

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 22, 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 is 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 focuses 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 explores 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	3.1.3	Integrate the Predictive Model into a Web-Based Data An-	
71		alytics Dashboard	16
72	3.1.4	System Development Framework	16
73	3.1.5	Design, Building, Testing, and Integration	17
74	3.2	Development Tools	18
75	3.2.1	Software	18
76	3.2.2	Hardware	19
77	3.2.3	Packages	19
78	3.3	Calendar of Activities	21
79	4	Preliminary Results/System Prototype	22
80	4.1	Data Gathering	22
81	4.2	Exploratory Data Analysis	23
82	4.3	Outbreak Detection	27
83	4.4	Model Training	28
84	4.4.1	LSTM Model	29
85	4.4.2	ARIMA Model	31
86	4.4.3	Kalman Filter Model	35
87	4.5	Preliminary System Requirements	37

88	4.5.1	Backend Requirements	37
89	4.5.2	User Interface Requirements	38
90	4.5.3	Security and Validation Requirements	40
91	4.5.4	Testing Process	41
92	4.6	System Prototype	42
93	4.6.1	Guest Interface	42
94	4.6.2	Personnel Interface	43
95		References	47
96		A Appendix Title	50
97		B Resource Persons	51

98 List of Figures

99	3.1 Workflow for forecasting the number of weekly dengue cases . . .	10
100	4.1 Snippet of the Combined Dataset	23
101	4.2 Data Contents	23
102	4.3 Dataset Statistics	24
103	4.4 Distribution of Dengue Cases	24
104	4.5 Frequency of Dengue Cases in Different Intervals	25
105	4.6 Trend of Dengue Cases	26
106	4.7 Ranking of Correlations	26
107	4.8 Ranking of Correlations (with lagged effects)	27
108	4.9 Comparison of Window Sizes	30
109	4.10 ARIMA Prediction Results for Test Set	31
110	4.11 Seasonal ARIMA Prediction Results for Test Set	33
111	4.12 Kalman Filter Prediction Results for Test Set	35
112	4.13 Entity-Relationship Database Schema Hybrid Diagram for DengueDash	
113	Database Structure	37
114	4.14 Use Case Diagram for Admin	38
115	4.15 Use Case Diagram for Encoder	39

116	4.16 Testing Process for DengueWatch	41
117	4.17 Dashboard for Guests	42
118	4.18 Login Page for Users	43
119	4.19 First Part of Case Report Form	44
120	4.20 Second Part of Case Report Form	44
121	4.21 Dengue Reports	45
122	4.22 Detailed Case Report	46
123	A.1 LSTM Prediction Results for Test Set	50

124 **List of Tables**

125	3.1	Timetable of Activities for 2024	21
126	3.2	Timetable of Activities for 2025	21
127	4.1	Comparison of Models	28
128	4.2	Comparison of Window Sizes	30

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.

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

160 1.2 Problem Statement

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

173 1.3 Research Objectives

174 1.3.1 General Objective

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

179 1.3.2 Specific Objectives

180 Specifically, this study aims to:

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

195 1.4 Scope and Limitations of the Research

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

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

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

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

221 1.5 Significance of the Research

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

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

245 awareness campaigns and community engagement initiatives, empowering
246 residents with knowledge and preventative measures to protect themselves
247 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

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

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

284 **2.3 Existing System: RabDash DC**

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

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

301 2.4 Deep Learning

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

315 2.5 Kalman Filter

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

331 Our study will compare ARIMA, seasonal ARIMA, Kalman Filter, and LSTM
332 models using our own collected dengue case data along with weather data to
333 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.

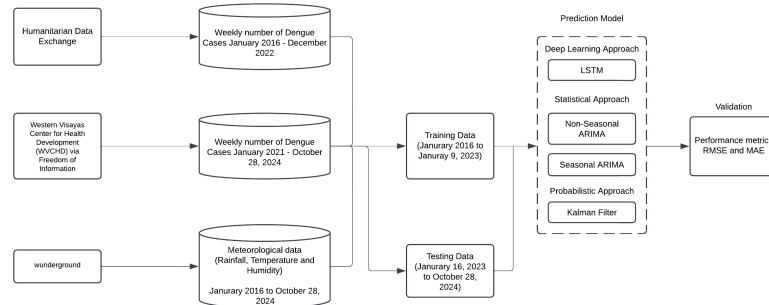


Figure 3.1: Workflow for forecasting the number of weekly dengue cases

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.

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

416 **Data Integration and Preprocessing**

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

424 **Exploratory Data Analysis (EDA)**

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

429 **Outbreak Detection**

430 To detect outbreaks, we computed the outbreak threshold value of dengue cases
431 using the formula,

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

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

433 3.1.2 Develop and Evaluate Deep Learning Models for 434 Dengue Case Forecasting

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

440 Data Preprocessing

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

447 LSTM Model

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

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

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

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

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

465 The training process was conducted using three distinct window sizes (5 weeks,
466 10 weeks, and 20 weeks) to determine the optimal sequence length of weeks to
467 input into the LSTM model for improved forecasting performance.

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

472 **Seasonal ARIMA (SARIMA):**

473 The SARIMA (Seasonal ARIMA) model was utilized to forecast weekly dengue
474 cases, incorporating seasonal patterns and exogenous weather variables (rainfall,
475 max temperature, and humidity). The dataset was divided into training (80%)
476 and testing (20%) sets while maintaining temporal continuity for validation. The
477 input data consisted of weekly dengue case counts as the target variable and
478 weather-related features as exogenous regressors.

479 The SARIMA model's parameters were set as follows:

- 480 • Order: (2, 0, 2)
- 481 • Seasonal Order: (0, 1, 1, 52)

482 The SARIMA model was trained using the training dataset, including exoge-
483 nous variables. The maximum number of iterations was set to 400 to ensure
484 convergence during fitting.

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

- 487 • Mean Squared Error (MSE): Quantifies average squared prediction error.
- 488 • Root Mean Squared Error (RMSE): Measures average prediction error on
489 the data's original scale.
- 490 • Mean Absolute Error (MAE): Measures the average magnitude of the abso-
491 lute errors between the predicted and actual values

492 ARIMA

493 The ARIMA model was employed to forecast weekly dengue cases using historical
494 weather data (rainfall, max temperature, and humidity) as exogenous variables
495 and historical case counts as the primary dependent variable. The dataset was
496 split into training (80%) and testing (20%) sets. To determine the optimal con-
497 figuration for the ARIMA model, a grid search was conducted over the following
498 parameter ranges:

- 499 • p (autoregressive order): 0 to 3
- 500 • d (differencing order): 0 to 2
- 501 • q (moving average order): 0 to 3

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

507 The fitted ARIMA model was used to forecast weekly dengue cases for the
508 test dataset. Predictions were directly assigned to the PredictedCases column in
509 the test dataset. Model performance was evaluated using the following metrics:

- 510 • Mean Squared Error (MSE): Quantifies average squared prediction error.
- 511 • Root Mean Squared Error (RMSE): Measures average prediction error on
512 the data's original scale.
- 513 • Mean Absolute Error (MAE): Measures the average magnitude of the abso-
514 lute errors between the predicted and actual values

515 Kalman Filter:

- 516 • Input Variables: The target variable (Cases) was modeled using three re-
517 gressors: 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 **Model Evaluation and Optimization**

- 529 • Compare the performance of all models to identify the most accurate fore-
530 casting approach.
- 531 • Iteratively optimize the selected model.

532 **3.1.3 Integrate the Predictive Model into a Web-Based** 533 **Data Analytics Dashboard**

534 **Dashboard Design and Development**

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

539 **Model Integration and Deployment**

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

542 **3.1.4 System Development Framework**

543 The Agile Model is the birthchild of both iterative and incremental approaches
544 in Software Engineering. It aims to be flexible and effective at the same time by

545 being adaptable to change. It's also important to note that small teams looking
546 to construct and develop projects quickly can benefit from this kind of method-
547 ology. As the Agile Method focuses on continuous testing, quality assurance is a
548 guarantee since bugs and errors are quickly identified and patched.

549 **3.1.5 Design, Building, Testing, and Integration**

550 **Design and Development**

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

564 **Testing and Integration**

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

571 **3.2 Development Tools**

572 **3.2.1 Software**

573 **Github**

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

579 **Visual Studio Code**

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

584 **Django**

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

593 **Next.js**

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

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

600 **Postman**

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

606 **3.2.2 Hardware**

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

610 **3.2.3 Packages**

611 **Django REST Framework**

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

616 **Leaflet**

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

623 **Chart.js**

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

631 **Tailwind CSS**

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

640 **Shadcn**

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

646 **Zod**

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

3.3 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/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.

4.2 Exploratory Data Analysis

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

```
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

- **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

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%.

	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

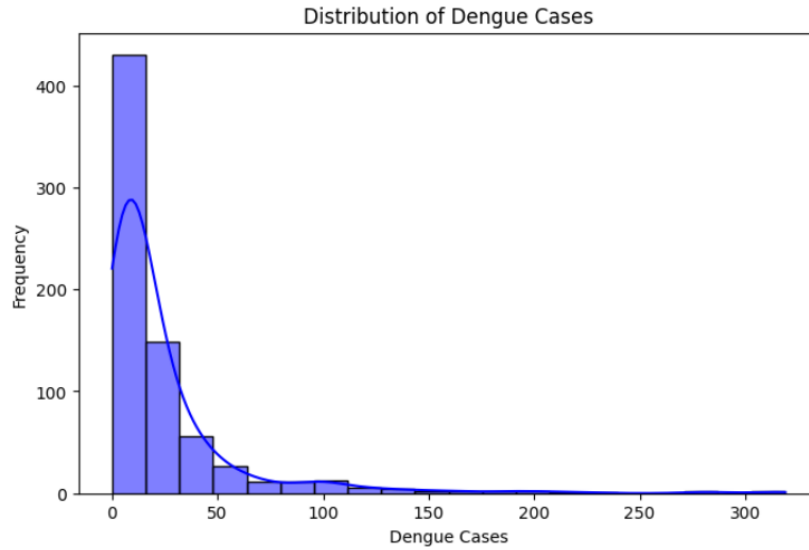


Figure 4.4: Distribution of Dengue Cases

688 In figure 4.4, a histogram of dengue cases shows a right-skewed distribution,
689 indicating that most weeks experience low case counts, while a few weeks record
690 outbreaks.

691 To further analyze the distribution, dengue cases were categorized into different
692 intervals (Figure 4.5): 0-5 cases, 6-15 cases, 16-30 cases, 31-100 cases and 101+
693 cases. The majority of weeks falls within the 0-5 cases and 6-15 cases categories,
694 indicating that most weeks have low dengue cases. Meanwhile, weeks with 101+
695 cases are rare, suggesting that extreme outbreaks are not frequent.

696 Figure 4.6 illustrates the trend of weekly dengue cases over time. The data
697 reveals periodic spikes in the number of cases, suggesting a seasonal pattern in
698 dengue cases. Notably, peak cases are observed during certain periods approx-
699 imately 3 years, potentially aligning with specific climatic conditions such as

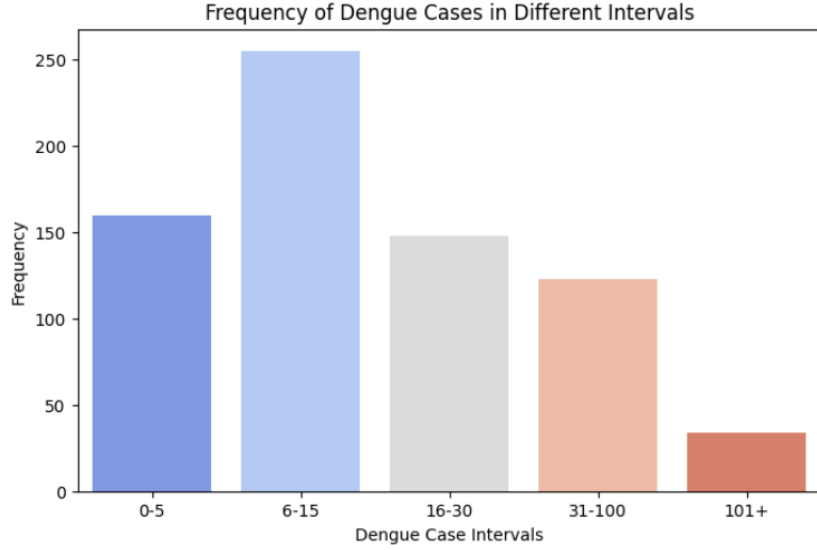


Figure 4.5: Frequency of Dengue Cases in Different Intervals

700 increased rainfall or temperature changes. This underscores the importance of
 701 incorporating climate variables into the forecasting model.

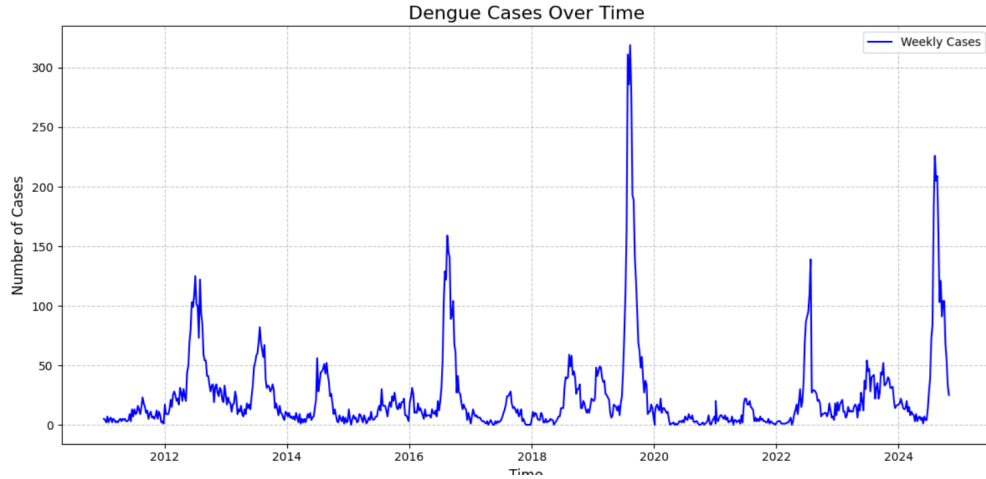


Figure 4.6: Trend of Dengue Cases

702 Figure 4.7 shows the ranking of correlation coefficients between dengue cases
 703 and selected features, including rainfall, humidity, maximum temperature, aver-
 704 age temperature, minimum temperature, and wind speed. Among these, rainfall
 705 exhibits the highest positive correlation with dengue cases (correlation coefficient
 706 0.13), indicating that increased rainfall may contribute to higher cases counts.
 707 This aligns with existing studies suggesting that stagnant water from heavy rain-
 708 fall creates breeding grounds for mosquitos. It is followed by humidity (0.10),

709 suggesting that higher humidity levels may enhance mosquito reproduction, lead-
 710 ing to more dengue cases. Temperature has a weak to moderate positive corre-
 711 lation with dengue cases, with maximum temperature (0.09) showing a stronger
 712 relationship than average and minimum temperature.

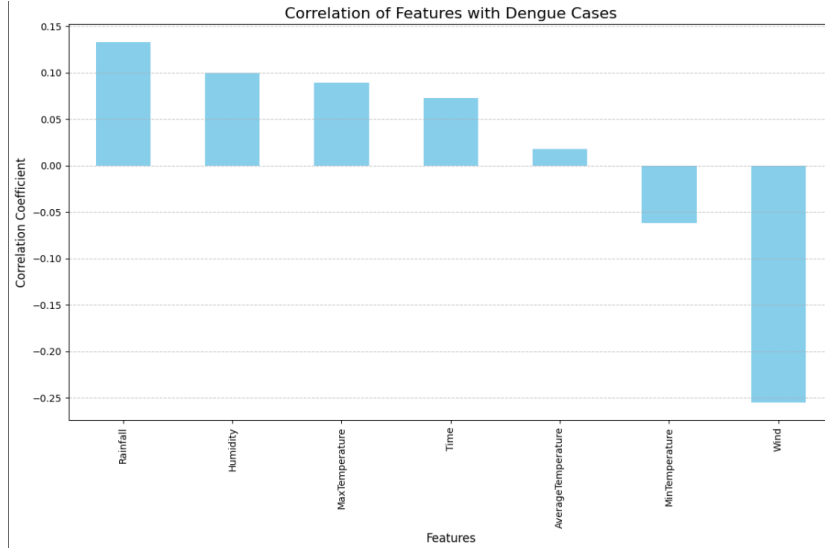


Figure 4.7: Ranking of Correlations

713 Figure 4.8 shows the ranking of correlation coefficients between dengue cases
 714 and selected features, with the addition of lagged effects. The analysis reveals no
 715 improvement in correlation when lagged variables are compared to direct observa-
 716 tions. This suggests that the observed values of rainfall, humidity, and maximum
 717 temperature remain the most significant predictors for dengue case forecasting.
 718 Overall, the exploratory data analysis highlights the significance of rainfall, hu-
 719 midity, and max temperature variables in dengue case forecasting.

720 4.3 Outbreak Detection

721 To identify outbreaks, we calculated the outbreak threshold value using the histor-
 722 ical mean as the endemic channel. The threshold is determined using the formula:

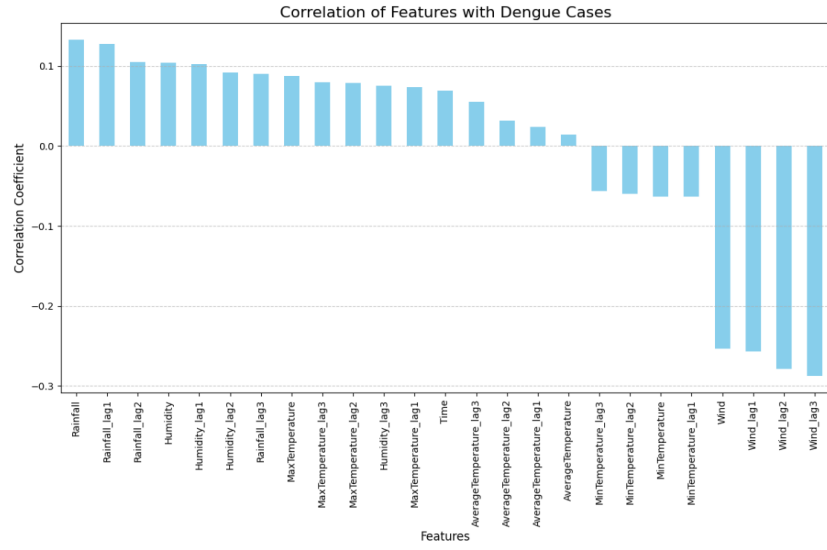


Figure 4.8: Ranking of Correlations (with lagged effects)

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

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

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

$$= 98.03407 \quad (4.4)$$

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

734 historical dengue cases from 2011 to 2024, with meteorological variables such as
735 max temperature, humidity, and rainfall.

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

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

749 The table below provides a summary and comparative analysis of each model's
750 results across these metrics, offering insights into the strengths and limitations of
751 each forecasting technique for dengue case prediction in Iloilo City. The acceptable
752 threshold for Mean Absolute Error (MAE) in forecasting dengue cases for it to be
753 considered accurate can vary depending on the context. However, related studies
754 often serve as benchmarks, with commonly cited acceptable values ranging from
755 20 to 30. For this study, we have established a threshold of 15 to emphasize the
756 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

757 4.4.1 LSTM Model

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

- 761 • Window Size: 5, 10, and 20 weeks, representing the time steps used in the
762 sequence data for each prediction.
- 763 • Epochs: 100 epochs were used for training, balancing sufficient training
764 time with computational efficiency also implementing early stopping to avoid
765 overfitting.
- 766 • Batch Size: 1, allowing the model to process one sequence at a time, which
767 is beneficial for small datasets but increases training time.
- 768 • Optimizer: The Adam optimizer was chosen for its adaptive learning capa-
769 bilities and stability in training. A custom learning rate of 0.0001 was set
770 to ensure gradual convergence and minimize risk of overfitting.

771 The dataset was split into training and test sets to evaluate the model’s per-
772 formance and generalizability:

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

778 The training process was conducted using three distinct window sizes—5 weeks,
779 10 weeks, and 20 weeks—to identify the optimal sequence length of weeks for input
780 into the LSTM model, thereby enhancing forecasting performance. The following
781 plots illustrate the performance of the model in predicting dengue cases for each
782 of the specified window sizes.

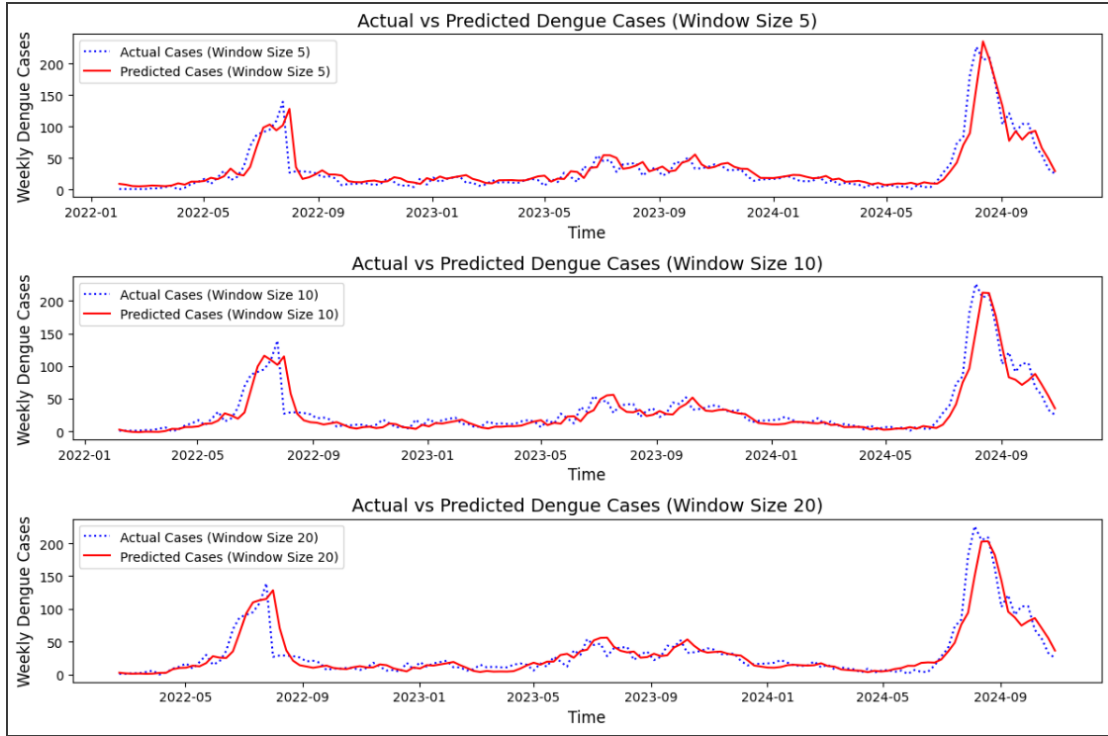


Figure 4.9: Comparison of Window Sizes

783 The evaluation metrics included Mean Squared Error (MSE), Root Mean
784 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

785

786 The results indicate that a window size of 10 weeks provides the most accurate
787 predictions, as evidenced by the lowest MSE and RMSE values. Although the
788 other two window sizes exhibit higher MAE values, the difference is not that
789 noticeable, with only approximately 0.1 differences. This suggests that using a
790 10-week sequence length effectively balances the temporal dependencies captured
791 by the model and the computational complexity of training.

792 Training and Testing Data Division for ARIMA 793 and Seasonal Arima

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

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

799 4.4.2 ARIMA Model

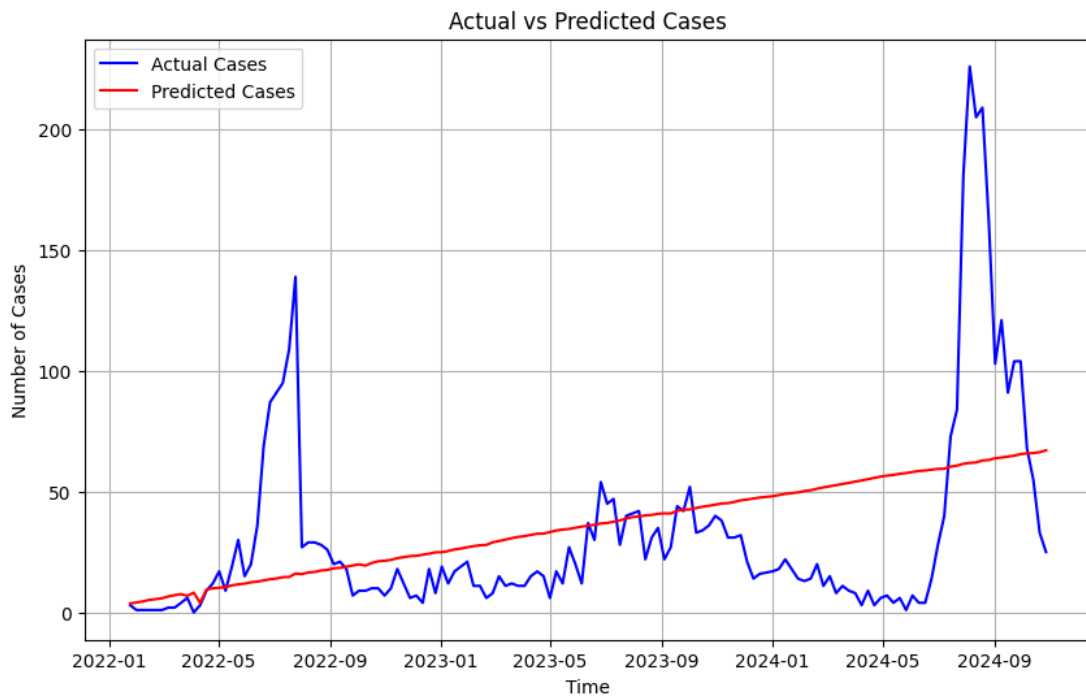


Figure 4.10: ARIMA Prediction Results for Test Set

800 The ARIMA model was developed to capture non-seasonal trends in the data. To
801 determine the best model configuration, grid search was used to explore various
802 combinations of ARIMA parameters, ultimately selecting **ARIMA(1, 2, 2)**. The
803 model was iteratively refined over **400 iterations** to ensure convergence to an
804 optimal solution. Key details are as follows:

- 805 1. **Data Preprocessing:** Prepare the dataset by handling any missing values
806 and scaling the data if necessary to improve model convergence and stability.
- 807 2. **Hyperparameter Tuning:** Use a grid search on potential ARIMA param-
808 eters (p, d, q) to identify the configuration that minimizes error. The optimal
809 parameters were found to be **(1, 2, 2)**.
- 810 3. **Model Training:**
- 811 • Set the number of iterations to 400 to ensure thorough training and
812 convergence.
 - 813 • Train the ARIMA model on 80% of the data and reserve 20% for test-
814 ing.
- 815 4. **Evaluation:** After training, the ARIMA model was evaluated on the test
816 data, yielding the following performance metrics:
- 817 • **MSE (Mean Squared Error):** 1521.48
 - 818 • **RMSE (Root Mean Squared Error):** 39.01
 - 819 • **MAE (Mean Absolute Error):** 25.80

820 Seasonal ARIMA (SARIMA) Model

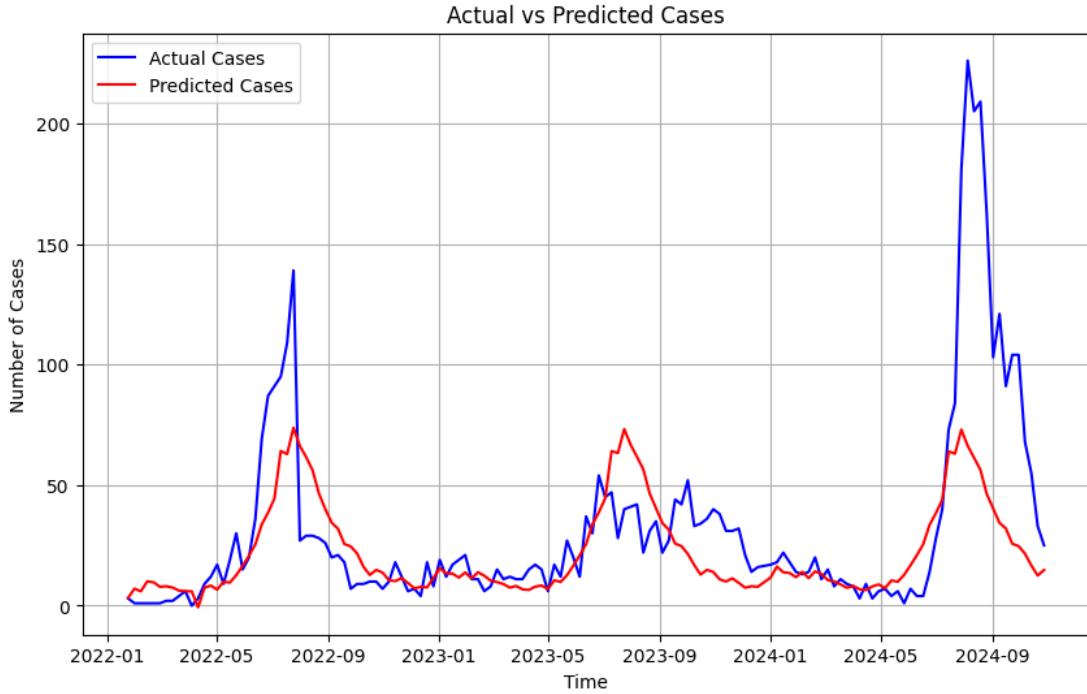


Figure 4.11: Seasonal ARIMA Prediction Results for Test Set

821 This model incorporates seasonal parameters, which were tuned using grid search
822 to find the best configuration: **SARIMA(2, 0, 2)(0, 1, 1)[52]**. As with ARIMA,
823 **400 iterations** were applied to ensure a robust fit.

824 Steps to Create the SARIMA Model:

- 825 1. **Data Preprocessing:** Ensure data readiness by filling any missing values
826 and scaling as needed.
- 827 2. **Seasonality Analysis:** Examine the dataset for seasonal patterns. A pe-
828 riodicity of **52 weeks** was identified, making SARIMA a suitable choice for
829 capturing yearly seasonality.
- 830 3. **Hyperparameter Tuning:** Conduct grid search to identify the best set of
831 parameters $(p, d, q)(P, D, Q)[S]$, where:
 - 832 • **(p, d, q)** are the non-seasonal parameters,

- 833 • **(P, D, Q)** are the seasonal parameters, and
834 • **S** is the season length.

835 The optimal configuration found was **(2, 0, 2)(0, 1, 1)**[52].

836 4. **Model Training:**

- 837 • Set the iteration count to 400 for enhanced model robustness.
838 • Train the model on the 80% training dataset and reserve the remaining
839 20% for testing.

840 5. **Evaluation:** The SARIMA model yielded the following error metrics:

- 841 • **MSE:** 1109.69
842 • **RMSE:** 33.31
843 • **MAE:** 18.09

844 The SARIMA model outperformed the ARIMA model in terms of lower
845 MSE and RMSE values, indicating its effectiveness in capturing the seasonal
846 patterns in the data.

847 4.4.3 Kalman Filter Model

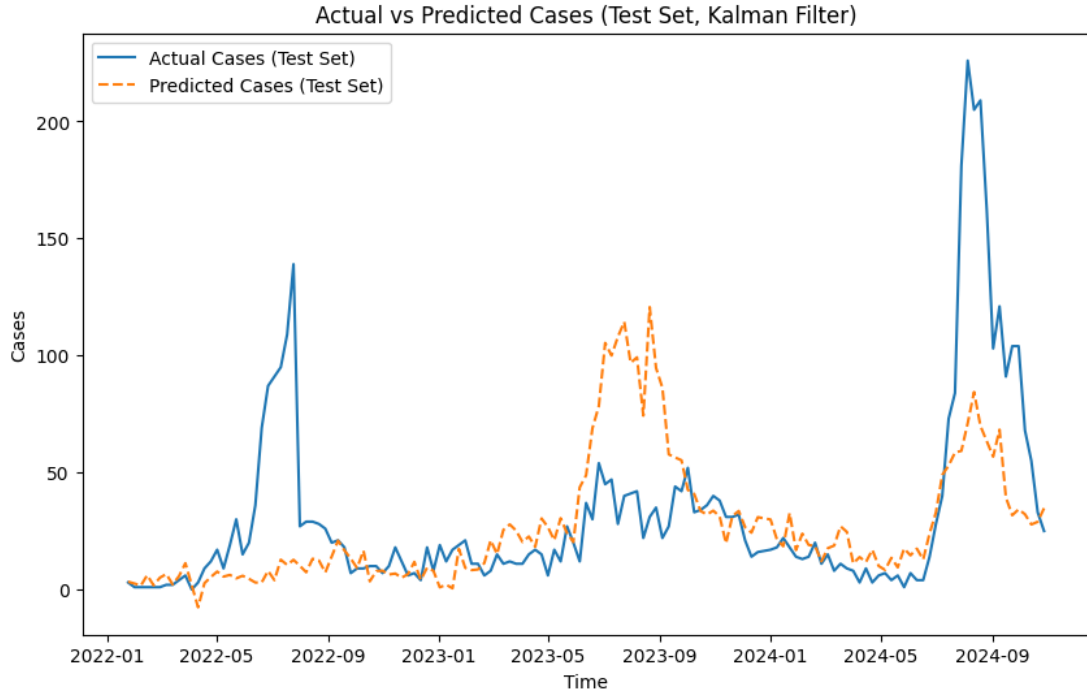


Figure 4.12: Kalman Filter Prediction Results for Test Set

848 Kalman Filter Methodology with Matrix Calculu- 849 lations

850 **Measurement Acquisition:** Obtain the measurement z_k of the system's state
851 with associated confidence. This measurement matrix provides a noisy observation
852 of the true state.

853 The dataset was split into training and test sets to evaluate the Kalman Filter's
854 performance and generalizability:

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

858 **Prediction Step:**

- 859 • Predict the next state:

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

860 where A is the state transition matrix and B is the control matrix.

- 861 • Update the state covariance:

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

862 where Q is the process noise covariance matrix.

863 **Compute Residual:** Calculate the residual

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

864 where H is the observation matrix. This residual represents the new information
865 from the measurement.

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

- 867 • Compute the Kalman Gain:

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

868 where R is the measurement noise covariance matrix.

- 869 • The Kalman Gain determines the weight of the measurement relative to the
870 prediction.

871 **State Update:**

- 872 • Update the state estimate:

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

873 blending the prediction and measurement.

874 **Uncertainty Update:**

- 875 • Update the state covariance:

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

876 where I is the identity matrix.

877 **Model Evaluation:** Upon testing, the Kalman Filter produced a Mean
878 Squared Error (MSE) of 1474.82, Root Mean Squared Error (RMSE) of 38.40
879 and Mean Absolute Error(MAE) of 22.34.

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.

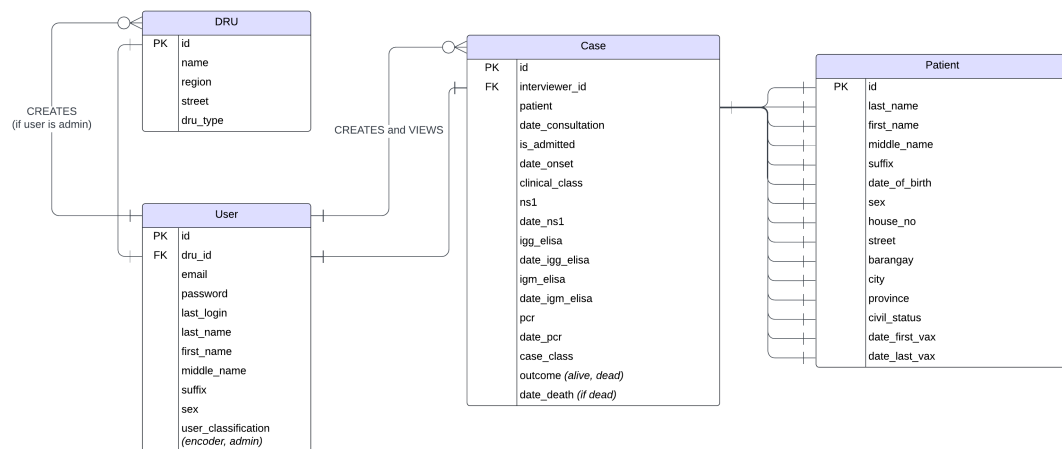


Figure 4.13: Entity-Relationship Database Schema Hybrid Diagram for DengueDash Database Structure

889 4.5.2 User Interface Requirements

890 Admin Interface

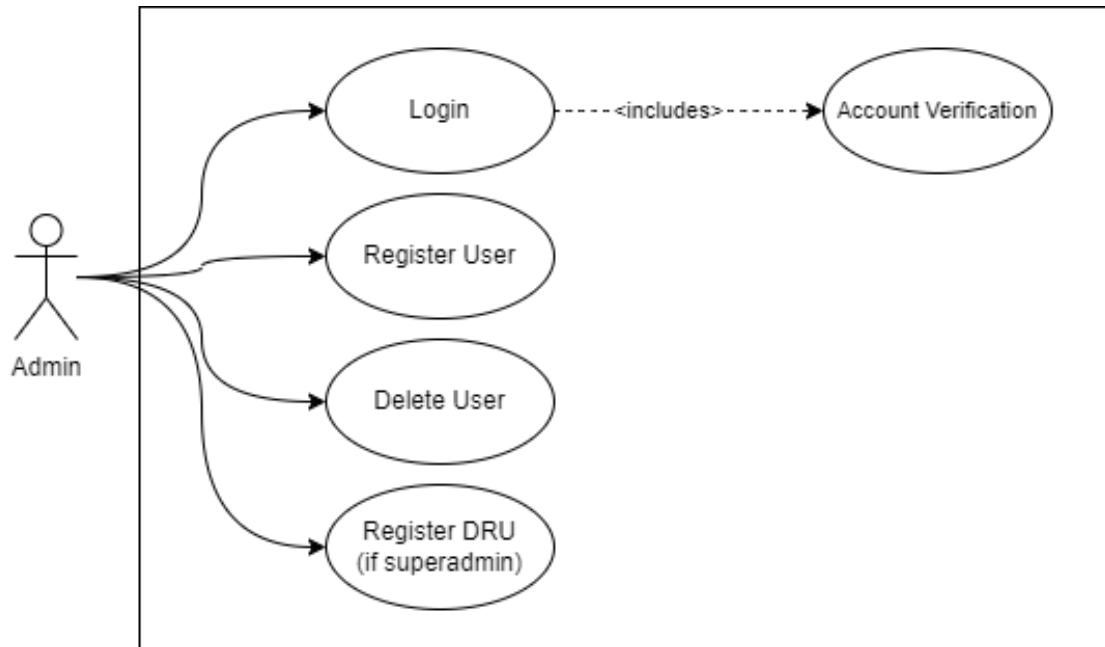


Figure 4.14: Use Case Diagram for Admin

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

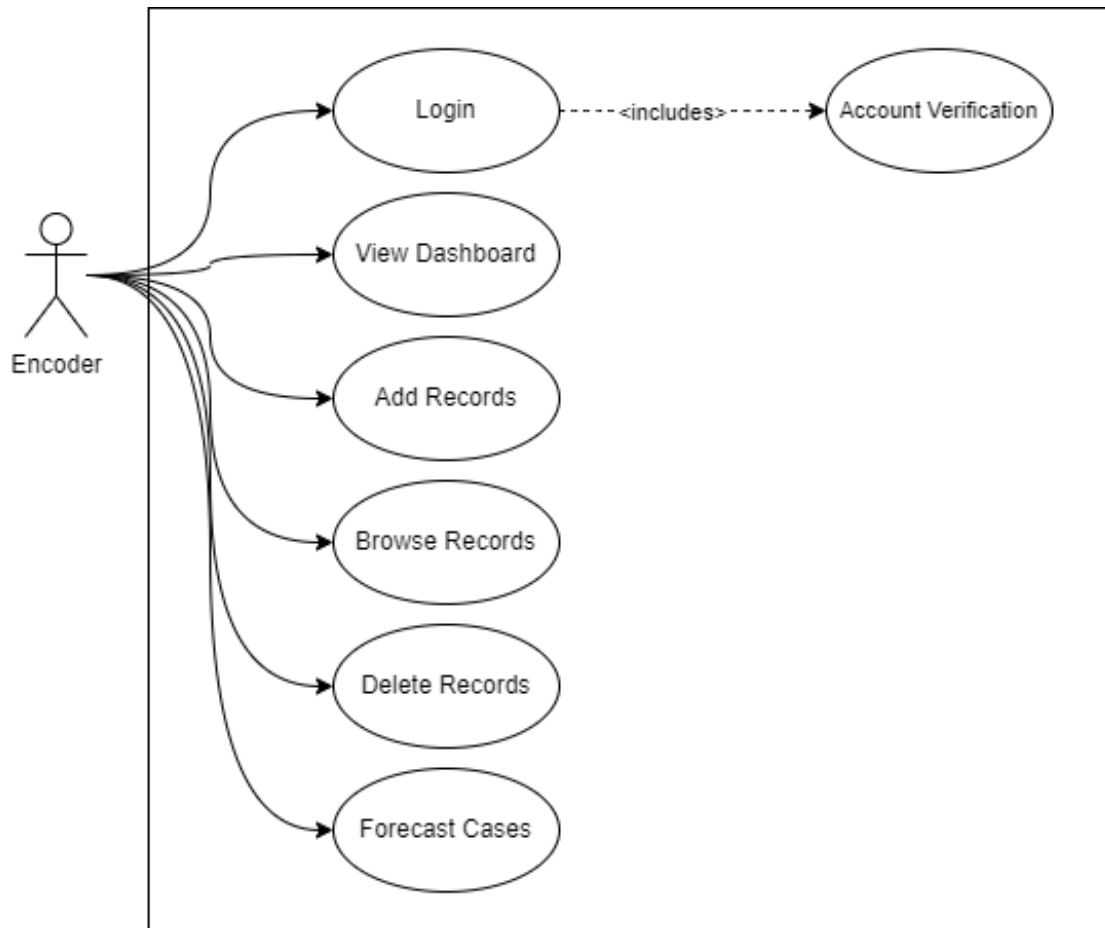


Figure 4.15: Use Case Diagram for Encoder

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

902 4.5.3 Security and Validation Requirements

903 Password Encryption

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

912 Authentication

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

918 Data Validation

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

927 4.5.4 Testing Process

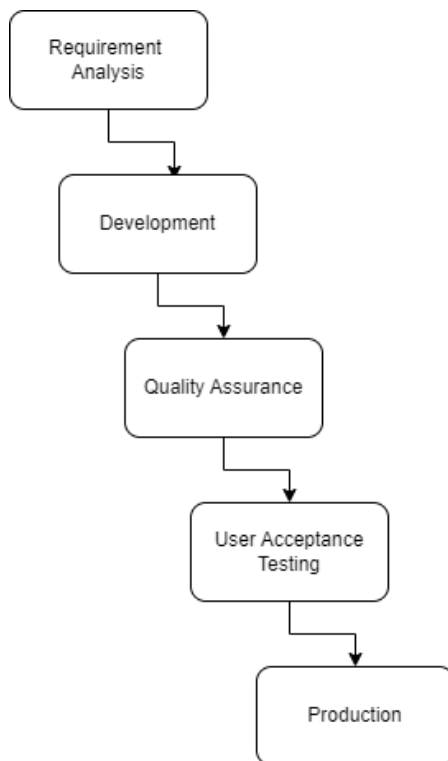


Figure 4.16: Testing Process for DengueWatch

928 As the system requirements and functionalities have been mentioned above, it
929 is important to implement testing to validate the system's performance and effi-
930 cacy. Since dengue reports include confidential information, anonymized historical
931 dengue reports were used to train the model and create the foundational architec-
932 ture of the system. By using functional tests, data validation and visualization can
933 be ensured for further continual improvements. Security testing is also important
934 as it is needed to safeguard confidential information when the system is deployed.
935 It includes proper authentication, permission views, and mitigating common in-
936 jection attacks. Finally, a user acceptance test from the prospected users, in this
937 case, the Iloilo City Epidemiology and Surveillance Unit, is crucial to assess its
938 performance and user experience. It enables the developers to confirm if the sys-
939 tem meets the needs of the problem, and once confirmed, it will be deployed and
940 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.

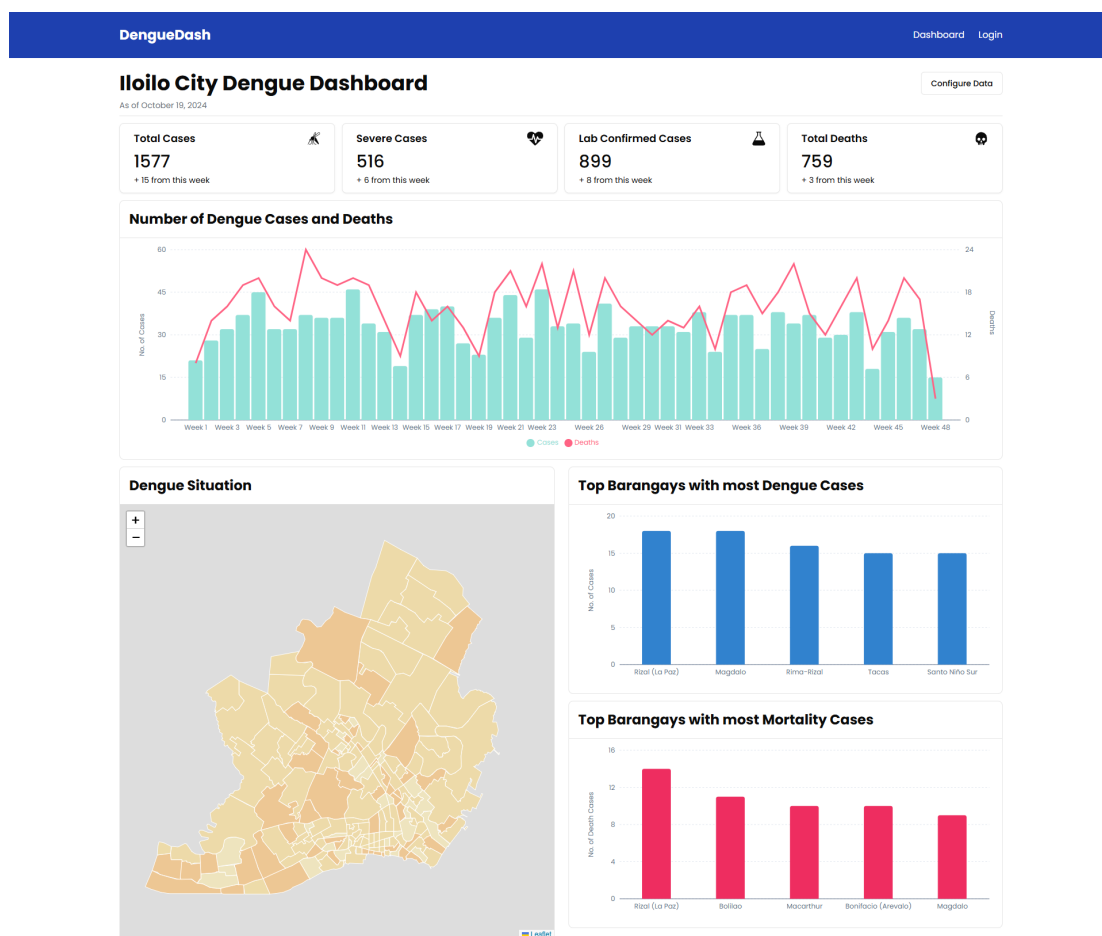


Figure 4.17: Dashboard for Guests

947 4.6.2 Personnel Interface

948 User Authentication, and Login

949 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
950 a different interface. As of the moment, registering a user is done using API via
951 Postman. In the login process, the system implements HTTP-only cookies that
952 contains the JSON Web Tokens (JWT) to protect against XSS attacks. After
953 proper credentials have been provided, it will redirect to the user's home page.
954

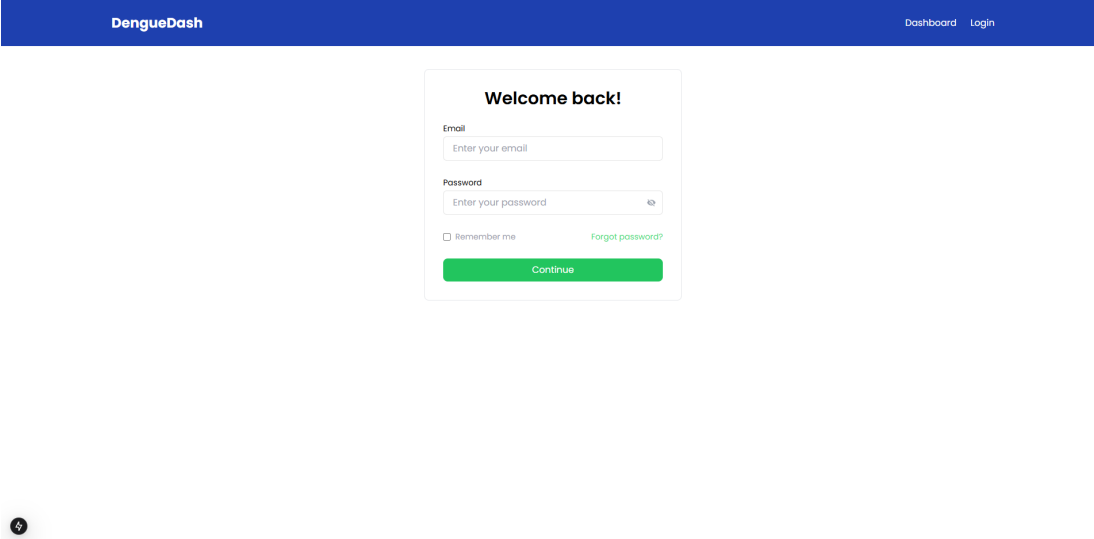


Figure 4.18: Login Page for Users

955 Encoder's View

956 Figures 4.19 and 4.20 show the digitized counterpart of the form obtained from the
957 Iloilo Provincial Epidemiology and Surveillance Unit. As the system aims to support
958 expandability for future features, some fields were modified to accommodate
959 more detailed input. It is worth noting that all of the included fields adhere to the
960 latest Philippine Integrated Disease and Surveillance Response (PIDSR) Dengue
961 Forms, which the referenced form was based on. By doing this, it is assumed
962 that the targeted users will have a familiarity when deployed on a national scale.
963 On a further note, the case form includes the patient's basic information, dengue
964 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

965 Once the data generated from the case report form is validated, it will be
 966 assigned as a new case and can be accessed through the Dengue Reports page, as
 967 shown in Figure 4.21. The said page displays basic information about the patient
 968 related to a specific case, including their name, address, date of consultation, and
 969 clinical and case classifications. It is also worth noting that it only shows cases
 970 the user is permitted to view. For example, in a local Disease Reporting Unit
 971 (DRU) setting, the user can only access records that came from the same DRU.
 972 On the other hand, in a consolidated surveillance unit such as a regional and
 973 provincial quarter, its users can view all the records that came from all the DRUs
 974 that report to them. Moving forward, Figure 4.22 shows the detailed case report
 975 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

976 References

- 977 *About GitHub and Git - GitHub Docs.* (n.d.). Retrieved from
 978 [https://docs.github.com/en/get-started/start-your-journey/
 979 about-github-and-git](https://docs.github.com/en/get-started/start-your-journey/about-github-and-git)
- 980 Ahmadini, A. A. H., Naeem, M., Aamir, M., Dewan, R., Alshqaq, S. S. A., &
 981 Mashwani, W. K. (2021). Analysis and forecast of the number of deaths,
 982 recovered cases, and confirmed cases from covid-19 for the top four affected
 983 countries using kalman filter. *Frontiers in Physics*, *9*, 629320.
- 984 Arroyo-Marioli, F., Bullano, F., Kucinskas, S., & Rondón-Moreno, C. (2021).
 985 Tracking r of covid-19: A new real-time estimation using the kalman filter.
 986 *PloS one*, *16*(1), e0244474.
- 987 Bosano, R. (2023). *Who: Ph most affected by dengue in western pacific*. Retrieved
 988 Use the date of access, from [https://news.abs-cbn.com/spotlight/12/
 989 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)
- 990 Brady, O. J., Smith, D. L., Scott, T. W., & Hay, S. I. (2015). Dengue disease
 991 outbreak definitions are implicitly variable. *Epidemics*, *11*, 92–102.
- 992 Bravo, L., Roque, V. G., Brett, J., Dizon, R., & L’Azou, M. (2014). Epidemiology
 993 of dengue disease in the philippines (2000–2011): a systematic literature
 994 review. *PLoS neglected tropical diseases*, *8*(11), e3027.
- 995 Carvajal, T. M., Viacrusis, K. M., Hernandez, L. F. T., Ho, H. T., Amalin, D. M.,
 996 & Watanabe, K. (2018). Machine learning methods reveal the temporal
 997 pattern of dengue incidence using meteorological factors in metropolitan
 998 manila, philippines. *BMC infectious diseases*, *18*, 1–15.
- 999 *Chart.js.* (n.d.). Retrieved from <https://www.chartjs.org/>
- 1000 Cheong, Y. L., Burkart, K., Leitão, P. J., & Lakes, T. (2013). Assessing weather
 1001 effects on dengue disease in malaysia. *International journal of environmental
 1002 research and public health*, *10*(12), 6319–6334.
- 1003 Christie, T. (n.d.). *Home - Django REST framework*. Retrieved from [https://
 1004 www.django-rest-framework.org/](https://www.django-rest-framework.org/)
- 1005 Colón-González, F. J., Fezzi, C., Lake, I. R., & Hunter, P. R. (2013). The effects
 1006 of weather and climate change on dengue. *PLoS neglected tropical diseases*,
 1007 *7*(11), e2503.

1008 Hemisphere, N. (2015). Update on the dengue situation in the western pacific
1009 region. *Update*.

1010 Hii, Y. L., Zhu, H., Ng, N., Ng, L. C., & Rocklöv, J. (2012). Forecast of dengue
1011 incidence using temperature and rainfall. *PLoS neglected tropical diseases*,
1012 6(11), e1908.

1013 Joel, C. (2021, 10). *6 reasons to use Tailwind over traditional CSS*. Re-
1014 trieved from [https://dev.to/charliejoel/6-reasons-to-use-tailwind](https://dev.to/charliejoel/6-reasons-to-use-tailwind-over-traditional-css-1nc3)
1015 [-over-traditional-css-1nc3](https://dev.to/charliejoel/6-reasons-to-use-tailwind-over-traditional-css-1nc3)

1016 Leaflet — an open-source JavaScript library for interactive maps. (n.d.). Retrieved
1017 from <https://leafletjs.com/>

1018 Lena, P. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use
1019 the date of access, from <https://www.pna.gov.ph/articles/1231208>

1020 Li, X., Feng, S., Hou, N., Li, H., Zhang, S., Jian, Z., & Zi, Q. (2022). Applications
1021 of kalman filtering in time series prediction. In *International conference on*
1022 *intelligent robotics and applications* (pp. 520–531).

1023 Ligue, K. D. B., & Ligue, K. J. B. (2022). Deep learning approach to forecasting
1024 dengue cases in davao city using long short-term memory (lstm). *Philippine*
1025 *Journal of Science*, 151(3).

1026 Perla. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use the
1027 date of access, from <https://www.pna.gov.ph/articles/1231208>

1028 RabDashDC. (2024). *Rabdash dc*. Retrieved Use the date of access, from [https://](https://rabdash.com)
1029 rabdash.com

1030 Runge-Ranzinger, S., Kroeger, A., Oliaro, P., McCall, P. J., Sánchez Tejeda, G.,
1031 Lloyd, L. S., ... Coelho, G. (2016). Dengue contingency planning: from
1032 research to policy and practice. *PLoS neglected tropical diseases*, 10(9),
1033 e0004916.

1034 Shadcn. (n.d.). *Introduction*. Retrieved from <https://ui.shadcn.com/docs>
1035 *Tailwind CSS - Rapidly build modern websites without ever leaving your HTML*.
1036 (n.d.). Retrieved from <https://tailwindcss.com/>

1037 Watts, D. M., Burke, D. S., Harrison, B. A., Whitmire, R. E., & Nisalak, A.
1038 (2020). Effect of temperature on the transmission of dengue virus by aedes
1039 aegypti. *The American Journal of Tropical Medicine and Hygiene*, 36(1),
1040 143–152.

1041 *What is Postman? Postman API Platform*. (n.d.). Retrieved from [https://](https://www.postman.com/product/what-is-postman/)
1042 www.postman.com/product/what-is-postman/

1043 WHO. (2023). *Dengue - global situation*. Retrieved Use the date of ac-
1044 cess, from [https://www.who.int/emergencies/disease-outbreak-news/](https://www.who.int/emergencies/disease-outbreak-news/item/2023-DON498)
1045 [item/2023-DON498](https://www.who.int/emergencies/disease-outbreak-news/item/2023-DON498)

1046 WHO. (2024). *Dengue and severe dengue*. Retrieved Use the date
1047 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)
1048 [dengue-and-severe-dengue](https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue)

1049 *Why Visual Studio Code?* (2021, 11). Retrieved from <https://code>

1050 `.visualstudio.com/docs/editor/whyvscode`
1051 World Health Organization (WHO). (2018). Dengue and severe dengue in the
1052 philippines. *WHO Dengue Factsheet*. (Available at: `https://www.who`
1053 `.int`)
1054 Zhou, S., & Malani, P. (2024). What is dengue? *Jama*, 332(10), 850–850.
1055 Zod. (n.d.). *TypeScript-first schema validation with static type inference*. Re-
1056 trieved from `https://zod.dev/?id=introduction`

1057 **Appendix A**

1058 **Appendix Title**

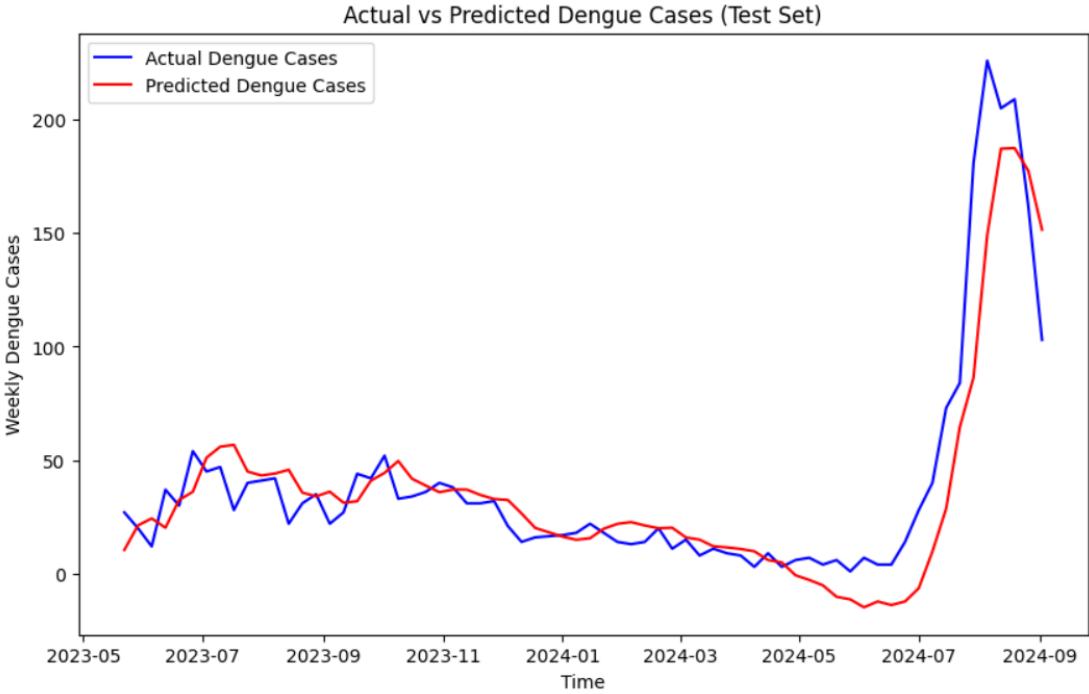


Figure A.1: LSTM Prediction Results for Test Set

1059 **Appendix B**

1060 **Resource Persons**

1061 **Mr. Firstname1 Lastname1**

1062 Role1

1063 Affiliation1

1064 emailaddr1@domain.com

1065 **Ms. Firstname2 Lastname2**

1066 Role2

1067 Affiliation2

1068 emailaddr2@domain.net

1069