***2. Overall Design***

**2.1 Operational Requirement**

The Course Resource Sharing Subsystem is a subsystem under the Teaching Service System. It works as a platform for teachers and students to share educational resources. It also allows the teaching staff to assign and collect homework and assignments.

**2.1.1 Functional Requirement**

The customer should login from a main system and enter our subsystem by clicking the URL link in main interface. The main interface page will also post the basic information about this subsystem.

The teaching resource sharing subsystem has two interfaces, Resource Management Interface and Homework Management Interface.

The Resource Management Interface includes command of viewing resource, command of uploading resource (limited by user’s privilege and level of resource (public, group only, class only and school only)), command of resource management, command of downloading resource, command of searching resource.

The Homework Management Interface includes command of homework assignment, command of homework submission, command of checking homework and command of searching homework.

**2.1.2 Performance Requirement**

The performance of this subsystem would have to depend on the server-side database, data transmitting time of the network and the amount of users online.

For the system itself, good design should be applied. The client-side should avoid repeating meaningless request to server which would greatly affect the performance, especially under a poor internet environment. Also, the UI should appear to be simple and user-friendly.

A stable server should be used so as to maximize the capable number of users and minimize the response time of the system. The server-side should have strategies designed to deal with emergency situations e.g. power cut.

**2.1.3 I/O Requirement**

A web application is acting as the client to provide a user-friendly UI for user. User can submit forms or click onto links/buttons in the webpage so as to manipulate the information in the database.

The results of whatever changes imposed by user would be shown on the webpage.

**2.1.4 Requirement of Data Management Capability**

*Security*:

File confidentiality – access of irrelevant individual to the system should be prevented;

Server security – server should be able to defense common attack from hackers.

*Performance*:

The server-side of the system should build up a permanent connection to the database, so as to avoid the wastage by reconnecting the database.

**2.2 Operating Environment**

**2.2.1 Minimum System Requirement**

*Computer:*

* CPU: ≥2.0GHz (e.g. Intel Mobile Core 2 Duo T5800)
* Memory: ≥4.0GB

*Hardware:*

* Keyboard: Usable
* Mouse: Required; Not required only if under Linux environment
* Monitor: Useable
* Hard Drive: ≥100GB, ≥7200rpm

*Network:*

* Network Interface Card: 100M
* Network Access: Local Area Network

*Client:*

* OS: Windows Vista or newer; Mac OS X or newer; Mobile Platform is not supported
* Browser: Microsoft Internet Explorer 10 or newer; Safari 7.0.3; Chrome 34.0.1847.137 or newer
* Network Access: Local Area Network

**2.2.2 Supporting Software**

*Server-side:*

* OS: Windows Server 2012 or newer; Windows 8 or newer; Mac OS X 10.9 or newer; Ubuntu 13.04 or newer
* PHP: PHP 5.0
* Database management : MySQL
* Web Server: Apache
* Browser: Microsoft Internet Explorer 10 or newer; Safari 7.0.3; Chrome 34.0.1847.137 or newer
* Adobe PDF viewer
* MS Office 2010 or newer; Pages 5.2 or newer; Numbers 3.2 or newer; Keynote 6.2 or newer

*Client-side:*

* Browser: Microsoft Internet Explorer 10 or newer; Safari 7.0.3; Chrome 34.0.1847.137 or newer

**2.3 Basic Design Concepts and Procedure**

This system can be divided into client-side and server-side.

For client-side, HTML and CSS is used to build up the website. Bootstrap is adopted for the UI design. JavaScript/JQuery is also used to

For server-side, apache is used as web server, PHP is used as the script language and MySQL is used as the database.

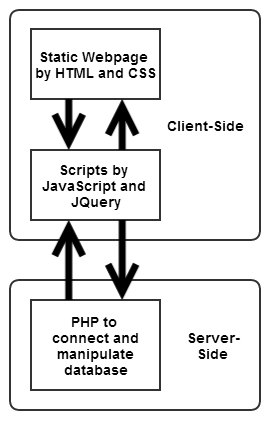


Figure 2.1 Procedure Flow Chart of the Course Resource Sharing Subsystem

**2.4 Structure**

**2.4.1 Structural Partitioning by Function (Horizontal partitioning)**

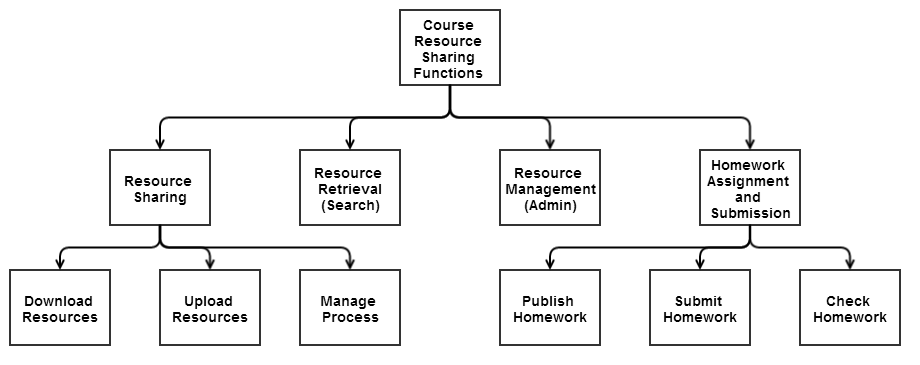


Figure 2.2 - HIPO of the Course Resource Sharing Subsystem

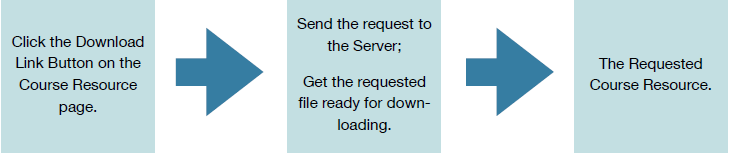


Figure 2.3 IPO of downloading resources

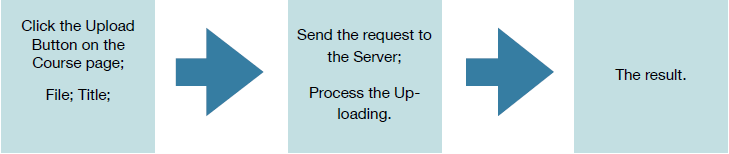


Figure 2.4 IPO of uploading resource

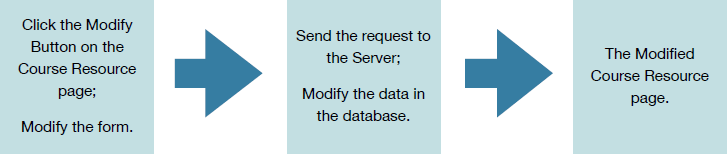


Figure 2.5 IPO of managing uploaded resources

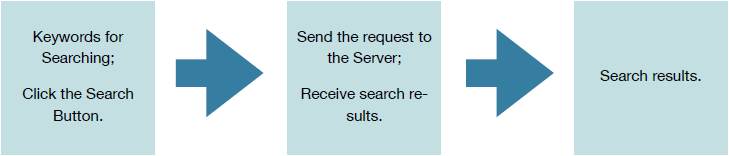


Figure 2.6 IPO of resource retrieval (search)

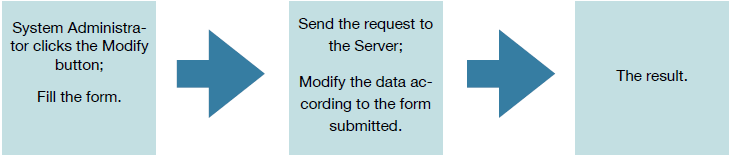


Figure 2.7 IPO of resource management (admin)

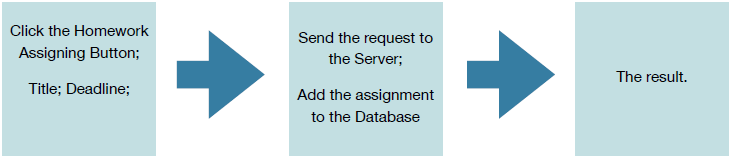


Figure 2.8 IPO of publishing homework

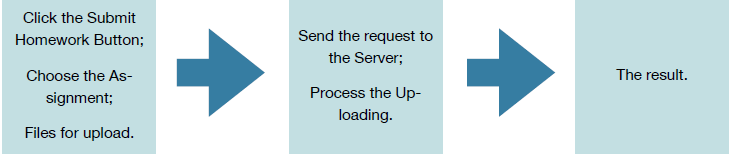


Figure 2.9 IPO of submitting homework

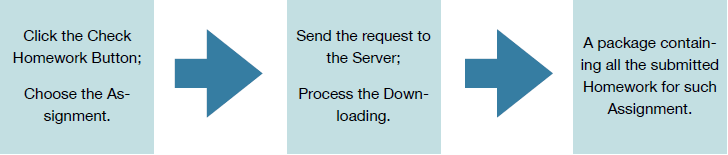


Figure 2.10 IPO of checking homework

**2.4.2 Structural Partitioning by Function by procedure (Vertical partitioning)**

2.4.2.1 Client

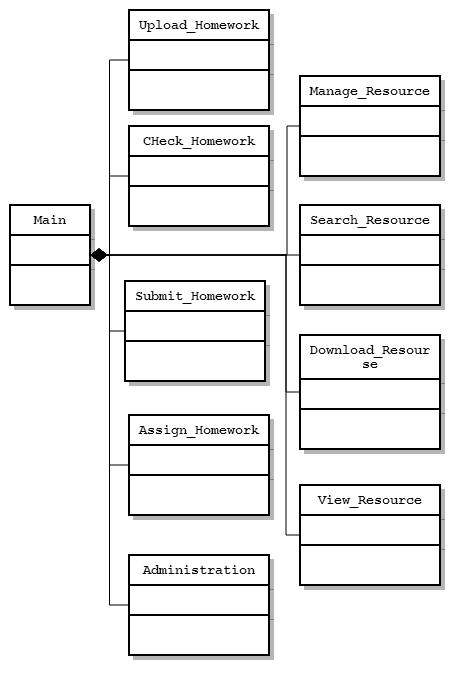


Figure 2.11 Class diagram of client-side programme

2.4.2.2 Server

This is mainly programmed by PHP.

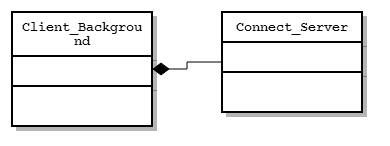


Figure 2.12 Class diagram of server-side programme

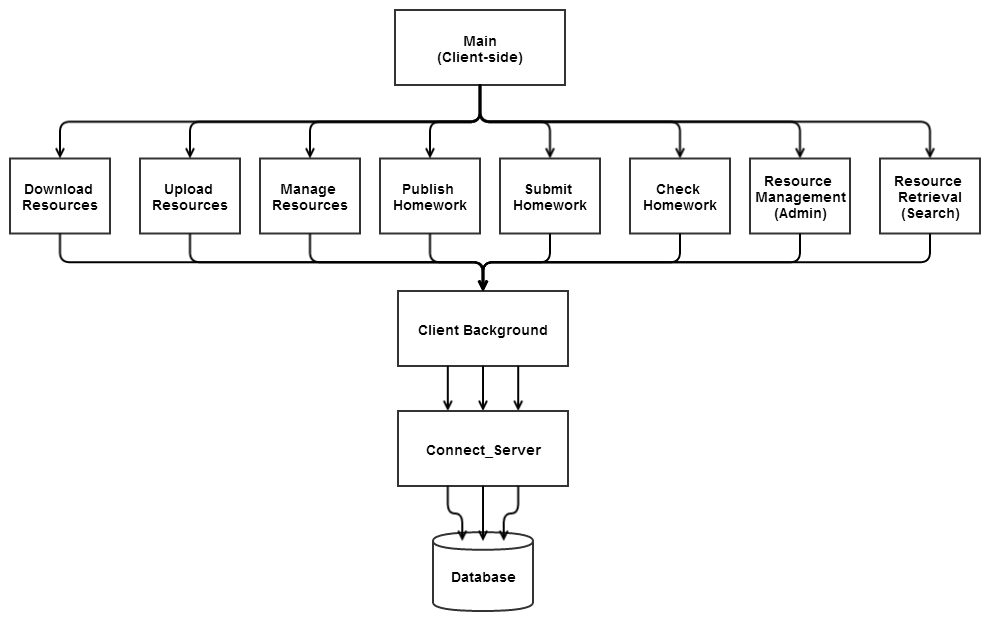


Figure 2.13 Relationship between different modules

**2.5 Interface**

The different modules in the system communicate by database sharing.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| \ | Resource Sharing | Homework Assignment and Submission | Resource Management (Administration) | Resource Retrieval (Search) |
| Client | Share the Resource database | Share the Homework database | Share the Resource and Homework database | Share the Resource database |

Figure 2.15 Database sharing across different modules in table

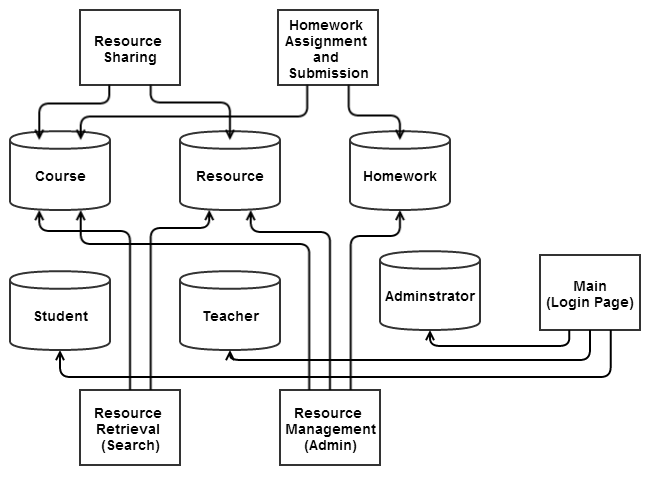


Figure2.16 Database sharing across different modules in graph

**2.6 Manual Error Handling**

If an error occurs and cannot be resolved by the programme itself, it would be handled manually.

**2.7 Unsolved issues**

The user cannot change his/her password. Also, there’s no means to deal with the situation when a user forgets password.