

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

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

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

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

16 Francis D. DIMZON
17 Adviser

18 April 26, 2025

Abstract

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

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

Contents

48	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	Backtesting Validation	18
71	3.1.4	Integrate the Predictive Model into a Web-Based Data An-	
72		alytics Dashboard	19
73	3.1.5	System Development Framework	19
74	3.1.6	Design, Building, Testing, and Integration	20
75	3.2	Development Tools	20
76	3.2.1	Software	20
77	3.2.2	Hardware	22
78	3.2.3	Packages	22
79	3.3	Calendar of Activities	24
80	4	Results and Discussion/System Prototype	25
81	4.1	Data Gathering	25
82	4.2	Exploratory Data Analysis	26
83	4.3	Outbreak Detection	30
84	4.4	Model Training Results	31
85	4.4.1	LSTM Model	31
86	4.4.2	ARIMA Model	33
87	4.4.3	Seasonal ARIMA (SARIMA) Model	34

88	4.4.4	Kalman Filter Model	35
89	4.5	Preliminary System Requirements	37
90	4.5.1	Backend Requirements	37
91	4.5.2	User Interface Requirements	38
92	4.5.3	Security and Validation Requirements	40
93	4.5.4	Testing Process	41
94	4.6	System Prototype	42
95	4.6.1	Guest Interface	42
96	4.6.2	Personnel Interface	43
97		References	47
98		A Appendix Title	50
99		B Resource Persons	51

List of Figures

100		
101	3.1	Workflow for forecasting the number of weekly dengue cases . . . 10
102	4.1	Snippet of the Combined Dataset 26
103	4.2	Data Contents 26
104	4.3	Dataset Statistics 27
105	4.4	Distribution of Dengue Cases 27
106	4.5	Frequency of Dengue Cases in Different Intervals 28
107	4.6	Trend of Dengue Cases 29
108	4.7	Ranking of Correlations 29
109	4.8	Ranking of Correlations (with lagged effects) 30
110	4.9	Comparison of Window Sizes 32
111	4.10	ARIMA Prediction Results for Test Set 33
112	4.11	Seasonal ARIMA Prediction Results for Test Set 34
113	4.12	Kalman Filter Prediction Results for Test Set 35
114	4.13	Entity-Relationship Database Schema Hybrid Diagram for DengueDash
115		Database Structure 37
116	4.14	Use Case Diagram for Admin 38
117	4.15	Use Case Diagram for Encoder 39

118	4.16 Testing Process for DengueWatch	41
119	4.17 Dashboard for Guests	42
120	4.18 Login Page for Users	43
121	4.19 First Part of Case Report Form	44
122	4.20 Second Part of Case Report Form	44
123	4.21 Dengue Reports	45
124	4.22 Detailed Case Report	46
125	A.1 LSTM Prediction Results for Test Set	50

126 **List of Tables**

127	3.1	Timetable of Activities for 2024	24
128	3.2	Timetable of Activities for 2025	24
129	4.1	Comparison of Models	31
130	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.

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

162 1.2 Problem Statement

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

175 1.3 Research Objectives

176 1.3.1 General Objective

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

181 1.3.2 Specific Objectives

182 Specifically, this study aims to:

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

197 1.4 Scope and Limitations of the Research

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

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

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

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

223 1.5 Significance of the Research

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

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

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

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

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

286 **2.3 Existing System: RabDash DC**

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

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

303 2.4 Deep Learning

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

317 2.5 Kalman Filter

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

333 Our study will compare ARIMA, seasonal ARIMA, Kalman Filter, and LSTM
334 models using our own collected dengue case data along with weather data to
335 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.

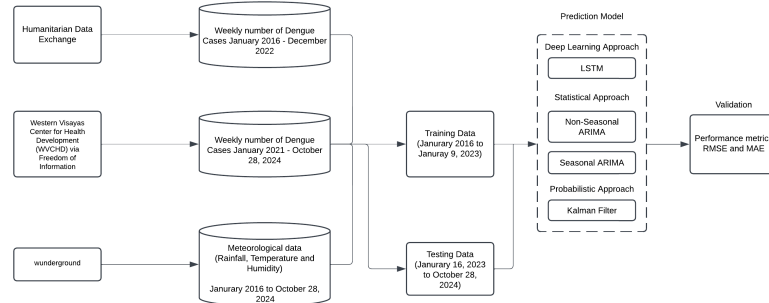


Figure 3.1: Workflow for forecasting the number of weekly dengue cases

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.

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

418 **Data Integration and Preprocessing**

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

426 **Exploratory Data Analysis (EDA)**

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

431 **Outbreak Detection**

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

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

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

435 3.1.2 Develop and Evaluate Deep Learning Models for 436 Dengue Case Forecasting

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

442 Data Preprocessing

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

449 LSTM Model

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

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

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

465 **LSTM units:**

- 466 • min value: 32
- 467 • max value: 128
- 468 • step: 16
- 469 • sampling: linear

470 **Learning Rate:**

- 471 • min value: 0.0001
- 472 • max value: 0.01
- 473 • step: None
- 474 • sampling: log

475 The tuner was instantiated with:

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

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

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

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.

510 Steps to Create the ARIMA Model:

- 511 1. **Data Preprocessing:** Prepare the dataset by handling any missing values
512 and scaling the data if necessary to improve model convergence and stability.
- 513 2. **Hyperparameter Tuning:** Use a grid search on potential ARIMA param-
514 eters (p, d, q) to identify the configuration that minimizes error. The optimal
515 parameters were found to be **(1, 2, 2)**.
- 516 3. **Model Training:**
 - 517 • Set the number of iterations to 400 to ensure thorough training and
518 convergence.
 - 519 • Train the ARIMA model on 80% of the data and reserve 20% for test-
520 ing.

521 Seasonal ARIMA (SARIMA)

522 1. Data Preprocessing

- 523 • Handle missing values through interpolation or imputation.
- 524 • Normalize or standardize features to ensure stable training.
- 525 • Split data into training (80%) and testing (20%) sets while maintaining
- 526 temporal continuity.

527 2. Seasonality Analysis

- 528 • Perform time series decomposition to examine trend, seasonality, and
- 529 residual components.
- 530 • Identify seasonality using autocorrelation plots and spectral analysis.
- 531 • A periodicity of **52 weeks** was detected, justifying the use of a seasonal
- 532 model.

533 3. Hyperparameter Tuning

- 534 • Conduct a grid search to optimize SARIMA parameters $(p, d, q)(P, D, Q)[S]$.
- 535 • Determine optimal configuration for seasonal and non-seasonal compo-
- 536 nents.
- 537 • Verify stationarity through Augmented Dickey-Fuller (ADF) test.

538 4. Model Training

- 539 • Fit the SARIMA model on the training dataset, incorporating exoge-
- 540 nous variables such as rainfall, temperature, and humidity.
- 541 • Set a maximum number of iterations to ensure convergence.
- 542 • Monitor model diagnostics (residual analysis) to confirm the absence
- 543 of autocorrelation in residuals.

544 5. Forecasting and Validation

- 545 • Generate out-of-sample forecasts for future dengue cases.
- 546 • Compare predicted values against actual data to assess real-world ap-
- 547 plicability.
- 548 • Visualize results with line plots and confidence intervals.

549 Kalman Filter:

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

557 The Kalman Filter's EM method was employed for training, iteratively esti-
558 mating model parameters over 10 iterations. The smooth method was used to
559 compute the smoothed state estimates for the training data. Observation matri-
560 ces for the test data were constructed similarly, ensuring compatibility with the
561 trained model.

562 Kalman Filter Methodology with Matrix Calculations

563 **Measurement Acquisition:** Obtain the measurement: (z_k) of the system's state
564 with associated confidence. This measurement matrix provides a noisy observation
565 of the true state.

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

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

571 Prediction Step:

- 572 • Predict the next state:

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

- 573 • Update the state covariance:

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

574 where Q is the process noise covariance matrix.

575 **Compute Residual:** Calculate the residual:

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

576 where H is the observation matrix. This residual represents the new information
577 from the measurement.

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

579 • Compute the Kalman Gain:

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

580 where R is the measurement noise covariance matrix.

581 • The Kalman Gain determines the weight of the measurement relative to the
582 prediction.

583 **State Update:**

584 • Update the state estimate:

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

585 blending the prediction and measurement.

586 **Uncertainty Update:**

587 • Update the state covariance:

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

588 where I is the identity matrix.

589 **3.1.3 Integrate the Predictive Model into a Web-Based** 590 **Data Analytics Dashboard**

591 **Dashboard Design and Development**

592 • Design an intuitive, user-friendly web-based dashboard incorporating:

- 593 – Interactive visualizations of yearly dengue case trends.
- 594 – Data input and update forms for dengue and weather data.
- 595 – Map display of dengue cases in each district in Iloilo City

596 **Model Integration and Deployment**

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

599 **3.1.4 System Development Framework**

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

606 **3.1.5 Design, Building, Testing, and Integration**

607 **Design and Development**

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

621 **Testing and Integration**

622 Each feature will be rigorously user-tested to ensure quality assurance, with par-
623 ticular emphasis on prerequisite features, as development cannot progress properly
624 if these fail. Moreover, integration between each feature serves as a pillar for a
625 cohesive user experience. Presently, we have not been able to use performance

626 metrics to measure the system’s performance, as developing and connecting the
627 core features is the utmost priority.

628 **3.2 Development Tools**

629 **3.2.1 Software**

630 **Github**

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

636 **Visual Studio Code**

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

641 **Django**

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

650 **Next.js**

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

657 **Postman**

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

663 **3.2.2 Hardware**

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

667 **3.2.3 Packages**

668 **Django REST Framework**

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

673 **Leaflet**

674 One of the features of the web application is the ability to map the number
675 of cases using a Choropleth Map. Leaflet is the only free, open-sourced, and

676 most importantly, stable JavaScript package that can do the job. With its ultra-
677 lightweight size, it offers a comprehensive set of features that does not trade
678 off performance and usability (*Leaflet — an open-source JavaScript library for*
679 *interactive maps*, n.d.).

680 **Chart.js**

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

688 **Tailwind CSS**

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

697 **Shadcn**

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

703 **Zod**

704 Data validation is integral in this web application since it will handle crucial data
705 that will be used for analytical inferences and observations. Since Zod is primarily
706 used for validating and parsing data, it ensures proper communication between
707 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 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

729 4.2 Exploratory Data Analysis

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

- 732 • **Time.** Weekly timestamps (e.g. "2011-w1")
- 733 • **Rainfall.** Weekly average rainfall (mm)
- 734 • **MaxTemperature, AverageTemperature, MinTemperature.** Weekly
735 temperature data (C)
- 736 • **Wind.** Wind speed (m/s)
- 737 • **Humidity.** Weekly average humidity (%)
- 738 • **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%.

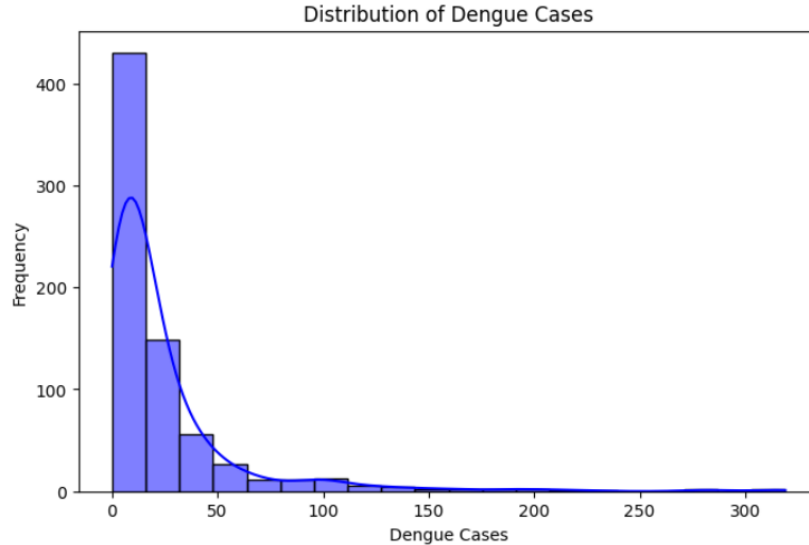


Figure 4.4: Distribution of Dengue Cases

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+

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

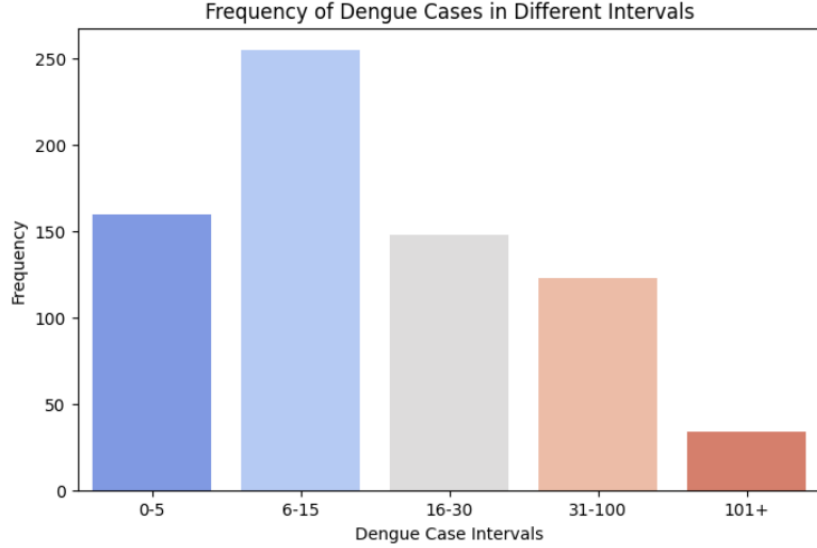


Figure 4.5: Frequency of Dengue Cases in Different Intervals

753

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

760 Figure 4.7 shows the ranking of correlation coefficients between dengue cases
 761 and selected features, including rainfall, humidity, maximum temperature, aver-
 762 age temperature, minimum temperature, and wind speed. Among these, rainfall
 763 exhibits the highest positive correlation with dengue cases (correlation coefficient
 764 0.13), indicating that increased rainfall may contribute to higher cases counts.
 765 This aligns with existing studies suggesting that stagnant water from heavy rain-
 766 fall creates breeding grounds for mosquitos. It is followed by humidity (0.10),
 767 suggesting that higher humidity levels may enhance mosquito reproduction, lead-
 768 ing to more dengue cases. Temperature has a weak to moderate positive corre-
 769 lation with dengue cases, with maximum temperature (0.09) showing a stronger
 770 relationship than average and minimum temperature.

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

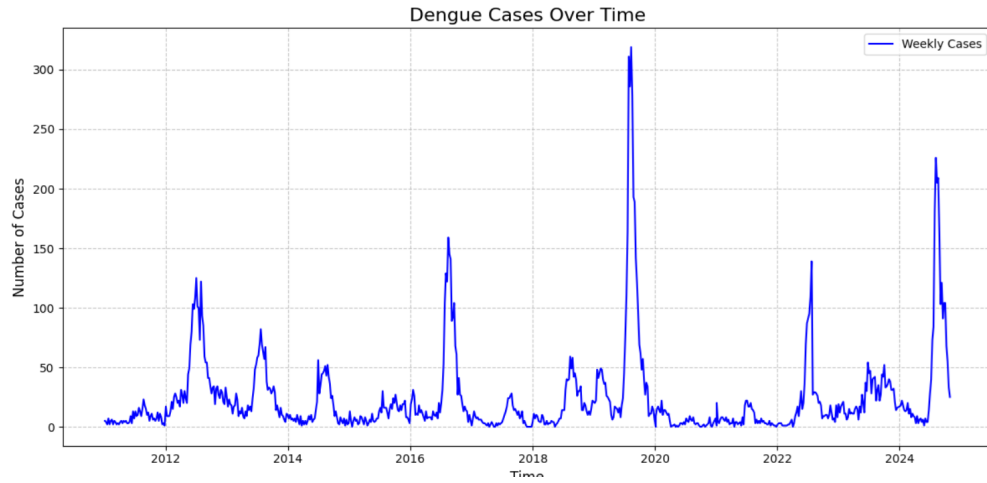


Figure 4.6: Trend of Dengue Cases

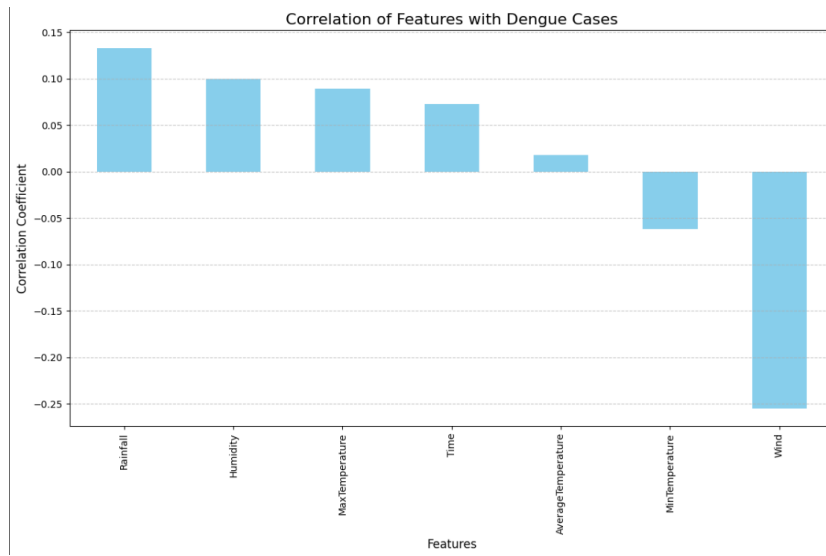


Figure 4.7: Ranking of Correlations

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.

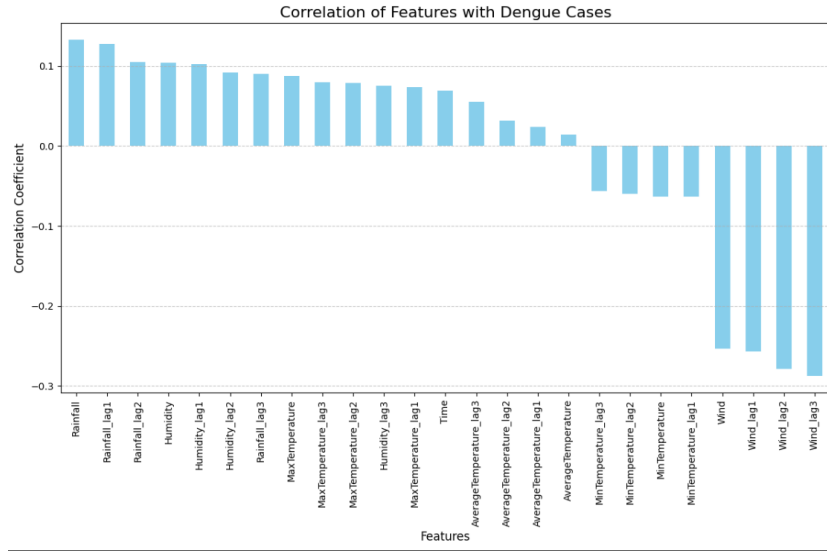


Figure 4.8: Ranking of Correlations (with lagged effects)

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.

787 4.4 Model Training Results

788 The models were evaluated using three metrics: MSE, RMSE, and MAE. The
 789 table below provides a summary and comparative analysis of each model’s results
 790 across these metrics, offering insights into the strengths and limitations of each
 791 forecasting technique for dengue case prediction in Iloilo City. The lower values
 792 of the three metrics indicate better forecasting performance. Table 4.1 shows that
 793 the models performed differently on testing data. LSTM outperformed the other
 794 models with the lowest RMSE, MSE, and MAE while the other three models had
 795 relatively higher values for the three metrics.

Method	LSTM	Seasonal ARIMA	ARIMA	Kalman Filter
Testing MSE	285.54	1109.69	1521.48	1474.82
Testing RMSE	16.90	33.31	39.00	38.40
Testing MAE	9.45	18.08	25.80	22.33
Best Parameters	Window Size: 5 Learning Rate: 0.01 Units: 96	(2, 0, 2)(0, 1, 1)	(1, 2, 2)	–

Table 4.1: Comparison of Models

796 4.4.1 LSTM Model

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

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

Table 4.2: Comparison of Window Sizes

801
 802 The results indicate that a window size of 5 weeks provides the most accurate
 803 predictions, as evidenced by the lowest MSE and RMSE values. Furthermore, the
 804 R² score of 0.83 indicates that 83% of the variability in the target variable (cases)
 805 is explained by the independent variables (the inputs) in the model, making it a
 806 reliable configuration overall.

807 Figure 4.9 illustrate the performance of the model in predicting dengue cases
 808 for each of the specified window sizes. The plots demonstrate that the predicted
 809 cases closely follow the trend of the actual cases, indicating that the LSTM model
 810 successfully captured the underlying patterns in the data. Despite the fact that the
 811 test data is unseen, the model shows a remarkable ability to generalize, suggesting
 812 that the model is effectively leveraging past observations to predict future trends.

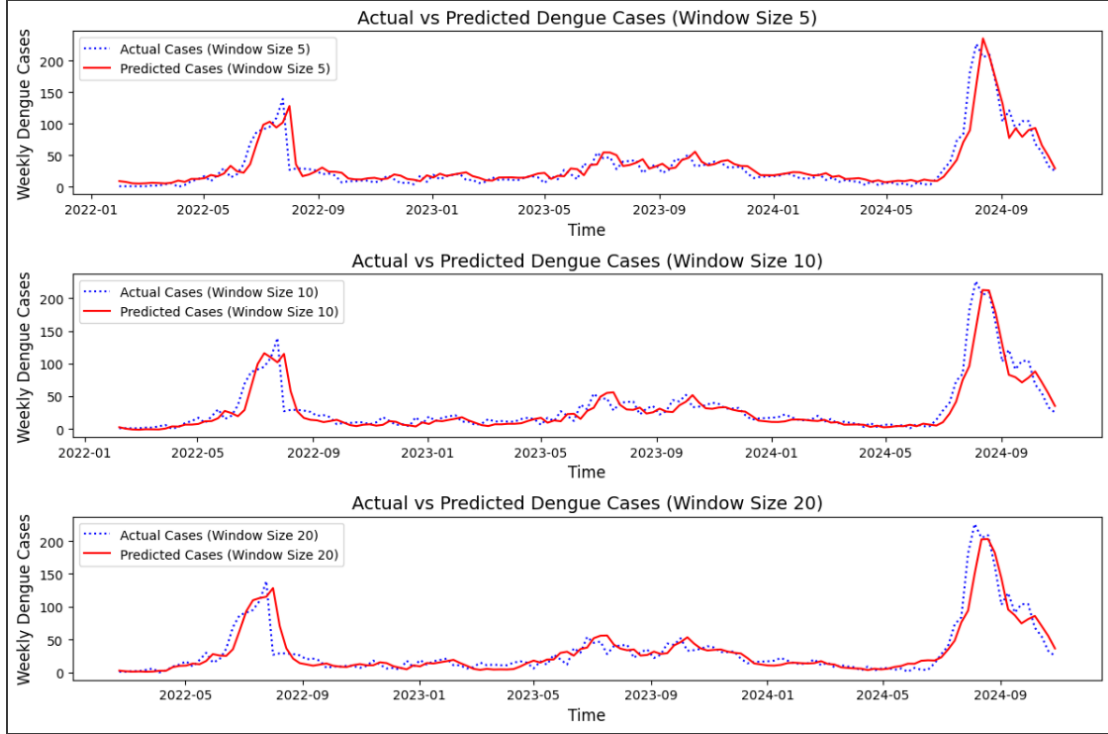


Figure 4.9: Comparison of Window Sizes

813 4.4.2 ARIMA Model

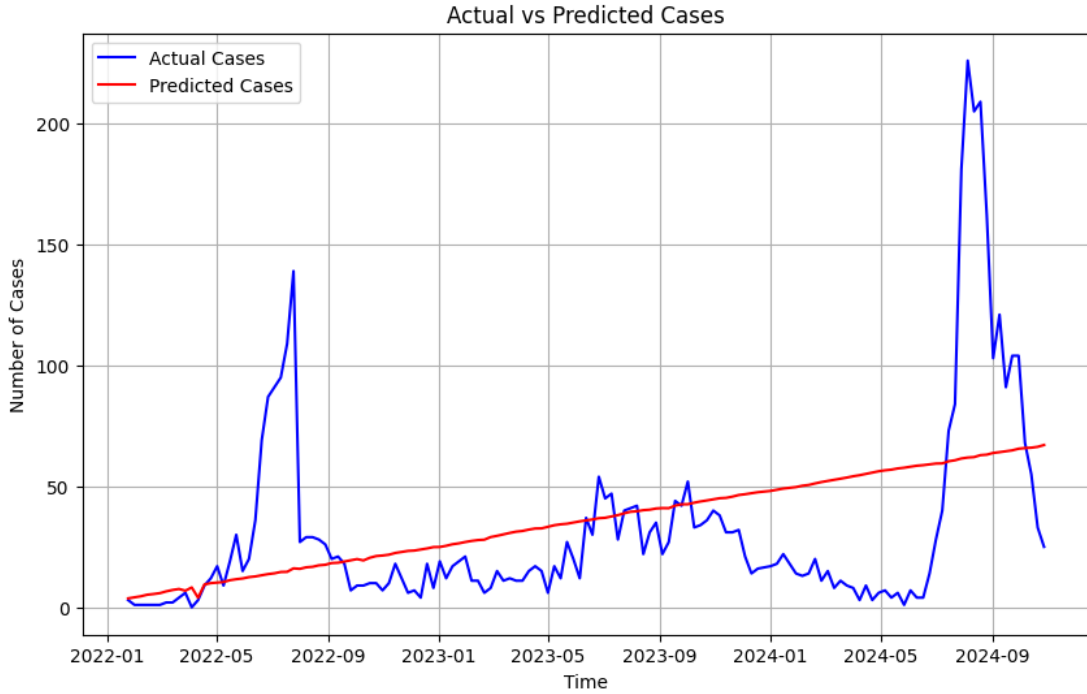


Figure 4.10: ARIMA Prediction Results for Test Set

814 The ARIMA model was developed to capture non-seasonal trends in the data.
 815 To determine the best model configuration, grid search was used to explore vari-
 816 ous combinations of ARIMA parameters, ultimately selecting **ARIMA(1, 2, 2)**.
 817 The model was iteratively refined over **400 iterations** to ensure convergence to
 818 an optimal solution. Figure 4.10 illustrates the comparison between actual and
 819 predicted dengue cases in the test set. As shown in the plot, the ARIMA model
 820 struggled to capture the non-linear characteristics and abrupt spikes in the data.
 821 Consequently, it failed to accurately reflect the fluctuations and outbreak patterns
 822 seen in the actual case counts.

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

- 825 • **MSE (Mean Squared Error):** 1521.48
- 826 • **RMSE (Root Mean Squared Error):** 39.01
- 827 • **MAE (Mean Absolute Error):** 25.80

4.4.3 Seasonal ARIMA (SARIMA) Model

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

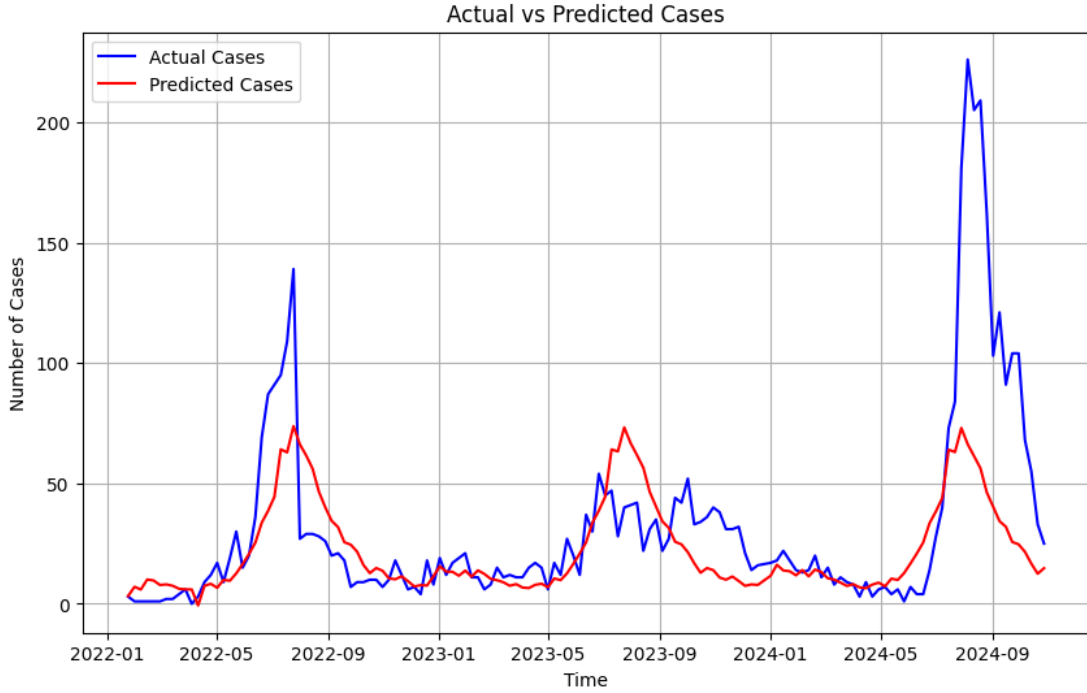


Figure 4.11: Seasonal ARIMA Prediction Results for Test Set

This model incorporates seasonal parameters, which were tuned using grid search to find the best configuration: **SARIMA(2, 0, 2)(0, 1, 1)[52]**. As with ARIMA, **400 iterations** were applied to ensure a robust fit. As shown in Figure 4.11, the SARIMA model demonstrates a notable improvement in performance. Unlike its non-seasonal counterpart, it effectively captures the general trend and aligns more closely with the peaks observed in the actual dengue cases, indicating its ability to model seasonal dynamics.

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

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

843

- MAE: 18.09

844

845

846

847

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

848

4.4.4 Kalman Filter Model

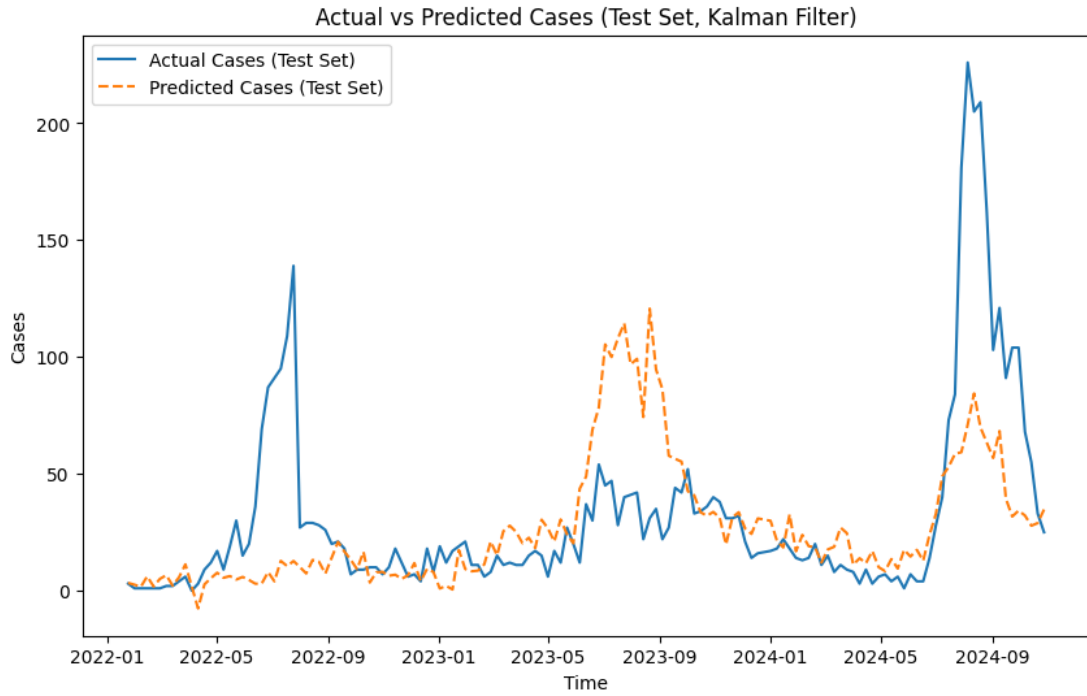


Figure 4.12: Kalman Filter Prediction Results for Test Set

849

850

851

852

853

854

855

856

Figure 4.12 shows the comparison between the actual dengue cases and the predicted values on the test set. As illustrated in the plot, the Kalman Filter model demonstrates a moderate ability to follow the general trend of the actual data. While it effectively captures some rising and falling patterns, it still struggles to accurately replicate the sharp peaks and extreme values found in the real case counts. This limitation is particularly noticeable during the large spikes in 2022 and 2024.

857 The model's performance was evaluated using standard regression metrics.
858 The results are as follows:

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

859 These metrics indicate that the Kalman Filter outperforms the ARIMA model
860 in terms of mean absolute error (MAE), suggesting better accuracy in captur-
861 ing day-to-day fluctuations. However, it still underperforms compared to the
862 SARIMA model, particularly in modeling seasonal trends and sharp outbreaks.
863 Despite its limitations, the Kalman Filter shows promise for short-term forecasting
864 due to its adaptability and real-time updating capability.

865 4.5 Preliminary System Requirements

866 4.5.1 Backend Requirements

867 Database Structure Design

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

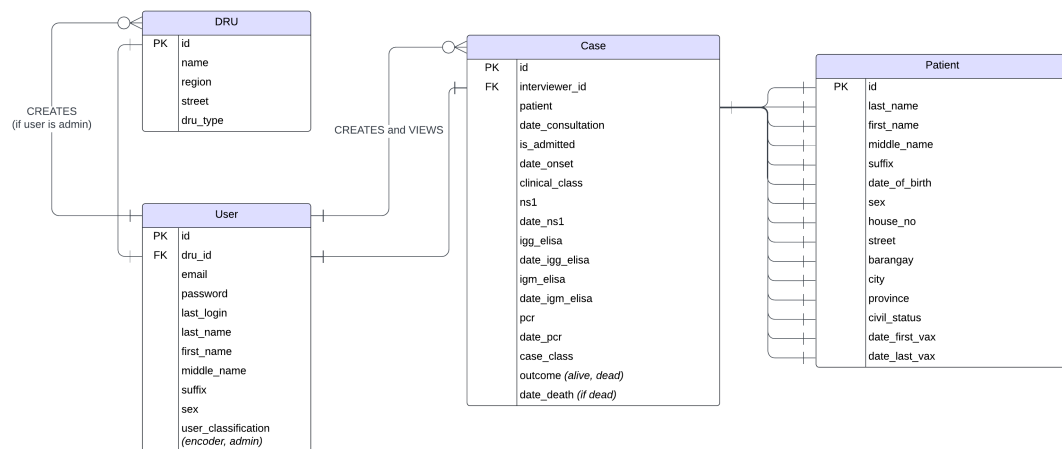


Figure 4.13: Entity-Relationship Database Schema Hybrid Diagram for DengueDash Database Structure

874 4.5.2 User Interface Requirements

875 Admin Interface

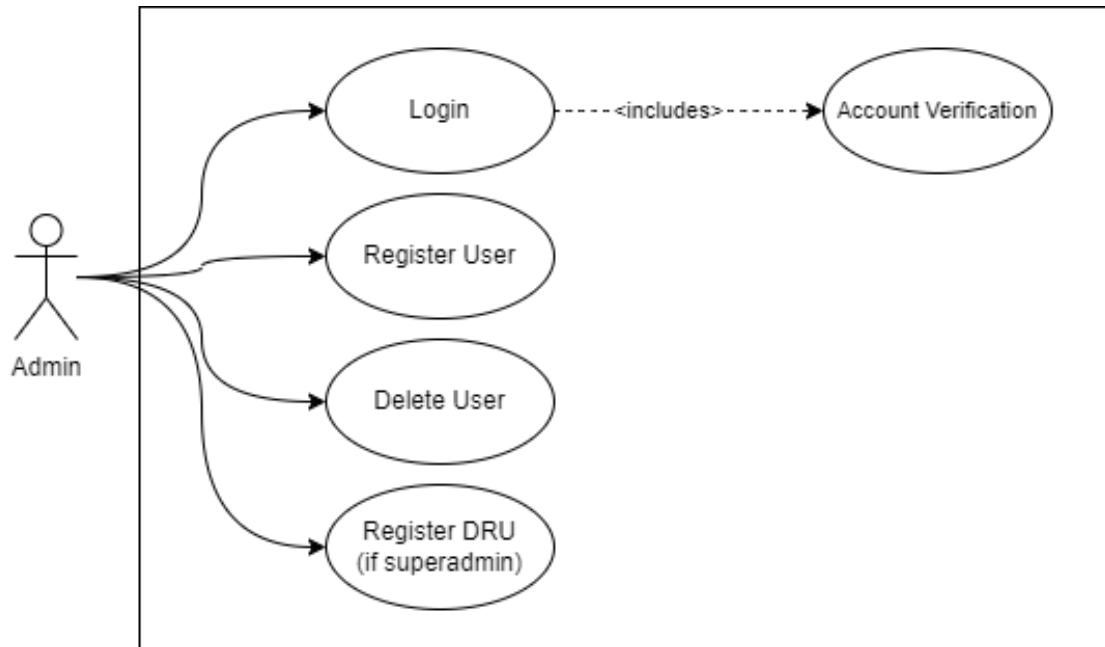


Figure 4.14: Use Case Diagram for Admin

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

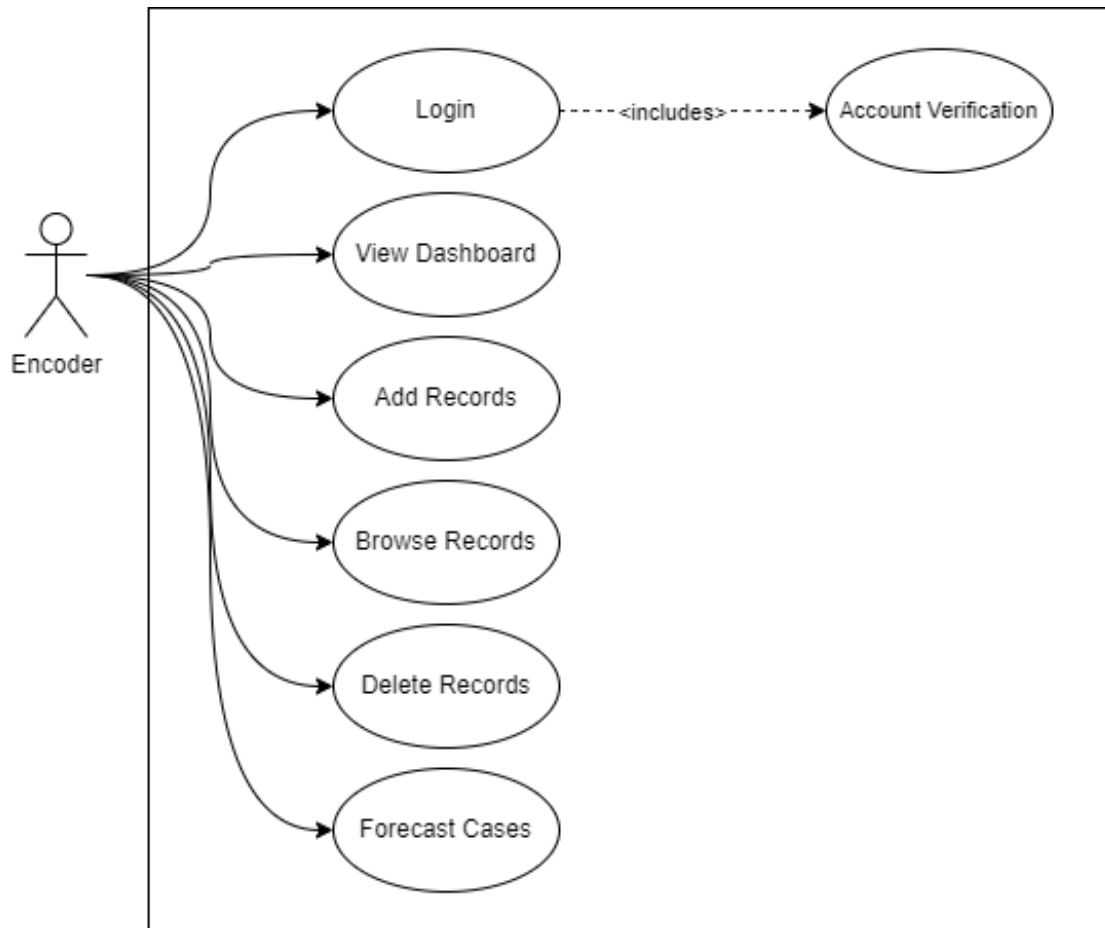


Figure 4.15: Use Case Diagram for Encoder

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

887 4.5.3 Security and Validation Requirements

888 Password Encryption

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

897 Authentication

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

903 Data Validation

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

912 4.5.4 Testing Process

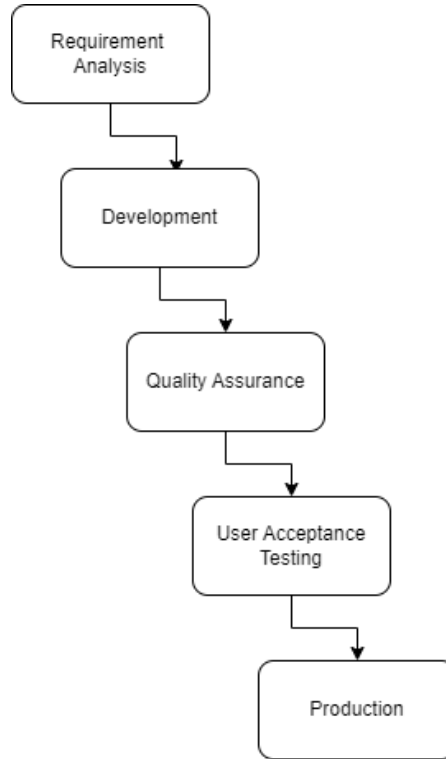


Figure 4.16: Testing Process for DengueWatch

913 As the system requirements and functionalities have been mentioned above, it
914 is important to implement testing to validate the system's performance and effi-
915 cacy. Since dengue reports include confidential information, anonymized historical
916 dengue reports were used to train the model and create the foundational architec-
917 ture of the system. By using functional tests, data validation and visualization can
918 be ensured for further continual improvements. Security testing is also important
919 as it is needed to safeguard confidential information when the system is deployed.
920 It includes proper authentication, permission views, and mitigating common in-
921 jection attacks. Finally, a user acceptance test from the prospected users, in this
922 case, the Iloilo City Epidemiology and Surveillance Unit, is crucial to assess its
923 performance and user experience. It enables the developers to confirm if the sys-
924 tem meets the needs of the problem, and once confirmed, it will be deployed and
925 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.

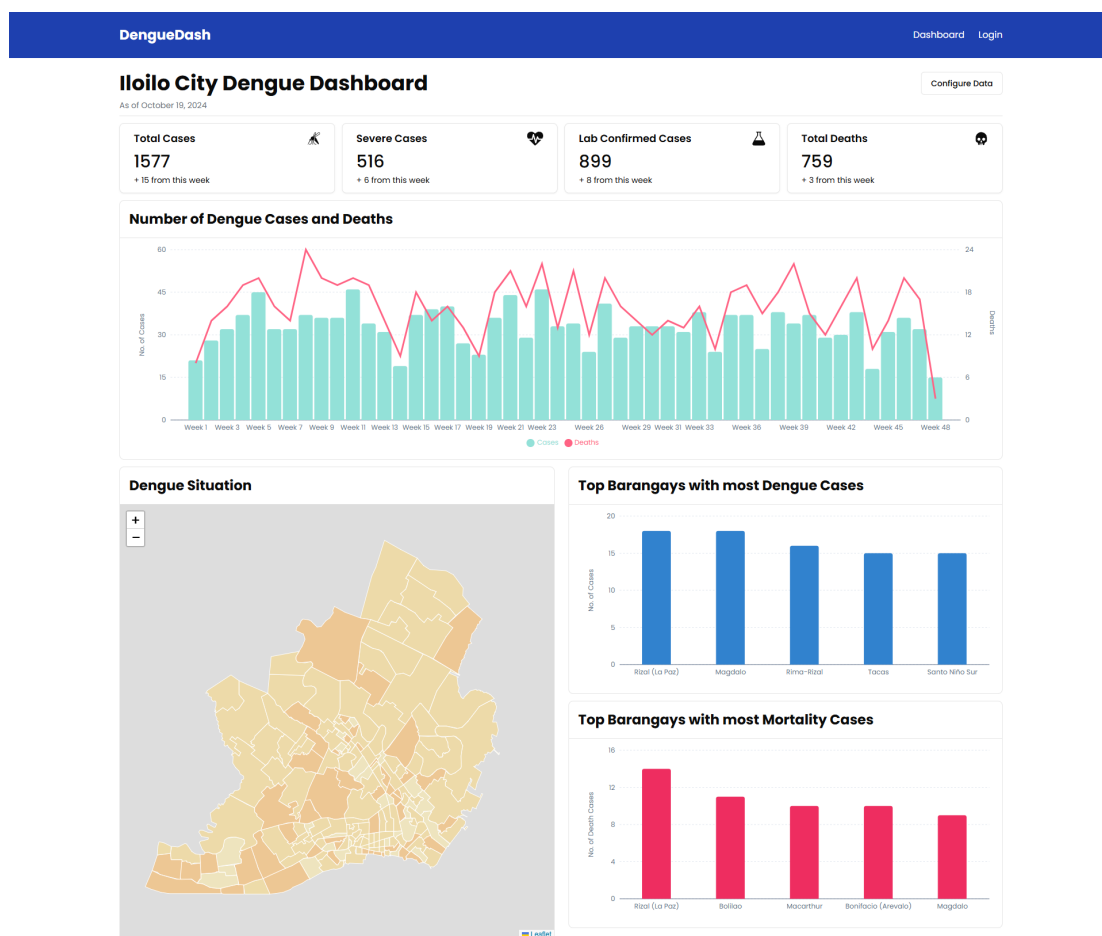


Figure 4.17: Dashboard for Guests

932 4.6.2 Personnel Interface

933 User Authentication, and Login

934 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
935 a different interface. As of the moment, registering a user is done using API via
936 Postman. In the login process, the system implements HTTP-only cookies that
937 contains the JSON Web Tokens (JWT) to protect against XSS attacks. After
938 proper credentials have been provided, it will redirect to the user's home page.
939

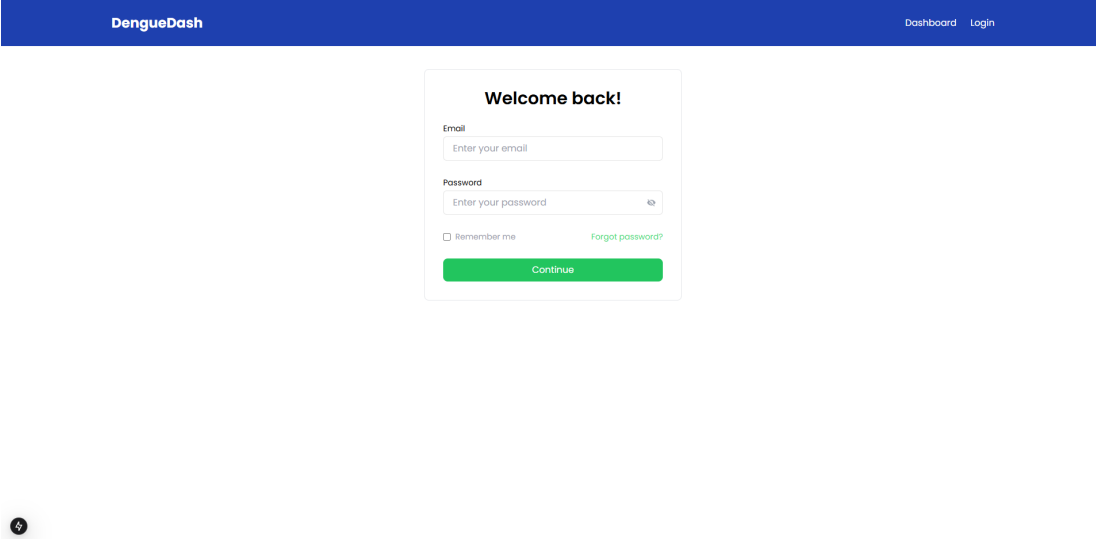


Figure 4.18: Login Page for Users

940 Encoder's View

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

DengueDash

Modules

Analytics

Forms

Case Report Form

Data Tables

Settings

CN

shadcn

m@example.com

0

Building Your Application

Data Fetching

Personal Information

Clinical Status

Personal Detail

First Name

Middle Name

Last Name

Suffix

Sex

Civil Status

Date of Birth

Address

House No.

Street

Barangay

City

Province

Vaccination

Date of First Vaccination

Date of Last Vaccination

Back

Next

Figure 4.19: 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

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

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

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

Figure 4.21: Dengue Reports

References

- 962 *About GitHub and Git - GitHub Docs.* (n.d.). Retrieved from
963 [https://docs.github.com/en/get-started/start-your-journey/
964 about-github-and-git](https://docs.github.com/en/get-started/start-your-journey/about-github-and-git)
- 965 Ahmadini, A. A. H., Naeem, M., Aamir, M., Dewan, R., Alshqaq, S. S. A., &
966 Mashwani, W. K. (2021). Analysis and forecast of the number of deaths,
967 recovered cases, and confirmed cases from covid-19 for the top four affected
968 countries using kalman filter. *Frontiers in Physics*, 9, 629320.
- 969 Arroyo-Marioli, F., Bullano, F., Kucinskas, S., & Rondón-Moreno, C. (2021).
970 Tracking r of covid-19: A new real-time estimation using the kalman filter.
971 *PloS one*, 16(1), e0244474.
- 972 Bosano, R. (2023). *Who: Ph most affected by dengue in western pacific*. Retrieved
973 Use the date of access, from [https://news.abs-cbn.com/spotlight/12/
974 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)
- 975 Brady, O. J., Smith, D. L., Scott, T. W., & Hay, S. I. (2015). Dengue disease
976 outbreak definitions are implicitly variable. *Epidemics*, 11, 92–102.
- 977 Bravo, L., Roque, V. G., Brett, J., Dizon, R., & L’Azou, M. (2014). Epidemiology
978 of dengue disease in the philippines (2000–2011): a systematic literature
979 review. *PLoS neglected tropical diseases*, 8(11), e3027.
- 980 Carvajal, T. M., Viacrusis, K. M., Hernandez, L. F. T., Ho, H. T., Amalin, D. M.,
981 & Watanabe, K. (2018). Machine learning methods reveal the temporal
982 pattern of dengue incidence using meteorological factors in metropolitan
983 manila, philippines. *BMC infectious diseases*, 18, 1–15.
- 984 *Chart.js.* (n.d.). Retrieved from <https://www.chartjs.org/>
- 985 Cheong, Y. L., Burkart, K., Leitão, P. J., & Lakes, T. (2013). Assessing weather
986 effects on dengue disease in malaysia. *International journal of environmental
987 research and public health*, 10(12), 6319–6334.
- 988 Christie, T. (n.d.). *Home - Django REST framework.* Retrieved from [https://
989 www.django-rest-framework.org/](https://www.django-rest-framework.org/)
- 990 Colón-González, F. J., Fezzi, C., Lake, I. R., & Hunter, P. R. (2013). The effects
991 of weather and climate change on dengue. *PLoS neglected tropical diseases*,
992 7(11), e2503.

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

995 Hii, Y. L., Zhu, H., Ng, N., Ng, L. C., & Rocklöv, J. (2012). Forecast of dengue
996 incidence using temperature and rainfall. *PLoS neglected tropical diseases*,
997 6(11), e1908.

998 Joel, C. (2021, 10). *6 reasons to use Tailwind over traditional CSS*. Re-
999 trieved from [https://dev.to/charliejoel/6-reasons-to-use-tailwind](https://dev.to/charliejoel/6-reasons-to-use-tailwind-over-traditional-css-1nc3)
1000 [-over-traditional-css-1nc3](https://dev.to/charliejoel/6-reasons-to-use-tailwind-over-traditional-css-1nc3)

1001 Leaflet — an open-source JavaScript library for interactive maps. (n.d.). Retrieved
1002 from <https://leafletjs.com/>

1003 Lena, P. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use
1004 the date of access, from <https://www.pna.gov.ph/articles/1231208>

1005 Li, X., Feng, S., Hou, N., Li, H., Zhang, S., Jian, Z., & Zi, Q. (2022). Applications
1006 of kalman filtering in time series prediction. In *International conference on*
1007 *intelligent robotics and applications* (pp. 520–531).

1008 Ligue, K. D. B., & Ligue, K. J. B. (2022). Deep learning approach to forecasting
1009 dengue cases in davao city using long short-term memory (lstm). *Philippine*
1010 *Journal of Science*, 151(3).

1011 Perla. (2024). *Iloilo beefs up efforts amid hike in dengue cases*. Retrieved Use the
1012 date of access, from <https://www.pna.gov.ph/articles/1231208>

1013 RabDashDC. (2024). *Rabdash dc*. Retrieved Use the date of access, from [https://](https://rabdash.com)
1014 rabdash.com

1015 Runge-Ranzinger, S., Kroeger, A., Oliaro, P., McCall, P. J., Sánchez Tejeda, G.,
1016 Lloyd, L. S., ... Coelho, G. (2016). Dengue contingency planning: from
1017 research to policy and practice. *PLoS neglected tropical diseases*, 10(9),
1018 e0004916.

1019 Shadcn. (n.d.). *Introduction*. Retrieved from <https://ui.shadcn.com/docs>
1020 *Tailwind CSS - Rapidly build modern websites without ever leaving your HTML*.
1021 (n.d.). Retrieved from <https://tailwindcss.com/>

1022 Watts, D. M., Burke, D. S., Harrison, B. A., Whitmire, R. E., & Nisalak, A.
1023 (2020). Effect of temperature on the transmission of dengue virus by aedes
1024 aegypti. *The American Journal of Tropical Medicine and Hygiene*, 36(1),
1025 143–152.

1026 *What is Postman? Postman API Platform*. (n.d.). Retrieved from [https://](https://www.postman.com/product/what-is-postman/)
1027 www.postman.com/product/what-is-postman/

1028 WHO. (2023). *Dengue - global situation*. Retrieved Use the date of ac-
1029 cess, from [https://www.who.int/emergencies/disease-outbreak-news/](https://www.who.int/emergencies/disease-outbreak-news/item/2023-DON498)
1030 [item/2023-DON498](https://www.who.int/emergencies/disease-outbreak-news/item/2023-DON498)

1031 WHO. (2024). *Dengue and severe dengue*. Retrieved Use the date
1032 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)
1033 [dengue-and-severe-dengue](https://www.who.int/news-room/fact-sheets/detail/dengue-and-severe-dengue)

1034 *Why Visual Studio Code?* (2021, 11). Retrieved from <https://code>

1035 `.visualstudio.com/docs/editor/whyvscod`
1036 World Health Organization (WHO). (2018). Dengue and severe dengue in the
1037 philippines. *WHO Dengue Factsheet*. (Available at: `https://www.who`
1038 `.int`)
1039 Zhou, S., & Malani, P. (2024). What is dengue? *Jama*, 332(10), 850–850.
1040 Zod. (n.d.). *TypeScript-first schema validation with static type inference*. Re-
1041 trieved from `https://zod.dev/?id=introduction`

1042 **Appendix A**

1043 **Appendix Title**

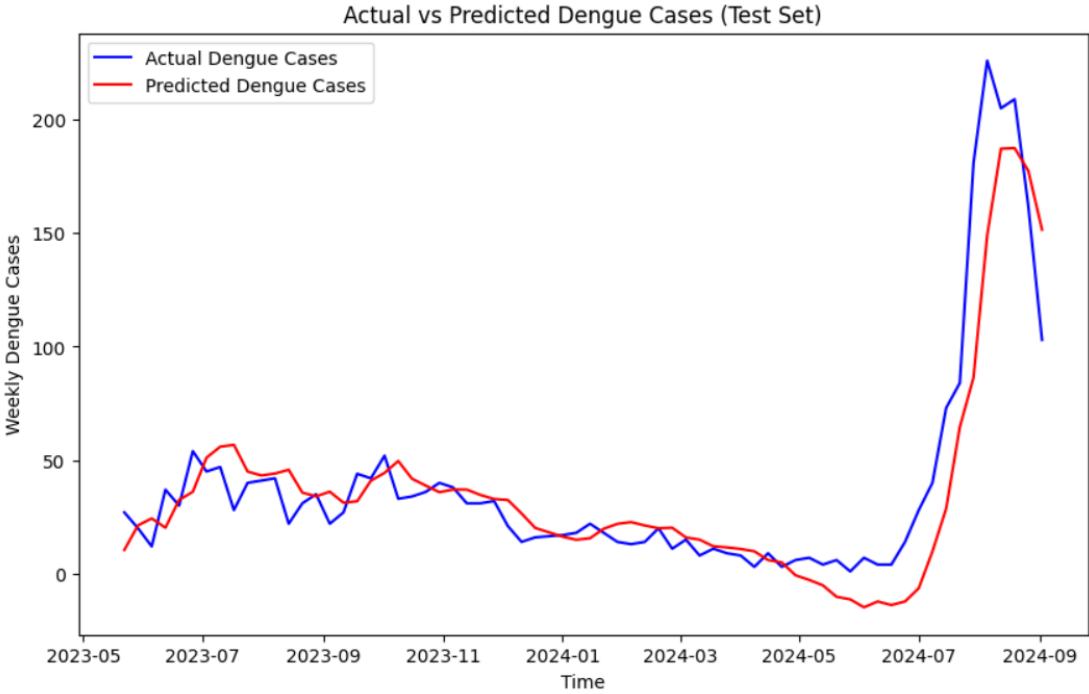


Figure A.1: LSTM Prediction Results for Test Set

1044 **Appendix B**

1045 **Resource Persons**

1046 **Mr. Firstname1 Lastname1**

1047 Role1

1048 Affiliation1

1049 emailaddr1@domain.com

1050 **Ms. Firstname2 Lastname2**

1051 Role2

1052 Affiliation2

1053 emailaddr2@domain.net

1054