

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 February 23, 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 16.18, 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	1 Introduction	1
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	2 Review of Related Literature	6
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	3 Research Methodology	10
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	3.2 Kalman Filter	15
71	3.3 Kalman Filter Methodology with Matrix Calculations	16
72	3.3.1 Integrate the Predictive Model into a Web-Based Data An-	
73	alytics Dashboard	17
74	3.3.2 System Development Framework	18
75	3.3.3 Design, Building, Testing, and Integration	18
76	3.4 Development Tools	19
77	3.4.1 Software	19
78	3.4.2 Hardware	20
79	3.4.3 Packages	20
80	3.5 Calendar of Activities	22
81	4 Results and Discussion/System Prototype	23
82	4.1 Data Gathering	23
83	4.2 Exploratory Data Analysis	24
84	4.3 Outbreak Detection	28
85	4.4 Model Training	29
86	4.4.1 LSTM Model	30
87	4.4.2 ARIMA Model	32

88	4.4.3	Seasonal ARIMA (SARIMA) Model	33
89	4.4.4	Kalman Filter Model	35
90	4.5	Preliminary System Requirements	36
91	4.5.1	Backend Requirements	36
92	4.5.2	User Interface Requirements	37
93	4.5.3	Security and Validation Requirements	39
94	4.5.4	Testing Process	40
95	4.6	System Prototype	41
96	4.6.1	Guest Interface	41
97	4.6.2	Personnel Interface	42
98		References	46
99		A Appendix Title	49
100		B Resource Persons	50

List of Figures

102	3.1	Workflow for forecasting the number of weekly dengue cases . . .	10
103	4.1	Snippet of the Combined Dataset	24
104	4.2	Data Contents	24
105	4.3	Dataset Statistics	25
106	4.4	Distribution of Dengue Cases	25
107	4.5	Frequency of Dengue Cases in Different Intervals	26
108	4.6	Trend of Dengue Cases	27
109	4.7	Ranking of Correlations	27
110	4.8	Ranking of Correlations (with lagged effects)	28
111	4.9	Comparison of Window Sizes	31
112	4.10	ARIMA Prediction Results for Test Set	32
113	4.11	Seasonal ARIMA Prediction Results for Test Set	33
114	4.12	Kalman Filter Prediction Results for Test Set	35
115	4.13	Entity-Relationship Database Schema Hybrid Diagram for DengueDash	
116		Database Structure	36
117	4.14	Use Case Diagram for Admin	37
118	4.15	Use Case Diagram for Encoder	38

119	4.16 Testing Process for DengueWatch	40
120	4.17 Dashboard for Guests	41
121	4.18 Login Page for Users	42
122	4.19 First Part of Case Report Form	43
123	4.20 Second Part of Case Report Form	43
124	4.21 Dengue Reports	44
125	4.22 Detailed Case Report	45
126	A.1 LSTM Prediction Results for Test Set	49

127 **List of Tables**

128	3.1	Timetable of Activities for 2024	22
129	3.2	Timetable of Activities for 2025	22
130	4.1	Comparison of Models	29
131	4.2	Comparison of Window Sizes	31

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.

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

163 1.2 Problem Statement

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

176 1.3 Research Objectives

177 1.3.1 General Objective

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

182 1.3.2 Specific Objectives

183 Specifically, this study aims to:

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

198 1.4 Scope and Limitations of the Research

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

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

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

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

224 1.5 Significance of the Research

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

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

248 awareness campaigns and community engagement initiatives, empowering
249 residents with knowledge and preventative measures to protect themselves
250 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

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

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

287 **2.3 Existing System: RabDash DC**

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

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

304 2.4 Deep Learning

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

318 2.5 Kalman Filter

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

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

2.6 Weather Data

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

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

2.7 Chapter Summary

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

The literature further underscores the significance of weather variables—such as temperature and rainfall—in forecasting dengue cases. Studies demonstrate that these variables contribute to accurate outbreak prediction models. Leveraging these insights, our study will incorporate both weather data and historical 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.

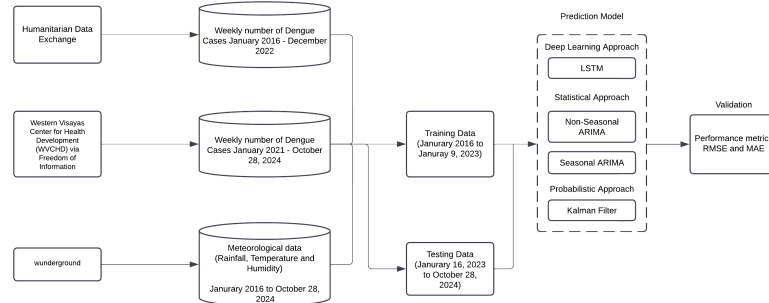


Figure 3.1: Workflow for forecasting the number of weekly dengue cases

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.

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

419 **Data Integration and Preprocessing**

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

427 **Exploratory Data Analysis (EDA)**

- 428 • Analyzed trends, seasonality, and correlations between dengue cases and
429 weather factors.
- 430 • Created visualizations like time series plots and scatterplots to highlight
431 relationships and patterns in the data.

432 **Outbreak Detection**

433 To detect outbreaks, we computed the outbreak threshold value of dengue cases
434 using the formula,

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

435 where μ is the historical mean and σ is the standard deviation.

436 3.1.2 Develop and Evaluate Deep Learning Models for 437 Dengue Case Forecasting

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

443 Data Preprocessing

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

450 LSTM Model

451 To prepare the data for LSTM, a sliding window approach was utilized. Sequences
452 of weeks of normalized features were constructed as input, while the dengue case
453 count for the subsequent week was set as the target variable. This approach en-
454 sured that the model leveraged temporal dependencies in the data for forecasting.

455 The LSTM model was designed using the TensorFlow and Keras libraries. The
456 architecture comprised the following layers:

- 457 • Input Layer: Accepting sequences of weeks with three features (rainfall, max
458 temperature, and humidity).
- 459 • LSTM Layer: A single LSTM layer with 64 units and ReLU activation,
460 capturing temporal dependencies and feature interactions.
- 461 • Dense Output Layer: A fully connected layer with a single neuron to predict
462 the dengue cases for the next week.

463 The model was trained for 100 epochs implementing early stopping with a batch
464 size of 1, enabling fine-grained weight updates. The training dataset consisted

465 of 80% of the sequences, while the remaining 20% was used as the test set to
466 evaluate model performance. Validation loss was monitored during training to
467 assess model generalization.

468 The training process was conducted using three distinct window sizes (5 weeks,
469 10 weeks, and 20 weeks) to determine the optimal sequence length of weeks to
470 input into the LSTM model for improved forecasting performance.

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

475 **ARIMA**

476 The ARIMA model was employed to forecast weekly dengue cases using historical
477 weather data (rainfall, max temperature, and humidity) as exogenous variables
478 and historical case counts as the primary dependent variable. The dataset was
479 split into training (80

- 480 • p (autoregressive order): 0 to 3
- 481 • d (differencing order): 0 to 2
- 482 • q (moving average order): 0 to 3

483 The combinations of these parameters were evaluated by fitting an ARIMA model
484 for each set of (p, d, q) values. The model's performance was assessed using the
485 mean squared error (MSE) between the predicted and actual dengue cases in the
486 test set. The combination yielding the lowest MSE was selected as the optimal
487 parameter configuration.

488 The fitted ARIMA model was used to forecast weekly dengue cases for the
489 test dataset. Predictions were directly assigned to the PredictedCases column in
490 the test dataset.

491 **Steps to Create the ARIMA Model:**

- 492 1. **Data Preprocessing:** Prepare the dataset by handling any missing values
493 and scaling the data if necessary to improve model convergence and stability.

- 494 **2. Hyperparameter Tuning:** Use a grid search on potential ARIMA param-
 495 eters (p, d, q) to identify the configuration that minimizes error. The optimal
 496 parameters were found to be **(1, 2, 2)**.
- 497 **3. Model Training:**
- 498 • Set the number of iterations to 400 to ensure thorough training and
 499 convergence.
 - 500 • Train the ARIMA model on 80% of the data and reserve 20% for test-
 501 ing.

502 **Kalman Filter:**

- 503 • **Input Variables:** The target variable (Cases) was modeled using three re-
 504 gressors: rainfall, max temperature, and humidity.
- 505 • **Training and Testing Split:** The dataset was split into 80% training and
 506 20% testing to evaluate model performance.
- 507 • **Observation Matrix:** The Kalman Filter requires an observation matrix,
 508 which was constructed by adding an intercept (column of ones) to the re-
 509 gressors.

510 The Kalman Filter’s EM method was employed for training, iteratively esti-
 511 mating model parameters over 10 iterations. The smooth method was used to
 512 compute the smoothed state estimates for the training data. Observation matri-
 513 ces for the test data were constructed similarly, ensuring compatibility with the
 514 trained model.

515 **3.2 Kalman Filter**

- 516 • **Input Variables:** The target variable (Cases) was modeled using three
 517 regressors: rainfall, max temperature, and humidity.
- 518 • **Training and Testing Split:** The dataset was split into 80% training and
 519 20% testing to evaluate model performance.
- 520 • **Observation Matrix:** The Kalman Filter requires an observation matrix,
 521 which was constructed by adding an intercept (column of ones) to the re-
 522 gressors.

523 The Kalman Filter’s EM method was employed for training, iteratively esti-
 524 mating model parameters over 10 iterations. The smooth method was used to
 525 compute the smoothed state estimates for the training data. Observation matri-
 526 ces for the test data were constructed similarly, ensuring compatibility with the
 527 trained model.

528 3.3 Kalman Filter Methodology with Matrix Cal- 529 culations

530 **Measurement Acquisition:** Obtain the measurement: (z_k) of the system’s state
 531 with associated confidence. This measurement matrix provides a noisy observation
 532 of the true state.

533 The dataset was split into training and test sets to evaluate the Kalman Filter’s
 534 performance and generalizability:

- 535 • **Training Set:** 80% of the data was used for training, enabling the Kalman
 536 Filter model to capture key patterns.
- 537 • **Test Set:** The remaining 20% of the data was reserved for testing.

538 Prediction Step:

- 539 • Predict the next state:

$$\hat{x}_{k|k-1} = A\hat{x}_{k-1|k-1} + Bu_k$$

- 540 • Update the state covariance:

$$P_{k|k-1} = AP_{k-1|k-1}A^T + Q$$

541 where Q is the process noise covariance matrix.

542 **Compute Residual:** Calculate the residual:

$$y_k = z_k - H\hat{x}_{k|k-1}$$

543 where H is the observation matrix. This residual represents the new information
 544 from the measurement.

545 **Scaling Factor (Kalman Gain):**

- 546 • Compute the Kalman Gain:

$$K_k = P_{k|k-1}H^T (HP_{k|k-1}H^T + R)^{-1}$$

547 where R is the measurement noise covariance matrix.

- 548 • The Kalman Gain determines the weight of the measurement relative to the
549 prediction.

550 **State Update:**

- 551 • Update the state estimate:

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k y_k$$

552 blending the prediction and measurement.

553 **Uncertainty Update:**

- 554 • Update the state covariance:

$$P_{k|k} = (I - K_k H)P_{k|k-1}$$

555 where I is the identity matrix.

556 **3.3.1 Integrate the Predictive Model into a Web-Based**
557 **Data Analytics Dashboard**

558 **Dashboard Design and Development**

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

563 **Model Integration and Deployment**

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

566 **3.3.2 System Development Framework**

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

573 **3.3.3 Design, Building, Testing, and Integration**

574 **Design and Development**

575 After brainstorming and researching the most appropriate type of application to
576 accommodate both the prospected users and the proposed solutions, the team has
577 decided to proceed with a web application. Given the time constraints and avail-
578 able resources, we believe this is the most pragmatic and practical move. The next
579 step is to select modern and stable frameworks that align with the fundamental
580 ideas we have learned at the university. The template obtained from WVCHD
581 and Iloilo Provincial Epidemiology and Surveillance Unit was meticulously ana-
582 lyzed to create use cases and develop a preliminary well-structured database that
583 adheres to the requirements needed to produce a quality application. The said use
584 cases serve as the basis of general features. Part by part, these are converted into
585 code, and with the help of selected libraries and packages, it resulted in the de-
586 sired outcome that may still modified and extended since it is continuously being
587 developed.

588 **Testing and Integration**

589 Each feature will be rigorously user-tested to ensure quality assurance, with par-
590 ticular emphasis on prerequisite features, as development cannot progress properly
591 if these fail. Moreover, integration between each feature serves as a pillar for a
592 cohesive user experience. Presently, we have not been able to use performance
593 metrics to measure the system's performance, as developing and connecting the
594 core features is the utmost priority.

595 3.4 Development Tools

596 3.4.1 Software

597 Github

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

603 Visual Studio Code

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

608 Django

609 Django is a free and open-sourced Python-based web framework that offers an
610 abstraction to develop and maintain a secure web application. As this research
611 aims to create a well-developed and maintainable application, it is in the best
612 interest to follow an architectural pattern that developers and contributors in the
613 future can understand. Since Django adheres to Model-View-Template (MVT)
614 that promotes a clean codebase by separating data models, business logic, and
615 presentation layers, it became the primary candidate for the application’s back-
616 bone.

617 Next.js

618 A report by Statista (2024) claims that React is the most popular front-end frame-
619 work among web developers. However, React has limitations that can be a nui-
620 sance in rapid software development, which includes routing and performance op-
621 timizations. This is where Next.js comes in—a framework built on top of React.

622 It offers solutions for React’s deficiency, making it a rising star in the framework
623 race.

624 **Postman**

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

630 **3.4.2 Hardware**

631 The web application is continuously being developed on laptop computers with
632 minimum specifications of an 11th-generation Intel i5 CPU and 16 gigabytes of
633 RAM.

634 **3.4.3 Packages**

635 **Django REST Framework**

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

640 **Leaflet**

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

647 **Chart.js**

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

655 **Tailwind CSS**

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

664 **Shadcn**

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

670 **Zod**

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

3.5 Calendar of Activities

A Gantt chart showing the schedule of the activities is included below. Each 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

4.2 Exploratory Data Analysis

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

- **Time.** Weekly timestamps (e.g. "2011-w1")
- **Rainfall.** Weekly average rainfall (mm)
- **MaxTemperature, AverageTemperature, MinTemperature.** Weekly temperature data (C)
- **Wind.** Wind speed (m/s)
- **Humidity.** Weekly average humidity (%)
- **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%.

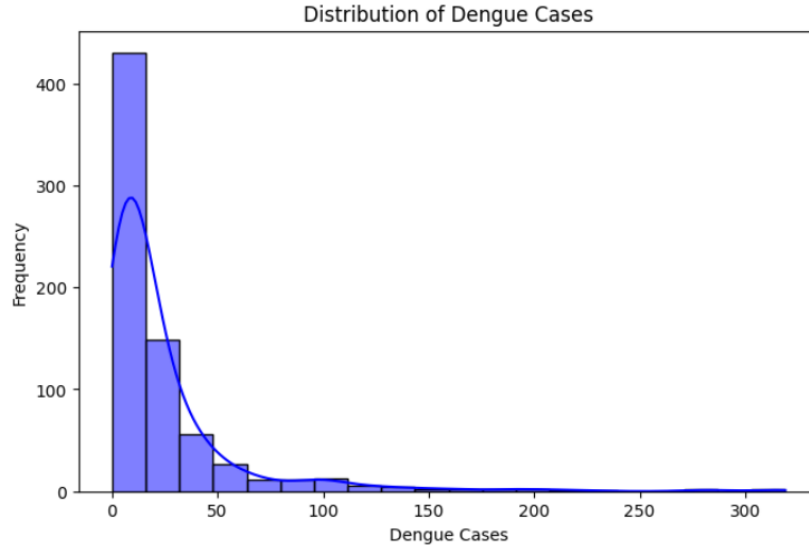


Figure 4.4: Distribution of Dengue Cases

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+

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

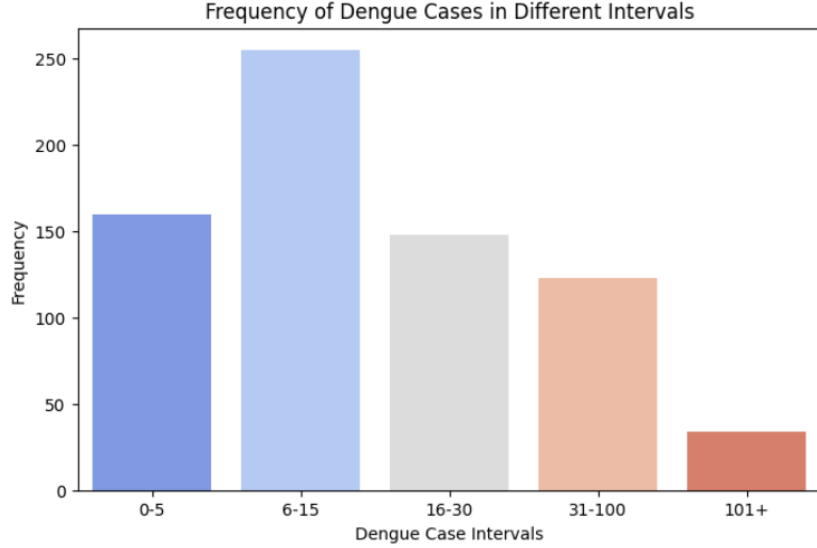


Figure 4.5: Frequency of Dengue Cases in Different Intervals

720

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

727 Figure 4.7 shows the ranking of correlation coefficients between dengue cases
 728 and selected features, including rainfall, humidity, maximum temperature, aver-
 729 age temperature, minimum temperature, and wind speed. Among these, rainfall
 730 exhibits the highest positive correlation with dengue cases (correlation coefficient
 731 0.13), indicating that increased rainfall may contribute to higher cases counts.
 732 This aligns with existing studies suggesting that stagnant water from heavy rain-
 733 fall creates breeding grounds for mosquitos. It is followed by humidity (0.10),
 734 suggesting that higher humidity levels may enhance mosquito reproduction, lead-
 735 ing to more dengue cases. Temperature has a weak to moderate positive corre-
 736 lation with dengue cases, with maximum temperature (0.09) showing a stronger
 737 relationship than average and minimum temperature.

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

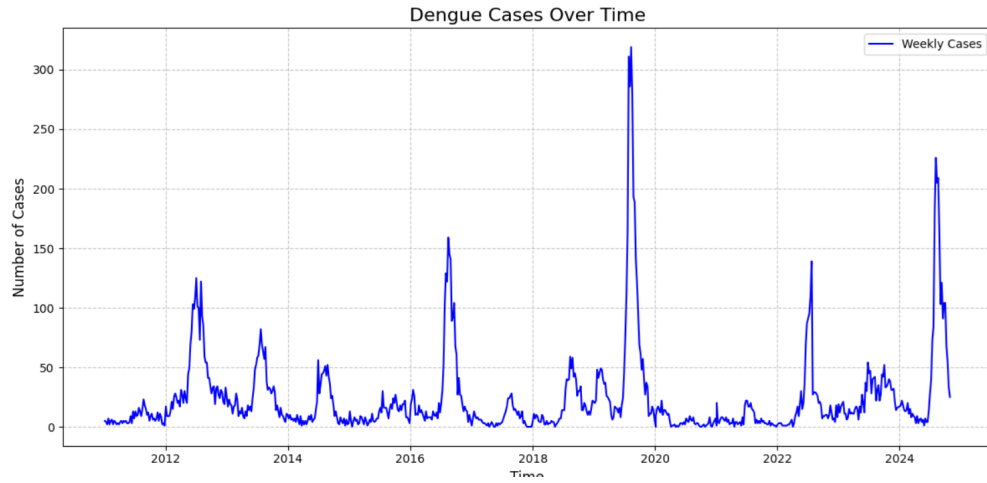


Figure 4.6: Trend of Dengue Cases

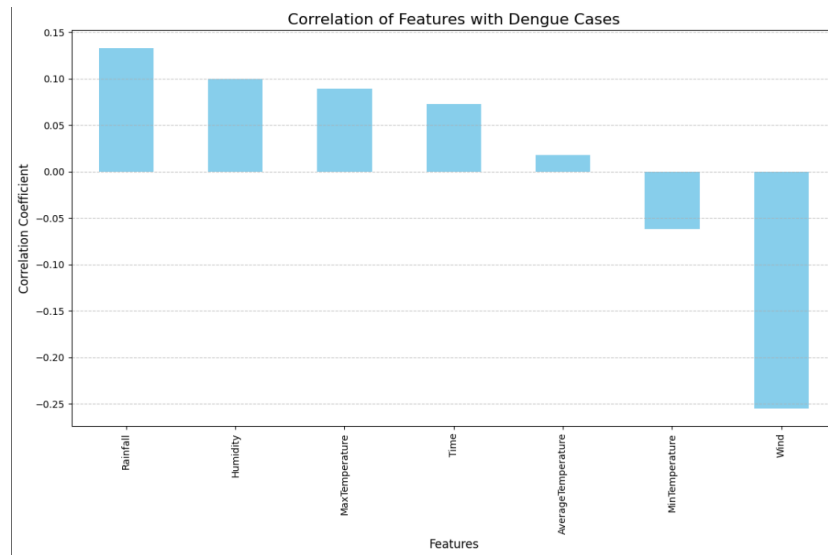


Figure 4.7: Ranking of Correlations

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.

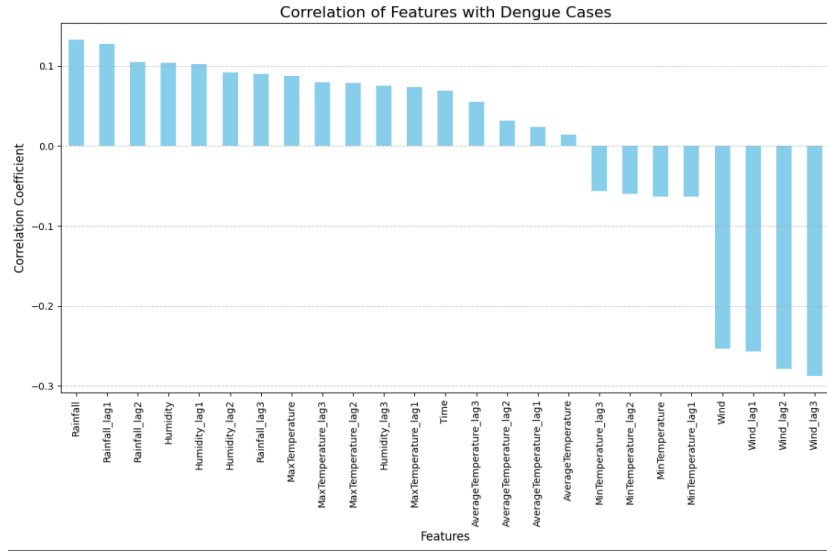


Figure 4.8: Ranking of Correlations (with lagged effects)

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 μ is the historical mean and σ 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.

4.4 Model Training

The proposed Dengue Watch system utilized four distinct models to forecast weekly dengue cases: Long Short-Term Memory (LSTM) networks, Autoregressive Integrated Moving Average (ARIMA), Seasonal ARIMA (SARIMA), and Kalman Filter. Each model was trained on a dataset containing 720 weeks of historical dengue cases from 2011 to 2024, with meteorological variables such as max temperature, humidity, and rainfall.

Using SARIMA and LSTM for dengue forecasting requires an adaptive approach due to seasonal changes and long-term trends. Dengue case data is updated every month, and weather data can be extracted manually every week. By continuously monitoring performance, incorporating external factors, and updating the model regularly (preferably monthly or semi-annually), forecasting accuracy can be maintained. If drastic environmental or epidemiological changes occur, more frequent retraining is necessary. This ensures that public health interventions remain proactive, effectively mitigating dengue outbreaks.

To optimize predictive performance, hyperparameter tuning was conducted individually for each model, refining parameters to achieve the most accurate and reliable forecasts. Following training, the models were rigorously evaluated against the dataset using a set of key performance metrics, including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Error (MAE).

The table below provides a summary and comparative analysis of each model's results across these metrics, offering insights into the strengths and limitations of each forecasting technique for dengue case prediction in Iloilo City. The acceptable threshold for Mean Absolute Error (MAE) in forecasting dengue cases for it to be considered accurate can vary depending on the context. However, related studies often serve as benchmarks, with commonly cited acceptable values ranging from 20 to 30. For this study, we have established a threshold of 15 to emphasize the significance of accurate dengue prediction.

Model	MSE	RMSE	MAE
LSTM	277.71	16.18	9.44
Seasonal ARIMA (2, 0, 2) (0, 1,1)	1109.69	33.31	18.09
ARIMA (1, 2, 2)	1521.48	39.01	25.80
Kalman Filter	1474.82	38.40	22.34

Table 4.1: Comparison of Models

782 4.4.1 LSTM Model

783 The LSTM model architecture consisted of an input layer, a single LSTM layer
784 with 64 units and ReLU activation, followed by a dense layer with a single output
785 neuron to predict the dengue case count. Key hyperparameters included:

- 786 • Window Size: 5, 10, and 20 weeks, representing the time steps used in the
787 sequence data for each prediction.
- 788 • Epochs: 100 epochs were used for training, balancing sufficient training
789 time with computational efficiency also implementing early stopping to avoid
790 overfitting.
- 791 • Batch Size: 1, allowing the model to process one sequence at a time, which
792 is beneficial for small datasets but increases training time.
- 793 • Optimizer: The Adam optimizer was chosen for its adaptive learning capa-
794 bilities and stability in training. A custom learning rate of 0.0001 was set
795 to ensure gradual convergence and minimize risk of overfitting.

796 The dataset was split into training and test sets to evaluate the model's per-
797 formance and generalizability:

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

803 The training process was conducted using three distinct window sizes—5 weeks,
804 10 weeks, and 20 weeks—to identify the optimal sequence length of weeks for input
805 into the LSTM model, thereby enhancing forecasting performance. The following
806 plots illustrate the performance of the model in predicting dengue cases for each
807 of the specified window sizes.

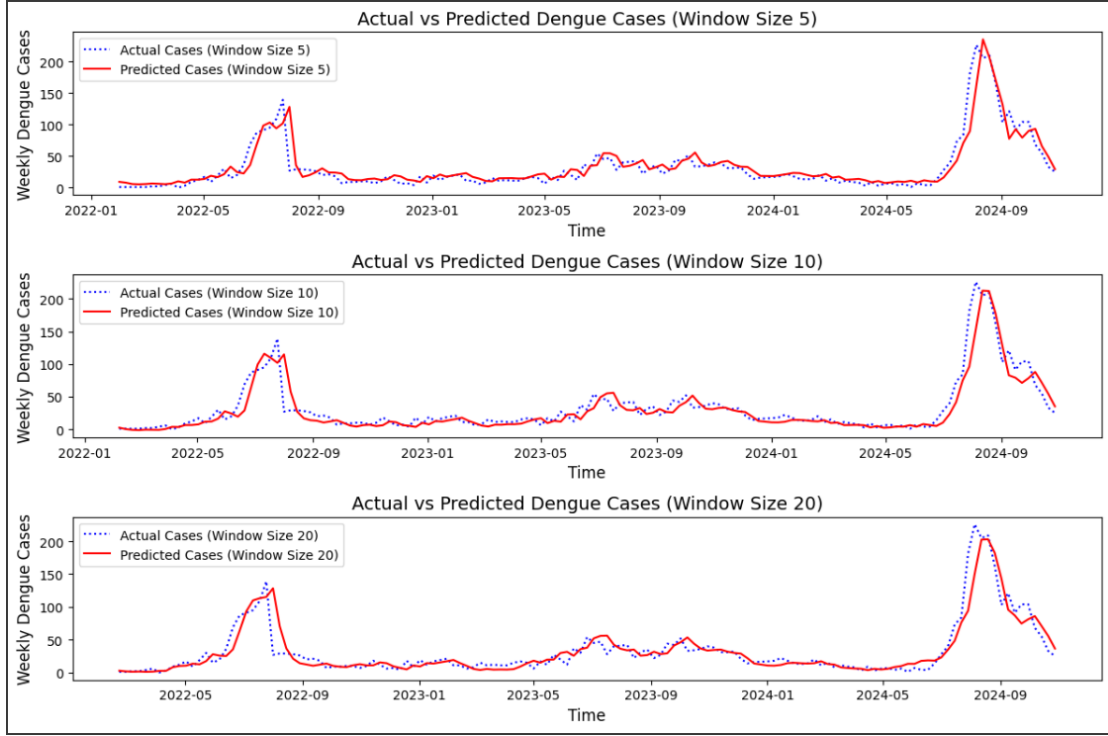


Figure 4.9: Comparison of Window Sizes

808 The evaluation metrics included Mean Squared Error (MSE), Root Mean
 809 Squared Error (RMSE) and Mean Absolute Error (MAE), which assess the accu-
 racy of the model's predictions.

Window Size	MSE	RMSE	MAE
5	282.69	16.81	9.29
10	277.71	16.18	9.44
15	289.63	17.02	9.30

Table 4.2: Comparison of Window Sizes

810

811 The results indicate that a window size of 10 weeks provides the most accurate
 812 predictions, as evidenced by the lowest MSE and RMSE values. Although the
 813 other two window sizes exhibit higher MAE values, the difference is not that
 814 noticeable, with only approximately 0.1 differences. This suggests that using a
 815 10-week sequence length effectively balances the temporal dependencies captured
 816 by the model and the computational complexity of training.

817 Training and Testing Data Division for ARIMA 818 and Seasonal Arima

819 Both models utilized an **80%-20% split** to evaluate generalizability:

- 820 • **Training Set:** 80% of the data was used for training, allowing the models
821 to learn underlying patterns in the dataset.
- 822 • **Test Set:** 20% of the data was reserved for testing, providing an unbiased
823 assessment of the models' performance on unseen data.

824 4.4.2 ARIMA Model

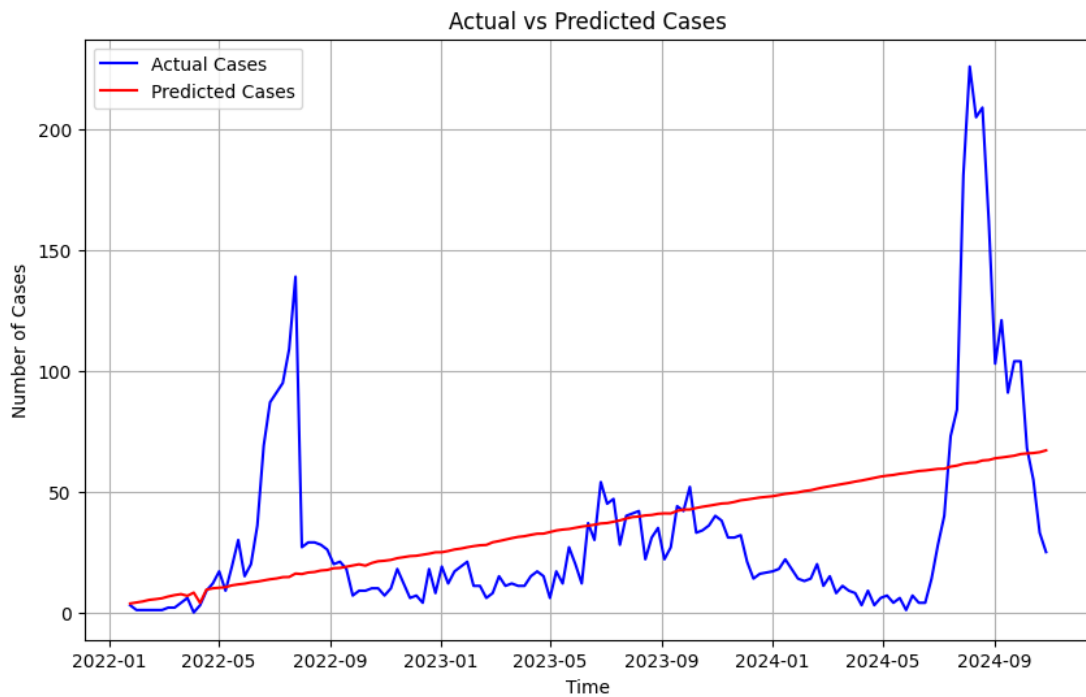


Figure 4.10: ARIMA Prediction Results for Test Set

825 The ARIMA model was developed to capture non-seasonal trends in the data. To
826 determine the best model configuration, grid search was used to explore various
827 combinations of ARIMA parameters, ultimately selecting **ARIMA(1, 2, 2)**. The
828 model was iteratively refined over **400 iterations** to ensure convergence to an
829 optimal solution.

830 The model's performance was assessed using regression metrics to evaluate its
831 forecasting capability:

- 832 • Mean Squared Error (MSE): Quantifies average squared prediction error.
- 833 • Root Mean Squared Error (RMSE): Measures average prediction error on
834 the data's original scale.
- 835 • Mean Absolute Error (MAE): Measures the average magnitude of the abso-
836 lute errors between the predicted and actual values.

837 The ARIMA model yielded the following error metrics:

- 838 • **MSE (Mean Squared Error):** 1521.48
- 839 • **RMSE (Root Mean Squared Error):** 39.01
- 840 • **MAE (Mean Absolute Error):** 25.80

841 4.4.3 Seasonal ARIMA (SARIMA) Model

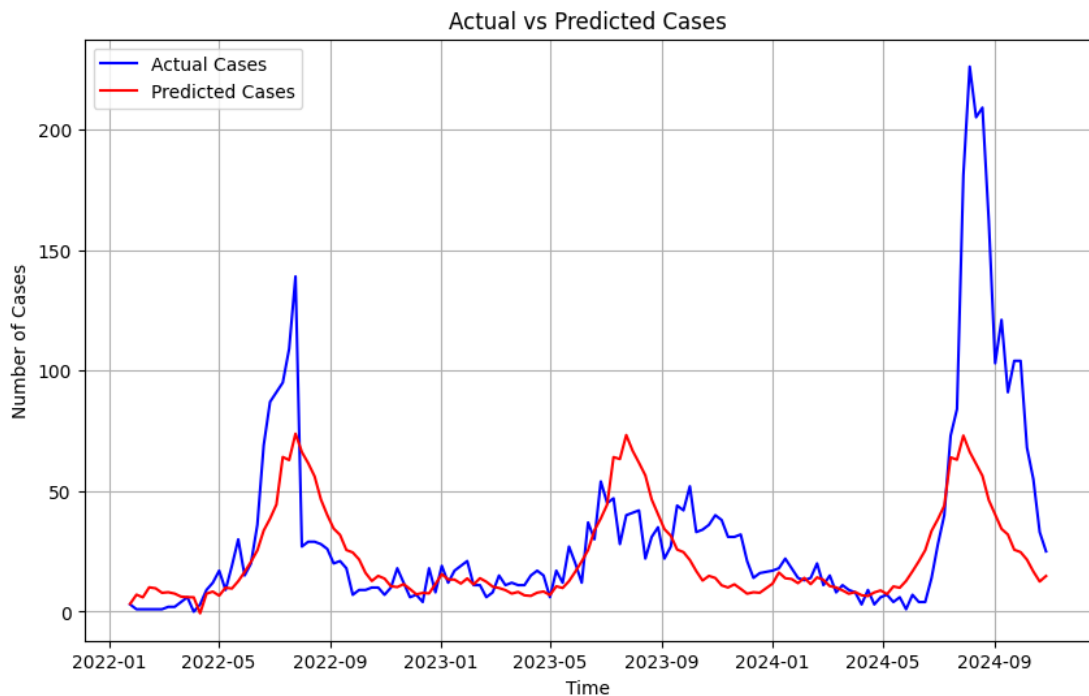


Figure 4.11: Seasonal ARIMA Prediction Results for Test Set

842 This model incorporates seasonal parameters, which were tuned using grid search
843 to find the best configuration: **SARIMA(2, 0, 2)(0, 1, 1)[52]**. As with ARIMA,
844 **400 iterations** were applied to ensure a robust fit.

845 The model's performance was assessed using regression metrics to evaluate its
846 forecasting capability:

- 847 • Mean Squared Error (MSE): Quantifies average squared prediction error.
- 848 • Root Mean Squared Error (RMSE): Measures average prediction error on
849 the data's original scale.
- 850 • Mean Absolute Error (MAE): Measures the average magnitude of the abso-
851 lute errors between the predicted and actual values.

852 The SARIMA model yielded the following error metrics:

- 853 • **MSE:** 1109.69
- 854 • **RMSE:** 33.31
- 855 • **MAE:** 18.09

856 The SARIMA model outperformed the ARIMA model in terms of lower MSE and
857 RMSE values, indicating its effectiveness in capturing the seasonal patterns in the
858 data.

859 4.4.4 Kalman Filter Model

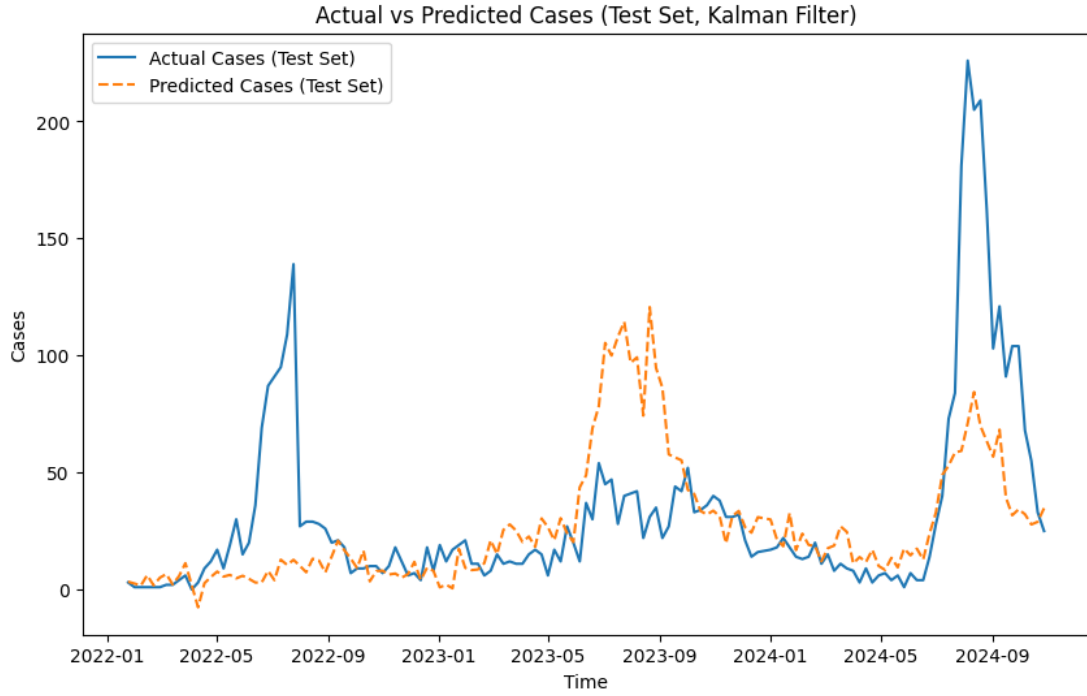


Figure 4.12: Kalman Filter Prediction Results for Test Set

860 **Model Evaluation:** Upon testing, the Kalman Filter produced the following
861 error metrics:

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

862 These results indicate the model's performance in predicting dengue cases,
863 where lower errors suggest a better fit to the observed data.

4.5 Preliminary System Requirements

4.5.1 Backend Requirements

Database Structure Design

Determining how data flows and how it would be structured is crucial in creating the system as it defines how extendible and flexible it would be for future features and updates. Thus, creating a comprehensive map of data ensures proper normalization that eliminates data redundancy and improves data integrity. Figure 4.13 depicts the designed database schema that showcases the relationship between the application's entities.

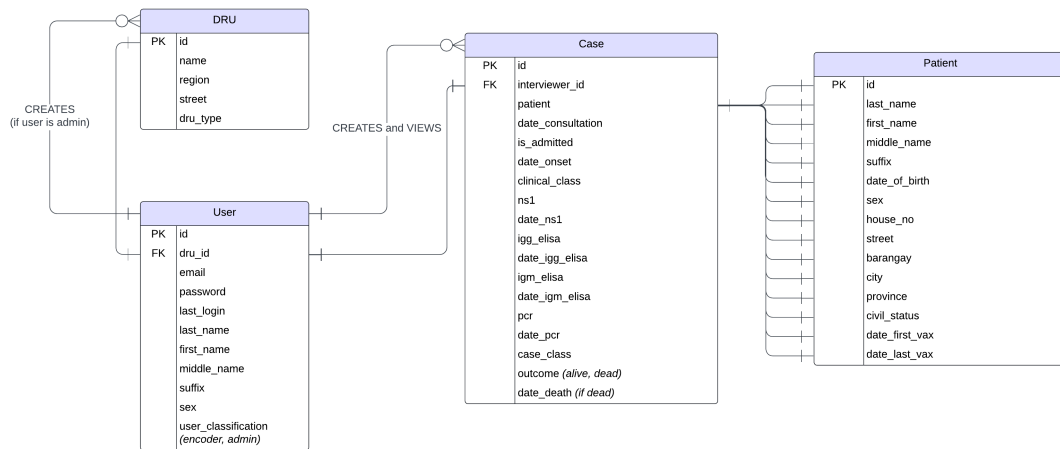


Figure 4.13: Entity-Relationship Database Schema Hybrid Diagram for DengueDash Database Structure

873 4.5.2 User Interface Requirements

874 Admin Interface

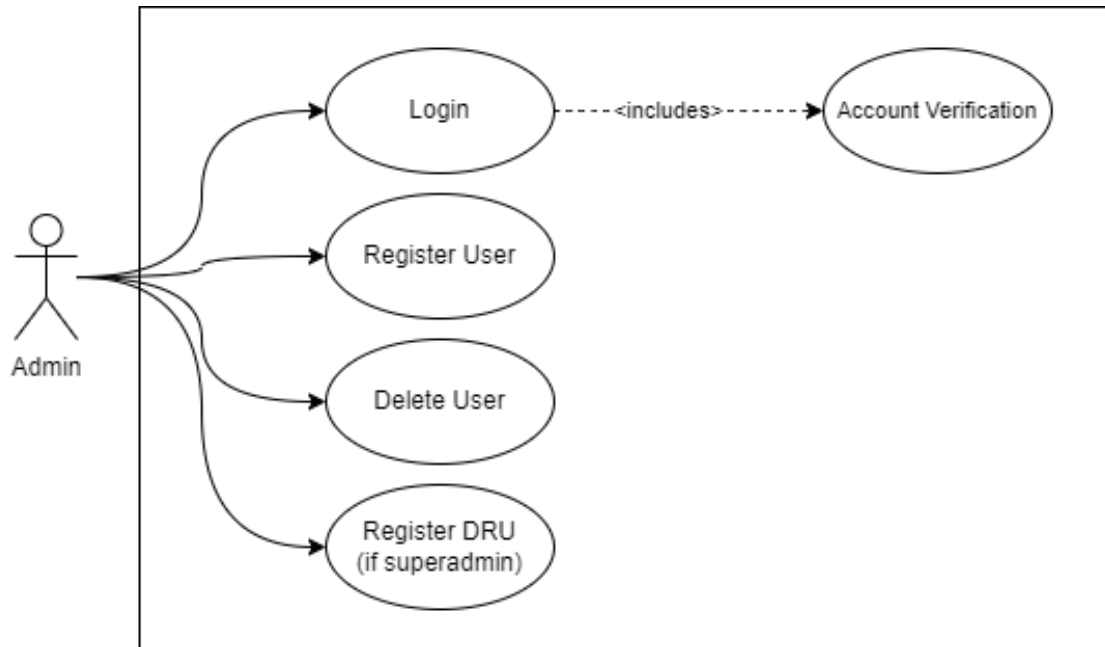


Figure 4.14: Use Case Diagram for Admin

875 Figure 4.14 shows the possible tasks that the admin can do in the application. To
876 protect the integrity of data, only the admins can register and delete accounts.
877 Both account creation and deletion will be done within the application.

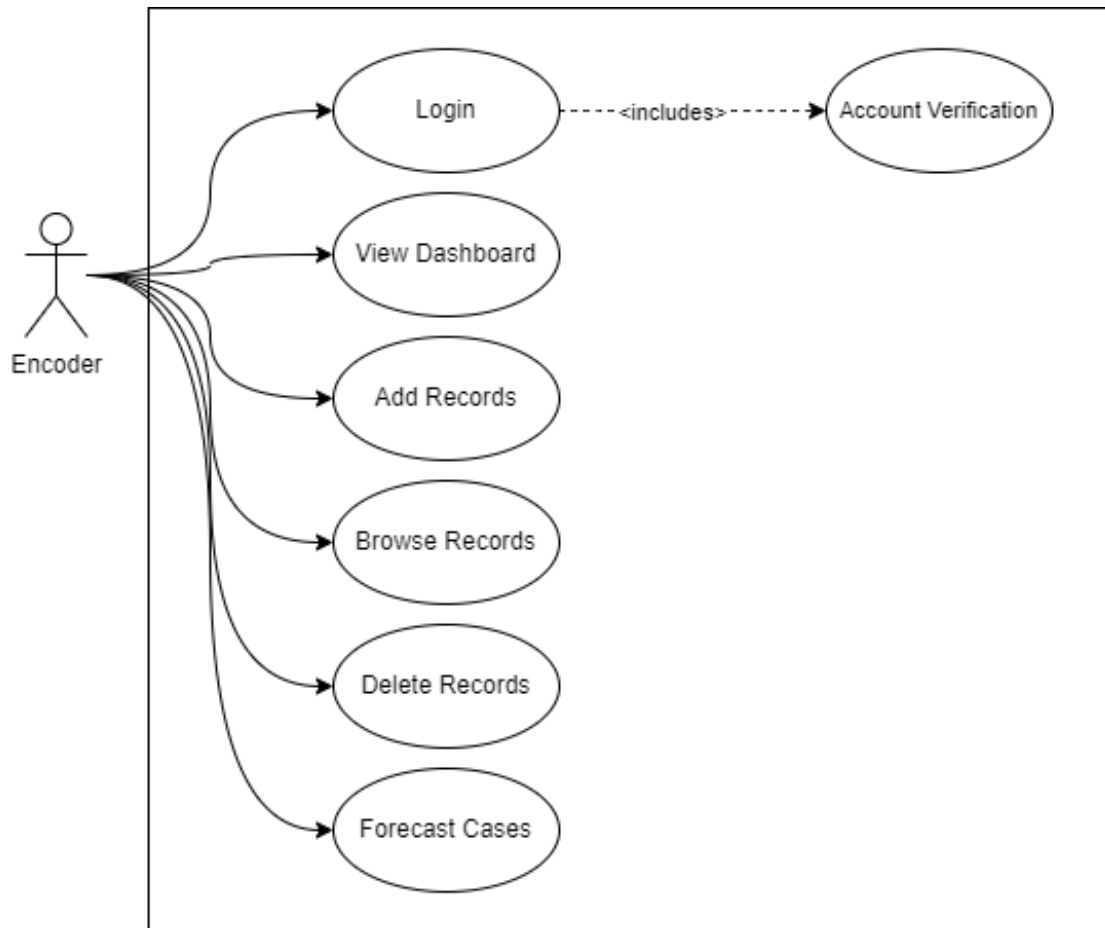


Figure 4.15: Use Case Diagram for Encoder

879 Figure 4.15, on the other hand, illustrates the use cases for the system's primary
880 users. Since only the admin accounts can register a user, the registration process
881 is not part of it. Instead, the main features, which include reporting and viewing
882 records, are the only permitted actions for this type of user. The said processes
883 can be done in the application by filling out a form with details required for each
884 dengue case. As data is entered, it will be consolidated for model training and
885 used for further forecasting of dengue cases.

886 4.5.3 Security and Validation Requirements

887 Password Encryption

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

896 Authentication

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

902 Data Validation

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

911 4.5.4 Testing Process



Figure 4.16: Testing Process for DengueWatch

912 As the system requirements and functionalities have been mentioned above, it
913 is important to implement testing to validate the system's performance and effi-
914 cacy. Since dengue reports include confidential information, anonymized historical
915 dengue reports were used to train the model and create the foundational architec-
916 ture of the system. By using functional tests, data validation and visualization can
917 be ensured for further continual improvements. Security testing is also important
918 as it is needed to safeguard confidential information when the system is deployed.
919 It includes proper authentication, permission views, and mitigating common in-
920 jection attacks. Finally, a user acceptance test from the prospected users, in this
921 case, the Iloilo City Epidemiology and Surveillance Unit, is crucial to assess its
922 performance and user experience. It enables the developers to confirm if the sys-
923 tem meets the needs of the problem, and once confirmed, it will be deployed and
924 further evaluated to ensure stability and reliability in live operation.

4.6 System Prototype

4.6.1 Guest Interface

The Guest Interface is intended for all visitors of the web application. It shows the related statistics for dengue cases in a particular area and time. As the system is still in its testing phase, the data converted into charts shown in Figure 4.17 are generated from Python's Faker library.

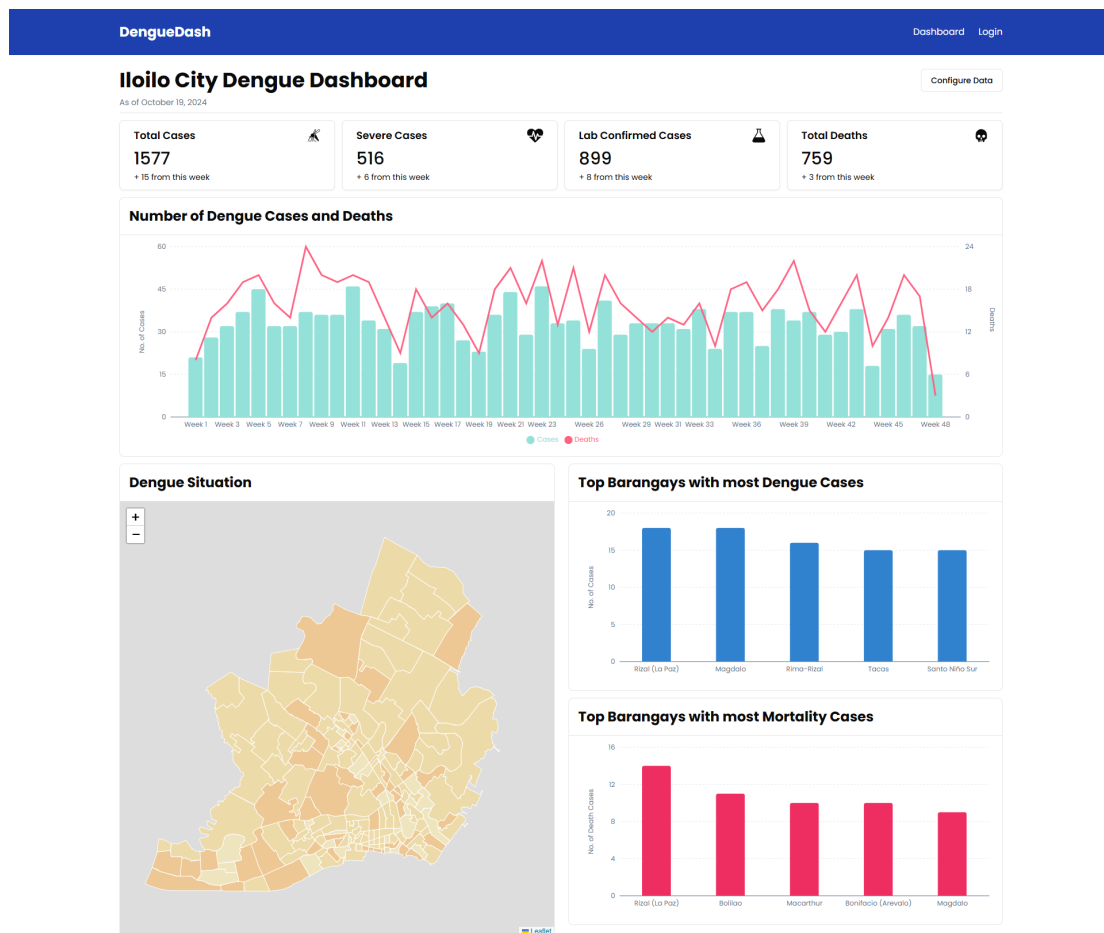


Figure 4.17: Dashboard for Guests

931 4.6.2 Personnel Interface

932 User Authentication, and Login

933 To protect the data's integrity in production, it has been decided that the registration process will not be visible. Instead, an admin must register a user using
934 a different interface. As of the moment, registering a user is done using API via
935 Postman. In the login process, the system implements HTTP-only cookies that
936 contains the JSON Web Tokens (JWT) to protect against XSS attacks. After
937 proper credentials have been provided, it will redirect to the user's home page.
938

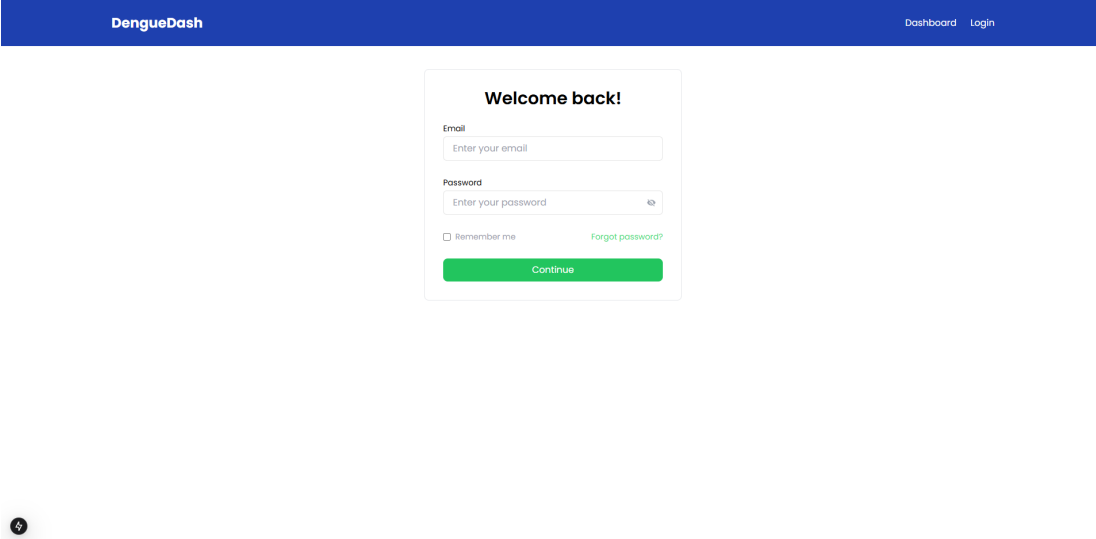


Figure 4.18: Login Page for Users

939 Encoder's View

940 Figures 4.19 and 4.20 show the digitized counterpart of the form obtained from the
941 Iloilo Provincial Epidemiology and Surveillance Unit. As the system aims to support
942 expandability for future features, some fields were modified to accommodate
943 more detailed input. It is worth noting that all of the included fields adhere to the
944 latest Philippine Integrated Disease and Surveillance Response (PIDSR) Dengue
945 Forms, which the referenced form was based on. By doing this, it is assumed
946 that the targeted users will have a familiarity when deployed on a national scale.
947 On a further note, the case form includes the patient's basic information, dengue
948 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 4.19: 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 4.20: Second Part of Case Report Form

949 Once the data generated from the case report form is validated, it will be
 950 assigned as a new case and can be accessed through the Dengue Reports page, as
 951 shown in Figure 4.21. The said page displays basic information about the patient
 952 related to a specific case, including their name, address, date of consultation, and
 953 clinical and case classifications. It is also worth noting that it only shows cases
 954 the user is permitted to view. For example, in a local Disease Reporting Unit
 955 (DRU) setting, the user can only access records that came from the same DRU.
 956 On the other hand, in a consolidated surveillance unit such as a regional and
 957 provincial quarter, its users can view all the records that came from all the DRUs
 958 that report to them. Moving forward, Figure 4.22 shows the detailed case report
 959 of the patient on a particular consultation date.

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

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 4.21: Dengue Reports

Building Your Application > Data Fetching

Personal Information

Full Name

Thomas, Stephanie John

Sex

Male

Full Address

4189 Rice Coves, Calubihan, Iloilo City, Iloilo

Date of Birth

September 19, 2010

Civil Status

Separated

Vaccination Status

First Dose

February 11, 2024

Last Dose

April 10, 2024

Case Record #24010060

Date of Consultation

November 23, 2024

Date Onset of Illness

November 22, 2024

Patient Admitted?

Yes

Clinical Classification

Severe dengue

Laboratory Results

NSI

Pending Result

IgG Elisa

Negative

IgM Elisa

Equivocal

PCR

Pending Result

Date Done

N/A

Date Done

November 29, 2024

Date Done

December 2, 2024

Date Done

N/A

Outcome

Case Classification

Confirmed

Date of Death

November 29, 2024

Outcome

Dead

Figure 4.22: Detailed Case Report

References

- 961 *About GitHub and Git - GitHub Docs.* (n.d.). Retrieved from
 962 [https://docs.github.com/en/get-started/start-your-journey/](https://docs.github.com/en/get-started/start-your-journey/about-github-and-git)
 963 [about-github-and-git](https://docs.github.com/en/get-started/start-your-journey/about-github-and-git)
- 964 Ahmadini, A. A. H., Naeem, M., Aamir, M., Dewan, R., Alshqaq, S. S. A., &
 965 Mashwani, W. K. (2021). Analysis and forecast of the number of deaths,
 966 recovered cases, and confirmed cases from covid-19 for the top four affected
 967 countries using kalman filter. *Frontiers in Physics*, *9*, 629320.
- 968 Arroyo-Marioli, F., Bullano, F., Kucinskas, S., & Rondón-Moreno, C. (2021).
 969 Tracking r of covid-19: A new real-time estimation using the kalman filter.
 970 *PloS one*, *16*(1), e0244474.
- 971 Bosano, R. (2023). *Who: Ph most affected by dengue in western pacific*. Retrieved
 972 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)
 973 [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)
- 974 Brady, O. J., Smith, D. L., Scott, T. W., & Hay, S. I. (2015). Dengue disease
 975 outbreak definitions are implicitly variable. *Epidemics*, *11*, 92–102.
- 976 Bravo, L., Roque, V. G., Brett, J., Dizon, R., & L’Azou, M. (2014). Epidemiology
 977 of dengue disease in the philippines (2000–2011): a systematic literature
 978 review. *PLoS neglected tropical diseases*, *8*(11), e3027.
- 979 Carvajal, T. M., Viacrusis, K. M., Hernandez, L. F. T., Ho, H. T., Amalin, D. M.,
 980 & Watanabe, K. (2018). Machine learning methods reveal the temporal
 981 pattern of dengue incidence using meteorological factors in metropolitan
 982 manila, philippines. *BMC infectious diseases*, *18*, 1–15.
- 983 *Chart.js.* (n.d.). Retrieved from <https://www.chartjs.org/>
- 984 Cheong, Y. L., Burkart, K., Leitão, P. J., & Lakes, T. (2013). Assessing weather
 985 effects on dengue disease in malaysia. *International journal of environmental*
 986 *research and public health*, *10*(12), 6319–6334.
- 987 Christie, T. (n.d.). *Home - Django REST framework.* Retrieved from [https://](https://www.django-rest-framework.org/)
 988 www.django-rest-framework.org/
- 989 Colón-González, F. J., Fezzi, C., Lake, I. R., & Hunter, P. R. (2013). The effects
 990 of weather and climate change on dengue. *PLoS neglected tropical diseases*,
 991 *7*(11), e2503.

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

994 Hii, Y. L., Zhu, H., Ng, N., Ng, L. C., & Rocklöv, J. (2012). Forecast of dengue
 995 incidence using temperature and rainfall. *PLoS neglected tropical diseases*,
 996 6(11), e1908.

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

1000 Leaflet — an open-source JavaScript library for interactive maps. (n.d.). Retrieved
 1001 from <https://leafletjs.com/>

1002 Lena, P. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use
 1003 the date of access, from <https://www.pna.gov.ph/articles/1231208>

1004 Li, X., Feng, S., Hou, N., Li, H., Zhang, S., Jian, Z., & Zi, Q. (2022). Applications
 1005 of kalman filtering in time series prediction. In *International conference on*
 1006 *intelligent robotics and applications* (pp. 520–531).

1007 Ligue, K. D. B., & Ligue, K. J. B. (2022). Deep learning approach to forecasting
 1008 dengue cases in davao city using long short-term memory (lstm). *Philippine*
 1009 *Journal of Science*, 151(3).

1010 Perla. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use the
 1011 date of access, from <https://www.pna.gov.ph/articles/1231208>

1012 RabDashDC. (2024). *Rabdash dc*. Retrieved Use the date of access, from [https://](https://rabdash.com)
 1013 rabdash.com

1014 Runge-Ranzinger, S., Kroeger, A., Oliaro, P., McCall, P. J., Sánchez Tejeda, G.,
 1015 Lloyd, L. S., ... Coelho, G. (2016). Dengue contingency planning: from
 1016 research to policy and practice. *PLoS neglected tropical diseases*, 10(9),
 1017 e0004916.

1018 Shadcn. (n.d.). *Introduction*. Retrieved from <https://ui.shadcn.com/docs>
 1019 *Tailwind CSS - Rapidly build modern websites without ever leaving your HTML*.
 1020 (n.d.). Retrieved from <https://tailwindcss.com/>

1021 Watts, D. M., Burke, D. S., Harrison, B. A., Whitmire, R. E., & Nisalak, A.
 1022 (2020). Effect of temperature on the transmission of dengue virus by aedes
 1023 aegypti. *The American Journal of Tropical Medicine and Hygiene*, 36(1),
 1024 143–152.

1025 *What is Postman? Postman API Platform*. (n.d.). Retrieved from [https://](https://www.postman.com/product/what-is-postman/)
 1026 www.postman.com/product/what-is-postman/

1027 WHO. (2023). *Dengue - global situation*. Retrieved Use the date of ac-
 1028 cess, from [https://www.who.int/emergencies/disease-outbreak-news/](https://www.who.int/emergencies/disease-outbreak-news/item/2023-DON498)
 1029 [item/2023-DON498](https://www.who.int/emergencies/disease-outbreak-news/item/2023-DON498)

1030 WHO. (2024). *Dengue and severe dengue*. Retrieved Use the date
 1031 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)
 1032 [dengue-and-severe-dengue](https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue)

1033 *Why Visual Studio Code?* (2021, 11). Retrieved from <https://code>

1034 `.visualstudio.com/docs/editor/whyvscode`
1035 World Health Organization (WHO). (2018). Dengue and severe dengue in the
1036 philippines. *WHO Dengue Factsheet*. (Available at: `https://www.who`
1037 `.int`)
1038 Zhou, S., & Malani, P. (2024). What is dengue? *Jama*, 332(10), 850–850.
1039 Zod. (n.d.). *TypeScript-first schema validation with static type inference*. Re-
1040 trieved from `https://zod.dev/?id=introduction`

1041 **Appendix A**

1042 **Appendix Title**

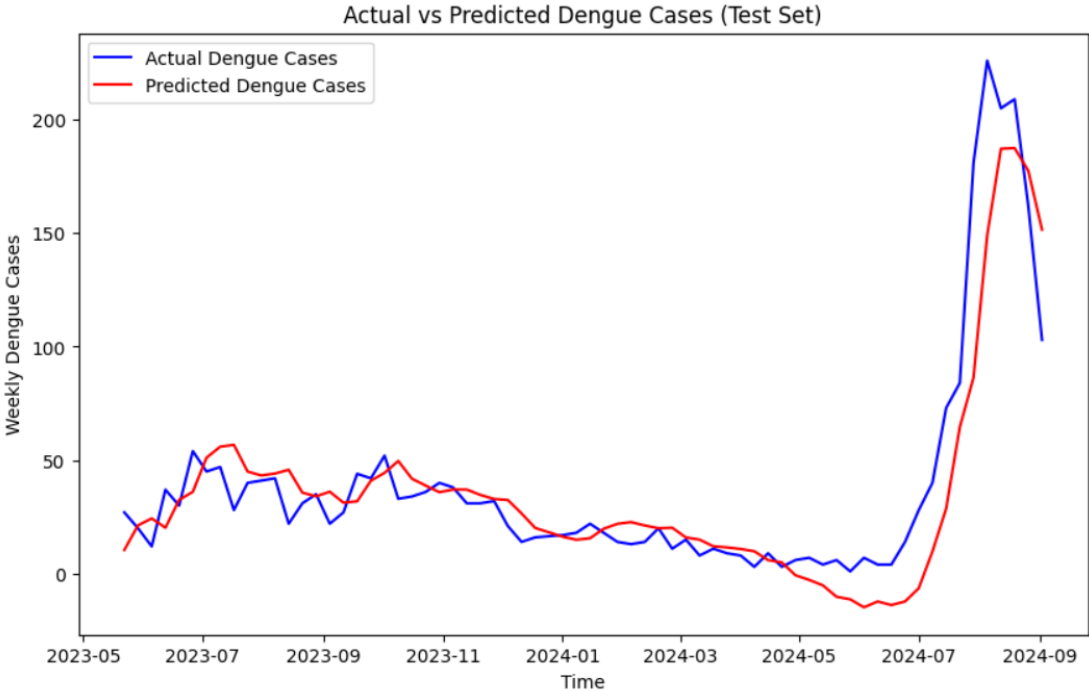


Figure A.1: LSTM Prediction Results for Test Set

1043 **Appendix B**

1044 **Resource Persons**

1045 **Mr. Firstname1 Lastname1**

1046 Role1

1047 Affiliation1

1048 emailaddr1@domain.com

1049 **Ms. Firstname2 Lastname2**

1050 Role2

1051 Affiliation2

1052 emailaddr2@domain.net

1053