EARLY WARNING FOR PRE AND POST FLOOD RISK MANAGEMENT

FINAL REPORT

BSc (Hons) in Information Technology Specializing in Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology

8th October 2021

EARLY WARNING FOR PRE AND POST FLOOD RISK MANAGEMENT

FINAL REPORT

Dissertation Submitted in Partial Fulfillment of The Requirements for The Bachelor Of Science In Information Technology

Department Of Information Technology

Sri Lanka Institute of Information Technology

8th October 2021

1 DECLARATION

We declare that this our own work, and this dissertation does not incorporate without acknowledgement any material previously submitted for a degree or Diploma in any other University or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, we hereby grant to Sri Lanka Institute of Information Technology, the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Name	Student ID	Signature
S. P. M. K. W Ilukkumbure	IT18022902	
V.Y Samarasiri	IT18012620	vidurasamarasiri
M. F. Mohamed	IT18003406	FMohamed
Vinobaji. S	IT17181648	S vinobaji

The above candidates have carried out research for the bachelor's degree Dissertation under my supervision.

Signature of the supervisor: Date:

2 ABSTRACT

Flooding has been a very treacherous situation in Sri Lanka. Enabling to develop a structure to forecast extreme weather conditions will be a great aid for citizens who are affected from disasters. In this study authors explores the use of Machine Learning (ML), Deep Learning (DL), IoT (internet of things) and crowdsourcing to provide insights to development of the pre and post flood management system as a solution to flood risk management and mitigate potential flood risks. Machine learning and deep learning algorithms are used to predict upcoming flooding situations and rainfall occurrences using predicted weather information and historical flood, rainfall data sets. Crowdsourcing is used as a novel method for identifying flood threatening areas. Weather information is gathered from citizens, and it will help to build a procedure to notify the public and authorities. The IoT device tracks the Real-time metrological conditions and monitor continuously. The overall outcome showcases that machine learning models, deep learning algorithms, IoT and crowdsourcing information are equally contributing to predict and forecast extreme weather conditions. The integration of the above components with machine learning techniques, together with the availability of historical data sets, can forecast flood occurrences and extreme weather conditions with above 0.70 accuracy in specific areas of Sri Lanka.

Keywords: Crowd sourcing, Deep Learning (DL), Internet of Things (IoT), Machine learning (ML), Rainfall prediction, Flood prediction.

3 ACKNOWLEDGEMENT

Looking back to the past year of unforgeable and hard-working Research Project studies, we are thankful to many people and many things. Firstly, we would like to convey our heartfelt appreciation and credits to our supervisor Mr. Samantha Rajapakse, who provided us the treasurable prospect to test ourselves at the uppermost level we could do. You have invigorated and helped us to grow up from a fresh graduate student to a mature researcher. We have learnt so much from your both educationally and personally which has bought us to a great place in our studentship times. Thank you for all the priceless comments and guidance on our research.

We would also like to express gratitude to our parents for all the backing they gave us with no anticipation of a compensation. Their constant help, support and tolerance have made our life full of success, and we are glad for everything they have done.

Table of Contents

4 LIST OF FIGURES

5 LIST OF TABLES

6 LIST OF ABBREVIATIONS

Abbreviation	**Description**
API	Application programming interface
AI	Artificial Intelligence
CPU	Central processing unit
DB	Database
GPS	Global Positioning System
IDE	Integrated development environment
IoT	Internet of Things
IP	Internet Protocol
JSON	JavaScript Object Notation
MAC	Medium Access Control
ML	Machine Learning
OLED	Organic Light-Emitting Diode
PCB	Printed circuit board
RTC	Real Time Clock
SDLC	Software Development Life Cycle
SIM	Subscriber identity module
SMS	Short Message Service
SQL	Structured query language
SSD	Solid-state drives

TCP	Transmission Control Protocol
WLAN	Wireless Local Area Network
RP	Rainfall Prediction
SVM	Support Vector Machine
LR	Logistic Regression
RF	Random Forest
DCT	Decision Tree
ANN	Artificial Neural Network
MSE	Mean Square Error
RMSE	Root Mean Square Error
MCDT	Multi-Criteria Decision Tree
DMC	Disaster Management Centre
ROI	Return on Investment
VM	Virtual Machine
ML	Simple Leaner Regression
CART	Classification and Regression Tree

7 INTRODUCTION

Incidence of floods is a known risk to several localities across the world. Although floods are a result of a natural hazard due to sudden excessive rainfall, reasons caused by people have increased the risk of floods even due to slight rainfalls. Those reasons are mostly connected with various misuses of the land resources that are unfriendly or obstructing natural means of water flows to lows lands and sea. Therefore, the flood risk management involves in both a short-term strategy and long-term strategy. The long-term strategy requires globally agreed efforts to control human activities to protect the environment and control the global warming below 1.5 degrees over the pre-industrial climate level. The short-term strategy is to protect people from ongoing risks of floods. This strategy requires the techniques for prediction of floods and flood disaster management system to issue alerts, evacuate people urgently from flooding areas.

Study area was selected as Kalu River basin area in Sri Lanka, which is prone for high number of flooding. A recent study has found that use of machine learning models and deep learning algorithms are widely used in building up prediction models for weather forecasting [1]. Historical flood and rainfall data are used as the inputs for training data set. IoT devices monitor the changing weather conditions continuously and the readings from the IoT devices are added to training the model for forecasting predictions. Crowdsourcing offers a technique for gathering information from public crowd and validation. The gathered data sets are analyzed and validated using statistical analysis techniques to organize and classify the final data sets. In this paper machine learning models are developed together with artificial neural networks for predicting flood occurrences and rainfall based on historical data sets. Furthermore, utilization of IoT device and validated crowdsourcing data sets provides supports early warning system to forecast weather information.

7.1 Literature Review

The concept of flood risk management is now growing fast in globally. Flood risk management is the framework for managing flooding situations. The main intention of flood risk management is to identify floods, being activated before a flooding situation occurs and reducing its possible impacts [2]. The flood risk management provides a set of conceptual frameworks for before, during and after a flooding situation occurs. These steps will help to implement strategies for early warning, weather forecasting and rehabilitation plans [3]. In the flood risk management, planning stage includes the main objective of minimizing the risks with the help of early warning systems and assessing the procedure for reducing flooding situations. To execute this strategy, extreme weather conditions and flooding situations should be monitored constantly. Geo graphic information systems are used to assess flood risks by using risk-based methodology [4] . Different data sets are used for forming rainfall predictions and flood predictions. These model-based predictions are used for developing early warning systems. Natural disasters can be tracked down and monitored using different devices such as sensors, IoT devices. Humans can act as sensors. Crowd-sourcing information can be taken as human sensors and those information which are gathered from devices, sensors and citizens will be helpful for building up the predictions models as well.

The proposed rainfall prediction model uses data sets (Temperature maximum, Sea Level Pressure average, and Relative Humidity minimum in day and month based on location) to check the contribution of each attribute and create models (Logistic Regression, SVM) for 3 different locations (Coastal and City, countryside, and hill area) to check the accuracy difference when using the same data set for 3 locations. Research papers regarding "Rainfall Prediction" used data sets such as temperature, vapor pressure, and relative humidity, as well as multiple models such as Logistic regression (LR), Decision Trees (DCT), Random Forest (RF), Support Vector Machine (SVM), Artificial Neural Network (ANN), LASSO regression, Naive Bayes, and check for the best model based on accuracy, error (RSE and RM-SE), recall, and F-Measure[1], [5], [6], [7].

Upon following a review on the current literature in Sri Lanka [8] and other countries, for machine learning models several methodologies were identified. These flood forecasting models have a number of approaches to the problem, ranging from traditional statistical approaches to machine learning and data mining methodologies [9]. The Simple Liner Regression based models' linear regression-based algorithm to forecast and SVM does supports hyperplane to define to groups.

Most of these models are classified as single and hybrid models. Hybrid machine learning models are widely used and has better performance on selected models. Artificial Neural Network (ANN) algorithms selecting add-on optimizer was recommended [8].

Identify these issues of existing flood forecast system and optimizing those methods to develop a more accurate systematic model to provide predictions using computational approach, will be discussed in bellow onwards.

Crowd sourcing approach on disaster risk reduction focuses on collecting and analyzing data in a disastrous event. The similar studies have been done in Brazil to implement a crowd source-based approach to collect weather information from volunteers to improve and forecast flood risk management [10]. similarly, Taiwan also has developed a system named TSER (Taiwan scientific earthquake report system) for disaster monitoring which obtains information from trained volunteers, and it is used to identify earthquakes and disastrous events before and during disasters [11]. The crowdsourced data sets that are gathered from volunteers are unstructured and unorganized.

These data sets should be organized and analyzed before using for further processes. To overcome this issue, research on crisis information management in the web 3.0 age implemented a machine learning based approach for transforming and classifying crowdsourcing data sets to a usable format to support making decisions in a disaster [12]. Authors on this research focus on classifying and analyzing crowdsourcing data sets in more technological way. Crowdsourcing establishes better and improved risk communication between affected people, volunteers and relief providing authorities [13]. Sahana is a volunteer coordination system which was implemented to manage and coordinate possible risks related to disasters. This crowdsourcing system is mainly focused on the outbreak of 2004 Asian Tsunami to find missing victims, relief persons and identify damages to buildings, houses, and belongings [14].

With a swinging change in the atmosphere incessantly there are damages to the people who dwells in adverse areas. Also, there are some areas where the people aren't informed about the consequences precisely in their areas, due to the rapid conversion of weather conditions regularly [15] In the current world, knowing the live environment has become a greatest challenge because of the impediments IoT encounters when it comes to assessing the live meteorological conditions [16]. Also, fields such as constructions, agriculture, manufacturing etc. also causes assorted defies in the nature's condition [16]. Since all of us are aware, agriculture performs a crucial role in the country's Economy.

Internet of Things (IoT) could be used to watch and control numerous devices remotely with the help of sensor network which has the ability of sensing, processing, and transmitting the data to a cloud. A cloud is a service which provides a proficiency of advance reckoning, storing, and producing information in a more precise format. From the cloud service these data could be made available on various platforms such as mobile applications, web applications etc. [17]. The internet acts as the heart of revolution while playing a dynamic responsibility in reliability, productivity, and quick communication from information from the device to the cloud contrarywise. IoT extends a wide range of connectivity of objects with miscellaneous procedures and properties of applications for acquiring a complete machine to machine collaboration [18]. Applying all the best practices and modern technologies in IoT the public and the relevant officials will be able to overcome majority of the intense weather conditions and be prepared for any circumstances.

Globally, the notion of flood risk management is rapidly expanding. Flood risk management is the framework for dealing with flooding emergencies.

The primary goal of flood risk management is to identify floods, activate before a flooding crisis develops, and reduce the potential consequences. Flood risk management offers a collection of theoretical foundations for before, during, and after a torrential downpour.

Prior research concluded that meteorological data provided a significant influence in flood forecasting. There are several research covered in area of flood damage mitigate methods [19].

Upon identifying the pre-existing flood forecast systems, most of the data not-real-time and the existing data wasn't sufficient for creating flood forecast systems. But machine-learning algorithms coupled to data mining methodologies can provide environmental forecasts based on historical data [19].

Current systems with low quality data input and machine learning models have a significance of existing rainfall gauges are incapable of accurately capturing the spatial distribution of precipitation. Estimates of rainfall can easily overstated and discharge data is only accessible during the flood season [19].

Another flood forecasting system conducted in the Kalu River basin area by Sri Lankan Irrigation Department and ICHARM Public Works research institute of Japan had concluded a simulation of Rainfall Run off Inundation (RRI) model has revealed that the river dimensions had a significant impact on model output. And even with the use of satellite data was used to validate the flood inundation [20]. In the same researchers had issue of not have enough the computational power of simulation in their existing systems.

7.2 Research Gap

Related studies and research have been conducted in worldwide by approaching crowdsource based model to capture disaster data from community. When it comes to volunteer-based systems, active participation of crowd plays a major role. These Studies have successfully conducted crowd source-based approach to enable disaster response by implementing a mobile app feature. Focusing on improving self-organizing and disaster resilience by participating volunteers in a disaster situation.

Considering some past research studies with related to same field of weather monitoring using IoT has being done. A fully functional smart weather monitoring device with all sensors included in it to monitor the weather is more productive than having many smart devices to monitor the weather data factors. Also, some studies have gathered data with the assistance of an IoT device and previewed on a web application despite of any verification or validation of data which may cause to misleading of information. The flood water level prediction using machine learning adequate training data. Which will affect the decision-making models key in. The decision-making model will be a supervised learning algorithm. This will result in a model with water level prediction and classification with warning decision.

This proposed system has the ability to monitor the main key factors of the environment which could be used to conduct a prediction for flooding and extreme weather conditions. These gathered information will be verified and validated with the other models in this system such as ML, AI, and crowdsourcing technologies. Since this information will be consumed by the authorities for various instances validation and accuracy is the main key feature of this system. For the users to access all the weather information for various procedures a mobile application and web application is developed Also, at times due to intense weather situations stakeholders might not be able to access the system with their usual platform, to avoid such disruption we have a SMS base weather providing system which any user could receive weather information from any location where an IoT weather monitoring device is place upon request.

This proposed research uses machine learning models such as Support Vector Machine and Logistic Regression to predict rainfall using temperature maximum, relative humidity, and sea level pressure as attributes. This study only checks secondary weather factors' contributions in three different locations, such as Colombo (coastal and city), Vavuniya (countryside), and Katugastota (Hills) of Sri Lanka, causing rainfall by using location and time (day, month). At the end of the research, we can conclude a small factor's contribution to rainfall based on the accuracy difference of each location model or prediction error.

The machine learning models for the

Research Papers	Flood Prediction	Rainfall Prediction	Crowdsourcing	Smart Weather monitoring device	Solution for Non-Subscribed users
[1]	✗	✓	✗	✓	✗
[7]	✗	✓	✗	✓	✗
[8]	✓	✗	✗	✗	✗
[10]	✓	✗	✓	✗	✗
[16]	✓	✓	✗	✓	✗
[21]	✗	✗	✓	✗	✗
Proposed Solution	✓	✓	✓	✓	✓

Table 1: Comparison with existing researches

7.3 Research Problem

When a flooding situation takes place at a specific location as the water rises quickly it will take some time to get fully prepared for public to evacuate from the area. When the water level reaches the peak, it will start to flood in their living area and cause immediate damage, therefore with the help machine learning based rainfall prediction models, flood prediction models, crowdsourcing and use of extreme weather and flood IoT device can implement a solution for this major matter.

At present, when a flooding situation takes place at a specific location as the water rises quickly, it will take some time to get fully prepared for public to evacuate from the area. When the water level reaches high, it will start to flood in their living area and cause immediate damage. When it comes to disaster management and flood tracking, the major problems that we identified are mentioned in bellow factors.

- Unavailability of an early warning tool will be very costly for most of the countries
- One of the major problems that countries face when a flooding situation takes place is, loss of human lives, property losses, agricultural losses, and economic losses.
- Due unadvanced system, poor coordination between people and the officials increases the flood disaster loses and recovery plans are delayed.
- To address these situations, we propose to develop an early warning structure to minimize the devastating destruction that could be caused.

Therefore, the proposed flood risk management system consists of rain fall prediction, flood prediction, crowdsourcing and extreme weather and flood tracking IoT device will overcome the above-mentioned problems

8 RESEARCH OBJECTIVE

8.1 Main Objective

Provide an early warning mechanism and predict severe weather conditions which may cause flooding with the use of real-time data and historical data

8.2 Specific Objective

To full fill above mentioned main objective, the specific objectives that need to be covered as follows,

1. Develop severe rainfall prediction model based on historic data analysis and provide warnings and suggestions to the end users
2. Create the Flood Forecasting Model to predict the flooding for the selected specific area using historic data collected.
3. Develop crowdsourcing solution to gather weather information from public crowd, analyze, validate and present crowdsource data to end users
4. Designing and implementing a smart device to gather live weather data information, transmitting gathered information to the applications in a precise manner, and reaching out to the users who aren’t able to access this system due to various reasons

9 METHODOLOGY

This section highlights approach for solving the research problem by applying suitable methodologies. This section consists with the overall system architecture, rainfall prediction model component, flood prediction model component, crowdsourcing component and flood and weather tracking IoT device implementation methodology, development process related to above mentioned components, data collection and requirement gathering, feasibility study, design components and commercialization of the overall system.

9.1 System Overview

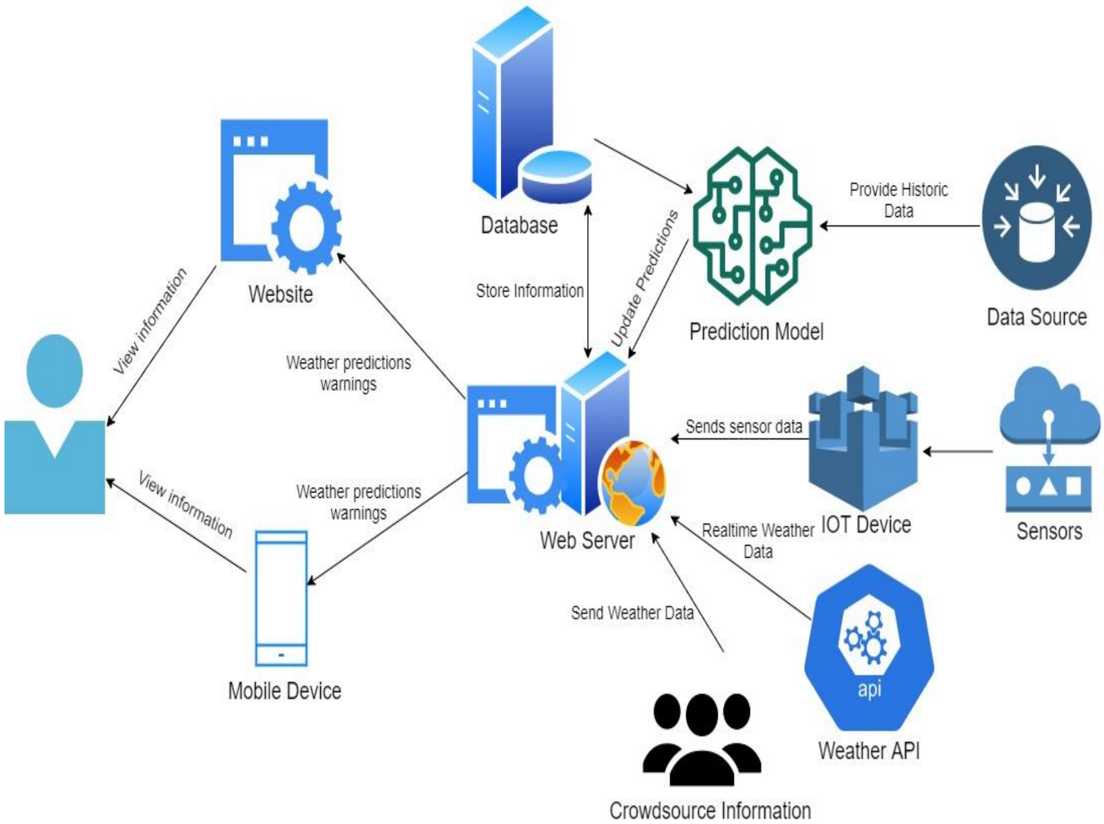


Figure 1: System Overview

9.2 Component Overview

9.2.1 Rainfall Prediction Model

This rainfall prediction system was developed to predict rainfall, which one day will be used in flood decision-making models and provide predictions for people. Historical weather data is used to build machine learning models and weather data collected from IoT devices is used to predict rainfall.

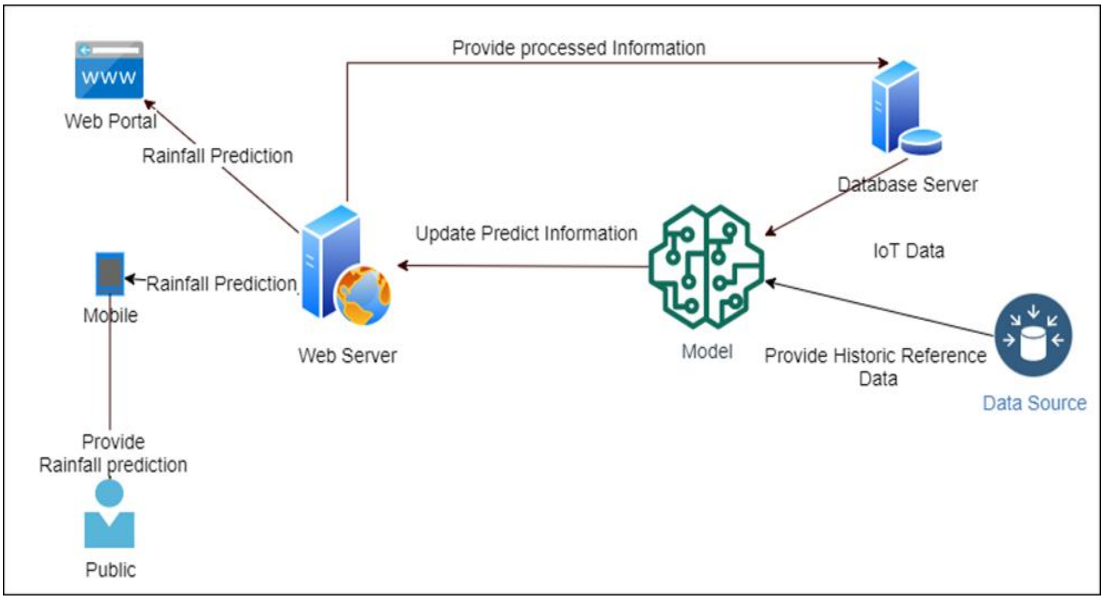


Figure 2:Rainfall Prediction System

In this system, the machine learning methods of logistic regression and Support Vector Machine will be used to predict whether it will rain or not and the rainfall amount range (mm) by using data sets and checking accuracy differences. After the observation of the prediction accuracy, you can make assumptions and test them with the needed evaluation in the training model again and again. We can draw some conclusions based on the accuracy difference between each data set. At the end of the research, they found out the best dataset for the prediction model and predicted the rainfall and rainfall range.

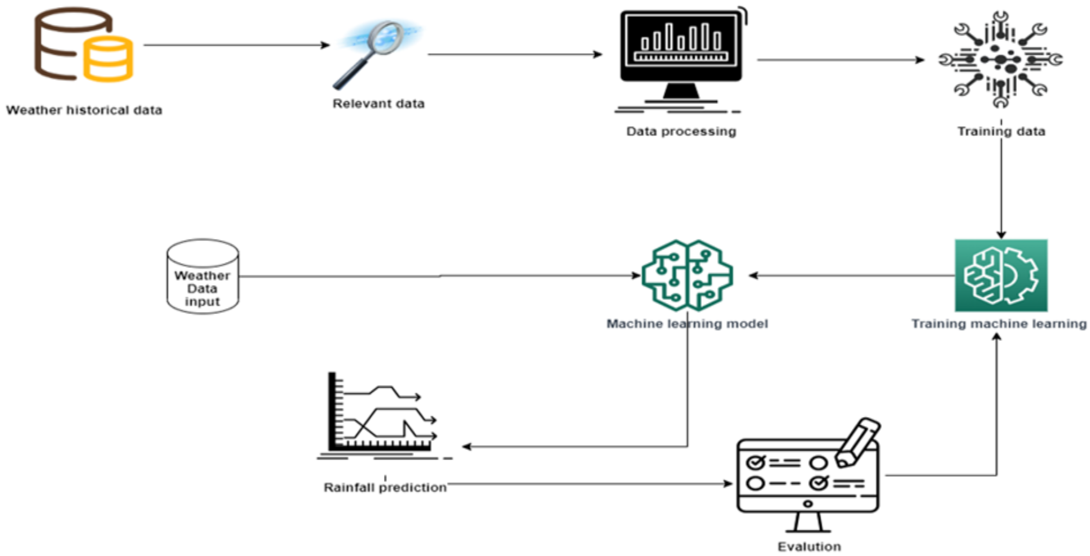


Figure 3:Process Diagram of Rainfall Prediction

9.2.2 Flood prediction component

9.2.2.1 Flood prediction component Development

Simple Liner Regression (SLR), SVM and ANN methods are used to develop the machine learning algorithm. The Deep Learning approach was much suitable for this scenario Artificial Neural Network (ANN) are commonly utilized in machine learning and most of the commonly used models are composed of an input layer, hidden layer and output layer as shown in Figure 2. The most complexity is handled in the hidden layer.

Decision making Model: Decision making model utilizes flood forecast of gauge stations, rainfall prediction model and IoT sensory data. It is a multi-criteria decision tree based (MCDT), model is most suitable for classification and can executed immediately and accurately.

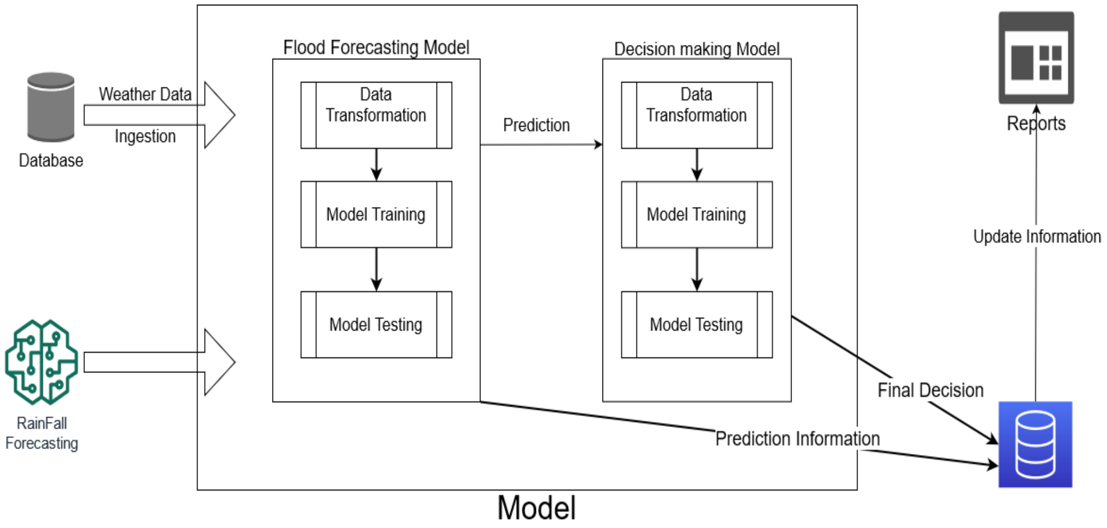


Figure 4 AI Model

As in Figure 4 AI Model these results of water level prediction model are be presented and passed on to the results in order for the decision-making model to make the final decision about the situation.

9.2.3 **Crowdsourcing Component**

This section highlights the component diagram of crowdsourcing, logical view of crowdsourcing solution, crowdsource process design and development approach. Volunteer weather information is gathered through the mobile application and stored in the real-time firebase database. When visualizing the crowdsourced data, the most accurate crowdsource data is displayed to the users with IoT device data and weather API data

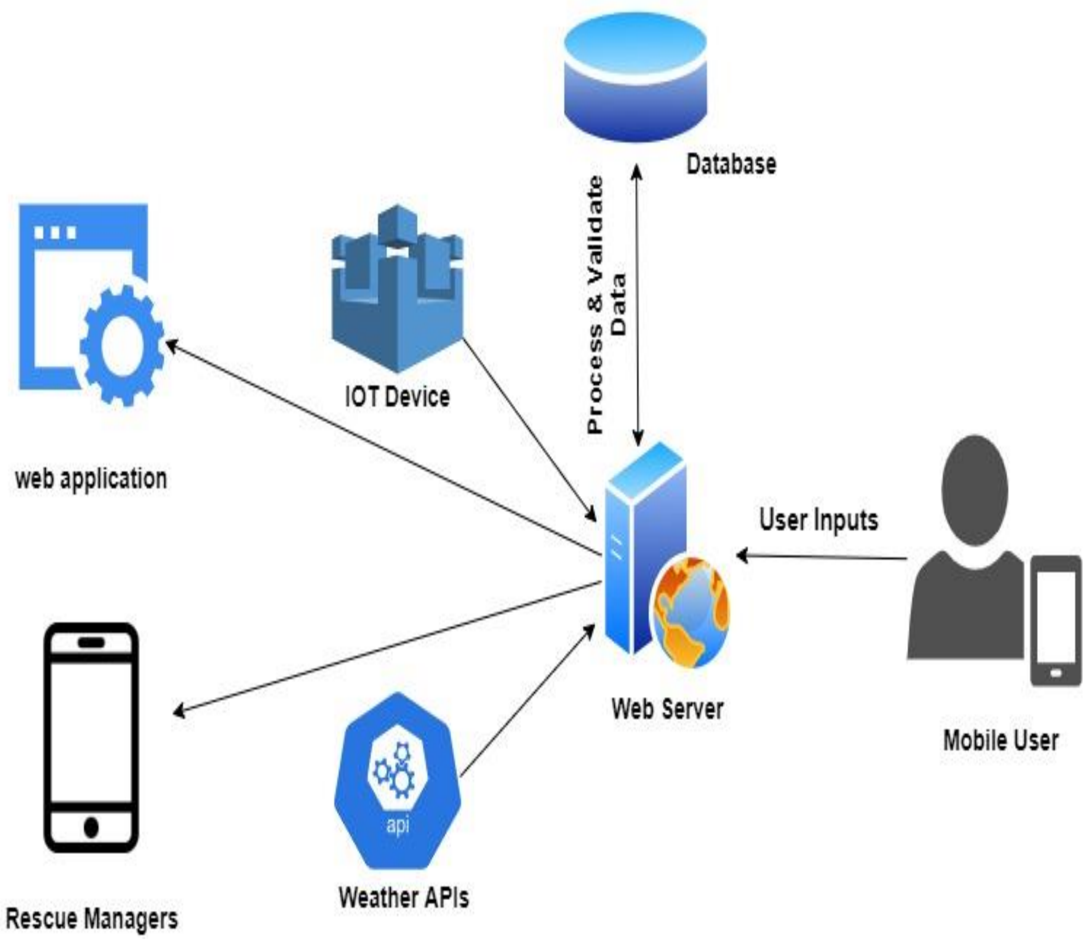


Figure 5 High-level View of Crowdsourcing

The logical view of crowdsourcing is represented in bellow figure

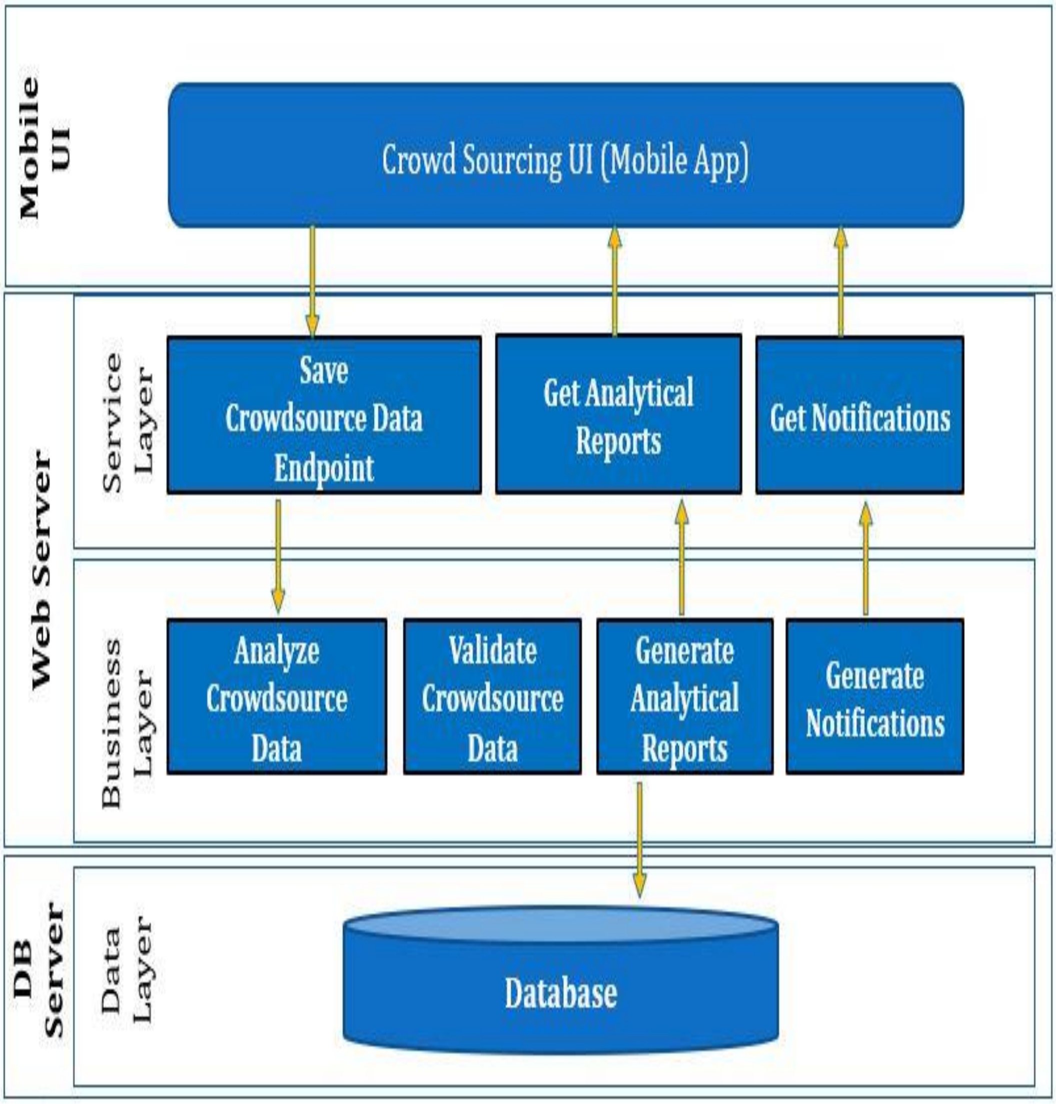


Figure 6 Logical View of Crowdsourcing

Crowdsource process design

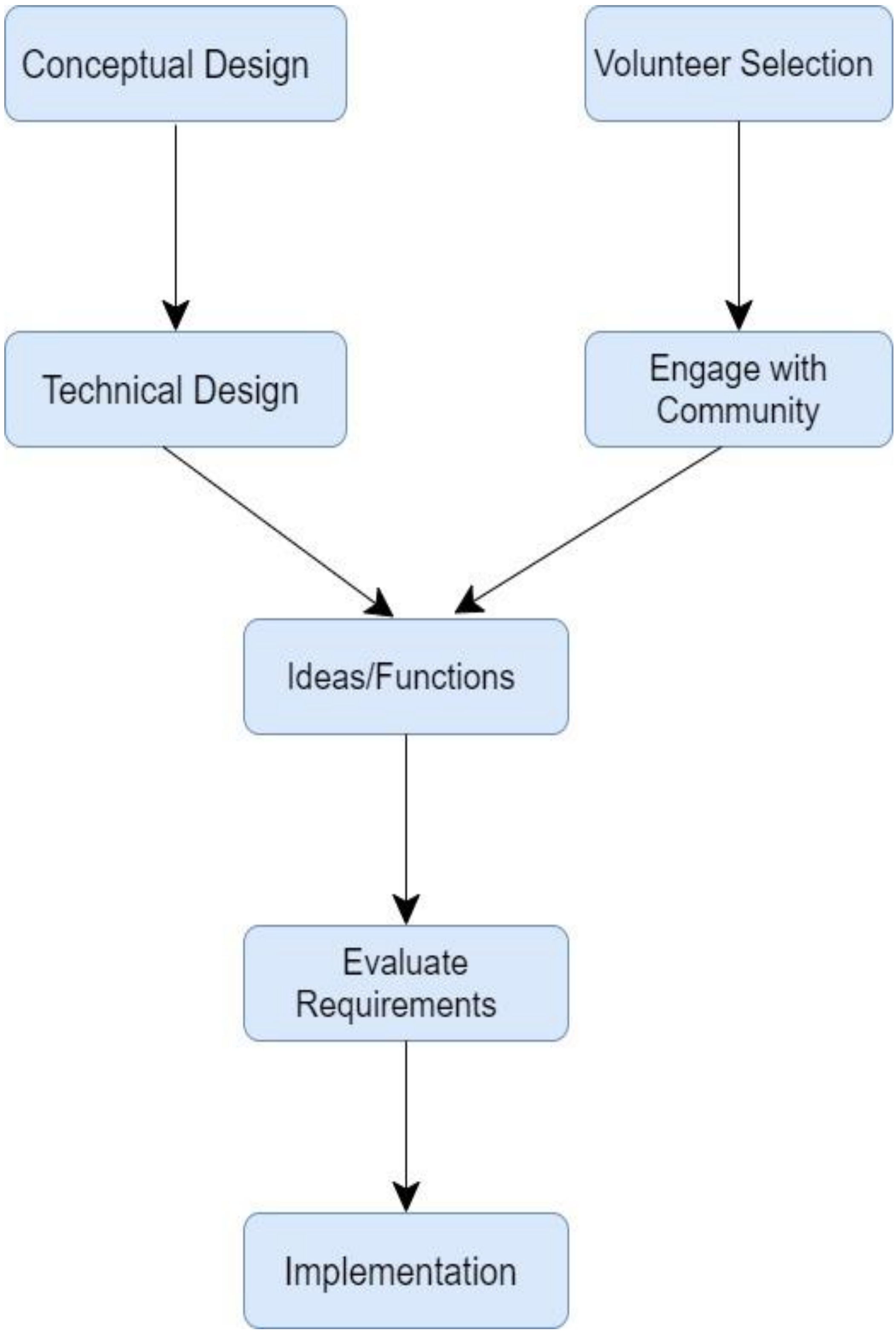


Figure 7 Crowdsource Process Design

9.2.4 Crowdsourcing Approach development

The proposed early warning system is capable and have the below mentioned functionalities in terms of crowdsourcing,

- Gathering data from public crowd
- Validating and analyzing the gathered data
- Structuring data in precise and concise format
- Crowdsource data comparison with weather API and IoT device
- Receiving weather information periodically from public crowd
- Providing analytical reports and notifications

Crowd sourcing offers a technique of devolved and a low-cost approach for gathering and exchanging information. The overall intention of providing this feature is to approach the public crowd to collect their current weather information and analyze and validate those information before publishing them to users. The crowdsourcing section consists with a UI containing predefined questionnaire to gather information from public crowd and it enables to receive real time updates in a specific location and helps users to monitor floods and extreme weather conditions in nearby locations during the crisis.

Since the solution gather information from the public crowd, the gathered data should be validated and analyzed. The gathered data sets will be stored in firebase live database. The data analysis will be done using Statistical Data Analysis methodologies to identify the most common and matching crowd sourcing data set based on the location and comparing datasets to identify how they are statistically different each other. The validity of analyzed data (outcome of data analysis) will be done by comparing with live data sources such as third-party weather APIs, inputs from the proposed flood tracking and weather monitoring IOT device. Apart from these implementations active participation of crowd plays a major role in crowd source-based systems, therefore, to enforce active participation the system will provide periodic based alerts for the users to send status of weather in their living area which will help nearby people to be in touch with the current weather situation. The final dashboard does a major role in visualizing data, and it provides real time data management After processing these series of data manipulations, the final condensed crowd sourcing data set is published through the system based on the location

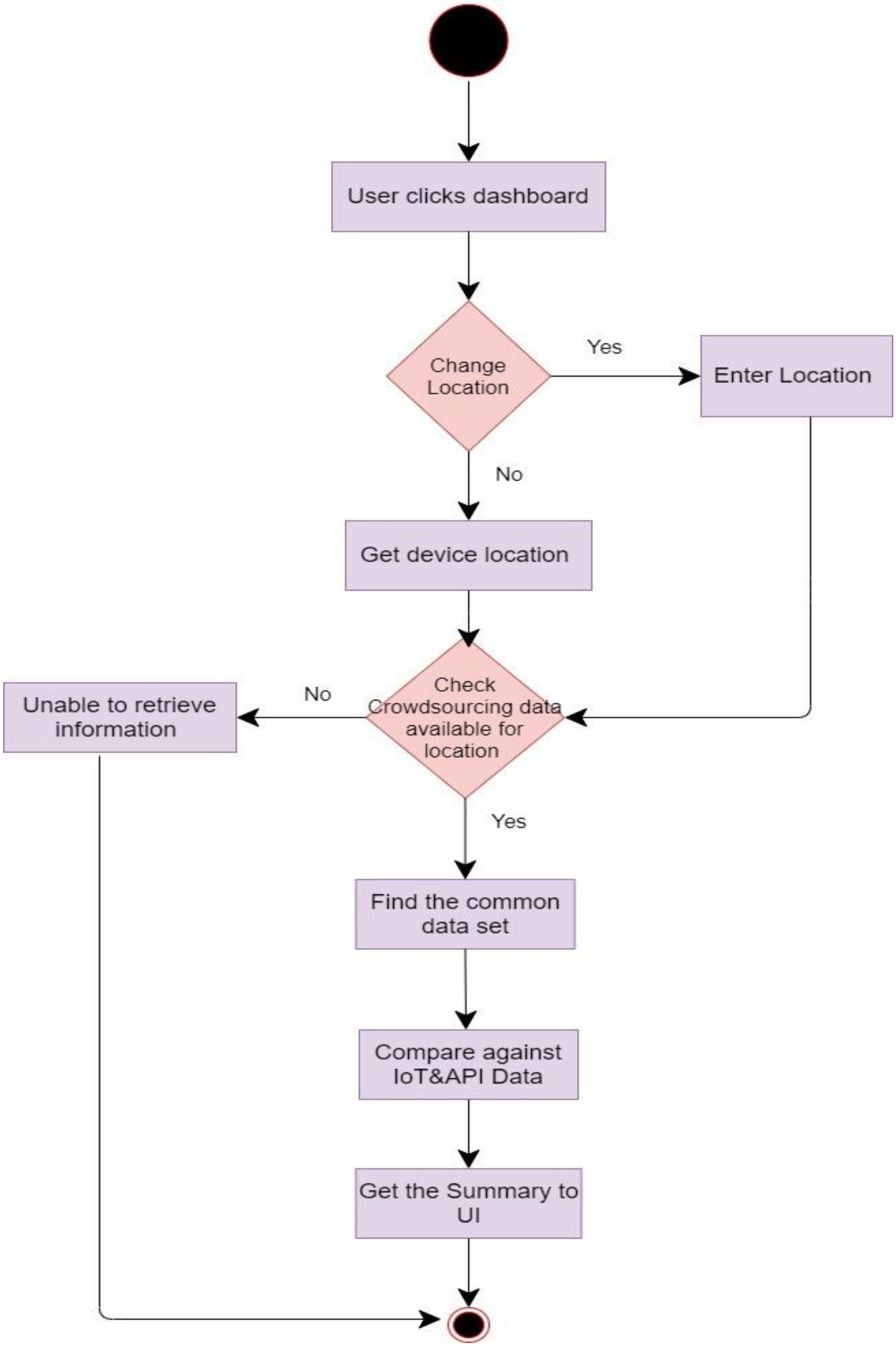


Figure 8 Crowdsource Data Manipulation Flow

9.2.5 crowdsource data analysis

Crowdsource data analysis is done by using K means algorithm to reduce number of false entries and get the most common data. The K means algorithm starts with first, picking the number of clusters which is known as K. After picking the K vale, next we randomly select the centroid of each cluster. As an example, if we want two clusters the K will be equal to two and then randomly select the centroid. Once centroids are initialized it assigns each data point in the cluster to the closest cluster centroid. calculating the distance from centroids to each data point is performed by Euclidean distance calculation. After assigning all the data points into two clusters the next step will be computing the centroids of newly formed clusters and it updates the new centroid. This iteration will continue till the centroid no longer changes.

In our mobile application we used K-means clustering algorithm to reduce invalid data error and get most common data from crowdsource data. When we consider current weather situation, there can be several values because of different users are experiencing different weather situations in their location. So, we must extract most common data from these different data set. To execute that using k-means algorithm, we divided our dataset into three clusters. Cluster is a set of data that contains similar data. Similarity is calculated by the distance of each data point to the centroid of the cluster. Centroid refers to the middle point of a cluster. From these 3 clusters we can find one cluster that contains more data values. It represents the cluster that contains higher number of similar kind of data. So, we can choose that cluster to achieve our goal. By choosing this cluster we can reduce error of invalid data, because invalid data are in remaining two clusters. even though we chose this cluster sometimes it will not contain exact same data. In that case again we must get most common data inside that cluster. To perform that operation, we can simply count the occurrences of each data values and get the most common value. After getting that common data value it will displayed in the text field in the relevant section of the dashboard.

9.2.6 iot Device & SMS System Component

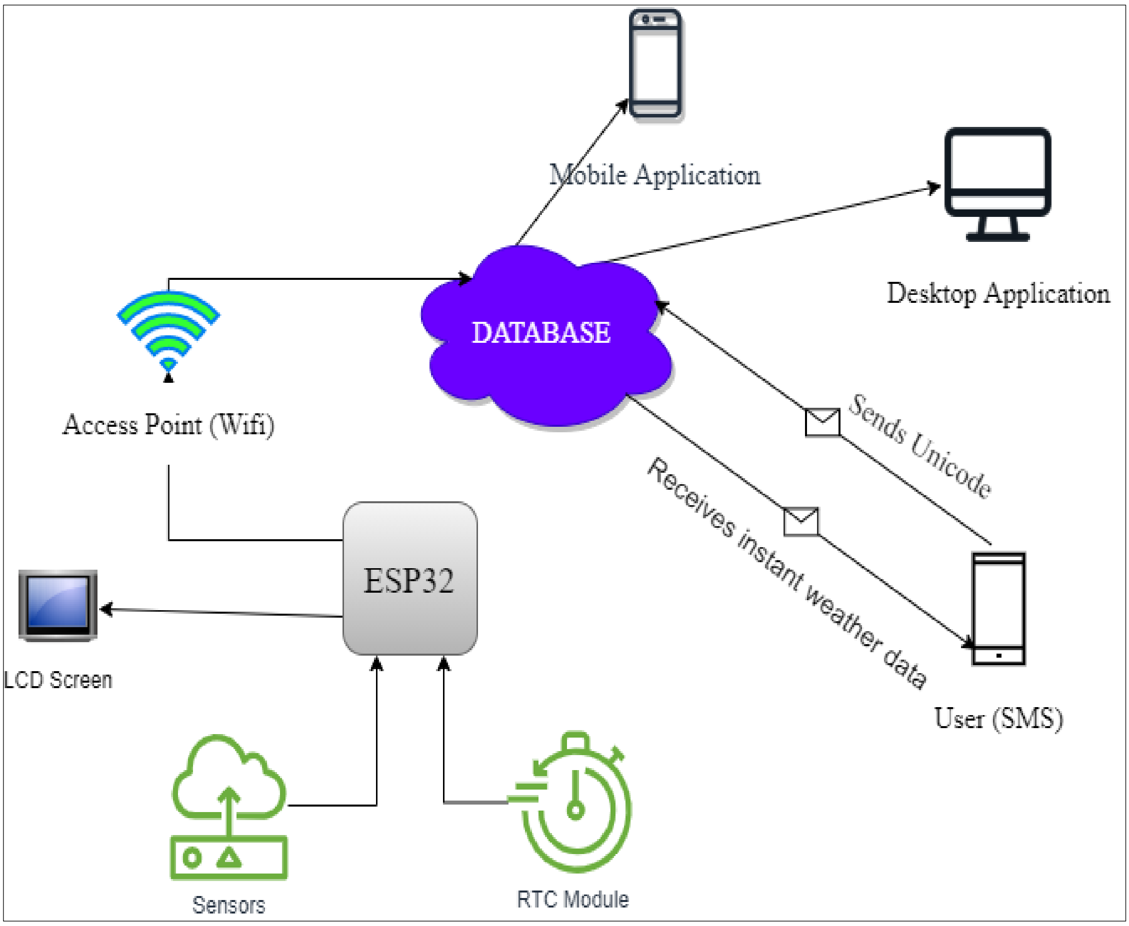


Figure 9: IoT Device & SMS System Overview

Smart Weather Monitoring Device Component Development

Before manufacturing and designing this weather monitoring device a deep study was done on Internet of Things(IoT) on its operations and functions and a way to design a PCB board which will also bring a feasible structure to the weather data monitoring device.

This developed early warning system has the capability and the ability of having the below functions,

1. Detection of Temperature & Humidity.
2. Detection of the intensity of Rainfall.
3. Detection of the increasing Water level.
4. Transmitting weather data information to the users via SMS on request.

This designed weather monitoring device will be playing a major role in this project as in many live weather data factors will be gathered by the help various types of sensors which are unique for their responsibilities, where these gathered data will be used for various models in this project.

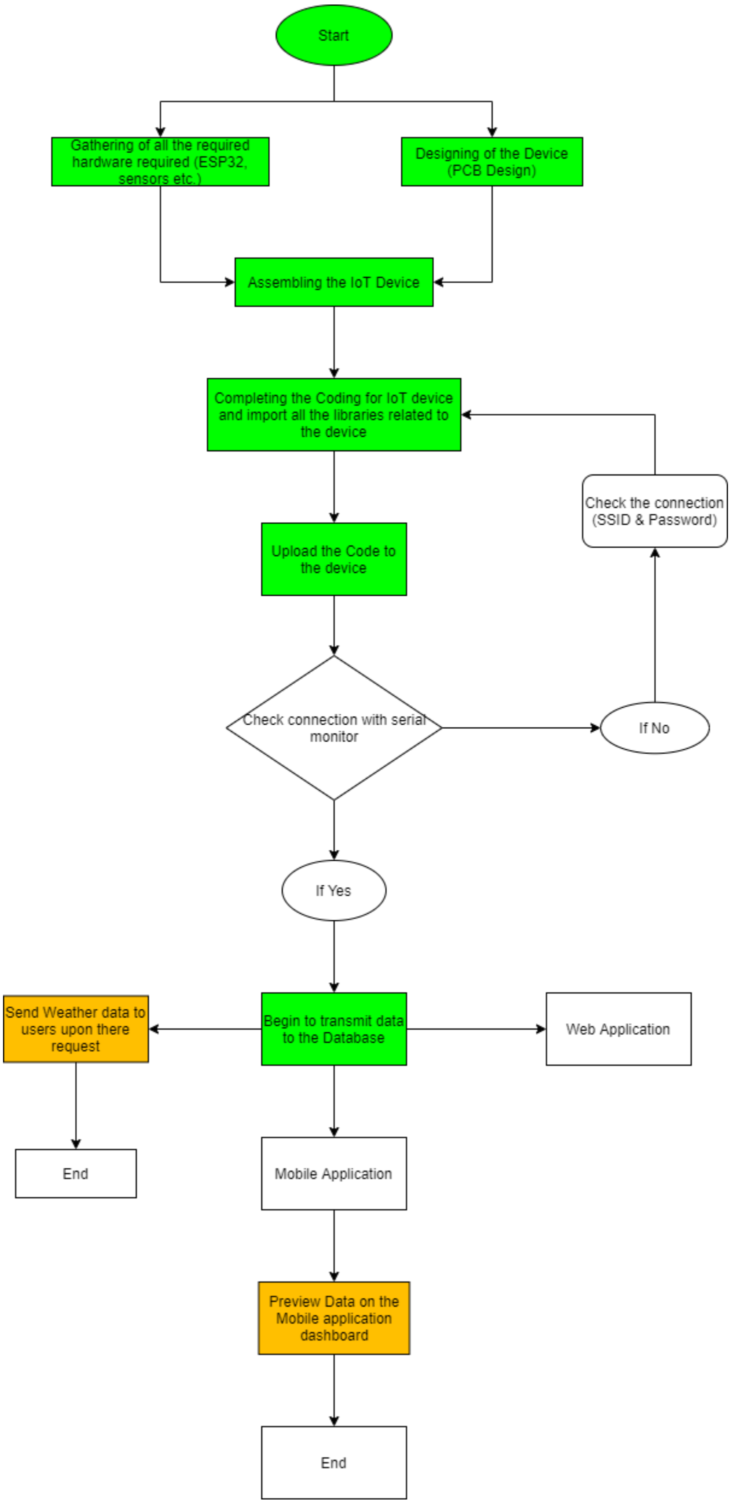


Figure 10: Flow diagram of the IoT Device

In this designed IoT aided weather censoring structure, an ESP32 assesses 4 factors by exploiting the sensors. They are, DHT11 Sensor, which is used to monitor the temperature and Humidity, AJ-SR04T sensor which is used to monitor the increasing of the water level, Rain Drop Detection sensor and a DS1307 Clock RTC Module which will provide a precise time and date which is used to note the readings of each respective sensor.

When it comes to the perspective of usage some users may use our mobile application/Web site some may not use due to various reasons. But we will still be providing a solution for such users to get details via our system. This will be done where the user will have to send a request to our system via SMS and the user will be able to get limited important details related to weather data via the same mode.

9.3 Development process

Software development life cycle (SDLC) is the approach for developing software applications. It is split up to six phases: Planning, requirement analysis, design, implementation, testing & integration, and maintenance. The phases are depending on the software scope. As per our requirements these are the main components approved for software development. Software development life cycle is the methodology for enhancing the software development process and it improves the efficiency of each phase of software development life cycle.

To develop the solution for the early warning for pre and post flood risk management system it is mandatory to adhere to proper software development methodologies. Planning stage is an important stage in the development process since that phase should be completed carefully with a proper idea of the development. Requirements gathering and analysis is done using various methodologies to gather available information and requirements for implementing flood risk management methodologies in terms of rainfall prediction, flood prediction, crowdsourcing and development of flood and weather tracking IoT device. Volunteer communication, field observations are done at this phase for implementing design strategies for mobile application and web application. Field observations are important to develop strategies for flood risk management.

Depending on the requirements, designing and sketches are done in the next phase of the software development life cycle. After selecting the technologies and tools the implementation is started adhering to latest implementation strategies and best practices. After completing implementation phase, software testing phase begins, and it will be followed by deployment and maintenance phase. The overview of the software development life cycle model which we selected to develop early warning for pre, and post flood risk management is outlined in bellow figure

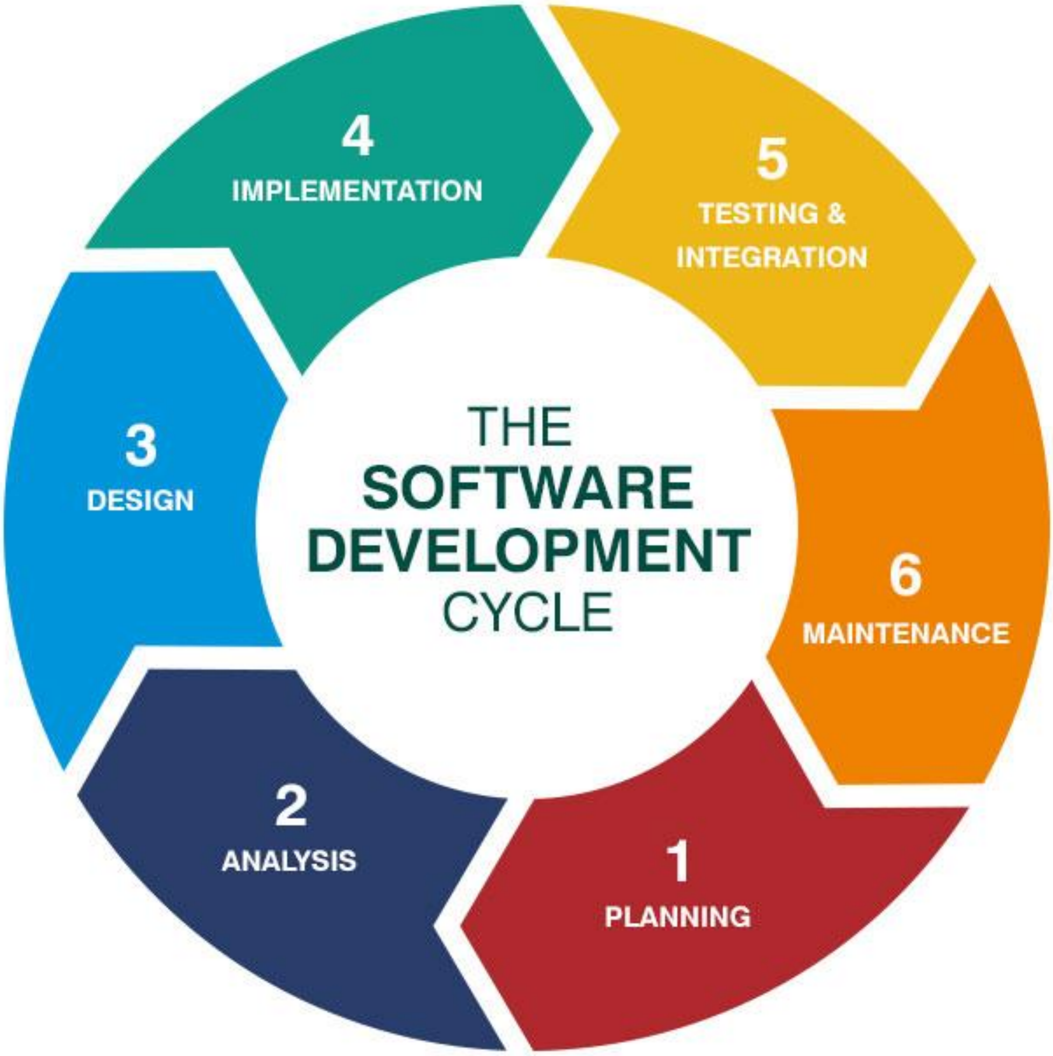


Figure 11: Software Development Life Cycle

9.4 Requirement Gathering and Data Collection

9.4.1 Rainfall Prediction

Historical weather data such as maximum temperature, minimum relative humidity, average pressure, and rainfall are collected daily from the meteorology department of Sri Lanka. The historical daily data from 2015–2019 period was collected from three different locations of Sri Lanka, such as Colombo, Vavuniya, and Katugastota.

Attribute	***Type***	***Description***
Relative Humidity	Numeric	Rh Min(percentage)
Temperature	Numeric	Max temperature (°C)
Pressure	Numeric	Sea Pressure mean(hpa)
Month	Numeric	month
Rain (targeted value)	Numeric	Boolean: 1 or 0
Rainfall range (targeted value)	Numeric	Ranges(0,1,2,3,4)

Table 2:Attributes

It is preferable to gather a large amount of data to improve the accuracy of the output of this rainfall prediction system. With four months of missing data, the collected historical data is substantially organized. Two extra attributes, "Rain_Or_Not" and "Rain Range" were created based on the collected rainfall attribute. The "Rain or Not" column was created based on a binary option if it rained (1) or not (0). The "Rain Range" column was created based on multi-classification tags such as (0.01mm as (0), (0.1-10) mm as (1), (10-20)mm as (2), (20-30)mm as (3), and (30+mm as (4).

	Station_Name	date	yy	mm	dd	Tem_Max	Pressure	RH_Min	Rainfall(mm)	Rain_Or_Not	Rain_Range
0	Katugastota	1/1/2015	2015	1	1	29.1	956.70	79	0.0	0	0
1	Katugastota	2/1/2015	2015	1	2	29.2	958.25	67	0.0	0	0
2	Katugastota	3/1/2015	2015	1	3	31.5	958.75	60	0.0	0	0
3	Katugastota	4/1/2015	2015	1	4	30.3	958.75	69	0.0	0	0
4	Katugastota	5/1/2015	2015	1	5	29.3	958.00	74	0.0	0	0
...	...	...	...	...	...	...	...	...	...	...	...
1791	Katugastota	27/12/2019	2019	12	27	29.8	957.05	70	0.0	0	0
1792	Katugastota	28/12/2019	2019	12	28	30.8	957.00	66	0.0	0	0
1793	Katugastota	29/12/2019	2019	12	29	29.7	957.50	63	0.0	0	0
1794	Katugastota	30/12/2019	2019	12	30	30.1	957.55	63	0.0	0	0
1795	Katugastota	31/12/2019	2019	12	31	30.6	958.45	69	0.0	0	0

1796 rows × 11 columns

Table 3:Katugastota Sample Data

9.4.1 Flood prediction

Data Collection

Data Sources are essential part of the building data-driven modes. The trusted data is collected by Metrology Department, Irrigation Department and Disaster Management Center in Sri Lanka. The data collected by past 3 years (2019 -2021) such as amount of rainfall occurrences and real-time basis, water levels change in river on time-basis, number of water levels discharged in a specific station, real-time water levels, rainfall durations, precipitation, elevation, flow directions of the river and rainfall are used in creation of machine learning models.

The data collected are sampled by twenty-four-hour intervals. And these collected data is collected from different source pre-processing of the data sets to normalize into common data frames was essential. The pre-processed datasets used in developing machine learning models for water level prediction models are presented in the Table 4 Water Model Processed Data Set Overview.

Table 4 Water Model Processed Data Set Overview

Factors	Time Interval	Type	Description
Tributary	2019 - 2021	Text	River / Sub River name
Station	2019 - 2021	Text	Gauge station name
Catchment Area	2019 - 2021	Km^2	Catchment area of the water near river gauge station.
Water Level Classification	24 hours	Remark {Normal, Alert,	Four levels of water classifications

		Minor, Major}	
Water Level Before	24 hours	m	Water level 1hour before read
Water Level at Time	Present	m	Water level when the readings were taken.
Fluctuation	Present	Remark {Normal, Rising, Falling}	Water level has a significant change since the 1hour reading to present.
Rainfall	24 hours	mm	Rainfall near the water gauge station.

In [3]:
```
river_data = pd.read_csv('SeptDataSet4ANN_forColab.csv')
river_data.head()
river_data
```

Out[3]:

	Alert Level	Minor Level	Major Level	Level before	Water Level at time	Rising 2 or Falling 1 or Normal 0	RF in mm	F4, F3, Alert 2, Normal 1, Empty 0
0	3.0	6.0	7.5	2.18	6.84	0	0.8	1
1	5.0	6.0	5.5	2.14	6.29	0	2.0	1
2	5.0	4.5	5.0	4.62	2.88	2	40.0	1
3	5.0	10.7	7.0	4.14	2.12	1	9.8	1
4	10.0	10.7	12.2	8.45	8.92	0	6.1	2
...	...	...	...	...	...	...	...	...
9995	4.0	6.0	7.5	3.12	2.43	0	9.0	1
9996	3.0	4.5	8.0	8.45	8.28	0	9.0	1
9997	10.0	10.0	12.2	4.57	2.40	0	17.1	1
9998	10.0	10.7	12.2	3.31	8.92	1	38.5	2
9999	4.0	6.0	7.5	2.87	4.12	0	0.7	1

10000 rows × 8 columns

Figure 12 Station 4 - Rathnapura dataset

Real-time IoT devices sends data through the IoT devices are intended to be used in detecting the water level model and rainfall model. The bellow Table 10 provides an insight into data used in creation of decision tree model for decision making.

Table 5 Decision Tree Model Processed Data Set Overview

Factors	Unit	Type	Description
Distance	cm	Numeric	Distance from water level surface to the sensor
Humidity	percentage	Numeric	Relative Humidity from sensor
Temperature	Celsius	Numeric	Relative Temperature from sensor
Rain	Binary	Boolean {Rain, Not Rain}	The water droplet sensor reading about currently raining or not.
Rain Range	Class	Remark {No Rain, Rain, Heavy Rain}	Raining Classification by the Rain forecast model at in interval.
Water Level	Class	Remark {Normal, Alert, Minor, Major}	Water level classification by Water Level forecasting model.
Warn	Binary	Boolean {warn, not warn}	Decision Tree Decision.

```
df.head()
df
```

	timestamp	distance	Humidity	Temp	Current_Rain	Rain_Range	Water_Level_Class	Warn
0	1626886919	3	82	31	0	0	0	0
1	1629734639	32	78	31	0	0	0	0
2	1628334639	32	78	31	1	2	0	0
3	1626886919	32	77	31	0	0	0	0
4	1626887929	3	87	31	1	2	1	0
...	...	...	...	...	...	...	...	...
995	1629734639	32	78	31	1	0	0	0
996	1626887039	3	87	31	1	2	3	1
997	1626887020	3	90	31	1	0	3	0
998	1626886919	32	88	31	1	2	1	0
999	1626887039	3	90	31	1	1	1	0

1000 rows × 8 columns

Figure 13 IoT Pannipitiya dataset

Data Processing

Data collected from these sources are used to with multiple methods of analysis.

1. Mathematical based approach
2. Artificial Intelligence and machine learning based approach

Mathematical based approach

As a mathematical based there are several dominant data analysis techniques are used. There is serval analysis can be utilized such as Time Series analysis, Factor analysis and Regression analysis. Most suitable for use on identifying the correlation between a dependent variable and other independent variables factors it.

Model Solution

Water Level Prediction Model

Linear Regression Analysis provided dependent variable factor of fluctuations (water level normal, rising or falling) with accuracy levels correlating with the river alert level, water level before, and water level at the time, rainfall, water level classification.

Artificial Intelligence and machine learning based approach.

Using of Artificial intelligence and machine learning to predict variables and identify significant relationships with the variables in modern approach in the creation of models as a solution. Serval machine learning methodologies are identified by Support Vector Machine (SVM) Algorithm, Fuzzy logic, Random Forest, and Artificial Neural Network (ANN). For the suitable cases as classification and regression challenges using SVM and ANN algorithms are adequate.

The bellow Figure 14 Models Building Algorithm contains the flowchart of algorithm used in training the machine learning models.

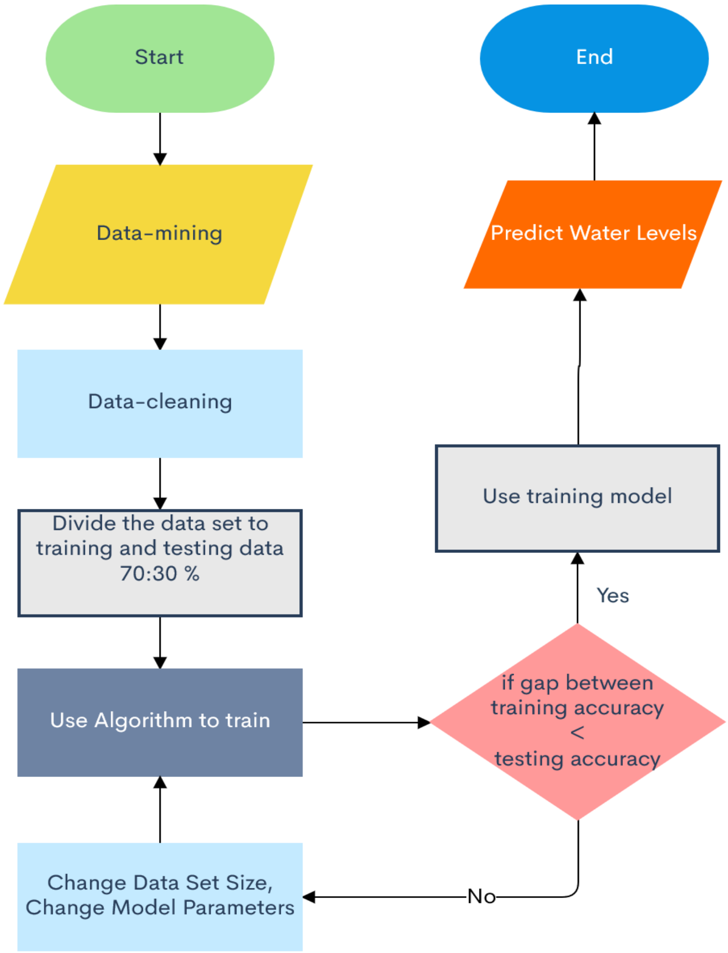


Figure 14 Models Building Algorithm

Support Vector Machines (SVM)

Support Vector Machines are a machine learning algorithm under supervised learning. And the plotted data in the dimensional space each number of features of values are in a particular coordinate. The SVM models are used in each 5 location data sets with attributes changed to identify contributions of the variables factor into main dependent variable.

For the accuracy of data, the data sets such as testing data and training data are normalized in to maintain the general distribution and ratio of the source data.

Artificial Neural Network (ANN)

Deep Learning approach of distinguishing other factors of dependent variable and classification in the use of utilizing multiple layers of combined neural layers with each other to provide a network of neurons. These networks of neurons are in hidden layer. As the input layer provides the inputs and output layer is number of dependent variables Figure 15 Artificial Neural Network Layers shows three inputs and two outputs.

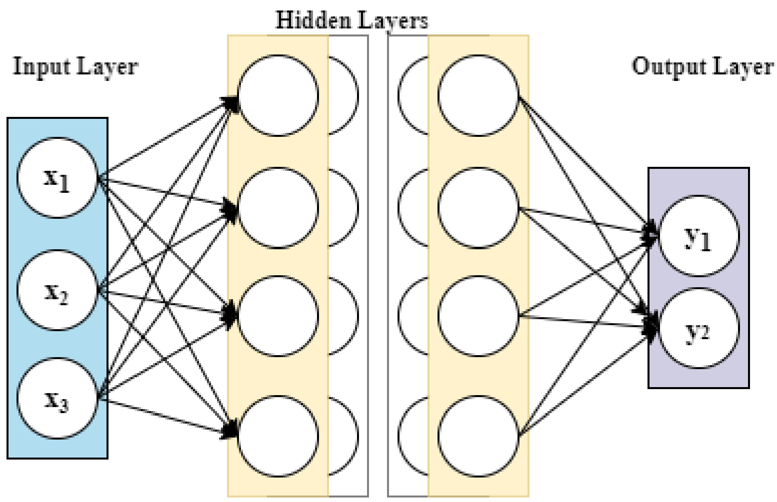


Figure 15 Artificial Neural Network Layers

As above Artificial network. The water model creation water level prediction initiated using TensorFlow and Keras Libraries.

Decision Making Model

Decision selection models are contained with decision chains. The data provided data has to calculate the operational possibilities of a decision. Most the systems as discussed in the 7.1 Literature Review as more knowledgebase centre expert system in use. The approach of gathering data for build an expert system is diagramed in Figure 16 Optimal Expert System Flow Diagram.

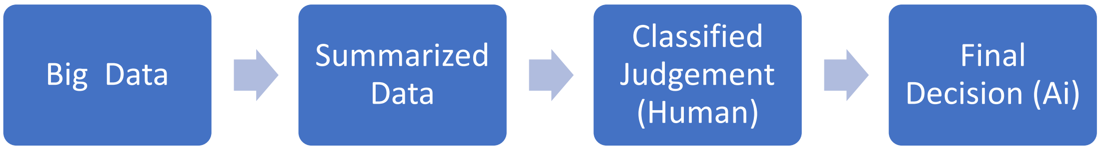


Figure 16 Optimal Expert System Flow Diagram

Decision Tree

The decision tree model is consisting of branches, edges, leaves. The model creation has two major steps induction and pruning. For this model it is utilizing the Gini index function because it is in the use of classification.

The data set used in this model creation is depicted in

Table 5 Decision Tree Model Processed Data Set Overview. Using Classification and Regression Tree (CART) to perform classification.

A data-driven water level prediction model is used to provide accurate water level prediction.

9.5 Software and Hardware Specifications

9.5.1 Software boundaries

- PyCharm

  It is a popular IDE for computer programming that is specifically designed for the Python programming language. JetBrains, a Czech business, created and delivered it. Graphic debuggers, code analysis, testing, and version control have all been improved. It is cross-platform, featuring versions for Windows, Mac OS X, and Linux.

- Flask

  The server-side calls in this application are built with Flask. The developed model was put into the flask framework to generate REST API calls.

- scikit-learn

  This is where the logistic regression and SVM algorithm is imported in this project. Techniques for classification, regression, and clustering are all included in this collection.

- Google's Colab

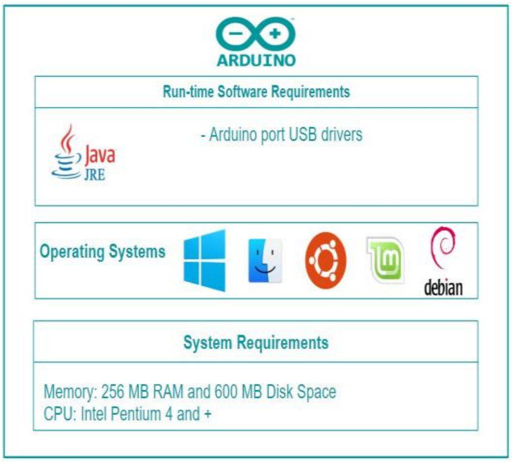
  Colab is a cloud-based notebook, like Jupyter Notebook. Unlike the jupyter notebook, you do not need to install any dependencies. The colab had all the machine learning libraries installed. All that's left to do now is import and start using the relevant libraries.

- Android Studio

  The proposed system includes an android mobile application. The mobile application plays a major role when it comes to crowd sourcing. The weather data is gathered from public crowd through the mobile application. Usability and user friendliness will be concerned more when developing the mobile application. Android studio IDE will be used as the development platform and java as the programming language.

- Arduino IDE

  This is an open-source cross application which could be used in Windows, MacOS, Linux which functions are written in C and C++.This IDE has the capability to upload the written codes to any Arduino compatible boards with 3rd party cores and libraries.



- Altium designer

  Altium designer is a software which is used to design printed circuit boards (PCB) for electronical devices. We have used this software to design a PCB board for the IoT device which will also give a long durability and stability to the IoT device.

  System requirements

  1. Intel® core™ i7 processor or higher
  2. 16GB RAM
  3. 10GB hard disk space
  4. High performance graphics card (supporting DirectX 10 or better), such as GeForce gtx 1060/Radeon rx 470
  5. Adobe® reader® (version xi or later for 3d pdf viewing)

- Database Handling

  The generated data will be stored in a database to retrieve the data when it is needed to display to end-users. Firebase will be used store these crowdsourcing data since interacting with public will generate lots of data. Firebase is used as the main data storage location

- GitLab

  GitLab is a web-based git repository management system that enables users to track the development lifecycle. GitLab is used as the main repository location of the project.

- Colab: Google Colaboratory

  Colab notebooks enables to merge executable code and rich text in a single document, as well as graphics, HTML, LaTeX, and other formats with running on the Jupyter Notebook environment. The tool is offered by Google and hosted on Cloud.

9.8.2 Hardware Specifications

Backend development process

- RAM : 16 GB RAM
- CPU : Any latest CPU
- Disk Space : 10 GB and another 1 GB for caches
- Monitor Resolution : 1024x768

Requirements to execute this mobile application in a desktop

- Operating System : Windows 07, Windows 10
- CPU : Any latest CPU
- Software : Android Studio

Requirements to execute this web application in a desktop

- Operating System : Windows 07, Windows 10
- CPU : Any latest CPU
- Software : Browsers

8.8.3 Communication Specifications

Weather app needs to identify users' current location for weather forecasting, Gmail, and Facebook authentication for login. Therefore, mobile application should be connected to Wi-Fi or mobile data to access these functions.

9.6 Designs

Designing of the Rainfall prediction Component

In the proposed rainfall prediction system, machine learning methods are used to create prediction models. This is the simplest method for entering data from the input form. By developing location-based rainfall prediction models, one of the key goals of this study topic is to increase the accuracy of predicting rainfall and rain range.

- Input weather data from.
- Prediction page of Rainy day.
- Prediction page of Sunny day.

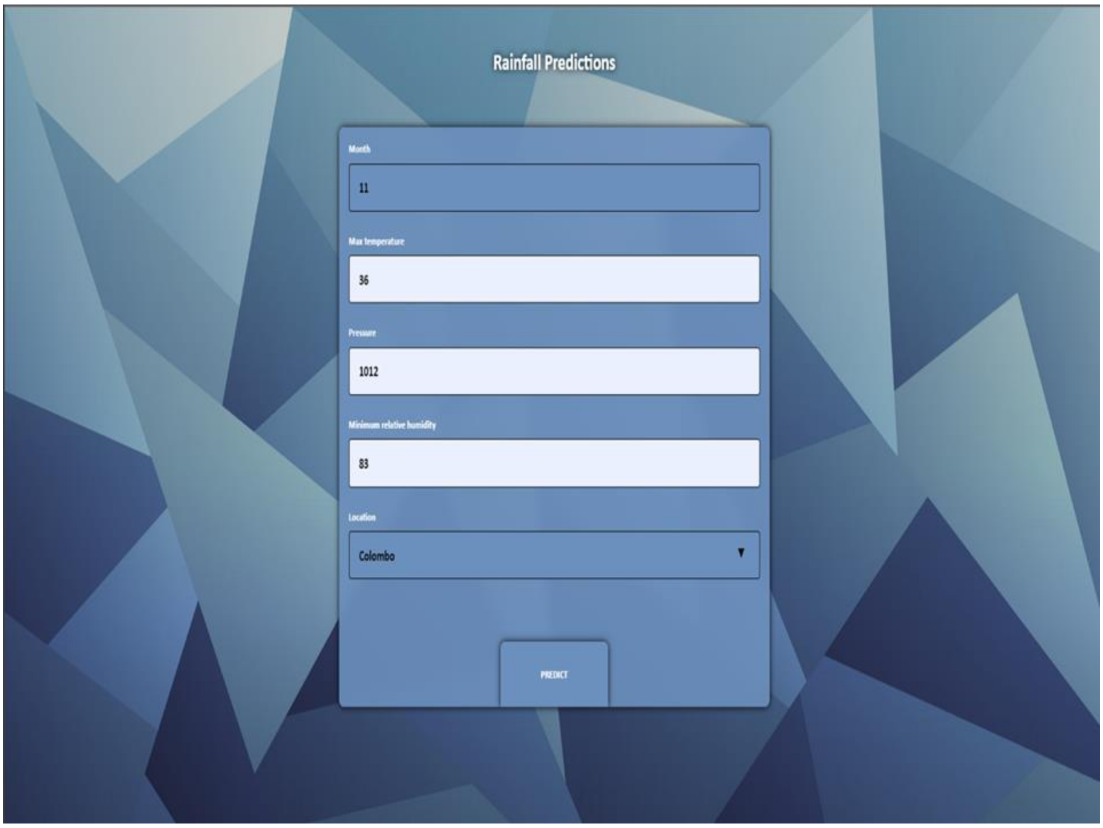


Figure 17:Weather Input Data Form

Figure 19:Prediction Page(Rainy Day)

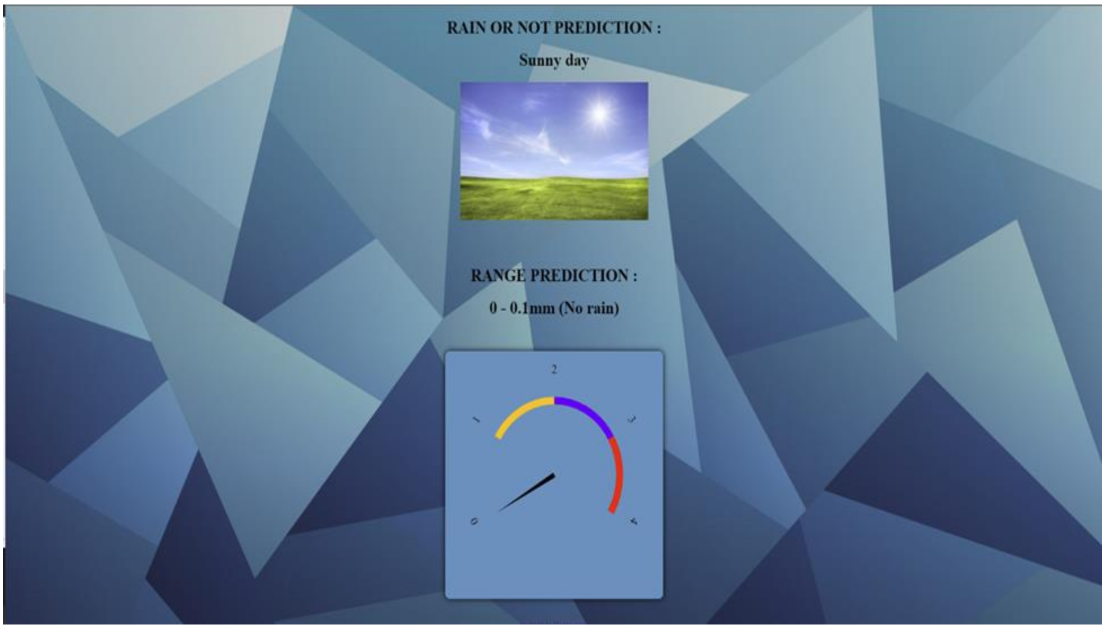


Figure 18:Prediction Page(Sunny Day)

The Kalu River Basin Water Level Prediction is Displayed in this user interface.

For each selected station, the flood classification prediction will be displayed as in Figure 20 Station Prediction Classification Dashboard Values.

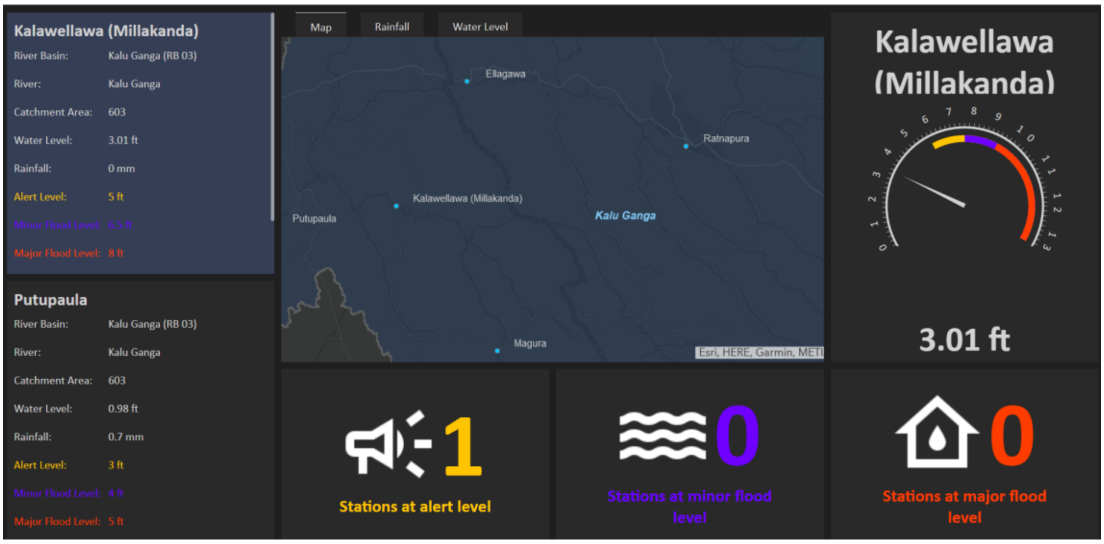


Figure 20 Station Prediction Classification Dashboard Values

Designing of the Crowdsourcing Component

The mobile application plays a major in interacting with volunteers to obtain crowdsourcing data. This section consists with sample wireframe designs which are used in design process in the development of mobile application.

Figure 22 Login UI Design

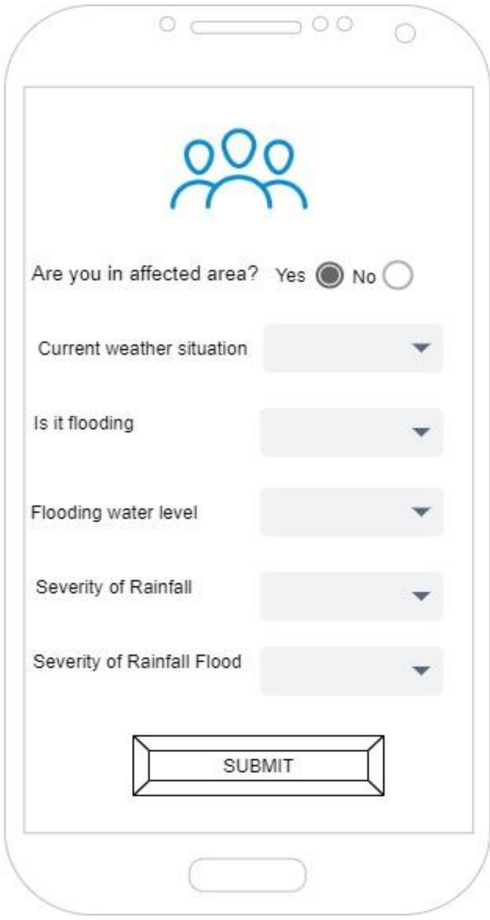


Figure 21 Crowdsource UI Design

Designing of the IoT Device and data preview

The IoT device will be performing a major role in this project as in the users will be receiving live weather data through the Web application or the Mobile application with the help of the IoT devices positioned in various locations. These IoT devices will be having a GPS tracker assembled in the device which will help to have live location tracking together with weather information too.

The IoT device was designed on a PCB board to ensure long time durability, high performance, scalability and to have precise structure of the device.

Below shows the schematic design, PCB Layout, and the 3D view of the device of the designed IoT device.

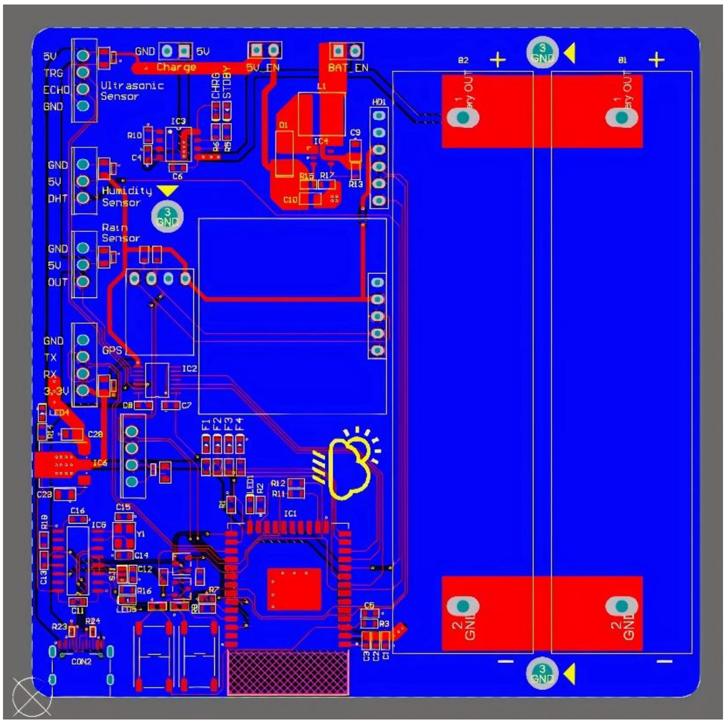


Figure 23: PCB Layout of the IoT Device

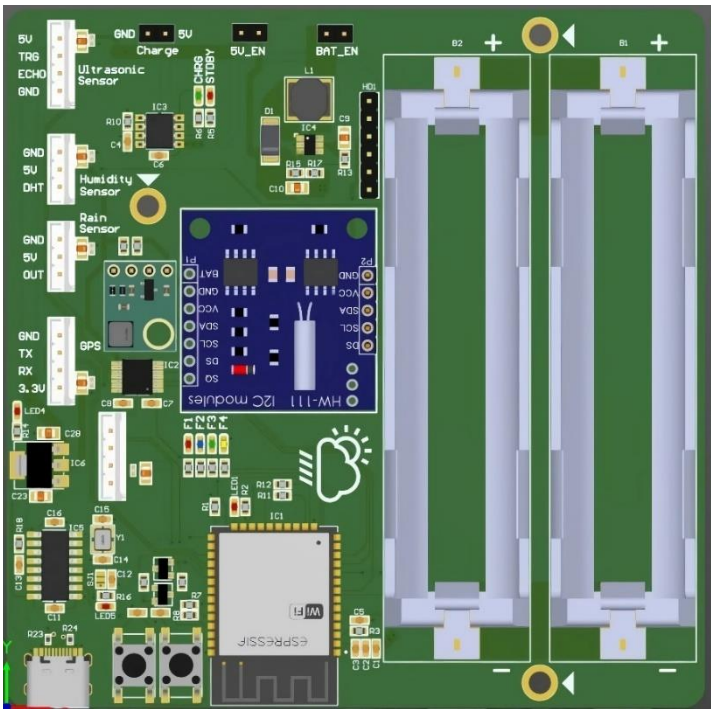


Figure 24: 3D view of the Designed PCB Board

The mobile application will have a dashboard to preview all the weather data factors (Temperature, Humidity, Rainfall intensity, Water level increase/decrease) gathered from the IoT device together with a graph view.

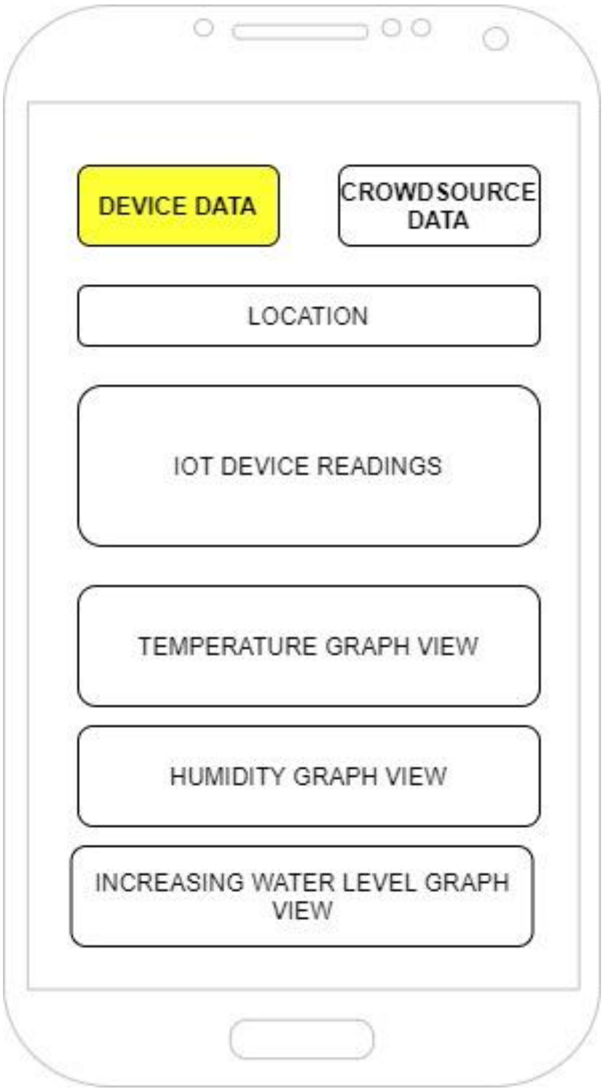


Figure 25: IoT Readings Dashboard Wireframe

The web application will also have a similar design to preview the IoT sensed weather data which will be beneficiary for both state and non-state members.

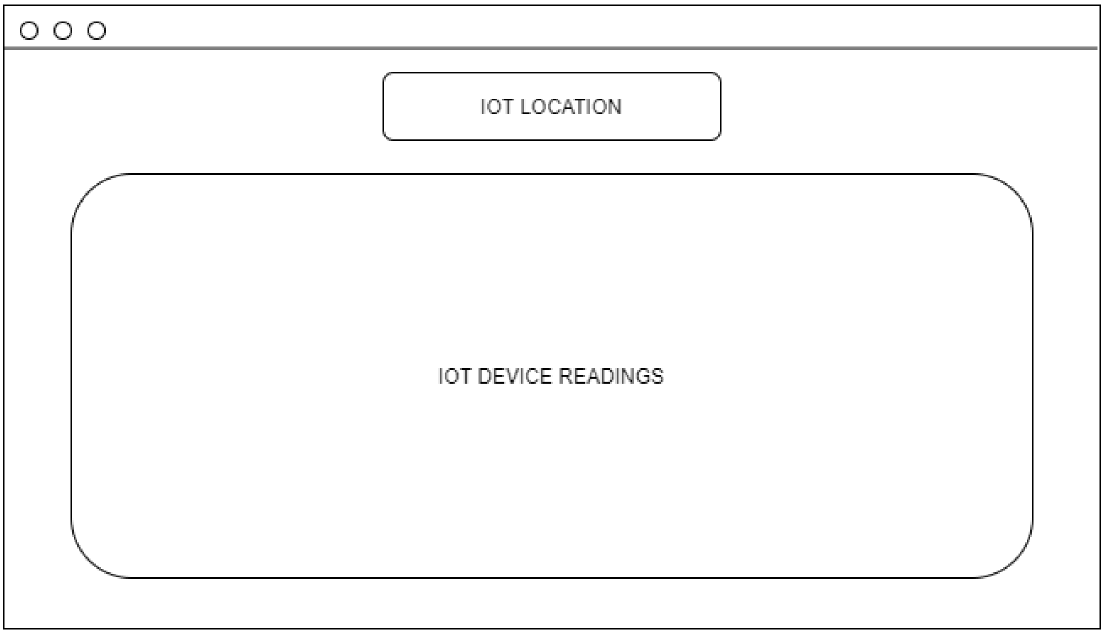


Figure 26: Web Application IoT device data preview wireframe

For the users who have is not a stakeholder of our system(Mobile app/ Web app) is still able to receive weather information upon users' request's location wise. Below is sample method on how they we receive weather data information once the users send a request to our system via SMS.

9.7 Commercialization

This risk management software is developed as a private initiative at a cost to the developers. This cost is considered as capital employed in a business venture. Therefore, the return to the developers depends on the extent of commercial operations of the venture. Unlike in many commercial ventures that involve in production of goods and services for profit margins in markets, this software development is a social responsibility project that requires the engagement and management by the state authorities. Therefore, its commercialization involves in contracting with the relevant state authorities responsible for flood risk management to take ownership of the project or risk management system.

This can take two forms. First, the software is sold to the state authorities at a price with a service maintenance period at a fee as agreed. Accordingly, the rest of the risk management is the responsibility of the authorities. Second, the software ownership along with the patent rights is retained with the developers while an annual service price is charged for utilization. As such, the developers become software contractor to the risk management system. In either case, the developers should be able to recover the capital with a competitive rate of return on capital within a short period calculated on the basis of investment appraisals such as internal rate of return (IRR) and pay-back periods. The government will benefit by effectively managing flood risk and minimizing the damaged to people and properties that politically detrimental to the ruling parties. As such, commercialization requires skills in negotiation for selling the product to the state and the society.

9.8 Testing and Implementation

9.8.1 Testing

Software testing is a method for evaluating the functionality of software is matching its specified requirements and to check whether the software is error free. software testing is an essential part in a development process. Software testing involves using manual tools and automated tools for testing. The main goal of software testing is to identify errors, bugs, software gaps or missing software requirements. There are series of levels in software testing. Since this an early warning system for pre and post flood risk management, it is highly encouraged to develop the application with minimum number of errors. As a result of lack of software testing, the entire mobile application, web application and IoT device will fail to meet its specified requirements. Therefore, there is a need of executing software testing strategies to keep functioning the system without any interruptions.

Sample test cases for Rainfall prediction system

Test case ID	001
Test case scenario	Validate input field
Test steps	a. Users navigate to prediction page b. Fill required fields c. Submit
Test data	data
Expected result	Re-direct to the prediction page screen
Actual result	As expected,
Pass/Fail	Pass

Test case ID	002
Test case scenario	Validate input field
Test steps	a. Users navigate to prediction page b. Fill required fields expect month field c. Submit
Test data	data
Expected result	Re-direct to the login screen
Actual result	As expected
Pass/Fail	Pass

Test case ID	003
Test case scenario	Validate input field
Test steps	a. Users navigate to prediction page b. Fill required fields expect max temperature c. Submit
Test data	data
Expected result	Re-direct to the login screen
Actual result	As expected
Pass/Fail	Pass

Test case ID	004
Test case scenario	Validate input field
Test steps	a. Users navigate to prediction page b. Fill required fields expect minimum relative humidity c. Submit
Test data	data
Expected result	Re-direct to the login screen
Actual result	As expected
Pass/Fail	Pass

Sample test cases of the mobile application

Test case ID	01
Test case scenario	User Registration
Test steps	a. User navigate to the Registration UI b. User enters incorrect email c. Sign up
Test data	User data
Expected result	Please enter a valid email address
Actual result	As expected
Pass/Fail	Pass

Test case ID	02
Test case scenario	User Login
Test steps	a. User navigate to the Login UI b. User enters email and keep password empty c. Log in
Test data	User data
Expected result	Please enter a valid password
Actual result	As expected
Pass/Fail	Pass

Test case ID	03
Test case scenario	Weather API
Test steps	a. User navigate to the weather forecast UI b. User enters location as Malabe
Test data	Live weather data
Expected result	Successfully retrieved weather information in Malabe
Actual result	As expected
Pass/Fail	Pass

Test case ID	04
Test case scenario	Weather API
Test steps	a. User navigate to the weather forecast UI b. User retrieve current location weather data
Test data	Live weather data
Expected result	Successfully retrieved weather information in user current location
Actual result	As expected
Pass/Fail	Pass

Test case ID	05
Test case scenario	Weather API
Test steps	a. User navigate to the weather forecast UI b. User has disabled the location service
Test data	Live weather data
Expected result	Please enable the GPS location function
Actual result	As expected
Pass/Fail	Pass

Test case ID	06
Test case scenario	Crowdsource Data Submission
Test steps	a. User navigate to Crowdsource UI b. User enters whether information c. Submit information
Test data	Crowdsource data
Expected result	Submitted successfully and database has been updated
Actual result	As expected
Pass/Fail	Pass

Test case ID	07
Test case scenario	Crowdsource Data Retrieve
Test steps	a. User navigate to the dashboard UI b. User enters location as Malabe
Test data	Crowdsource data
Expected result	Successfully retrieved validated crowdsource weather information in Malabe
Actual result	As expected
Pass/Fail	Pass

Sample test cases of the IoT Device

Test ID	**TC-A01**
Test case scenario	Checking if the IoT device is capable of synchronizing data in regular intervals
Entry Criteria	1. Minimum of 2 IoT devices needed 2. The device should be assembled in a method where is could be transfer data in a regular interval.
Test Procedure	1. IoT Device 1 & IoT Device 2 enlists to the network and a establishes a connection. 2. Check the DB if the data readings are transmitted within a regular interval. 3. Step 1 should be done repeatedly to ensure the connectivity to the network and DB is successful and data is passed within regular intervals.

Exit Criteria (pass)	IoT device 1 & 2 will send their network connection requests & DB connection request within a short time and transmit data to the DB in regular intervals.

Test ID	**TC-A02**
Test case scenario	Checking the accuracy of the data readings of the IoT device in long and short intervals.
Entry Criteria	1. Fully assembled and functional IoT device. 2. IoT device being connected to the DB.
Test Procedure	1. Set the IoT device reading to the interval to 5 minutes. 2. IoT device connects to the network and establishes a connection to the DB. 3. Check the DB if the data readings are transmitted within given interval (5 minutes).
Exit Criteria (Fail)	Reading from the IoT device becomes irregular after a certain time.

Test ID	**TC-A0**
Test case scenario	Checking the accuracy of the data readings of the IoT device in long and short intervals.
Entry Criteria	1. Fully assembled and functional IoT device. 2. IoT device being connected to the DB.
Test Procedure	1. Set the IoT device reading to the interval to 1 minute. 2. IoT device connects to the network and establishes a connection to the DB. 3. Check the DB if the data readings are transmitted within given interval (1 minutes).
Exit Criteria (Pass)	IoT device data transmission to the DB stays constant.

Machine Leaning models testing

Software testing is a method of determining the accuracy of software by taking into account all of its characteristics (reliability, scalability, portability, reusability, and usability) along with reviewing the operation of software components to detect bugs, errors, and faults. Software testing has the ability to identify any errors and weaknesses during development.

Various types of testing enable us to detect flaws that are only noticeable during runtime. Using testing components summarized as Unit testing, Regression testing and Integration testing.

The machine learning models cannot be tested as a software regular testing. In the machine learning algorithms, the testing of the components includes two main data sets other than the training set.

1. Validation Set

   For hyperparameter tuning, having merely a training set and a testing set is insufficient. And that can result in overfitting. To avoid that, you can select a small validation data set to evaluate a model. Only after you get maximum accuracy on the validation set, you make the testing set come into the game.

2. Testing Test

   For assurance of the models guaranteed real world implications, choose samples for a testing set from training set instances that the machine has used in training.

   It is critical to remain objective during the selection process and to draw samples at random. Also, avoid training on testing data by not using the same set several times.The diagram bellow Figure 27 Data Sets using for Model Validation show the process of validating the model.

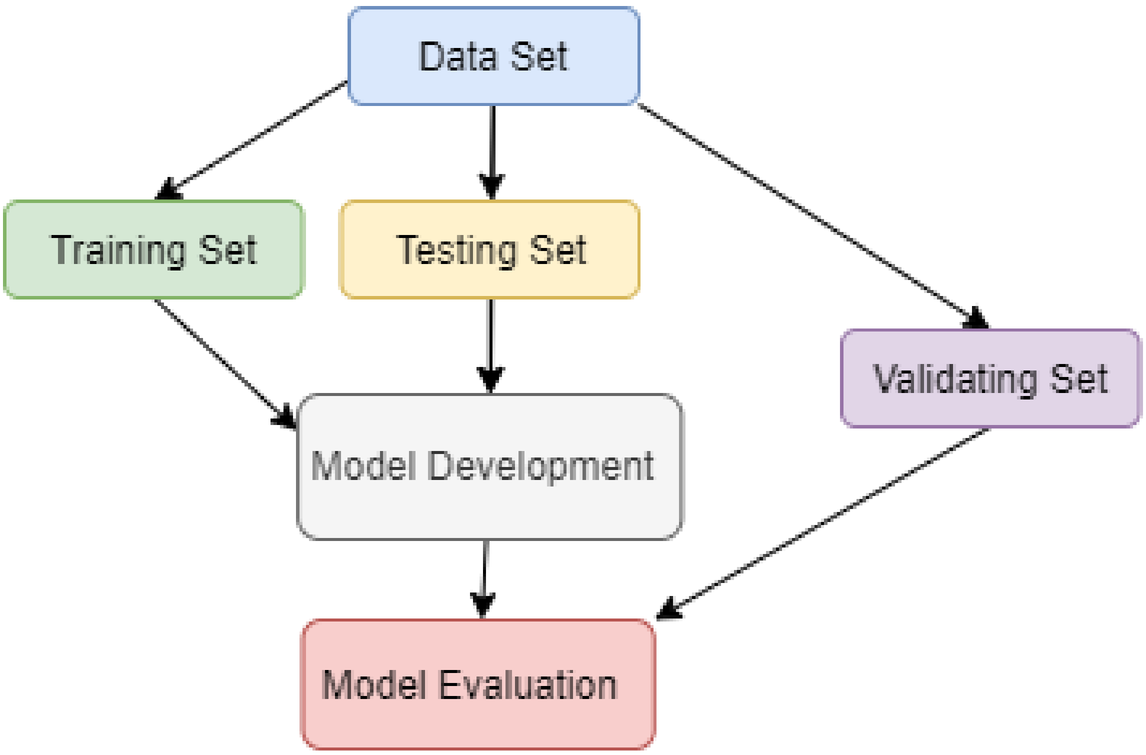


Figure 27 Data Sets using for Model Validation

There are other model testing approaches,

1. Pre-Train testing

   Testing the data sets labels, and missing labels and missing values.

2. Post-Train testing

   Checking the performance of trained models and check it performs accordingly. As an example, how much the classification of the models results. And minimal functionality testing of the model.

9.8.2 Implementation

Rainfall Prediction Component Implementation

The proposed framework was developed as a web application. This rainfall prediction system is where we input weather data manually. The model predicts whether there will be rainfall or not, and the rainfall amount ranges. Individuals can enter relevant weather data and receive a prediction. This research mainly considers two main data sets as the daily basis for predict rainfall. Such as

Data set 1

1. Temperature maximum
2. Relative humidity minimum
3. Sea level pressure Average

Data set 2

1. Dataset 1
2. Time (Month)
3. Locations (Colombo(Coastal and City), Vavuniya (Countryside), and Katugastota(Hills)

Data set one is the main factor that is causing rainfall and that is already being used in exciting research. Based on location, data set two, data set one, and replacement factor time (month) is used. This weather historical data for the past 5 years was collected from the metrology department of Sri Lanka.

Train the Model

This is a crucial stage of the study because the results of this phase have a big impact on the system's output. During this technique, we must divide the dataset into two pieces. Analyzed data is split into 20% test data and 80% trained data used in the trained model. Accuracy checks with each attribute change and understands the attribute's contribution. The right answer must be included in the training data, often known as an aim or target attribute. The learning algorithm looks for patterns in the training data that map the input data attributes to the goal and then produces a machine learning model that captures these patterns. We used two machine learning algorithms to train the model. As an example,

- Logistic Regression(LR)
- Support Vector Machine(SVM)

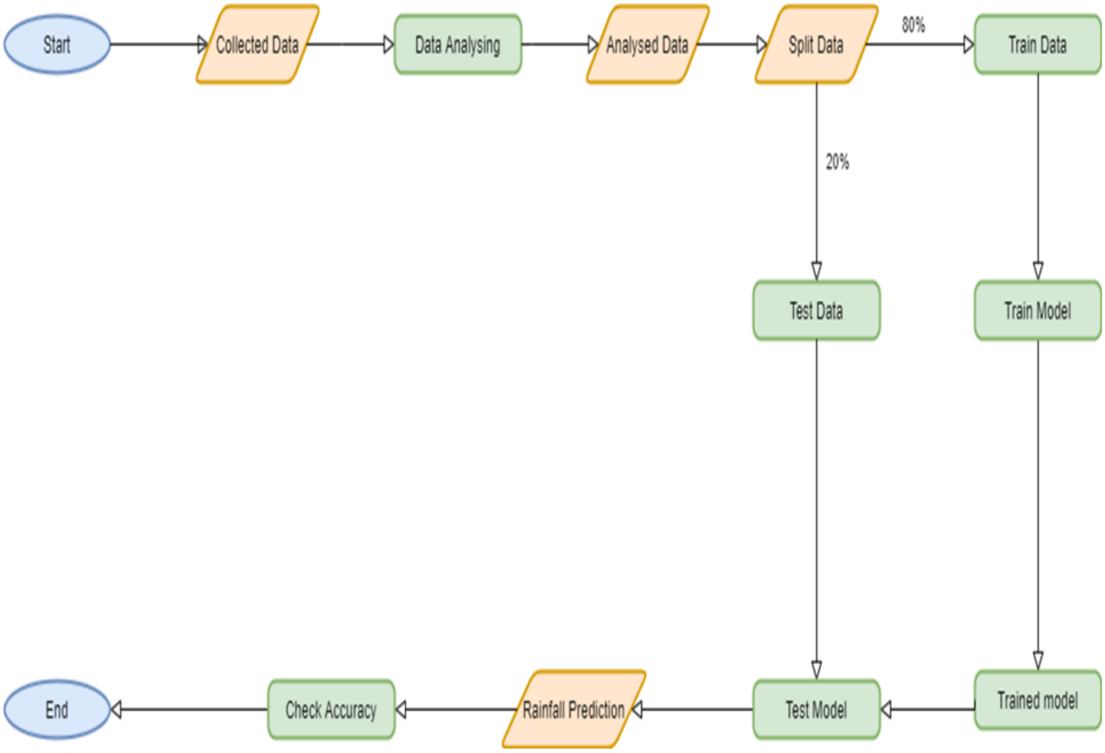


Figure 28:Model Training

Logistic Regression

The technique of logistic regression is one of the ways of doing a regression analysis on binary input parameters. The sigmoidal function, which is at the basis of the entire approach, is the focus of the logistic regression function represented in Fig 9. It uses an S-shaped format to map out the features of any given input data between 0 and 1. The LR model is a linear one that employs a sigmoid function, as indicated in equation (1). This equation is used to undertake classifications with results ranging from 0 to 1, giving the LR the capacity to do the probabilistic interpretation. In this proposed system, logistic regression is used to predict for binary classification whether it will rain or not.

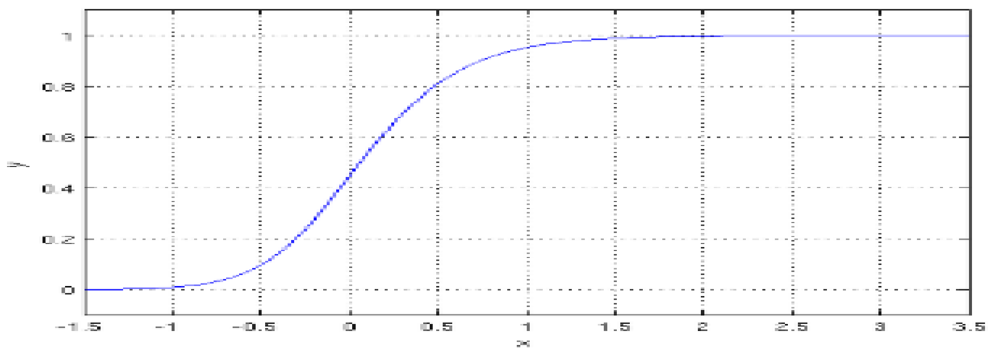


Figure 29:Logistic Function

$$F(x) = \frac{1}{(1 - e^{x})}$$

1:Sigmoid Function

In most cases, the regression coefficients are evaluated using maximum probability estimation. Most of the writers used statistical models to predict rainfall, such as univariate or multivariate binary logistic regression.

Support Vector Machine

The SVM model was created in 1995 by Cortes and Vapnik. The SVM model is a well-known model that uses a hyperplane to distinguish two classes. The best element of the SVM model is the kernel function, which converts primary input into high-dimensional data and then finds a hyperplane to separate the groups. As a result, SVM is a useful technique for categorization. Furthermore, bit determination has a major impact on categorization execution. It's difficult to decipher the RBF and other non-linear kernels. The linear kernel does not provide a great prediction for non-separable datasets, either.

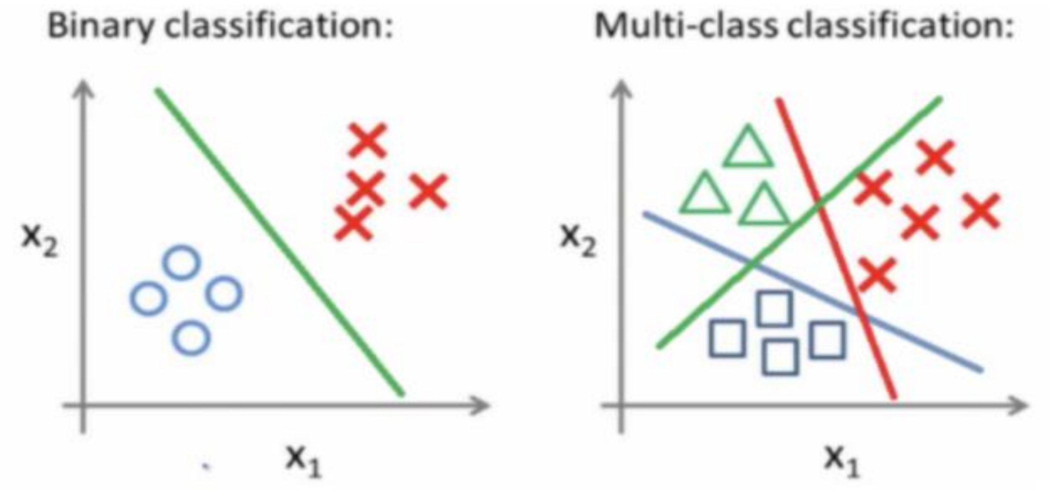


Figure 30:SVM Binary & Multi classification

In this system, SVM is used to predict for binary classification “rain or not” behavior and for multi-classification divide, the rain range into five categories (No Rain (0.1mm), Drizzle (0.1-10mm), Normal (10-20) mm), Strong (20-30) mm), and Heavy (30+ mm)).

Rainfall Prediction logical view

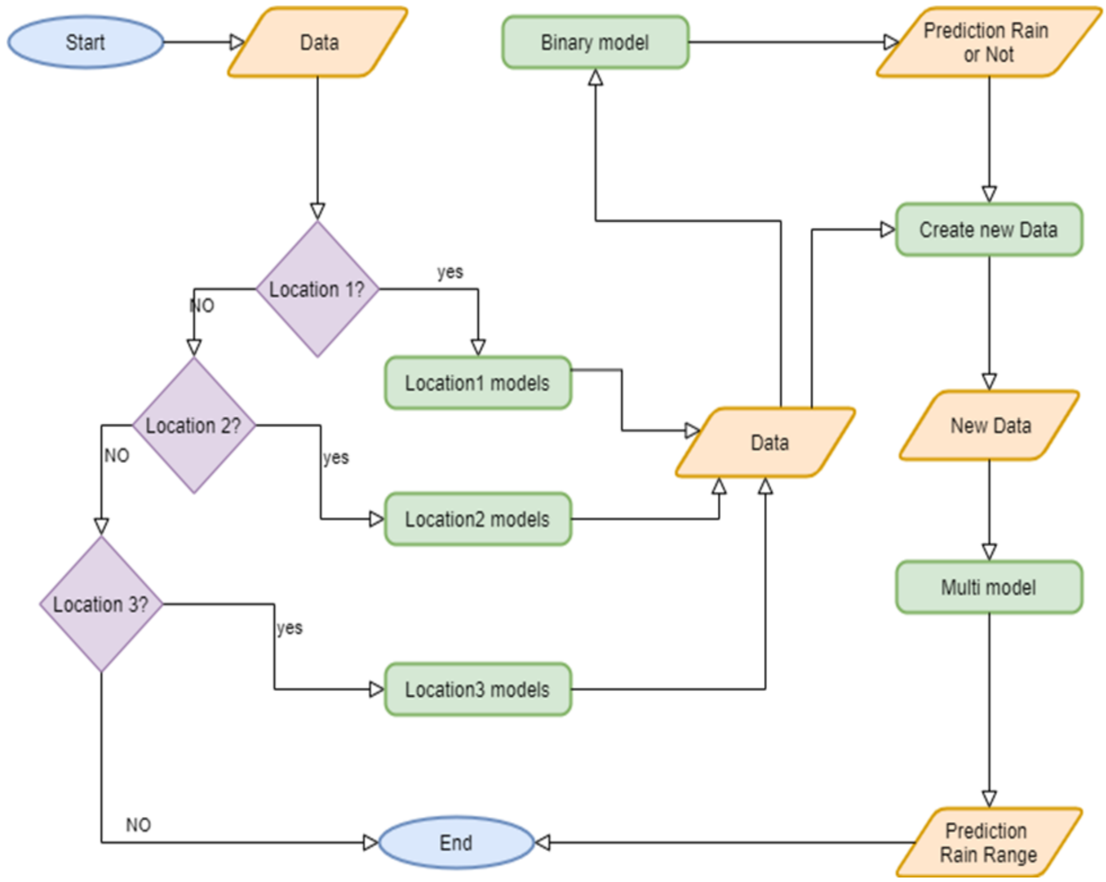


Figure 31:Prediction Flow Chart

The final prediction system selects a prediction model based on location. Once the model has selected the input data to use to predict the binary classification "Rain or Not", This output adds to input data and new input data will be created. new data used as input for the multi-classification "Rain Range"

In this final rainfall prediction model, binary classification model output (rain or not) is used as input in the multi-classification model to increase the accuracy of multi-classification. The final Rainfall prediction system makes a prediction based on IoT data and a prediction model for specific IoT device locations.

Flood forecasting Component Implementation

The proposed model was created to be providing predicated classification as the daily records get updated.

The created model has been implemented as an API to provide predictions in Figure 32 API Demo Request.

Other API developed are attached in APPENDICES.

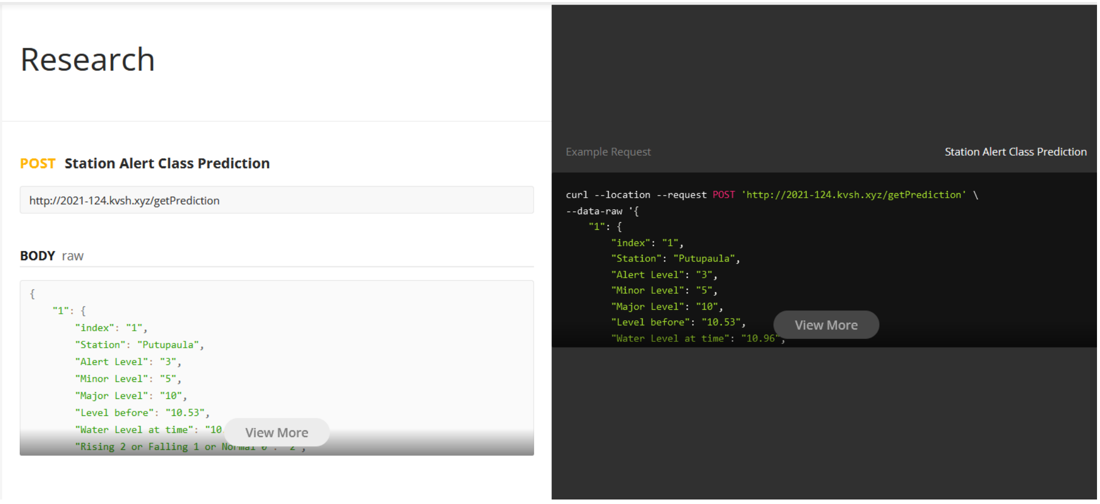


Figure 32 API Demo Request

The output of the model is displayed as bellow in Figure 33 Forecast Prediction Response.

Figure 33 Forecast Prediction Response

The Table 6 Classification levels are displayed of the classification values and levels.

Table 6 Classification levels

Classification	Value
Normal	1
Alert Level	2
Minor Level	3
Major Level	4

The currently collected data are on firebase database so it can utilize in the future development and dashboarding the predictions.

Dashboard view of the Water Level Reading, This User Interface provides the information of water level in meters.

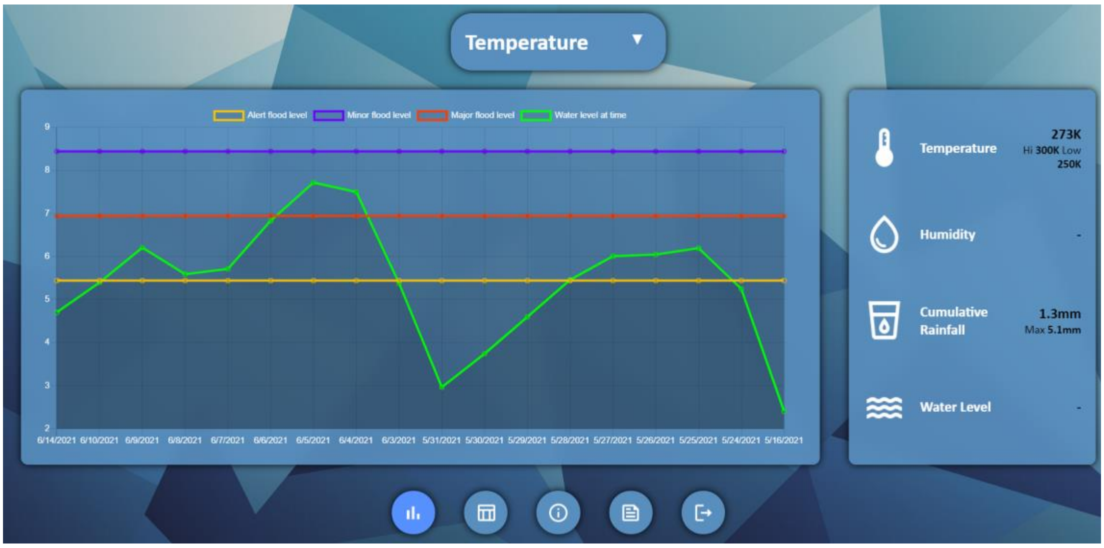


Figure 34 Water level Dashboard View

For each selected station, the flood classification prediction will be displayed as in Figure 20 Station Prediction Classification Dashboard Values.

Crowdsourcing Component Implementation

The mobile application is developed using android studio and the volunteers will use the mobile application to submit weather information. Figures below are the sample screenshots of the implemented mobile application using above mentioned methodologies.

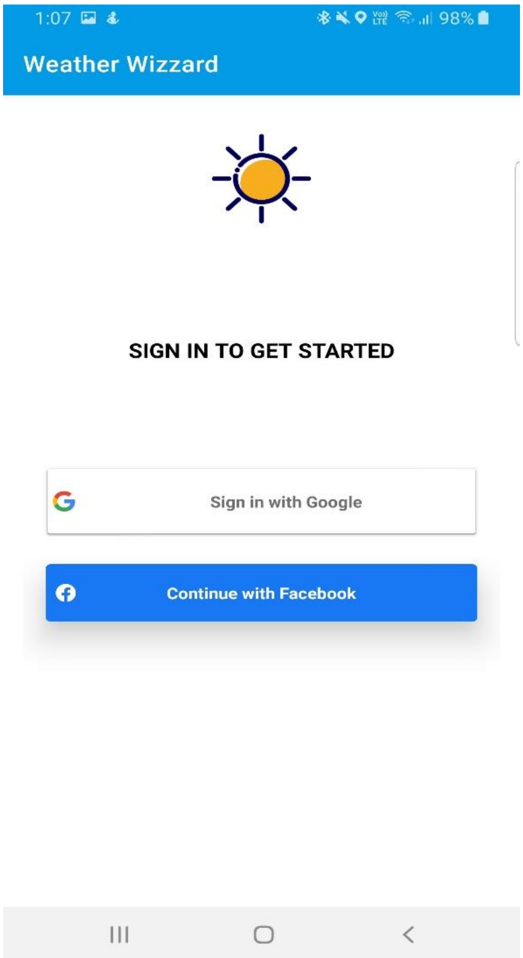


Figure 36 Login UI

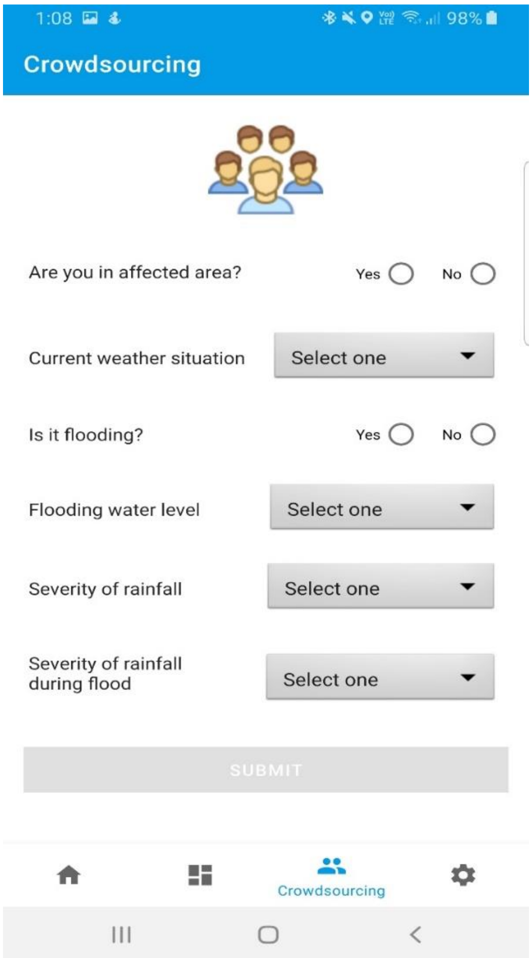


Figure 35 Crowdsource UI

IoT device and the implementation

The IoT device sketch was initially designed by using an online tool which helped us to have a clear idea on the wiring system the IoT device will have to go through, for the sensors to functions properly.

Below is the sketch circuit design.

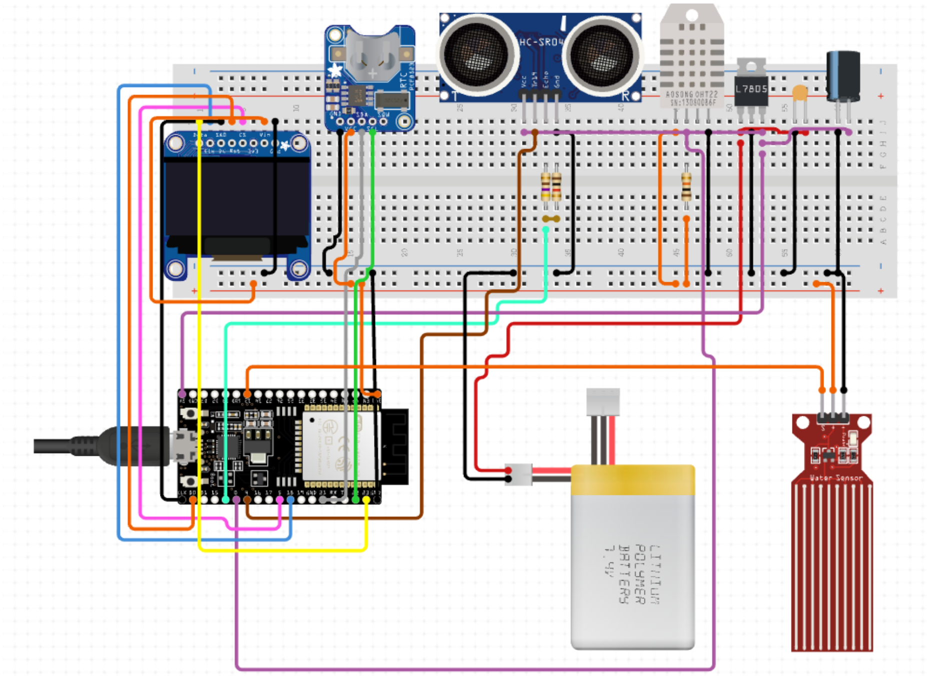


Figure 37: IoT device circuit design

To ensure a long use and a better accuracy and scalability of the device we have designed a PCB board for the IoT device whereas all the sensors, Power supply etc. Is directly embedded to the board. By having such precise structure on the device, the durability and scalability of the device will also increase and have a better function in the device.

Below is a schematic design of the designed IoT device

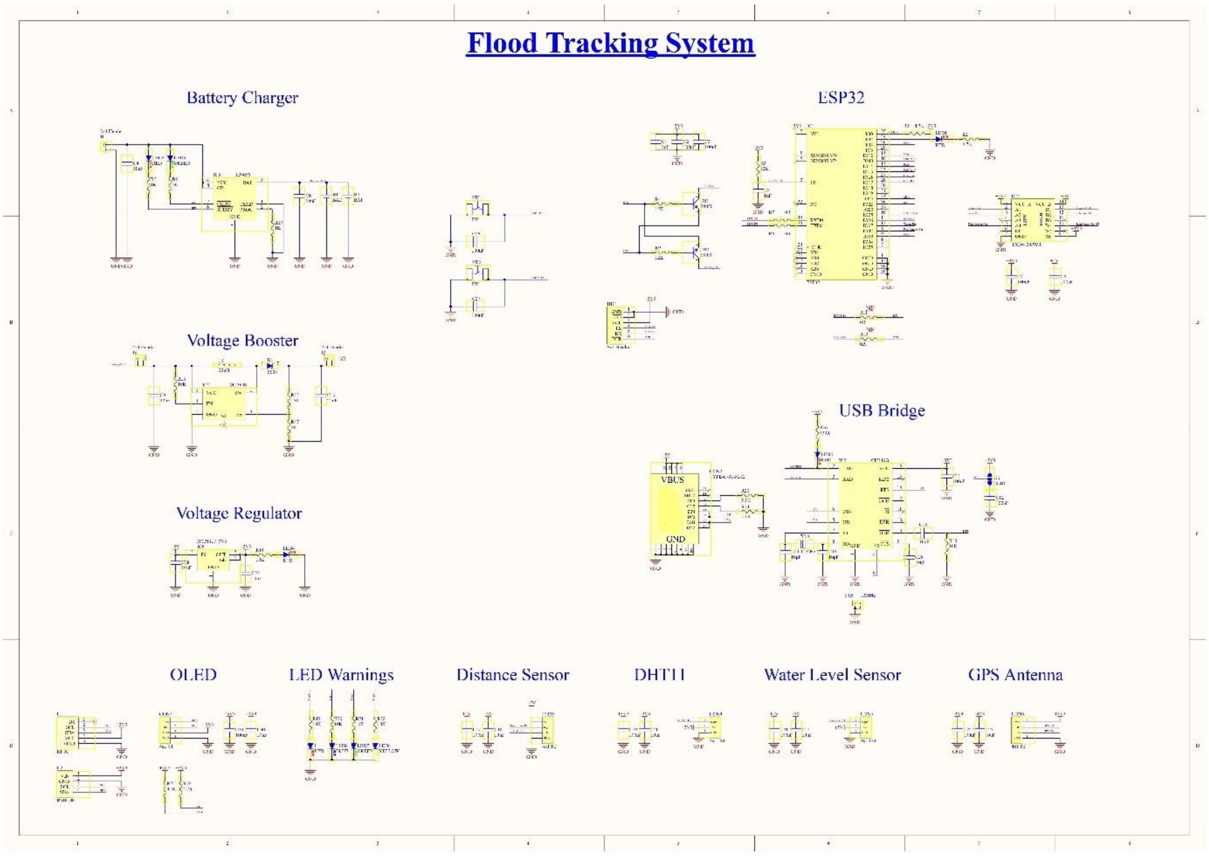


Figure 38: Schematic Design of the IoT Device

10 Results and Discussion

10.1 Results

Rainfall prediction Results

Models Accuracy Comparison with Data Set Changes - 1

Figure 19 shows the training and testing accuracy comparison between the Logistic Regression binary classification models for each dataset. The first graph shows the comparison of the accuracy of the combined data (Temperature maximum, relative humidity minimum, month) set without a specific location. The second graph shows the comparison of the accuracy of the dataset (Temperature maximum, relative humidity minimum, month, and station). The third graph shows the comparison of the accuracy of the dataset (Temperature maximum, Relative humidity minimum, month) with the specific location of Colombo. The fourth graph shows the comparison of the accuracy of the dataset (Temperature maximum, Relative humidity minimum, month) with the specific location of Katugastota. The fifth graph shows the comparison of the accuracy of the dataset (Temperature maximum, Relative humidity minimum, month) with the specific location of Vavuniya.

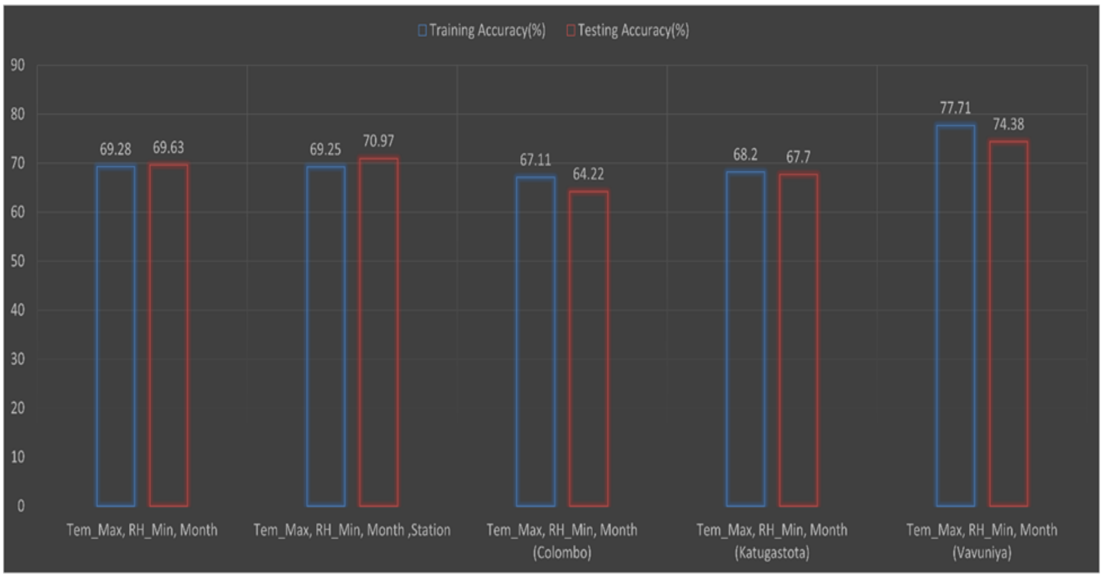


Figure 39:Models Accuracy Comparison with Data Set Changes – 1

3.1.2 Model Accuracy Comparison with Data Set Changes - 2

Figure 18 shows the training and testing accuracy comparison between the Logistic Regression binary classification models for each dataset. The three graphs show the comparison of the accuracy of the combined data (Temperature maximum, relative humidity minimum, month, and sea level pressure) set with specific locations in Vavuniya, Katugastota, and Colombo. The fourth and fifth graphs show the comparison of the accuracy of the dataset (Temperature maximum, relative humidity minimum, month, sea level pressure) and the dataset (Temperature maximum, relative humidity minimum, month, sea level pressure) for a specific location in Vavuniya.

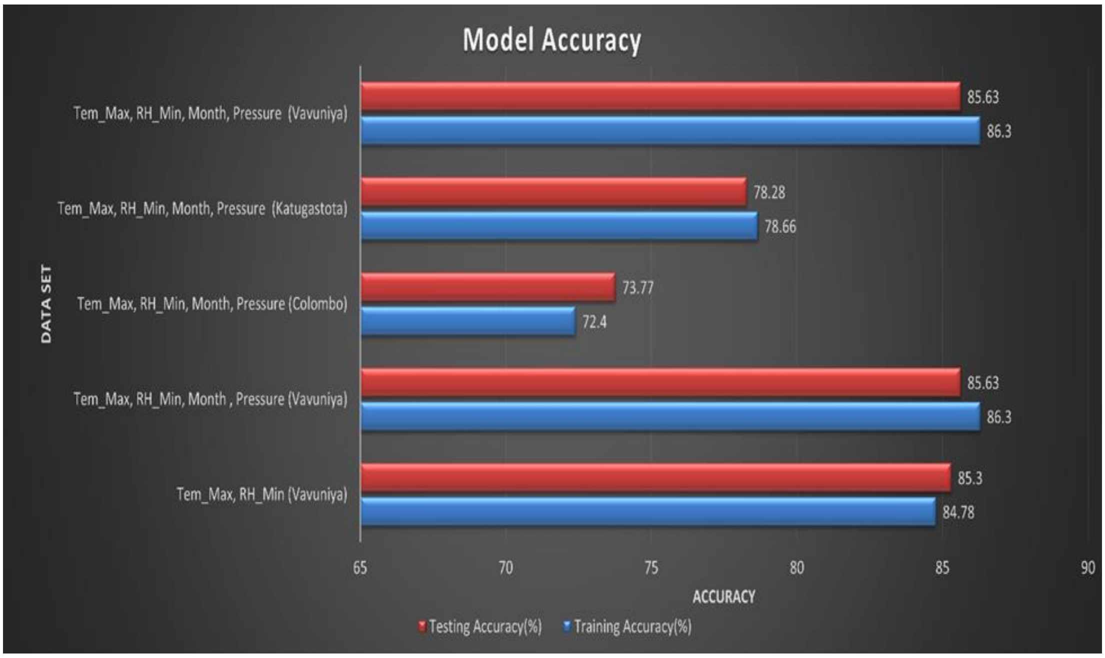


Figure 40:Model Accuracy Comparison with Data Set Changes - 2

3.1.3 All Models Accuracy Comparison

Figure 17 shows an accuracy comparison between the Logistic Regression (LR) binary classification model, Support Vector Machine (SVM) binary classification model, Support Vector Machine (SVM) multi-classification model, Auto-ML best model (Random Forest) binary classification, and Auto-ML best model (Random Forest) multi-classification for the dataset (Temperature maximum, relative humidity minimum, month, and sea level pressure) for specific locations in Colombo, Katugastota, and Vavuniya.

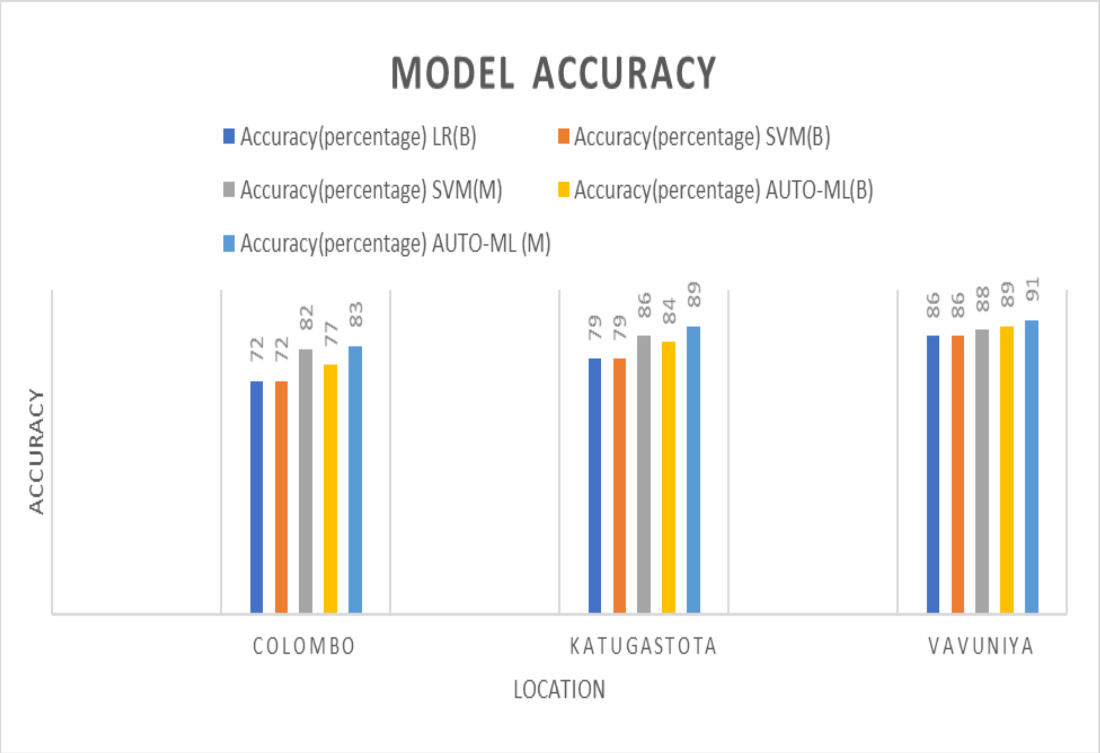


Figure 41:All Models Accuracy Comparison

Water Level Classification Model

The created linear regression model was evaluated using mean squared error, which in results.

The contains utilization of python SKLEARN and NUMPY libraries for the calculation of accuracy of liner regression model Figure 42 Python SKLEARN to calculate Mean Squad Error Image.

```
In [6]: linear = linear_model.LinearRegression()

        linear.fit(x_train, y_train)
        acc = linear.score(x_test, y_test)
        print(acc)

        0.79320129057141306

In [7]: print('Coefficient : \n', linear.coef_)
        print('Intercept: \n', linear.intercept_)

        predictions = linear.predict(x_test)

        Coefficient :
         [[  0.83113332   0.46795598  -0.32625304]
         [  0.42533608  -1.16877498   1.22473926]
         [ -0.83744845 -77.75884296  78.60561722]]
        Intercept:
         [-3.33659835 -2.00999089 14.64257374]

In [8]: mse = metrics.mean_squared_error(y_test, predictions)

In [9]: print("Mean Squared Error {}".format(mse))

        Mean Squared Error 0.3134568743884474
```

Figure 42 Python SKLEARN to calculate Mean Squad Error Image

Simple Liner Regression (SLR) Water Model, Regression model.

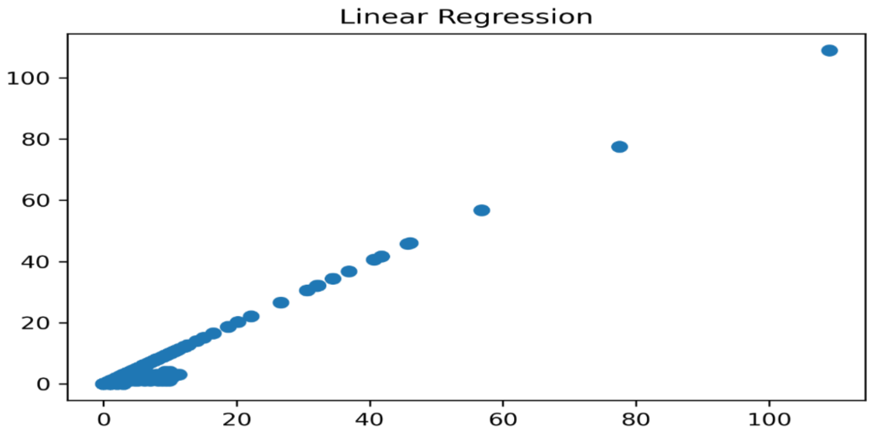


Figure 43 Water Model Leaner Regression

1. (SVM) Model Confusion Matrix in Figure 44 SVM Model Confusion Matrix.

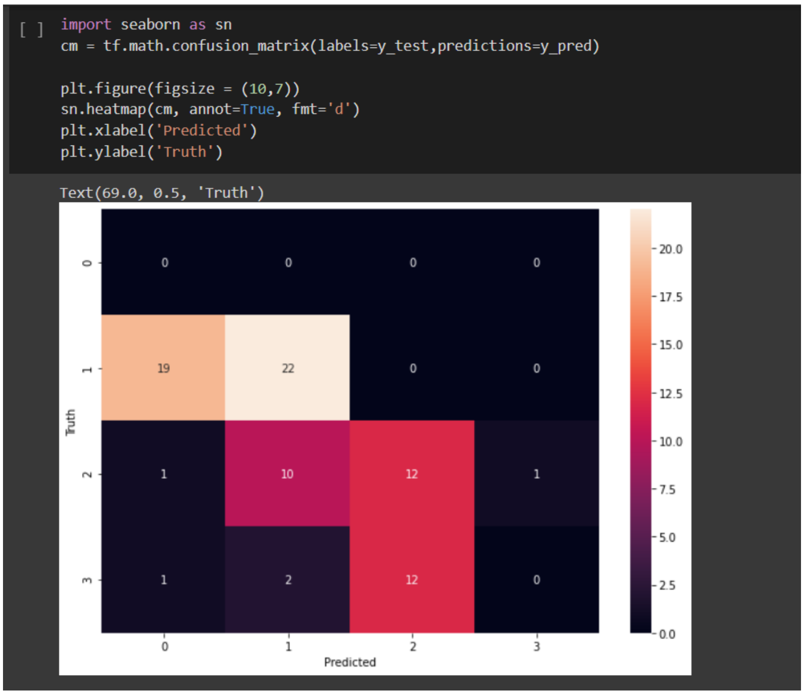


Figure 44 SVM Model Confusion Matrix

2. ANN Model Confusion Matrix in Figure 45 ANN Model Confusion Matrix

```python
import seaborn as sn
cm = tf.math.confusion_matrix(labels=y_test,predictions=y_pred)

plt.figure(figsize = (10,7))
sn.heatmap(cm, annot=True, fmt='d')
plt.xlabel('Predicted')
plt.ylabel('Truth')
```

```
Text(69.0, 0.5, 'Truth')
```

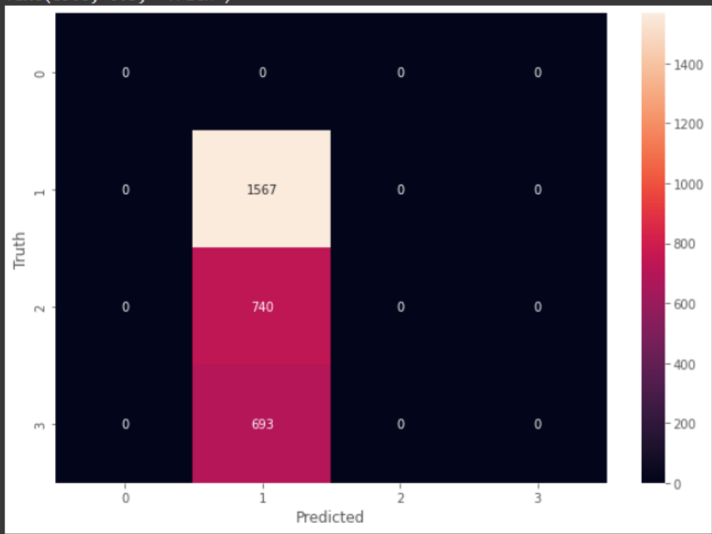


Figure 45 ANN Model Confusion Matrix

3. Value Loss Over Epoch in Figure 46 Value Lost Over Epoch.

```
plt.style.use('ggplot')
plt.plot(hist.history['loss'], label = 'loss')
plt.plot(hist.history['val_loss'], label='val loss')
plt.title("Loss vs Val_Loss")
plt.xlabel("Epochs")
plt.ylabel("Loss")
plt.legend()
plt.show()
```

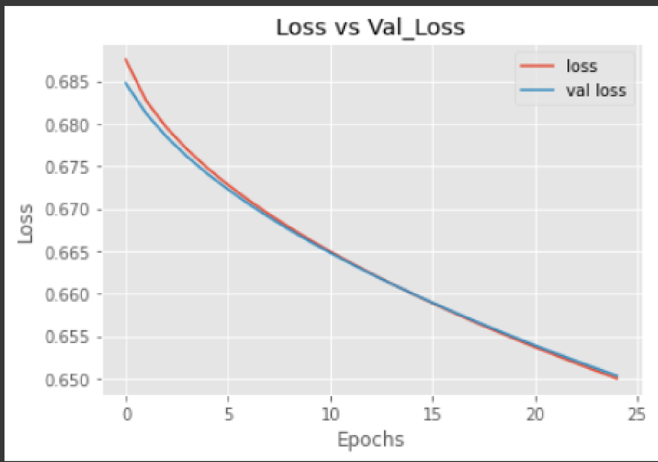


Figure 46 Value Lost Over Epoch

- Decision Tree Model with Figure 48 Decision Tree at alpha = 0.05

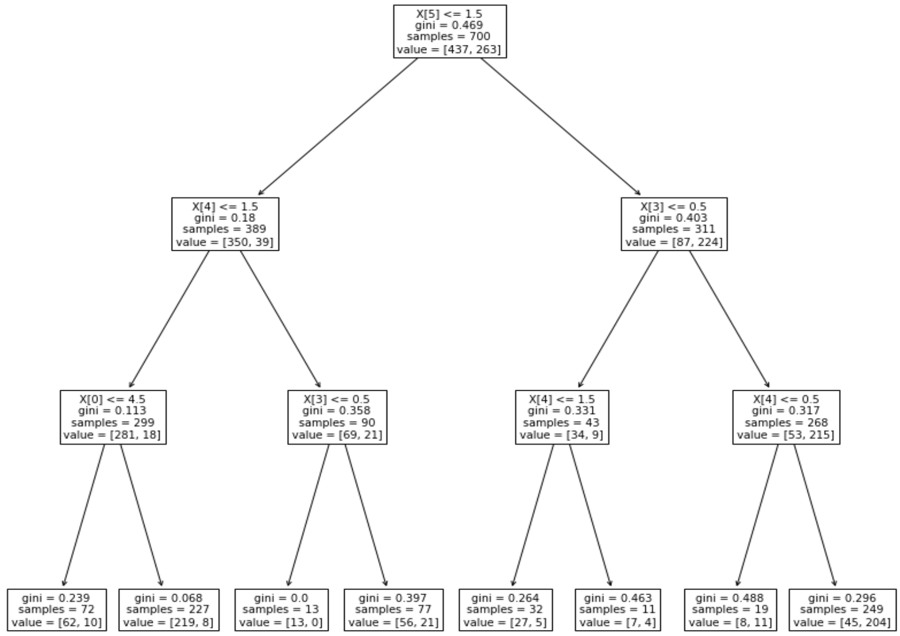


Figure 47 Decision Tree at alpha = 0.05

Leaf Index

Values

- X[0] = Distance
- X[1] = Humidity
- X[2] = Temp
- X[3] = Current Rain
- X[4] = Rain Rage
- X[5] = Water levels

Table 7 Accuracy table of water prediction models

Data	**Accuracy(percentage)%**		
	SLR	SVM	ANN
Train	81.2	79.4	89.01
Test	79	77.0	84.3

Table 8 Accuracy table of decision tree model

Data	Decision Tree
Train	50.4%
Test	52.4%

Crowdsourcing outcome evaluation

The crowdsourcing data is gathered through the implemented mobile application. bellow tables represent the sample data sets that we gathered in different weather conditions in three different locations by participating volunteers

Location	Weather condition	Flood	Flood water level	Severity of rainfall	Severity of rainfall with flooding
Pannipitiya	Overcast Clouds	No	None	None	None
	Light Rain	No	None	Low	None
	Heavy Rain	No	None	High	None

Table 9 Crowdsource Dataset 01

Location	Weather condition	Flood	Flood water level	Severity of rainfall	Severity of rainfall with flooding
Nugegoda	Clear Sky	No	None	None	None
	Thunderstorm	No	None	High	None
	Heavy Rain	No	None	High	None

Table 10 Crowdsource Dataset 02

The above tables represent the gathered crowdsourcing datasets. The next step is to visualize gathered crowdsourcing results. The below UI shows the visualization dashboard of the gathered crowdsourcing results in a specific location

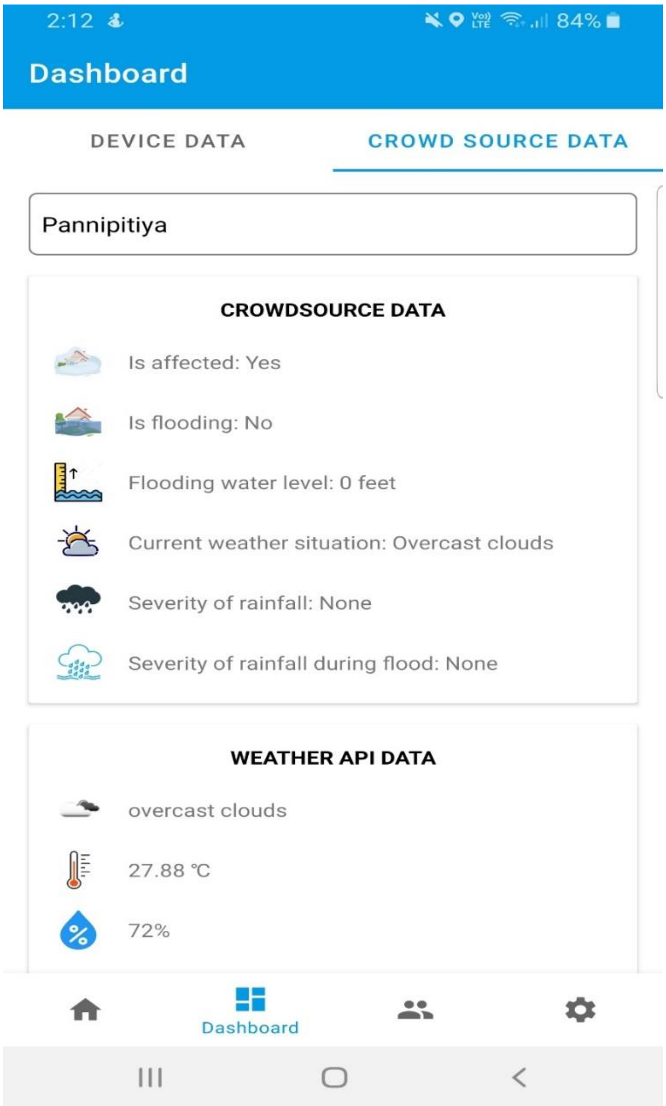


Figure 48 Crowdsource Dashboard

IoT Device Functionality

Device connection to the network

After the completion of the code writing to the Weather monitoring device which was written to detect the respective weather factors from the relevant sensors the code was successfully uploaded to the device via the Arduino IDE. All sensors need to be plugged to the device with the relevant pins in the correct slots. To feed the code to the device it needs to be plugged to the PC/Laptop with an USB connection. Once the device is plugged the connection to the network could be witnessed on the serial monitor of the IDE and the Weather monitoring device OLED too.

Figure 49: Network Connection preview on OLED

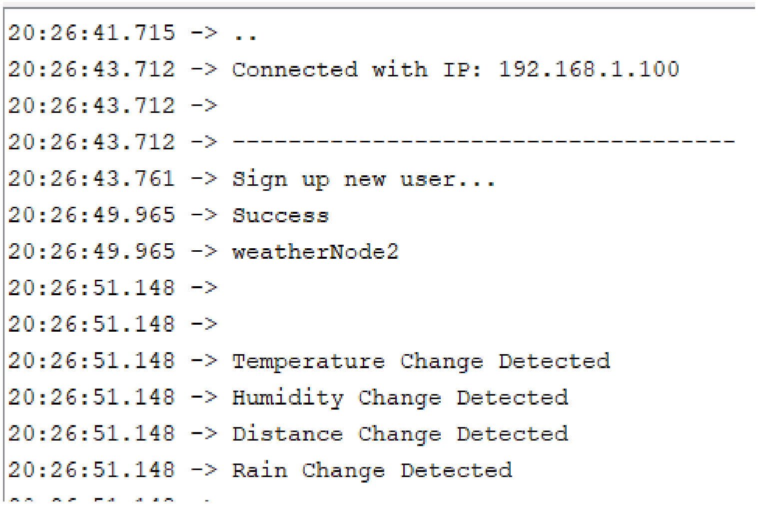


Figure 50: Network Connection preview on Serial Monitor

Data Preview on the OLED screen and data transmission to the Database

Once the connection to the network is successfully established the sensor detected data will be previewed on the OLED screen and will be passed onto the Database with a regular interval.

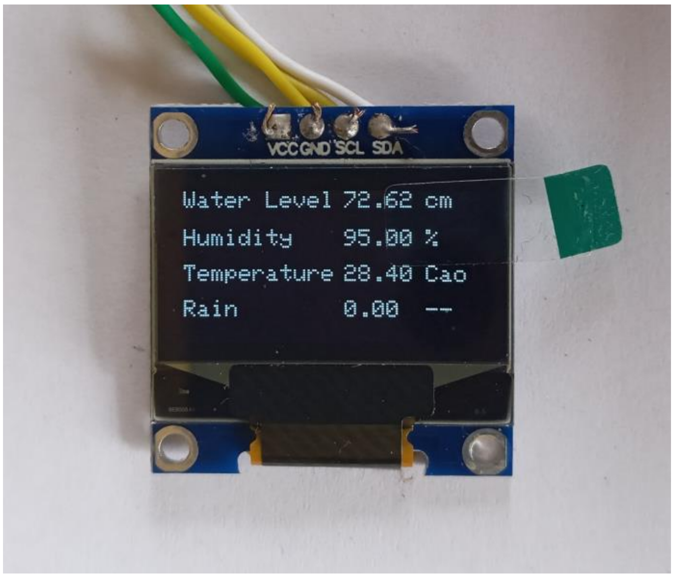


Figure 51: Sensor Data displayed on the OLED screen

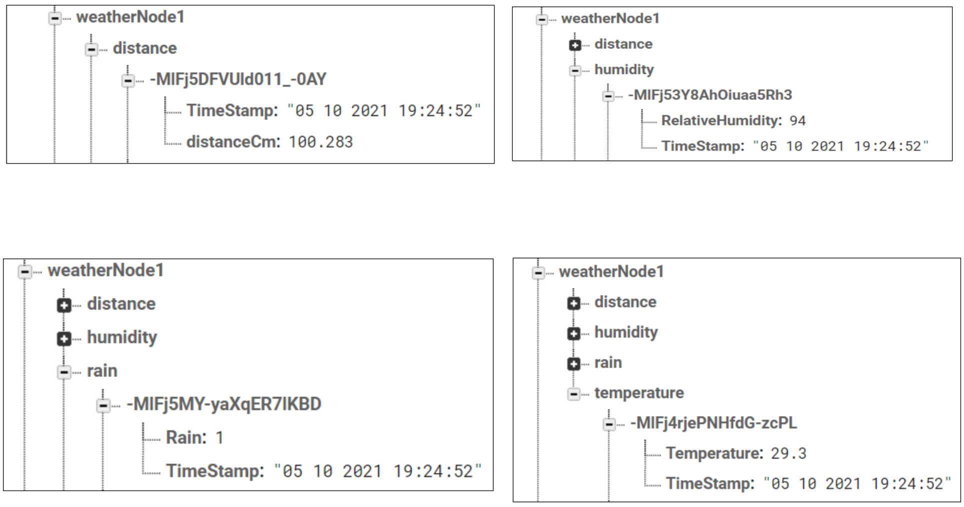


Figure 52: Sample data from the Database which were transmitted from the IOT

Sensor functionalities

DHT11 Temperature and Humidity Sensor

The DHT11 sensor which is used to scrutinize the temperature and the humidity of the environment of a specific location is coded to read the factors as the temperature as Celsius and the humidity as a percentage(%).

```
void read_DHT() {
  delay(delayMS);
  sensors_event_t event;
  dht.temperature().getEvent(&event);
  if (isnan(event.temperature)) {
    Serial.println(F("Error temperature!"));
  }
  else {
    //Serial.print(F("Temperature: "));
    //Serial.print(event.temperature);
    temp = event.temperature;
    //temp = floorf(event.temperature * 100) / 100;
    //Serial.println(F("°C"));

  }
  dht.humidity().getEvent(&event);
  if (isnan(event.relative_humidity)) {
    Serial.println(F("Error humidity!"));
  }
  else {
    //Serial.print(F("Humidity: "));
    //Serial.print(event.relative_humidity);
    humidity = event.relative_humidity;
    //Serial.println(F("%"));
  }
}

```

Figure 53: DHT11 Sensor reading implementation

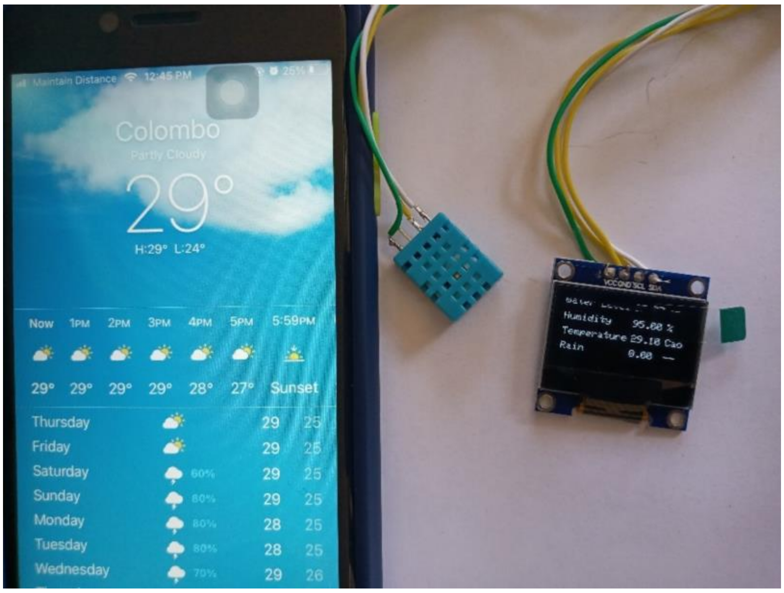


Figure 54: Sensor data reading against actual data

The code which is implemented to monitor the temperature and the humidity clearly shows that the accuracy of the sensor reading is correct against verified temperature data.

AJ-SR04T Distance Sensor Module

This distance sensor module will read the increasing of the water level in that location where the device is placed. When the distance between the sensor and the water level is decreasing which means the water level has increased.

Distance Calculation- 1 Centimeter = 0.4 Inches

Sensor distance between the water level(Increase/Decrease) Water level(Increase/Decrease)

```
void read_distance() {
  // Clears the trigPin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);

  // Calculate the distance
  distanceCm = duration * SOUND_SPEED / 2;

}
```

Figure 55: Distance detection implementation

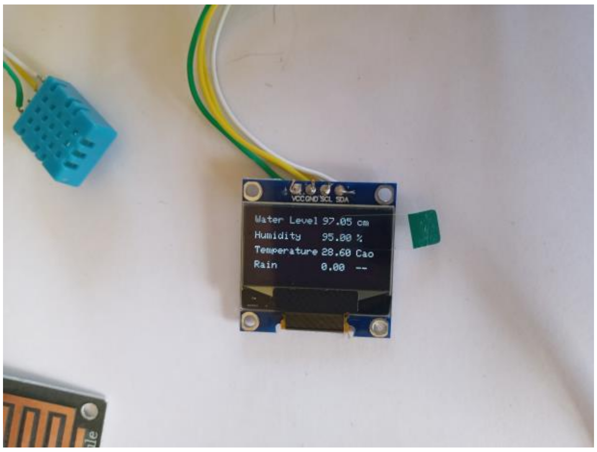


Figure 56: Distance module reading

Rain Drop Detection Sensor

The rain drop detection sensor will give a digital output as 1 or 0 as in 1 defines as when a rainfall is detected and 0 defines as when there is no rainfall.

```
void read_rain() {
  if (digitalRead(Rain_sensor) == HIGH) {
    Rain = 0.0;
  }
  else {
    Rain = 1.0;
  }
}
```

Figure 57: Rain Detection sensor module implementation

Figure 58: Rainfall sensor reading

<u>Non-Subscribed user solution</u>

<u>SMS Weather Data providing</u>

For users who are not able to access the mobile application and the web application due various reasons they are yet able to receive live weather data information, which is gathered from the weather data monitoring device via SMS. This solution is provided to the user upon user requests. Whereas the user will be able to receive weather data information via SMS only from the locations where an IoT device is placed. To fulfill this concept, we have used Twilio as our service provide to succeed this concept.

Mobile number purchased- (205) 832-3294

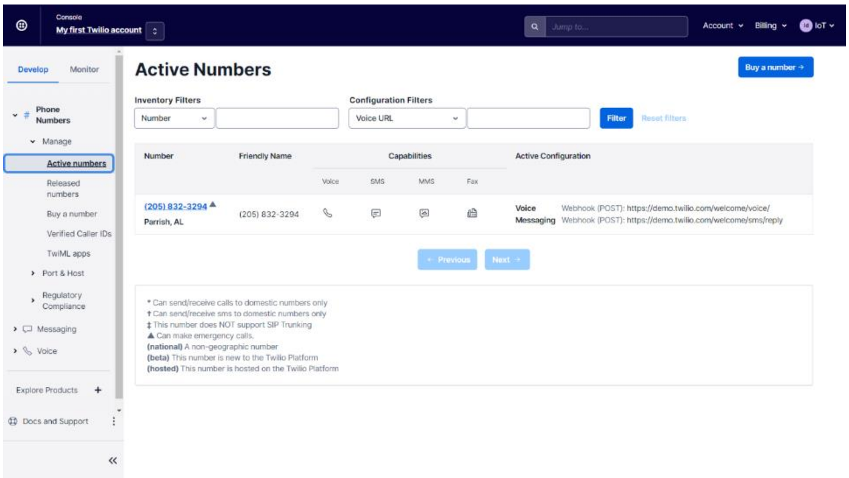


Figure 59: Twilio purchased mobile number

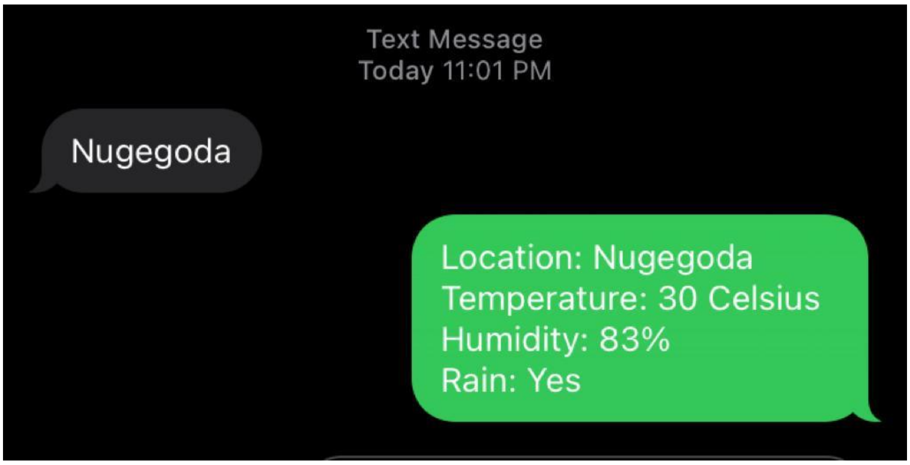


Figure 60: Sample SMS with weather information received upon user request

Data Output and Preview

The data detected by the sensors will be passed on to the Database and once the JSON file is extracted and processed to a precise format below will be preview in results.

Location	Date and Time	Distance (CM)	Humidity (%)	Rainfall (1/0)	Temperature (C)
Nugegoda	05 10 2021 19:24:52	100.283	94	1	29.3
	05 10 2021 19:25:22	100.249	94	1	29.4
	05 10 2021 19:25:52	100.266	93	1	29.4
	05 10 2021 19:26:23	100.266	93	1	29.4
	05 10 2021 19:26:55	100.283	93	1	29.4

Table 11: Data readings from the IoT device of Nugegoda

Location	Date and Time	Distance (CM)	Humidity (%)	Rainfall (1/0)	Temperature (C)
Pannipitiya	05 10 2021 19:26:55	100.283	93	1	29.4
	05 10 2021 19:28:28	100.266	93	1	29.4
	05 10 2021 19:28:59	100.266	93	1	29.4
	05 10 2021 19:29:29	100.249	93	1	29.4
	05 10 2021 19:30:00	100.249	93	1	29.4

Table 12: Data readings from the IoT device of Pannipitiya

10.2 Research findings

The goal of this research is to better understand the attribution contributions of rainfall based on three different locations in Sri Lanka and to improve the process' overall performance. The data was gathered in three different locations: Colombo (coast and city), Vavuniya (countryside), and Katugastota (hills).

The contribution of temperature and relative humidity is higher in causing rainfall, but the contribution of the attribute "month" can be neglected, and the attribute pressure gives less contribution to rainfall when compared to temperature and relative humidity. Maximum temperature and minimum relative humidity with a time frame of the day (24 hours) and location as a constant are significant values. While using the same attribute data set for each specific location selected from Sri Lanka, such as Colombo (coastal and city), Vavuniya (countryside), and Katugastota (Hills), the prediction error or accuracy of each location model is different. Coastal and city location prediction errors seem higher than in hills or rural areas.

SVM and Logistic Regression are roughly similar in terms of accuracy. While using the Binary model output "Rain or Not" attribute as an extra input to the multi-classification model, its accuracy has increased. Auto sci-kit learns to select random forests as the best accuracy model. Random forest gives better accuracy than support vector machine or logistic regression. Because random forest uses many decision trees and prioritizes the model with the highest accuracy, it provides better accuracy.

The above crowdsourcing tables represent the crowdsourcing data sets that have been gathered in different weather conditions in cities. Integrating mobile technology with crowdsourcing has experimented a positive impact in public during a disastrous event. Authors identified crowdsourcing approach is a conventional way of collecting information in a natural disaster. The outcome of the crowdsourcing data is capable of saving millions of lives by promoting availability of disaster data. The mobile application helps users to view and track weather conditions and flood information in specific areas.

Based on the research results authors have come to a decision that crowdsourcing platform is effective in collecting accurate volunteer information with the help of majority decision function. Majority decision is used to reduce number of false entries. Since the mobile application gather crowdsource data from citizens it is important reduce number of false entries. Majority decision function is based on a K means clustering algorithm to eliminate number of false entries and analyze the most common crowdsource data set. When displaying the crowdsource data, majority decision function experimented positive results by displaying most accurate crowdsource data set in the mobile dashboard. These accurate crowdsource data sets will support the decision-making procedures in disaster risk management

In the current world there are numerous microcontrollers which can perform various functions in terms of IoT. We have chosen the ESP32 NodeMCU module as in this microcontroller has the functionality of connecting to the network via Wi-Fi which is also meant by there this microcontroller has and inbuilt Wi-Fi module. Currently in many other research conducted seems to be having an issue in connecting to the internet as in they have used an external Wi-Fi module. Which also reduces the performance load on that sensor which is leads to data transmission issues. To avoid all possible issues in the network connection and the data transmission to the DB we have chosen to proceed with an ESP32.

Reaching out to the user who does not have access to the system or application was one of our main targets as in we need to reach out to all the possible stakeholder for a better information providing. Therefore, we have chosen to reach out to such users via SMS as in this method is more feasible than approaching them through an internet connection due to various reasons such as not having an internet functioned mobile phone, out of data etc. reaching out to all the stakeholders in system has been an emerging area in the current world as in some systems.

Most flood forecasting models use average results as shown in Table 7, yet the implementation of ANN has higher accuracy than all. Furthermore, results accuracy is check by conducting a metric of Mean Absolute Error for standardization. Regardless of the location of the gauge station, the accuracy of the test data does not vary.

Decision making model utilizes flood forecast of gauge stations, rainfall prediction model and IoT sensory data in Figure 48 Decision Tree at alpha = 0.05. It is a multi-criteria decision tree based (MCDT), model is most suitable for classification and can executed immediately and accurately Table 8.

10.3 Discussion

The contribution of temperature and relative humidity is higher in causing rainfall as expected, but the contribution of the attribute “month” can be neglected while using specific locations, and the attribute pressure gives less contribution to rainfall. While using the same attribute data set for each specific location selected from Sri Lanka, such as Colombo (coastal and city), Vavuniya (countryside), and Katugastota (Hills), the Vavuniya and Katugastota models give better accuracy than the Colombo model.

SVM and Logistic Regression are roughly similar in terms of accuracy. While using the Binary model output "Rain or Not" attribute as an extra input to the multi-classification model, its accuracy has increased. Auto sci-kit learn was used to compare the best classifier accuracy and our model accuracy for binary and multi-classification. Auto sci-kit learns to select random forests as the best accuracy model. Random forest gives better accuracy than support vector machine or logistic regression. Because random forest uses many decision trees and prioritizes the model with the highest accuracy, it provides better accuracy. While the split data set for each decision tree is important, the misclassified data from the earlier decision trees get more priority.

We tested the mobile application by participating volunteers in Colombo area and gathered responses in different weather conditions accordingly. Using the mobile application and the location tracking function enables users to submit correct and accurate information. Based on the gathered data sets it will enforce disaster risk limiting. Disaster risk limiting can directly benefit from crowdsourcing as an example by actively participating and communicating information will overcome unavailability of information before, during and after an extreme weather situation. Finally integrating overall crowdsourcing data sets will be a great aid for increasing the accuracy of other weather forecasts and early warning systems.

This designed IoT device was designed using a PCB board which will give more feasible and precise structure. Also, in terms of the durability and scalability is more efficient and higher than a normal design based on the breadboard. The relevant sensors are plugged on to the PCB board accordingly to read the relevant weather data factors. This device in in size (Width, Length, Height) is 104mm x 104mm x 20mm. Whereas a feasible size and structure was used to avoid damages occurring and easy handling in the device.

And ESP32 has a Wi-Fi embedded module in it which has the capability of connecting to the internet to transmit data to the DB or to the relevant platform. We have used this module instead of adding an additional Wi-Fi module to device which could lead to performance issues due to the increasing size of the code.

This proposed and implemented IoT device is a low cost and open source-based platform. Low cost (Table 3) is meant by that the sensors and the components used in this assembled device could be easily found at a low cost by several vendors. These components such the sensors could be easily used as in plug and use once the code is fed to the device on the functionality of the IoT device. The Arduino device is capable to read, write & deliver the output to a certain platform.

User who could not access our system will not be able to get the relevant weather data information. Due to this reason, we have provided a solution for them whereas they could send a SMS with the location they would like to have the weather updated and the system will return the weather data gathered from the IoT device placed in that specific location. This solution was implemented to ensure that every stakeholder could benefit from this system and could be aware of the weather information accordingly. There might be many reasons where a user isn’t able to access to our system such as lack of hardware facilities, software not supporting etc. Due to this reason since an SMS is a basic functionality in every mobile any user could benefit from this system without any issues.

The contribution for water level predictions, from the river basin data as each station has the daily fluctuations of the water levels, will not provide the whole predictions of an hourly occurrence. Providing a live data set from the gauge stations and remarks of the Disaster Management Centre (DMC) of Sri Lanka has a huge data change in the reports they provide. Building another IoT base device to water levels for River basin areas will provide continuous live river basin water readings.

The decision-making models currently uses decision tree has a lower accuracy than expected. As I can suggest using Decision Making Neural Decision Forests model will result in more accurate predictions.

10.4 Summary of student contribution

Member	Component	Task
Ilukkumbure S. P. M. Kaveesha. W	Designing and developing machine Learning Model for water level prediction of the river station. Design and develop the machine learning models for decision making.	Feasibility study to perceive the requirement of the aspect. Develop APIs to precent predictions. Develop API to integrate system with other government systems. Develop Dashboard to Precent Predictions. Design Web Application and Fronted.
Vinobaji. S	Development of a machine learning algorithm for rainfall prediction based on locations based which one use in Decision making model as input.	A feasibility study is conducted to determine the aspect's requirements. Identify the best dataset based on attributes contribution in accuracy which will be used in rainfall prediction model. Create a framework for obtaining weather data as an input.

		Test the generated machine learning model on a dataset with known results to verify accuracy. To increase accuracy, use binary classification model output as input for Multi classification model.
V.Y Samarasiri	Development of crowdsourcing component	Feasibility study to perceive the requirement of the aspect. Implement the mobile app user registration Establishing connection to the open weather map API, using API unique key and retrieve live weather data and display using a suitable way Finalize crowdsourcing data that has to be gathered from volunteers

		Implement the crowdsourcing UI Save crowdsource data and retrieve crowdsourcing weather data from DB and find the common data set and validate the data set against IOT data & weather API data Implement the dashboard for visualize validated crowdsource data.
M. F. Mohamed	Designing and developing an IoT weather monitoring system. Development of SMS weather data providing system	Extensive Study on IoT device designing & implementation. Implementation of a IoT based weather monitoring device which includes a Wi-Fi module for data transmission. Integration between the IoT device and database

		for the other system functionalities. Previewing the IoT gathered data on the mobile application in a detailed manner. Develop a method for users to receive weather data information via SMS.

11 BUDGET JUSTIFICATION

Component	**Quantity**	**Price**
ESP32 Microcontroller	1	LKR 1,550.00
DHT11 Temperature and Humidity Sensor	1	LKR 325.00
Rain Drop Sensor	1	LKR 200.00
HCSR04 Ultrasonic Sensor	1	LKR 250.00
RTC Module	1	LKR 160.00
Historical Weather data for rainfall prediction		LKR 28800.00
Azure VM: 1 vCPU 1GB $7.592/month	6	LKR 9156.78
Irrigation Department Reports for Flood Level Prediction Three Years (2019-2021-July) Rs.40 per station / month	36	LKR 7200.00
Total		**LKR 47641.78**

Table 13: Budget

12 CONCLUSION

The contribution of temperature and relative humidity is higher in causing rainfall as expected because of the maximum temperature and minimum relative humidity with time frame day (24hrs) and location as constant being significant values. but the contribution of the attribute "month" shows less contribution can be neglected while using specific locations, and the attribute pressure gives less contribution to rainfall. While using the same attribute data set for each specific location selected from Sri Lanka, such as Colombo (coastal and city), Vavuniya (countryside), and Katugastota (Hills), the Vavuniya and Katugastota models give better accuracy than the Colombo model. Location Colombo can be considered as another factor that influences place. This may be due to coastal areas or artificial factors (Global Warming, deforestation).

I've found that SVM and Logistic Regression are roughly similar in terms of accuracy. Even though the auto-machine learning model auto sci-kit learn shows the best rainfall forecast as Random Forest. The data used as input for prediction and classification has a significant impact on the percentage of accuracy and prediction. All models have advantages and disadvantages, and the most difficult part is deciding which model is the best. Logistic Regression for binary classification and SVM for multi-classification models indicate a proficient and appropriate model for this rainfall prediction.

Auto sci-kit learn was used to compare the best classifier accuracy and our model accuracy for binary and multi-classification. The best accuracy model, according to Auto Sci-kit Learn, is random forest. Because random forest uses numerous decision trees and prioritizes the best accuracy models, it provides improved accuracy. While dividing the data set for each decision tree, the misclassified data from the previous decision tree is given higher priority.

Future work can be considered as comparing another rural coastal location's accuracy with a city coastal location, considering artificial factors that influence rainfall (air pollution level of a location, etc.), planning to use a deep learning model to predict rainfall, and finding another time factor which can be used to replace the month.

The conducted research of experimental water level forecasting machine learning models with use of Linear Regression, Support Vector Machines and Artificial Neural

Network based models provided accuracies did not affect regardless the station of the station so I conclude Kalu River Basin and slope has an equal distribution of water level and each of the factor of water level catchment area does not effectiveness of the flooding models because it’s a constant.

The natural factor of data does affect the area of Sabaragamuwa province and Western Province, and the Sea Level was not added to equation and not identify as a contributing factor. The sudden variations of water model prediction really have contribution to the decision trees and the values really do affect the predictions accuracy. Data driven solutions are highly dependable on data quality, therefore advance data collection and analyzing methods should be carried out. The final flood forecast, and predictions are displayed on the web application and mobile application to distribute predictions. Implementation of this kind of hybrid models require further research.

After conducting a successful research on crowdsourcing for flood risk management, the presented work fills the gap that has been mentioned in the research gap section in terms of crowdsourcing approach. In this crowdsourcing approach volunteers are taken as human sensors that will contribute information about metrological factors such as weather condition, flooding water level and flooded areas. Regarding the crowd sourcing-based approach further improvements and research can focus in these core areas. Weather forecast efficiency can be improved in collecting more factors from the crowd such as picture uploading function in a crisis with combining qualitative and quantitative inputs from the crowd and it will further benefit early warning systems. The success of the crowd sourcing solution depends on active participation of the crowd therefore there is a need for enabling and improving the contributing information in a disaster event. Further research can be done in implementing an offline method for involving crowd in a natural disaster and expanding the involvement in crowd for more cities.

Weather situations are one of the key factors in the day-to-day human lives. Whereas for agriculture, education, development activities etc. are some of the main areas where the weather conditions impact directly. Due to these reasons, having a proper and accurate weather change information has a valuable responsibility in enlightening the human lives.

Even though with the emerging technology in the world scrutinizing weather condition through satellites and high technology equipment's, still there are hurdles in monitoring and providing accurate weather information to a miniature area.

In this designed system we have implemented an IoT devices which has a connection with internet in terms of live data transmission and sensors embedded to the IoT device to detect weather data and will be previewed on the mobile application and web application dashboard respectively. These data could be used by various industries as in there is a direct impact of the weather situation. These industries are agriculture, traffic, logistics, development, and other day to day human activities etc.

These lively gathered data will be validated and verified with the help of the other models in this system such as using the ML, AI, and the crowdsourcing technology. Therefore, the data which is presented to the stakeholders of this system are well validated and verified data.

For the users who are unable to access the weather data via our mobile application and web application they could request weather data location wise via SMS and the relevant data of that location will be delivered to the users via the same mode. This methodology was implemented to this system as in our main aim was to reach every stakeholder of the system considering all the possible hurdles the users will face too.

As further improvements with regard to this designed Iot device it could be embed with an CMOS Camera Module which could be used for image processing where the gathered data will be verified clearly. Also, this device could be embedded more with other sensor modules such as an Anemometer, light detection sensor etc. to improve the productivity of the device. This device could also be incorporated with a GSM module which will also benefit in terms network connection where the device will not need to connect to an external network to establish the DB connection. Since the ESP32 contains of a Bluetooth facility also this could be used to create a mesh network with the other weather monitoring IoT devices to avoid data transmission disruption.

With regard to the SMS based weather data providing methodology this could be improved by having an automated weather information providing method to user. This could be setup as the user could enroll to our service and they will receive weather data updates in a regular interval inevitably.

13 REFERENCES

[1] “Weather Forecasts Based on Rainfall Prediction Using Machine Learning Methodologies,” Adalya J., vol. 9, no. 6, 2020, doi:

[2] E. J. Plate, “Flood risk and flood management,” J. Hydrol., vol. 267, no. 1–2, pp. 2–11, 2002, doi: 10.1016/S0022-1694(02)00135-X.

[3] E. M. Mendiondo, “Flood Risk Management of Urban Waters in Humid Tropics: Early Warning, Protection and Rehabilitation,” Work. Integr. Urban Water Manag., no. April, pp. 2–3, 2005.

[4] P. Deckers, W. Kellens, J. Reyns, and W. Vanneuville, “Geospatial Techniques in Urban Hazard and Disaster Analysis,” Geospatial Tech. Urban Hazard Disaster Anal., no. December, 2010, doi: 10.1007/978-90-481-2238-7.

[5] “Rainfall Prediction for Udaipur, Rajasthan using Machine Learning Models Based on Temperature, Vapour Pressure and Relative Humidity,” Int. J. Recent Technol. Eng., vol. 8, no. 6S, pp. 133–137, 2020, doi: 10.35940/ijrte.f1024.0386s20.

[6] K. Dutta and P. Gouthaman, “Rainfall Prediction using Machine Learning and Neural Network,” Int. J. Recent Technol. Eng., vol. 9, no. 1, pp. 1954–1961, 2020, doi: 10.35940/ijrte.a2747.059120.

[7] S. B and J. K.S, “Rainfall Prediction Using Machine Learning Techniques and an Analysis of the Outcomes of These Techniques,” Int. J. Eng. Appl. Sci. Technol., vol. 04, no. 09, pp. 365–371, 2020, doi: 10.33564/ijeast.2020.v04i09.047.

[8] H. Thilakarathne and K. Premachandra, “Predicting Floods in North Central Province of Sri Lanka using Machine Learning and Data Mining Methods,” no. October, pp. 44–49, 2017.

[9] A. Mosavi, P. Ozturk, and K. W. Chau, “Flood prediction using machine learning models: Literature review,” Water (Switzerland), vol. 10, no. 11, pp. 1–40, 2018, doi: 10.3390/w10111536.

[10] L. C. Degrossi, J. P. De Albuquerque, M. C. Fava, and E. M. Mendiondo, "Flood citizen observatory: A crowdsourcing-based approach for flood risk management in Brazil," Proc. Int. Conf. Softw. Eng. Knowl. Eng. SEKE, vol. 2014-January, no. January, pp. 570–575, 2014.

[11] W. T. Liang, J. C. Lee, and N. C. Hsiao, "Crowdsourcing platform toward seismic disaster reduction: The Taiwan scientific earthquake reporting (TSER) system," Front. Earth Sci., vol. 7, no. April, pp. 1–12, 2019, doi: 10.3389/feart.2019.00079.

[12] A. Schulz, H. Paulheim, and F. Probst, "Crisis information management in the Web 3.0 age," ISCRAM 2012 Conf. Proc. - 9th Int. Conf. Inf. Syst. Cris. Response Manag., no. April, pp. 2–6, 2012.

[13] O. Mejri, S. Menoni, K. Matias, and N. Aminoltaheri, "Crisis information to support spatial planning in post disaster recovery," Int. J. Disaster Risk Reduct., vol. 22, pp. 46–61, Jun. 2017, doi: 10.1016/J.IJDRR.2017.02.007.

[14] M. Careem, C. De Silva, R. De Silva, L. Raschid, and S. Weerawarana, "Sahana: Overview of a disaster management system," 2nd Int. Conf. Inf. Autom. ICIA 2006, no. January, pp. 361–366, 2006, doi: 10.1109/ICINFA.2006.374152.

[15] M. Prasanna, M. Iyapparaja, M. Vinothkumar, B. Ramamurthy, and S. S. Manivannan, "An Intelligent Weather Monitoring System using Internet of Things," Int. J. Recent Technol. Eng., vol. 8, no. 4, pp. 4531–4536, 2019, doi: 10.35940/ijrte.d8464.118419.

[16] S. R, S. V.P, S. E, and J. B, "Real Time Water Monitoring System using IoT," Ijarcce, vol. 6, no. 3, pp. 378–380, 2017, doi: 10.17148/ijarcce.2017.6386.

[17] P. Yadav, N. Nigam, and R. Yadav, "IoT based- Advanced Weather Monitoring System IoT based- Advanced Weather Monitoring System," 2020.

[18] A. M. Bhagat, A. G. Thakare, and K. A. M. | N. S. M. | P. V. Choudhary, "IOT Based Weather Monitoring and Reporting System Project," Int. J. Trend Sci. Res. Dev., vol. Volume-3, no. Issue-3, pp. 365–367, 2019, doi: 10.31142/ijtsrd21677.

[19] B. Li et al., “On the operational flood forecasting practices using low-quality data input of a distributed hydrological model,” Sustainability (Switzerland), vol. 12, no. 19, 2020, doi: 10.3390/su12198268.

[20] S. Kokularamanan, A. W. M. Rasmy, D. Perera, and T. Koike, “Development of a Flood Forecasting and Data Dissemination System for Kalu River Basin in Sri Lanka,” Annual Sessions of IESL, The Institution of Engineers, Sri Lanka, vol. 1, pp. 205–210, 2017.

[21] M. J. D. Fernando, D. A. K. K. Pathirana, W. J. K. T. D. Jayasooriya, S. A. H. Rathnaweera, and L. Rupasinghe, “Intelligent Flood Management System,” in 2019 International Conference on Advancements in Computing (ICAC), 2019, pp. 79–84. doi: 10.1109/ICAC49085.2019.9103407.

APPENDICES

Weather Historical data collection from the Meteorology Department of Sri Lanka.

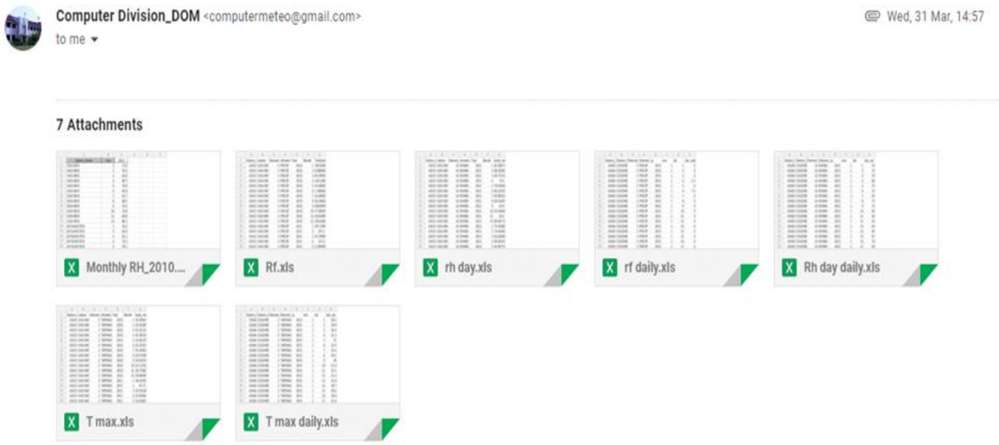


Figure 61: Weather historical data collection

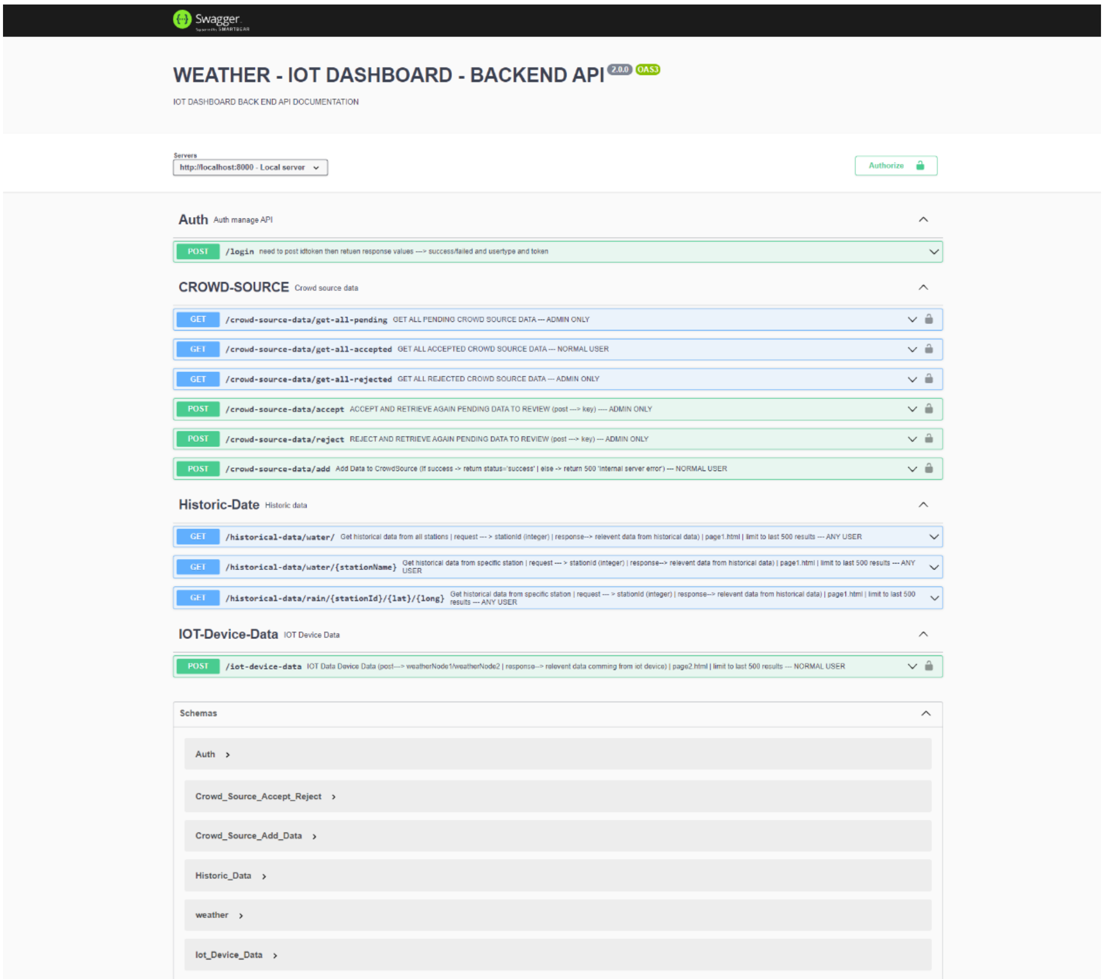


Figure 62Weather API for Integration

Figure 63 IoT Dashboard