CVDMS: A Centralized Vaccine Data Management System

Ruiyu Gao (118010074)
School of Data Science
The Chinese University of Hong Kong, Shenzhen
ruiyugao@link.cuhk.edu.cn

Xinfei Li (118010151)
School of Data Science
The Chinese University of Hong Kong, Shenzhen
xinfeili@link.cuhk.edu.cn

Jingjing He (118010092)
School of Data Science
The Chinese University of Hong Kong, Shenzhen
jingjinghe@link.cuhk.edu.cn

Ke Li (118020231)
School of Data Science
The Chinese University of Hong Kong, Shenzhen
keli@link.cuhk.edu.cn

ABSTRACT

It has been one and a half years since the first COVID-19 case broke out in Wuhan. We notice that there is an increasing demand of vaccination around the world. Therefore, we are going to design a vaccination system, which serves as a centralized database that stor- ing the related vaccination information. Our vaccination system consists of three main entities, which are vaccinators, clinics and vaccine manufacturers respectively. This database system is designed to realize two main functions. The first one is to serve as a database for an appointment platform where the potential vaccinators can book an appointment for vaccination in a specified clinic. The second is to serve as a database for clinicals to store the information of vaccinators, including personal information like name, birthday, gender, appointment date, vaccination location and vaccination status. It will also store the information of vaccination, like manufacturers and inventory. We carefully design a friendly user interface (UI) to help data query and processing. This database will definitely improve the efficiency of vaccination and help control the diffuse of epidemic diseases.

1 INTRODUCTION AND MOTIVATION

It has been almost one and a half years since the first COVID-19 case was discovered in Wuhan province. Globally, new COVID-19 cases increased for the ninth consecutive week, with nearly 5.7 million new cases reported in the last week – surpassing previous peaks. The number of new deaths increased for the sixth consecutive week, with over 87000 new deaths reported. In this edition, a special focus update is provided on COVID-19 variants. The world is in the midst of a COVID-19 pandemic. As WHO and partners work together on the response – tracking the pandemic, advising on critical interventions, distributing vital medical supplies to those in need— they are racing to develop and deploy safe and effective vaccines. Vaccines save millions of lives each year. Vaccines work by training and preparing the body's natural defenses – the immune system – to recognize and fight off the viruses and bacteria they target. After vaccination, if the body is later exposed to those disease-causing germs, the body is immediately ready to destroy them, preventing illness. Several safe and effective vaccines prevent people from getting seriously ill or dying from COVID-19.

Figure 1: COVID-19 Vaccine - Vacuna

Vaccines are a critical new tool in the battle against COVID-19 and it is hugely encouraging to see so many vaccines proving successful and going into development. Working as quickly as they can, scientists from across the world are collaborating and innovating to bring us tests, treatments, and vaccines that will collectively save lives and end this pandemic. It is to provide immunization on specific illnesses or pandemics that getting the vaccination. Immunization is one of the most cost-effective public health interventions. Since the creation of the Expanded Program on Immunization (EPI) in 1974, millions of deaths and disabilities from vaccine-preventable diseases have been prevented around the world.

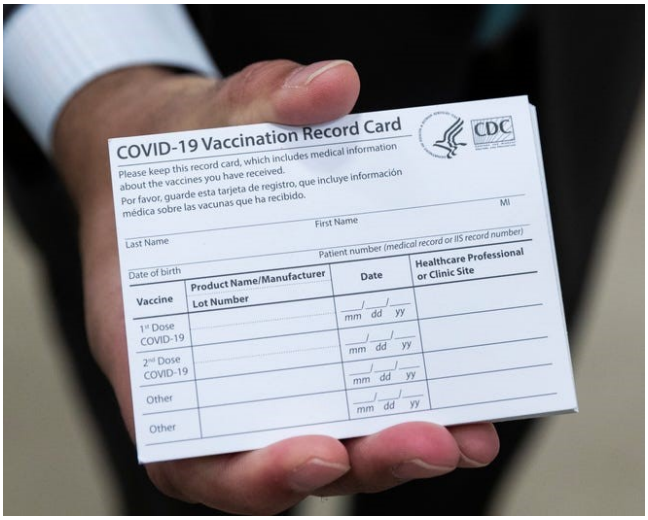

Figure 2: COVID-19 Record Card

However, providing COVID-19 vaccines nationwide requires unprecedented logistics and coordination effort among public health authorities and private-sector partners. Government still requires vaccinators to fill in vaccination record cards. Concerning the global situation that the world is in the midst of a COVID-19 pandemic, there is an increasingly significant demand to build a vaccine database system for the convenience of vaccinators and clinics, which aims to manage vaccine distribution and improve vaccine implementation strategies. Tracking COVID-19 vaccine distribution and administration activities requires collaboration between public and private information technology (IT) systems and integration of existing and newly developed IT systems. For example, the CDC of the USA has a website that guides potential vaccinators on how to get vaccination (Figure 3) .

Our centralized vaccine data management systems will be integrated to ensure successful tracking and reporting of COVID-19 vaccine distribution and administration data. Each section includes an overview and information on training to support implementation.

Centralized Vaccine Data Management System (CVDMS) refers to an integrated IT system —both public and private, as well as new and existing—are used to ensure successful vaccine allocation, distribution, administration, monitoring, and reporting. In the vaccinator's position, CVDMS could help vaccinators to make an appointment for vaccination, which works like an immunization registry. CVDMS has varying capacities to automate processes and to handle a large volume of appointment data. From the clinics' view, CVDMS access people's vaccination records. CVDMS can support clinics by:

- Creating a centralized data repository for vaccination information specific to that jurisdiction
- Approving vaccine orders from enrolled providers and submitting orders to CVDMS
- Monitoring vaccine distribution and changes in vaccine inventory, including accounting for wasted, spoiled, expired, and transferred vaccine
- Providing vaccination coverage assessments

Last but not least, for vaccine manufactures, CVDMS works as a platform for ordering all COVID-19 vaccines. Manufactures can use CVDMS to do following activities:

- View vaccine allocations for each program
- Place and manage vaccine orders for their providers
- Generate reports throughout the vaccine distribution process, from vaccine order placement through distribution
- Track vaccine shipments

CVDMS is a user-friendly system that provides an accessible user interface. And it has been deployed on Tencent Cloud, which means that CVDMS users can access the website anywhere. CVDMS generates data via data generators written in Python via CSV files. Group members fetch the data from reality (mainly in Shenzhen City, Guangdong Province, PRC, and data have been desensitized) to make the generated data more applicable and practical. We set a rigorous user authorization to secure the database to protect user privacy and data confidentiality.

2 SYSTEM DESIGN

2.1 Overview of Database

Three main entities involved in vaccination are users, clinics, and vaccines. The database store necessary information about entities and their relationships. Entity Schemas include

- User {UserID, UserName, Gender, date}
- Clinic {ClinicalID, ClinicalName, Province, City, Street, OpenTime, Telephone}
- VaccineInfo {Batch, Produced_area, Produced_date, Period_of_Validity}

2.1.1 *Relational Schemas.*

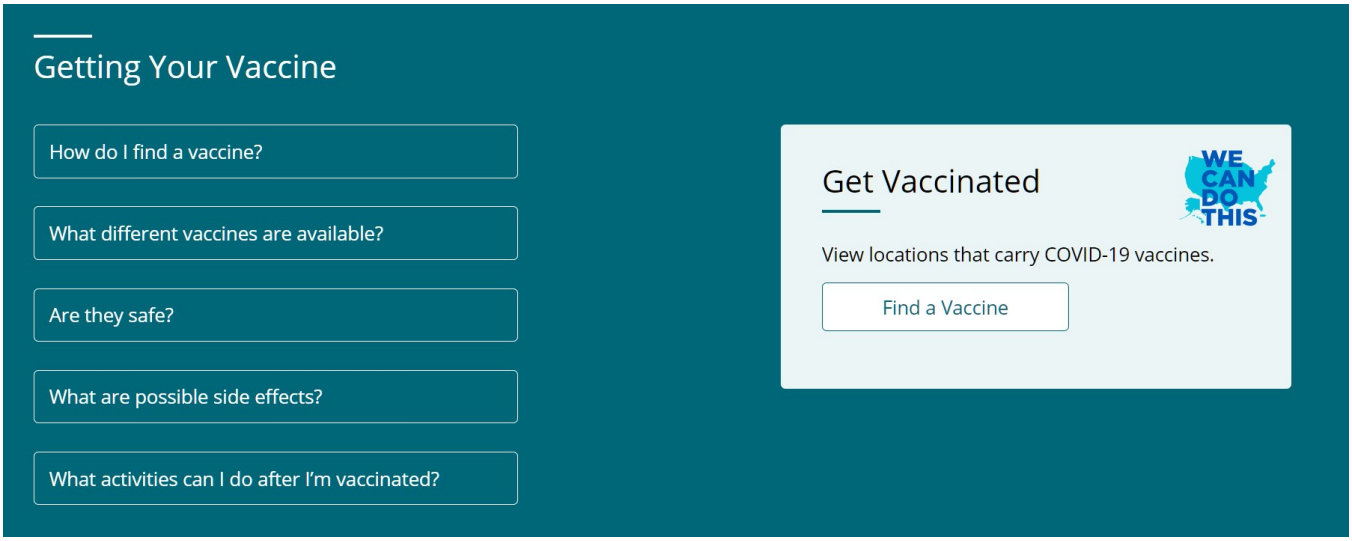

Figure 3: CDC Vaccines for COVID-19 Appointment

- VaccinationAppointment {AppointmentID, UserID, ClinicalID, ScheduledDate, ScheduledTime}
- VaccineStorage {ClinicalID, Batch, PurchaseDate, BuyQuantity, RemainQUantity}

2.1.2 Foreign Key Referencing.

- "UserID" in VaccinationAppointment references User. Explanation: Table "VaccinationAppointment" and table "User" is linked via column "UserID". In this way, we can refer to the user who made a specific appointment. An appointment cannot present in table if the user does not present in table "User".
- "ClinicID" in VaccinationAppointment references Clinic. Explanation: Table "VaccinationAppointment" and table "Clinic" is linked via column "ClinicID". In this way, we can refer to the clinic where the appointment happens. An appointment cannot present in table if the clinic does not present in table "Clinic".
- "ClinicID" in VaccineStorage references Clinic.Explanation: Table "VaccineStorage" and table "Clinic" is linked via column "ClinicID". In this way, we can refer to the clinic where a batch of vaccines is stored. A vaccine storage record cannot present if clinic is not present in table "Clinic".
- "Batch" in VaccineStorage references VaccineInfo.Explanation: Table "VaccineStorage" and table "VaccineInfo" are linked via column "Batch". In this way, we can refer to information of the batch of vaccines in a clinic's vaccine storage. A vaccine storage record cannot present of the vaccine batch number is not valid.

2.1.3 Normalization.

- The database schema meets third norm form (3NF).

2.1.4 Functional Dependency.

- UserID → UserName, Gender, date
- ClinicalID → ClinicalName, Province, City, Street, OpenTime, Telephone
- AppointmentID → UserID, UserName, Gender, BirthDate, ScheduledDate, ScheduledTime, Completed, ClinicalID, ClinicalName, Province, City, Street, OpenTime, Telephone
- Batch → Produced_area, Produced_date, Period_of_Validity, Supplier
- {ClinicalID, Batch, PurchaseDate} → Buy Quantity, RemainQuantity

2.1.5 ER Diagram.

2.1.6 Relationship Cardinalities. The Relationship Cardinalities is shown in Figure 5

2.1.7 Database Structure. The Database Structure of our system is shown in Figure 6

2.1.8 Indexing.

- VaccinationAppointment {AppointmentID, UserID, ClinicalID, ScheduledDate, ScheduledTime}

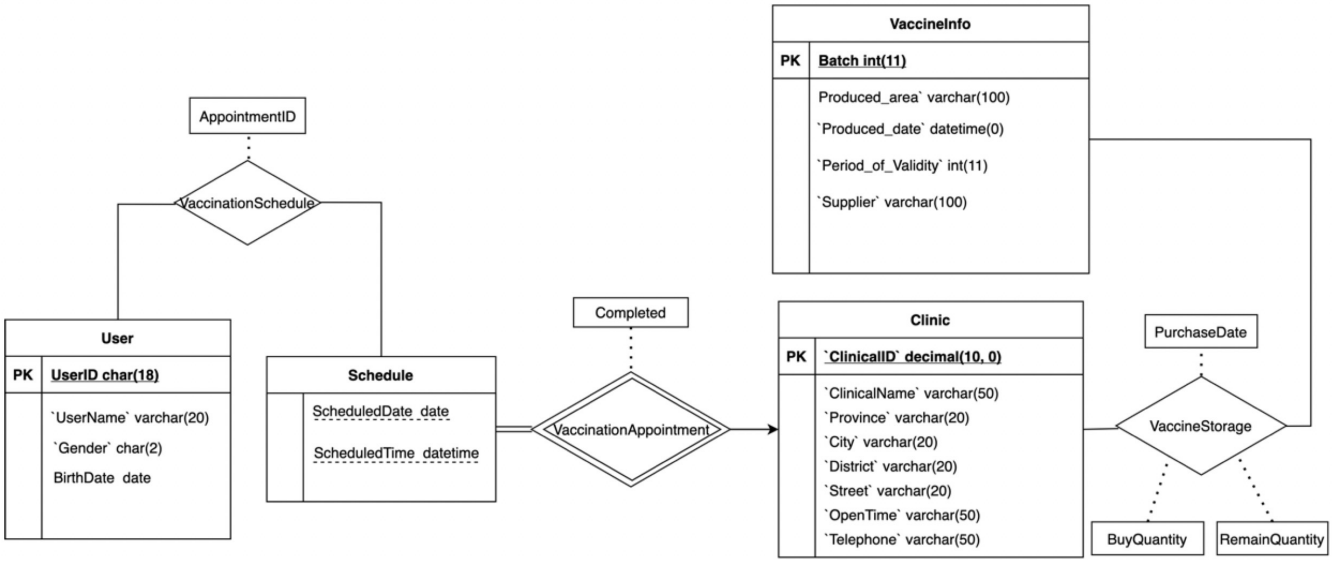

Figure 4: ER diagram

Relationship	Cardinality Constraint	Explanation
User, Schedule	Many to many	A user can have multiple appointments. A time slot of appointment could contain appointments from multiple users
Schedule, Clinic	Many to one	A clinic has many vaccination schedules. A vaccination schedule happens in one single clinic.
Clinic, VaccineInfo	Many to many	A clinic can store multiple batches of vaccines. A batch of vaccines can be stored in different clinics.

Figure 5: Relationship Cardinalities

Indexed Column: AppointmentID, UserID, ClinicalID
Explanation:AppointmentID serves as the primary key of the whole table. We need an appointmentID to uniquely pick out an appointment record.UserID is a foreign key referencingtable “User”. Consider a real case example where user what to findall appointment records made by a username, weneed to find out the appointment records that present with selected UserID. Having UserID indexed would make the query much more efficient.oClinicID is a foreign key referencingtable “Clinic”. Consider a real case example where user what to find all appointment records in a specific clinic, the database needs to find out the appointment records that present with the ClinicID. Having ClinicID indexed would make the query much more efficient.

Database Table	Description
User	Contains all of the customer's information.
Clinic	Contains all information of a clinic.
VaccineInfo	Contains all information of a specific batch of vaccine.
VaccinationAppointment	Contains all information of a vaccination appointment.
VaccineStorage	Contains the storage information of a batch of vaccine in a clinic.

Figure 6: Database Structure

2.1.9 Functionality. The database and web application allow technicians to complete the following functions:

- • Add a record to tables
- • Edit a record in tables
- • Delate a record in tables
- Search a record in a table
- Show the result of query in analytical purpose

3 IMPLEMENTATION

3.1 Overview of Implementation

CVDMS is a web-enabled database that allows users manage tables stored in database. A user interface is developed with direct access via IP address http://34.80.164.190/

3.2 Technology Stack

Here are the major technology stack components: Hypertext Markup Language (HTML), Cascading Style Sheets (CSS), JavaScript and PHP(Hypertext Preprocessor). For more details, the main body frame is written in HTML. The interactive operations are implemented by PHP and Layui, a popular front-end supporter written in JavaScript and CSS.

3.3 Functionality Implementation

- Users can access five tables individually via the first five tabs lies on the left bar of website.
- Users can add a record into current table by clicking "Add" button on the right up corner of the current page.
- Users can edit an existing record by clicking "Edit" button present in the last segment of the record.
- Users can delete an existing record by clicking "Delete" button present in the laast segment of the record.

3.4 Project Structure

Our project is developed on Linux. PHP file under /var/www/html/ directory can be directly visit by HTTP request from internet. In consideration of the convenience of development and deployment, we use integrated environment PHPSTUDY as our running environment. For more details, PHP-7.3.22 and MySQL8.0.16 are used in the integrated environment.

3.5 Website Framework Implementation

The static framework of our website is implemented by HTML and Layui, which is embedded in PHP file. The functionality of interactive operations will be done by PHP.

For more details, PHP code can be divided into three parts by their functions as shown in Figure 11.

3.5.1 Database Connection. We use Mysqli extension in PHP to complete these works. For convenience of use in other parts, we abstract four functions as shown in figure in the connection part.

3.5.2 Data Deletion. Deletion of data can be done on current table with the value of primary key. This demand is fulfilled by the "Delete" button in website.

3.5.3 Website Construction. We used the functions all above to constructs our excellent website. The static framework of website is constructed by HTML.

The buttons can be divided into three types:

- Page jumping buttons. We only need to change the URL because the PHP files can be directly access via name by HTTP

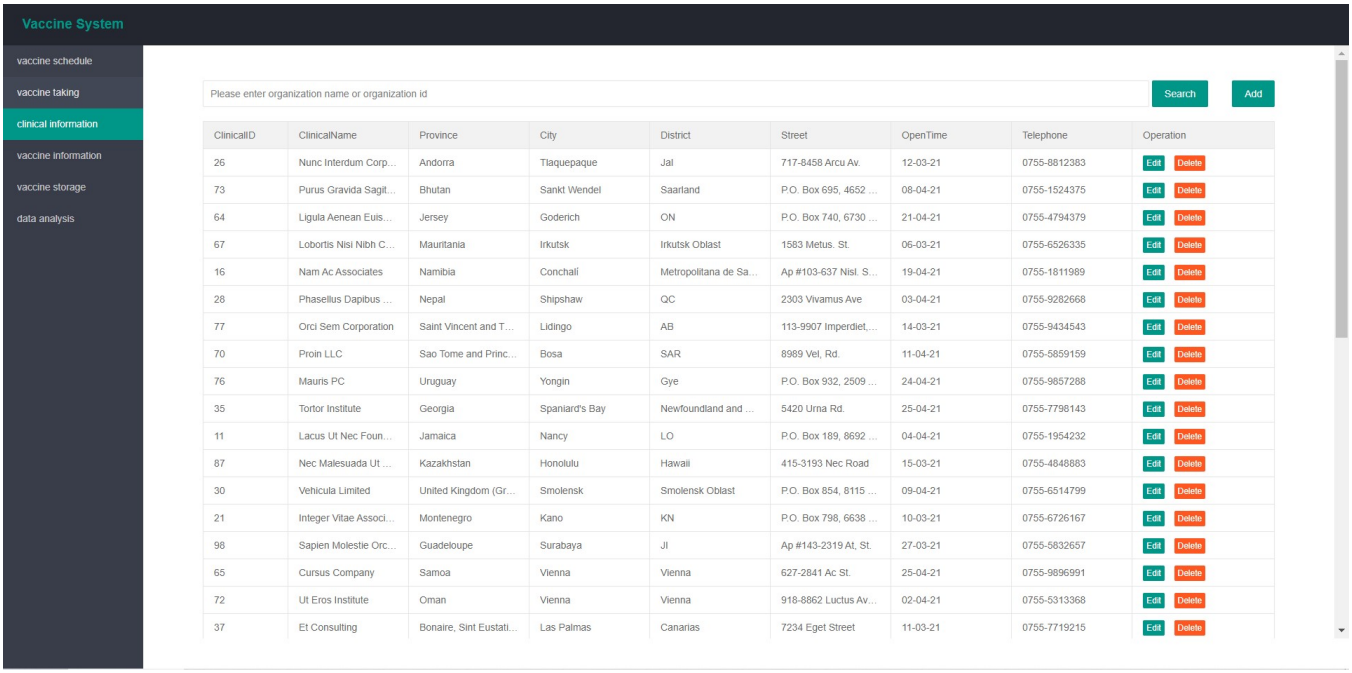

Figure 7: Access

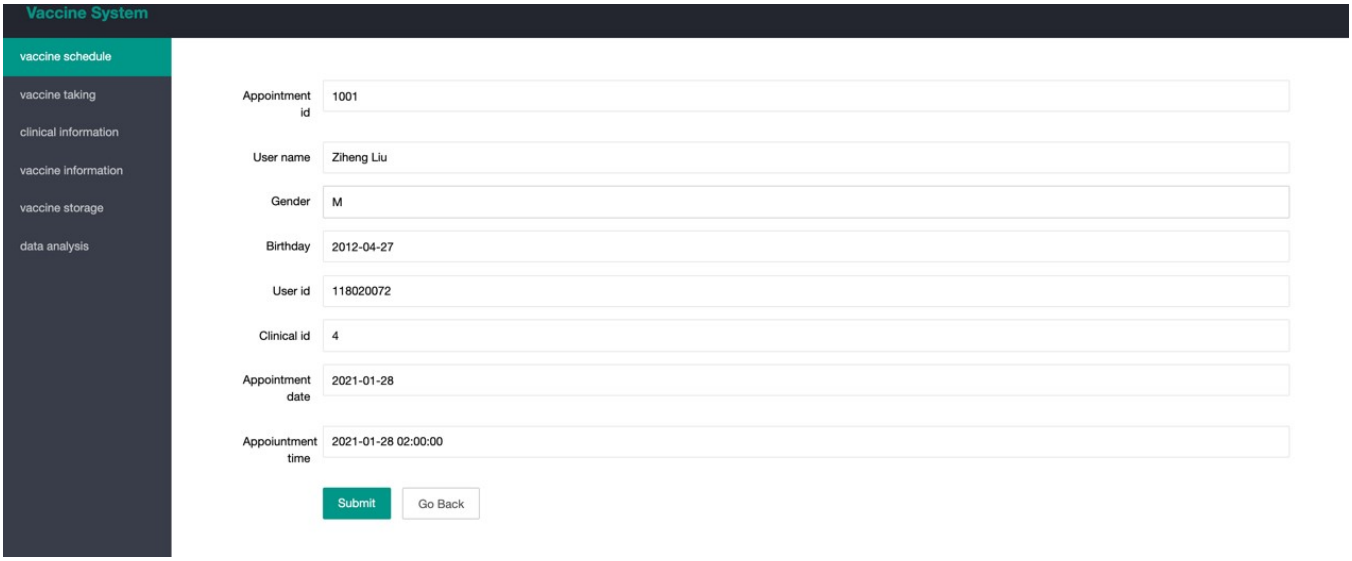

Figure 8: Add and Edit

request. We can easily change it by herf attribute in HTML code, which will directly change the URL we visit.

- Delete/search buttons. Firstly, we should state the communication method between HTML and PHP codes. We use _POST method in both HTML and PHP codes to get the input/selection from users. The information transmitted in this way cannot be seen by any other user, which guarantees the security of our data.

Secondly, we constructed the table layout by table widget in LayUI. For more details, we created a table widget in HTML

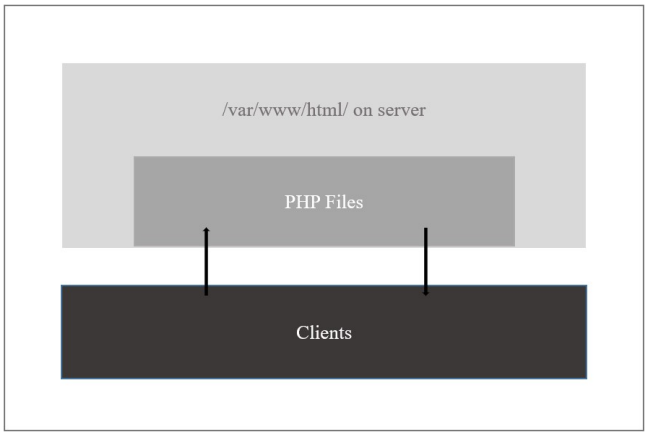

Figure 9: Program Structure

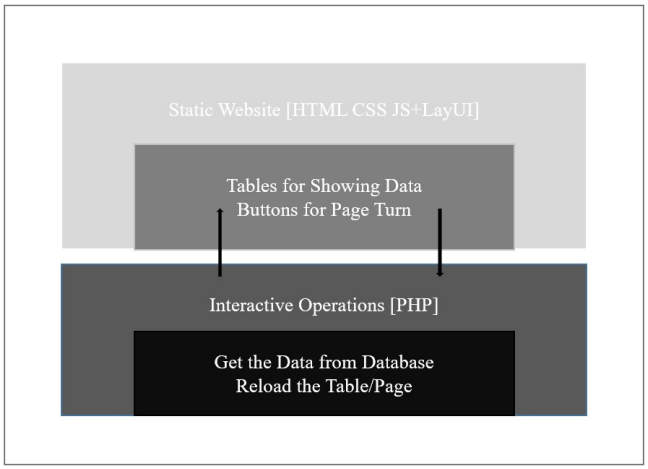

Figure 10: Website Framework Implementation

code and set the URL of table as corresponding PHP file, which will return the result of SELECT operations on database. When there are delete/search operations, we will reload the table with new Query statements.

- Add/edit buttons. When users click these buttons, we should jump to an additional page. The Query operations behind add/edit are similar to search/delete.

4 DATA ANALYSIS AND DATA MINING

Utilizing our CVDMS system and combining with machine learning algorithms, we can perform some data analysis and data mining besides the basic data query operations above. For example, if we want to investigate the geographical distribution of vaccination, in this case, we need to count the vaccination events that happen in every city each month. Below I will show the data analysis using three examples.

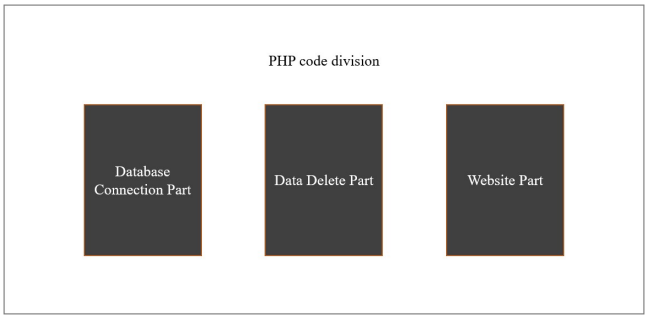

Figure 11: PHP code structure

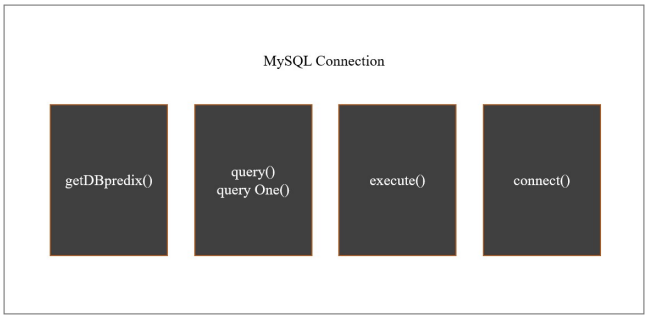

Figure 12: Database Connection

4.1 The proportion of the number of vaccine consumption in different cities

As Figure13 shows, Beijing accounts for around 17 percent in terms of the number of vaccine consumption. When we hover on a specific city, we can see the detailed number of consumed vaccine in the last month. For example, there are 5,500 vaccination taken in total in Beijing yesterday.

4.2 The distribution of the total number of daily appointment in different cities.

As we can see from Figure 14, the average number of total daily appointment in three cities are similar, which is close to from 600 to 800. However, their degree of centralization can be quite different. For example, in Beijing, the number is closely around 600 on most of the days. While in Shanghai, the number can be a random number between 200 to 1400.

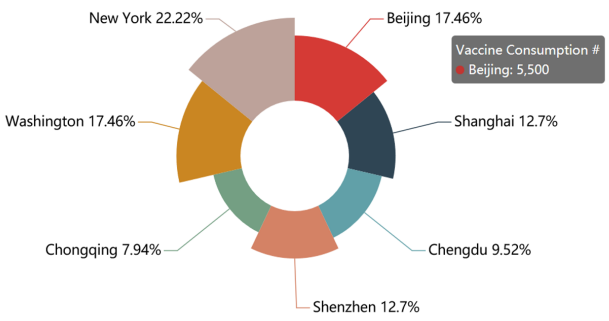

Figure 13: the proportion of the of vaccine consumption in different cities on April 27th 2021.

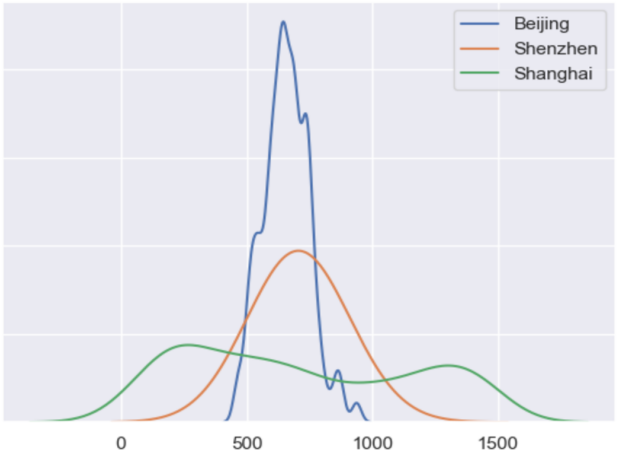

Figure 14: The distribution of the total number of daily appointment in different cities.

4.3 The consumption trend of two types of vaccine.

From the line graph, we can easily see the variation tendency and make comparisons. For example, as we can see, the demand for vaccine 2 is relatively larger than vaccine 1, which helps the clinics to make better decisions when purchasing the vaccine next time.

Leveraging the historical data, we can do more fantastic thing! For example, we can make predictions using data mining algorithms like time series or machine learning. We predicted that in the next half month, we demand for 3200 of vaccine 1 and 3500 of vaccine 2. This is one of the wide AI's applications on health care, which can have great impact on managing the vaccine storage more efficiently for government.

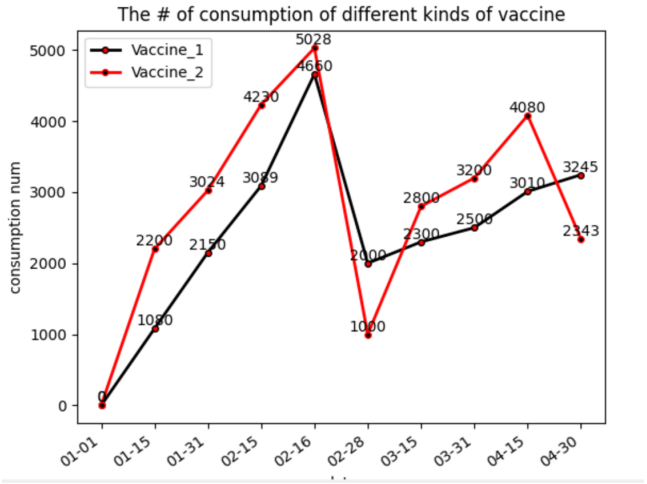

Figure 15: The number of consumption of different vaccine types

5 CONCLUSIONS AND SELF-EVALUATION

In this project, we build a Centralized Vaccine Data Management System (CVDMS) to integrate all the possible vaccination operations: vaccine allocation, distribution, administration, monitoring for clinics and vaccine manufactures, and vaccination appointment for potential vaccinators. By applying MySQL, PHP, CSS+HTML+JS, and Python for database building, front-end construction, and data generation and analytics., our CVDMS shows significant superiority in the efficiency and concurrency of vaccine distribution and implementation strategies.

Taken together, CVDMS is a far more efficient mechanism to store and organize vaccine data and information related to vaccination than spreadsheets; it allows for a centralized facility that can easily be modified and quickly shared among multiple users. Having a web-based front end removes the requirement of users having to understand and use MySQL directly, and allows users to connect from anywhere with an internet connection and a basic web browser. It brings the inspiriting possibility of queries to obtain information on various vaccinators for clinics. Due to the flexibility of access, potential vaccinators can appoint vaccination with the appropriate specific clinic. Lastly, it can dynamically predict stocks of vaccines, which helps vaccine manufacturers come up with a rational plan to organize vaccine production.

Limited by time and data privacy, it is a well-developed centralized

vaccine data management system consisting of considerable design, thought-provoking ideas, and theoretical support for vaccine distribution and implementation strategies.

6 FUTURE WORK

While this project has supplied the basics for a system to keep track of vaccinators, vaccine storage, many further enhancements are desired for greater control of the information.

The first major enhancement would be for clinics to be able to log in directly without authorization and see the information for their vaccinators. They should be able to see what revaccinations the vaccinators have taken, and still need to take, as well as information about their job, pay, location, and email address. Also useful would be the ability to record with vaccine manufactures, which makes the prediction more applicable.

Another useful addition to CVDMS would be an extended system by which the vaccinators should exit the system. Currently, the vaccinators can use the system without checking any personal information; instead, CVDMS is supposed to be set that allows users to log in and check or update their personal information directly. This is currently being worked on, and which are unmet.

A prototype for a more visually appealing and intuitive template was worked on and implemented for the matrix manager. This new, more user-friendly design could be propagated throughout the system.

Finally, a general query capability should be added. This feature will be very useful when attempting to fill out surveys regarding vaccinators, to appointment vaccination from clinics, or to obtain information for various reports.

REFERENCES

[1] Velavan, Thirumalaisamy P and Meyer, Christian G, Tropical medicine & international health, vol. 25, pp. 278.

[2] Le, T Thanh and Andreadakis, Zacharias and Kumar, Arun and Román, R Gómez and Tollefsen, Stig and Saville, Melanie and Mayhew, Stephen and others. The COVID-19 vaccine development landscap, vol. 19, pp.305–306.

[3] Pereira, Jennifer A and Quach, Susan and Hamid, Jemila S and Heidebrecht, Christine L and Quan, Sherman D and Nassif, Jane and Diniz, Amanda Jane and Van Exan, Robert and Malawski, Jeffrey and Gentry, Adrian and others. Exploring the feasibility of integrating barcode scanning technology into vaccine inventory recording in seasonal influenza vaccination clinics. vol. 30, pp.794–802, 2012

[4] Silberschatz, Abraham and Korth, Henry F and Sudarshan, Shashank and others. Database system concepts.vol. 4, 1997.