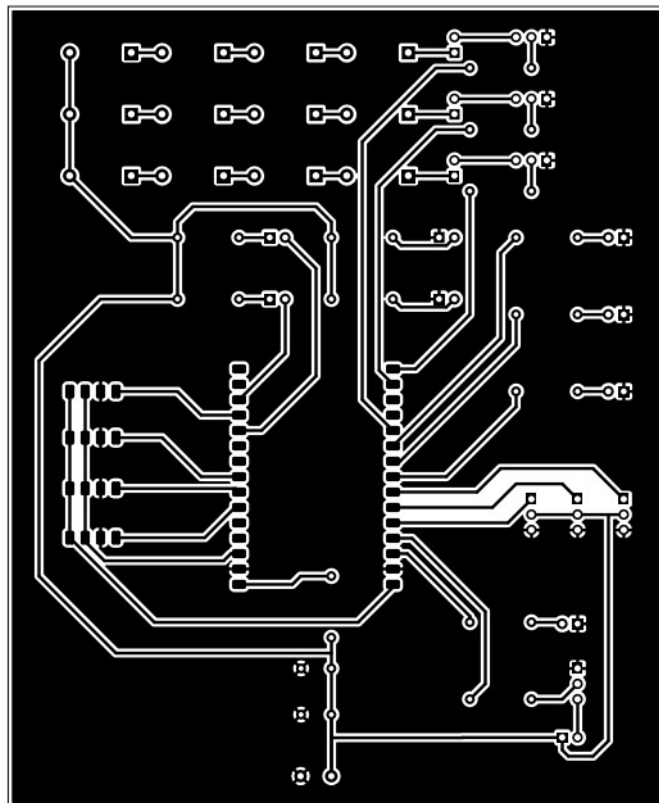


Figura 8: PCB para o circuito da placa

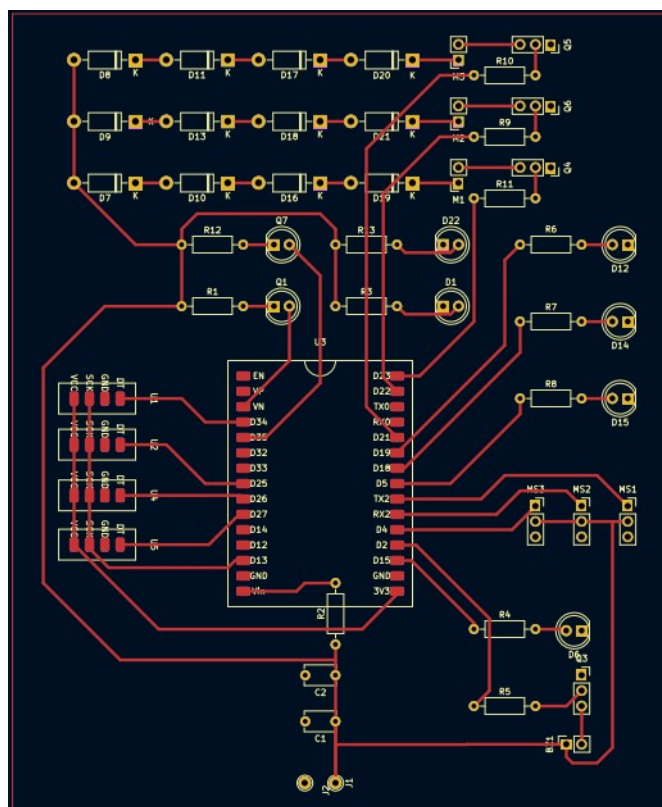


Figura 9: Disposição dos componentes na PCB

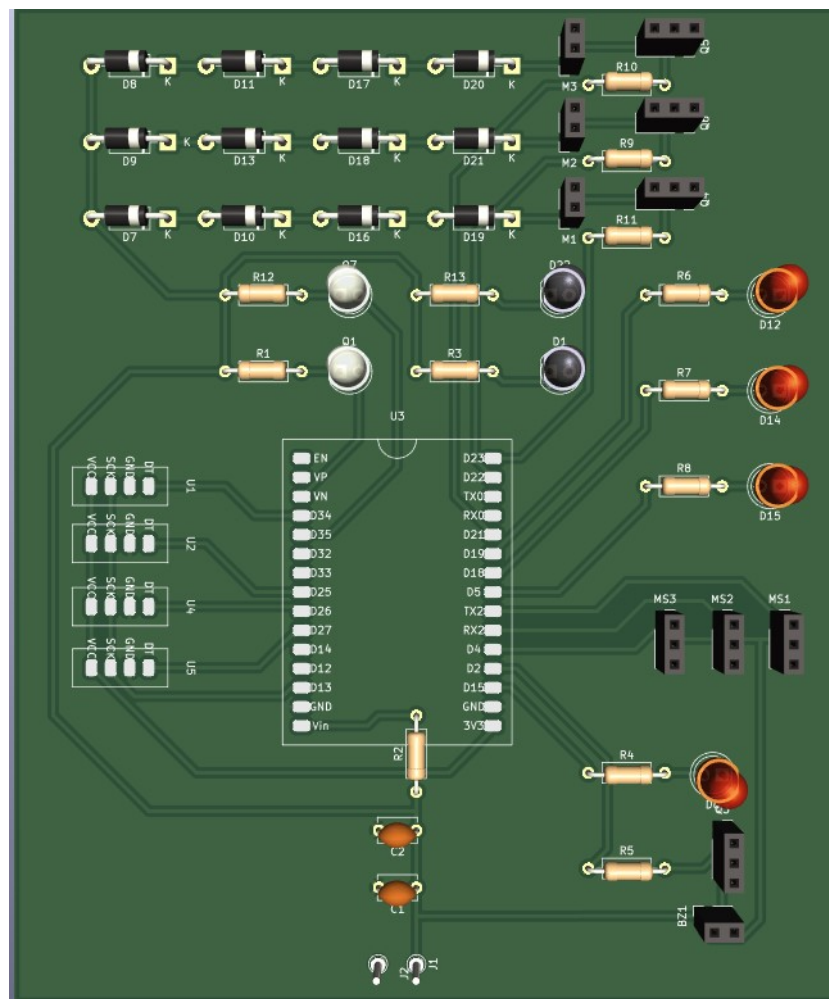


Figura 10: Representação 3D da PCB

3.4 PROJETO DE SOFTWARE

O projeto de *software* foi iniciado a partir da elaboração dos casos de uso da VMG. Foram identificados três atores no sistema: o Cliente, a Máquina e o Aplicativo, e dois casos de uso: Adicionar Créditos e Fazer um Pedido. O diagrama de casos de uso elaborado a partir disso pode ser visto na Figura 11.

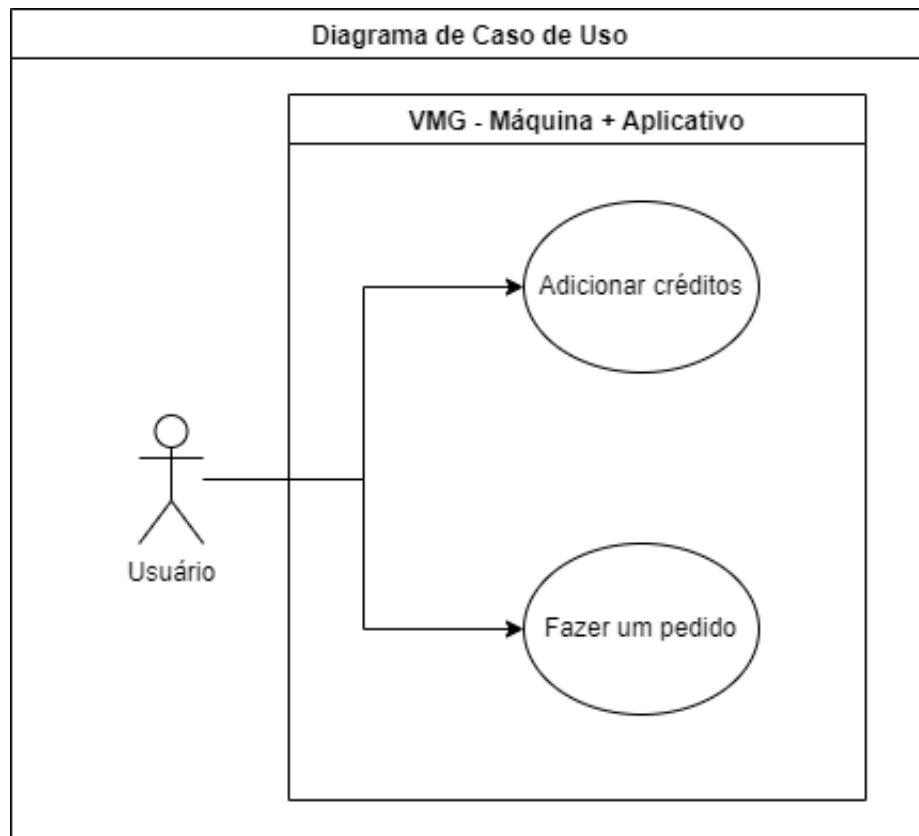


Figura 11: Diagrama de casos de uso do projeto.

A partir deste levantamento foram identificados os requisitos funcionais referentes a cada caso de uso:

- **Fazer um Pedido: RF1-15, RF18-19**
- **Adicionar Créditos: RF16-17**

O caso de uso de Fazer um Pedido foi devidamente identificado e especificado na Tabela 3. O de Adicionar Créditos é simples o suficiente para ser entendido pela Figura 13.

Tabela 2: Diagrama de caso de uso 1: UC001

Nome	Venda de Produtos
Descritor	UC001
Atores	Atores principais: Cliente, Máquina e Aplicativo
Descrição	Caso de uso executado quando é feito um pedido à VMG
Pré-condições	VMG deve estar ligada e funcional
Pós-condições	Cliente recebe os produtos desejados

Tabela 3: Especificação do caso de uso 1: UC1, Parte 1

Fluxo Básico		
Ações do Usuário	Ações da Máquina	Ações do Aplicativo
1. Usuário escolhe comprar produtos		
		2. Aplicativo inicia uma conexão com a Máquina
	3. Máquina conecta ao aplicativo e envia informações dos produtos e de crédito do usuário	
		4. Aplicativo troca para tela de Seleção de Produtos
5. Usuário seleciona os produtos desejados		
		6. Aplicativo troca para tela de seleção de pesos
7. Usuário seleciona os pesos desejados		
		8. Aplicativo troca para tela de confirmação de pesos
9. Usuário confirma o pedido		
		10. Aplicativo envia informações do pedido para a máquina
	11. Máquina recebe o pedido e envia uma confirmação de recebimento	
	12.1. Máquina retira créditos do usuário e emite os produtos sequencialmente	12.2 Aplicativo troca para tela de conexão

Tabela 4: Especificação do caso de uso 1: UC1, Parte 2

Regras de Negócio		
Identificador	RN001	
Nome	Política de entrega para manter opinião positiva do cliente	
Descrição	A VMG deve sempre entregar 10g a mais do que o pedido pelo cliente no mínimo, de modo a suprir qualquer erro de funcionamento e manter a aprovação do cliente.	
Fluxo alternativo 1		
	1. A qualquer momento a máquina encontra um erro, emite um aviso sonoro e trava em estado de tratamento de erro, impede novas conexões e o dinheiro do cliente é devolvido	
2. O cliente pode avisar um funcionário sobre o ocorrido ou tentar realizar o pedido em outra VMG próxima		
Fluxo alternativo 2		
		1. A qualquer momento o aplicativo encontra um erro, avisa o cliente e cancela a operação atual, retornando à tela de conexão
2. O cliente tenta novamente ou desiste do pedido		

Após identificar e especificar os casos de uso para o *software*, foram feitos os diagramas de sequência para cada caso de uso, que podem ser vistos nas Figuras 12 e 13¹

¹Estes diagramas podem ser encontrados em tamanho ampliado no Apêndice A.

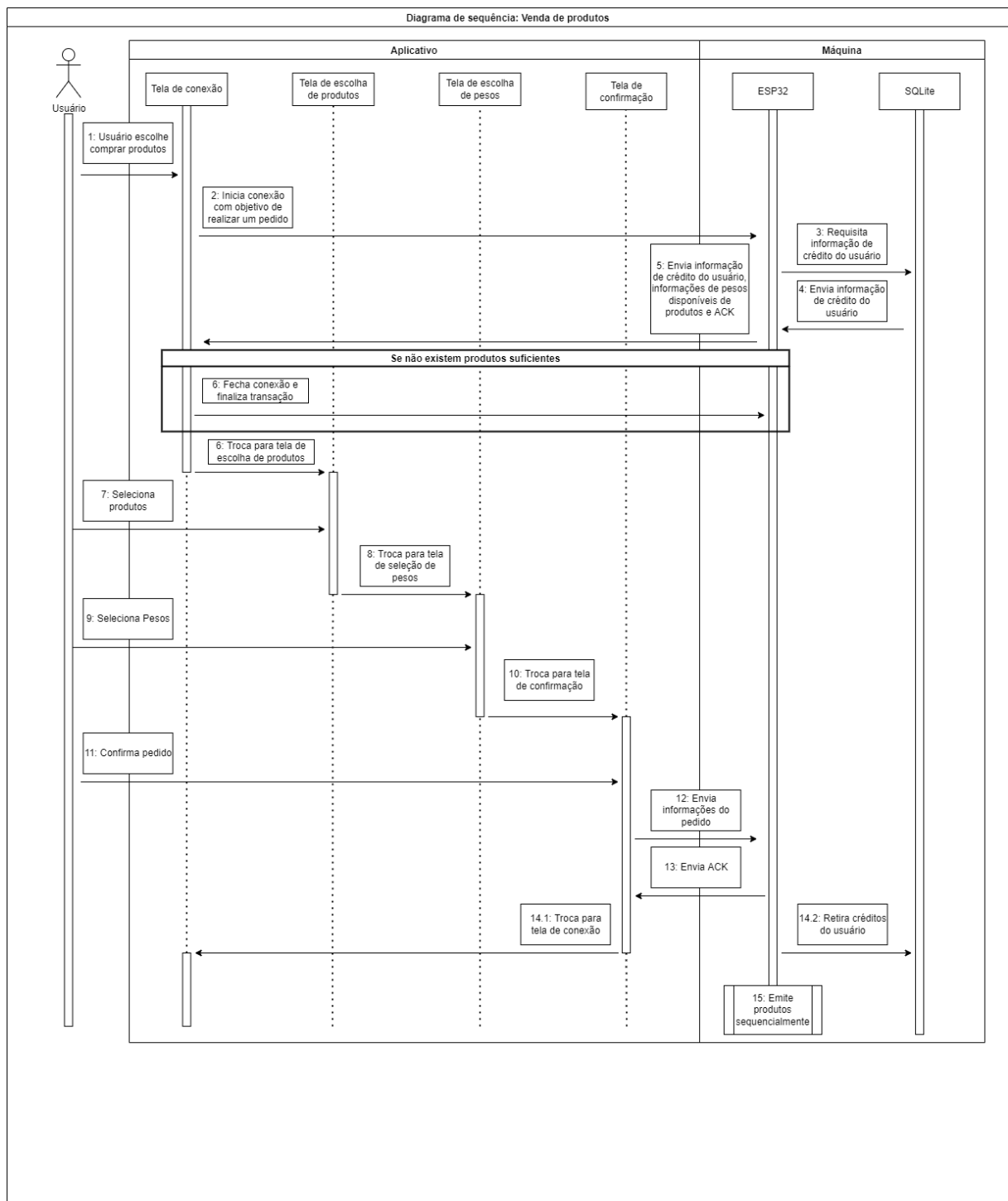


Figura 12: Diagrama de seqüências para o caso de uso UC001.

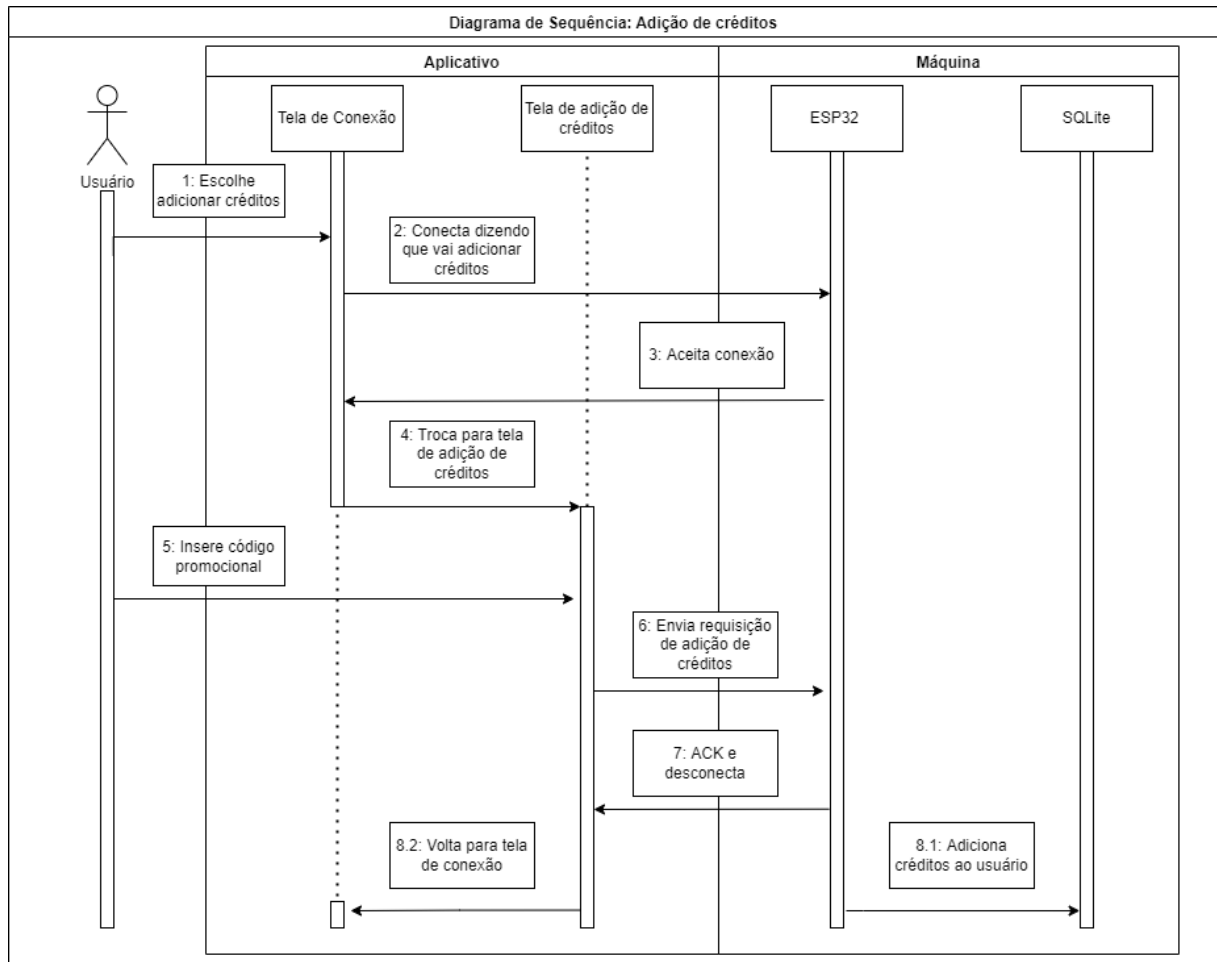


Figura 13: Diagrama de sequências para o caso de uso UC002.

A partir dos diagramas de sequencia foi elaborado o diagrama de estados e transições (figura 14) de modo a planejar a implementação das funcionalidades necessárias no firmware da máquina e diagramas de telas (figuras 15, 16, 17 e 18) e fluxo de telas (Figura 19) para planejar a implementação do aplicativo.

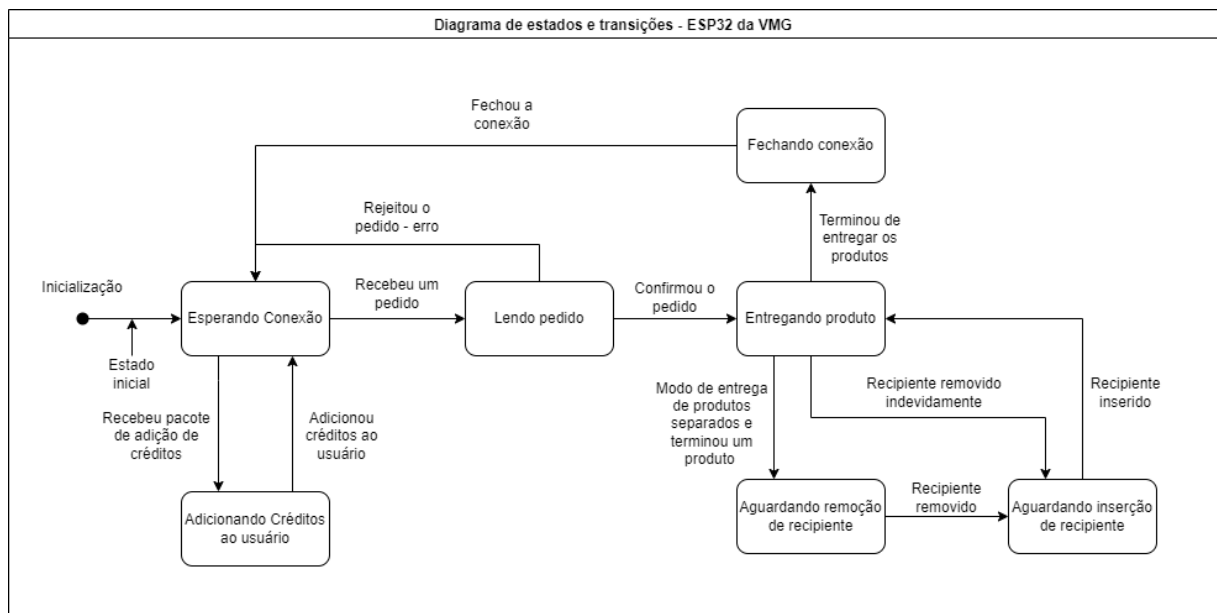


Figura 14: Diagrama de estados e transições do firmware da máquina.

Tela de conexão

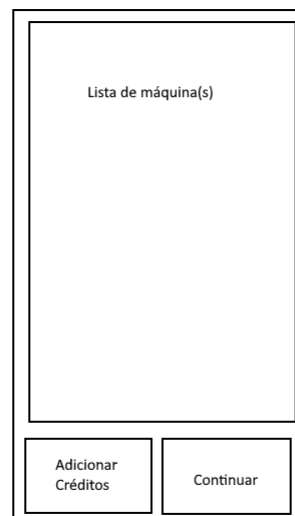
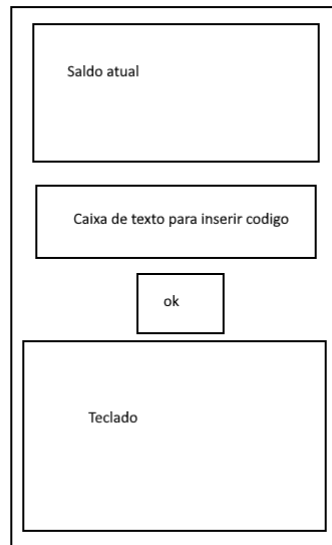
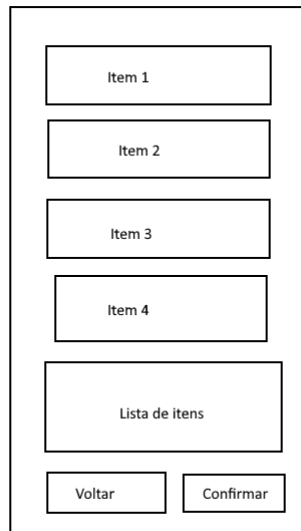


Figura 15: Diagrama visual da tela de conexão à máquina.

Credit management interface

**Figura 16: Diagrama visual da tela de adição de créditos.**

Tela de escolha de item

**Figura 17: Diagrama visual da tela de escolha de produtos.**

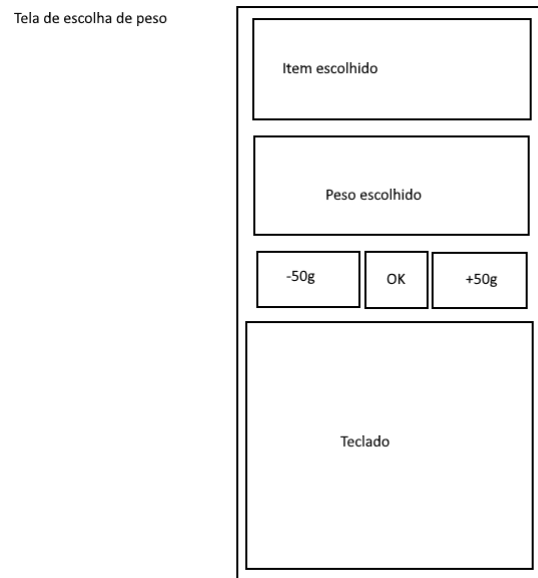


Figura 18: Diagrama visual da tela de seleção de pesos.

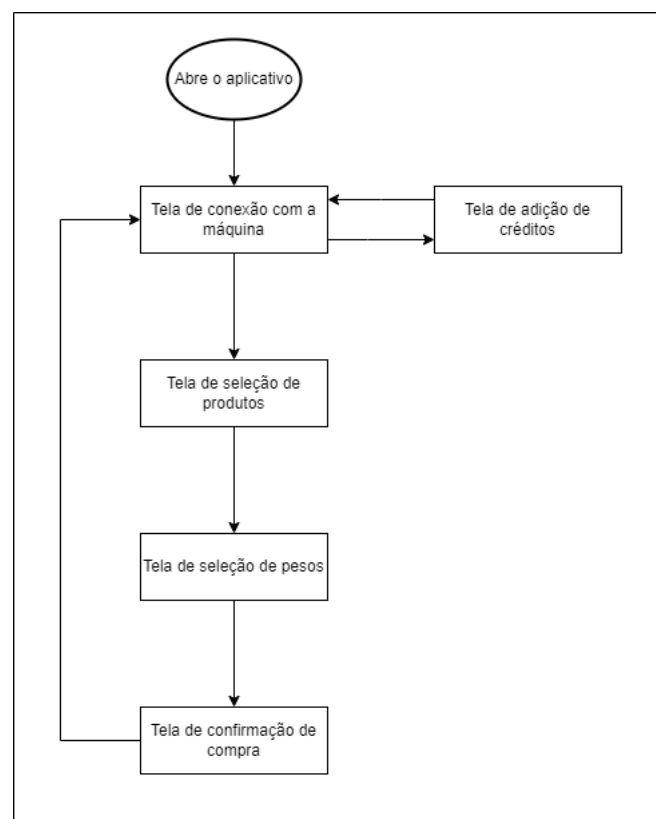


Figura 19: Diagrama de fluxo de telas do aplicativo.

Para planejar a comunicação entre a máquina e o aplicativo foi feita também uma representação gráfica dos pacotes utilizados na comunicação (Figura 20)²

²Este diagrama está disponível de maneira ampliada no apêndice A.

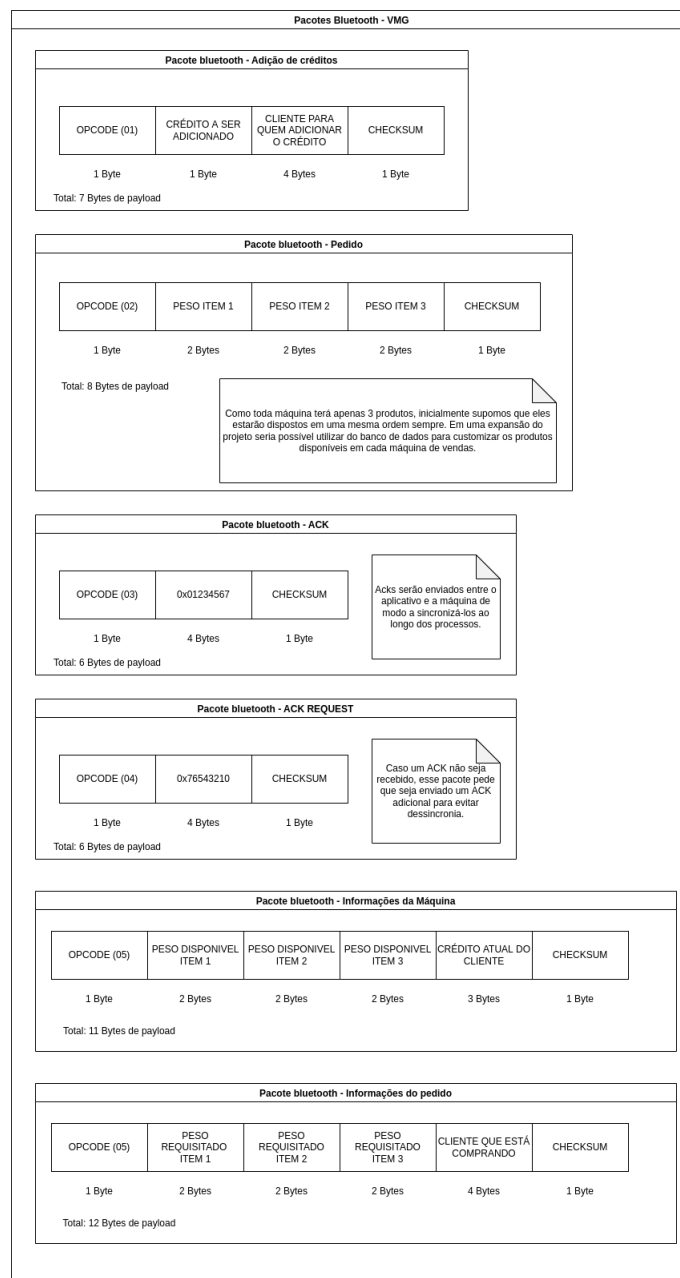


Figura 20: Planejamento dos pacotes Bluetooth.

Após a projetagem, foi iniciado o processo de implementação utilizando da linguagem C++ na IDE ArduinoIDE para o desenvolvimento do firmware da máquina, por ser um jeito fácil e rápido de atingir os objetivos propostos, e Kotlin na IDE AndroidStudio para o desenvolvimento do aplicativo, pelo fato de um dos membros já ter experiência prévia no desenvolvimento utilizando dessas ferramentas. Foi utilizado o Git hospedado em GitHub.com para controle de versionamento e paralelização do processo de programação.

3.5 INTEGRAÇÃO

A integração do projeto foi feita em partes para possibilitar a verificação e isolamento de erros provenientes da junção das partes anteriormente feitas. Considerando a grande possibilidade de erros causados por jumpers, devido ao grande tamanho físico da estrutura mecânica do projeto, o foco inicial da integração se revolveu ao redor da fixação e proteção dos cabos que conectam os componentes das caixas à placa.

Após a fixação da placa na estrutura mecânica e teste das conexões, foram conectados os cabos e testados os softwares de teste. Assim, foi possível verificar se não haviam erros de medição para a construção das portas do projeto, além da correta e suficiente vibração dos motores quando acopladas às caixas.

Depois, foi testado um protótipo da máquina de estados, sem requerer conexão com o aplicativo, para verificar se a integração da estrutura mecânica estava completa com a parte do hardware. Após a confirmação dessa etapa, conserto de erros e alterações de planejamento baseado nos resultados, passou-se para a integração do software.

Ao enviar diversos pacotes e verificar o comportamento da máquina e os pacotes de bits no monitor serial, foi possível verificar se não havia erros lógicos no código, partes indevidamente comentadas ou numerações e cálculos errados.

Após a verificação de todas essas etapas, só restou então o teste da máquina de estados funcionando na íntegra juntamente com a máquina. Após alguns ajustes de requisitos que não puderam ser concluídos ou que foram levemente alterados do planejamento inicial, foi possível atestar um funcionamento satisfatório da máquina.

4 EXPERIMENTOS E RESULTADOS

Diversos experimentos foram realizados durante o projeto. Primeiramente, foram testados todos os sensores e atuadores de maneira individual, a fim de garantir o funcionamento das partes. Então, após essa verificação, foi testada a integração de todas as partes. Os testes da integração também foram realizados de maneira cumulativa, i.e., as partes foram sendo testadas em pequenos conjuntos. Dessa forma, foi possível identificar erros e inconsistências que necessitaram de alterações ao longo do projeto.

Primeiramente, foi testada a estrutura mecânica do projeto. Ao expor a estrutura a testes de estresse, foi observado que um peso acima do necessário era suportado pela estrutura por um período de tempo considerável. No entanto, no primeiro teste da canaleta, foi observado que houve erros de planejamento, uma vez que os produtos necessitavam de uma força superior a fornecida dos motores de vibração para caírem na região de saída. Assim, foi necessário um replanejamento do projeto, o qual aumentou o tempo necessário para finalizar a estrutura, mas que retirou a necessidade de um motor de vibração na canaleta.

Outro problema observado na parte mecânica se deu por dificuldades na produção das portas inicialmente propostas. Assim, as portas acabaram sendo as mais genéricas, as quais "abrem para cima e fecham para baixo". Isso é problemático por possibilitar o travamento da porta caso um produto esteja saindo da porta, podendo afetar o funcionamento da máquina. Outra dificuldade encontrada foi o "encapsulamento" do projeto, i.e., o revestimento previsto no planejamento inicial acabou não sendo cumprido por questões de custo e facilidade de manejo. Isso acabou cortando gastos e facilitando o manejo dos fios e verificação do funcionamento do projeto. Contudo, ao retirar um compartimento devidamente escuro para a inserção da embalagem, não foi possível utilizar os sensores infravermelhos da maneira inicialmente pensada, uma vez que a influência da iluminação externa do ambiente não podia mais ser desprezada. Então, foi pensado em utilizar fitas isolantes, as quais acabaram prejudicando ainda mais a estética do projeto e acabaram não sendo implementadas por não haver tempo hábil de testar efetivamente em diversos ambientes.

Depois, foram realizados testes na parte de hardware. Os testes foram inicialmente conduzidos com o uso de um Arduino UNO, uma vez que erros na comunicação e verificação de componentes já obtidos causaram uma ausência inicial da ESP32. Assim, apesar do funcionamento normal do circuito no esquemático, devido à algumas peculiaridades da ESP32 que dificultariam a implementação do software forçaram alterações no esquemático e consequentemente na pcb produzida, atrasando consideravelmente o andamento do projeto.

Após duas semanas de atraso, foi possível testar individualmente as partes do esquemático em uma placa universal, uma vez que a pcb produzida falhou por ser espelhada incorretamente. Todavia, o uso de jumpers na placa, juntamente com o deslocamento excessivo da placa causou problemas de conexão. Dessa forma, foram necessárias mais duas semanas para um replanejamento do circuito sobre uma placa perfurada e testes de funcionamento completo da placa. Enquanto isso, a placa de pcb foi produzida mais algumas vezes, mas nenhuma foi bem-sucedida, seja por problemas de corrosão excessiva ou transmissão térmica desigual. A placa perfurada apresentou o desempenho esperado, além de funcionar em todos os testes individuais de hardware definidos no projeto.

Nas Figuras 21, 22, 23 é possível observar a placa perfurada, assim como sua posição na estrutura mecânica em três ângulos diferentes.

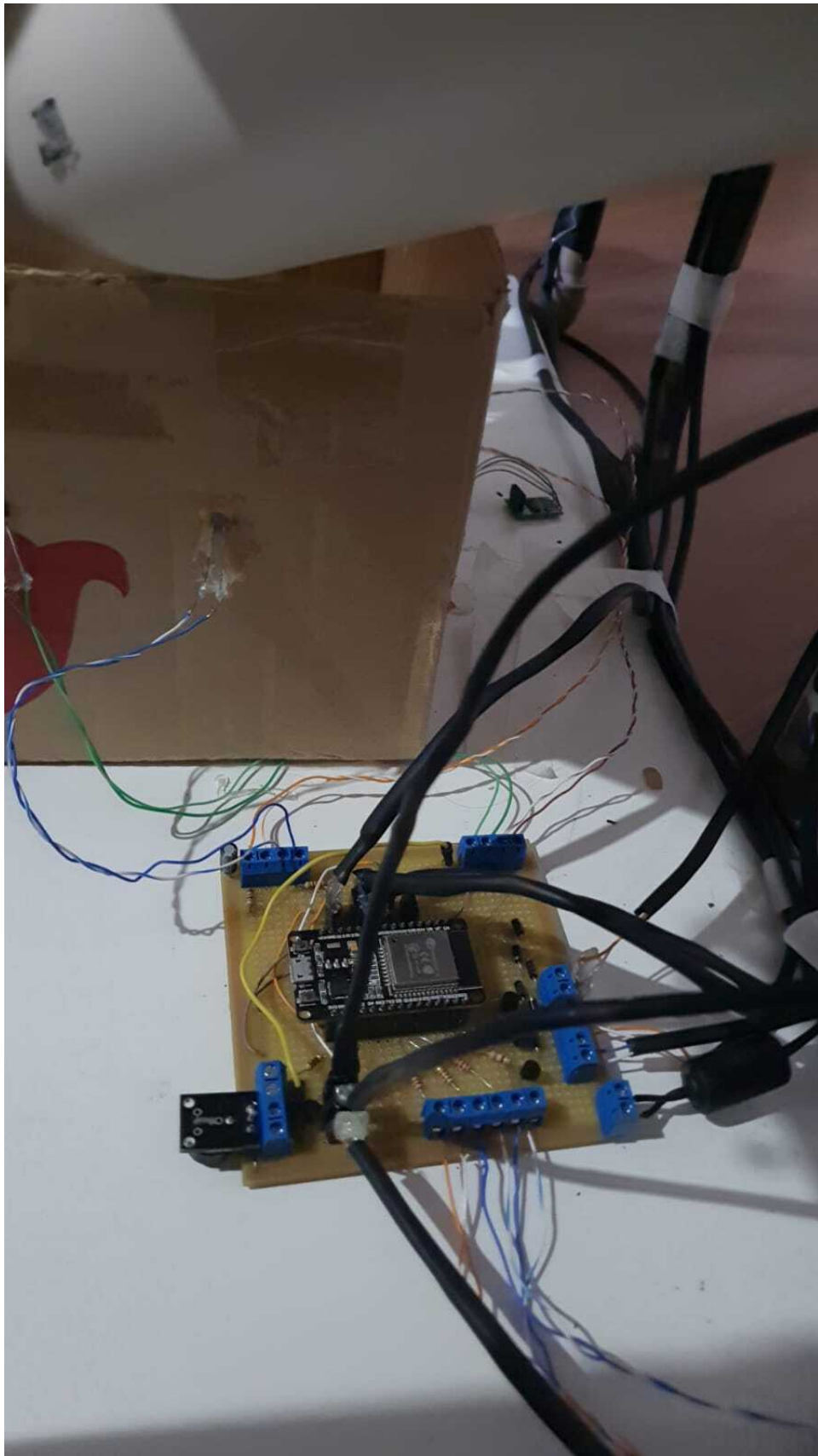


Figura 21: Placa perfurada na máquina

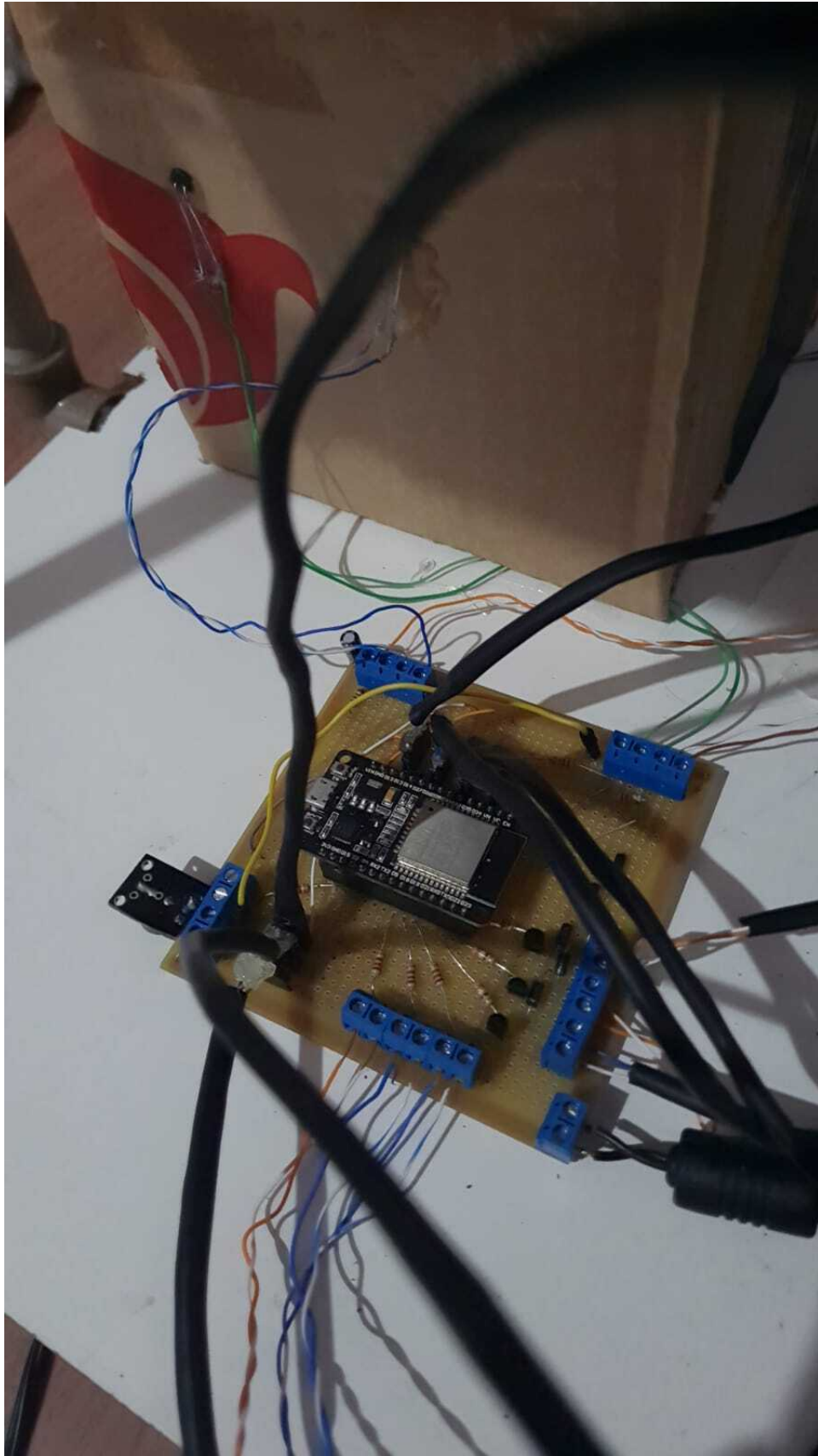


Figura 22: Placa perfurada na máquina

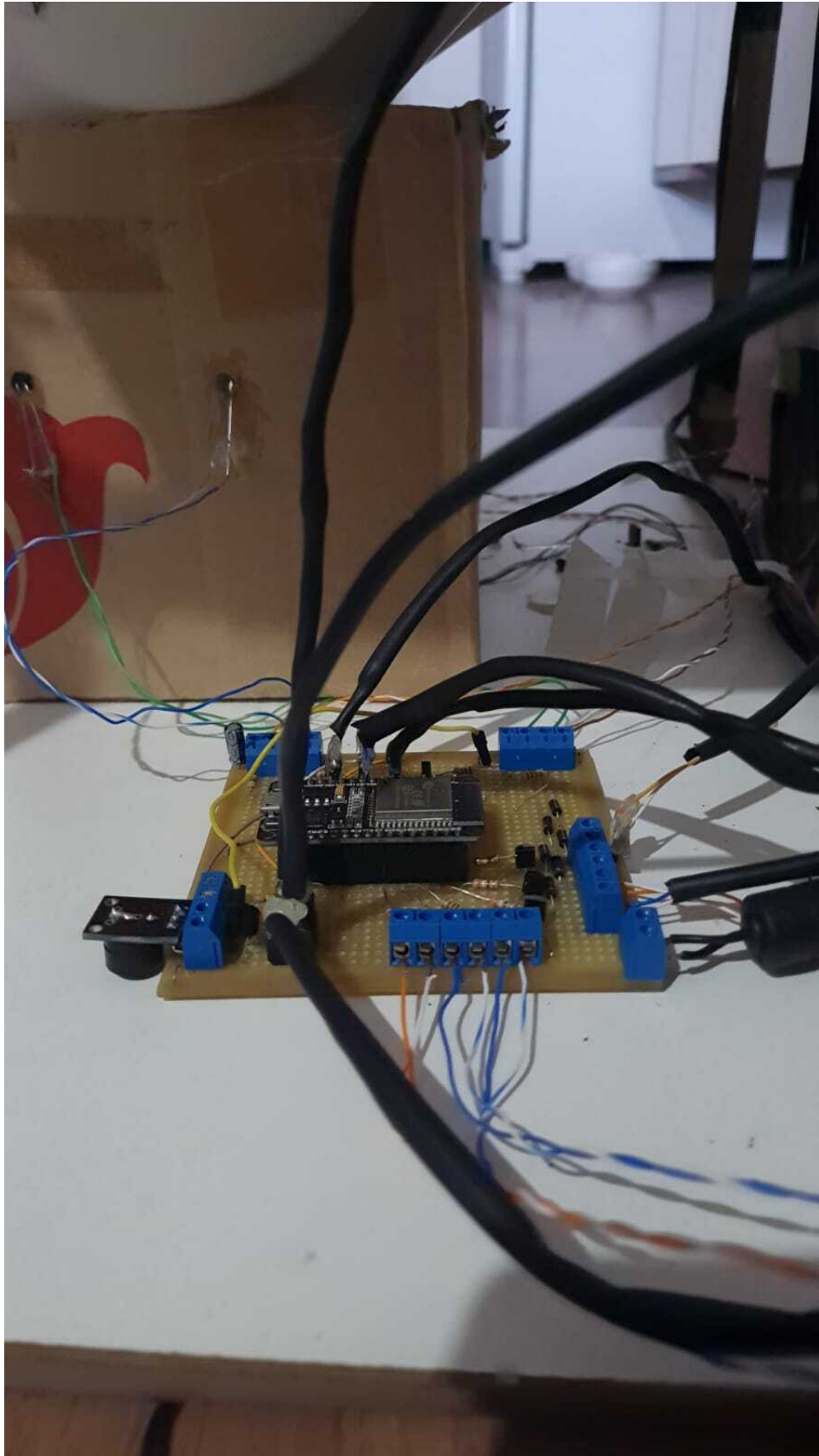


Figura 23: Placa perfurada na máquina

É possível verificar que ainda que não esteticamente agradável, há uma boa divisão dos componentes a serem conectados devido aos conectores borne e às similaridades com a pcb que foi planejada para este projeto.

Paralelamente aos outros experimentos foram realizados diversos testes para verificar a funcionalidade da implementação em software. Como suporte à implementação do aplicativo foram realizados diversos testes em emuladores de android, certificando-se que as soluções escolhidas condiziam com os nossos objetivos - em mais de uma ocasião, esses testes revelaram maneiras mais simples e eficientes de executar as funções do aplicativo do que as planejadas inicialmente.

Inicialmente, esses testes tinham como objetivo experimentar com a interação entre os componentes de interface disponíveis para os aplicativos android, como listas, botões, caixas de texto e overlays, verificando que os elementos funcionavam como uma unidade para depois verificar seu funcionamento em conjunto.

Após terminados os testes de interface foram iniciados juntamente a um simples servidor *bluetooth* em uma ESP32 externa os testes com dispositivos Android reais, verificando a funcionalidade da conexão e a disposição dos elementos de interface em diferentes tamanhos de tela. Foi necessário realizar esses testes em um dispositivo fixo devido a uma das limitações de utilizar um emulador Android - eles não possuem capacidade de conexão via *bluetooth* (LINUXTOPIA, 2022). Os testes iniciais foram complicados, com o aplicativo parando completamente diversas vezes devido a varias nuances na implementação da conexão *bluetooth* em dispositivos android - para que uma conexão seja funcional, é necessário no mínimo 3 *threads* no aplicativo, uma para conexão, uma para transmissão de dados e outra para o recebimento. Essa informação é dada de maneira confusa na documentação oficial do Android (GOOGLE, 2021), e apenas com muita tentativa e erro foi identificada a solução de criar essas *threads*.

Depois de finalizados os testes de conexão, o processo restante seguiu de modo tranquilo, com alguns modelos de máquina de estados sendo testados no *firmware* da máquina com resultados de lógica satisfatórios após alguns experimentos com valores *mock*.

Ocorreram experimentos com diversos modelos de pacote de dados para estabelecer a transmissão segura de informações entre a máquina e o aplicativo, além de diversos diagramas para auxiliar no processo de desenvolvimento das partes mais sutis do software. Os pacotes, incluindo aqueles que não foram utilizados, estão disponíveis na figura 20 e os diagramas auxiliares podem ser observados no apêndice B.

Ao final, foram realizados os testes de integração. Os testes iniciais foram bem sucedidos, apresentando uma boa sincronia entre a parte mecânica de estrutura e hardware. As portas apresentaram funcionamento dentro do esperado, e os motores de vibração ajudavam no controle do fluxo dos produtos para uma saída melhor da caixa. O maior problema nos primeiros testes de integração evidenciaram-se no controle do fluxo do amendoim em específico. Devido ao seu tamanho e oleosidade, tornou-se difícil evitar o despejo de medições erradas do produto, obrigando uma modificação no plano inicial quanto ao controle das portas. Foi verificado que o controle preciso e simultâneo do servo motor com a leitura das balanças era inviável. A leitura das balanças, apesar de precisas, não eram suficientemente rápidas para o controle fino da porta. Especialmente no caso do amendoim, o fluxo significativamente alto impossibilitou essa sincronia entre as partes. Então, foi utilizado um método mais simples, o qual consiste em abrir a porta por um curto período de tempo, fechar a porta e então fazer uma leitura, repetindo o processo até o valor pedido pelo cliente ser atingido.

Além disso, foram encontrados problemas na conectividade de alguns dos cabos. Assim, após alguns dias de discussão e testes, outros cabos foram produzidos com sucesso, ainda que não esteticamente agradáveis. Um dos maiores problemas nessa parte da integração se deu pelo fato dos cabos serem longos, sendo exaustiva a reposição quando houvesse qualquer quebra de conectividade. Além disso, para garantir a estabilidade da estrutura mecânica, o plano de perfurar os canos de pvc para inserção dos cabos foi abandonado, tendo em vista que nesse ponto do projeto, uma perfuração incorreta poderia ocasionar a perda da base da VMG e consequente não completude do trabalho. Assim, para proteger os cabos, improvisações com tubos termo-retráteis foram utilizadas.

Um outro problema encontrado na integração foi uma estranha "não aceitação" de pinos quando o projeto todo era ligado. Ainda que em testes isolados fosse observado o funcionamento das partes do projeto, quando ligado em conjunto, o projeto apresentava problemas de leitura ou funcionamento. Após horas de reavaliação de solda, código e conexões, foi identificado que ao trocar os pinos com funções idênticas e mesmo código, essas partes voltaram a funcionar.

Já na parte da integração do software ao resto, os resultados foram bem melhores. A conexão Bluetooth foi bem-sucedida, sendo possível o pareamento com a máquina e a realização de um pedido. Os maiores erros na parte do software acabaram sendo encontrados bem rapidamente, graças à possibilidade de verificar o funcionamento do restante da máquina juntamente com o monitor serial.

5 CRONOGRAMA E CUSTOS DO PROJETO

5.1 CRONOGRAMA

Para a construção do cronograma, foram definidas todas as atividades necessárias para a realização do projeto. Com essa definição, foram estipuladas datas visando uma melhor organização do tempo. O cronograma pode ser dividido em 6 partes distintas, sendo cada uma delas voltada mais para cada um dos respectivos entregáveis, como está presente nas Figuras 25, 26, 27, 28, 29, e 30, além da entrega final na Figura 31. As cores ao lado indicam a situação conforme o esquema de cores apresentado na Figura 24.

Done in time	Completed late	Todo	
In progress	Late	Started Early	Completed early

Figura 24: Esquema de cores do cronograma

Activity / Deliverable	Responsible	Assistant	Assistant	Status
Project Inception	All			
Project specification	All			
Set up blog	Cássio			
Make Requirements Definition	All			
Make Risk Analysis	All			
Make Schedule	Paulo		Cássio	
Make Budget	Gomes		Paulo	
Beautify blog	Cássio			
Write Project charter	Paulo		Cássio	
Write blog posts	Gomes		Paulo	
Prepare project charter presentation	Paulo			
Purchase and Gather Materials	Gomes		Cássio	
Deliverable #1: Blog				

Figura 25: Tarefas planejadas para o entregável 1

Case design	Gomes		Paulo	
Box design	Gomes		Paulo	
Design of the machine as a whole	Gomes		Cássio	
Case build	Gomes	Cássio	paulo	
Container holder design	Gomes		Paulo	
Box build	Gomes	Cássio		
Dispenser design	Gomes		Paulo	
Pipeline design	Gomes		Paulo	
Dispenser build	Gomes	Cássio		
Pipeline build	Gomes	Cássio		
Container holder build	Gomes		Paulo	
Deliverable #2: Mechanical Structure				

Figura 26: Tarefas planejadas para o entregável 2

Component research	Cássio	Paulo	Gomes	
Sensor circuit design	Cássio	Paulo	Gomes	
Motor Circuit design	Cássio	Paulo	Paulo	
Container holder circuit	Cássio	Paulo	Paulo	
Dispenser circuit	Cássio	Paulo	Gomes	
PCB Design	Cássio			
Run tests on arduino	Cássio	Paulo		
Test sensors' interference	Cássio			
Test error range of circuit	Cássio	Gomes		
Making PCB/Circuit	Gomes	Cássio		
Test stability of holder	Gomes			
Test door movement	Gomes			
Deliverable #3: Hardware				

Figura 27: Tarefas planejadas para o entregável 3

Update the schedule	Paulo			
Software planning and design (with UML)	Paulo			
Product selection interface planning	Paulo			
Weight selection interface planning	Paulo			
Credit management interface planning	Paulo			
Writing Blog Post	Paulo			
Software planning and design (with UML) - corrections	Paulo			
Product selection interface programming	Paulo			
Weight selection interface programming	Paulo			
Credit management interface programming	Paulo			
Placeholder app integration	Paulo			
Firmware planning and design	Paulo			
Container door driver programming	Paulo			
Presence detection driver programming	Paulo			
Bluetooth communication driver programming	Paulo			
Weight detection driver programming	Gomes			
Vibration motor driver programming	Paulo			
Database driver programming	Cássio	Paulo		
Firmware programming	Paulo	Gomes	Cássio	
App-Machine integration	Paulo			

Figura 28: Tarefas planejadas para o entregável 4 - parte 1

App-Machine integration	Paulo			
Error treatment	Paulo	Cássio		
Database planning	Paulo	Cássio		
Database programming	Cássio	Paulo		
Credit viewing functionality test	Paulo			
Promotional code functionality test	Paulo			
Weight selection functionality test	Paulo			
Product selection functionality test	Paulo			
Database information retrieval test	Cássio	Paulo		
Database information input test	Cássio	Paulo		
Screen selection functionality test	Paulo			
Documentation	Paulo			
Write second blog post	Paulo			
Write Partial Evaluation 1 report	Gomes	Paulo		
Prepare Partial Evaluation 1 presentation	Gomes	Cássio		
Deliverable #4: Software				

Figura 29: Tarefas planejadas para o entregável 4 - parte 2

Box and case integration	Gomes		Cássio	
Box and case testing	Cássio		Gomes	
Electronics and case integration	Cássio	Paulo	Gomes	
Electronics and case testing	Paulo		Cássio	
Electronics and box integration	Gomes	Paulo	Cássio	
Electronics and box testing	Cássio		Gomes	
Machine and app integration	Paulo		Gomes	
Machine and app testing	Paulo		Gomes	
Complete build testing	Paulo		Cássio	
Write Partial evaluation 2 report	Paulo		Cássio	
Prepare Partial Evaluation 2 presentation	Cássio		Gomes	
Deliverable #5: Integration				

Figura 30: Tarefas planejadas para o entregável 5

Working prototype demonstration	All			
Write technical report	Paulo		Cássio	
Prepare Partial Evaluation 3 presentation	Gomes		Paulo	
Shoot Video	All			
Demonstration				Até agora
Total Time				Semana
Prepare Final Evaluation presentation	All			
Final Evaluation				Até agora
Total Time				Semana

Figura 31: Tarefas planejadas para o entregável final

Por fim, no gráfico da Figura 32, tem-se a divisão das horas trabalhadas, em azul, e das horas previstas, em vermelho. Pode-se verificar que o número de horas realmente efetivadas foi bem maior que o tempo planejado. Isso se deve ao tempo extra tomado para replanejamentos e conserto de erros ao longo do projeto. Todo o cronograma está disponível online¹

¹Disponível em: <https://docs.google.com/spreadsheets/u/1/d/1bdrTKiWSXUvHpyr5hWXgXmDE-O8Ac1qOoMfqdSIT6I/edit?usp=sharing>

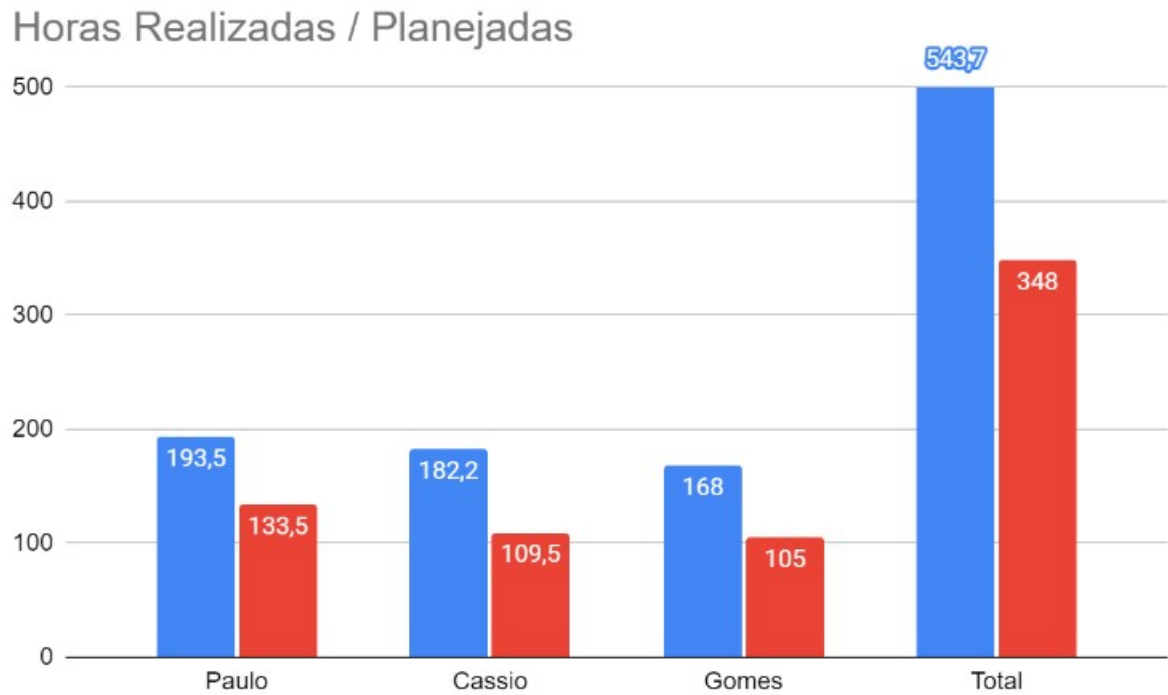


Figura 32: Gráfico de horas trabalhadas e estimadas semanalmente por integrante

5.2 CUSTOS

Após a conclusão do planejamento foi possível gerar um orçamento de todos os materiais necessários para o desenvolvimento da VMG. O orçamento inicial foi de R\$877,00, ou R\$292,00 para cada integrante, conforme a tabela 5.

Componente	Quantidade	Preço Total
Célula de Carga	13	R\$260,00
HX711	4	R\$80,00
ESP32	1	R\$50,00
Buzzer	1	R\$3,00
Bateria de 12V	1	R\$70,00
Servomotor SG90	3	R\$69,00
Potes e Embalagem	3	R\$60,00
Motor de Vibração	3	R\$30,00
Conectores	3	R\$30,00
Suporte	3	R\$30,00
Materiais (Madeira, Tubos de PVC e afins)	Diversos	R\$200,00
Custo total		R\$877,00

Tabela 5: Tabela do orçamento inicial.

Ao longo do projeto foi necessário repensar a lista de componentes devido a descobertas durante os experimentos e para repor aqueles materiais que porventura estragaram ou foram consumidos no processo de experimentação, trazendo o custo total a R\$1157,00, ou R\$385,00 para cada integrante, como mostra a tabela 6.

Componente	Quantidade	Preço Total
Materiais de construção (Madeira, Tubos de PVC e afins)	Diversos	R\$250,00
LEDs	4	R\$5,00
ESP32	2	R\$90,00
Buzzer	1	R\$3,00
HX711	4	R\$51,60
Célula de Carga	7	R\$245,00
Potes e Embalagem	3	R\$15,30
Motor de Vibração	3	R\$9,00
Conectores	3	R\$50,00
Servomotor SG90	3	R\$69,00
Componentes eletrônicos em geral	Diversos	R\$80,00
Custo total		R\$1157,00

Tabela 6: Tabela de custos de produção e pesquisa.

6 CONCLUSÕES

6.1 CONCLUSÕES

Diante dos resultados citados anteriormente, é possível concluir que o desenvolvimento do projeto não obteve sucesso completo. Os integrantes do grupo não apenas subestimaram muitos dos requisitos propostos no plano de projeto, mas também não tomaram em consideração a capacidade de se encontrar tendo em vista os cronogramas individuais de cada membro. No entanto, foi possível verificar que os conhecimentos de diversas outras matérias foram aplicadas com sucesso no projeto.

6.2 TRABALHOS FUTUROS

Como trabalho futuro, é possível observar diversas melhorias no projeto atual. Primeiramente, ao invés de construir uma estrutura mecânica do zero, o uso de uma estrutura já pronta como uma sapateira, ou ainda, o uso de estruturas metálicas perfuradas que podem ser conectadas por parafusos teriam sido melhores opções tanto no quesito de tempo, quanto em estética. O uso de uma estrutura já pronta, só necessitaria de pequenas alterações, além de já possuir medidas que garantem sua estabilidade, poupando alguns dias de trabalho lixando e cortando madeira e tubos de PVC. Já a estrutura metálica, apesar de mais cara, tornaria mais fácil e seguro o posicionamento dos fios, pois sua estrutura interna possibilitaria uma espécie de "canaleta de fios", assegurando uma melhor estética (sem cabos à vista), além de menos risco de dano aos fios.

Ainda sobre a estrutura mecânica, seria possível melhorar as portas das caixas aderindo ao funcionamento de "baixo para cima" inicialmente proposto, a fim de evitar que ocorram emperramentos na porta. Outra maneira, seria adquirir caixas que já vêm com um sistema de dispensa, ainda que isso possa ser consideravelmente mais caro.

Referente à parte de hardware, seria possível melhorar caso houvesse um melhor planejamento referente à testes de integração. Ainda que a integração propriamente dita

fosse mais tarde, seria possível testar a integração entre a mecânica e o hardware com um código simples de estresse, em que todos os componentes funcionassem em ciclos, testando simultaneamente somente os que poderiam funcionar juntos na prática (para evitar um uso de corrente excessivo, já que o projeto utiliza uma quantidade considerável de motores). Assim, seria possível comprovar o funcionamento da placa com bastante tempo de sobra, e então uma encomenda da placa poderia ser feita. Dessa forma, a pcb poderia ser planejada em quatro camadas, além de ser resinada por uma empresa especializada na produção de placas. Assim, erros na produção de pcb caseira seriam evitados, economizando alguns dias de trabalho.

Outra melhoria possível no hardware seria a utilização mais sábia das células de carga. Por ser a primeira vez que o grupo trabalhou com esse componente, além de compreensões errôneas sobre a precisão das células de carga, o projeto teve sua complexidade e custos elevados pelo uso de um número maior de sensores de peso que o necessário. Para este projeto, a utilização de somente uma célula de carga de capacidade de 5kg para cada balança teria sido suficiente, e não duas de 1kg por balança, como foi utilizado neste trabalho.

REFERÊNCIAS

CODELLOS, F. **Amplificadores, suas funções e tipos**. 2013. Disponível em: <<https://revistaautomotivo.com.br/amplificadores-suas-funcoes-e-tipos/>>. Acesso em: 21 de junho de 2022.

GOOGLE. **Bluetooth overview**. 2021. Disponível em: <<https://developer.android.com/guide/topics/connectivity/bluetooth>>. Acesso em: 13 de Abril de 2022.

LINUXTOPIA. **Android Emulator Limitations**. 2022. Disponível em: <https://www.linuxtopia.org/online_books/android/devguide/guide/developing/tools/android_emulator_limitations.html>. Acesso em: 13 de Abril de 2022.

OMEGA. **Célula de Carga**. Disponível em: <<https://br.omega.com/prodinfo/celulas-de-carga.html>>. Acesso em: 21 de junho de 2022.

SEMICONDUCTOR, A. **24-Bit Analog-to-Digital Converter (ADC) for Weigh Scales**. Disponível em: <https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/hx711_english.pdf>. Acesso em: 21 de junho de 2022.

WIKIPEDIA. **Servomotor**. 2020. Disponível em: <<https://pt.wikipedia.org/wiki/Servomotor>>. Acesso em: 21 de junho de 2022.

WIKIPEDIA. **DC Motor**. 2021. Disponível em: <https://en.wikipedia.org/wiki/DC_motor>. Acesso em: 21 de junho de 2022.

WIKIPEDIA. **Bluetooth**. 2022. Disponível em: <<https://pt.wikipedia.org/wiki/Bluetooth>>. Acesso em: 20 de Junho de 2022.

APÊNDICE A – DIAGRAMAS AUMENTADOS

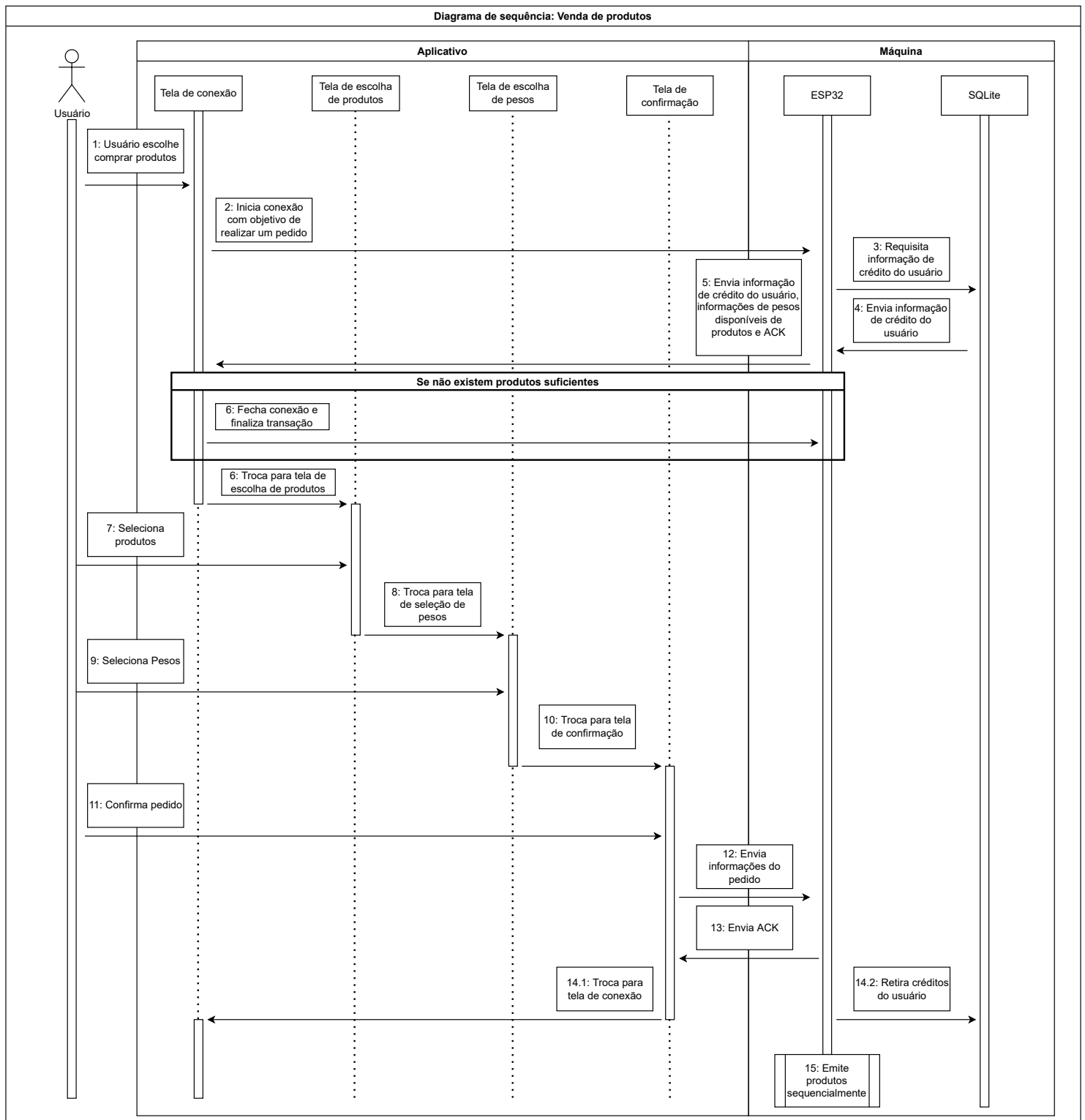
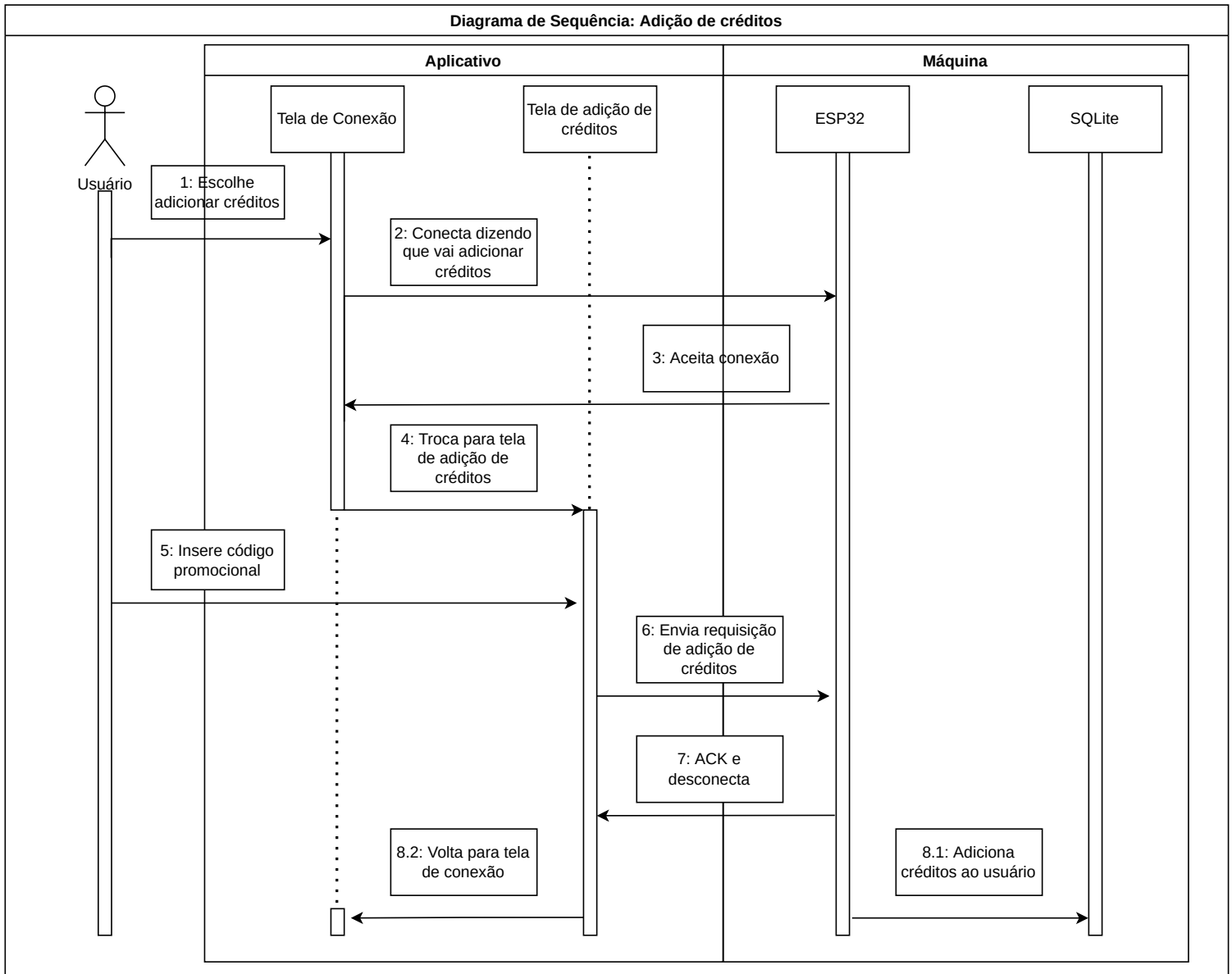
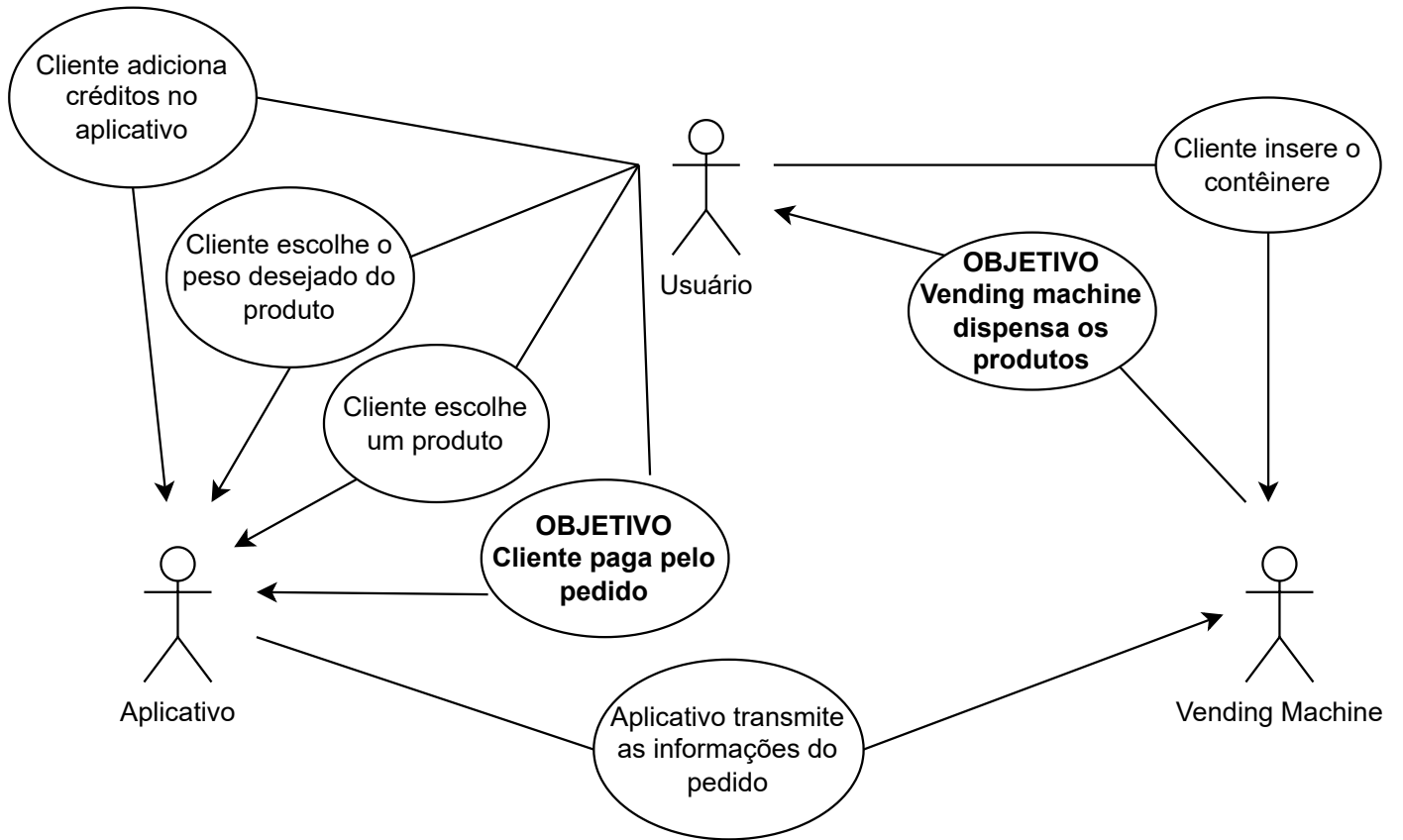


Diagrama de Sequência: Adição de créditos



APÊNDICE B – DIAGRAMAS AUXILIARES E VERSÕES INICIAIS

Diagrama de Caso de Uso (Versão inicial) - VMG



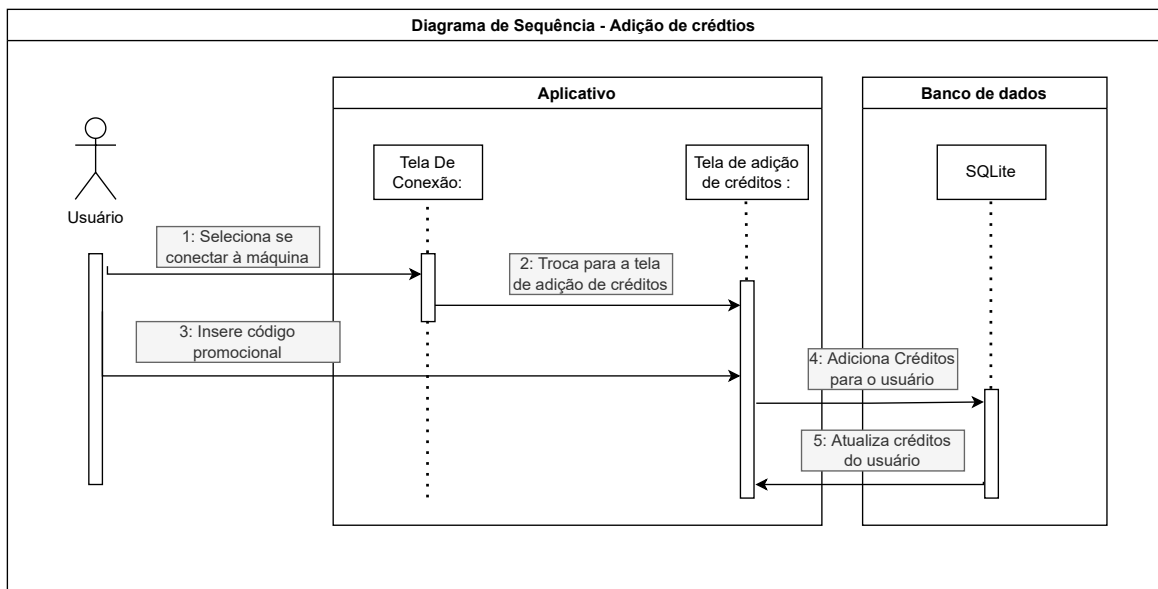


Diagrama de sequência: inserção de contêiner

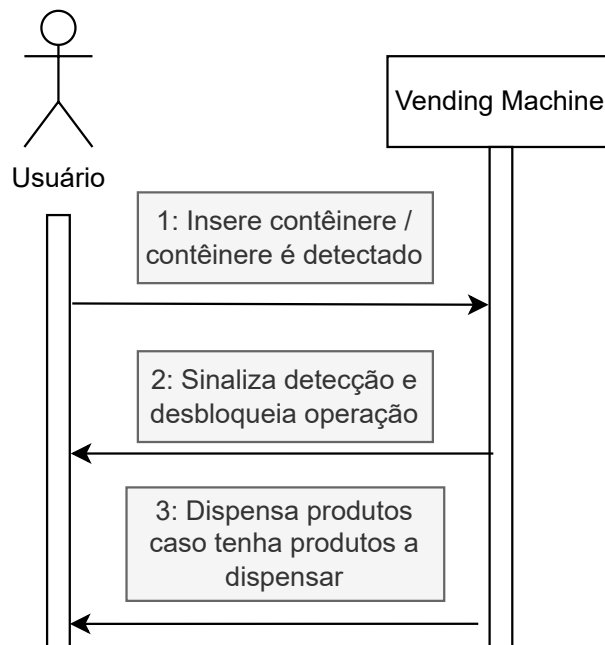
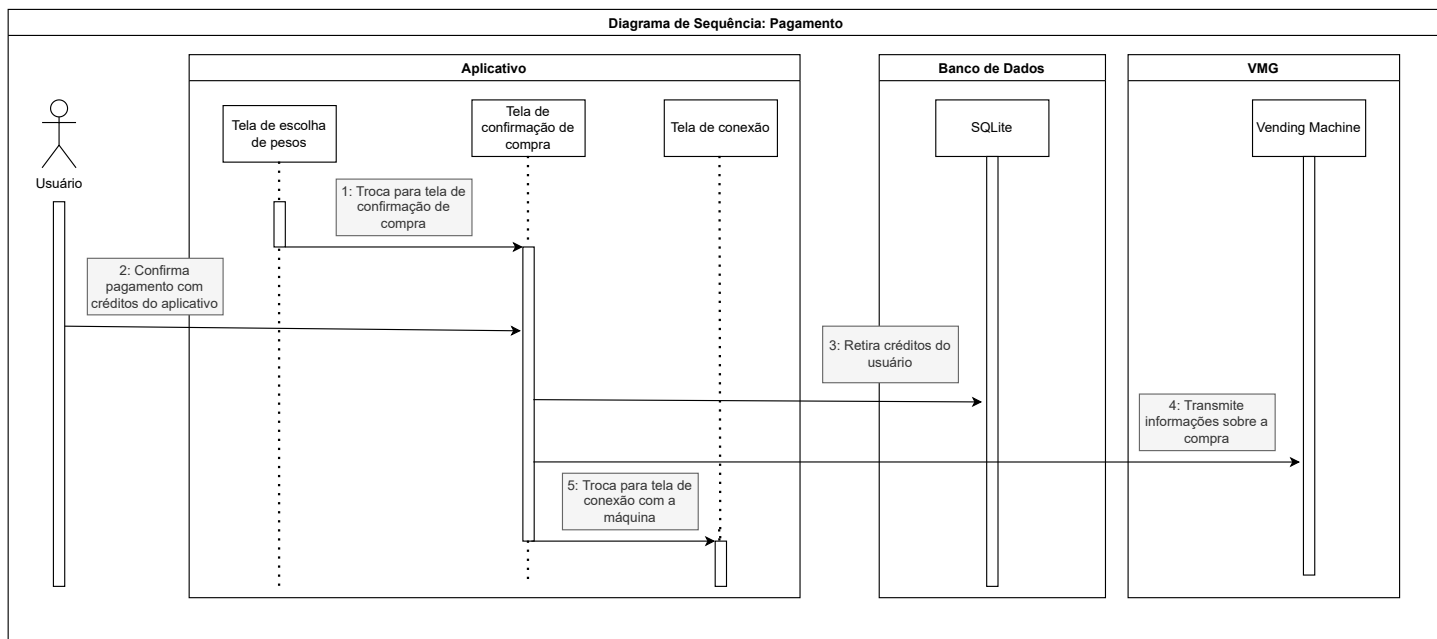
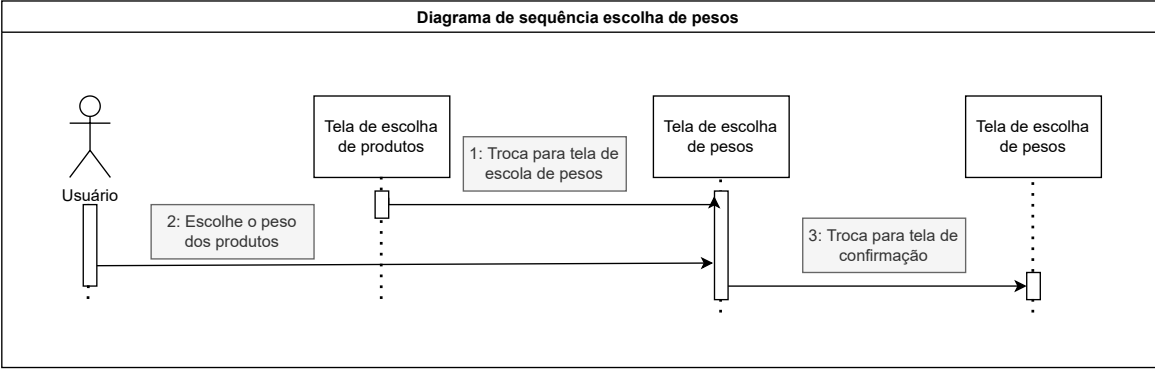


Diagrama de Sequência: Pagamento





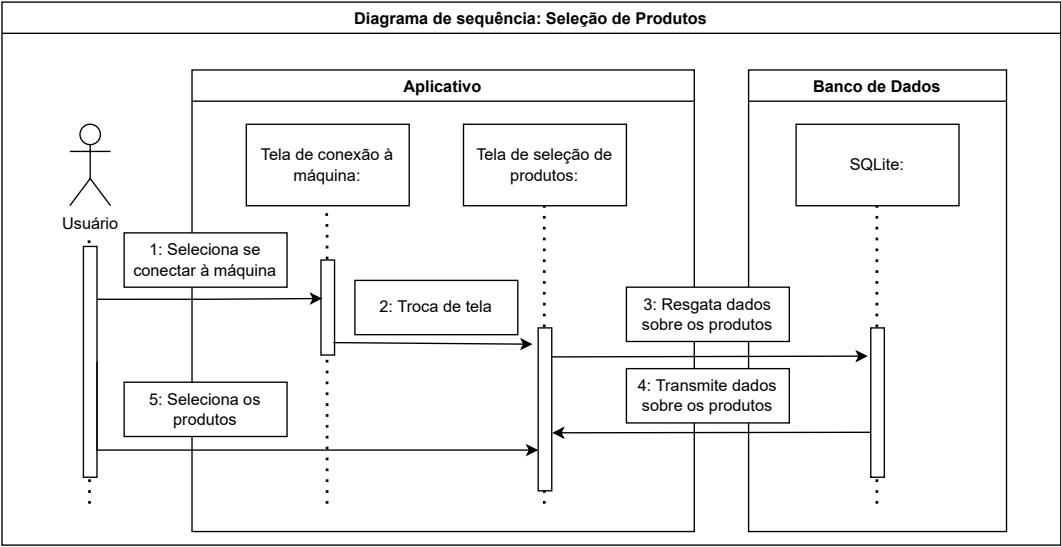


Diagrama de sequência: transmissão de dados

