DEVELOPING A CLOUD BASED INTERNET OF THINGS (IoT) GREENHOUSE MONITORING SYSTEM

By

PATISWA CHWAYITA MPANGELE

(21649816)

Submitted in fulfilment of the requirements for the

MASTER OF INFORMATION AND COMMUNICATIONS TECHNOLOGY DEGREE

in the

DEPARTMENT OF INFORMATION TECHNOLOGY

in the

FACULTY OF ACCOUNTING AND INFORMATICS

at

DURBAN UNIVERSITY OF TECHNOLOGY

Supervisors:

Dr P Mtshali PhD

Dr A Singh DTech

DECLARATION

I, Patiswa Chwayita Mpangele, declare that this dissertation is my own work. This work has not been submitted in any form for another degree at any university or institution of higher learning. All information cited from published or unpublished works have been acknowledged.

Patiswa Chwayita Mpangele

Student Signature

DEDICATION

I dedicate this dissertation to my beloved husband, Dr Lamla Thungatha (PhD), who has given me, and our daughter the unconditional love and friendship. I also dedicate this dissertation to my daughter, Eminathi Indiphile Thungatha, who is so precious to us, and I finally dedicate this dissertation to both the Mbambeli Mpangele and Mthuthuzeli Thungatha families. Without their enduring support and encouragement, the writing of this dissertation would not be a success.

ACKNOWLEDGEMENTS

First and foremost, I thank the Almighty God who makes everything possible for me.

I would like to express my sincere appreciation to my supervisors for their assistance, support and mentorship which helped me through this work; without their constructive advice and continuous motivation, it would not have been possible to complete this work.

Special thanks to Lubabalo Mbangata for his assistance to make this study a success and to all my colleagues. My huge appreciation to my parents, Mrs Nozuko Mpangele and Mr Mbambeli Mpangele, for their prayers and immense support during tough times.

I acknowledge all my siblings, colleagues, friends and loved ones, too numerous to mention, who have contributed in one way or another towards the successful completion of this study. To all of you, I offer my grateful thanks.

Finally, special thanks to Centre of Praise Ministries for taking care of my spiritual well-being.

" Never worry about anything. Instead, in every situation let your petitions be made known to God through prayers and requests, with thanksgiving" **Philippians 4:6**

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATION

WSN	Wireless Sensor Network
CAN	Controller Area Network
SMS	Short Message Service
IoT	Internet of Things
RFID	Radio Frequency Identification
NFC	Near Field Communication
UWP	Universal Windows Platform
DSRM	Design Science Research Methodology
SOA	Service Oriented Architecture
IT	Information Technology

ABSTRACT

The introduction of the greenhouse reduces the negative impact of difficult to control outdoor factors on crop production. The greenhouses offer a better environment in which to grow high quality agricultural products in large quantities, but in a small surface area. Factors including temperature, humidity and soil moisture each have a different impact on the quality of the crop. These factors contribute to how crops develop either directly or indirectly. Poor environmental conditions can damage crops or increase the likelihood of crop diseases. Hence, this study set out to develop a greenhouse monitoring system to assist farmers to monitor soil moisture, humidity and temperature. Internet of Things (IoT) and cloud-computing are the major components underlying the monitoring system.

The reviewed literature reveals that there are minimal studies that have been conducted for the convergence of cloud computing and IoT in agriculture, particularly in greenhouse farming. Despite the researcher's best efforts, literature revealed very few studies that have successfully implemented these emerging technologies in innovative ways in the sector of agriculture. Studies are even sparser in relation to greenhouse applications.

This study aimed to develop a cloud-based Internet of Things (IoT) greenhouse monitoring system that is cost effective and easily accessible and provides unlimited data storage to farmers for analysis. This aim is achieved by three key objectives, that is: by critically reviewing existing literature of IT based greenhouse monitoring systems supporting crop production; by developing a wireless cloud-based IoT greenhouse monitoring system; and by evaluating the performance of the developed system.

The design science research methodology (DSRM) is applied as an overall methodology to conduct this study. DSRM is a popular methodology when research aims to develop innovative artefacts. This instance of DSRM followed five steps: awareness; suggestion; development; evaluation; and conclusion to design and implement and evaluate the artefact. DSRM allowed for a robust development process which promoted higher quality in monitoring the three main factors of the greenhouse environment, which are temperature, humidity and soil moisture.

The results of the evaluation of the developed system indicates an efficient, cost-effective and easy to access monitoring system for three key environmental factors, namely, temperature, humidity, and soil moisture.

CHAPTER 1: INTRODUCTION

1.1 Context of the study

The increased concern over food security and demand for high-quality fresh produce are some of the issues that have afforded the speedy initiation of the greenhouse in the agriculture industry. The rapidly growing population (Chawla 2016) and exponential demand on fresh produce also influence this. Greenhouse production is a rapidly growing sector globally (Hu *et al.* 2017; Ghoulem *et al.* 2019). Greenhouses isolate crops from the environment, which permits the production of crops that were not feasible growing at that exact site (Ramyateja and Kishore 2016). The greenhouse enclosure allows the manipulation of the crop environment. Castilla (2013:1) highlights that, greenhouse crops "perform at their best while in the most suitable environment through maintaining the temperature, the soil moisture, and the humidity at the optimal level for photosynthesis".

A greenhouse environment requires precise continuous monitoring of these parameters in order to avoid hostile exposure to unhealthy ambient surroundings. Computerized environment monitoring systems are conducive solutions in granting continuous monitoring. Song (2010) has proposed different systems for monitoring greenhouse environments, for instance, the intelligent greenhouses based on the Controller Area Network (CAN) bus by ((Pengzhan and Baifen (2010)), RS485 bus by. (Yulong and Jiaqiang (2011))), and the Wireless Sensor Network (WSN) greenhouse monitoring system. These precedent works perform an analysis of several climatological variables by using wired and wireless networks for data transportation.

Presently, there are many available greenhouse environment-monitoring systems in the market, some providing abundant or as little of monitoring as may be feasible. In the past days of the greenhouse, it was enough to have data sent to a mobile device or webpage for the farmer to monitor the current situation; farmers never bothered to keep the data for future reference. In the modern day, keeping data for future reference is vital since it can help farmers with the prediction of the final yield based on the previous turn out conditions' yield. They also need to be alert about which conditions are conducive for the crop production based on the results obtained in previous seasons. Cloud-based IoT systems can provide effective greenhouse monitoring.

Dan *et al.* (2015:488) state that, “IoT communication is used to collect the measurements and to communicate between the centralized control unit and the actuators located at the different parts of the greenhouse compared to the cabled systems. The installation of IoT is fast, cheap and easy”. Moreover, it is easy to relocate the measurement points when needed by just moving sensor nodes from one location to another, within a communication range of the coordinator device (Ravi Kishore Kodali, Jain and Karagwal 2016). IoT means connecting, transferring data from one device to another via the Internet. The IoT controls appliances anytime and anywhere while cloud computing provides storage and computing resources to implement a web application (Jie *et al.* 2013). Cloud computing practiced in greenhouse monitoring will improve the productivity of greenhouse production, save water and support conventional agriculture move towards precision farming. According to JoSEP *et al.* (2010:51), cloud computing is defined as “both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services”.

1.2 Motivation of the study

The subjects of interest in this study is to monitor soil moisture, temperature and humidity with soil moisture as the main factor of this study. The cloud based IoT greenhouse monitoring system is based on a wireless network for data collection and storage. Soil moisture directly relates to water. Hence, the monitoring of soil moisture can provide positive results to water usage. In recent years, the shortage of water has been getting worse because of the impact of the climate change. As for Department of Environmental Affairs (2011) “South Africa is a water-scarce country (annual freshwater availability is less than 1 700 m3 per capita), with limited average rainfall of about 450 mm/year and unevenly distributed water resources”.

The motivation for doing this research on developing a cloud-based IoT greenhouse monitoring system is encouraged by the fact that human beings are still more involved in the currently available greenhouse systems. As much as there are computerized systems, most of them require manual control. To be effective, they need an observer available onsite all the time especial during the day. Hence, the continuous human visitation to farm sites is required, causing a lot of burden to the farmers. In recent years, the shortage of farming labor force has been getting worse because of the falling interest in youth and the aging farmers (Kitamura and Oka 2005).

Greenhouse farming is continuously developing, making a good use of machine-to-machine technology, by deploying different sensor nodes that act as human observers. These sensor nodes swap human observations and measure the environmental parameters and conditions within the greenhouse in order to obtain measurements that are more accurate with the anticipated sampling frequency. Furthermore, they permit collaboration and processing of the gathered measurements in order to come to conclusion and determine the actual status of the greenhouses (Ibrahim *et al.* 2019)

The IoT is marked as the third wave of the world's information industry following computer and the Internet. The IoT has accomplished human to machine amalgamation, improved the physical world real-time control and accurate management degree of intelligence, and attained optical allocation of resources and scientific intelligence decision, and meticulous management (Guo and Zhong 2015). The utilization of IoT in greenhouse horticulture contributes to its growth. Since, the data gathered from the sensors, within and outdoors of the greenhouse, can be analyzed and backed-up on the central cloud data storage for long term referencing and data mining as well as stored on cloud edge points for faster processing (Danita *et al.* 2018).

1.3 Research Problem

The closed environment farming method, like a greenhouse, has its own unique requirements when compared to outdoor farming environments (Salleh *et al.* 2013). The precise continuous greenhouse monitoring system is required to control various environmental parameters (Ahonen, Virrankoski and Elmusrati 2008; Shaker and Imran 2013). However, according to Matijevics and Simon (2010), most available systems are not conducive to the greenhouse environment since they use a *wired* network. These systems are not easy to install and extend; maintenance costs are high and there is limited data storage space. Song (2010) and Dhumal and Chitode (2013) continue to note the issue of unfavorable wired systems in a greenhouse and limited data storage space. In addition, Baviskar *et al.* (2014) notes that the cabling system is hostile to a greenhouse, and mentions the issue of limited data storage on the currently used greenhouse monitoring systems. There are several disadvantages of wired systems, such as the need for long cables, which leads to difficult installation, maintenance, high costs and technical complications during deployment (Erazo *et al.* 2015).

1.4 Research aim

The aim of this study is to develop a cloud-based IoT greenhouse monitoring system that is wireless, cost-effective and easily accessible; and provides unlimited data storage space to farmers for analysis. This system should effectively monitor temperature, humidity and soil moisture levels within the greenhouse. Further, it must provide real time and accurate information and send data to the cloud for storage.

1.5 The research objectives:

- To critically reviewing existing literature of IT based greenhouse monitoring systems supporting crop production
- To developing a wireless cloud-based IoT greenhouse monitoring system; and
- To evaluate the system performance of the developed system.

1.6 The research hypothesis

The aim of the study was to prove the following research hypothesis:

The use of IoT for greenhouse environment monitoring will allow for the monitoring of humidity, temperature and moisture in order to use resources more efficiently.

1.7 Proposed research methodology

This study conforms to the design science research methodology (DSRM). DSRM uses an iterative process involving five steps: awareness; suggestion; development; evaluation, and conclusion (Oates 2005). Hevner (2007) defines DSRM as a research technique for producing innovative artefacts that extend human and social capabilities within a specific domain. DRSM requires the creation of an innovative and useful artefact. Evaluation for the artefact is necessary in order to ensure its effectiveness for the specified problem (Hevner 2007). Furthermore, the artefact must either be innovative or provide a more effective solution to a given problem. Both, "the construction and evaluation of the artefact must be done rigorously, and the results of the research presented effectively to technology-oriented and management-oriented audiences" (Hevner 2007:5). Von Alan *et al.* (2004) stress that, to obtain viable results in DSRM, a dual research methodology is relevant.

March and Smith (1995), Hevner *et al.* (2004) and Hevner and Chatterjee (2010) propose a design research methodology which is a research framework in which IT research can be framed by integrating two complementary disciplines. The first of these is a natural science, in which research focus on theorizing and justifying, and the second is design science research (DSR), where the research focus on building and evaluating the process for IT artefacts. How does this "integration" occur?

1.8 Importance of the study/contribution of the study

Reviewed literature reveals that studies have been conducted and some are still in progress to monitor the greenhouse environment using cloud-based IoT technology. The literature showed evidence of the existence of several studies using radio frequency (RF), global system for mobile communication (GSM) and WSN. The development of a cloud-based IoT greenhouse monitoring system will contribute to the following:

- Using an Internet of Things based system which is based on a wireless sensor network as a solution to the issues of wired network systems.
- Using cloud-computing service for the storage service.as a solution to the limited storage space.

1.9 Thesis Outline

This thesis is organized into five chapters. A summary of the five chapters is given below.

Chapter one: This chapter forms the background of the study of the greenhouse monitoring system, by highlighting the following: why a greenhouse is used in farming and the importance of monitoring the greenhouse environment (motivation); problem statement; research aim, research questions and objectives; the contribution of the study; research outline; and summary.

Chapter two: This chapter presents greenhouse farming, the importance of temperature, humidity, and soil moisture, WSN for monitoring greenhouse, IoT for greenhouse monitoring, cloud computing, and the convergence of IoT and cloud computing technologies for greenhouse monitoring.

Chapter three: This chapter discusses the design science research methodology, which is the adopted research methodology for this study.

Chapter four: This chapter discusses the implementation of design science research methodology and the results obtained.

Chapter five: The conclusion and recommendations are drawn and presented.

1.10 Chapter summary

A rising world population places increasing demands on a nation's agricultural capability. Recent statistics indicate a growing rate of malnutrition and hunger all over the world. The IoT has promising avenues which are capable of offering many solutions towards the modernization of this type of agriculture. It aims to integrate seamlessly both physical and digital worlds in one single ecosystem that makes up a new intelligent era of the Internet. This study sets out to develop a temperature, humidity and soil moisture monitoring system using cloud-based IoT designed specifically for a greenhouse environment. The next chapter will be reviewing the current existing greenhouse monitoring systems.

CHAPTER 2: LITERATURE REVIEW

This chapter focuses on reviewing existing greenhouse monitoring systems. It starts by discussing greenhouse farming and pertinent greenhouse environmental factors, which are temperature, humidity and soil moisture. It also critically evaluates developments in the application of new trending technologies, such as wireless sensor networks (WSN), IoT, and cloud computing in a greenhouse environment. The chapter then goes on to discuss how these technologies are used in a greenhouse environment by reviewing their system architecture, as well as uncovering some challenges faced by these technologies, such as data storage and wireless connectivity. Lastly, the potential benefits of integrating new technologies for greenhouse environment monitoring purpose are reviewed.

2.1 A brief description of a greenhouse

Ramanjaneya (2015) highlights that, since ancient times, farmers have known that adjustment to the environment plays a key role in the productivity of crops. The first attempt to adjust the natural environment was the introduction of greenhouses built in Italy in the sixteenth century to house the exotic plants that explorers were bringing back from their voyages to the tropics (Kerr 1989). The greenhouse is defined as a covered structure that provides plants with an optimally controlled environment for adjustment of climate-growth conditions, to reduce the cost of production and increase crop yield (Gallant 2018). According to Jiménez *et al.* (2012), a greenhouse is an indoor covered place where plants are cultivated and grown. These structures vary in size from small shelters to industrial size structures (Shamshiri and Ismail 2013).

There is a need for concrete systems at each stage of the food cycle from food production, harvesting, transport and distribution, in order to overcome food insecurity (Yadav *et al.* 2012). This study is focusing on the food production stage. The enhancement of food production in developing countries is an urgent matter (Ordonez 2016). Rouphael *et al.* (2010) argue that, to provide for an ever-growing population, strategies should be initiated based on how to increase yield production and also work on the quality of the product. These two points: product quantity and product quality, should be the primary objectives for all farmers together with the speedy development of greenhouse farming in the agriculture industry for the production stage.

The introduction of the greenhouse for crop production significantly reduces the impact of unpredictable outdoor factors (Prathibha, Hongal and Jyothi 2017). Van Lenteren (2000) highlights that greenhouses offer an excellent opportunity to grow high-quality products in large quantities on a small surface area. During the production process, crops go through various, such as germination, sprouting, flowering and fruit development. Crops can also be affected by the environment in which they grow. A farming environment can consist of many factors, including temperature, humidity and soil moisture. These factors contribute or indicate how crops develop either directly or indirectly, and the poor environmental conditions can damage crops or increase the likelihood of diseases.

2.1.1 Temperature management inside the greenhouse

One of the major factors directly affecting crop development and health is extreme temperature. Temperature influences most plant development processes, including photosynthesis, transpiration, absorption, respiration and flowering (Mastalerz 1977). Different types of crops utilize different best growing temperatures which can differ for the roots and shoot environment, and for growth stages throughout the lifespan of the crop (Dias *et al.* 2016). Jain, Bhakar and Singhal (2017) state that, since farmers are usually concerned about crop growth and development, they need to provide these best crop conditions throughout the entire production cycle. Current approaches of temperature management in greenhouses appear far from ideal since they are based on wires, complex structures, complicated installation, high maintenance costs, poor tension and mobility.

2.1.2 Humidity management inside the greenhouse

Water vapour inside the greenhouse is one of the most noteworthy variables that can be disturbing to crop growth. Humidity can be defined as present water vapour in the air

(Stefko and Kollar 2015). It is categorised into two forms: the absolute humidity and the relative humidity. According to Grasley *et* al. (2006:51), humidity is viewed as "the ambient vapor pressure P to P_s at the ambient temperature T_a and is generally expressed as a percent". Even though humidity is viewed in terms of temperature, it relatively occurs due to temperature. Measuring humidity is greatly done in the ground-based weather stations with the primary purpose of reporting and forecasting weather (Bell *et al*., 2017). This is normally done for the provision of information about the presence of water and the actions of water vapour. Humidity is significant to plants because it moderately controls the moisture loss from the plant. The leaves of plants have small pores, CO_2 enters the plants via these pores, and oxygen and water leave via them. Transpiration rates lessening equivalently to the total amount of humidity in the air. This is because water disseminates from areas of higher concentration to areas of lower concentration. Current approaches of humidity management in greenhouses, as in the case of temperature management, appear far from perfect; they, too, are based on electrical wirings, complex structures, complicated installation, high maintenance costs, poor tension, and mobility. Moore (2019) identifies instances of archaic practices where farmers use the traditional method of putting a basket filled with water inside the greenhouse to manipulate the humidity or place the crop pots on top of a tray filled with water.

2.1.3 Soil moisture management inside the greenhouse

The importance of having soil well moistened is well understood; however, it is also necessary for the farmers to be aware of the negative effects of overly moist soil on the development of crops (Jain, Bhakar and Singhal 2017). Monitoring or measuring soil moisture is one way of determining when crops need irrigation and how much irrigation water to apply. Champagne *et al.* (2019) state that the soil moisture is critical for seed germination and the uptake of nutrients by the plants. Excess water in the soil may slow down or stop gaseous exchange between the soil and the atmosphere, which reduces root respiration and growth. Optimal levels of moisture support root growth lengthen the flowering period and increase the overall yield of the crop (Kramer 1944). As for Satyanarayana and Mazaruddin (2013) and Gangasani (2016), soil moisture plays a critical role for proficient photosynthesis, respiration, transpiration and transportation of minerals and other nutrients through the plant. Current approaches of moisture management in greenhouses are more challenging since they are based on complicated installation and high maintenance costs, poor tension and mobility, wirings and complex structures, and, sometimes, on the traditional ways of moisture management.

This study proposes to monitor greenhouse soil moisture, humidity and temperature. Kirshboum (1995, 2006) states that temperature, humidity and soil moisture are the most important environmental factors influencing soil organic material decomposition and production of greenhouse gases' terrestrial environment. According to Kirkham (2014), the absorption and transportation of water and nutrients are reliant on the soil conditions. Thus, it is important to maintain the temperature and the moisture level in the soil at an optimum level to keep the crop healthy. Therefore, this research considers soil moisture, humidity and temperature as the primary factors affecting crop growth. These factors are difficult to monitor manually inside a greenhouse, and current approaches are outdated and laden with inaccuracy (Jiménez *et al.* 2012).

Nachidi, Benzaouia and Tadeo (2006) highlight that it is commonly known that greenhouses are structures that permit the establishment of an indoor microclimate for crop growth, defending it from adverse outdoor conditions. Azaza *et al.* (2016) proposed the construction of a Takagi-Sugeno (T-S) fuzzy model from a simplified nonlinear dynamic model of the greenhouse climate to control air temperature and humidity inside the greenhouse. In addition, the designed system achieved the desired climate conditions in a greenhouse, however, this study was conducted on an empty greenhouse.

Micro-environments, such as greenhouses, are highly used due to the fact that they can be controlled easily. Baviskar *et al.* (2014) note that the current use of the greenhouse monitoring system enables precise monitoring, but it requires a lot of cabling or wiring which is hostile for a greenhouse. They also emphasize the need for the introduction of an efficient and effective wireless system in this area of greenhouse farming.

2.2 Wireless Sensor Networks

A wireless sensor network (WSN) is a network shaped by a large number of sensor nodes where each node is equipped with a sensor to detect physical phenomena (Estrin *et al.* 1999; Lopez *et al.* 2010). WSNs are regarded as a groundbreaking information gathering method to shape the information and communication system which will importantly improve the reliability and efficiency of infrastructure systems. Compared with the wired solution, WSNs feature easier deployment and better flexibility of devices. With the rapid technological development of sensors, WSN has become the key for IoT (Xia *et al.* 2011; Akyildiz *et al.* 2002). Deepika and Rajapirian (2016) posit that a WSN in an agricultural environment is established to collect data from various sensors, as shown in Figure 2.1 below. The architecture model below is adopted from Horvat, Sostaric and Zagar (2012) and Po'ad, Ismail and Jusoh (2017), and it shows the logical view of a WSN.

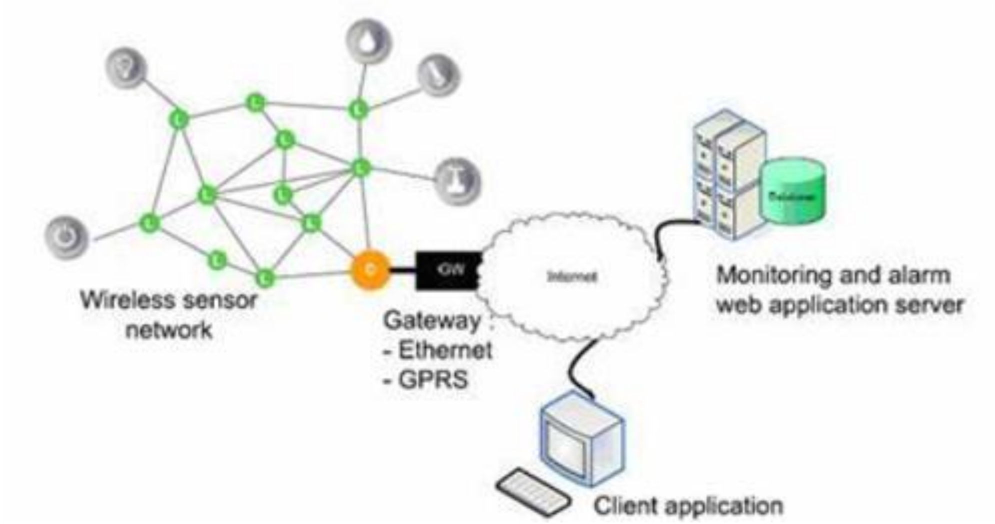

Figure 1: WSN architecture adapted from Horvat, Sostaric and Zagar (2012) and Po'ad, Ismail and Jusoh (2017)

Figure 1 above presents WSN architecture that is made up of several components named nodes. According to Udaykumar (2015), a node is encompassed of a processor, memory, sensors, radio and a battery. The sensors are installed in the field and the real parameters are monitored. In the figure above, there are three basic functions performed: sensing, communication and computation by using hardware, software and algorithm. For sensing, different sensors can be deployed in the targeted area with the selected microcontroller. Wireless communication technologies, such as ZigBee, Bluetooth and Wi-Fi, are normally used.

Tamayo, Ibarra and Macías (2010) agree that WSN technology could give farmers the ability to accurately measure environment aspects affecting production. WSN has been used in agriculture for various purposes, such as the work of Jiber, Harroud and Karmouch (2011), where it is used in building a decision support system. The sensors are used to collect data and then the collected data are analyzed and serve as input to the decision support system. As for Abbasi, Islam and Shaikh (2014), sensor-based monitoring systems in agriculture have a variety of terminologies, such as precision agriculture, smart agriculture, variable rate technology, precision farming, and global positioning system. These terminologies depend on the type of sensors used and the environment deployed (Ojha, Misra and Raghuwanshi 2015). WSNs can be categorized into many broad classes according to the deployment environments**.** However, for the interest of this study, five categories are discussed below.

2.2.1 WSN applications in agriculture

Ruiz-Garcia *et al.* (2009) state that WSN technology has attracted a number of researchers in the agricultural field in the past years. The use of WNS technology is a suitable strategy for data collection and monitoring in a tough environment, such as a greenhouse, croplands, warehouses and refrigerated trucks. In agriculture, a WSN can be used for a collection of diverse requirements of weather, crop and soil information for proactive solutions rather than reactive solutions. Moreover, WSN can be used for multiple crops on a single piece of land, monitoring of distributed land and different fertilizer and water requirements to different pieces of uneven land (Abbasi, Islam and Shaikh 2014). According to Aqeel-ur-Rehman, Shaikh and Islam (2010), the utilization of WSN in the field of agriculture has reached advanced stages. To shed light on these developments, the following sub-sections review the implementation of WSN in the agricultural services of irrigation and horticulture.

2.2.1.1 WSN for irrigation

Abbasi, Islam and Shaikh (2014) define irrigation as the artificial application of water in agricultural farm soil. To improve the traditional method of irrigation, there has been several different irrigation systems in use, such as drip irrigation and sprinkler irrigation that are primarily used to manage the water wastage problem developed using advanced technologies (Rajalakshmi and Mahalakshmi 2016). Many studies have proposed and developed WSN as a measure to monitor irrigation, such as the studies of Damas *et al.* (2001), Morais, Valente and Serôdio (2005) and Kim and Evans (2009). The importance of an automated irrigation system is one of the primary objectives of WSNs for irrigation. Vieira et al. (2017) posit that automated irrigation would be helpful to identify the right moments to irrigate, and the amount of water necessary to fulfil crop water requirements. Consequently, this would lead to a better water use. This is supported by the study conducted by Vieira et al. (2017), based on the design of a long range WSN for precision irrigation. Water flow measurements caught the interest of Granda *et al.* (2017) who posit that the amount of water that the irrigation system can provide per hour should be carefully monitored to save water in dry areas. Hence, they proposed the irrigation measurement system for dry areas based on WSN.

2.2.1.2 WSN for horticulture

Kuroda, Ibayashi and Mineno (2015:19) contend that "In agriculture, there are three types of horticulture systems. The first type is for collecting and analyzing temperature data in agricultural fields, such as a grape plantation. The second is used for plant factories that continuously produce vegetables, regardless of location or season, in an artificially controlled environment. The third is greenhouse horticulture for raising vegetables, fruit trees, and flowering plants". Horticulture mostly deals with the cultivation, production, distribution, and use of flowers, fruits, greenhouse, and ornamentals and is known as small scale or low-intensity farming (Zhang, Kantor and Singh 2004). Greenhouse and viticulture are discussed under this section due to the major concern of the research community towards these areas. *Greenhouse horticulture* is the production of horticultural crops within, under or sheltered by structures to provide modified growing conditions and/or protection from pests, diseases and adverse weather (Van Os 1999). *Viticulture* is the branch of horticulture that focuses on cultivation and harvesting of grapes (Cook, Kliewer and Lider 1974). For the interest of this study, intensive review for application of WSN in a greenhouse is presented in the section below.

2.2.1.3 Wireless sensor networks for greenhouse monitoring

This study reviewed WSN to establish its applicability towards monitoring a greenhouse environment. The greenhouse needs to be monitored and controlled to trace down the local climate parameters in different angles of the greenhouse. A study was conducted by Morais *et al.* (1996) to monitor the greenhouse climate using a WSN to collect indoor and outdoor climate data. However, the system was using old technology devices, such as the Microchip PIC16C71 and the Intel 87C592 microcontrollers.

Stipanicev and Marasovic (2003) propose a greenhouse monitoring and controlling system, which is an embedded web server unit system based on TINI board. TINI board is based on Java computer "that uses TINI chip set plus 512 Kbytes commodity SRAM, and interface circuitry in a 68-pin SlMM stick form factor" (Stipanicev and Marasovic (2003:1352). Using the simple 1-wire local network to collect data from distributed sensors and activating actuators and the web server is connected to the internet through an Ethernet or dial-up network. Stipanicev and Marasovic (2003) claim that the developed system showed all advantages of network embedded system technology (NEST), as the possibility of changing physical topology and low dimensions. Moreover, the system showed to be a cost-effective comparison to PC based systems whilst preserving the full functionality at the same time. However, the system has limited reliability and its physical topology is not flexible.

A study conducted by Mancuso and Bustaffa (2006) indicates that tomatoes can be grown inside a greenhouse and soil temperature can be monitored using WSN, through Sensicast devices for soil temperature measurements. They also developed a web-based crop monitoring system so that greenhouse farmers can read the measurements over the internet. The system can also send an alarm to farmers through mobile phone by short message service (SMS) or general packet radio services (GRPS) if there is any change that needs to be done. The aim was to maximize crop production and quality.

Liu, Meng and Cui (2007) have proposed to develop and test a WSN model for environmental monitoring inside the greenhouse. The study was presented in two parts. In the first part, many sensor nodes were used to measure temperature, humidity and soil moisture and send data to the sink node. A sink node is a device that receives all the collected data by the sensor nodes in a WSN (Akyildiz *et al.* 2002). A sink node was installed indoors for collecting and transferring data wirelessly to a remote computer terminal using short message service (SMS). The second part consists of the global message service (GMS) module and management-based software database running on the remote computer terminal. The aim of their study was to provide dynamic and real-time data to the farmers anytime and anywhere using mobile devices. However, they did not focus on using an internet web-based system for wider and possibly cheaper accessibility.

A study by Tamayo, Ibarra and Macías (2010) proposed better crop management with decision support systems on the WSN model. This offered valuable information with regard to the surrounding environment and potential on the crop status to eliminate unwanted diseases and pest-related diseases. The model can also be used to predict tomato and pumpkin zucchini crops' plagues and diseases, particularly in the open fields as well as in greenhouses. However, they have reckoned that this system can incorporate web-based technology.

In China, Ai and Chen (2011) proposed the green house environment monitor technology implementation based on the android mobile platform. They have implemented a technology which uses a mobile phone as a monitoring terminal in a monitoring greenhouse environment. In this system, they used two sensors: temperature and humidity sensors. Sensors are cable type. General packet radio services (GPRS) are used to send messages.

Chavan and Karande (2014) presented a smart WSN to monitor the agricultural environment for soil temperature, humidity and soil moisture. They used Zigbee to remotely monitor the environment sensor nodes and transmit data wirelessly to the central server, which collects, stores and sends data on demand to client mobile devices. Zigbee R224.24 is an inexpensive wireless device that can be used to connect to another device on the network. Zigbee proved to be an effective, yet cost-effective device.

Majone *et al.* (2013) developed a WSN soil moisture monitoring system, connected in multiple hop configuration. The system is based on independent sensor nodes, which allow for both real-time and historical data management and was connected through an input/output interface to a WSN platform. However, this study did not check sensor flexibility.

The agriculture domain has a potential for WSN implementation, as shown by the above studies. However, after reviewing these studies, it was noted that WSN solutions are still too complex to implement with maintenance that requires major technical support. This influenced the researcher to conduct a literature review on the utilization of IoT application software in integration with WSN network to try to minimize the complexity.

2.2.2 Wireless Sensor Network Categories

Terrestrial: In terrestrial wireless sensor network (TWSN), the nodes are dispensed over the environment and the improvement in Micro-Electro-Mechanical System (MEMS) technology has allowed the formation of smart, small-sized, although lowcost sensors. These vigorous sensors permit a sensor node to precisely gather the encircling data (Yu *et al.* 2012; Ari *et al.* 2015; Ojha, Misra and Raghuwanshi 2015). Yigit *et al.* (2018) note that the terrestrial sensor networks are designed to function on land, therefore, they need air as a communication channel. A typical terrestrial sensor network is composed of a transmitter and receiver part. It uses electromagnetic radio waves for carrying the information.

Underground: In this type, the wireless sensors are rooted within the soil**.** Sensors are rooted underground using wireless technology to enable them to communicate. The Underground wireless sensor network (UWSN) is used for agricultural purposes to monitor conditions in the soil. It has a land node to transfer sensed information from the underground nodes to the base station (Yu *et al*. 2012; Yu *et al.* 2013; Lino 2014).

Underwater: This kind of network is used under water and it imposes challenges due to the hostility of the deployed environment (Ari *et al.* 2015)**.**

Multimedia: This WSN is proposed to enable tracking and monitoring of events in the form of multimedia, such as imaging, video and audio, using cameras and microphones as sensors (Ari *et al.* 2015).

Mobile: These networks consist of a collection of sensor nodes that can be moved on their own and can interact with the physical environment. The mobile nodes have the ability to compute, sensing and communicate (Pfeifer, Olariu and Fersha 2005).

2.3 A description of the architecture of Internet of Things

According to Ng and Wakenshaw (2017), citing Ashton (2009), the term 'Internet-of-Things (IoT)' was first introduced by Kevin Ashton to describe how IoT can be created by "adding radio frequency identification and other sensors to everyday objects" (Ashton 2009). Over time, the term has grown into one that describes the IoT as a network of entities that are connected through any form of sensor, permitting these entities known as Internet-connected constituents, to be located, identified and operated.

Gubbi *et al.* (2013) define the Internet of Things (IoT) as a broad term that describes the interconnection of different daily life electronic objects through the internet. In the concept of IoT, every object is connected with each other through a unique identifier so that it can transfer data over the network without human interaction. Zhao *et al.* (2010) define IoT as a network of Internet-enabled objects, together with web services that interact with these objects. Underlying the Internet of Things are technologies such as RFID (radio frequency identification), sensors and smartphones. The basic idea of the loT is that virtually every electronic component in this world can also become a computer that is connected to the Internet. Stojkoska and Trivodaliev (2017) define IoT as "the collection of smart devices or objects capable of communicating and computing, ranging from simple sensor nodes to home appliances and sophisticated smartphones". However, for this study, IoT is defined as the communication of sensors and the physical world.

Similar to any other information system, the IoT depends on an amalgamation of hardware, software and architecture. Whitmore, Agarwal and Da Xu (2015) highlight that IoT critical hardware infrastructure includes radio frequency identification (RFID), near field communication (NFC) and sensor network. However, Stojkoska and Trivodaliev (2017) state that the standard IoT usually contains many WSN and radio frequency identification (RFID) devices. Oppermann, Boano and Römer (2014) highlight that WSN is a model that has tremendously been explored by the research community in the last two decades and consists of smart sensing devices that can interconnect through direct radio communication. In comparison to WSN, RFID devices are not as experienced. RFID mainly comprise of two parts: an integrated circuit with some computational capabilities and an antenna for communication. Atzori, Iera and Morabito (2010) identify IoT as the main enabling factor for the integration of different technologies and communication solution.

For the purpose of this study, sensor network is the main hardware infrastructure that was used. Sensors are devices that monitor features of the environment or other objects, such as temperature, humidity, movement and quantity. When several sensors are used in conjunction and relate to each other, they are called a wireless sensor network. Whitmore, Agarwal and Da Xu (2015) highlight that these hardware infrastructures have been explored previously, e.g., WSN has been used previously in agriculture.

The IoT uses the currently presented hardware infrastructure to a great extent, however, innovative software should be written for backing the interoperability among several assorted devices and searching the data formed by them (Li, Da Xu and Zhao 2015; Whitmore, Agarwal and Da Xu 2015). The vital technology in the comprehension of IoT systems is middleware, which is a software that provides interoperability among incompatible devices and applications (Ngu *et al.* 2016). Whitmore, Agarwal and Da Xu (2015) articulate that several researchers have proposed the application of semantic middleware to interoperate the diverse classes of devices collaborating through different communication arrangements.

2.3.1 Definition of Internet of Things architectural model

This trending technology of IoT brings many emerging applications and services, which add value to the market place. Zheng *et al.* (2011) stress that this kind of technology includes architecture models, network and communication technology, discovery and search engine, and security and privacy technology. There are many different sentiments concerning the number of layers in IoT. However, Zheng *et al.* (2011), Atzori *et al.* (2012) and Zhao and Ge (2013) all agree that IoT has a three-layer architectural model. Figure 2 below is adapted from Mahmoud *et al.* (2015) to give a logical background of how IoT works in everyday life. The diagram shows three layers described from the bottom up.

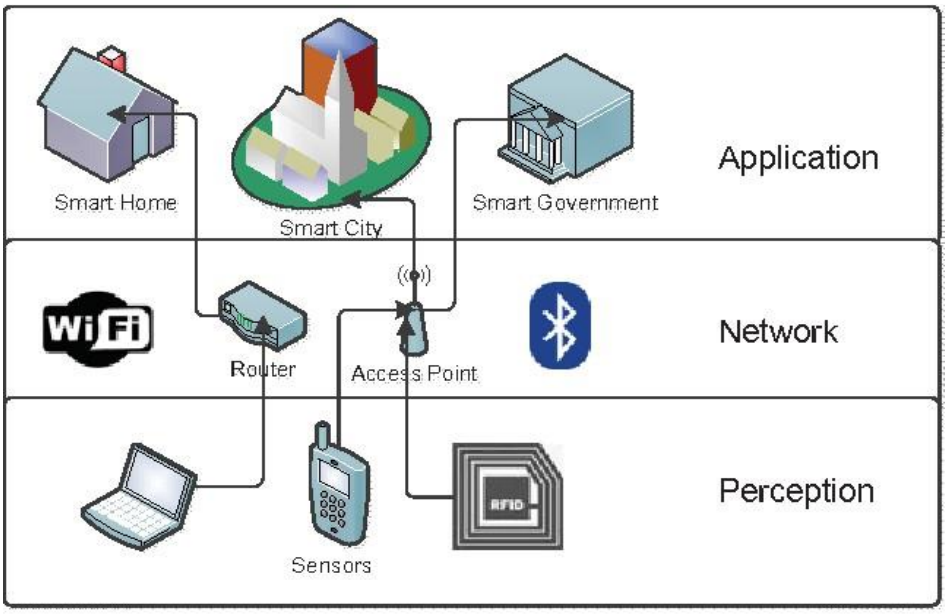

Figure 2: Internet of Things architectural model adopted from Mahmoud et al. (2015)

Layer one (Perception/sensing layer): The main function of this layer is to collect data from the physical world with the help of sensors, laptops, smartphones and actuators depending on the aim of the system. Data collected can be temperature, humidity, concentration, etc. and then transferred to the network layer. Moreover, data collaboration can also be performed on this layer (Atzori *et al.* 2012). Zheng *et al.* (2011) define collaboration as the amalgamation of behaviour in diverse systems.

Layer two (Network layer): The main function of this layer is to connect all things and components that are part of the perception layer, such as sensors, laptop, physical objects, and let them be aware of each other. The network layer allows data routing and transmission to different hubs and devices over the Internet (Leo *et al.* 2014). This layer is built up from the combination of varied networks that may include technologies such as Wi-Fi, LTE, Bluetooth, 3G, Zigbee, etc.

Layer three (Application layer): This layer guarantees the authenticity, integrity and confidentiality of the data. After storing, processing and analyzing data intelligently, this layer delivers the data based on the users' requests.

2.3.2 IoT for agriculture

Agricultural IoT has become the inevitable trend of agricultural society (Dan *et al.* 2015). In agriculture, IoT provide several services such as reducing number of site visitation, which can be costly and manage inadequate water dissemination during irrigation; however, the primary aim is to increase production. It is more vital to find technique that give faultless analyzing and controlling to develop proper environment. Sumithabhashini (2020) highlights that, IoT focuses mostly on greenhouse environment because it is easy to monitor an already controlled environment compared to an open field. Similar to WSN, IoT uses sensors, wireless technologies and mobile devices as useful assets of IoT technology. Hence, several authors describe WSN as the primary key feature of IoT (Xia *et al.* 2011; Akyildiz and Stuntebeck 2006). This section focuses on reviewing factors affecting greenhouse environment during production. Furthermore, this section also reviews integrated systems that uses IoT technology

2.3.2.1 IoT based greenhouse environment monitoring systems

Lan and Ma (2014) conducted a study to develop a greenhouse environment monitoring system with the aim of observing and managing the development of crops in the greenhouse. The system can automatically collect greenhouse environmental parameters, such as air temperature, air humidity, illumination, and soil temperature and soil moisture. This system adopted ZigBee chip integrated wireless sensors and data collecting modules. The shortfall of this system is that ZigBee technology has a low channel bandwidth of 1MHz, so the connection might be slow. ZigBee is restricted to wireless personal area networks (WPAN) and reaches an average of 10 to 30 meters for usual applications (Yick, Mukherjee and Ghosal 2008; Ray 2015).

Shenoy and Pingle (2016) proposed a system to monitor the greenhouse environment during the production period using different sensors to monitor PH level, temperature, humidity, nutrients and soil moisture, with an IoT-based system. This system sent data to the server via Bluetooth or Wi-Fi. The server kept the records of when certain events were triggered, and which action were taken. The issue with this system was relying more on Bluetooth, which is distance limited.

Liao *et al.* (2017) highlight that traditional methods for monitoring the environmental factors of a greenhouse and the growth of phalaenopsis orchids often suffer from low spatiotemporal resolution, high labour-intensity, requiring much time, and a lack of automation and synchronization. To solve these problems, the authors developed IoT based monitoring system with wireless network using ZigBee protocol, which is one of the low-power and short transmission range communication protocols and is often adopted by wireless sensor networks (Yick, Mukherjee and Ghosal 2008). The architecture is adapted from Liao *et al.* (2017). Figure 3 shows how this system was designed.

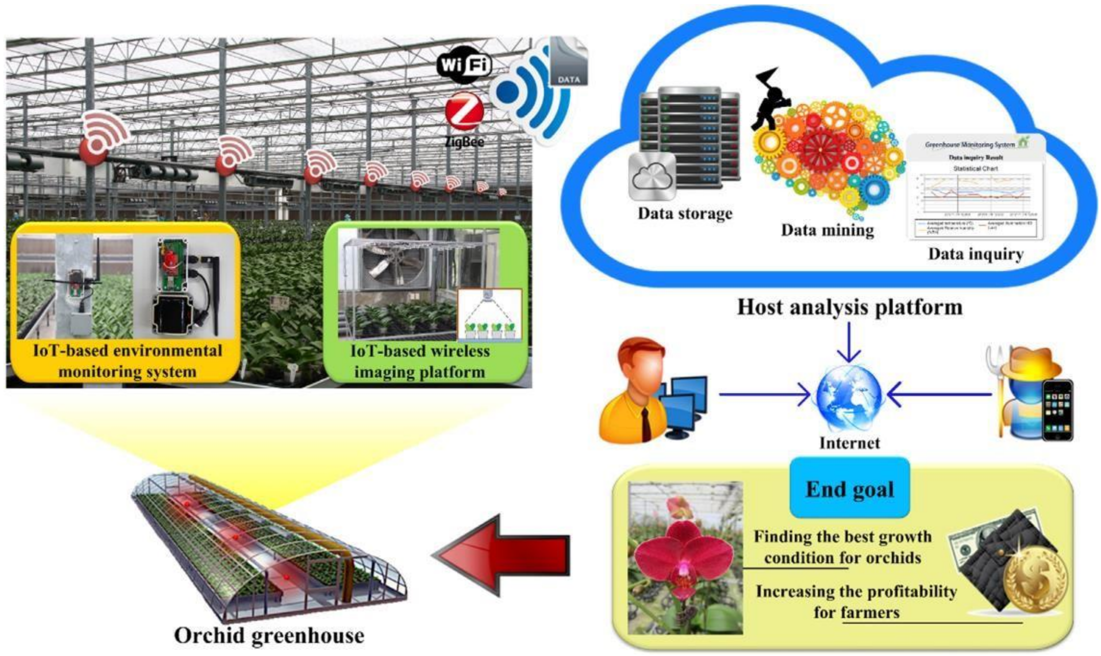

Figure 3: Adopted architecture model (Liao et al. 2017)

The diagram above presents an IoT based system; this system is divided into three subsystems: An IoT-based environmental monitoring system, an IoT-based wireless imaging platform, and a host analysis platform (HAP). An IoT-based monitoring system was used to collect environmental factor and send the data to the gateway through the Zigbee protocol. The gateway organizes the data and sends it to the HAP through the Wi-Fi connection. The IoT-based wireless imaging platform is built upon a set of embedded boards, such as raspberry pi, and it is accountable for apprehending the orchid images and conveying them to the HAP via the file transfer protocol.

Akkaş and Sokullu (2017) conducted a study to present WSN prototype consisting of MicaZ nodes to measure greenhouses' temperature, light, pressure and humidity. The generated data would be shared with the help of IoT. With this system, farmers can control their greenhouse from their mobile phones or computers, which have internet connection.

Vimal and Shivaprakasha (2017) proposed IOT based Greenhouse environment monitoring and controlling system using Arduino platform. Using GSM (Global system for mobile communication) modem to send SMS (short message service) that convey the present status of the environmental parameters and Ethernet. Arduino function as the heart of the system. Here pH sensor also used to measure pH of the soil. The pH indicates a solutions acidity or alkalinity. The pH sensor consists of pH probe and a pH sensor module. The Arduino Ethernet shield allows Arduino board to connect to the internet. The GSM sends SMS to the user when sensor value exceeds a defined level. The user turns on the actuator by sending another SMS. All environmental parameters are sending to server through Ethernet and stored in the database. Therefore, the user can monitor and control parameter through android mobile application. However, both systems do not cater for data storage for long term referencing.

Kitpo *et al.* (2019) proposed the IoT system to monitor the growth of tomatoes using camera as an IoT devices capturing images for further analysis in a greenhouse. For data analyses, they considered and examined fruits regions in tomato plant images. Since the dataset of tomato plants images includes the background of soils and leaves, this study detects and extract only the fruits regions, then these regions were classified in different growing stages. However, this system was developed for tomato greenhouse environment only.

Subahi and Bouazza (2020) presents a work, based on the IoT new advances in using sensors equipment, they planned to develop a smart Energy-Efficient (EE) system which monitors and controls internal greenhouse temperature. The proposed system will allow increased and improved productivity. The main study objective was not only to build a consistent growing environment, but also to automate the whole system and make it smart to save energy and production costs. The proposed approach focuses on monitoring and controlling greenhouse internal temperature, but it is also open to expanding other kinds of properties, e.g. Carbon dioxide (CO2) and humidity. The proposed system is regarded as intelligent because it is capable, autonomously, to monitor the outside temperature and the energy consumption rush hours, in order to precisely produce the appropriate reference temperature, and ensure that the greenhouse temperature reaches this reference temperature. In addition, this system can identify the angle of the Sun rays in order to control the opening and closing of the awnings, which results in reducing the effects of high temperatures.

2.3.2.2 Production phases integrated into an IoT-based system

Gondchawar and Kawitkar (2016) proposed an IoT-based smart agriculture aimed at developing an integrated system which will take care of all factors affecting the productivity in every stage, such as cultivation, harvesting, and post-harvesting storage, by using automation and IoT technologies. This work was focusing on three system features: firstly, smart GPS-based remote-controlled robot used for performing the operations, such as weeding, spraying and moisture sensing. Secondly, smart irrigation was also included with smart control and an intelligent decision-making system based on precise real-time field data. Thirdly, smart warehouse management, focuses on controlling temperature, humidity and theft exposure in the warehouse. These tasks were regulated through any remote smart device or computer connected to the Internet and the tasks were performed by interfacing sensors, Wi-Fi or ZigBee modules, camera and actuators with a micro-controller, such as raspberry pi.

The use of IoT in the greenhouse environment has shown a great potential of improving crop production, as the above literature review has shown. However, it is worth noting that the storage of gathered data is still a great challenge when using IoT only. Hence, this leads to the need to review the application of cloud computing for the purpose of data storage with the integration of IoT. The following section focuses on reviewing cloud computing for greenhouse farming.

2.4 A Definition of cloud computing

Kundra (2011) and Dash *et al.* (2012) define cloud computing as a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction. Rupanagudi *et al.* (2015) define cloud computing as the practice of using network remote servers hosted on the internet to store, manage and process data, rather than a local server or a personal computer.

Sindhanaiselvan and Mekala (2014:1) highlight that “cloud computing is a term used to describe both a platform and type of application. A cloud-computing platform provides dynamically provisions, configures and reconfigures servers as needed". Servers in the cloud can be virtual machines which are an alternative to having local servers handling applications. The clients of the cloud computing network usually have no idea where the servers are physically located, they just spin up their application and start working. Cloud computing provides several compelling features that make it attractive to different fields. It has economical and technical advantages that can contribute towards running a smooth business, for example, cloud computing can help by reducing business risk and maintenance expenses and it is easy to access.

Zerger *et al.* (2010), Mahesh, Savitha and Dinesh (2014) and Botta *et al.* (2016) contend that the general architecture of cloud computing can be fragmented into four layers: datacenter layer (hardware), infrastructure layer, platform layer and application layer. Each of these layers can be perceived as a service for the layer above and as a consumer for the layer underneath.

Datacenter layer (Hardware): This layer is accountable for managing the physical resources of the cloud and includes servers, routers, switches, power and cooling system. This layer usually contains thousands of servers that are organized in racks interconnected through switches and routers.

Infrastructure layer (Virtualization): This layer creates a pool of storage and processing resources by partitioning the physical resources using virtualization technologies, such as VMware. The infrastructure is an important component of cloud computing since it has many key features, such as dynamic resource assignment, and is only made available through virtualization technologies.

Platform layer: This layer is constructed on top of the infrastructure layer and consists of operating systems and application frameworks. The function of this layer is to minimize the load of deploying application straight into VMware.

Application layer: At the top level of the hierarchy, the application layer consists of the actual cloud application. Figure 4 below is adapted from Zhang, Cheng and Boutaba (2010) and explains cloud computing services.

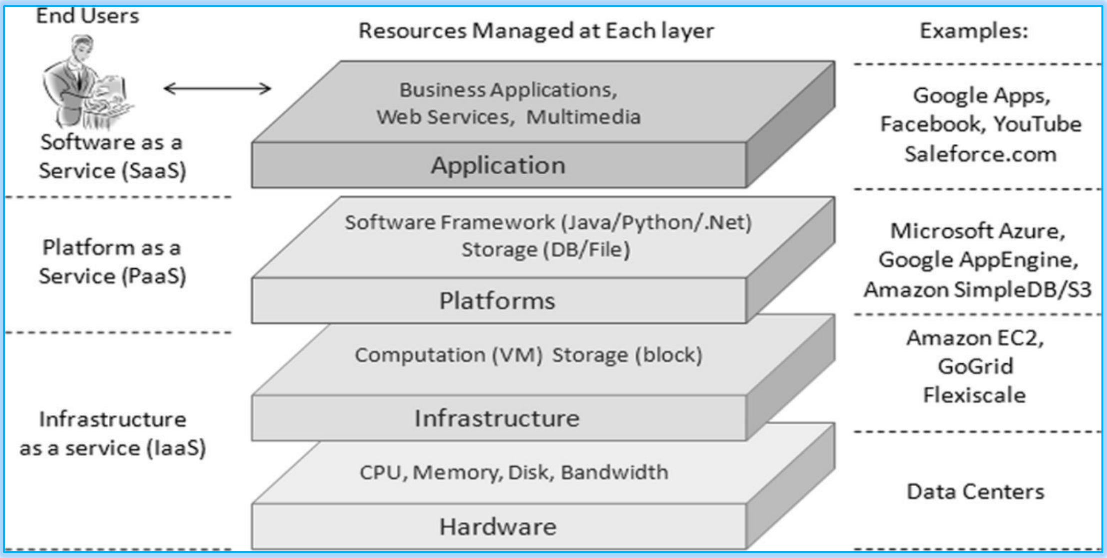

Figure 4: Cloud computing architecture from Zhang, Cheng and Boutaba (2010)

2.4.1 The application of cloud computing in agriculture

Cloud computing has been used and promoted in numerous fields, such as medicine and medical, manufacturing, financial services, energy, communication and other key areas, which play an important role for improving the efficient use of resources, information sharing and integration. According to Kun Qian (2012), the application of cloud computing in agriculture has the potential to resolve the bottleneck problem of agricultural modernization and agricultural information dissemination. Moreover, cloud computing provides an opportunity for farmers to have access to tangible information to enhance farming procedures by breaking knowledge limitations. In addition, according to Yanxin (2013), cloud computing applications in agriculture in 2013 was still on theoretical research phase in most of the developing countries and seemingly immature, although this technology is promising to rapidly improve the management of information in the agriculture industry. However, a report from IoT and research (2017) posits that smart farming through cloud computing is expected to grow at a linear rate.

Cloud computing can be applied in agriculture for different purposes, such as large-scale information storage (Jaeger, Lin and Grimes 2008), low-cost access to IT resources (Rao, Sasidhar and Kumar 2012), cloud agriculture system and easy solution to farming queries (Roy *et al.* 2017).

Large-scale information storage: Cloud computing offers high size data stores to store the massive scale data and information (Jaeger, Lin and Grimes 2008). The databases relating to the farming community, such as crop, weather, market information, farmers' experiences of the agricultural processes, and information about the pesticides and prescriptions can be all easily stored in the cloud. According to Wang *et al.* (2014), IoT utilizes heterogeneous devices to collect data, and the efficient storage of the data collected becomes a problem. Gubbi *et al.* (2013) indicate that cloud computing can provide virtual infrastructure for such utility computing which integrates massive heterogeneous monitoring devices, storage devices, analytics tools, visualization platforms and clients' delivery. Wang *et al.* (2014) propose the use of cloud computing for agricultural systems to monitor heterogeneous sensors. In the study conducted by Rubala, Anitha and Student (2017), cloud computing was used as the data server to access and control the agricultural production, and the results of the study showed improved competence, productivity and profitability in the farming production system.

Low-cost access to IT resources: Cloud computing offers low-cost access to many IT resources. It has introduced the pay as you use model to try to motivate the farming community not to invest in owning the IT resources since it is not necessary to do so (Rao, Sasidhar and Kumar 2012). Instead, they can access the required resource through renting arrangement from the cloud computing providers. It is a much cheaper and reliable method.

Cloud agro system: The cloud agro system is the cloud-based electronic system used to monitor the overall functionalities associated with agriculture (Goraya and Kaur 2015; Roy *et al.* (2017). The system has online service techniques accessible to all the farmers, from any region of the country and at any time.

The reviewed literature shows that cloud computing carries more pros in the greenhouse industry. However, cloud computing depends on the distribution of resources, which are crucial requirements for an effective IoT platform. Hence, the following section reviews the convergence of cloud computing and IoT for greenhouse farming.

2.5 Convergence of cloud computing and IoT for improved greenhouse monitoring

Liu *et al.* (2015) and Botta *et al.* (2016) highlight that the two worlds of cloud and IoT have seen a rapid and independent evolution. However, plenty of complementary characteristics on these worlds are the result of their convergence, this complement of cloud and IoT has been identified in literature and foretell the future. The primary requirement for an IoT platform using cloud computing is grounded in the unlimited capacity and the resource sharing to umbrella, and reimburse its technological constraints such as storage, processing and communication (Biswas and Giaffreda 2014).

Cloud computing provides elasticity and scalability of resources and applications. The services and resources are simply accessible and available (Wang, Chen and Wang 2015). Internet of Things can benefit from the cloud computing by stretching its scope to deal with things in the real world in a more disseminated and dynamic wayf and to provide new services on a large number of real-life scenarios (Botta *et al.* 2016).

Therefore, an innovative IT prototype in which cloud and IoT are two complementary technologies merged is anticipated to disrupt both the current and future worlds.

2.6 Chapter summary

This chapter provided a clear insight on the nature, characteristics and contributions of WSN, IoT and cloud computing in agriculture, by pointing out the potential of these technologies as the tool to overcome the food insecurity and water wastage issues. Reported literature reveals that studies have been conducted and some are still in progress to monitor the greenhouse environment using IT-based greenhouse monitoring systems supporting crop production. The literature showed evidence of the existence of several studies using, WSN, IoT and cloud computing. Authors such as ((Ruiz-Garcia *et al.* (2009)), (Abbasi, Islam and Shaikh (2014)). and (Aqeel-ur-Rehman, Shaikh and Islam (2010))) focused on the application of WSN in agriculture production. One of the WSN main benefits is the capability that it offers to generate alarms when the environmental variables are reaching critical points. This fact permits farmers to make arrangements for neutralizing the negative of high temperatures or humidity. Another benefit is the remote monitoring and automation of the environment variables of the greenhouse (Dan *et al.* (2015)). (Sumithabhashini (2020))) Highlighted that, IoT provides several services such as reducing number of site visitation, which can be costly and manage inadequate water dissemination during irrigation. IoT focuses mostly on greenhouse environments because it is easy to monitor an already controlled environment than an open field. Cloud computing can be applied in agriculture for different purposes, such as large-scale information storage (Jaeger, Lin and Grimes 2008).

However, these technologies have yet to resolve challenges that are hindering farmers. These remarkable challenges are noted as technology limitations, which are generated from the fact that these technologies were designed by looking at specific needs. However, the expansion of knowledge opens doors for new ideas that these technologies will utilize in different fields. One of the generic issues is the fact that most sensors do not have data storage capacity. Thus, there is a need for them to work together in order to surmount these issues and challenges. The succeeding chapter describes the research methodology to be followed by this study and how it is adopted to achieve the development of the cloud-based IoT greenhouse monitoring system.

CHAPTER 3 RESEARCH METHODOLOGY

This chapter explains the design science research method as the main approach used in this study, and how it was adopted to achieve the development of the cloud-based IoT greenhouse monitoring system. The chapter begins by presenting an overview of the variety of research methods that can be used in the computing field by mentioning six different paradigms in the computing field and the reason for selecting design science research (DSR) as the philosophical approach adopted in this study. This chapter also discusses the case site for this study and the reasons why the Horticulture Department of the Durban University of Technology (DUT) was used.

3.1 Research paradigms in Information Technology

A paradigm is defined as a set of ethics, beliefs and assumptions, ideas, and activities which guide a group of individuals (researchers, farmers) on how to perform and behave (Al-Debei 2010). Originally, in Information Technology (IT) research, three major paradigms were commonly employed in most studies, which can be distinguished as a *positivist*, *interpretive* and *critical*. However, later a fourth paradigm was proposed as a major paradigm called the *design science* paradigm, with the aim to enhance IT research to be more appropriate to academic research. Furthermore, the study conducted by Niehaves and Stahl (2006) provides six examples of paradigms that actually exist in the IS field, namely: *positivism*, *interpretivism*, *behavioural science research*, *design science research*, *critical research paradigm* and *non-critical research paradigm*.

A review of these six paradigms was conducted with the aim of finding the most applicable one for this study. Therefore, the design science research (DSR) was selected because it concentrates on building innovative technology systems. The DSR paradigm concentrates on understanding the people’s problems and organizations by means of design, construction, utilization and evaluation of artefacts that pursue to transform the current condition to a more desirable one (Hevner *et al.* 2004; Vaishnavi and Kuechler 2004; Kuechler, Vaishnavi and Kuechler Sr 2007).

Oates (2005) and Abbasi, Islam and Shaikh (2014) opine that DSR concentrates on designing and developing new artefacts. The artefacts that evolve from a DSR-oriented study are broadly classified by March and Smith (1995) into four categories: i) constructs; ii) models; iii) methods; and iv) instantiations. The literature on DSR emphasizes that it is critical that the artefacts are utilized for their purpose and be evaluated.

3.2 Describing the design science research paradigm

Research in information technology is considered a DSR if the key purpose is to change a current situation linked to an organization or social systems into a more desirable one through the development of new artefacts (Hevner *et al.* 2004). DSR attempts to create things that serve human purposes. Vaishnava and Kuechler (2004) also mention that DSR includes the formation of novel knowledge through design of novel or innovative artefacts (things or processes) and the analysis of the use and performance of such artefacts, along with reflection and abstraction, to advance and understand the behaviour of aspects of the computing field. Frank (2006) mentions the importance of DSR concentrations on developing new innovative artefacts to cover the boundaries of human problem solving and organizational competencies. As for Venable *et al.* (2011), DSR has a goal of developing new technologies and new uses of technologies. The use of DSR extends the domain of research beyond examining existing technologies and their interaction within a socio-technical environment to embrace the development of new technologies to support and improve the ways to work and live. DSR-oriented studies are broadly classified by March and Smith (1995) "into four groupings: i) constructs, ii) models, iii) methods, and iv) instantiations".

Constructs: is a set of concepts that form the vocabulary of a domain. They shape a conceptualization used to describe problems within the domain and to suggest their solutions Oates (2005).

Models: use constructs to symbolize situations as problem and solution statements. Consequently, modelling is a set of propositions or statements that formulate relationships among constructs Oates (2005).

Method: is defined as a set of algorithms used to accomplish a task. These algorithms provide solutions to solve problems by using models and constructs. Additionally, the method can be reflected as a translator from one model to another for solving a problem Oates (2005).

Instantiation: is defined as the realization of an artefact in its environment (a working system that demonstrates that constructs, models, and method can be implemented in an Information Technology-based system) Oates (2005).

DSR consists of two basic activities, i.e., build and evaluate (March and Smith 1995). The building is the process of constructing an artefact for a specific purpose, while evaluation is the process of determining how well the artefact performs. Based on the two underlying design processes of DSR which were recognized by March and Smith (1995) and Khubisa (2017), there is a common connection between these processes and the objectives of this study. DSR was selected as the approach for this study, firstly, to achieve the aim of this study which is *to develop an effective cloud-based IoT greenhouse monitoring system*; this is associated with the building process, and, secondly, *to validate the system for accuracy and performance;* this is associated with evaluation process.

Finally, this study adopted DSR as the research methodology because its main aim is to change the current situation at the DUT horticulture nursery using IoT, WSN and cloud technology to monitor greenhouse environment. To ensure success, the users of the proposed system were involved throughout the study as change agents. This helped nursery workers to be able to monitor and control the greenhouse environment without being physically there all the time. This was accomplished through designing and developing a cloud-based IoT greenhouse monitoring system.

3.3 Applying design science research methodology

The study conducted by Peffers *et al.* (2007) noted that a number of successful studies have been conducted inside and outside of the IT field, to provide some guidance to define DSR. However, Peffers *et al.* (2007) reviewed these studies and noted a lack of accepted developed framework templates for design science research methodology (DRSM). In response, the authors developed a framework template for DSRM to carry out and present a DSR in IT and it was successfully accepted. Refer to the Figure 7 below for the adopted DSRM.

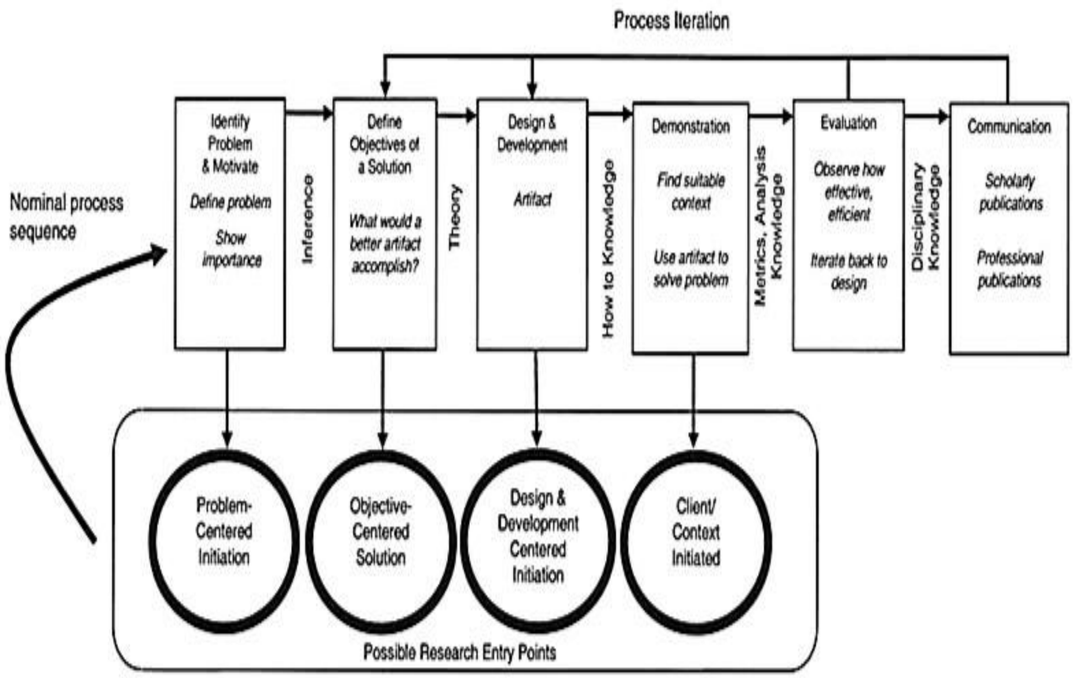

Figure 5: Design science research methodology process framework (adopted from Peffers et al.2007)

Figure 5 above presents the DSRM process framework, which was applied in order to realize the artefact of this research. This figure shows rigorous problems to contribute, to evaluate the design, and to communicate the results to appropriate spectators (Hevner *et al.* 2004). As for Peffers *et al.* (2007), it is not compulsory for a researcher to sequentially follow the activities of DSRM in their order. Depending on the nature of the problem, the researcher may initiate the study in any of the activities; except the last two, which are not connected to any research entry points. In this study, the DSRM process framework was applied following the normal sequential order, as explained below.

3.3.1 Identify problem and motivation

The first step of the DSRM framework, according to Peffers *et al.* (2007), is problem identification and motivation which justify the value of the problem. This motivates the spectators of the research and the researcher. For this study problem, a comprehensive literature review was conducted to find the existing greenhouse monitoring systems in conjunction with physical visits to the site of the experiment. The first objective of this study is to critically evaluate the existing greenhouse monitoring systems. The literature review revealed that a considerable number of systems were available. However, the need for wireless monitoring system was raised by Matijevics and Simon (2010), Dhumal and Chitode (2013), Baviskar *et al.* (2014) and Erazo *et al.* (2015), who note that a greenhouse is hostile to cabling system and limited data storage issue. Several disadvantages were shown by a wired system, such as the need for long cables, which lead to difficult installation, maintenance, high costs and technical complication during deployment.

During the visit to the DUT nursery (the site of the experiment), the formulated questions were addressed to the manager and ground workers. According to Benbasat and Zmud (2003), the questions asked should be related to what the researcher wants to achieve, e.g., is the DUT nursery currently using an IT-based system to monitor greenhouse environment? Also, observations were made, and it was confirmed that there is no IT-related system that is currently used to monitor greenhouse environment; only the traditional methods of farming were the core driver of the nursery.

A suggestion to develop an IT-based system for DUT's nursery was proposed. The motivation to use DUT's nursery was raised after the visitation to the Horticulture Department at the Durban University of Technology. Consequently, the suggestion was made for the integration of new trending technology, such as the Internet of Things, cloud computing, and social media applications, such as Twitter, Facebook, etc. as part of the solution to the current situation.

3.3.2 Define the objectives for a solution

The second step of the DSRM framework deals with the establishment of objectives of a solution from the problem definition and knowledge of what is possible and feasible.

After identifying the problems that are associated with the current nursery at the DUT, this study aimed at developing a cloud-based IoT greenhouse monitoring system. The proposed system should use fewer cables and be more effective, efficient to nursery workers and easy to use.

3.3.3 Design and development

The third step of the DSRM framework focuses on the design and development of an artefact. It involves defining the artefact's desired functionality and its construction, and then creating the actual artefact. According to March and Smith (1995) and Hevner *et al.* (2004), the possible artefacts of IS are constructs, models, methods and instantiations. This study developed an instantiation, which is a cloud-based IoT greenhouse monitoring system. Reggio (2018) mentions that developing a system based on the Internet of Things is a novel task which is scarcely supported by software engineering, as stated by Larrucea *et al.* (2017:25), who note that "there is no consolidated set of software engineering best practices for the IoT that has emerged". Therefore, this study adopted the stance by Larrucea *et al.* (2017:25) that "past software engineering techniques can be harnessed and adapted to the challenge of today's IoT." Hence, Huang *et al.* (2010) adopted the novel web-based application development life cycle. . Figure 6 demonstrates a diagrammatic illustration of this novel life cycle model.

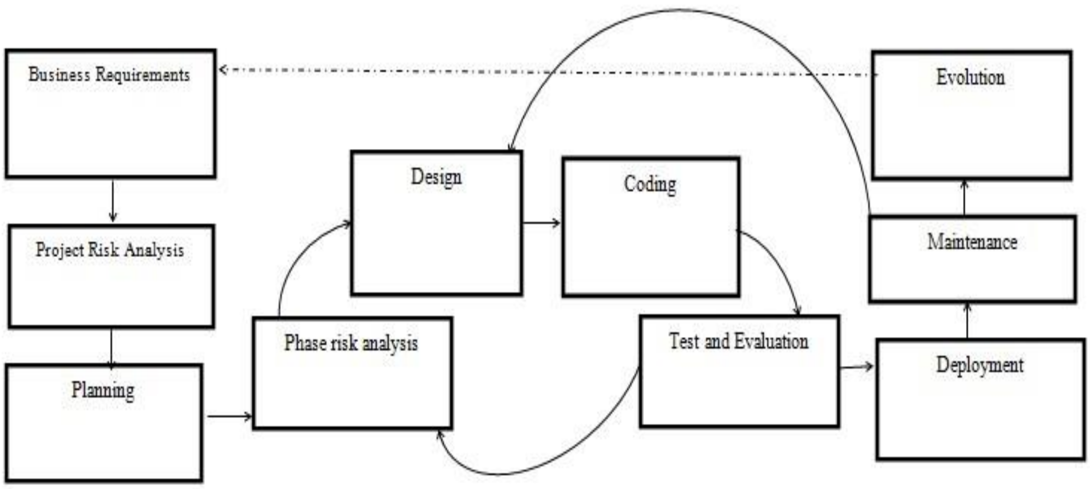

Figure 6: A novel web-based application development life cycle model from Huang et al. (2010)

This model consists of three sets of processes, namely: requirement processes; development processes; and evolution processes. Though this step in DSRM concentrates on design and development, nevertheless, it was essential to first consider the requirement process because the system of greenhouse monitoring would be designed based on farmers' requirements.

3.3.3.1 Requirement processes

The set of tasks of the required processes contain business requirements, project risk analysis and planning. As for this study, the researcher defines business requirements as the greenhouse monitoring system requirements. For this phase, Huang *et al.* (2010) posit that the researcher should anticipate on the functionality and performance of the system, such as what will be needed for this system to function well in the greenhouse environment, and also laying down the criteria for determining good system performance. To come up with suitable greenhouse monitoring system requirements, it was essential for the researcher to visit the greenhouse. A unified modelling language (UML) proposed by Reggio (2018) for IoT system requirements specification was adopted for this study. This UML states that, eliciting and specifying the requirements of an IoT system, requires a combination of software engineering techniques, such as the preliminary modelling of the system domain and the goal-oriented requirements, and the service-oriented UML modelling method (Reggio 2018). In the next chapter, a set of greenhouse monitoring system requirements that were gathered by the researcher are clearly outlined.

The project risks were analyzed and then a strict project plan, with deadlines, was developed. According to Huang *et al.* (2010), in a project risk analysis, the risks, such as unrealistic schedules, misunderstanding of requirements, frequently changing requirements, underestimation of the complexity of non-functional requirements, new technologies, low scalability, and reusability for cloud-based IoT greenhouse monitoring system development, are specifically addressed. The risks were identified for the cloud-based IoT greenhouse monitoring system for DUT's nursery greenhouse using one of the risk identification tools called interviews. The planning was formulated as a project risk analysis.

3.3.3.2.1 Generic life cycle model starts with a generic IoT device

An IoT device hosts software units that are classified into system units, service units, and application units. A generic life cycle for IoT device illustrates the platforms that an IoT device drives through from (re)construction to decommissioning. Figure 7 below demonstrates the generic life cycle model for an IoT device as adapted from Rahman, Ozcelebi and Lukkien {2018).

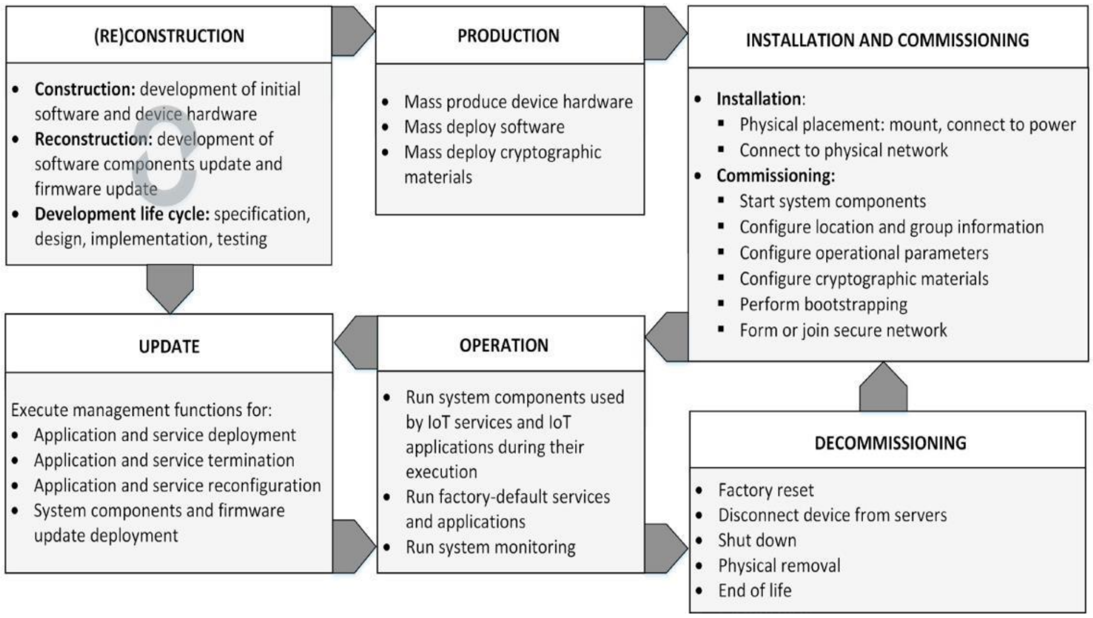

Figure 7: Generic life cycle model starts with generic IoT device adapted from Rahman, Ozcelebi and Lukkien (2018)

The *Construction and re-Construction* phase relates to the installation and updates of device hardware and its primary software (Charalabidis *et al.* 2018). For this study, specific sensors, microcontrollers, operating system and an integrated development environment were used. Furthermore, implementation of re-construction of device firmware and initials software was performed for checking the latest updates for the used software and firmware. Both construction and re-construction must follow the accepted product development life cycle of design equipment, implementation and testing to guarantee functional relevance (Rahman, Ozcelebi and Lukkien 2018). Application on standard conditions, plug-first testing and accreditation during software and hardware development help to accomplish the required level of interoperability

(Khajenasiri *et al.* 2017). Once this stage is completed, an IoT device life cycle drives to the production stage, whereas the re-construction activity starts iteration freely, equivalent to the rest of the device life cycle.

The *production* stage: This stage is built up of manufacturing the hardware and mass deploying of the preliminary software onto the hardware and the mass deployment of the cipher material scan (Rahman, Ozcelebi and Lukkien 2018).

The *installation* and *commission* stage assemble the devices for application and secures communication inside the network. This stage includes two activities: installation and commission. Installation activity includes connecting hardware to power and connect to the network. Commission activity includes starting the system components, configuration of location and group information, configuration of operation parameters, configuration of cryptographic material, performing bootstrapping, and forming or joining a secure network.

The *operation* stage: This phase supports the implementation of IoT services and applications by running applicable system components and runs system monitoring. The presentation of the system counts on how sound these system components accomplish their functions.

The *update* stage: The device goes to the update phase when it desires to execute supervision duties for updating the devices. These updates come in distinct constructions and all these updates need a network connection.

The *decommissioning* stage: In some instances, an IoT device needs to be decommissioned, for example, when it is commissioned to another network or devices reach the end of their lifespan and need to be replaced.

3.3.3.2 Development processes

The development processes handle the list of stages described in the planning processes. For the code to be recyclable, easy to scale, easy to maintain, and robust, risks analysis has to be applied for each phase on the planning processes. To initiate the development process, the researcher performed risks analysis on the planning process phases and a couple of exceptions were notified. These exceptions were reported to the greenhouse supervisor for discussions and to come up with resolutions. According to Huang *et al.* (2010), after risks analysis is conducted, a list of potential risks should be detailed and discussed by developers and other people who are involved in the project.

In order to present better the structure and the functionalities of the proposed system, before commencing with the development phase, it is essential to model the proposed system. The paradigm followed in this study does not have specific modelling techniques for demonstrating the system, the structure and the functionalities. The commonly used object-oriented modelling called UML was used to model the system (Rumbaugh, Jacobson and Booch 2004). This study adopted a generic life cycle model for IoT proposed by Rahman, Ozcelebi and Lukkien (2018) to perform the design and coding activities. “There are three main elements of Generic Life Cycle Model namely: IoT device, IoT service, and IoT application were identified” (Rahman, Ozcelebi and Lukkien 2016:1058), after reviewing service oriented architecture (SOA) for IoT systems. However, this study focusses on the two elements, i.e., IoT device and IoT service. The generic life cycle for an application is excluded from this study because there is no IoT application that is specially developed for the purpose of this study.

3.3.3.2.2 Generic life cycle for the IoT services

According to Rahman, Ozcelebi and Lukkien (2018), a service is a prescribed detailed whole process of a system. For this study, service is the greenhouse environment (temperature, humidity and soil moisture) monitoring process. An IoT service is utilized by IoT applications to accomplish the desired goals. All the data collected during the monitoring process should be accessed through IoT application. Any IoT service must follow the generic life cycle for IoT service. Conversely, the comprehension of the tasks within the life cycle stages and the conforming stakeholders’ concerns differ from one system to another (Rahman, Ozcelebi and Lukkien 2018). Like the IoT device life cycle, the IoT service cycle starts with the construction and reconstruction stages, and their functions are similar, with the difference of the execution stage, reconfiguration stage and termination stage that are explained below (see Figure 8).

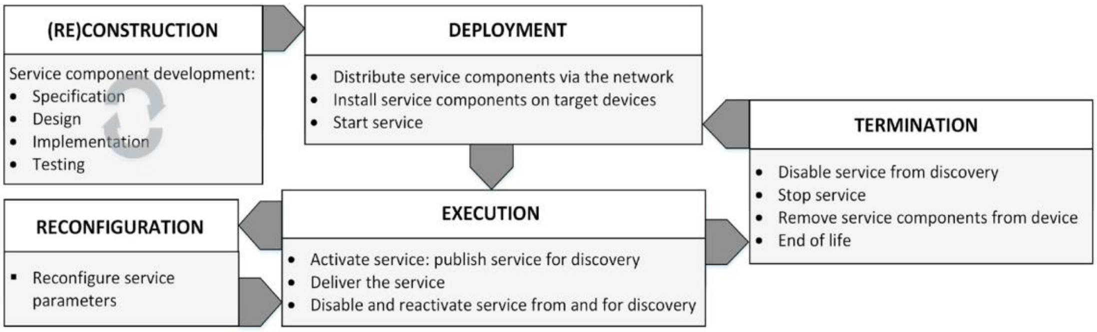

Figure 8: Generic life cycle for the IoT services adapted from Rahman, Ozcelebi and Lukkien (2018)

The *execution* stage: During the run time, the service is issued for discovery and is prepared to attend to requests from applications or other services (Rahman, Ozcelebi and Lukkien 2018). The aim of this phase is to permit the service to be utilized by a rising number of applications and other services while maintaining adequate performance, or in other words, to achieve service scalability.

The *reconfiguration* stage: This is backed by the device supervision functions throughout the update phase of the device life cycle (Charalabidis *et al.* 2018). Appropriate application of this stage helps adjustability as service parameters can be adjusted during execution.

The *termination* stage: Two options are likely for service cessation: firstly, the service units' update is available and prepared to be deployed; secondly, the service is no longer utilized, which is the case when the service is reaching its termination.

3.3.3.3 Evolution processes

Evolution processes start when the developed system is deployed. Once the development of the cloud-based IoT monitoring system is done, the system is deployed, after fixing all the issues that arose during evaluation and maintenance. The web-based application development model promotes the flexibility of moving back into the development process. Huang *et al.* (2010:395) state that, "evolution is to modify an existing product after it has been released".

3.3.4 Demonstration

The fourth step of the DSRM model focuses on demonstrating the use of the artefact to solve one or more instances of the problem. After successful development, the cloud-based IoT greenhouse-monitoring system was deployed in the greenhouse at the DUT nursery. The selected set of participants were authorized to use the system, examine all the functionalities of the system and report their perceptions. The demonstration of the cloud-based IoT greenhouse- monitoring system is shown in the next chapter of this study. The system is illustrated in the form of screenshots.

3.3.5 Evaluation

The fifth step of the DSRM model is the evaluation of the resulting artefact. This step involves comparing the objectives of a solution to actual observed results from use of the artefact in the demonstration. It could include items, such as a comparison of the artefact's functionality with the solution objectives from the second step of the DSRM.

3.3.6 Communication

The last step of the DSRM model involves communicating the problem and its importance, the artefact, its utility and novelty, the rigour of its design, and its effectiveness to researchers and other relevant audiences, such as practicing professionals (Peffers *et al.* 2007). This dissertation, as a master's degree project, can be viewed as communication that clearly argues the greenhouse problem and an artefact called the cloud-based greenhouse monitoring system addressed it and the target academic audience. Furthermore, a journal article will be produced from this study, which will report on the work accomplished in this study.

3.4 Chapter summary

This chapter has presented the design science research paradigm that was adopted for this research study. The chapter has also shown how the DSRM was utilized to carry out the research study by highlighting the steps that were accomplished. A generic life cycle model for IoT was tied to the DSRM, specifically for the purpose of developing a quality cloud-based IoT system within limited resources and time. The next chapter presents the design, development, implementation and evaluation results of the cloud-based IoT greenhouse monitoring system.

CHAPTER 4: SYSTEM DEVELOPMENT, IMPLEMENTATION, AND EVALUATION

This chapter presents the development, implementation and evaluation of a cloud-based IoT greenhouse monitoring system. This chapter is divided into 2 sections: development and implementation; and evaluation. The development and implementation section has the following sub-sections: business and system requirements; system and architectural design; and system implementation. Evaluation has two sub-sections: evaluation method; and criteria-based evaluation strategy. The last part is the summary of this chapter.

4.1. Greenhouse monitoring system: Business/User requirement

Different requirements are included in the process of developing a novel system(s) or enhancing the existing one(s) (Pohl 2010). A cloud-based IoT greenhouse monitoring system requires an effective method for eliciting and specifying its requirements (Reggio 2018). When developing an IT-based artefact, business requirements are the most noteworthy requirements. The business requirement is rated as the highest level on the requirements' sequence and should clearly demonstrate the vision of the system and the value attached to it, which will be of benefit to the audience (farmers) (Wiegers and Beatty 2013). Selioukova (2002) highlights that business requirements should be associated with all other requirements. Wiegers and Beatty (2013) state that software requirements include three distinct levels of IS solution: business requirements; user requirements; and functional requirements. There is a common interrelation between business requirements and user requirements, which influence the system functionality. The business requirement is the manner of stating each user requirements and structure them into comprehensive agreement (Pohl 2010: 1).

The goal of the requirements is to describe what the system should do, allowing the developer and the user to agree on that description. The contextual inquiry method and document analysis method proposed by Wiegers and Beatty (2013) and Privitera (2015) were adopted for gathering business and user requirements for the cloud-based IoT greenhouse monitoring system. Their requirements for this greenhouse monitoring system include:

- To develop and implement greenhouse remotely monitoring system;
- To develop and implement a system that increase production yield; and
- To develop and implement a cost-effective and efficient system.

User requirements explain goals or tasks the users must be able to perform with the system (Wiegers and Beatty 2013). Approaches to present user requirements include cases, user stories and event-response tables. Table 1 below presents the user requirements for the system.

Table 1: User requirements

Level	Description	Priority
Manager	Should be able to use any mobile computing device to access the results remotely	High priority
Manager	Must be able to access the results from previous days/months. (backup system)	High priority
Groundworker	The result shall be available on the site display on the monitor	Desirable
Groundworker	Must receive the urgent report via Twitter or Facebook personal account	High priority

According to Palmer, Liang and Want (1990), requirement classifications are helpful as an approach for problem-solving and requirement analysis, and to detect conflict among requirements. Of course, there were conflicts among these cloud-based IoT monitoring system requirements during the process of establishment. However, the researcher implemented conflict resolution that is suitable for the suggestions that are maintained in the literature concerning requirements elicitation and, subsequently, an agreement was reached.

4.1.1 Greenhouse monitoring system functional specification

This section is composed of both functional requirements and non-functional requirements of the intended system domain. Functional specifications describe the conceptualization of what has to hold in the intended system domain regarding not only the information system but also its environment and their interactions (Chung 1991; Sørensen *et al.* 2011). Functional requirements (FRs): they define the expected services, what the system is supposed to do or should do (Faisandier 2012). Functional and non-functional requirements, for this study, are as follows: continuous monitoring, fewer cables used, cost-effective, mobile access, and accessible via social media. The cloud-based IoT greenhouse monitoring system should be able to store data on the cloud and sent a message via Twitter/Facebook when the temperature, humidity or moisture reach the threshold (Jiménez *et al.* 2012). The development of the cloudbased IoT greenhouse monitoring system fulfils all these requirements through the combination of WSN, IoT, and cloud computing. The researcher decided to use UML, because it is a standard modelling language that allows for specifying, visualizing, constructing and documenting IS. A UML-based proposed for IoT system requirements specification by Reggio (2018) was adopted for this study. For eliciting and specifying the requirements of an IoT system requires a combination of software engineering techniques, such as the preliminary modelling of the system domain and the goal-oriented requirements, and the service-oriented UML modelling method (Reggio 2018).

Reggio (2018) claims that an IoT system presents uncommon features that should be considered to have an effective method for discovering and specifying its requirements. The data collected by the cloud-based IoT greenhouse monitoring system have to be backed up on the cloud according to date and time. This will help farmers to access data from previous days and compare with the current one. The location needs to be specified so that farmers monitoring more than one greenhouse should be aware of which greenhouse is sending data. For this study, the DUT's greenhouse is used.

4.1.2 Case diagram of cloud-based IoT greenhouse monitoring system

Figure 9 shows a use case diagram that demonstrates the interaction that is anticipated to take place between the actors and commands of the cloud-based IoT greenhouse monitoring system. Escalona and Koch (2004) state that the use case diagram defines the functionality of a software system in a graphics format. The dependencies between use cases stated through «extends» and «includes» are vital because they elucidate the interactions that exist, which, in return, help to reduce uncertainty in the system. The «extends» use case signifies optional behaviour whereas the «includes» use case signifies compulsory behaviour.

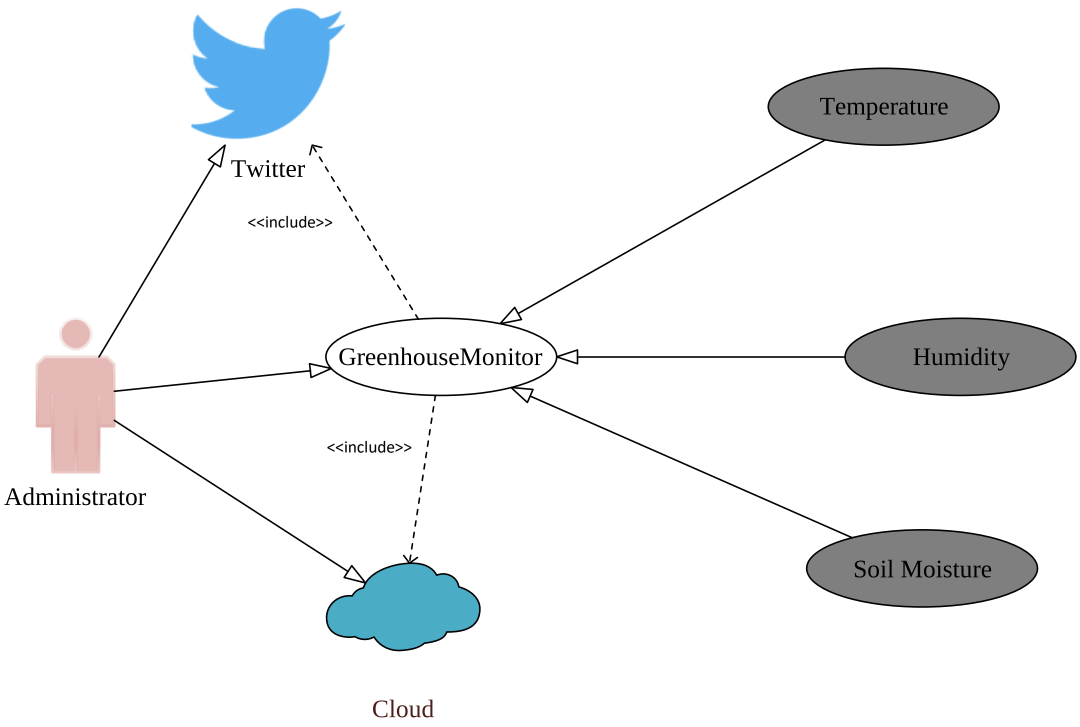

Figure 9: Use case diagram for cloud-based IoT greenhouse monitoring system

The Table 2 below shows the actors that interrelate with the system and the roles. An actor may also be recognized with a role it plays rather than by its name, with respect to a given use case (Wegmann and Genilloud 2000). Reggio (2018) posits that the first step to specify a goal is to determine which participants and objects of the domain are involved or which roles they will play.

Table 2 Roles of actors in cloud-based IoT monitoring system

User	Responsibility
Administrator	The administrator has power over the system, is able to check greenhouse status anytime and anywhere. The administrator is in control of access right to grant, edit and delete both manager and ground worker accounts.

4.1.3 System structure showing activities for cloud-based IoT greenhouse monitoring system

A cloud-based IoT greenhouse monitoring system uses soil sensors and environmental sensors. Sensors are installed inside the greenhouse in order to collect soil and environmental data relevant to the growth of the crop, such as temperature, humidity and soil moisture. These sensors constitute a wireless sensor network to collect environmental and soil information from the greenhouse. In addition, sensed data are sent to the cloud to be analyzed and stored if the received data exceed or are below the threshold message sent to via twitter account to the user. Figure 10 below shows the system structure for a proposed cloud-based IoT greenhouse monitoring system.

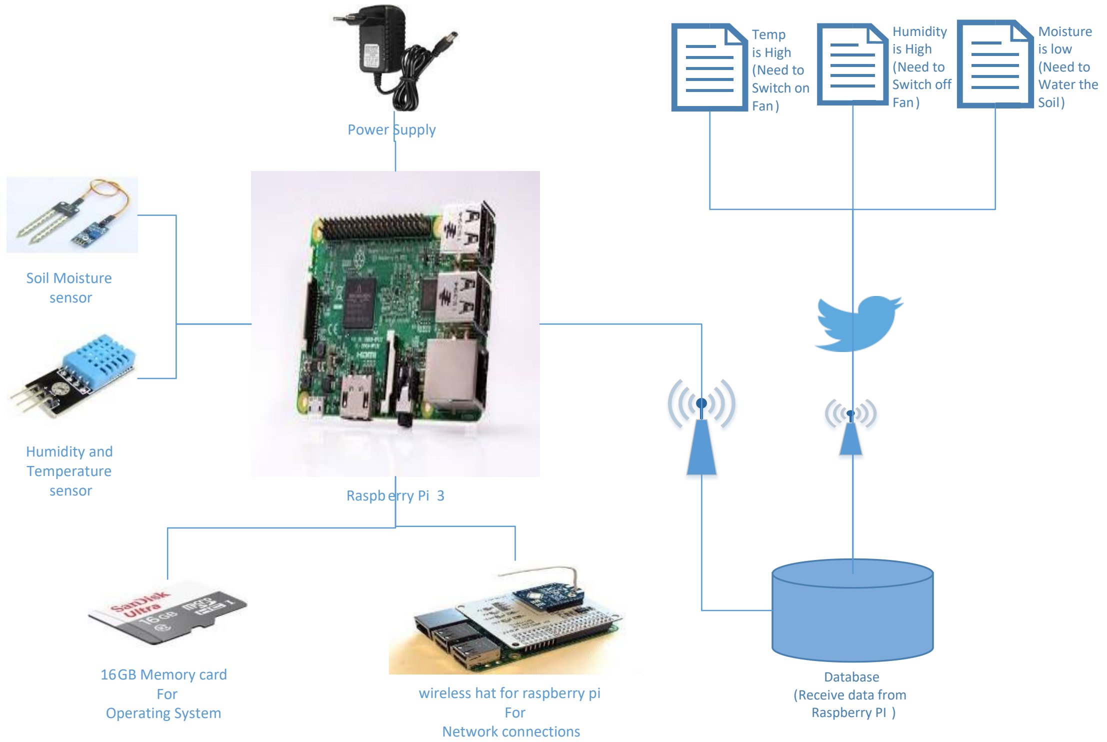

Figure 10: System overview for cloud-based IoT greenhouse monitoring

A generic life cycle model for IoT devices, adapted from Rahman, Ozcelebi and Lukkien (2018), was used to explain this system structure. A generic life cycle model for IoT is used to demonstrate the system development and functionality of cloudbased IoT greenhouse-monitoring system. This cycle has 3 life cycles: IoT device life cycle; IoT service life cycle; and IoT application life cycle. The hardware design of the system includes soil moisture, and Dht11 for temperature and humidity sensor, and raspberry pi as the microcontroller. According to Rahman, Ozcelebi and Lukkien (2018), this stage pertains to the installation and updates of device hardware and its initial software.

Soil moisture sensor

This sensor senses the moisture content based on capacitive effect; it consists of a hygroscopic dielectric material sandwiched between a pair of electrodes forming a small capacitor; in this case, it is the soil acting as a dielectric material. Soil moisture sensor measures the water content in the soil. It uses the property of the electrical resistance of the soil. The sensor is inserted in the soil to sense the existence of water. An electric current can easily pass through if there is moisture.

Temperature and humidity node

This node senses the temperature and humidity inside the greenhouse using a pyroelectric film for temperature and a hygrometer which is a resistive type humidity sensor. The Dht11 used in this study is a simple, low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air and spits out a digital signal on the data pin (no analog input pins are needed).

Raspberry Pi

Raspberry pi is a credit card size low-cost, high performance computer, which was developed in the United Kingdom by the raspberry pi foundation. The researcher used the model B of the third generation of raspberry pi (Coppock 2015; Kirby, Chapman and Chapman 2018). The raspberry pi is an open-source computing platform consisting of a microcontroller board and a development environment for writing software. Raspberry pi 3 is a versatile solution to the issue of cost, mobility and reliability when it comes to stimulus presentation (Vineela *et al.* 2018). This device is inexpensive, lightweight and highly versatile (Kuziek, Shienh and Mathewson 2017). The Windows 10 IoT core operating system was used.

4.1.3.1 The Production

Rahman, Ozcelebi and Lukkien (2018) mention that production includes manufacturing the hardware and mass deploying the initial software onto the hardware. In this phase, windows 10 IoT core is installed on the raspberry pi through the insertion of 16 GB SD card. The SD memory card is configured in such a way that it acts as a hard drive to the raspberry pi processor. Raspberry pi functions in the same way as standard PC; it needs a keyboard for command entry, a display unit (monitor) and power supply (Vujovic and Maksimovic 2014). Furthermore, the administrator password was created to protect the access; this was done to promote privacy on the system. Rahman, Ozcelebi and Lukkien (2018) mention that the mass deployment of cryptographic material scan is very important for this stage. Security is a major concern in every network to avoid attacks, such as hackers pushing erroneous data into the network; access confidential information; or disabling the entire network (Gubbi *et al.* 2013). A laptop running windows 10 was used to write the code using C#.

4.1.3.2 The installation and commission

This phase prepares the devices for operation and secure communication within the network. Device interfacing is conducted in this phase; the sensor is connected to the raspberry pi through the GPIO (general-purpose input and output). Raspberry pi is plugged to the power supply 5V and is connected to the network using RJ45 cable. A soil moisture sensor was inserted on the pot plant to monitor soil moisture and the results were sent to raspberry pi via GPIO pins. A Dht11 temperature and humidity sensor was placed next to the plant pot to detect temperature levels in the greenhouse environment and the results were sent to the raspberry pi via GPIO pins.

Once the installation of the hardware is done, the commissioning starts to run all the necessary commands. According to Rahman, Ozcelebi and Lukkien (2018), configuring operational parameters during commission allows the device to operate as expected. This process includes configuration of keys and certificates for bootstrapping. As for Garcia-Carrillo and Marin-Lopez (2016), bootstrapping in IoT relates to the process of acquiring configurations and secret keys for authentication and secure communication with other devices in the network. Then the hardware part for the system is ready to function.

4.1.3.3 The production

The IP scanner is used to detect the IP address of the raspberry pi which enables the remote login using the secure shell (SSH) communication protocol. SSH provides a secure channel over an unsecured network in a client-server architecture (Reshma et al. 2018). According to Rahman, Ozcelebi and Lukkien (2018), this phase supports the execution of IoT services and applications by running relevant system components and running system monitoring. The code is formulated on the laptop for data collection. During this stage, the latest update is required to execute the code; after this update, the system can be decommissioned to another network or devices that have reached the end of their lifespan need to be replaced (Rahman, Ozcelebi and Lukkien 2018).

4.2 Design of the architecture model for cloud-based IoT greenhouse monitoring system

The proposed model aimed at developing a cloud-based IoT greenhouse-monitoring system depicted in Figure 11 below, aiming to provide an ideal greenhouse-cropping environment. This model is a three-layer model, namely: sensing layer; network layer; and application layer. According to Zhou, Chen and Chen (2016), the cloud-based system is composed of three-layer architecture, namely: the application layer; business layer; and physical layer. Furthermore, Tzounis *et al.* (2017) state that the structure of IoT is based on the three layers, namely: the perception layer (sensing); the network layer (data transfer); and the application layer (data storage and manipulation).

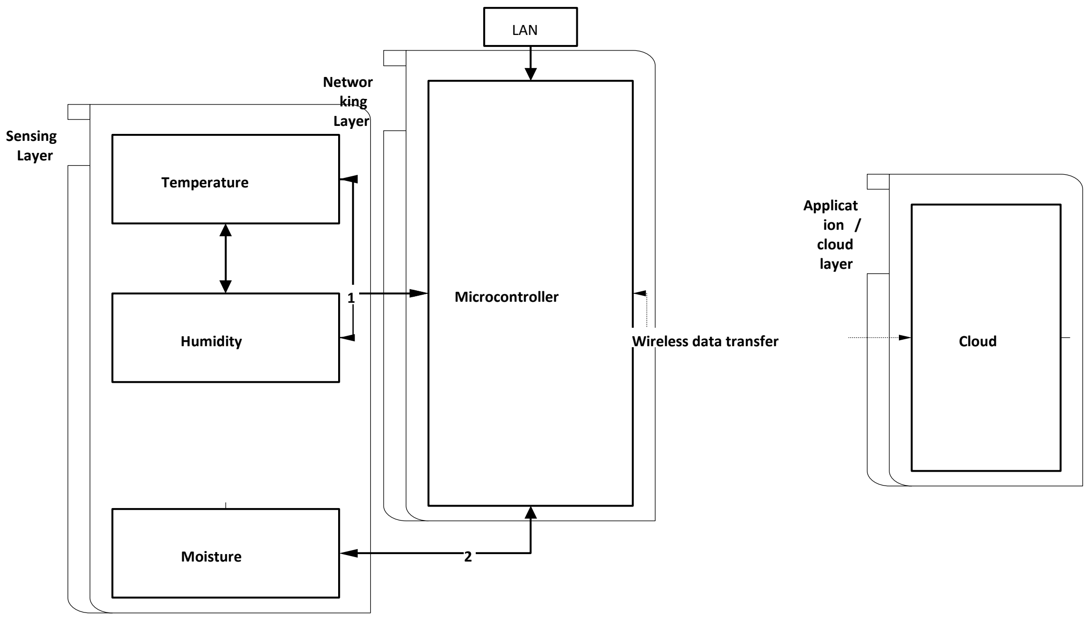

Figure 11: Proposed architecture model

4.2.1 Sensing layer

At this layer, heterogeneous sensors are deployed inside the greenhouse to collect soil moisture, humidity and temperature. Raspberry pi is operating as a standard computer in this study to connect the power supply of 5 V, receiving all the results from the sensors displaying them on the monitor inside the greenhouse. The Internet connectivity was through an Ethernet/LAN cable and or through a USB dongle (WiFi connectivity). The architecture below (see Figure 12) shows the overview of the installation and commission stages.

Dht11: *pin = GpioController.GetDefault().OpenPin(GPIO27, GpioSharingMode.Exclusive); dhtSensor = new Dht11(pin, GpioPinDriveMode.Input);* Soil Moisture: *pinMoistureSensor = gpioController.OpenPin(GPIO17); pinMoistureSensor.SetDriveMode(GpioPinDriveMode.Input);*

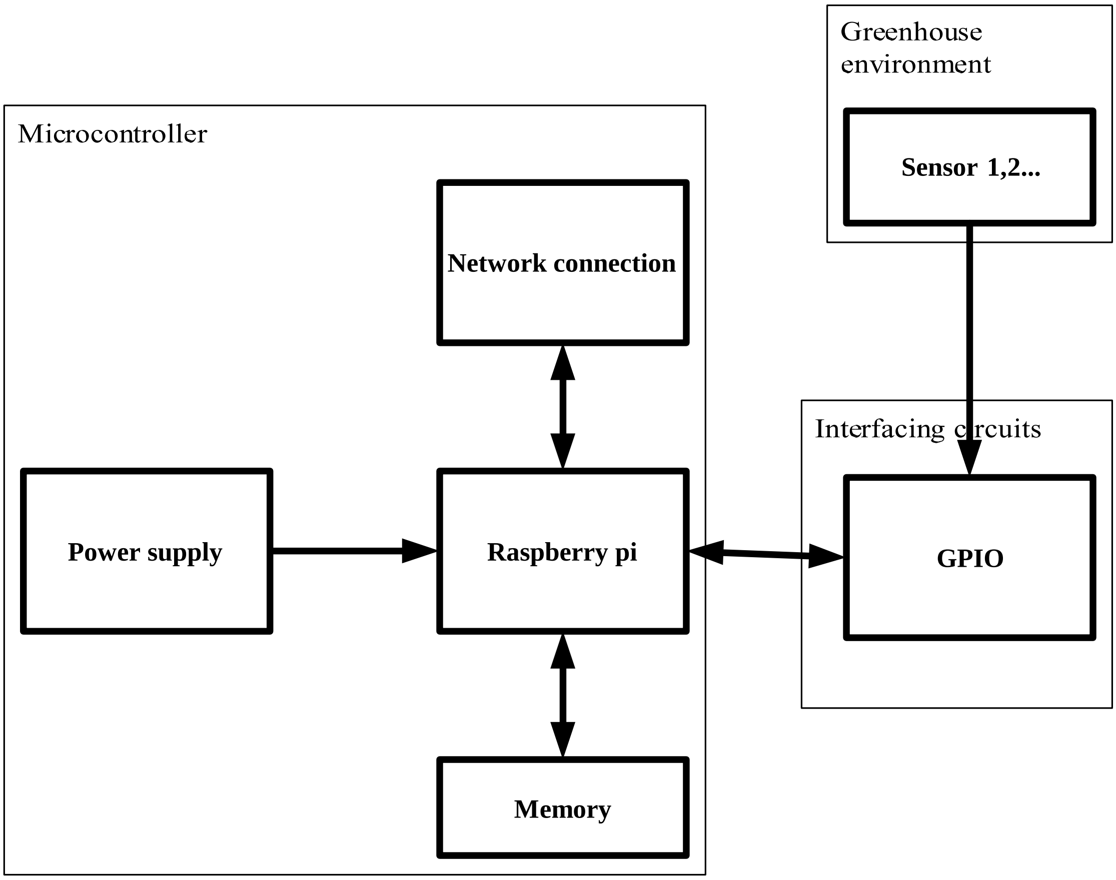

Figure 12: Proposed overview architecture for sensing layer

4.2.2 Networking layer

At this layer, after the microcontroller interacted with sensors and with physical objects or environment and the communication is successful, the microcontroller starts to communicate with the nearest gateway and the data are forwarded to cloud for analysis, processing, dissemination and storage. For this process to be performed successfully, specific communication protocols need to be utilized, as Ray (2018) mentions that IoT communication protocols generally work in the data link layer, network layer, transport layer and application layer. Such protocols include ZigBee, ONE-NET, Sigfox, WirelessHART, ISA100.11a, and 6LowPan. For this study, wirelessHart is used.

1.*private static string s_deviceConnectionString =*

"HostName=ProgressIoTHub.azuredevices.net;DeviceId=fezhatpc;SharedAccessKey= DUIfdy+IS4AXUfYB5xkusX9+Dz k76XWhcWlglFKjoDY=";

2. private static TransportType s_transportType = TransportType.Amqp;

4.2.3 Application layer

This layer is responsible for facilitating the end-users' capability of accessing the sensed data (see Figure 13). This is succeeded by executing several services including, but not limited to, data storage, data analytics and data visualization in accumulation to providing a suitable application program interface (API) and software tools through which the end user can access the data. The application layer cloud server has a large database at its core that can accommodate a huge amount of data relayed through the network layer from the sensing layer. The end user can be able to send the message back to the network layer by typing something.

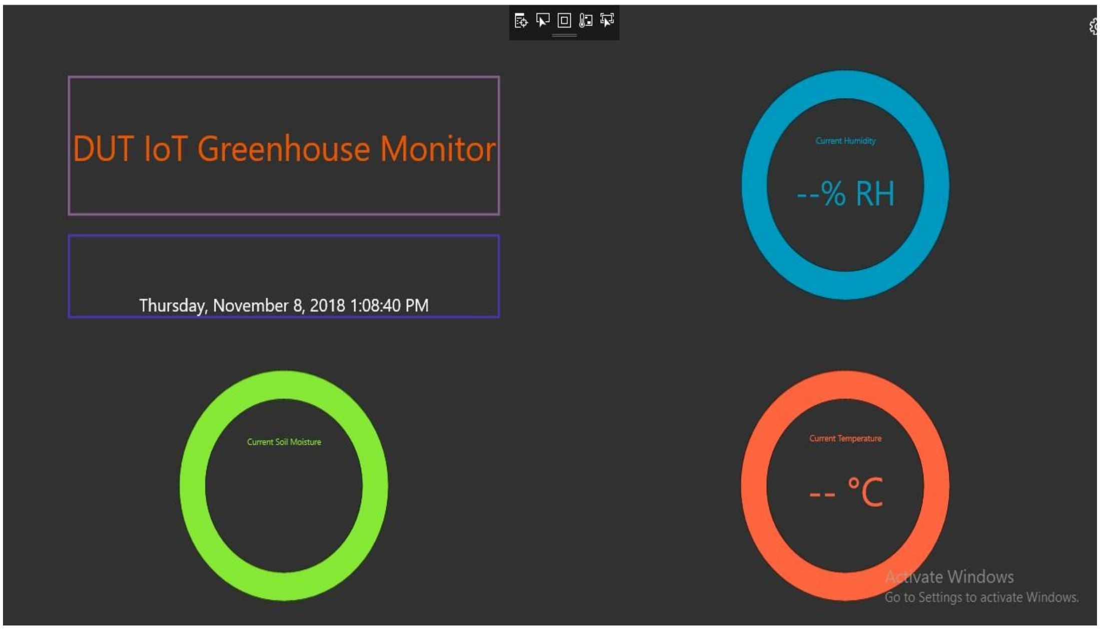

Figure 13: Local end user image

4.3 The implementation of cloud-based IoT greenhouse monitoring system

The main aim of this study was to develop a cloud-based IoT greenhouse monitoring system that is wireless, cost-effective, easily accessible and provide unlimited data storage space to farmers. The development of a cloud-based IoT greenhouse monitoring system followed a novel life cycle model for web-based application development (Huang *et al.* 2010). The development of this system adopted a Microsoft visual studio as an integrated development environment. A visual studio is the development environment for creating software solution targeted for the Microsoft .Net platform. Visual studio is used to write the application code, build the solution, debug and deploy the application to other computing devices (Oancea and Donald 2002; Watson *et al.* 2012). The Microsoft .Net platform is a development and execution environment that allows different programming languages and libraries to work together (Mayo 2008). In this way, it is easier to build, manage, deploy and integrate with other networked systems. For this study, it was used to build universal windows platform application using C#. It incorporates many of the best features from other languages with the aim to clear up their problems (Watson *et al.* 2012), and C++ programing languages which are part of .Net platform.universal applications which are the programs that can be used across all compatible Microsoft windows devices, including personal computer (PC), tablets, smartphones and IoT. Universal Windows Platform (UWP) application. These applications require a development environment with installed universal windows app development tools, which include SDK tools and device emulator.

Figures 14 and 15 represent the system prototype, the temperature and humidity sensor node, soil moisture sensor node, with the water jar hardwired into the raspberry pi. The sensors nodes are interfaced using the standard peripheral interface.

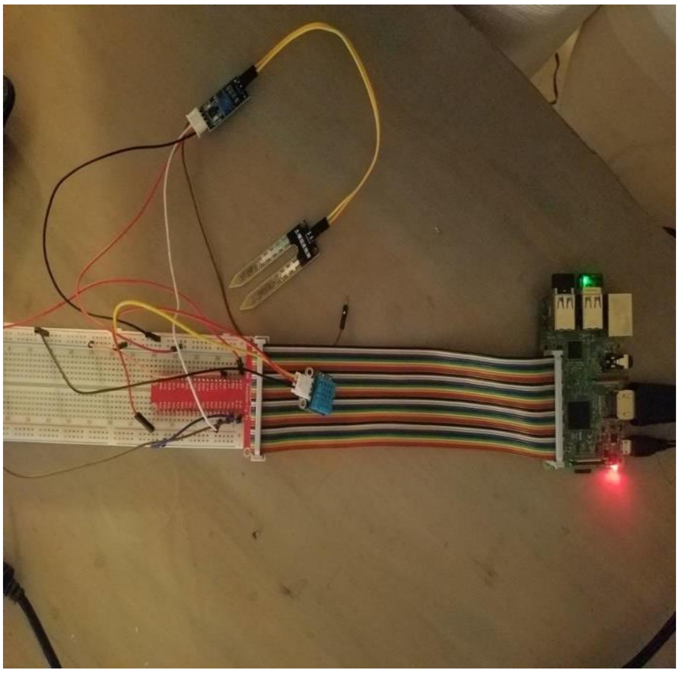

Figure 14: System prototype

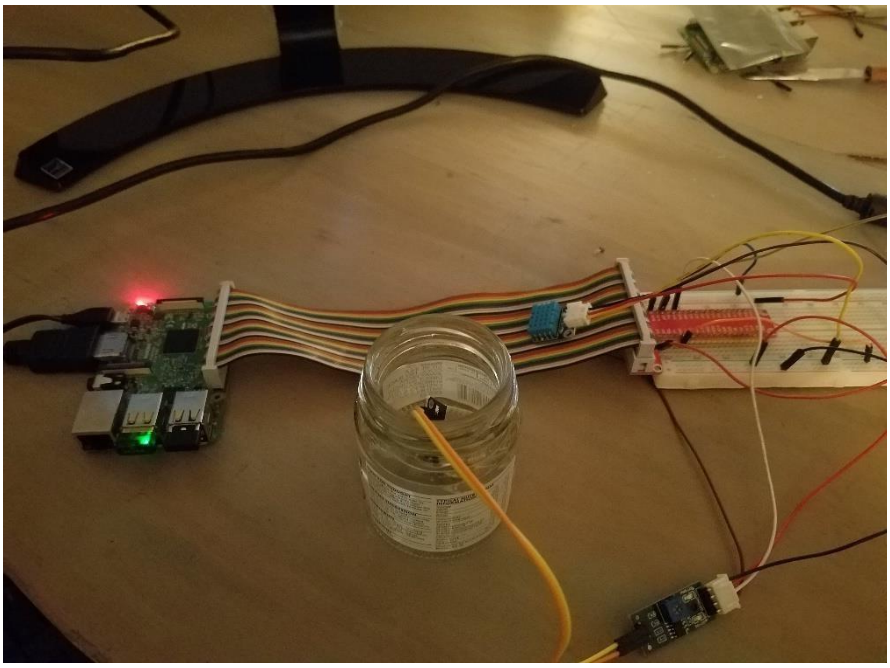

Figure 15: System prototype with water jar to test moisture status

4.3.1 Functional description of cloud-based IoT greenhouse monitoring on a local station

This monitoring system allows a farmer to view the results on the site using a desktop. Figure 16 shows how the results should be displayed on site. The blue circle is displaying humidity in percentage (%), the orange circle displays temperature in degrees C, the green circle displays wet/dry soil moisture, the navy rectangle displays the day, date and time the results were captured, and the purple rectangle displays the location of the system.

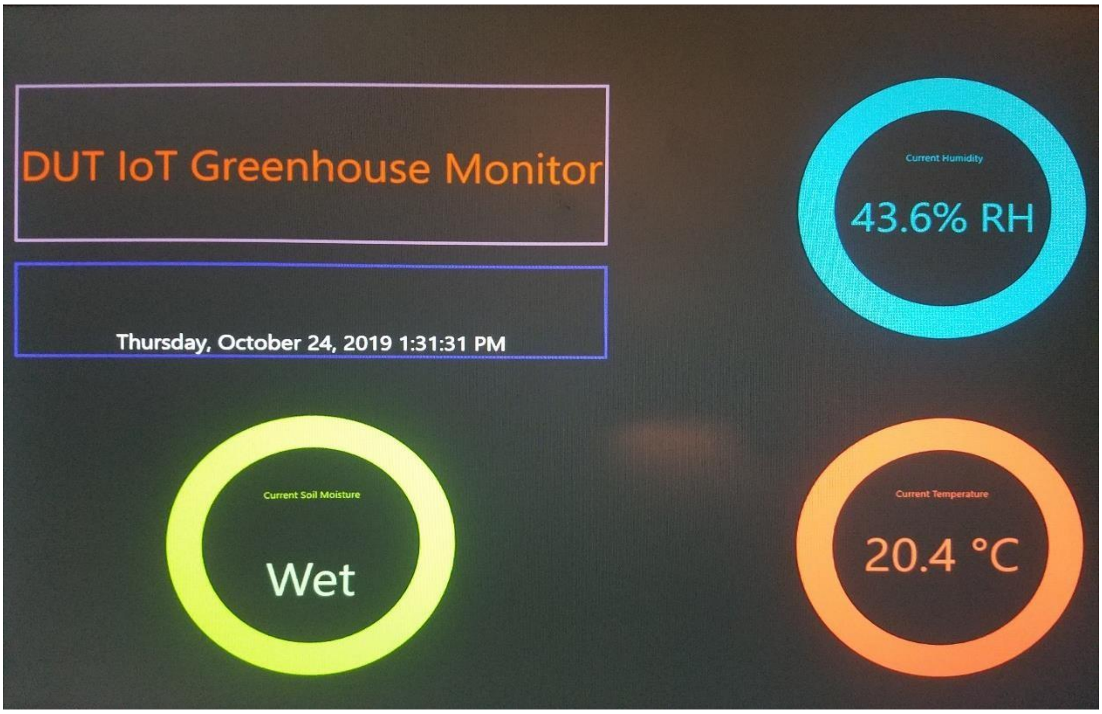

Figure 16: Local image with wet results

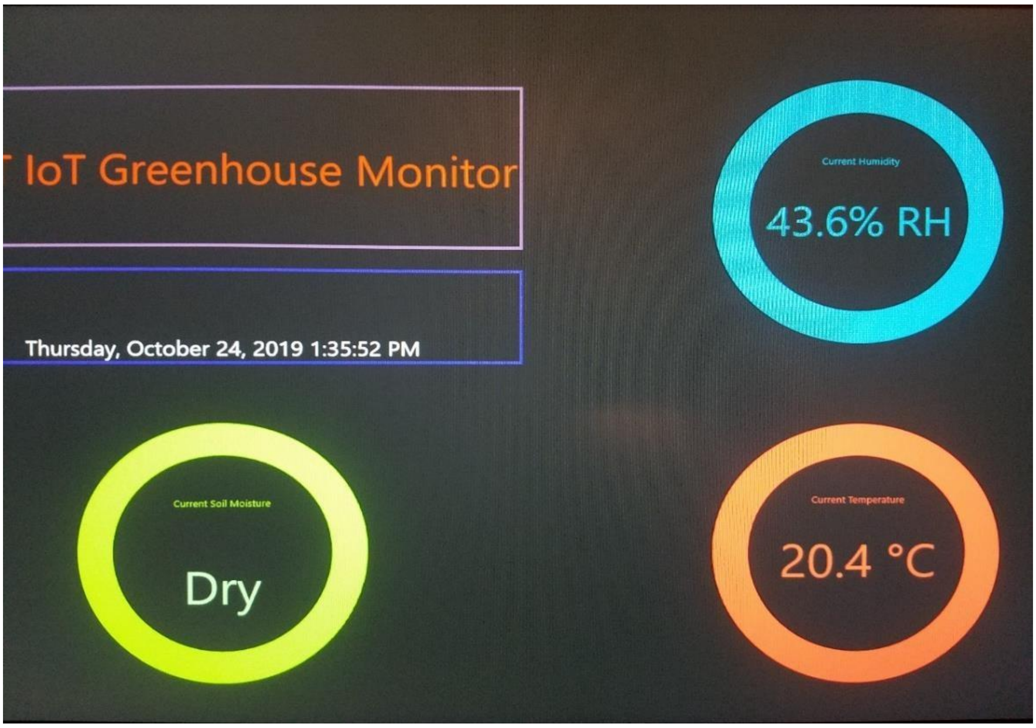

Figure 17: Local image with dry results

4.3.2 Functional description of cloud-based IoT greenhouse monitoring system on the cloud

The file DUTIOTGRHMO.log contains the data posted to the ProgressIoTHub in Windows Azure. The ConnectionString in the C# code contains the ConnectionString that points to the telemetry data (temperature, humidity and moisture). Each data point has a timestamp, a messaged and the telemetry data. Appendix 1 shows the data points and telemetry data collected during prototyping. The screenshots below (see Figure 18) shows the messages that were received by the IoT Hub:

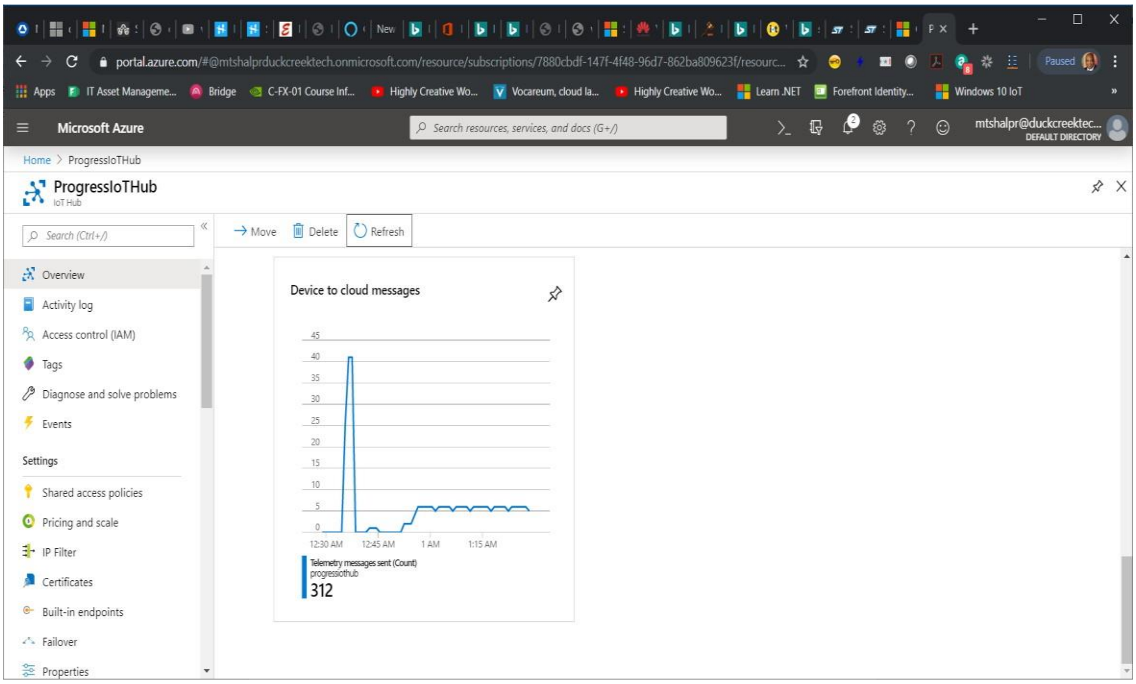

Figure 18: IoT hub results

The screenshot below (see Figure 19) is an expansion of the device to cloud messages above. The timestamps are different because Windows Azure uses the timestamp of the cloud environment.

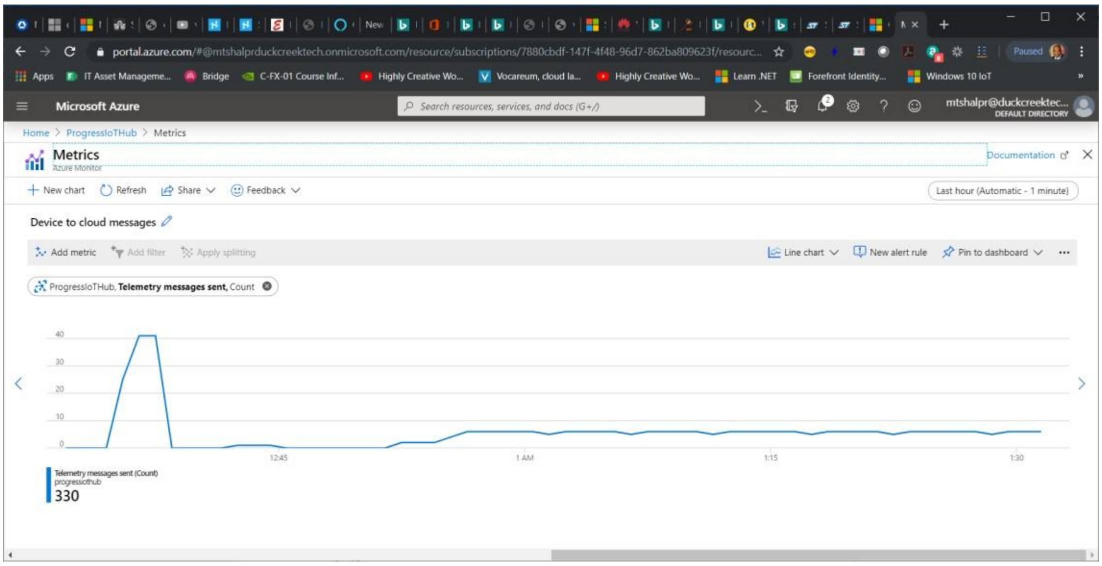

Figure 19: Device to cloud messages

4.4 Evaluation

This section demonstrates and evaluates the artefacts of the cloud-based IoT greenhouse monitoring system. This system is designed to monitor the greenhouse climate environment. Peffers *et al.* (2007) divide what others call evaluation into two activities, i.e., demonstration and evaluation. The demonstration is like a light-weight evaluation to demonstrate that the artefact feasibly works to "solve one or more instances of the problem", while, evaluation in DSR is concerned with the evaluation of design science outputs including theory and artefacts (Venable, Pries-Heje and Baskerville 2016). Nunamaker Jr, Chen and Purdin (1990) identified a number of methods for evaluation or what they termed experimentation. These included computer and lab simulations, field experiments and lab experiments. However, Nunamaker Jr, Chen and Purdin (1990) regard these activities as experimentation and observation. Venable *et al.* (2011) called these activities artificial evaluation and naturalistic evaluation, openly recognizing the evaluative nature of the observation activity. The artificial evaluation includes laboratory experiments, field experiments, simulations, criteria-based analysis, theoretical arguments and mathematical proofs. For this study, experimentation is employed.

The two most important categories of evaluation are defined by Venable, Pries-Heje and Baskerville (2016) as formative vs summative evaluation and ex-ante vs ex-post evaluation. However, for this study, formative vs summative is omitted, and ex-ante vs ex-post is appointed. The system measurements were adopted from the study of Prat, Comyn-Wattiau and Akoka (2014) which informs the researcher on what is to be evaluated in IS artefacts and how the evaluation should be conducted.

Evaluation of design artefacts is a crucial part of design science research (March and Smith 1995; Vaishnavi and Kuechler 2004; Hevner and Chatterjee 2010). According to Oates (2005), evaluation creates a platform for identifying strengths and weaknesses of the artefact which can lead to the modification of the artefact. Venable, Pries-Heje and Baskerville (2016) posit that there are at least six main purposes for evaluation in DSR. Firstly, the purpose is to determine how well a design artefact achieves its expected main aim, secondly, to determine the quality and the knowledge gain from the artefact, thirdly, to compare the new artefact with other artefacts to determine whether the artefact makes an improvement, fourthly, to evaluate the utility, style, quality and efficacy rigorously. Vaishnavi and Kuechler (2004), Hevner (2007) and Oates (2005) posit that an artefact might be evaluated for another impact, and the evaluation can further elaborate on the knowledge outcomes by finding out why an artefact works or does not work. The following section explains the evaluation method that was employed to evaluate the artefacts and their design of the study.

4.4.1 Evaluation method

The evaluation of DSR artefacts in IS can be carried out in different methods (Peffers *et al.* 2007). However, it is argued that the most suitable evaluation method would be contingent on the nature of the artefact itself and the problem is solved. The artefact of this study is categorized as an instantiation - a system (Peffers *et al.* 2007; Hevner and Chatterjee 2010). Prat, Comyn-Wattiau and Akoka (2014) argue that IS artefacts, whereby a DSR process was followed, are regarded as a system. In the common IS literature, evaluation is commonly regarded from these two perspectives: *ex-ante* perspective and *ex-post* perspective (Prat, Comyn-Wattiau and Akoka 2014; Venable, Pries-Heje and Baskerville 2016). The *ex-ante* perspective is the evaluation of uninstantiated artefacts (Venable, Pries-Heje and Baskerville 2016). In addition, Prat, Comyn-Wattiau and Akoka (2014) elaborate that *ex-ante* evaluation is for the purpose of deciding which of the several competing technologies should be acquired or adopted. It happens before design and construction begin. Whereas*Ex-post* is the evaluation of instantiated artefacts (Prat, Comyn-Wattiau and Akoka 2014). For this study, only ex-post is employed. For an *ex-post* perspective, the actual system prototype performance was evaluated. Figure 20 demonstrates the points where each evaluation perspective was considered.

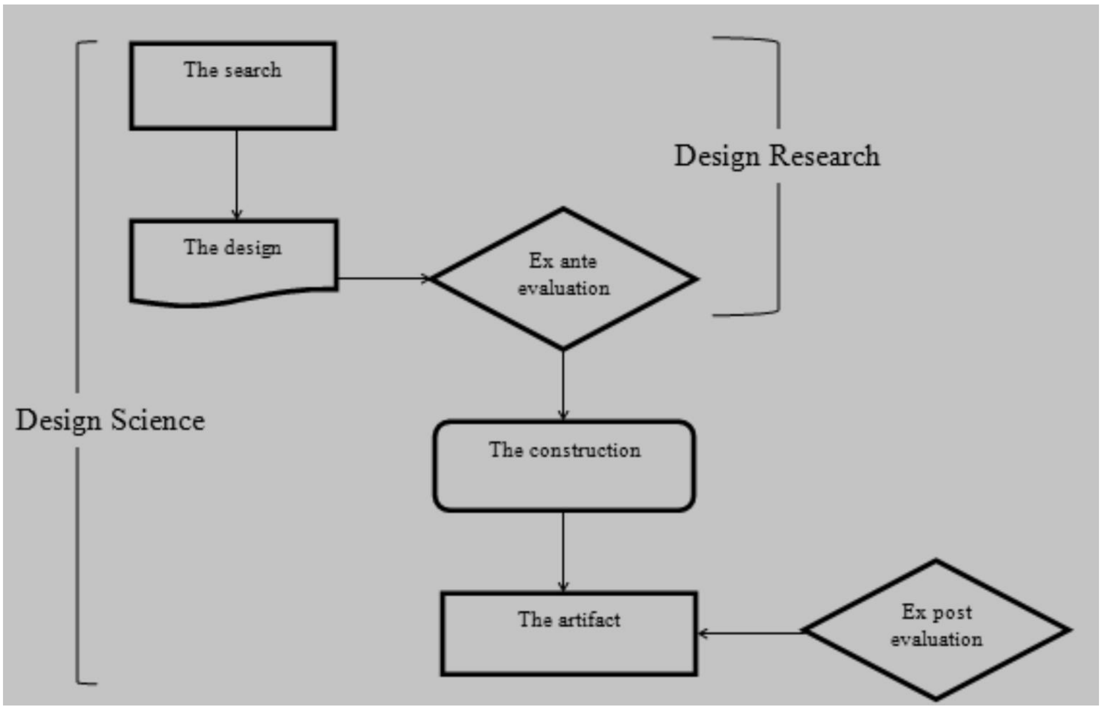

Figure 20: The ex-ante and ex-post perspectives (Prat, Comyn-Wattiau and Akoka 2014)

Two evaluation processes were adopted by Venable, Pries-Heje and Baskerville (2016), namely, artificial evaluation and naturalistic evaluation. The artificial evaluation process refers to a non-realistic evaluation that involves laboratory experiments, mathematical proofs and simulations (Venable, Pries-Heje and Baskerville 2016). On the other hand, the naturalistic evaluation process involves engaging directly with users in a real-life environment, observing them while they're using the system, and asking them questions in order to get an understanding of their behaviour (Sun and Kantor 2006). For this study, the artificial evaluation is omitted; only the ex-post perspective of the naturalistic evaluation process was employed.

Cronholm and Goldkuhl (2003) state there are three types of strategies that can be used to evaluate artefacts: as a goal-based evaluation, as a goal-free evaluation, and as a criteria-based evaluation. However, for this study, only the criterion-based evaluation was adopted.

4.4.2 Criteria based evaluation strategy

March and Smith (1995) posit that, "evaluation of instantiations considers the efficiency and effectiveness of the artefact and its impacts on the environment and its users". There are several criteria-based approaches, such as checklists, heuristics, principles or quality ideas (Cronholm and Goldkuhl 2003). For this study, the hierarchy of evaluation criteria was adopted from Prat, Comyn-Wattiau and Akoka (2014). In their study, Prat, Comyn-Wattiau and Akoka (2014) built a hierarchy of evaluation criteria which are coupled to five system dimensions. Out of the five system dimensions, only 3 were of interest in the evaluation of a cloud-based IoT greenhouse monitoring system. These dimensions are goal, environment and activity. The evaluation criteria which were used in this study were taken from their hierarchy; however, the evaluation criteria that did not address the objective of this study were prohibited. The objective of this study which is centred on evaluation is to evaluate the system for performance. This objective was aligned with the goal, environment and activity of the system and mapped to *effectiveness* and *understandability*. Conversely, for these evaluation criteria to apply to this work, they had to be related to the evaluation criteria in the hierarchy of Prat, Comyn-Wattiau and Akoka (2014). Effectiveness was reflected as efficacy. This union is admitted by Hevner *et al.* (2004), where they articulated that these terms are interchangeable. In the hierarchy of Prat *et al*. (2014), efficacy is organized along the goal system dimension. Efficacy is defined as the degree in which artefacts archive the expected results and can be applied to machines and automata (Carvalho 2012). Understandability refers to the user comprehension towards the system and, in this study, the criteria were considered as consistency with people (Prat, Comyn-Wattiau and Akoka 2014).

4.5 Chapter summary

This chapter presented the development, implementation and the evaluation of cloud-based IoT greenhouse monitoring system. Development of this system is presented in the first section of this chapter, demonstrating how the user requirements of this system were expressed and employed methods to prompt these user requirements.

The requirements evaluation was essentially conducted, in order to find out which functions will be essential for the cloud-based IoT greenhouse monitoring system. These functions were presented by means of UML to optimize understanding of the design features of the proposed system. The implementation of the system section demonstrates how the system works by prototyping the system. The last section of this chapter presents artefact evaluation called the cloud-based IoT greenhouse monitoring system. The artefact evaluated is used to monitor the greenhouse environment. It uses sensors to collect temperature, humidity and soil moisture levels inside the greenhouse.

This artefact is presented to ground workers and nursery managers, and it was evaluated on the third research objective, which is: to evaluate the system performance. The focus group method was used to discuss the effectiveness of the system and to check how understandable the system is to the users. The next chapter summarizes the current research study, presents the overall contributions made, and recommends directions for future research.

CHAPTER 5: CONCLUSION

This chapter provides a summary of the complete study and discusses the completion of the research aim and the research hypothesis. A discussion on the study concentrating on the contributions made is presented. The limitations of this research, as well as the lessons learned in carrying out this study and the recommendations for possible further research work arising from the study, are expressed. A wireless cloud-based IoT greenhouse monitoring system was developed to remotely monitor and automate the control of temperature, humidity and soil moisture of the greenhouse using a prototype. The system has a number of attractive features, including low-cost, compact, scalable, easy to customize, easy to deploy, easy to maintain and finally the system is very useful in managing the use resources efficiently, which is the focal hypothesis of this study. The system was developed successfully using IoT and cloud computing. The system was able to monitor usage and store the data in the cloud for monitoring trends thus leading to efficient water usage. However, due to the number of limitations the system performance was not conducted and that is left as an exercise for other researchers. The following sections explain more about the number of limitations as well as the recommendations for future studies.

5. 1 Limitations

The main limitations of this research are as follows: The difficulty of this project was given by the variety of confines types. The system could not achieve one of the aims of system which was the evaluation of the system by nursery workers due to the following issues:

- DUT's nursery has no Wi-Fi or LAN (Local Area Network) internet connection. The researcher found it hard to work in this environment. However, the experiment was conducted in a room setup with all sensors connected to the relevant places.
- Limited power points: The Raspberry Pi used in this study needs power in order to function. The DUT greenhouse has no power socket close or inside the greenhouse, that gave the researcher a difficult time to connect the Raspberry Pi inside the greenhouse.

- Due to restricted room space there was no need to use many Raspberry Pi's, only one Raspberry Pi was used instead of multiple Pi's to cover the whole room.

5. 2 The recommendations for possible further research work arising

The developed monitoring system can be improved in many ways. Therefore, the following directions might be interesting and important in future research:

- To further the scope of this work, the designed framework for this study, as well as the infrastructure, can be evaluated.
- The testing of this system against other system can be recommended for future studies.
- The proposed system can use any available storage applications, such as google Documents and Microsoft Office Suite;
- This work can be implemented using Arduino board and the water sensor can be modified in such a way that it would display the actual water value available in the soil;
- The proposed system could also be tied to an automatic irrigation system; once system detects that the soil is dry, it would trigger the automatic irrigation system to begin watering.
- The output of this work was to design and test the system, the next step/future work will be to calibrate and miniaturize the system for real world application.
- The researcher recommends that the farmers can use the experimental guide provided below to conduct the experience:
  - For Wi-Fi/ internet coverage in a greenhouse a signal booster can be used to extended the signal ,
  - To cover the whole greenhouse 4 Raspberry PI should be used,
  - 4 temperature and humidity, and soil moisture sensors
  - The code for this experiment is attached at Appendix 2

The system could not send data to the cloud via the DUT network due to DUT has closed ports access. The researcher had to use a private network.

5.3 Summary

This study aimed to develop a cloud-based IoT greenhouse monitoring system that could be used for monitoring greenhouse environments, specifically focusing on temperature, humidity, and soil moisture. This study emphasizes the use of wireless technology through the convergence of cloud computing and the Internet of Things, based on best practices, rather than wired technology. As reported in the literature review of this dissertation, many authors have used wired technology to develop a greenhouse environment monitoring system. However, wired technology is not conducive to the greenhouse environment in terms of installation and high costs. For this study, small portable sensors were used. They were communicating like human beings in real-time to send notifications to the greenhouse worker and to send data to the cloud for storage as reference for future purposes. This dissertation has addressed the processes leading to the design and development of the proposed system. The proposed system could be a solution to the many issues faced with the usage of the current system, as discussed in Chapter 1. The proposed system is developed as a simple and cost effective one.

REFERENCES

Abbasi, A. Z., Islam, N. and Shaikh, Z. A. 2014. A review of wireless sensors and networks' applications in agriculture. *Computer Standards & Interfaces*, 36 (2): 263270.

Ahonen, T., Virrankoski, R. and Elmusrati, M. 2008. Greenhouse monitoring with wireless sensor network. In: Proceedings of *Mechtronic and Embedded Systems and Applications, 2008. MESA 2008. IEEE/ASME International Conference on*. Beijing, China, Oct. 12-15 2008. IEEE, 403-408.

Ai, W. and Chen, C. 2011. Green house environment monitor technology implementation based on android mobile platform. In: Proceedings of *Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), 2011 2nd International Conference on*. IEEE, 5584-5587.

Akkaş, M. A. and Sokullu, R. 2017. An IoT-based greenhouse monitoring system with Micaz motes. *Procedia Computer Science*, 113: 603-608.

Akyildiz, I. F. and Stuntebeck, E. P. 2006. Wireless underground sensor networks: Research challenges. *Ad Hoc Networks*, 4 (6): 669-686.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y. and Cayirci, E. 2002. Wireless sensor networks: a survey. *Computer networks*, 38 (4): 393-422.

Al-Debei, M. M. 2010. The design and engineering of innovative mobile data services: An ontological Framework founded on business model thinking. Brunel University, School of Information Systems, Computing and Mathematics.

Aqeel-ur-Rehman, Z., Shaikh, N. A. and Islam, N. 2010. An integrated framework to develop context-aware sensor grid for agriculture. *Australian Journal of Basic and Applied Sciences*, 4 (5): 922-931.

Ari, A. A. A., Gueroui, A., Labraoui, N. and Yenke, B. O. 2015. Concepts and evolution of research in the field of wireless sensor networks. *arXiv preprint arXiv:1502.03561*.

Ashton, K. 2009. That 'internet of things' thing. *RFID journal*, 22 (7): 97-114.

Atzori, L., Iera, A. and Morabito, G. 2010. *The internet of things: A survey*. *Computer networks*, 54 (15): 2787-2805.

Atzori, L., Iera, A., Morabito, G. and Nitti, M. 2012. The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization. *Computer networks*, 56 (16): 3594-3608.

Azaza, M., Tanougast, C., Fabrizio, E. and Mami, A. 2016. Smart greenhouse fuzzy logic based control system enhanced with wireless data monitoring. *ISA transactions*, 61: 297-307.

Baviskar, J., Mulla, A., Baviskar, A., Ashtekar, S. and Chintawar, A. 2014. Real time monitoring and control system for green house based on 802.15. 4 wireless sensor network. In: Proceedings of *Communication Systems and Network Technologies (CSNT), 2014 Fourth International Conference on*. IEEE, 98-103.

Benbasat, I. and Zmud, R. W. 2003. The identity crisis within the IS discipline: Defining and communicating the discipline's core properties. *MIS quarterly*: 183-194.

Biswas, A. R. and Giaffreda, R. 2014. IoT and cloud convergence: Opportunities and challenges. In: Proceedings of *2014 IEEE World Forum on Internet of Things (WFIoT).* IEEE, 375-376.

Botta, A., De Donato, W., Persico, V. and Pescapé, A. 2016. Integration of cloud computing and internet of things: a survey. *Future Generation Computer Systems*, 56: 684-700.

Pengzhan, C. and Baifen, L. 2010. Construction of intelligent greenhouse control system based on CAN bus. *International Conference on Computer Application and System Modeling*, 10: 631-634,.

Carvalho, J. Á. 2012. Validation criteria for the outcomes of design research. In: Proceedings of *Pre-ECIS workshop on IT Artefact Design & Workpractice Intervention*. June 10, 2012, Barcelona

Castilla, N. 2013. *Greenhouse technology and management*. Cabi.

Champagne, C., White, J., Berg, A., Belair, S. and Carrera, M. 2019. Impact of Soil Moisture Data Characteristics on the Sensitivity to Crop Yields Under Drought and Excess Moisture Conditions. *Remote Sensing*, 11 (4): 372.

Charalabidis, Y., Zuiderwijk, A., Alexopoulos, C., Janssen, M., Lampoltshammer, T. and Ferro, E. 2018. The Multiple Life Cycles of Open Data Creation and Use. In: *The World of Open Data*. Springer, 11-31.

Chavan, C. and Karande, M. P. 2014. Wireless monitoring of soil moisture, temperature & humidity using zigbee in agriculture. *International Journal of Engineering Trends and Technology (IJETT)*–Volume 11.

Chawla, L. 2016. *Growing up in an urbanizing world*. Routledge.

Chung, L. 1991. Representation and utilization of non-functional requirements for information system design. In: Proceedings of *International Conference on Advanced Information Systems Engineering*. Springer, 5-30.

Cook, J. A., Kliewer, W. and Lider, L. A. 1974. *General viticulture*. Univ of California Press.

Coppock, J. 2015. Development of a Raspberry Pi based, SDI-12 sensor environmental data logger. In Obtaining credit in Australia. University of Southern Queensland

Cronholm, S. and Goldkuhl, G. 2003. Strategies for information systems evaluationsix generic types. *Electronic Journal of Information Systems Evaluation*, 6 (2): 65-74.

Damas, M., Prados, A., Gómez, F. and Olivares, G. 2001. HidroBus® system: fieldbus for integrated management of extensive areas of irrigated land. *Microprocessors and Microsystems*, 25 (3): 177-184.

Dan, L., Xin, C., Chongwei, H. and Liangliang, J. 2015. Intelligent agriculture greenhouse environment monitoring system based on IOT technology. In: Proceedings of *Intelligent Transportation, Big Data and Smart City (ICITBS), 2015 International Conference on*. IEEE, 487-490.

Dash, S. K., Sahoo, J. P., Mohapatra, S. and Pati, S. P. 2012. Sensor-cloud: assimilation of wireless sensor network and the cloud. In: Proceedings of *International Conference on Computer Science and Information Technology*. Springer, 455-464.

Deepika, G. and Rajapirian, P. 2016. Wireless sensor network in precision agriculture: A survey. In: Proceedings of *2016 International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS)*. IEEE, 1-4.

Dhumal, Y. and Chitode, J. 2013. Green house automation using Zigbee and smart phone. *International Journal of Advanced Research in Computer Science and Software Engineering Research Paper,* 3 (5) 495-501.

Dias, P. C., Cadavid, D., Ortega, S., Ruiz, A., França, M. B. M., Morais, F. J., Ferreira, E. C. and Cabot, A. 2016. Autonomous soil moisture sensor based on nanostructured thermosensitive resistors powered by an integrated thermoelectric generator. *Sensors and actuators A: Physical*, 239: 1-7.

Erazo, M., Rivas, D., Pérez, M., Galarza, O., Bautista, V., Huerta, M. and Rojo, J. L. 2015. Design and implementation of a wireless sensor network for rose greenhouses monitoring. In: Proceedings of *Automation, Robotics and Applications (ICARA), 2015 6th International Conference on*. IEEE, 256-261.

Escalona, M. J. and Koch, N. 2004. Requirements engineering for web applications-a comparative study. *J. Web Eng.*, 2 (3): 193-212.

Estrin, D., Govindan, R., Heidemann, J. and Kumar, S. 1999. Next century challenges: Scalable coordination in sensor networks. In: Proceedings of *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*. ACM, 263-270.

Faisandier, A. 2012. Systems opportunities and requirements, vol. 2. *Engineering and Architecting Multidisciplinary Systems*. Sinergy'Com, France.

Frank, U. 2006. *Towards a pluralistic conception of research methods in information systems research*. ICB-research report.

Gallant, J. 2018. *Controlled environment greenhouse*. Google Patents.

Gangasani, J. R. 2016. Soil moisture modeling using artificial neural network. Doctoral Dessertation:, Alabama Argicultural and Mechanical University.

Garcia-Carrillo, D. and Marin-Lopez, R. 2016. Lightweight coap-based bootstrapping service for the internet of things. *Sensors*, 16 (3): 358.

Ghoulem, M., El Moueddeb, K., Nehdi, E., Boukhanouf, R. and Calautit, J. K. 2019. Greenhouse design and cooling technologies for sustainable food cultivation in hot climates: Review of current practice and future status. *Biosystems Engineering*, 183: 121-150.

Gondchawar, N. and Kawitkar, R. 2016. IoT based smart agriculture. *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, 5 (6): 177-181.

Goraya, M. S. and Kaur, H. 2015. Cloud computing in agriculture. *HCTL Open International Journal of Technology Innovations and Research (IJTIR)*, 16: 23211814.

Granda, J., Almanza, M., Fontalvo, J. and Calle, M. 2017. Irrigation Measurement System for Dry Areas Based on WSN. *International Journal of Interdisciplinary Telecommunications and Networking (IJITN)*, 9 (3): 10-20.

Gubbi, J., Buyya, R., Marusic, S. and Palaniswami, M. 2013. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29 (7): 1645-1660.

Hevner, A. and Chatterjee, S. 2010. Design science research in information systems. In: *Design research in information systems*. Springer, 9-22.

Hevner, A. R. 2007. A three cycle view of design science research. *Scandinavian journal of information systems*, 19 (2): 4.

Hevner, R., Daza, R., Englund, C., Kohtz, J. and Fink, A. 2004. Postnatal shifts of interneuron position in the neocortex of normal and reeler mice: evidence for inward radial migration. *Neuroscience*, 124 (3): 605-618.

Horvat, G., Sostaric, D. and Zagar, D. 2012. Remote environmental noise monitoring using wireless multimedia sensor networks. In: Proceedings of *Proceedings of the 30th International Conference Science in Practive*.

Hu, W., Zhang, Y., Huang, B. and Teng, Y. 2017. Soil environmental quality in greenhouse vegetable production systems in eastern China: current status and management strategies. *Chemosphere*, 170: 183-195.

Huang, W., Li, R., Maple, C., Yang, H.-J., Foskett, D. and Cleaver, V. 2010. A novel lifecycle model for Web-based application development in small and medium enterprises. *International Journal of automation and computing*, 7 (3): 389-398.

IoT, T. and Research. 2017. *Smart Farming to be Driven by Growth in Cloud Computing*. Available: https://talkiot.co.za/2017/11/22/smart-farming-to-be-drivenby-growth-in-cloud-computing/ (Accessed

J. Yulong and Y. Jiaqiang. 2011. Design an Intelligent Environment Control System for GreenHouse Based on RS485 Bus. *International Conference on Digital Manufacturing and Automation,*, 2: 361-364.

Jaeger, P. T., Lin, J. and Grimes, J. M. 2008. Cloud computing and information policy: Computing in a policy cloud? *Journal of Information Technology & Politics*, 5 (3): 269-283.

Jain, N., Bhakar, S. and Singhal, R. 2017. A Review of Greenhouse Climate Control Application for Cultivation of Agriculture products. *Int. J. Eng. Trends. Technol,* 46: 305-308.

Jiber, Y., Harroud, H. and Karmouch, A. 2011. Precision agriculture monitoring framework based on WSN. In: Proceedings of *2011 7th International Wireless Communications and Mobile Computing Conference*. Istanbul, Turkey, July 4-8 2011. IEEE, 2015-2020.

Jie, Y., Pei, J. Y., Jun, L., Yun, G. and Wei, X. 2013. Smart home system based on iot technologies. In: Proceedings of *Computational and Information Sciences (ICCIS), 2013 Fifth International Conference on*. IEEE, 1789-1791.

Jiménez, A., Jimenez, S., Lozada, P. and Jiménez, C. 2012. Wireless sensors network in the efficient management of greenhouse crops. In: Proceedings of *Information Technology: New Generations (ITNG), 2012 Ninth International Conference on*. Las Vegas, NV, USA, April 16-18 2012. IEEE, 680-685.

JoSEP, A.D., KAtz, R., KonWinSKi, A., Gunho, L.E.E., PAttERSon, D. and RABKin, A., 2010. A view of cloud computing. *Commun. ACM*, *53*(4), pp.50-58

Kerr, R. A. 1989. Greenhouse skeptic out in cold. *Science*, 246 (4934): 1118-1120.

Khajenasiri, I., Estebsari, A., Verhelst, M. and Gielen, G. 2017. A review on Internet of Things solutions for intelligent energy control in buildings for smart city applications. *Energy Procedia,* 111: 770-779.

Khubisa, F. M. 2017. Developing a frugal information system to support very small enterprise business transactions.

Kim, Y. and Evans, R. 2009. Software design for wireless sensor-based site-specific irrigation. *Computers and electronics in agriculture*, 66 (2): 159-165.

Kirby, J., Chapman, L. and Chapman, V. 2018. Assessing the Raspberry Pi as a lowcost alternative for acquisition of near infrared hemispherical digital imagery. *Agricultural and Forest Meteorology*, 259: 232-239.

Kirkham, M. B. 2014. *Principles of soil and plant water relations*. Academic Press.

Kitpo, N., Kugai, Y., Inoue, M., Yokemura, T. and Satomura, S. 2019. Internet of Things for Greenhouse Monitoring System Using Deep Learning and Bot Notification Services. In: Proceedings of *2019 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 1-4.

Kramer, P. J. 1944. Soil moisture in relation to plant growth. *The Botanical Review*, 10 (9): 525-559.

Kuechler, W., Vaishnavi, V. and Kuechler Sr, W. L. 2007. Design [science] research in IS: a work in progress. In: Proceedings of *Proceedings of the second international conference on design science research in information systems and technology (DESRIST 2007)*. 1-17.

Kundra, V. 2011. Federal cloud computing strategy.

Kuroda, M., Ibayashi, H. and Mineno, H. 2015. Affordable 400MHz long-haul sensor network for greenhouse horticulture. In: Proceedings of *2015 International Conference on Information Networking (ICOIN)*. IEEE, 19-24.

Kuziek, J. W., Shienh, A. and Mathewson, K. E. 2017. Transitioning EEG experiments away from the laboratory using a Raspberry Pi 2. *Journal of neuroscience methods*, 277: 75-82.

Lan, Z. and Ma, L. 2014. *Internet of things based farm greenhouse monitor and alarm management system*. Google Patents.

Larrucea, X., Combelles, A., Favaro, J. and Taneja, K. 2017. Software engineering for the internet of things. *IEEE Software*, (1): 24-28.

Leo, M., Battisti, F., Carli, M. and Neri, A. 2014. A federated architecture approach for Internet of Things security. In: Proceedings of *Euro Med Telco Conference (EMTC), 2014*. IEEE, 1-5.

Li, S., Da Xu, L. and Zhao, S. 2015. The internet of things: a survey. *Information Systems Frontiers*, 17 (2): 243-259.

Liao, M.-S., Chen, S.-F., Chou, C.-Y., Chen, H.-Y., Yeh, S.-H., Chang, Y.-C. and Jiang, J.-A. 2017. On precisely relating the growth of Phalaenopsis leaves to greenhouse environmental factors by using an IoT-based monitoring system. *Computers and electronics in agriculture*, 136: 125-139.

Lino, D. B. S. 2014. A New MAC for Wi-Fi based Underground Networks.

Liu, H., Meng, Z. and Cui, S. 2007. A wireless sensor network prototype for environmental monitoring in greenhouses. In: Proceedings of *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*. Shanghai, China, 21-25 Sept. 2007. IEEE, 2344-2347.

Liu, Y., Dong, B., Guo, B., Yang, J. and Peng, W. 2015. Combination of cloud computing and internet of things (IOT) in medical monitoring systems. *Management*, 8: 12.

Lopez, J., Roman, R., Agudo, I. and Fernandez-Gago, C. 2010. Trust management systems for wireless sensor networks: Best practices. *Computer Communications*, 33 (9): 1086-1093.

Mahesh, D. S., Savitha,S. and Dinesh, K. A. 2014. A cloud computing architecture with wireless sensor networks for agricultural applications. 2: 34-38. Available: http://www.ijcncs,org .Access date 31 January 2020.

Mahmoud, R., Yousuf, T., Aloul, F. and Zualkernan, I. 2015. Internet of things (IoT) security: Current status, challenges and prospective measures. In: Proceedings of *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST).* IEEE, 336-341.

Majone, B., Viani, F., Filippi, E., Bellin, A., Massa, A., Toller, G., Robol, F. and Salucci, M. 2013. Wireless sensor network deployment for monitoring soil moisture dynamics at the field scale. *Procedia Environmental Sciences*, 19: 426-435.

Mancuso, M. and Bustaffa, F. 2006. A wireless sensors network for monitoring environmental variables in a tomato greenhouse. In: Proceedings of *IEEE International Workshop on Factory Communication Systems*. 107-110.

March, S. T. and Smith, G. F. 1995. Design and natural science research on information technology. *Decision support systems*, 15 (4): 251-266.

Mastalerz, J. W. 1977. *The greenhouse environment*.New York, USA. John Wiley & Sons.

Matijevics, I. and Simon, J. 2010. Control of the greenhouse's microclimatic condition using wireless sensor network. *IPSI J TIR*, 6 (2): 35-38.

Mayo, J. 2008. *C# 3.0 Unleashed: With the. NET Framework 3.5*. United States of America.Pearson Education.

Moore, S. 2019. *What Is the Ideal Humidity Level for a Greenhouse?*. Available: http://homeguides.sfgate.com/ideal-humidity-level-greenhouse-77050.html. (Accessed 31 Jnuary 2020)

Morais, R., Cunha, J. B., Cordeiro, M., Serodio, C., Salgado, P. and Couto, C. 1996. Solar data acquisition wireless network for agricultural applications. In: Proceedings of *Electrical and Electronics Engineers in Israel, 1996., Nineteenth Convention of*. IEEE, 527-530.

Morais, R., Valente, A. and Serôdio, C. 2005. A wireless sensor network for smart irrigation and environmental monitoring: A position article. In: Proceedings of *5th European federation for information technology in agriculture, food and environement and 3rd world congress on computers in agriculture and natural resources (EFITA/WCCA)*. 845-850.

Nachidi, M., Benzaouia, A. and Tadeo, F. 2006. Temperature and humidity control in greenhouses using the Takagi-Sugeno fuzzy model. In: Proceedings of *Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control, 2006 IEEE*. IEEE, 2150-2154.

Ng, I. C. and Wakenshaw, S. Y. 2017. The Internet-of-Things: Review and research directions. *International Journal of Research in Marketing*, 34 (1): 3-21.

Ngu, A. H., Gutierrez, M., Metsis, V., Nepal, S. and Sheng, Q. Z. 2016. IoT middleware: A survey on issues and enabling technologies. *IEEE Internet of Things Journal*, 4 (1): 1-20.

Niehaves, B. and Stahl, B. C. 2006. Criticality, epistemology and behaviour vs.

Design-information systems research across different sets of paradigms. In: Proceedings of *ECIS*.   50-61.

Nunamaker Jr, J. F., Chen, M. and Purdin, T. D. 1990. Systems development in information systems research. *Journal of management information systems*, 7 (3): 89106.

Oancea, G. and Donald, R. P. 2002. *Visual Basic. NET by example*.United States of America. Que Publishing.

Oates, B. J. 2005. *Researching information systems and computing*.California Sage.

Ojha, T., Misra, S. and Raghuwanshi, N. S. 2015. Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges. *Computers and electronics in agriculture*, 118: 66-84.

Oppermann, F. J., Boano, C. A. and Römer, K. 2014. A decade of wireless sensing applications: Survey and taxonomy. In: *The Art of Wireless Sensor Networks*. Springer, 11-50.

Ordonez, R. V. 2016. Demand for safer food in developing countries. University of Maryland, College Park.

Palmer, J. D., Liang, Y. and Want, L. 1990. Classification as an approach to requirements analysis. *Advances in Classification Research Online*, 1 (1): 131-138.

Peffers, K., Tuunanen, T., Rothenberger, M. A. and Chatterjee, S. 2007. A design science research methodology for information systems research. *Journal of management information systems*, 24 (3): 45-77.

Pfeifer, T., Olariu, S. and Fersha, A. 2005. Wireless sensor networks and their applications. *Special issue of Computer Communications*, 28.

Po’ad, F., Ismail, W. and Jusoh, J. 2017. Improving RF Transmit Power and Received Signal Strength in 2.4 GHz ZigBee Based Active RFID System with Embedded Method. In: Proceedings of *IOP Conference Series: Materials Science and Engineering*. IOP Publishing, 012101.

Pohl, K. 2010. *Requirements engineering: fundamentals, principles, and techniques*. Springer Publishing Company, Incorporated.

Prat, N., Comyn-Wattiau, I. and Akoka, J. 2014. Artefact Evaluation in Information Systems Design-Science Research-a Holistic View. In: Proceedings of *PACIS*. 23.

Prathibha, S., Hongal, A. and Jyothi, M. 2017. IOT Based Monitoring System in Smart Agriculture. In: Proceedings of *Recent Advances in Electronics and Communication Technology (ICRAECT), 2017 International Conference on*. IEEE, 81-84.

Privitera, M. B. 2015. Contextual Inquiry Methods. *Contextual inquiry for medical device design*: 47-71.

Rahman, L. F., Ozcelebi, T. and Lukkien, J. 2018. Understanding IoT systems: a life cycle approach. *Procedia Computer Science*, 130: 1057-1062.

Rahman, L. F., Ozcelebi, T. and Lukkien, J. J. 2016. Choosing your IoT programming framework: Architectural aspects. In: Proceedings of *2016 IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*. IEEE, 293-300.

Rajalakshmi, P. and Mahalakshmi, S. D. 2016. IoT based crop-field monitoring and irrigation automation. In: Proceedings of *Intelligent Systems and Control (ISCO), 2016 10th International Conference on*. IEEE, 1-6.

Ramanjaneya, B. 2015. Greenhouse grown tomato (Lycopersicon esculentum L.) To drip irrigation and n-fertigation in semiarid tropics. Professor jayashankar telangana state agricultural university, rajendranagar, hyderabad.

Ramyateja, K. and Kishore, G.S., 2016. Greenhouse Monitoring And Controlling Using Arm. *International Journal of Computer Science and Mobile Computing*, *5*(8), pp.32-37.

Rao, N. M., Sasidhar, C. and Kumar, V. S. 2012. Cloud computing through mobilelearning.International Journal of Advanced Computer Science and Applications (IJACSA)-2010. *arXiv preprint arXiv:1204.1594*,

Kodali, R.K., Jain, V. and Karagwal, S., 2016, December. IoT based smart greenhouse. In *2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)* (pp. 1-6). IEEE.

Ray, B. 2015. *The ZigBee Vs WiFi Battle For M2M Communication*. Available: https://www.link-labs.com/blog/zigbee-vs-wifi-802-11ah (Accessed 31 January 2020)

Ray, P. P. 2018. A survey on Internet of Things architectures. *Journal of King Saud University-Computer and Information Sciences*, 30 (3): 291-319.

Reggio, G. 2018. A UML-based proposal for IoT system requirements specification. In: Proceedings of *Proceedings of the 10th International Workshop on Modelling in Software Engineering*. ACM, 9-16.

Reshma, U., Bangera, P. P., Chethana, H. and Keerthi, D. 2018. *Raspberry Pi based Soil Parameters Monitoring Device using Sensors*. International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 6.887 Volume 6 Issue V, May 2018- Available at www.ijraset.com

Rouphael, Y., Schwarz, D., Krumbein, A. and Colla, G. 2010. Impact of grafting on product quality of fruit vegetables. *Scientia Horticulturae*, 127 (2): 172-179.

Roy, S., Ray, R., Roy, A., Sinha, S., Mukherjee, G., Pyne, S., Mitra, S., Basu, S. and Hazra, S. 2017. IoT, big data science & analytics, cloud computing and mobile app based hybrid system for smart agriculture. In: Proceedings of *2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON)*. IEEE, 303-304.

Rubala, J. I., Anitha, D. and Student, P. 2017. Agriculture Field Monitoring using Wireless Sensor Networks for Improving Crop Production. *International Journal of Engineering Science*, 5216

Ruiz-Garcia, L., Lunadei, L., Barreiro, P. and Robla, I. 2009. A review of wireless sensor technologies and applications in agriculture and food industry: state of the art and current trends. *Sensors*, 9 (6): 4728-4750.

Booch, G., Rumbaugh, J. and Jacobson, I., 1999. *The unified modeling language reference manual* (Vol. 2). Reading: Addison-Wesley.

Rupanagudi, S. R., Ranjani, B., Nagaraj, P., Bhat, V. G. and Thippeswamy, G. 2015. A novel cloud computing based smart farming system for early detection of borer insects in tomatoes. In: Proceedings of *Communication, Information & Computing Technology (ICCICT), 2015 International Conference*.IEEE, pp:1-6.Salleh, A., Aziz, A., Abidin, M. Z., Misran, M. H. and Mohamad, N. R. 2013. Development of greenhouse monitoring using wireless sensor network through ZigBee technology. *International Journal of Engineering Science Invention (IJESI)*, 2 (7): 6-12.

Satyanarayana, G. and Mazaruddin, S. 2013. Wireless Sensor Based Remote Monitoring System for Agriculture Using ZigBee and GPS. In: Proceedings of *Conference on Advances in Communication and Control Systems*. India, Atlantis Press, 110-114.

Shaker, M. and Imran, A. 2013. Greenhouse Micro Climate Monitoring Based On WSN with Smart Irrigation Technique. *World Academy of Science, Engineering and Technology, International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 7 (12): 1566-1571.

Shamshiri, R. and Ismail, W. I. W. 2013. A review of greenhouse climate control and automation systems in tropical regions. *J. Agric. Sci. Appl*, 2 (3): 176-183.

Shenoy, J. and Pingle, Y. 2016. IOT in agriculture. In: Proceedings of *Computing for Sustainable Global Development (INDIACom), 2016 3rd International Conference*. IEEE, 1456-1458.

Sindhanaiselvan, K. and Mekala, T. 2014. A survey on sensor cloud: architecture and applications. *International journal of P2P network trends and technology (IJPTT)*, 6: 49-53.

Song, J. 2010. Greenhouse monitoring and control system based on zigbee wireless senor network. In: Proceedings of *Electrical and Control Engineering (ICECE), 2010 International Conference*. IEEE, 2785-2788.

Sørensen, C., Pesonen, L., Bochtis, D., Vougioukas, S. and Suomi, P. 2011. Functional requirements for a future farm management information system. *Computers and electronics in agriculture*, 76 (2): 266-276.

Stefko, J. and Kollar, M. 2015. Interaction of materials and humidity of wood based construction walls. *Annals of Warsaw University of Life Sciences-SGGW. Forestry and Wood Technology*, 90.

Stipanicev, D. and Marasovic, J. 2003. Networked embedded greenhouse monitoring and control. In: Proceedings of *Control Applications, 2003. CCA 2003. Proceedings of 2003 IEEE Conference*. IEEE, 1350-1355.

Stojkoska, B. L. R. and Trivodaliev, K. V. 2017. A review of Internet of Things for smart home: Challenges and solutions. *Journal of Cleaner Production*, 140: 14541464.

Subahi, A. F. and Bouazza, K. E. 2020. An Intelligent IoT-Based System Design for Controlling and Monitoring Greenhouse Temperature. *IEEE Access*,

Sumithabhashini, D. 2020. IOT based Greenhouse monitoring and alerting system using Raspberry Pi.

Sun, Y. and Kantor, P. B. 2006. Cross- Evaluation: A new model for information system evaluation. *Journal of the American Society for Information Science and Technology*, 57 (5): 614-628.

Tamayo, R. A. C., Ibarra, M. L. and Macías, J. A. G. 2010. Better crop management with decision support systems based on wireless sensor networks. In: Proceedings of *Electrical Engineering Computing Science and Automatic Control (CCE), 2010 7th International Conference*. Tuxtla Gutierrez, Mexico, 8-10 Sept. 2010. IEEE, 412-417.

Tzounis, A., Katsoulas, N., Bartzanas, T. and Kittas, C. 2017. Internet of Things in agriculture, recent advances and future challenges. *Biosystems Engineering*, 164: 3148.

Udaykumar, R. 2015. Development of WSN system for precision agriculture. In: Proceedings of *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*. IEEE, 1-5.

Vaishnavi, V. and Kuechler, W. 2004. Design research in information systems. January 20, 2004, last updated August 16, 2009. *URL: http://desrist. org/designresearch-in-information-systems*,(Accessed 31 January 2020)

Vimal, P. and Shivaprakasha, K. 2017. IOT based greenhouse environment monitoring and controlling system using Arduino platform. In: Proceedings of *2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*. IEEE, 1514-1519.

Van Lenteren, J. C. 2000. A greenhouse without pesticides: fact or fantasy? *Crop protection*, 19 (6): 375-384.

Van Os, E. A. 1999. Closed soilless growing systems: a sustainable solution for Dutch greenhouse horticulture. *Water Science and Technology,* 39 (5): 105-112.

Venable, J., Pries-Heje, J. and Baskerville, R. 2016. FEDS: a framework for evaluation in design science research. *European Journal of Information Systems*, 25 (1): 77-89.

Venable, J. R., Pries-Heje, J., Bunker, D. and Russo, N. L. 2011. Design and diffusion of systems for human benefit: toward more humanistic realisation of information systems in society. *Information Technology & People*, 24 (3): 208-216.

Vineela, M.T., NagaHarini, J., Kiranmai, C., Harshitha, G. and AdiLakshmi, B., 2018. IoT Based Agriculture Monitoring and Smart Irrigation System Using Raspberry Pi. *International Research Journal of Engineering and Technology*, *5*(1), pp.14171420.

Von Alan, R. H., March, S. T., Park, J. and Ram, S. 2004. Design science in information systems research. *MIS quarterly*, 28 (1): 75-105.

Vujovic, V. and Maksimovic, M. 2014. Raspberry Pi as a Wireless Sensor node: Performances and constraints. In: Proceedings of *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2014 37th International Convention on*. IEEE, 1013-1018.

Wang, H. Z., Lin, G. W., Wang, J. Q., Gao, W. L., Chen, Y. F. and Duan, Q. L. 2014. Management of big data in the internet of things in agriculture based on cloud computing. In: Proceedings of *Applied Mechanics and Materials*. Trans Tech Publ, 1438-1444.

Wang, Y., Chen, R. and Wang, D.-C. 2015. A survey of mobile cloud computing applications: Perspectives and challenges. *Wireless Personal Communications*, 80 (4): 1607-1623.

Watson, K., Hammer, J. V., Reid, J. D., Skinner, M., Kemper, D. and Nagel, C. 2012. *Beginning Visual C# 2012 Programming*.in Canada. John Wiley & Sons.

Whitmore, A., Agarwal, A. and Da Xu, L. 2015. The Internet of Things—A survey of topics and trends. *Information Systems Frontiers*, 17 (2): 261-274.

Wiegers, K. and Beatty, J. 2013. *Software requirements*. Pearson Education.

Xia, J., Tang, Z., Shi, X., Fan, L. and Li, H. 2011. An environment monitoring system for precise agriculture based on wireless sensor networks. In: Proceedings of *Mobile Ad-hoc and Sensor Networks (MSN), 2011 Seventh International Conference* . IEEE, 28-35.

Yadav, S., Swarnkar, K., Chauhan, T. and Jain, R. 2012. How to improve food security and farming systems by using wireless sensor networks (WSN). *J. Inf. Eng. Appl*, 2 (6): 11-20.

Yick, J., Mukherjee, B. and Ghosal, D. 2008. Wireless sensor network survey. *Computer networks*, 52 (12): 2292-2330.

Yigit, M., Yildiz, H. U., Kurt, S., Tavli, B. and Gungor, V. C. 2018. A survey on packet size optimization for terrestrial, underwater, underground, and body area sensor networks. *International Journal of Communication Systems*, 31 (11): e3572.

Yu, X., Wu, P., Han, W. and Zhang, Z. 2012. Overview of wireless underground sensor networks for agriculture. *African Journal of Biotechnology*, 11 (17): 39423948.

Yu, X., Wu, P., Han, W. and Zhang, Z. 2013. A survey on wireless sensor network infrastructure for agriculture. *Computer Standards & Interfaces*, 35 (1): 59-64.

Zerger, A., Rossel, R. V., Swain, D., Wark, T., Handcock, R. N., Doerr, V., BishopHurley, G., Doerr, E., Gibbons, P. and Lobsey, C. 2010. Environmental sensor networks for vegetation, animal and soil sciences. *International Journal of Applied Earth Observation and Geoinformation*, 12 (5): 303-316.

Zhang, Q., Cheng, L. and Boutaba, R. 2010. Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*, 1 (1): 7-18.

Zhang, W., Kantor, G. and Singh, S. 2004. Integrated wireless sensor/actuator networks in an agricultural application. In: Proceedings of *Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM, 317-317.

Zhao, J.-c., Zhang, J.-f., Feng, Y. and Guo, J.-x. 2010. The study and application of the IOT technology in agriculture. In: Proceedings of *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*. IEEE, 462-465.

Zhao, K. and Ge, L. 2013. A survey on the internet of things security. In: Proceedings of *Computational Intelligence and Security (CIS), 2013 9th International Conference on*. IEEE, 663-667.

Zheng, L., Zhang, H., Han, W., Zhou, X., He, J., Zhang, Z., Gu, Y. and Wang, J. 2011. Technologies, applications, and governance in the internet of things. *Internet of thingsGlobal technological and societal trends. From smart environments and spaces to green ICT.* In China

Zhou, L., Chen, N. and Chen, Z. 2016. A Cloud Computing-Enabled Spatio-Temporal Cyber-Physical Information Infrastructure for Efficient Soil Moisture Monitoring. *ISPRS International Journal of Geo-Information*, 5 (6): 8

APPENDIX 1

DUT IoT Greenhouse Monitor Telemetry Data -> Azure IoT Hub

Date	Time	MessageId	Temperature	Humidity	moisture
10/30/2019	0:54:18	"1749e8af-c16b-4e26-821f-bbf9b58fe0b1"	21.23713534	62.56688283	Wet
10/30/2019	0:54:29	"6a165010-a28d-4141-bdcd-a7a7482fd7f8"	20.85031252	62.33004177	Wet
10/30/2019	0:54:39	"b2b11b54-389b-4719-9b4b-7b271a957f32"	20.73879192	62.33339237	Wet
10/30/2019	0:54:50	"bd46ae23-13d2-4010-80ae-9fb912fc56c6"	20.52314746	61.29721474	Wet
10/30/2019	0:55:00	"d93f91c1-dc3f-4181-8057-5fd14dda4b5f"	20.18021229	62.06044249	Wet
10/30/2019	0:55:10	"861d20a5-fcb3-4d2a-80b4-82f41c624162"	21.50116557	61.78692171	Wet
10/30/2019	0:55:20	"19af49ce-9a96-4355-a9a6-b43cc0d1d9d9"	20.33624572	61.30122058	Wet
10/30/2019	0:55:31	"3f2d0a54-8e11-4a73-aec4-30f902602171"	20.92118496	62.14804033	Wet
10/30/2019	0:55:41	"d1b6b028-ecbb-46f8-8a46-9819fc2e3776"	21.42233593	62.26735264	Wet
10/30/2019	0:55:51	"5b13b259-0a31-4a4b-8acd-c3d18904db2b"	21.51710709	61.00231209	Wet

10/30/2019	0:56:02	"ab5a03b8-73f0-4025-8b13-003e90a45b56"	21.91810206	62.48171685	Wet
10/30/2019	0:56:12	"8f23510e-9729-4b1e-8f13-c019a5b11724"	20.99616481	60.02377798	Wet
10/30/2019	0:56:22	"41e98372-8dd3-4b8f-a373-9ce7076da22b"	21.11807478	61.2889832	Wet
10/30/2019	0:56:33	"fba52a1a-01ca-451d-b3b5-73a9adcf14a8"	21.30562161	62.42753269	Wet
10/30/2019	0:56:43	"b51a4ec9-1e83-4b41-932d-562ae9a731a3"	20.39014816	60.9081136	Wet
10/30/2019	0:56:53	"79b197e2-544d-42d2-8b79-d511eb629d9c"	20.41354897	60.62351307	Wet
10/30/2019	0:57:04	"1772e452-5c16-4910-aec7-a92ca179c724"	20.36407143	61.63915038	Wet
10/30/2019	0:57:14	"2e9093f6-b832-486c-afd6-e59ad596d162"	20.26046251	60.26869516	Wet
10/30/2019	0:57:24	"c0392de5-6a41-4471-a441-b7a74e149024"	21.37668488	62.10836111	Wet
10/30/2019	0:57:34	"901498a4-d1de-4728-8eaa-6a4957a7d418"	21.30953826	60.01444468	Wet
10/30/2019	0:57:45	"01a434b4-fc34-41f7-bbb1-8e001547314e"	21.86348325	60.44565095	Wet
10/30/2019	0:57:55	"a5470f02-8c41-4b36-9fba-2015d195fd4c"	21.63548953	62.3832333	Wet
10/30/2019	0:58:06	"e2f14e45-d65c-4e24-9f77-f09d207f7051"	21.03476782	61.94915164	Wet
10/30/2019	0:58:16	"bc87f479-1eb3-4db5-9b22-3fc5185a34c2"	20.15737567	60.96906082	Wet

10/30/2019	0:58:26	"887f1916-4066-4119-af92-c96260398216"	21.39618906	61.22399713	Wet
10/30/2019	0:58:36	"60b8ad4a-d136-46ee-be4a-67fbe6835439"	21.51186951	62.80535304	Wet
10/30/2019	0:58:47	"c2e832a8-a602-48a1-b69e-9b7ba935bdf2"	21.60007986	60.92634667	Wet
10/30/2019	0:58:57	"1f78d82c-a855-468b-8c47-9b692e3c979b"	20.83622224	62.37398616	Wet
10/30/2019	0:59:08	"aae25212-fad4-48b7-8425-973153dc9d3c"	20.71509041	61.25169079	Wet
10/30/2019	0:59:18	"b9ef2fd9-7c28-44a0-9bc7-fd96a444dcf5"	20.4352864	62.81920467	Wet
10/30/2019	0:59:28	"22faac91-22cf-4e9d-b80b-d96599875d2e"	20.24997331	61.3571885	Wet
10/30/2019	0:59:38	"9f343dba-d05d-453a-b24b-ac41b51c505f"	20.47466167	62.36220484	Wet
10/30/2019	0:59:49	"2bc76035-17a2-418c-873a-c6c12e0dfc01"	20.96007936	61.62823529	Wet
10/30/2019	0:59:59	"98f2ad68-a7e5-4f21-b55e-1e1ba8df684d"	20.82720972	61.8652182	Wet
10/30/2019	1:00:09	"8c1a7852-85a5-4231-80c9-b39b8759d578"	20.60701788	61.11308228	Wet
10/30/2019	1:00:19	"eef51c32-bcdc-4628-9774-1098e488c4cf"	20.05534201	60.02514277	Wet
10/30/2019	1:00:30	"6b37f21c-7491-422d-9092-1429b971671e"	20.20203017	62.26121595	Wet
10/30/2019	1:00:40	"1da6f82c-fb34-48cf-abbd-de83ba337802"	20.80472481	62.43150214	Wet

10/30/2019	1:00:51	"fe52c3ff-6878-4b13-b120-12446c8f37fc"	20.01898837	62.11101392	Wet
10/30/2019	1:01:01	"5fb86918-90fa-4ec0-bf3d-eb861b9e50e8"	20.98108417	62.72796054	Wet
10/30/2019	1:01:11	"b24eb19d-7fd7-4bb4-9c5d-045c39f109dc"	20.70194646	60.98937158	Wet
10/30/2019	1:01:21	"979c3a23-573b-4036-9f06-a24d4e64e4be"	20.22216607	62.36122511	Wet
10/30/2019	1:01:32	"7674a4d8-9ebc-4d16-9726-007cbf09d59f"	21.09353955	60.8149704	Wet
10/30/2019	1:01:42	"cfa68910-635b-4f81-9edb-fd74bd98a261"	20.81559551	62.61976047	Wet
10/30/2019	1:01:52	"12d7d90c-59e3-426e-b577-436ece505906"	20.25654468	61.0167076	Wet
10/30/2019	1:02:03	"cba9616c-69c5-47e7-816c-e055e2e00ad0"	21.4640397	60.81333652	Wet
10/30/2019	1:02:13	"fd1b955d-7800-4168-a387-6cb545ebdd38"	20.97028768	62.14510271	Wet
10/30/2019	1:02:23	"f5635ca4-758e-45e6-b0a3-72ae6bb86866"	21.75965648	61.43803637	Wet
10/30/2019	1:02:34	"29b6b2a7-f0b0-425c-b3df-d56f396439b1"	21.82243897	60.09102945	Wet
10/30/2019	1:02:44	"2971c36c-eea1-4fa8-8ff5-76b85a41c998"	20.62874284	62.92391644	Wet
10/30/2019	1:02:55	"ae2558a4-ba15-47bd-a6cc-cca145c13ac5"	20.47222471	61.37084531	Wet
10/30/2019	1:03:05	"e1e9bfa0-17d8-4b8e-bf2b-1901d3fbe108"	20.03902267	60.9812013	Wet

10/30/2019	1:03:15	"075e1308-c4a8-46f5-8cc8-8e7c6b5c4a44"	20.76065607	62.2426615	Wet
10/30/2019	1:03:25	"04351d88-d211-401c-bdba-109af5c682ce"	21.66820519	60.13890385	Wet
10/30/2019	1:03:36	"916e77fa-584d-4302-a13a-34ffd4b3865a"	21.81283963	61.82283122	Wet
10/30/2019	1:03:46	"8bb7e90e-fb10-49ad-8005-5c6ad120ee23"	21.56366907	61.9616217	Wet
10/30/2019	1:03:57	"08b5f768-e6e7-4ef9-af84-437c71c628f5"	21.85337636	60.92496906	Wet
10/30/2019	1:04:07	"20adabac-b995-4549-bd01-ef3891581873"	21.17752332	62.38558838	Wet
10/30/2019	1:04:17	"184c0356-280d-48ad-848f-77d4fc347954"	20.68262626	61.35076739	Wet
10/30/2019	1:04:28	"f2c0943a-560e-4735-bb5a-4d18817a1654"	20.48126368	60.62514863	Wet
10/30/2019	1:04:38	"82cbc42b-761c-4ec6-836d-4c6a6bbafd44"	20.26989468	61.6210541	Wet
10/30/2019	1:04:48	"b0698c89-320a-42d5-97e5-7597ebb9074f"	20.98693412	62.89381922	Wet
10/30/2019	1:04:59	"55375166-7cd5-4115-baee-36c90926b4e2"	21.27801516	62.26967649	Wet
10/30/2019	1:05:09	"428aab90-d061-443e-9bf4-d779574347bd"	21.04647417	61.15794254	Wet
10/30/2019	1:05:19	"5156d8ec-b709-4a82-9290-908298310a00"	21.74782082	62.76905085	Wet
10/30/2019	1:05:30	"2e17b791-3326-4bf6-b4ad-d146083482d2"	21.79856246	62.9374531	Wet

10/30/2019	1:05:40	"63d76206-5f44-4d97-a4a2-bc63d84d01f4"	20.77859977	61.54770982	Wet
10/30/2019	1:05:50	"a11a3fe4-0d16-4fef-b8f1-f63b33a3a158"	21.34641565	62.68207438	Wet
10/30/2019	1:06:00	"eb6956b7-f1b0-44b8-8d82-abbf6bff3f2f"	20.62055839	62.3520478	Wet
10/30/2019	1:06:11	"e3a1c4a0-2966-4031-b8dd-0ae8ade979d4"	20.813494	62.39764783	Wet
10/30/2019	1:06:21	"a9b85938-e18a-418c-97d4-a30926f13659"	20.87510841	61.08448942	Wet
10/30/2019	1:06:31	"f68ecb55-edea-43da-aab1-d42b02aa3c6f"	21.93366735	61.56198579	Wet
10/30/2019	1:06:42	"d9704747-b4af-44d9-a6f1-0429a4ecd7d5"	21.11413599	60.23443784	Wet
10/30/2019	1:06:52	"e70f4229-56bf-4fba-b9d3-5e3819b9229a"	20.68884799	60.53046247	Wet
10/30/2019	1:07:02	"5603a2d5-70c5-42f7-a801-fbf18039fa78"	20.25254516	60.25389634	Wet
10/30/2019	1:07:13	"75abd1c2-167e-4336-9dc0-6b409d514479"	20.27606465	61.38289106	Wet
10/30/2019	1:07:23	"1d085fb4-ea35-4b57-b177-8984b9c6d4fe"	21.57942262	60.31796563	Wet
10/30/2019	1:07:34	"6689527a-850c-4567-8198-8eb4f18847da"	21.67938294	62.08728296	Wet
10/30/2019	1:07:44	"1133ab46-4553-4baf-8161-da9c76eee37b"	21.92696275	60.16471479	Wet
10/30/2019	1:07:54	"11c6e9d9-e2c3-4d7c-91b6-b096b4d4615e"	21.37611904	61.87130762	Wet

10/30/2019	1:08:05	"507a88d0-5815-4b49-a0f6-4286a796210e"	20.4486335	61.34436525	Wet
10/30/2019	1:08:15	"054a8d6e-1bae-4cca-a2ca-c7326f61a11e"	20.34478175	62.9502086	Wet
10/30/2019	1:08:25	"4b27ffb1-b79b-44fc-a81e-2487e8397ac4"	21.41665835	62.40805052	Wet
10/30/2019	1:08:36	"01b5efac-27c1-4694-96fd-dfb16c7f0099"	20.52914803	61.23235473	Wet
10/30/2019	1:08:46	"ad25a9ee-8866-49e6-9839-7c00d222eabd"	21.27023039	62.08421059	Wet
10/30/2019	1:08:56	"e78845ca-ea19-49a1-8f41-3835ff8ffe60"	20.96983387	61.67189159	Wet
10/30/2019	1:09:06	"c7a5e369-b096-45d3-8745-fbea276c0942"	20.08701759	61.32424769	Wet
10/30/2019	1:09:17	"e92f127c-0421-45ba-a592-eb34dbb6b9ef"	21.54165735	62.3203526	Wet
10/30/2019	1:09:27	"e9488585-feda-4a18-8d86-93c02ddfa0d9"	21.14703539	62.91841539	Wet
10/30/2019	1:09:37	"e6a6e331-d4ed-4161-a2e5-ecff5575512c"	20.81507682	61.68258489	Wet
10/30/2019	1:09:48	"09dcbb7d-d913-4b2a-88e7-d68a2d7bc81a"	20.82426967	61.03924879	Wet
10/30/2019	1:09:58	"05083bf1-ba1c-4f76-bd5f-ebebdf7a9ea0"	20.51941654	62.36783489	Wet
10/30/2019	1:10:08	"a28fe1cb-f5a8-43bb-bf69-5db31e4b11fe"	21.56996924	61.31495263	Wet
10/30/2019	1:10:18	"573d1dd3-20d2-469c-8178-d2a70ee59bbd"	20.37887944	60.84993403	Wet

10/30/2019	1:10:29	"3c1f1a68-ba62-45da-b4b3-060e66b09e5a"	21.3524236	62.93234051	Wet
10/30/2019	1:10:39	"19e83676-51b2-41d6-96e0-6cc0c6a55db9"	21.86108683	60.76612279	Wet
10/30/2019	1:10:49	"10415245-3710-4e44-804b-db8903933098"	20.19191282	62.34893338	Wet
10/30/2019	1:11:00	"d4f19a95-5f5b-4361-b18b-595fff5cc22a"	21.14979839	62.96829737	Wet
10/30/2019	1:11:10	"244d5b76-c0b4-4c1e-8f8d-c1ab16510f0f"	20.3782112	62.95029242	Wet
10/30/2019	1:11:20	"2bea8b17-824e-4d53-940f-b86b5cb6ec8a"	21.62466551	62.26315347	Wet
10/30/2019	1:11:31	"99b71566-ab3b-46d2-a79f-bf051412cb35"	21.62714387	62.80091725	Wet
10/30/2019	1:11:41	"3573d7cd-1797-4e84-ae19-5d85517c0b91"	21.08341125	61.29460714	Wet
10/30/2019	1:11:51	"68b78dc2-e86f-41f1-97f1-5602e000fedf"	21.19943036	61.74220539	Wet
10/30/2019	1:12:01	"549a6f43-cd6e-4ac6-8e8f-74d3931011fa"	20.83490978	61.04488625	Wet
10/30/2019	1:12:12	"d6759461-0378-41a2-acf5-433ae74a5ba2"	20.67031974	60.19872181	Wet
10/30/2019	1:12:22	"f18c3a0f-da5f-4739-96b2-968d654edfd7"	20.24448659	62.97202874	Wet
10/30/2019	1:12:32	"814151f2-d6c9-4136-b8b5-a292ef45daf4"	21.29473304	60.38159367	Wet
10/30/2019	1:12:42	"eb0b042a-e508-4d2a-aee5-6ed3be842eb7"	20.42326874	62.63370146	Wet
10/30/2019	1:12:53	"50e84ed1-0c59-4abc-8827-71087846eb9d"	20.37682696	61.14933773	Wet

10/30/2019	1:13:03	"535e5e06-fee7-4836-8b9b-80eacb1daaec"	20.39683649	60.8100349	Wet
10/30/2019	1:13:13	"9f5ba39f-3ff0-466e-a21d-6d665f61214d"	21.22648757	62.94378801	Wet
10/30/2019	1:13:24	"383e7ba7-bb8c-4712-8a07-3e0609cec18b"	21.46914621	61.97300507	Wet
10/30/2019	1:13:34	"100ef77c-54e5-49e1-a099-7c9b5683a378"	21.5283869	60.68862802	Wet
10/30/2019	1:13:44	"fbe9f9dc-5c5b-4e8c-b78a-22b1158366d1"	20.92268157	60.78159301	Wet
10/30/2019	1:13:55	"1442367c-8cf3-408d-afcd-74b1942ab9d6"	21.7330334	62.34418865	Wet

10/30/2019	1:14:05	"31d03325-4438-408f-9106-8ec2e6b79f49"	21.16869053	61.77026066	Wet
10/30/2019	1:14:15	"4c4453fb-ec28-435e-b365-da8f706727de"	20.32094475	61.95946175	Wet
10/30/2019	1:14:26	"d772a106-59b2-4a26-a6aa-3ffa0a6859e2"	21.4945794	61.43548726	Wet
10/30/2019	1:14:36	"b6d15a89-0902-48a9-ba77-1a598e99a856"	21.60942127	62.48451766	Wet
10/30/2019	1:14:46	"a4d7b4dd-de66-48f8-b048-841104cedb97"	20.37912624	60.61274847	Wet
10/30/2019	1:14:56	"6ff4bdc7-ae2e-4e3c-bdb3-06af340df077"	20.0417253	62.52343291	Wet
10/30/2019	1:15:07	"f614677d-b662-42a7-a94f-8059aa4de026"	21.89630295	61.8299136	Wet

10/30/2019	1:15:17	"3bc27ad3-b862-4e12-84bd-278610397bb1"	21.71040709	62.81960151	Wet
10/30/2019	1:15:28	"2f0f28a2-3bfd-4b4a-891e-7b1c6736bd90"	21.21601548	62.90627556	Wet
10/30/2019	1:15:38	"619beeb1-9db6-4445-9db4-be31a62f8f7e"	21.14268684	62.09099613	Wet
10/30/2019	1:15:48	"310d2b50-4b08-45bd-a4b0-a0f500d780d0"	21.60203795	62.41797908	Wet
10/30/2019	1:15:58	"689383ef-71fe-45b3-bfd6-6a0f9bbd2a8e"	21.57002512	61.62696336	Wet
10/30/2019	1:16:09	"4917c6a1-c2a0-431f-b6f1-6de6575179e9"	20.28228141	62.49692779	Wet
10/30/2019	1:16:19	"ecfd3737-0013-43e3-b720-c8b5ff090650"	20.39813195	62.65211181	Wet
10/30/2019	1:16:29	"aa328417-f2df-4d6f-99c8-a454a09f199d"	21.33468453	61.11051752	Wet
10/30/2019	1:16:40	"d47cdff6-450c-43f5-9dfe-1f61d8c0f63b"	20.23878869	60.47077166	Wet
10/30/2019	1:16:50	"0b9c92f8-fb0b-4e3a-b97d-b8d8e7856e9d"	20.96355743	61.66129095	Wet
10/30/2019	1:17:00	"20f98e06-b817-456a-9a9e-75e288053c11"	20.96379067	61.59646922	Wet
10/30/2019	1:17:11	"5361b7c2-4770-4560-ac17-943489b44f75"	21.66040774	62.98263781	Wet
10/30/2019	1:17:21	"86260242-a8ee-44ac-a082-abcdc62d586b"	20.75981639	62.19941586	Wet
10/30/2019	1:17:31	"c98f97d2-d110-4278-80c7-0d6784c397bc"	20.1463797	61.08072278	Wet

10/30/2019	1:17:41	"68b09718-16ae-46ae-913e-d18b7a1a150c"	20.38709891	62.98250626	Wet
10/30/2019	1:17:52	"cd26e54f-f1f0-434c-8eb2-6e14e18dbb40"	20.49829796	62.31629845	Wet
10/30/2019	1:18:02	"40db4ed1-c4db-4a36-8942-0603a79aba1a"	20.06622239	60.37380572	Wet
10/30/2019	1:18:13	"ca15f3df-f131-4021-b34f-cf454f37e19e"	21.60492963	62.47297883	Wet
10/30/2019	1:18:23	"dcbd1e6c-a3ca-431d-bf09-4beca3fd4774"	21.24306986	61.47774679	Wet
10/30/2019	1:18:34	"45675157-6138-4cea-9f39-f0f888f6e9c2"	21.37837516	60.50855397	Wet
10/30/2019	1:18:44	"3a5d2f49-ae77-4c16-8a90-d23f6e8d04d2"	21.96075449	62.33505672	Wet
10/30/2019	1:18:54	"322842c6-862a-4799-98ba-6ff9c162aa98"	21.61014865	61.85445377	Wet
10/30/2019	1:19:05	"22e6b151-afed-49dd-baf1-debbfcfa32f6"	21.02402642	62.7449413	Wet
10/30/2019	1:19:15	"b1521d33-bc69-478b-9cad-ff5c1d900809"	20.55885955	62.7620201	Wet
10/30/2019	1:19:25	"aa2ec09b-2a82-4234-8cf7-30db3b21cbc8"	20.32166058	62.45817184	Wet
10/30/2019	1:19:36	"44f343b2-c536-45d5-895f-9d83716acfb8"	20.16971029	62.59181629	Wet
10/30/2019	1:19:46	"45590c41-7ac7-4928-821d-9bb29320aa39"	20.71873118	61.18316347	Wet
10/30/2019	1:19:56	"857c112d-a754-42f2-abea-ebed2f1c4f4f"	20.25615173	60.96914141	Wet

10/30/2019	1:20:07	"92ed6310-b6bb-48d1-9099-1dddd9efa153"	20.2193266	61.84138541	Wet
10/30/2019	1:20:17	"4ae0273d-776b-4478-b6f3-242f6bd831dc"	21.17770065	62.51461441	Wet
10/30/2019	1:20:27	"b2b13a87-10ef-4e2f-8a65-d72cc27fb932"	21.24761772	61.32233415	Wet
10/30/2019	1:20:38	"3d65866d-fca7-45a1-8c73-d9f2b725baa5"	21.22488714	62.37253142	Wet
10/30/2019	1:20:48	"44da75d5-3ff0-4a1f-9e6f-b31552458cbf"	20.58634428	61.10712367	Wet
10/30/2019	1:20:58	"bb00139d-5856-4105-afb2-7626ba331dd6"	21.59839613	60.22339221	Wet
10/30/2019	1:21:09	"5c8abf2f-d41d-4eb0-8e25-92744a7f54d1"	20.16314491	62.52904797	Wet
10/30/2019	1:21:19	"2d6ca24d-b16b-4694-9b6c-770fe3a2cf75"	21.49727515	61.88043624	Wet
10/30/2019	1:21:30	"5b9e5b8b-344c-4b7e-9cfa-0d80d9c028f1"	20.93547261	60.93678217	Wet
10/30/2019	1:21:40	"4f038313-c926-4984-8e8a-652948527085"	21.14677282	62.11062928	Wet
10/30/2019	1:21:50	"c6e70ddb-4265-4026-bb6c-b09f9f474cdb"	21.45416417	60.63615784	Wet
10/30/2019	1:22:00	"a9c966d7-8f8e-402c-bea5-ae8d22ee5f98"	20.96439879	61.39478328	Wet
10/30/2019	1:22:11	"68742cf9-3068-4c6b-be97-df59ff89f51f"	20.90741769	60.45754945	Wet
10/30/2019	1:22:21	"858ed0ca-8572-4613-860f-faf04acdc01d"	20.39882127	61.12247653	Wet

10/30/2019	1:22:31	"d826c8a2-9742-4135-bfdb-92600de3173e"	21.81862721	61.15563065	Wet
10/30/2019	1:22:42	"b478fbd0-5cab-4b5c-ae3f-153443695547"	20.82546779	61.91617012	Wet
10/30/2019	1:22:52	"5a39b21c-d2db-4940-9c00-66518e1cf514"	21.99305208	61.43825304	Wet
10/30/2019	1:23:02	"cc3eeee5-634b-40df-be99-867b239064d2"	21.42932596	60.02321939	Wet
10/30/2019	1:23:13	"c61bce5f-c23e-4f14-b26d-2ebacfa0af50"	21.80746387	62.55294833	Wet
10/30/2019	1:23:23	"817b2e28-c103-47a6-b11f-e16f8aee2bc1"	21.09141826	61.61879758	Wet
10/30/2019	1:23:33	"f31f22ac-6a97-4eee-af0a-79eef25a8a8b"	20.33181734	60.04269155	Wet
10/30/2019	1:23:43	"5deb6f62-0122-4344-8a08-a069ac38425b"	20.64995824	60.42891596	Wet
10/30/2019	1:23:54	"1f1fdc65-6101-425a-a71f-e842ee9d4093"	20.23156474	60.61489712	Wet
10/30/2019	1:24:04	"9bf8820b-87ca-4318-9d1f-5c8cfa26e73a"	21.67820182	60.9534429	Wet
10/30/2019	1:24:14	"bb25e856-507b-448f-a2f4-74a68dc5b20d"	20.14150608	61.3741292	Wet
10/30/2019	1:24:24	"9223b566-2eca-4ee7-bdd6-2aa9dc0875c6"	20.79240491	60.14131459	Wet
10/30/2019	1:24:35	"8c92bbf6-0250-4faa-9475-c22f96d9a124"	21.64200283	61.27359832	Wet
10/30/2019	1:24:45	"9d1b1008-db1e-492c-9ff0-c4e421f246c5"	21.1919301	62.69283206	Wet

10/30/2019	1:24:55	"5ba4890d-7039-4ac2-a2e2-4d8a6c7516d9"	20.26319148	60.37176771	Wet
10/30/2019	1:25:06	"8a94f793-b99f-4d49-abe2-2c2228082c63"	20.76899192	61.41387712	Wet
10/30/2019	1:25:16	"37ac34bf-cf18-43ce-917e-dcfb5f8e0340"	20.48273482	60.71485418	Wet
10/30/2019	1:25:26	"4a2a3ab2-1720-404e-9ff9-fd1e7fad0962"	21.7630604	62.72388576	Wet
10/30/2019	1:25:36	"5198dfe2-f81f-4754-b090-dce26c263e2e"	21.91261466	60.48142408	Wet
10/30/2019	1:25:47	"fb8e04d7-681a-4c16-9796-a33c1d450d62"	20.15587606	61.80646433	Wet
10/30/2019	1:25:57	"63658e3a-650f-4472-871f-8b5b99bb9cb7"	20.25670602	62.22269333	Wet
10/30/2019	1:26:08	"b4902ec9-3981-41bb-8b3a-c64cb70e60bc"	21.44616029	61.85026059	Wet
10/30/2019	1:26:18	"e09572e1-9564-46b1-b793-12fb168d1977"	21.53310319	60.10136164	Wet
10/30/2019	1:26:28	"8dd1664a-9986-4922-87b6-4cbe26e1e942"	21.07526884	62.1385547	Wet
10/30/2019	1:26:39	"404dd36c-7dec-494f-a8bf-65985fa49968"	21.77414699	61.01768222	Wet
10/30/2019	1:26:49	"6def52db-a356-414c-aaf6-dc2460b19f01"	20.69829078	60.74622941	Wet
10/30/2019	1:26:59	"537bade4-e293-4c43-bde4-3011d3551e06"	20.62683115	62.64478901	Wet
10/30/2019	1:27:09	"aa6a3e41-0ddd-4e1c-bf72-ff8ceadef7bb"	20.60681161	61.35381162	Wet

10/30/2019	1:27:20	"39ca722b-6ed5-4f56-af89-7506568a4c42"	21.97801552	61.09688709	Wet
10/30/2019	1:27:30	"d25b9919-68e1-46b6-8bca-939071f5d8c8"	21.63544391	61.71597981	Wet
10/30/2019	1:27:41	"0cfc0da5-4da4-43cd-8aac-3519e676aa91"	21.76690525	62.45115688	Wet
10/30/2019	1:27:51	"e4868cbe-efbf-43b9-89a9-0babd9a42d72"	21.75228811	62.33942811	Wet
10/30/2019	1:28:01	"0a0844e7-8128-4422-acf7-b0ab09f66240"	20.93297363	60.22325027	Wet
10/30/2019	1:28:11	"db38f9f2-2be0-41da-9870-a6cbc729c7d2"	20.59646356	62.78287184	Wet
10/30/2019	1:28:22	"869313b5-1fa8-4241-b1f7-40f88419b6a1"	20.26540064	61.25105159	Wet
10/30/2019	1:28:32	"8a85e86e-fa1a-4796-b0d0-97f3ee543e21"	20.37332932	62.86592302	Wet
10/30/2019	1:28:42	"921b8e42-a82d-4b4e-a52d-434bd78aa60f"	20.25405535	60.71083839	Wet
10/30/2019	1:28:53	"3c2f509b-3f38-458c-bd55-0e6f805de3db"	20.37892258	60.98956578	Wet
10/30/2019	1:29:03	"7bc7ad3c-a7d3-4593-b8ff-3656b366c84f"	21.46992585	62.33834678	Wet
10/30/2019	1:29:13	"f2051233-92f0-450d-83a3-056bf2a8e0c0"	20.56110654	62.36164261	Wet
10/30/2019	1:29:23	"1f283b41-7cd1-4b3c-b09c-569fa266d490"	21.7737967	62.64934046	Wet
10/30/2019	1:29:34	"f52109db-65a8-4d5e-ad65-90004bd5db08"	20.02107438	62.37710501	Wet

10/30/2019	1:29:44	"a002205d-3526-44e6-bd28-267b27371c65"	21.54955436	60.40725847	Wet
10/30/2019	1:29:54	"0d72bc8f-e507-479d-b31f-c31b7493d248"	20.3157536	61.07931322	Wet
10/30/2019	1:30:05	"fdf183d7-fcf6-451b-b286-10b7e799a6ae"	21.86975784	61.29077898	Wet
10/30/2019	1:30:15	"6ffdecfc-302e-4fe4-a859-b92ac89ad229"	21.83790832	61.77203489	Wet
10/30/2019	1:30:26	"79f065b8-0230-48f0-b979-8fc042f0df76"	20.68550548	61.51783428	Wet
10/30/2019	1:30:36	"596af878-91e2-4a6c-b0df-43579f47519b"	21.4374043	60.93390215	Wet
10/30/2019	1:30:46	"ad34aab7-be60-4e7c-a189-3406e1ef19b7"	21.72950745	62.82981232	Wet
10/30/2019	1:30:56	"75eba95f-f8bd-43de-a026-95ccb9bcc485"	20.30053343	62.07640451	Wet
10/30/2019	1:31:07	"3ba6ee7a-9363-452a-999a-76c336436fc1"	21.47111087	61.83003142	Wet
10/30/2019	1:31:17	"0d547676-ab22-4064-ab9d-e553752ffec0"	21.1603025	61.4101689	Wet
10/30/2019	1:31:27	"1dd77d16-e8c6-40fa-a42f-60a1a0315dc7"	20.3051255	61.18151315	Wet
10/30/2019	1:31:38	"c191e0f3-98eb-40a8-9154-46cb587fe9a7"	20.24343092	60.22940833	Wet
10/30/2019	1:31:48	"e4035c40-9242-469c-9f55-744b86516cfa"	21.3842701	62.92065163	Wet
10/30/2019	1:31:58	"448d8785-b583-457b-93f1-5b26a95e69ac"	21.43261523	60.6286765	Wet

10/30/2019	1:32:09	"eb564423-167e-41f6-ada4-8c63300668e8"	20.17015152	60.09152326	Wet
10/30/2019	1:32:19	"eb2127d8-04b8-44e6-923b-b6152fd257a9"	21.37018984	60.00101742	Wet
10/30/2019	1:32:29	"f558f705-8599-4ae0-85e7-75ae4cf36577"	21.61303021	60.25132715	Wet
10/30/2019	1:32:40	"32eec0aa-743b-4240-9817-6690ae255647"	20.01006438	60.99220198	Wet
10/30/2019	1:32:50	"cbe4d66c-2974-4886-91fa-d0247d9a5862"	21.83307992	62.81538212	Wet
10/30/2019	1:33:00	"3f22f911-e5c2-4121-a7de-fc0d9f2c65c6"	21.98549005	62.10732198	Wet
10/30/2019	1:33:11	"a3e12d61-4580-4e8a-935a-7c1c8914a224"	20.99612607	61.78795908	Wet
10/30/2019	1:33:21	"1fc5a0d0-24ac-4be1-81ef-1c7ecdd1deda"	21.22510987	61.86324875	Wet
10/30/2019	1:33:31	"8ec0af26-eda2-4560-8467-35edcd0f6b0d"	21.03648796	62.63448171	Wet
10/30/2019	1:33:41	"a652acc2-f2dd-473a-a798-44e3ce65be98"	20.93020307	62.37507645	Wet
10/30/2019	1:33:52	"90c20c49-6e5d-4940-8e0f-a5e17d5a49fa"	21.25836443	61.7652553	Wet
10/30/2019	1:34:02	"4c17c7b2-2c5f-4734-b96e-9abc01a34dc9"	20.10331754	60.83066363	Wet
10/30/2019	1:34:13	"5df9b302-12c5-4b92-8ac7-f8d034bb92b4"	20.60592448	61.62144267	Wet
10/30/2019	1:34:23	"cca750c0-9b9c-429c-849d-d8cc0683df60"	21.27961118	61.14281838	Wet
10/30/2019	1:34:33	"89f361cb-cca4-472b-9905-c1bb019e4a1f"	20.02807473	60.24422206	Wet

10/30/2019	1:34:43	"4f090717-b8e1-476c-b2a7-3aec531d9758"	21.33527205	62.88398513	Wet
10/30/2019	1:34:54	"abcbf2d3-4e8f-41c4-95cc-fb5f9f3013d5"	21.42790408	62.12818598	Wet
10/30/2019	1:35:04	"538242e0-3198-4729-b65c-24a5d87596ad"	21.01120508	60.93673496	Wet
10/30/2019	1:35:14	"56eb3288-052d-434d-a45a-fac0edd9c47d"	21.64169969	62.15508903	Wet
10/30/2019	1:35:25	"4eb724f7-78a8-4ad1-a3e4-d15147dd626f"	21.00957123	62.34293402	Wet
10/30/2019	1:35:35	"9babdd10-ec6a-403c-8289-170f8dc1ddee"	21.87647716	62.45859816	Wet

APPENDIX 2

SOIL MOISTURE

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;
using System.Runtime.InteropServices.WindowsRuntime;
using Windows.Foundation;
using Windows.Foundation.Collections;
using Windows.UI.Xaml;
using Windows.UI.Xaml.Controls;
using Windows.UI.Xaml.Controls.Primitives;
using Windows.UI.Xaml.Data;
using Windows.UI.Xaml.Input;
using Windows.UI.Xaml.Media;
using Windows.UI.Xaml.Navigation;
```

```
using Windows.Devices.Gpio;

namespace SoilMoistureLab
{
    /// <summary>
    /// An empty page that can be used on its own or navigated to within a Frame.
    /// </summary>
    public sealed partial class MainPage : Page
    {
        // Arbitralily selected GPIO 17 - you can use any
        private const int GPIO17 = 17;

        // We use a timer to take readings at specified intervals
        private DispatcherTimer timer;

        private GpioPinValue pinValue;
        GpioPin pin;

        public MainPage()
        {
```

```
    this.InitializeComponent();
    timer = new DispatcherTimer();
    timer.Interval = TimeSpan.FromMilliseconds(1000);
    timer.Tick += Timer_Tick;
    InitGPIO();

    // If we have an active pin, let's start the time so we can periodically
    // check the sensor
    if (pin != null)
    {
        timer.Start();
    }
}

private void InitGPIO()
{
    var gpioController = GpioController.GetDefault();

    // Show an error if there is no GPIO controller
    if (gpioController == null)
```

```
    {
        return;
    }

    // Open GPIO pin 17 for output - sensor will send the value through this pin
    pin = gpioController.OpenPin(GPIO17);
    pin.SetDriveMode(GpioPinDriveMode.Input);

}

private void Timer_Tick(object sender, object e)
{
    DateTime tm = DateTime.Now;
    pinValue = pin.Read();

    if (GpioPinValue.Low == pinValue)
    {
        //WanterIndicator.Text = " [" + tm.ToString("H:mm:ss") + "] " + "Water Detected!";
    }
    else
```

```
        {
          // WanterIndicator.Text = " [" + tm.ToString("H:mm:ss") + "] " + "No Water Detected!";
        }
    }
```

HUMIDITY AND TEMPERATURE

```
using namespace concurrency;
using namespace std;
using namespace Platform;
using namespace Windows::Foundation;
using namespace Windows::Foundation::Collections;
using namespace Windows::System::Threading;
using namespace Windows::Devices::Gpio;
using namespace Sensors::Dht;

_Use_decl_annotations_

Dht22::Dht22(Windows::Devices::Gpio::GpioPin^ pin, Windows::Devices::Gpio::GpioPinDriveMode
inputReadMode)
```

```
{
    // ***
    // *** Set Drive Mode to Input
    // ***
    this->_inputReadMode = inputReadMode;
    pin->SetDriveMode(this->_inputReadMode);

    // ***
    // *** Save the Pin
    // ***
    this->_pin = pin;
}

Dht22::~Dht22()
{
    this->_pin = nullptr;
}

Windows::Foundation::IAsyncOperation<DhtReading>^ Dht22::GetReadingAsync()
{
```

```
        return this->GetReadingAsync(DEFAULT_MAX_RETRIES);
}

Windows::Foundation::IAsyncOperation<DhtReading>^ Dht22::GetReadingAsync(int maxRetries)
{
        return create_async([this, maxRetries]
        {
                DhtReading returnValue;
                int i = 0;

                for (i; i < maxRetries; i++)
                {
                        returnValue = this->InternalGetReading();

                        if (returnValue.IsValid)
                        {
                                break;
                        }
                }
```

```
            returnValue.RetryCount = i;

            return returnValue;
        });
}

DhtReading Dht22::InternalGetReading()
{
        DhtReading returnValue;

        // ***
        // *** Create a buffer for the 40-bit reading
        // ***
        std::bitset<40> bits;

        // ***
        // *** Query the performance counter frequency
        // *** to calculate the correct timing.
        // ***
        LARGE_INTEGER qpf;
```

```
QueryPerformanceFrequency(&qpf);

// ***
// *** This is the threshold used to determine whether a bit is a '0' or a '1'.
// *** A '0' has a pulse time of 76 microseconds, while a '1' has a
// *** pulse time of 120 microseconds. 110 is chosen as a reasonable threshold.
// *** We convert the value to QPF units for later use.
// ***
const unsigned int oneThreshold = static_cast<unsigned int>(110LL * qpf.QuadPart /
1000000LL);

// ***
// *** Latch low value onto pin
// ***
this->_pin->Write(GpioPinValue::Low);

// ***
// *** Set pin as output
// ***
this->_pin->SetDriveMode(GpioPinDriveMode::Output);
```

```
// ***
// *** Wait for at least 18 ms
// ***
Sleep(SAMPLE_HOLD_LOW_MILLIS);

// ***
// *** Set pin back to input
// ***
this->_pin->SetDriveMode(this->_inputReadMode);

// ***
// *** Read the current value
// ***
GpioPinValue previousValue = this->_pin->Read();

// ***
// *** catch the first rising edge
// ***
const ULONG initialRisingEdgeTimeoutMillis = 1;
```

```
ULONGLONG endTickCount = GetTickCount64() + initialRisingEdgeTimeoutMillis;
for (;;)
{
	if (GetTickCount64() > endTickCount)
	{
		returnValue.TimedOut = true;
		return returnValue;
	}

	GpioPinValue value = this->_pin->Read();
	if (value != previousValue)
	{
		// ***
		// *** Rising edge?
		// ***
		if (value == GpioPinValue::High)
		{
			break;
		}
```

```
			previousValue = value;
		}
	}

	LARGE_INTEGER prevTime = { 0 };

	const ULONG sampleTimeoutMillis = 10;
	endTickCount = GetTickCount64() + sampleTimeoutMillis;

	// ***
	// *** Capture every falling edge until all bits are received or
	// *** timeout occurs
	// ***
	for (unsigned int i = 0; i < (bits.size() + 1);)
	{
		if (GetTickCount64() > endTickCount)
		{
			returnValue.TimedOut = true;
			return returnValue;
		}
```

```
GpioPinValue value = this->_pin->Read();
if ((previousValue == GpioPinValue::High) && (value == GpioPinValue::Low))
{
    // ***
    // *** A falling edge was detected
    // ***
    LARGE_INTEGER now;
    QueryPerformanceCounter(&now);

    if (i != 0)
    {
        unsigned int difference = static_cast<unsigned int>(now.QuadPart -
prevTime.QuadPart);
        bits[bits.size() - i] = difference > oneThreshold;
    }

    prevTime = now;
    ++i;
}
```

```
        previousValue = value;
    }

    returnValue = this->CalculateValues(bits);

    return returnValue;
}

DhtReading Dht22::CalculateValues(std::bitset<40> bits)
{
    DhtReading returnValue;

    unsigned long long value = bits.to_ullong();

    unsigned int checksum =
        ((value >> 32) & 0xff) +
        ((value >> 24) & 0xff) +
        ((value >> 16) & 0xff) +
        ((value >> 8) & 0xff);
```

```
returnValue.IsValid = (checksum & 0xff) == (value & 0xff);

if (returnValue.IsValid)
{
        unsigned long long value1 = bits.to_ullong();
        returnValue.Humidity = (((value >> 24) & 0xff00) + ((value >> 24) & 0xff)) / 10.0;

        unsigned long long value = bits.to_ullong();
        double temp = (((value >> 8) & 0x7f00) + ((value >> 8) & 0xff)) / 10.0;
        if ((value >> 16) & 0x80) temp = -temp;
        returnValue.Temperature= temp;
}
else
{
        returnValue.Humidity = 0;
        returnValue.Temperature = 0;
}

return returnValue;
```

```
}
```

REVIEW REPORT OF EXAMINER'S CORRECTIONS

Student's Name:	Patiswa Chwayita Mpangele
Student Number:	21649816
Qualification for which registered:	MASTER OF INFORMATION AND COMMUNICATIONS TECHNOLOGY DEGREE
Department in which registered:	DEPARTMENT OF INFORMATION TECHNOLOGY
Name of supervisor:	Dr P Mtshali PhD

Name of co-supervisor:	Dr A Singh DTech

Title of the study:	**DEVELOPING A CLOUD BASED INTERNET OF THINGS (IOT) GREENHOUSE MONITORING SYSTEM**

QUERIES/COMMENTS	**CORRECTIONS/REVISIONS** **(Indicate clearly how you addressed each comment)**
Examiner 2: In technical writing, we do not normally write a section like 5.3; however, the student's enthusiasm for her accomplishment is	section 5.3 **Lessons learned** has been removed from this thesis.

apparent and I would retain this section for this reason.	
Examiner 2: For this to be research, we need to define a hypothesis, and the thesis would test the hypothesis.	The hypothesis for this study has been stated in page 5 section **1.6 The research hypothesis**
Examiner 2: The research questions are very broad; this might be due to the lack of a stated hypothesis.	The research questions have been removed section **1.4 The research questions** from this study and the detailed hypothesis has been provided
Literature review **Examiner 1**: This section is comprehensive and it was perfectly researched. The only improvement I can suggest is to point the entire section towards the argument that IoT and cloud computing is ideal for green house monitoring.	Section 2.2.1 **Wireless Sensor Network Categories**, page 17-18 the discussion of this WSN devices is to give the background about the different type of sensors, which was of help in selecting which sensors to use for this study. several authors describe WSN as the primary key feature of IoT (Xia *et al.* 2011; Akyildiz and Stuntebeck 2006).

For instance in section 2.2.2. a presentation of Wireless Sensor Network Categories must present each of them as a candidate of the incumbent project i.e. why it is appropriate or not. Alternatively the sign posting must cristalise this as options that could have been used and the chosen one must be sponsored in comparison.	
Examiner 2: I understand that my role of examiner is due to my involvement with data science and computing; I am not an expert in agriculture. However, I am aware that there is a great deal of automation in agriculture; it's hard to believe agritech hasn't taken the extra step of IoT by now. I recommend that the	More literature has been added, on the following sections **2.3.2 IoT for agriculture,** third paragraph, page 21. **2.3.2.1 IoT based greenhouse environment monitoring systems** the last paragraph from page 21-24, **2.4 A Definition of cloud computing** page 25-27 **and**

student search the literature again and revise the lit review as needed.	**2.4.1 the application of cloud computing in agriculture** page 27-28
Examiners 1: I however feel that the motivation of the study must be linked to the research problem although the candidate may have been motivated less serious reason **Examiner 2:** Page 2 Horticulture uses “knowledge from books” –not a proper style; rather “uses published data or historical records”	**Section1.2 Motivation of the study** page2-3 has been revised more information from books and papers has been used.
Examiner 2: The sensor readings don’t seem to be independently verified by non-digital means (wet- and dry-bulb thermometers). I searched the document for calibration and	Yes, the sensor readings weren’t verified because the researcher was only focusing on the concept of developing a system. The concept of calibration and miniaturize the system for real world application is reserved for future studies see page

didn't find anything. Where is the verification that the digital readings are correct?	67 section **5.2 The recommendations for possible further research work arising** bullet number 5.
Examiner 2: Regarding the limitations expressed in the conclusions, it is a pity that the student did not undertake the very simple steps needed to overcome these, especially since it seems that there would have been verification by farmers and hence addressed one of the chief issues here. Wifi coverage in a greenhouse can easily be extended by a signal booster, and a raspberry pi only costs about R700. It seems that for a total cost of perhaps R2000, the whole greenhouse could have been covered and the validation and testing issues overcome. These are the sorts of things that IoT is supposed to address, and which are lauded in the early parts of the thesis. It's a pity that in the conclusion it seems as if IoT doesn't live up to the promise**.**	Guidelines have been stated in section **5.2 The recommendations for possible further research work arising.** To guide farmers/ future studies on how to design the system, for experiment purposes

Intuition instead of intuition	This has been removed, since section **1.2** has been revised
Page 40 First paragraph: improper use of semi-colon in two places. Please check whole thesis.	Semi-colon has been removed page 40
Typographical errors	There were all corrected

Corrections approved by supervisor

Yes	✓	**No**	

Supervisor's signature: **Date:** 05/11/2021

Co-supervisor's signature ______ **Date:** 05/11/2021

Student's signature: ________ **Date:** 05/11/2021

Routing: Student ⟶ Supervisor/Co-Supervisor ⟶ Faculty Officer ⟶ FRC