2. Analysis

(Project Title)

**Personal Health Management System Based on JSP and Servlet**

(Logo) - option



(Student No, Name, E-mail)

**Student No:** 22113877  
**Name:** 딩카이  
**E-mail:** q1[793632722@gmail.com](mailto:793632722@gmail.com)

[ Revision history ]

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision date** | **Version #** | **Description** | **Author** |
| 2025/3.25 | 1.00 | 1 Draft | Author name |
| 2025/4/25 | 1.10 | 2 |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

= Contents =

1. Introduction ...........................................................................................

2. Use case analysis ....................................................................................

3. Domain analysis .........................................................................................

4. User Interface prototype ............................................................................

5. Glossary ......................................................................................................

6. References .................................................................................................

1. Introduction

* Summarize the contents of this document.

- Describe the prominent features of your project (usefulness, significance, expandability etc).

* 12pt, 160%.

This project aims to develop a Personal Health Management System using JSP and Servlet technologies within a Browser/Server (B/S) architecture. As public awareness of health and wellness increases, the need for accessible and user-friendly digital health tools has become vital. This system addresses this need by allowing users to record, track, and consult on personal health data in a structured and secure environment.

The system supports three types of users:

Regular Users: Log health data (diet, exercise, sleep), view physical examination records, access health education content, and consult doctors.

Doctors: Respond to health inquiries from users through the system.

Administrators: Manage user and doctor accounts, oversee content publication, and maintain system integrity and security.

Technically, the system integrates JSP for the frontend, Servlets for backend logic, and SQL Server for persistent storage. The system design emphasizes modularity, clarity, and scalability for future development.

2. Use case analysis

- Build a use case diagram.

* Make detailed description for each use case (Use case description).
* 12pt, 160%.

2.1 Feasibility Analysis

Developing any system is subject to constraints such as time and resources. Therefore, a feasibility analysis must be conducted before starting any project to reduce development risks and avoid wasting manpower, material, and financial resources. The following analysis is conducted from four aspects: technical, economic, operational, and legal feasibility.

2.1.1 Technical Feasibility

This system is developed using MyEclipse and the SQL Server database. The programming language is Java, primarily utilizing J2EE technology. Java is an object-oriented programming language that is simple, easy to learn, and highly flexible. The Java course was already studied in the third year, and J2EE was systematically studied during the fourth year. The overall development difficulty of the Personal Health Management System is not high, with database design and operations being the core of the system. Courses such as Software Engineering, Software Testing, and UML Unified Modeling Language were studied during university, and each semester involved corresponding course projects. Therefore, with acquired skills in system analysis, design, and testing, implementing this system is technically feasible.

2.1.2 Economic Feasibility

In today’s information age, digital management allows personal health management to become more systematic, faster, and comprehensive. This system does not require high computer configurations. Even low-end computers retired from internet cafés or computer labs can fully meet the system requirements. Therefore, the system is economically feasible.

2.1.3 Operational Feasibility

The system is simple to operate. Most input pages use drop-down menus; in some pages, information is automatically generated without manual input. Date entries use a calendar control, making operation convenient. It requires minimal training for operators who are familiar with Windows. Additionally, the system has a highly intuitive visual interface, so there is no significant difficulty in terms of usability.

2.1.4 Legal Feasibility

The Personal Health Management System is self-developed and has practical significance. The development environment and the database used are based on open-source software. Unlike commercial software, this system does not involve infringement issues. Therefore, it is legally feasible.

In conclusion, building a Personal Health Management System is both necessary and feasible.

2.2 System User Use Case Diagrams

2.2.1 Use Case Diagram for General Users

After registering an account and logging into the system, general users can manage their health information, including daily health records, physical examination data, health education activity records, browsing health-related news, viewing doctor information, and consulting with doctors online. The use case diagram for general users is shown in Figure 2.1:

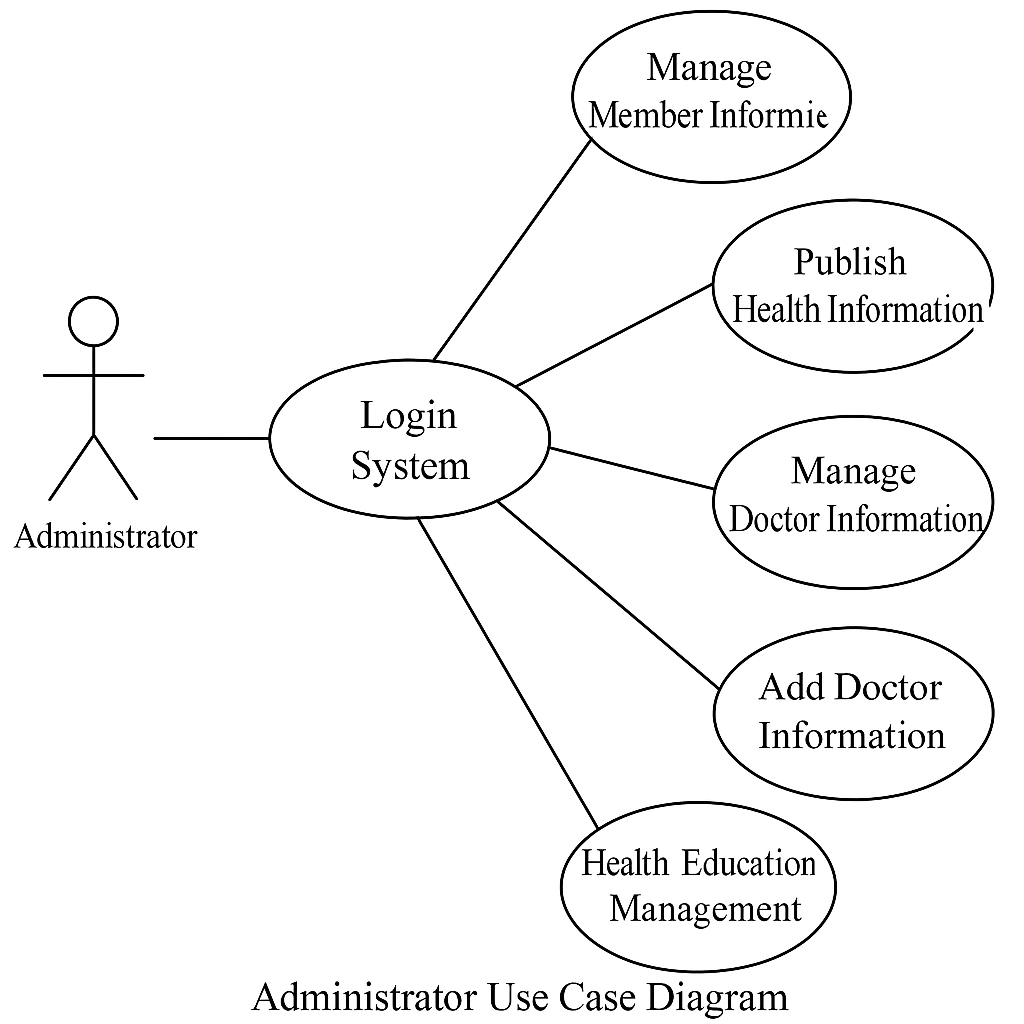


Figure 2.1 Use Case Diagram for General Users

2.2.2 Use Case Diagram for Administrators

Administrators are responsible for managing the information of both general users and doctors. They can also publish health-related information for users to view.

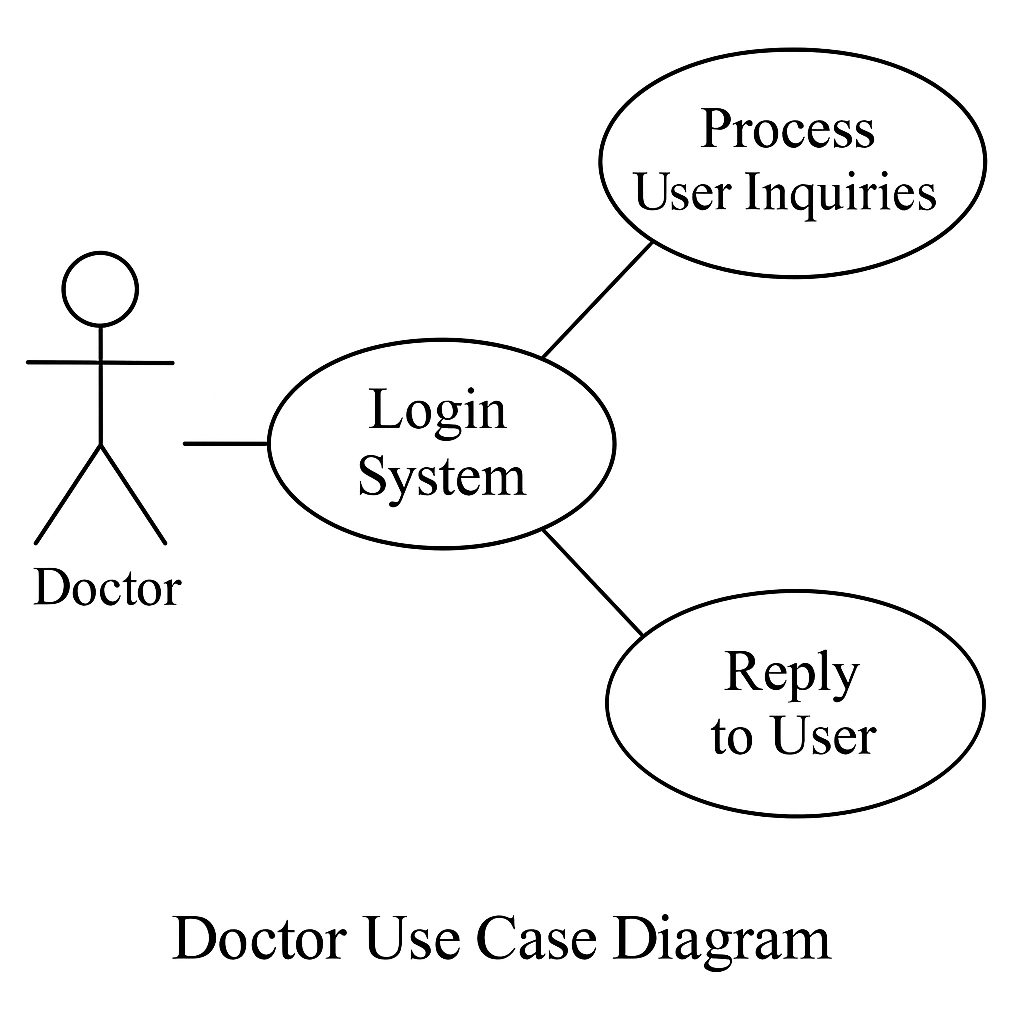
Figure 2.2 Use Case Diagram for Administrators



2.2.3 Use Case Diagram for Doctors

Doctors are responsible for handling health consultations from users. They can view user information and provide professional advice.

Figure 2.3 Use Case Diagram for Doctors



2.3 Functional Module Requirements Analysis

The system is characterized by its simplicity and user-friendly prompts. It will implement the following basic functions:

A clean and elegant interface with simple operations and user-friendly error messages

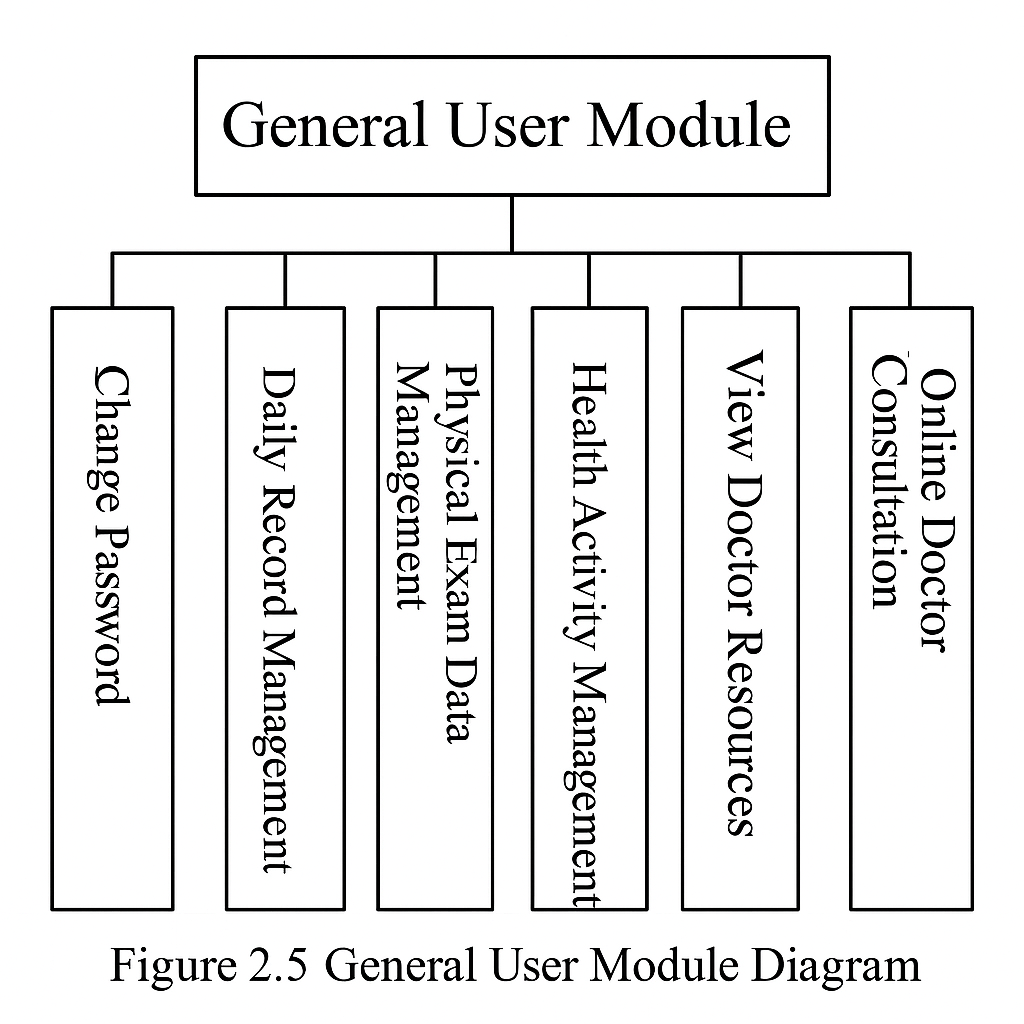
Administrator functions include user management, health information management, doctor information management, publishing health articles, and modifying personal password

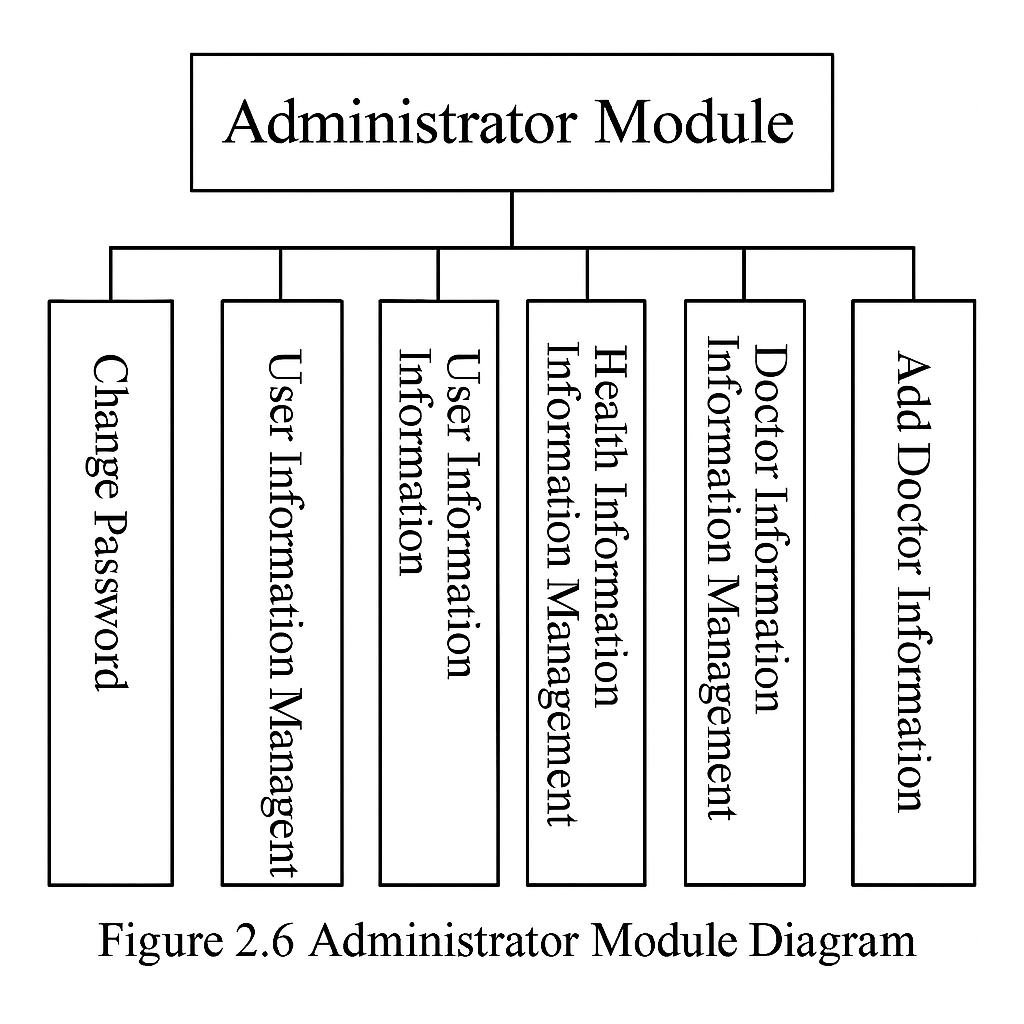
General user functions include managing daily health records, physical examination data, health education activity records, and consulting with doctors

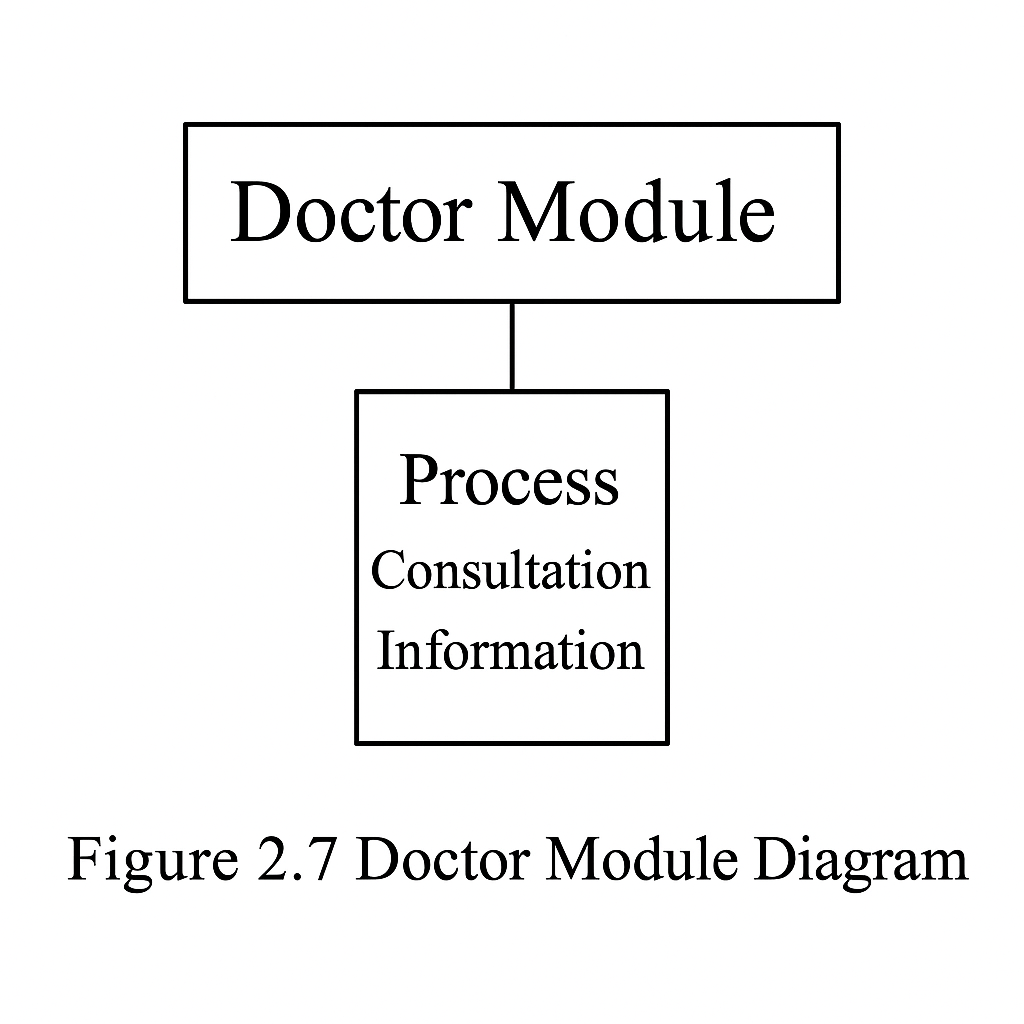
Doctor functions include handling consultation requests from users

High-level security features to prevent malicious user actions

The system's functional architecture is divided into three platforms: general user platform, administrator platform, and doctor user platform.







3. Domain analysis

- Find all of classes in your project.

- Describe the meaning of each class.

* 12pt, 160%.

3.1 Database Analysis and Design

A computer information system is centered around its database. With the support of a database management system (DBMS), it enables the collection, organization, storage, retrieval, updating, processing, analysis, and distribution of information. The database has become the foundation and core of modern information systems. The quality of a database design directly affects the performance and quality of the entire system.

Generally, database design follows five steps: planning, requirement analysis, conceptual design, logical design, and physical design.

3.1.1 Conceptual Database Design

Conceptual design involves building the conceptual structure of the entire system’s database from the bottom up based on data analysis. It is conducted from the user’s perspective by first designing individual views, then integrating those views, and finally optimizing the integrated structure to obtain the final result.

The Entity-Relationship (E-R) model is used for conceptual database design. The E-R model consists of entities, attributes, and relationships. E-R diagrams illustrate the objects involved in a user’s work environment, and attributes describe the characteristics of those entities.

The goal of conceptual design is to produce a conceptual schema that reflects the information requirements of an organization. This schema is independent of the logical structure of the database, the DBMS used, and even the computer system itself.

Based on the above database requirements analysis and the characteristics of the system's conceptual model, the following E-R diagrams have been created:

3.1.2 Logical Database Design

Since the conceptual data model is independent of any specific DBMS, it needs to be converted into a logical relational model based on the features of the DBMS in use. The transformation from the E-R model to the relational model should follow these principles:

Each entity is converted into a relation (table)

All primary keys must be defined as NOT NULL

Binary relationships should define foreign keys based on their types (one-to-many, weak to strong, one-to-one, many-to-many)

Based on the E-R model, the logical data structure for the Personal Health Management System is designed as follows:

(1) General User Table  
Table 3.1: General User Table (t\_yonghu)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| id | int | 4 | No | Yes | ID |
| loginName | varchar | 50 | No | No | Account Name |
| loginPass | varchar | 50 | No | No | Password |
| xingming | varchar | 50 | No | No | Name |
| xingbie | varchar | 50 | No | No | Gender |
| shengri | varchar | 50 | No | No | Date of Birth |

(2) Daily Record Table  
Table 3.2: Daily Record Table (t\_jilu)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| id | int | 4 | No | Yes | ID |
| yonghu\_id | varchar | 4 | No | No | User ID |
| shijian | varchar | 50 | No | No | Record Time |
| yinshi | varchar | 50 | No | No | Diet Status |
| yundong | varchar | 50 | No | No | Exercise Status |
| shuimian | varchar | 50 | No | No | Sleep Status |
| beizhu | varchar | 50 | No | No | Remarks |

(3) Physical Examination Table  
Table 3.3: Physical Examination Table (t\_tijian)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| id | int | 4 | No | Yes | ID |
| yonghu\_id | int | 4 | No | No | User ID |
| shijian | varchar | 50 | No | No | Examination Time |
| jieguo | varchar | 50 | No | No | Examination Result |

(4) Health Education Activity Table  
Table 3.4: Health Education Activity Table (t\_huodong)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| id | int | 2 | No | Yes | ID |
| yonghu\_id | int | 50 | No | No | User ID |
| shijian | varchar | 50 | No | No | Activity Time |
| didian | varchar | 50 | No | No | Location |
| xingshi | varchar | 50 | No | No | Activity Form |
| zhuti | varchar | 50 | No | No | Activity Theme |
| zuzhizhe | varchar | 50 | No | No | Organizer |
| neirong | varchar | 50 | No | No | Activity Content |

(5) Doctor Information Table  
Table 3.5: Doctor Information Table (t\_yisheng)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| id | int | 4 | No | Yes | ID |
| xingming | varchar | 50 | No | No | Name |
| xingbie | date | 8 | No | No | Gender |
| nianling | int | 4 | No | No | Age |
| biye | int | 4 | No | No | Graduate School |
| loginname | int | 4 | No | No | Username |
| loginpw | int | 4 | No | No | Password |

(6) Consultation Table  
Table 3.6: Consultation Table (t\_liuyan)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| id | int | 4 | No | Yes | ID |
| content | varchar | 5000 | No | No | Content |
| liuyanshijian | varchar | 50 | No | No | Consultation Time |
| huifu | varchar | 50 | No | No | Reply Content |
| huifushijian | varchar | 50 | No | No | Reply Time |
| yonghu\_id | int | 4 | No | No | User ID |
| yisheng\_id | int | 4 | No | No | Doctor ID |

(7) Administrator Table  
Table 3.7: Administrator Table (t\_admin)

| Column Name | Data Type | Length | Nullable | Primary Key | Description |
| --- | --- | --- | --- | --- | --- |
| userId | int | 4 | No | Yes | ID |
| userName | varchar | 50 | No | No | Username |
| userPw | varchar | 50 | No | No | Password |

3.1.3 Database Connectivity Principle

The system uses JDBC to connect to the database. By simply importing the corresponding database JAR package into the project, the database can be accessed easily. The Class.forName() method is used to load the driver, and the DriverManager.getConnection() method creates a connection.

The system follows the DAO (Data Access Object) pattern for database operations. DAO is a classical design pattern in Java programming, widely used and a fundamental part of the persistence layer in the J2EE architecture. DAO is based on a layered software architecture and abstracts access to data sources. This abstraction means developers don’t need to know the physical location or type of the database—they can simply operate on encapsulated data objects.

DAO Pattern Class Diagram:  
Figure 3.10 DAO Pattern Class Diagram

BusinessObject: The business object, which acts as the client of the DAO pattern

DataTransferObject: Transfers data between different layers of the application

DataObjectAccess: Encapsulates basic operations for the data source

DataSource: Refers to the data source itself

This structure effectively separates business logic from data logic, resulting in well-layered and maintainable code. The system uses a DBContent class to simplify database connections. Example code is as follows:

public DBContent(){

String CLASSFORNAME = "com.microsoft.sqlserver.jdbc.SQLServerDriver";

String url = "jdbc:sqlserver://localhost:1433;databaseName=db\_jiankang";

String user = "sa";

String password = "sa";

try {

Class.forName(CLASSFORNAME);

con = DriverManager.getConnection(url, user, password);

stmt = con.createStatement(ResultSet.TYPE\_SCROLL\_SENSITIVE, ResultSet.CONCUR\_UPDATABLE);

} catch (Exception ex) {

ex.printStackTrace();

}

}

Wherever a database connection is required, you can simply instantiate a DBContent object to establish and operate on the database.

4. User Interface prototype

* Develop user interfaces of your system.

- It will be easy if you just think that you make a preliminary user manual of your system based on your user interface.

* 12pt, 160%.

The user interface is designed for simplicity and functional clarity. Each role has a custom dashboard:

Login Page: Users select their role and enter credentials. Validation occurs server-side.

Regular User Dashboard: Access to daily record logging, exam history, articles, and doctor consultation.

Doctor Dashboard: Inbox for user inquiries and submission interface for replies.

Admin Dashboard: User and doctor management, content publishing, password reset.

All pages are implemented using JSP, styled with CSS, and supported by JavaScript for form validation and interactivity. The interface is responsive, lightweight, and intuitive to use.

5. Glossary

- Specifically describe all of the terms used in this documents.

* 12pt, 160%.

5.1 Purpose and Significance of System Testing

System testing is a very important and lengthy phase in the development cycle of a Management Information System (MIS). Its importance lies in the fact that it is the final checkpoint for ensuring system quality and reliability. It serves as the ultimate review of the entire development process, including system analysis, design, and implementation.

The primary goal of system testing is to thoroughly identify errors in the program and improve the reliability of the software system. Its objective is to evaluate how well the system has been built. This phase can be divided into three main steps:

Unit Testing – to verify the correctness of each module.

Integration Testing – to test whether the interfaces between modules are functioning properly.

Acceptance Testing – to determine whether the entire software system meets user functional and performance requirements.

Once issues are identified, debugging is carried out to locate and correct the errors. System testing is black-box testing based on the overall system requirements specification and should cover the integrated components of the system. The goal of system testing is to verify whether the system meets the defined requirements and to identify any nonconformities or inconsistencies with those specifications.

5.2 Testing Process

At the beginning of the testing phase, the database initially contained randomly entered data, which lacked practical relevance. In order to enhance the system's completeness, all randomly input data was cleared, and a set of valid test data was entered. This allowed for a more realistic assessment of the system’s functionality.

Through continuous problem discovery, repeated testing, and debugging, issues were resolved one by one until the system was able to function normally.

5.2.1 Login Module Test of the Main Page

Test Procedure:

Open the system homepage and enter incorrect login information.

Attempt to log in.

Enter correct login credentials.

Attempt to log in again.

Test Results:

| Module Name | Test Case | Expected Result | Actual Result | Pass/Fail |
| --- | --- | --- | --- | --- |
| Login Module | Username: null, Password: null | Display error message: “Please enter username” | Login failed, message shown as expected | Pass |
| Login Module | Username: m1, Password: null | Display error message: “Please enter password” | Login failed, message shown as expected | Pass |
| Login Module | Username: m1, Password: 1 | Display error: “Incorrect username or password” | Login failed, error message shown as expected | Pass |

5.3 Other Errors

During the system design and implementation process, various unpredictable errors are inevitable. These may include syntax errors, data type mismatches, parameter passing issues, etc. Such errors must be carefully examined and resolved through repeated debugging and meticulous checks.

For instance, misconfiguration in servlet path mapping due to carelessness can result in runtime errors, such as the one shown in Figure 5.4.

6. References

- Describe all of your references (book, paper, technical report etc).

* 12pt, 160%.

Personal Health Management System

A web-based platform that allows users to manage health-related data, consult doctors, and access educational content.

**User**

A general user who manages their own health records through the system.

Doctor

A certified medical professional responding to health inquiries in the system.

Administrator

A role responsible for managing users, doctors, and system content.

Daily Record

A log of health behaviors such as meals, sleep duration, and physical activity.

Physical Examination

Formal medical test records submitted by users or healthcare staff.

Health Activity

Educational or participatory events aimed at improving health literacy.

Consultation

A system feature allowing users to ask doctors health-related questions and receive replies.

**Login Authentication**

A system mechanism for verifying a user's identity with credentials.

**Session Management**

The system's method of maintaining user sessions and ensuring secure interaction during login duration.

[1] Bai Qiuchan, Gao Aihua, Shen Xianlai. Physical Examination Management Information System for General Users Based on VFP [J]. Computer and Digital Engineering, 2006, 34(01): 143–145.

[2] Shu Pan, Chen Jingang. Implementation and Design of Physical Examination Management System in Digital Campus Construction [J]. Journal of Wuhan Institute of Technology, 2008, 30(04): 108–111.

[3] Xu Chaoyi. Object-Oriented Analysis of Personal Health Management System [J]. Journal of Anhui University of Science and Technology (Natural Science Edition), 2005, 25(03): 62–64.

[4] Ai Lingxian. Design and Implementation of Department-Level Website Construction Platform in Colleges and Universities [J]. Science and Technology Information, Issue 16, 2008.

[5] Zhang Guoyu, Mou Zongguo. Discussion on Strengthening and Improving Personal Health Management Work [J]. Journal of Sichuan College of Education, 2007, 23(Supplement): 19–20.