Advancement of Real-Time Flood Monitoring and Navigation system via LoRaWan-based Wireless sensor Networks and AI-Powered Chatbot

John Loyd Maispag
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
johnloyd.masipag@tup.edu.ph

Jemimah Aniel Climaco
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
jemimahaniel.climaco@tup.edu.ph

Khen Kyrill Dado
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
khenkyrill.dado@tup.edu.ph

Jericho Martin Franco
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
jerichomartin.franco@tup.edu.ph

Julius Magayon
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
julius.magayon@tup.edu.ph

Jomel Manalastas
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
jomel.manalastas@tup.edu.ph

Romalyn Manansala
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
romalyn.manansala@tup.edu.ph

Christian Ceballos
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
christian.ceballos@tup.edu.ph

Justice Colili
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
justice.colili@tup.edu.ph

Angelyn Magsino
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
angeline.magsino@tup.edu.ph

Andrei Miguel Marquez
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
andreimiguel.marquez@tup.edu.ph

Carl Justin Viernes
Department of Electronics Engineering
Technological University of the Philippines
Manila, Philippines
carljustin.viernes@tup.edu.ph

Abstract*—The Flood and Road Eye Navigation System (FRENS) addresses urban flooding in Metro Manila by offering real-time flood and road monitoring. It comprises Mother and Sub Nodes equipped with sensors measuring atmospheric pressure, floodwater levels, and precipitation. These nodes communicate seamlessly using LoRa technology, with data transmission facilitated by a GSM module to a centralized database. An IP camera aids in traffic detection and monitoring, while solar panels ensure continuous operation, even in remote areas. FRENS integrates computer vision and a chatbot, utilizing Google Maps and the Dijkstra algorithm for route optimization. Leveraging YOLOv8 Nano and ByteTrack technologies, it detects vehicles and processes video sequences for real-time traffic monitoring. The chatbot, driven by GPT-3.5 NLP, provides users with real-time updates and facilitates interactive communication. This comprehensive approach integrates data visualization, advanced analytics, and real-time communication, offering commuters and drivers alternative routes. By enhancing road safety and bolstering disaster resilience in flood-prone urban areas like Manila, FRENS serves as a vital tool in addressing the challenges of urban flooding.

***Keywords*—real-time monitoring, LoRa, GSM module, IP camera, traffic detection, solar panels, computer vision, chatbot, Google Maps, Dijkstra algorithm, YOLOv8 Nano, ByteTrack, GPT-3.5 NLP**

I. INTRODUCTION

Due to the increasing prevalence and severity of extreme weather events linked with climate change, the necessity for effective and efficient flood monitoring and management systems has become more critical than ever. Monitoring, forecasting, simulation, evaluation, and analysis are essential in mitigating the impacts of flooding. In this context, integrating Artificial Intelligence (AI) can significantly enhance flood events' real-time detection and forecasting, resulting in more accurate and timely responses.

This project proposes a system that employs AI algorithms and flood sensor data to detect and analyze flooding situations. Upon detection, the information is displayed on a navigational map within an application. This real-time updating of flood-related data enables users to make informed decisions about the safest and most efficient routes during such events. The system goes beyond detecting and displaying flood events by offering optimized alternative routes that consider flood conditions and current traffic situations. This ensures users avoid flood-impacted areas and take the least congested routes, providing a lower estimated arrival time. The project explores this system's development and potential impact, demonstrating how AI can be harnessed to address the challenges posed by navigation during flooding events.

The Philippines' vulnerability to tropical cyclones, which result in torrential rainfall, floods, and severe damage to infrastructure, is a significant concern. Approximately 20 cyclones enter the country's area of responsibility annually, leading to severe flooding that exacerbates traffic congestion and disrupts the transportation system. Despite efforts by government agencies, traditional flood monitoring and navigation systems often need to be improved for immediate decision-making due to their reliance on crowd-sourced data and official reports. This reliance can intensify navigation problems for users.

The emergence of LoRaWAN-based wireless sensor networks (WSNs) presents a promising solution to the limitations of traditional flood monitoring systems. LoRaWAN technology offers advantages such as low power consumption, long-range communication capabilities, and cost-effectiveness, making it suitable for deploying sensors in remote or inaccessible flood-prone areas. The deployed system can provide real-time data on flood levels, rainfall amounts, and temperature by integrating float switches, rain gauges, and barometric sensors. Unlike traditional systems, this technology allows users to monitor the latest flood conditions based on sensor data, enhancing their ability to make informed decisions during flood events.

A. Google Map

As Google Maps depends on real-time data from multiple sources, it cannot provide past flooding data. Although it successfully provides up-to-date traffic information and specific weather conditions, including suggesting alternative routes based on traffic circumstances, it cannot monitor and display historical occurrences of floods. The limitation hinders its ability to provide extensive analysis of places prone to flooding and the possible pathways impacted by past flood occurrences.

The proposed system employs Google Maps as a mapping tool to aid users in navigating securely, utilizing data supplied by the web application (see Figure 1). The system produces alternate routes and estimates arrival times (ETA) based on Google Maps, depending on the conditions in the deployment locations. These estimations take into consideration the extra delays caused by floods and traffic. Essentially, Google Maps is restricted in its presentation of information and cannot show previous instances of floods. However, this technology improves user navigation by recommending alternate routes based on current conditions and offering updated estimated arrival times, considering delays caused by floods and traffic.

In order to improve user engagement, the system integrates a chatbot powered by OpenAI. Many current applications need more meaningful user involvement, which can worsen critical circumstances. The chatbot is programmed to provide users with general guidance and tailored responses using the collected data. Through artificial intelligence, the system can efficiently monitor flood occurrences and provide recommendations for the most advantageous navigation paths. This integration signifies notable progress in enhancing the ability to withstand floods and reducing their influence on navigation.

Additionally, the system utilizes LoRaWAN-based wireless sensor networks (WSNs) to monitor floods in real time. The chatbot capability is implemented to solve the need for more user involvement in present apps. It supports users by offering responses based on the data that has been collected. Artificial intelligence allows for the creating of intelligent systems that can effectively monitor flood events and optimize navigation routes. The development of real-time flood monitoring and navigation systems using LoRaWAN-based WSNs and artificial intelligence is a significant step forward in improving flood resilience and reducing the impact of floods in a navigation context.

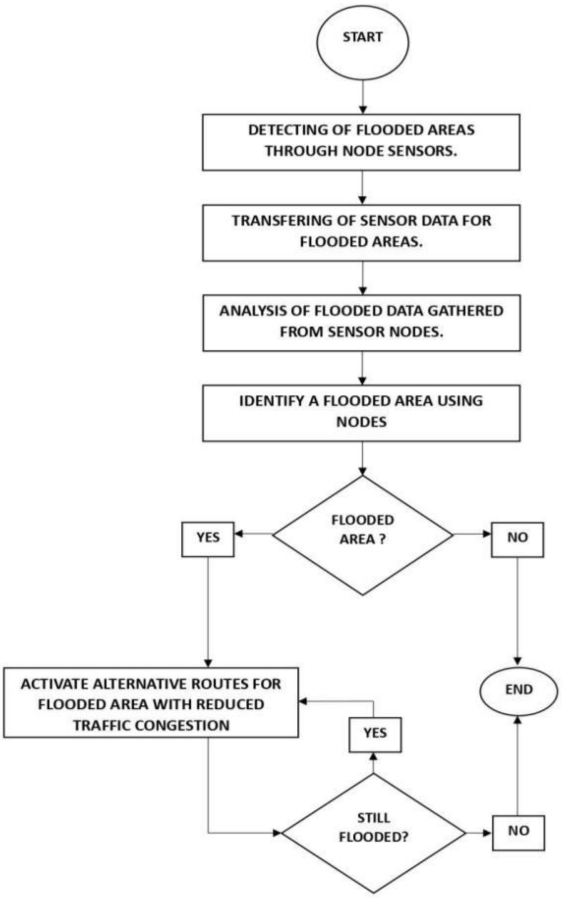

Fig 1. Flood Detection and Navigation flowchart

II. OBJECTIVES

This study aims to develop a flood and road monitoring system for flood-prone areas in Manila. It mainly aims to (1) design and develop a mother node and sub-node for flood and road monitoring systems using ESP32, Arduino Mega, and integrated sensors such as Rain Gauge, Barometric Sensor, and Float switches. (2) develop a Wireless Sensor Network (WSN) using LoRaWAN and Internet of Things (IoT) to integrate the mother nodes and sub-nodes of the system. (3) develop an acquisition and road monitoring system using the YOLO computer vision algorithm. (4) develop a navigation system that provides real-time information on road traffic and flood mapping for alternative routes and estimated time of arrival (ETA) using artificial neural networks by deep learning. (5) develop a web-based application with an integrated chatbot and voice assistant. (6) test and evaluate the efficiency, functionality, and reliability of the flood and road eye navigation system by ISO 9126.

The study aims to design a system comprising a mother node and a sub-node capable of detecting, monitoring, and rerouting users during floods and heavy traffic occurrences within the City of Manila. The mother nodes serve as the primary nodes of the system. They collect various data, including rainfall data using a rain gauge, weather conditions using a barometric sensor, and flood data using float switches. To facilitate communication and data collection from the sub-nodes, a LoRa Adafruit is integrated into the mother node. A GSM module is also incorporated to provide internet connectivity to the mother node, enabling real-time transmission of collected sensor data to the web application.

Conversely, the sub-node has float switches and a LoRa Adafruit to communicate and transmit data to the mother node. The mother node and the sub-node are powered by a solar panel and a battery, ensuring continuous system functionality even during extreme weather conditions. A camera and a Raspberry Pi are also utilized to capture and provide live traffic situations and traffic volume using the YOLO algorithm in the deployment area, which informs the users of the web application about the traffic situation within specified areas in Manila.

The system's web application features a Google Map interface designed to assist users in navigating from one location to another, taking into account both flood and traffic data to provide the best possible route for the users. Additionally, a chatbot utilizing OpenAI and voice recognition is integrated into the web application, enhancing user convenience during interaction with the system. The proponent aims to provide comprehensive flood and traffic monitoring capabilities, ultimately enhancing the safety and efficiency of navigation in flood-prone areas of the city. This innovative approach can empower communities to proactively respond to adverse weather events and alleviate the impacts of floods and traffic congestion, contributing to overall resilience and sustainable development.

III. REVIEW OF RELATED LITERATURE

A study by J.G. Natividad and J.M. Mendez [4] on Flood Monitoring and Early Warning Systems uses an ultrasonic sensor to measure floodwater levels and a GSM module to send alert messages on the water level to concerned agencies. The study employs a stem-mounted float switch with a 0.5 ft increment from 1ft to 4ft instead of an ultrasonic sensor to measure the flood level due to the sensitivity of ultrasonic sensors to environmental factors in the deployment that could alter the correctness of the collected data. The study also utilizes IoT technology using the GSM/GPRS capability of the GSM module to connect the system's microcontroller to the internet and upload all the necessary data collected to the website.

The study by M.L. Roopa et al. [5] about IoT-Based Real-time Weather Prediction systems utilizes a Node MCU, a DHT11 sensor, a barometric sensor, and a rainfall sensor to create a weather station. It contains various sensors and gadgets that work together but in specific ways to transmit proper and accurate data on weather parameters. These can be checked through a mobile and web application. The system proposes an Arduino Mega 2560 for the system's central controller where weather sensors are connected, such as BMP 280 for temperature and air pressure measurement and a Rain gauge for rainfall amount monitoring.

W. Indrasari and L.V. Kadarwati [6] proposed a float equipped with a permanent magnet that serves as an indicator directly exposed to water and is affixed vertically onto support poles. The Hall Effect UGN3503 magnetic sensor senses the position of the permanent magnet within the float. Altitude measurements are derived from the displacement of the permanent magnet within the float caused by water level changes and readings from a vertically mounted ultrasonic sensor.

The proposed system consists of a mother-node and a sub-node that utilize Lora technology. A Lora Adafruit is utilized to communicate and transmit the data from the mother node and sub-node—a study by N.M. Yoeseph [7] about the early warning system architecture consists of several key components: a LoRa node connected to multiple sensors, a LoRa gateway, a LoRaWAN network server, and a database. The LoRa gateway bridges the LoRa wireless network with an IP network, utilizing WiFi and Ethernet connections. The system operates within The Things Network (TTN) infrastructure, leveraging it as the LoRaWAN network server. The LoRa nodes function as the LoRa wireless module responsible for transmitting data gathered from ultrasonic sensors and water flow meters.\

N. Algiriyage [8] The study investigates real-time traffic flow estimation from low-quality surveillance video data. The traffic flow estimation module counts vehicles based on the direction of movement and the class of the vehicle from CCTV video data. The module is divided into three sub-tasks: vehicle detection, vehicle tracking, and traffic flow estimation. The vehicle detection module draws a bounding box around vehicle objects to locate them within a frame. In contrast, the vehicle tracking module tracks the movement of a vehicle object between different frames.

The study by Ruan et al. [9] focuses on the navigation system based on the vehicle terminal, the system's overall framework, and each module's function. According to the study, optimizing the design and application of the Dijkstra algorithm saved the intermediate result storage space in the process of system operation, improved the accuracy of the navigation path, reduced the complexity of the Dijkstra algorithm, and significantly improved the system's efficiency. This previous study met the needs of the user travel map management and updating, supplemented the current traffic information query, query destination, and shortest path planning, and gave travelers absolute convenience.

Most currently used flood warning analysis and notification systems rely on water level sensors and precipitation forecasts and must be capable of providing near real-time and automated flood monitoring and warning signal analysis. The study by Priya and Kala [10] developed an early detection and warning system for floods using a video streaming process, and the warning information is provided if it exceeds a certain threshold.
The previous study about navigation systems improved the Djikstra algorithm. Similar to the study, a custom bidirectional Dijkstra algorithm is used to loop each sensor every time the vehicle passes the route. At the same time, the study for early flood detection and warning systems does not rely only on water level sensors. Integrating a live viewing of traffic conditions and flood levels will provide the most accurate information for users to take precautionary actions.

The study by Sehgal U. et al. [11] provides a foundational understanding of chatbot applications powered by Natural Language Processing (NLP) techniques. The study categorizes chatbots into two types: rule-based and self-learning. They also discuss their application in various industries for customer service and query resolution functions. Rule-based systems are artificial intelligence that follow predefined rules or logic to solve problems. To provide real-time functionality for the Generative Pre-trained Transformer (GPT), it is necessary to establish a connection between the flood monitoring database and the GPT framework, such as OpenAI's GPT. This connection will enable the model to effectively comprehend and respond to user inquiries regarding flood conditions, safety measures, and navigational assistance.

The study by Bhutada S. et al. [12] indicates the potential integration of ChatGPT for the Pharma Industry, designed to interact with people through text or speech. Their goal is to operate with conversational capabilities like human interactions. The chatbot can understand spoken input from the users using the speech to text. Text-to-speech (TTS) technology could also synthesize the chatbot's response to the user through speech. Similarly, the approach for this study is beneficial in aiming for human-like conversation for the chatbot, especially the users, who are drivers, travelers, and commuters, making it more easily accessible.

IV. METHODOLOGY

A. *Hardware Development*

a) *System Design/Architecture*

The system architecture integrates IoT sensors (Float switches, barometric sensors, and rain gauges) as input devices for sub-nodes and mother nodes. Arduino Mega processes data from these sensors, while LoRa Transceivers facilitate inter-node communication. The sub-node's transceiver transmits data to the mother-node. A GSM Module sends the collected data to the cloud, enhancing accessibility. The ultimate output is delivered through a user-friendly web application, offering real-time insights into environmental variables. This concise and interconnected system efficiently captures, processes, and presents critical data, providing stakeholders with a comprehensive view of float levels, barometric pressure, and rainfall amounts.

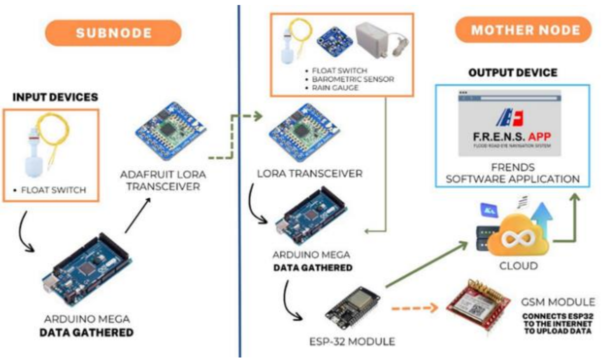

Fig 2 Mother Node and Sub node System Architecture

b) *Data Collection and Transmission*

The initial phase of the proposed system, as illustrated in Figure 3.4, involves deploying IoT sensors at both nodes for data acquisition. These sensors, positioned strategically, enable the Arduino to interpret incoming analog data. Data collected at the Sub-Node is transmitted to the Mother Node via the Adafruit LoRa communication protocol. Upon reaching the Mother Node, the data is transmitted to the ESP32 Microcontroller through the GSM Module for internet connectivity. This facilitates seamless data transmission to a designated database for storage and analysis. Including the GSM Module ensures adaptability to various environmental conditions and locations. A Solar Panel and Solar Charge Controller are integrated to enhance sustainability, providing a continuous and reliable power source. Additionally, a user-friendly Web Application allows stakeholders to access real-time updates on critical parameters such as flood levels, rainfall, temperature, and atmospheric pressure, making it a comprehensive tool for monitoring dynamic environmental conditions captured by the sensors.

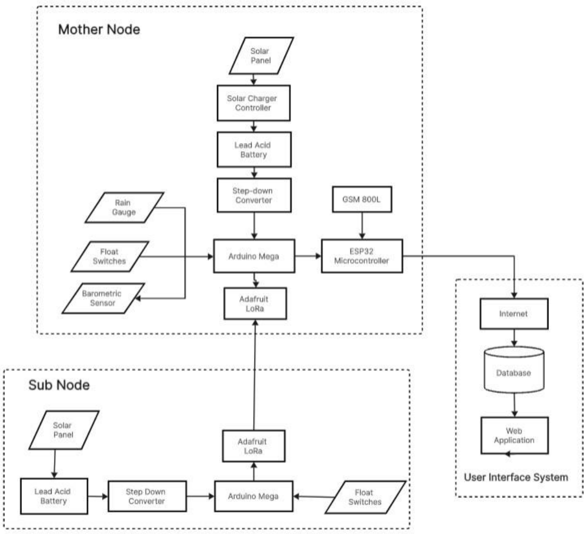

Fig 3 Block Diagram

c) System Hardware Model

Figures 4 to 7 present a detailed illustration of the external hardware model of the Mother Node and Sub Node, highlighting its essential components and design complexity. With a height of 12 ft, the Mother Node and Sub Node feature a dedicated component box for data transmission equipment, ensuring organized and efficient operation. Its power system incorporates a lead-acid battery and solar panel, providing reliable and sustainable energy. Moreover, the Mother Node has integrated essential input devices such as float switches, rain gauges, and a barometric sensor, enhancing the system's functionality for flood and road monitoring applications. This comprehensive hardware model significantly contributes to the field, offering insights into the design and implementation of robust environmental monitoring systems.

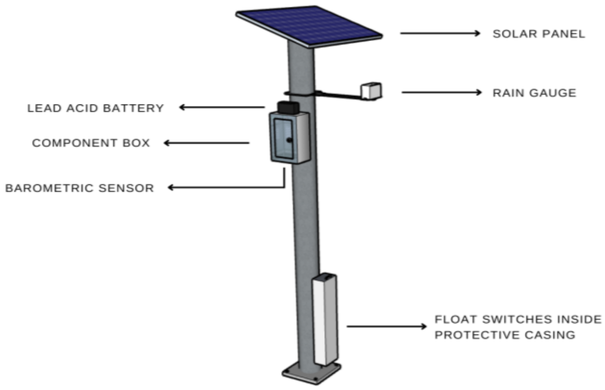

Fig 4 Mother Node

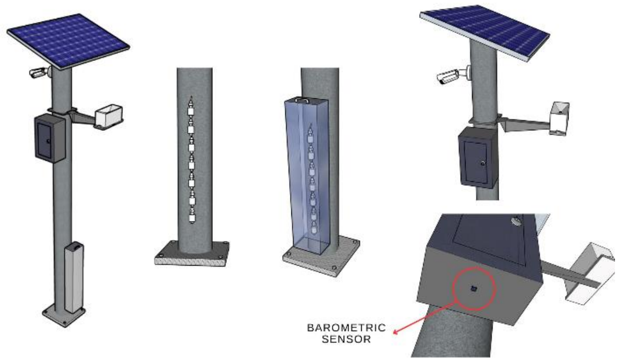

Fig 5 Mother Node Parts

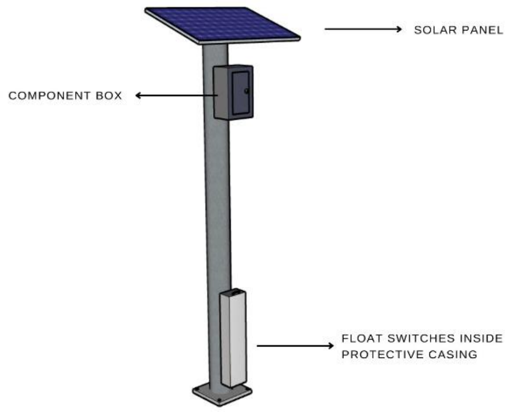

Fig 6 Sub Node

Fig 7 Sub Node Parts

d) Component Box – Mother Node

The component box of the mother node is primarily comprised of components capable of collecting, processing, and uploading data inputs from the wireless sensor network, such as the rain gauge, float switches, and the barometric pressure sensor. The mother node's enclosure components include an Arduino Mega, ESP32, GSM Module, LoRa Adafruit, solar charge controller, and a step-down converter.

Arduino Mega is tasked with gathering all data inputs from the wireless sensor networks. Conversely, the ESP32, empowered by the GSM module, collects and uploads all the data collected by the Arduino Mega to the database. The LoRa Adafruit functions as the principal communication link between the mother node and subnode, facilitating the wireless transfer of data collected by the subnode using the LoRa radio frequency capability.

The solar charge controller plays a crucial role in managing the energy collected by the solar panel, preventing overcharging that may damage the system. Additionally, the step-down converter is responsible for converting the 12-volt power input to 5 volts, ensuring compatibility with the components of the system.

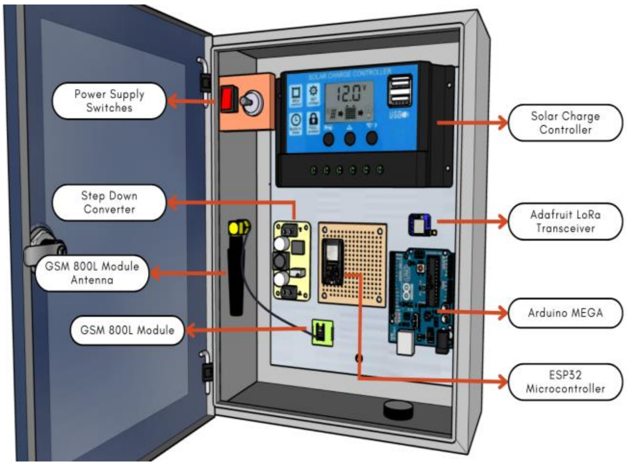

Fig 8 Component Box of Mother Node

e) Component Box – Sub Node

The component box of the sub-node comprises the Arduino Mega, LoRa Adafruit, solar charge controller, and step-down converter. The Arduino Mega collects data from the float switches, and subsequently, the LoRa Adafruit transmits the acquired data to the mother node. The solar charge controller regulates the energy collected by the system's solar panel to prevent overcharging, mitigating potential damage. Additionally, the step-down converter plays a pivotal role in converting the 12-volt power input to 5-volt. This combination of components ensures the efficient functioning of the sub-node, allowing for precise data collection, transmission, and power management within the specified voltage parameters.

Fig 9 Component Box of Sub Node

f) Flood Gauge and Deployment Area

The flood gauge system utilizes float switches to measure flood levels, with each switch representing a certain flood height. The "Low" level, often known as the Half-Knee Level, permits the passage of all vehicles within a one-foot range. The "Medium" level at a depth of 2 feet deters the use of light and moderate vehicles. Heavy trucks are not allowed at the "High" level while it is at a height of 3 feet. The 4-foot level and higher (at waist level) impose limitations on all types of vehicles due to safety hazards.

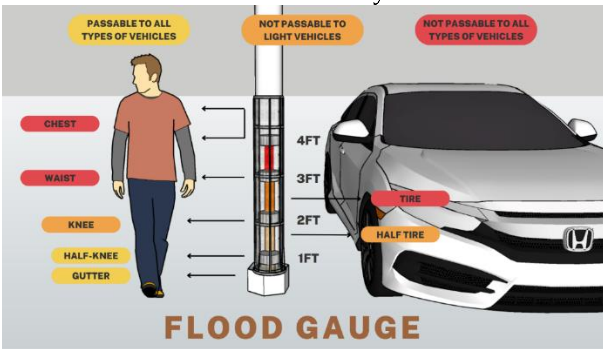

Fig 10 System Flood Gauge

The study concentrates on District 5 of Manila, implementing Flood Monitoring Systems in regions prone to flooding, such as San Marcelino – Ayala Boulevard, Pedro Gil, Ayala Boulevard – Taft Avenue, UN Avenue, Maria Orosa - Kalaw, and Padre Faura. High-traffic zones like Padre Burgos, UN Avenue - Roxas Boulevard, and UN Avenue - Romualdez have been equipped with Traffic and Road Monitoring Systems.

Fig 11 Proposed Deployment Area

B. *Software Development*

a) *System Architecture*

Figure 12 shows a comprehensive system design for a web application focused on real-time flood monitoring and navigation systems for road users. It seamlessly integrates front-end and back-end components to enable user access to real-time information and facilitate efficient administration. Front-end components utilize modern web technologies to provide an intuitive interface, while back-end components handle data processing and storage, integrating with a database management system. An API-based integration layer ensured seamless interaction among the system's components. The platform offered real-time data visualization and interactive features for users alongside management tools for administrators. It was fortified with security measures to safeguard information and was built with scalability and performance enhancements to ensure efficient operation. This architecture laid a solid groundwork for an all-encompassing solution to monitor floods and navigate roads effectively.

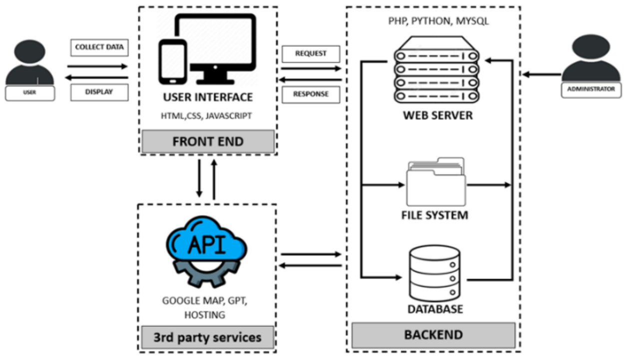

Fig 12. System Architecture of the Software System

b) *Estimated time of Arrival and Alternative Route*

The block diagram (Figure 14) depicts a simplified procedure for acquiring the Estimated Time of Arrival (ETA) by integrating Google Maps and optimizing IoT sensors and cameras. The system gathers vehicle traffic and water level data from various sensors, including cameras, and sends it to a central database.

Google Maps uses route analysis to verify the presence of specified nodes along the user's chosen path. Without nodes, it creates a route and calculates an estimated arrival time (ETA). When the system encounters nodes, it utilizes the Google Maps Geocoding API to collect node-specific data.

Computer vision and deep learning techniques enhance estimated arrival time (ETA) forecasts by analyzing current traffic, flood conditions, and node data. Users are provided with an optimized route that considers many factors, such as node impacts, resulting in a more accurate estimated arrival time (ETA). This is achieved by efficiently combining Google Maps and sensor data, allowing for individualized route calculations.

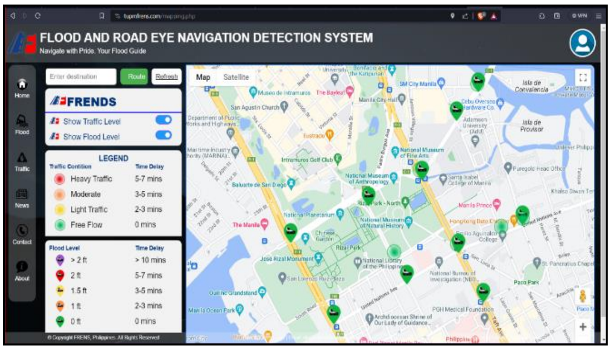

Fig. 13. Mapping of the System

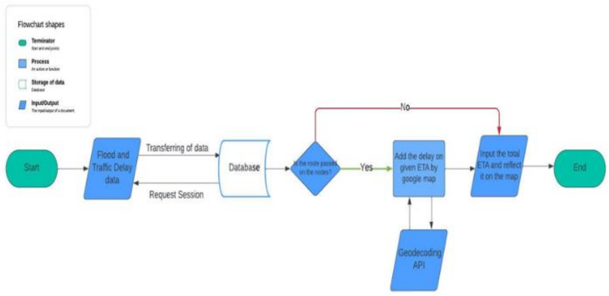

Fig. 14. Flood and road detection Flowchart

c) *Dataset connection to OpenAI (blue)*

This diagram shows (See figure X) how the back end works in the chatbot. There are two parts to this process: information retrieval and data pre-processing. First, the system collects all information from the datasets, flood sensors, traffic sensors, and trained information through the database, then injects it into OpenAI as a prompt to be temporarily remembered by its engine. This data will be stored as a prompt, so when the user asks a question, the prompt will be sent before the question; hence, the chatbot will understand the user's queries.

Figure 16 shows how the chatbot decides where it will retrieve the information. This process is called AI model interaction; the chatbot will check if the user queries are in line with the pre-trained datasets, which are all data in the website, and if it is not, the chatbot will decide to collect the information to general pre-trained information and give the output back to the users.

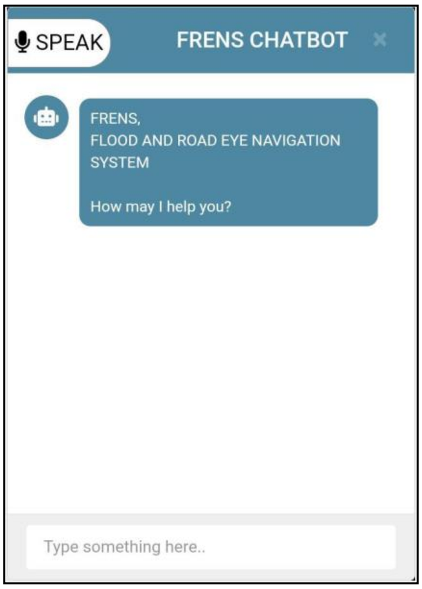

Fig 15 Chatbot Interface

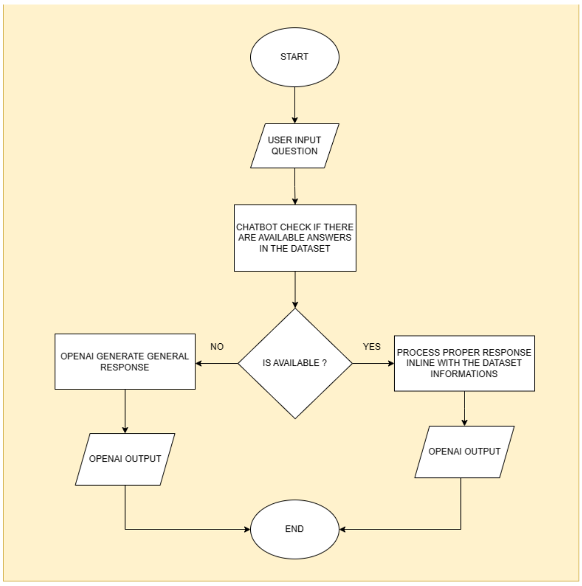

Fig 16 Process and Output response flowchart

IV. RESULTS AND DISCUSSION

A. *System Integration and Testing*

TABLE I. INITIAL DATA GATHERING ON SITE

Temp	Pressure	Date	Time	Rain amount D	Rain amount E	Rain amount F	Rain Amount G	Rain amount H	Rain amount J
36.43	103457.15	May 10, 2024	9:05 PM	0	0	0	0	0	0
36.45	100324.54	May 13, 2024	11:58 PM	0	0	0	0	0	0
32.52	101453.06	May 14, 2024	3:55 AM	0	0	0	0	0	0
31.67	101324.56	May 15, 2024	8:15 PM	1.20	3.15	0.90	4.50	3.50	2.10
30.80	100345.45	May 16, 2024	9:49 AM	0	0	0	0	0	0
29.50	101802.25	May 17, 2024	11:04 AM	0	0	0	0	0	0
31.52	100476.32	May 18, 2024	12:01 AM	1.02	1.80	1.35	0.90	1.35	1.90
31.98	101976.92	May 19, 2024	12:01 AM	0.90	1.50	0.60	0.30	0.45	0.30
31.03	101076.94	May 20, 2024	11:59 PM	0	0	0	0	0	0
31.15	100258.47	May 21, 2024	11:59 PM	0	0	0	0	0	0
36.45	100920.45	May 22, 2024	11:59 PM	0	0	0	0	0	0
36.25	101485.95	May 23, 2024	11:59 PM	0	0	0	0	0	0
35.15	101235.25	May 24, 2024	11:59 PM	0.84	0	0.54	0	0	0
35.12	100258.14	May 25, 2024	11:58 PM	1.2	0.93	0.30	1.35	1.80	0.90
26.78	101345.14	May 26, 2024	3:04 AM	0	0	0	0	0	0

The table below depicts initial tests of the mother node and subnode for real-time data transmission displayed on a website. Researchers evaluate the nodes' performance under varied conditions, emphasizing their ability to transmit data swiftly. Additionally, the tests determine the maximum transmission range, considering obstacles in the line of sight. These figures offer insights for optimizing node deployment in real-world scenarios, ensuring reliable data transmission even in the presence of obstacles.

B. *Rerouting and ETA Accuracy*

TABLE II: ETA GENERATED FROM FRENS APP AND WAZE APP

LOCATION	FRENS ROUTING			WAZE ROUTING			DISTANCE (kilometers)
	ETA (minutes)	TRAVEL DURATION		ETA (minutes)	TRAVEL DURATION		
		Start	End		Start	End	
TUP to Philippine General Hospital	5	4:02 pm	4:09 pm	4	4:02 pm	4:07 pm	1.3
Philippine General Hospital to Manila Doctors Hospital	6	3:10 pm	3:18 pm	4	3:10 pm	3:14 pm	1.1
Manila Doctors Hospital to Manila City Hall	7	3:16 pm	3:24 pm	4	3:16 pm	3:21pm	1.3
Manila City Hall to Padre Burgos	7	3:25 pm	3:33 pm	5	3:25 pm	3:30 pm	1.4
Padre Burgos to Roxas Blvd Manila	5	3:32 pm	3:39 pm	3	3:32 pm	3:35 pm	0.8
Roxas Blvd to Supreme Court Taft Ave Manila	6	3:35 pm	3:44 pm	4	3:35 pm	3:39 pm	1.3
Supreme Court Taft Ave Manila to LRT-1 Pedro Gil Station	7	3:42 pm	3:51 pm	5	3:42 pm	3:47 pm	1.3
LRT-1 Pedro Gil Station to Astral Tower Taft Ave Manila	5	3:48 pm	3:54 pm	2	3:48 pm	3:50 pm	0.5
Astral Tower Taft Ave Manila to BPI UN Avenue	4	3:53 pm	3:56 pm	3	3:53 pm	3:53 pm	0.9
BPI UN Avenue to SM Manila	6	3:54 pm	4:02 pm	5	3:54 pm	3:59 pm	1.5

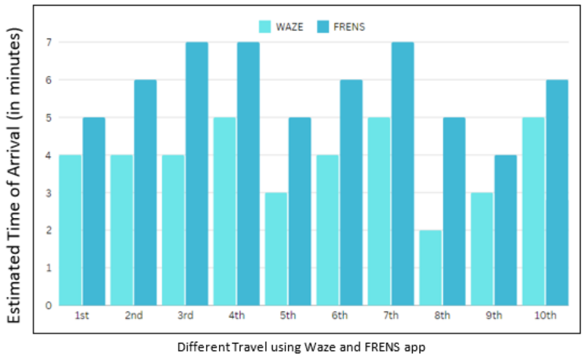

Fig 17 Waze VS. FRENS App

The gathered data were used to differentiate the FRENS's estimated arrival from that of the existing navigation application. The researchers determined the accuracy of the ETA generated by the system to the actual arrival time. The Mean Absolute Error(MAE) was utilized to measure the average magnitude of errors between the estimated and actual arrival time.

C. Chatbot Dataset

The dataset should be in text form, containing descriptive information similar to how one would teach something to another person. This approach ensures that the chatbot comprehends the material and provides responses based on its understanding.

id	information	date_added
1	FRENS is all about traffic and flood information around ermita manila to ensure the safety of drivers.	2024-02-29 17:26:36
12	FRENS own by students from TUP Manila.	2024-02-29 19:10:48
36	The purpose of our website is to provide real-time information and updates about flood conditions and road traffic to help drivers make informed decisions about their travel routes. By monitoring these critical factors, our goal is to enhance safety, reduce travel time, and minimize the risks associated with flooding and heavy traffic. Our platform integrates sensors and data analytics to monitor flood levels and traffic patterns across various location that are around TUP Manila. This data is then processed and displayed on our website in a user-friendly format, allowing users to quickly identify areas to avoid.	2024-05-24 16:23:42
37	To use our website to check road conditions, follow these detailed steps: 1. Access the Website: Open your web browser and navigate to our website by entering the URL provided. Ensure you have a stable internet connection for the best experience. 2. Home Page Overview: Upon landing on the home page, you will see an interactive map. This map is the central feature for monitoring road conditions. Additionally, you will see sections and menus that offer more detailed information and options. 3. Interactive Map: The interactive map is designed to provide real-time updates on road conditions. 4. Search destination: After searching your desired destination, it will automatically select your location and to your destination. 5. Indicators: The indicators such as the flood and traffic nodes will be avoided if the location are flooded and heavy traffic. 6. Live viewing: If you want to see the live monitoring you can select the traffic lines to see the live video feed.	2024-05-24 16:23:49
38	Currently, there is no mobile app version of our website. However, our development team is working diligently to enhance the website responsiveness to mobile browsers. This ensures that you can still access all the essential features and real-time updates on road and flood conditions directly from your mobile device.	2024-05-24 16:23:53
39	How to Contact the Developers for Feedback 1. Navigate to the "Contact" Tab: Visit our website and locate the "Contact" tab or section in the navigation menu. 2. Fill Out the Contact Form: Once on the "Contact" page, you will find a form to fill out with your details. Enter your name, email address, and any other required information. 3. Fill up "Subject" for Developers: Write a subject that specify your message 4. Compose Your Message: In the message box, provide your feedback, suggestions, or any issues you would like to address with the developers. Be as detailed as possible to ensure the developers understand your feedback clearly. 5. Submit Your Message: Once you have completed the form, click the "Submit" or "Send" button to send your message to the developers. 6. Confirmation Email: After the developers have reviewed your message, they will respond to you via email. Make sure to check your email inbox (and possibly your spam folder) for their response.	2024-05-24 16:24:09
40	You can reach our support team via email at frendsapp@yahoo.com. Additionally, you can contact us by phone at +639169638514 for further assistance. Feel free to reach out to us with any questions, concerns, or feedback you may have. We are here to help!	2024-05-24 16:24:14
41	To check if an area is flooded, utilize our website interactive map and flood tab: Interactive Map: Indicator Colors: Green indicates areas that are not flooded. By examining the maps color-coded indicators, you can quickly identify regions unaffected by flooding. Flood Tab: Vehicle Compatibility: Access the flood tab to determine if your chosen vehicle can safely navigate through the flooded area. This is automatically avoided when you navigate using the map.	2024-05-24 16:24:58
42	How do I know if my vehicles is not passable? You can determine if your vehicle is not passable based on its vehicle type classification: Vehicle Types and Flood Depth Tolerance: 1. Type A - All types of trucks: • Not passable if flood depth exceeds 2 feet. 2. Type B - Bus, van type cars, and Jeepneys: • Not passable if flood depth exceeds 2 feet. 3. Type C - Taxi type cars and big bikes: • Not passable if flood depth exceeds 1.5 feet. 4. Type D - Motorcycles, electronics tricycles, and bicycles: • Not passable if flood depth exceeds 1 foot. By identifying your vehicle type and referring to the corresponding flood depth tolerance, you can make informed decisions regarding your ability to navigate flooded areas safely.	2024-05-24 16:26:39
43	What areas are most affected by floods? Currently, the areas most affected by floods around TUP Manila, as analyzed and researched by the Frens team, include: • Ayala - San Marcelino Intersection • San Marcelino Street • Ayala - Finance Rd - Taft Ave Intersection • UN Ave. Romualdez St. Intersection • UN Ave. Taft Ave Intersection • Padre Faura St. - Taft Ave. Intersection • Pedro Gil St. Taft Ave. Intersection • Padre Burgos - Gen Luna St. Ma. Orosa St. Int. • Padre Burgos Avenue • UN Ave. - Roxas Blvd East Rd Intersection • UN Ave. - Ma. Orosa St. Intersection	2024-05-24 16:27:02

Page number: 1/2

Fig 18. Chatbot Dataset

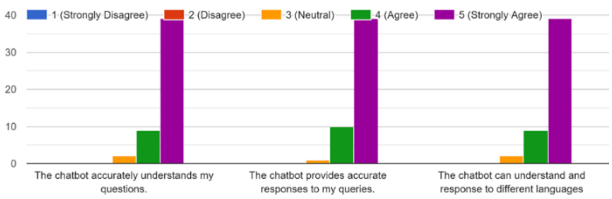

Fig 19. Chatbot Response Accuracy

This part of the survey assesses how accurately the chatbot understands and responds to user queries. The survey results indicated a high level of satisfaction, with a computed score exceeding 90%. This score signifies that the majority of users found the chatbot's responses to be accurate and satisfactory.

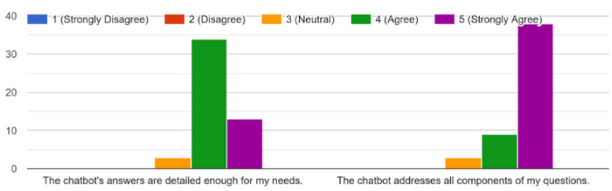

Fig 20 Chatbot Completeness

This survey assessed whether the chatbot's responses were detailed and thorough. The survey results showed high satisfaction, with a computed score exceeding 90%. This score indicates that the majority of users found the chatbot's responses to be complete and satisfactory.

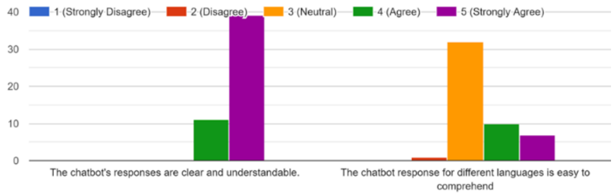

Fig 21 Chatbot Clarity

This survey specifically focused on evaluating the clarity and understandability of the chatbot's responses. The survey results revealed a perfect satisfaction score of 100% for the clarity of the chatbot's responses in the primary language. This indicates that users found the chatbot's responses exceptionally clear and easily understood.

However, when evaluating the chatbot's performance in different languages or dialects, the satisfaction score dropped significantly to 34%. This lower score highlights a critical area for improvement.

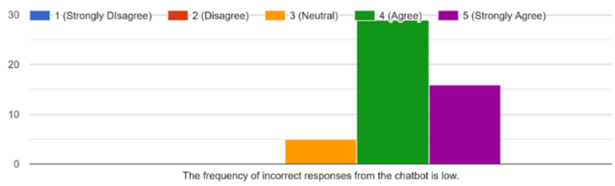

Fig 22 Error Rate

In this part of the survey, we assessed the frequency of malfunctions experienced by users when interacting with the chatbot. According to the survey results, the satisfaction score was impressively high, exceeding 90%. This high satisfaction score suggests that the chatbot's error rate is relatively low, indicating that it performs reliably and meets user expectations in most cases. The remaining 10% may be attributed to the clarity of the chatbot, as most respondents use different languages when asking queries.

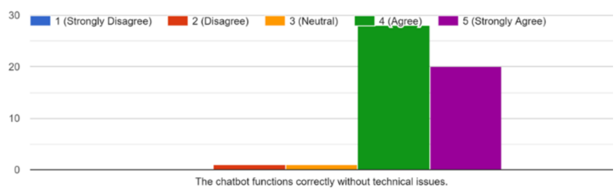

Fig 23 Chatbot Functionality Test

This part of the survey assessed whether the chatbot operates smoothly without technical issues. Based on the survey, the satisfied computed score was above 90%, which signifies good chatbot functionality.

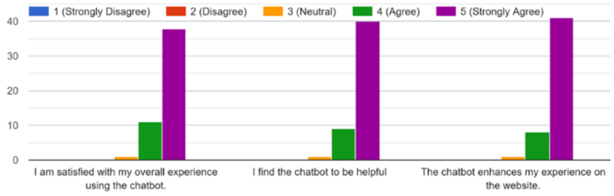

Fig 24 Chatbot User Satisfaction

This survey focused on gauging overall satisfaction with the chatbot's performance and its impact on the website experience. According to the survey, the satisfaction score was above 90%, indicating that using the chatbot is very helpful. Respondents are also amazed that technology like this can now be integrated into our website. Some comments suggest that the advancement of this technology might replace customer service.

V. CONCLUSION

The successful implementation of this vehicle detection system opens possibilities for future enhancements, such as integrating real-time traffic predictive analytics. By combining the captured vehicle information with dynamic routing algorithms, users can enjoy a seamless and efficient navigation experience. This integration would enable the system to automatically calculate and suggest optimal routes, reducing travel time and congestion and minimizing potential delays. Moreover, utilizing an online database for vehicle tracking lays the groundwork for incorporating additional features, such as pre-registered route preferences, and enhancing the system's capability to handle a large volume of traffic efficiently.

By integrating these two features, the application enables users to utilize the functionalities of the FRENS app. With this application, users can navigate without the risk of being stuck by flooded roads with 86.9% accuracy of rerouting. Furthermore, the application stores historical data on past traffic and flood conditions, providing users with valuable insights for their specific needs. The app offers chatbot functionalities, allowing users to ask questions about the newly developed application. Hence, Analyzing the satisfaction scores from the survey, almost all questions received exemplary ratings, resulting in 98% overall satisfactory score, except for the chatbot's responses in different languages for text-to-speech. The main concern on text-to-speech was that the output was directly translated without fully understanding the language, resulting in poor grammar, choice of words, and sentence structure.

Training the model better to understand the language, including its characteristics and slang, can improve this. Even though the chatbot was continuing to improve, it is fully functional and, even in its current improving stage, is found to be very helpful. It enhances the user experience, mainly since the web application features are still new. Overall, the chatbot effectively accommodates all user queries.

VI. RECOMMENDATION

The Flood and Road Eye Navigation System (FRENS) has demonstrated significant potential in providing real-time flood and traffic monitoring for Manila. This research recommendation aims to explore and propose future endeavors and innovations to enhance and expand FRENS.

1. Implement a more adaptable mapping system that provides real-time Estimated Time of Arrival (ETA) and rerouting options to address unexpected situations on the road effectively.
2. Incorporate a feature that integrates number coding information, enabling users to identify the best available routes based on the coding restrictions applicable during their travel time.
3. Develop a dedicated mobile application compatible with Android and iOS platforms, providing users with a seamless and accessible experience while utilizing the flood and road monitoring system on their mobile devices.
4. Explore the utilization of satellite APIs to enhance the precision of routing plans, particularly in addressing flood and traffic conditions on the map.

REFERENCES

[1] Chen, S.P. 1990. Remote Sensing Analysis in Geoscience. Beijing Mapping Press, Beijing, China

[2] "Climate Change: Global Sea Level," NOAA Climate.gov, 2021

[3] B. Popa, “Google Maps Rival Creates System To Detect Flooded Roads and Send Warnings to Drivers,” autoevolution, Jul. 17, 2023. https://www.autoevolution.com/news/google-maps-rival-creates-system-to-detect-flooded-roads-and-send-warnings-to-drivers-218203.html

[4] J. G. Natividad and J. M. Mendez, “Flood Monitoring and Early Warning System Using Ultrasonic Sensor,” IOP Conference Series: Materials Science and Engineering, vol. 325, pp. 012020–012020, Mar. 2018, doi: https://doi.org/10.1088/1757-899x/325/1/012020.

[5] L. Karlo et al., “Real Time Flood Detection, Alarm and Monitoring System Using Image Processing and Multiple Linear Regression,” 2022. Available: https://www.dlsu.edu.ph/wp-content/uploads/pdf/research/journals/jciea/vol-7-1/2tolentino.pdf

[6] W Indrasari and L. V. Kadarwati, “Prototype of water level monitoring system using magnetic sensor and ultrasonic based on Arduino Mega 2560,” Journal of Physics: Conference Series, vol. 2193, no. 1, pp. 012052–012052, Feb. 2022, doi: https://doi.org/10.1088/1742-6596/2193/1/012052.

[7] N.M. Yoeseph, F.A. Purnomo, R. Hartono, and None Nuryani, “Lora-based IoT sensor node for Real-time Flood Early Warning System,” IOP Conference Series: Earth and Environmental Science, vol. 986, no. 1, pp. 012060–012060, Feb. 2022, doi: https://doi.org/10.1088/1755-1315/986/1/012060.

[8] N. Algiriyage et al., “Towards Real-time Traffic Flow Estimation using YOLO and SORT from Surveillance Video Footage.” Accessed: Apr. 04, 2024. [Online]. Available: https://idl.iscram.org/files/nilanialgiriyage/2021/2311_NilaniAlgiriyage_etal2021.pdf

[9] Ruan, C., Luo, J., & Wu, Y. C. (2014). Map navigation system based on optimal Dijkstra algorithm. CCIS2014. https://doi.org/10.1109/ccis.2014.7175798

[10] K. P. Menon and L. Kala, “Video surveillance system for realtime flood detection and mobile app for flood alert,” in 2017 International Conference on Computing Methodologies and Communication (ICCMC), 2017, doi: 10.1109/ICCMC.2017.8282518.

[11] U. Sehgal and S. Bhardwaj, "Building a Chatbot using Natural Language Processing," 2023 Second International Conference on Informatics (ICI), Noida, India, 2023, pp. 1-5, doi: 10.1109/ICI60088.2023.10421577.

[12] S. Bhutada, S. S. D. Uppala, S. Nagavajyula and T. Pottigari, "C2-PILS: A Chatbot using Chat-GPT for Pharma Industry and Life Sciences," 2023 International Conference on Advanced Computing Technologies and Applications (ICACTA), Mumbai, India, 2023, pp. 1-6, doi: 10.1109/ICACTA58201.2023.10392641.