Design and implementation of java-based community epidemic management system

Junjie Hao, Pu Wang, Di Jiao, Huixin Wang

School of Computer Science, Qufu Normal University, Rizhao, 276800, China

Abstract: *During the Newcastle pneumonia epidemic, the traditional community epidemic prevention and control work is cumbersome, backward record means and low efficiency. The functions of the system include registration of community residents' body temperature, practical updating of community epidemic information and registration of people entering and leaving the community. The system test shows that the system can reduce the workload of community workers, complete the collection of health conditions, improve the efficiency of the community, and has good prospects for application.*

Keywords: *Smart Community; B/S; Development Model; Management System*

1. Introduction

The new pneumonia epidemic is the fastest spreading, most widespread and most difficult to prevent and control major public health emergencies that have occurred since the founding of New China, posing a serious challenge to public health security in China and the world. The community is not only the front line of joint prevention and control of the epidemic, but also the most effective first line of defense against external importation and internal spread. In the wake of the epidemic, strict community grid management has been implemented nationwide to effectively reduce human contact, but the closed management has brought various problems along with the convenience of prevention and control. With a wide range of communities and a large population base, it is a huge challenge for community managers to register people in and out of the community, collect information on community residents, and keep statistics and updates on the epidemic situation in the community. At the same time, unexpected situations such as missing persons put great pressure on the work of community managers. In order to relieve the pressure of community managers' work and improve the efficiency of community management, the system was born. The system adopts a unified data model, a standard code system, and a friendly user interface to promote the informationization of community governance services with a scientific work operation mechanism, providing a perfect solution in improving the efficiency and accuracy of community epidemic prevention and control, recording the entry and exit of outsiders in a timely manner, and providing residents with practical community epidemic information, and moving epidemic prevention and control and community management to a holistic, systematic, scientific, standardized, standardized, and orderly manner. systematic, scientific, standardized, standardized and orderly direction with good practicality [1].

The community outbreak management system is a system designed for the period of the epidemic. The system combines Web technology and B/S development mode to achieve network access at any time. The system provides two clients for community administrators and community residents, which is convenient for community workers to manage the community and for community residents to grasp the actual surrounding epidemic situation. The system interface is simple and easy to use, convenient to operate, and adopts B/S system architecture, which has the advantages of zero installation, zero maintenance, strong interactivity, easy to expand, and low development cost, and is especially suitable for the design and implementation of community epidemic management system.

2. System feasibility and requirements analysis

2.1 System feasibility analysis

(1) Economic feasibility

The system is simpler and more efficient than the traditional community outbreak prevention and

control work, which greatly reduces the human and material expenses and allows users to perform registration and management operations on the webpage. Therefore, the design and development of this system is economically feasible.

(2) Technical feasibility

The system is supported by B/S framework, and the project is developed in client-server-database mode. The front-end uses Bootstrap technology based on Html5 and Css3 to realize the page part of the website, which is one of the mainstream front-end frameworks and has the feature of simple, flexible and easy to start, and can be developed quickly for the front-end page. Front-end basic operations through Jquery, Javascript and Ajax technology to achieve, compatible with the majority of the current browser. The back-end is developed in client-server-database mode, with the presentation layer (UI) interacting with end-users, obtaining end-user requirements and displaying the corresponding pages, the business logic layer (BLL) processing data logically, and the data access layer (DAL) using the database to analyze and process data, and then feeding it back to the presentation layer. Therefore, the design and development of this system is technically feasible.

(3) Operational feasibility

The system is compatible with most of the current browsers and has a friendly and easy-to-use interface that allows users to understand the system functions and operate it accurately as soon as possible. The system is designed and developed to meet the needs of community epidemic work in a simple and easy-to-operate manner, and to promote and use the community epidemic management system in major communities in the future.

2.2 System requirements analysis

(1) User Needs Analysis

The system is mainly facing the community managers and residents, through the community personnel to conduct relevant research, and the results of the research valuable information to organize the system user needs analysis. The system needs to have a simple and beautiful interface, simple operation, no advertising, and high confidentiality of user data. Therefore, the system adopts a simple and beautiful design style, and the system is modularized to achieve uniformity, simplify the user operation process, improve the ease of use of the system, and do our best to meet the user requirements.

System users include community residents and administrators, and users with different roles have different authority functions.

For community resident users, they have the functions of inquiring about the epidemic situation in the community, inquiring about their own situation and modifying their basic information.

For community administrator users, they have the functions of inquiring the situation of residents in the community, modifying residents' information and registering outsiders.

(2) Functional Requirements Analysis

The community epidemic management system mainly serves community administrators and community residents, so the clients are divided into two, namely the resident side and the administrator side. During the epidemic period, the community is under closed management, and the information of resident personnel needs to be entered in advance. If there is an unregistered person entering the community to live, the person can be added to the system after confirmation, and if someone needs to leave the community for a long time for some reason, the resident can also be deleted from the system so as to achieve accurate information collection. The administrator side can directly realize the operation of body temperature registration, community personnel entry and exit records and query related personnel information through the interface. At the same time, residents can also check their status in the last three days or perform operations such as community epidemic information search in order to understand the surrounding situation and take appropriate preventive measures in a timely manner.

The system functions can be divided into the following five areas, including:

a. Community resident information input function

The module contains the resident's name, residence details, phone number, daily temperature, personal status, and number of days in isolation.

b. Community mobility input function

The module contains the name of the person, the person's ID number, telephone number, body temperature and the time of entering and leaving the community, etc. The administrator can register the movement of people in the community in this module.

c. Community resident information search function

The module contains the resident's name, phone number, daily temperature and personal status, etc. The administrator can make inquiries about the community residents' information in this module.

d. Community epidemic status inquiry function

The module contains the community building number, unit number, suspected confirmed diagnosis and the number of confirmed diagnoses.

e. Resident personal status inquiry and modification function

The contents of the module include information such as temperature for the last three days and change of account password, etc. Residents can inquire and modify their personal information in this module.

(3) Performance Requirements Analysis

The system is developed to solve the community epidemic management problem. The primary feature of the system is to be simple, easy to use and convenient to operate for the community residents and managers. The system requires to be operated anywhere without installing any special software, as long as there is a device with Internet access to operate. The system should achieve zero client installation, zero maintenance, strong interactivity, easy expansion, and no need to upgrade multiple clients when upgrading the system, only the server can be upgraded. The system does not take up too many system resources, can run smoothly on most browsers, has good concurrency, and supports more users to connect at the same time. The system should be able to respond quickly, and the response time should be maintained within 2 seconds.

3. Program Design

Due to the rapid population movement and large population base in the community, the traditional community epidemic prevention and control work has problems such as backward recording means, complicated work and low efficiency. After preliminary research and analysis, the system needs to achieve the functions of timely recording the entry and exit of community residents and outsiders and providing residents with practical community epidemic information, so as to improve the efficiency and accuracy of community epidemic prevention and control.

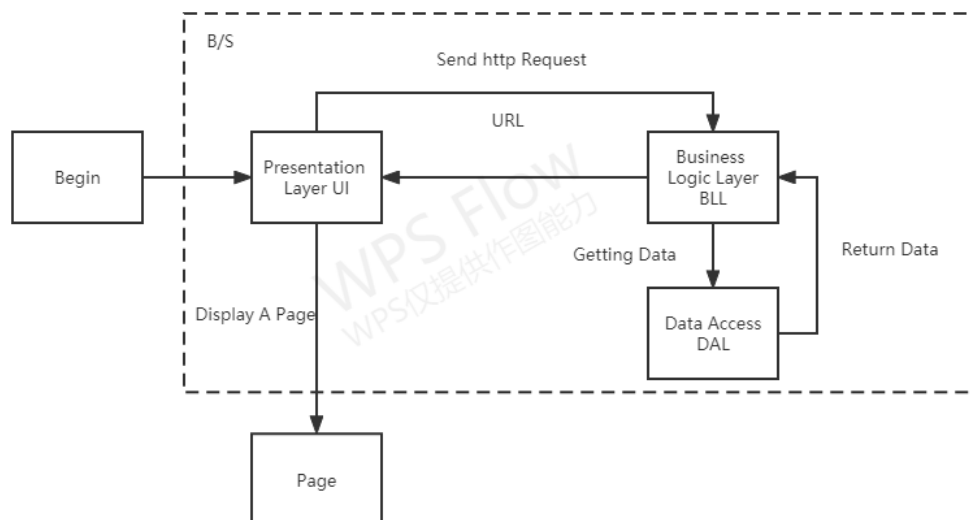### 3.1 System architecture

Figure 1: Overall system structure diagram

The framework structure of the system is divided into 3 layers, where the main role of the presentation

layer (UI for short) is to realize the interaction with the end-user, to obtain the end-user requirements and display the corresponding page. The Business Logic Layer (BLL for short) is for logical processing of data, and the Data Access Layer (DAL for short) uses the database for data analysis and processing, and then feeds it back to the presentation layer. The Data Access Layer (DAL for short) is directly connected to the database, and its main role is to receive information and instructions from the business logic layer and process them through the database, and feed the processing results back to the business logic layer [2]. The overall structure of the system is shown in Figure 1.

3.2 System functions

From the analysis of the various requirements of the system, the functions of the community epidemic management system can be derived as shown in Figure 2.
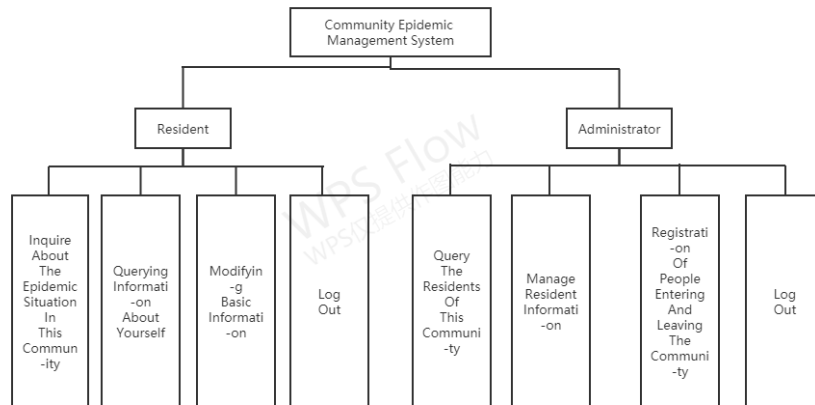

Figure 2: System overall functional structure diagram

3.3 Database design

(1) Conceptual structure design

The following data items and data structures required to record community epidemic information can be listed according to the system functional analysis[3].

1) Residents: serial number, name, gender, telephone, address, current status, current temperature.

2) Resident management: serial number, name, quarantine time, details

3) outsiders: serial number, name, ID number, telephone, body temperature when entering the cell, time of entering the cell, time of leaving the cell

4) Activity records: serial number, name, date, temperature in the morning, temperature in the afternoon, physical condition, time of leaving the cell, time of entering the cell

5) User login: serial number, user name, password.

(2) Logical structure design

Table 1: Basic form for residents' personal information

Field Name	Data Type	Empty	Description
Serial number	Int	No	Number of each resident
Name	Varchar (20)	No	Name of each resident
Gender	Varchar (20)	No	Gender of each resident
Phone	Varchar (11)	Yes	Phone number for each resident
Address	Varchar (100)	No	Each resident's unit building
Current Status	Int	No	0 means normal 1 means infected
Body temperature	Float	No	Resident's body temperature on the day

Based on the conceptual structure design, the logical structure design requires the design of a resident personal information form, a resident isolation form, a user login form, and a record form for people entering and leaving the community. Take the basic resident personal information table as an example, the details are shown in Table 1[4].

4. System Implementation

Since the system is community-oriented, the system should be simple and easy to use, so that users can understand the system functions and operate it accurately as soon as possible. The overall design of the system is based on the needs of community epidemic work, and the system should be simple and easy to use to meet the needs of community epidemic work as much as possible, so that the community epidemic management system can be promoted and used in major communities in the future[5].

4.1 Front End Pages

The front-end webpage is mainly simple and easy to operate. According to the demand, the interface is divided into community residents' information inquiry interface, community residents' temperature registration interface, incoming and outgoing community personnel registration, community situation inquiry, and residents' personal situation inquiry and modification interface[6]. Since the system is community-oriented, it is only divided into two clients: community staff and residents, using the login interface to distinguish the identity of the loggers and the corresponding series of interfaces to complete the functions set by the system.

(1) System login screen

As shown in the Figure 3, users can enter the system through the login screen, only when the account number and password are completely correct, if there is an input error, it will show login failure[7].

Figure 3: Login screen preview

(2) Community resident information inquiry interface

The administrator can check the residents' situation here, including the residents' door number, residents' telephone number and residents' physical condition, etc. By entering the residents' names, the administrator can quickly find the relevant residents' personal information, which brings great convenience to the work of the community administrator. The specific interface is shown in Figure 4: Community resident information inquiry *interface*[8].

Figure 4: Community resident information inquiry interface

(3) Resident personal information inquiry and modification interface

Residents can make password changes or log out in this interface, and at the same time, they can view their personal physical condition and time records of entering and leaving the community in the past
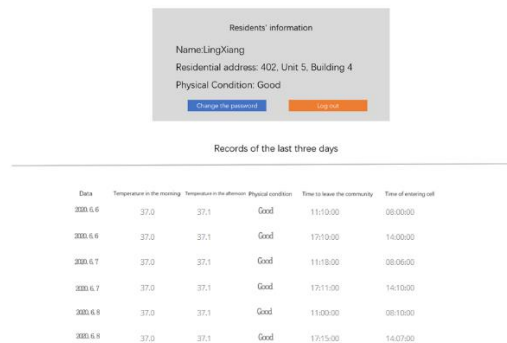
three days, as shown in Figure 5[9].

Figure 5: Resident personal information inquiry and modification interface

(4) Community outbreak status inquiry interface

Residents can check the number of suspected new crowns and the number of patients with new crowns for the corresponding building number and unit number in this interface, so that they can understand the surrounding situation and take better preventive and control measures. The specific interface is shown in Figure 6[10].

Figure 6: Community outbreak status inquiry interface

4.2 Database Management

Table 2: Residents' access to the district table

Field Name	Data Type	Is empty	Description
Serial number	Int	No	The Flowing Water
Name	Varchar (20)	No	Name of each resident
ID number	Varchar (45)	No	Foreigner ID number
Phone	Int	No	Foreigner phone number
Body temperature	Varchar (45)	No	Body temperature when entering and leaving the cell
Date	Varchar (20)	No	Record residents' travel dates
Time to enter the cell	Time	No	Time for residents to enter the neighborhood
Time of leaving the cell	Time	No	Residents go out for community time

The database is based on MySQL, and each basic table is created using Navicat tool. Create residents' personal information table and use it as the main table, set the serial number in it as the primary key (to prevent duplicate names) and set the name as the unique key. Create the table of residents' access to the district, as in Table 2, and link the unique key name in the table of residents' personal information by foreign key. Create a resident isolation table, as in Table 3, and link it by foreign key referencing the primary key serial number in the resident personal information table. Create a user login table and link it by a foreign key referencing the primary key serial number in the resident personal information table. Create a separate outsider record table to record the movement of people in and out of the community. After the data table is successfully created, there is no actual data in the database yet. To ensure that the external keys can be used, the permanent resident data needs to be entered in advance, such as the serial

number in the resident personal information table and the name in the resident personal information table[11].

Table 3: Resident isolation table

Field Name	Data Type	Is empty	Description
Serial number	Int	No	Number of each resident
Name	Varchar (20)	No	Name of each resident
Number of days of isolation	Int	No	0 means no quarantine or end of quarantine
Details	Varchar (200)	Yes	Detailed description of the resident's condition

5. System Testing

After debugging the system code in eclipse and uploading the system to the server, we opened the system interface with multiple browsers and performed several operations such as inputting, deleting and modifying information, the system was able to run normally and the test results basically reached the expectation after going through the basic addition, deletion, modification and switching of each function cycle, and the system stability was excellent during the test[12].

6. Conclusion

The community epidemic management system is closely integrated with the development of B/S technology to meet the current needs of epidemic prevention and control. The system links community administrators with community dwellers, which greatly improves the efficiency of community administrators, relieves the work pressure of community administrators, and is of great practical value to the standardized management of community epidemics.

References

[1] Yin Yunxia, He Yunlong, Li Yan. Development of college financial aid management system based on B/S framework and SQL Server database [J]. Microcomputer Applications, 2020, 36(12):53-55.
[2] Xia Wan, Li Quan, Zhang Jingjing. Construction of intelligent community management system under epidemic [J]. Information Technology and Informatization,2020(08):67-70.
[3] Yan De-ru, Zhang Shuji. The organizational integration path of grassroots party organizations in coordinating community emergency governance [J]. Exploration,2021(01):125-138.
[4] Yang Wei, Geng Liming, Dong Zhenjie. Design and implementation of professional practice process management system based on B/S architecture [J]. Journal of Science and Technology Economy, 2020,28(34):45-46.
[5] Lu Shenghui, Liu Yue, Zhou Ziyu, Wang Jiawei, Chang Jianing, Li Yangqi. Design of B/S-based nucleic acid detection information management system [J]. Network Security Technology and Applications, 2021(01):55-57.
[6] Liao B. Analysis of MVC pattern in JAVA-based Web application development [J]. Electronic Technology and Software Engineering, 2020(21):49-50.
[7] Pei-Pei Lu. Design and implementation of an epidemic monitoring and warning system based on SIR model [D]. Dalian University of Technology, 2021.
[8] Dong Hanlei, Liu Paul, Chen Zhuo, Shu Yunxing. Design and implementation of a Java-based B/S logistics management system [J]. Changjiang Information Communication, 2021,34(05):84-86.
[9] Zhang Shuai. Design and implementation of a big data visualization cloud platform for smart cities [D]. Shenyang University, 2021.
[10] Zhang Yuxin. Design and implementation of patriotic education website based on SpringMVC framework [J]. Electrical and mechanical information, 2020(20): 131+133.
[11] Zhang Y. Design and implementation of the backstage of intelligent pipeline information system based on IOT cloud platform [D]. Beijing University of Posts and Telecommunications,2020.
[12] Zhou Y, Zhang ZF, Zhou ZT, Zhang Y. Design and application of a Java Web-based intelligent medical consultation management system[J]. China Medical Equipment, 2021,18(08):132-135.