



Development and user testing study of MozzHub: a bipartite network-based dengue hotspot detector

Jane Labadin¹ · Boon Hao Hong¹ · Wei King Tiong¹ · Balvinder Singh Gill² · David Perera³ · Andrew Ragai Henry Rigit⁴ · Sarbhan Singh² · Cia Vei Tan² · Sumarni Mohd Ghazali² · Jenarun Jelip⁵ · Norhayati Mokhtar⁵ · Norafidah binti Abdul Rashid⁵ · Hazlin Bt Abu Bakar⁵ · Jyh Hann Lim⁵ · Norsyahida Md Taib⁵ · Aaron George⁵

Received: 25 March 2022 / Revised: 14 October 2022 / Accepted: 25 October 2022 /
Published online: 11 November 2022

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract

Traditionally, dengue is controlled by fogging, and the prime location for the control measure is at the patient's residence. However, when Malaysia was hit by the first wave of the Coronavirus disease (COVID-19), and the government-imposed movement control order, dengue cases have decreased by more than 30% from the previous year. This implies that residential areas may not be the prime locations for dengue-infected mosquitoes. The existing early warning system was focused on temporal prediction wherein the lack of consideration for spatial component at the microlevel and human mobility were not considered. Thus, we developed MozzHub, which is a web-based application system based on the bipartite network-based dengue model that is focused on identifying the source of dengue infection at a small spatial level (400 m) by integrating human mobility and environmental predictors. The model was earlier developed and validated; therefore, this study presents the design and implementation of the MozzHub system and the results of a preliminary pilot test and user acceptance of MozzHub in six district health offices in Malaysia. It was found that the MozzHub system is well received by the sample of end-users as it was demonstrated as a useful (77.4%), easy-to-operate system (80.6%), and has achieved adequate client satisfaction for its use (74.2%).

Keywords Vector control · Source of infection · Fogging · Malaysia · Model implementation

1 Introduction

Over the last five decades, dengue has evolved and become the world's fastest-growing mosquito-borne disease. With approximately half of the world's population residing in

✉ Jane Labadin
ljane@unimas.my

dengue high-risk areas, this has resulted in an estimated 100–400 million annual infections alongside an increasing number of dengue mortalities. Consequently, healthcare systems worldwide are burdened with rising numbers of dengue infections, which unfortunately has no effective treatment or cure to date. Malaysia is a country located in the Southeast Asian region, which had first reported dengue infections in 1902. Since then, dengue has become a major public health issue for Malaysians with the first major outbreak being reported in 1973 [27]. In Malaysia, dengue control measures are primarily focused on source reduction (i.e., larviciding) and vector control measures (i.e., thermal fogging, ultra-low-volume spray, and indoor residual spray), which are achieved through active engagement between health authorities and the local community [28]. Effective dengue control measures are crucial in breaking the chain of transmission and subsequently reducing dengue-related morbidity and mortality [29]. Despite the institution of the existing dengue control measure, we remain unable to eradicate dengue; consequently, dengue cases and outbreaks continue to increase [8]. One of the reasons for this could be attributed to challenges in identifying the correct source of dengue infection. As the majority of individuals tend to have frequent daily movements from one location to another (i.e., a person will visit the workplace, school, premises, park, and other places), this makes identification of the exact location of infection highly challenging. Furthermore, the accuracy of the location of infection is highly dependent on the mobility history provided by the patient, which could be subject to recall bias. All of this, in essence, may result in dengue control measures being directed at incorrect locations (i.e., conducting residual spraying only in the housing area and omitting other potential sources of infection, including the workplace, park, etc.), thereby defeating the purpose.

Malaysia reported a 30.3% reduction in dengue cases in 2020 (88,845) compared with that in 2019 (127,407) [26]. In 2020, a movement control order (MCO) was instituted in Malaysia to curb the Coronavirus disease (COVID-19) outbreak, wherein during the MCO, all individuals were urged to stay at home to avoid getting infected by COVID-19. Studies have suggested that the reduction in dengue cases observed in 2020 could be attributed to the MCO, which restricted the mobility of individuals; consequently, several individuals were confined to their homes. If this was the case, then it would be safe to imply that most residences may not be the common source location of dengue infections; otherwise, we would have experienced an increase in dengue cases during 2020. Therefore, dengue control measures should not primarily target residential areas but should also consider other potential locations (i.e., other potential sources of infections based on patient mobility patterns) [10].

Typical studies on vector-borne disease transmission applied data-driven methods, such as the statistical approach [7, 20, 33] and machine learning techniques [6, 19, 30, 32], for integrating the environmental factors and epidemiology datasets as alarm variables for predicting the disease outbreak. These studies were focused on the seasonality occurrences of the diseases to early identify a high-risk region and implemented as an early warning system (EWS) [15, 32] to prevent the outbreak. However, the regions identified by those studies are at the macro level, such as a city, district, province, and state in a country, which is the current limitation for EWS [16]. Hussain-Alkhateeb et al. [16] reported that the features in most of the current EWS lack the ability to identify the high transmission areas at a microlevel. The area at a macrolevel may not be suitable for identifying the probable sources of dengue infection for vector control measures because

high spatial prediction ability is defined as microlevel [16]. Moreover, vector-borne diseases, including dengue, are influenced by several inter-related factors, such as human mobility [34, 35], which were not considered in those studies or the approaches required a large scale of data [32].

To identify the probable sources of dengue infection, the mobility history of patients with dengue taken 2 weeks before the onset of symptoms is critical. Each location visited will need to be quantified based on selected parameters, including the environmental factors that affect the dengue mosquito's survival rates. However, the run time of data-driven approaches is shorter than those of other approaches, which is suitable for use in a real-time system [5]. However, most EWSs for dengue were not operated in a real-time environment; therefore, a model that can integrate the heterogeneity of human effect and the location visited is more essential in an EWS. Deep learning approaches were widely employed to overcome the shortage of machine learning approaches, particularly in spatio-temporal problems [1, 2, 31], which take human factors and environmental predictors into account. Despite that, the input size must be large enough to generate a high-accuracy model, and high computational power is required for model training [38]. However, the decision-making process to develop a trained model is not frequently interpretable by humans and lacks common sense. This can lead to several issues, particularly for policymakers, since the reasons must be provided before enforcing new policies [38].

To address the abovementioned issues, the bipartite network approach was used by researchers [11, 23] to study the identification of disease hotspots. A bipartite network model (BNM) is a model made up of two discrete entities that consist of different sets of nodes. Each node in one entity is linked to a node in the other entity. Links between connected nodes are weighted to define the contact strength of the link between the two nodes. As highlighted by [23], the least amount of data is sufficient to formulate a network model to provide effective prediction; therefore, the bipartite network has been widely used in several areas, including biology [12, 18, 24], sociology [3, 4, 18, 25], and ecology [37]. Most diseases can be formulated into a bipartite network, which is human and location. By assigning selected parameters to the human and location entities, the link weight can be quantified to rank the locations based on the contact strength between human and location. Therefore, the patients' mobility history and environmental factors can be used in a BNM to identify the most probable source of dengue infection.

There is an urgent need to accurately detect the potential sources of dengue infections to achieve effective dengue control. However, to date, there are limited dengue models that are able to accurately identify sources of dengue infection. Therefore, this study presents a validated dengue network model [22, 23] that evaluates patient mobility history and environmental data, including temperature, relative humidity, and precipitation, to identify sources of dengue infections in Malaysia. This study primarily aimed to discuss the development, design, and implementation of the well-established model [23] as a user-friendly tool—MozzHub system—which is a bipartite network-based dengue hotspot detector for detecting the potential source of dengue infection. Furthermore, we performed user acceptance testing (UAT) of the MozzHub system in several district health offices in Malaysia. We believe that the MozzHub system would be able to assist public health authorities in efficiently deciding on deploying dengue control measures and preventing future dengue outbreaks in Malaysia.

2 Methods

2.1 Requirement analysis

Kok et al. (2017) previously discussed the formulation of the bipartite network-based dengue model, which is consisted of the following three sub-modules: 1) location node quantification, 2) link weight quantification, and 3) visualization via heatmap [19]. These sub-modules were developed to facilitate bipartite network-related research that incorporates location as one of its entities within the network. Details of the sub-modules are described in the following paragraphs.

Sub-module I: Location node quantification As aforementioned, the bipartite network can be applied to several real-world problems that deal with mobility [10, 14, 23, 37]. In previous studies, locations were converted to global positioning system (GPS) coordinates in the form of latitude and longitude. These coordinates are used as inputs of distance matrix to remove any redundant location and used to retrieve environmental data from a weather application programming interface (API). This sub-module assists the researcher to automate the process of converting GPS coordinates, location clustering, and retrieving environmental data. Based on the model formulated [23], the maximum flight range of the *Aedes* mosquito is 400 m; thus, a new location is declared in the database if the distance between the new location node and the existing location node is greater than 400 m. This sub-module requires the user to provide good input on locality as well as the date and time of locations visited in a spreadsheet format and subsequently input the data to this sub-module. The parameters included for the location node in the model are relative humidity (H), precipitation (Pre), life cycle index (Lc), survival index (S), biting rate (B) of *Aedes* mosquito, altitude (Al), and the number of times a location visited by the respective human (Fl) [23]. Lc , S , and B are calculated by using temperature (T). The three weather parameters, including T , H and Pre , are obtained using the weather API that is provided by World Weather Online.

Sub-module II: Link weight quantification In this sub-module, the link weight between the connected human and location nodes will be quantified. In this study, the parameters of the human node are the number of times a human visited a location (Flh) and the time duration of the human stay at one location (Du) [23], which is retrieved based on the patient mobility data. Therefore, both parameters constitute the human node's quantification for the network. To quantify the link weight, all the paired nodes require to be sorted in a spreadsheet by listing all the parameter values for both nodes. The link weight of the connected nodes will be quantified by adopting the summation rule [23], and all of the parameters will be normalized before quantification. In sub-module II, the weighted contact network will be formulated and used to generate the ranking of the location node. The ranking algorithm adopted in this study is the hyperlink-induced topic search (HITS) [21].

Sub-module III: Visualization via heatmap A heatmap is a graphical data representation method with different variety densities of colors to define the relationship of data values in the heatmap. The objective of this sub-module is to generate a heatmap to better visualize the output on the relevant geographical map generated from the previous sub-modules. Similar to

the previous sub-modules, users are required to list the locations with the corresponding GPS coordinates and rank the locations in a spreadsheet to allow this sub-module to generate the heatmap.

In summary, these three sub-modules improved the workflow of a BNM-related study, thereby allowing the user to reuse the sub-modules in their case study. However, these three sub-modules are researcher oriented and may not be user-friendly for non-researchers. Public health officers are the main users of the bipartite network-based dengue model and may possess insufficient knowledge in modeling and programming techniques to be able to use the previous sub-modules. Additionally, data preparation is time-consuming. Therefore, these three sub-modules should be integrated and implemented into an automated system. The design of the integrated system is presented in the following subsection.

2.2 System design

The overall system architecture of MozzHub is shown in Fig. 1. The MozzHub is deployed and hosted at a workstation in the Universiti Malaysia Sarawak research facility. As MozzHub is a web-based application, users are required to access the MozzHub system using a web browser, such as Google Chrome, Mozilla Firefox, or Microsoft Edge. The user interface is designed using React, which is an open-source front-end JavaScript library. In this study, sub-module III is implemented on the client side by requesting the data from the backend. Leaflet is an open-source JavaScript (JS) library used to develop interactive maps for a web application. Thus, it is selected to realize the heatmap in MozzHub. To assist users in easily adopting the system, the format of the actual investigation form used by public health authorities was converted into hypertext markup language (HTML) form. Therefore, all the data collected by

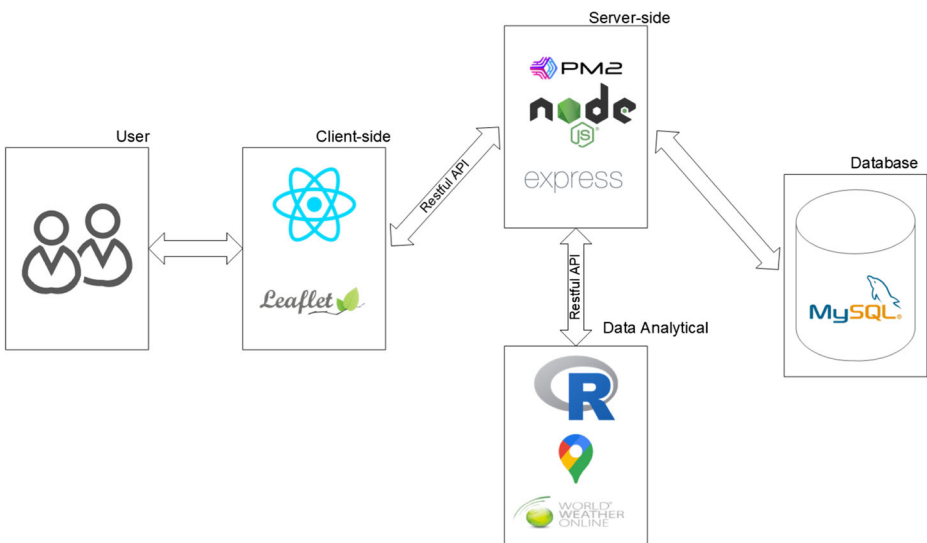


Fig. 1 System architecture of MozzHub

MozzHub are based on the investigation form approved by the Malaysia Ministry of Health.

Moreover, the server side was deployed and hosted on the same workstation and built based on Node JS and Express JS for RESTful API. The process manager 2 (PM2) is used to manage and monitor the server script for MozzHub. Therefore, the server side will act as a proxy between the client side and database; data analytics and database to perform data management. All requests will be handled by the server side, including user authentication, data retrieval, data storing, and other relevant database processes. To ensure the correct data format before storing it in the database, preliminary error checking of the data submitted by the user from the client side is performed at the server side.

As previously mentioned, sub-modules I and II correspond to the formulation of a bipartite network for dengue. To ensure compatibility with the overall system architecture, these sub-modules were rewritten in R language. One of the objectives of this study is to automate the process of formulating the network; therefore, all data will be retrieved by the script itself via RESTful API. Thus, the user is not required to recognize the R language since the network model will be automatically formulated. Data analytics scripts are hosted on the Windows platform. Therefore, Task Scheduler provided by Windows is used to set the scripts' execution schedule.

The database management system (DBMS) used in MozzHub is MySQL. MySQL is an open-source relational DBMS. Thus, it is suitable for the database design in MozzHub. The only proxy to interact with the database is the server side. Data update, retrieval, and definition are initially handled by the server side and subsequently passed to the database. All the required data for analysis and output are requested and stored by calling the ResfulAPI, which is the only way to communicate with the database via the server side.

2.3 Data flow design

In this study, a data flow diagram (DFD) is used to depict the design of the data flow in MozzHub. DFDs are constructed in layers wherein the top level is defined as the context diagram (Level-0); Level-1 is presented in the following subsections.

2.3.1 Context diagram

The context diagram is a graphical representation of the data flow at the highest level containing the entire MozzHub system. The major data flows are presented in Fig. 2. The external entity in MozzHub is the main user, which is the person that interacts with the system.

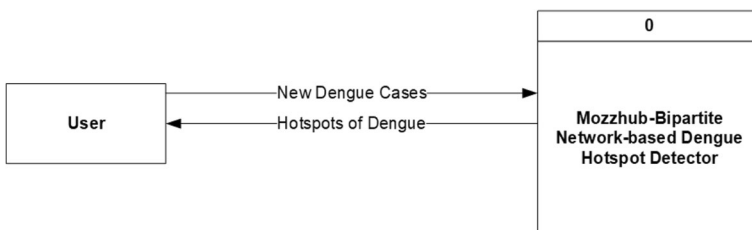


Fig. 2 Context diagram of MozzHub

Dengue cases will be inputted by the user into MozzHub by following the investigation form. MozzHub will subsequently process and analyze the data to generate the hotspot of dengue as the output.

2.3.2 Level-1 DFD

The level-1 data flow corresponds to the processes within the context diagram. The details of each child’s processes in the context diagram are illustrated in Fig. 3. A total of seven child processes and five data stores are defined in level-1 DFD. The only data inputted by the user is the dengue case details with patient movement records. As mentioned in system architecture, data will be stored in the database via the server side; therefore, data insertion or retrieval is performed by calling RESTful API. Therefore, case details provided by the user will be stored in DS1 after being processed by process 1. Users can review the existing cases that users previously submitted via process 2, which will retrieve the required data from DS1 and return it to the user. Processes 3–6 are mainly focused on the data analytical component highlighted in Fig. 1. Each process represented the main steps to generate a bipartite network, which were used to identify the hotspots of dengue. All the required data will be returned by calling RESTful API and stored in a database in the same way. Details of these four processes are explained in the model implementation section by showing the algorithms of these processes. Process 7 corresponds to the sub-module III, as previously explained. The heatmap is implemented at the front-end of MozzHub wherein the required data will be returned from the server side and generate a heatmap on the client side.

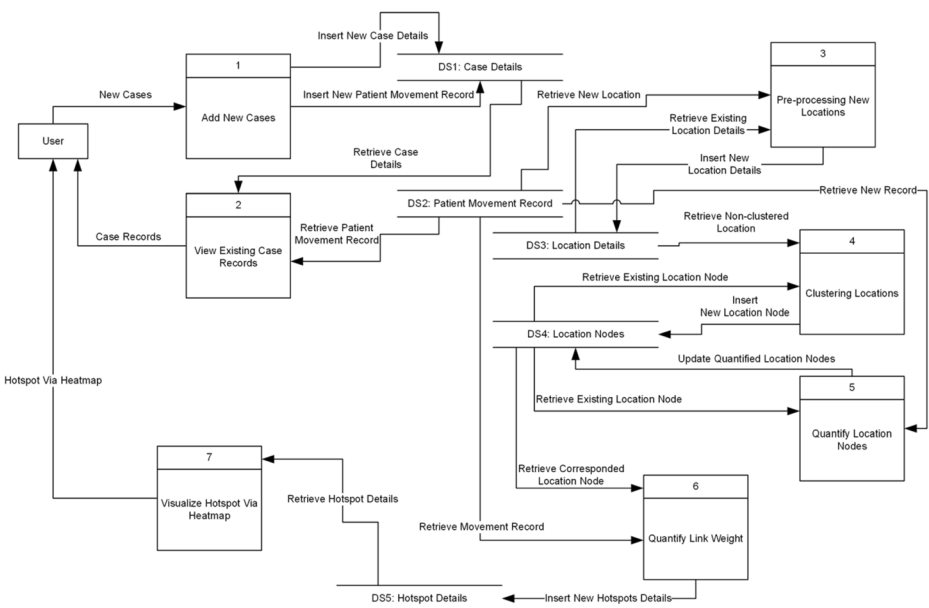


Fig. 3 Level-1 data flow diagram of MozzHub

2.4 System development and implementation

This section is divided into two parts, wherein the first part explains the development of each function that appears in the MozzHub user interface, and in the second, the details of the model implementation of MozzHub integrating all the sub-modules are presented.

2.4.1 System functionalities development

MozzHub was developed using React JavaScript library and React Semantic UI for ease of user interface (UI) development. Figure 4 shows the user authentication section of MozzHub, wherein the system can only be accessed by authenticated users. Users are required to log in to the system using an assigned ID and password.

The overview of the MozzHub dashboard is shown in Fig. 5. To identify the possible source of infection of dengue and take action as early as possible, the district health officer or public health authority as the user can obtain the latest information displayed on the dashboard. The name of the health officer is displayed in the top-left section and followed by a list of the potential sources of infection generated by MozzHub below it. If the distance between two locations is <400 m, both of the locations will be clustered into the same location node. If the location node consists of more than one location, there will be more than one address locations listed. The total cases indicate the number of dengue cases that were used to formulate the network for identifying the dengue hotspot. The date is mentioned when the network has been formulated.

The heatmap shown in Fig. 6 is one of the abovementioned sub-modules. In MozzHub, the heatmap is rendered on the client side and supported by the Leaflet, which is an open-source JavaScript library for interactive maps. The red circles in the map indicate the locations of the dengue hotspots identified by the system, with higher color intensity indicating higher priority for dengue control action. The color intensity of the circle is obtained from the following eq. (1):

$$intensity = \text{floor}(255 - \text{Location_Ranking}_i \times 255) \quad (1)$$

where i is the ranking of the i th location. The color scheme used in this system is based on the RGB scheme. Thus, the value of r is fixed at 255, and the values of g and b depend on

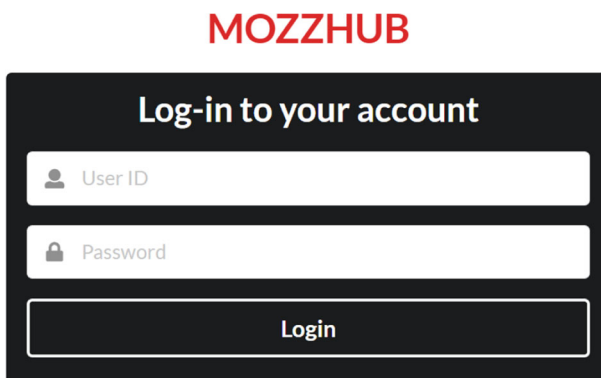


Fig. 4 MozzHub: User authentication interface

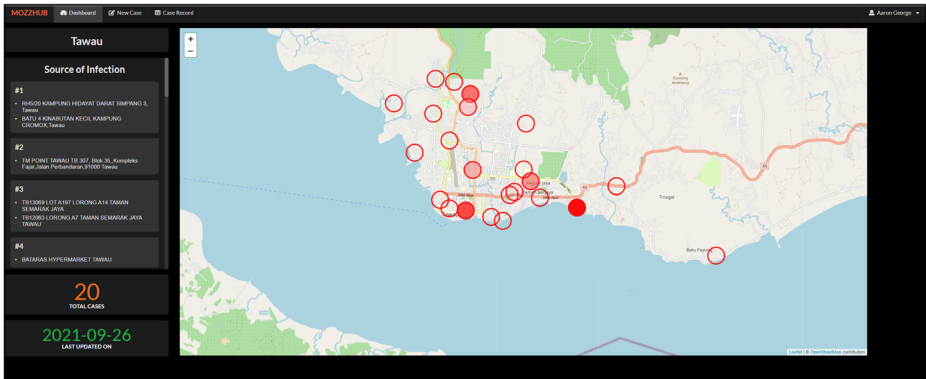


Fig. 5 MozzHub: Overview of the dashboard

the value of the intensity that is computed based on eq. (1). When the user hovers across or clicks the circle, the ranking and the coordinates of the location node will pop up.

The MozzHub interface for entering a new case is presented in Fig. 7. The interface was designed based on the format of the dengue case investigation form currently used by public health authorities for recording the case details, patient information, symptoms of the patient, and the movement of the patient in the past 14 days from the date of onset. The case ID is assigned by the system and will serve as a unique ID that corresponds to the human node ID in the network [23].

The MozzHub interface for inputting patient information (left) and notification information (right) is shown in Fig. 8. Residential and other addresses (e.g., school and workplace) will be used as an option for quick input in the patient movement record section. As it will be used to generate the dates of the last 14 days from the onset date in the patient movement record, the onset date column in the notification is one of several compulsory inputs.

The MozzHub interface for keying in the patient’s past 14-day movement record is depicted in Fig. 9. As those records will be stored and used as input for the location node quantification

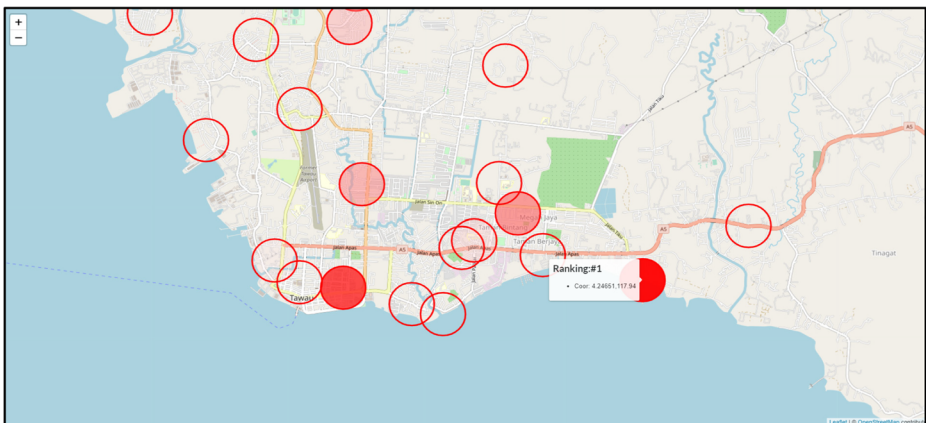


Fig. 6 MozzHub: Visualization of the source of infection via heatmap

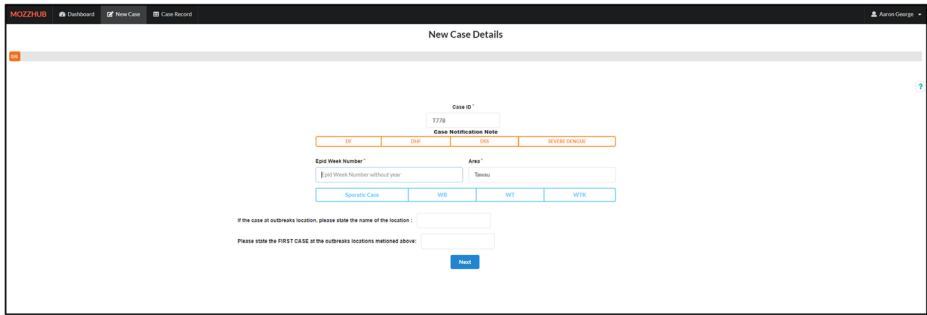


Fig. 7 MozzHub: New case entry I

process and link weight quantification process to generate the ranking of the hotspot, the 14-day movement history requires to be recorded. The dates from day 1 to 14 are generated by the system, and the start and end times are used to calculate the duration/length of time the patient was in the particular location. The duration will be used as a parameter in the human node of the bipartite network.

The case ID and patient name are displayed at the top of the page to indicate which patient and case are being referred to. In the real world, a patient may visit more than one location on the same day; therefore, it is not realistic if the form provides only one row for each day record. Thus, the interface of the patient movement record was designed such that the user is able to add additional rows for each day or reduce the number of rows by clicking the PLUS and MINUS buttons. As previously mentioned, the residential and other addresses in the previous section are displayed at the top of the form and can be used as a shortcut to enable the user to select it instead of repeatedly keying in the data.

The case review section of MozzHub is illustrated in Fig. 10. All the cases submitted by the user will be listed in a table by case ID. In the table, a green tick symbol in the mobility column indicates that there is mobility data attached to the case ID; otherwise, a red cross symbol will be shown. The user is allowed to edit the data, including the details of patient movement records, if the particular case has not yet been formulated into the network. A green tick symbol in the Editable column means the user is allowed to edit the data, whereas a red cross symbol signifies edits are not allowed.

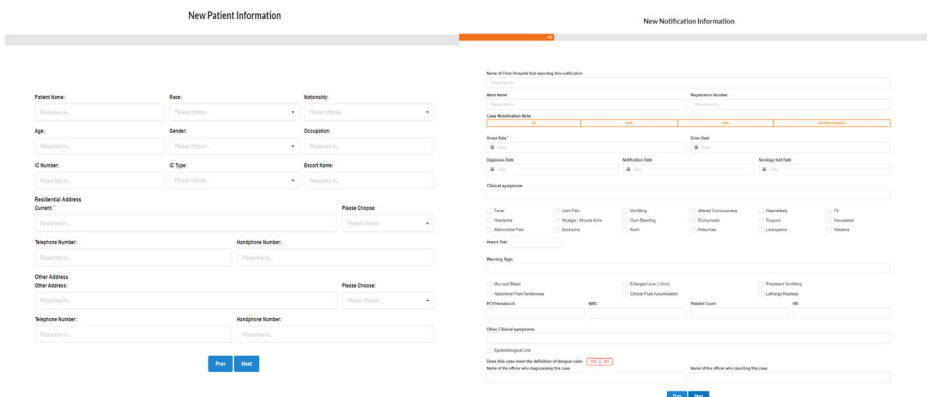


Fig. 8 MozzHub: New patient information (left) and notification information (right)

Patient Movement Record

Case ID:	Onset Date:	House Address:	Other Address:
<input type="text" value="11"/>	<input type="text" value="2021-06-24"/>	<input type="text" value="Kota Samarahan"/>	<input type="text"/>

DAY 1 (2021-06-10)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 2 (2021-06-11)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 3 (2021-06-12)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 4 (2021-06-13)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 5 (2021-06-14)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 6 (2021-06-15)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 7 (2021-06-16)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 8 (2021-06-17)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY 9 (2021-06-18)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY10 (2021-06-19)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY11 (2021-06-20)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY12 (2021-06-21)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY13 (2021-06-22)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>
DAY14 (2021-06-23)	Start Time	HH:MM Example 1430	End Time	HH:MM Example 1500	Address	Location	<input type="button" value="+"/>	<input type="button" value="-"/>

Fig. 9 MozzHub: Patient movement record

2.4.2 Model implementation

One significant aspect of the model implementation is the automation of the dengue bipartite network formulation process [23]. The abovementioned sub-modules were rewritten to integrate the script with the server side via RESTful API (prefixed with /). The following four main scripts were implemented: data preprocessing of the new location, location clustering, location node parameter quantification, and bipartite

No.	Case ID	Mobility	Editable	Report
1	T1113	✓	✗	<input type="button" value="CL view"/>
2	T1162	✓	✗	<input type="button" value="CL view"/>
3	T1233	✓	✗	<input type="button" value="CL view"/>
4	T1295	✓	✗	<input type="button" value="CL view"/>
5	T1365	✓	✗	<input type="button" value="CL view"/>
6	T1369	✓	✗	<input type="button" value="CL view"/>
7	T1398	✓	✗	<input type="button" value="CL view"/>
8	T1408	✓	✗	<input type="button" value="CL view"/>
9	T1578	✓	✗	<input type="button" value="CL view"/>
10	T355	✓	✗	<input type="button" value="CL view"/>

Fig. 10 MozzHub: Case review interface

network formulation. All of the scripts were written in the R language in the following algorithm:

Algorithm 1: Preprocessing new location

INPUT: new mobility dataset from /unprocess, existing geocoded location dataset from /getlocationlist
 OUTPUT:
 START

1. Request the new mobility data from the database via /unprocess and set it to Loc_{new}
2. Check the length of Loc_{new} . If 0, exit and end the script; otherwise, continue
3. Request the existing geocoded location from the database via /getlocationlist and set it to Loc_{exist}
4. Compare Loc_{exist} and Loc_{new} , remove duplicate values, and set to Loc_{new}
5. Calling the google_geocoding function by passing Loc_{new} as input and setting the return value to Loc_{new}
6. Save Loc_{new} to the database and set the status of the location to 0 via /geocoded
7. Set the status of the mobility data to /status

END

Algorithm 1 delineates the steps in data preprocessing of new locations in MozzHub. The input of the process is requested from the database via RESTful API. API /unprocess requests the database to retrieve the patient movement record that is new in the database and indicated by the attribute “status.” If the status is 0, this indicates that the particular record is new, and data preprocessing requires to be performed. API /getlocationlist requests the existing geocoded location from the database and compares both data to remove duplicated elements. Removing duplicated elements helps to reduce the number of calls of Google Map API. After the location has been geocoded, it is stored in data via API /geocoded using the POST method. The status of the new patient movement records is subsequently set to 1, indicating that it has been processed.

Algorithm 2: Clustering location

INPUT: Existing geocoded location wherein status = 0, existing location node
 OUTPUT: List of new location nodes with child location
 START

1. Request the existing geocoded location from the database via /uncluster and set it to $Loc_{uncluster}$
2. Check the length of $Loc_{uncluster}$. If 0, exit and end the script; otherwise, continue
3. Request the Existing location node from the database via /getLocationNode and set it to Loc_{node}
4. Insert $Loc_{uncluster}[1]$ to $Loc_{node}[h+1]$, where h is the length of Loc_{node}
5. Calling distance_matrix function by passing Loc_{node} . Set the return value to Loc_{node}
6. Delete $Loc_{uncluster}[1]$
7. Repeat steps 4–5 until the length of $Loc_{uncluster} = 0$
8. Save the Loc_{node} to database via /clustered

END

Algorithm 2 outlines the location clustering process. The data requested via API /uncluster will return the list of the location that has not yet been clustered and assigned to $Loc_{uncluster}$. The $Loc_{uncluster}$ includes the location ID, LatLng, and elevation of the location. The list of existing location nodes will be returned by calling API /getLocationNode and assigned to Loc_{node} . The list consists of the location node ID, latitude and longitude of the location node, the elevation of the location node, and followed by the location ID that has been clustered into the node. The top of the element in the list of $Loc_{uncluster}$ will be appended to the last row of Loc_{node} . By comparing the last row of the matrix, the Loc_{node} will be inputted to the distance_matrix function. If the value is <400 with one of the cells in the matrix, the location will be clustered to the nearest location node and deleted from $Loc_{uncluster}$ and Loc_{node} . Otherwise, the location is kept in Loc_{node} , and the process is repeated until all the locations

listed in $Loc_{incluster}$ have been clustered. The final Loc_{node} will be posted to the database, and the status of the location will be set to 1 via /clustered.

Algorithm 3: Location node's parameter quantification

INPUT: /graph, /getLocationNode, area

OUTPUT: List of quantified location node's parameters

START

1. Initial temp, humi, pre-variables; loc_node_params as dataframe
2. Request the data from the database via /graph by passing area and 1 as query and set to Mob_{new}
3. Check the length of Mob_{new}. If 0, exit and end the script; otherwise, continue
4. Request the existing location node from the database via /getLocationNode and set it to Loc_{node}
5. Obtain the total visit date of the location node in Mob_{new} and set it to date_{visit}
6. Calling the weather API to request the data by passing LatLng, date_{visit} -14, and date_{visit} as a query.
7. Compute the average value of temperature, humidity, and precipitation from the return values of the weather API
8. Set the average value of temperature, humidity, and precipitation to temp, humi, and pre, respectively
9. Compute the value of Lc, S, and B by passing temp as input. All the values are assigned to Lc, S, and B, respectively
10. Compute the value Fl and assigned it to Fl
11. Append the location node, temp, humi, pre, elevation, Lc, S, B, and Fl to loc_node_params
12. Repeat steps 5–10 until all the location nodes listed in Mob_{new} are quantified
13. Save loc_node_params to the database via /quantified

END

In this study, the parameters implemented in this model [23] are *H*, *Pre*, *Lc*, *S*, *B* of *Aedes* mosquito, *Al*, and *Fl* for the location node; *Fh* and *Du* are the parameters for the human node. Algorithm 3 lists the steps of location node parameter quantification. The data returned by /graph consist of the location node in the area that has not yet been quantified. To quantify the location node, the environmental properties 14 days before the visit date to the location need to be calculated. The weather API used in MozzHub is WordWeatherOnline API, the query needs to include the LatLng and the start and end dates to request the average value for the period. The environmental properties of a location visited will be kept in the database until a control measure is acted on the location. If there is no control measure taken on a particular location node and the same locality is being visited by another patient, then current environmental property values are added to the previous ones, and the average values will be stored.

Algorithm 4: Bipartite network formulation.

INPUT: Area, /graph, /quantifiedLocNode

OUTPUT: Hotspot list for the area

START

1. Request the data from the database via /graph by passing area and 2 as query and set to raw_{graph}
 2. Compute the duration by subtracting the raw_{graph}\$end_time and raw_{graph}\$start_time and set the Dur
 3. Compute the Fh in raw_{graph} by summarizing all the common locations with particular human node ID
 4. Create a dataframe named human_node consisting of human_node_ID, visited_location_node, Fh, and Du
 5. Normalize the value of Fh and Du
 6. Request the data from the database via /quantifiedLocNode and set it to quantified_location_node
 7. Select the location node's parameter value of human_node\$visited_location_node from quantified_location_node and set to location_node
 8. Normalize all the numeric values in location_node
 9. Extract the unique human_node_ID and set it to col
 10. Extract the unique location_node_ID and set it to row
-

The flow of the last script that covers human node quantification, link weight quantification, and HITS algorithm implementation is listed in algorithm 4. Passing the area and 2 to /graph, the relationship table of the human and location nodes will be returned for the

particular area. This returned value is set to raw_{graph} , which is the input to formulate the bipartite graph of the area. Human node quantification includes Du (step 2) and Fh (step 3). The quantified location nodes will be returned by $/quantifiedLocNode$ and will be matched with the location node that appears in raw_{graph} . The parameter values will be normalized (steps 5 and 8) to ensure all the values are within a range from 0 to 0.9.

A contact matrix that represents the bipartite graph will be formed by assigning the number of human nodes as the number of columns and the number of location nodes as the number of rows. To convert the bipartite graph to a bipartite network, the link weight value will be assigned to each cell (step 12). The bipartite network will be inputted to the HITS function to generate the ranking of the location and human nodes. The primary aim of MozzHub is to identify the hotspot of dengue; therefore, only the ranking of location nodes will be returned from the HITS function (step 17). The ranking of location nodes will act as a hotspot and be stored in the database via $/ranking$.

2.5 System testing

Two main tests were performed in this phase, including functional testing to ensure all components in MozzHub work as expected and user testing to evaluate the usability of the system by a sample of target users.

2.5.1 Functional testing

In this testing, the main interface of the integrated system was analyzed to ensure all the data inputted were captured by the system. Functional testing was performed with the following three test objectives: 1) to verify the functioning of the heatmap display, 2) verify the proper functioning of new case entry case details, and 3) verify the proper functioning of case movement records entry. Results of the functional testing by test objective are shown in Tables 1, 2 and 3.

2.5.2 UAT

Study design and participants This was a cross-sectional pilot study conducted from August to December 2021 to examine the user acceptance of the MozzHub system. Participants were

Table 1 Test plan of dashboard heatmap

Test objective: examine dashboard component: heatmap				
Test case	Input data	Expected result	Actual result	Pass/Fail
The system loaded when there is an Internet connection	When the page loaded	The map displays success, and the heatmap is rendered in the map. All tiles are rendered when zooming in or out on the map.	The same as the expected result.	Pass
The system loaded when there is no Internet connection	When the page loaded	The map displays success, and the heatmap is missing. Some tiles are missing when zooming in or out on the map.	The same as the expected result.	Pass

Table 2 Test plan of new case enter (1)

Test objective: verify the new case enter function: case details				
Test case	Input data	Expected result	Actual result	Pass/Fail
All the required columns are entered	onClick=DF and WB Epid Week Number=30 Area=Kota Samarahan	After clicking the SAVE button, no message pops up.	The same as the expected result.	Pass
Some required columns are empty	onClick=DF and WB Area=Kota Samarahan	After clicking the SAVE button, the warning message pops up.	The same as the expected result.	Pass

recruited from seven district health offices in three states in Malaysia as in Table 4 and given access to the MozzHub system. These districts were selected after discussions with the Vector Control Division, Ministry of Health Malaysia and the system developers, wherein the selection was based on the fulfillment of the following criteria: low dengue burden, absence of concurrent dengue research activities in the district, having sufficient computers with Internet connection for accessing the system, and low burden of COVID-19 cases. These criteria would enable the staff at the district health office to use the MozzHub system and provide sufficient responses for the study. The participants recruited for the user acceptance test were designated users of the MozzHub system at the respective district health offices, consisting of key officers involved in managing dengue surveillance, including the dengue control officer and the dengue control environmental health officers. For the model to generate the probable source of infection in a single locality within the district, these officers were required to key in a minimum of 20 dengue cases into the MozzHub system, and they were subsequently asked to complete a user acceptance questionnaire for the MozzHub system.

Study instrument This study is a 16-item self-administered questionnaire based on a 5-point Likert scale response options, ranging from 1-Strongly Disagree, 2-Disagree, 3-Neutral, 4-Agree, and 5-Strongly Agree. The questionnaire is designed by referring to the concept of the

Table 3 Test plan of new case enter (2)

Test objective: verify the new case enter function: patient movement record				
Test case	Input data	Expected result	Actual result	Pass/Fail
Generate more input field	“PLUS” button click=TRUE	After clicking the PLUS button, the input field is generated for a specific day, and the MINUS button is activated for the generated input field.	The same as the expected result.	Pass
Remove the redundant input field	First field=FALSE “MINUS” button click=TRUE	After clicking the MINUS button, the redundant input field is removed.	The same as the expected result.	Pass
Save the input	Save button click=TRUE Next button click=TRUE	After clicking the SAVE button, the NEXT button is clicked, and the system will pop up a complete message for the user.	The same as the expected result.	Pass

Table 4 Distribution of participants (n = 31)

State (Districts)	n	%
Terengganu		
PKD Kuala Terengganu	6	19
PKD Marang	5	16
Johor		
PKD Kulai	6	19
PKD Tangkak	4	13
Sabah		
PKD Kota Kinabalu	5	16
PKD Tawau	5	16

technology acceptance model (TAM) that was created by Davis (1985) [9]. TAM is a theory to study how the user accepts the technology and how they use it [9]. In TAM, the following were the two major cognitive factors raised to measure user acceptance: 1) Perceived of Usefulness (PU) and 2) Perceived Ease of Use (PE). PU is concerned about how the system improves the productivity and effectiveness of user performance, whereas PE is related to how users are able to easily access the system and its display. Thus, the questionnaire consisted of the following three domains that assess user acceptance of the system: 1) usefulness of the system (8 items), 2) ease of use (4 items), and 3) user satisfaction (4 items). The eight items in the first domain are designed based on PU, whereas the four items in the second domain are designed based on PE in TAM. Based on TAM, user satisfaction will come after the cognitive factors of a user (PU and PE) toward the system. Thus, the third domain of the questionnaire is to collect the overall experience of the user after accessing the system.

A forward-backward translation of the questionnaire was undertaken to ensure consistency [36]. The questionnaire, which was initially in English, was translated into Bahasa Malaysia and back-translated into English by four Malaysian translators with similar educational backgrounds and similar command of English and Bahasa Malaysia. Variations in the original and translated versions were discussed and resolved by a joint agreement involving all four translators. The final questionnaire was presented in a dual-language format (English and Bahasa Malaysia) and distributed to the participants using Google Forms.

2.6 Statistical analysis

The internal reliability of the questionnaire was examined using Cronbach's alpha wherein the alpha value was reported at 0.984, indicating good internal consistency of the questionnaire. Removal of any of the items did not increase the Cronbach's alpha value by more than 0.002 (Appendix 1); therefore, all items were retained. For ease of analysis, the initial 5-point Likert scale response was reduced to the following 3-point response: agree (strongly agree and agree), neutral, and disagree (strongly disagree and disagree). Descriptive statistical analysis for the user testing was performed by calculating the percentage and frequency of responses for individual items and each domain. The scores for all items in each domain were averaged to obtain domain-wise agreement wherein average scores of more than 3, less than 3, and equals

3 were interpreted as “Agree,” “Disagree,” and “Neutral,” respectively. All analyses were performed using SPSS software version 24.

3 Results

UAT A total of 31 participants from the six districts completed the user acceptance questionnaire [17] (Table 4). Each district had almost similar numbers of respondents with an average of five respondents per district with proportions ranging from 13% to 19%.

For the first domain, which is the usefulness of the system, individual item agreement ranged from 61.3% (“It saves me time when I use”) to 77.4% (“It is useful”). Domain-wise, 77.4%, 16.1%, and 6.5% of the participants agreed, disagreed, and were neutral, respectively, regarding the items on the usefulness of the system. For the second domain (ease of use), agreement for individual items ranged from 67.7% (“I can use it without written instruction”) to 74.2% (“It requires the fewest steps possible to accomplish what I want to do with”). Domain-wise, 80.6%, 9.7%, and 9.7% of the participants agreed, disagreed, and were neutral, respectively, regarding the items relating to the system’s ease of use. In the third domain (user satisfaction), agreement for individual items ranged from 58.1% (“I feel I need to have it”) to 71.0% (“I am satisfied with it”). Domain-wise, 74.2%, 12.9%, and 12.9% of the participants indicated satisfaction, disagreed, and were neutral with the system, respectively. Overall, the highest agreement was reported in the ease of use (80.6%) followed by the usefulness of the system (77.4%) and user satisfaction (74.2%) domain. The user acceptance questionnaire responses for individual items within each domain and the overall domain are presented in Table 5.

Table 5 User acceptance questionnaire responses

Item	Agree % (n)	Neutral % (n)	Disagree % (n)
Usefulness of the system			
QU1 It helps me be more effective	77.4 (24) *	6.5 (2) *	16.1 (5) *
QU2 It helps me be more productive	74.2 (23)	16.1 (5)	9.7 (3)
QU3 It is useful	74.2 (23)	16.1 (5)	9.7 (3)
QU4 It makes the things I want to accomplish easier to get done	77.4 (24)	12.9 (4)	9.7 (3)
QU5 It saves me time when I use	74.2 (23)	16.1 (5)	9.7 (3)
QU6 It meets my needs	61.3 (19)	29.0 (9)	9.7 (3)
QU7 The source of infection generated from the system is accurate to the real world	67.7 (21)	22.6 (7)	9.7 (3)
QU8 It does everything I would expect it to do	64.6 (20)	19.4 (6)	16.1 (5)
Ease of use			
QE1 It is easy and simple to use	80.6 (25) *	9.7 (3) *	9.7 (3) *
QE2 It requires the fewest steps possible to accomplish what I want to do with	71.0 (22)	22.6 (7)	6.5 (2)
QE3 I can use it without written instruction	74.2 (23)	16.1 (5)	9.7 (3)
QE4 I can use it successfully every time	67.8 (21)	25.8 (8)	6.5 (2)
User satisfaction			
QS1 I am satisfied with it	71.0 (22)	19.4 (6)	9.7 (3)
QS2 It works the way I want it to work	61.3 (19)	29.0 (9)	9.7 (3)
QS3 I feel I need to have it	58.0 (18)	29.0 (9)	12.9 (4)
QS4 It is pleasant to use	64.6 (20)	35.5 (8)	9.7 (3)

* Domain-wise agreement based on the average score

4 Discussion

To date, the identification of dengue hotspots as targets for instituting control measures has been based on conjectures. According to Hussain-Alkhateeb et al. [16], most of the existing EWSs are focused on temporal prediction for vector-borne diseases. Research works on spatial predictions for EWS are limited. Although few EWSs demonstrated their ability in spatial prediction, a large scale of data is required for those EWSs [13, 32] to establish stable prediction, and the areas of prediction are normally at a macro level [6, 7]. EWSs with microlevel spatial analysis are limited, which is essential for vector control measures [16]. The comparison between selected studies and MozzHub based on the three aspects is tabulated in Table 6.

As presented in Table 6, the studies [30, 32] did not consider patient movement effect, suggesting the absence of heterogeneity of humans in these studies. As previously mentioned, to identify the hotspot of dengue, the movement of the patient should not be neglected. Therefore, the MozzHub system has been developed as a potential solution for addressing this issue. MozzHub uses a bipartite network to identify the probable locations of dengue infection based on the 14-day history of dengue case movements. As aforementioned, BNM only requires a small amount of data to provide an efficient result for prediction; thus, it is the best-fit solution to integrate the 14-day history of dengue case movement data. Although the entomological data were not directly inputted to MozzHub, the parameters Lc , S , and B were quantified using the entomological data of the vector during the model formulation. Moreover, since the primary concern of the study was to forecast the dengue outbreaks instead of identifying dengue hotspots, the spatial component was absent in [32]. The study [30] involved a spatial component in the prediction. However, the spatial component is focused on the data collected in the selected districts and forecasts the trend of dengue for the given district only. Thus, it is not suitable to use for identifying dengue hotspots for implementing vector control measures. The probable locations of dengue infection identified by MozzHub were focused on a small spatial level (i.e., 400 m) to allow public health to utilize the resources in these areas for vector control measures.

The system is designed to automatically generate locations that may replace the currently used manual hotspot identification method. The only mandatory input of the system is the 14-day history of dengue case movements, and all the data preprocessing and analysis will be automatically performed by MozzHub, making it suitable for less-skilled or nontechnical users who can easily use the system without any modeling and statistical knowledge.

In this study, we reported on the usefulness, ease of use, and user satisfaction with the MozzHub system using a validated and reliable instrument. From this analysis, we found that a large majority of users were in agreement with the domains relating to the ease of use (80.6%), usefulness (77.4%), and user satisfaction (74.1%) of the MozzHub system. More specifically, users reported high agreement on items regarding the system being useful (usefulness domain),

Table 6 Comparison between selected studies and MozzHub

Study	Mobility of patient	Spatial component	User testing in Malaysia
[32]	No	No	No
[30]	No	No	No
[15]	No	Yes, district level	Yes
MozzHub	Yes	Yes, 400 m	Yes

the system requiring a few steps to accomplish the outcome (ease of use domain), and general satisfaction with the system (user satisfaction). The MozzHub system must achieve desirable agreement levels in the system user testing phase as this would ensure its use more efficiently and effectively.

Across each domain, several items scored lower agreement levels, including the items relating to saving time (usefulness domain), ability to use without written instruction (ease of use domain), and the need to have the system (user satisfaction). Several reasons could be attributed to these findings. Among the reasons that could result in a low agreement on the item relating to saving time are user-related factors, such as being first-time users, lack of familiarity with the system, and infrequent system use, thereby requiring more time to get familiarized and operate the system. Furthermore, there could be several system-related factors that could contribute to time-consuming factors, including the requirement for data entry into the system, such as mobility history, data reentry into the spreadsheets and the system, and redundancies in the spreadsheets. Additionally, the inability of users to use the system without written instructions could be due to their limited training. Furthermore, the low agreement on the need to have the system could be due to the presence of several existing dengue surveillance systems that the staff are already more familiar with. Moreover, it is possible that the users are not privy to the advantages and validity of the newly introduced MozzHub system.

Based on the UAT of the MozzHub system, it is evident that the system satisfies the criteria of being useful, easy-to-operate, and able to provide adequate satisfaction to its users. However, several improvements could be made at the system and user levels to further improve the user acceptance of the system. These include reducing redundancies in the data spreadsheet and minimizing data reentry to improve time constraints at the system level. To address the user-related factors, more training, supervision, and daily use of the system are required. Future validation studies for MozzHub are required to further ascertain its reliability and validity. The prediction power of the MozzHub system will be examined using the retrospective dengue cases collected earlier; furthermore, prospective dengue cases from 2022 onward will continuously be used for further validation analysis.

5 Conclusions

In this study, the design, implementation, and UAT of the MozzHub system has been demonstrated and documented. The MozzHub system is a bipartite network-based dengue hotspot detector that only requires the least amount of input to identify the potential source of dengue infection within a 400-m radius spatial size by factoring in both the effects of population mobility and environmental factors into the model. The MozzHub system was developed based on a previously validated bipartite dengue network model and is designed with a certain degree of automation to reduce the system complexity for end-users. Consequently, the MozzHub system appeals well to the end-users as it has been demonstrated to be a useful, easy-to-operate system along with having achieved adequate client satisfaction for its use. With the currently available evidence, the MozzHub system is a system that could greatly assist the MOH, Malaysia in the management and control of dengue as it would provide crucial information on the location of sources of infection, which should be the target area for instituting dengue control measures. Thus, it is also hoped that the MozzHub system will be expanded to all district health offices in Malaysia in the future.

Appendix 1

	Scale mean if item deleted	Scale variance if item deleted	Corrected item-total correlation	Cronbach's alpha if item deleted
1. It helps me be more effective	54.23	128.114	0.932	0.982
2. It helps me be more productive	54.23	128.314	0.920	0.982
3. It is useful	54.16	127.806	0.922	0.982
4. It makes the things I want to accomplish easier to get done	54.26	128.531	0.950	0.982
5. It saves me time when I use	54.35	128.970	0.863	0.983
6. It meets my needs	54.23	127.781	0.864	0.983
7. The source of infection generated from system is accurate to the real world	54.39	127.312	0.870	0.983
8. It does everything I would expect it to do	54.35	127.970	0.918	0.982
1. It is easy and simple to use	54.19	130.028	0.832	0.983
2. It requires the fewest steps possible to accomplish what I want to do with	54.23	128.847	0.890	0.982
3. I can use it without written instruction	54.23	130.447	0.801	0.984
4. I can use it successfully every time	54.26	129.065	0.871	0.983
1. I am satisfied with it	54.29	128.280	0.958	0.982
2. It works the way I want it to work	54.39	131.245	0.778	0.984
3. I feel I need to have it	54.45	128.389	0.895	0.982
4. It is pleasant to use	54.32	129.292	0.848	0.983

Acknowledgments The authors thank the Malaysia Ministry of Higher Education for the Prototype Research Grant Scheme PRGS/1/2020/ICT05/UNIMAS/02/1 awarded to J. Labadin.

Data availability The datasets generated during and/or analysed during the current study are not publicly available due to request from Malaysia Ministry of Health (MoH) but are available from the corresponding author upon approval from the MoH.

Declarations

Conflict of interests We declared there is no conflict of interest in terms with this submission.

References

1. Ali A, Zhu Y, Zakarya M (2021) A data aggregation based approach to exploit dynamic spatio-temporal correlations for citywide crowd flows prediction in fog computing. *Multimed Tools Appl* 80(20):31401–31433
2. Ali A, Zhu Y, Zakarya M (2021) Exploiting dynamic spatio-temporal correlations for citywide traffic flow prediction using attention based neural networks. *Inf Sci* 577:852–870
3. Alzahrani T, Horadam KJ (2014) *Analysis of two crime-related networks derived from bipartite social networks*. In: 2014 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2014). IEEE
4. Alzahrani T, Horadam KJ, Boztas S (2014) Community detection in bipartite networks using random walks. In: *Studies in Computational Intelligence* Springer, 2014. p. 157–165
5. Ambati LS, El-Gayar O (2021) Human activity recognition: a comparison of machine learning approaches. *J Midwest Assoc Inf Syst (JMWAIS)* 2021(1):49


6. Anno S, Hara T, Kai H, Lee MA, Chang Y, Oyoshi K, Mizukami Y, Tadono T (2019) Spatiotemporal dengue fever hotspots associated with climatic factors in Taiwan including outbreak predictions based on machine-learning. *Geospat Health* 14(2). <https://doi.org/10.4081/gh.2019.771>
7. Bowman LR, Tejada GS, Coelho GE, Sulaiman LH, Gill BS, McCall PJ, Olliaro PL, Ranzinger SR, Quang LC, Ramm RS, Kroeger A, Petzold MG (2016) Alarm variables for dengue outbreaks: a multi-Centre study in Asia and Latin America. *PLoS One* 11(6):e0157971
8. Bujang MA, Mudin RN, Haniff J, Ikhwan T, Sidik T, Nordin NM, Sulaiman N, Md-Sani SS, Kadir SSA, Mohammed S, Pin GP (2017) Trend of dengue infection in Malaysia and the forecast up until year 2040. *Intern Med J* 24(6):438–441
9. Davis FD (1985) *A technology acceptance model for empirically testing new end-user information systems: theory and results*. Massachusetts Institute of Technology
10. Eze M, Labadin J, Lim T (2011) *Emerging computational strategy for eradication of malaria*. In: IEEE symposium on computers and informatics. IEEE
11. Eze M, Labadin J, Lim T (2011) *Contact strength generating algorithm for application in malaria transmission network*. In: 2011 7th international conference on information Technology in Asia. IEEE
12. Ge M, Li A, Wang M (2016) A bipartite network-based method for prediction of long non-coding RNA–protein interactions. *Genom Proteom Bioinform* 14(1):62–71
13. Guo P, Liu T, Zhang Q, Wang L, Xiao J, Zhang Q, Luo G, Li Z, He J, Zhang Y, Ma W (2017) Developing a dengue forecast model using machine learning: a case study in China. *PLoS Negl Trop Dis* 11(10): e0005973
14. Hong BH, Labadin J, Tiong WK, Lim T, Chung MHL (2021) Modeling COVID-19 hotspot using bipartite network approach
15. Hussain-Alkhateeb L, Kroeger A, Olliaro P, Rocklöv J, Sewe MO, Tejada G, Benitez D, Gill B, Hakim SL, Gomes Carvalho R, Bowman L, Petzold M (2018) Early warning and response system (EWARS) for dengue outbreaks: recent advancements towards widespread applications in critical settings. *PLoS One* 13(5):e0196811
16. Hussain-Alkhateeb L, Rivera Ramirez T, Kroeger A, Gozzer E, Runge-Ranzinger S (2021) Early warning systems (EWSs) for chikungunya, dengue, malaria, yellow fever, and Zika outbreaks: what is the evidence? A scoping review. *PLoS Negl Trop Dis* 15(9):e0009686
17. Hwang W, Salvendy G (2010) Number of people required for usability evaluation: the 10±2 rule. *Commun ACM* 53(5):130–133
18. Isah H, Neagu D, Trundle P (2015) *Bipartite network model for inferring hidden ties in crime data*. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in SocialN Analysis and Mining
19. Jain R, Sontisirikit S, Iamsirithaworn S, Prendinger H (2019) Prediction of dengue outbreaks based on disease surveillance, meteorological and socio-economic data. *BMC Infect Dis* 19(1):272
20. Jayaraj VJ, Avoi R, Gopalakrishnan N, Raja DB, Umasa Y (2019) Developing a dengue prediction model based on climate in Tawau, Malaysia. *Acta Trop* 197:105055
21. Kleinberg JM (1999) Hubs, authorities, and communities. *ACM Comput Surv* 31(4es):5
22. Kok WC, Labadin J (2019) Validation of bipartite network model of dengue hotspot detection in Sarawak. *J Comput Sci Technol*. Singapore: Springer Singapore:335–345
23. Kok WC, Labadin J, Perera D (n.d.) *Modeling dengue hotspot with bipartite network approach*. In: 2017 International conference on computational science and technology. Springer.
24. Lee J, Lee J (2013) Hidden information revealed by optimal community structure from a protein-complex bipartite network improves protein function prediction. *PLoS One* 8(4):e00372. <https://doi.org/10.1371/journal.pone.0060372>
25. Liljeros F, Edling CR, Amaral LAN, Stanley HE, Åberg Y (2001) The web of human sexual contacts. *Nature* 411(6840):907–908
26. Ministry of Health Malaysia (2020) *Press Statement of the Director General of Health Malaysia: Current situation of dengue fever, Zika and chikungunya in Malaysia (For Week 51/2020 from 13 December to 19 December 2020)*. Putrajaya: disease control division ministry of health. Malaysia
27. Mohd-Zaki AH, Brett J, Ismail E, L’Azou M (2014) Epidemiology of dengue disease in Malaysia (2000–2012): a systematic literature review. *PLoS Negl Trop Dis* 8(11):e3159
28. Packierisamy PR, Ng CW, Dahlui M, Inbaraj J, Balan VK, Halasa YA, Shepard DS (2015) Cost of dengue vector control activities in Malaysia. *Am J Trop Med Hyg* 93(5):1020–1027
29. Rozilawati H, Lee HL, Mohd Masri S, Mohd Noor I, Rosman S (2005) Field bioefficacy of deltamethrin residual spraying against dengue vectors. *Trop Biomed* 22(2):143–148. <https://pubmed.ncbi.nlm.nih.gov/16883280/>

30. Salim NAM, Wah YB, Reeves C, Smith M, Yaacob WFW, Mudin RN, Dapari R, Sapri NNFF, Haque U (2021) Prediction of dengue outbreak in Selangor Malaysia using machine learning techniques. *Sci Rep* 11(1):939
31. Shaffiee Haghshenas S, Pirouz B, Haghshenas SS, Pirouz B, Piro P, Na K-S, Cho S, Geem ZW (2020) Prioritizing and analyzing the role of climate and urban parameters in the confirmed cases of COVID-19 based on artificial intelligence applications. *Int J Environ Res Public Health* 17(10):3730
32. Shi Y, Liu X, Kok SY, Rajarethinam J, Liang S, Yap G, Chong CS, Lee KS, Tan SS, Chin CK, Lo A, Kong W, Ng LC, Cook AR (2016) Three-month real-time dengue forecast models: an early warning system for outbreak alerts and policy decision support in Singapore. *Environ Health Perspect* 124(9):1369–1375
33. Siswanto S, Risva R, Marlina N (2019) Epidemiology forecasting analysis of dengue haemorrhagic fever with seasonal autoregressive integrated moving average in tropical area. *Public Health of Indonesia* 5(2):38–47. <https://doi.org/10.36685/phi.v5i2.261>
34. Stoddard ST, Morrison AC, Vazquez-Prokopec GM, Paz Soldan VP, Kochel TJ, Kitron U, Elder JP, Scott TW (2009) The role of human movement in the transmission of vector-borne pathogens. *PLoS Negl Trop Dis* 3(7):e481
35. Stoddard ST, Forshey BM, Morrison AC, Paz-Soldan VA, Vazquez-Prokopec GM, Astete H, Reiner RC, Vilcarrromero S, Elder JP, Halsey ES, Kochel TJ, Kitron U, Scott TW (2013) House-to-house human movement drives dengue virus transmission. *Proc Natl Acad Sci U S A* 110(3):994–999
36. Su TA, Hoe VCW (2008) Reliability of a Malay-translated questionnaire for use in a hand-arm vibration syndrome study in Malaysia. *Singap Med J* 49(12):1038–1045
37. Ying LC, Labadin J, Chai WY, Tuen AA, Peter C (2015) Applying bipartite network approach to scarce data: modeling habitat suitability of a marine mammal species. *Procedia Comput Sci* 60:266–275
38. Zohuri B, Moghaddam M (2020) Deep learning limitations and flaws. *Mod Approaches Mater Sci* 2(3): 241–250

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Affiliations

Jane Labadin¹  • Boon Hao Hong¹ • Wei King Tiong¹ • Balvinder Singh Gill² • David Perera³ • Andrew Ragai Henry Rigit⁴ • Sarbhan Singh² • Cia Vei Tan² • Sumarni Mohd Ghazali² • Jenarun Jelip⁵ • Norhayati Mokhtar⁵ • Norafidah binti Abdul Rashid⁵ • Hazlin Bt Abu Bakar⁵ • Jyh Hann Lim⁵ • Norsyahida Md Taib⁵ • Aaron George⁵

¹ Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Malaysia

² Institute for Medical Research, Ministry of Health, Kuala Lumpur, Malaysia

³ Institute for Health and Community Medicine, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Malaysia

⁴ Faculty of Engineering, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Malaysia

⁵ Ministry of Health, Putrajaya, Malaysia