

1 DENGUEWATCH: A SYSTEM FOR REAL-TIME  
2 DENGUE MONITORING AND FORECASTING IN ILOILO  
3 PROVINCE

4 A Special Problem Proposal  
5 Presented to  
6 the Faculty of the Division of Physical Sciences and Mathematics  
7 College of Arts and Sciences  
8 University of the Philippines Visayas  
9 Miag-ao, Iloilo

10 In Partial Fulfillment  
11 of the Requirements for the Degree of  
12 Bachelor of Science in Computer Science by

13 AMODIA, Kurt Matthew A.  
14 BULAONG, Glen Andrew C.  
15 ELIPAN, Carl Benedict L.

16 Francis D. DIMZON  
17 Adviser

18 April 26, 2025

## Abstract

Dengue fever remains a significant public health concern in the Philippines, with cases rising dramatically in recent years. Nationwide outbreaks have placed immense strain on healthcare systems, underscoring the need for innovative approaches to surveillance and response. In Iloilo City, this national trend was reflected in a significant surge, with the Iloilo Provincial Health Office reporting 4,585 cases and 10 fatalities as of August 10, 2023—a 319% increase from the previous year’s 1,095 cases and one death. This research focused on developing a centralized system for monitoring and forecasting dengue trends in Iloilo City, incorporating graphical visualizations such as heatmaps, trends, and historical graphs. The study explored the application of artificial intelligence (AI) in dengue prediction, utilizing deep learning models. The performance of the Long Short-Term Memory (LSTM) model was compared with traditional statistical methods, including non-seasonal and seasonal Autoregressive Integrated Moving Average (ARIMA) models and the Kalman Filter. Evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Square Error (RMSE) were used to identify the most effective model for integration into the system. Forecasting was based on climate variables such as temperature, rainfall, relative humidity, and previous monthly case counts. The LSTM model emerged as the best performer with an RMSE of 16.15, demonstrating its suitability for time-series predictions, while the Kalman Filter showed the poorest performance with an RMSE of 38.40. By integrating predictive analytics with real-time data visualization, the proposed system aims to support public health agencies, such as the Department of Health (DOH), by providing actionable insights for proactive intervention strategies. This AI-driven solution enhances traditional outbreak reporting systems by enabling timely, data-informed decisions to mitigate the impact of dengue in the region.

**Keywords:** ARIMA, artificial intelligence, dengue prediction, LSTM, Kalman Filter, deep learning, climate variables, public health, outbreak mitigation

# Contents

48	<b>1 Introduction</b>	<b>1</b>
49	1.1 Overview . . . . .	1
50	1.2 Problem Statement . . . . .	2
51	1.3 Research Objectives . . . . .	2
52	1.3.1 General Objective . . . . .	2
53	1.3.2 Specific Objectives . . . . .	2
54	1.4 Scope and Limitations of the Research . . . . .	3
55	1.5 Significance of the Research . . . . .	4
56	<b>2 Review of Related Literature</b>	<b>6</b>
57	2.1 Dengue . . . . .	6
58	2.2 Outbreak Definition . . . . .	6
59	2.3 Existing System: RabDash DC . . . . .	7
60	2.4 Deep Learning . . . . .	8
61	2.5 Kalman Filter . . . . .	8
62	2.6 Weather Data . . . . .	9
63	2.7 Chapter Summary . . . . .	9

64	<b>3 Research Methodology</b>	<b>10</b>
65	3.1 Research Activities . . . . .	11
66	3.1.1 Gather Dengue Data and Climate Data to Create a Com-	
67	plete Dataset for Forecasting . . . . .	11
68	3.1.2 Develop and Evaluate Deep Learning Models for Dengue	
69	Case Forecasting . . . . .	13
70	3.1.3 Integrate the Predictive Model into a Web-Based Data An-	
71	alytics Dashboard . . . . .	17
72	3.1.4 System Development Framework . . . . .	17
73	3.2 Development Tools . . . . .	19
74	3.2.1 Software . . . . .	19
75	3.2.2 Hardware . . . . .	20
76	3.2.3 Packages . . . . .	20
77	3.3 Application Requirements . . . . .	23
78	3.3.1 Backend Requirements . . . . .	23
79	3.3.2 User Interface Requirements . . . . .	24
80	3.3.3 Security and Validation Requirements . . . . .	27
81	3.4 Calendar of Activities . . . . .	27
82	<b>4 Results and Discussion/System Prototype</b>	<b>29</b>
83	4.1 Data Gathering . . . . .	29
84	4.2 Exploratory Data Analysis . . . . .	30
85	4.3 Outbreak Detection . . . . .	34
86	4.4 Model Training Results . . . . .	35
87	4.4.1 LSTM Model . . . . .	35

88	4.4.2	ARIMA Model . . . . .	37
89	4.4.3	Seasonal ARIMA (SARIMA) Model . . . . .	38
90	4.4.4	Kalman Filter Model . . . . .	39
91	4.5	System Prototype . . . . .	41
92	4.5.1	Home Page . . . . .	41
93	4.5.2	User Registration, Login, and Authentication . . . . .	42
94	4.5.3	Personnel Interface . . . . .	43
95	4.6	User Testing . . . . .	46
96		<b>References</b>	<b>48</b>
97		<b>A Appendix Title</b>	<b>51</b>
98		<b>B Resource Persons</b>	<b>52</b>

## 99 List of Figures

100	3.1	Workflow for forecasting the number of weekly dengue cases . . .	10
101	3.2	Testing Process for DengueWatch . . . . .	18
102	3.3	Entity-Relationship Database Schema Hybrid Diagram for DengueDash	
103		Database Structure . . . . .	23
104	3.4	Use Case Diagram for DRU Admin . . . . .	24
105	3.5	Use Case Diagram for Surveillance Unit Admin . . . . .	25
106	3.6	Use Case Diagram for Encoder . . . . .	26
107	4.1	Snippet of the Combined Dataset . . . . .	30
108	4.2	Data Contents . . . . .	30
109	4.3	Dataset Statistics . . . . .	31
110	4.4	Distribution of Dengue Cases . . . . .	31
111	4.5	Frequency of Dengue Cases in Different Intervals . . . . .	32
112	4.6	Trend of Dengue Cases . . . . .	33
113	4.7	Ranking of Correlations . . . . .	33
114	4.8	Ranking of Correlations (with lagged effects) . . . . .	34
115	4.9	Training Folds - Window Size 5 . . . . .	36
116	4.10	Testing Folds - Window Size 5 . . . . .	36

117	4.11 ARIMA Prediction Results for Test Set . . . . .	37
118	4.12 Seasonal ARIMA Prediction Results for Test Set . . . . .	38
119	4.13 Kalman Filter Prediction Results for Test Set . . . . .	40
120	4.14 Home Page . . . . .	41
121	4.15 Sign Up Page . . . . .	42
122	4.16 Login Page . . . . .	43
123	4.17 First Part of Case Report Form . . . . .	44
124	4.18 Second Part of Case Report Form . . . . .	44
125	4.19 Dengue Reports . . . . .	45
126	4.20 Detailed Case Report . . . . .	46
127	A.1 LSTM Prediction Results for Test Set . . . . .	51

# 128 List of Tables

<small>129</small>	3.1	Timetable of Activities for 2024 . . . . .	28
<small>130</small>	3.2	Timetable of Activities for 2025 . . . . .	28
<small>131</small>	4.1	Comparison of different models for dengue prediction . . . . .	35
<small>132</small>	4.2	Comparison of Window Sizes . . . . .	35
<small>133</small>	4.3	Comparison of SARIMA performance for each fold . . . . .	39
<small>134</small>	4.4	Comparison of KF-LSTM performance for each fold . . . . .	40
<small>135</small>	4.5	Computed System Usability Scores per Participant . . . . .	47



# Chapter 1

## Introduction

### 1.1 Overview

From 2020 to 2022, dengue cases declined due to reduced surveillance during the COVID-19 pandemic (WHO, 2023), but cases surged in 2023 as restrictions were lifted. This year saw an increase in dengue outbreaks worldwide, with over five million cases and more than 5,000 deaths reported in over 80 countries (Bosano, 2023). Dengue is endemic in the Philippines, leading to longer and more widespread seasonal outbreaks. Globally, dengue infections have increased significantly, posing a major public health challenge. The World Health Organization reported a ten-fold rise in cases between 2000 and 2019, with a peak in 2019 when the disease spread across 129 countries (WHO, 2024).

Iloilo City and Province are intensifying efforts to curb the rising dengue cases (Lena, 2024). As of August 10, 2023, the Iloilo Provincial Health Office recorded 4,585 cases and 10 deaths, a 319% increase from last year's 1,095 cases and one death. Governor Arthur Defensor Jr. confirmed that the province has reached the dengue outbreak threshold based on Department of Health (DOH). Local government units (LGUs) have been informed, and the province's disaster management office is on blue alert, indicating disaster mode. (Perla, 2024)

In Iloilo City, 649 dengue cases were recorded this year 2024, with two deaths. Cases cluster in 40 out of 180 barangays, meaning multiple cases are being reported in these areas over several weeks. The city's health officer, Dr. Roland Jay Fortuna, reported high utilization of non-COVID-19 hospital beds, reaching over 76%, prompting concerns about hospital capacity.

160 This study explores the monitoring and forecasting of dengue outbreaks by an-  
161 alyzing key factors such as temperature, relative humidity, and historical dengue  
162 cases, using different models. The findings aim to provide an advanced, AI-driven  
163 alternative for dengue prevention and control, targeting agencies like the Depart-  
164 ment of Health (DOH). By aligning with the national AI Roadmap, particularly  
165 in Iloilo City, this research aspires to improve outbreak responses through cutting-  
166 edge technology rather than traditional reporting methods.

## 167 1.2 Problem Statement

168 Dengue remains a critical public health challenge worldwide, with cases increasing  
169 due to the easing of COVID-19 restrictions and heightened global mobility. While  
170 a temporary decline in cases was observed during the pandemic (2020–2022) due  
171 to reduced surveillance efforts, 2023 marked a resurgence, with over five million  
172 cases and more than 5,000 deaths reported across 80 countries. In dengue-endemic  
173 regions like the Philippines, the threat is particularly severe. In Iloilo City and  
174 Province, dengue cases rose by 319% as of August 2023, overwhelming local health-  
175 care systems. This surge strained resources, with over 76% of non-COVID-19 hos-  
176 pital beds occupied by dengue patients, highlighting the urgent need for effective  
177 predictive tools. The lack of a reliable system to monitor and forecast dengue  
178 outbreaks contributes to delayed interventions, exacerbating public health risks  
179 and healthcare burdens in the region.

## 180 1.3 Research Objectives

### 181 1.3.1 General Objective

182 This study aims to develop an AI-based dengue forecasting and monitoring system  
183 for Iloilo City and Province. The researchers will train and compare multiple deep  
184 learning models to predict dengue case trends based on climate data and historical  
185 dengue cases to help public health officials in possible dengue case outbreaks.

### 186 1.3.2 Specific Objectives

187 Specifically, this study aims to:

- 188 1. Gather dengue data from the Iloilo Provincial Health Office and climate data  
189 (including temperature, rainfall, wind, and humidity) from online sources.  
190 Combine and aggregate these data into a unified dataset to facilitate com-  
191 prehensive dengue case forecasting.
- 192 2. Evaluate deep learning models for predicting dengue cases using metrics  
193 such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE),  
194 and Mean Squared Error (MSE). Compare the performance of these models  
195 to determine the most accurate forecasting approach.
- 196 3. Develop a web-based analytics dashboard that integrates a predictive model  
197 and provides data management system for dengue cases in Iloilo City and  
198 the Province.
- 199 4. Assess the usability and effectiveness of the analytics dashboard through  
200 structured feedback and surveys involving health professionals and policy-  
201 makers.

## 202 1.4 Scope and Limitations of the Research

203 This study aims to gather dengue data from the Iloilo Provincial Health Office  
204 and climate data from online sources such as PAGASA or weatherandclimate.com.  
205 These data will be preprocessed, cleaned, and combined into a unified dataset to  
206 facilitate comprehensive dengue case forecasting. However, the study is limited by  
207 the availability and completeness of historical data. Inconsistent or missing data  
208 points may introduce biases and reduce the quality of predictions. Furthermore,  
209 the granularity of the data will be in a weekly format.

210 To evaluate deep learning models for predicting dengue cases, the study will  
211 train and compare the performance of various models, using metrics like Mean  
212 Absolute Error (MAE) and Root Mean Square Error (RMSE). While these models  
213 aim to provide accurate forecasts, their performance is heavily influenced by the  
214 quality and size of the dataset. Limited or low-quality data may lead to suboptimal  
215 predictions. Additionally, the models cannot fully account for external factors  
216 such as public health interventions or socio-economic conditions which may impact  
217 dengue transmission dynamics.

218 The study also involves developing a web-based analytics dashboard that in-  
219 tegrates predictive models and provides a data management system for dengue  
220 cases in Iloilo City and the Province. This dashboard will offer public health  
221 officials an interactive interface to visualize dengue trends, input new data, and

222 identify risk areas. However, its usability depends on feedback from stakeholders,  
223 which may vary based on their familiarity with analytics tools. Moreover, exter-  
224 nal factors such as limited internet connectivity or device availability in remote  
225 areas may affect the system’s adoption and effectiveness. While the dashboard  
226 provides valuable insights, it cannot incorporate all factors influencing dengue  
227 transmission, emphasizing the need for ongoing validation and refinement.

## 228 1.5 Significance of the Research

229 This study’s development of an AI-based dengue forecasting and monitoring sys-  
230 tem has wide-reaching significance for various stakeholders in Iloilo City:

- 231 • **Public Health Agencies:** Organizations like the Department of Health (DOH)  
232 and local health units in Iloilo City and Province stand to benefit greatly  
233 from the system. With dengue predictions, we can help these agencies opti-  
234 mize their response strategies and implement targeted prevention measures  
235 in high-risk areas before cases escalate.
- 236 • **Local Government Units (LGUs):** LGUs can use the system to support  
237 their disaster management and health initiatives by proactively addressing  
238 dengue outbreaks. The predictive insights allow for more efficient planning  
239 and resource deployment in barangays and communities most vulnerable to  
240 outbreaks, improving overall public health outcomes.
- 241 • **Healthcare Facilities:** Hospitals and clinics, which currently face high bed  
242 occupancy rates during dengue season will benefit from early outbreak fore-  
243 casts that can help in managing patient inflow and ensuring adequate hos-  
244 pital capacity.
- 245 • **Researchers and Policymakers:** This AI-driven approach contributes valu-  
246 able insights for researchers studying infectious disease patterns and policy-  
247 makers focused on strengthening the national AI Roadmap. The system’s  
248 data can support broader initiatives for sustainable health infrastructure  
249 and inform policy decisions on resource allocation for dengue control.
- 250 • **Community Members:** By reducing the frequency and severity of outbreaks,  
251 this study ultimately benefits the community at large. This allows for timely

252 awareness campaigns and community engagement initiatives, empowering  
253 residents with knowledge and preventative measures to protect themselves  
254 and reduce the spread of dengue.

## Chapter 2

## Review of Related Literature

### 2.1 Dengue

Dengue disease is a tropical and subtropical mosquito-borne viral illness and is a major health concern in the Philippines (Bravo, Roque, Brett, Dizon, & L’Azou, 2014). The majority of individuals with dengue experience no symptoms. Fever is the most common symptom, typically 4 to 7 days after being bitten by an infected mosquito (Zhou & Malani, 2024). In recent years, the trend of dengue cases in the Philippines has shown notable fluctuations, with periodic outbreaks occurring every 3 to 5 years, often influenced by climatic and environmental changes. According to the Department of Health (DOH), the number of reported cases has steadily increased over the past decades, attributed to urbanization, population growth, and inadequate vector control measures (World Health Organization (WHO), 2018). Moreover, studies suggest that El Niño and La Niña events have significant effects on dengue incidence, with warmer temperatures and increased rainfall providing favorable breeding conditions for mosquitoes (Watts, Burke, Harrison, Whitmire, & Nisalak, 2020). The study of Carvajal et. al. highlights the temporal pattern of dengue cases in Metropolitan Manila and emphasizes the significance of relative humidity as a key meteorological factor, alongside rainfall and temperature, in influencing this pattern (Carvajal et al., 2018).

### 2.2 Outbreak Definition

The definition of an outbreak is a critical factor in disease surveillance, as it determines the threshold at which an unusual increase in cases is considered a

278 public health concern. Studies suggest that outbreak thresholds should be context-  
 279 specific, given the variability in transmission dynamics across different locations  
 280 (Runge-Ranzinger et al., 2016). Outbreak definitions defined using the Endemic  
 281 Channel often base thresholds on 2 standard deviations (SD) above the mean  
 282 number of historic dengue cases. Other studies (Hemisphere, 2015) also used an  
 283 alert threshold of a 5-year mean plus 2 standard deviations. A study by (Brady,  
 284 Smith, Scott, & Hay, 2015) discussed that definitions of dengue outbreaks differ  
 285 significantly across regions and time, making them inconsistent and incomparable.

286 From the studies above, this research implements an outbreak definition basing  
 287 the threshold on 2 standard deviations (SD) above the mean number of historic  
 288 dengue cases. It is important to note that for future outbreak definitions, addi-  
 289 tional local context like available hospital space etc., must be taken into account  
 290 for a more effective outbreak definition.

## 291 **2.3 Existing System: RabDash DC**

292 RabDash, developed by the University of the Philippines Mindanao, is a web-  
 293 based dashboard for rabies data analytics. It combines predictive modeling with  
 294 genomic data, enabling local health authorities to optimize interventions and al-  
 295 locate resources more effectively. RabDash’s modules include trend visualization,  
 296 geographic hotspot mapping, and predictive forecasting, utilizing Long Short-  
 297 Term Memory (LSTM) models for time-series forecasting (RabDashDC, 2024).

298 For DengueWatch, RabDash serves as a strong inspiration, particularly in  
 299 its monitoring, historical trend visualization, and forecasting capabilities. These  
 300 features align well with the needs of dengue control efforts, providing real-time  
 301 insights into outbreak trends and enabling more effective, data-driven decision-  
 302 making. RabDash’s architecture is relevant to the DengueDash, as dengue out-  
 303 breaks similarly require time-series forecasting models. By using LSTM, RabDash  
 304 effectively models trends in outbreak data, which provides a framework for adapt-  
 305 ing LSTM to dengue forecasting. Research indicates that LSTM models outper-  
 306 form traditional methods, such as ARIMA and MLP, in handling the complexities  
 307 of time-dependent epidemiological data (Ligue & Ligue, 2022).

## 308 2.4 Deep Learning

309 The study of (Ligue & Ligue, 2022) highlights how data-driven models can help  
310 predict dengue outbreaks. The authors compared traditional statistical meth-  
311 ods, such as non-seasonal and seasonal autoregressive integrated moving average  
312 (ARIMA), and traditional feed-forward network approach using a multilayer per-  
313 ceptron (MLP) model with a deep learning approach using the long short-term  
314 memory (LSTM) architecture in their prediction model. They found that the  
315 LSTM model performs better in terms of accuracy. The LSTM model achieved a  
316 much lower root mean square error (RMSE) compared to both MLP and ARIMA  
317 models, proving its ability to capture complex patterns in time-series data (Ligue  
318 & Ligue, 2022). This superior performance is attributed to LSTM’s capacity  
319 to capture complex, time-dependent relationships within the data, such as those  
320 between temperature, rainfall, humidity, and mosquito populations, all of which  
321 contribute to dengue incidence (Ligue & Ligue, 2022).

## 322 2.5 Kalman Filter

323 The Kalman Filter is another powerful tool for time-series forecasting that can be  
324 integrated into our analysis. It provides a recursive solution to estimating the state  
325 of a linear dynamic system from a series of noisy measurements. Its application  
326 in epidemiological modeling can enhance prediction accuracy by accounting for  
327 uncertainties in the data (Li et al., 2022). Studies have shown that Kalman filters  
328 are effective in predicting infectious disease outbreaks by refining estimates based  
329 on observed data. A study published in *Frontiers in Physics* utilized the Kalman  
330 filter to predict COVID-19 deaths in Ceará, Brazil. They found that the Kalman  
331 filter effectively tracked the progression of deaths and cases, providing critical in-  
332 sights for public health decision-making (Ahmadini et al., 2021). Another research  
333 article in *PLOS ONE* focused on tracking the effective reproduction number ( $R_t$ )  
334 of COVID-19 using a Kalman filter. This method estimated the growth rate of  
335 new infections from noisy data, demonstrating that the Kalman filter could main-  
336 tain accurate estimates even when case reporting was inconsistent (Arroyo-Marioli,  
337 Bullano, Kucinskas, & Rondón-Moreno, 2021).

338 Our study will compare ARIMA, seasonal ARIMA, Kalman Filter, and LSTM  
339 models using our own collected dengue case data along with weather data to  
340 identify the most effective model for real-time forecasting.



## 341 2.6 Weather Data

342 The relationship between weather patterns and mosquito-borne diseases is inher-  
343 ently nonlinear, meaning that fluctuations in disease cases do not respond propor-  
344 tionally to changes in climate variables(Colón-González, Fezzi, Lake, & Hunter,  
345 2013) Weather data, such as minimum temperature and accumulated rainfall, are  
346 strongly linked to dengue case fluctuations, with effects observed after several  
347 weeks due to mosquito breeding and virus incubation cycles. Integrating these  
348 lagged weather effects into predictive models can improve early warning systems  
349 for dengue control(Cheong, Burkart, Leitão, & Lakes, 2013). A study also suggests  
350 that weather-based forecasting models using variables like mean temperature and  
351 cumulative rainfall can provide early warnings of dengue outbreaks with high sen-  
352 sitivity and specificity, enabling predictions up to 16 weeks in advance(Hii, Zhu,  
353 Ng, Ng, & Rocklöv, 2012).

354 We will utilize weather data, including variables such as temperature, rainfall,  
355 and humidity, as inputs for our dengue forecasting model. Given the strong, non-  
356 linear relationship between climate patterns and dengue incidence, these weather  
357 variables, along with their lagged effects, are essential for enhancing prediction  
358 accuracy and providing timely early warnings for dengue outbreaks.

## 359 2.7 Chapter Summary

360 This chapter reviewed key literature relevant to our study, focusing on existing  
361 systems, predictive modeling techniques and the role of weather data in forecast-  
362 ing dengue outbreaks. We examined systems like RabDash DC, which integrates  
363 predictive modeling with real-time data to inform public health decisions, provid-  
364 ing a foundational structure for our Dengue Watch System. Additionally, deep  
365 learning approaches, particularly Long Short-Term Memory (LSTM) networks,  
366 were highlighted for their effectiveness in time-series forecasting, while alternative  
367 methods such as ARIMA and Kalman Filters were considered for their ability to  
368 model complex temporal patterns and handle noisy data.

369 The literature further underscores the significance of weather variables—such  
370 as temperature and rainfall—in forecasting dengue cases. Studies demonstrate  
371 that these variables contribute to accurate outbreak prediction models. Lever-  
372 aging these insights, our study will incorporate both weather data and historical  
373 dengue case counts to build a reliable forecasting model.

# Chapter 3

## Research Methodology

This chapter lists and discusses the specific steps and activities that will be performed to accomplish the project. The discussion covers the activities from pre-proposal to Final SP Writing.

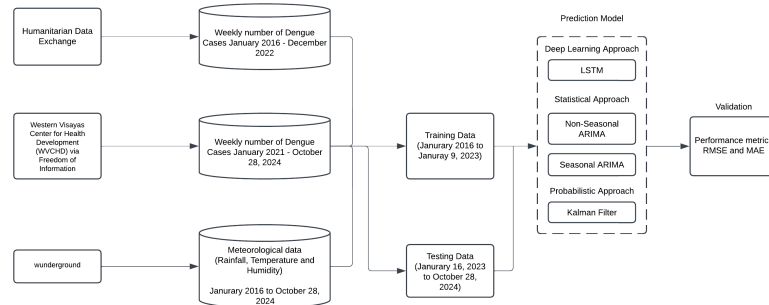


Figure 3.1: Workflow for forecasting the number of weekly dengue cases

This summarizes the workflow for forecasting the number of weekly dengue cases. This workflow focuses on using statistical, deep learning, and probabilistic models to forecast the number of reported dengue cases. The approach involves deploying several models for prediction, including ARIMA and Seasonal ARIMA as statistical approaches, LSTM as a deep learning approach, and the Kalman Filter as a probabilistic approach. These methods are compared with each other to determine the most accurate model.

## 3.1 Research Activities

### 3.1.1 Gather Dengue Data and Climate Data to Create a Complete Dataset for Forecasting

#### Acquisition of Dengue Case Data

The historical dengue case dataset used in this study was obtained from the Humanitarian Data Exchange and the Western Visayas Center for Health Development (WVCHD) via Freedom of Information (FOI) requests. The decision to use weekly intervals was driven by the need for precision and timeliness in capturing fluctuations in dengue cases and weather conditions. Dengue transmission is influenced by short-term changes in weather variables such as rainfall and temperature, which impact mosquito breeding and virus transmission cycles. A weekly granularity allowed the model to better capture these short-term trends, enabling more accurate predictions and responsive public health interventions.

Moreover, using a weekly interval provided more data points for training the models compared to a monthly format. This is particularly critical in time series modeling, where larger datasets help improve the robustness of the model and its ability to generalize to new data. Also, the collection of weather data was done by utilizing web scraping techniques to extract weekly weather data (e.g., rainfall, temperature, and humidity) from Weather Underground (wunderground.com).

#### Data Fields

- **Time.** Represents the specific year and week corresponding to each entry in the dataset.
- **Rainfall.** Denotes the observed average rainfall, measured in millimeters, for a specific week.
- **Humidity.** Refers to the observed average relative humidity, expressed as a percentage, for a specific week.
- **Max Temperature.** Represents the observed maximum temperature, measured in degrees Celsius, for a specific week.
- **Average Temperature.** Represents the observed average temperature, measured in degrees Celsius, for a specific week.

- 417     • **Min Temperature.** Represents the observed minimum temperature, mea-  
418       sured in degrees Celsius, for a specific week.
- 419     • **Wind.** Represents the observed wind speed, measured in miles per hour  
420       (mph), for a specific week.
- 421     • **Cases.** Refers to the number of reported dengue cases during a specific  
422       week.

## 423   **Data Integration and Preprocessing**

424   The dengue case data was integrated with the weather data to create a com-  
425   prehensive dataset, aligning the data based on corresponding timeframes. The  
426   dataset underwent a cleaning process to address any missing values, outliers, and  
427   inconsistencies to ensure its accuracy and reliability. To ensure that all features  
428   and the target variable were on the same scale, a MinMaxScaler was applied to  
429   normalize both the input features (climate data) and the target variable (dengue  
430   cases).

## 431   **Exploratory Data Analysis (EDA)**

- 432     • Analyzed trends, seasonality, and correlations between dengue cases and  
433       weather factors.
- 434     • Created visualizations like time series plots and scatterplots to highlight  
435       relationships and patterns in the data.

## 436   **Outbreak Detection**

437   To detect outbreaks, we computed the outbreak threshold value of dengue cases  
438   using the formula,

$$\text{Outbreak Threshold Value} = \mu + 2\sigma \quad (3.1)$$

439   where  $\mu$  is the historical mean and  $\sigma$  is the standard deviation.

### 440 3.1.2 Develop and Evaluate Deep Learning Models for 441 Dengue Case Forecasting

442 The deep learning models were developed and trained to forecast weekly dengue  
443 cases using historical weather data (rainfall, temperature, wind, and humidity)  
444 and dengue case counts. The dataset was normalized and divided into training and  
445 testing sets, ensuring temporal continuity to avoid data leakage. The methodology  
446 for preparing and training the model are outlined below.

#### 447 Data Preprocessing

448 The raw dataset included weekly aggregated weather variables (rainfall, tempera-  
449 ture, wind, humidity) and dengue case counts. The "Time" column was converted  
450 to a datetime format to ensure proper temporal indexing. To standardize the data  
451 for training, MinMaxScaler was employed, normalizing the feature values and tar-  
452 get variable to a range of 0 to 1. This step ensured that the models could efficiently  
453 process the data without being biased by feature scaling differences.

#### 454 LSTM Model

455 The dataset was split into training and test sets to evaluate the model's perfor-  
456 mance and generalizability:

- 457 • **Training Set:** 80% of the data (572 sequences) was used for model training,  
458 enabling the LSTM to learn underlying patterns in historical dengue case  
459 trends and their relationship with weather variables.
- 460 • **Test Set:** The remaining 20% of the data (148 sequences) was reserved for  
461 testing

462 To prepare the data for LSTM, a sliding window approach was utilized. Se-  
463 quences of weeks of normalized features were constructed as input, while the  
464 dengue case count for the subsequent week was set as the target variable. This  
465 approach ensured that the model leveraged temporal dependencies in the data for  
466 forecasting. To enhance the performance of the LSTM model in predicting dengue  
467 cases, Bayesian Optimization was employed using the Keras Tuner library. The  
468 tuning process aimed to minimize the validation loss (mean squared error) by  
469 adjusting key model hyper-parameters. The search space is summarized below:

470     **LSTM units:**

- 471     • min value: 32
- 472     • max value: 128
- 473     • step: 16
- 474     • sampling: linear

475     **Learning Rate:**

- 476     • min value: 0.0001
- 477     • max value: 0.01
- 478     • step: None
- 479     • sampling: log

480     The tuner was instantiated with:

- 481     • **max trials = 10:** Limiting the search to 10 different configurations
- 482     • **executions per trial = 3:** Running each configuration thrice to reduce  
483         variance
- 484     • **validation split = 0.2:** Reserving 20% of the training data for validation

485     The hyperparameter tuning was conducted for three different window sizes of  
486     data: 5, 10, and 20. This allows the model to have the optimal hyperparameters  
487     used for each window size. Training was conducted over 100 epochs with early  
488     stopping to prevent overfitting while maintaining computational efficiency. A  
489     batch size of 1 was used, enabling the model to process individual sequences,  
490     which is suitable for smaller datasets but results in longer training times. The  
491     Adam optimizer, known for its adaptive learning capabilities and stability was  
492     employed.

493     To validate the effectiveness of the model, cross-validation was implemented.  
494     However, standard k-fold cross-validation randomly shuffles the data, which isn't  
495     suitable for time series since the order of observations is important. To address  
496     this, a time series-specific cross-validation strategy was used with TimeSeriesS-  
497     plit from the scikit-learn library. This method creates multiple train-test splits

498 where each training set expands over time and each test set follows sequentially.  
499 This approach preserves the temporal structure of the data while helping reduce  
500 overfitting by validating the model across different time segments.

501 After training, predictions on both the training and test datasets were rescaled  
502 to their original scale using the inverse transformation of MinMaxScaler. Model  
503 performance was evaluated using the mean squared error (MSE), root mean  
504 squared error (RMSE) and mean absolute error (MAE).

## 505 **ARIMA**

506 The ARIMA model was utilized to forecast weekly dengue cases, leveraging histor-  
507 ical weather data—including rainfall, maximum temperature, and humidity—as  
508 exogenous variables alongside historical dengue case counts as the primary depen-  
509 dent variable. The dataset was partitioned into training (80%) and testing (20%)  
510 sets while maintaining temporal consistency.

511 To identify the optimal ARIMA configuration, a comprehensive grid search  
512 was performed across the following parameter ranges:

- 513 • Autoregressive order (p): 0 to 3
- 514 • Differencing order (d): 0 to 2
- 515 • Moving average order (q): 0 to 3

516 Each combination of (p,d,q) was used to fit an ARIMA model, and perfor-  
517 mance was evaluated based on the mean squared error (MSE) between the pre-  
518 dicted and actual dengue cases on the test set. The parameter set that achieved  
519 the lowest MSE was selected as the final model configuration.

520 Following model selection, the best-fit ARIMA model was retrained on the  
521 training set and subsequently used to forecast dengue cases for the test period.  
522 The predictions were assigned to the **PredictedCases** column in the test dataset.  
523 Model performance was further assessed using key evaluation metrics, including  
524 MSE, root mean squared error (RMSE), and mean absolute error (MAE). Visual  
525 comparisons between actual and predicted dengue cases were produced through  
526 line plots to better illustrate the model’s forecasting accuracy.

## 527 Seasonal ARIMA (SARIMA)

528 The SARIMA modeling process began with data preprocessing, which included  
529 handling missing values through interpolation or imputation, and standardizing  
530 features to ensure stable model training. The dataset was then split into training  
531 and testing sets in an 80:20 ratio, preserving the temporal order of observations.  
532 Seasonality analysis was conducted using time series decomposition and autocor-  
533 relation plots, which revealed a periodicity of 52 weeks—justifying the adoption  
534 of a seasonal model. To fine-tune the model, a grid search was performed over a  
535 range of SARIMA parameters  $(p,d,q)(P,D,Q)[S]$ , while stationarity was validated  
536 using the Augmented Dickey-Fuller (ADF) test. The model was then trained  
537 on the dataset using rainfall, temperature, and humidity as exogenous variables,  
538 with convergence ensured by setting a maximum number of iterations. Residual  
539 diagnostics were used to confirm that residuals were uncorrelated, indicating a  
540 good model fit. For evaluation, forecasts were compared against actual values,  
541 and results were visualized with line plots. Finally, to validate the model’s gener-  
542 alizability across different time periods, Time Series Cross-Validation with three  
543 folds was applied. This allowed assessment of the model’s performance on multi-  
544 ple time segments, providing insights into its robustness in real-world forecasting  
545 scenarios.

## 546 Kalman Filter:

- 547 • Input Variables: The target variable (Cases) was modeled using three re-  
548 gressors: rainfall, max temperature, and humidity.
- 549 • Training and Testing Split: The dataset was split into 80% training and  
550 20% testing to evaluate model performance.
- 551 • Observation Matrix: The Kalman Filter requires an observation matrix,  
552 which was constructed by adding an intercept (column of ones) to the re-  
553 gressors.

554 The Kalman Filter’s Expectation-Maximization (EM) method was employed  
555 for training, iteratively estimating model parameters over 10 iterations. After  
556 training, the smoothing method was used to compute the refined state estimates  
557 across the training data. Observation matrices for the test data were constructed  
558 in the same manner as for the training set, ensuring compatibility with the learned  
559 model parameters. On the test data, the Kalman Filter applied these parameters  
560 to predict and correct the estimated dengue cases, providing more stable and  
561 accurate forecasts compared to direct regression models. Additionally, a hybrid



562 Kalman Filter–LSTM (KF-LSTM) model was developed to combine the strengths  
563 of both approaches. In this setup, the LSTM model was first used to predict  
564 dengue cases based on historical data and weather features. The Kalman Filter  
565 was then applied as a post-processing step to the LSTM predictions, smoothing  
566 out noise and correcting potential errors.

### 567 **3.1.3 Integrate the Predictive Model into a Web-Based** 568 **Data Analytics Dashboard**

#### 569 **Dashboard Design and Development**

- 570 • Design an intuitive, user-friendly web-based dashboard incorporating:
  - 571 – Interactive visualizations of yearly dengue case trends.
  - 572 – Data input and update forms for dengue and weather data.
  - 573 – Map display of dengue cases in each district in Iloilo City

#### 574 **Model Integration and Deployment**

- 575 • Deploy the best-performing model within the dashboard as a backend service  
576 to enable real-time or periodic forecasting.

### 577 **3.1.4 System Development Framework**

578 The Agile Model is the birthchild of both iterative and incremental approaches  
579 in Software Engineering. It aims to be flexible and effective at the same time by  
580 being adaptable to change. It's also important to note that small teams looking  
581 to construct and develop projects quickly can benefit from this kind of method-  
582 ology. As the Agile Method focuses on continuous testing, quality assurance is a  
583 guarantee since bugs and errors are quickly identified and patched.

#### 584 **Design and Development**

585 After brainstorming and researching the most appropriate type of application to  
586 accommodate both the prospected users and the proposed solutions, the team  
587 has decided to proceed with a web application. Given the time constraints and  
588 available resources, it has been decided that the said means is the most pragmatic

589 and practical move. The next step is to select modern and stable frameworks  
 590 that align with the fundamental ideas learned by the researchers in the university.  
 591 The template obtained from WVCHD and Iloilo Provincial Epidemiology and  
 592 Surveillance Unit was meticulously analyzed to create use cases and develop a  
 593 preliminary well-structured database that adheres to the requirements needed  
 594 to produce a quality application. The said use cases serve as the basis of general  
 595 features. Part by part, these are converted into code, and with the help of selected  
 596 libraries and packages, it resulted in the desired outcome that may still modified  
 597 and extended to achieve scalability.

## 598 Testing and Integration

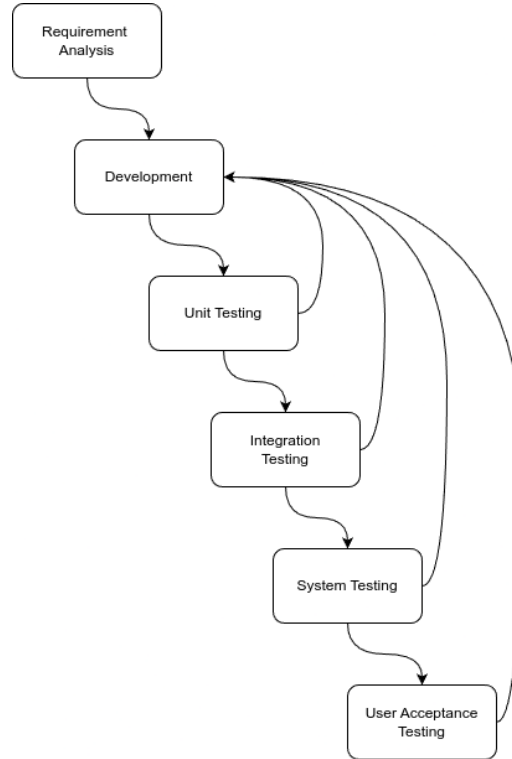


Figure 3.2: Testing Process for DengueWatch

599 Implementing testing is important to validate the system's performance and ef-  
 600 ficacy. Thus a series of tests were conducted to identify and resolve bugs during  
 601 the developmental phase. Each feature was rigorously tested to ensure quality as-  
 602 surance, with particular emphasis on prerequisite features, as development cannot  
 603 progress properly if these fail. Because of this, integration between each feature

604 serves as a pillar for a cohesive user experience. Since dengue reports include  
605 confidential information, anonymized historical dengue reports were used to train  
606 the model and create the foundational architecture of the system. By using func-  
607 tional tests, data validation and visualization can be ensured for further continual  
608 improvements. Security testing is also important as it is needed to safeguard  
609 confidential information when the system is deployed. It includes proper authen-  
610 tication, permission views, and mitigating common injection attacks. Finally, a  
611 user acceptance test from the prospected users, in this case, doctors, nurses, and  
612 other health workers is crucial to assess its performance and user experience. It  
613 enables the developers to confirm if the system meets the needs of the problem.

## 614 **3.2 Development Tools**

### 615 **3.2.1 Software**

#### 616 **Github**

617 GitHub is a cloud-based platform that tracks file changes using Git, an open-  
618 source version control system (*About GitHub and Git - GitHub Docs*, n.d.). It is  
619 used in the project to store the application’s source code, manage the system’s  
620 source version control, and serve as a repository for the Latex files used in the  
621 actual research.

#### 622 **Visual Studio Code**

623 Visual Studio Code is a free, lightweight, and cross-platform source code editor  
624 developed by Microsoft (*Why Visual Studio Code?*, 2021). As VS Code supports  
625 this project’s programming and scripting languages, it was chosen as the primary  
626 source code editor.

#### 627 **Django**

628 Django is a free and open-sourced Python-based web framework that offers an  
629 abstraction to develop and maintain a secure web application. As this research  
630 aims to create a well-developed and maintainable application, it is in the best  
631 interest to follow an architectural pattern that developers and contributors in the

632 future can understand. Since Django adheres to Model-View-Template (MVT)  
633 that promotes a clean codebase by separating data models, business logic, and  
634 presentation layers, it became the primary candidate for the application’s back-  
635 bone.

## 636 **Next.js**

637 A report by Statista (2024) claims that React is the most popular front-end frame-  
638 work among web developers. However, React has limitations that can be a nui-  
639 sance in rapid software development, which includes routing and performance op-  
640 timizations. This is where Next.js comes in—a framework built on top of React.  
641 It offers solutions for React’s deficiency, making it a rising star in the framework  
642 race.

## 643 **Postman**

644 As the application heavily relies on the Application Programming Interface (API)  
645 being thrown by the backend, it is a must to use a development tool that facilitates  
646 the development and testing of the API. Postman is a freemium API platform  
647 that offers a user-friendly interface to create and manage API requests (*What is*  
648 *Postman? Postman API Platform*, n.d.).

## 649 **3.2.2 Hardware**

650 The web application was developed on laptop computers with minimum specifica-  
651 tions of an 11th-generation Intel i5 CPU and 16 gigabytes of RAM. Furthermore,  
652 an RTX 3060 mobile GPU was also utilized to retrain the integrated model locally.

## 653 **3.2.3 Packages**

### 654 **Django REST Framework**

655 Django Rest Framework (DRF) is a third-party package for Django that provides a  
656 comprehensive suite of features to simplify the development of robust and scalable  
657 Web APIs (Christie, n.d.). These services include Serialization, Authentication  
658 and Permissions, Viewsets and Routers, and a browsable API .

## 659 Leaflet

660 One of the features of the web application is the ability to map the number  
661 of cases using a Choropleth Map. Leaflet is the only free, open-sourced, and  
662 most importantly, stable JavaScript package that can do the job. With its ultra-  
663 lightweight size, it offers a comprehensive set of features that does not trade  
664 off performance and usability (*Leaflet — an open-source JavaScript library for*  
665 *interactive maps*, n.d.).

## 666 Chart.js

667 Another feature of the application is to provide users with informative, approach-  
668 able data storytelling that is easy for everyone to understand. The transformation  
669 of pure data points and statistics into figures such as charts is a big factor. Thus,  
670 there is a need for a package that can handle this feature without compromising  
671 the performance of the application. Chart.js is a free and open-source JavaScript  
672 package that is made to meet this criteria as it supports various types of charts  
673 (*Chart.js*, n.d.).

## 674 Tailwind CSS

675 Using plain CSS in production-quality applications can be counterproductive.  
676 Therefore, CSS frameworks were developed to promote consistency and accelerate  
677 the rapid development of web applications (Joel, 2021). One of these is Tailwind,  
678 which offers low-level utility classes that can be applied directly to each HTML  
679 element to create a custom design (*Tailwind CSS - Rapidly build modern web-*  
680 *sites without ever leaving your HTML.*, n.d.). Given the limited timeline for this  
681 project, using this framework is a wise choice due to its stability and popularity  
682 among developers.

## 683 Shadcn

684 Shadcn offers a collection of open-source UI boilerplate components that can be  
685 directly copied and pasted into one's project. With the flexibility of the provided  
686 components, Shadcn allows developers to have full control over customization and  
687 styling. Since this is built on top of Tailwind CSS and Radix UI, it is supported  
688 by most modern frontend frameworks, including Next.js (Shadcn, n.d.).

## 689 **Zod**

690 Data validation is integral in this web application since it will handle crucial data  
691 that will be used for analytical inferences and observations. Since Zod is primarily  
692 used for validating and parsing data, it ensures proper communication between  
693 the client and the server (Zod, n.d.).

## 694 3.3 Application Requirements

### 695 3.3.1 Backend Requirements

#### 696 Database Structure Design

697 Determining how data flows and how it would be structured is crucial in creating  
 698 the system as it defines how extendible and flexible it would be for future features  
 699 and updates. Thus, creating a comprehensive map of data ensures proper nor-  
 700 malization that eliminates data redundancy and improves data integrity. Figure  
 701 3.3 depicts the designed database schema that showcases the relationship between  
 702 the application's entities.

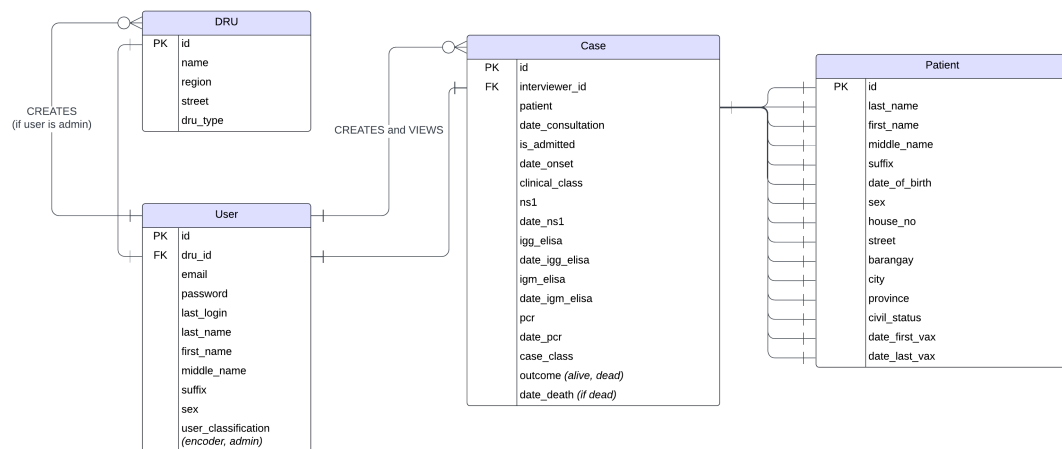


Figure 3.3: Entity-Relationship Database Schema Hybrid Diagram for DengueDash Database Structure

### 3.3.2 User Interface Requirements

#### Disease Reporting Unit Admin Interface

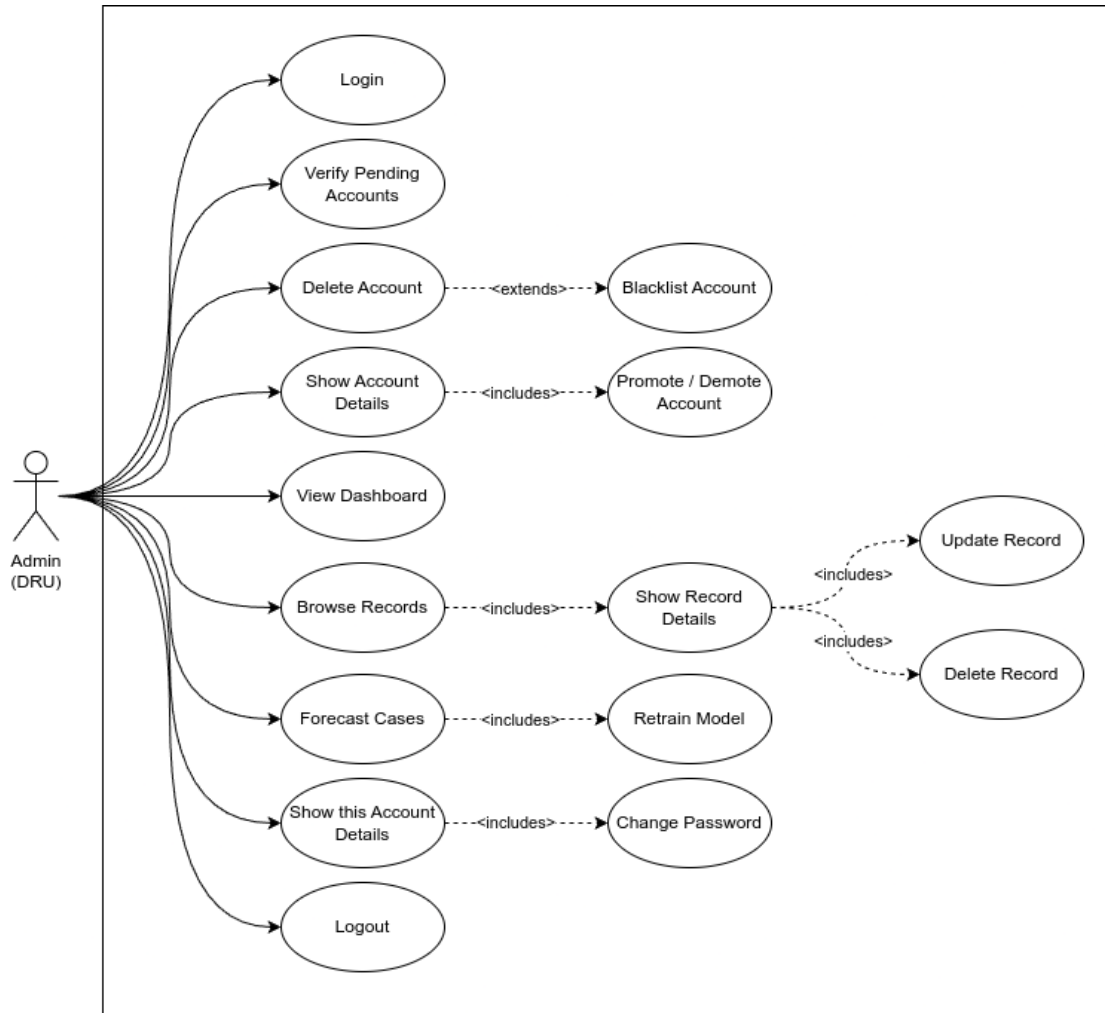


Figure 3.4: Use Case Diagram for DRU Admin

#### Surveillance Unit Admin Interface

Figure 3.4 shows the actions an admin for a specific Disease Reporting Unit (DRU) can take in the application. These include managing accounts, browsing records, and forecasting and retraining all the consolidated data under the unit. To protect the integrity of data, encoders that register to a DRU must first be verified by these users, and then the encoder's account can only be authorized to use the



711 application.



Figure 3.5: Use Case Diagram for Surveillance Unit Admin

712 While the previous use case focuses on hospitals, clinics, and other reporting  
 713 units, the use case presented in Figure 3.5 has a one-step higher authorization as  
 714 it manages these DRUs. It has the same features as the DRU admin but with  
 715 extra management of the DRUs under a specific surveillance unit. At this point,  
 716 only the authorized surveillance unit administrator can register and create a DRU  
 717 to uphold transparency and accountability.

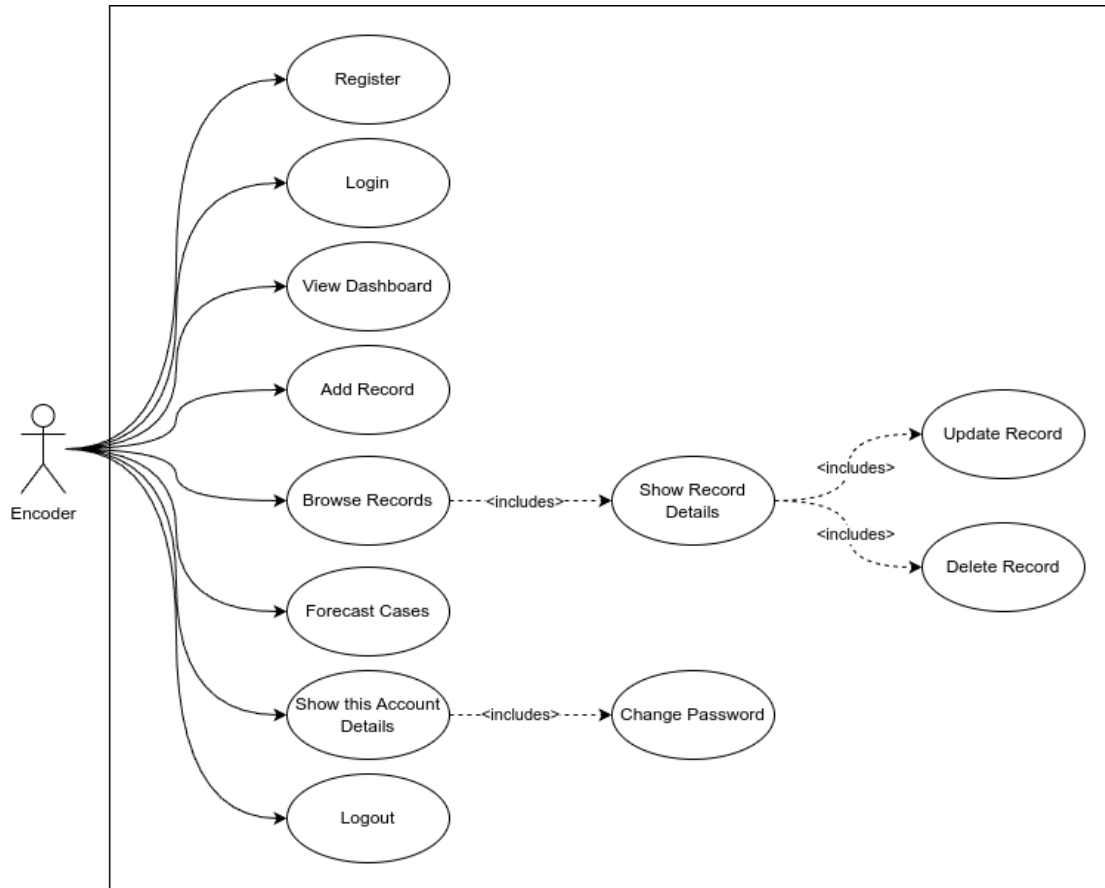


Figure 3.6: Use Case Diagram for Encoder

Figure 3.6, on the other hand, illustrates the use cases for the system's primary users. These users can register but must wait for further verification to access the application. Similar to the previous interfaces, encoders can browse and manage records, as well as forecast the consolidated cases under a specific surveillance or disease reporting unit, but they are not allowed to retrain the model. Lastly, they are the only type of user that can file and create dengue cases by filling out a form with the required details.

### 726 3.3.3 Security and Validation Requirements

#### 727 Password Encryption

728 Storing passwords as plain text in the database is a disgrace and a mortal sin in  
729 production. It is important to implement precautionary methods such as hashing  
730 and salting, followed by encryption with a strong algorithm, to prevent bad actors  
731 from using the accounts for malicious transactions. By default, Django generates  
732 a unique random salt for each password and encrypts it with Password-Based Key  
733 Derivation Function 2 (PBKDF2) with a SHA256 hash function. Utilizing these  
734 techniques ensures that in the event of a data breach, cracking these passwords  
735 would be time-consuming and useless for the attackers.

#### 736 Authentication

737 DengueWatch utilizes JSON Web Tokens (JWT) to authenticate the user. Since  
738 the mechanism operates in a stateless manner, tokens are served only after a  
739 successful login, eliminating the need for the server to keep a record of the token,  
740 which is vulnerable to session hijacking. In addition, these tokens are signed with  
741 a secret key, ensuring they have not been tampered with.

#### 742 Data Validation

743 Both the backend and frontend should validate the input from the user to preserve  
744 data integrity. Thus, Zod is implemented in the latter to help catch invalid inputs  
745 from the user. By doing this, the user can only send proper requests to the server  
746 which streamlines the total workflow. On the other hand, Django has also a built-  
747 in validator that checks the data type and ensures that the input matches the  
748 expected format on the server side. These validation processes ensure that only  
749 valid and properly formatted data is accepted, which reduces the risk of errors  
750 and ensures consistency across the web application.

## 751 3.4 Calendar of Activities

752 A Gantt chart showing the schedule of the activities is included below. Each  
753 bullet represents approximately one week of activity.

Table 3.1: Timetable of Activities for 2024

Activities	Aug	Sept	Oct	Nov	Dec
Project Initiation and Team Formation	••				
Literature Review and Data Gathering	••	••••			
Data Cleaning and Feature Selection		••		•	•
Creating System Dashboard		••	••••	•	
Analysis and Interpretation of Results			•		•
Documentation	••	••••	••••	••••	••••

Table 3.2: Timetable of Activities for 2025

Activities	Jan	Feb	Mar	Apr	May
Create Admin Dashboard	•	•••			
Integrate the Best Model to the System	•	••••			
Extend Features to Accommodate a National Setting		•	••		
User Testing			••	•	
System Deployment				•••	
Documentation	••	••••	••••	••••	••••

## Chapter 4

# Results and Discussion/System Prototype

### 4.1 Data Gathering

The data for dengue case prediction was gathered from a variety of reliable sources, enabling a comprehensive dataset spanning from January 2011 to October 2024. This dataset includes 720 rows of data, each containing weekly records of dengue cases along with corresponding meteorological variables, such as rainfall, temperature, and humidity.

1. Dengue Case Data: The primary source of historical dengue cases came from the Humanitarian Data Exchange and the Western Visayas Center for Health Development (WVCHD). The dataset, accessed through Freedom of Information (FOI) requests, provided robust case numbers for the Western Visayas region. The systematic collection of these data points was essential for establishing a reliable baseline for model training and evaluation.
2. Weather Data: Weekly weather data was obtained by web scraping from Weather Underground, allowing access to rainfall, temperature, wind, and humidity levels that correlate with dengue prevalence.

```
data.head()
```

	Time	Rainfall	MaxTemperature	AverageTemperature	MinTemperature	Wind	Humidity	Cases
0	2011-01-03	9.938571	29.444400	25.888890	23.888900	11.39	86.242857	5
1	2011-01-10	8.587143	30.000000	26.705556	24.444444	7.32	88.028571	4
2	2011-01-17	5.338571	30.000000	26.616667	25.000000	7.55	84.028571	2
3	2011-01-24	5.410000	30.555556	26.483333	20.555556	10.67	80.971429	7
4	2011-01-31	2.914286	28.333333	25.283333	18.650000	11.01	74.885714	2

Figure 4.1: Snippet of the Combined Dataset

#	Column	Non-Null	Count	Dtype
0	Time	720 non-null		datetime64[ns]
1	Rainfall	720 non-null		float64
2	MaxTemperature	720 non-null		float64
3	AverageTemperature	720 non-null		float64
4	MinTemperature	720 non-null		float64
5	Wind	720 non-null		float64
6	Humidity	720 non-null		float64
7	Cases	720 non-null		int64

dtypes: datetime64[ns](1), float64(6), int64(1)  
memory usage: 45.1 KB

Figure 4.2: Data Contents

## 772 4.2 Exploratory Data Analysis

773 From the summary above, the dataset consists of 720 weekly records with 8  
774 columns:

- 775 • **Time.** Weekly timestamps (e.g. "2011-w1")
- 776 • **Rainfall.** Weekly average rainfall (mm)
- 777 • **MaxTemperature, AverageTemperature, MinTemperature.** Weekly  
778 temperature data (C)
- 779 • **Wind.** Wind speed (m/s)
- 780 • **Humidity.** Weekly average humidity (%)
- 781 • **Cases.** Reported dengue cases

	Time	Rainfall	MaxTemperature	AverageTemperature	MinTemperature	Wind	Humidity	Cases
count	720	720.000000	720.000000	720.000000	720.000000	720.000000	720.000000	720.000000
mean	2017-12-02 11:22:00	13.957499	32.191142	28.110319	25.038472	6.172417	81.609442	23.744444
min	2011-01-03 00:00:00	0.000000	-14.600000	24.494444	12.222222	1.910000	73.185714	0.000000
25%	2014-06-21 06:00:00	1.270000	31.666667	27.504167	25.000000	4.117500	79.885713	5.000000
50%	2017-12-07 12:00:00	4.318000	32.222222	28.161111	25.000000	5.725000	81.771429	12.000000
75%	2021-05-11 18:00:00	10.414000	32.777778	28.751389	25.555556	7.860000	83.503571	26.000000
max	2024-10-28 00:00:00	445.008000	58.333333	30.916667	32.222222	19.200000	89.571429	319.000000
std	NaN	35.448846	2.616379	0.999800	1.291659	2.446703	2.831674	37.144813

Figure 4.3: Dataset Statistics

From the statistics in figure 4.3, the number of cases ranges from 0 to 319. The average number of dengue cases per week is 23.74, with a median of 12 cases and a standard deviation of 37.14. The distribution is highly skewed, with some weeks experiencing significant number of cases (up to 319 cases). Rainfall shows a wide variation (0 to 445mm), while temperature remains relatively stable, with an average of 28.1 degree celsius. Humidity levels ranges from 73% to 89% with a mean of 81.6%.

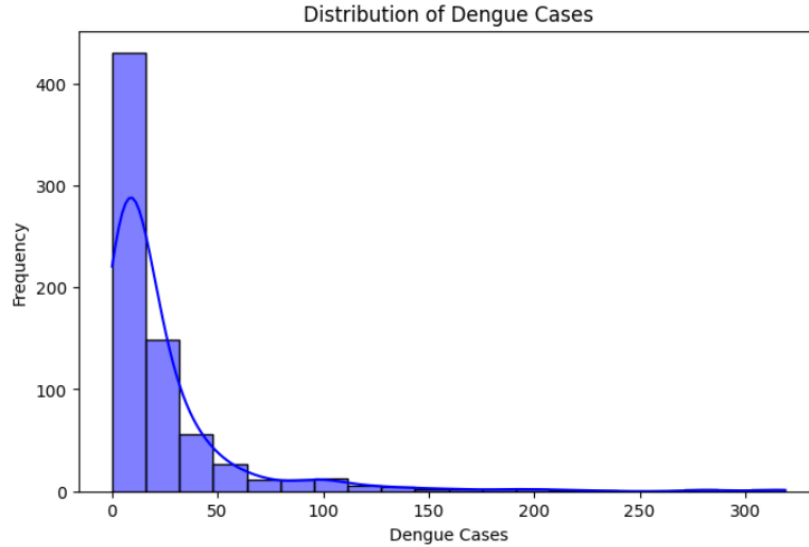


Figure 4.4: Distribution of Dengue Cases

In figure 4.4, a histogram of dengue cases shows a right-skewed distribution, indicating that most weeks experience low case counts, while a few weeks record outbreaks. To further analyze the distribution, dengue cases were categorized into different intervals (Figure 4.5): 0-5 cases, 6-15 cases, 16-30 cases, 31-100 cases and 101+

794 cases. The majority of weeks falls within the 0-5 cases and 6-15 cases categories,  
 795 indicating that most weeks have low dengue cases. Meanwhile, weeks with 101+  
 cases are rare, suggesting that extreme outbreaks are not frequent.

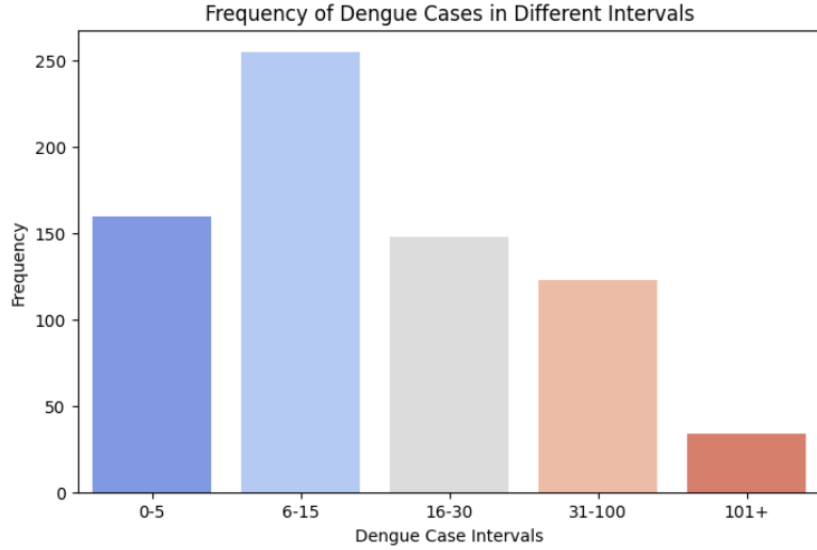


Figure 4.5: Frequency of Dengue Cases in Different Intervals

796

797 Figure 4.6 illustrates the trend of weekly dengue cases over time. The data  
 798 reveals periodic spikes in the number of cases, suggesting a seasonal pattern in  
 799 dengue cases. Notably, peak cases are observed during certain periods approx-  
 800 imately 3 years, potentially aligning with specific climatic conditions such as  
 801 increased rainfall or temperature changes. This underscores the importance of  
 802 incorporating climate variables into the forecasting model.

803 Figure 4.7 shows the ranking of correlation coefficients between dengue cases  
 804 and selected features, including rainfall, humidity, maximum temperature, aver-  
 805 age temperature, minimum temperature, and wind speed. Among these, rainfall  
 806 exhibits the highest positive correlation with dengue cases (correlation coefficient  
 807 0.13), indicating that increased rainfall may contribute to higher cases counts.  
 808 This aligns with existing studies suggesting that stagnant water from heavy rain-  
 809 fall creates breeding grounds for mosquitos. It is followed by humidity ( 0.10),  
 810 suggesting that higher humidity levels may enhance mosquito reproduction, lead-  
 811 ing to more dengue cases. Temperature has a weak to moderate positive corre-  
 812 lation with dengue cases, with maximum temperature (0.09) showing a stronger  
 813 relationship than average and minimum temperature.

814 Figure 4.8 shows the ranking of correlation coefficients between dengue cases  
 815 and selected features, with the addition of lagged effects. The analysis reveals no



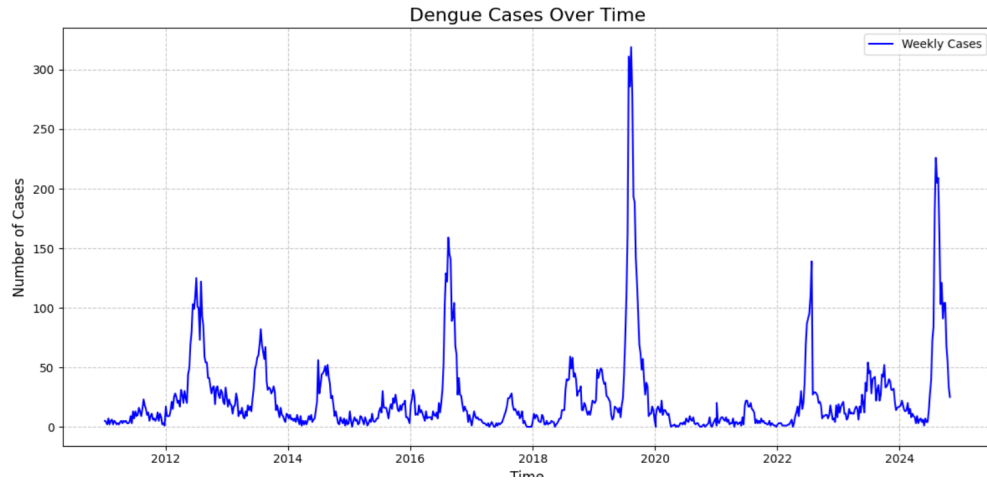


Figure 4.6: Trend of Dengue Cases

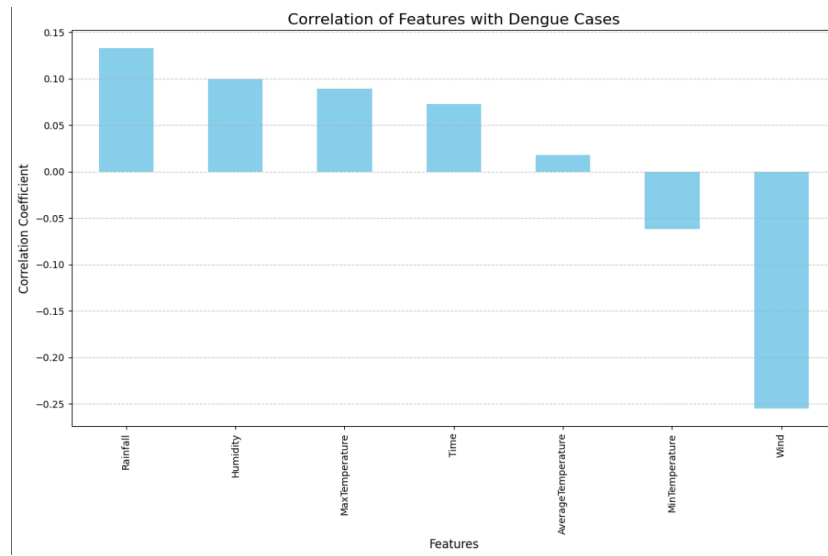


Figure 4.7: Ranking of Correlations

improvement in correlation when lagged variables are compared to direct observations. This suggests that the observed values of rainfall, humidity, and maximum temperature remain the most significant predictors for dengue case forecasting. Overall, the exploratory data analysis highlights the significance of rainfall, humidity, and max temperature variables in dengue case forecasting.

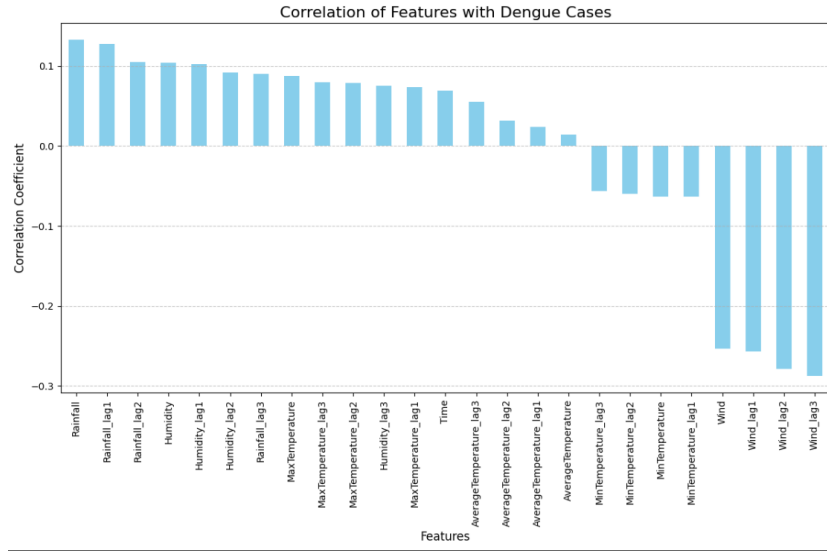


Figure 4.8: Ranking of Correlations (with lagged effects)

## 4.3 Outbreak Detection

To identify outbreaks, we calculated the outbreak threshold value using the historical mean as the endemic channel. The threshold is determined using the formula:

$$\text{Outbreak Threshold Value} = \mu + 2\sigma \quad (4.1)$$

$$= 23.744444 + 2(37.144813) \quad (4.2)$$

$$= 23.744444 + 74.289626 \quad (4.3)$$

$$= 98.03407 \quad (4.4)$$

where  $\mu$  is the historical mean and  $\sigma$  is the standard deviation.

This result indicates that dengue cases exceeding 98 in Iloilo City can be considered an outbreak. However, it is important to note that this threshold serves only as a baseline. Additional parameters, such as the number of hospital beds available in the city, must be considered to compute a more effective threshold and develop an appropriate response strategy.

## 830 4.4 Model Training Results

831 The models were evaluated using three metrics: MSE, RMSE, and MAE. The  
 832 table below provides a summary and comparative analysis of each model's results  
 833 across these metrics, offering insights into the strengths and limitations of each  
 834 forecasting technique for dengue case prediction in Iloilo City. The lower values  
 835 of the three metrics indicate better forecasting performance. Table 4.1 shows that  
 836 the models performed differently on testing data. LSTM outperformed the other  
 837 models with the lowest RMSE, MSE, and MAE while the other three models had  
 838 relatively higher values for the three metrics.

Method	LSTM	Seasonal ARIMA	ARIMA	Kalman Filter	KF-LSTM
Testing MSE	285.54	1261.20	1521.48	1474.82	785.35
Testing RMSE	16.90	34.45	39.00	38.40	25.56
Testing MAE	9.45	18.73	25.80	22.33	14.55
Best Parameters	Window Size: 5 Learning Rate: 0.01 Units: 64	(2,0,2)(0,1,1)	(1,2,2)	Observation Covariance: 10.0 Transition Covariance: $0.1 \times \text{Identity}$	Same as LSTM

Table 4.1: Comparison of different models for dengue prediction

### 839 4.4.1 LSTM Model

840 The LSTM model was tuned for the following parameters: learning rate and units.  
 841 The hyperparameter tuning was conducted for each window size, finding the best  
 842 parameters for each window size. Further evaluating which window size is most  
 843 suitable for the prediction model, Table 4.2 shows the evaluation metrics for each  
 window size used in the LSTM model training.

Window Size	MSE	RMSE	MAE	R <sup>2</sup>
5	285.54	16.90	9.45	0.83
10	334.63	18.29	9.85	0.80
20	294.85	17.17	9.35	0.83

Table 4.2: Comparison of Window Sizes

844

845 The results indicate that a window size of 5 weeks provides the most accurate  
 846 predictions, as evidenced by the lowest MSE and RMSE values. Furthermore, the  
 847 R<sup>2</sup> score of 0.83 indicates that 83% of the variability in the target variable (cases)  
 848 is explained by the independent variables (the inputs) in the model, making it a  
 849 reliable configuration overall.

850 Figure 4.9 illustrates the model's performance in predicting dengue cases for  
 851 each fold using a window size of 5. As shown in the plot, the training set progres-

852 sively increases with each fold, mimicking a real-world scenario where more data  
853 becomes available over time for dengue prediction. Figure 4.10 demonstrates that  
854 the predicted cases closely follow the trend of the actual cases, indicating that the  
855 LSTM model successfully captures the underlying patterns in the data. It is also  
856 evident that as the fold number increases and the training set grows, the accuracy  
857 of the predictions on the test set improves. Despite the test data being unseen,  
858 the model exhibits a strong ability to generalize, suggesting it effectively leverages  
859 past observations to predict future trends.

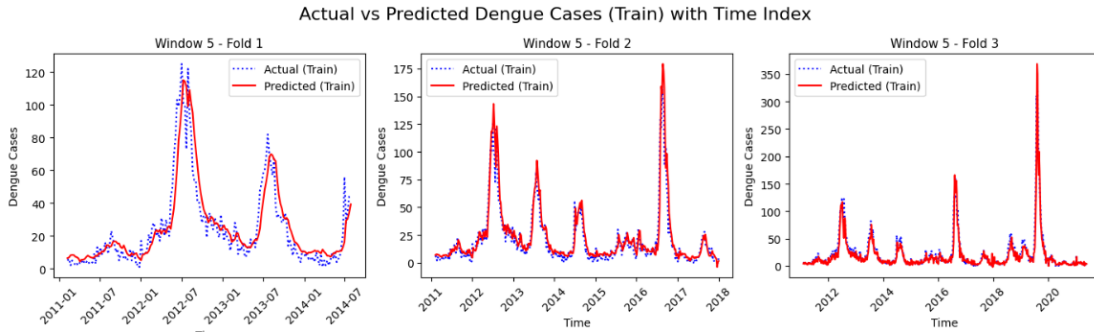


Figure 4.9: Training Folds - Window Size 5

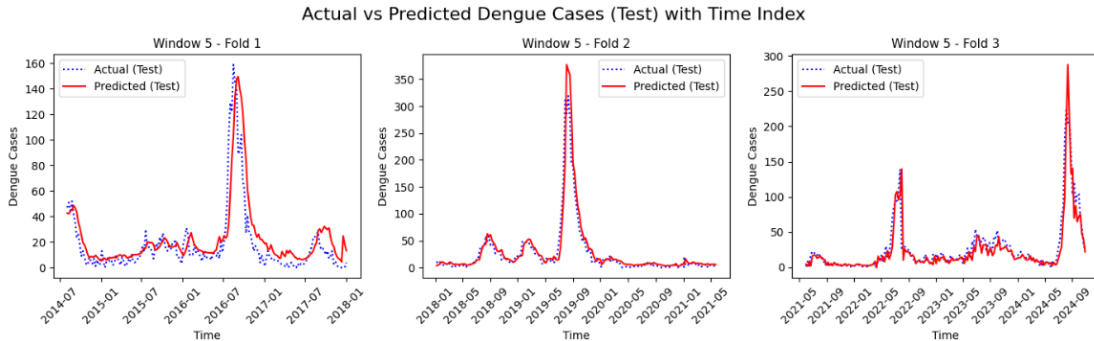


Figure 4.10: Testing Folds - Window Size 5

## 860 4.4.2 ARIMA Model

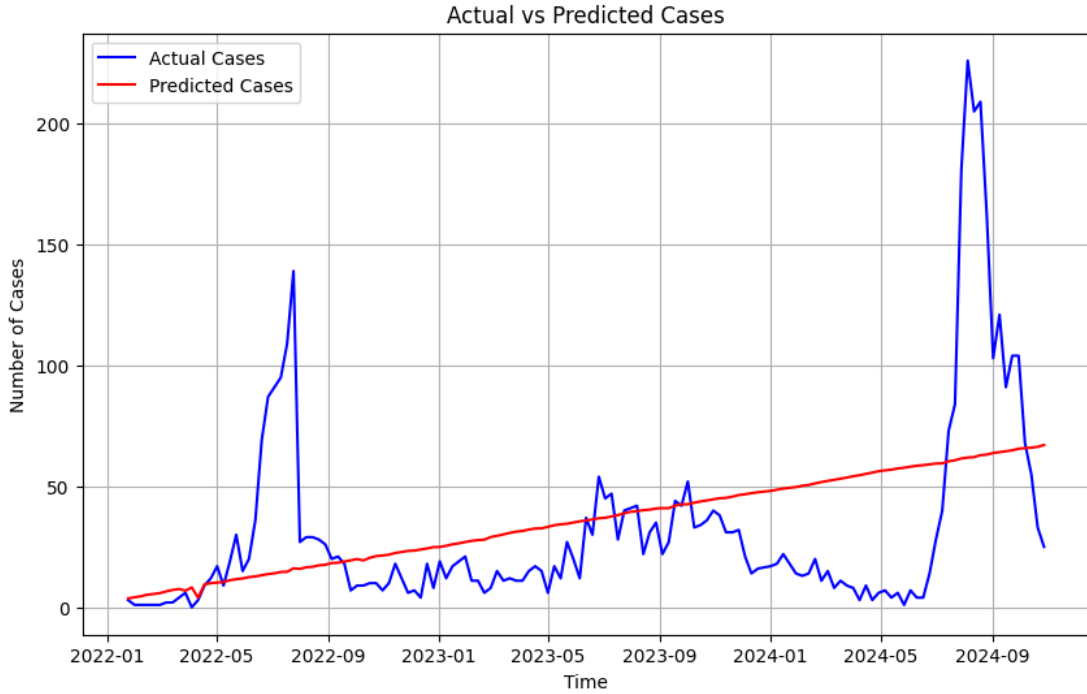


Figure 4.11: ARIMA Prediction Results for Test Set

861 The ARIMA model was developed to capture non-seasonal trends in the data.  
 862 To determine the best model configuration, grid search was used to explore vari-  
 863 ous combinations of ARIMA parameters, ultimately selecting **ARIMA(1, 2, 2)**.  
 864 The model was iteratively refined over **400 iterations** to ensure convergence to  
 865 an optimal solution. Figure 4.11 illustrates the comparison between actual and  
 866 predicted dengue cases in the test set. As shown in the plot, the ARIMA model  
 867 struggled to capture the non-linear characteristics and abrupt spikes in the data.  
 868 Consequently, it failed to accurately reflect the fluctuations and outbreak patterns  
 869 seen in the actual case counts.

870 The model's performance was assessed using regression metrics to evaluate its  
 871 forecasting capability. The ARIMA model yielded the following error metrics:

- 872 • **MSE (Mean Squared Error):** 1521.48
- 873 • **RMSE (Root Mean Squared Error):** 39.01
- 874 • **MAE (Mean Absolute Error):** 25.80

### 4.4.3 Seasonal ARIMA (SARIMA) Model

To address the limitations of the ARIMA model, a Seasonal ARIMA (SARIMA) model was developed to capture both non-seasonal and seasonal variations in the data.

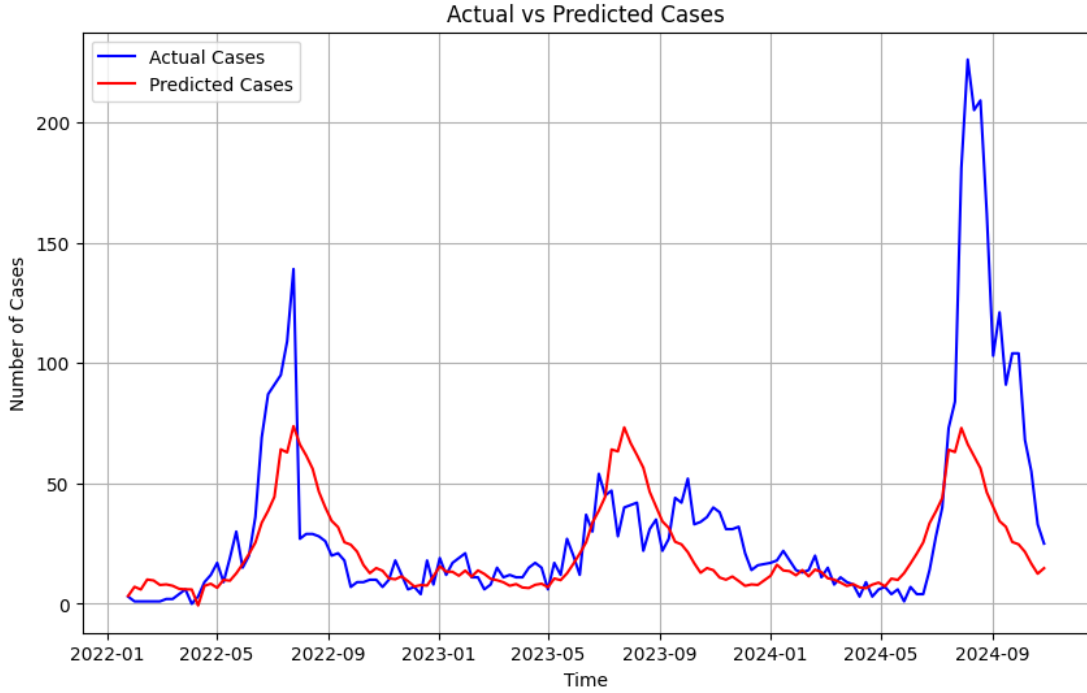


Figure 4.12: Seasonal ARIMA Prediction Results for Test Set

This model incorporates seasonal parameters, which were tuned using grid search to find the best configuration: **SARIMA(2, 0, 2)(0, 1, 1)[52]**. As with ARIMA, **400 iterations** were applied to ensure a robust fit. As shown in Figure 4.12, the SARIMA model demonstrates a notable improvement in performance. Unlike its non-seasonal counterpart, it effectively captures the general trend and aligns more closely with the peaks observed in the actual dengue cases, indicating its ability to model seasonal dynamics.

The model's performance was assessed using regression metrics to evaluate its forecasting capability. The SARIMA model yielded the following error metrics:

- **MSE:** 1109.69
- **RMSE:** 33.31

890

- **MAE: 18.09**

891  
892  
893  
894

The lower error values, when compared to the ARIMA model, highlight the SARIMA model's superior capability in forecasting dengue cases. Its effectiveness in capturing seasonal patterns contributed to a more accurate representation of the actual cases.

895  
896  
897  
898  
899  
900

After training the model, the SARIMA model was validated using the same Time Series Cross-Validation strategy employed in the LSTM model. Table 4.3 presents the performance metrics for each fold, as well as the average metrics across all folds. The average RMSE and MAE values were close to those obtained during the initial training phase, indicating that the SARIMA model performed consistently across different time segments.

Fold	MSE	RMSE	MAE
1	659.68	25.68	16.00
2	2127.49	46.12	21.30
3	996.43	31.56	18.89
<b>Average</b>	<b>1261.20</b>	<b>34.45</b>	<b>18.73</b>

Table 4.3: Comparison of SARIMA performance for each fold

901

#### 4.4.4 Kalman Filter Model

902  
903  
904  
905  
906  
907  
908  
909

Figure 4.13 shows the comparison between the actual dengue cases and the predicted values on the test set. As illustrated in the plot, the Kalman Filter model demonstrates a moderate ability to follow the general trend of the actual data. While it effectively captures some rising and falling patterns, it still struggles to accurately replicate the sharp peaks and extreme values found in the real case counts. This limitation is particularly noticeable during the large spikes in 2022 and 2024. The model's performance was evaluated using standard regression metrics. The results are as follows:

$$\text{MSE} = 1474.82, \quad \text{RMSE} = 38.40, \quad \text{MAE} = 22.34$$

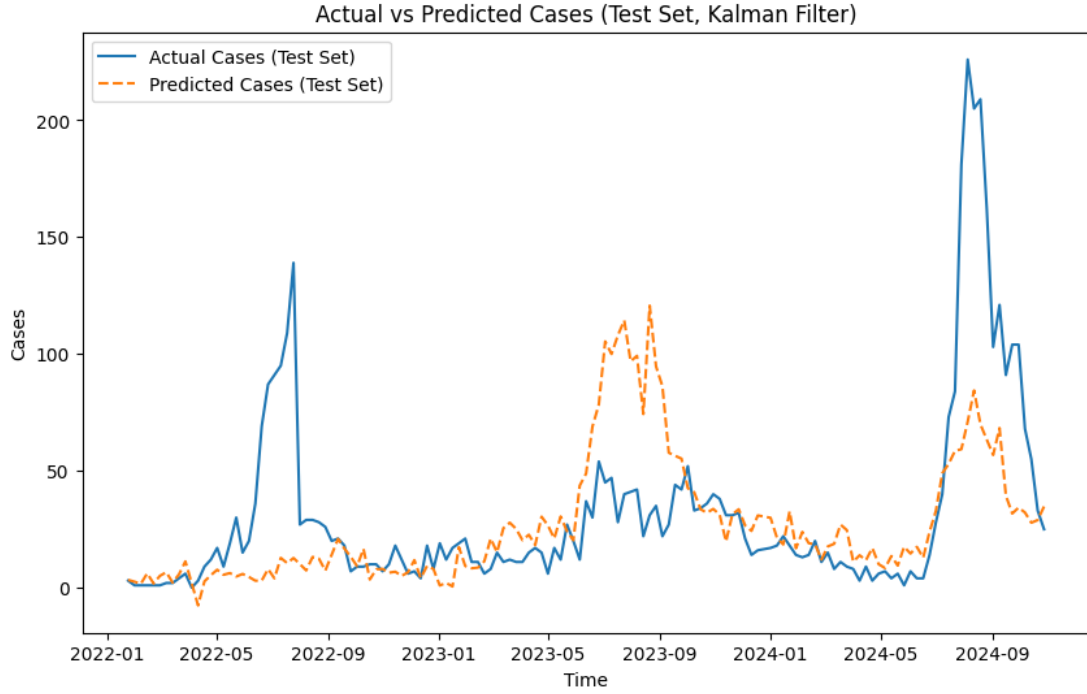


Figure 4.13: Kalman Filter Prediction Results for Test Set

910 The Kalman Filter was then combined with the LSTM model in order to see  
 911 improvements in its predictions. Table 4.4 shows the metrics across three folds  
 912 using the same Time Series Cross Validation Strategy employed in the previous  
 913 models to see how it performed on different time segments.

Fold	MSE	RMSE	MAE
1	113.59	10.66	6.42
2	752.51	27.43	12.11
3	1489.95	38.60	25.13
<b>Average</b>	<b>785.35</b>	<b>25.56</b>	<b>14.55</b>

Table 4.4: Comparison of KF-LSTM performance for each fold

914 2

915 As can be seen in the table above, the performance of the hybrid model demon-  
 916 strated improvements in all metrics as compared to just using the Kalman Filter  
 917 alone.



## 4.5 System Prototype

### 4.5.1 Home Page

The Home Page is intended for all visitors of the web application. The Analytics Dashboard, which displays relevant statistics for dengue cases at a certain year and location, is the primary component highlighted, as seen in Figure 4.14. This component includes a combo chart that graphs the number of dengue cases and deaths per week in a specific year, a choropleth map that tracks the number of dengue cases per location, and various bar charts that indicate the top locations affected by dengue.

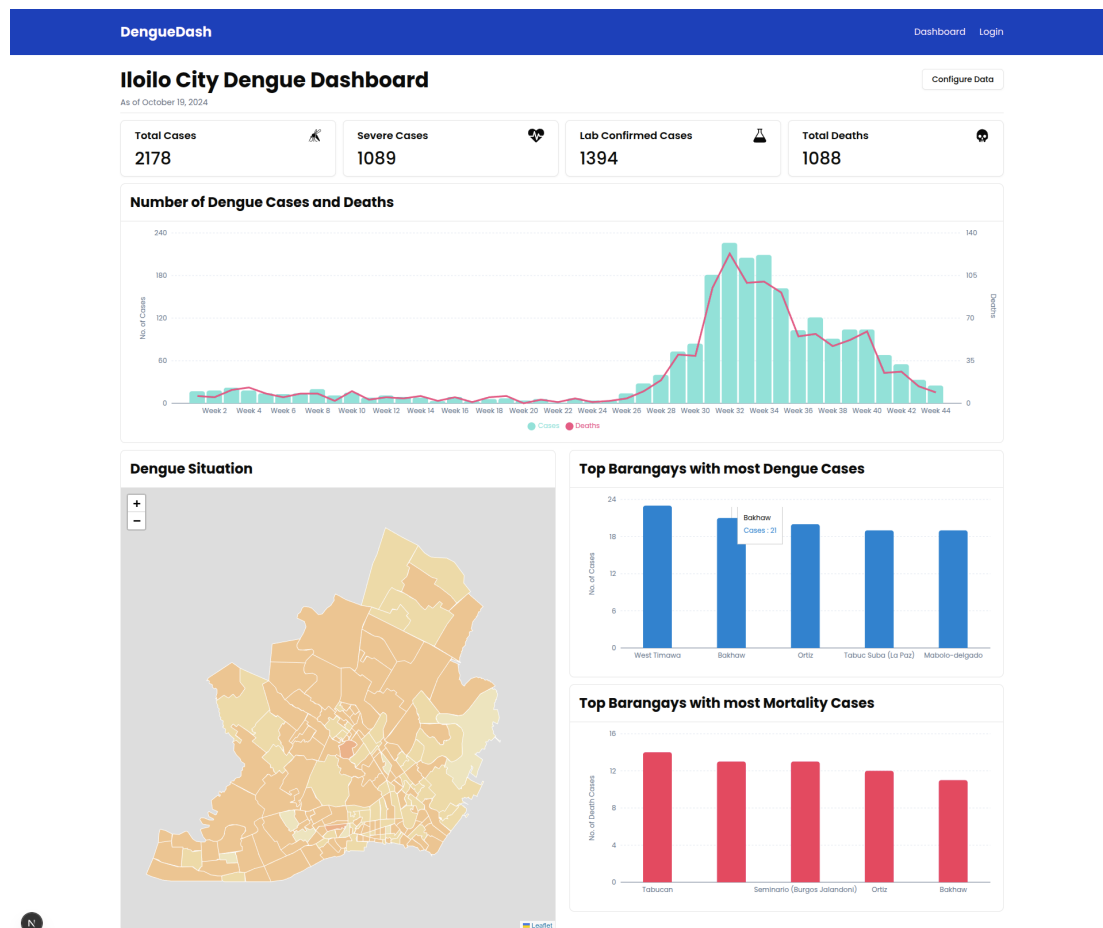


Figure 4.14: Home Page

## 4.5.2 User Registration, Login, and Authentication

The registration page, as shown in Figure 4.15, serves as a gateway to access the authenticated pages of the web application. Only prospected encoders can create an account since administrator accounts are only made by existing administrator accounts to protect the data's integrity in production. After registering, the "encoder account" cannot access the authorized pages yet as it needs to be verified first by an administrator managing the unit the user entered. Once verified, the user can log in to the system through the page shown in Figure 4.16. After entering the correct credentials, which consist of an email and password, the system uses HTTP-only cookies containing JSON Web Tokens (JWT) to prevent vulnerability to Cross-site Scripting (XSS) attacks. It will then proceed to the appropriate page the type of user belongs to.

**DengueDash** [Dashboard](#) [Login](#)

### sign Up

Create your account to get started

<b>First Name</b> <input type="text" value="John"/>	<b>Middle Name (optional)</b> <input type="text" value="David"/>
<b>Last Name</b> <input type="text" value="Doe"/>	<b>Sex</b> <input type="text" value="Select gender"/>
<b>Email</b> <input type="text" value="john@example.com"/>	<b>Region</b> <input type="text" value="Select region"/>
<b>Surveillance Unit</b> <input type="text" value="Select surveillance unit"/>	<b>DRU</b> <input type="text" value="Select DRU"/>
<b>Password</b> <input type="text"/> <small>Must be at least 8 characters long</small>	<b>Confirm Password</b> <input type="text"/>

[Create Account](#)

[Already have an account? Sign in](#)

Figure 4.15: Sign Up Page

Figure 4.16: Login Page

### 939 4.5.3 Personnel Interface

#### 940 Encoder's View

941 Figures 4.17 and 4.18 show the digitized counterpart of the form obtained from the  
 942 Iloilo Provincial Epidemiology and Surveillance Unit. As the system aims to sup-  
 943 port expandability for future features, some fields were modified to accommodate  
 944 more detailed input. It is worth noting that all of the included fields adhere to the  
 945 latest Philippine Integrated Disease and Surveillance Response (PIDSRS) Dengue  
 946 Forms, which the referenced form was based on. By doing this, it is assumed  
 947 that the targeted users will have a familiarity when deployed on a national scale.  
 948 On a further note, the case form includes the patient's basic information, dengue  
 949 vaccination status, consultation details, laboratory results, and the outcome.

DengueDash

Modules

Analytics

Forms

Case Report Form

Data Tables

Settings

CN

shadcn

m@example.com

0

Building Your Application

Data Fetching

Personal Information

Clinical Status

Personal Detail

First Name

Middle Name

Last Name

Suffix

Sex

Civil Status

Date of Birth

Address

House No.

Street

Barangay

City

Province

Vaccination

Date of First Vaccination

Date of Last Vaccination

Back

Next

Figure 4.17: First Part of Case Report Form

DengueDash

Modules

Analytics

Forms

Case Report Form

Data Tables

Settings

CN

shadcn

m@example.com

0

Building Your Application

Data Fetching

Personal Information

Clinical Status

Consultation

Date Admitted/Consulted/Seen

Is Admitted?

Date Onset of Illness

Clinical Classification

Laboratory Results

NSI

Date done (NSI)

IgG ELISA

Date done (IgG ELISA)

IgM ELISA

Date done (IgM ELISA)

PCR

Date done (PCR)

Outcome

Case Classification

Outcome

Date of Death

Back

Submit

Figure 4.18: Second Part of Case Report Form

950 Once the data generated from the case report form is validated, it will be  
951 assigned as a new case and can be accessed through the Dengue Reports page, as  
952 shown in Figure 4.19. The said page displays basic information about the patient  
953 related to a specific case, including their name, address, date of consultation, and  
954 clinical and case classifications. It is also worth noting that it only shows cases  
955 the user is permitted to view. For example, in a local Disease Reporting Unit  
956 (DRU) setting, the user can only access records that came from the same DRU.  
957 On the other hand, in a consolidated surveillance unit such as a regional and  
958 provincial quarter, its users can view all the records that came from all the DRUs  
959 that report to them. Moving forward, Figure 4.20 shows the detailed case report  
960 of the patient on a particular consultation date.

The screenshot displays the DengueDash application. On the left is a sidebar menu with the following items: Analytics, Forms, Data Tables (expanded to show Dengue Reports and Another Report), and Settings. The main content area is titled 'Building Your Application > Data Fetching' and contains a table of dengue reports. The table has columns for Case ID, Name, Barangay, City, Date Consulted, Clinical Classification, Case Classification, and an Action button labeled 'Open'. There are 12 rows of data. At the bottom of the table, there is a pagination control showing '< Previous', a page number '1' in a box, '2', '...', '218', and 'Next >'.

Case ID	Name	Barangay	City	Date Consulted	Clinical Classification	Case Classification	Action
24010965	Robinson, Raymond Todd	Balabago	Iloilo City	2024-11-27	Severe dengue	Confirmed	Open
24010975	Harmon, Michelle Donna	Yulo-Arayo	Iloilo City	2024-11-26	No warning signs	Suspect	Open
24010960	Thomas, Stephanie John	Calubihan	Iloilo City	2024-11-23	Severe dengue	Confirmed	Open
24010972	Cooper, Richard Rodney	PHHC Block 17	Iloilo City	2024-11-23	With warning signs	Probable	Open
24010583	Ramos, Joshua James	Dungan A	Iloilo City	2024-11-22	No warning signs	Confirmed	Open
24009896	Howe, Mark Curtis	Taal	Iloilo City	2024-11-21	With warning signs	Probable	Open
24010481	Lambert, Mark Laura	Aguinaldo	Iloilo City	2024-11-19	With warning signs	Suspect	Open
24009948	Cannon, Michael Victoria	Legaspi dela Rama	Iloilo City	2024-11-18	No warning signs	Confirmed	Open
24010606	Pham, Timothy Lauren	Molo Boulevard	Iloilo City	2024-11-17	Severe dengue	Confirmed	Open
24010668	Nguyen, Lisa Emily	Boilao	Iloilo City	2024-11-17	Severe dengue	Probable	Open

Figure 4.19: Dengue Reports

DengueDash

Modules

Analytics

Forms

Data Tables

Dengue Reports

Another Report

Settings

CN

shadcn

mj@example.com

0

Building Your Application > Data Fetching

Personal Information

Full Name

Thomas, Stephanie John

Date of Birth

September 19, 2010

Sex

Male

Civil Status

Separated

Full Address

4189 Rice Coves, Calubihan, Iloilo City, Iloilo

Vaccination Status

First Dose

February 11, 2024

Last Dose

April 10, 2024

Case Record #24010060

Date of Consultation

November 23, 2024

Patient Admitted?

Yes

Date Onset of Illness

November 22, 2024

Clinical Classification

Severe dengue

Laboratory Results

NSI

Pending Result

Date Done

N/A

IgG Elisa

Negative

Date Done

November 29, 2024

IgM Elisa

Equivocal

Date Done

December 2, 2024

PCR

Pending Result

Date Done

N/A

Outcome

Case Classification

Confirmed

Outcome

Dead

Date of Death

November 29, 2024

Figure 4.20: Detailed Case Report

## 4.6 User Testing

To evaluate the usability of the system, the System Usability Scale (SUS) was utilized. SUS is a Likert-scale-based questionnaire comprising 10 items that are critical to assessing system usability. A total of five participants completed the survey. Their responses were processed following the step-by-step calculation method adopted from (Babich, 2015). The resulting usability scores for each participant are shown in Table 4.5.

The average System Usability Scale (SUS) score across systems is typically 68 (Babich, 2015). In this testing, the system achieved an average SUS score of 88.5, indicating a highly positive user experience. This score suggests that participants found the system not only enjoyable to use but also intuitive enough to recommend to others. Furthermore, it demonstrates that the system is suitable

<b>Participant</b>	<b>Usability Score</b>
1	95.0
2	90.0
3	85.0
4	87.5
5	85.0
<b>Average</b>	<b>88.5</b>

Table 4.5: Computed System Usability Scores per Participant

<sup>973</sup> for real-world applications without presenting significant complexity for first-time  
<sup>974</sup> users.

# References

- 976 *About GitHub and Git - GitHub Docs.* (n.d.). Retrieved from  
 977 [https://docs.github.com/en/get-started/start-your-journey/](https://docs.github.com/en/get-started/start-your-journey/about-github-and-git)  
 978 [about-github-and-git](https://docs.github.com/en/get-started/start-your-journey/about-github-and-git)
- 979 Ahmadini, A. A. H., Naeem, M., Aamir, M., Dewan, R., Alshqaq, S. S. A., &  
 980 Mashwani, W. K. (2021). Analysis and forecast of the number of deaths,  
 981 recovered cases, and confirmed cases from covid-19 for the top four affected  
 982 countries using kalman filter. *Frontiers in Physics*, 9, 629320.
- 983 Arroyo-Marioli, F., Bullano, F., Kucinskas, S., & Rondón-Moreno, C. (2021).  
 984 Tracking r of covid-19: A new real-time estimation using the kalman filter.  
 985 *PloS one*, 16(1), e0244474.
- 986 Babich, N. (2015). *How to use the system usability scale (sus) to*  
 987 *evaluate the usability of your website.* Usability Geek. Retrieved  
 988 from [https://usabilitygeek.com/how-to-use-the-system-usability](https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/)  
 989 [-scale-sus-to-evaluate-the-usability-of-your-website/](https://usabilitygeek.com/how-to-use-the-system-usability-scale-sus-to-evaluate-the-usability-of-your-website/) (Ac-  
 990 cessed: 2025-04-26)
- 991 Bosano, R. (2023). *Who: Ph most affected by dengue in western pacific.* Retrieved  
 992 Use the date of access, from [https://news.abs-cbn.com/spotlight/12/](https://news.abs-cbn.com/spotlight/12/22/23/who-ph-most-affected-by-dengue-in-western-pacific)  
 993 [22/23/who-ph-most-affected-by-dengue-in-western-pacific](https://news.abs-cbn.com/spotlight/12/22/23/who-ph-most-affected-by-dengue-in-western-pacific)
- 994 Brady, O. J., Smith, D. L., Scott, T. W., & Hay, S. I. (2015). Dengue disease  
 995 outbreak definitions are implicitly variable. *Epidemics*, 11, 92–102.
- 996 Bravo, L., Roque, V. G., Brett, J., Dizon, R., & L’Azou, M. (2014). Epidemiology  
 997 of dengue disease in the philippines (2000–2011): a systematic literature  
 998 review. *PLoS neglected tropical diseases*, 8(11), e3027.
- 999 Carvajal, T. M., Viacrusis, K. M., Hernandez, L. F. T., Ho, H. T., Amalin, D. M.,  
 1000 & Watanabe, K. (2018). Machine learning methods reveal the temporal  
 1001 pattern of dengue incidence using meteorological factors in metropolitan  
 1002 manila, philippines. *BMC infectious diseases*, 18, 1–15.
- 1003 *Chart.js.* (n.d.). Retrieved from <https://www.chartjs.org/>
- 1004 Cheong, Y. L., Burkart, K., Leitão, P. J., & Lakes, T. (2013). Assessing weather  
 1005 effects on dengue disease in malaysia. *International journal of environmental*  
 1006 *research and public health*, 10(12), 6319–6334.



1007 Christie, T. (n.d.). *Home - Django REST framework*. Retrieved from [https://](https://www.django-rest-framework.org/)  
1008 [www.django-rest-framework.org/](https://www.django-rest-framework.org/)

1009 Colón-González, F. J., Fezzi, C., Lake, I. R., & Hunter, P. R. (2013). The effects  
1010 of weather and climate change on dengue. *PLoS neglected tropical diseases*,  
1011 7(11), e2503.

1012 Hemisphere, N. (2015). Update on the dengue situation in the western pacific  
1013 region. *Update*.

1014 Hii, Y. L., Zhu, H., Ng, N., Ng, L. C., & Rocklöv, J. (2012). Forecast of dengue  
1015 incidence using temperature and rainfall. *PLoS neglected tropical diseases*,  
1016 6(11), e1908.

1017 Joel, C. (2021, 10). *6 reasons to use Tailwind over traditional CSS*. Re-  
1018 trieved from [https://dev.to/charliejoel/6-reasons-to-use-tailwind](https://dev.to/charliejoel/6-reasons-to-use-tailwind-over-traditional-css-1nc3)  
1019 [-over-traditional-css-1nc3](https://dev.to/charliejoel/6-reasons-to-use-tailwind-over-traditional-css-1nc3)

1020 Leaflet — an open-source JavaScript library for interactive maps. (n.d.). Retrieved  
1021 from <https://leafletjs.com/>

1022 Lena, P. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use  
1023 the date of access, from <https://www.pna.gov.ph/articles/1231208>

1024 Li, X., Feng, S., Hou, N., Li, H., Zhang, S., Jian, Z., & Zi, Q. (2022). Applications  
1025 of kalman filtering in time series prediction. In *International conference on*  
1026 *intelligent robotics and applications* (pp. 520–531).

1027 Ligue, K. D. B., & Ligue, K. J. B. (2022). Deep learning approach to forecasting  
1028 dengue cases in davao city using long short-term memory (lstm). *Philippine*  
1029 *Journal of Science*, 151(3).

1030 Perla. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use the  
1031 date of access, from <https://www.pna.gov.ph/articles/1231208>

1032 RabDashDC. (2024). *Rabdash dc*. Retrieved Use the date of access, from [https://](https://rabdash.com)  
1033 [rabdash.com](https://rabdash.com)

1034 Runge-Ranzinger, S., Kroeger, A., Oliario, P., McCall, P. J., Sánchez Tejeda, G.,  
1035 Lloyd, L. S., ... Coelho, G. (2016). Dengue contingency planning: from  
1036 research to policy and practice. *PLoS neglected tropical diseases*, 10(9),  
1037 e0004916.

1038 Shadcn. (n.d.). *Introduction*. Retrieved from <https://ui.shadcn.com/docs>  
1039 *Tailwind CSS - Rapidly build modern websites without ever leaving your HTML*.  
1040 (n.d.). Retrieved from <https://tailwindcss.com/>

1041 Watts, D. M., Burke, D. S., Harrison, B. A., Whitmire, R. E., & Nisalak, A.  
1042 (2020). Effect of temperature on the transmission of dengue virus by aedes  
1043 aegypti. *The American Journal of Tropical Medicine and Hygiene*, 36(1),  
1044 143–152.

1045 *What is Postman? Postman API Platform*. (n.d.). Retrieved from [https://](https://www.postman.com/product/what-is-postman/)  
1046 [www.postman.com/product/what-is-postman/](https://www.postman.com/product/what-is-postman/)

1047 WHO. (2023). *Dengue - global situation*. Retrieved Use the date of ac-  
1048 cess, from <https://www.who.int/emergencies/disease-outbreak-news/>

1049       item/2023-DON498

1050   WHO.   (2024).   *Dengue and severe dengue*.   Retrieved Use the date

1051       of access, from [https://www.who.int/news-room/fact-sheets/detail/](https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue)

1052       dengue-and-severe-dengue

1053   *Why Visual Studio Code?*   (2021, 11).   Retrieved from [https://code](https://code.visualstudio.com/docs/editor/whyvscode)

1054       .visualstudio.com/docs/editor/whyvscode

1055   World Health Organization (WHO). (2018). Dengue and severe dengue in the

1056       philippines. *WHO Dengue Factsheet*. (Available at: [https://www.who](https://www.who.int)

1057       .int)

1058   Zhou, S., & Malani, P. (2024). What is dengue? *Jama*, 332(10), 850–850.

1059   Zod. (n.d.). *TypeScript-first schema validation with static type inference*. Re-

1060       trieved from <https://zod.dev/?id=introduction>

1061 **Appendix A**

1062 **Appendix Title**

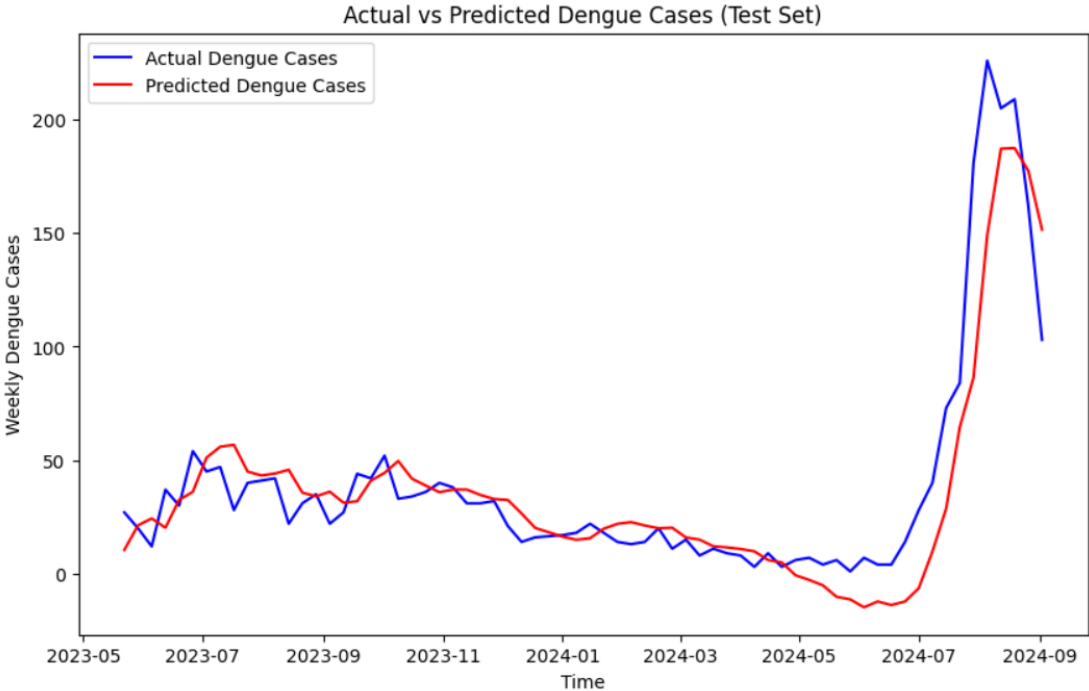


Figure A.1: LSTM Prediction Results for Test Set

## 1063 **Appendix B**

### 1064 **Resource Persons**

1065 **Mr. Firstname1 Lastname1**

1066 Role1

1067 Affiliation1

1068 emailaddr1@domain.com

1069 **Ms. Firstname2 Lastname2**

1070 Role2

1071 Affiliation2

1072 emailaddr2@domain.net

1073 ....