

**IOT-BASED CONTROL AND MONITORING SYSTEM OF A SOLAR-POWERED
BRUSHLESS DC MOTOR FOR AGRO-MACHINES – THE CASE OF A TANZANIAN-
MADE OIL PRESS MACHINE**

Gilbert Minja

**A Dissertation Submitted in Partial Fulfilment of the Requirements for the Degree of Master of
Science in Embedded and Mobile Systems of the Nelson Mandela African Institution of Science
and Technology**

Arusha, Tanzania

July, 2022

ABSTRACT

The impulse in designing local agricultural machinery for curbing post-harvest losses in most African countries particularly Tanzania is unmatched. Locally made agricultural machines have proven to elevate the life of many small-scale farmers, which has increased the need to incorporate machine drives and controls to ease the process and operations. With potentials in Solar Energy, powering machine drive systems that operate in off-grid areas has been the best solution. Using the principles of Internet of Things (IoT) together with advancement in motor designs and readily available off the shelf microcontrollers such as the Raspberry Pi and Arduino UNO in the market, we achieve machinery that caters for our needs and the local content. Mobile apps play a huge role in industrialization where monitoring and even controls of machines can be performed by the mobile phones. This project incorporated Agile-Scrum methods to develop a control and monitoring system for a locally made avocado oil extraction machine that is powered by a solar system with 1600W panel arrays and 800Ah battery pack, and uses a Brushless Direct Current Motor coupled with electric solenoid valve, relay modules and a controller unit assisting on the control process and collecting crucial motor operation data such as voltage and current. The designed Mobile app ‘Blue’ acquire motor operation data from the Raspberry Pi via Bluetooth technology, delivering data to cloud server for later analysis. Easing data acquisition in off grid areas when engineers, technicians or operators have a physical access to the stations. It was concluded that this novel design would provide an effective control and monitoring mechanism with an acceptance on reliability, usability and effectiveness of up to 85.65% for a plethora of locally-made machinery that available in the market which still uses the manual means of operation emphasizing ease of use and productivity, thence joining hands with the global world on attaining some of the Sustainable Development Goals.

Keywords: IoT, BLDC Motors, Solar Energy, Solar Photo Voltaic, Batteries, Off-grid, Bluetooth Technology, Mobile Application, Control Systems, Sustainable Development Goals

DECLARATION

I, Gilbert Minja, do hereby declare to the Senate of the Nelson Mandela African Institution of Science and Technology that this dissertation is my original work and that it has neither been submitted nor concurrently submitted for a degree or similar award in any other institution.

Gilbert Minja



29/07/2022

Name and Signature of Candidate

Date

The above declaration is confirmed by:

Dr. Jema David Ndibwile



29/07/2022

Name and Signature of Supervisor 1

Date

Prof. Shubi Felix Kaijage



29/07/2022

Name and Signature of Supervisor 2

Date

COPYRIGHT

This dissertation is copyright material protected under the Berne Convention, the Copyright Act of 1999 and other international and national enactments, on that behalf, on intellectual property. It must not be reproduced by any means, in full or in part, except for short extracts in fair dealing; for researcher's private study, critical scholarly review or discourse with an acknowledgement, without a written permission of the Deputy Vice Chancellor for Academic, Research and Innovation, on behalf of both the author and the Nelson Mandela African Institution of Science and Technology (NM-AIST).

CERTIFICATION

The undersigned certify that they have read and hereby recommend for acceptance by the Nelson Mandela African Institution of Science and Technology, a dissertation entitled “*IOT-Based Control and Monitoring System of a Solar-Powered Brushless Dc Motor for Agro-Machines: The Case of a Tanzanian-Made Oil Press Machine*” in Partial Fulfilment of the Requirements for the Award of the Degree of Master of Science in Embedded and Mobile Systems (EMoS) of the Nelson Mandela African Institution of Science and Technology.

Dr. Jema David Ndibwile

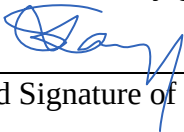


29/07/2022

Name and Signature of Supervisor 1

Date

Prof. Shubi Felix Kaijage



29/07/2022

Name and Signature of Supervisor 2

Date

ACKNOWLEDGEMENTS

First and foremost, I want to express my gratitude to the Almighty God for leading me to the Nelson Mandela African Institution of Science and Technology to pursue a Master's degree (NM-AIST). I'd also want to thank God for blessing me with courage, strength, and wellness during my studies, which led to this accomplishment.

I'd like to convey my heartfelt appreciation to the: Center of Excellence for ICT in East Africa for their financial help in sponsoring my studies and project work, Dr. Jema Ndibwile, Dr. Shubi Kaijage, and Mr. Joseph Bundala, my industrial supervisor, for their encouragement, advice, and unwavering support during my internship and project.

The Access to Energy Institute (A2EI) and Imara Tech Company in Arusha deserve special recognition for providing me with an internship opportunity as well as continued assistance and encouragement throughout the project.

My parents' and family's love, prayers, moral support, and encouragement have been invaluable throughout my studies.

I'd also want to express my gratitude to my comrades, instructors, and friends for their support and encouragement during my time at NM-AIST. They were pleasant, cooperative, and sympathetic during my studies and stay at NMA-IST, and they played an important role.

DEDICATION

This dissertation is dedicated to my lovely wife Jacqueline, our two children Talia-Kristen and Gian-Kenzo, and my parents and family, who have always believed in my skills.

TABLE OF CONTENTS

ABSTRACT	i
DECLARATION	ii
COPYRIGHT	iii
CERTIFICATION.....	iv
ACKNOWLEDGEMENTS	v
DEDICATION	vi
TABLE OF CONTENTS	vii
LIST OF TABLES	xi
LIST OF FIGURES.....	xii
LIST OF APPENDICES	xiv
LIST OF ABBREVIATIONS AND SYMBOLS	xv
CHAPTER ONE	1
INTRODUCTION.....	1
1.1 Background of the Problem.....	1
1.2 Statement of the Problem	2
1.3 Rationale of the Project	3
1.4 Project Objectives.....	3
1.4.1 Main Objectives	3
1.4.2 Specific Objectives.....	3
1.5 Project Questions.....	3
1.6 Significance of the Project	4
1.7 Delineation of the Project.....	4
CHAPTER TWO.....	5
LITERATURE REVIEW.....	5
2.1 Introduction	5
2.2 Brush-Less Direct Current Motor.....	5

2.3	Machine Control Systems	5
2.3.1	Open-loop and Closed-loop Control	6
2.4	Machine Monitoring Systems	6
2.5	Importance of Automation Systems	6
2.6	Importance of Monitoring Systems	7
2.7	Internet of Things	7
2.7.1	Internet of Things Architecture	7
2.8	Communication Technologies in Internet of Things.....	8
2.8.1	LoRa	9
2.8.2	ZigBee Protocol.....	10
2.8.3	Bluetooth	10
2.9	Solar Photovoltaic Power	12
2.10	Related Works	12
CHAPTER THREE.....		14
MATERIALS AND METHODS		14
3.1	Introduction	14
3.2	Project Design Approach.....	14
3.3	Relevance and Rigor Cycle	15
3.4	Project Area.....	15
3.5	Requirement Data Gathering.....	16
3.6	System Modelling.....	17
3.7	System Implementation.....	18
3.7.1	Components Selection Method	18
3.7.2	Hardware Components.....	19
3.8	System Design Architecture	19
3.8.1	Power Supply Unit	19
3.8.2	Control Unit	21

3.8.3	Transmission Unit	23
3.8.4	Data Acquisition Unit.....	28
3.9	Advantages of the Developed System.....	33
3.9.1	Technical Feasibility	33
3.9.2	Availability of Monitoring Data from Controller	33
3.9.3	Use of micro-controllers (Arduino and RPi).....	33
3.9.4	Future Development.....	33
3.10	System Testing and Validation.....	33
3.10.1	Test Requirements.....	33
CHAPTER FOUR.....		35
RESULTS AND DISCUSSION		35
4.1	Results	35
4.1.1	System Implementation Overview.....	35
4.1.2	Control System Unit.....	35
4.1.3	The Data Acquisition Mobile Application	38
4.1.4	The Power Unit	43
4.1.5	Met Functional and Non-functional Requirements.....	45
4.1.6	System Modelling	46
4.1.7	Scalability.....	46
4.1.8	Validation.....	46
4.2	Discussion	49
CHAPTER FIVE.....		51
CONCLUSION AND RECOMMENDATIONS.....		51
5.1	Conclusion.....	51
5.2	Recommendations	52
REFERENCES.....		53
APPENDICES.....		60

POSTER PRESENTATION	74
---------------------------	----

LIST OF TABLES

Table 1:	Functions of the Core Bluetooth Protocols	12
Table 2:	Requirements gathered from A2EI	16
Table 3:	Advantages and drawbacks of Python	25
Table 4:	Raspberry Pi and Arduino UNO comparison with other Single Board Computers (SBCs)	26
Table 5:	Described use case requirements	30
Table 6:	Total Power consumption by the system	43
Table 7:	The Functional Requirements	45
Table 8:	Non-Functional Requirements	46
Table 9:	Interpretation of the Likert Scale Mean	48
Table 10:	System Acceptance testing checklist.....	49
Table 11:	System acceptance testing survey questionnaire.....	68

LIST OF FIGURES

Figure 1:	The IoT Architecture (Shahinzadeh <i>et al.</i> , 2019).....	8
Figure 2:	LoRa Protocol Stack (Loraalliance, 2020).....	9
Figure 3:	Zigbee Protocol Stack (Prakash, 2020).....	10
Figure 4:	Bluetooth Protocol Stack (Zeadally <i>et al.</i> , 2019).....	11
Figure 5:	Agile- scrum Methodology (a) and System Design (ASD) with a focus on Outcomes (Van Casteren, 2017)	15
Figure 6:	Tanzania map showing Meru District in Arusha Region (Mbabaye <i>et al.</i> , 2018)	15
Figure 7:	Flow of components/modules selection Method (Van Casteren, 2017)	18
Figure 8:	The Solar panel (a) the MPPT charge controller (b) and the solar battery (c) (Robles Algarín <i>et al.</i> , 2017)	20
Figure 9:	The LM2596 Buck DC-DC Converter (Ferrer <i>et al.</i> , 2017)	20
Figure 10:	Arduino UNO (a) A 4-channel 5-VDC Relay Module (b) (Arduino, 2021)	22
Figure 11:	The BLDC Motor Model KY130AS0430 (a) the KEYA Electric KYDBL48150-1E model controller (b) and the DC Solenoid valve (c) (Keya, 2021; Aliexpress, 2020)	23
Figure 12:	Raspberry Pi 3 Model B (Raspberrypi, 2021)	25
Figure 13:	The RS232 Port configuration (Erraissi <i>et al.</i> , 2018)	28
Figure 14:	The data logger system architecture	28
Figure 15:	Use Case diagram showing the administrator and users (Engineer/Technician).....	30
Figure 16:	Flutter application structure (Flutter, 2021).....	31
Figure 17:	Arduino IDE used to write the control codes.....	32
Figure 18:	The block diagram of the motor control and monitoring system.....	35
Figure 19:	The flow of events for the control system and monitoring data acquisition system.	36
Figure 20:	The complete Circuit diagram of a Power Supply, Control and Transmission with their interconnections	37
Figure 21:	The Developed Control Box circuit.....	38
Figure 22:	The Data flow of events for the ‘Blue’ mobile application.	39

Figure 23:	The Login and Registration page (a) and the Paired Control and Monitoring Station (b).....	40
Figure 24:	Snippets of the Blue app showing Uploaded Items (a), Internet connectivity status (b) and App exit query (c)	41
Figure 25:	The Firestore Database showing files uploaded from the mobile phone client	42
Figure 26:	The Power Supply Unit.....	43
Figure 27:	The final developed control and monitoring system with the oil-press machine	50

LIST OF APPENDICES

Appendix 1:	Python Code for collecting Data on USB Port from the RaspberryPi	60
Appendix 2:	Control codes for the Arduino UNO	64
Appendix 3:	System Development in Pictures.....	67
Appendix 4	Questionnaire for Validating the System	68
Appendix 5	Setting up VNC for Raspberry Pi wireless connectivity to PC.....	69
Appendix 6:	Blue Android Application Manifest.....	72

LIST OF ABBREVIATIONS AND SYMBOLS

A2EI	Access to Energy Institute
AC	Alternating Current
ASD	Agile System Development
BLDC	Brush-Less DC
BT	Bluetooth
BT	Bluetooth
DAU	Data Acquisition Unit
DC	Direct Current
DFD	Data Flow Diagram
ICT	Information and Communication Technologies
IDE	Integrated Developed Environment
IDE	Integrated Development Environment
iOS	iPhone Operating System
IoT	Internet of Things
ISM	Industrial, Scientific, and Medical Equipment
IT	Information Technology
JSON	JavaScript Object Notation
L2CAP	Logical Link Control and Adaptation Protocol
LMP	Link Manager Protocol
LoRa	Low-Power, Long-Range Wireless Protocol
LP-WAN	Low Power Wide Area Network
MAC	Media Access Control
MCU	Microcontroller Unit
MPPT	Maximum Power Point Tracking
NM-AIST	Nelson Mandela African Institution of Science and Technology
PC	Personal Computer
PID	Proportional Integral Derivative
PLC	Programmable Logic Control
PV	Photo-Voltaic
PV	Photovoltaic
PWM	Pulse Width Modulation
R&D	Research and Development

RPi	Raspberry Pi
SBC	Single Board Computers
SCADA	Supervisory Control and Data Acquisition
SDG	Sustainable Development Goals
SDK	Software Development Kit
SDP	Service Discovery Protocol
SMBs	Small and Medium Businesses
SQL	Structured Query Language
UART	Universal Asynchronous Receiver-Transmitter
UI	User Interface
Wi-Fi	Wireless Fidelity

CHAPTER ONE

INTRODUCTION

1.1 Background of the Problem

The Access to Energy Institute (A2EI) aspires to be the world's first philanthropic and collaborative R&D platform for the solar off-grid business. The A2EI, which was established in 2019, collaborates with entrepreneurs, innovators, and organizations to assist the creation and study of solar-powered machinery, agricultural technology for productive use, home appliances, and energy-related services (A2EI, 2019).

As part of A2EI's productive-use and agricultural mechanization, equipment, machinery and implements are used to maximize agricultural and food output. The A2EI is a pioneer in the use of machinery, equipment, and tools in day-to-day farm/end-product operations to boost marginal productivity in food production and poverty eradication, whilst working closely to achieve the three 2030 Sustainable Development Goals (SGDs), that is SGD1 -Poverty. The SGD7- Affordable Clean Energy and SDG 9 -Industrial, Innovation and Infrastructure (A2EI, 2019).

Recently, modern testing has played a major part in attempts to screen and control electrical and mechanical equipment drives. It is possible to observe, control and collect the dynamic condition of electrical and mechanical devices or machinery drives. To improve quality of operations, quick preparation, maintenance and low power usage are used (Hemalatha *et al.*, 2017). Several methods for monitoring and controlling current procedures, such as Zigbee, PLC-SCADA, and The Internet of Things (IoT) are accessible (Jagadesh *et al.*, 2019).

The Internet of Things (IoT) refers to the interconnection of physical devices, electrically powered machines, cars (also known as "connected devices" or "smart devices"), buildings, and other objects that are embedded with electronics, software, sensors, actuators, and network connections in order to collect and share data (Hemalatha *et al.*, 2017). The Internet of Things (IoT) allows things to be sensed or controlled remotely utilizing existing network infrastructure, allowing for more seamless integration of the real world into computer-based processes, resulting in increased efficiency, accuracy, and economic benefit, as well as reduced human interaction (Ganga & Ramachandran, 2018).

Our civilization need energy because it ensures our level of life and supports all other aspects of our economy. Renewable energy technologies provide clean, abundant energy from renewable resources such as the sun, wind, soil and plants (Mohtasham, 2015). Solar energy, as a major potential renewable energy source, is becoming one of the most essential energies in the future,

particularly in places where the sun is abundant, such as Tanzania, which has yearly averages of insolation ranging from 4.9 to 6.6 KWh/(m²day) (Hammar, 2011).

Photovoltaic (PV) solar panels have traditionally been used for smaller-scale energy generation, mainly in business or residential facilities or individual buildings. The efficiency of these panels ranges from 18% to 12%, and the entire cost of a solar panel is determined by its size (in W), brand, physical size, longevity/durability, and any certifications it may have (Islam & Meade, 2013).

Advanced electronics and software in sophisticated instrumentation, computers, and digital signal processors have simplified and increased our capacity to instrument and analyze machines, not least in the critical area of controlling and displaying the results of complicated conditions-monitoring analyses (Tavner *et al.*, 2008).

The project investigates how IoT technology may be used to control and monitor the activities of a solar-powered oil-press machine. To collect monitoring data from the oil-extraction station, a Raspberry Pi, Arduino microcontroller, and ultimately a cloud-based IOT system were used. The system focuses primarily on preset iterations for a whole oil pressing process, while also recording essential motor metrics such as voltage and current that may be used to determine motor performance, efficiency, and longevity. Through the use of a cloud-based computing system, the appropriate operation levels have been determined by the oil-press station technical personnel.

1.2 Statement of the Problem

The introduction of new technologies for improving techniques of machinery control, data gathering and analysis, plant efficiency, uneven outcomes, and a lack of resources are some of the practical problems encountered by productive-use personnel (Ganga & Ramachandran, 2018).

The locally made Agro-Machines that are currently deployed by A2EI as pilot test-case in off-grid areas that require renewable power sources particularly solar energy, rely solely on manual operations and technical knowhow of the operators. Notwithstanding, the crucial parameters monitoring of the core driving entities of the machines drives such as the motor are inadequate.

A need of more efficient and sustainable power source, modes of control/automation and monitoring of these machines in areas where there is inadequate or no communication is of a paramount.

1.3 Rationale of the Project

The purpose of the project is to look at the available control and monitoring parameters for the solar-powered oil-press machine. The aim is to come up with an affordable (cost-effective) system design that will cater the needs of machine operations based on the behavior according to the critical parameters and operator's mode of use required. We chose an oil press machine among other Agro-Machines available at A2EI as they have posed an immediate operational challenge on motor control, hydraulic jack system protection and motor parameters monitoring.

In comparison to existing monitoring devices in the market, the project will employ widely accessible components such as microcontrollers and sensors to construct the control and monitoring system, which will greatly reduce the cost and maintenance issues.

1.4 Project Objectives

1.4.1 Main Objectives

To develop a solar-powered control and operation data acquisition system for a Brushless DC Motor deployed in Agro-Machines, the Oil Press machine locally made in Tanzania.

1.4.2 Specific Objectives

- (i) To identify requirements for a solar-powered IoT based system for control and data acquisition for industrial machine drives.
- (ii) To develop a solar power unit, control and data acquisition mechanism for industrial machine drives (BLDC Motors).
- (iii) To validate the system developed.

1.5 Project Questions

- (i) What are the requirements for development of a solar-powered IoT-Based system for control and data acquisition for industrial machine drives?
- (ii) How should the proposed system be developed to meet the intended objectives?
- (iii) Does the developed system meet the intended goals?

1.6 Significance of the Project

- (i) A system is designed to assist small scale avocado farmers to cold-press avocado oil to curb post-harvest losses, providing them with extra income.
- (ii) The System will also address the use of renewable energies (in particularly Solar Energy) throughout its operations.
- (iii) The system will also collect motor operation data, which will then assist in the future optimization of power and creating simpler machine drives with high output.
- (iv) The System will help local engineers with the footprint in the modelling and design of control systems for machine drives for the small-scale industries.

1.7 Delineation of the Project

This project assumes that all local Oil Press machine operators have the ability to operate an electrically driven system once trained on. Equally, this project assumes that all engineers and technicians involved on a periodic maintenance of the machine are conversant with smart phone applications including apps navigation and initiating connectivity such as Bluetooth and other control functionalities, all upon a proper training.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

By analyzing recent relevant research, the purpose of this chapter is to offer an overview of the context in which this project will be carried out. As per the introduction on the previous chapter this project proposes the design and development of a control and monitoring system for Solar Powered BLDC Motors deployed in locally made Agro-Machines.

It is important to analyze significant literature linked with the control of BLDC Motors and Electric solenoid valves as well as BLDC Motor monitoring and data acquisition methods and identify gaps in order to lay the groundwork for the project in subject.

2.2 Brush-Less Direct Current Motor

Shunt Motors, Compound Motors, Permanent Magnet DC motors, Series Motors, and BLDC Motors are among the several types of DC motors readily available on the market. Different operational concepts have been established for diverse industrial applications for these DC motors (Vaishakh & Vanitha, 2019).

The BLDC motor has gained enormous acceptance over the last several decades in industrial automation equipment (Shifat & Jang-Wook, 2020). The BLDC motors have become more commonly used in industrial applications and instruments than typical DC motors due to attractive qualities such as high efficiency, great torque, incredible speed, low noise, compact area, and long life (Ibrahim *et al.*, 2019). Moreover, Ibrahim *et al.* (2019) iterated, because these motors don't have brushes like the normal DC motors, no sparks are produced during operation. This enables the use of the motor in high-risk situations.

Furthermore, Hussain *et al.* (2019) pointed out, since the BLDC motor is brushless, no brushes have to be replaced, hence requiring no maintenance. As a result, there is no need for the consumer to be concerned about its maintenance.

2.3 Machine Control Systems

Brushless DC Motor closed loop control is an important component of many industrial processes, and several control strategies have been developed to improve the variety of control executions of brushless DC motor drives such as speed, current, protection and so forth (Hameed, 2018).

Numerous adaptive control methods have been used to regulate BLDCs one being a combination of Fuzzy Logic and PIDs. However, these algorithms are mostly for linear models and are not relevant to nonlinear models (Somwanshi *et al.*, 2019). Fuzzy control is based on fuzzy logic and fuzzy set theory, thus, too complex to deploy in small scale systems (Anshory *et al.*, 2019). There is no self-tuning in the generic PID control. Only skilled technicians can identify it based on the controlled item's step response profile, and the parameters of a PID control object can be obtained via a variety of experimental artificial techniques which are deemed complex (Zhang *et al.*, 2018).

2.3.1 Open-loop and Closed-loop Control

The two types of control loops are closed loop (feedback) and open loop (no feedback). The controller's independence from the "process output" is a defining characteristic of open loop control. In closed loop control, the controller's control action is determined by the process output (Gonzalez & Asada, 2019). A feedback Control loop that tends to keep a specified relationship of variables from one process to another by comparing functions of these variables and applying the difference of control to produce the needed output can be used to ensure the integrity of a closed loop system (Schmidt *et al.*, 2018).

2.4 Machine Monitoring Systems

Monitoring motor characteristics is becoming increasingly popular in industries as a way to predict early motor failures/defects. The BLDC motors necessitate constant monitoring of the motor's functioning. The electric motor's performance may be maintained by monitoring changes in numerous parameters associated with the motor operation (Izza & Robandi, 2016).

The most frequent and essential metrics monitored for assessing the operation of an electric motor in industries are voltage, current, active power, reactive power, and temperature. For instance, Vaishakh and Vanitha (2019) demonstrated a Lab VIEW based design method for monitoring various motor characteristics. The significance of monitoring is highlighted since it is dangerous to manage and monitor these devices directly during continuous operation (Magdum & Agashe, 2016).

2.5 Importance of Automation Systems

Automation is described as a sequence of processes or procedures carried out without the assistance of humans. There are several control systems that employ automation to control multiple operations without the need for human intervention (Forster *et al.*, 2019).

Increased production rates and productivity, as well as more effective resource consumption, higher product quality, enhanced safety, shorter worker workweeks, and shorter manufacturing lead times,

are all frequently ascribed to automation (Bello & Steiner, 2019). Increased production and productivity are two of the most important reasons for introducing automation. Despite promises of greater quality from humans, automated systems usually conclude the production process with less variability than human labor, resulting in better product control and consistency. Increased process control also allows for more effective resource usage, which results in less waste (Dotoli *et al.*, 2019).

2.6 Importance of Monitoring Systems

The necessity of monitoring the condition of engineering machine drives has grown as more engineering operations are automated and the personnel required to run and oversee is lessened (Jagadesh *et al.*, 2019).

Rotating electrical drive machines, whether driven by BLDC motors or induction motors, are at the core of most engineering processes, and as they are developed to tighter margins, there is a rising need to monitor their behavior and performance for reliability, efficiency and sustainability's sake (Izza & Robandi, 2016).

2.7 Internet of Things

The Internet of Things (IoT) is the collection and sharing of data by combining physical equipment, machines, buildings, and other things with electronics, software, sensors, actuators and network connections (Jagadesh *et al.*, 2019). The Internet of Things (IoT) is widely used to expand device, system, and service connectivity that extends beyond machine-to-machine (M2M) interactions and encompasses a wide range of protocols, domains and applications (Senthilkumar *et al.*, 2020).

The Internet of Things has facilitated the growth of Internet-connected automation into a slew of new domains. The IoT is also predicted to create significant volumes of data from a variety of sources, demanding faster data aggregation as well as a greater requirement to index, store and analyze such data more efficiently (Ganga & Ramachandran, 2018).

2.7.1 Internet of Things Architecture

The basic IoT architecture consists of three levels (Fig. 1). These levels are the perception layer, the network layer, and the application layer.

(i) Perception Layer

Sensor nodes acquire data about the environment in this level (e.g., machines). Voltage sensors to check voltage levels in order to ensure the proper operation of an electrical machine. Current sensors

to measure the current used by a load in an electric drive/machine, DHT sensors detect humidity and temperature, PH sensors detect soil pH, and PIR sensors detect movement/motion of an item and the like (Alihamidi *et al.*, 2019). Normally, this layer detects other smart things or normal objects equipped with sensors or senses crucial parameters during the machine drives operations.

(ii) Network Layer

This layer is responsible for receiving, processing, and delivering sensed data to various network components. This layer consist of gateways, routers and micro-controller devices like Arduino, and Raspberry pi, and communication protocols (Malche *et al.*, 2019). Communication protocols like Hyper Text Transfer Protocol (HTTP), Bluetooth, MQ Telemetry Transport (MQTT), Zigbee and Wi-Fi provide the capability of the Network layer devices to communicate with Perception Layer as well as Application Layer (Shahinzadeh *et al.*, 2019).

(iii) Application Layer

All collected data is managed, processed, analyzed, and presented in a user-friendly manner in this layer. Furthermore, the application layer communicates with the sensor layer to change ambient conditions utilizing actuators such as valves and relay switches. This layer is made up of Webservers, Databases and Clients (Shahinzadeh *et al.*, 2019).

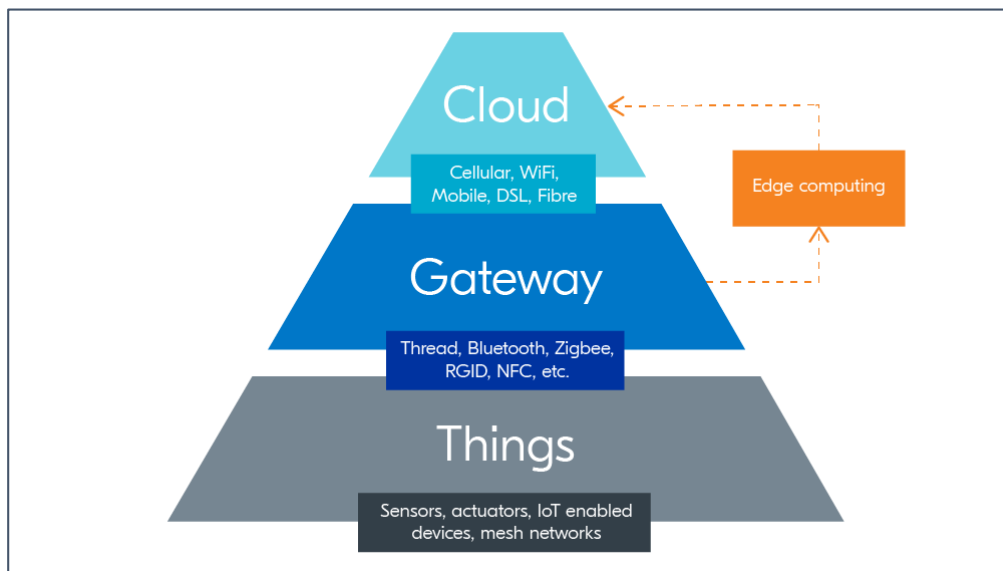


Figure 1: The IoT Architecture (Shahinzadeh *et al.*, 2019)

2.8 Communication Technologies in Internet of Things

Wireless technologies enable us to send information across multiple locations that are not wired together. It is also dependent on the distances between locations that we wish to send data.

For machine monitoring systems, communication technologies such as Zigbee, Bluetooth, Near Field Communication (NFC), Long Range Radio (Lo-Ra), Infrared (IR), Wireless Fidelity (Wi-Fi), or even the visible light spectrum are often used (known as VLC). In this work, the Raspberry Pi3 is utilized to monitor machine operation parameters then by using Bluetooth-enabled hand-held devices to upload the monitored patch files to the cloud server for more analysis. Bluetooth was chosen since it is the most widely used communication technology in commercial cellphones today. As a result, the technology may be used without the need for an extra module on either the Raspberry Pi or the smartphone.

Before diving into the technology of our choice for this project (i.e., Bluetooth), we looked at alternative wireless communication technologies that were available.

2.8.1 LoRa

Semtech created the LoRa communication technology, which is a low-power, long-range wireless protocol. This technology has a variety of significant benefits, including low power consumption, great extensibility, and high efficiency, however, it only has a small transmission rate (Bertoldo *et al.*, 2018).

The LoRa Alliance developed a Low Power Wide Area Network (LP-WAN) protocol and system architecture, with LoRa specifying the system's physical layer (Loraalliance, 2020). Furthermore, LoRaWAN is a media access control (MAC) layer protocol developed on top of the LoRa physical layer. It describes the network architecture that runs in an unlicensed spectrum below 1000 MHz (San *et al.*, 2017). The LoRaWAN employs AES-128 encryption for security, which is represented by the AppsKEY and NwkKey (Tsai *et al.*, 2019).

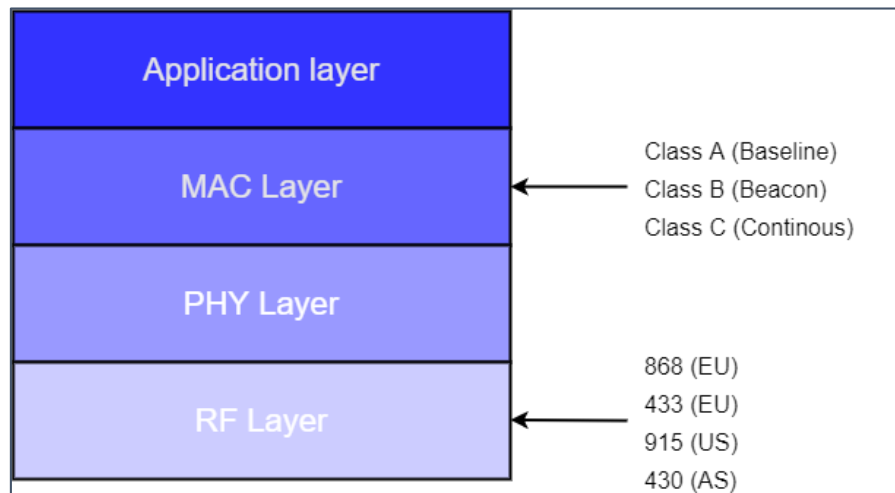


Figure 2: LoRa Protocol Stack (Loraalliance, 2020)

2.8.2 ZigBee Protocol

The Zigbee wireless communication system is a low-power, low data-rate wireless communication technology that has a data throughput of 20 to 250 kilobits per second and a range of up to 200 meters (Adi & Kitagawa, 2019). The Zigbee alliance creates the network and application layers above the data link layer, based on the IEEE 802.15.4 standard, which defines the lowest levels such the physical layer and data link layer. The Zigbee, like many other communication protocols, operates in the ISM radio band, which is separated into three frequency bands and contains 27 channels (868 MHz, 915 MHz and 2.4GHz). Channel 0 will have a data rate of 20 Kbps and a frequency of 868 MHz. Channels 1 through 10 will operate at 915 MHz with a 40 Kbps data rate, while the remaining channels will run at 2.4 GHz with a 250Kbps data rate (Ali *et al.*, 2019). The Zigbee protocol stack is shown in Fig. 3.

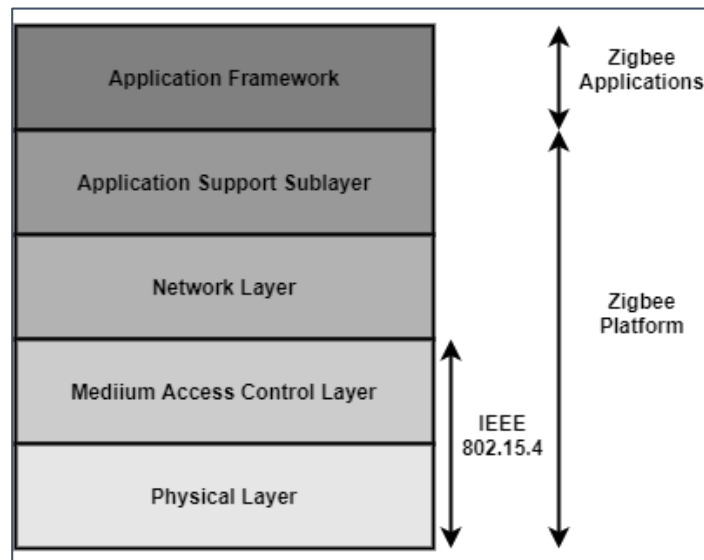


Figure 3: Zigbee Protocol Stack (Prakash, 2020)

2.8.3 Bluetooth

Bluetooth is a low-power, high-speed wireless communication network that connects mobile phones and other devices. It is a standard (IEEE 802.15.1) for wirelessly linking phones, computers, and other network devices across short distances. Bluetooth transmissions generally have a range of only a few meters (up to 10 meters) (Zeadally *et al.*, 2019).

It has a maximum data throughput of 721 Kbps and three voice channels and operates on the 2.45 GHz frequency spectrum. An international agreement compatible with 1.0 devices has permitted the use of this frequency range for industrial, scientific, and medical equipment (ISM). Bluetooth may connect up to "eight devices" at the same time, each with its own 48-bit address based on the IEEE 802 standard, and point-to-point or multipoint connections are both available (Terán *et al.*, 2017).

Instead of using the OSI or TCP/IP models, the Bluetooth design uses its own protocol stack. Another differentiating feature is that not all Bluetooth devices are needed to use all the stack's protocols. This is due to the fact that Bluetooth is designed to be used by a wide range of applications, and each application defines which portion of the protocol stack to use (Lonzetta *et al.*, 2018).

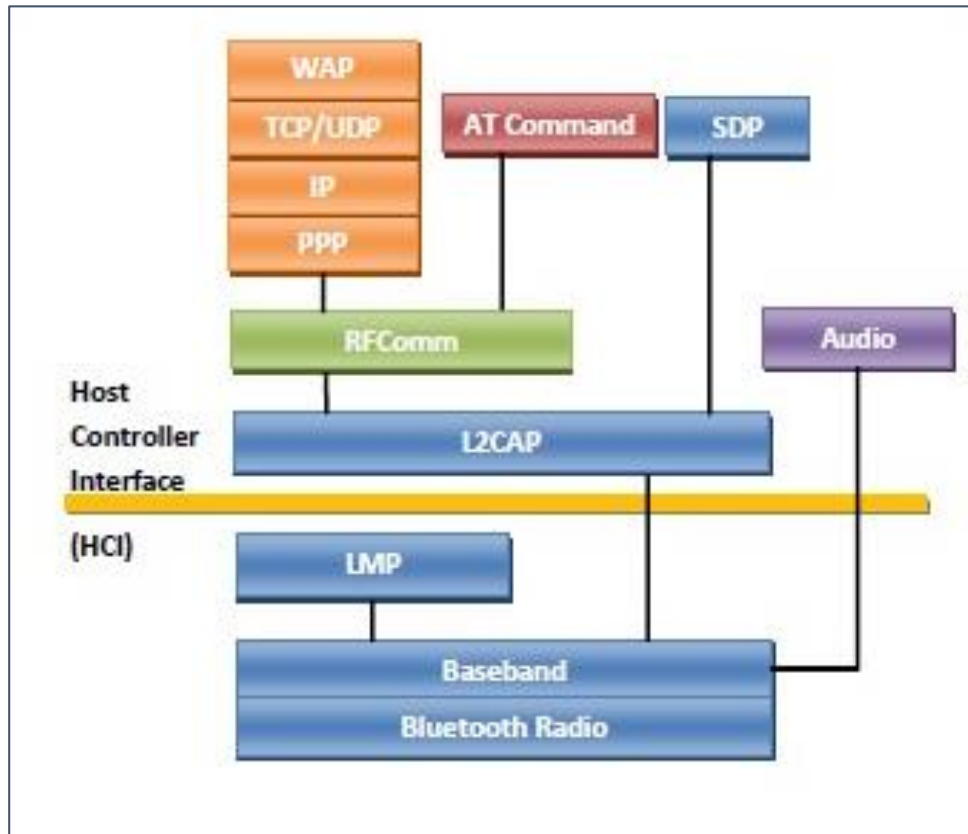


Figure 4: Bluetooth Protocol Stack (Zeadally *et al.*, 2019)

Table 1: Functions of the Core Bluetooth Protocols

Protocol	Functions
Radio	This protocol describes the physical structure and requirements for radio wave transmission at the physical layer. The air interface, frequency ranges, frequency hopping parameters, and modulation algorithms are all defined in this document.
Baseband	Radio protocol services are used in this protocol. The addressing mechanism, packet frame format, timing algorithms, and power control methods are all specified in this document.
Link Manager Protocol (LMP)	To facilitate communication, LMP develops and maintains logical links between Bluetooth devices. Device authentication, message encryption, and packet size negotiation are among LMP's other key responsibilities.
Logical Link Control and Adaptation Protocol (L2CAP)	L2CAP may be used to change the upper layer frame and baseband layer frame formats. The L2CAP supports both connection-oriented and connectionless services.
Service Discovery Protocol (SDP)	To establish a connection between competing Bluetooth devices, SDP addresses service-related inquiries such as device information.

2.9 Solar Photovoltaic Power

Solar PV systems can generate either direct current (DC) or alternating current (AC) power (AC). Since the DC output of the PV array is directly linked to a DC load, it may also be a direct coupled system (Rehman & El-Amin, 2012). It's vital to realize that directly linked systems don't have a battery bank and can only operate during daylight hours.

A maximum power point tracker (MPPT) is also employed between the load and the array to help make greater use of the PV array's available maximum power output as well as match the electrical load's impedance to the PV array's maximum power output (Adam & Fashina, 2019).

2.10 Related Works

Cristian, Constantin, Zoltan, Adina and Florica demonstrated the value and utility of using a PLC to regulate and examine mechanical applications in driving engines. In this system design, the motors were controlled using Siemens PLC and SCADA programming (Cristian *et al.*, 2014).

However, both PLC and SCADA system deployments are quite costly and need extensive training and specialized staff.

Furthermore, a publication on mechanical computerization using Zigbee was presented by Cristian *et al.* (2014). The Zigbee concept is based on a wireless sensor arrangement that is used to manage and administrate various forms in operations without diverting other processes. Nonetheless, Zigbee transmission and control would need a vast wireless sensor network, making this a more complicated system for seemingly easier tasks like machine drive control and monitoring.

Truong and Vu (2012) presented a study on employing remote hardware and the Android platform to remotely control and guide modern operations or applications. For example, a CNC machine was monitored using a remote sensor network. Setting up remote gear and control hardware at the controlling station, on the other hand, was somewhat costly. Large companies/factories with financial clout and experience would gain from this.

Senthilkumar *et al.* (2020), proposed an M2M based Industrial automation using Advanced Microcontroller and IoT with auto-sensing technology based on deployment of ARM workstation, where the workstation is used as both controller and server. The system would automatically monitor and control the industrial electrical appliances whilst giving emergency alerts to authorized personnel without human interference. Acquisitions from machine activities, on the other hand, had to be physically accessed from the ARM controllers, which necessitated the use of technically skilled individuals. Furthermore, when it comes to distribution to Small and Medium Businesses (SMBs), ARM workstations are complex.

Deshmukh and Bhuyar (2018), proposed a control and monitoring systems using various technologies. While interaction between a user and a system is largely accomplished online through wireless communication systems like Bluetooth, RF and ZigBee, SCADA programs are equally employed to develop user interfaces. However, SCADA programs are not adaptable for users owing to their costly libraries. Since the distance between the sender and receiver is short and a small amount of data in transmission, RF, Bluetooth and ZigBee technologies, commonly used in simple-to-use applications, are employed. However, Bluetooth and ZigBee wireless communication methods are largely restricted to simple applications owing to data protection, and their slow communication rates and distances. This gives offline data acquisition systems a design edge for seamless integration of the technologies.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

This Chapter describes the materials and methods used in developing a solar energy powered innovative control and monitoring system of a BLDC Motor and corresponding parts such as a solenoid valve used in a locally manufactured Oil Press Machine using Raspberry Pi. The study was carried out in order to investigate simpler and more effective solutions for controlling and monitoring machine drives. The Chapter is organized in such a manner that it investigates existing Oil press machine operations in the project study region, system requirements, architectural design, and prototype requirements utilized in the creation of the control and monitoring units. It also includes the design of a mobile app for offline data collection and monitoring of the motor's important parameters.

3.2 Project Design Approach

To implement the system, Agile Scrum System Design Methodology was deployed. Since customer demands change so fast, today's industrial environment is defined by complexity. As a result, suppliers are faced with the issue of shortening speed to value while still offering new products that clients expect (Soeteman-Hernandez *et al.*, 2019).

In project management, the Agile scrum technique focuses on incremental development. Each iteration is divided into two to four-week sprints, with the objective of completing the most important features first and delivering a possibly deliverable product at the end of each sprint. In future sprints, more features are introduced to the product, which is subsequently modified depending on stakeholder and customer input in the interim (Dhir *et al.*, 2019).

Agile System Development (ASD) seeks to create operational efficiencies in short iterations and provides value such as on-time execution and product quality (Schön *et al.*, 2017).

The Agile Manifesto contains ideas and concepts that aid in the optimization of the software development process and have a significant impact on today's modern ASD team cooperation. The four essential principles stated below are included in the Agile Manifesto (Van Casteren, 2017).

- (i) Processes and tools vs. individuals and interactions.
- (ii) Overwhelming documentation with the working system.
- (iii) Collaboration with clients is encouraged rather than contract negotiations.

- (iv) Changing one's behavior in conformity with a goal.

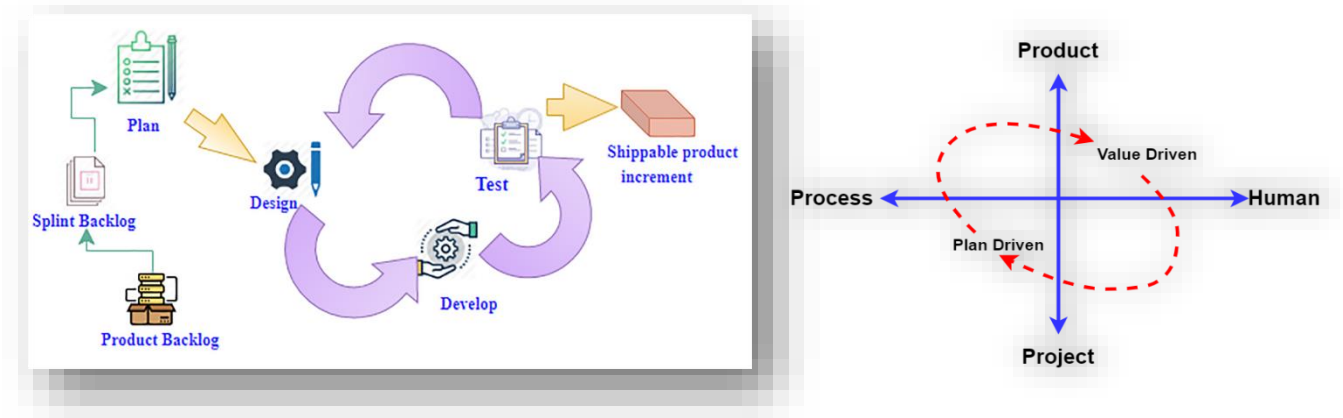


Figure 5: Agile- scrum Methodology (a) and System Design (ASD) with a focus on Outcomes (Van Casteren, 2017)

3.3 Relevance and Rigor Cycle

The relevance and rigor cycle elements were used to direct the process of collecting system requirements in this project by recognizing challenges presented by the created oil press machine. This was accomplished by reviewing similar systems, holding brainstorming sessions, and speaking with support teams.

3.4 Project Area

The project is part of an ongoing piloting of a locally manufactured Oil-Press Machine which was hand operated. The machine is specifically designed for oil-pressing avocado produce for off-grid farmers in Meru District in Arusha region of the Northern Tanzania.

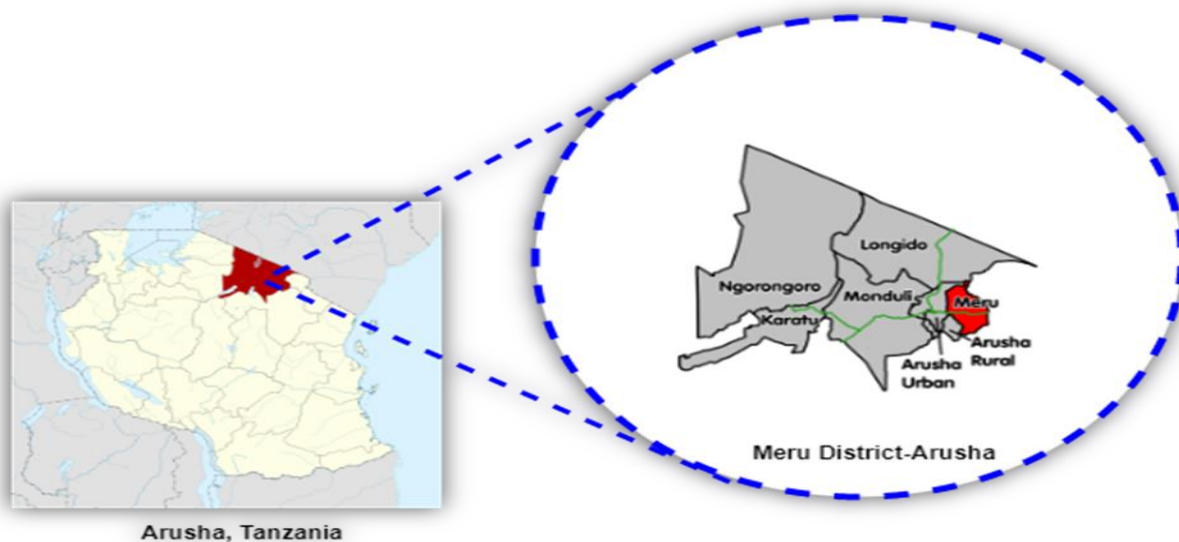


Figure 6: Tanzania map showing Meru District in Arusha Region (Mbabaye et al., 2018)

3.5 Requirement Data Gathering

A two-step data collecting strategy was used to get adequate and relevant information for creating the control and monitoring system for a BLDC Motor for Oil Press Machine utilizing Arduino UNO and Raspberry Pi (RPi). The first stage was to gather existing data on the manual control system and the issues it faced, and the second stage was to use the data acquired in step one to design and test the control and monitoring system utilizing the microcomputer mentioned earlier (Arduino UNO and RPi). The necessary requirements were gathered from the test lab from the 22nd of June to the 4th of July 2021, where both services and constraints were divided into functional and non-functional demands (Table 2). With reference to current knowledge on control and monitoring systems in place and obstacles faced. Several techniques of requirement collection were used in this process, including:

- (i) Personal Interviews, in which engineers and technical personnel in the designing department were asked face-to-face questions using an interview guide. The interview focused on the official's expertise with what control measures they would want and what motor characteristics they would like to capture.
- (ii) System analysis in which need data was gathered by studying the fundamentals of machine drive operation in relation to incorporating a solar powered BLDC motor, and extracting information related to machine control measures. It entailed direct interaction with the machine, as well as the use of books and research papers on BLDC motor controls and monitoring.

Table 2: Requirements gathered from A2EI

Functional Requirements	Non-Functional Requirements
The system shall automate the whole motor and solenoid valve switching process	Security
The system shall have an option for a manual or automatic control	Accessibility
The system shall collect motor parameters (voltage and current) and store in SD Card	Usability
The system shall automatically upload data to a client phone once Bluetooth pairing is detected	
The system shall stop operations once the batteries are below 48V	
The created app will be requested to upload the data to a cloud server once the data is on the client phone.	
The system shall be powered by solar energy and have a backup power plan	

3.6 System Modelling

In order to reflect the functions of the process while decreasing the complexity of the descriptions, modeling was employed during the system's development. The functions, interactions, and outputs of the processes involved in system development were studied. The following are the system modeling approaches utilized in this project:

- (i) A circuit diagram is a representation of an electric or electronic circuit in graphical form. It displays the electrical connections between various electronic components. The circuit diagram was used in this project during the design of a control unit or data acquisition that included motors, controllers, and other modules with varying electrical characteristics. The circuit schematics were created using Fritzing software. The Fritzing program was chosen because it is the most widely used electronic software for making schematic diagrams, has a large number of supporting libraries, provides significant online help, and strongly supports engineering process design (Wu *et al.*, 2017).
- (ii) Flow charts are visual representations of algorithms. Shapes are used to depict steps, whereas arrows are used to illustrate logical flows (Aschwanden-Granfelt, 2017). Flowchart diagrams were primarily employed in this project to construct decision algorithms that were used during the control of machine drives as well as the reading and transmission of recorded data between the controller and a mobile smart phone.
- (iii) A use case diagram is a visual depiction of how a system functions from the user's perspective. It's a diagrammatic representation of how system actors interact with the system to meet functional objectives (Firesmith, 2014). The following factors influenced our decision to use a use case diagram as a design tool in this project: It is used to verify higher system models and validate system requirements; it is used to manage complex applications and provide timing requirements; it is simple to understand and can be used to communicate with system users; and it is used to manage complex applications and provide timing requirements.
- (iv) The Data Flow Diagram (DFD) is a graphic depiction of a system's logic. It depicts the flow of data from the source to the steps of processing and storage (Li & Chen, 2009). The DFD was used to mimic the interconnectedness of systems and subsystems in the proposed system.

3.7 System Implementation

This section presents the processes utilized to create a control and monitoring system for a locally manufactured Oil Press machine. It comprises the following subsections' information for hardware selection, mobile application development using flutter, Visual Studio editor, and template creation

3.7.1 Components Selection Method

Seven stages were taken to choose components for the creation of a Motor Control and Data Acquisition Unit (DAU) for the proposed system: Determining the best approach to getting control information from human viewpoints, identifying the control target based on system needs and information to be controlled, determining the control target viewpoint based on determining the best approach to getting control information from human viewpoints, Examination of module conditions in relation to system requirements, listing of module restrictions, and selection of qualified modules to be used in system development based on module type and physics, when compared to an actual physical phenomenon, the module operation level is identified, examination of module conditions in relation to system requirements, listing of module restrictions, and selection of qualified modules to be used in system development.

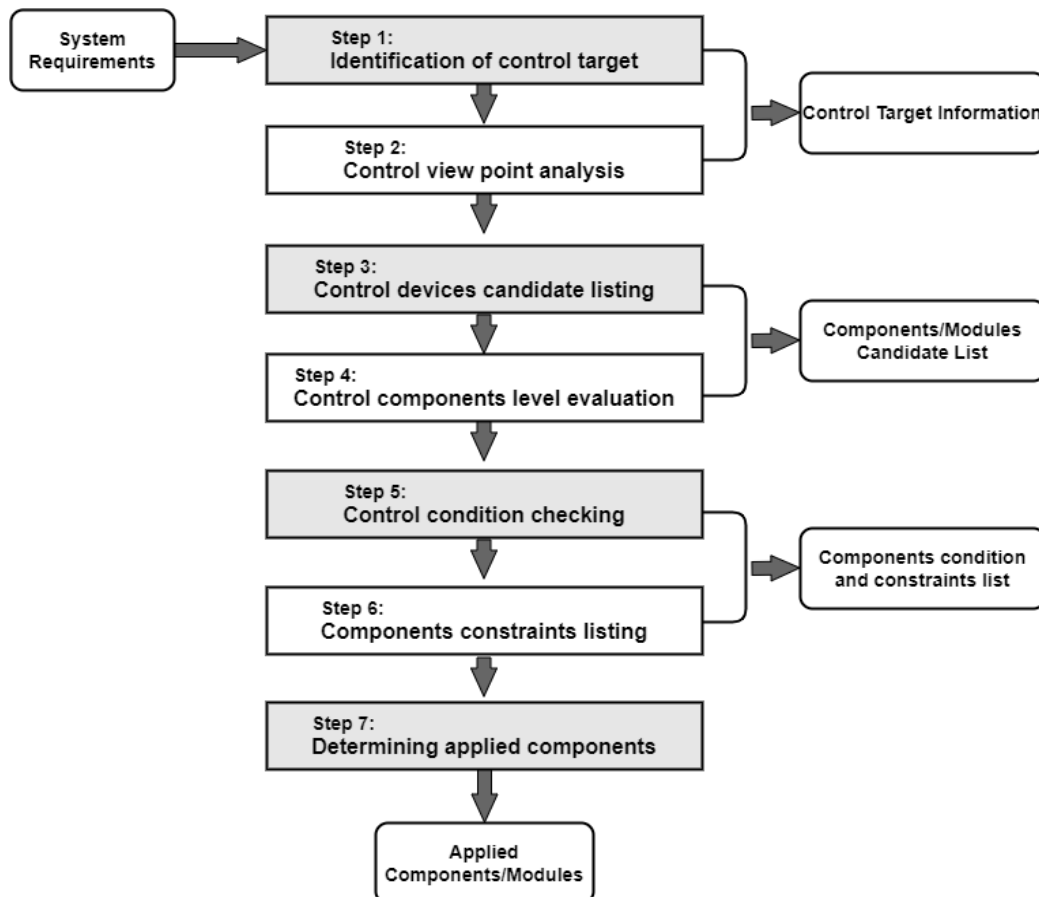


Figure 7: Flow of components/modules selection Method (Van Casteren, 2017)

3.7.2 Hardware Components

Hardware used included Raspberry Pi, Arduino Mega Microcontroller, 4-Channel Relay, a DC Fan, and an android phone (Samsung) which has internet as well as Bluetooth capabilities all grouped in as either the power supply unit, the control unit, the transmission unit or the monitoring unit as they will be discussed in detail in the sections that follow.

3.8 System Design Architecture

The Power Supply Unit, Control Unit, Transmission Unit, and Monitoring Unit are the four components of the system design architecture. The following is a list of these components:

3.8.1 Power Supply Unit

This is a subsystem of our system that supplies DC voltage to our machine drives and controllers. It is a result of harnessing solar energy where to supply adequate power to other subsystems we had an array of components that made up the whole power supply system.

(i) Photovoltaic (PV) Solar Panel (200 W)

The PV panels are used to generate power directly from sunlight. These panels are made up of several individual cells that are linked together to create a specific voltage of power. The PV are inherently DC devices, producing DC voltages with varying currents depending on the solar radiation level for the day. This current produced is a necessary attribute for charging the battery bank where our total system uses DC only (no inverters).

We have chosen the 200 W PV panels specifically depending on the power needed to run the 3 kW Motor for the Oil Press machine for at least eight (8) hours a day without recharging. More calculations on how we arrived to the choice are elaborated in the Chapter Four.

(ii) Maximum Power Point Tracking Solar Charge Controller

A DC to DC converter transforms the voltage output of solar panels to a voltage that would be used to charge a battery bank. The MPPT charge controllers allow the PV array output voltage to be greater than the battery bank output voltage without causing power loss. This controller has been chosen specifically for battery protection and providing the appropriate voltage output (i.e., 48V) for our load.

(iii) Solar Battery

Even in the roughest of circumstances, the 12V 200Ah Deep Cycle Hybrid GEL Battery is an excellent choice for standby or everyday power demands. This battery is made to last, with a leak-proof, maintenance-free design that provides excellent performance and a long life. Because of its capacity to resist more than 750 charge/discharge cycles at 50% DOD, this battery is ideal for sensitive applications. We have made use of these batteries as a power bank, since the machine operations are solely dependent on the power harnessed and stored in the batteries.



Figure 8: The Solar panel (a) the MPPT charge controller (b) and the solar battery (c) (Robles *et al.*, 2017)

(iv) LM2596 Buck Converter

This is a 20W DC-DC buck converter module that may be adjusted. A screw potentiometer, which is likewise included on the board, may be used to modify the output. Batteries, power transformers, DIY adjustable power supplies, 24 V car power supplies, industrial equipment, 12 V to 3.3 V, 12 V to 5 V, 24 V to 5 V, 24 V to 12 V, 36 V to 24 V, and so on, may all benefit from this module. Self-calibration is possible using the on-board voltage meter (Alsumady *et al.*, 2021).

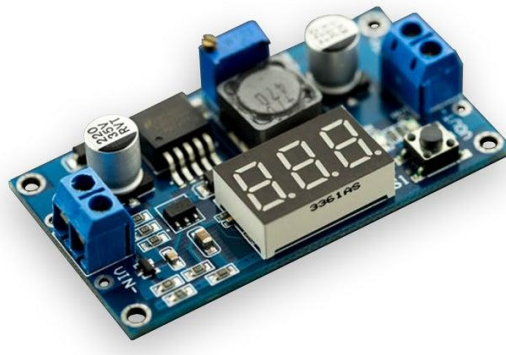


Figure 9: The LM2596 Buck DC-DC Converter (Ferrer *et al.*, 2017)

This Buck converter was used to provide different voltage levels for powering different subsystems that require a specific voltage to operate.

3.8.2 Control Unit

This represents the heart of machine drive and control mechanism of the oil press machine which consists of a BLDC Motor and its controller, an Arduino microcontroller, relay switches and the solenoid valve.

(i) Arduino UNO

This is a USB-connected microcontroller board featuring an ICSP header, a 16 MHz ceramic resonator, 14 digital input/output pins, a power connector, a reset button, and 6 analog inputs. The board features 32 KB of flash memory (0.5 KB of which is utilized by the boot loader), 1 KB of EEPROM, a clock speed of 16 MHz, and 2 KB of SRAM (Nussey, 2013). It provides a set of digital and analog pins that may be used in conjunction with a variety of other circuits and boards that serve entirely different functions in the design. For loading the programs from the computer, the board has a USB serial-communication port. The integrated development environment (IDE) for Arduino is a piece of software that fully supports the C and C++ programming languages for programming the board (Deshmukh & Bhuyar, 2018). We deployed this microcontroller as the heart of our control system, defining trigger conditions and timing profiles for switching between the motor and solenoid valve.

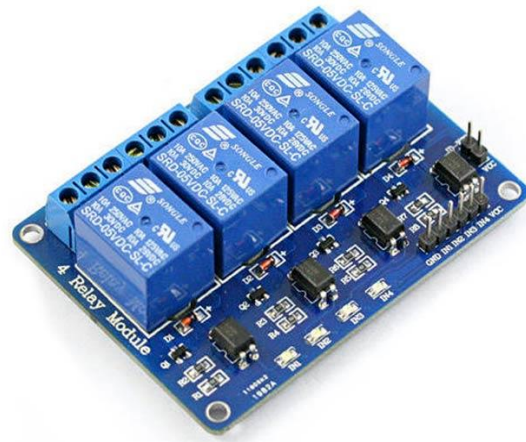
(ii) Relay Switch

Since a relay is an electromagnetic switch, its heart is an electromagnet powered by a little current that functions as a lever or as the switch itself, this played a critical part in timely switching of the Motor and Solenoid valve for machine control. This enables very modest electric currents to leverage and regulate considerably larger electric currents (Bagde *et al.*, 2021).

Sensors are delicate devices that create little quantities of electric current; nevertheless, in order for a sensor to power larger machines, it requires something to turn them on by allowing larger currents to flow. The magnetic that attracts the armature is powered by the modest control current. The armature connects to the other end of the circuit, completing it and allowing electricity to flow. There are several sorts of relays. Attraction Type Electromagnetic Relays, Induction Type Relays, Hybrid Relay, Thermal Relay, Reed Relay are all examples of electromagnetic relays. We have deployed a four-channel relay switch module to accomplish the timing switch between motor and solenoid via microcontroller (Arduino UNO).



(a)



(b)

Figure 10: Arduino UNO (a) A 4-channel 5-VDC Relay Module (b) (Arduino, 2021)

(iii) The BLCD Motor -KY130AS0430 Model

This is a 48v Brushless DC Servo Motor having a total revolution per minute (rpm) of 3000 rpm. The motor has an IP55 protection grade and an insulation grade F. The motor has unique characteristics such as winding overhang structure optimization to reduce copper and iron loss, compact volume, light weight, low temperature increase, and high efficiency. Additionally, the model has a very high coercivity - the highest magnetic energy product NdFe35 permanent magnetic materials, a high resistance to demagnetization, and therefore a highly steady motor performance (Keya, 2021).

The model also has low noise (only 45dB (A)), low vibration, low moment of inertia, high torque (up to 25 000 Nm/kgm²), rapid dynamic reaction (time constant 20 ms), a wide speed range, and a robust overload capacity (four times), making it an excellent choice for the Oil-Press machine.

The motor model was specifically used to drive hydraulics fluid for the hydraulic winch during the compression of the avocado materials. The choice of the motor was solely based on its powertrain (3 kW) and the current ratings which in return would determine how long the Oil-Press machine would run with a fully charged battery pack.

(iv) The Motor Controller (KYDBL48150-1E)

The brushless DC controller KYDBL48150-1E is clever. It uses a 32-bit high-performance MCU with an advanced motion control algorithm that is suited for the outside quadrature encoder input to complete the open loop and closed loop speed, as well as the closed loop torque of motor sport (Keya, 2021). The functionalities of a controller with numerous analog input ports, pulse input

ports, and digital I/O ports may be redefined using specific software. The CAN bus, a universal serial port communication protocol, is widely used in automation (Keya, 2021).

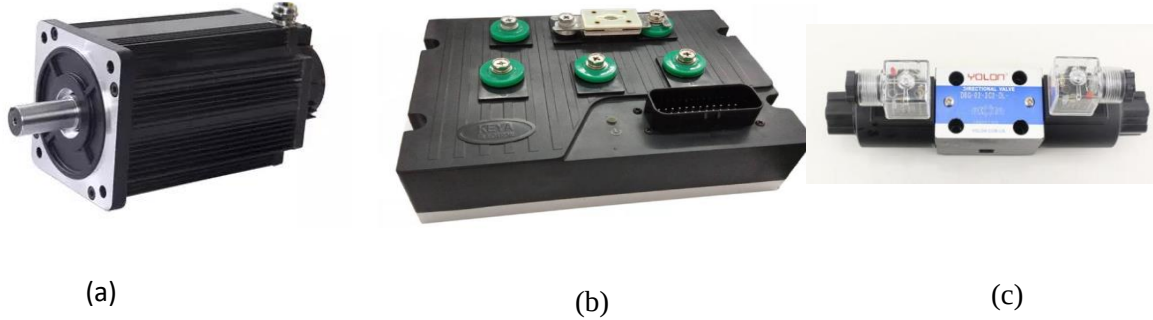


Figure 11: The BLDC Motor Model KY130AS0430 (a) the KEYA Electric KYDBL48150-1E model controller (b) and the DC Solenoid valve (c) (Keya, 2021; Aliexpress, 2020)

This Motor controller coupled with the BLDC Motor together completes the drive train of the oil press machine. The Arduino UNO microcontroller adds an external control functionality that sets aside the procedures and timings of the motor operation, and the solenoid valve as will be explained later in the next chapters.

(v) Solenoid Valve (DSG-02-3CS2-DL)

These are DC-powered valves that have been modified with robust wet-type solenoid and circuit designs to achieve high pressure (300 Bar), big volume flow (80 L/min), and minimal pressure drop. This solenoid consumes the least amount of electricity, with an operating current as low as 1.3A. We have deployed this solenoid to work in tandem with the hydraulics system of the Oil press machine. The bidirectional functionality offers the compression and release of a pressing shaft.

3.8.3 Transmission Unit

This part consists of a Raspberry Pi (RS232 port) and an in-built Bluetooth Wireless Communication capability of the RPi. The core functions of the Raspberry Pi are to record operational parameters for the motor from the motor control connected to it via the RS232 port of the controller and the USB port of the RPi and to relay the saved data to client mobile equipment. The RPi Model 3B has a built-in Bluetooth capability which via the technology we passively ‘airlink’ the saved files using Bluetooth to the client’s application on the android mobile phone and then to the cloud once internet resources are made available.

(i) Raspberry Pi

The Raspberry Pi 3 Model B is a pocket computer with an updated ARMv7 multi core CPU and Gigabytes of RAM. It appears to be a pocket computer that can transition from being a "toy computer" to meeting real-world desktop PC needs (Upton & Halfacree, 2014). The Fig. 12 shows the circuit board diagram of raspberry Pi. The major update is from the BCM2836 (single core ARMv6) to the BCM2837 (dual core ARMv6) (quad core ARMv7). When compared to multi-core CPUs, the processing speed rises by two times. The speed of a system may be increased by 4 to 7.5 times by effectively utilizing architecture. This processor boosts the speed with which you can browse the web and play games. All additional daughter boards would run at 99% efficiency on the Pi 3.

'Sudo apt-upgrade' is how the Pi 3 works. The Raspberry Pi 3 has a quad-core 64-bit CPU, Wi-Fi, and Bluetooth built in. The RAM stays at 1GB, and the USB and Ethernet ports remain unchanged. The Pi 3 should, however, be able to use more power-hungry USB devices thanks to improved power management.

Setting up VNC to Support Remote Control of Raspberry Pi

Because there was no HDMI monitor on site, the VNC protocol was used to offer graphical remote access to the Raspberry Pi. To begin, get the Raspberry Pi's IP address and connect to it through terminal using any SSH program. After having access to Raspberry Pi, it was necessary to install Tight VNC Server, configure and start it. Finally, all that was required to create a connection using a computer was to enter the RaspberryPi's IP address into any VNC program. To prevent having to start the VNC server manually every time the Raspberry Pi is turned on, a script to start it automatically was added to the boot sequence. The steps to do this are available in the Appendix 5.

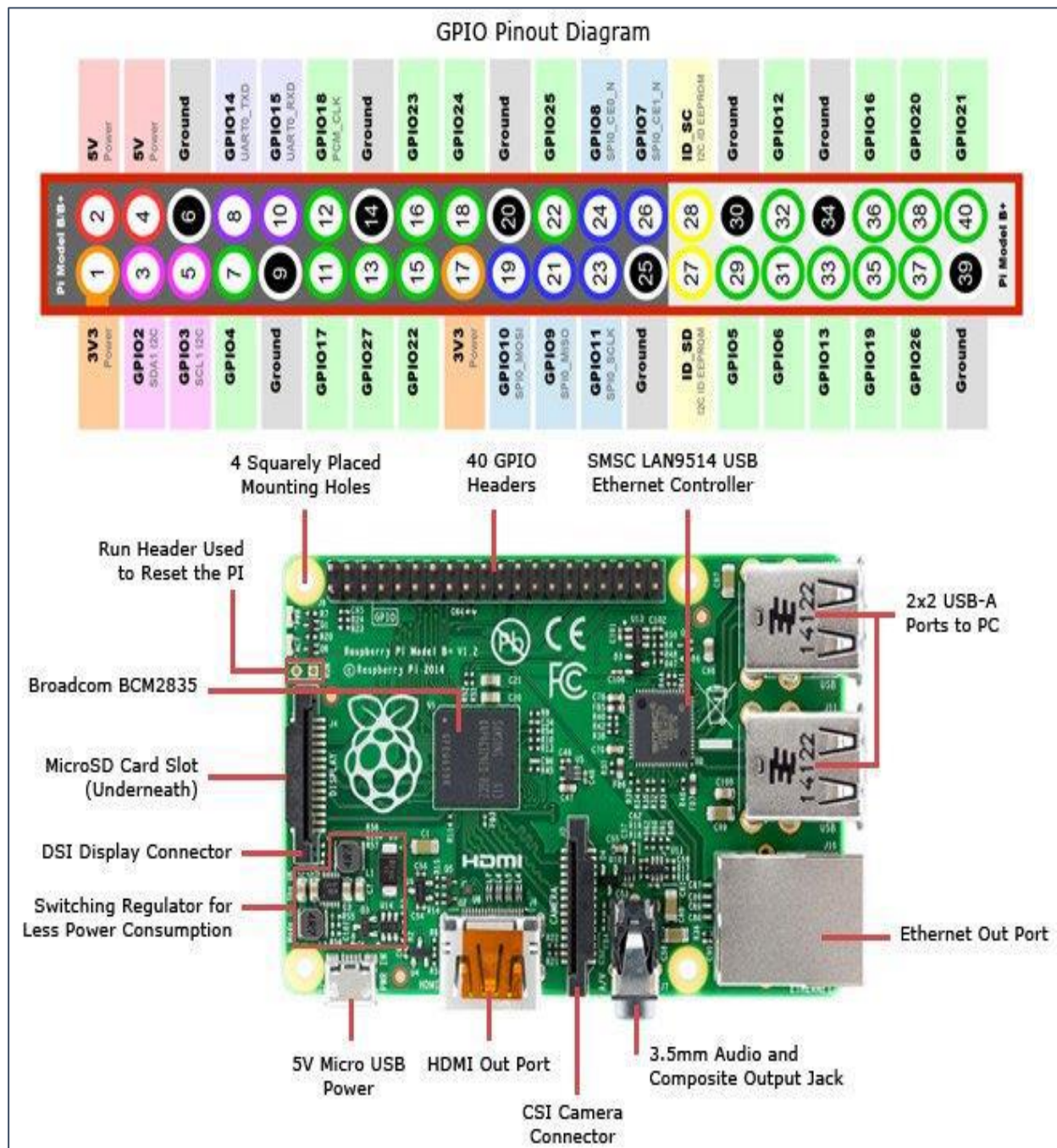


Figure 12: Raspberry Pi 3 Model B (Raspberrypi, 2021)

Python

Python was created in the late 1980s and may be run on any operating system that has a Python interpreter. Table 3 presents the advantages and disadvantages of using Python. Without changing the software, the code may be created once and run on practically every computer. It may be used to construct games, online applications, network servers, scientific computing, media tools, application scripting, and pretty much anything else on a computer. The Google search engine, the YouTube video-sharing website, and the NASA Corporation all utilize it on a regular basis (Van Rossum & Drake, 2014).

Table 3: Advantages and drawbacks of Python

At the moment Python supports both 2 and 3 versions. The version 2.7.13 is the latest stable version,

Advantages	Drawbacks
• Easy to read code • Object Oriented Programming • Large number of libraries for Python • Open source • Great community	• Python is slower than compiled languages • Python is not a very good language for embedded systems development • Lack of true multiprocessor support

which will be supported up to 2020. For this project Python 2.7.13 was chosen, this decision came from the fact that Python 2.7.x has more relevant supporting libraries.

Table 4: Raspberry Pi and Arduino UNO comparison with other Single Board Computers (SBCs)

Parameter	Raspberry Pi	Arduino UNO	Intel Galileo	UDOO NEO	Beagle Bone
Processor	Broadcom BCM2837 ARMv7 Quad core 1.3Ghz	ATMEGA8, ATMEGA 1280	Intel Quark X1000-400MHz single core	Free scale I MX 6soloX 1Ghz ARM Cortex M4	AM335 1GHz Cortex A8
RAM	1 Gb	16-32Kb	512Kb on chip SRAM 256Mb DRAM	512Mb or 1Gb (full)	512Mb DDR3 RAM
Power	10W	5W	15W	10W	15W
OS	Raspian, Debian, Fedora, ARCH Linux ARM and FreeBSD	N.A	Arduino Linux distribution for galileo Windriver Rocket	Android Lollipop and Linux UDOObuntu2 (14.04 LTS)	Android, Debian, Angstrom, yacto, Fedora and Ubuntu
COST	\$40	\$30	\$70	\$65	\$55

Kaup *et al.* (2018)

When the Raspberry Pi model and the Arduino boards are compared to Single Board Computers (SBCs), as shown in Table 2, The Arduino Uno microcontroller and the Raspberry Pi are the most cost-effective and energy-efficient options. Finally, the raspberry pi utilized in the above table is the most robust raspberry pi, namely the raspberry pi model 3 B+, which is a cost-effective SBC when compared to others and is also capable of complicated job management.

(ii) Communication Protocols

Communication technologies play an essential role in the IoT domain. On the Internet of Things, devices exchange data with one another over wired or wireless protocols. However, wireless technology is gaining popularity over wired technology due to the fact that wireless sensors are less expensive and can be installed even in locations where cabling is difficult. Wireless protocols are

required for devices to communicate wirelessly. Bluetooth, RS232, Wi-Fi, Zigbee and Low Power Wide Area Network are some examples of wireless protocols (Feng *et al.*, 2019).

Bluetooth

Bluetooth began as a wireless, short-range cable replacement technology, but it has progressed significantly over the past two decades. Bluetooth radios are now present in nearly all computer devices, including PCs, smartphones, smart watches, and microcontrollers (Zeadally *et al.*, 2019).

The Raspberry Pi 3 Model B comes with an in-built Bluetooth capability where one can connect a plethora of accessories that are also Bluetooth enabled including the Android smart phone that we are deploying on this project.

Once the Motor operation files have been ‘airlinked’ from the RPi to the mobile phone via the designed mobile application, the same are going to be uploaded to the desired server for cloud storage, sharing and analysis purposes.

RS232

During the operation of the drive system, the RPi microcontroller sends instructions to the servo motor controller to gather operational data. As a result, the communication system requires communication between the RPi microprocessor and the driving system.

The RS-232 is a standard contemporary computer interface that contains ports COM1 and COM2, the latter of which features a 9-pin connection. The RS-232 communication connection is connected to the current generation computer through a 9-pin connector (DB9) (Fig. 13). The method employs a three-wire connection strategy to enable trustworthy and real-time transmission, which means the GND, RXD, and TXD pins of the RS-232 port are linked to the external port (Han & Kong, 2010).

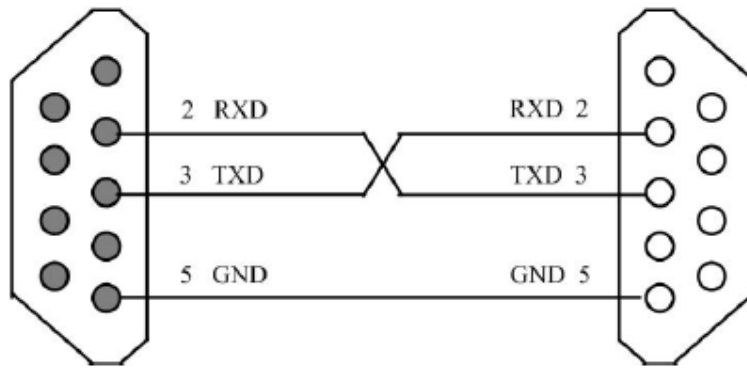


Figure 13: The RS232 Port configuration (Erraissi *et al.*, 2018)

3.8.4 Data Acquisition Unit

This is the application software that allows the user to monitor the operation mechanisms of the motor drives. Use case diagram and database schema were developed before implementing the actual mobile application. The diagrams serve as reference when implementing the system.

Conceptual Design

The platform structure/components are presented in the design concept. Based on the project discoveries, there are several problems monitoring locally made agricultural machinery, some of these problems local methods of preventive maintenance data collection based on experience of operators. To resolve these issues, a Bluetooth enabled mobile app will be developed. The architectural design as shown in Fig. 14, provides links between the control systems and the mobile app for collecting useful operation information of the motor drive.

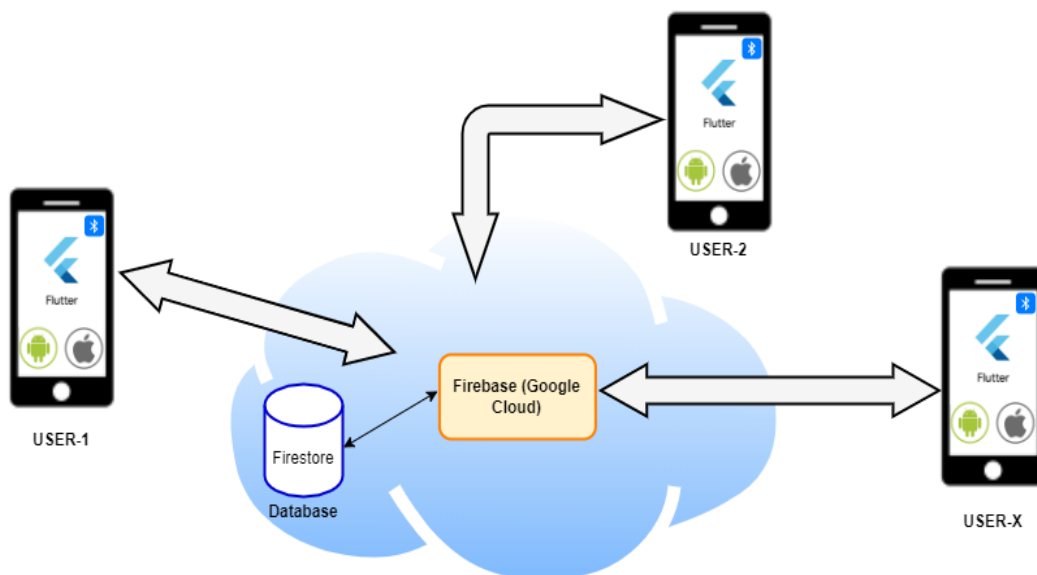


Figure 14: The data logger system architecture

The proposed data logging system consists of the mobile app which will be developed using flutter and firebase, the Google cloud service in which the authentication procedures, database and storage will be hosted.

Use Case Diagram

This depicts the relationship between the application system and its users. It defines the types of activities or procedures that a user may accomplish in the system. Actions are referred to as use cases as is shown in Fig. 15. Cases are referred to as actors which are the users of the system for example: the system administrator, engineers, and field technicians. The use cases include; login, logout, add new user, delete user, change the name, change the user password, pair a node, collect node data and export node data to a cloud server. Refer to Table 3 for descriptions of use cases requirements.

(iii) Database Design

This is the process of organizing information according to a database model and the method of distinguishing individuals, their relationships, and all their attributes. The database designer decides on the information that should be processed and how to relate it to other information. This is to say that database design is the process of classifying and determining the interrelationships of data (Bolchini *et al.*, 2007).

Firestore database from firebase cloud service was deployed in our system, which does not include Structured Query Language (SQL), but also provides easy navigation and sort functions for queries.

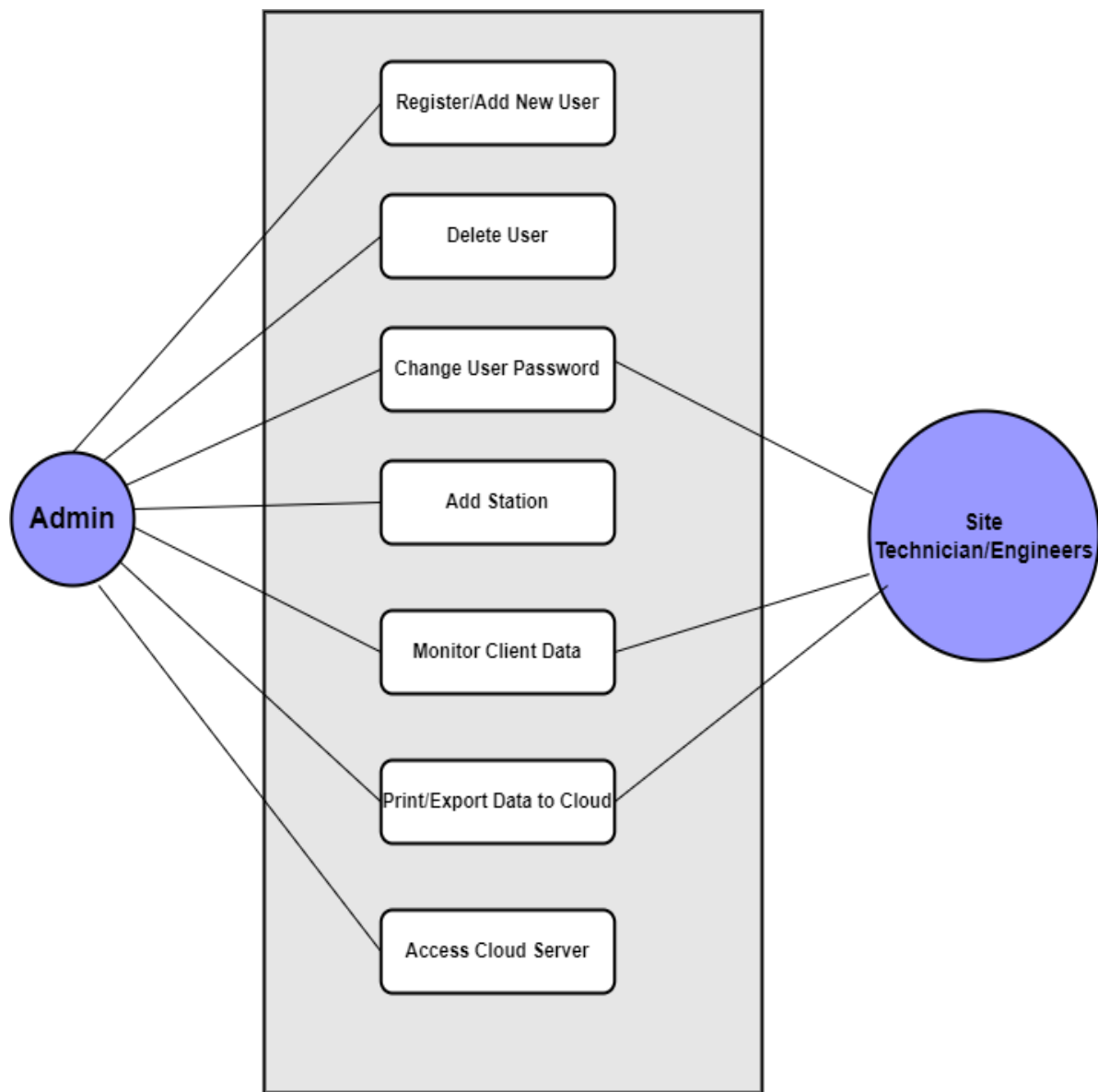


Figure 15: Use Case diagram showing the administrator and users (Engineer/Technician)

Table 5: Described use case requirements

SN	Use Case	Description
1	Register/Add New User	Admin should be able to Register New User so as the users
2	Delete User	The system should aloe admin only should be able by to delete users
3	Add Station	The system should allow the admin should be able to add stations
4	Monitor Data	Admin/User should be able to see data in App
5	Print/Export Data to cloud	Admin and User should be able to upload data to cloud
6	Access Cloud Server	Only Admin should be able to access cloud storage

(iv) **Mobile Application System Development Tools**

During the development of a data logging system, we made use of several readily available free tools such as Android Development Studio, Flutter Software Development Kit (SDK), Firebase and a source code editor, the Visual Studio.

Android Development Studio

The official Integrated Development Environment (IDE) for Android app development is Android Studio, which is based on IntelliJ IDEA. Android Studio, in addition to IntelliJ's advanced code editor and development tools, offers extra features that help you create Android apps faster (Smyth, 2017). It was chosen for this project because it is a flexible Gradle-based build system with a fast and feature-rich emulator, a single environment where you can develop for all Android devices, and it allows you to make changes to your running app without having to restart it (Esmaeel, 2015).

According to Craig and Gerber (2015) It includes code templates and GitHub integration to make it easier to develop common app features and import sample code, as well as extensive testing tools and frameworks, lint tools to detect performance, usability, version compatibility, and other issues, C++ and NDK support, and built-in support for Google Cloud Platform, making it simple to integrate Google Cloud Messaging and App Engine.

Flutter Software Development Kit

Flutter is a portable user interface toolkit developed by Google for producing appealing natively created programs for mobile, web, and desktop from a single codebase (Napoli, 2019). Tim Sneath, Google Group Product Manager, describes it as "a robust general-purpose open UI toolkit" Only the iOS and Android mobile platforms have built-in compatibility at the moment (Fig. 16).

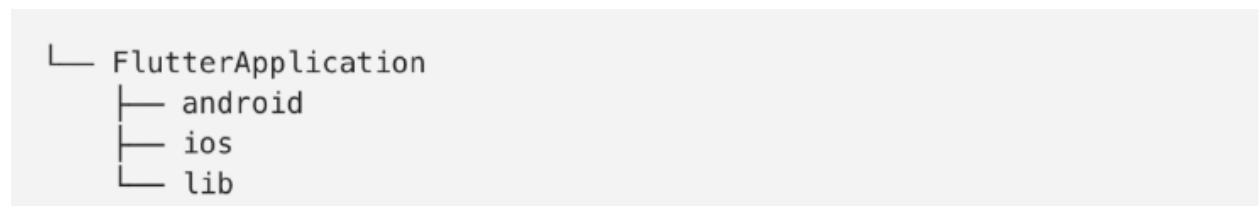


Figure 16: Flutter application structure (Flutter, 2021)

Flutter is the most forward-thinking way to cross-platform app development. It has its own set of interface objects, renderer, and engine that implements flutter's animation, graphics, file, and network I/O, among many other essential components (Flutter, 2021).

Flutter projects are written in the Dart programming language and constructed ahead-of-time to the native platform architecture, resulting in unmatched performance.

At its most basic level, Flutter features widgets that are made up of several widgets to create the most common interface components that we're used to seeing on the iOS and Android platforms. Developers may create their own widgets by combining existing widgets at any level of the layered architecture, thanks to flutter's open and layered design (Zammetti, 2019). The Flutter team built the current high-level widgets in this fashion, and there is no reason why developers shouldn't do the same. The iOS and Android UI toolkits with hierarchical implementations and limited access levels cannot match this level of customization flexibility (Dagne, 2019).

Source Code Editors (Visual Code)

These are applications designed exclusively for writing and editing computer program source codes. They can be separate applications or frameworks, like Notepad++, or they can be incorporated into an IDE or web browser (Sulír *et al.*, 2018). They are extremely important programming tools because they allow the development of an executable source code file for usage with a compiler/linker. Example, while configuring and programming the Arduino UNO microcontroller and sensors, the program created in C by an Arduino IDE, as illustrated in Fig. 17, is used.

Visual Studio has a code editor with IntelliSense (code completion) and refactoring capabilities. The inbuilt debugger may be used to debug both source and machine code. Other built-in tools include a code profiler, a designer for creating GUI apps, a web designer, a class designer, and a database schema designer (Johnson, 2012). It enables plug-ins that enhance functionality at nearly every level, such as support for source control systems (such as Subversion and Git) and the inclusion of new toolsets for domain-specific languages or toolsets, such as editors and visual designers (Amann *et al.*, 2016).

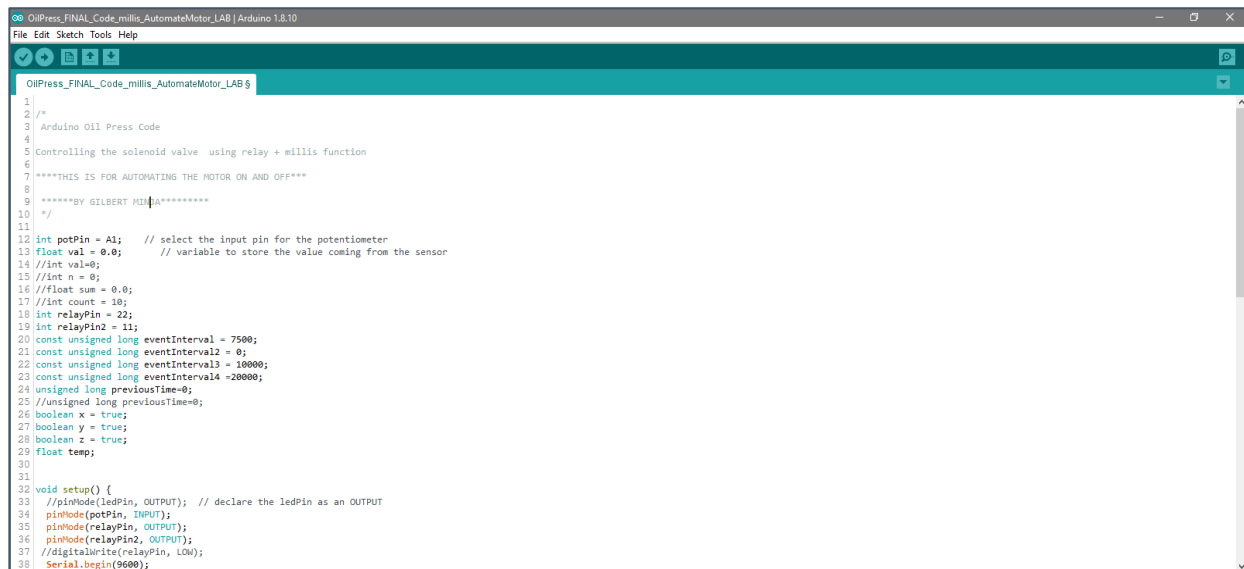


Figure 17: Arduino IDE used to write the control codes

3.9 Advantages of the Developed System

3.9.1 Technical Feasibility

This system is specifically built for authorized users who may utilize the system the system controls during the Oil press machine operation and initiate secure communication over Bluetooth on the outskirts of the present application. Technically, the system is viable. All the resources necessary for the system are simply accessible.

3.9.2 Availability of Monitoring Data from Controller

Even if aberrant situations are discovered as they occur and solutions are readily available, the collected data can assist in mitigating future failures. Remote access to motor operating data can be obtained by technicians/ engineers from the comfort of their own homes.

3.9.3 Use of micro-controllers (Arduino and RPi)

They are low-powered architecture with good online support and quick prototyping capabilities that can communicate data wirelessly to the server via a device with many GPIOs and PWM capabilities, as well as being developer-friendly. Arduino UNO would aid a time-to-time upgrade of the program by simply replacing the microchip with an updated one.

3.9.4 Future Development

This DC-based project for Motor control may be developed further by supplying new modules, sensors, and functionality to a wide range of businesses that employ machine drives. Companies might have greater management and surveillance of their businesses with the use of Android applications. Furthermore, it allows for the integration of various systems that may be merged to ensure efficiency in the field of industrial automation in industries.

3.10 System Testing and Validation

Following the development of the system, unit testing was carried out to ensure that each module functions properly and accurately. The effectiveness and dependability of data monitoring were then assessed by evaluating system performance. Finally, a representative group of A2EI engineers participated in user acceptability testing.

3.10.1 Test Requirements

The test requirements for IoT devices and generated mobile applications are widely focused on security, analytics, device, networks, processors, operating systems, platforms, and standards. The

following are the test criteria and goals: A huge quantity of data and its application by completing user authentication testing.

Security testing: In the mobile app environment, numerous users access a large quantity of data. As a result, it is critical to confirm the user through authentication by supplying a username and password:

- (i) As part of the security testing, it is vital to assess the data integrity and security of the established system, as well as its authentication and data privacy procedures.
- (ii) Performance testing is essential for building a strategic strategy for developing and implementing a testing plan.
- (iii) Testing for Reliability and Scalability: Creating an IoT test environment necessitates modeling building pieces using virtualization, which necessitates testing for reliability and scalability.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Results

This Chapter summarizes the project's findings and examines the outcomes. It also details the design and execution of a control and monitoring system for a solar-powered BLDC motor for an oil press machine. In addition, the findings of the system's modeling and testing are provided and debated.

4.1.1 System Implementation Overview

The project created a system prototype that operates a BLDC motor externally through the motor's control unit, gathers motor operating data, stores it on an SD card, and passively uploads it to a cloud storage service. The prototype indicated that the motor control for the operation of the oil press machine was executed, and that operation data such as voltage and current were passively communicated to the firebase database for storage and subsequent viewing. The A2EI engineers were able to access the cloud-stored data for a study of the motor. This allows engineers to monitor the motor's critical working status. The proposed method assists in the creation of awareness in the areas of motor and machine drive control, as well as boosting operational efficiency. As indicated in Appendix 1, the created system in Fig. 18 is a mix of hardware and software, and the computer code was built in Thonny IDE using the Python language.

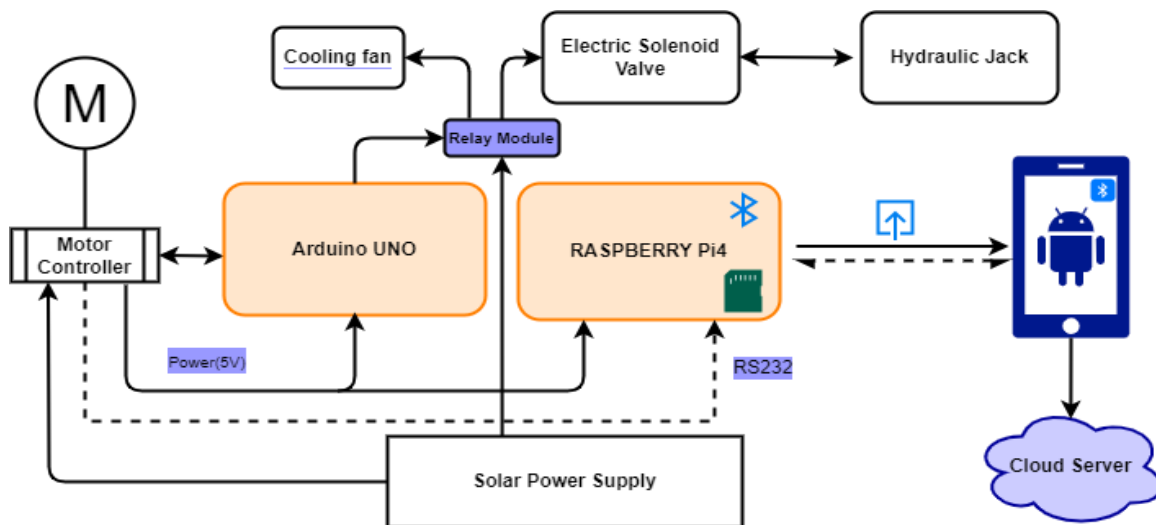


Figure 18: The block diagram of the motor control and monitoring system

4.1.2 Control System Unit

This subsystem involves properly connecting an Arduino Uno with several modules, and it performs well and delivers the intended outcomes as specified in the project. The predefined control parameters based on timing were executed appropriately as per the requirements. The MCU in

tandem with the Motor controller produced results that were expected of the machine drives operation. Figure 21 shows the prototype of the whole system including the control subsystem.

(i) Flow Charts

Figure 19 displays how the flow of events proceeds during monitoring data acquisition for the control system.

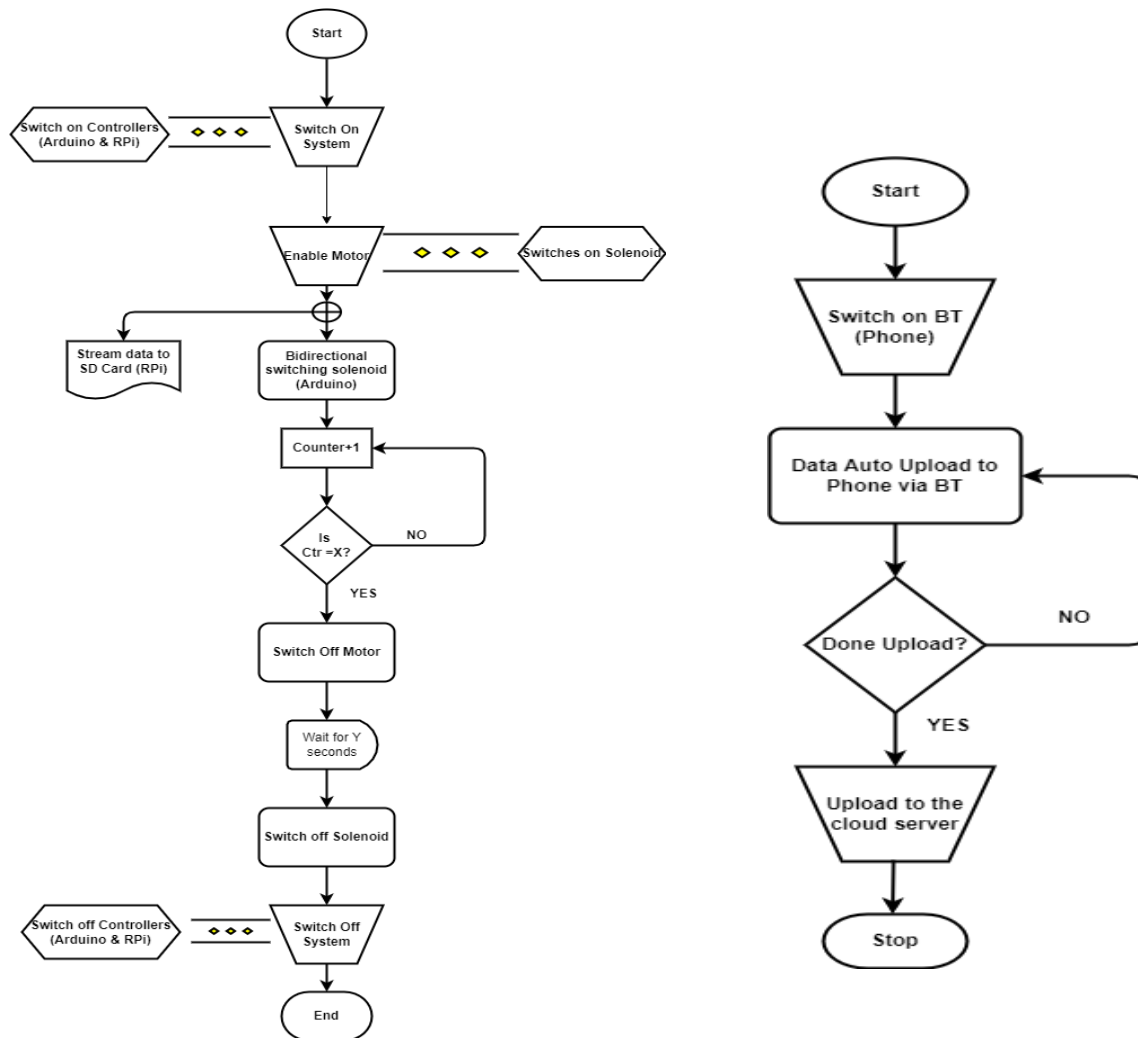


Figure 19: The flow of events for the control system and monitoring data acquisition system

(ii) The Circuit Diagram

The Circuit diagram in Fig. 20 depicts the interconnection of all the building blocks of the System from the power supply unit, control unit and the monitoring unit.

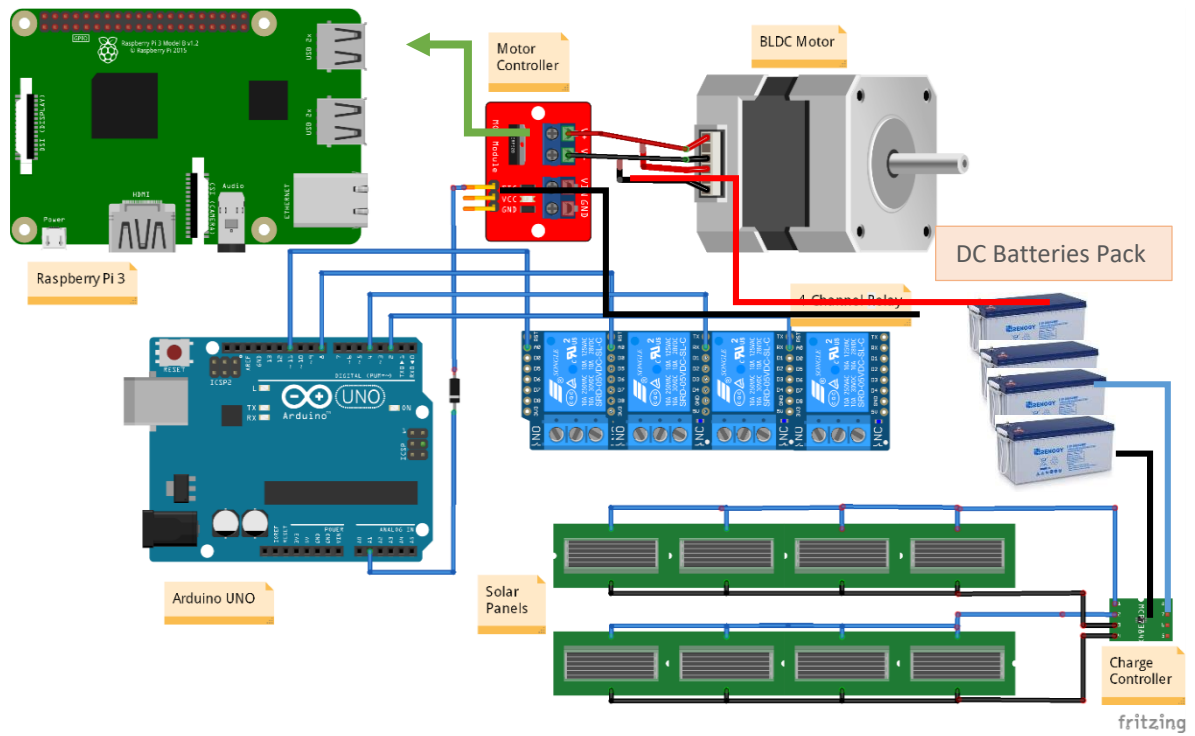


Figure 20: The complete Circuit diagram of a Power Supply, Control and Transmission with their interconnections

(iii) The Developed Control Box Circuit

This developed control box circuitry represented the interconnect of all building blocks of the control and monitoring system by piecing together the Power supply unit with the DC buck converters, microcontrollers (Raspberry Pi and Arduino UNO), the Relay switches, the cooling fan and the solenoid connection as seen in Fig. 21.

The control box was also equipped with switches that would provide the ON/OFF functionality of the machine and the control box, together with the switches for the bidirectional solenoid when an operator wants to change direction of the oil pressing shaft.

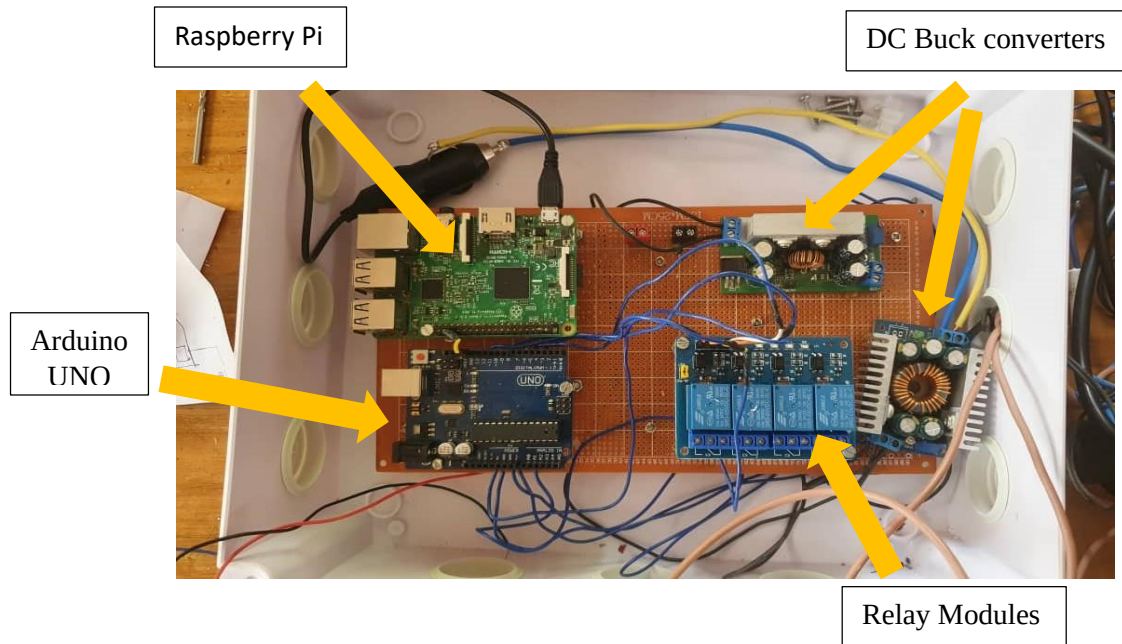


Figure 21: The Developed Control Box circuit

4.1.3 The Data Acquisition Mobile Application

The designed mobile application system would be mostly accessible via the company's premises. The app-based data collecting tool for system monitoring includes user administration, station management, account settings, and a logs dashboard. The system's privileged users are administrators and A2EI engineers. The administrator has the authority to increase or reduce the number of features to which other parties have access. End-user functionality is limited to self-registration, report uploads to the server, report printing, and password recovery, as depicted in Fig. 22.

(i) Data flow

Figure 22 presents the flow of events of the blue mobile application.

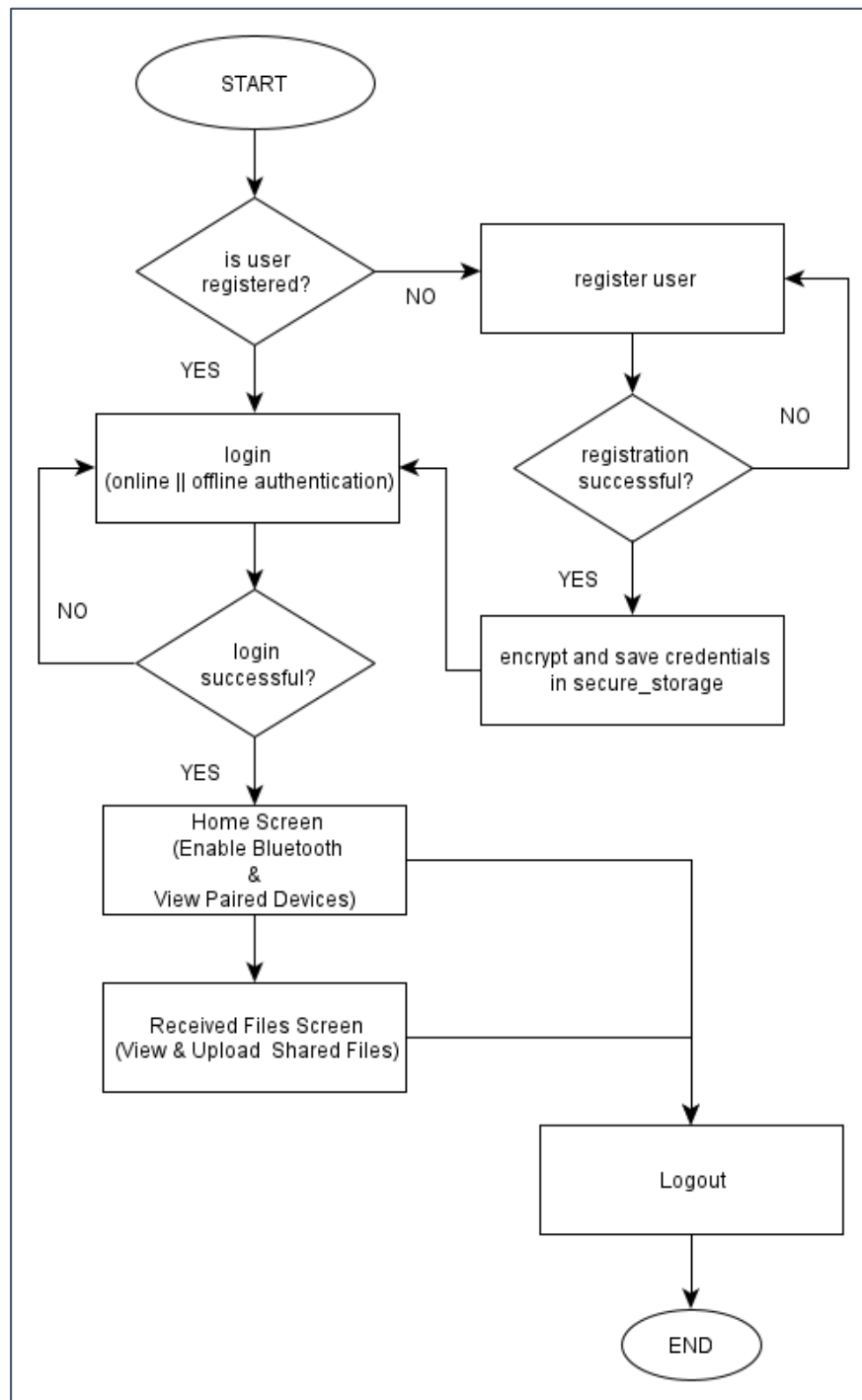


Figure 22: The Data flow of events for the 'Blue' mobile application

(ii) The Designed Mobile App

A Preview

The user is able to register, login and logout. *Firebase Authentication* service has been used for authentication. To register and login, the user's *email and password* is required. The email format

should be valid (i.e., example@gmail.com) and the password should be of at least 6 characters long (i.e., 123456).

Using `crypt` and `flutter_secure_storage`; the login credentials are encrypted and securely stored locally upon successful registration to be used during offline login. The user can enable Bluetooth, view paired devices, and open Bluetooth settings. `flutter_bluetooth_serial` library has been used for adapter status monitoring, turning adapter on and off, opening Bluetooth settings, discovering devices and listing paired devices.

The user can view shared files. Using `path_provider`, `path`, and `flutter_file_manager`, the shared files via Bluetooth are displayed to the user. The user can upload the received files to *Firestore Storage*. Upon successful upload, the instance is logged on *Firestore Database*, showing the file uploaded, time of upload and the user who has uploaded the file.

Figure 23 and 24 present the designed Blue mobile app that acquires monitoring data from the RPi controller via the Bluetooth communication technology and uploads to a firebase storage as seen in Fig. 25.

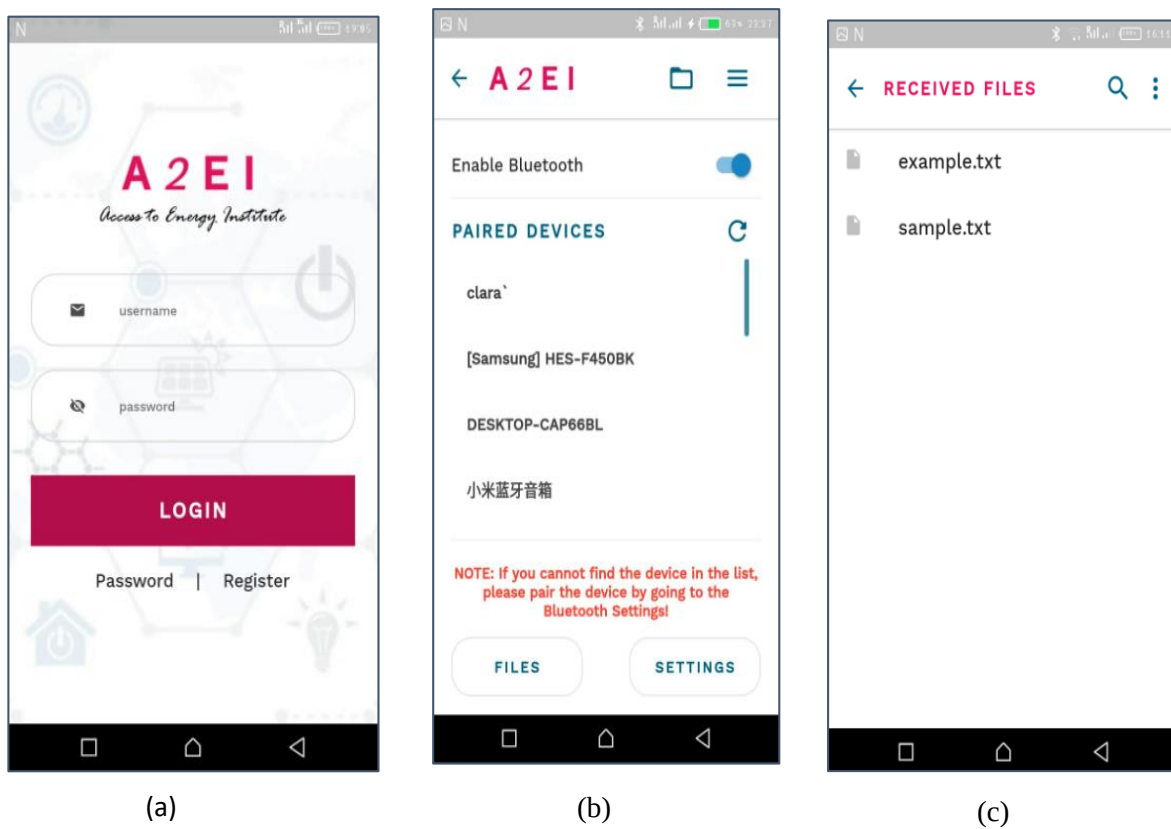
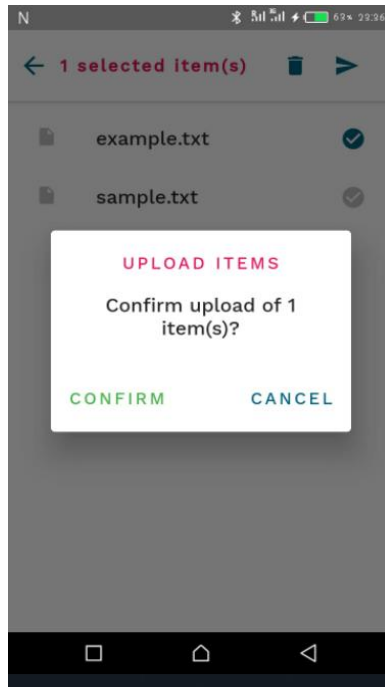
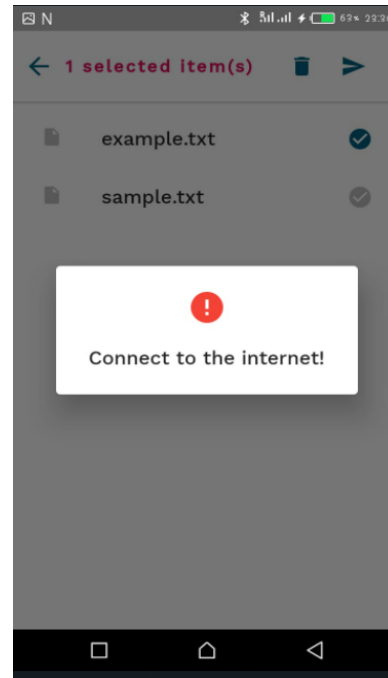


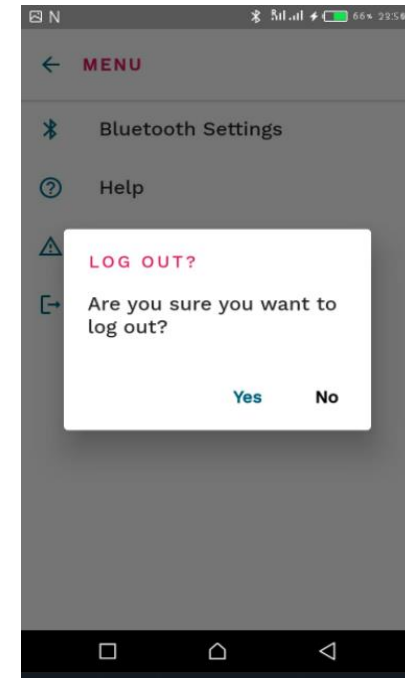
Figure 23: The Login and Registration page (a) and the Paired Control and Monitoring Station (b)



(a)



(b)



(c)

Figure 24: Snippets of the Blue app showing Uploaded Items (a), Internet connectivity status (b) and App exit query (c)

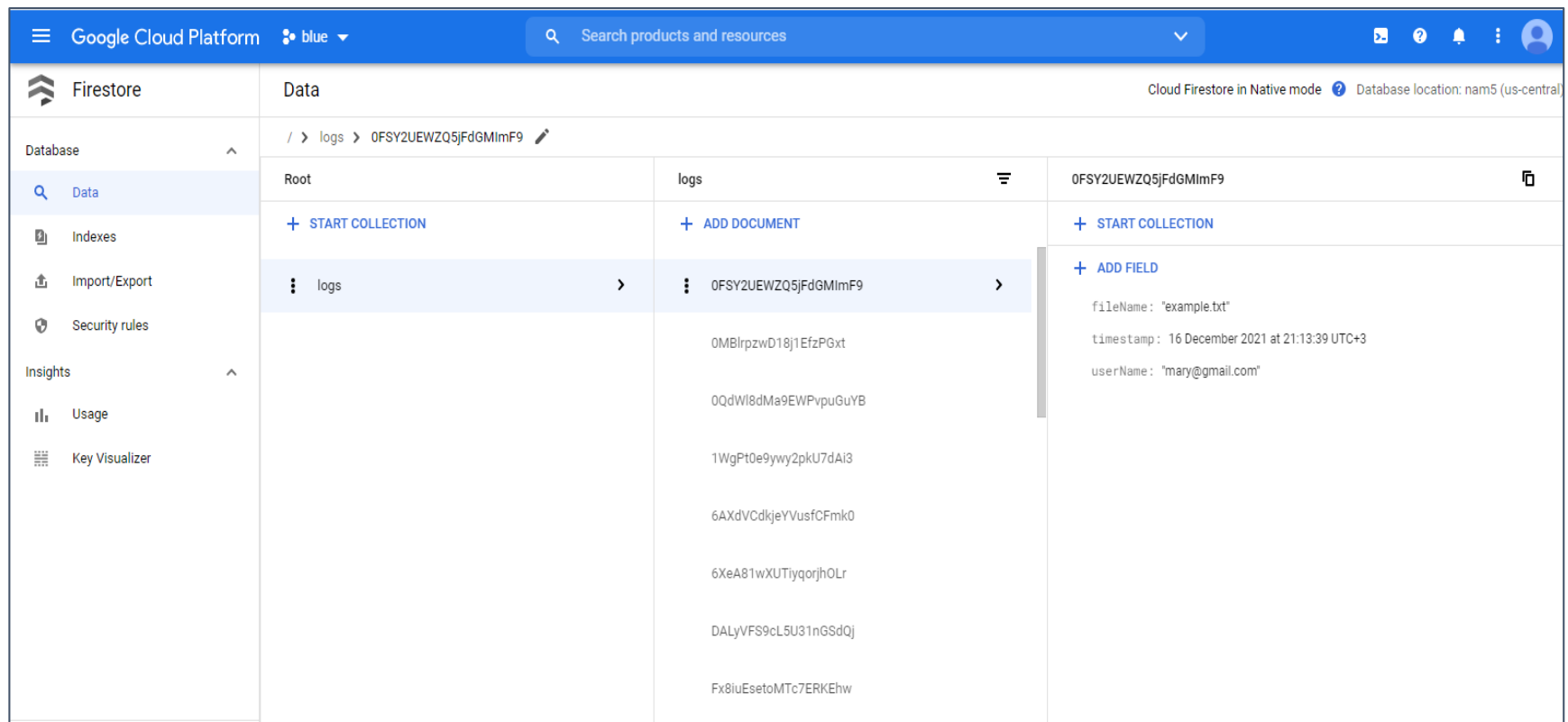


Figure 25: The Firestore Database showing files uploaded from the mobile phone client

4.1.4 The Power Unit

(i) A preview

To cater for power needs to for the oil press machine the PV solar system was implemented as per the requirements (Fig. 26). The system would produce enough power to run the machine for at least eight (8) hours in a day or continuously any time in a twenty four hours day without power shortage/outbreak.

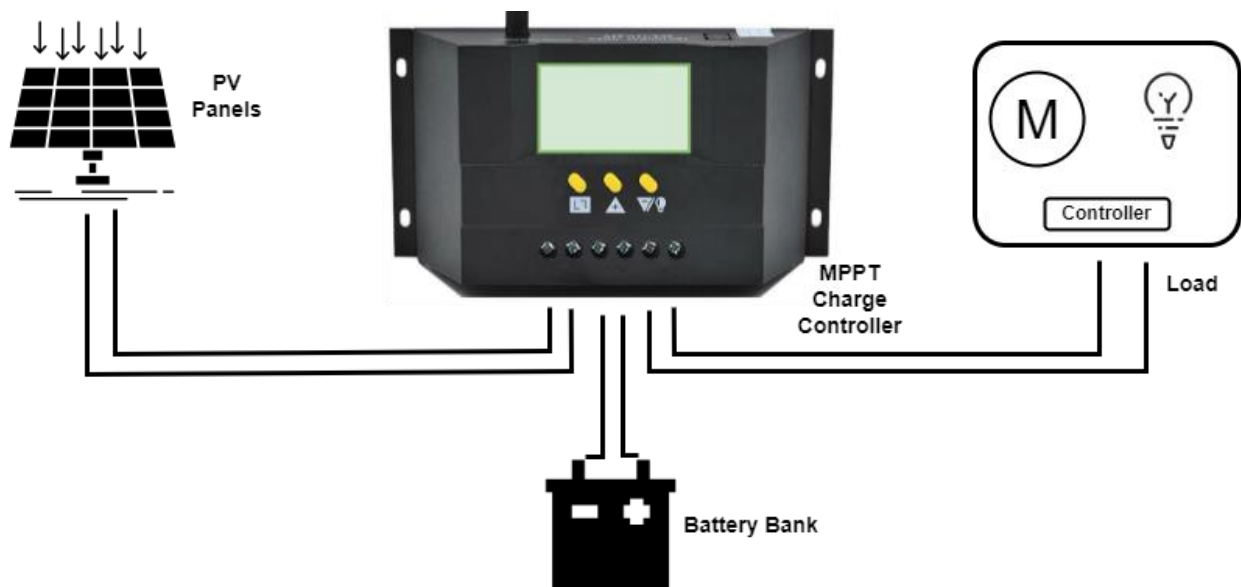


Figure 26: The Power Supply Unit

To size the system, we gathered the consumption data of all electrical components in Wattages including the motor rating, the solenoid valve ratings, the controller ratings and the DC bulbs rating (for lighting during the night works). The total consumption data (Table 4) was then used to calculate the number and size of Solar PV panels to be used, the rating of an appropriate charge controller and the size of battery bank in accordance with desired number of hours for the oil press operation.

Table 6: Total Power consumption by the system

Component	Consumption
Motor + Motor Controller	3000 W
Raspberry Pi + Arduino Microcontrollers	1.69 W
5- DC bulbs (each 3W)	15 W
Solenoid Valve	31.2 W
Total Consumption	3047.89 W

Battery Sizing

Deep cycle batteries used in our system are designed to be swiftly recharged after being depleted to a low energy level, or to cycle charged and discharged for years on end. The batteries are powerful enough to run the oil press.

Given the nominal operating voltage of the system 48V and machine operation time of at most 8 hours. Using the formula below, we arrive to the required capacity of 507.98 Ah

Equation 1: Calculation for the required system capacity

$$\text{Required Capacity (Ah)} = \frac{\text{Total Power (W)} \times \text{Total Time (h)}}{\text{Nominal Voltage (V)}}$$

With a battery Depth of Discharge (DOD) at 50%, and Battery Loss (BL) of 0.85 (as per the datasheet)

Equation 2: Calculation for the Battery Capacity Required

$$\text{Battery Capacity} = \text{Required Capacity} \times \text{DOD} \times \text{BL}$$

Hence, the battery capacity needed is 863.41Ah

Using four (4) 200Ah hour batteries sufficed the need for our system to run the number of hours that were required.

Photovoltaic Panel Sizing

The PV solar panels used had to be selected depending on the total system power consumption requirement as per the table above. So, based on global data, we had to first compute the overall Watt-peak rating required for PV modules with a panel production factor of 4.2 (Smartsolar, 2021). The number of Solar PV panels was determined by dividing the result obtained in formula 3 by the rated output Watt-peak of the PV modules supplied by the datasheet (8).

Equation 3: Calculation for Total Watt Peak

$$\text{Total Watt Peak} = \frac{\text{Total Watt Hours per Day}}{\text{Panel generation factor}}$$

Solar Charge Controller

This charge controller depended on the number of solar PV panels used (that is 8) together with specification of the Short Circuit current provided from the panel data sheet of 7.5 A, arriving to a

total solar charge controller rating of 78 A. Whence, the 150 v, 80 A MPPT charge controller was chosen for the system.

Equation 4: Calculation for the Charge Controller Rating

$$\text{Charge controller rating} = \text{No of panels} \times I_{sc} \times \text{Energy lost factor} (1.3)$$

4.1.5 Met Functional and Non-functional Requirements

A functional requirement is a set of functions that the system must be able to perform regardless of physical constraints (Alsaleh & Haron, 2016). It explains the system's input and output behavior. Non-Functional Requirements are the qualities that the system's final result must have and characterize the system's non-behavioral features (Alsaleh & Haron, 2016). In this project, there are seven system functional needs and five non-functional requirements. Table 5 and 6 described the functional and non-functional criteria that were employed in the system design and a description on their achievements.

Table 7: The Functional Requirements

Functional Requirements	Description
The system shall automate the whole motor switching process.	Once the mains have been switched on, the Motor was ready for operation.
The system shall have an option for a manual or automatic control.	With a single button switch, this was achieved.
The system shall collect motor parameters (voltage and current) and store in SD card.	Via RS232 interface, data will be streamed from the motor controller into SD card of the RPi.
The system shall automatically upload data to a client phone once Bluetooth pairing is detected.	Once the client mobile phone is in the controller vicinity, stored data in RPi was uploaded to the mobile phone Blue app.
The system shall stop operations once the batteries are below 48V.	The charger controller limited the machine use when voltage dropped below 48V.
The created app will be requested to upload the data to a cloud server once the data is on the client phone.	Once in the internet zone, the engineer/technician uploaded data to the cloud server.
The System shall be powered by solar energy and have a backup power plan.	A system with eight solar panels and four backup batteries was designed.

Table 8: Non-Functional Requirements

Non-Functional Requirements	Description
Security	The physical system was guarded by padlocks against safety and the app by login credentials
Accessibility	The system is accessible 24 hours, 7 days a week, 365 days a year
Usability, Reliability and Portability	Ease of use, easily maintained and the control box could be easily moved

4.1.6 System Modelling

Low cost, scalability, security, efficiency, availability, and ease of use were all factors in the development of a BLDC motor control and monitoring system using Arduino UNOs. The adoption of low-cost control devices for automating machine drive operations has resulted in a low-cost element. The ability of a system to store and retrieve output data/information at any time is referred to as scalability. The capacity of the system to provide confidentiality, integrity, and availability of data transmitted and stored in both the client and the database server is referred to as security. The ability of a system to provide the intended output with the least amount of power, energy, and time is referred to as efficiency. Usability refers to the system's ability to be operated easily by its users. Availability refers to the system's ability to be accessible at any time when needed.

4.1.7 Scalability

We developed a scalable system that allows operators to easily control the operations of an oil press machine by the help of buttons at the control box. For instance, when the user starts the machine, the controller can automatically enable and disable the motor and the solenoid valve and concurrently acquire the motor operation parameters and store them in an SD card. The system automatically checks if the motor controller is powered then subsequently powers the solenoid valve that protects the valve from overheating. Temperature controlled by switching on a fan cooling the MCUs in the control box by timing intervals. Generally, the same approach/design can be utilized on a plethora of agricultural machineries such as peanut shelling machines, milling machines and so forth.

4.1.8 Validation

Validation is the process of examining key indicators to ensure that the system specifications meet the intended purpose. Its purpose is to determine if the user specifications and needs have been met.

Validation is therefore a method of ensuring that specifications were properly written and that the relevant system standards were met (Kamalrudin & Sidek, 2015).

(i) Unit Test

Unit testing is a technique for ensuring the functional validity of each system module (Dybå & Dingsøyr, 2008). The following modules were evaluated for this built system: The Raspberry Pi3 Model, The Solenoid Valve and Relay Switch Modules. Each was tested for appropriate powering and the connection with RPi for functionality.

(ii) Integration Test

This is a testing approach used to determine if the modules evaluated during the unit testing stage can be combined seamlessly and correctly (Nidhra & Dondeti, 2012). Different functional elements were combined and tested to ensure that they worked effectively together. For instance, the modules such as 4-Channel Relay and DC fan were all connected to Arduino UNO in a bottom-up approach and tested for any possible errors. Different functional elements were combined and tested to ensure that they worked effectively together.

(iii) System Test

System testing is a process that involves assembling the system as a whole and running tests to ensure that it functions properly and satisfies the needs of the intended end user. During this system testing step, the overall output of all the modules was successfully tested. System testing examines the visible functional accuracy of the final product rather than the structural dimension of program codes. For motor operation data storage, the device was passively interfaced with the firebase server cloud database. Following the connection, the system was evaluated as a whole and confirmed to work as expected.

(iv) Acceptance Test

After the internal staff of the A2EI and Imara Tech Companies tested the system prototype, it was transferred to the company's technicians to collect their observations on the system usability. This activity drew a total of eight technicians. The responses of the survey were gathered on a five-point Likert scale as 5: Strongly Agree, 4: Agree, 3: Neutral, 2: Disagree, and 1: Strongly Disagree as the scale responses (Lozano *et al.*, 2008). The Instructions given to the users on how to fill the survey questionnaire is found in Appendix 4.

Using the Linkert Equation:

$$\text{Mean Score} = (5*n5 + 4*n4 + 3*n3 + 2*n2 + 1*n1) / (n5 + n4 + n3 + n2 + n1)$$

Where: n1 = number of respondents who answered “Strongly Disagree”

n2 = number of respondents who answered “Disagree”

n3 = number of respondents who answered “Neutral”

n4 = number of respondents who answered “Agree”

n5 = number of respondents who answered “Strongly Agree”

The hypothesized mean of the Likert scale employed can be compared to the mean of the replies to a specific question. The hypothesized mean, for example, is the mid-point value of 3 on a 5-point Likert scale. Comparing the mean score with 3 determines whether or not there is considerable agreement with the concept being examined (Coakes & Steed, 2001).

Table 9: Interpretation of the Likert Scale Mean

Mean Score	Mean Score (%)	Interpretation
0.01 – 1.00	0.0 – 20.0	Strongly Invalid
1.01 – 2.00	20.1 – 40.0	Invalid
2.01 – 3.00	40.1 – 60.0	Neutral
3.01 – 4.00	60.1 – 80.0	Valid
4.01 – 5.00	80.1 – 100.0	Strongly Valid

Table 10: System Acceptance testing checklist

NO	Acceptance Variable	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Mean Score	Mean Score (%)
1	The Controls of the system are easy to use and operate	0	0	0	7	1	4.12	82.4
2	The interface of the application is easy to use and interactive	0	0	0	5	3	4.38	87.6
3	I am satisfied with general performance of the system	0	0	1	6	1	4.0	80.0
4	I would recommend the use of this system to others	0	0	1	1	6	4.63	92.6

The results of the system acceptance test are displayed in Table 9. The average score for each response was greater than 3.80 (76.0%), as per Table 8, this indicated that most respondents were pleased with the general system usability, reliability and performance.

4.2 Discussion

A motor control and monitoring system's developed circuitry had combined all the machine drive building blocks, wireless communication, and programming components into a single module (Fig. 27). Furthermore, within a digital assembly environment, the device could collect significant variations in motor operation conditions, which is advantageous in short or long-time operating systems because a motor mounted at an oil press machine can have a centrally-located means of passively gathering data for reliability and efficiency monitoring. This also illustrates that the system can achieve the project's future goal of gathering data and sending out alerts in real-time.

Using the MPPT solar charge controller, the system can also notice and deactivate load power supply if vital operating parameters such as voltage and current fall below the ideal values (48V for our system). While the control system meets the design goals of a prototype, operational system, and proof of concept system, there are still a few aspects that might be modified in future iterations, such as a fully automation mode that includes material loading and offloading for oil production, the app user interface to make data more understandable to machine operators, and the addition of

sensors such as temperature and location sensors to monitor the amount of internal temperature generated in the solenoid and to determine the machine's position on a regular basis.

In general, the monitoring system has proved the feasibility of developing a low-power machine monitoring system using a single wireless device protocol. As radio, Bluetooth, and MCU technologies advance, users will be able to create smaller and less expensive wireless modes of communication that do not rely on the continuous availability of internet services. Before these systems can be widely adopted in the industry, the developing technologies must be simplified and stabilized.

Wireless monitoring systems, particularly offline 'applinks', may be a valuable tool in monitoring industrial machine drives especially in areas where there is limited or no access to internet services. Their implementation should be simple enough for any engineer to comprehend and utilize.

Converting locally manufactured agricultural machinery to more advanced machinery through the use of renewable energy and widely accessible machine drives and smart controllers has proven critical to improving the living standards of many local small scale farmers' communities. To enhance the system design, this project proposes developing a framework for the fundamental design that all other developers may use as a guide for gathering requirements and implementing a system that would serve their needs. The suggested design will assist any company or organization evaluating new technology to improve its functionality.

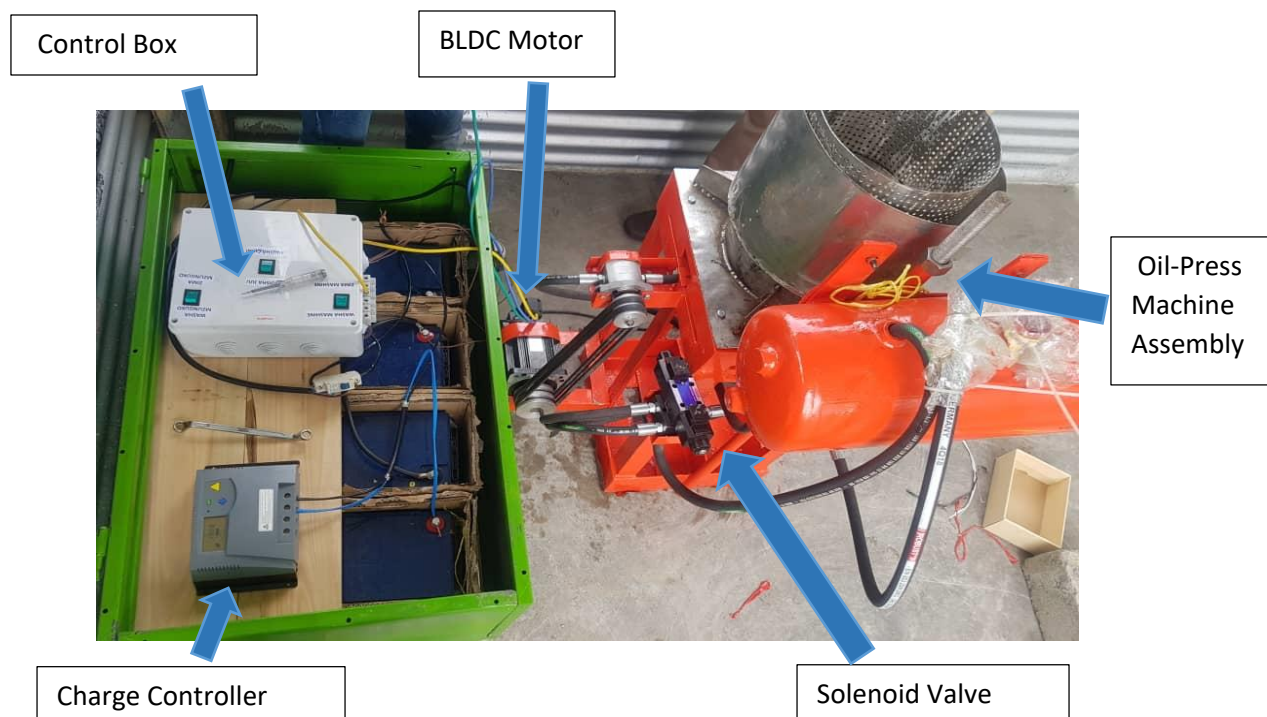


Figure 27: The final developed control and monitoring system with the oil-press machine

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The discussions on the findings reported in Chapter 4 are summarized in this chapter. As a result, the discussions provided are intended to detail and completely explain the implications and conclusion of the findings in relation to the currently available theories as well as the available literatures. A control and monitoring system for the solar powered oil-press machine was developed with a fully integrated electronic control system. Readily available devices/components were deployed in coming up with a solar-based power supply unit that would cater for the oil-press machine operation. Appropriate programming on timing was attempted to calibrate the information for the machine operation. A four-channel relay module and Arduino UNO microcontroller were successfully interfaced with the rest of the system such as the Solenoid Valve and DC buck converters to arrive to a required system. A Raspberry Pi together with the designed mobile application enabled acquisition of crucial motor operation data and store to the cloud server. The power supply unit was designed, embracing solar energy harvesting that is clean, sustainable and enabled operations of the machine for up to eight (8) hours without recharging the batteries. All observations and experimental tests prove that the developed solution suffices the oil pressing activities and control requirements for the machine.

A new kind of computing in which dynamically scaled and typically virtualized resources are supplied as a service through the Internet. The troubleshooting of the motor drive was made simple by using the Raspberry Pi to collect operating parameters such as current and voltage which are then 'app-linked' to a smartphone through Bluetooth technology and ultimately to the cloud when internet connectivity was made available. This is crucial since the machines are in areas with no Internet connectivity.

The threshold voltages and currents collected would then be used to calibrate the motors based on operating values from the previous months. Threshold levels can vary based on a variety of factors such as battery charge capacity, quality of pressed materials, and wear and tear of the machine's moving components, all of which can reduce a machine's efficiency and output.

Implementing such a system in the field could significantly boost productivity while also saving time, lowering carbon emissions, and allowing operators and small-scale farmers to participate fully in value-chain addition of the avocado produce. This in turn the project will join hands with the global world in addressing three 2030 Sustainable Development Goals in SGD1 -Poverty. SGD7-Affordable Clean Energy and SDG 9 -Industrial, Innovation and Infrastructure. Hence, the

developed Solar-Powered IoT-Based system suit the Tanzanian context addressing challenges revealed from the oil-press machine and secondary literature, and also serves as both the blue-print and framework for further work on agricultural mechanization.

5.2 Recommendations

Industrial machine drives control and monitoring involves the definition of control parameters and steps as well as measurement of different machine parameters including voltage, current, and temperature. Due to the limitation of scope, the developed system was designed to control and monitor only a few parameters. For better control of the BLDC motor-powered machine drives, it highly needs the above-mentioned parameters to be considered in a closed loop system.

This system is also limited by the power stored in the solar batteries that cannot go beyond eight (8) hours of a constant operation. To increase the operation time, larger Solar PV and battery assembly system have to be put in place as well as machine cooling system especially for the hydraulic systems as a result of solenoid valve operation.

On the other hand, due to time, parameters monitoring dashboard system would be a novel addition to the system, this category needs to be considered to ensure industrial data and efficiency of operation of the machine drives are closely monitored. To achieve a higher functioning system, we propose allocating more time and resources to such initiatives that solves the local problems with a novel solutions that are easily adaptable.

REFERENCES

- A2EI. (2019). *Access to Energy Institute* [Online]. <https://a2ei.org/news>
- Adam, J. S., & Fashina, A. A. (2019). Design of a hybrid solar photovoltaic system for Gollis University's administrative block, Somaliland. *International Journal of Physical Research*, 7(2), 37-47.
- Adi, P. D. P., & Kitagawa, A. (2019). ZigBee Radio Frequency (RF) performance on Raspberry Pi 3 for Internet of Things (IoT) based blood pressure sensors monitoring. *International Journal of Advanced Computer Science and Applications*, 10(5), 18-27.
- Ali, A. I., Partal, S. Z., Kepke, S., & Partal, H. P. (2019). *Zigbee and Lora Based Wireless Sensors For Smart Environment and Iot Applications*. In *2019 1st Global Power, Energy and Communication Conference* (pp. 19-23). <https://scholar.google.com>
- Aliexpress. (2020). *SolenoidValve*. <https://scholar.google.com>
- Alihamidi, I., Madi, A. A., & Addaim, A. (2019). *Proposed Architecture of E-health IOT*. In *2019 International Conference on Wireless Networks and Mobile Communications*. <https://scholar.google.com>
- Alsaleh, S., & Haron, H. (2016). The Most Important Functional and Non-Functional Requirements of Knowledge Sharing System at Public Academic Institutions: A Case Study. *Lecture Notes on Software Engineering*, 4(2), 157.
- Alsumady, M., Alturk, Y., Dagamesh, A., & Tantawi, M. A. (2021). Controlling of DC-DC Buck Converters Using Microcontrollers. *International Journal of Circuits*, 15, 197-202.
- Amann, S., Proksch, S., Nadi, S., & Mezini, M. (2016, March). A Study of Visual Studio Usage In Practice. In *2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering*. <https://scholar.google.com>
- Anshory, I., & Robandi, I. (2016). *Monitoring and Optimization of Speed Settings for Brushless Direct Current using Particle Swarm Optimization*. In *2016 IEEE Region 10 Symposium*. <https://scholar.google.com>
- Anshory, I., Robandi, I., Wirawan, W., & Jamaaluddin, J. (2019). Identification and implementation hybrid fuzzy logic and PID controller for speed control of BLDC motor. *Journal of Physics: Conference Series*, 1402(3), 033080.

- Arduino. (2021). *Arduino Hardware*. <https://www.arduino.cc/en/hardware>
- Aschwanden-Granfelt, S. (2017). *Process Flow Documentation: A Flowchart Guide for Micro & Small Business*. <https://scholar.google.com>
- Bagde, S., Ambade, P., Batho, M., Duragkar, P., Dahikar, P., & Ikhar, A. (2021). Internet of Things (IOT) Based Smart Switch. *Journal of IoT in Social, Mobile, Analytics, and Cloud*, 3(2), 149-162.
- Bello, L. L., & Steiner, W. (2019). A perspective on IEEE time-sensitive networking for industrial communication and automation systems. *Proceedings of the Institute of Electrical and Electronics Engineers*, 107(6), 1094-1120
- Bertoldo, S., Carosso, L., Marchetta, E., Paredes, M., & Allegretti, M. (2018). Feasibility analysis of a LoRa-based WSN using public transport. *Applied System Innovation*, 1(4), 1-12.
- Bolchini, C., Schreiber, F. A., & Tanca, L. (2007). A methodology for a very small data base design. *Information Systems*, 32(1), 61-82.
- Coakes, S., & Steed, L. (2001). *SPSS without anguish: Version 10.0 for Windows*. <https://scholar.google.com>
- Gerber, A., Craig, C., & Selvaraj, D. (2015). *Learn Android Studio: Build Android Apps Quickly and Effectively*. Apress. <https://scholar.google.com>
- Cristian, B., Constantin, O., Zoltan, E., Adina, P. V., & Florica, P. (2014). *The Control of an Industrial Process With PLC*. In *2014 International Conference on Applied and Theoretical Electricity*. <https://scholar.google.com>
- Dagne, L. (2019). *Flutter for Cross-Platform App and SDK Development*. <https://scholar.google.com>
- Deshmukh, N. S., & Bhuyar, D. L. (2018). A Smart Solar Photovoltaic Remote Monitoring and Controlling. In *2018 Second International Conference on Intelligent Computing and Control Systems*. <https://scholar.google.com>
- Dhir, S., Kumar, D., & Singh, V. B. (2019). *Success and Failure Factors that Impact on Project Implementation using Agile Software Development Methodology*. In *Software Engineering*. Springer, Singapore. <https://scholar.google.com>

- Dotoli, M., Fay, A., Miśkowicz, M., & Seatzu, C. (2019). An overview of current technologies and emerging trends in factory automation. *International Journal of Production Research*, 57(15-16), 5047-5067.
- Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, 50(9-10), 833-859.
- Erraissi, N., Raoufi, M., Aarich, N., Akhsassi, M., & Bennouna, A. (2018). Implementation of a low-cost data acquisition system for “PROPRE. MA” project. *Measurement*, 117, 21-40.
- Feng, X., Yan, F., & Liu, X. (2019). Study of wireless communication technologies on Internet of Things for precision agriculture. *Wireless Personal Communications*, 108(3), 1785-1802.
- Ferrer, J., Tan, J., Tiongson, A., & Hong, M. (2017). *Fuzzy Logic Controlled DC-DC Buck Converter*. <https://scholar.google.com>
- Firesmith, D. G. (2014). *Common System and Software Testing Pitfalls*. <https://scholar.google.com>
- Flutter. (2021). *Build Apps For Any Screen*. <https://scholar.google.com>
- Forster, Y., Hergeth, S., Naujoks, F., Krems, J., & Keinath, A. (2019). User education in automated driving: Owner’s manual and interactive tutorial support mental model formation and human-automation interaction. *Information*, 10(4), 143.
- Ganga, D., & Ramachandran, V. (2018). IoT-based vibration analytics of electrical machines. *Internet of Things Journal*, 5(6), 4538-4549.
- Gonzalez, D. J., & Asada, H. H. (2019). Hybrid open-loop closed-loop control of coupled human–robot balance during assisted stance transition with extra robotic legs. *Robotics and Automation Letters*, 4(2), 1676-1683.
- Hameed, H. S. (2018). *Brushless DC Motor Controller Design using MATLAB Applications*. In *2018 1st International Scientific Conference of Engineering Sciences-3rd Scientific Conference of Engineering Science*. <https://scholar.google.com>
- Hammar, L. (2011). *Distribution of Wind and Solar Energy Resources in Tanzania and Mozambique*. <https://scholar.google.com>
- Han, X., & Kong, X. (2010). *The Designing of Serial Communication Based On RS232*. In *2010 First ACIS International Symposium on Cryptography, and Network Security, Data Mining and Knowledge D*. <https://scholar.google.com>

- Hemalatha, M. C., Nagarajan, M. R., Suresh, P., Shankar, G. G., & Vijay, A. (2017). Brushless DC Motor Controlled by using Internet of Things. *International Journal of Science Technology & Engineering*, 3(09), 373-377.
- Hussain, S. S., Abbasi, M. D., Ameer, M., Majeed, M. K., Khawer, S. K., & Khan, M. A. (2019). *Sensor-Less Control of A BLDC Motor for the Conversion of a Conventional Petrol Vehicle Into a Solar-Electric Powered Vehicle. In 2019 13th International Conference on Mathematics, Actuarial Science, Computer Science and Statistics.* <https://scholar.google.com>
- Ibrahim, M. A., Mahmood, A. K., & Sultan, N. S. (2019). Optimal PID controller of a brushless DC motor using genetic algorithm. *International Journal of Power Electronics and Drive Systems*, 2088(8694), 8694.
- Islam, T., & Meade, N. (2013). The impact of attribute preferences on adoption timing: The case of photo-voltaic (PV) solar cells for household electricity generation. *Energy Policy*, 55, 521-530.
- Jagadesh, M., Saravanan, M., Narayanan, V., Vadhana, M. P., & Logeshwaran, K. (2019). *Monitoring System in Industry using IoT. In 2019 5th International Conference on Advanced Computing & Communication Systems.* <https://scholar.google.com>
- Johnson, B. (2012). *Professional Visual Studio 2012.* <https://scholar.google.com>
- Kamalrudin, M., & Sidek, S. (2015). A review on software requirements validation and consistency management. *International Journal of Software Engineering and its Applications*, 9(10), 39-58.
- Kaup, F., Hacker, S., Mentzendorff, E., Meurisch, C., & Hausheer, D. (2018). *The Progress of the Energy-efficiency of Single-board Computers.* <https://scholar.google.com>
- Keya. (2021). *Keya DC Motors.* <https://scholar.google.com>
- Li, Q. & Chen, Y. L. (2009). Data flow diagram. *Modeling and Analysis of Enterprise and Information Systems.* <https://scholar.google.com>
- Lonzetta, A. M., Cope, P., Campbell, J., Mohd, B. J., & Hayajneh, T. (2018). Security vulnerabilities in Bluetooth technology as used in IoT. *Journal of Sensor and Actuator Networks*, 7(3), 28.

- Loraalliance. (2020). *LoRaWAN Specifications*. <https://scholar.google.com>
- Lozano, L. M., García-Cueto, E. & Muñiz, J. (2008). Effect of the number of response categories on the reliability and validity of rating scales. *Methodology*, 4, 73-79.
- Magdum, A. D., & Agashe, A. A. (2016). *Monitoring and Controlling the Industrial Motor Parameters Remotely using LabVIEW*. In *2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology*. <https://www.google.com>
- Malche, T., Maheshwary, P., & Kumar, R. (2019). Environmental monitoring system for smart city based on secure Internet of Things (IoT) architecture. *Wireless Personal Communications*, 107(4), 2143-2172.
- Mbabaye, G., Minja, R., Mtalo, F., Legonda, I., & Mkongo, G. (2018). Fluoride occurrence in domestic water supply sources in Tanzania: A case of Meru district Arusha region. *Tanzania Journal of Science*, 44(3), 72-92.
- Mohtasham, J. (2015). Renewable energies. *Energy Procedia*, 74, 1289-1297.
- Napoli, M. L. (2019). *Beginning Flutter: A Hands On Guide To App Development*. <https://scholar.google.com>
- Nidhra, S., & Dondeti, J. (2012). Black box and white box testing techniques-a literature review. *International Journal of Embedded Systems and Applications*, 2(2), 29-50.
- Nussey, J. (2013). *Arduino for Dummies*. <https://scholar.google.com>
- Prakash, S. (2020). *Zigbee based Wireless Sensor Network Architecture for Agriculture Applications*. In *2020 Third International Conference on Smart Systems and Inventive Technology*. <https://scholar.google.com>
- Raspberrypi. (2021). *Raspberry Pi 3 Model 3*. [https:// www. raspberrypi. com/ products/raspberry-pi-3-model-b/](https://www.raspberrypi.com/products/raspberry-pi-3-model-b/)
- Rehman, S., & El-Amin, I. (2012). Performance evaluation of an off-grid photovoltaic system in Saudi Arabia. *Energy*, 46, 451-458.
- Robles Algarín, C., Taborda Giraldo, J. & Rodriguez Alvarez, O. (2017). Fuzzy logic based MPPT controller for a PV system. *Energies*, 10, 2036.

- San Cheong, P., Bergs, J., Hawinkel, C., & Famaey, J. (2017). *Comparison of LoRaWAN Classes and their Power Consumption*. In *2017 IEEE Symposium on Communications and Vehicular Technology*. <https://www.google.com>
- Schmidt, R. A., Lee, T. D., Winstein, C., Wulf, G., & Zelaznik, H. N. (2018). *Motor Control and Learning: A behavioral Emphasis*. <https://www.google.com>
- Schön, E. M., Thomaschewski, J., & Escalona, M. J. (2017). Agile Requirements Engineering: A systematic literature review. *Computer Standards & Interfaces*, 49, 79-91.
- Senthilkumar, S., Prabakaran, C., Saranya, M., Keerthana, V., Kayalvizhi, S., & Shinthana, S. (2020). M2M Based Industrial Automation using Advanced Microcontroller and IoT with Auto Sensing Technology. *International Research Journal of Innovations in Engineering and Technology*, 4, 101.
- Shahinzadeh, H., Moradi, J., Gharehpetian, G. B., Nafisi, H., & Abedi, M. (2019). *IoT Architecture for Smart Grids*. In *2019 International Conference on Protection and Automation of Power System*. <https://www.google.com>
- Shifat, T. A., & Jang-Wook, H. (2020). Remaining useful life estimation of BLDC motor considering voltage degradation and attention-based neural network. *Access*, 8, 168414-168428.
- Smartsolar. (2021). *Solar Power Potential Tanzania*. <https://scholar.google.com>
- Smyth, N. (2017). *Android Studio 2.3 Development Essentials-Android 7 Edition*. <https://scholar.google.com>
- Soeteman-Hernandez, L. G., Apostolova, M. D., Bekker, C., Dekkers, S., Grafström, R. C., Groenewold, M., Handzhiyski, Y., Herbeck-Engel, P., Hoehener, K., Karagkiozaki, V., & Kelly, S. (2019). Safe innovation approach: Towards an agile system for dealing with innovations. *Materials Today Communications*, 20, 100548.
- Somwanshi, D., Bundeale, M., Kumar, G., & Parashar, G. (2019). Comparison of fuzzy-PID and PID controller for speed control of DC motor using LabVIEW. *Procedia Computer Science*, 152, 252-260.
- Sulír, M., Bačíková, M., Chodarev, S., & Porubán, J. (2018). *Visual Augmentation of Source Code Editors*. <https://scholar.google.com>

- Tavner, P., Ran, L., Penman, J., & Sedding, H. (2008). *Condition Monitoring of Rotating Electrical Machines*. <https://scholar.google.com>
- Terán, M., Aranda, J., Carrillo, H., Mendez, D., & Parra, C. (2017). *IoT-based System for Indoor Location using Bluetooth Low Energy*. In *2017 IEEE Colombian Conference on Communications and Computing*. <https://scholar.google.com>
- Tsai, K. L., Leu, F. Y., You, I., Chang, S. W., Hu, S. J., & Park, H. (2019). Low-power AES data encryption architecture for a LoRaWAN. *Access*, 7, 146348-146357.
- Upton, E., & Halfacree, G. (2014). *Raspberry Pi user Guide*. <https://scholar.google.com>
- Vaishakh, J. K., & Vanitha, V. (2019). Remote monitoring of BLDC motor using LabVIEW and zigbee. *Conference Series: Materials Science and Engineering*, 561(1), 012118.
- VanCasteren, W. (2017). The Waterfall Model and the Agile Methodologies: A comparison by project characteristics. *Research Gate*, 2, 1-6.
- Van Rossum, G., & Drake F. L. (2014). *The Python Language Reference*. <https://www.google.com>
- Wu, T. Y., Wang, B., Lee, J. Y., Shen, H. P., Wu, Y. C., Chen, Y. A., Ku, P. S., Hsu, M. W., Lin, Y. C., & Chen, M. Y. (2017). *CircuitSense: Automatic Sensing of Physical Circuits and Generation of Virtual Circuits to Support Software Tools*. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. <https://www.google.com>
- Zammetti, F. (2019). *Practical Flutter*. <https://www.google.com>
- Zeadally, S., Siddiqui, F., & Baig, Z. (2019). 25 years of Bluetooth technology. *Future Internet*, 11(9), 1-24.
- Zhang, W., Zhang, F., Zhang, J., Zhang, J., & Zhang, J. (2018). Based on Brushless DC Motor of Fuzzy and PID Control System. *Conference Series: Materials Science and Engineering*, 452(4), 042075.

APPENDICES

Appendix 1: Python Code for collecting Data on USB Port from the RaspberryPi

```
import os
import time
import shutil
import numpy as np
import pandas as pd
import serial
import subprocess

from time import sleep
from datetime import datetime

log = open("/home/pi/cumsum_data.csv", "a")
if os.stat("/home/pi/cumsum_data.csv").st_size == 0:

log.write("timestamp,current,motor_voltage,internal_voltage,battery_voltage,power,energy,cumulative_energy\n")

file = open("/mnt/flashdisk/cumsum_data.csv", "a")
if os.stat("/mnt/flashdisk/cumsum_data.csv").st_size == 0:

file.write("timestamp,current,motor_voltage,internal_voltage,battery_voltage,power,energy,cumulative_energy\n")

ser = serial.Serial(
    port='/dev/ttyUSB0', #Replace ttyS0 with ttyAM0 for Pi1,Pi2,Pi0
    baudrate = 115200,
    parity=serial.PARITY_NONE,
    stopbits=serial.STOPBITS_ONE,
    bytesize=serial.EIGHTBITS,
    timeout=1
)

while 1:
    now = datetime.now()
    now = now.replace(microsecond=0)
    print("timestamp: ", now, end = ' ')
    ser.write(str.encode("?A\r"))
    #time.sleep(1)
```

```

i = ser.readline()
if i == b" or i == b'}' or i == b'W' or i == b'\xbf' or i == b'\xbb' or i == b '[' or i == b']':
    i = 0;
    print ("Current: ", i, end = ' ')
else:
    i = str(i)
    #print("wrong current",i)
    i = int("".join(filter(str.isdigit, i))) * 0.1
    i = round(i, 2)
    print ("Current: ",i, end = ' ')

ser.write(str.encode("?V [2]\r"))
#time.sleep(1)
v =ser.readline()
if v == b" or v == b'}' or v == b'W' or v == b'\xbf' or v == b'\xbb' or v == b '[' or v == b']':
    v = 0;
    print ("motor_voltage: ", v, end = ' ')
else:
    v = str(v)
    #print("wrong voltage",v)
    v = int("".join(filter(str.isdigit, v)))
    v = str(v)
    v = v[1:]
    v = int(v) * 0.1
    v = round(v,2)
    print("motor_voltage: ", v, end = ' ')

ser.write(str.encode("?V [1]\r"))
#time.sleep(1)
vi =ser.readline()
if vi == b" or vi == b'}' or vi == b'W' or vi == b'\xbf' or vi == b'\xbb' or vi == b '[' or vi == b']':
    vi = 0;
    print ("internal_voltage: ", vi, end = ' ')
else:
    vi = str(vi)
    #print("wrong vi",vi)
    vi = int("".join(filter(str.isdigit, vi)))
    vi = str(vi)

```

```

vi = vi[1:]
vi = int(vi) * 0.1
vi = round(vi, 2)
print(",internal_voltage: ", vi, end = ' ')

vb = vi + v # total battery voltage
print(",battery_voltage:", vb, end = ' ')

p = i * v
p = round(p, 4)
print ("power: ",p, end = ' ')
e = (p * 60) / (3600 * 1000)
e = round(e, 4)
#e = (p * 300) / (1000 * 3600) # five minutes resolution
print ("energy: ", e, end = ' ')

#cume = e # initially uncomment this to create csv snf run twice
df = pd.read_csv('/home/pi/cumsum_data.csv')
#new = list(accumulate(e))
#cume = df.energy
cume = df.at[df.index[-1], 'cumulative_energy'] + e # save to pi folder
print(",cum_energy", round(cume,6))

dl = pd.read_csv('/mnt/flashdisk/cumsum_data.csv')
#new = list(accumulate(e))
#cume = df.energy
cumel = dl.at[dl.index[-1], 'cumulative_energy'] + e # save to flash

log.write(format(now.strftime("%Y-%m-%d %H:%M:%S")+","+str(i)+","+str(v)+
                "+str(vi)+","+str(vb)+","+str(p)+","+str(e)+","+str(cume)+"\n"))
file.write(format(now.strftime("%Y-%m-%d %H:%M:%S")+","+str(i)+","+str(v)+
                "+str(vi)+","+str(vb)+","+str(p)+","+str(e)+","+str(cumel)+"\n"))

log.flush()
time.sleep(1)
file.flush()

```

```
#time.sleep(3) #57 is one minute
```

```
#shutil.copyfile('/home/pi/cumsum_data.csv', '/mnt/flashdisk/', follow_symlinks=True)
```


Appendix 2: Control codes for the Arduino UNO

```
/*
```

```
Oil Press Control Code
```

```
Controlling the solenoid valve using relay + millis function
```

```
****THIS IS FOR AUTOMATING THE MOTOR ON AND OFF****
```

```
*****BY GILBERT MINJA*****
```

```
*/
```

```
int potPin = A1; // select the input pin for the MOTOR TRIGGER
```

```
float val = 0.0; // variable to store the values
```

```
int relayPin = 22;
```

```
int relayPin2 = 11;
```

```
const unsigned long eventInterval = 7500;
```

```
const unsigned long eventInterval2 = 0;
```

```
const unsigned long eventInterval3 = 10000; // Motor ON interval
```

```
const unsigned long eventInterval4 = 20000; // Motor OFF interval
```

```
unsigned long previousTime=0;
```

```
//unsigned long previousTime=0;
```

```
boolean x = true;
```

```
boolean y = true;
```

```
boolean z = true;
```

```
float temp;
```

```
void setup() {
```

```
pinMode(potPin, INPUT);
```

```
pinMode(relayPin, OUTPUT);
```

```
pinMode(relayPin2, OUTPUT);
```

```
//digitalWrite(relayPin, LOW);
```

```
Serial.begin(9600);
```

```
}
```

```
void loop() {
```

```
    unsigned long currentTime = millis(); // millis function used for delays in switching On and OFF the motor and Solenoid
```

```
/**read the value from the sensor and convert it to range of 0 - 5 volts**/
```

```
val = ((float)analogRead(potPin)/1023)*5;
```

```
Serial.print("MCU Voltage: ");
```

```
Serial.println(val);
```

```
if (val < 3.9 && x == true) // Holding the solenoid OFF when Motor is not enabled
```

```
{
```

```
if (currentTime - previousTime >= eventInterval2){
```

```
    digitalWrite(relayPin, HIGH); //NO used in the relay
```

```
    previousTime = currentTime;
```

```
}
```

```
}
```

```
if (val >= 4.0 && y == true){ //Switching relays for both Motor and Relay and toggling operation for a Motor state (ON--> OFF)
```

```
    digitalWrite(relayPin, LOW);
```

```
    digitalWrite (relayPin2, LOW);
```

```
    x = false;
```

```
    z = false;
```

```
}
```

```
if (x == false){ //Switching off the Motor after 60seconds
```

```
if (currentTime - previousTime >= eventInterval3) {
```

```
    digitalWrite (relayPin, HIGH);
```

```
    previousTime = currentTime;
```

```
    y = false;
```

```
    x = true;
```

```
}
```

```
}
```

```
if ( y == false )// Switching ON the Motor after 90seconds have elapsed
```

```
{
```

```
if (currentTime - previousTime >= eventInterval4)
```

```
{
```

```
    digitalWrite( relayPin, LOW);
```

```
    previousTime = currentTime;
```

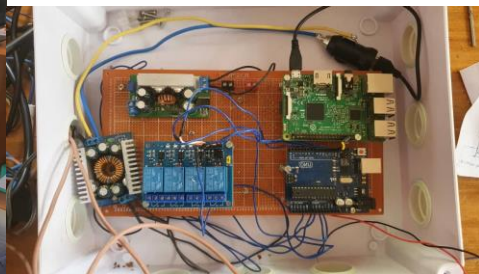
```
    y = true;
```

```
}
```

```
}
```

```
if (val < 3.9 && z == false) // Switching off relay after 7.5 seconds when Enable switch is disabled
{
if (currentTime - previousTime >= eventInterval){
digitalWrite(relayPin, HIGH); //NO used in the relay
digitalWrite(relayPin2, HIGH);
previousTime = currentTime;
}
}
}
```

Appendix 3: System Development in Pictures



Appendix 4 Questionnaire for Validating the System

Introduction

The primary goal of this survey is to obtain feedback on usability of the developed control and monitoring system for a solar powered oil press machine.

Respondents:

Engineers and Technicians at A2EI

Please tick the box corresponding to your views on the developed system

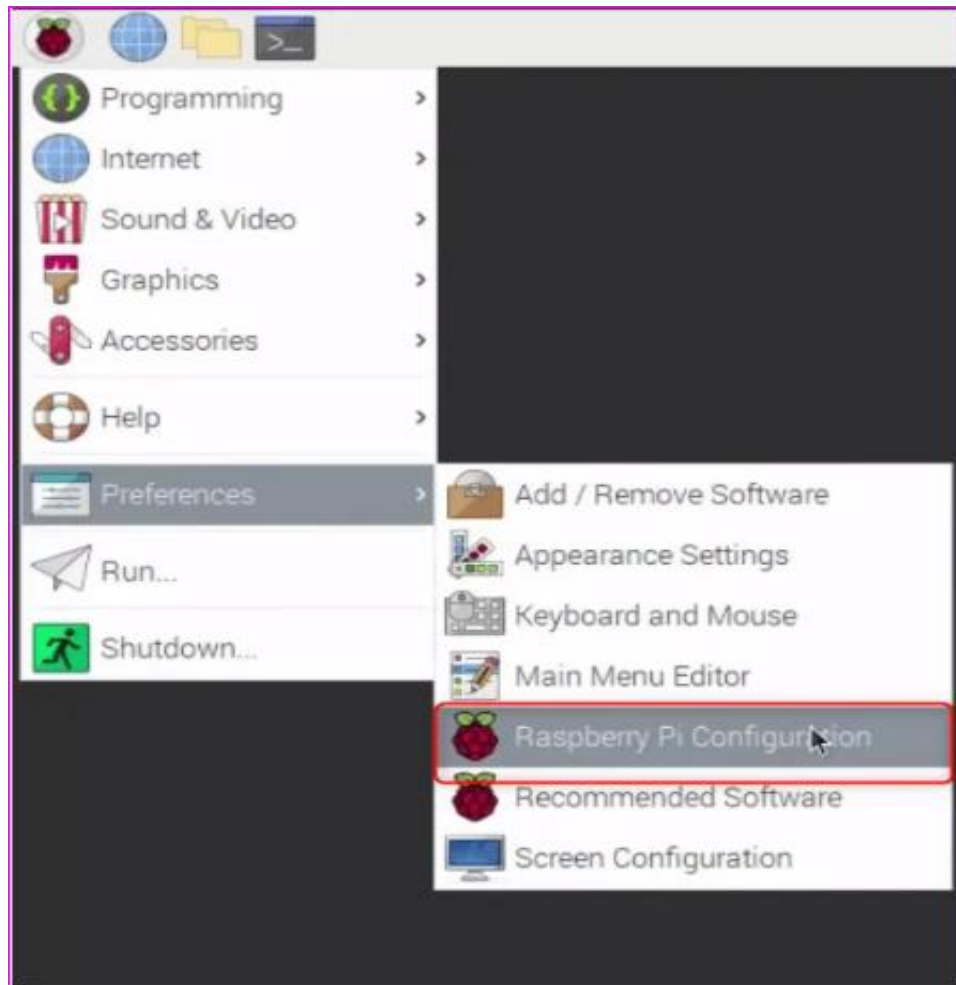
Table 11: System acceptance testing survey questionnaire

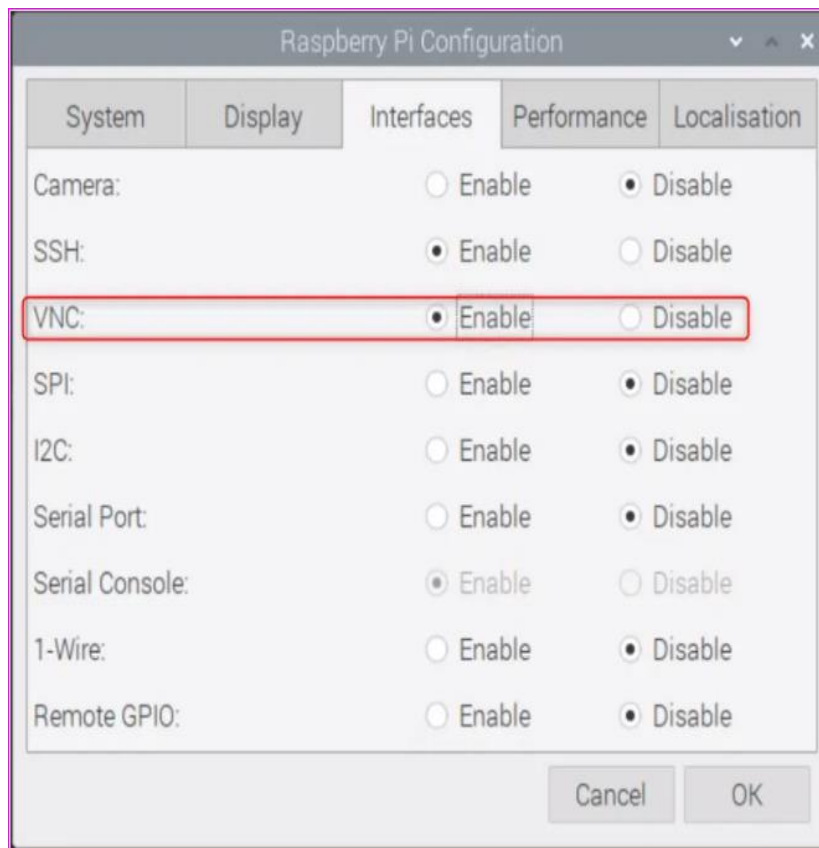
NO	Acceptance Variable	Strongly Disagree	Disagree	Neutral	Agree	Strongly Agree	Mean Score
1	The Controls of the system are easy to use and operate						
2	The interface of the application are easy to use and interactive						
3	I am satisfied with general performance of the system						
4	I would recommend the use of this system to others						

Appendix 5 Setting up VNC for Raspberry Pi wireless connectivity to PC

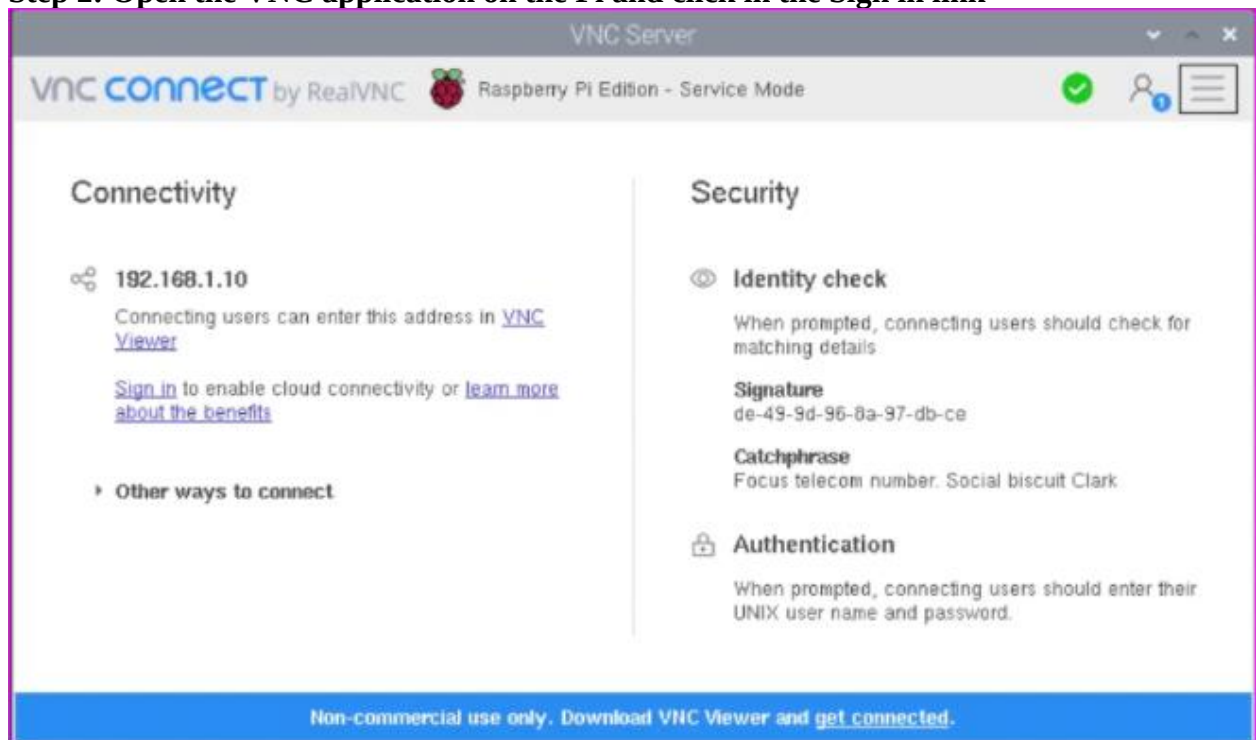
Step 1:

Make sure VNC is enabled on the Raspberry Pi. to ensure that, go to main menu -> preferences -> Raspberry Pi configuration as shone here. And, go to interfaces tab there you can enable the VNC and hit OK.

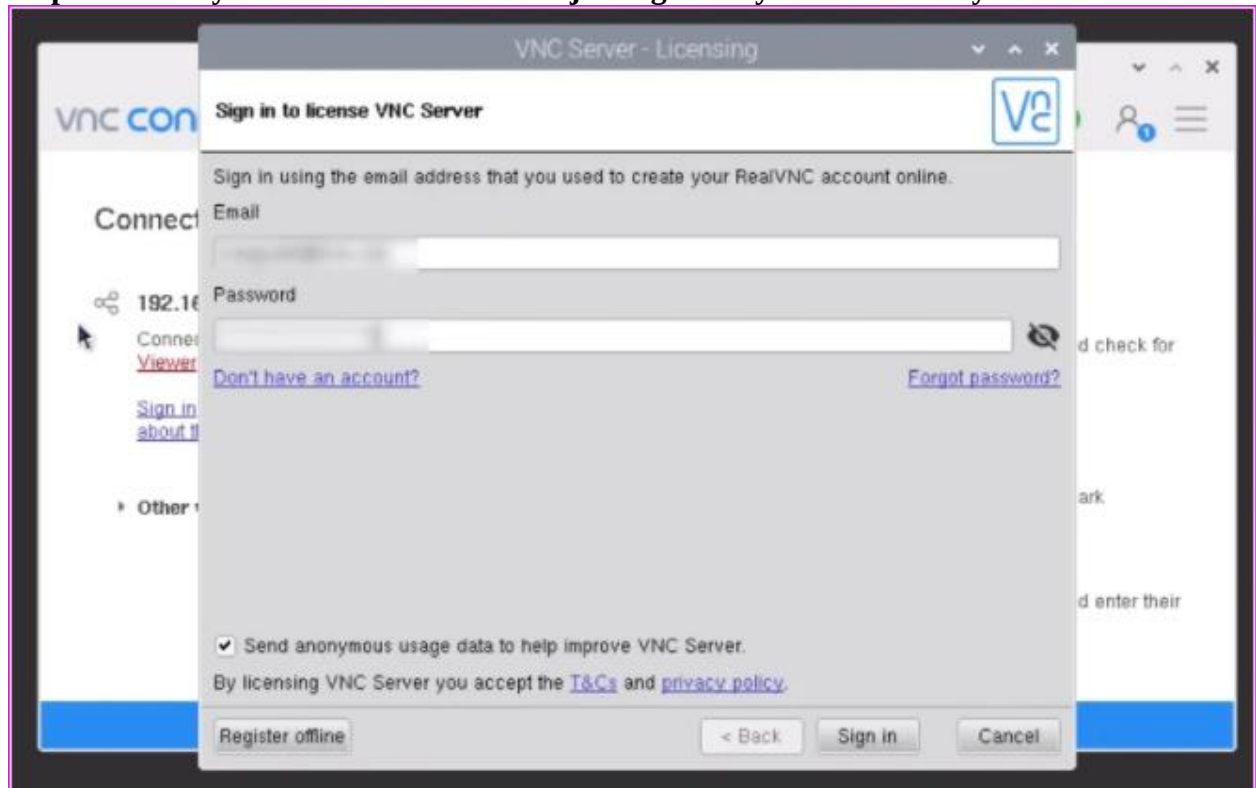




Step 2: Open the VNC application on the Pi and click in the Sign in link



Step 3: Create your own account here or just sign in if you have already



Now using the IP Address assigned to your Raspberry Pi, you can wirelessly get connected using your PC without the use of external screen.

Appendix 6: Blue Android Application Manifest

```
manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.blue">

    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.MANAGE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:label="blue"
        android:icon="@mipmap/launcher_icon"
        android:usesCleartextTraffic="true"
        android:requestLegacyExternalStorage="true">

        <activity
            android:name=".MainActivity"
            android:launchMode="singleTop"
            android:theme="@style/LaunchTheme"
            android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestScreenSize|locale|layoutDirection|fontScale|screenLayout|density|uiMode"
            android:hardwareAccelerated="true"
            android:windowSoftInputMode="adjustResize">
                <!-- Specifies an Android theme to apply to this Activity as soon as
                    the Android process has started. This theme is visible to the
                    user
                    while the Flutter UI initializes. After that, this theme
                    continues
                    to determine the Window background behind the Flutter UI. -->
                <meta-data
                    android:name="io.flutter.embedding.android.NormalTheme"
                    android:resource="@style/NormalTheme"
                />
                <!-- Displays an Android View that continues showing the launch
                    screen
                    Drawable until Flutter paints its first frame, then this splash
                    screen fades out. A splash screen is useful to avoid any visual
                    gap between the end of Android's launch screen and the painting
                    of
                    Flutter's first frame. -->
                <meta-data
                    android:name="io.flutter.embedding.android.SplashScreenDrawable"
                    android:resource="@drawable/launch_background"
                />
                <intent-filter>
                    <action android:name="android.intent.action.MAIN"/>
                    <category android:name="android.intent.category.LAUNCHER"/>
                </intent-filter>
            </activity>
            <!-- Don't delete the meta-data below.
```

```
    This is used by the Flutter tool to generate  
GeneratedPluginRegistrant.java -->  
    <meta-data  
        android:name="flutterEmbedding"  
        android:value="2" />  
    </application>  
</manifest>
```

POSTER PRESENTATION