

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE ENGENHARIA QUÍMICA
CURSO DE ENGENHARIA QUÍMICA**

MARCO VINICIUS FONTENELE AMORIM HUF

**IMPLEMENTAÇÃO DE SISTEMA DE AUTOMAÇÃO NO PROCESSO
DE PRODUÇÃO DE FORMULADOS EM PÓ**

TRABALHO DE CONCLUSÃO DE CURSO

LONDRINA

2021

MARCO VINICIUS FONTENELE AMORIM HUF

**IMPLEMENTAÇÃO DE SISTEMA DE
AUTOMAÇÃO NO PROCESSO DE PRODUÇÃO
DE FORMULADOS EM PÓ**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Química da Universidade Tecnológica Federal do Paraná, *Campus Londrina*, como requisito parcial à obtenção do título de Engenheiro Químico.

Orientador: Prof. Dr. Felipi Luiz de Assunção Bezerra

LONDRINA

2021

MARCO VINICIUS FONTENELE AMORIM HUF

**IMPLEMENTAÇÃO DE SISTEMA DE AUTOMAÇÃO NO PROCESSO DE
PRODUÇÃO DE FORMULADOS EM PÓ**

Trabalho de Conclusão de Curso de Graduação para
obtenção do título de Bacharel em Engenharia
Química da Universidade Tecnológica Federal do
Paraná (UTFPR).

Data de aprovação: 02/dezembro/2021

Prof. Dr. Felipi Luiz de Assunção Bezerra
Universidade Tecnológica Federal do Paraná – *Campus Londrina*

Prof. Dr. Guilherme Duenhas Machado
Universidade Tecnológica Federal do Paraná – *Campus Londrina*

Prof. Dr. Lucas Bonfim Rocha
Universidade Tecnológica Federal do Paraná – *Campus Londrina*

RESUMO

O presente projeto foi idealizado a partir de experiências vividas do autor durante o período de estágio na indústria de pós e massas alimentícias. Uma das operações é a pesagem de matéria-prima e o ensacamento de uma mistura pré-definida de acordo com a demanda. Entretanto, a pesagem e ensacamento são feitos de forma manual, o que gera erros que não são considerados no balanço de material. O controle de operações e gerenciamento de dados automatizados são ferramentas imprescindíveis para uma gestão eficiente de processos. O presente trabalho desenvolve um protótipo com *software* e *hardware* para adquirir dados de produção na etapa de pesagem da matéria-prima utilizada no processo de produção dos pós alimentícios. O Sistema envia os dados para um sistema, o qual gera relatórios e acelera a reatividade da gestão para desvios de qualidade e de produção.

Palavras-chave: Arduino, gestão de produção, Sistema Supervisório.

ABSTRACT

This project was conceived based on the author's experiences during his internship in the powder and pasta industry. One of the operations is the weighing of raw material and the bagging of a pre-defined mixture according to demand. However, weighing and bagging is done manually, which generates errors that are not considered in the material balance. Automated operations control and data management are essential tools for efficient process management. The present work develops a software and a hardware to acquire production data in the step of weighing the raw material used in the production process of food powders. The System sends the data to a system, which generates reports and accelerates management reactivity to quality and production deviations.

Keywords: Arduino, production management, supervisory system.

SUMÁRIO

1. INTRODUÇÃO	1
2. OBJETIVOS.....	4
2.1 Objetivo geral	4
2.2 Objetivos específicos.....	4
3 REVISÃO BIBLIOGRÁFICA.....	4
3.1 Automação de sistemas de produção	4
3.2 Sensores e atuadores	8
3.3 Célula de carga	9
3.4 Controlador lógico programável.....	10
3.5 Sistema de supervisão e aquisição de dados (SCADA)	12
3.6 Arduino.....	13
3.7 Arduino versus CLP	15
4. METODOLOGIA	16
4.1 Processo produtivo atual da empresa.....	16
4.2 Processo produtivo pós implementação do sistema de automação em pesagem.....	18
4.3 Protótipo construído com Arduino	19
4.3.1 Materiais.....	19
4.3.2 Montagem da estrutura física da balança.....	20
4.3.3 Conexões do Arduino	21
4.3.4 Softwares	28
4.3.5 Bibliotecas utilizadas no Arduino IDE	30
4.3.5.1 HX711	30
4.3.5.2 LCD_I2C.....	30
4.3.5.3 Pushbutton	30
4.3.5.4 rExcel	30

4.4.6 Bibliotecas utilizadas no Python	31
4.4.6.1 XLWINGS	31
4.4.6.2 XLRD	31
4.4.6.3 pySimpleGui	31
5. RESULTADOS E DISCUSSÃO	31
5.1 Protótipo	32
5.2 Calibração	33
5.3 Fluxo de funcionamento da aplicação.....	34
5.4 Orçamento do protótipo construído	40
6. CONCLUSÕES	41
REFERÊNCIAS.....	43
ANEXO A – CÓDIGO DA CALIBRAÇÃO UTILIZANDO A BIBLIOTECA HX711 ...	47
APÊNDICE A – CÓDIGO FONTE DO ARDUINO IDE – PROJETO DE BALANÇA DE AUTOMAÇÃO CONSTRUÍDA COM ARDUINO.....	49
APÊNDICE A.1 – Instância de variáveis	49
APÊNDICE A.2 – Setup inicial – Executado ao Arduino ser ligado	50
APÊNDICE A.3 – Loop Arduino.....	50
APÊNDICE A.4 – Função que detecta se o usuário alterou a receita a ser produzida	52
APÊNDICE A.5 – Função que mostra informações do ingrediente a ser pesado e o peso atual da balança.....	52
APÊNDICE A.6 – Função utilizada para tarar a balança	53
APÊNDICE A.7 – Função utilizada para exibir um erro no display caso o valor pesado não esteja dentro do solicitado	53
APÊNDICE A.8 – Função utilizada para escrever no Excel o ingrediente, valor pesado, valor solicitado, data e hora do processo	53
APÊNDICE B – CÓDIGO FONTE DESENVOLVIDO NO PYTHON – PROJETO DE BALANÇA DE AUTOMAÇÃO CONSTRUÍDA COM ARDUINO	55

LISTA DE FIGURAS

Figura 1 - Sistema de automação industrial desenvolvido pela empresa Citysystems para a cervejaria Imperatriz.....	6
Figura 2 – Sistema de automação industrial desenvolvido pela Citysystems para a Qualimix – Nutrição animal.....	7
Figura 3 - Sistema de automação industrial para pesagem em receita de ração animal	7
Figura 4 - Célula de carga	9
Figura 5 - Módulo conversor amplificado HX711.....	10
Figura 6 - Controlador Lógico Programável - CLP	11
Figura 7 - Estrutura básica de funcionamento de um CLP	11
Figura 8 - Fluxo de automação do processo	12
Figura 9 - Arduino Uno	14
Figura 10 - Fluxograma do processo produtivo atual	17
Figura 11 - Fluxograma do processo após o processo de automação	19
Figura 12 – Materiais utilizados na montagem do protótipo	20
Figura 13 - Disposição da célula de carga no protótipo do autor	21
Figura 14 - Conexões entre a célula de carga, o módulo HX711 e o Arduino.....	22
Figura 15 - Módulo I2C para display LCD	23
Figura 16 - Conexões com o módulo I2C para display LCD	24
Figura 17 - Push button utilizado no Arduino	25
Figura 18 - Conexões com o Push button utilizado no Arduino.....	26
Figura 19 - Esquema completo de conexões do projeto	28
Figura 20 - Protótipo de balança construída com Arduino	32
Figura 21 - Calibração confirmada com a Balança Arduino	34
Figura 22 - Fluxograma de funcionamento do Software.....	35
Figura 23 - Demonstração de planilha de receitas	35
Figura 24 - Tela para conectar o Arduino e o Excel	36
Figura 25 - Interface construída no Python para o usuário selecionar a receita e o peso a ser produzido	36
Figura 26 - Planilha de receitas após usuário selecionar o peso na interface.....	37

Figura 27 - Planilha de pesagem criada automaticamente após confirmado o peso e receita a serem produzidos	37
Figura 28 - Protótipo informando ao usuário o peso e o ingrediente	38
Figura 29 - Mensagem de erro ao colocar um peso fora da faixa permitida na balança	39
Figura 30 - Dados gravados no Excel após apertar o botão “1” do protótipo	39
Figura 31 - Protótipo mostrando o próximo ingrediente a ser pesado após o ingrediente anterior ter sido gravado no Excel.....	40

LISTA DE TABELAS

Tabela 1 - Comparação entre o Arduino e o CLP	16
Tabela 2 - Conexões entre a célula de carga e o módulo HX711	22
Tabela 3 - Conexões entre o módulo HX711 e o Arduino	22
Tabela 4 - Conexões entre o módulo I2C, display LCD e o Arduino	23
Tabela 5 - Interligações do Arduino	27
Tabela 6 - Orçamento do protótipo	41

1. INTRODUÇÃO

A qualidade, independente do ramo em que se atua, não deve ser vista pelas organizações apenas como um diferencial, mas sim, como um mecanismo para antever, evitar ou solucionar os problemas que venham correr. Diante de um cenário repleto de novidades, novos produtos, marcas e concorrentes, as ferramentas dessa área empresarial auxiliam na estruturação de estratégias fundamentais para manutenção e fortificação do elo com clientes e consumidores finais. Conseqüentemente, dentre as várias vantagens dessa aplicação, ocorrem o aumento de vendas e da competitividade, consolidação da imagem, fidelização de clientes, dentre outras (NASCIMENTO; ROSA; NANTES, 2016).

A ausência de controle de qualidade, tanto em processos gerenciais, como nos produção, acarreta conseqüências financeiras, custos desnecessários, retrabalhos, perdas, desperdícios e transtornos aos clientes (NARVAES, 2009).

O controle de qualidade na área alimentícia trabalha para padronizar, regulamentar, fiscalizar e monitorar tudo que ocorre na cadeia produtiva, ajudando a reduzir ao máximo o desperdício de alimentos (MARTINS, 2021).

A padronização da fabricação de cada produto e o monitoramento constante dos processos são fundamentais para se obter êxito na qualidade, já que ele pode auxiliar na rápida tomada de providências e prevenção de ameaças futuras (NARVAES, 2009). O processo de monitoramento pode ser viabilizado através da automação industrial (ROSÁRIO, 2009).

A automação é definida como um sistema que faz uso de técnicas computadorizadas ou mecânicas para agilizar as tarefas quase sempre sem a interferência humana. Esta ferramenta é aplicada e implementada por um *software* que gera instruções para um *hardware*, o qual influencia no processo produtivo. A Automação é uma aliada significativa no desempenho do processo, pois por meio da tecnologia, são gerados indicadores que otimizam processos, auxiliam na gestão e reduzem o tempo que o colaborador levaria para realizar a atividade (ROSÁRIO, 2009).

O gerenciamento de um processo de manufatura, seja ele qual for utilizando o recurso da automação torna muito mais eficaz e confiável as ações da gestão da manutenção, gerando índices operacionais, como o *Operational Performance*

Indicators (OPI's) e gerenciais, como o *Key Performance Indicators (KPI's)* (RONCOLI; REGATTIERI, 2018).

Os indicadores são de suma importância para a tomada de decisões em relação a manutenções preventivas e preditivas, aumentando as chances de se evitar uma manutenção corretiva, o que muito prejudica qualquer processo (RONCOLI; REGATTIERI, 2018).

O estágio obrigatório do autor foi realizado na indústria de pós alimentícios e massas alimentícias Hoshino Rizzo. Todas as etapas de produção da indústria Hoshino Rizzo são realizadas de forma puramente manual, o que torna o processo de fabricação lento e com pouco controle, suscetível a erros em quase todas as etapas de produção. Foi observado ocasiões em que o colaborador se desatenta ao pesar a matéria-prima, deixando uma matéria-prima de lado e comprometendo a qualidade do produto final, acarretando em perdas financeiras consideráveis.

Se tratando de indústrias que utilizam como matéria-prima de seus produtos grãos ou pós, encontra-se a necessidade de equacionar volumes ou quantidades desses materiais com eficiência para que o produto final tenha uniformidade e exatidão do consumo da matéria-prima, proporcionando o planejamento da produção e de seus custos, além de um padrão de qualidade (ALVES; MANFIO, 2018).

Analisando as indústrias que precisam quantificar o seu produto acabado em pequenas frações para o consumidor final, também se faz necessário uma exatidão na pesagem de cada fração, pois além de não exceder o peso determinado para evitar prejuízos, impede que falte produto naquela porção pré-determinada (ALVES; MANFIO, 2018).

Na etapa de pesagem de matéria-prima no processo de produção da indústria de pós alimentícios Hoshino Rizzo, é necessário a mistura de diversos componentes em quantidades diferentes. Para agilizar e facilitar o processo, alguns desses componentes são pesados e preparados anteriormente à mistura ou formulação, o que chamamos de pré-batch ou pré-mix.

O pré-mix varia de acordo com cada processo industrial e, na maioria dos casos, possui componentes, pesos e necessidades específicas para a fabricação de produtos com formulações diferentes, exigindo um alto nível de controle e organização para que não aconteçam interferências e falhas (CANTON, 2019).

Em um processo onde o operador prepara as bateladas manualmente, podem ocorrer problemas por diversos motivos, o que afeta o processo como um todo,

gerando perdas e atrasos na produção. É possível desenvolver um sistema de dosagem e pesagem automáticos para resolver o problema (CANTON, 2019).

Um sistema de dosagem industrial automatizado pode ser definido como sendo um conjunto de equipamentos com a finalidade de manipular matérias-primas (líquidos e sólidos) para produzir produtos químicos tais como como bebidas, derivados químicos, mistura de nutrientes para ração animal (pré-mix, sal mineral), aditivos e líquidos para indústria alimentícia em geral (CITYSYSTEMS, 2016).

Os sistemas de dosagem industrial autocontrolados exigem a aplicação de controladores lógicos programáveis (CLPs). Estes controladores são capazes de adquirir sinais de sensores de acordo com valores pré-determinados e realizar o monitoramento e controle automático da dosagem. Eles possuem ainda um baixo tempo de resposta e são responsáveis por alterar condições do processo que exigem mudanças frequentes nas condições de dosagem industrial (CITYSYSTEMS, 2016).

O presente trabalho foi idealizado a partir do período de estágio obrigatório do autor em uma empresa que produz e comercializa pós e massas alimentícias, com produção completamente manual, demonstrando potencial de melhoria em seu processo de produção através da automação.

Durante o estágio notou-se problemas ocorrendo na etapa de pesagem, e então foi orçado equipamentos de dosagem e pesagem automáticos, porém, foi constatado que o equipamento não era viável para empresa, devido ao seu custo elevado, dependendo do equipamento, o custo era maior que o próprio faturamento da empresa.

Partindo disso, uma das melhorias propostas é substituir o sistema de pesagem atual por um sistema automatizado de pesagem de baixo custo utilizando Arduino, a fim de permitir o controle do processo e a integração com um sistema de supervisão.

2. OBJETIVOS

2.1 OBJETIVO GERAL

O objetivo geral desse trabalho é projetar um sistema de automação e controle de baixo custo para uma indústria de pós alimentícios, especificamente na área da pesagem da matéria-prima, bem como, realizar o levantamento de custo do projeto.

2.2 OBJETIVOS ESPECÍFICOS

Para alcançar o objetivo geral deste projeto, procura-se alcançar os seguintes objetivos específicos:

- Especificar a instrumentação necessária para o projeto
- Implementar a interface de comunicação entre Arduino, Excel e Python
- Levantar com fornecedores o custo de cada material utilizado na montagem do projeto
- Reduzir as perdas de produção no processo de pesagem de matéria-prima.

3. REVISÃO BIBLIOGRÁFICA

3.1 AUTOMAÇÃO DE SISTEMAS DE PRODUÇÃO

A automação é definida como um sistema automático pelo qual os mecanismos controlam seu próprio funcionamento, quase sem a interferência do ser humano. Em termos práticos, a automação surge da substituição dos humanos em atividades de controle de máquinas e processos, empregando para isso diferentes elementos tecnológicos para se obter informações emitidas por sensores, processá-las por meio de elemento de decisão e efetuar as ações calculadas sobre o processo utilizando atuadores (MELCONIAN, 2013).

O sistema é capaz de atuar levando em conta informações recebidas pelos sensores e/ou transdutores, podendo então executar ações corretivas e repetitivas, aumentando a eficiência do processo e prevenindo erros humanos na produção (ESTEPA, 2010).

Entre os elementos que constituem os sistemas automatizados, além dos sensores e atuadores estão sistemas de controle e supervisão, sistemas de transmissão e coleta de dados e aplicações de *software* para a programação do funcionamento do processo (ESTEPA, 2010).

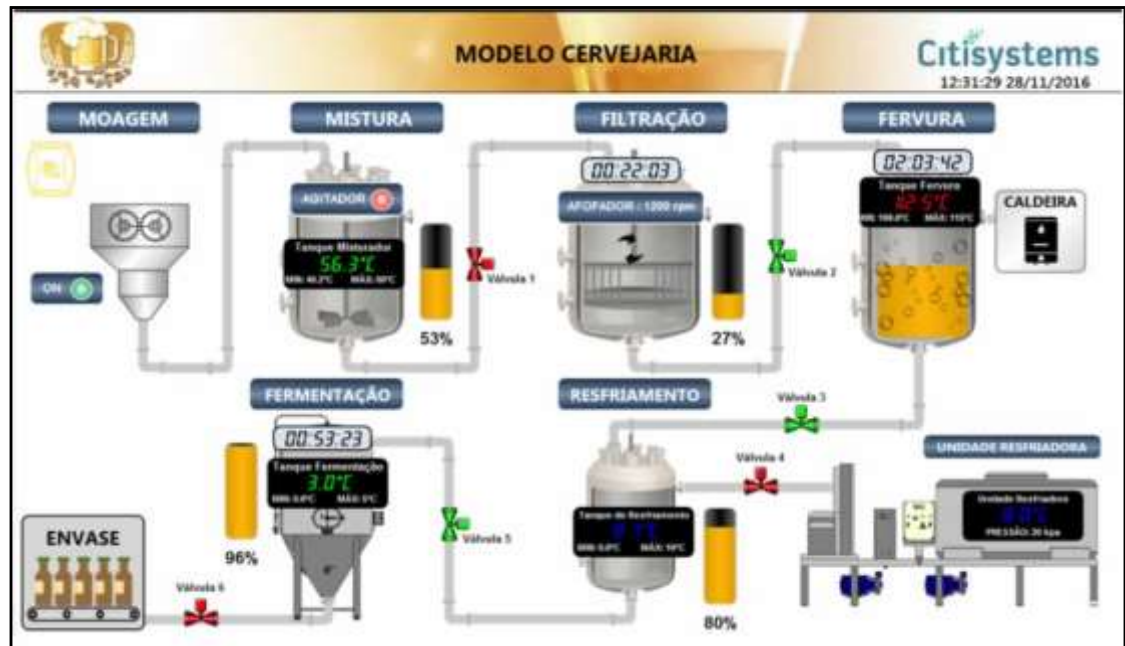
Dentro desse contexto, a aplicação e modernização da automação nas indústrias contribuem de forma significativa para aumentar a possibilidade de sobrevivência das empresas, fazendo com que os sistemas de produção se tornem mais eficientes, possibilitando maior controle de qualidade, repetibilidade, velocidade na produção, facilidade de manipulação, alívio nas condições de trabalho dos operários, segurança, confiabilidade, flexibilidade e produtividade do processo. Isto melhora a competitividade dos produtos manufaturados no mercado atual em que as exigências dos consumidores são cada vez maiores (ESTEPA, 2010).

Um exemplo de aplicação de sistema de automação industrial na indústria alimentícia é o sistema utilizado no processo de produção da cervejaria Imperatriz, pela empresa Citysystems, o sistema é apresentado na Figura 1.

O sistema desenvolvido é capaz de prover o controle dos tempos e temperaturas do processo. Além disso, o sistema permite que o processo como um todo seja direcionado, mesmo que uma determinada etapa não tenha viabilidade de ser controlada ou monitorada (CITYSYSTEMS, 2016).

O sistema fornece informações sobre o ponto do processo em que se encontra, dentro da sequência de produção da cerveja. Comunica todos os eventos importantes que necessitam de intervenção, enviando uma sinalização para o sistema supervisorio. Através do supervisorio, é possível controlar as variáveis de processo como a temperatura, acionamento do moedor do malte, acendimento da chama automatizada de acordo com a temperatura, abertura e fechamento da válvula de gás, de acordo com a lógica de controle e mecanismo de segurança e variáveis de tempo (CITYSYSTEMS, 2016).

Figura 1: Sistema de automação industrial desenvolvido pela empresa Citysystems para a cervejaria Imperatriz.

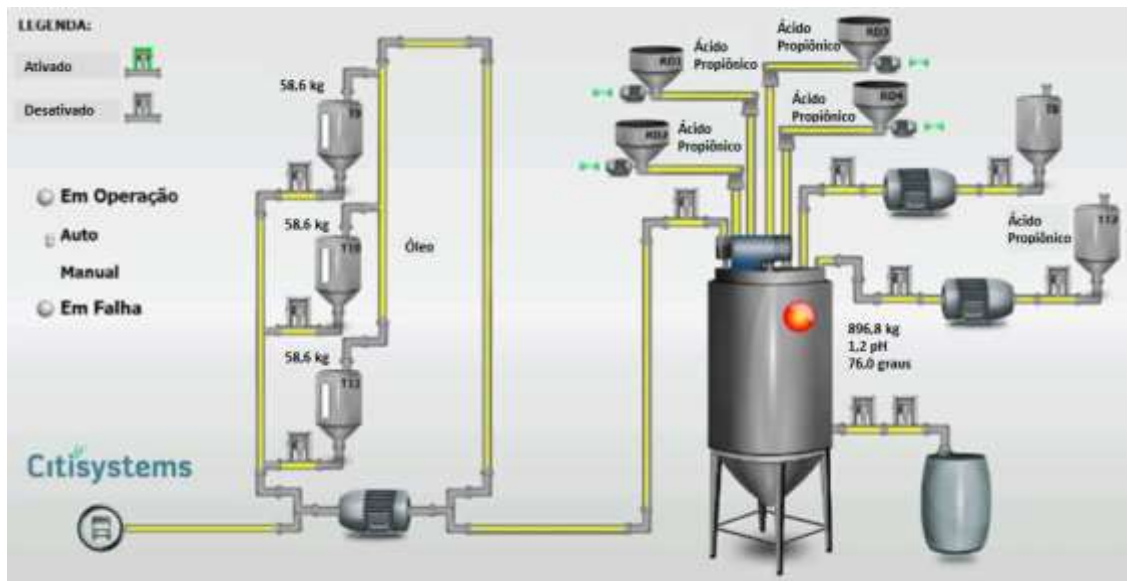


Fonte: Citysystems, 2016.

A aplicação do sistema de automação no processo de produção na cervejaria gera um ganho visível em agilidade, confiabilidade e repetibilidade. A automação de etapas onde a menor variação pode influenciar o produto, possibilita a obtenção de lotes de cerveja muito próximos uns dos outros, fazendo com que a qualidade varie sempre para melhor (FERNANDES, 2011).

Além do sistema desenvolvido para a cervejaria, a Citysystems desenvolveu também um sistema supervisorio para a Qualimix – Nutrição animal. O supervisorio é utilizado para a dosagem de rações animais (líquidos e sólidos) com o monitoramento de níveis, temperaturas e peso, o sistema é apresentado na Figura 2.

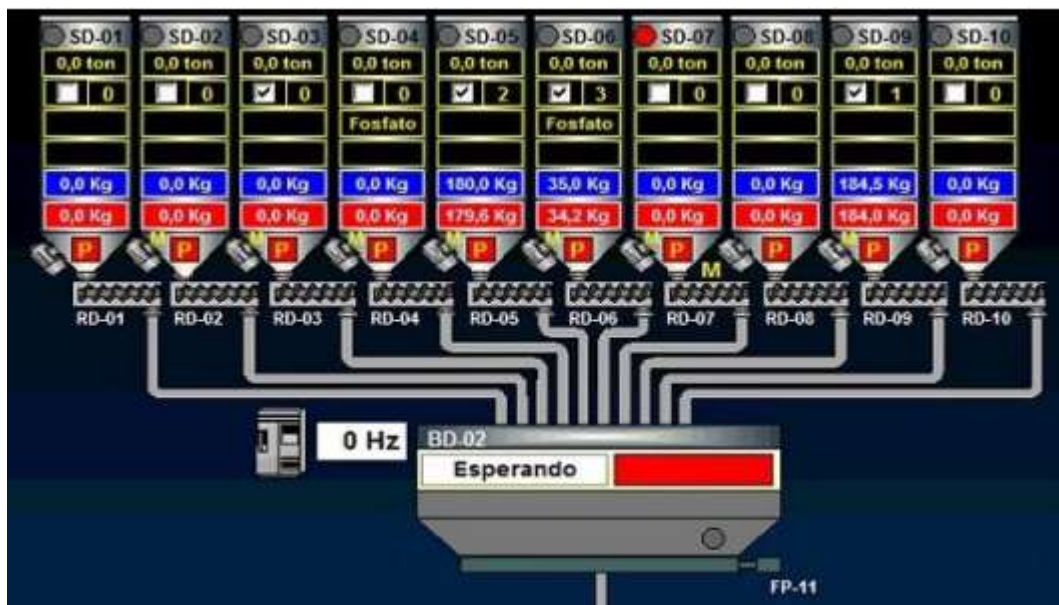
Figura 2: Sistema de automação industrial desenvolvido pela Citisystems para a Qualimix – Nutrição animal.



Fonte: Citisystems, 2016.

O sistema automático de dosagem na fabricação de ração, de maneira geral, pode ser ilustrado pela Figura 3.

Figura 3: Sistema de automação industrial para pesagem em receita de ração animal.



Fonte: Fehrenbach, 2017.

Conforme ilustrado na Figura 3, um exemplo de sistema de dosagem automatizado conta com Silos de Armazenamento (identificados como SD-01 a SD-

10), roscas dosadoras (identificadas como RD-01 a RD-10) e uma balança (identificada como BD-02) (FEHRENBACH, 2017).

O sistema realiza a dosagem de forma automatizada, pesando cada matéria-prima necessária para a receita utilizando sistemas de dosagem como rosca, registro gaveta (guilhotina) e grade hidráulica. Dependendo da matéria-prima e/ou a quantidade a ser dosada, pode-se dividir as formas de dosagem, pois diferentes ingredientes possuem propriedades de escoamento variadas (FEHRENBACH, 2017).

3.2 SENSORES E ATUADORES

Um sensor é definido como um dispositivo que converte um fenômeno físico em sinal elétrico. Como tal, os sensores representam parte da interface entre o mundo físico e o mundo dos dispositivos elétricos, tais como os computadores. A outra parte dessa interface é representada pelos atuadores, que convertem os sinais elétricos em fenômenos físicos (MELCONIAN, 2013).

Os sensores são indispensáveis em qualquer instalação automática, pois eles permitem obter informação sobre o funcionamento dos vários sistemas e da evolução do processo a ser controlado. O processo de medição utiliza sensores acoplados a sistemas de aquisição que convertem o sinal do sensor em um formato tratável por um sistema computacional, o qual por sua vez pode gerar um sinal para os atuadores, visando corrigir o erro do processo (FRADEN, 2003).

Sensores discretos são os que fornecem uma resposta em 1 ou 0, verdadeiro ou falso, ligado e desligado. Esse tipo de sensor é muito utilizado na indústria para detecção de objetos e posicionamento de partes móveis de equipamentos. Como exemplo, cita-se chave fim de curso, sensores de proximidade, nível entre outros (FEHRENBACH, 2017).

Sensores analógicos possuem um transdutor acoplado que fornece, em sua saída, um sinal proporcional a grandeza medida. São alguns exemplos de analógicos o sensor de nível, pressão, temperatura, pesagem, régua potenciométrica, entre outros (FEHRENBACH, 2017).

Os atuadores são dispositivos que modificam uma variável controlada. Recebem um sinal proveniente de um controlador e agem sobre um sistema controlado. Normalmente por possuírem potências mais elevadas, necessitam de uma interface entre ele e o controlador. Exemplos de atuadores são válvulas e cilindros

pneumáticos, relés estáticos e eletromecânicos, motores e solenoides (ALBUQUERQUE; THOMAZINI, 2018).

3.3 CÉLULA DE CARGA

O dispositivo que realiza a operação de pesagem é um sensor chamado célula de carga (Figura 4). Trata-se de uma barra de alumínio, com perfil projetado para sofrer deformação conforme o peso atua sobre ela, e a medição dessa deformação, feita por um extensômetro. Embora pequena, essa deformação é suficiente para gerar um sinal de saída linear e compatível com a carga aplicada (ALBUQUERQUE; THOMAZINI, 2018).

Figura 4: Célula de carga



Fonte: Weigtech, 2021.

As células de carga são dispositivos versáteis que proporcionam desempenho preciso e robusto em uma grande variedade de aplicações. Tais sensores são essenciais em diversos processos industriais e comerciais, da automação na fabricação de automóveis e à pesagem de matérias-primas em indústrias de diversos setores (MELCONIAN, 2013).

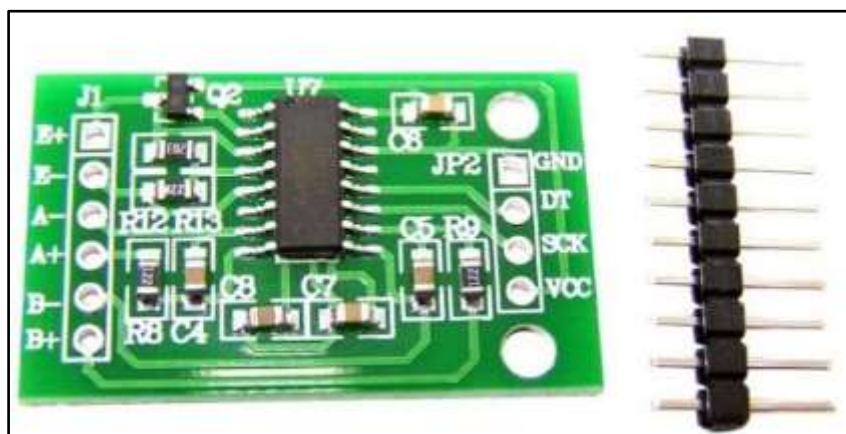
As células de carga baseiam-se na variação da resistência elétrica de um extensômetro quando submetido a deformação, essa variação decorre do estreitamento da seção transversal do extensômetro, provocando pequenas variações de sua resistência elétrica (ALBUQUERQUE; THOMAZINI, 2018).

Quando usamos células de carga, uma extremidade é normalmente presa a uma estrutura ou base, enquanto a outra extremidade é livre para fixar o peso ou o elemento de suporte de peso. Quando a força é aplicada ao corpo da célula de carga, ela se flexiona levemente sob a tensão. Como o circuito do extensômetro está

firmemente colado à célula de carga, ela se deforma com o corpo principal, alterando a resistência elétrica do circuito em proporção ao alongamento do fio. Utilizando eletrônicas de condicionamento de sinal, podemos produzir um sinal de saída proporcional à força aplicada, tendo assim uma balança de carga (MELCONIAN, 2013).

Para realizar o condicionamento do sinal recebido pela célula de carga, é necessário um amplificador de sinal e um conversor análogo/digital. O módulo conversor HX711 (Figura 5) foi desenvolvido pela empresa AVIA Semiconductor para fazer a conversão dos valores analógicos da resistência das células de carga em dados digitais por meio do circuito ADC de 24-bit. Tal conversor possui também um amplificador de sinal que oferece grande precisão (AVIA, 2021).

Figura 5: Módulo conversor amplificado HX711



Fonte: Avia Semiconductor, 2021.

3.4 CLP – CONTROLADOR LÓGICO PROGRAMÁVEL

Um grande salto nos processos automatizados ocorreu com a chegada dos Controle Lógico Programável (CLP) (THOKEIN, 2013). O CLP (Figura 6) é um equipamento eletrônico dotado de *software* e *hardware* capaz de automatizar, controlar e monitorar máquinas, processos específicos ou linhas de produção. Segundo a *National Electrical Manufacturers Association* (NEMA), é um aparelho eletrônico digital que utiliza uma memória programável para armazenar internamente instruções e para implementar funções específicas, tais como lógica, sequenciamento, temporização, contagem e aritmética, controlando, por meio de módulos de entradas e saídas, vários tipos de máquinas ou processos. O programa desenvolvido para um

CLP é totalmente personalizável, o que o torna o CLP um equipamento muito dinâmico que pode ser usado em qualquer processo automático de acionamento e/ou monitoramento de máquinas e processos (BERALDO; BUENO; FERREIRA, 2011).

Figura 6: Controlador Lógico Programável - CLP



Fonte: CSA Automação, 2021.

A Figura 7 representa o processo de funcionamento de um CLP, baseado em microprocessadores: entrada, processamento de informação e saída.

Figura 7: Estrutura básica de funcionamento de um CLP



Fonte: Carrilho, 2005.

Na indústria, há diversos equipamentos presentes na automatização e controle de processos se comunicando. A comunicação pode acontecer do CLP para o PC, do equipamento para o CLP e do CLP para o equipamento, mas para transmitir essa informação, é necessária uma rede de comunicação (BERALDO; BUENO; FERREIRA, 2011).

A comunicação entre sensores, atuadores e os sistemas ocorrem através de redes de comunicação industrial. Cada protocolo de comunicação é um conjunto de regras pré-definidas com o intuito de garantir uma comunicação confiável e precisa entre os equipamentos envolvidos (BERALDO; BUENO; FERREIRA, 2011).

Podemos citar a Ethernet como a principal rede utilizada atualmente, mas existem ainda outros protocolos de comunicação, como DeviceNet, Profibus, entre outras (BERALDO; BUENO; FERREIRA, 2011).

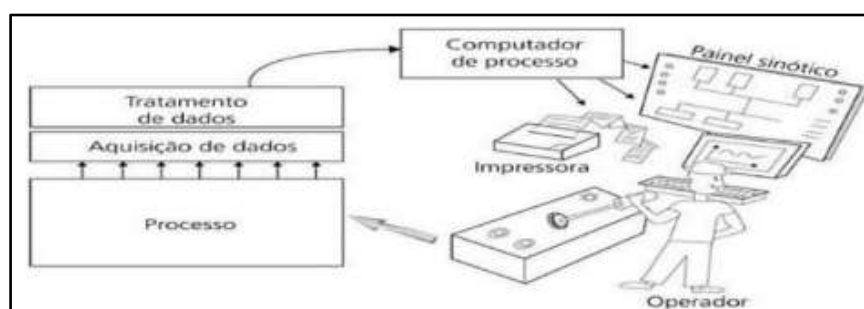
O CLP é um computador robusto e de grande utilidade no ambiente industrial, mas não atua sozinho no “chão de fábrica”. Os vários CLP’s presentes em uma indústria estão ligados através de cabos de rede à computadores. Esses computadores pessoais ficam abrigados em uma sala especial, denominada sala de controle, com todas as características necessárias para o bom funcionamento desses. (BERALDO; BUENO; FERREIRA, 2011).

3.5 SISTEMA DE SUPERVISÃO E AQUISIÇÃO DE DADOS (SCADA)

Um sistema supervisório destina-se a capturar e armazenar em um banco de dados, informações sobre um processo de produção. As informações vêm de sensores que capturam dados específicos (conhecidos como variáveis de processo) da planta industrial. É uma importante ferramenta que permite ao supervisor ou gestor monitorar os processos de uma planta industrial em tempo real, acompanhando as informações desejadas da planta, pois uma falha não identificada inicialmente pode gerar uma parada na linha de produção, acarretando uma perda financeira considerável. Um sistema de monitoramento eficiente permite a rápida detecção do problema, permitindo sua correção de forma mais efetiva (RONCOLI; REGATTIERI, 2018).

A Figura 8 representa o ciclo de informações da aquisição de dados, tratamento de dados e envio para o sistema supervisório.

Figura 8: Fluxo de automação do processo



Fonte: Spinola, 2004.

Os computadores abrigados na sala de controle possuem um *software* denominado sistema supervisório. Esse aplicativo possui a função de monitorar e controlar todo o processo industrial, em comunicação com os CLP's. Através do sistema supervisório é possível definir os diversos parâmetros necessários para o bom funcionamento da produção sem precisar acessar o CLP diretamente. Isto é a grande vantagem desse *software*, pois em um computador pessoal há a possibilidade de criar telas com várias opções de controle, além de poder monitorar e selecionar os vários processos existentes (BERALDO; BUENO; FERREIRA, 2011).

O SCADA é tipicamente uma combinação de elementos de *software* e de *hardware*, como controladores lógicos programáveis (CLPs) e unidades terminais remotas (UTRs) (FAGUNDES, 2021).

A aquisição de dados começa com os CLPs, que se comunicam com os equipamentos de chão de fábrica, como máquinas e sensores. Após a captação e processamento dessas informações, é necessária a utilização de uma Interface Homem-Máquina para permitir o operador controlar e supervisionar o sistema.(BERALDO; BUENO; FERREIRA, 2011). Os dados coletados dos equipamentos são enviados a sala de controle, onde os operadores poderão supervisionar os CLPs usando interfaces homem-máquina (IHMs) (FAGUNDES, 2021).

As IHMs são um elemento importante dos sistemas SCADA. Elas são as telas utilizadas pelos operadores para a comunicação com o sistema SCADA. Os dados adquiridos podem ser apresentados de forma amigável em telas sinópticas com gráficos de tendências, consultas a dados históricos e sinalização de alarmes e falhas (PUPO, 2002).

3.6 ARDUINO

O Arduino (Figura 9) foi criado em 2005 por um grupo de 5 pesquisadores: Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis, com o principal objetivo fornecer uma maneira fácil e de baixo custo para iniciantes e profissionais criarem dispositivos que interajam com seu ambiente usando sensores e atuadores (RENNA et al, 2013). Suas principais características são:

- Baixo custo na prototipagem, permitindo testar o produto com um baixo custo e assim detectar possíveis falhas no projeto.

- Sua construção é de software aberto, permitindo incluir novas funcionalidades alterando os módulos presentes na placa para efetuar serviços específicos.

Figura 9: Arduino Uno



Fonte: Robocore, 2021.

A principal finalidade do Arduino num sistema é facilitar a prototipagem, implementação ou emulação do controle de sistemas interativos, a nível doméstico, comercial ou móvel, da mesma forma que o CLP controla sistemas de funcionamento industriais. Com ele é possível enviar ou receber informações de basicamente qualquer sistema eletrônico, como identificar realizar medições de nível em recipientes, automação de casas, aplicações na área de impressão 3D, robótica, engenharia de transportes, engenharia agrônômica e na área musical (RENNA et al, 2013).

Além de prototipagem, esse dispositivo se mostra muito versátil, pois auxilia em várias áreas de aplicação. Dentre alguns trabalhos publicados sobre o uso do microcontrolador Arduino, destacam-se:

Gimenes e Pereira (2015) construíram um sistema de controle de nível de líquidos em um reservatório por meio de um sistema com o Arduino UNO e um investimento total de R\$291,50. Concluíram que o sistema construído funcionou com excelência, e seu custo foi muito baixo se comparado a outros sistemas já utilizados na indústria.

Muynarsk (2014) tiveram ótimos resultados em seu projeto de monitoramento de motores em uma indústria, por meio da plataforma Arduino. O sistema

acompanhava constantemente o funcionamento de motores e identificava defeitos rapidamente, dessa forma, reduzindo o tempo de parada e de manutenção.

Keler et al. (2018) desenvolveram um equipamento de proteção coletiva contra a inalação excessiva de gás amônia por vazamento, o equipamento foi construído utilizando um sensor de gás ligado a um Arduino, que media o índice de gás amônia e acionava uma sirene caso esse índice fosse maior que o permitido, garantindo assim a segurança dos operadores.

Junior et al. (2019) construíram um sistema de automação na medição de temperatura de uma indústria de componentes eletrônicos utilizando Arduino Uno. O processo produtivo anterior à automação se baseava em julgamento humano do operador que, após verificar a temperatura no visor de um termo higrômetro, anotava em uma planilha impressa e comparava com as especificações de temperatura da pasta de solda, liberando ou não a produção. Após a automação, os autores concluíram que houve maior eficiência na medida e diminuição de erros humanos no processo.

3.7 ARDUINO VERSUS CLP

O CLP é um sistema projetado para ser robusto, ele vem com certificações industriais e tem muitas características de segurança, suportam altas e baixas temperaturas, trabalhando em alta eficiência mesmo sem refrigeração, com vida útil de 15 a 20 anos (RAJA, 2020).

Em um certo nível, o CLP e o Arduino funcionam de maneira similar, ambos são controladores lógico programáveis, porém, o CLP é mais utilizado em aplicações industriais mais hostis (RAJA, 2020). Dentre as condições hostis pode-se citar: temperaturas extremas; umidade elevada; ar de automação com sólidos particulados ou com alto teor de óleo; altas vibrações. (RAJA, 2020). A Tabela 1 apresenta diferenças entre o Arduino e o CLP.

Tabela 1: Comparação entre o Arduino e o CLP

Arduino	CLP
Programação em C/C++	Programação em Ladder, Structured
Baixo custo	Text, Instruction List e GRAFCET
Diversas aplicações	Alto custo
Não consegue trabalhar em condições	Aplicações industriais
hostis	Trabalha em condições hostis
Performance satisfatória	Alta performance
Pequeno e compacto	Grande e robusto

Fonte: RAJA (2020).

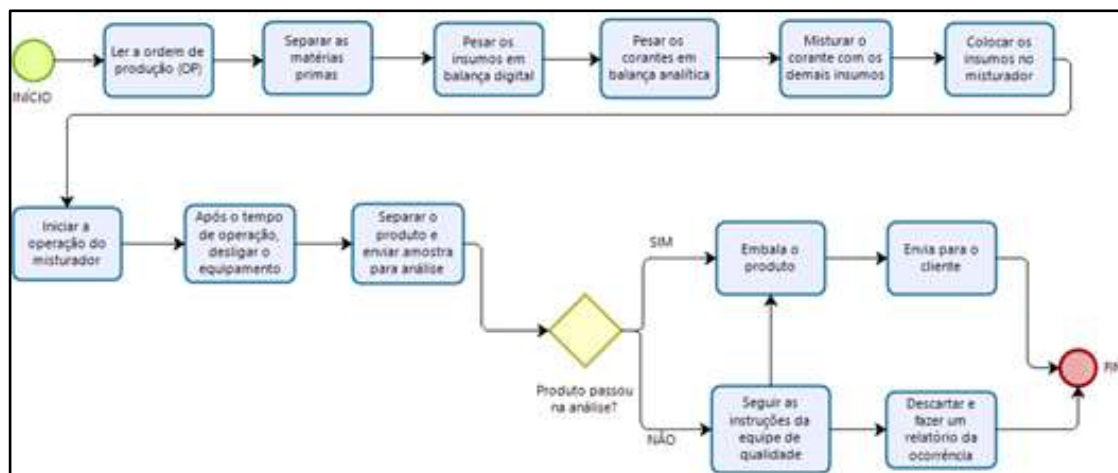
4. METODOLOGIA

No presente trabalho foi desenvolvido um protótipo de sistema de pesagem automatizado utilizando Arduino, Excel e Python, onde serão monitoradas e controladas operações e a variável de peso do processo, armazenando as informações da produção de maneira automática, eliminando a necessidade de anotações em papel e o retrabalho de inserir essas informações nas planilhas eletrônicas.

4.1 PROCESSO PRODUTIVO ATUAL DA EMPRESA

O processo de fabricação de pós alimentícios é dividido em oito etapas: Recebimento da matéria-prima; Armazenagem; Pesagem; Produção; Análise de qualidade; Envase; Embalagem; Armazenamento; Expedição, apresentados na Figura 10.

Figura 10: Fluxograma do processo produtivo atual



Fonte: Autoria própria, 2021.

A matéria-prima chega até a fábrica através de caminhões, a mercadoria é conferida e anota-se a quantia, código da matéria-prima, validade e lote. A quantidade anotada é posteriormente transferida para o Excel para realizar a gestão do estoque.

Após ser gerada a ordem de produção (OP), a matéria-prima necessária é separada e levada até a sala de produção de pós alimentícios. Pesa-se a matéria-prima necessária conforme a receita estabelecida na OP e as separa em sacos plásticos.

Posteriormente, pesa-se os corantes em balança analítica, é necessário precisão na pesagem dos corantes, visto que a quantidade utilizada conforme a receita é muito pequena, e caso haja desvios irá alterar a cor do produto final.

Com a receita do formulado em pó separada, o misturador Y é abastecido, fechado e ligado, operando o tempo necessário conforme estabelecido em cada receita.

A produção é por bateladas de aproximadamente 75 a 100 kg, dependendo da quantia que se deseja produzir. O equipamento consegue atingir uma produção de até 200 kg/h dependendo do produto.

Após passado o tempo estabelecido, o produto final é retirado do misturador e armazenado em sacos plásticos, é então enviada uma amostra do produto para o laboratório, que faz a análise de qualidade, caso o produto final seja aprovado, ele é separado e enviado para a etapa de envase.

Na etapa de envase, o produto final é pesado manualmente e colocado em embalagens primárias de 200 a 500 gramas, variando de acordo com o produto final.

Feita a embalagem primária, os produtos são colocados em embalagem secundária, uma embalagem maior contendo 10 pacotes em cada.

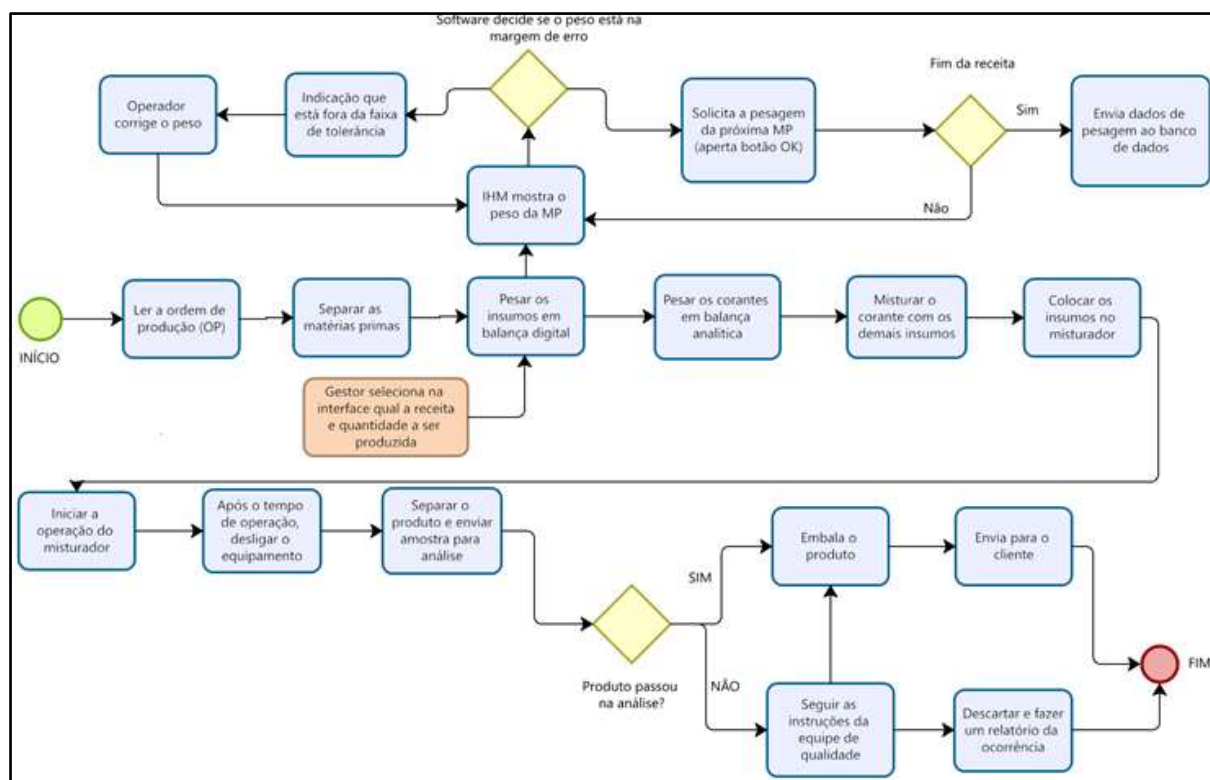
A empresa envia caixas de 20 kg de produto final para o cliente, a quantidade de pacotes necessários para preencher a caixa é dependente do produto.

4.2 PROCESSO PRODUTIVO PÓS IMPLEMENTAÇÃO DO SISTEMA DE AUTOMAÇÃO EM PESAGEM

A mudança do processo se dará na etapa de pesagem da matéria-prima. Inicialmente o gestor irá selecionar em uma interface qual a receita que deseja produzir e a quantidade que se deseja produzir. Após clicar em confirmar, o Arduino é informado automaticamente o ingrediente e o peso necessário.

O operador irá verificar na Interface Humano-Máquina qual o peso necessário e a matéria-prima a ser pesada, após o operador colocar o ingrediente e apertar no botão de confirmar, o Arduino irá detectar se o peso está dentro da faixa de tolerância permitida para o ingrediente, valor esse determinado em 5%, garantindo uma melhor precisão na hora da pesagem. Caso não esteja dentro da faixa, será disparado um erro para o operador corrigir o peso, já se o peso estiver dentro da faixa, será enviado o peso, nome do ingrediente, horário e data para o banco de dados e solicitará a pesagem do próximo ingrediente ao operador. O fluxograma pós implementação do sistema de automação em pesagem é apresentado na Figura 11.

Figura 11: Fluxograma do processo após o processo de automação.



Fonte: Autoria própria, 2021.

4.3 PROTÓTIPO CONSTRUÍDO COM ARDUINO UNO

4.3.1 MATERIAIS

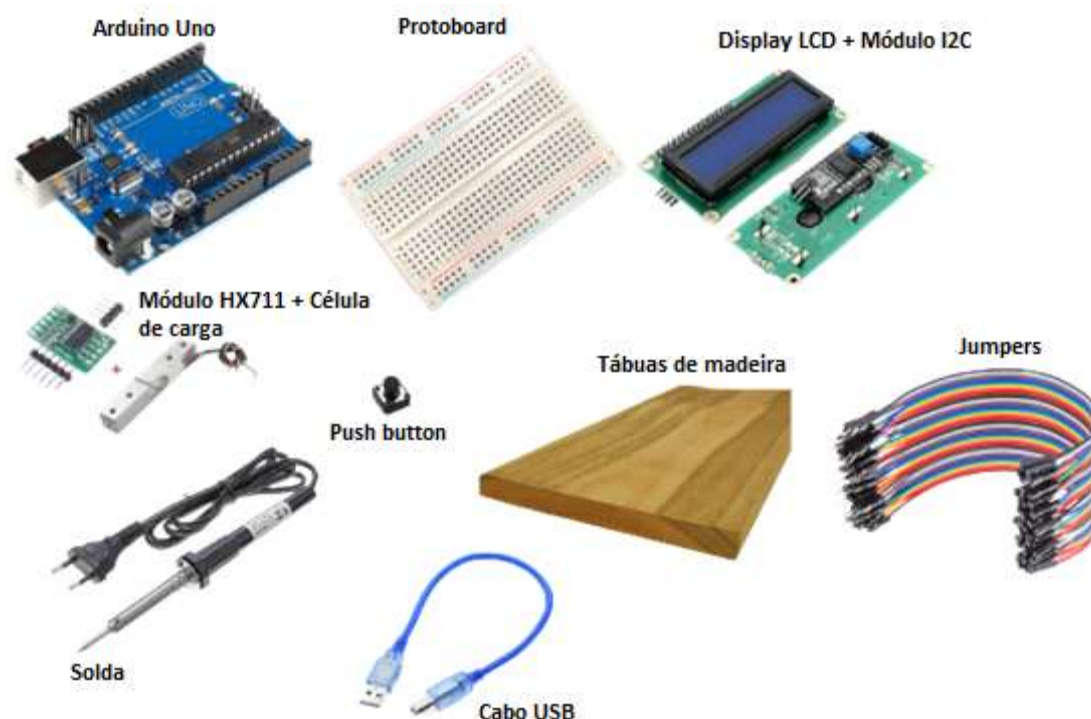
Para a criação do protótipo, foram necessários os seguintes materiais:

- Arduino: O Arduino utilizado foi Arduino Uno, devido ao seu baixo custo e vasta gama de possibilidades de criação de projetos
- Protoboard: Utilizada devido a facilidade de inserção de componentes eletrônicos
- Célula de carga: Quando o material for colocado na balança, a célula de carga irá se deformar levemente, enviando um sinal elétrico para o módulo HX711
- Módulo HX711: Utilizado para amplificar os sinais recebidos pela célula de carga, o módulo irá enviar os sinais recebidos para o Arduino
- Tábuas de madeira, utilizadas para montar a estrutura da balança
- Display LCD 16x2 para Arduino
- Módulo I2C para o Display 16x2

- 2x Push Button, utilizados para realizar a tara e efetivar a pesagem
- Jumpers Macho e Fêmea
- Solda: Utilizada para soldar os componentes no módulo HX711
- Cabo de energia USB: Para ligar o Arduino

Os materiais são apresentados na Figura 12.

Figura 12: Materiais utilizados na montagem do protótipo



Fonte: Autoria própria, 2021.

4.3.2 MONTAGEM DA ESTRUTURA FÍSICA DA BALANÇA

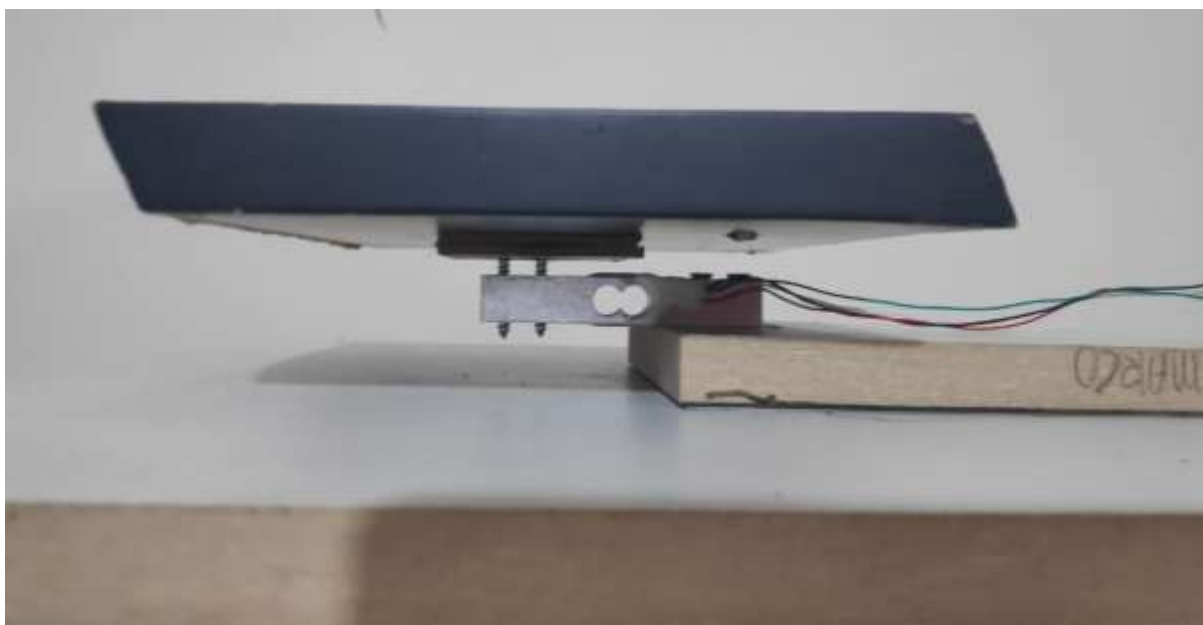
O projeto da balança possui algumas propriedades isoladas que devem ser levadas em consideração, principalmente quanto a sua instalação e os métodos de pesagem. Se a montagem não for executada de forma correta podem ocorrer erros na leitura.

A célula de peso deve ser instalada de maneira que seu centro fique livre, proporcionando uma certa distância do sensor entre a base de instalação e a base de coleta de dados. O objetivo é que qualquer força aplicada à placa móvel faça com que

a célula de carga se curve. Sem espaçadores, a carga seria diretamente transferida da placa móvel para a placa da base, sem afetar a célula de carga.

A estrutura da balança com a célula de carga construída pelo autor é apresentada na Figura 13.

Figura 13: Disposição da célula de carga no protótipo do autor.



Fonte: Autoria própria, 2021.

4.3.3 CONEXÕES DO ARDUINO

Após construída a estrutura física da balança, é necessário realizar as conexões do sensor de peso com o módulo HX711 e o Arduino. A conexão é feita fundamentalmente de duas fases, uma delas envolvendo a ligação da célula de carga no módulo HX711 e a outra entre o módulo HX711 e o Arduino. O esquema de ligação entre a célula de carga e o módulo é apresentado na Tabela 2, já entre o módulo e o Arduino é apresentado na Tabela 3.

Tabela 2: Conexões entre a célula de carga e o módulo HX711

Célula de Carga	Módulo HX711
Fio Vermelho	Pino E+
Fio Preto	Pino E-
Fio Verde	Pino A-
Fio Branco	Pino A+

Fonte: Autoria própria, 2021.

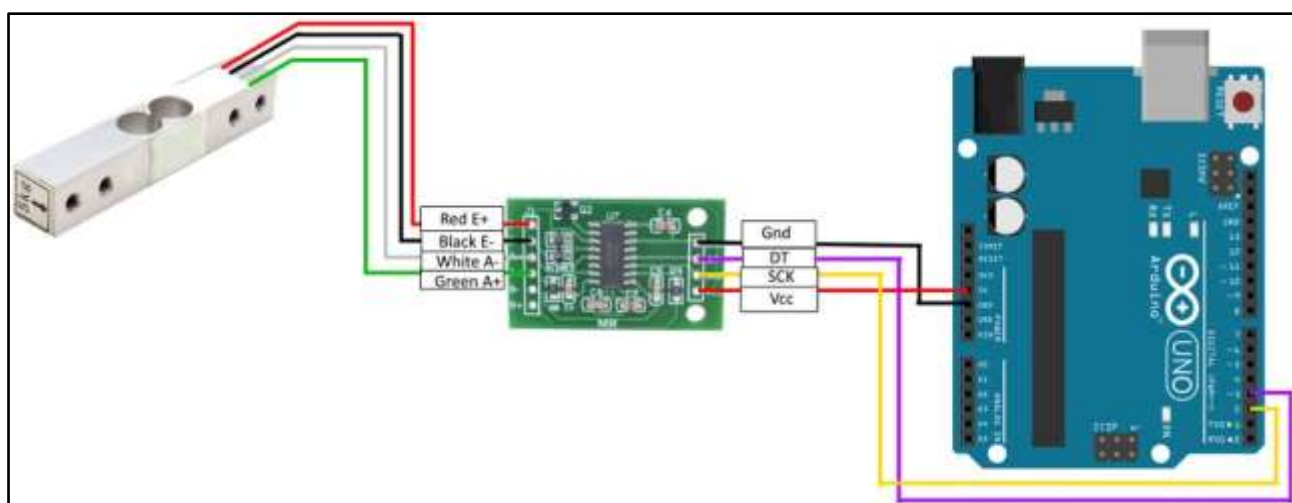
Tabela 3: Conexões entre o módulo HX711 e o Arduino

Célula de Carga	Módulo HX711
Pino GND	Pino GND
Pino DT	Pino 3
Pino SCK	Pino 2
Pino VCC	Pino 5V

Fonte: Autoria própria, 2021.

A Figura 14 apresenta o esquema completo de conexões entre a célula de carga, o módulo e o Arduino.

Figura 14: Conexões entre a célula de carga, o módulo HX711 e o Arduino



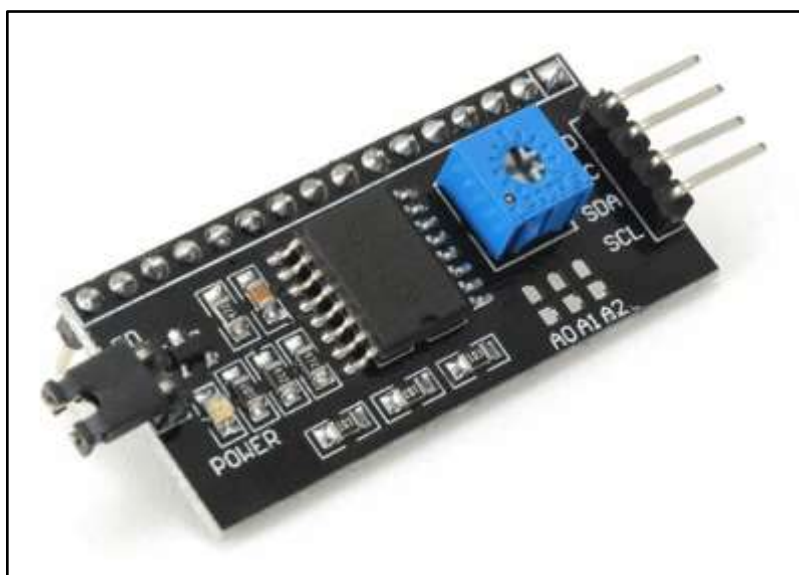
Fonte: Instructables, 2021.

O display LCD (Liquid Crystal Display) permite uma interface visual entre homem e máquina (HMI), barata e simples de usar. No LCD é transmitido os dados que estão sendo coletados pelo Arduino, permitindo o colaborador visualizar a medida

de peso. Este recurso irá nos possibilitar utilizar a balança sem a necessidade de um computador por perto para a exibição dos valores obtidos, elemento que faltava para completar a Balança Arduino.

O display LCD utiliza pelo menos 6 jumpers para conexão (ARDUINOECIA, 2014). Em placas com um número menor de portas, como o Arduino Uno, isso significa sacrificar algumas portas que poderiam ser utilizadas para ligação de outros componentes, como sensores ou motores. Para contornar esse problema, pode ser utilizado o módulo I2C para display LCD, apresentado na Figura 15 (ARDUINOECIA, 2014).

Figura 15: Módulo I2C para display LCD



Fonte: Arduinoecia, 2021.

O módulo possui 16 pinos que podem ser ligados diretamente ao display ou a protoboard, o esquema de ligações é mostrado na Tabela 4.

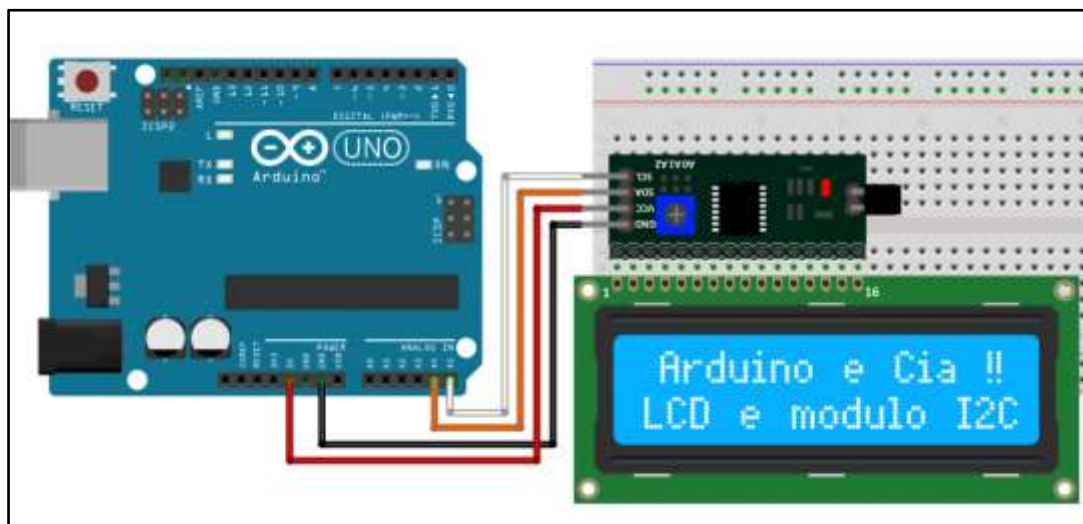
Tabela 4: Conexões entre o módulo I2C, display LCD e o Arduino

Módulo I2C	Arduino
Pino VCC	Pino 5V
Pino GND	Pino GND
Pino SDA	Pino A4
Pino SCL	Pino A5

Fonte: Autoria própria, 2021.

A Figura 16 apresenta visualmente a conexão entre o Arduino e o módulo I2C na protoboard.

Figura 16: Conexões com o módulo I2C para display LCD



Fonte: Arduinoecia, 2021.

Já foi construído a estrutura física da balança Arduino e a visualização dos dados, mas para que o operador consiga utilizá-la e enviar os dados para o computador, são necessários botões, um botão, que será utilizado para realizar a tara da balança, e outro botão que será utilizado para efetivar a pesagem.

O *Push button* (botão de pressão) é uma chave que, quando pressionado o botão, ela abre ou fecha o circuito, convertendo assim, um comando mecânico em elétrico. Geralmente eles têm um contato de ação momentânea, abrindo ou fechando o circuito apenas de modo momentâneo (SILVER, 2020). O *Push button* é apresentado na Figura 17.

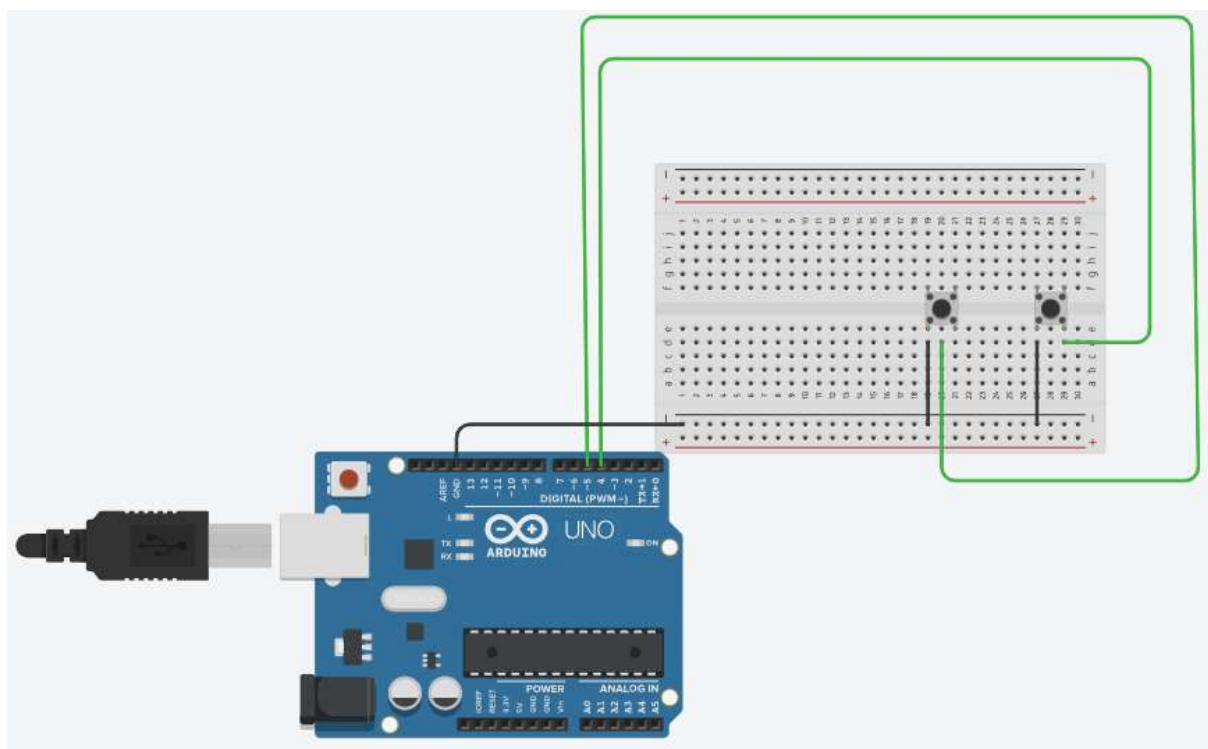
Figura 17: Push button utilizado no Arduino.



Fonte: Silver, 2020.

As chaves podem ser do tipo NA (Normalmente aberta) que quando pressionado fecha o circuito permitindo assim a passagem de corrente, e mandando um sinal *HIGH* (1) para o Arduino ou uma chave do tipo NF (normalmente fechada), que quando pressionada, abre o circuito impedindo a passagem de corrente, mandando assim um sinal *LOW* (0) para o Arduino (SILVER, 2020). As conexões são apresentadas na Figura 18.

Figura 18: Conexões com o *Push button* utilizado no Arduino.



Fonte: Autoria própria, 2021.

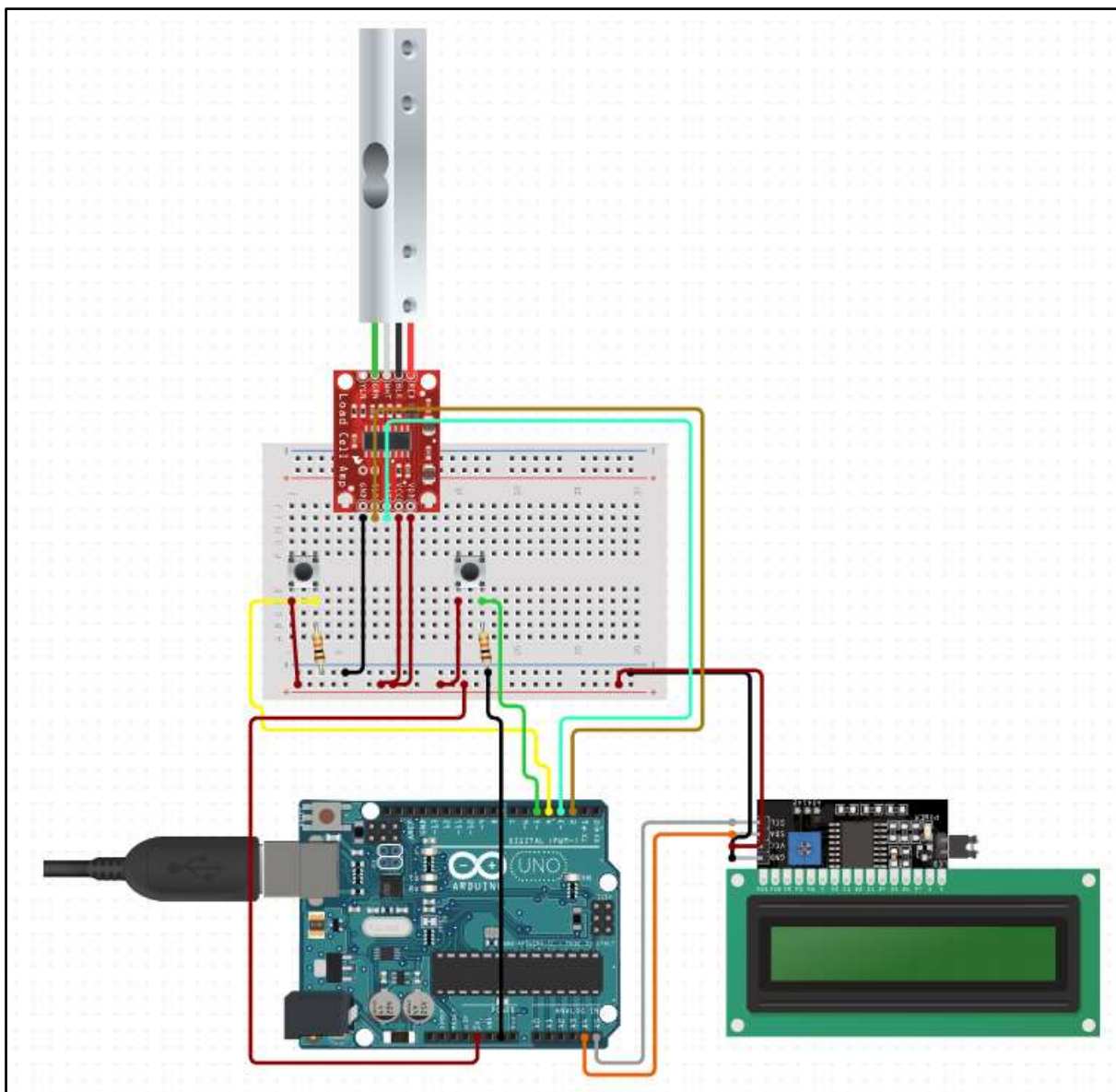
Para o correto funcionamento dos comandos e dos dados do protótipo, se faz necessário a interligação de todos os equipamentos exatamente como o descrito conforme Tabela 5 e Figura 19. Essa é a parte no qual chamamos de hardware em um sistema de automação de balança utilizando Arduino.

Tabela 5: Interligações do Arduino

Interligações do Arduino	
Pino 5V	Conectado na fileira positiva da protoboard para a alimentação dos equipamentos
Pino GND	Conectado na fileira negativa da protoboard para a alimentação dos equipamentos
Pino 2	Conectado ao pino DT do módulo HX711
Pino 3	Conectado ao pino SCK do módulo HX711
Pino 4	Conectado ao botão utilizado para efetivar a pesagem
Pino 5	Conectado ao botão utilizado para tarar a balança
Pino A4	Conectado ao pino SCA do módulo I2C do display
Pino A5	Conectado ao pino SCL do módulo I2C do display

Fonte: Autoria própria, 2021.

Figura 19: Esquema completo de conexões do projeto.



Fonte: Autoria própria, 2021.

4.3.4 SOFTWARES

Para escrever um código que possibilita o Arduino executar suas funções, é necessário o uso do *Arduino Integrated Development Environment* (Arduino IDE), uma aplicação de plataforma cruzada, escrita nas linguagens de programação C e C++ (ARDUINO, 2021).

O Arduino IDE é um software gratuito que facilita o desenvolvimento e a gravação de códigos diretamente no microcontrolador. Através deste, é possível

realizar o *upload* dos códigos para a placa tanto em sistemas operacionais Windows quanto Linux (USINAINFO, 2021).

O Arduino utiliza uma linguagem de programação denominada *Wiring*, que é adaptada a linguagem *Processing*. O que torna esse tipo de linguagem simples é o fato de que quando fazemos *upload* do código escrito, ele é traduzido para a linguagem C. (CAMPOS, 2014).

O sistema de gerenciamento de banco de dados escolhido foi o Excel. A empresa que o autor realizou o estágio obrigatório já utiliza o Excel para o controle de receitas, bem como a gestão da produção e de estoque, com isso, facilitaria para o gestor a utilização de receitas já criadas e do manejo dos dados coletados.

A linguagem de programação utilizada será o Python, uma linguagem de programação de alto nível, interpretada de script, imperativa, orientada a objetos, de tipagem dinâmica e forte.

O fato de o autor já ter utilizado Python, bem como sua vasta gama de bibliotecas prontas que facilitam o desenvolvimento, contribuíram para a adoção dessa linguagem.

As bibliotecas são funcionalidades a mais para utilizarmos códigos com função de programar dispositivos novos ou manipular dados e comando. Quando utilizamos a biblioteca, estamos dando uma ampla expansão de programação quando comparados as funções básicas do Arduino, para utilizarmos uma biblioteca, basta acessarmos o Arduino IDE, nele já temos algumas bibliotecas pré-instaladas. Mas para novas bibliotecas, devemos baixar e adicioná-las na pasta de bibliotecas do Arduino IDE (TOFÓLI, 2014).

No projeto, serão utilizadas bibliotecas pré construídas para facilitar o seu desenvolvimento.

Todo o código do protótipo desenvolvido é apresentado nos Apêndices A e B.

4.3.5 BIBLIOTECAS UTILIZADAS NO ARDUINO IDE

4.3.5.1 HX711

Biblioteca *open source*, teve o início de seu desenvolvimento em 2012 por Weihong Guan em 2012, posteriormente, em 2014, Bogdan Necula assumiu o projeto.

A biblioteca será utilizada para realizar a leitura do módulo HX711, além de utilizar suas funções de tara e calibração.

4.3.5.2 LCD_I2C

Biblioteca *open source* desenvolvida em 2020 por David Aristi, a biblioteca em questão é utilizada para enviar ao módulo LCD as leituras realizadas pela balança, bem como a pesagem do ingrediente necessária para a receita, e mostrar mensagens de erro e sucesso ao realizar uma pesagem.

4.3.5.3 PUSHBUTTON

Biblioteca utilizada para realizar a leitura dos botões e enviar um sinal ao Arduino caso o colaborador o pressione.

4.3.5.3 REXCEL

Desenvolvida por Roberto Valgolio, com seu início em 2015, rExcel é uma biblioteca *open source* utilizada para fazer a interface de conexão entre o Arduino e o Excel, trocando dados em tempo real, em ambas as direções, entre os dois.

Para tal fim, a biblioteca utiliza uma planilha do Excel com código em VBA para realizar a leitura, sendo necessário instalar a mesma através do site do autor.

4.3.6 BIBLIOTECAS UTILIZADAS NO PYTHON

4.3.6.1 XLWINGS

Biblioteca open source e gratuita, utilizada para automatizar o Excel através do Python, é utilizada para trocar dados em tempo real, dessa forma, podendo escolher a receita a ser produzida, bem como o peso necessário a ser produzido.

4.3.6.2 XRLD

Com o início de seu desenvolvimento em 2005 por Stephen Machin, a biblioteca é utilizada para realizar as leituras das planilhas do Excel, no projeto em questão, a planilha foi utilizada para conseguir detectar quais seriam as planilhas de receitas necessárias a serem visualizadas na interface do Python.

4.3.6.3 PYSIMPLEGUI

Foi lançada oficialmente em 2018, PySimpleGui é uma biblioteca amplamente utilizada dentro do Python para a criação de interfaces, com ela, é possível criar uma interface para o usuário final realizar suas ações, sem a necessidade de executar o Software através de linhas de comando.

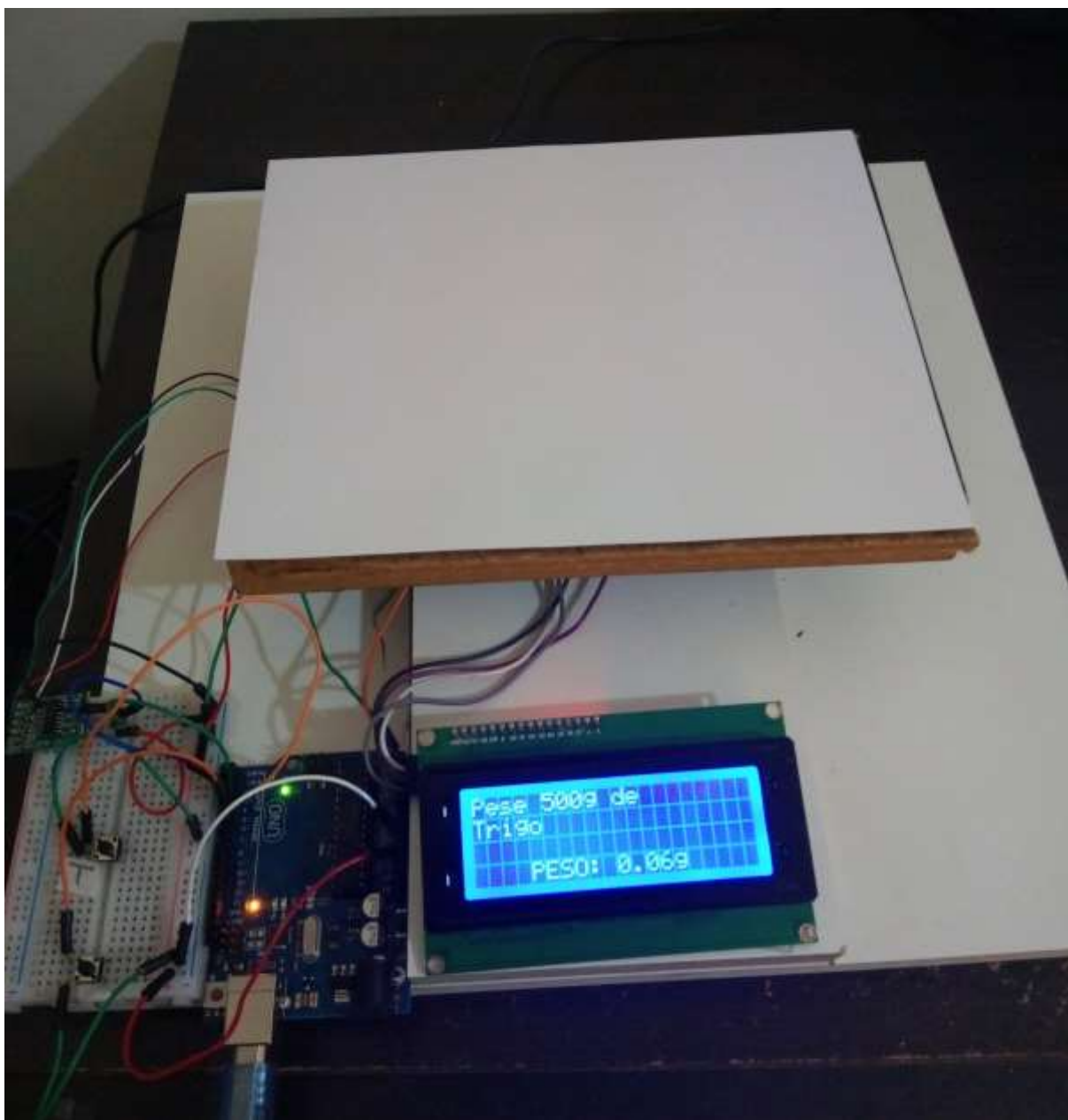
5. RESULTADOS E DISCUSSÃO

Como se trata de uma prototipagem para fins acadêmicos, foram realizados apenas testes de implementação em escala reduzida. Porém, por meio de um estudo e avaliação é possível constatar sua viabilidade comercial e sua aplicação em ambientes diversos, como empresas e outras indústrias que utilizam o processo de pesagem.

5.1 PROTÓTIPO

Com os procedimentos descritos na metodologia, a montagem do protótipo foi concluída, conforme pode ser observado na Figura 20.

Figura 20: Protótipo de balança construída com Arduino.



Fonte: Autoria própria, 2021.

5.2 CALIBRAÇÃO

Após instaladas todas as bibliotecas necessárias e com a estrutura da balança e suas conexões com o Arduino montadas, é necessário realizar a calibração da balança.

A calibração de célula de carga consiste em determinar a relação entre a tensão lida na ponte (no caso retornada pelo HX711) e a força nela aplicada (no caso, o peso do objeto colocado na balança).

Para o projeto em questão, o código utilizado para realizar a calibração foi desenvolvido pelo canal Brincando com Ideias e é apresentado no Anexo A.

A calibração é realizada variando a função `scale.set_scale` apresentada no Anexo A, a função recebe de parâmetro um valor, que conforme é alterado, influencia na calibragem do Arduino.

Foi utilizado um objeto de massa conhecida para realizar a calibração, o autor variou o seu parâmetro até atingir um valor de 104500 para a escala, conforme é apresentado no código fonte do Apêndice A.

A Figura 21 apresenta a comparação entre a pesagem de um objeto utilizando uma balança convencional, após ser pesado, o mesmo objeto foi colocado na balança Arduino para confirmar a sua calibração, assim definindo o fator de calibração de 104500 para o protótipo em questão.

Figura 21: Calibração confirmada com a Balança Arduino



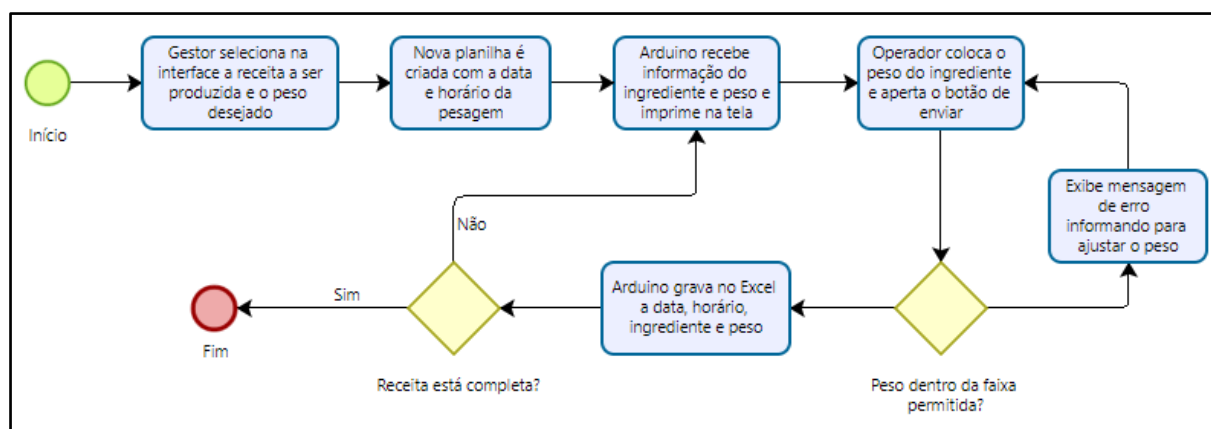
Fonte: Autoria própria (2021).

A matéria-prima de maior peso utilizada no pré-mix que é feito na Hoshino Rizzo pesa menos de 2 kg, o fator de calibração de 104500 foi testado para o peso máximo de 5 kg em um ingrediente, demonstrando ótima precisão, dessa forma, foi determinado satisfatório.

5.3 FLUXO DE FUNCIONAMENTO DA APLICAÇÃO

O fluxograma de funcionamento do Software é apresentado na Figura 22, cada etapa do funcionamento será apresentada em sequência.

Figura 22: Fluxograma de funcionamento do Software



Fonte: Autoria própria, 2021.

Inicialmente, se faz necessário o usuário adicionar manualmente as receitas em planilha do Excel, o nome da planilha será o nome da receita, e é necessário com que o usuário adicione as frações e o nome de cada ingrediente a ser pesado, os *inputs* são apresentados na Figura 23.

Figura 23: Demonstração de planilha de receitas

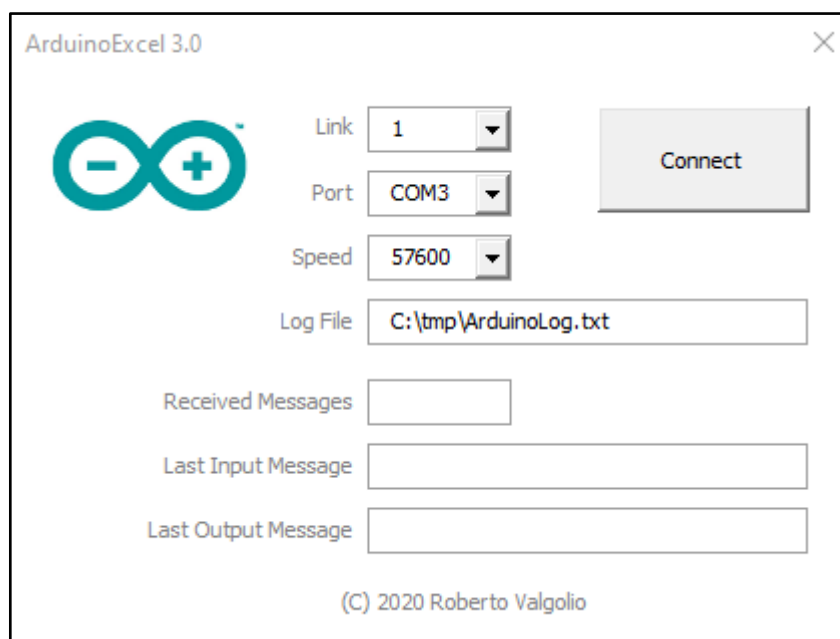
	A	B	C	D	E
1	Pudim	Peso (g)	Fração %		Total a ser produzido (kg)
2	Acucar	0	20%		
3	Sal	0	5%		
4	Trigo	0	10%		
5	Óleo	0	5%		

Fonte: Autoria própria, 2021.

Visto que na indústria que o autor estagiou já utilizavam planilha de Excel para controle de receitas, entre outros, seria necessário apenas repassar para essa nova planilha o que já está feito.

Adicionadas as receitas, é necessário conectar o Arduino com o Excel. Após abrir o Excel, irá aparecer a tela apresentada na Figura 24, apertando em “Connect” irá fazer com que o Arduino escute as ações que ocorrem na planilha.

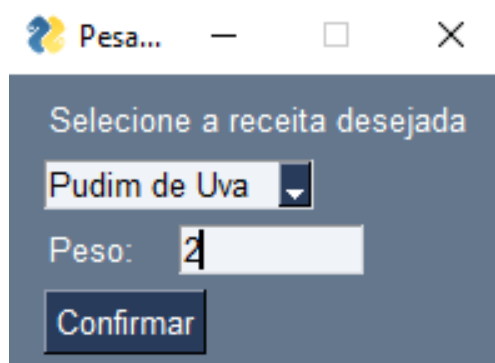
Figura 24: Tela para conectar o Arduino e o Excel



Fonte: Autoria própria, 2021.

Com o Arduino e o Excel conectados, a aplicação se inicia ao usuário selecionar na interface apresentada na Figura 25 qual a receita que se deseja produzir e o peso a ser produzido.

Figura 25: Interface construída no Python para o usuário selecionar a receita e o peso a ser produzido



Fonte: Autoria própria, 2021.

Ao clicar em confirmar, é enviado um comando ao Excel para alterar a quantidade a ser produzida e mostrar o peso necessário de cada ingrediente, conforme mostrado na Figura 26. Ainda, é criada uma nova planilha no Excel com a pesagem dessa nova produção, apresentada na Figura 27.

Figura 26: Planilha de receitas após usuário selecionar o peso na interface

	A	B	C	D	E
1	Pudim	Peso (g)	Fração %		Total a ser produzido (kg)
2	Acucar	400	20%		2
3	Sal	100	5%		
4	Trigo	200	10%		
5	Óleo	100	5%		

Fonte: Autoria própria, 2021.

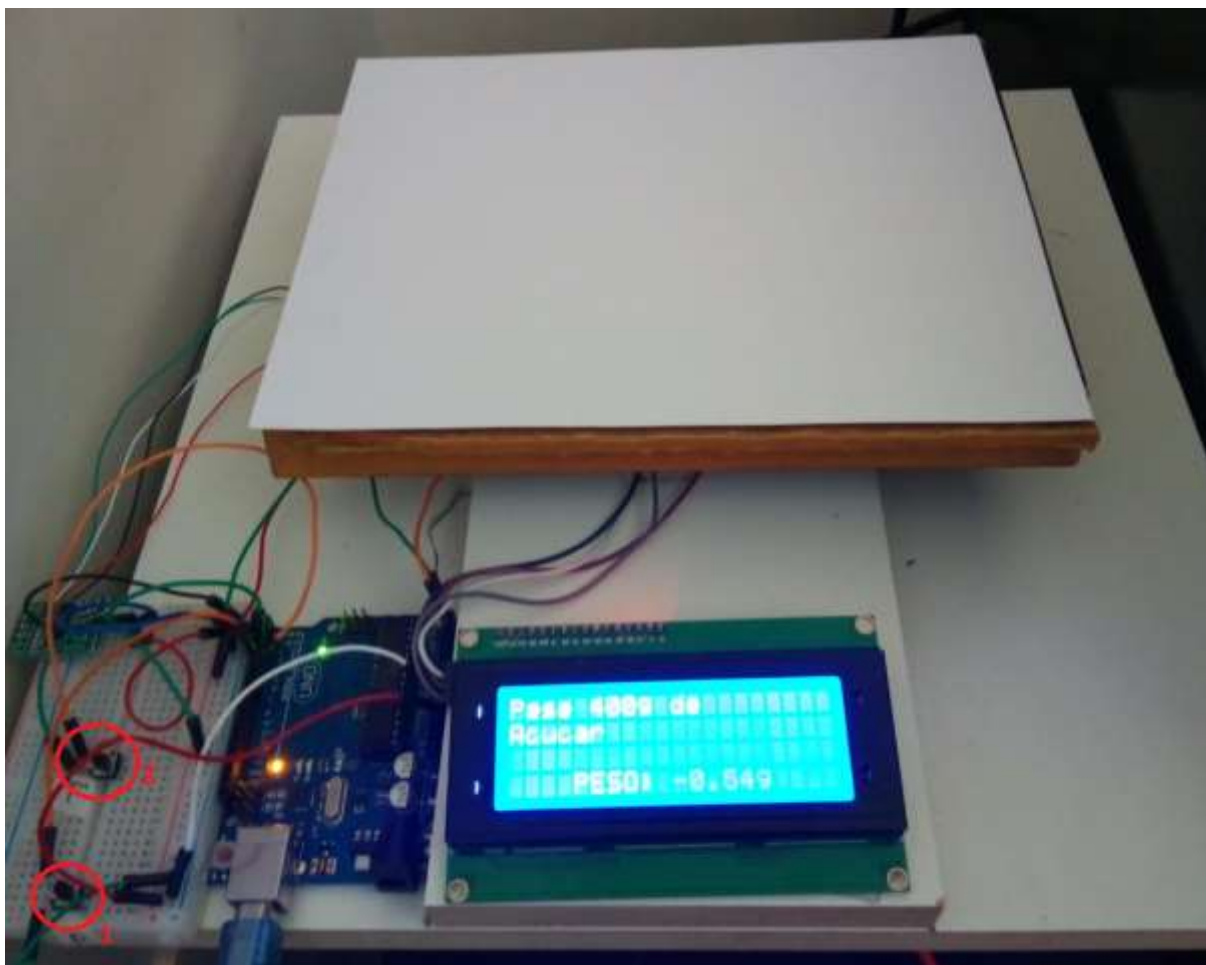
Figura 27: Planilha de pesagem criada automaticamente após confirmado o peso e receita a serem produzidos

	A	B	C	D	E
1	Ingrediente	Peso(g) receita	Valor pesado (g)	Data	Horario
2					
3					
4					

Fonte: Autoria própria, 2021.

No protótipo há dois botões, um botão que possui a funcionalidade de realizar a tara, já o outro botão possui a finalidade de enviar o ingrediente, seu peso, data e horário pro Excel, caso o peso informado esteja dentro da faixa válida de peso. Determinou-se um desvio de 5% no peso para o ingrediente, valor determinado de forma a adquirir precisão satisfatória na pesagem de matérias-primas. Na Figura 28, é apresentado o protótipo já mostrando ao usuário qual o ingrediente e o peso necessário da receita, o botão circulado “1” no protótipo é o botão de enviar os dados, já o botão “2” é o botão de realizar a tara.

Figura 28: Protótipo informando ao usuário o peso e o ingrediente



Fonte: Autoria própria, 2021.

Ao colocar o peso do ingrediente na balança e pressionar o botão “1”, caso o peso não esteja dentro da faixa especificada, irá exibir uma mensagem de erro informando ao usuário para corrigir o peso, é apresentado na Figura 29.

Figura 29: Mensagem de erro ao colocar um peso fora da faixa permitida na balança



Fonte: Autoria própria, 2021.

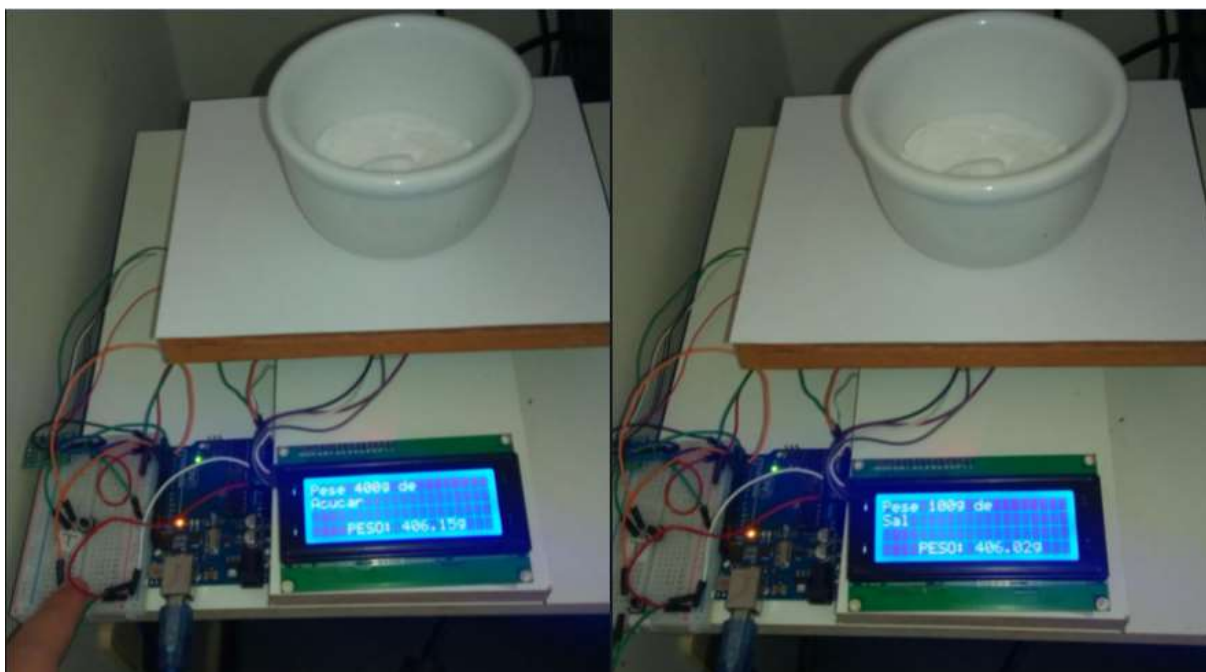
Agora, caso o peso colocado na balança esteja dentro da faixa permitida e o usuário pressionar o botão “1”, é efetivada a pesagem, o Arduino envia ao Excel os dados do peso, ingrediente, data, tempo e o peso que é especificado na receita, conforme apresentado na Figura 30, e é solicitado o próximo ingrediente conforme Figura 31.

Figura 30: Dados gravados no Excel após apertar o botão “1” do protótipo

	A	B	C	D	E
1	Ingrediente	Peso(g) receita	Valor pesado (g)	Data	Horario
2	Acucar	400	406	10-Nov-21	10:20:27 AM

Fonte: Autoria própria (2021).

Figura 31: Protótipo mostrando o próximo ingrediente a ser pesado após o ingrediente anterior ter sido gravado no Excel.



Fonte: Autoria própria (2021).

As informações da receita gravadas no Excel permitem ao gestor um melhor controle no processo, monitorando variáveis de peso e permitindo agir rapidamente caso haja alguma inconsistência nos dados recebidos.

5.4 ORÇAMENTO DO PROTÓTIPO CONSTRUÍDO

Antes de realizar a compra dos materiais, uma pesquisa de preços foi realizada em diferentes sites e lojas de produtos do gênero, com o propósito de encontrar o menor preço e oferecer um custo menor para o desenvolvimento e construção do projeto. Todos os materiais foram comprados em lojas brasileiras e o frete está incluso no valor total.

Da mesma maneira, para reduzir custos, a compra de alguns itens pode ser feita através da importação de outros países, dispensando a compra em mercado nacional, o que pode baratear o custo do projeto.

Na Tabela 6 são apresentados os materiais utilizados para a montagem do protótipo, além de seus custos de compra no comércio local.

Tabela 6: Orçamento do protótipo

Material	Quantidade	Preço unitário	Preço total
Arduino Uno R3 com cabo USB	R\$ 1,00	R\$ 35,60	R\$ 35,60
Protoboard 400 pontos	R\$ 1,00	R\$ 13,90	R\$ 13,90
Jumper Macho	R\$ 10,00	R\$ 0,86	R\$ 8,60
Jumper Femea	R\$ 8,00	R\$ 0,86	R\$ 6,88
Display LCD com módulo I2C	R\$ 1,00	R\$ 23,99	R\$ 23,99
Célula de carga 20kg + módulo HX711	R\$ 1,00	R\$ 25,00	R\$ 25,00
Tábuas de madeira	R\$ 2,00	R\$ 5,00	R\$ 10,00
Push Button	R\$ 2,00	R\$ 0,18	R\$ 0,36
Valor Total			R\$ 124,33

Fonte: Autoria própria (2021).

O valor total do protótipo foi de R\$ 124,33. Orçamentos de dosadoras automatizadas foram realizados em empresas locais e nacionais, valores esses na faixa de R\$ 15.000,00 – R\$ 60.000,00, valores muito superiores ao do protótipo construído.

É necessário levar em conta o tempo de montagem do protótipo. Visto que o código fonte do projeto já está construído, se faz necessário apenas a montagem da estrutura física e as conexões do Arduino para realizar uma réplica, dessa forma, em um dia pode-se construir o protótipo apresentado.

6. CONCLUSÕES

É possível perceber que a plataforma Arduino pode ser uma boa solução para desenvolver sistemas de automação. O projeto desenvolvido durante esse trabalho consegue atender os requisitos básicos de um sistema de automação de pesagem e dar início a automação do processo em indústria de formulados em pó.

Com um baixo orçamento (R\$ 124,33) é possível implementar um nível a mais de automação no processo, deixando a etapa de pesagem de forma mais controlada. O recebimento de dados pelo gestor além de facilitar o seu trabalho coletando os dados de produção de maneira mais precisa, auxilia na intervenção do processo, visto

que caso haja diferença na matéria-prima pesada e a da receita, comprometerá a qualidade do produto final, gerando despesas para a empresa.

O protótipo em questão não foi testado nas linhas de produção da Hoshino Rizzo, visto que o autor já não havia mais contato com a empresa quando o projeto foi construído.

O projeto pode seguir como referência para um projeto de automação em indústria formulados em pó e em outros tipos de indústria, segmentos comerciais e residenciais.

Como sugestão de melhoria do projeto, pode-se utilizar o módulo Wi-fi do Arduino, visto que com seu uso, não seria necessário a passagem de um cabo Ethernet para se conectar ao Arduino. Além disso, a melhoria na interface de *Hardware* do protótipo, como adição de botões maiores e uma interface mais intuitiva, facilitando o trabalho do colaborador com a balança.

Portanto, o protótipo apresentado neste trabalho se mostrou funcional, com potencial para a melhoria do processo produtivo e dando o início a automação da empresa.

REFERÊNCIAS

ALBUQUERQUE PEDRO; THOMAZINI DANIEL. **Sensores Industriais – Fundamentos e aplicações**. 8. ed.

AVIA. **HX711**. Disponível em: <<http://en.aviaic.com/detail/730856.html>>. Acesso em: 24 out. 2021.

BERALDO SILVEIRA SILVA, G.; BUENO DE PAULA, G.; FERREIRA DE PAULA SILVA, S. **Automação industrial, seus protocolos de automação industrial utilizando CLP, sistema supervisório e seus protocolos de comunicação**. [s.l: s.n.].

CANTON FERNANDO. **Evite erros na dosagem e pesagem com automação industrial**. Disponível em: <<https://www.solucoesdynamicair.com.br/blog/automacao-no-processo-de-pesagem-industrial>>. Acesso em: 24 out. 2021.

CITYSYSTEMS. **Sistemas supervisórios de automação industrial**. Disponível em: <<https://www.citisystems.com.br/sistemas-supervisorios-automacao-industria/>>. Acesso em: 24 out. 2021.

ESTEPA CASTILLO. **Proposta de Arquitetura de Supervisão e Controle para uma Plataforma Automatizada (WebLar) Orientada a Formação e Pesquisa em Automação e Robótica**. Campinas: [s.n.]. Disponível em: <<http://www.repositorio.unicamp.br/handle/REPOSIP/265314>>. Acesso em: 24 out. 2021.

FAGUNDES PATRICIA. **O que é um sistema SCADA?** Disponível em: <<https://www.hitecnologia.com.br/blog/o-que-e-um-sistema-scada/>>. Acesso em: 24 out. 2021.

FEHRENBACH, R. B. **DESENVOLVIMENTO DE SISTEMA AUTOMATIZADO PARA CONTROLE DE DOSAGEM DE RAÇÃO ANIMAL**.

FRADEN JACOB. **Handbook of modern sensors: Physics, designs, and applications.**

GALO FEBRÔNIO ALVES, R.; ROBERTO MANFIO, E. **DOSAGEM PROGRAMADA DE MATÉRIA-PRIMA PARA OS DIVERSOS PROCESSOS DA INDÚSTRIA.**

MARIA, A.; FERNANDES, R. **Automação e Controle em uma Micro Cervejaria Artesanal.**

MELCONIAN SÉRGIO. **MELHORIA DA AUTOMAÇÃO NO PROCESSO DE DOSAGEM INDUSTRIAL COMO FERRAMENTA PARA O AUMENTO DA PRODUTIVIDADE.**

NARVAES PATRÍCIA. **Dicionário ilustrado de meio ambiente.** São Caetano do Sul, 2021.

NASCIMENTO, L.; ROSA, A.; NANTES, E. **A IMPORTÂNCIA DO CONTROLE DE QUALIDADE EM INDÚSTRIA DO SEGMENTO ALIMENTÍCIO.**

PUPO MAURÍCIO. **Interface homem-máquina para supervisão de um CLP em controle de processos através da WWW.** São Carlos, 2020.

RIBEIRO CARLOS; DAMASIO EDILSON. **SOFTWARE LIVRE PARA BIBLIOTECAS, SUA IMPORTÂNCIA E UTILIZAÇÃO: O CASO GNUTECA.** Curitiba, 2016.

RONCOLI, A. M.; REGATTIERI, C. R. **O impacto de um sistema supervisório no gerenciamento da manutenção.** Revista Interface Tecnológica, v. 15, n. 2, p. 527–538, 30 dez. 2018.

ROSÁRIO JOÃO MAURÍCIO. **Automação industrial.** [s.l.] Baraúna, 2009.

THOKEIN ROGER. **Fundamentos de eletrônica digital.** [s.l.] Editora AMHG, 2013.

ARDUINOECIA. **Como usar o módulo I2C com Arduino e display LCD 16x2.** <<https://www.arduinoecia.com.br/modulo-i2c-display-16x2-arduino/>> Acesso em: 9 de outubro de 2021.

STRAUB, M. **Arduino IDE – O Software para gravação de códigos no Arduino.** <<https://www.usinainfo.com.br/blog/arduino-ide-o-software-para-gravacao-de-codigos-no-arduino/>> Acesso em: 9 de outubro de 2021.

SILVER, L. **Push Button Arduino: 3 modos de ligar botões no Arduino.** <<https://guiarobotica.com/push-button-arduino/>> Acesso em: 9 de outubro de 2021.

Brincando com Ideias. **O Arduino funciona como uma balança? Sim! Ele é capaz!** - **Módulo HX711.** Youtube, 16 de julho de 2019. Disponível em: <<https://www.youtube.com/watch?v=-qLfybfvsHw>>. Acesso em: 1 de maio de 2021.

CAMPOS, R. A. F. **Automação Residencial Utilizando Arduino e Aplicação Web.** Brasília, junho, 2014. Disponível em <http://repositorio.uniceub.br/bitstream/235/5461/1/Monografia_Roberto.pdf>

GIMENES, A. H.; PEREIRA, P. R. A. **Utilização da plataforma arduino como ferramenta de controle para um sistema de nível de líquidos.** 2015. Disponível em: <<http://pdf.blucher.com.br.s3-sa-east1.amazonaws.com/chemicalengineeringproceedings/cobeqic2015/472-34099-262068.pdf>>. Acesso em: 10 de novembro 2021.

KELER, A. C. et al. **Desenvolvimento de um equipamento de proteção coletiva para identificação de vazamento de gás amônia em ambientes refrigerados.** 2018. Disponível em: <http://www.abepro.org.br/biblioteca/TN_STO_261_500_36491.pdf> Acesso em: 10 de novembro de 2021.

MUYNARSK, O. G. **Sistema de monitoramento e controle de máquinas elétricas, utilizando microcontrolador arduino e supervisor Elipse scada para diminuição de parada não programadas para a manutenção.** 2014. Disponível em:

<<https://ojs.eniac.com.br/index.php/Anais/article/view/191/220>>. Acesso em: 10 de novembro de 2021.

JUNIOR, V. et al. **O uso do microcontrolador (Arduino) para automação industrial na medição de temperatura.** 2019. Disponível em: <<http://www.jornacitec.fatecbt.edu.br/index.php/VIIIJTC/VIIIJTC/paper/viewFile/1850/2468>>. Acesso em: 10 de novembro de 2021.

RENNA, R. et al. **Introdução ao kit desenvolvimento Arduino.** 2021. Disponível em: <http://www.telecom.uff.br/pet/petws/downloads/tutoriais/arduino/Tut_Arduino.pdf> Acesso em: 10 de novembro de 2021.

RAJA, D. **Overview of PLC and Various Arduino based PLC boards.** 2020. Disponível em: <<https://circuitdigest.com/tutorial/an-overview-of-plc-and-various-arduino-plc-boards>> Acesso em: 10 de novembro de 2021.

MARTINS, E. **Como fazer o controle de qualidade de alimentos.** 2021. Disponível em: <<https://blog-pt.checklistfacil.com/controle-de-qualidade-de-alimentos/>>. Acesso em: 03 de dezembro de 2021.

ANEXO A – CÓDIGO DE CALIBRAÇÃO UTILIZANDO A BIBLIOTECA HX711

```
/*
  Q0481-Sketch-Calibrar
  AUTOR: BrincandoComIdeias
  LINK:  https://www.youtube.com/brincandocomideias ; https://cursodearduino.net/
  COMPRE: https://www.arducore.com.br/
  SKETCH: Calibrar Balanca com módulo HX711
  DATA: 04/07/2019
*/

// INCLUSÃO DE BIBLIOTECAS
#include <HX711.h>

// DEFINIÇÕES DE PINOS
#define pinDT 2
#define pinSCK 3
#define pinBotao 4

// INSTANCIANDO OBJETOS
HX711 scale;

// DECLARAÇÃO DE VARIÁVEIS
float medida = 0;

void setup() {
  Serial.begin(57600);

  scale.begin(pinDT, pinSCK); // CONFIGURANDO OS PINOS DA BALANÇA
  scale.set_scale(); // LIMPANDO O VALOR DA ESCALA

  delay(2000);
```

```
scale.tare(); // ZERANDO A BALANÇA PARA DESCONSIDERAR A MASSA DA  
ESTRUTURA
```

```
Serial.println("Balança Zerada");  
}
```

```
void loop() {
```

```
    medida = scale.get_units(5); // SALVANDO NA VARIÁVEL O VALOR DA MÉDIA DE  
5 MEDIDAS
```

```
    Serial.println(medida, 3); // ENVIANDO PARA MONITOR SERIAL A MEDIDA COM  
3 CASAS DECIMAIS
```

```
    scale.power_down(); // DESLIGANDO O SENSOR  
    delay(1000); // AGUARDA 5 SEGUNDOS  
    scale.power_up(); // LIGANDO O SENSOR  
}
```

Fonte: Brincando com Ideias (2019).

APÊNDICE A – CÓDIGO FONTE DO ARDUINO IDE - PROJETO DE BALANÇA DE AUTOMAÇÃO CONSTRUÍDA COM ARDUINO

A.1 – INSTÂNCIA DE VARIÁVEIS

```
/*
Projeto TCC - Marco Vinicius F.A. Huf - 2021
*/
// INCLUSÃO DE BIBLIOTECAS
#include <HX711.h>
#include <PushButton.h>
#include <rExcel.h>
#include <LCD_I2C.h>

LCD_I2C lcd(0x27);
// DEFINIÇÕES DE PINOS
#define pinDT 2
#define pinSCK 3
#define pinBotao 4
#define pinBotaoTara 5

// DEFINIÇÕES
#define pesoMin 0.010
#define pesoMax 30.0

#define escala 104500

// INSTANCIANDO OBJETOS
HX711 scale;
PushButton botao(pinBotao);
PushButton botaoTara(pinBotaoTara);

rExcel myExcel;          // Classe utilizada para realizar a troca de informações com
o Excel
```

```
// DECLARAÇÃO DE VARIÁVEIS
float medida = 0;
int idx = 0;          // index

//Variáveis que serão trocadas com o Excel
char ingrediente[32];
char pesoDoIngrediente[32];
char receitaSelecionada[32];
char receitaSelecionadaAntiga[32];
char planilhaPesada[32];
//
int indexReceitas = 2;
float desvioPadrao = 0.2;
```

A.2 – SETUP INICIAL – EXECUTADO AO ARDUINO SER LIGADO

```
void setup() {
  lcd.begin();
  lcd.backlight();
  Serial.begin(57600);
  scale.begin(pinDT, pinSCK); //Configurando os pinos da balança
  scale.set_scale(escala); // Enviando o valor de calibração a ser utilizado
  myExcel.clearInput();
  scale.tare();
}
```

A.3 – LOOP ARDUINO

```
void loop() {

  //Verifica qual a receita que está sendo pesada
  myExcel.get("Medidas", "B8", receitaSelecionada);
  myExcel.get("Medidas", "C8", receitaSelecionadaAntiga);
```

```

myExcel.get("Medidas", "D8", planilhaPesada);

//Primeiro, irá verificar se a receita a ser pesada foi alterada
resetaBalancaSeMudouReceita();

//Verifica qual o ingrediente atual e o seu peso
myExcel.getIndex(receitaSelecionada, indexReceitas, 1, ingrediente);
myExcel.getIndex(receitaSelecionada, indexReceitas, 2, pesoDoIngrediente);
int pesoIngredienteInt = atoi(pesoDoIngrediente);

//Printa as informações na tela
printaInformacoesDePeso(pesoIngredienteInt);

//Inicia a leitura dos botões de tara e envio de informações
botaoTara.button_loop();
botao.button_loop();

if (botaoTara.pressed()){
    tararBalanca();
}

if(botao.pressed()){

    medida = scale.get_units(5); //Salva na variável medida a média de 5 medidas de
peso.

    if (medida <= pesoMin ){ // Caso a medida esteja negativa, tarar a balança
        tararBalanca();
    }
    else {
        float medidaCorrigida = medida*1000;

        if (medidaCorrigida < (pesoIngredienteInt - pesoIngredienteInt*desvioPadrao) ||
medidaCorrigida > (pesoIngredienteInt + pesoIngredienteInt*desvioPadrao)){

```

```

        exibirErroForaDaFaixaDePeso();
    }
    else{
        escreverNoExcelValorPesado(medidaCorrigida);
    }
}
}
}
}

```

A.4 – FUNÇÃO QUE DETECTA SE O USUÁRIO ALTEROU A RECEITA A SER PRODUZIDA

```

void resetaBalancaSeMudouReceita(){
    if (receitaSelecionadaAntiga[0] == 't'){
        lcd.print("Receita alterada");
        scale.tare();
        myExcel.clearInput();
        indexReceitas = 2;
        idx = 0;
        myExcel.write("Medidas", "C8", "f");
    }
}
}

```

A.5 – FUNÇÃO QUE MOSTRA INFORMAÇÕES DO INGREDIENTE A SER PESADO E O PESO ATUAL DA BALANÇA.

```

void printInformacoesDePeso(int pesoDoIngrediente){
    lcd.setCursor(0, 0);
    lcd.print("Pese ");
    lcd.print(pesoDoIngrediente);
    lcd.print("g de ");
    lcd.setCursor(0, 1);
    lcd.print(ingrediente);
}

```

```

lcd.setCursor(20, 1);
lcd.print("  PESO: ");
lcd.print(scale.get_units(5)*1000);
lcd.print("g");

delay(500);
lcd.clear();
}

```

A.6 – FUNÇÃO UTILIZADA PARA TARAR A BALANÇA

```

void tararBalanca(){
    scale.tare();
    medida = 0;
    lcd.clear();
    lcd.print("Tara realizada");
}

```

A.7 – FUNÇÃO UTILIZADA PARA EXIBIR UM ERRO NO DISPLAY CASO O VALOR PESADO NÃO ESTEJA DENTRO DO SOLICITADO

```

void exibirErroForaDaFaixaDePeso(){
    lcd.clear();
    lcd.print("O peso esta fora da faixa permitida");
    delay(2500);
    lcd.clear();
}

```

A.8 – FUNÇÃO UTILIZADA PARA ESCREVER NO EXCEL O INGREDIENTE, VALOR PESADO, VALOR SOLICITADO, DATA E HORA DO PROCESSO

```

void escreverNoExcelValorPesado(int valorMedido){
    myExcel.writeIndexed(planilhaPesada, idx+2, 1, ingrediente);
    myExcel.writeIndexed(planilhaPesada, idx+2, 2, pesoDoIngrediente);
    myExcel.writeIndexed(planilhaPesada, idx+2, 3, valorMedido, 2);
}

```

```
myExcel.writeIndexed(planilhaPesada, idx+2, 4, "%date%");  
myExcel.writeIndexed(planilhaPesada, idx+2, 5, "%time%");  
  
idx++;  
indexReceitas++;  
}
```


APÊNDICE B – CÓDIGO FONTE DESENVOLVIDO NO PYTHON - PROJETO DE BALANÇA DE AUTOMAÇÃO CONSTRUÍDA COM ARDUINO

```

#!/usr/bin/python3
import xlwings as xw
import PySimpleGUI as sg
#Biblioteca para leitura do excel
import xlrd
import datetime

class TelaPython:
    def __init__(self):
        book_ler = xlrd.open_workbook("Arduino_Excel_30.xls")
        arrayReceitas = book_ler.sheet_names()
        arrayReceitas.remove('Medidas')
        arrayReceitas.remove('Parameters')

        arrayFiltrado = [ x for x in arrayReceitas if "-" not in x ]
        self.telaSelecionarReceita(arrayFiltrado);

    def telaSelecionarReceita(self, listaReceitas):
        layout = [[sg.Text('Selecione a receita desejada')],
                 [sg.Combo(listaReceitas, key='combobox')],
                 [sg.Text('Peso: ', size=(5, 1)), sg.InputText(key='peso',size=(10,3))],
                 [sg.Button('Confirmar')]]

        janela = sg.Window("Pesagem de receitas").layout(layout)

        while True:          # Event Loop
            event, values = janela.read()
            print (event)
            if event in (None, 'Exit'):
                break
            if callable(event):
                event()

```

```

elif event == 'Confirmar':
    selecionada = values['combobox']
    peso = values['peso']

    nomeNovaPlanilha = datetime.datetime.now().strftime("%m-%d-%H;%M")
    self.alterarReceitaSelecionada(selecionada, peso, nomeNovaPlanilha)
def alterarReceitaSelecionada(self, selected, pesoNecessario, nomeNovaPlanilha):

    wb = xw.Book('Arduino_Excel_30.xls')
    wb.sheets.add(nomeNovaPlanilha)
    sht1 = wb.sheets[nomeNovaPlanilha]
    sht1.range('A1').value = 'Ingrediente'
    sht1.range('B1').value = 'Peso(g) receita'
    sht1.range('C1').value = 'Valor pesado (g)'
    sht1.range('D1').value = 'Data'
    sht1.range('E1').value = 'Horario'
    receitaSelecionada = selected
    sht2 = wb.sheets[selected]
    sht2.range('E2').value = pesoNecessario

    sht3 = wb.sheets['Medidas']
    sht3.range('B8').value = selected
    sht3.range('C8').value = 't'
    sht3.range('D8').value = nomeNovaPlanilha

tela = TelaPython()

```