Software Requirements Specifications

IOT BASED SMART POULTRY FARM

University of Sargodha

Project Code:

Internal Advisor:

Mam Samreen

Project Manager:

Anees ur Rehman

Project Team:

Naila Batool(BSEF19E041)
Anees ur Rehman(BSEF19E028)
Mustaqeem Haider(BSEF19E035)

Submission Date:

12-2-2022

____

Project Manager's Signature

Document Information

Category	Information
Customer	CS & IT, UOS
Project	IOT BASED SMART POULTRY FARM
Document	Requirement Specifications
Document Version	1.0
Identifier	PGBH01-2020-RS
Status	Draft
Author(s)	Naila Batool Mustaqeem Haider Anees ur Rehman
Approver(s)	PM
Issue Date	October 10, 2022
Document Location	
Distribution	1. Advisor 2. PM 3. Project Office

Definition of Terms, Acronyms and Abbreviations

This section should provide the definitions of all terms, acronyms, and abbreviations required to interpret the terms used in the document properly.

Term	Description
ASP	Active Server Pages
RS	Requirements Specifications

Table of Contents

1.Introduction

1.1 Purpose of Document

This document provides comprehensive details of project. It describes the scope, cost, problem, solution of the project. This document describes how user monitor and also control the climate of poultry farm.

1.2 Project Overview

Recently, the use of IoT (Internet of Things) based system has been expanded. The system describes the automation of poultry farm that facilitate control and supervision regardless of distance and time. In a poultry farm both temperature and humidity levels should be monitored regularly in ensuring the system runs smoothly. With the help of IOT based technology, we can easily control, monitor and manage the current situation of the farm through mobile application at anytime and anywhere.

1.3 Scope

As poultry farming is very popular in so many countries all over the world. Manual checking and maintaining of poultry farm require a lot of hard work. Therefore, we are going to develop an IoT based Smart Poultry farm that will allow users to automatically control and monitor the poultry farm. It will lessen a lot of work and save a lot of time of the users. We can easily market the project as it will make a lot of easiness for poultry farm owners.

2. Overall System Description

In the proposed system, almost all factors including such as temperature, humidity, light and manual works like food feeding and water supply system is fully automated system is designed to perform these activities. This system reduces manpower, improves health and growth of chicken and increases eggs production. For this purpose, mobile application can be used for monitoring farm activities and internal environment.

2.1 User characteristics

IOT based smart poultry farm has two main users who will be using the system. First One is the admin who will be going to install the IoT hardware and using the Arduino UNO Software will be uploading the data on the Database and other one will be the user who is going to utilize that data for controlling and monitoring the smart poultry farm. For this purpose, mobile application can be used by user for monitoring farm activities and internal environment.
These two users perform following activities

- Sensing/Monitoring Unit
- Control Unit
- Feedback Unit
- Updating Unit

User Roles:

View the Data Sensed by the System
Control the overall system
Maintain temperature, food level, water and all other processes

Admin Roles:

Manage the IoT hardware devices and Sensors
Using the data sensed and uploading that to Database
Update the system

2.2 System constraints

Identify any constraints or limitations on the system. Constraints may include the following:

- Investment- Getting your poultry farm started.
- Select farm location.
- Gather necessary equipment.
- Purchase chickens

Software constraints

- Android Version 4.1 min or API level 16
- Arduino UNO Desktop Application

Hardware constraints

- Computer or Laptop with Windows OS
- Temperature sensor
- Arduino Mega
- Ultrasonic sensors
- Arduino UNO
- Water level sensors
- Gas sensors
- Light sensor

Cultural constraints (includes language etc.)

- There must be an internet connection so the user can use this and application literate(English Language)

User constraint

- As this project is for the people who not manage their time to go physically and manage poultry farm system.

3. Functional Requirements

Functional Requirements contains the actions, services, tasks or functions that are going to be performed by the system.

Sensors

All sensor will be monitoring the activities performing in farm and send it to Arduino.

Arduino UNO

Arduino is u sed to control the data sensed by the sensors and uploading that data on the database. This data is used by application to control and monitor poultry farm activities.

Android App

This app will be showing the data sensed by the sensors (temperature sensor, water level sensor, gas sensor) by which user will be see the data on application and perform activities for managing it.

Monitor activities

App will be showing the activities in farm by using the sensors which are installed.

Controlling activities

App will be to take corrective actions and steps to manage their food level, water level and gas level.

4. Non-functional Requirements

1.1 Performance Requirements

Speed:

All the data will be shown to the users in real-time as it is a time sensitive data so that user can take corrective actions to control the pollution level.

Precision:

App will be showing the data of smart poultry farm so it is a very sensitive data that requires a great precision.

Reliability:

The data shown by the app must be reliable as it is regarding the health issues of the users.

1.2 Safety Requirements

Data shown must be precise and the corrective measures suggested by the application to manage the activities.

1.3 Security Requirements

Smart poultry farm data must be accurately shown to the users and all the users device data must the secure.

4.1 User Documentation

List the user documentation components that will be delivered along with the software, such as user manuals, online help, and tutorials.

5. Assumptions and Dependencies

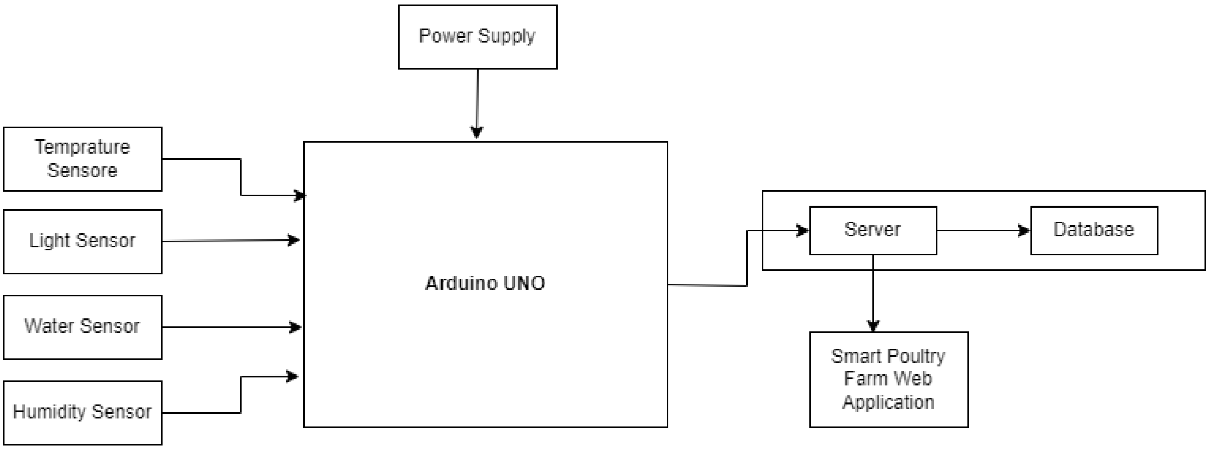
List any assumed factors that could affect the stated requirements. These factors are not system constraints, but areas where future changes might drive changes in the requirements. The project could be affected if these assumptions are incorrect, are not shared, or changed.

Also, identify any dependencies the project has on external factors. For example, if you expect to integrate into the system some components that are being developed by another project, you are dependent upon that project to supply the correctly operating components on schedule.

6. System Architecture

Following architecture level components of the system.

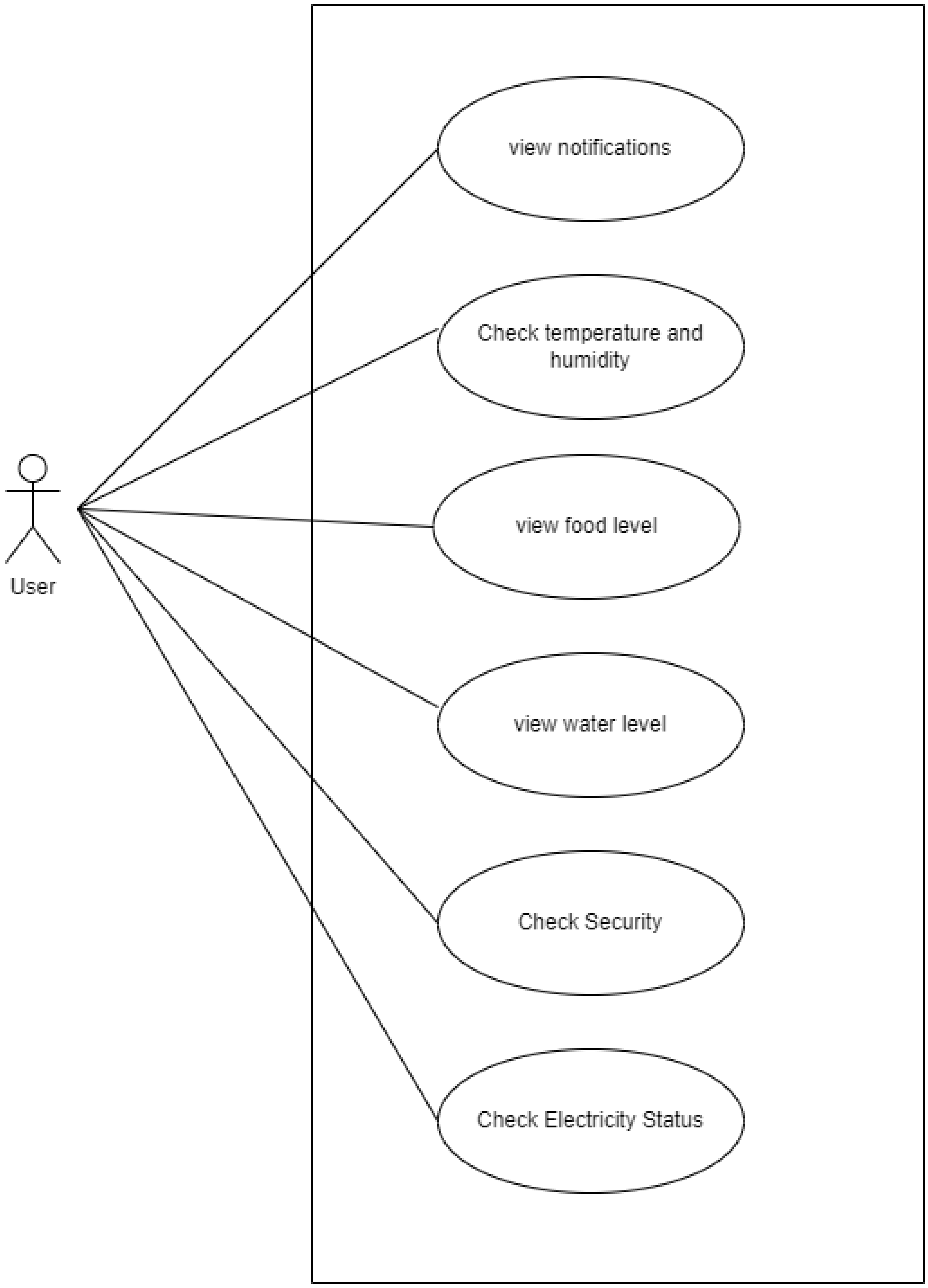
- Temperature Sensor
- Water Level Sensor
- Gas Sensor
- Ultrasonic Sensor



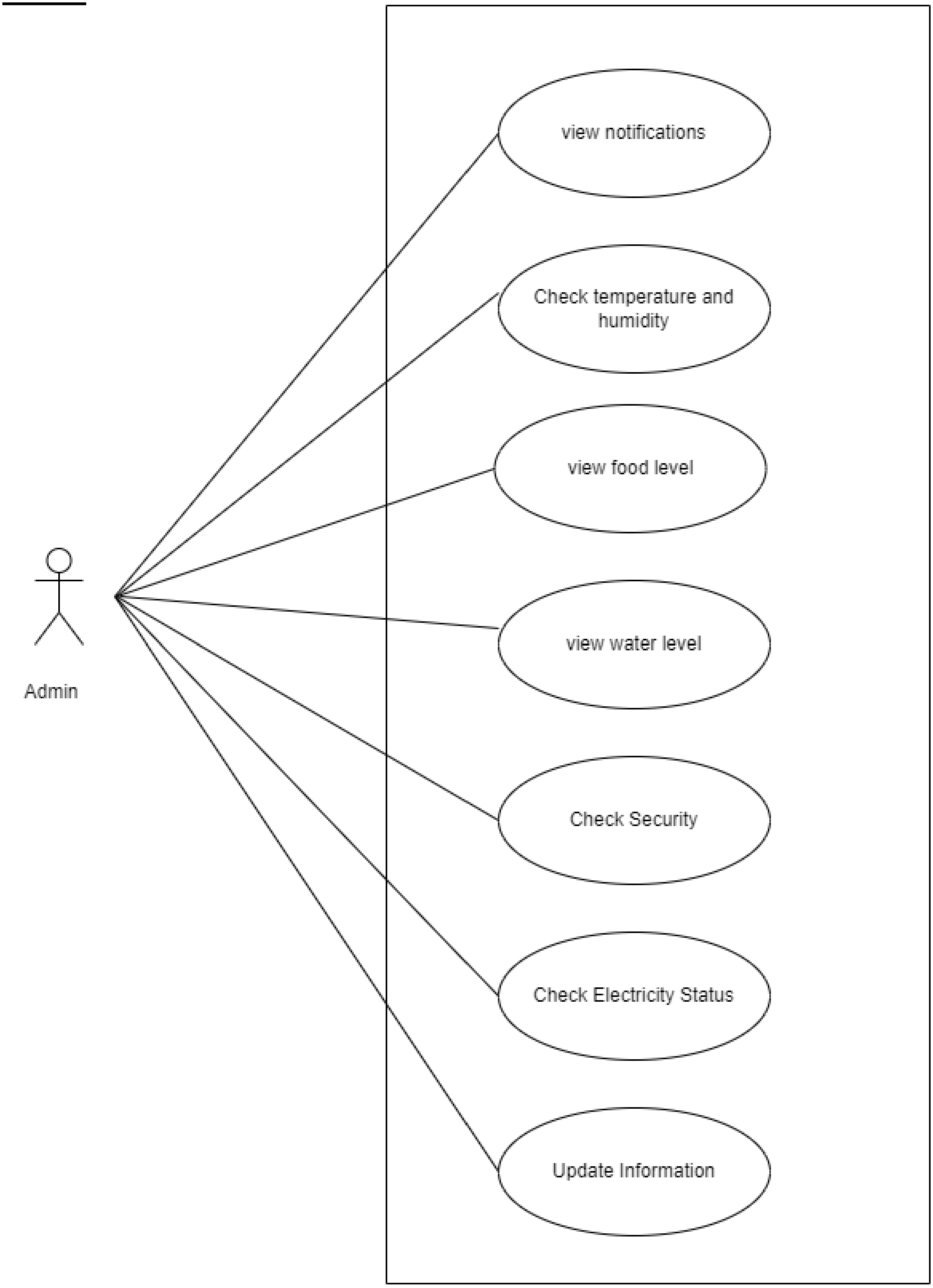
- WIFI kit

6.1 Use Case Diagrams

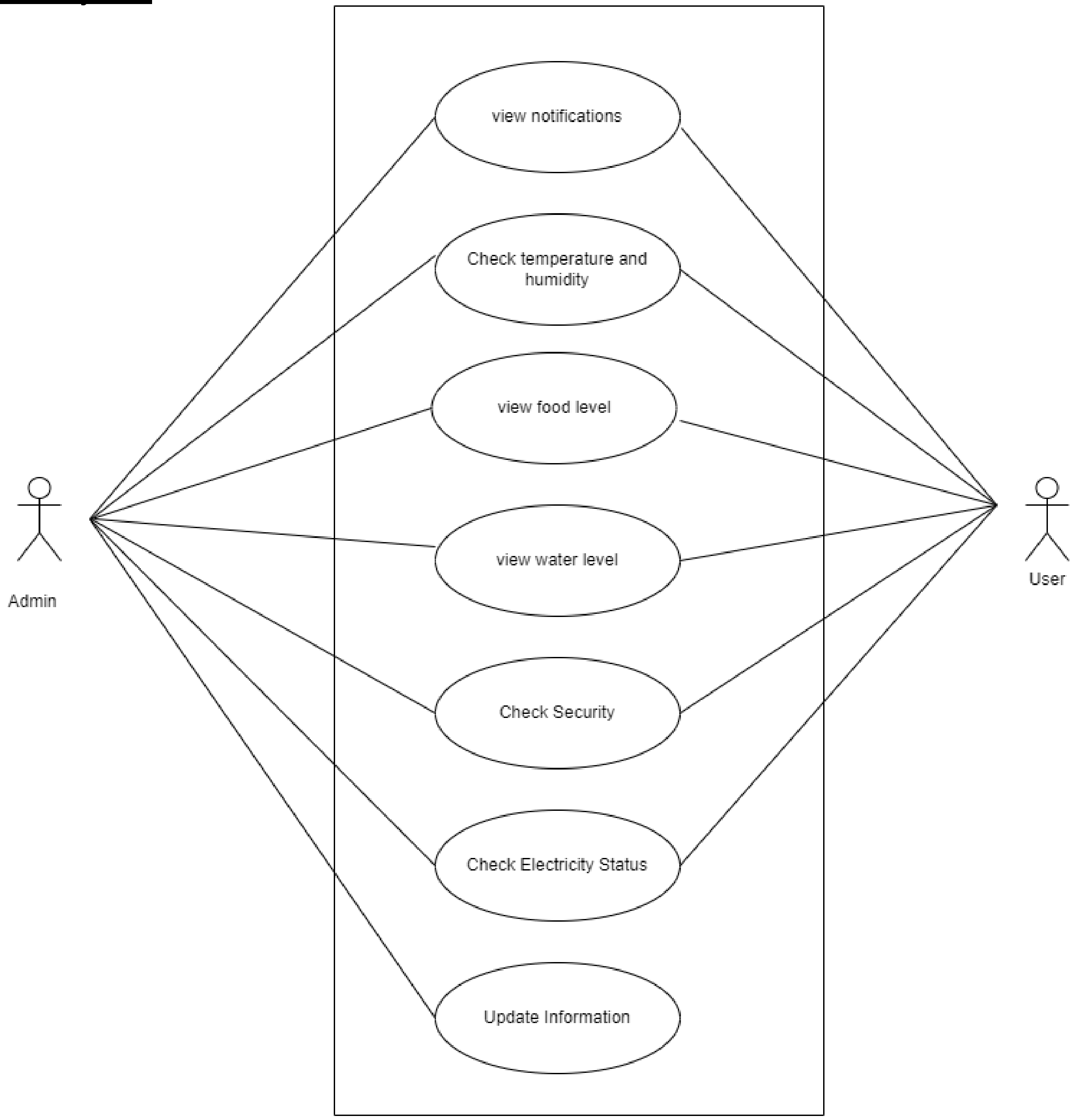
USER



ADMIN


view notifications
Check temperature and humidity
view food level
view water level
Check Security
Check Electricity Status
Update Information
Admin

Combine System



Use Case Description

Each Use Case has a description, which describes the functionality that will be built in the proposed system. The template for Use Case description is given below:

<Use case Id: user>
Actors: User
Feature: check temperature, water level, gas level

Use case Id:	1
Pre-condition:	Install App Internet Connection Registration Login

Scenarios

Step#	**Action**	**Software Reaction**
1.	App Login	Authenticate user
2.	Check Gas	Show if any toxic gas is present or not
3.	Water Level	Show the level of water
4.	Check Temperature	Show Temperature of Poultry farm

Post Conditions

Step#	**Description**
	After performing activity, water level, and temperature is maintained and controlled. Suggestions are updated to the admin.

Use Case Cross referenced	2
User Interface reference	List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.

Concurrency and Response
Give an estimate of the following

- ♦ Multiple users
- ♦ Fast

<Use case Id: admin>
Actors: admin
Feature: check temperature, water level, gas level, update suggestions

Use case Id:	2
Pre-condition:	Install App Internet Connection Registration Login

Scenarios

Step#	**Action**	**Software Reaction**
1.	App Login	Authenticate user
2.	Check Gas	Show if any toxic gas is present or not

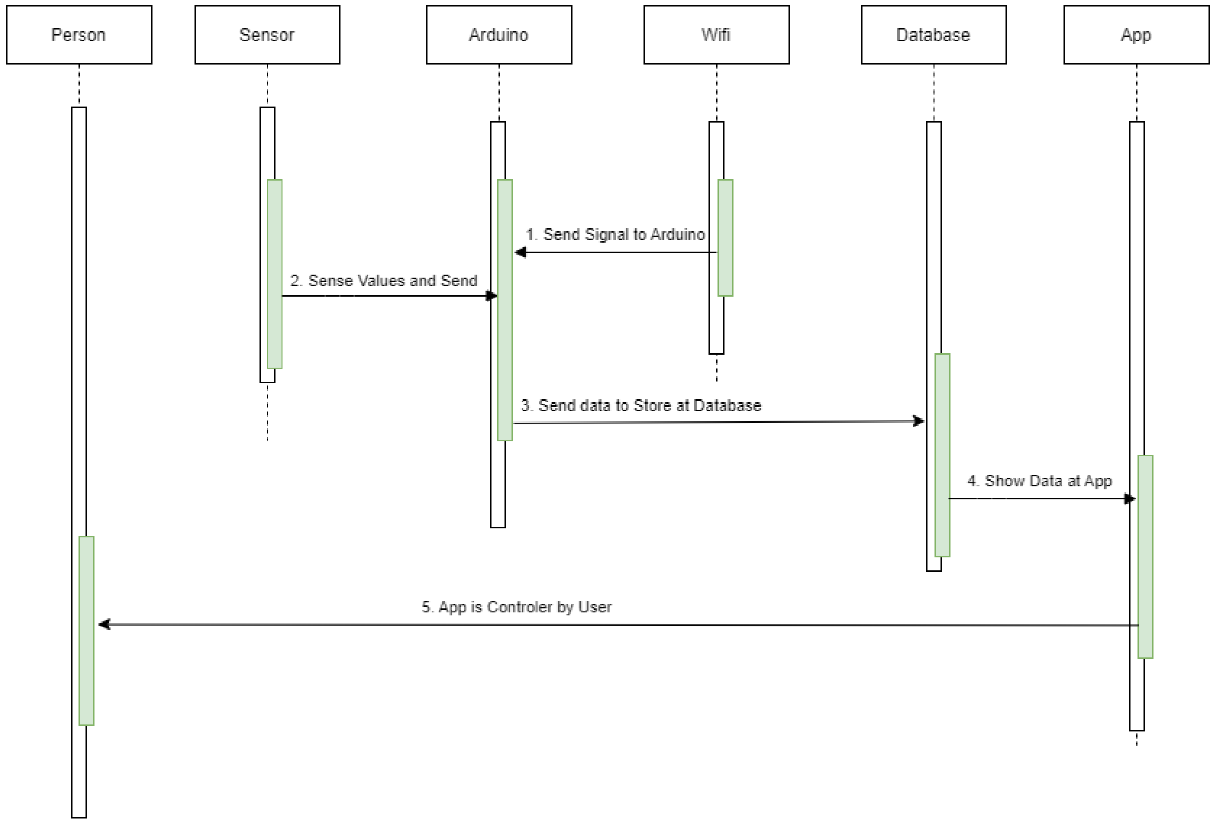
3.	Water Level	Show the level of water
4.	Check Temperature	Show Temperature of Poultry farm
5.	Update Suggestions	Suggestions are show to users
Post Conditions		
Step#	**Description**	
	After performing activity, water level, and temperature is maintained and controlled. Suggestions are updated to the user.	
Use Case Cross referenced	1	
User Interface reference	List user interface(s) that are related to this use case. Use numbered list in case of more than one user interface elements.	
Concurrency and Response Give an estimate of the following ♦ 1 ♦ Fast		

8. Graphical User Interfaces

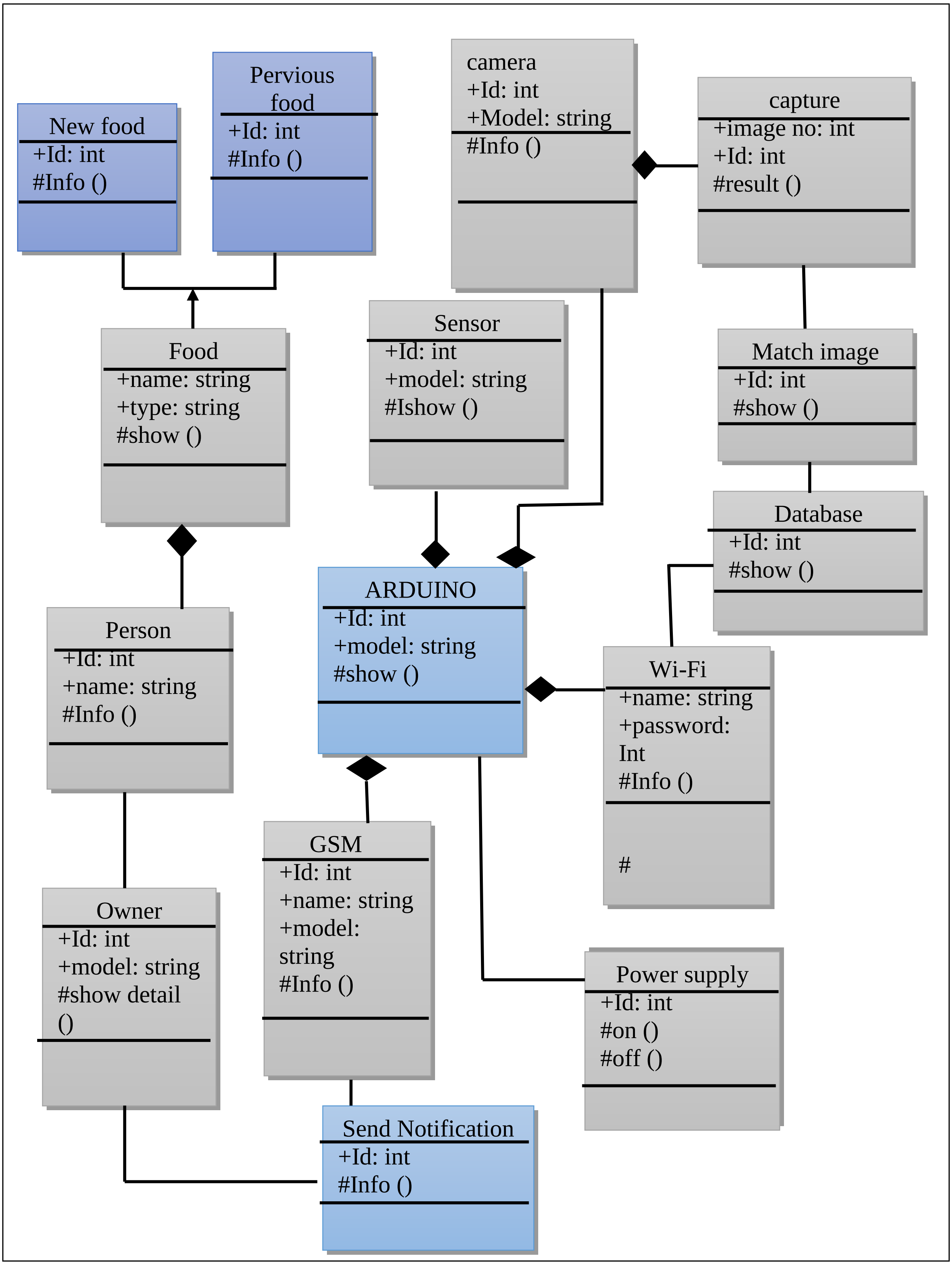
Give a detailed account of user interfaces included in this project.

<User Interface Id: Title>	
Interface Id.	Write the reference number assigned to this UI.
Use case Reference	Refer to the use case invoking this UI.
Snapshot	
Include a labeled snapshot of the user interface.	
Data dictionary reference	
Label	**Data dictionary identifier**
	Refer to fields in data dictionary

9. Sequence Diagram

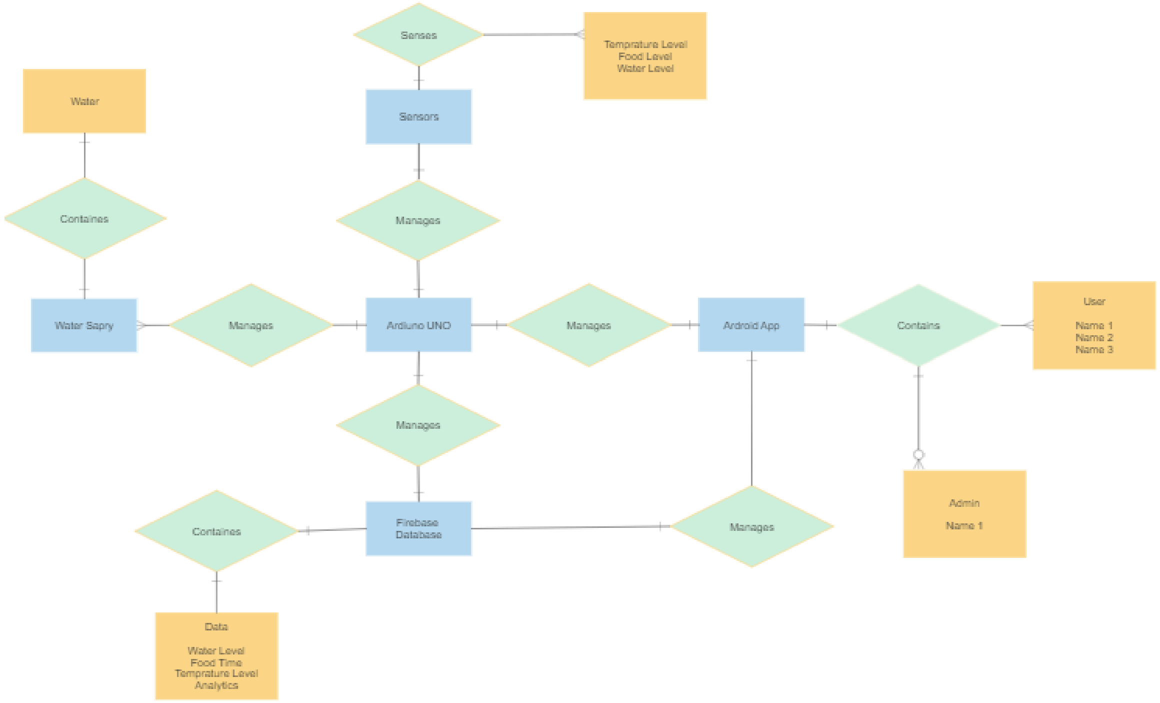

Person
Sensor
Arduino
Wifi
Database
App
1. Send Signal to Arduino
2. Sense Values and Send
3. Send data to Store at Database
4. Show Data at App
5. App is Controler by User

10.Class Diagram

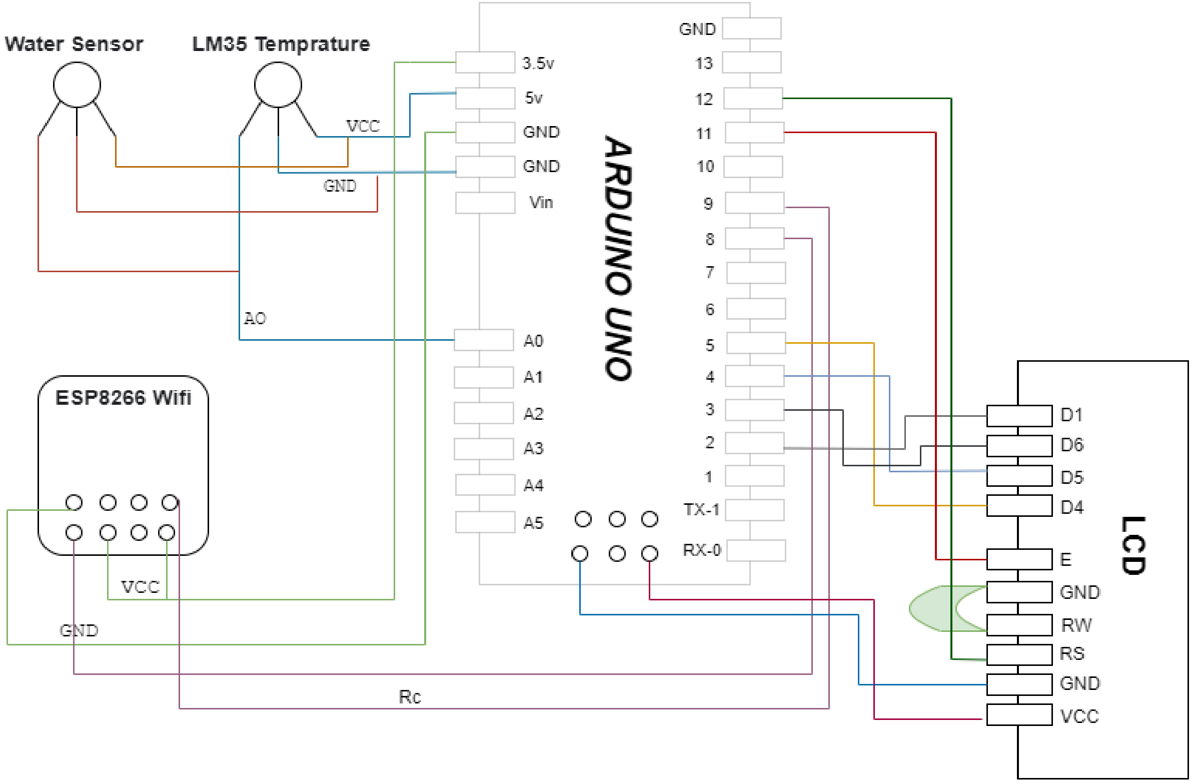


11. High Level Design

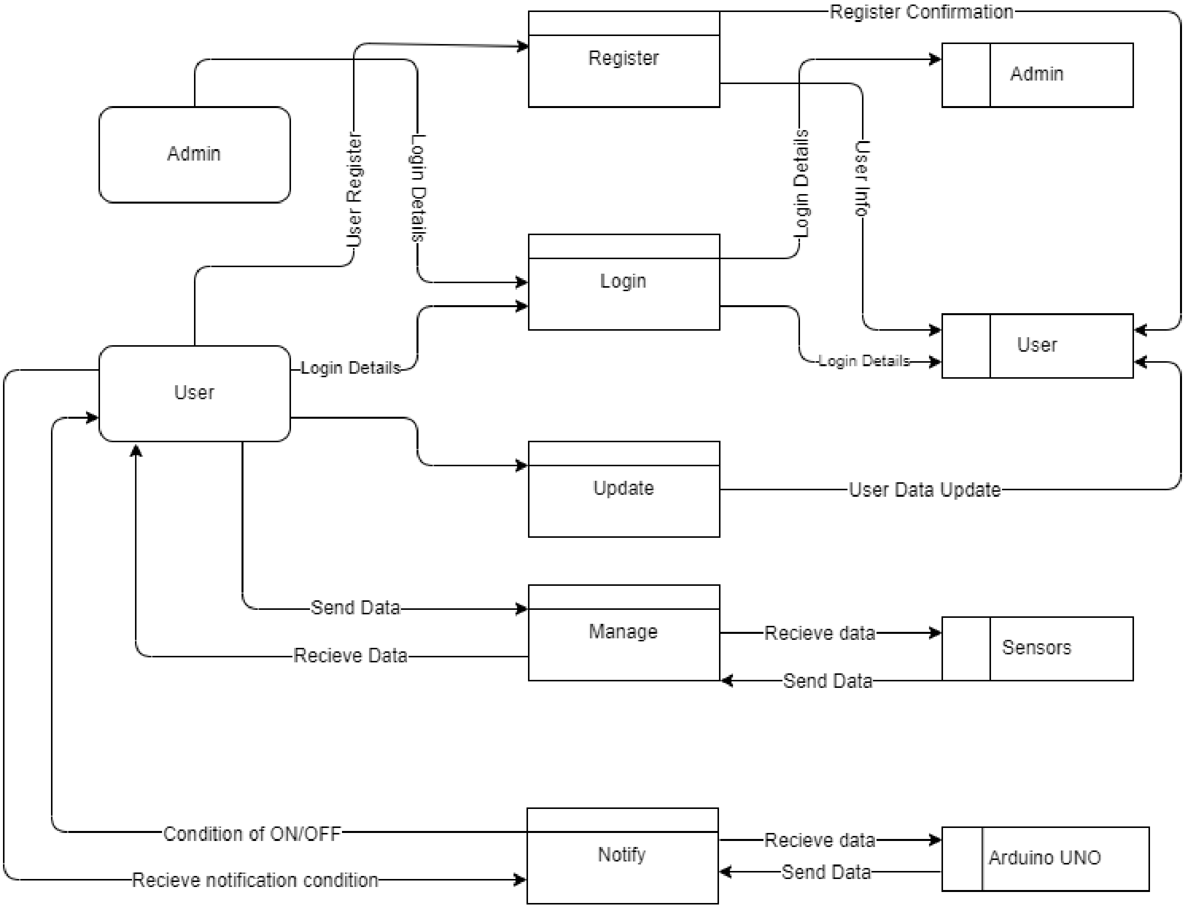
11.1 ER Diagram



11.2 Circuit Diagram



11.3 DFD


Register Confirmation
Register
Admin
Admin
User Register
Login Details
Login Details
User Info
Login
Login Details
Login Details
User
User
Update
User Data Update
Send Data
Manage
Recieve Data
Recieve data
Send Data
Sensors
Condition of ON/OFF
Recieve notification condition
Notify
Recieve data
Send Data
Arduino UNO

11.4 Data Dictionary

The convention recommended for writing the data dictionary is as follows.

i. **Data 1**

Temperature.

ii. **Data 2**

Water Level

iii. **Data 2**

Gas Level

< Data 1>	
Name	Temperature
Alias	Humidity
Where-used/how-used	Temperature is used in app to control the temperature of the poultry farm.
CONTENT DESCRIPTION	Temperature is sensed by sensor and it is used by admin and users to control the temperature of the poultry farm.

< Data 2>	
Name	Water Level
Alias	Level of water
Where-used/how-used	It is used in app by the admin and user to maintain the water level in poultry farm.
CONTENT DESCRIPTION	Water level is sensed in sensor then sent to Arduino UNO. There it is sent to database so that user can check and control the level of the water by the app.

< Data 3>	
Name	Gas Level
Alias	Level of gas
Where-used/how-used	They are commonly used to detect toxic or explosive gasses and measure gas concentration.
CONTENT DESCRIPTION	Gas sensors (also known as gas detectors) are electronic devices that detect and identify different types of gasses.

12.Requirements Traceability Matrix

The requirements trace-ability matrix is a table used to trace project life cycle activities and work products to the project requirements. The matrix establishes a thread that traces requirements from identification through implementation.

Sr. #	Feature	Use case ID	UI ID	Priority	Build Number	Use Case Cross reference (Related Use Cases)
1	View temperature View water level Check security	1		1		2
2	View temperature View water level Update system	2		1		1

The columns carry the following meaning:

- Feature: Lists system features based on which use cases are built.
- Use Case ID: Write the ID of the use case for easy lookup
- UI ID: Write the user interface ID for this use case.
- Priority: Give an appropriate rating to each use case according to its priority
- Build Number: Write the reference number to which this feature belongs.
- Use Case Cross Ref: Write the related use cases separated with commas.

13.Risk Analysis

Perform an analysis of the constraints and identify the potential problems that may arise in the project due to the constraints. For this section cover the following:

- Storage Risk:

As all the data sensed by the sensors would have to be stored permanently at a storage device that can build a massive amount of data.

- Natural Disaster:

As it is a hardware-based project sensor can damage through natural disaster like rain ,flood etc.

- Hardware Failure:

There could be a risk that hardware can be damaged by multiple reasons. Or can start measuring wrong data due to mud contamination or some other issues.

14.Cost Estimation

1.	**Software development cost**	8,000
2.	**Packaged software**	20,000
3.	**Hardware**	
4.	**Network**	5,000
5.	**Client**	
6.	**Misc.**	5,000
		Total cost =38,000

15.References

- **[1]** Archana M P1, Uma S K2, “Monitoring and controlling of poultry farm using IOT”, International Journal of Innovative Research in Computer and Communication Engineering, Vol. 6, Issue 4, April 2018.
- **[2]** Zainal H. C. Soh1, Mohd H. Ismail1, “Development of automatic chicken feeder using Arduino Uno”, IEEE, Dec 2017
- **[3]** R. Brian “Farms of the Future: The Rise of IoT in Agriculture”. Retrieved from https://www.link-labs.com/blog/rise-of-iot-in-agriculture, 2016
-
- **[4]** Lopez Research LLC, “An Introduction to the Internet of Things. Part 1. of The IoT Series”, 2003.