KONGUNADU COLLEGE OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

PROJECT TITLE

IoT Based Smart Crop Protection System For Agriculture

TEAM ID: PNT2022TMID13381

TEAM MEMBER

1. KIRUTHIKA B (TEAM LEAD)
2. KOWSIKA P
3. AARTHY M
4. HEMANITHI J

MENTOR

MR. HARI KUMAR S

+

INDEX

1. **INTRODUCTION**
1.1 Project Overview
1.2 Purpose
2. **LITERATURE SURVEY**
2.1 Existing problem
2.2 References
2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
3.1 Empathy Map Canvas
3.2 Ideation & Brainstorming
3.3 Proposed Solution
3.4 Problem Solution fit
4. **REQUIREMENT ANALYSIS**
4.1 Functional requirement
4.2 Non-Functional requirements
5. **PROJECT DESIGN**
5.1 Data Flow Diagrams
5.2 Solution & Technical Architecture
5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
6.1 Sprint Planning & Estimation
6.2 Sprint Delivery Schedule
6.3 Reports from JIRA
7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
7.1 Feature 1
7.2 Feature 2
7.3 Database Schema (if Applicable)
8. **TESTING**
8.1 Test Cases
8.2 User Acceptance Testing
9. **RESULTS**
9.1 Performance Metrics
10. **ADVANTAGES & DISADVANTAGES**
11. **CONCLUSION**
12. **FUTURE SCOPE**
13. **APPENDIX**
Source Code
GitHub & Project Demo Link

1.INTRODUCTION

1.1 PROJECT OVERVIEW

Crops in farms are many times ravaged by local animals like buffaloes, cows, goats, birds etc. This leads to huge losses for the farmer. Due to over population, it occurs a deforestation this results in shortage of food, water and shelter in forest areas. So, animal's interference in residential areas is increasing day by day which affects human life and property causes human animal conflict but as per nature's rule every living creature on this earth has important role in eco-system. Elephants and other animals coming in to contact with humans, impact negatively in various means such as by depredation of crops, damaging grain stores, water supplies, houses and other assets, injuring and death of humans. One of the major economic issues faced by the country is agriculture as this is the sector which is source of livelihood for about 54% of Indians till date. Still today this sector is not well developed and faces lots of problems resulting into low productivity of crops. As 43% of land in India, is used for farming but contributes only 18% of the nation's GDP. The poor condition of agriculture in the country is the point of concern for Indians. The rural farmers in India suffer from poverty and most of them are illiterate so there is lack of good extension services. The problem of wild life attack on crops i.e., crop Vandalization is becoming very common in the states of Tamil Nadu, Himachal Pradesh, Punjab, Haryana, Kerala and many other states. Wild animals like monkeys, elephants, wild pigs, deer, wild dogs, bison, nilgais, estray animals like cows and buffaloes and even birds like parakeets cause a lot of damage to crops by running over them, eating and completely vandalizing them. According to World Bank's "India: Priorities for Agriculture and Rural Development", India's large agricultural subsidies are hampering productivity-enhancing investment. Over regulation of agriculture has increased costs, price risks and uncertainty. Government intervenes in labour, land, and credit markets. India has inadequate infrastructure and services. World Bank also says that the allocation of water is inefficient, unsustainable and inequitable. The irrigation infrastructure is deteriorating. Illiteracy, general socio-economic backwardness, slow progress in implementing land reforms and inadequate or inefficient finance and marketing services for farm produce. Very small (less than 20,000 m^2) size of land holdings due to fragmentation, land ceiling acts and family disputes. Such small holdings are often over-manned, resulting in disguised unemployment and low productivity of labor.

1.2 PURPOSE

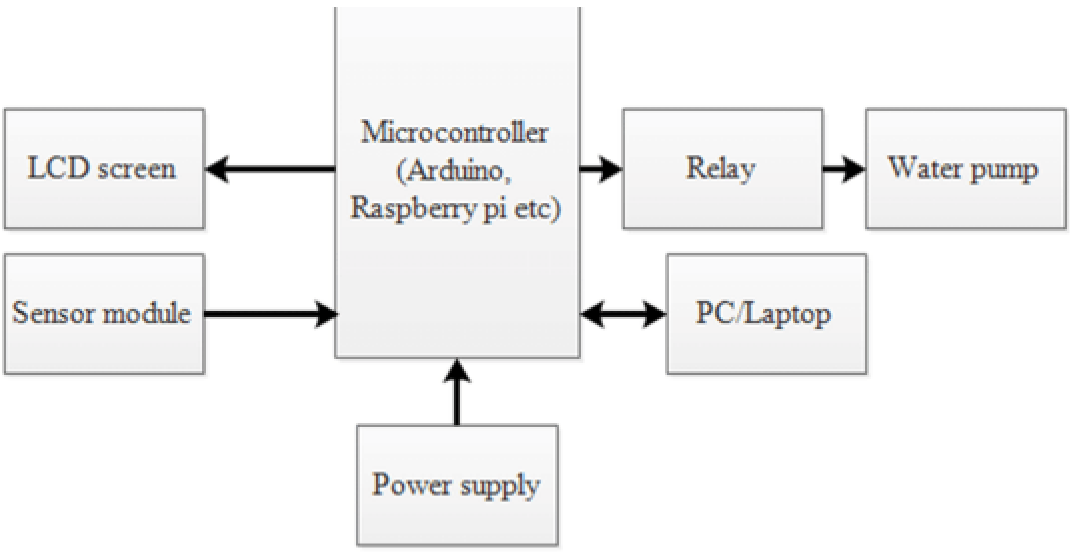
Illiteracy of farmers and their ignorance in the field of modern agricultural practices and technology, hampered by high costs and impracticality in the case of small land holdings. Inadequate Irrigation facilities and dependence of farmers on monsoon season, where good monsoon results in a vigorous growth while a poor monsoon leads to a sluggish growth for the economy as a whole. Ministry of Agriculture is also working in direction to improve the conditions of farmers by employing different programs like Insurance plan and ITC Limited plan. Under Insurance plan Agriculture Insurance Company of India insures farmers cultivating wheat, fruit, rice and rubber in the event of natural disasters or catastrophic crop failure, under the supervision of the Ministry of Agriculture. ITC Limited plan aims to connect 20,000 villages to the Internet by 2013 providing provide farmers with up-todate crop prices for the first time, which should minimize losses incurred from neighbouring producers selling early and in turn facilitate investment in rural areas.

(usually pronounced as "pick") is a family of Microcontroller made by Micro Technology, derived from the PIC1650 originally developed by General Instrument's Microelectronics Division. The name PIC initially referred to Peripheral Interface Controller,] and is currently expanded as Programmable Intelligent Computer. The first parts of the family were available in 1976; by 2013 the company had shipped more than twelve billion individual parts, used in a wide variety of embedded systems. Early models of PIC had read-only memory (ROM) or field programmable EPROM for program storage, some with provision for erasing memory. All current models use flash memory for program storage, and newer models allow the PIC to reprogram itself. Program memory and data memory are separated. Data memory is 8-bit, 16-bit, and, in latest models, 32-bit wide. Program instructions vary in bit-count by family of PIC, and may be 12, 14, 16, or 24 bits long. The instruction set also varies by model, with more powerful chips adding instructions for digital signal processing functions. A buzzer is a loud noise maker. Most modern ones are civil defense or air- raid sirens, tornado sirens, or the sirens on emergency service vehicles such as ambulances, police cars and fire trucks. There are two general types, pneumatic and electronic. A buzzer or beeper is an audio Signaling device, which be mechanical, electromechanical, or piezoelectric (piezo for short).

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

The socio-economic backgrounds of farmers differed little across participating farms, with a mean age of 22 to 14years. Most farmers (71%) had attained a primary school education level, but only 29% had achieved higher. The average farm size and perimeter length was 3900m2_2300 and 890m_315, respect. Not correlated with distance to forest edge Farmers grew their crops for subsistence (56%) and commercial (44%) purposes. Among the subsistence crops edible to wildlife species, all farmers grew chickpea and wheat as their main crops, while banana, chili, maize, papaya plant and water spinach were among the other subsistence crops. The main commercial crops grown were wheat (62%), soybean (18%) or a combination of both (20%) crops, which are unpalatable to most mammal species. Most farmers thought that cutting down the forest would increase flooding (94%), soil erosion (88%) and attacks from insect crop pests (66%). In our proposed work, when the animal enters into the farm area. The LDR's placed in the vertical positions help us to detect the size of the animal whereas PIR sensors are used to detect position of the animal.



Immediately, the APR board will be on, and the sound is played to divert the animal. During night time the flash light will be on and the message will be sent to the farmer. The LCD display the presence of animal and LDR readings. The GSM module is used for sending a message to warn the farmer about the intrusion.

This device is using Embedded PIC Microcontroller. It comprises LCD (16×2) (JHD162A), PIC Microcontroller, PIEZO Buzzer, GSM based SIM900A module, rheostat (10k), battery 9v, LED. Whenever there is attack by animals by crops in agriculture field, this system detects sound produced by buzzer and generate SMS alert within seconds to field owner. This device is based on motion detecting sensor and is developed especially for crop monitoring in agriculture fields, farms, wet lands, forests etc. GSM technology is use to send SMS alert to user on mobile whenever there is fire broken out in field. It will also generate buzzer sound to alarm nearby people to take proper action to diminish crops protected by smart farming.

Electric fences were used to control livestock in the United States in the early 1930s, and electric fencing technology developed in both the United States and New Zealand. An early application of the electric fence for livestock control was developed in 1936–1937 by New Zealand inventor Bill Gallagher. One of the major disadvantages of having an electric fence installed is that it requires regular maintenance. There are many rules and regulations involved, one that you need to check with local council to ensure even installing one is approved. Another is that you must constantly maintains the surrounding plant life. If trees and grass are not properly trimmed back, they could be considered a fire hazard. During night time the flash light will be on and the message will be sent to the farmer. The LCD display the presence of animal and LDR readings. The GSM module is used for sending a message to warn the farmer about the intrusion.

2.2 REFERENCE

i. Mr.Pranav shitap, Mr.Jayesh redij, Mr.Shikhar Singh, Mr.Durvesh Zagade, Dr. Sharada Chougule. Department of ELECTRONICS AND TELECOMMUNICATION ENGINEERING, Finolex Academy of Management and technology, ratangiri, India.

ii. N.Penchalaiah, D.Pavithra, B.Bhargavi, D.P.Madhurai, K.EliyasShaik,S.Md.sohaib.Assitant Professor, Department of CSE,AITS, Rajampet,India UG Student, Department of CSE,AITS,Rajampet, India.

iii. Mr.P.Venkateswara Rao, Mr.Ch Shiva Krishna ,MR M Samba Siva ReddyLBRCE,LBRCE,LBRCE.

iv. Mohit Korche,Sarthak Tokse, ShubhamShirbhate, Vaibhav Thakre,S. P. Jolhe(HOD). Students , Final Year,Dept.of Electrical engineering,Government

College of engineering,Nagpur head of dept.,Electrical engineering,Government College of engineering,Nagpur.

2.3 PROBLEM STATEMENT DEFINITION

Climate-smart agriculture (CSA) (or climate resilient agriculture) is an integrated approach to managing landscapes to help adapt agricultural methods, livestock and crops to the effects of climate change and, where possible, counteract it by reducing greenhouse gas emissions from agriculture, at the same time taking into account the growing world population to ensure food security. Thus, the emphasis is not simply on carbon farming or sustainable agriculture, but also on increasing agricultural productivity. "CSA ... is in line with FAO's vision for Sustainable Food and Agriculture and supports FAO's goal to make agriculture, forestry and fisheries more productive and more sustainable".

Global Livestock Environmental Assessment Model (GLEAM): This simulates the interaction of activities and processes involved in livestock production (milk and meat production) and the environment. The model is designed to evaluate several environmental impact categories, such as greenhouse gas emissions, nutrient and water use, land use and land degradation and biodiversity interactions.

Sustainability Assessment of Food and Agriculture (SAFA) system: The guidelines of SAFA is a framework for sustainability performance assessment in the food and agriculture sector, including crop and livestock production, forestry and fisheries. The monitoring and evaluation of activities set baselines, define indicators, measure progress and evaluate successes and setbacks in CSA interventions.

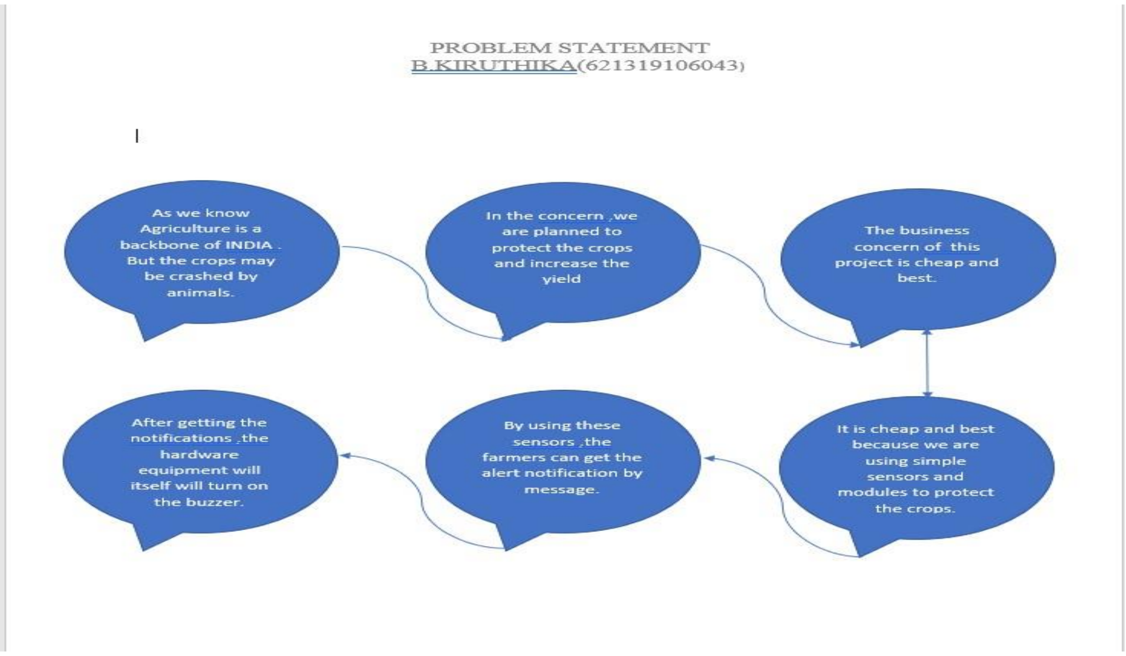
Economics and Policy Innovations for Climate-Smart Agriculture (EPIC): The programme works with governments, universities, research centres and other institutional partners in support of their transition to CSA through economic and policy analysis. It does this by identifying and harmonizing climate-smart agricultural policies, impacts analysis, effects, costs and benefits as well as incentives and barriers to the adoption of climate-smart agricultural practices.

Ex-Ante Carbon-balance Tool (EX-ACT): This appraisal system was developed by FAO. In the project development phase, it provides ex-ante estimates of the impact of agriculture and forestry development projects, programmes and policies on the carbon-balance.

Climate Risk Management (CRM): This integrated approach addresses vulnerabilities to short-term climate variability and longer-term climate change within the framework of sustainable development. The key component of the FAO's CRM involves the provision of weather and climate information products for farmers, fishers and livestock herders for the assessment of risks so as to improve opportunities at local level.

Gender mainstreaming: In order to achieve CSA in a socially sustainable way; there is a need to understand the roles, capabilities and responsibilities of men and women to ensure equal access to CSA policies and practices benefits.

Monitoring and Assessment of Greenhouse Gas Emissions and Mitigation Potential in Agriculture (MAGHG) project: This project falls under the MICCA (Mitigation of Climate Change in Agriculture) programme. Under this project, member countries are supported in gathering and reporting data on GHG emissions in the agriculture, forestry and other land use (AFOLU) sector for UNFCCC related reporting requirements.



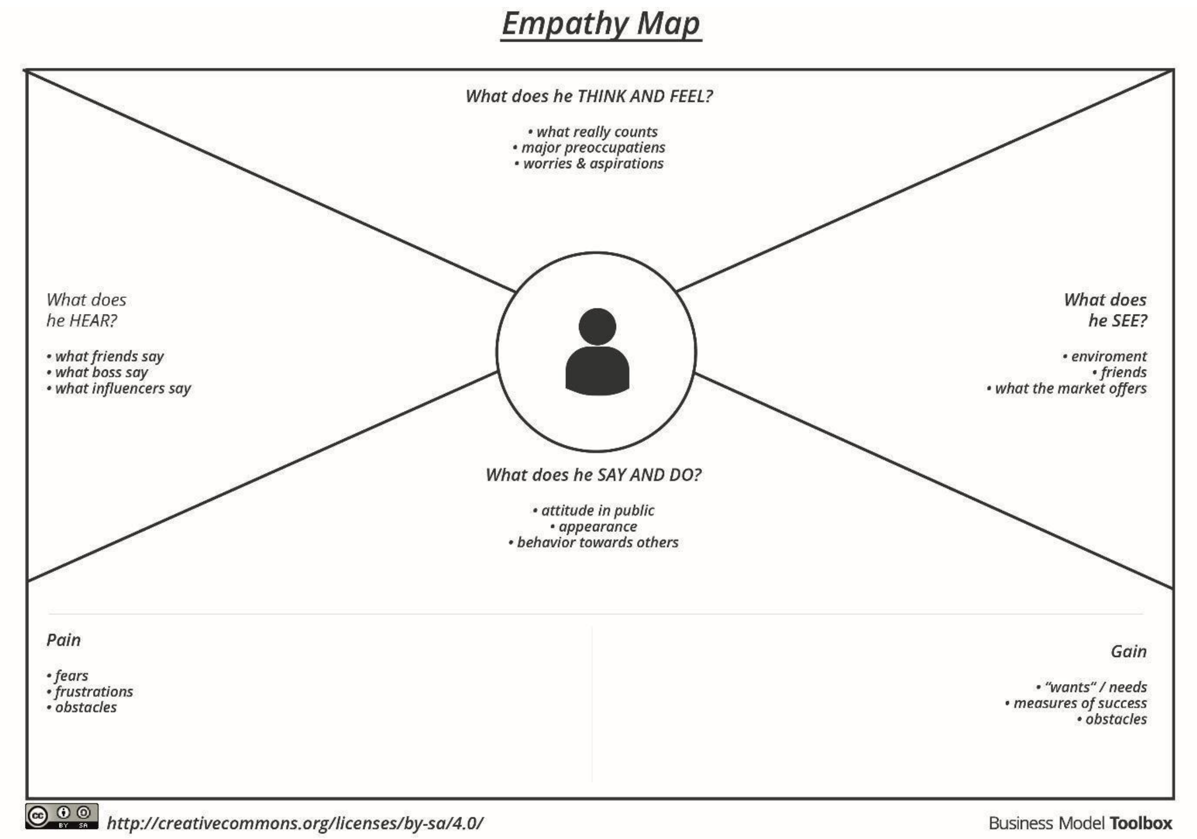
Carbon farming is a name for a variety of agricultural methods aimed at sequestering atmospheric carbon into the soil and in crop roots, wood and leaves. The aim of carbon farming is to increase the rate at which carbon is sequestered into soil and plant material with the goal of creating a net loss of carbon from the atmosphere. Increasing a soil's organic matter content can aid plant growth, increase total carbon content, improve soil water retention capacity and reduce fertilizer use. As of 2016, variants of carbon farming reached hundreds of millions of hectares globally, of the nearly 5 billion hectares (1.2×10^{10} acres) of world farmland. In addition to agricultural activities, forests management is also a tool that is used in carbon farming.

3.IDEATION AND PROPOSED SOLUTION

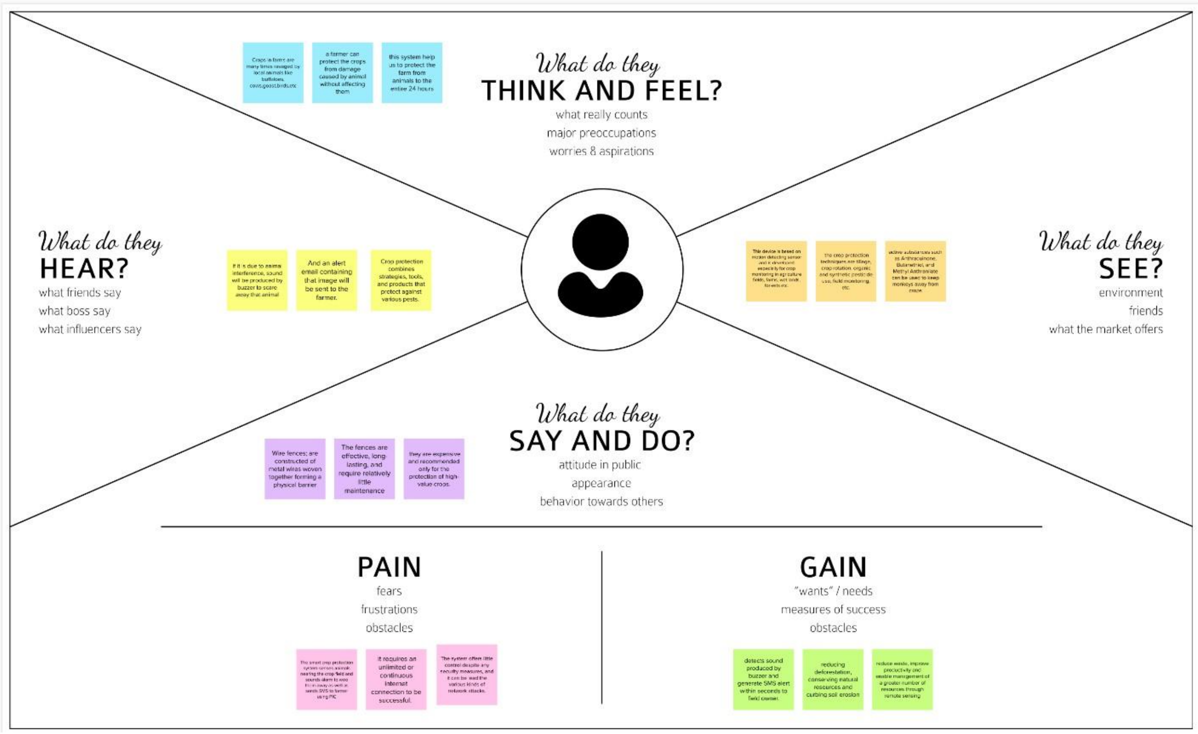
3.1 EMPATHY MAP CANVAS

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

It is a useful tool to helps teams better understand their users.
Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

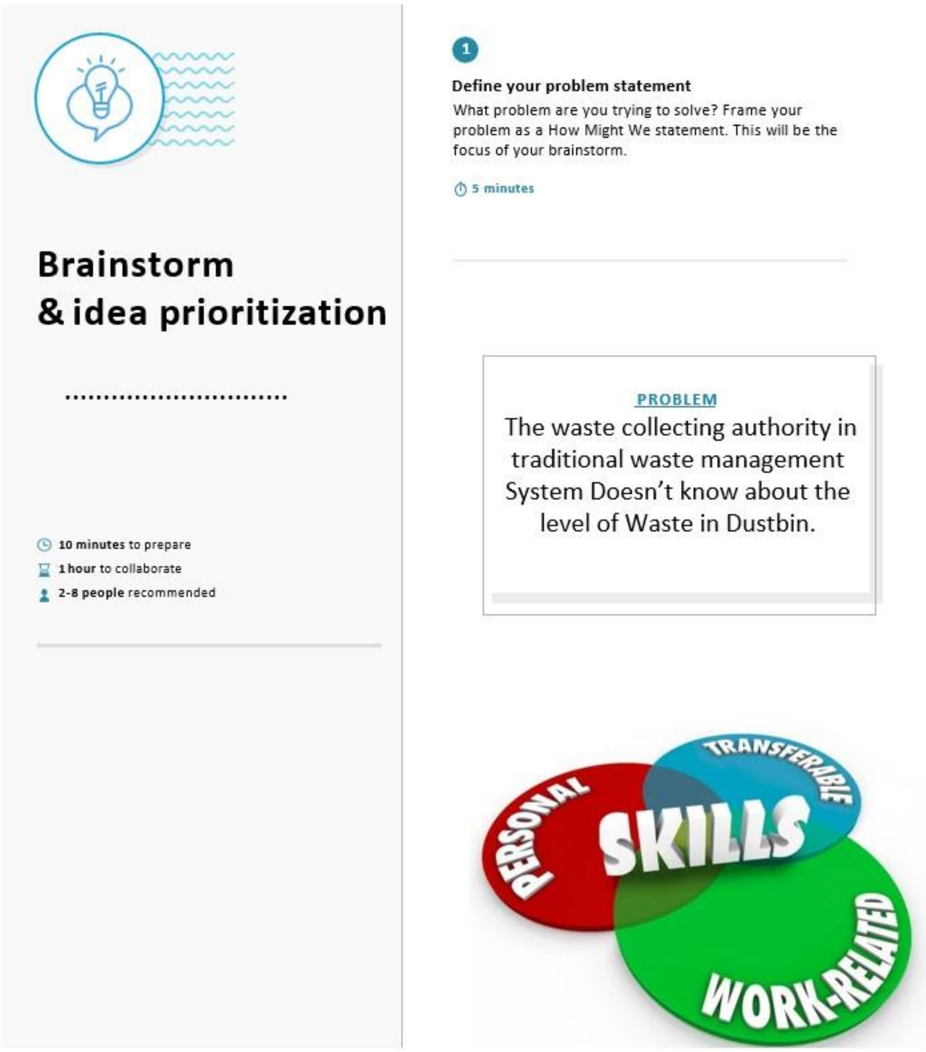


Reference:
https://app.mural.co/invitation/mural/hemanithi8707/1664113912019?sender=u815ea915f388 dc7374017715&key=8ace806a-9163-461e-bef9-c4bec2daf69f



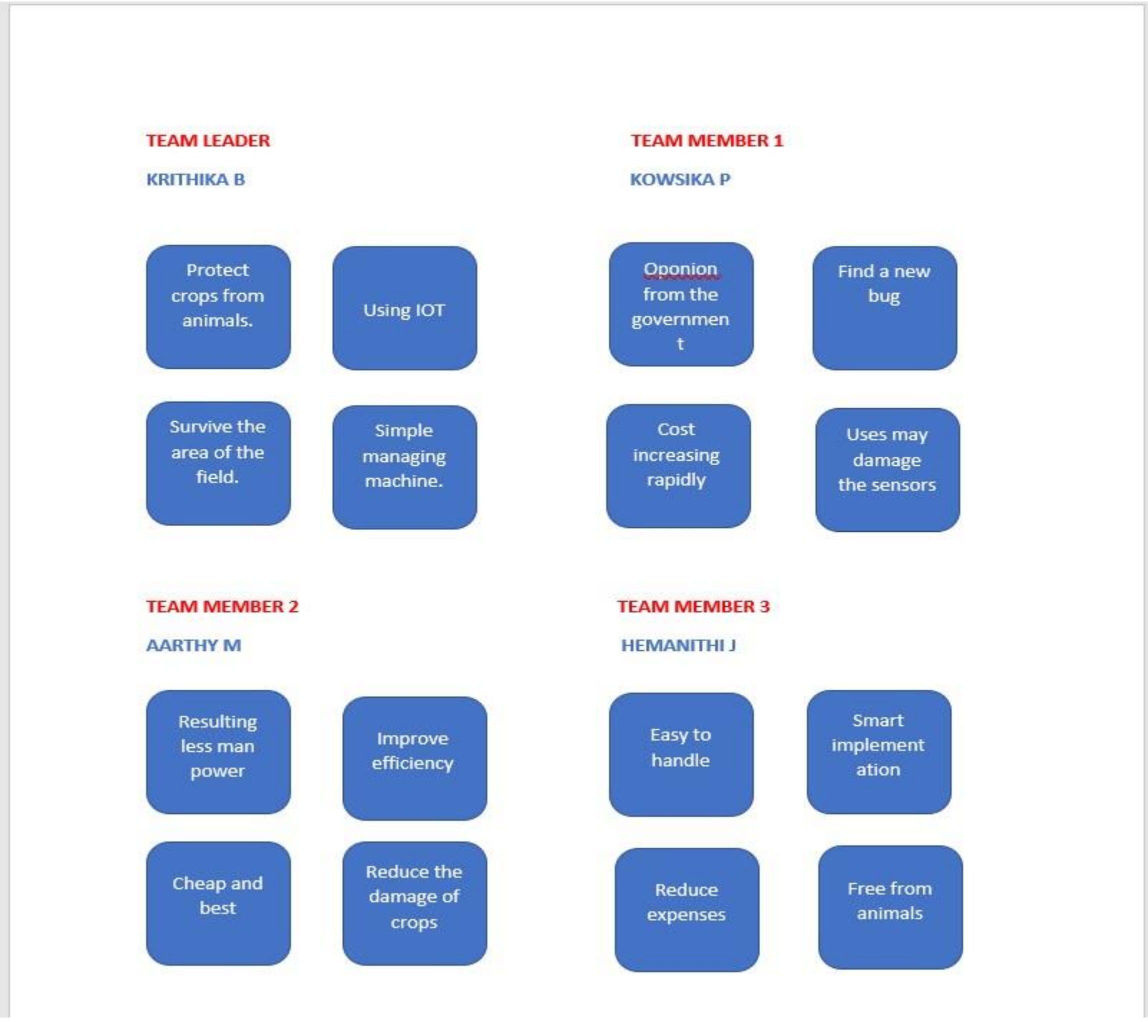
Men, women, boys, and girls are affected by climate change in different ways. To increase the effectiveness and sustainability of CSA interventions, they must be designed to address gender inequalities and discriminations against people at risk. Gender gap in agriculture implies that men and women farmers have varying access to resources to prepare for and respond to climate change. Women farmers are more prone to climate risk compared to men. It has been reported that in developing countries, women have less access compared to men to productive resources, financial capital, and advisory services. They often tend to be excluded from decision making which may impact on their adoption of technologies and practices that could help them adapt to climatic conditions. A gender-responsive approach to CSA tries to identify and address the diverse constraints faced by men and women and recognizes their specific capabilities. Climate Smart Agriculture presents opportunities for women in agriculture to engage in sustainable production. Climate change affects men and women differently. There is need to level the field and CSA is an opportunity for women in agriculture to engage more productively. Gender gap in agriculture implies that men and women farmers have varying access to resources to prepare for and respond to climate change. Women farmers are more prone to climate risk compared to men. It has been reported that in developing countries, women have less access compared to men to productive resources, financial capital, and advisory services.

3.2 IDEATION AND BRAINSTORMING



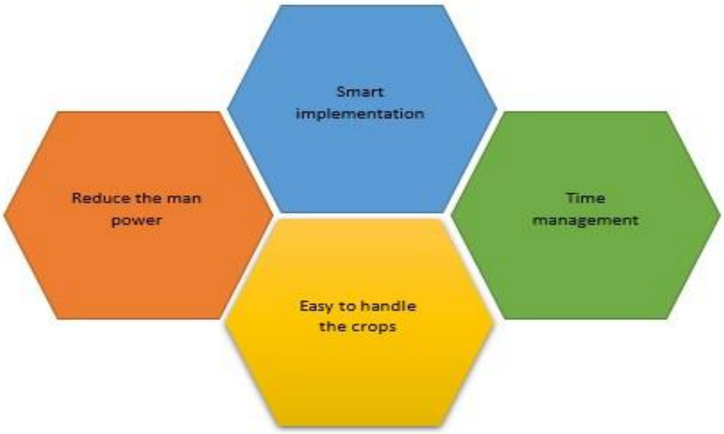
Precision agriculture (PA) is a farming management concept based on observing, measuring and responding to inter and intra-field variability in crops. First conceptual work on PA and practical applications go back in the late 1980s. The goal of precision agriculture research is to define a decision support system (DSS) for whole farm management with the goal of optimizing returns on inputs while preserving resources.

Among these many approaches is a phytogeomorphological approach which ties multi-year crop growth stability/characteristics to topological terrain attributes. The interest in the phytogeomorphological approach stems from the fact that the geomorphology component typically dictates the hydrology of the farm field.This data is used in conjunction with satellite imagery by variable rate technology (VRT) including seeders, sprayers, etc. to optimally distribute resources. However, recent technological advances have enabled the use of real-time sensors directly in soil, which can wirelessly transmit data without the need of human presence.



PUBLIC:

GOVERNMENT



Precision agriculture has also been enabled by unmanned aerial vehicles that are relatively inexpensive and can be operated by novice pilots. These agricultural drones can be equipped with multispectral or RGB cameras to capture many images of a field that can be stitched together using photogrammetric methods to create orthophotos. These multispectral images contain multiple values per pixel in addition to the traditional red, green blue values such as near infrared and red-edge spectrum values used to process and analyze vegetative indexes such as NDVI maps.[11] These drones are capable of capturing imagery and providing additional geographical references such as elevation, which allows software to perform map algebra functions to build precise topography maps. These topographic maps can be used to correlate crop health with topography, the results of which can be used to optimize crop inputs such as water, fertilizer or chemicals such as herbicides and growth regulators through variable rate applications.

3.3 PROPOSED SOLUTION

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	**To protect the crops from damage caused by animals as well as divert the animal without any harm.crops in farms are many times ravaged by local animals like buffaloes,cows,goats,birds etc.. this leads to huge losses for the farmers. We proposed a automatic crop protection system from animals and also animal detection system id designed to detect the presence of animal.**
2.	Idea / Solution description	As we know agriculture is the backbone of INDIA this may help the farmers to protect their crops from the animals.
3.	Novelty / Uniqueness	The farmers can get the notification message by using this technique.
4.	Social Impact / Customer Satisfaction	May be the animals can crash the hardware.
5.	Business Model (Revenue Model)	It can be applied to an business model like in forest areas.
6.	Scalability of the Solution	Its scalability is upto measurable distance like 50meter.

Each farmer produced enough food to feed about 26 people during this time. The 1960s prompted the Green Revolution with new methods of genetic modification, which led to each farmer feeding about 156 people.

3.4 PROBLEM SOLUTION FIT

Precision agriculture uses many tools but here are some of the basics: tractors, combines, sprayers, planters, diggers, which are all considered auto-guidance systems. The small devices on the equipment that uses GIS (geographic information system) are what makes precision ag what it is. You can think of the GIS system as the “brain.” To be able to use precision agriculture the equipment needs to be wired with the right technology and data systems. More tools include Variable rate technology (VRT), Global positioning system and Geographical information system, Grid sampling, and remote sensors. Intra and inter-field variability may result from a number of factors. These include climatic conditions (hail, drought, rain, etc.), soils (texture, depth, nitrogen levels), cropping practices (no-till farming), weeds and disease. Permanent indicators—chiefly soil indicators—provide farmers with information about the main environmental constants. Point indicators allow them to track a crop's status, i.e., to see whether diseases are developing, if the crop is suffering from water stress, nitrogen stress, or lodging, whether it has been damaged by ice and so on. This information may come from weather stations and other sensors (soil electrical resistivity, detection with the naked eye, satellite imagery, etc.). Soil resistivity measurements combined with soil analysis make it possible to measure moisture content. Soil resistivity is also a relatively simple and cheap measurement.Predictive approach: based on analysis of static indicators (soil, resistivity, field history, etc.) during the crop cycle.Control approach: information from static indicators is regularly updated during the crop cycle by sampling: weighing biomass, measuring leaf chlorophyll content, weighing fruit, etc.remote sensing: measuring parameters like temperature (air/soil), humidity (air/soil/leaf), wind or stem diameter is possible thanks to Wireless Sensor Networks[19] and Internet of things (IoT) proxy-detection: in-vehicle sensors measure leaf status; this requires the farmer to drive around the entire field.aerial or satellite remote sensing: multispectral imagery is acquired and processed to derive maps of crop biophysical parameters, including indicators of disease. Airborne instruments are able to measure the amount of plant cover and to distinguish between crops and weeds.

Decisions may be based on decision-support models (crop simulation models and recommendation models) based on big data, but in the final analysis it is up to the farmer to decide in terms of business value and impacts on the environment- a role being takenover by artificial intelligence (AI) systems based on machine learning and artificial neural networks.It is important to realize why PA technology is or is not adopted, "for PA technology adoption to occur the farmer has to perceive the technology as useful and easy to use. It might be insufficient to have positive outside data on the economic benefits of PA technology as perceptions of farmers have to reflect these economic considerations." To be able to use precision agriculture the equipment needs to be wired with the right technology and data systems. More tools include Variable rate technology (VRT), Global positioning system and Geographical information system, Grid sampling, and remote sensors. Intra and inter-field variability may result from a number of factors. These include climatic conditions (hail, drought, rain, etc.), soils (texture, depth, nitrogen levels), cropping practices (no-till farming), weeds and disease. Permanent indicators—chiefly soil indicators—provide farmers with information about the main environmental constants. Point indicators allow them to track a crop's status, i.e., to see whether diseases are developing, if the crop is suffering from water stress, nitrogen stress, or lodging, whether it has been damaged by ice and so on. This information may come from weather stations and other sensors.

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

The concept of precision agriculture first emerged in the United States in the early 1980s. In 1985, researchers at the University of Minnesota varied lime inputs in crop fields. It was also at this time that the practice of grid sampling appeared (applying a fixed grid of one sample per hectare). Towards the end of the 1980s, this technique was used to derive the first input recommendation maps for fertilizers and pH corrections. The use of yield sensors developed from new technologies, combined with the advent of GPS receivers, has been gaining ground ever since. Today, such systems cover several million hectares.

In the American Midwest (US), it is associated not with sustainable agriculture but with mainstream farmers who are trying to maximize profits by spending money only in areas that require fertilizer. This practice allows the farmer to vary the rate of fertilizer across the field according to the need identified by GPS guided Grid or Zone Sampling. Fertilizer that would have been spread in areas that don't need it can be placed in areas that do, thereby optimizing its use.

Around the world, precision agriculture developed at a varying pace. Precursor nations were the United States, Canada and Australia. In Europe, the United Kingdom was the first to go down this path, followed closely by France, where it first appeared in 1997-1998. In Latin America the leading country is Argentina, where it was introduced in the middle 1990s with the support of the National Agricultural Technology Institute. Brazil established a state-owned enterprise, Embrapa, to research and develop sustainable agriculture. The development of GPS and variable-rate spreading techniques helped to anchor precision farming management practices. Today, less than 10% of France's farmers are equipped with variable-rate systems. Uptake of GPS is more widespread, but this hasn't stopped them using precision agriculture services, which supplies field-level recommendation maps.

One third of the global population still relies on agriculture for a living. Although more advanced precision farming technologies require large upfront investments, farmers in developing countries are benefitting from mobile technology. This service assists farmers with mobile payments and receipts to improve efficiencies. For example, 30,000 farmers in Tanzania use mobile phones for contracts, payments, loans, and business organization.

The economic and environmental benefits of precision agriculture have also been confirmed in China, but China is lagging behind countries such as Europe and the United States because the Chinese agricultural system is characterized by small-scale family-run farms, which makes the adoption rate of precision agriculture lower than other countries. Therefore, China is trying to better introduce precision agriculture technology into its own country and reduce some risks, paving the way for China's technology to develop precision agriculture in the future. The development of GPS and variable-rate spreading techniques helped to anchor precision farming management practices. Today, less than 10% of France's farmers are equipped with variable-rate systems. Uptake of GPS is more widespread, but this hasn't stopped them using precision agriculture services, which supplies field-level recommendation maps.

4.2 NON-FUNCTIONAL REQUIREMENTS

Precision agriculture, as the name implies, means application of precise and correct amount of inputs like water, fertilizer, pesticides etc. at the correct time to the crop for increasing its productivity and maximizing its yields. Precision agriculture management practices can significantly reduce the amount of nutrient and other crop inputs used while boosting yields. Farmers thus obtain a return on their investment by saving on water, pesticide, and fertilizer costs.The second, larger-scale benefit of targeting inputs concerns environmental impacts. Applying the right amount of chemicals in the right place and at the right time benefits crops, soils and groundwater, and thus the entire crop cycle. Consequently, precision agriculture has become a cornerstone of sustainable agriculture, since it respects crops, soils and farmers. Sustainable agriculture seeks to assure a continued supply of food within the ecological, economic and social limits required to sustain production in the long term.A 2013 article tried to show that precision agriculture can help farmers in developing countries like India. Precision agriculture reduces the pressure by the agriculture on the environment by increasing the efficiency of machinery and putting it into use. For example, the use of remote management devices such as GPS reduces fuel consumption for agriculture, while variable rate application of nutrients or pesticides can potentially reduce the use of these inputs, thereby saving costs and reducing harmful runoff into the waterways. Self-steering tractors have existed for some time now, as John Deere equipment works like a plane on autopilot. The tractor does most of the work, with the farmer stepping in for emergencies. Technology is advancing towards driverless machinery programmed by GPS to spread fertilizer or plow land. Autonomy of technology is driven by the demanding need of diagnoses, often difficult to accomplish solely by hands-on farmer-operated machinery. In many instances of high rates of production, manual adjustments cannot sustain. Other innovations include a solar powered machine that identifies weeds and precisely kills them with a dose of herbicide or lasers. Agricultural robots, also known as AgBots, already exist, but advanced harvesting robots are being developed to identify ripe fruits, adjust to their shape and size, and carefully pluck them from branches.

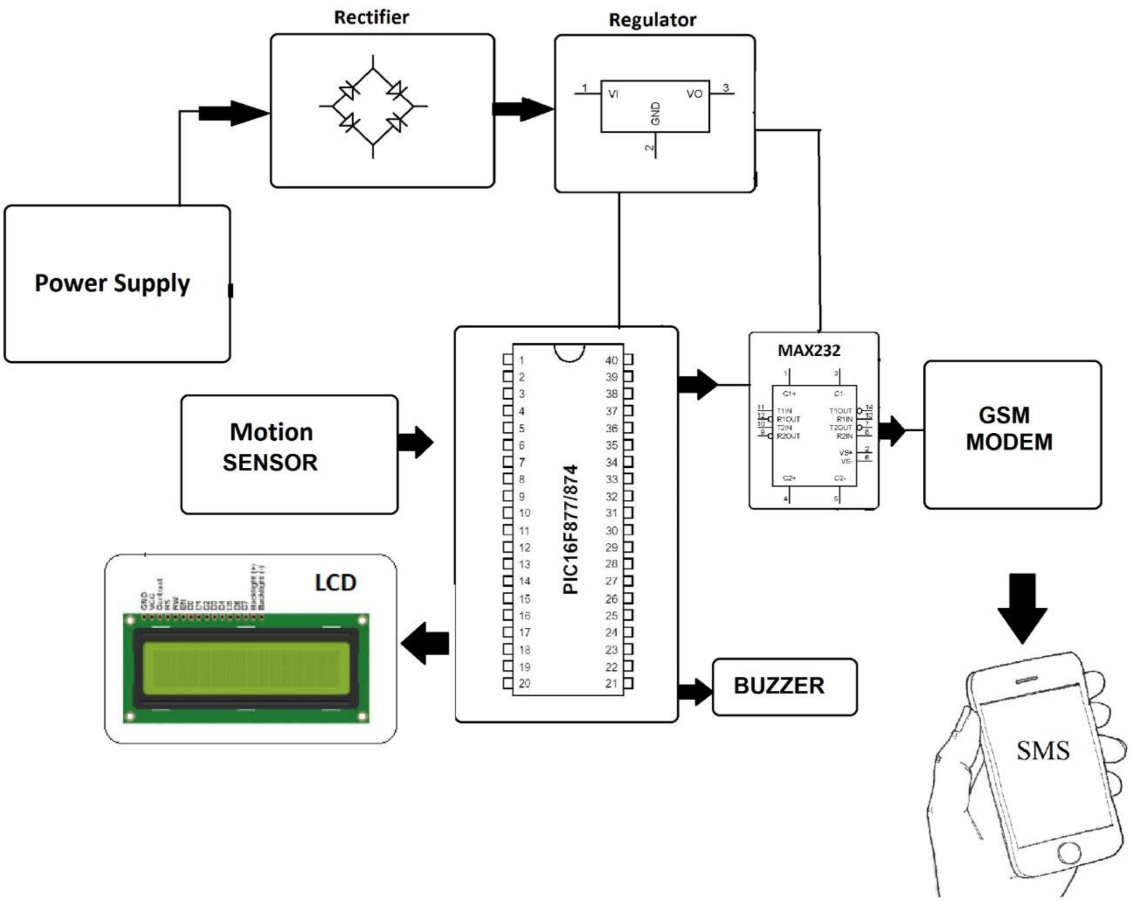
The Internet of things is the network of physical objects outfitted with electronics that enable data collection and aggregation. IoT comes into play with the development of sensors and farm-management software. For example, farmers can spectroscopically measure nitrogen, phosphorus, and potassium in liquid manure, which is notoriously inconsistent. They can then scan the ground to see where cows have already urinated and apply fertilizer to only the spots that need it. This cuts fertilizer use by up to 30%.Moisture sensors in the soil determine the best times to remotely water plants. The irrigation systems can be programmed to switch which side of tree trunk they water based on the plant's need and rainfall.

Innovations are not just limited to plants—they can be used for the welfare of animals. Cattle can be outfitted with internal sensors to keep track of stomach acidity and digestive problems. External sensors track movement patterns to determine the cow's health and fitness, sense physical injuries, and identify the optimal times for breeding. All this data from sensors can be aggregated and analyzed to detect trends and patterns.

As another example, monitoring technology can be used to make beekeeping more efficient. Honeybees are of significant economic value and provide a vital service to agriculture by pollinating a variety of crops.

5.PROJECT DESIGN

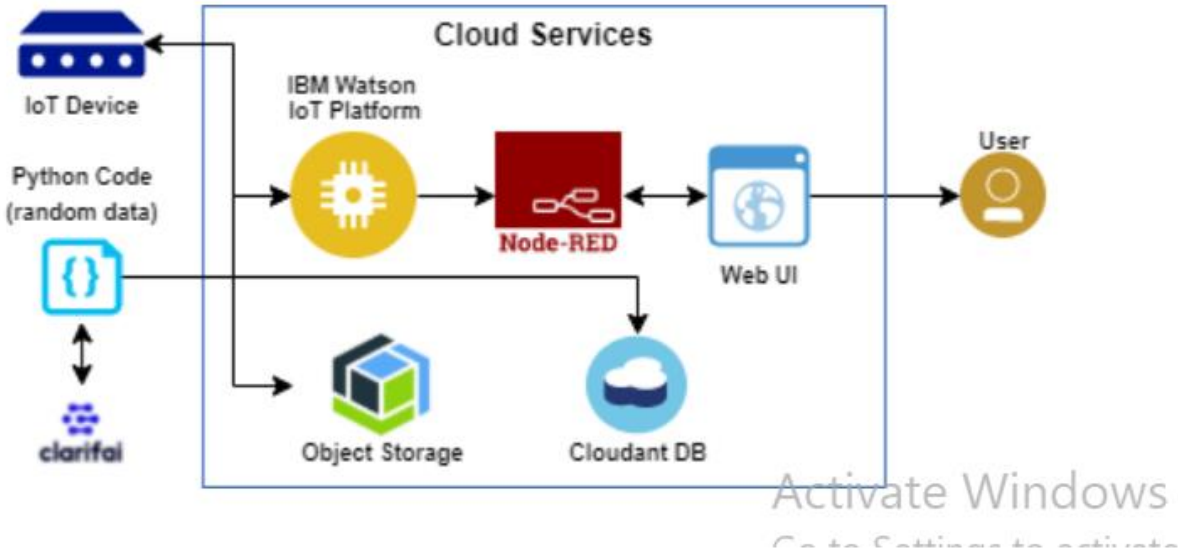
5.1 DATA FLOW DIAGRAM



The history of agriculture has been shaped by technological advances. Historians have described a number of agricultural revolutions, which identify major shifts in agricultural practice and productivity. These revolutions have been closely connected to technological improvements. Irrigation technology was developed independently by a number of different cultures, with the earliest known examples dated to the 6th millennium BCE in Khuzistan in the south-west of present-day Iran. A major turning point for agricultural technology is the Industrial Revolution, which introduced agricultural machinery to mechanise the labour of agriculture, greatly increasing farm worker productivity. In modern mechanised agriculture powered machinery has replaced many farm jobs formerly carried out by manual labour or by working animals such as oxen, horses and mules.

5.2 SOLUTION AND TECHICAL ARCHITECTURE

The 20th century saw major advances in agricultural technologies, including the development of synthetic fertilizers and pesticides, and new agricultural machinery including mass produced tractors and agricultural aircraft for aerial application of pesticides. More recent advances have included agricultural plastics, genetically modified crops, improved drip irrigation, and soilless farming techniques such as hydroponics, aquaponics, and aeroponics. In the first decades of the 21st century, Information Age technologies have been increasingly applied to agriculture. Agricultural robots, agricultural drones and driverless tractors have found regular use on farms, while digital agriculture and precision agriculture make use of extensive data collection and computation to improve farm efficiency.[7] Precision agriculture includes such areas as precision beekeeping, precision livestock farming, and precision viticulture. Agro-textiles is the segmented class of technical textiles that deals focuses on the agriculture sector, with an approach to crop protection and crop development and reducing the risks of farming practices. Primarily agro-textiles offer weather resistance and resistance to microorganisms and protection from unwanted elements and external factors. Agro-textiles helps to improve the overall conditions with which crop can develop and be protected. There are the various textile products, fabrics forms, fibers and techniques used in agro-textiles which are useful for agriculture mainly for crop protection and in crop development for instance shade nets, thermal insulation and sunscreen materials, windshield, antibird nets, which provide minimal shading and proper temperature, air circulation for protecting plants from direct sunlight and birds. Agrotextiles involves mulch mats, hail protection nets, and crop covers, etc. Agro-textiles are useful in Horticulture, aquaculture, landscape gardening and forestry also. More examples of use and application are covering livestock protection, suppressing weed and insect control, etc.



5.3 USER STORIES

Push–pull technology involves use of behaviour-modifying stimuli to manipulate the distribution and abundance of stemborers and beneficial insects for management of stemborer pests. It is based on in-depth understanding of chemical ecology, agrobiodiversity, plant-plant and insect-plant interactions, and involves intercropping a cereal crop with a repellent intercrop such as Desmodium uncinatum (silverleaf) (push), with an attractive trap plant such as Napier grass (pull) planted as a border crop around this intercrop. Gravid stemborer females are repelled from the main crop and are simultaneously attracted to the trap crop. The "push" in the intercropping scheme is provided by the plants that emit volatile chemicals (kairomones) which repel stemborer moths and drive them away from the main crop (maize or sorghum). The most commonly used species of push plants are legumes of the genus Desmodium (e.g. silverleaf Desmodium, D. uncinatum, and greenleaf Desmodium, D. intortum). The Desmodium is planted in between the rows of maize or sorghum, where they emit volatile chemicals (such as (E)-β-ocimene and (E)-4,8-dimethyl-1,3,7-nonatriene) that repel the stemborer moths. These semiochemicals are also produced in grasses such as maize when they are damaged by insect herbivores, which may explain why they are repellent to stemborers. Being a low-growing plant, Desmodium does not interfere with the growth of crops, but can suppress weeds and help improve soil quality by increasing soil organic matter content, fixing nitrogen, and stabilizing soils from erosion. It also serves as a highly nutritious animal feed and effectively suppresses striga weeds through an allelopathic mechanism. Another plant showing good repellent properties is molasses grass (Melinis minutiflora), a nutritious animal feed with tick-repelling and stemborer larval parasitoid attractive properties. The approach relies on a combination of companion crops to be planted around and among maize or sorghum. Both domestic and wild grasses can help to protect the crops by attracting and trapping the stemborers. The grasses are planted in the border around the maize and sorghum fields where invading adult moths become attracted to chemicals emitted by the grasses themselves. Instead of landing on the maize or sorghum plants, the insects head for what appears to be a tastier meal. These grasses provide the "pull" in the "push–pull" strategy. They also serve as a haven for the borers' natural enemies.

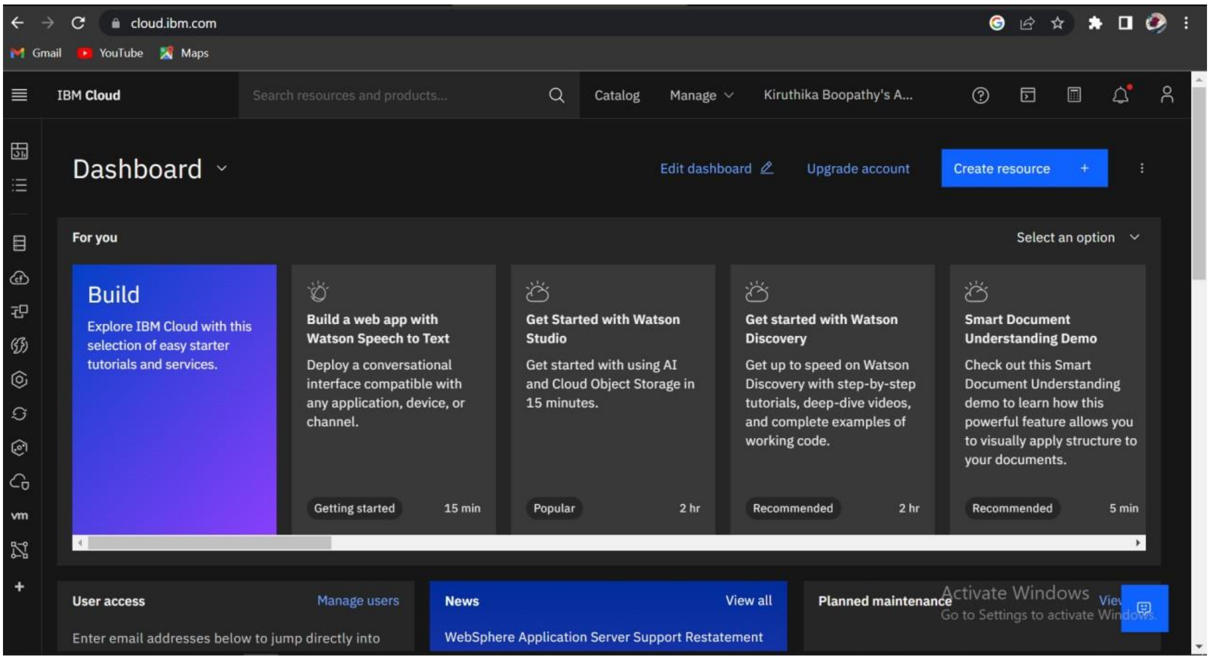
Good trap crops include well-known grasses such as Napier grass (Pennisetum purpureum), Signal grass (Brachiaria brizantha), and Sudan grass (Sorghum vulgare sudanense). Napier grass produces significantly higher levels of attractive volatile compounds (green leaf volatiles), cues used by gravid stemborer females to locate host plants, than maize or sorghum. There is also an increase of approximately 100-fold in the total amounts of these compounds produced in the first hour of nightfall by Napier grass (scotophase), the period at which stemborer moths seek host plants for laying eggs, causing the differential oviposition preference. However, many of the stemborer larvae, about 80%, do not survive, as Napier grass tissues produce sticky sap in response to feeding by the larvae, which traps them, causing the death of about 80% of larvae.

6.PROJECT PLANNING AND SCHEUDLING

6.1 SPRINT PLANNING AND ESTIMATION

STEP 1:

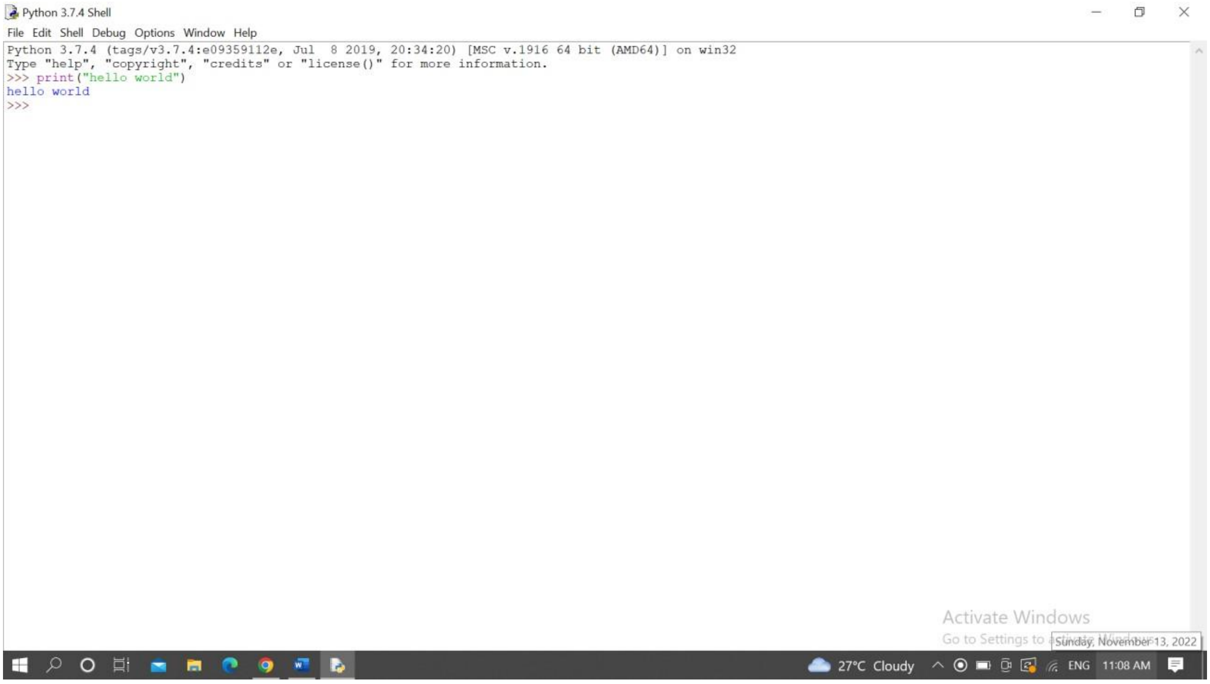
IBM Cloud Services



Desmodium also controls the parasitic weed, Striga, resulting in significant yield increases of about 2 tonnes/hectare (0.9 short tons per acre) per cropping season. In addition to benefits derived from increased nitrogen availability and competition for light, it was found that D. uncinatum strongly suppresses striga growth through allelopathy. These effects are thought to be related to isoflavanones produced in Desmodium roots, which can either promote the germination of striga seeds or inhibit seedling growth, depending on their structure. Together, these effects result in the phenomenon known as "suicidal germination", thus reducing the striga seed bank in the soil.[3] Other Desmodium species have also been evaluated and have similar effects on stemborers and striga weed and are currently being used as intercrops in maize, sorghum and millets.

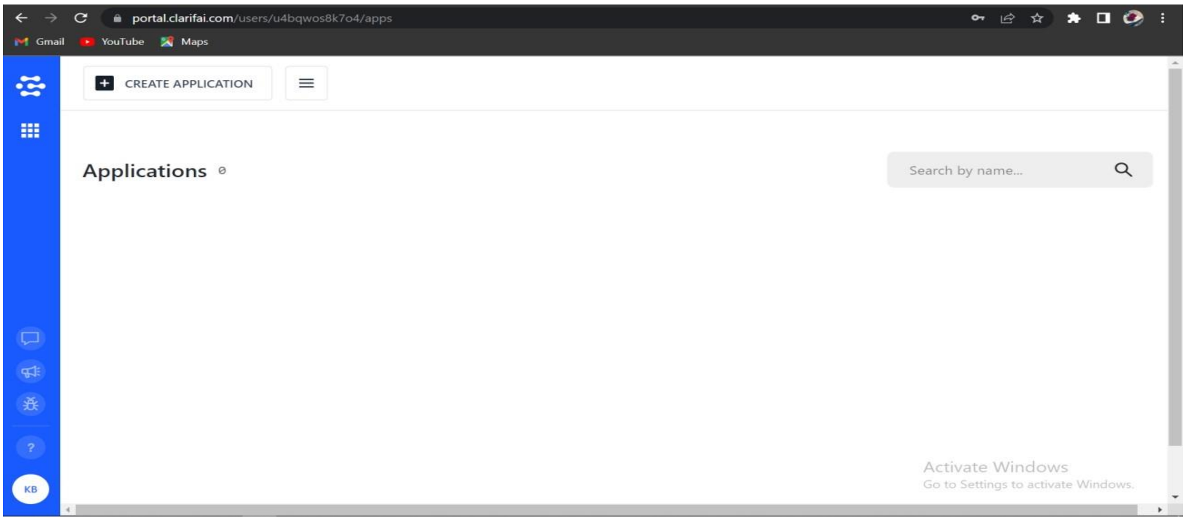
STEP 2:

SOFTWARE



STEP-3

Clarifai Account

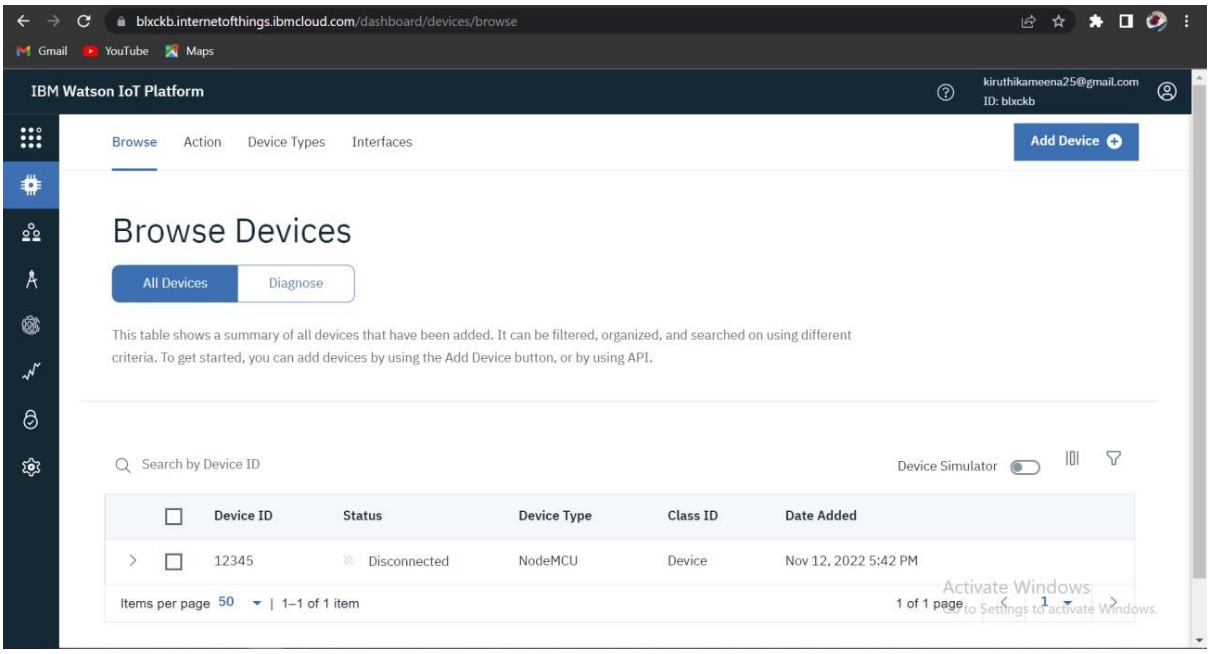


6.2 SPRINT DELIVERY SCHEDULE

Create and configure IBM cloud services:

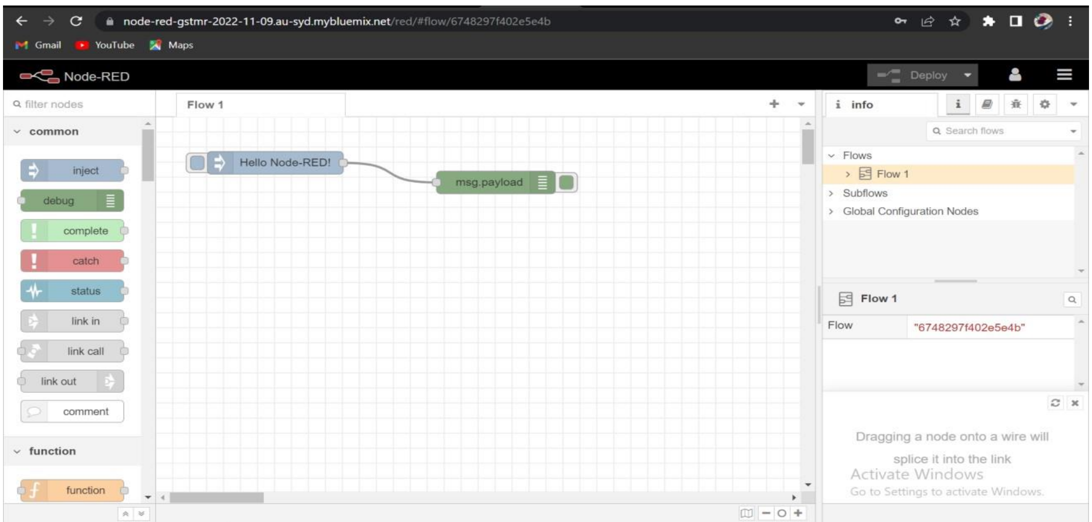
STEP 1:

IBM Watson IoT Platform and device



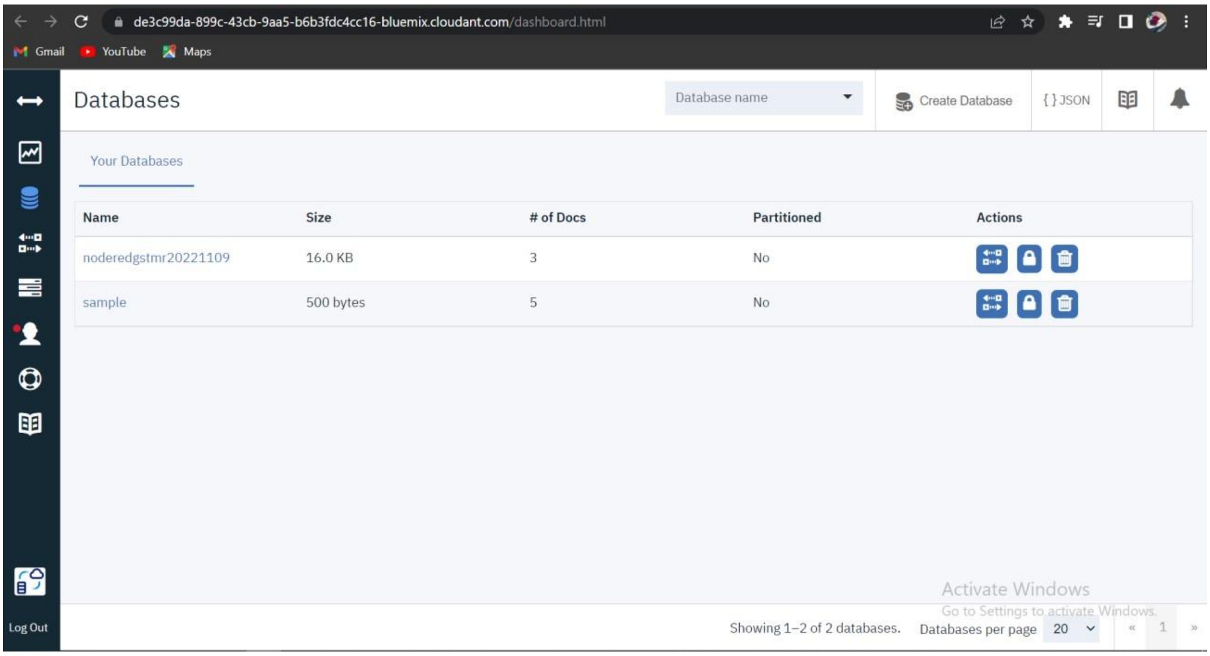
STEP 2:

Create Node-red services



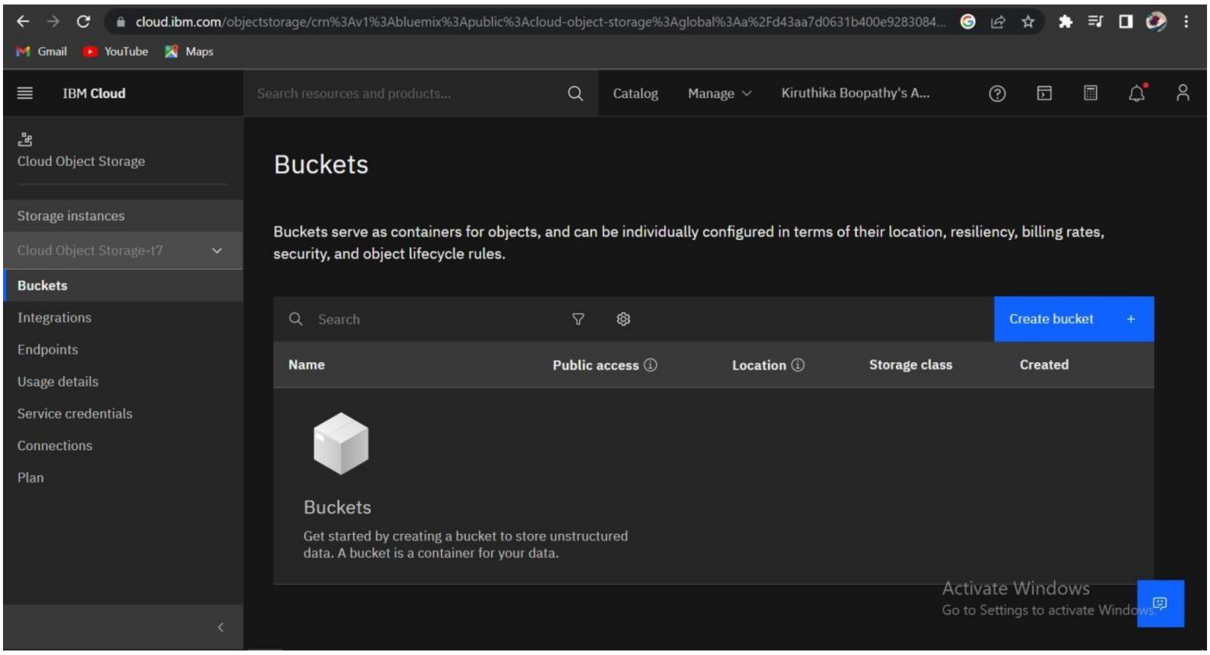
Step 3:

Create an databases in Cloudant db



Step 4:

Cloud object storage device



SPRINT 3

```
import cv2
import numpy as np
import wiotp.sdk.device
import playsound
import random
import time
import datetime
import ibm_boto3

from ibm_botocore.client import Config, ClientError
#CloudantDB
from cloudant.client import Cloudant
from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel
from clarifai_grpc.grpc.api import service_pb2_grpc
stub = service_pb2_grpc.V2Stub(ClarifaiChannel.get_grpc_channel())
from clarifai_grpc.grpc.api import service_pb2, resources_pb2
from clarifai_grpc.grpc.api.status import status_code_pb2
# This is how you authenticate.
metadata = (('authorization' , 'Key bc885e5165d74ef48f42f6f6a2c9eb87'),)
COS_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud" # Current list avaiable at https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints
COS_API_KEY_ID = " f6Ap-ct18m07S9UZL7XPbAF7170ome PLLUQOzqmnAzb5" # eg "W00YiRnLW4a3fTj MB-odB-2ySfTrFBIQQ'Wanc -- P3byk"
```

```
COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"

COS_RESOURCE_CRN = "crn:v1:bluemix:public:cloudantnosqldb:eu-gb:a/d43aa7d0631b400e9283084df08f9f60:502851d6-a240-4b22-8d4b-3642ed2bc3a8::" # eg "crn:vl:bluemix:public:cloud-object-storage:global:a/6b644a3fda97448b888c23eeef263ed6:199ab1e5-0d9d-420f-8e4a-98d868c04368 ::"

clientdb = Cloudant("apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2", "64455b04f35e5d5f9b4fc25bb38904af", url = "https://apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2:64455b04f35e5d5f9b4fc25bb38904af@de3c99da-899c-43cb-9aa5-b6b3fdc4cc16-bluemix.cloudantnosqldb.appdomain.cloud",

username = "apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2")

clientdb.connect()

#Create resource

cos=ibm_boto3.resource("s3",

ibm_api_key_id=COS_API_KEY_ID,

ibm_service_instance_id=COS_RESOURCE_CRN,

ibm_auth_endpoint=COS_AUTH_ENDPOINT,

config=Config(signature_version="oauth"),

endpoint_url=COS_ENDPOINT

)

def multi_part_upload(bucket_name, item_name, file_path) :

try:

print("Starting file transfer for {0} to bucket: {1}\n" .

format(item_name, bucket_name))

#set 5 MB chunks

part_size = 1024*1024 * 5

#set threadhold to 15 MB

file_threshold = 1024 * 1024 * 15
```

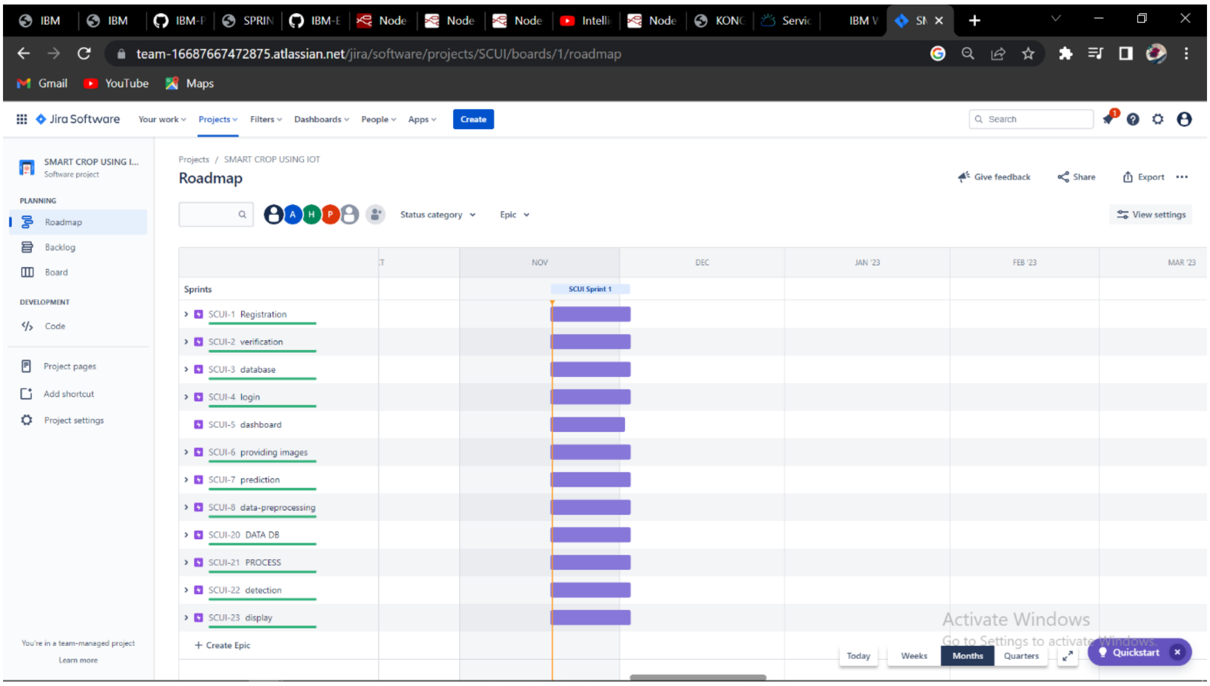
```
#set the transfer threshold and chunk size
transfer_config = ibm_boto3.s3.transfer.TransferConfig(
multipart_threshold=file_threshold,
multipart_chunksize=part_size
)
# the upload_fileobj method will automatically execute a multi-part upload
# in 5 MB chunks for all files over 15 MB
with open(file_path, "rb") as file_data:
cos.Object(bucket_name, item_name) .upload_fileobj(
Fileobj=file_data,
Config=transfer_config
)
print("Transfer for {0} Complete!\n". format(item_name))

except ClientError as be:
print("CLIENT ERROR: {0}\n" . format(be))
except Exception as e:
print("Unable to complete multi-part upload: {0}" .format(e))
def myCommandCallback(cmd) :
print("Command received: %s" % cmd.data)
command=cmd.data[ ' command']
print(command)
if(command =='lighton'):
print('lighton')
elif(command =='lightoff'):
print('lightoff')
elif(command =='motoron'):
```

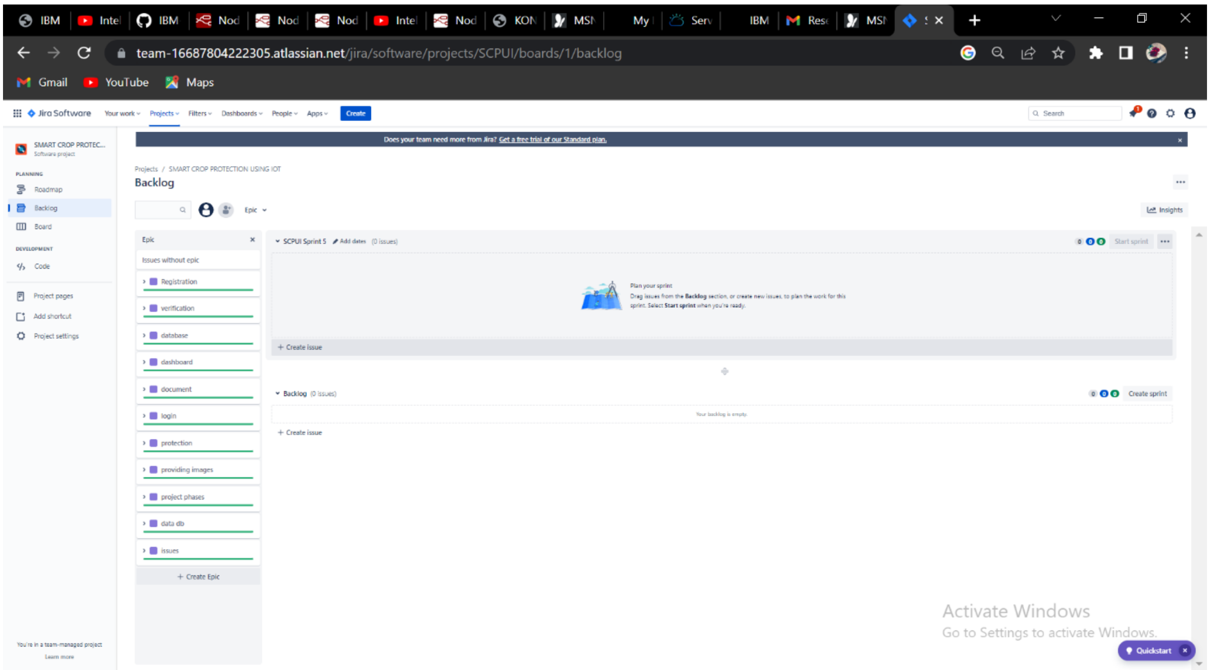
```
print('motoron')
elif(command =='motoroff') :
print('motoroff')
myConfig = {
"identity": {
"orgId": "blxckb",
"typeId": "NodeMCU",
"deviceId": "12345"
},
"auth": {
"token": "12345678"
}
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
database_name = "sample"
my_database = clientdb.create_database(database_name)
if my_database.exists():
print(f"¹ {database_name} ' successfully created.")
cap=cv2.VideoCapture('garden.mp4')
if(cap.isOpened()== True) :
print('File opened')
else:
print('File not found')
while(cap.isOpened()) :
ret, frame=cap.read()
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

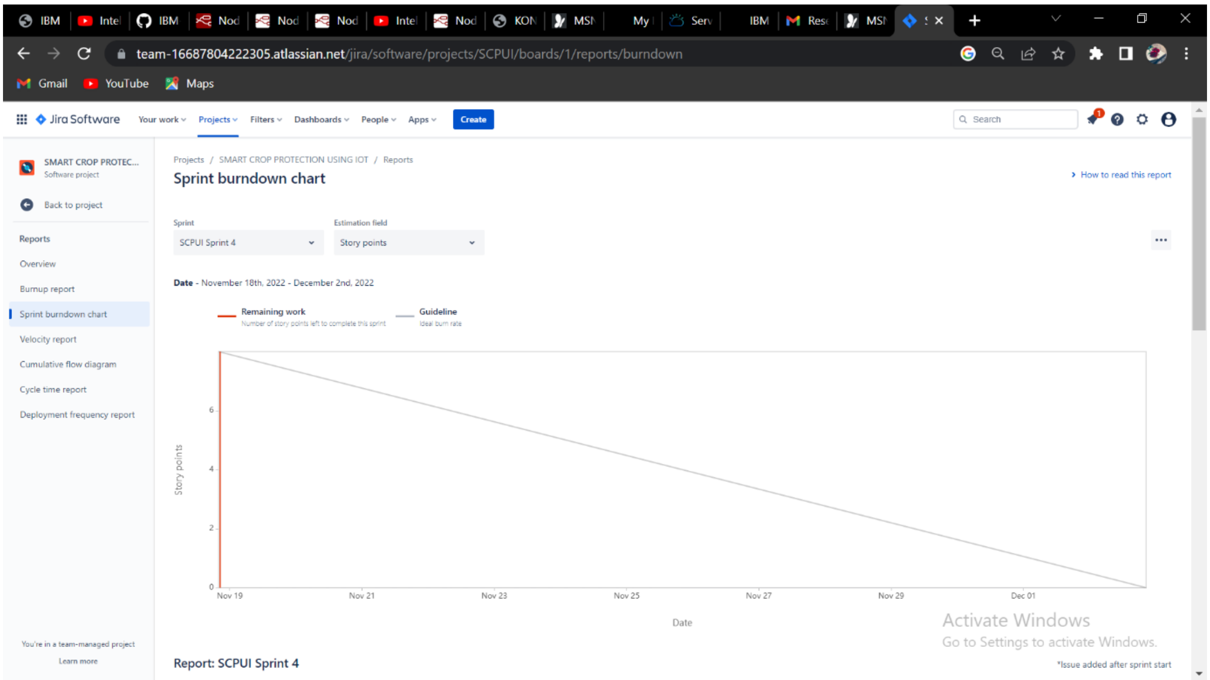
```
ims = cv2.resize(frame, (960, 540))
cv2.imwrite('ex.jpg' ,ims)
with open("ex.jpg", "rb") as f:
file_bytes = f.read()
# This is the model ID of a publicly available General model. You may use any other public or custom model ID.
request = service_pb2.PostModelOutputsRequest(
model_id='aaa03c23b3724a16a56b629203edc62c',
inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes))
)])
response = stub.PostModelOutputs(request, metadata=metadata)
if response.status.code != status_code_pb2.SUCCESS:
raise Exception("Request failed, status code: " + str(response.status.code))
detect=False
for concept in response.outputs[0] .data.concepts:
#print('%12s: %.2f¹ % (concept.name, concept.value))
if(concept.value>0.98):
#print(concept.name)
if(concept.name =="animal") :
print("Alert! Alert! animal detected")
playsound.playsound('alert.mp3')
# playsound.playsound('alert.mp3')
picname=datetime.datetime.now() . strftime("%Y-%m-%d-%H-%M")
cv2.imwrite(picname+ '.jpg',frame)
multi_part_upload('kiruthika2001' , picname+ '.jpg' , picname+ '.jpg')
json_document={"link":COS_ENDPOINT+'/'+'kiruthika2001'+'/'+picname+'.jpg'}
```

```
new_document = my_database.create_document(json_document)
if new_document.exists():
print(f"Document successfully created.")
time.sleep(5)
detect=True
moist=random.randint(0,100)
humidity=random.randint(0,100)
myData={ 'Animal' : detect, 'moisture' :moist, 'humidity':humidity}
print(myData)
if(humidity!=None):
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
onPublish=None)
print("Publish Ok ..")
client.commandCallback = myCommandCallback
cv2.imshow('frame ' , ims)
if cv2.waitKey(1) & 0xFF == ord('q'):
break
client.disconnect()
cap.release()
cv2.destroyAllWindows()
```

6.3 REPORT FROM JIRA







team-16687804222305.atlassian.net/jira/software/projects/SCPUI/boards/1/reports/burndown
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
SMART CROP PROTEC...
Software project
Back to project
Reports
Overview
Burnup report
Sprint burndown chart
Velocity report
Cumulative flow diagram
Cycle time report
Deployment frequency report
Projects / SMART CROP PROTECTION USING IOT / Reports
Sprint burndown chart
How to read this report
Sprint
SCPUI Sprint 4
Estimation field
Story points
Date - November 18th, 2022 - December 2nd, 2022
Remaining work
Guideline
Story points
Nov 19
Nov 21
Nov 23
Nov 25
Nov 27
Nov 29
Dec 01
Date
Report: SCPUI Sprint 4
*Issue added after sprint start

7.CODING AND SOLUTION

7.1 FEATURE 1

```
import cv2
import numpy as np
import wiotp.sdk.device
import playsound
import random
import time
import datetime
import ibm_boto3

from ibm_botocore.client import Config, ClientError
#CloudantDB
from cloudant.client import Cloudant
from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel
from clarifai_grpc.grpc.api import service_pb2_grpc
stub = service_pb2_grpc.V2Stub(ClarifaiChannel.get_grpc_channel())
from clarifai_grpc.grpc.api import service_pb2, resources_pb2
from clarifai_grpc.grpc.api.status import status_code_pb2
# This is how you authenticate.
metadata = (('authorization' , 'Key bc885e5165d74ef48f42f6f6a2c9eb87'),)
COS_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud" # Current list avaiable at https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints
```

```
COS_API_KEY_ID = " f6Ap-ct18m07S9UZL7XPbAF7170ome PLLUQOzqmnAzb5" # eg "W00YiRnLW4a3fTj MB-odB-2ySfTrFBIQQ'Wanc -- P3byk"

COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"

COS_RESOURCE_CRN = "crn:v1:bluemix:public:cloudantnosqldb:eu-gb:a/d43aa7d0631b400e9283084df08f9f60:502851d6-a240-4b22-8d4b-3642ed2bc3a8::" # eg "crn:vl:bluemix:public:cloud-object-storage:global:a/6b644a3fda97448b888c23eeef263ed6:199ab1e5-0d9d-420f-8e4a-98d868c04368 ::"

clientdb = Cloudant("apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2", "64455b04f35e5d5f9b4fc25bb38904af", url = "https://apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2:64455b04f35e5d5f9b4fc25bb38904af@de3c99da-899c-43cb-9aa5-b6b3fdc4cc16-bluemix.cloudantnosqldb.appdomain.cloud",

username = "apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2")

clientdb.connect()

#Create resource

cos=ibm_boto3.resource("s3",

ibm_api_key_id=COS_API_KEY_ID,

ibm_service_instance_id=COS_RESOURCE_CRN,

ibm_auth_endpoint=COS_AUTH_ENDPOINT,

config=Config(signature_version="oauth"),

endpoint_url=COS_ENDPOINT

)

def multi_part_upload(bucket_name, item_name, file_path) :

try:

print("Starting file transfer for {0} to bucket: {1}\n" .

format(item_name, bucket_name))

#set 5 MB chunks

part_size = 1024*1024 * 5
```

```
#set threadhold to 15 MB
file_threshold = 1024 * 1024 * 15
#set the transfer threshold and chunk size
transfer_config = ibm_boto3.s3.transfer.TransferConfig(
multipart_threshold=file_threshold,
multipart_chunksize=part_size
)
# the upload_fileobj method will automatically execute a multi-part upload
# in 5 MB chunks for all files over 15 MB
with open(file_path, "rb") as file_data:
cos.Object(bucket_name, item_name) .upload_fileobj(
Fileobj=file_data,
Config=transfer_config
)
print("Transfer for {0} Complete!\n". format(item_name))

except ClientError as be:
print("CLIENT ERROR: {0}\n" . format(be))
except Exception as e:
print("Unable to complete multi-part upload: {0}" .format(e))
def myCommandCallback(cmd) :
print("Command received: %s" % cmd.data)
command=cmd.data[ ' command']
print(command)
if(command =='lighton'):
print('lighton')
elif(command =='lightoff'):
```

```
print('lightoff')
elif(command =='motoron'):
```

7.2 FEATURE 2

```
print('motoron')
elif(command =='motoroff') :
print('motoroff')
myConfig = {
"identity": {
"orgId": "blxckb",
"typeId": "NodeMCU",
"deviceId": "12345"
},
"auth": {
"token": "12345678"
}
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
database_name = "sample"
my_database = clientdb.create_database(database_name)
if my_database.exists():
print(f"¹ {database_name} ' successfully created.")
cap=cv2.VideoCapture('garden.mp4')
if(cap.isOpened()== True) :
print('File opened')
```

```
else:

print('File not found')

while(cap.isOpened()) :

ret, frame=cap.read()

gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

ims = cv2.resize(frame, (960, 540))

cv2.imwrite('ex.jpg' ,ims)

with open("ex.jpg", "rb") as f:

file_bytes = f.read()

# This is the model ID of a publicly available General model. You may use any other public or custom model ID.

request = service_pb2.PostModelOutputsRequest(

model_id='aaa03c23b3724a16a56b629203edc62c',

inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.Image(base64=file_bytes))

)])

response = stub.PostModelOutputs(request, metadata=metadata)

if response.status.code != status_code_pb2.SUCCESS:

raise Exception("Request failed, status code: " + str(response.status.code))

detect=False

for concept in response.outputs[0] .data.concepts:

#print('%12s: %.2f¹ % (concept.name, concept.value))

if(concept.value>0.98):

#print(concept.name)

if(concept.name =="animal") :

print("Alert! Alert! animal detected")

playsound.playsound('alert.mp3')

# playsound.playsound('alert.mp3')
```

```
picname=datetime.datetime.now() . strftime("%Y-%m-%d-%H-%M")
cv2.imwrite(picname+ '.jpg',frame)
multi_part_upload('kiruthika2001' , picname+ '.jpg' , picname+ '.jpg')
json_document={"link":COS_ENDPOINT+'/'+'kiruthika2001'+'/'+picname+'.jp
g'}
new_document = my_database.create_document(json_document)
if new_document.exists():
print(f"Document successfully created.")
time.sleep(5)
detect=True
moist=random.randint(0,100)
humidity=random.randint(0,100)
myData={ 'Animal' : detect, 'moisture' :moist, 'humidity':humidity}
print(myData)
if(humidity!=None):
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
onPublish=None)
print("Publish Ok ..")
client.commandCallback = myCommandCallback
cv2.imshow('frame ' , ims)
if cv2.waitKey(1) & 0xFF == ord('q'):
break
client.disconnect()
cap.release()
cv2.destroyAllWindows()
```

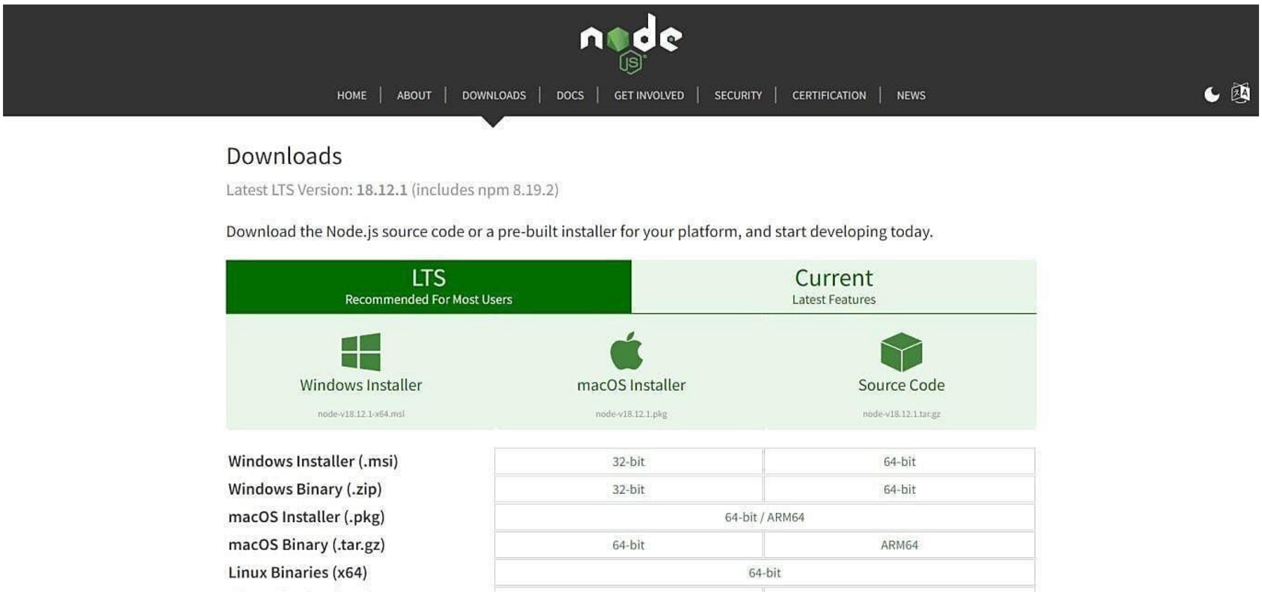
8.TESTING

8.1 TEST CASES

TEAM ID : PNT2022TMID13381
JECT : IOT BASED SMART CROP PROTECTION SYSYTEM FOR AGRICULT
DATE : 19 NOVEMBER 2022

TESTCASE ID	TESTCASE	TEST SCENARIO	TEST STEPS	INPUTS	EXPECTED OUTPUT	ACTUAL OUTPUT	TEST RESULT	TEST COMMENTS	BUG ID	TESTED BY
1	M WATSON IOT PLATFOR	To check whether the ibm watson is get connected	login to ibm watson iot platform	id , password	it should get login to the watson page	it has been logged in to the login page	PASS	GOOD		Aarthy M
			check whether it has the separate organization id	new id	it should shows the organization id	separate organization id has been shown	PASS	GOOD		Aarthy M
			check whether team mates are get connected	team mates id	it should shows the all the team members name / id	it is showing all the team members	PASS	GOOD		Aarthy M
			check whether separate device name , id , authentication token generated	device name , type	new device should be created	new device has been created	PASS	GOOD		Aarthy M
			to check whether it is showing output	device code and inputs	it should shows device gets connected and should show the output	its showing that device gets connected and output are verified	PASS	GOOD		Aarthy M
2	Python Compiler and Wokwi	to check the connection is established in	To check the whether the pH value is shown are not	pH reading	it need to show the ph value sometimes may random	it show the ph value for the input	PASS	GOOD		Hemanithi J
			to check whether the Temperature and humidity are sh	Temperature & humidity	it should show temperature & humidity	it show the temperature & humidity value for input	PASS	GOOD		Hemanithi J
3	NODE-RED	to check whether node-red is connected and shows the output	login in to node-red	id , password	it should get login to the node-red page	its get entered into the login page	PASS	GOOD		Hemanithi J
			check whether all the necessities are imported and connected	nodes	it should not show any error on nodes	it is not showing any errors	PASS	GOOD		Hemanithi J
			check whether all the nodes ar connected	node connection	blocks should gets	blocks has been connected	PASS	GOOD		Hemanithi J
			check whether the output are shown in nodered	output found or not	output should be obtained	output has been obtained	PASS	GOOD		Hemanithi J
4	MIT App Invertor	check whether the outputs are shown in	check whether the login is created	id,password	Get into the MIT app	MIT App invertor is getted	PASS	GOOD		Kiruthika B
			check whether new project is created in MIT	Project created	the new project is created	the new project is created	PASS	GOOD		Kiruthika B
			check whether the designer page is ready to use	create app	it should created	it is created successfully	PASS	GOOD		Kiruthika B
			check whether it the block page is created	create block	block should created	it is created successfully	PASS	GOOD		Kiruthika B
			check whether the block run successfully without error	run block	it should get input from cloud	it has been connected and provide output	PASS	GOOD		Kiruthika B
5	QR CODE	check whether qr code is generated	check whether the code shows any error	code	it should not shows any error	it is not showing any errors	PASS	GOOD		Kiruthika B
			check whether the MIT provide QR code	QR Code	QR code has been generated	QR code is generated	PASS	GOOD		Kiruthika B
			check whether the MIT app is installed in mobile	install in mobile	user should install mobile	app is install successfully	PASS	GOOD		Kiruthika B
			check whether the QR code get connected	app link	mobile gets connected	mobile has been connected	PASS	GOOD		Kiruthika B
			check whether the screen is found in mobile	screen found	screen should be generated	screen has been generated	PASS	GOOD		Kowsika P
6	TESTING	check entire process	check watson is connected	watson	iot watson should produce its output	iot watson has producing its output	PASS	GOOD		Kowsika P
			check node-red is connected	node-red	node-red should produce its output	node-red has been producing its output	PASS	GOOD		Kowsika P
			check whether python is connected	python	python should gets	python has been connected	PASS	GOOD		Kowsika P
			check whether details are shown	MIT App	details in MIT should be	details in MIT should be	PASS	GOOD		Kowsika P

8.2 USER ACCEPTANCE TESTING



9.RESULT

9.1 PERFORMANCE METRICS

The problem of crop vandalization by wild animals and fire has become a major social problem in current time.

It requires urgent attention as no effective solution exists till date for this problem. Thus this project carries a great social relevance as it aims to address this problem. This project willhelp farmers in protecting their orchards and fields and save them from significant financial losses and will save them from the unproductive efforts that they endure for the protection their fields. This will also help them in achievingbetter crop yields thus leading to theireconomic wellbeing.

1O.ADVANTAGES AND DISAVANTAGES

ADVANTAGES

Controllable food supply. you might have droughts or floods, but if you are growing the crops and breeding them to be hardier, you have a better chanceof not straving. It allows farmers to maximize yields using minimum resources such as water,fertilizers.

DISADVANTAGES

The main disadvantage is the time it can take to process the information.in order to keep feeding people as the population grows you have to radically change theenvironment of the planet.

11.CONCLUSION

A IoT Web Application is built for smart agricultural system using Watson IoT platform, Watsonsimulator, IBM cloud and Node-RED.

12.FUTURE SCOPE

In the future, there will be very large scope, this project can be made based on Image processing in which wild animaland fire can be detected by cameras and if it comes towards farmthen system will be directly activated through wireless networks. Wild animals can also be detected by using wireless networks such as laser wireless sensors and by sensing this laser or sensor's security system will beactivated.

13.APPENDX

SOURCE CODE

```
import cv2
import numpy as np
import wiotp.sdk.device
import playsound
import random
import time
import datetime
import ibm_boto3

from ibm_botocore.client import Config, ClientError
#CloudantDB
from cloudant.client import Cloudant
from cloudant.error import CloudantException
from cloudant.result import Result, ResultByKey
from clarifai_grpc.channel.clarifai_channel import ClarifaiChannel
from clarifai_grpc.grpc.api import service_pb2_grpc
stub = service_pb2_grpc.V2Stub(ClarifaiChannel.get_grpc_channel())
from clarifai_grpc.grpc.api import service_pb2, resources_pb2
from clarifai_grpc.grpc.api.status import status_code_pb2
# This is how you authenticate.
metadata = (('authorization' , 'Key bc885e5165d74ef48f42f6f6a2c9eb87'),)
COS_ENDPOINT = "https://s3.jp-tok.cloud-object-storage.appdomain.cloud" # Current list avaiable at https://control.cloud-object-storage.cloud.ibm.com/v2/endpoints
```

```
COS_API_KEY_ID = " f6Ap-ct18m07S9UZL7XPbAF7170ome
PLLUQOzqmnAzb5" # eg "W00YiRnLW4a3fTj MB-odB-2ySfTrFBIQQ'Wanc
-- P3byk"

COS_AUTH_ENDPOINT = "https://iam.cloud.ibm.com/identity/token"

COS_RESOURCE_CRN = "crn:v1:bluemix:public:cloudantnosqldb:eu-
gb:a/d43aa7d0631b400e9283084df08f9f60:502851d6-a240-4b22-8d4b-
3642ed2bc3a8::" # eg "crn:vl:bluemix:public:cloud-object-
storage:global:a/6b644a3fda97448b888c23eeef263ed6:199ab1e5-0d9d-420f-
8e4a-98d868c04368 ::"

clientdb = Cloudant("apikey-v2-
1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2",
"64455b04f35e5d5f9b4fc25bb38904af", url = "https://apikey-v2-
1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2:64455b04f35e5d5f9b4fc2
5bb38904af@de3c99da-899c-43cb-9aa5-b6b3fdc4cc16-
bluemix.cloudantnosqldb.appdomain.cloud",

username = "apikey-v2-1wveoo6739lo7qj5cy7kqtpfsku8dumxlvp6dy62rwu2")

clientdb.connect()

#Create resource

cos=ibm_boto3.resource("s3",

ibm_api_key_id=COS_API_KEY_ID,

ibm_service_instance_id=COS_RESOURCE_CRN,

ibm_auth_endpoint=COS_AUTH_ENDPOINT,

config=Config(signature_version="oauth"),

endpoint_url=COS_ENDPOINT

)

def multi_part_upload(bucket_name, item_name, file_path) :

try:

print("Starting file transfer for {0} to bucket: {1}\n" .

format(item_name, bucket_name))

#set 5 MB chunks

part_size = 1024*1024 * 5
```

```
#set threadhold to 15 MB
file_threshold = 1024 * 1024 * 15
#set the transfer threshold and chunk size
transfer_config = ibm_boto3.s3.transfer.TransferConfig(
multipart_threshold=file_threshold,
multipart_chunksize=part_size
)
# the upload_fileobj method will automatically execute a multi-part upload
# in 5 MB chunks for all files over 15 MB
with open(file_path, "rb") as file_data:
cos.Object(bucket_name, item_name) .upload_fileobj(
Fileobj=file_data,
Config=transfer_config
)
print("Transfer for {0} Complete!\n". format(item_name))

except ClientError as be:
print("CLIENT ERROR: {0}\n" . format(be))
except Exception as e:
print("Unable to complete multi-part upload: {0}" .format(e))
def myCommandCallback(cmd) :
print("Command received: %s" % cmd.data)
command=cmd.data[ ' command']
print(command)
if(command =='lighton'):
print('lighton')
elif(command =='lightoff'):
```

```
print('lightoff')
elif(command =='motoron'):
print('motoron')
elif(command =='motoroff') :
print('motoroff')
myConfig = {
"identity": {
"orgId": "blxckb",
"typeId": "NodeMCU",
"deviceId": "12345"
},
"auth": {
"token": "12345678"
}
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()
database_name = "sample"
my_database = clientdb.create_database(database_name)
if my_database.exists():
print(f"¹ {database_name} ' successfully created.")
cap=cv2.VideoCapture('garden.mp4')
if(cap.isOpened()== True) :
print('File opened')
else:
print('File not found')
while(cap.isOpened()) :
```

```
ret, frame=cap.read()

gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

ims = cv2.resize(frame, (960, 540))

cv2.imwrite('ex.jpg' ,ims)

with open("ex.jpg", "rb") as f:

file_bytes = f.read()

# This is the model ID of a publicly available General model. You may use any
other public or custom model ID.

request = service_pb2.PostModelOutputsRequest(

model_id='aaa03c23b3724a16a56b629203edc62c',

inputs=[resources_pb2.Input(data=resources_pb2.Data(image=resources_pb2.I
mage(base64=file_bytes))

)])

response = stub.PostModelOutputs(request, metadata=metadata)

if response.status.code != status_code_pb2.SUCCESS:

raise Exception("Request failed, status code: " + str(response.status.code))

detect=False

for concept in response.outputs[0] .data.concepts:

#print('%12s: %.2f¹ % (concept.name, concept.value))

if(concept.value>0.98):

#print(concept.name)

if(concept.name =="animal") :

print("Alert! Alert! animal detected")

playsound.playsound('alert.mp3')

# playsound.playsound('alert.mp3')

picname=datetime.datetime.now() . strftime("%Y-%m-%d-%H-%M")

cv2.imwrite(picname+ '.jpg',frame)

multi_part_upload('kiruthika2001' , picname+ '.jpg' , picname+ '.jpg')
```

```
json_document={"link":COS_ENDPOINT+'/'+'kiruthika2001'+'/'+picname+'.jpg'}
new_document = my_database.create_document(json_document)
if new_document.exists():
print(f"Document successfully created.")
time.sleep(5)
detect=True
moist=random.randint(0,100)
humidity=random.randint(0,100)
myData={ 'Animal' : detect, 'moisture' :moist, 'humidity':humidity}
print(myData)
if(humidity!=None):
client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
print("Publish Ok ..")
client.commandCallback = myCommandCallback
cv2.imshow('frame ' , ims)
if cv2.waitKey(1) & 0xFF == ord('q'):
break
client.disconnect()
cap.release()
cv2.destroyAllWindows()
```

GitHub Link

https://github.com/IBM-EPBL/IBM-Project-1595-1658401588

Project Demo Link

https://youtu.be/QC2PvwGjZMU