Requirements Analysis for a Student Information System

Jilang Pan

BSc Information System & Management Studies
Session (2003/2004)

The candidate confirms that the work submitted is their own and the appropriate credit has been given where reference has been made to the work of others.

I understand that failure to attribute material which is obtained from another source may be considered as plagiarism.

(Signature of student)

Acknowledgements

The completion of this project owes a great deal to the help, advice and assistance given by numerous people along the way. Primarily, I must thank my supervisor, Mr. Owen Johnson, who has been unfailingly helpful and always available to assist me with my problems. Secondly I would like to thank everybody with School of Computing, Leeds University Business School and Shanghai University of Finance and Economics whom I interviewed or received information from. They are:

Dr Les Proll
Dr Sarah Fores
Mr. Jonathan Ainsworth
Mr. Owen Johnson
Dr John Stell
Miss Millitza Callinan
Mr. Nicolas Forsans
Mr. Tony Jones
Mr. Paul Wheatley
Mrs. Haiying Zhao
Miss Ye Gong
Miss Jin Xu

Jilang Pan

April 2004

Summary

In order to successfully complete my project, a number of objectives firstly defined at the start of the project:

- Investigate what the current student information systems of three different university schools are
- Produce a detailed requirement specification for the potential new student information System for general university
- Evaluate and analysis the requirement specification of the potential new student information module.

The minimum requirements of this project have been set in the beginning of the project. They are:

- Conduct a requirement specification from first principles that will be based around the generic requirement of a typical university school
- An investigation of potential customers (university school)
- Business process modelling and information capture requirement
- Providing evidence of investigation
- At least one presentation for summarise requirement specification

In order to deliver a successful and usable requirement specification for generic university schools, all those requirements not only be met but some are exceeded in the project. A detailed system requirement specification was generated eventually on the basis of four current system investigations.

Table of Content

Chapter 1 Introduction

1.1 Introduction of the project

This project has been undertaken as a fundamental requirement of a BSc (Hons) Information System and Management Studies at the University of Leeds. The purpose is to investigate the current Student Information Systems that are used in university schools. Once this has been done, detailed requirement specification will be generated from the current systems' investigation, analysis and comparison. The requirement specification will be analysed and evaluated.

This chapter explains the project objectives and the scope of the investigation. The problem domain is described, the key terms are introduced and the structure and format of this report are briefly summarised.

1.2 Objectives, Requirement and Scope of this studies

In order to successfully complete my project, a number of objectives firstly need to be defined:

- Investigate what the current Student Information Systems of three different university schools are. Those chosen systems are: School Information System (SIS) in School of Computing (SoC), System in Leeds University Business School (LUBS), Student information System in Shanghai University of Finance and Economics (SUFE).

- Produce a detailed requirement specification for the potential new Student Information System of general university school

- Evaluate and analysis the requirement specification of the potential new Student Information System

Within those objectives, the scope of the project has been set. Those three Student Information Systems will be carefully investigated. Due to the vague boundary and relation between university school's sub-system and university wide system in China, the system in SUFE will also be covered in the investigation to generate a more interesting comparison of three Systems and hopefully, some useful mutually improvement recommendations can be made eventually.

The minimum requirements of this project have been set in the beginning of the project. They are:

- Conduct a requirement specification from first principles that will be based around the generic requirement of a typical university school
- An investigation of potential customers (university school)
- Business process modelling and information capture requirement
- Providing evidence of investigation
- At least one presentation for summarise requirement specification

In order to deliver a successful and usable requirement specification for generic university schools, the modelling of those three targeted Student Information Systems and their comparison and evaluation will be critical to the success of this intention.

1.3 What is a Student Information System

First of all, let’s see what an information system is. From the literature review, Sauer (1993) argued that some people will see them as an expense, others as a solution, a control mechanism, a threat to the quality of working life or even as a technical problem in 1993. The diverse perceptions of the Information system resulted in a huge differences in the definitions of IS that are available. Lucas (1995) describes an information system as helping to control operations in an organization, whilst Laudon & Laudon (2002) defines it technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making, coordination, and control in an organization. Information systems are seen as a strategic resource within the organization, that is, they have an important impact on key operations which determine the livelihood of the organization (Martin, 1992).

Student information system literally means the general information systems for maintaining and providing student information. It exists in all the schools, colleges, universities and any other education institutions. However, those information systems vary. Some of them are paper based; heavily manual work is involved in managing and maintaining information such as student personal records files. However, recently, most schools, even down to the very smallest, utilize computers in some way or anther. The uses to which the computers are put vary enormously, ranging from word processing and spreadsheet through to worldwide on-line access, complicated user access permission system and vast functionalities.

1.4 Why do university schools need Student Information System?

Martin (1992) rightly says that it is probably true that many of today’s business simply could not function effectively without automated information processing systems of some form or another, so do university schools. All of them have common tasks such as collecting, storing and processing information regarding their students, staff and the work done within the department. With the increase of information, it is unwise to adopting the traditional paper based system which is slow to access and therefore, inefficient. Recent years, many systems were developed either by the universities or the software companies in order to partially automate many of the processes carried out by the department. Those developments dramatically reduce the time take in searching information and should enable the school to maintain precise and up-to-date information. For example, previously, students would have to enquire for much of his information from the school student office, which required more time and effort, particularly from members of staff. The new system is more efficient and also often presents more accurate information. However, those systems vary even within the same university. Some are quite well developed and implemented, some still adopting the inferior and out-of-date technology. This is also one of the project's aims: to analysis some sample systems in order to attain the strengths and eliminate the weaknesses in developing the new potential student information system.

1.5 Structure and Milestone of the project

The report is split into five separate chapters, references and appendices. It is structured to reflect how the study was attempted and to try to show the logic with which this project was approached. Indeed, the report is largely in an order of how the study was completed. Chapter 2 of the report discusses the primary researches that were carried out in order to proceed with the project. It explains and justifies the manner in which this project was approached. Chapter 3 details 3 case studies carried out during the investigation phase of the project of 3 different current Student Information Systems. Their functionalities were compared and strengths and weaknesses were analyzed. Chapter 4 shows the product of the project, which is the detailed requirement specification of the new Student Information System. That chapter also discusses how it was derived from the previous investigation. Chapter 5 is the final chapter to finish the report; it shows the evaluation of the outcomes of the project, which includes identifying the weaknesses in processing the project and providing further enhancements to the system development.

The overall objectives of the study can be broken down to produce a set of milestones that become achievable as the project progresses. The first milestone will be the background research phase which clears which system develop methodologies will be used throughout the system requirement modelling and how this can be achieved (all dealt with in Chapter 2). The next milestone is the actual system requirements capture phase (Chapter 3). Then comes the outcome phase to show the product of the project (Chapter 4). Last comes the analysis phase, which dealt with the systems analysis as well as the outcome evaluation (Chapter 5). Detailed project schedule see APPENDIX A

Chapter 2. Background Research

2.1 Approaches to system development

The choice of system development approach will heavily influence the quality of the system being produced. Therefore, make a right choice of appropriate system development approach is vital before develop an actual system. Systems develop literature reviews showed two main approaches - the original hard systems approach (more relevant to technical, engineered systems) and the more recent soft systems approach (more relevant to human and social systems).

Hard system approach

Waterfall method and RAD:

"Waterfall" (also known as the Traditional method) divides the project into well-defined sequential stages with intermediate milestones. The final product is not delivered until all phases are finished. "Waterfall" method has many advantages, as it is the most direct way to the objective with the shortest development time and cost possible. However, the drawbacks of this method include little flexibility for scope changes, system limitations not being discovered until later in the development cycle, and clients not being able to see the product until it is completely finished (Bull, 1989).

In contrast to "Waterfall" method, RAD methodology (also known as the Spiral or Iterative methods) breaks the development process in cycles each containing multiple development phases (Stapleton, 1997). For example, often, 80% of the system functionality can be accomplished with 20% of the development effort during the first project cycle. The remaining functionality and improvements can be implemented during subsequent cycles. The advantages of RAD methodology include greater flexibility for scope changes, being able to identify limitations earlier in the development process, and to deliver partial functionality sooner than with "Waterfall" method.

OO Approach :

Object-Oriented (OO) technology has been heralded as a solution to the problems of software engineering. The claims are that OO technology promotes understandability, extensibility, resolvability, reusability, and maintainability of systems, and that OO systems are easy to understand and use (Bennett, 2002). It has been applied widely in the past for mapping from real-world problem to abstractions from which software can be developed effectively. Moreover, the object-orientation provides good conceptual structures, which can be used to deal with the increasingly complex information system.

System development methodologies domain model

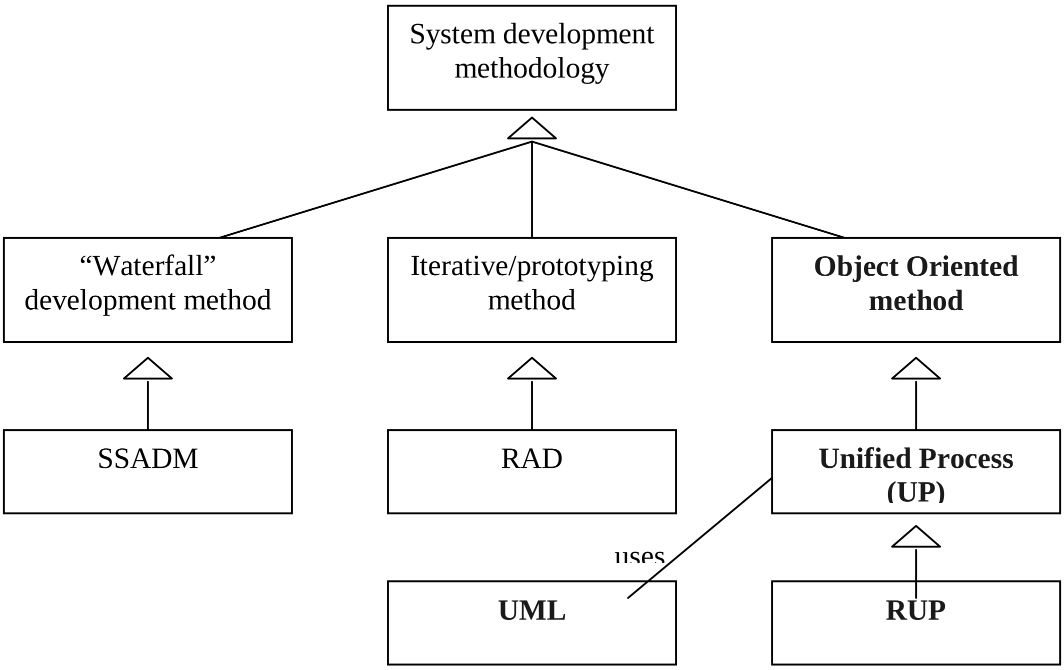

System development methodology
"Waterfall" development method
Iterative/prototyping method
Object Oriented method
SSADM
RAD
Unified Process (UP)
uses
UML
RUP

FIGURE 2-1 (Johnson, O, IS21 lecture 2 slides)

Soft System approach ---SSM

With soft system applications, system thinking should be regarded as a contribution to problem solving, rather than as a goal-directed methodology and this applies to all situations where the task itself cannot be entirely and objectively defined. The most famous soft approach is SSM (soft system methodology), which was first described by Peter Checkland (1981). Checkland argues for a cyclic learning system, which uses models of human activity systems to explore with the actors in a real-world problem situation their perceptions of that situation and their readiness to decide upon purposeful action, which accommodates different actors' perceptions, judgments and values.

Several people are currently conducting research into ways of overcoming the problems inherent with SSM. There is research ongoing at the University of Ulster dealing with the enhancement of SSM through Formal Methods and Risk Analysis techniques. A more practical approach is to use SSM to generate HARD (in Peter's point of view, HARD refer to the traditional way of system developing) questions, which can then be dealt with by the, more traditional, HARD methodologies.

2.2 Why use OO Approach?

In order to produce an actual system, an appropriate hard system methodology is necessary as well as the people centred soft approach. A right choice of both and apply them in the practice properly is vital to produce a suitable system that will enhance the productivity of the organization as well as the people in the organization after it is fully completed. Here in this project, I am required to do the generic requirement specification of student information system for university schools, which is the first and basic phase of the system development. As people said, ‘a good beginning leads to half of the success’ the right choice of system development methodology and proper beginning is crucial to the success of the whole system development as well as the project itself. Comparing those hard approaches discussed above, I summed up the criteria of the methodologies as below:

Criteria of Comparison	***Waterfall Method***	***RAD***	***OO Approach***
Responsiveness to environment	During planning only	Throughout the project	Throughout the project
Team dynamic and creativity	Limited	Unlimited during iterations	Unlimited during iterations
Final product determination	Determined during planning	Set during project	Set during project
Completion date	Determined during planning	Set during project	Set during project
Project flexibility	Structured	Semi-structured	Flexible
Communication in the team	Follow the plan rather than communicating	Communication during each iteration	UML enables sharing of information
Probability of success	Low	High	High

As the table shows, OO approach gets many better features than the others, though waterfall and RAD are also widely used system approaches. In this project, I am undertaking requirement analysis phase rather than the entire system development; therefore, the flexible OO approach will definitely be more suitable. However, the RAD theory of ‘MoSCoW rules’ (Stapleton, 1997) will also be adapted during the common user requirements selection. Method will be detailed in Chapter 4.2.6 Business context. The whole study will also take advantages of the SSM to involve users in the development phase in order to increase user acceptance eventually.

Another reason for choosing OO approach is that OO approach is also associated with various development tools. UML could be a very good example that can be applied in various system development environments.

One other successful software development is called RUP (*FIGURE 2-1*), which provides a central, common process definition that all software development team members can share, helping to ensure clear and unambiguous communication between team members (RUP, 2004). This helps the system developer to play the part expected of him in the project team by making it clear what his responsibilities are. Those tools enable better communication within the system develop team as well as eliminate the ambiguity. Therefore, RUP and UML will be adapted throughout the system modelling process.

2.3 What information is needed?

The information required to complete this project can be split into two parts. The first part is the modelling of the current student information system within University of Leeds and Shanghai university of Finance and Economics. That information will be the basis from which the further generic system requirement will be generated. Therefore the accuracy of the investigations of those targeted systems will be crucial to the success of the project and will continuously influence the further development of the system. Required information for current student information system can be further divided into three parts, they are: Information from student, staff and system's prospective respectively. In order to achieve this, various requirement capture techniques will be applied in the investigations for different kinds of required information, which will also be discussed in section 2.4.1- Fact-Finding Techniques

The second part of the information required addresses the objectives of this project. In order to review the proper strengths of each existing system followed by applying the improvement to the new student information system, researches into system analysis and system failures will be required. A comparison of the existing system is also vital for generating the generic system requirements

2.4 How will the information be found?

There are various fact-finding techniques available to carry out the investigations, however, in order to apply the most effective and efficient ways in the investigation as well as the research, those fact-finding techniques need to be studied and evaluated.

2.4.1 Fact finding techniques

There are five main fact finding techniques that are used by analysts to investigate requirement. They are background reading, interviewing, observation, document sampling, and questionnaires. The three existing systems that I have targeted at the beginning of the project have their own special characteristics; therefore, different fact-finding techniques will be applied to them respectively. (e.g. Geographic barriers eliminate me from face to face interview and observation for the system back China)

2.4.1.1 Personal use

Where possible, hands on the system gives the actual perception from the user's point of view. Personal use of the systems will be used to help identify the current functionalities available to that group of users as well as draw conclusions as to the strengths and weaknesses of the system. However, as a great deal of the data held is private and confidential, it is likely that permission will not be forthcoming to use some of the systems. Therefore, personal use is a valid method of discovering current system functions and citing weaknesses or strengths only for a particular system or function of a system, which lists students as its main users. In other words, this method could only provide information from a student's perspective.

2.4.1.2 Interviews

Interviewing is probably the most widely used fact-finding technique; a well-structured interview with a right member of staff will enable the necessary facts to be gathered. It can provide required information from staff's perspective. If I can find the staff who manages the current system, some system's technical details may also be gathered. However, it is also the one that requires the most skill and sensitivity, therefore, enough preparation before the interview is crucial to the success. Guidelines should also be followed. Bennett et al (2002) provide useful and detailed guidelines for interviewer to adopt before, at the start of, during and after the interview. Due to the distance barriers, some interviews should also be done through e-mail or telephone. Core questions will be asked, answered and understood for each system. Details about interviewees who were involved in the system investigation as well as the questions asked will be provided in appendix B interview scripts

Chapter 3. Case Studies of current student information systems

3.1 Introduction

In this chapter, the investigations of those three chosen systems will be presented respectively. During the unfolding of each system, details of technology, architecture, business requirement strength weakness etc will be listed. That information is mainly gathered from personal use and interviews.

3.2 Case Study 1 – SIS

SIS stands for School Information System, which is the main administrative database system within the School of Computing (SoC). All members of the SoC have the access via its main interface (web-based) intranet web pages at http://www.comp.leeds.ac.uk/cgi-bin/sis/main/index.cgi by providing their Linux username and password. The value proposal to the user is the effective and efficient way of broader range of information access. From the student's prospect, it also means convenient coursework submitting and results checking whereas, for the staff, it helps staff generate workload report and ease administration process.

3.2.1 Overview of the System

System Scope: The web based SIS is available for both staff and student of SoC to browse, and provides them with different information from any Internet enabled computer. The SIS and dynamic web page generation system interface with a number of existing systems, including university student information management system Banner, Edass and several administrative systems, such as those used to issue standard letters to students, the sorting of post and oversight of placement student are connected dynamically to the SIS database.

Technology: Dynamic web information system provides updated and accurate information to both students and staff. Modern IT technologies are used within the system integration and database management to provide as much information as the system can to the end users.

Stakeholder and User profile:
System users: Total head count of SoC is 852, which includes 514 full-time students. Stakeholder summary and user profile will be detailed in the SIS requirement specification in Appendix D.

3.2.2 Technical Architecture

Security: Users of SIS are issued a Linux username and password, which are intensely managed by the computing administrative staff.

Platform, Database and Development language

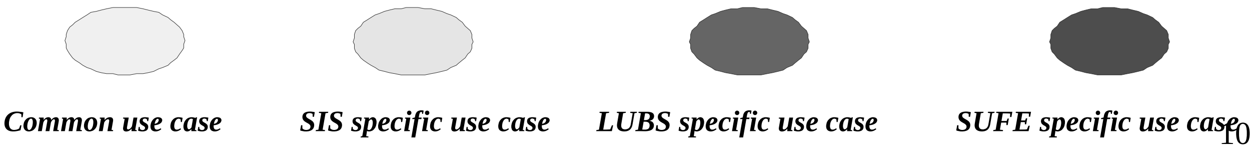
Name	*Architecture*
SIS – staff/postgrads/research grants	Ingres
SIS – taught student	MySQL
Student Handbook	Access
'Students'	Access
Placements	Access
Mail	Access
Edass (module assessment)	C program and data files
Yamp (degree assessment)	Perl scripts
Undergraduate admissions	Access
MSc admissions	Access
Postgraduate research admissions	Word
Research Publications	Access
192(staff and postgraduate directory)	Perl scripts and data files
Room booking	Microsoft Schedule+
School web site	Static HTML pages

User Environment:

The user will be accessing the system via a connection to the Internet. This is most likely to be from home or Lab, which are the excellent baseline, as this category includes the slowest types of Internet connection. However, the staff may normally use the system at work from the university's PCs and network while they are working.

3.2.3 Business requirements and System Scope

The following use case diagram (*FIGURE 3-1*) shows the types of users and their pursuing activities respectively. Activities are placed in different permission zones in order to identify the particular users' privilege of carrying out that activity. Students' use cases are detailed in the *FIGURE 3-2*.the detailed use case descriptions see APPENDIX D



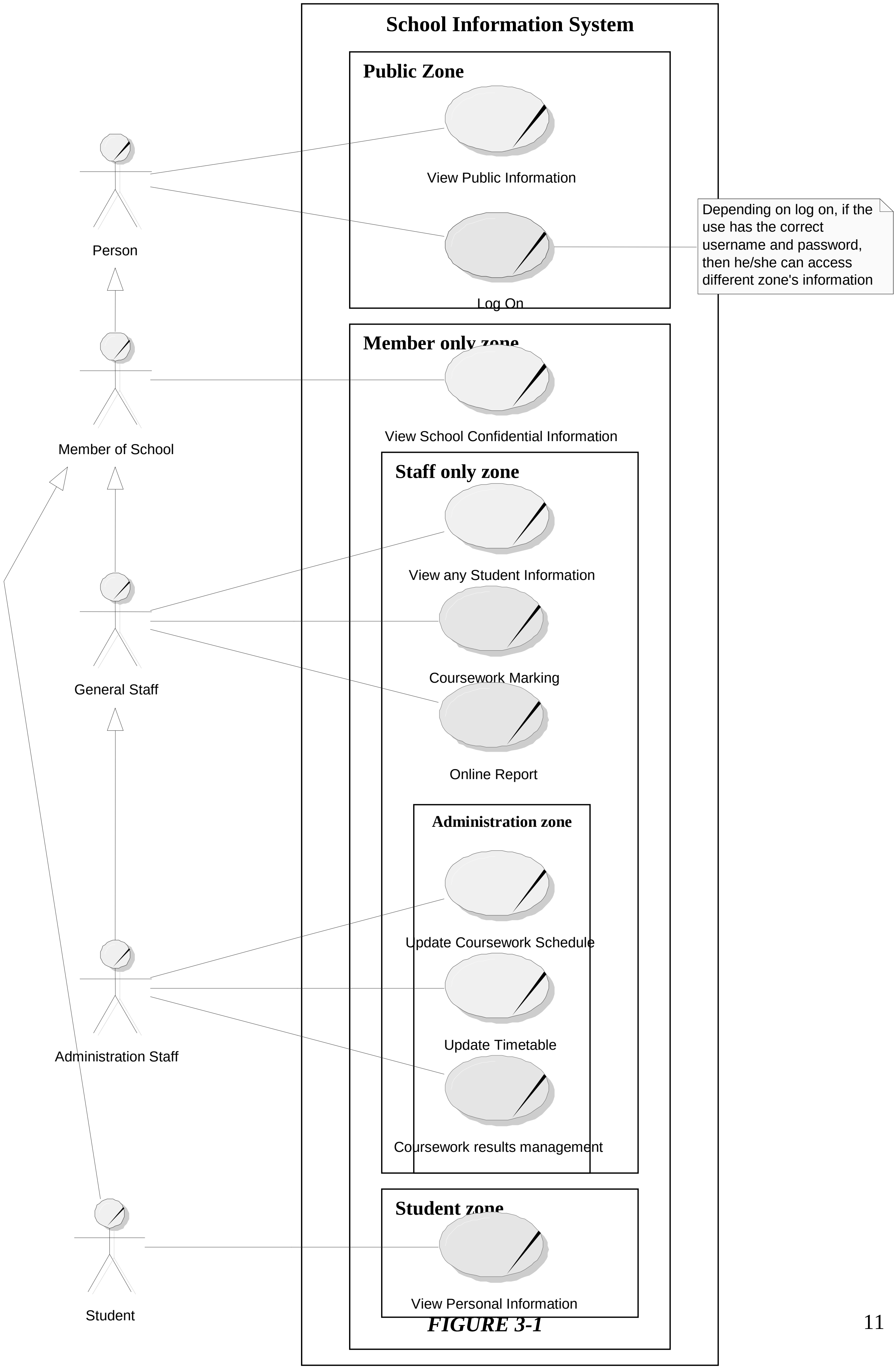

School Information System
Public Zone
View Public Information
Log On
Depending on log on, if the use has the correct username and password, then he/she can access different zone's information
Person
Member only zone
View School Confidential Information
Member of School
Staff only zone
View any Student Information
Coursework Marking
Online Report
General Staff
Administration zone
Update Coursework Schedule
Update Timetable
Coursework results management
Administration Staff
Student zone
View Personal Information
Student

FIGURE 3-1

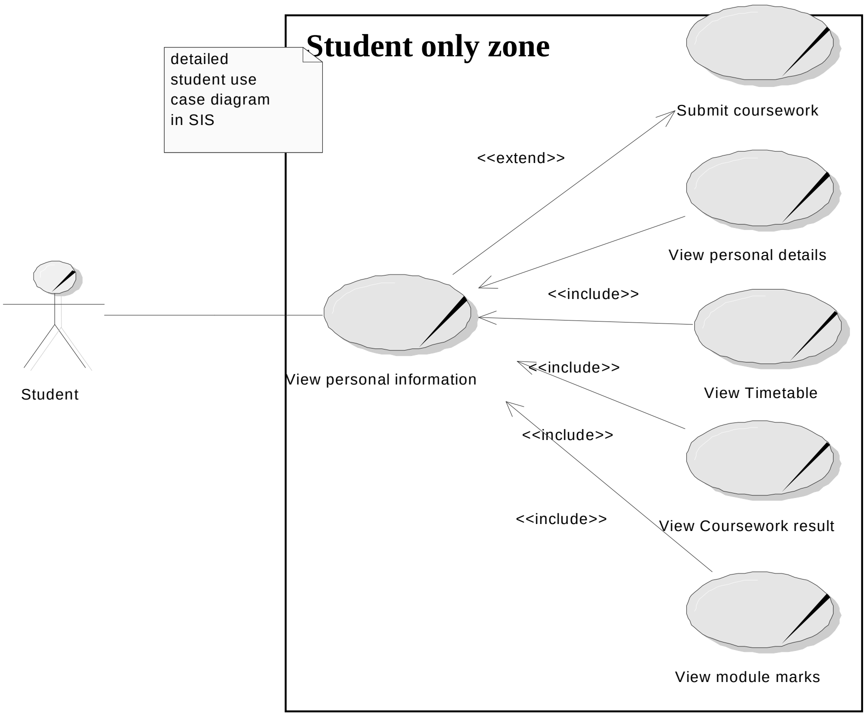


FIGURE 3-2

3.3 Case Study 2 – Systems at LUBS

LUBS is the one of the biggest departments within University of Leeds which has got over 1000 students and 111 academic staff and 53 support staff. However, there is no direct access for the students to get any personal or academic information online through the LUBS’s intranet. The only interaction between the students and the school through Internet is the university online facility called Nathan Bodington System (which will be discuss later in this chapter). Generally speaking, the current system within LUBS is highly confidential which can be accessed by the student office administration staff directly only, other academic staff use the Web for Faculty, which is provided by the university to access the student information.

The student information of LUBS is the third system that I investigated. The reason I discuss it here after SIS is because they have some similarities that I do not need to repeat myself.

3.3.1 Overview of the System

The LUBS system consists of several independent sub-systems (see the systems list in the architecture table). Till now, there is rarely a link between those databases or systems. They all operate separately within the department. Each system possesses one or more functionalities for a particular school activity. Joining those sub-systems, they form a large Student Information System, which covers most of the LUBS's activities as SIS does. However, student is not a part of the local system user, which brings a lot inconvenience to them. As a kind of university's local system, it also interacts with the university central student database through ODBC Link and uses a lot of the university's facilities such as Web for Faculty and Nathan Bodington System

3.3.2 Technical Architecture

Security: only member of staff can get access to the student personal information direct or indirectly through the LUBS local system. Each system operates independently thus provides high security. From the interview with the director of UG, I also found out that only 7 or 8 UG admission staff get the direct access to those local LUBS systems.

Individual system, database and development languages

The LUBS systems use very basic technology such as buy-in systems like Microsoft Access as its main system databases without public web based interface. The web site for LUBS is under construction.

Name	*Architecture*
Admission database	Gent
MBA student database	Microsoft Access
UG student database	Microsoft Access
Taught MSc database	Microsoft Access
Research Master database	Microsoft Access
Research student database	Microsoft Access
Nathan Bodington System	Java script and data files
Web for faculty	Java script and data files
Web site of LUBS	Java script and data files
Exam system	Microsoft Access and Spreadsheet

3.3.3 Business requirements and System Scope

The following use case diagram (*FIGURE 3-3*) shows the types of users and their pursuing activities respectively. Activities are placed in different permission zones in order to identify the particular users' privilege of carrying out that activity. Student use cases with Nathan Bodington System are detailed in the *FIGURE 3-4.* Detailed use case description see APPENDIX E.

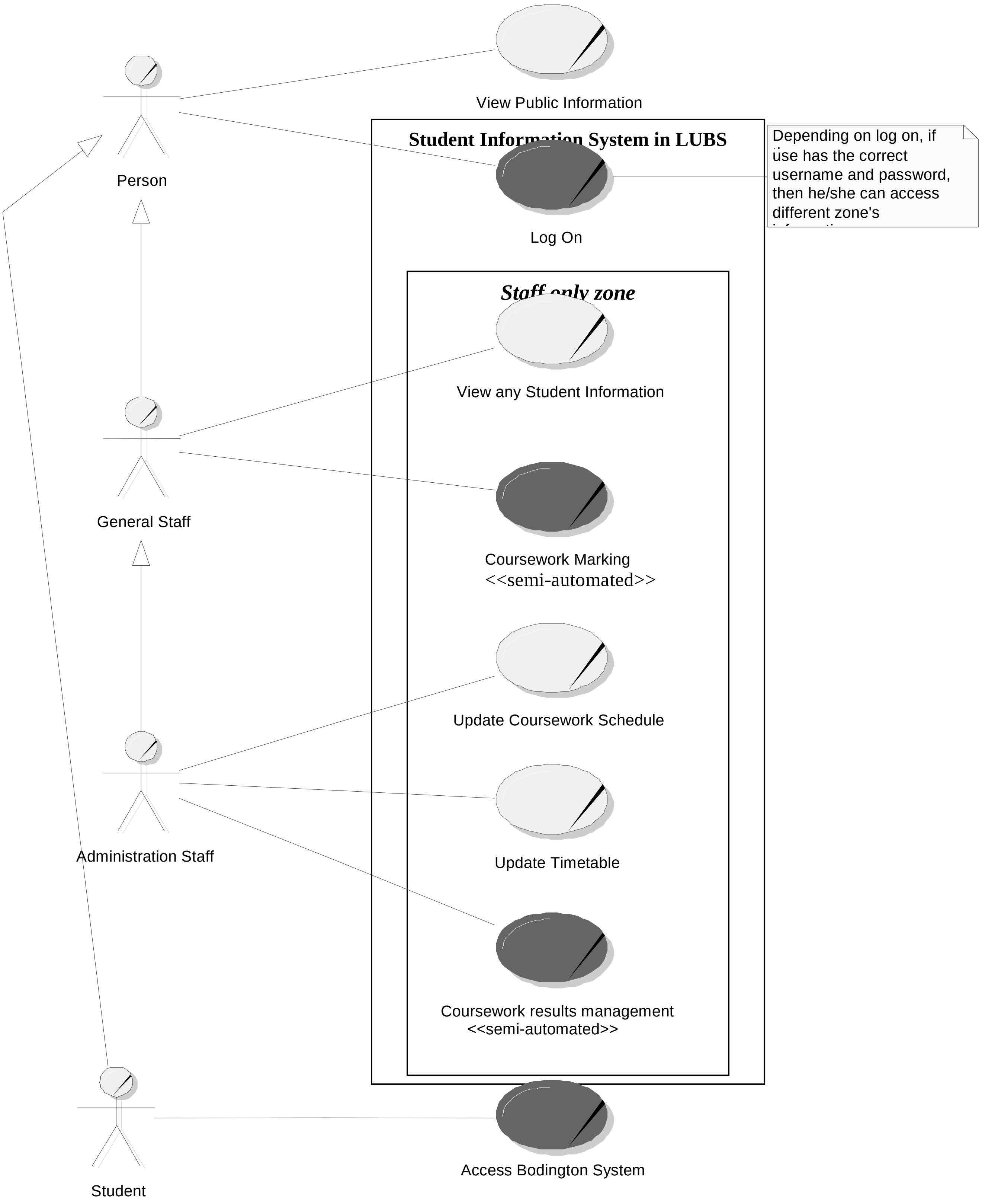

View Public Information
Person
Student Information System in LUBS
Depending on log on, if use has the correct username and password, then he/she can access different zone's
Log On
Staff only zone
View any Student Information
General Staff
Coursework Marking
<<semi-automated>>
Update Coursework Schedule
Update Timetable
Administration Staff
Coursework results management
<<semi-automated>>
Access Bodington System
Student

FIGURE 3-3

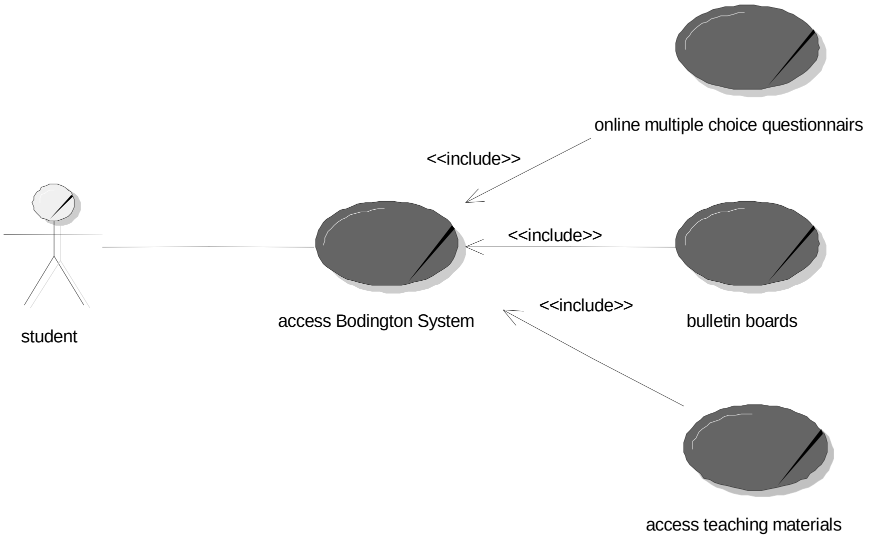


FIGURE 3-4

3.4 Case Study 3 - System at Shanghai University of Finance and Economics (SUFE)

All members of the university have been issued a username and password to get access to the internal university information system through http://www.shufe.edu.cn/. The system provides a university level administration due to the Chinese university structure, which will be detailed in APPENDIX F.

In contrast to those school level Student Information Systems, SUFE’s shows its speciality as a university level system which also provide some high level services such as module selection, major changing etc. Since all local system development should always match the pace of the central system development, the understanding of the university level system of SUFE is also necessary in designing the new Student Information System for school users

3.4.1 Overview of the System

The university central student records system maintains all students’ details as well as university general information forums. Although SUFE consists of colleges and faculties, there is no specific individual student information system dedicated for each college or faculty. Central database system manages all the information alone. Each department uses only a part of the system to get access to the central records

Each student of SUFE was issued a student account when they first registered. Currently the system provides more than 16000 student accounts. The student information system does not only provide student general information (e.g. personal details, module details etc) but also enable student to select and change both optional and elective modules. Student can change their modules in a certain period of time while the module register database is active.

1176 staff of the university access the system from a different interface, which is dedicated for admission; they can get access to all student data. This part of the system is relatively standard in China, since there are only two versions of staff admission systems in the market currently.

The system in Shanghai University of Finance and Economics is more flat than the general Student Information System in UK. In UK, most university manages their student records in two levels: school level (or department, faculty level) and university level (e.g. Banner), whereas in China it is simply managed by one big system. There are a few advantages and disadvantages with this kind of data management system, which will be discussed in 3.4.5 SWOT analysis.

3.4.2 Technical Architecture

Security: students and staff use their own username and password to log into the system though different interface. They have different privileges in accessing the central database. And any module selection or major change can only be done within a special period of time during the academic year while the databases are partially opened up allowing all students to select and change modules.

Platform, Database and Development language

Name	*Architecture*
Student web system	Java script, HTML
Central student database	Oracle and SQL
Admission system	Access
MSc admission	Access
Module selection	Java script and data files
Major change	Java script and data files
Apply external examinations	PHP

3.4.3 Business requirements and System Scope

As a university level system, it covers much wider range of activities than the two systems I mentioned previously, such as the functions of module selection and change across campus. Those functions can only be done at the university level whereas single department cannot hold all those enrolment information alone. However, because it is such an enormous system that cannot look after all aspects of each single department, (e.g. the formats of coursework among departments vary) it only provides the wide, general rather than specific information services to the end users. In other words, those coursework handling and marking are still down to each module leader in paper form. We can see that from the use case diagram (*FIGURE 3-5*) below. Detailed use case descriptions see APPENDIX F

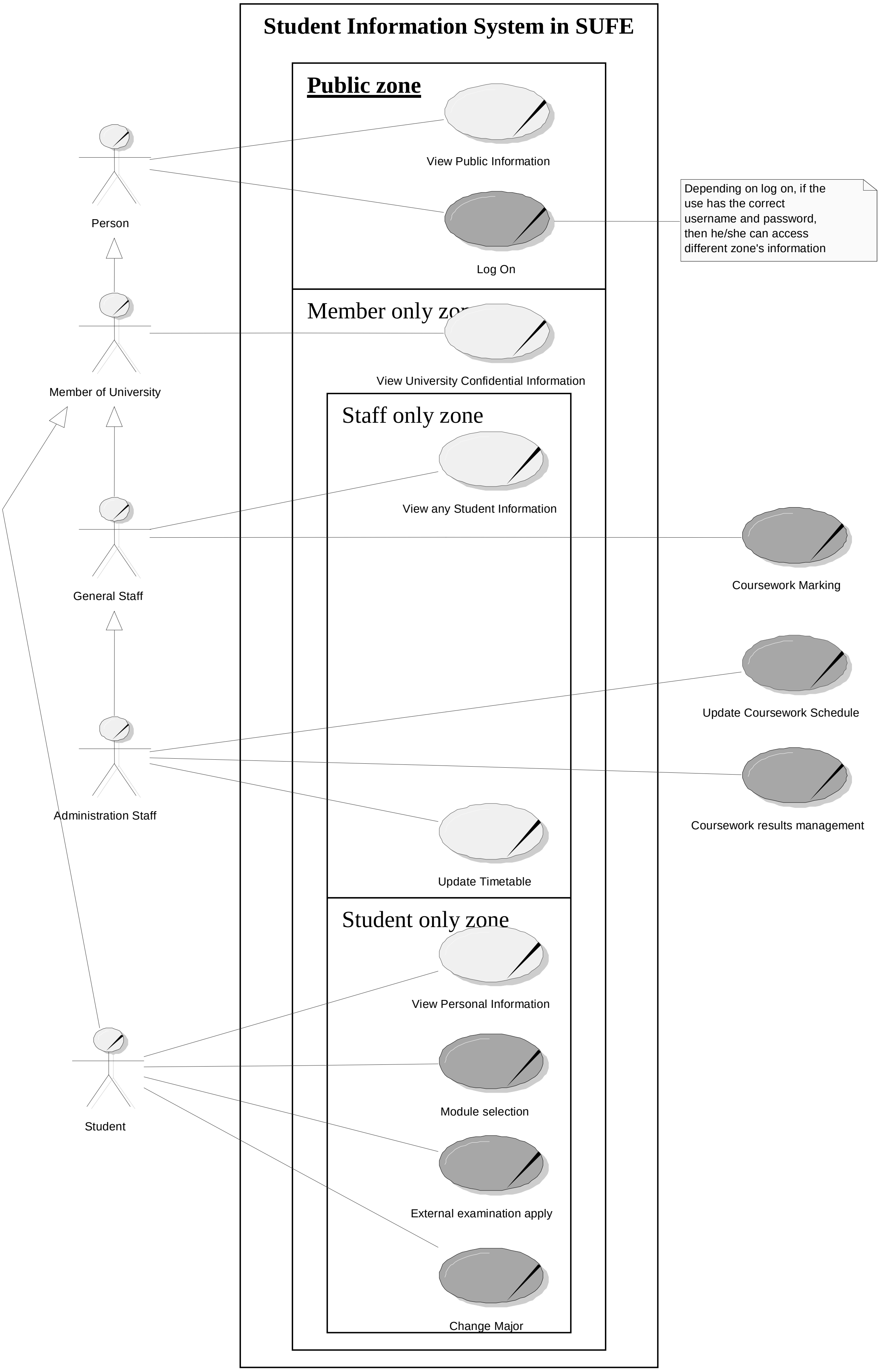

Student Information System in SUFE
Public zone
View Public Information
Log On
Depending on log on, if the use has the correct username and password, then he/she can access different zone's information
Person
Member only zone
View University Confidential Information
Member of University
Staff only zone
View any Student Information
General Staff
Coursework Marking
Update Coursework Schedule
Coursework results management
Administration Staff
Update Timetable
Student only zone
View Personal Information
Module selection
External examination apply
Change Major
Student

FIGURE 3-5

3.5 System functionality and SWOT analysis

All those three systems cover the basic system requirements such as maintaining students records, staff information, and general school information (marked as light blue colour in the use case diagrams) etc. However, most of the functions within them differ. Detailed system functions will be showed in the following table. (● means has, — means doesn't have)

System current functionality	**SIS**	**LBUS**	**SUFE**
Student records [undergraduate, taught postgraduate and research students]	●	●	●
Staff information	●	●	●
Research information [research publications and research grand]	●	●	●
General school information	●	●	●
Student Admission [UG and Post graduate applicants]	●	●	●
Information on all modules within school	●	●	●
Web based interface	●	—	●
Room Booking System [meeting rooms in SoC]	●	—	—
192 Facilities [contact details etc]	●	—	—
Performance statistics [No. of students registered, research publication…etc]	●	—	—
Module review back function	●	—	—
Student project information	●	—	—
Tutorial support	●	●	—
Electronic version coursework schedule	●	—	—
Electronic version module timetable	●	—	●
Electronic coursework submission	●	—	—
Electronic reporting system (e.g. equal opportunity report, self-checking report. etc.)	●	—	—
Nathan Bodington facilities	●	●	—
Student and staff forum	●	—	●
Online module selection	—	—	●
Online major change	—	—	●
Apply external examinations	—	—	●
Study plan for the whole university life	—	—	●
Class homepage facilities	—	—	●

SWOT analysis

The strength, weakness, opportunity and threat will be illustrated in the following table, by analysis and compare three systems, I will try to develop the generic student information systems on the basis of their strengths and correct the weaknesses.

SWOT	SoC	LUBS	SUFE
Strengths	*1.Flexible accessibility and much broader range of information can be attained at a time from the system.* *2. Historical records can be retrieved easily.* *3. Data integrity and consistency*	*1. Separated system provided highly secured environment. (Even academic staff get really limited access to the internal system directly).*	*1. Student data integrity and consistency* *2. Efficient module enrolment.* *3. Equal opportunities across all university students* *4. Various interactions and activities between university database and the end users* *5. Standardized system means regular update and improvement is made by the system developer*
Weaknesses	*1. Heavily rely on the database management; technical fault of server and database will lead to bad outcomes.*	*1. Duplicated data entry results in data inconsistency.* *2. Insecure data transfer between the systems (rely on disk or other media)* *3. Inconvenient data access special for student and academic staff.* *4. Limited resource available to the user* *5.Not functionally rich, lack of functionality. Many works are still rely on the paper based documents (such as timetables)*	*1. Security is the major weakness of the system. The various interactions are built on putting important data into risk e.g. module selection.* *2. Lack of specific functionalities for each department or faculty. Department specific activities still need to be undertaken by local staff by local 'paper file system' version and handwriting.* *3. During the module selection period, database management and security are critical to the maintenance work. In other words, it is the easiest dada vulnerable period.*
Opportunities	*To make the system functionally better in the future, the ability of provision of standardised report is needed.* *Fully integration with Edass is another goal. Edass is written in C, therefore can be accessed only from the Linux system. SIS development team are trying to make it fully*	*Great gap means the great opportunities to make improvement; many useful functions can be easily implemented.* *A good improvement will involve the student system interaction. A decent web based interface can provide student much more useful information and interactions*	*Good username and password control and security attention in module selection period will significantly improve the system's performance* *Great opportunities left for further development such as the ability to deal with coursework results management.*

	web based, which can be accessed by any browser. Future users will get more and more information from the system and staff can be able to monitor student progression through the system particularly.	Online timetabling and coursework schedule can be developed.	
Threats	The requirements from the university decide how SIS is going to organise its data. For example, this year all the module catalogues have been updated into the new format as well as the program catalogues followed by the request from the university. SIS is a sub-system in the university; the interface may change as the university system changes. Faculty issue. This seriously impact on the future of SIS. Since other members of the faculty join in, will they be allowed to use SIS or we need something else instead. Complexity itself is a serious security issue. As the system grows bigger and bigger, the structure within the system will get more complex, it will also ease data violation and vulnerability.	Traditional system thinking rely on the development of the university system may reduce the pace of self-development. Staff education will be critical in system security. Fast IT development may influence the whole system and many change them all. Same as SIS, the requirements change from the university may influence the local system completely. Data inconsistency is the most important factors that may cause serious error in either timetabling or coursework schedule	Large amount of user access may cause server crash. Standardised system version may raise serious security issue. Attacker can get detailed confidential system information. Loose password management may lead to unauthorised access and confidential information loses.

After investigating all those three systems and analysing each system's functionality, it is quite clear that SIS is outstanding among those three, which is developed most properly. The system of LUBS is categorized as a functionally average system after the investigation. There are two reasons that the word 'average' is given here.

First of all, it is found out throughout the investigation that most of the university schools within University of Leeds are using quite similar systems as LUBS does. From the conversations with some students from other departments such as school of sociology, school of mathematics, and school of biochemistry, etc, it is found that most schools are using the university web facilities (such as Nathan Bodington System) as their main interface with students rather than developing their own systems on the local databases. The resources from the central databases are so limited that they cannot be specified to school level (for example, it cannot provide coursework schedules and timetable for individual department)

Secondly, as the functionality analysis table showed in section 3.5, it is clear that LUBS's system is relatively rough since some business processes are still semi-automated and it is lack of functionality. Therefore, those individual school systems are categorized as an 'average' whereas SIS is functionally better.

SUFE shows its speciality as a university level system, which also provide some high level services such as module selection, major changing etc. Its special strengths can be learnt and built on top of other strengths discovered in the investigations.

3.6 Other Case Studies

After found out that university system facilities are still the most commonly used mediate with students within University of Leeds, it is very necessary to understand what those facilities are and how they work, which will be critical to the requirement of the Student Information System for potential university schools. Because what the university system facility does, is what the local school system is lack of. Therefore, a new investigation of Nathan Bodington System is set up.

Nathan Bodington System:

The Bodington system provides support for delivering online learning to the students of the University of Leeds. The overall aim of the service is to enhance teaching at the University by providing the teachers with the mechanisms they need to effectively deliver e-Learning.

Bodington provides a structured and secure environment for teaching materials, peer to peer and peer to tutor discussion (bulletin boards), online submission of coursework, online testing using multiple choice questionnaires, etc.

Taking a student of SoC as an example, the system deployment diagram (FIGURE 3-6) shows the systems interactions between SIS, Banner and Bodington facility

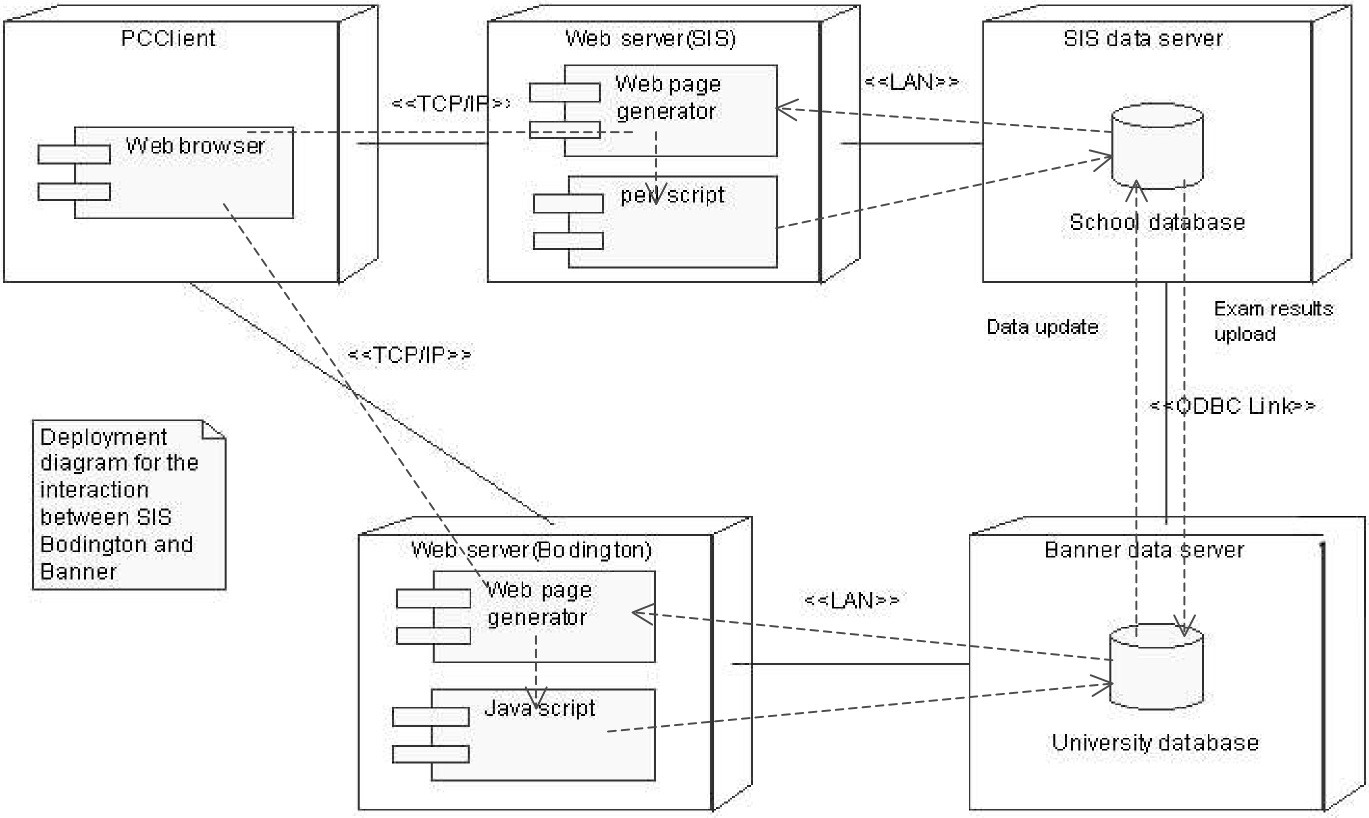


FIGURE 3-6

SIS is based on a post-greSQL. It is a single database. The main way that Banner interacts with SIS is through the script running automatically nightly. Data is imported from Banner then there is a separated program comparing the difference between those data. If any difference is found, then the data of SIS will be automatically changed to follow the Banner's. Therefore, any changes made in Banner on students of SoC will update SIS's records nightly. On the other hand, SIS also informally updates Banner with the exam results. So it appears as a two-way communication in the *FIGURE3-6*.

Any student registered in University of Leeds has the access of Bodington facilities through the Bodington system server. After they entered the valid university username and password, all the resource there are free to access.

3.7 Summary

This chapter is concentrated on the 3 case studies, which illustrate the scope of each system, technology they used, system requirement and functionality, and the SWOT analysis of each. An extra system investigation of Nathan Bodington System is also showed. From those case studies, we can see that each system got its own characteristics which different from others. SIS is definitely functionally better than LUBS's. The one of SUFE's is quite special as a university level system, but its distinctive functions my also be learnt and applied in the future school Student Information System such as the online module selection. Although those systems illustrate three totally different architectures in different education systems, some basic functions remain the same and most of common academic activities are covered, which will be the basis to generate the generic student information system for the potential university schools.

Chapter 4 Generic Student Information System

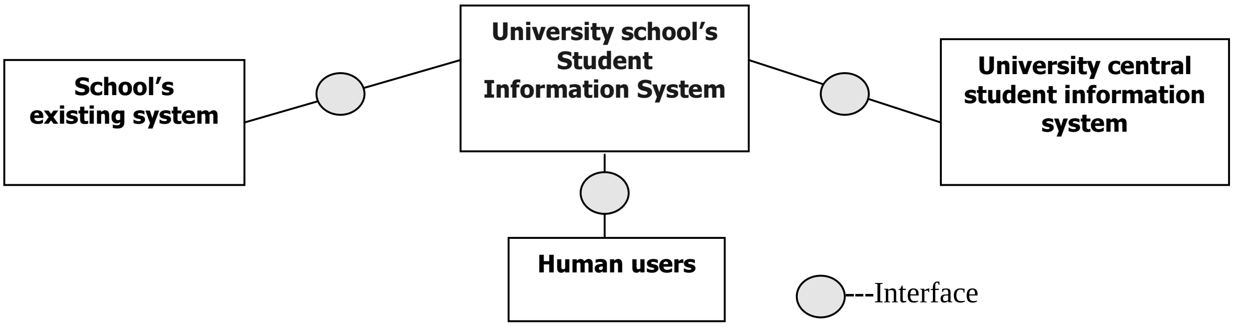
4.1 Introduction

This chapter explains the technical architecture and requirement specification for Student Information System within potential university schools. As a product of this project, how it is generated from the previous investigations will be further discussed and followed by the evaluation.

4.2 Requirement specification of Student Information System

4.2.1 System Scope and Architecture

This system modelling and requirements analysis apply to the university school's Student Information System, which is going to be developed by Aquilo. The system will be installed and maintained at the university school level. The web based Student Information System will be available for both staff and students of that individual school to browse, and will provide them with different information authorizations from any Internet enabled computer. The reason of chosen web based interface will be detailed in 4.2.5 System Technical Scope. Here is a diagram of Student Information System's position within a university.


University school's Student Information System
School's existing system
University central student information system
Human users
---Interface

The Student Information Database and Dynamic Web page Generation System will interface with a number of existing systems, including university central student information management system (e.g. Banner), and school's local existing systems.

4.2.2 Positioning - Problem Statement

The aim of the system is to enable the university schools to provide a level of services to both staff and students from which they would benefit. This includes an extensive student information access, which provides student personal details, coursework schedules and results, module details, timetables etc. Ensuring that students and staff are well informed. The system also stores and maintains the student information as well as coursework, which can directly reduce a lot workforce involved.

Efficiency and quality of web based service; the Student Information System will provide a reliable information service to the students and staff within that university school based on a central, powerful relational database with data updated frequently by other systems such as university's central student records system (e.g. Banner)

4.2.3 Stakeholder and User Descriptions

This section describes the users and the stakeholders of the student information system.

Stakeholder Summary

Name	Represents	Role
Head of the School	University School	Responsible for project funding approval, and monitoring progress of the project.
Database Administration and Software development Manager	Database Admin. And Software development Team	Ensures that the system is integrated with the database effectively, upholding data integrity. Also involved with providing content, and developing and updating the System, as well as site maintenance.
Director of UG studies	UG teaching staff	Responsible for all the under graduation studies affairs including the student records approval
Project Manager	Aquilo system develop Team	Primary role is integration of the different facets of the development project.

User Summary

Name	Description	Stakeholder
Student of School	Uses the system via a web browser to access student Info.	Head of University School
Director of UG studies	Uses the system to manage coursework results and generate staff workload report	Self-represented
General Administration	Use the system to help acceptance and clearance process. Coursework schedule and timetable updating	Director of UG studies
General staff	Access any student records	Director of UG studies
Database Administration	Involved with data integrity and database performance issues, allowing seamless use of system.	Database Administration Manager

User Environment

The user will be accessing the system via a connection to the Internet. This is most likely to be from home or labs, which is an excellent baseline, as this category includes the slowest types of Internet connection. However, the staff may normally use the system at work from the university's PCs and network.

In order to achieve flexible system access and broader range of information at the same time, web based system interface was chosen. However, issues of compatibility between different computer types still exist, although testing of the web site will have to be carried out over a number of different web-browsers. (e.g. Netscape, IE)

User Needs

Need	Priority	Concerns	Current Solution	Proposed Solutions
Effective and Efficient student information Administration	**High**	Mass Information management is difficult and time consuming	Require intensive data entry by staff and paper work.	User 'Self Service' approach using Internet based central database information distribution system.
Database Integrity, and efficient accessibility.	**High**	Data can be lost due to security breaches, improper database design and hardware failure.	Firewall and virus detection software.	Firewall and virus software, coupled with regular data backup, use of mySQL and sound internal password management schemes.
Information accuracy	**High**	The wrong information may result in serious mistake (e.g. wrong classification)	Human data input and paper record system involves certain human mistake.	Update daily from Banner No body can change them while they are in system except certain position staff.
24 Hour Internet Access to student information system	**Medium**	Flexible access whenever and wherever you are	Hand book or go to the school reception	24/7 system access provide by the system

4.2.4 Product Overview

Product Perspective

The system will reside on a web server, which will send data to users on request via the World Wide Web. This will be integrated with a Database Server, which is capable of managing two databases individually. One for storing information on taught students; research students, research grants, staff, admissions enquiries, and module etc. the other one is used to stored electronic coursework and coursework marks. They are interlinked, however, only the first one provides information to the students through dynamic web page generation technology (e.g. perl script or PHP) whereas the second one is confidential and staff only. This is really for security concern. The following system deployment diagram (***FIGURE 4-1***) illustrates the design

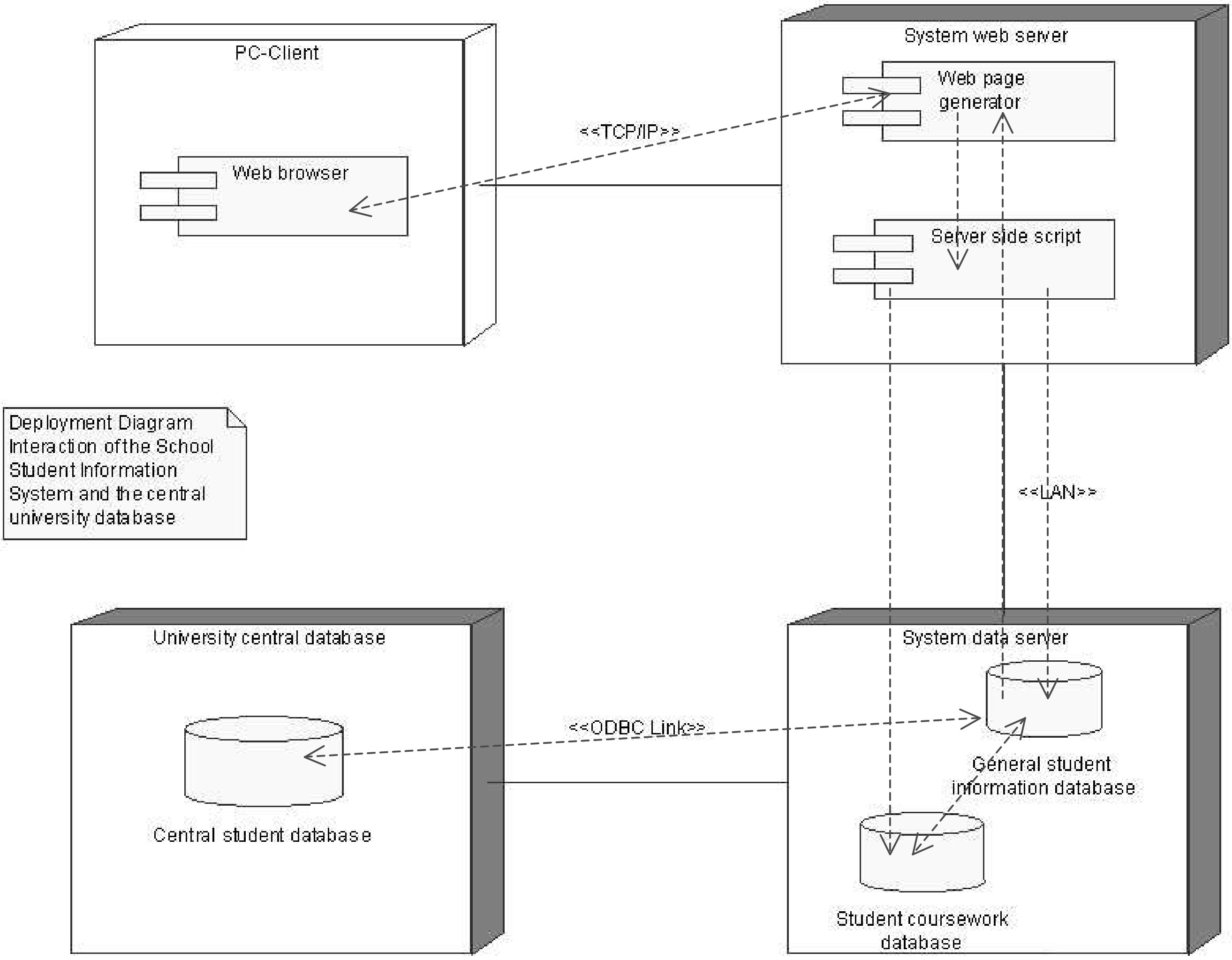

PC-Client
Web browser
<<TCP/IP>>
System web server
Web page generator
Server side script
Deployment Diagram Interaction of the School Student Information System and the central university database
<<LAN>>
University central database
Central student database
<<ODBC Link>>
System data server
General student information database
Student coursework database

FIGURE 4-1

Users will be required to input their username and password in order to access the system. The university central student record system will update student records regularly. Therefore the student records in school student information system are monitored and refreshed which any change has been made.

Summary of Capabilities

Customer Benefit	Supporting Features
24 Hour system access	Student and staff of the school can access the web-server of system anytime from any place. Server request central relational database to retrieve user requested information.
Great range of Information	Centralized relational database holds information on almost all aspect of student and school.
Fast, Efficient Submit	Fast intranet upload to the database through web interface
Latest Results	Coursework results update as soon as it is recorded.
Economic	Cost saving in human resource management.

4.2.5 System Technical Scope

Elements within the scope of this system deal with aspects of the services provides by student information system. This includes coursework submitting, view student personal details, timetable management and coursework management. Advanced functionality online report, which can be implemented, but will only be modelled at the abstract level in this specification. In order to provide all those dynamic interactions between system and user, web-based interface was listed as first choice. Although other methods such as client-server and UNIX can also be adopted, however, after evaluating all the advantages and disadvantages of each and from the investigation of current reality samples, web-based interface is definitely much more effective and easier to implement and use. Furthermore, by using the web-based interfaces, some good architecture strengths can be adopted from my previous investigations.

Interface	*Interface Type*	*Interface Method*	**Description**
Any visitor to System	Human – System	Web browser	Any people can view school public information via a web browser using an Internet connection.
Student to System	Human – System	Web browser and login	Students will have a password to log into the system, then they can access the school internal student resources
Staff to System	Human – System	Web browser and login	Staff will have a password to log into the system, then they can access confidential staff only resources and services
Administrator to System	Human – System	Web browser and login or direct login	Administrators will have the high privilege for student coursework database maintenance as well as timetable or coursework schedule update.
Database to dynamic web-page	System – System	(e.g. perl script)	Unique web-pages can be created from the database providing relevant information to individual user
School student records database to university central database	System – System	(e.g. ODBC link)	University database regularly updates the local database ensure student data consistency and information integrity
General student information database to student coursework database	System - System		The reason to have two-separated databases is for security concern as well as ease of database maintenance. The electronic version of coursework will be stored in the coursework database before marking

4.2.6 Business Context

School student information system is developed to provide high-level information service to students and staff of that particular university school.

Great amount of information available	Centralized relational database stores great amount of information
Accurate information	Updated and checked by module leader and system administrator, system update script to ensure the data consistency with university central database
Flexible information access	24/7 any time any place.
High security	Only staff can access the electronic coursework but no modification can be made after submitted
Reduce formalized workload	Coursework submitted electronically
Access from the comfort of home or lab.	Via connection to World Wide Web.

- Costs associated with staff required in help desk or reception can be reduced, moreover, cost of paper work will be significantly reduced.
- Information administration and paperwork will be minimized, with details on users stored in computer databases.
- Integration of the existing system (e.g. university central student admission system -- Banner) will make sure the information is updated as soon as any changes occur.

In the following user requirements defining part, I will adopt the RAD theory of 'MoSCoW rules' (Stapleton, 1997) to distinguish between what the users 'needs' in a system and what the users 'would like'. User requirements are divided into four categories depending on their relative importance and practical use in the proposed system. The first category is 'must have' requirements which states "are fundamental to the system". They are all the common use cases from the 3 case studies. The second category is the 'should have' requirements, which are important requirements, but the system would still be useful and usable without them. The third category is the 'could have' requirements, which would add value to a proposed system, which could be left out of the basic development process. The fourth category is 'want to have' but will not have this time around. This is less vital to this particular project because the size of the final system would mean that all requirements probably could be met which has been demonstrated in the first three categories.

Business Use Case Diagram

The following diagrams (*FIGURE 4-2, 4-3*) show the main users involved in the system and the major activities carried out by them. Detailed use case description and the sequence diagrams of how each activity takes place will be presented fully in the Appendix C.

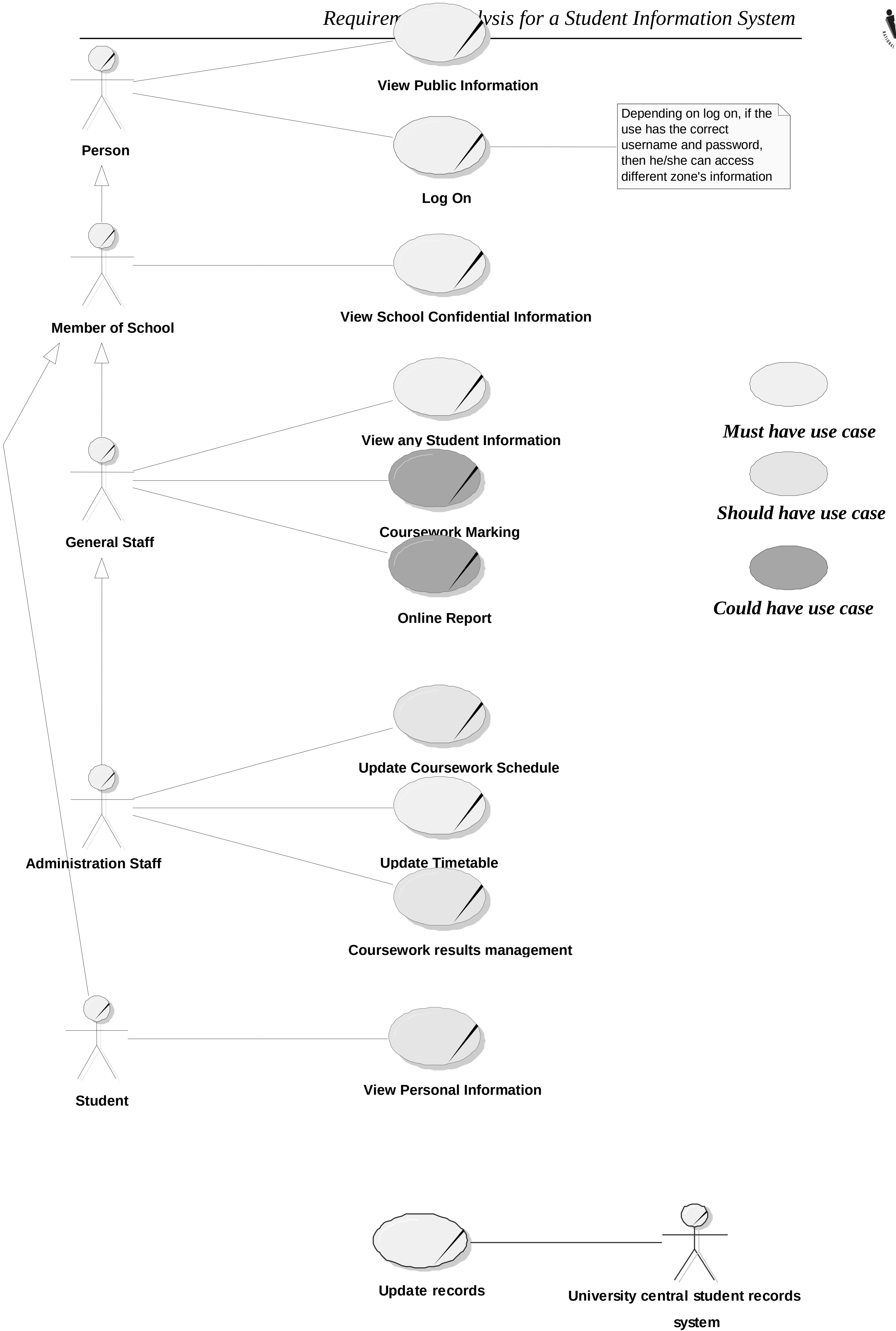

View Public Information
Person
Log On
Depending on log on, if the use has the correct username and password, then he/she can access different zone's information
Member of School
View School Confidential Information
General Staff
View any Student Information
Coursework Marking
Online Report
Must have use case
Should have use case
Could have use case
Administration Staff
Update Coursework Schedule
Update Timetable
Coursework results management
Student
View Personal Information
Update records
University central student records system

FIGURE 4-2

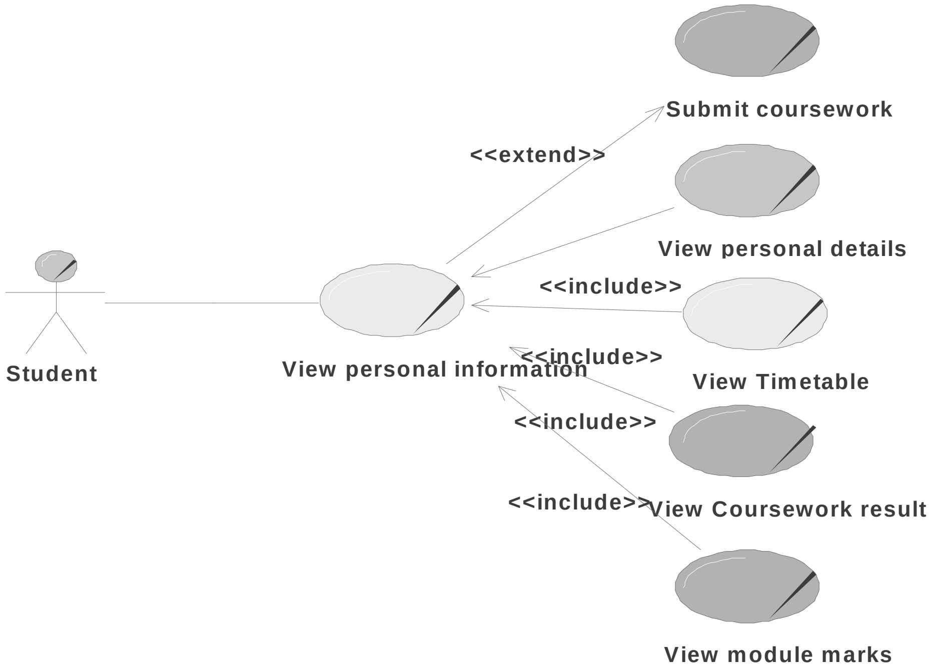


FIGURE 4-3

As *FIGURE 4-3* shows, student users of the new Student Information System must be able to at least view their personal details; they should be able to view the timetable as well. Based on different situations in different school, for example, various formats of the coursework required, they may be able to submit it electronically (if the school accepts electronic version coursework). The other enhancements would be the functions to view coursework results and module results, which will require the system not only recording the individual coursework results but also calculating the final results by combining the coursework and exam results.

Class Design

The Class Diagram (*FIGURE 4-4*) shows the system from a technical perspective, allowing software engineers to start developing the system.

9 main classes have been identified which are Timetable, Module, Enrolment, ModuleResult, Coursework, CourseworkSchedule, CourseworkResult, UniversitySchool and Member. (Student and AcedemicStaff are the sub-class of Member).

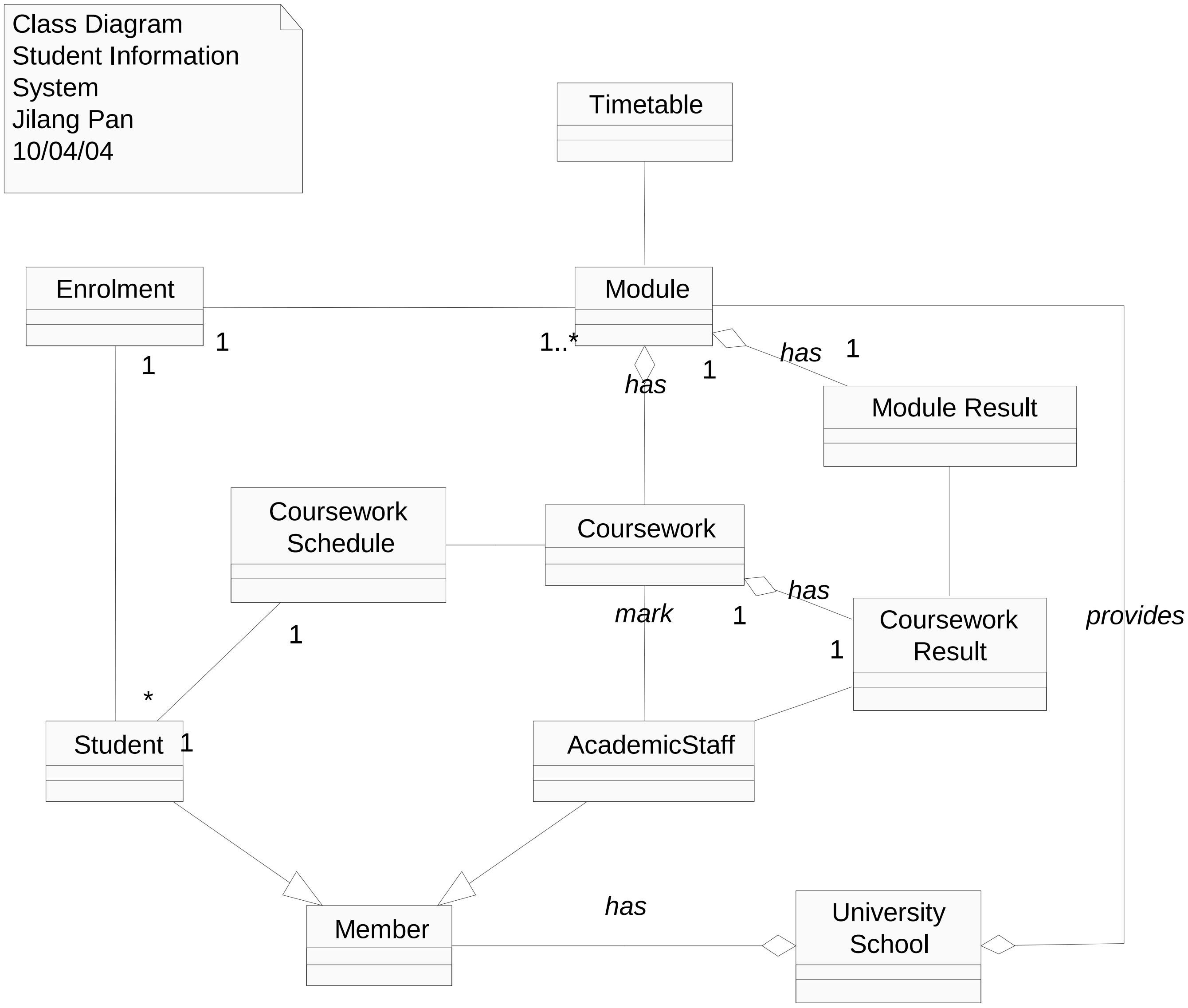


FIGURE 4-4

4.3 Summary

This Chapter shows the detailed requirement specification of the new Student Information System. It is built on the strengths of those investigated and trying to correct current system's weakness. The final product of the project will be the foundation of the future Student Information System development. It provides detailed user requirements as well as system architecture, which ease the further development work to be carried out.

Chapter 5 Evaluation

5.1 Introduction

This chapter looks at how successful this project has been in terms of meeting the initial objectives and requirements of the study. The final milestone of the report is to evaluate the investigation as well as the new Student Information System for potential university school and suggest areas of further development from these conclusions.

5.2 Acceptance criteria

As the objectives indicated, the product of this study is the requirement capture and analysis rather than system design and implementation. Therefore, we cannot judge directly from the outcomes from the first phase of the system development whether the development is success or not. Great a lot of work is left to the future system developers and programmers. A more sensible way of evaluate this kind of product will be assessing the way of carrying out the investigation and analysis in order to find out whether theoretical methodologies are appropriately used in practice and whether it can be improved. Adapting the evaluation, those criteria are firstly defined:

- **All objectives and minimum requirements are covered --**go through the project, whether the outcomes of the project have matched the initial objectives and minimum requirements.
- **User acceptance –** present the product to the potential end users as well as the system develop team (Aquilo), get advices and suggestions from them in order to make system enhancement
- **Functionality –** system functionality need to be analysis. A comparison of the potential Student Information System and the current average system will illustrate the system strengths as well as opportunities for further development
- **User involvement –** crucial part as discussed in chapter 2 (SSM), exam whether this success criteria has been satisfied throughout the requirement analysis phase and advices on continues adopting both hard and soft system development methodologies.

5.2.1 what has the project achieved?

The initial objectives and the requirements has not only be met but also extended during the process of the project.

- **Investigate what the current Student Information Systems of university schools are**

Investigations were completed at the beginning of the second semesters. During the investigation, three school information systems and a university system facility named Bodington System were chosen and fully investigated through both personal use and interview. As a joint honour student, I am quite familiar with both School of Computing's SIS and LUBS's system, so they are definitely my first choices because I was quite confident that I would get most information from them (since I can get access to both of them). Moreover, my friends are studying in Shanghai University of Finance and Economics; it also eases me to get some information from their university system. After four months investigation, three detailed system requirement specification were produced (see Appendix D). However, the investigation for the Student Information System in LUBS was not as fluent as I expected. Many admission staff had rejected me that the system is highly confidential. Fortunately, after the negotiation with the Director of UG, an interview was set up finally, which provided me some internal and useful information of the systems in LUBS. It enabled me finishing the LUBS's system modelling eventually. From this event, I have learnt that it is very important to get the clients' (university schools) permissions before targeting the place you want to investigate. This may cause investigation failure if you choose the wrong one.

The extra system investigation of Nathan Bodington System suddenly came out my mind after all those scheduled researches. I felt the necessity of understanding this university level education facility system while I was analyzing the captured system requirements. As explained earlier (Chapter 3.5), most schools make use of Nathan Bodington System as their main interaction mediate with students.

- **Produce a detailed requirement specification for the potential new Student Information System for general university school**

Found in Appendix C, a detailed system requirement specification for the potential new Student Information System for general university school is produced. A detailed system architecture and initial class design are also introduced. This will be a useful basis on which the further system development will be carried out.

5.2.2 User acceptance

User acceptance test is another part of the evaluation plan. Since the potential users of the Student Information System will be both students and staff, evaluation should be done from both sides. Their perception of the system is extremely crucial to whether they will use them or not. Therefore, a few students and two lecturers from LUBS were chosen to help me evaluate the system requirement and architecture. Since they do not understand the technical terminology of the way the project is undertaken, a brief presentation about the system was given and each system function was explained. Their opinions were collected after the presentation:

All of them are very happy to see the system to be developed, web-based system interface is very desirable among the students, and they are expecting the system to provide them more information about both school and themselves. Lecturers are more interested in the timetabling and electronic coursework schedule and submitting. One suggestion has been made by one of the student is that: the timetable interface can also link to the module web site. When you click on the module code in the timetable, the correlated module information will be displayed. Therefore, there can be an integration of module information, timetabling and coursework schedule to corresponding module.

Another problem was identified by my assessor during the progress meeting that, if more current systems investigations were undertaken, it will result in less use cases in common and more difficult to group those use cases into 'Must have' 'Should have' and 'Could have' categories. It is true in reality that those system functionalities greatly differ, however, their serving purposes are always the same—make the process more effective and efficiently. The process differences doesn't mean the use cases differ, therefore, as long as the project not exceeding the time constraints, the more systems you investigated, the better results turn up.

Finally, the requirement specification has been sent to the Aquilo system development team for their evaluation. Positive feedback was given and they are quite happy to take it as a basis to develop an actual Student Information System on. The acceptance of Aquilo which is also a part of the project requirements has been achieved.

5.2.3 Functionality

As the first phase of system development, system strengths should be built from the beginning. The functionality analysis was presented in the following table, which containing the comparison of the 'average' (discussed in Chapter 3.5) Student Information System currently used within the University of Leeds and the 'New' system for potential users.

Functionality analysis (•--has, — does not have, √ -- could have)

System current functionality	**‘Average’**	**New**
Student records [undergraduate, taught postgraduate and research students]	•	•
Staff information	•	•
Research information [research publications and research grand]	•	•
General school information	•	•
Student Admission [UG and Post graduate applicants]	•	•
Make use of University facilities	•	•
Information on all modules within school	•	•
Tutorial support	•	•
Student project information	—	•
Web based interface	—	•
Electronic version coursework schedule	—	•
Electronic version module timetable	—	•
Electronic coursework submission	—	•
Electronic reporting system (e.g. equal opportunity report, self-checking report. etc.)	—	√
Online module selection	—	√
Online major change	—	√
Student and staff forum	—	√
Study plan for the whole university life	—	√
Performance statistics [No. of students registered, research publication…etc]	—	√
Module review back function	—	√

Those shaded functions with a √ show the flexibility of the new Student Information System that varies with the different university school’s different requirements. It could also be the further enhancement of the system development.

From the analysis above, it is clear that the new Student Information System is functionally better than the current system in University of Leeds. Those flexible functional options can be applied depend on the potential user requirements.

5.2.4 User involvement

When new Student Information System requirement specification were developing, the end users were involved in the development cycle. For example, interviewee's opinions toward the system improvement were always asked throughout the interviews. Their perceptions of current system functionalities and ideal way of business process contribute to my system design results in system enhancement. However, it is not enough in the new system development cycle. As Checkland (1981) suggests, user involvement is crucial throughout the system development life cycle. The specification of the final system modelling may have caused me to develop the system primarily in a way that I thought was best (such as the categorizing the MoSoCoW use cases), rather than in the way the user thought was best. Therefore, it will be much better if a survey of system functionality analysis could be done among the users (staff and students) to reflect which system functions are most desirable. It is also very important to keep take user involvement issues into account in the future development.

5.3 Summary

In this Chapter, evaluation was adopted firstly through the product development in terms of meeting the initial objectives and requirements. They have not only been met but some of them are exceeded. Followed by assessing the studies by other three success criteria: user acceptance, functionality and user involvement. A few problems were identified and further enhancement was introduced. The reflections from students, lecturers and Aquilo system development team are positive and very encouraging. To sum up, the first phase of the system development – requirement capture and analysis is successful and the further development is promising.

References

Avison, D & Fizgerald, A (2002)*"Information systems development: methodologies, techniques and tools",* McGraw-Hill.

Avison D.E & Shah H.U. (1997) " *The information systems development life cycle: a first course in information systems",* McGraw-Hill.
Stapleton J. (1997). "*DSDM, Dynamic Systems Development Method"*. Addison-Wesley.

Bennett, S. McRobb, S and Farmer, R (1999) *"Object-Oriented Systems Analysis and Design using UML"* McGraw-Hill.

Bull, M (1989) *"Systems Development using Structured Techniques"*, Chapman and Hall Computing.

Checkland, P. B. (1981) *"System Thinking, Systems Practice*" john Wiley and Sons Ltd, Chichester, West Sussex.

Dix A, Finlay J, Aboud G & Beale R, (1998)*"Human Computer Interaction",* Prentice Hall Europe.

Fores, S *"Title from Previous Years"* http://www.comp.leeds.ac.uk/fyproj/previous-years.html [10/12/03]

Hughes, B and Cotterell, M (1999) *"Software Project Management"*, McGraw-Hill.

Laudon, K. C. and Laudon, J. P. (2002) *"Management Information System: managing the digital firm"*, 7th Edition, Prentice Hall.

Johnson, O (2002) University of Leeds, *"IS21-Topic1 course Not*es" UK.

Lucas, H (1995) "*Information System Concepts for Management*" 3rd Edition, Macmillan Publishing, New York.

Martin, C and Powell, P (1992) "*Information System: a management perspective"*, McGraw-Hill.

Rational Unified Process (RUP)(2004) *"C:\ProgramFiles\Rational\RationalUnifiedProcess\index.htm*" SoC, University of Leeds, UK. [10 January 2004]

Sauer, C (1993) "*Why Information Systems Fail: A Case Study Approach"* Alfred Waller Ltd, Henley-on-Thames, Oxfordshire.

Simon Bennett, John Skelton, Ken Lunn.(2002) "*Schaum's outline of UML"*. UK.

Appendix A

The project offers me a first chance to put my theoretical knowledge into actual system developing which give me a great sense of achievement after completing the project. It has also been an interesting experience for me to undertake such a large-scale investigations. I am pleased with the outcomes as well as experiencing the duration of solving encountered problems.

The project as a whole went well. The initial project plan and schedule turned out to be suitable for the project undertaken. Looking back on the project after completion, I have gained substantial knowledge in system modelling and in putting RUP into practice.

I have learnt a lot through completing this project and have a number of relevant pieces of advice for student wishing to complete a similar project using similar methodology.

1. Since this project is heavily relied on the findings of the investigations, getting the permissions of the potential system users you want to investigate are very important due to some information of the current system is highly confidential

2. Don’t be put off by the rejections from interviewees, try the others, they may be helpful and friendly

3. System modelling is like talking in another language (in this case I am talking all the project through using UML) in order to present it to the assessors outside the system developing team, make sure they understand what you are talking about

4. Planning is crucial to the success of the project due to its size and time constraints. Estimating how long you will spend on each section is extremely important and can be a great motivator, but try to stick to it. As I was advised at the beginning of the project, a well constructed Gantt chart will help informing yourself all the time

5. Time management is another success criteria. Do not leave everything to the last minute since you never know whether you have underestimated the time taken to finish each part of your project.

Project Schedule

Requirements Analysis for a Student Information System

░ Project Commitments

█ Non-Project Commitments

▓ Deliverables

Task ID	Task Description	Nov 2003				Dec 2003				Jan 2004				Feb 2004				Mar 2004					April 2004			
		03/11	10/11	17/11	24/11	08/12	15/12	22/12	29/12	05/01	12/01	19/01	26/01	02/02	09/02	16/02	23/02	02/03	09/03	16/03	23/03	30/03	06/04	13/04	20/04	27/04
	Background Phase	◄														►										
1	Understand the Problem	░	░	░	░	░	░						░													
2	Investigate similar projects		░	░																						
3	Research on the system development methodology				░	░	░							░	░											
4	Compile Information					░	░							░	░	░										
5	Mid-Project Report				░	░	▓																			
	Requirement Capture Phase				◄													►								
6	Investigate the SIS of school of computing				░	░	░																			
7	Requirement specification for SIS of school of computing					░	░																			
8	Investigate the typical Chinese University SIS							░						░	░											
9	Requirement specification for SIS of Chinese University													░	░											
10	Investigate the SIS of business school															░	░	░								
11	Requirement specification for SIS of business school																░	░								
	Analysis Phase															◄					►					
12	Comparison of three systems															░	░	░	░	░						
13	Evaluate the results																	░	░	░	░					
	Outcome Phase																		◄				►			
14	Requirement specification for the potential new SIS																		░	░	░	░				
	Examination						◄						►													
15	Exam Revision						█	█	█	█	█	█														
16	Exam Period										█	█	█													
	Deadlines																		◄							►
17	Submit Contents & Chapters																			▓						
18	Submit Final Project Report																									▓
	Final Report Writing																	◄								►
20	Draft of the Project																	░	░	░	░	░	░			
21	Finalized the Project																							░	░	░

Appendix B Interview Scripts

1. To define the requirements of the SIS modelling in the School of Computing, the following questions were answered after a meeting with Les Proll.

Q: How long has SIS been developed?
Les: It has been developed over lots of years since 1975. From 1990, SIS started to be integrated. Considerable work has been undertaken to rationalise and integrate the disparate information systems since 2001 with the establishment of the new post of Information System Support Officer

Q What a member of staff can do with the student information?
Les: Mainly most of the staffs have only read permission of taught course student. Generally speaking, a user can access the system only if he has a Linux username and password. SIS itself has got different permissions for different tasks (e.g. admi). People's access permissions are grouped by tasks they are allowed to. Permissions are allocated to different groups, for example, all staff have the permission to view all taught student records which shows on my screen that: those staff are in a group with permission to perform the task –view_all_ts.
Group is a kind of membership. Furthermore, a particular academic staff will have the permission with the particular modules he is teaching. Here is another good example; some special activities can only performed by certain group of people like "exam_admi" is the permission given to the examination officer. The permission allocation system within SIS is fully developed.

Q. In your opinion, what does student information mean? Information for student or students' information.
Les: I think it means both, which is also the purpose of building SIS to serve both contexts as long as within the budget.

Q. What is the main programming language used in SIS development?
Les: Mainly Perl, and some JavaScript as well as postgrasp database.

Q. Did SIS development team do any requirement capture and analysis before building the system?
Les: Yes we did. And we are still doing that now for the further development. However, it is not in the formal format as you do.

Q. How does SIS integrate with other systems such as Edass and Banner?
Les: Well, Edass is the front end of SIS, till now; we have already been able to store all the data in the SIS instead of Edass, which is a great improvement and a basis for further integration. The communication between Banner and SIS is through ODBC link nightly.

Q. Does Banner update SIS or the other way around?
Les: Generally speaking, Banner updates SIS with both taught student records and admission applications through ODBC link overnight. However, in some sense, there is an indirect way that SIS updates exam results to Banner as well.
Q. What is the current development in SIS?
Les: Previously we got a lot of little discrete systems within School of Computing, which are mainly Microsoft Access databases or C language such as Edass. We are currently integrating all of them into SIS. Moreover, our development is also undertaken in the process of Admission for MSC and research student.

Q. What do you expect SIS to develop in the future? Any other functions need to be developed?
Les: Yes, definitely it is not enough now, though SIS covers most of the school's activities currently. To make the system functionally better in the future, I think another important function we need or in other words a gap to fill in is the ability of provision of standardised report.
Fully integration with Edass is another goal. It is written in C, therefore can be accessed only from the Linux system. We are trying to make it fully web based, which can be accessed by any browser.
The education and training of the future user is also an important issue. I am retiring in two years. I hope more people can get better understanding of SIS and do more work for the development.

Q. As a manager of SIS, what is your responsibility of the system?
Les: Well, my responsibility is quite wide but not technical.(1) I am responsible to make sure the data is accurate and consistent. (2) Also define what development we are going to have. (3) Seek opportunities for development

as well as look ahead for development. The technical bit will be passed to JN Ainsworth. System Development Support Officer.

Q. While you are developing the system, are there any business rules you made for SIS?
Les: Well, we are concentrate on actual involving rather than the common layout. We do not have any formal documented business rules. But what I can tell you is our general policies and strategies, which is to make SIS as dynamic as possible though there are some static information on the system, we want the user can get as much information as possible.

Q. Do UG (undergraduate) staffs generally get greater access than academic staffs?
Les: Yes, they do. They can update the timetables as well as coursework schedules. As I told you, they are a group of people in SIS get particular accessibility. However, nowadays we normally doesn't make any changes to the student personal details as they are formally updated by Banner

<u>2. To define the requirements of the SIS modelling in the School of Computing, the following questions were answered after a meeting with Jonathan Ainsworth.</u>

Q: What is the overall aim of SIS?
J: SIS stands for School Information System, which covers wide range of information requirements ruling of school of computing.

- A part of SIS does is student records (undergraduate, taught postgraduate and research students)
- Staff information
- Research information which includes research publications and research grand)
- Student Administration (UG and Post applicants)
- Room Booking System (meeting rooms in SoC)
- 192 Facilities, which includes contact details etc.
- Performance statistics (how many students are currently registered in the school, how many research publications we have got, how much research grand we got, etc)

The most basic things about SIS is the link with Banner. Banner imports student data nightly to update the student records in SIS. This may cause many events such as new student records creation (usually at the beginning of the academic year), module registration change, change of personal details (such as address) etc.
As well as that, much information is direct inputted into SIS. That information includes:

- All information on modules of School of Computing
- A detailed module list which is comprised of teaching staff, assessors and external examiners who are responsible for each module as well as the students registered on that particular module.
- SoC Module Review Back System. Once every year, all academic staff need to fill in a module review form and send back to the central student office as a university rule. That information can be sent by SIS with the web based form to be filled in.
- Student project information. All those information is recorded by SIS as well as the allocation of the student (supervisors and assessors list)
- Tutorial support. The information for each tutor of the tutees (includes contact details, personal details, photos etc)
- Coursework Schedule
- Module timetable
- Assessment and examinations
- Coursework submission eases the submission of the electronic version of coursework.

Last but not the least, is the reporting system, which report to the central university administration. E.g. equal opportunities report (male, female statistics, etc). Most of them are self-checking report which monitoring how well students are doing, how people run the modules etc.

Q: What activities does SIS cover?
J: The primary activities covered by SIS are:

- Module catalogue
- Student details (program taken, personal detail…etc)
- Module registration
- Assessment
- Timetabling
- Coursework scheduling

- Project administration
- Reporting

Q: What a member of staff can do with the student information?
J: There is a complex permission system in SIS to allow different staff to undertake different actions depends upon what role he/she is.

Normal lectures cannot update or amend any student records. They can only look at individual student records. If they were the tutor at the same time, they would have some more detailed information of their tutees such as progress checking. Moreover, they need to do the module review once every year as I mentioned before. They also have the permission to enter marks into Edass.

Few selected staffs such as assessment tutor can administrate all assessment or project. He/she may also responsible for returning the grade to the university.

The senior staff will be responsible for running the report in order to look after the progression

Those members of staff working in the CSO are responsible for entering some types of data such as student letters, student illness, examination results, overall degree classification … etc.

There is also an over all administration who is responsible for everything really going on in SIS.

Q: Did SIS development team do any requirement capture and analysis before actually building the system?
J: Let me tell you how things happen then, this may give you an general idea of how we develop the system. Before I came here, it was a mess student record system with 13 individual separated systems. What we did is mainly the integration of those little systems in to a big SIS. Although some of the functions are requested by the senior member of staff, we have never carried out a survey. The development is driven by the aim of fitting with Banner. The reason why we haven't done a formal requirement analysis is because SIS is built on the basis of the old system, they were not lack of functionality but lack of integrity and data consistency. The admission part of SIS was built from scratch thus I did asked for some advice from the admission officers, but not really a formal analysis.

Q: How does SIS integrated with other systems such as Edass and Banner?
J: Let's start from Banner; SIS is based on a post-greSQL. It is a single database. The main way that Banner interacts with SIS is through the Script running automatically nightly. Data is imported from Banner then there is a separated system comparing the difference between those data. If any difference is found, then the data of SIS will be automatically amended to follow the Banner's. Therefore, any changes in Banner of the student in SoC will update SIS nightly.

Edass is written in C program, used to have its own data files. A year ago, it is changed. Now it is using the SIS's database but still the same interface.

Q: What is the current development in SIS?
J: In student records part of SIS, this year, we have done the coursework schedule, but there is still a lot of work needs to do with joining coursework scheduling, submission and Edass. Due to the historical reason, the system is separated; therefore, such as coursework marks may be entered more than once.

The other area we are working on is the reporting. Our aim is to get more and more information from the system and monitor student progression particularly.

Q: What do you expect SIS to develop in the future? Any other functions need to be developed?
J: Well, the development of SIS in the future will be influenced by both internal and external factors. Externally, the development of SIS in certain way, which is required by the university. The requirements from the university decide how SIS is going to organise its data. For example, this year all the module catalogues have been updated into the new format as well as the program catalogues followed by the request from the university. SIS is a sub-system in the university; the interface may change as the university system changes.

The other external factor is the faculty issue. This seriously impact on the future of SIS. Since other members of the faculty join in, will they be allowed to use SIS or we need something else instead.

Internally, the main area of development is to trying to make more use of data as we have got now. More reporting system as well as extracting information from those information available in the current system.

There is a current testing application of SIS, which is to help tutors. The purpose of the application is to check whether student has met the requirement of the program he/she has registered. If anything fails (such as less then 120 credits), system will e-mail both student and his/her tutor. It is trying to make the administration as easier as possible.

Q: As a Information System Support Officer, what is your responsibility of the system?
J: I am responsible for looking after day-to-day operations, checking whether the overnight script works properly. Checking database as well as identifying any problems with the data, for example, if anything wrong with the taught undergraduate student module registration, I will notify Kevin.

I am also responsible for the day-to-day software maintenance. Identifying the errors as well as fixing them.

Finally, I am greatly responsible for the development of SIS. I make major decisions on either creating or changing of the system.

3. To define the requirements of the information system modelling in the Leeds University Business School,(web for faculty) the following questions were answered after two meetings with Miss Millitza Callinan and Mr Nicolas Forsans respectively (academic staff: lecture and personal supervisor)

Q. How long have you been using the system?
M: I started using the system when I first started working in Business school, which is last November.
N: last September.

Q: What do you do with the system?
M: I am doing various activities with the system.
For teaching, I can get access to the module information as well as setting lecture structure from the system. I am also a personal tutor for a group of students. I can get the student information of them from the system as well.
For marking, there is another separated marking system in the school for security reason. What I can do is inputting the results into a formal Excel spreadsheet, which will be further transferred into anther system (let's say student coursework mark system) by the staff of UG office. So what I view the system is a combination of module information system, marking system and student information system.

N: I only use the system to get a list of students that I will teach the following academic year at the beginning of each semester. During the academic year, I don't actually use the system very often unless I need to e-mail my students.

Q: What kind of information you can get accessed from the system?
M: Any details of student that I am teaching rather than student in general. Since I am the personal tutor, I can get access to the group of student that I am supervising.

N: the system can provide me a list of students that I am teaching and their personal details. But only those taking my module or in my tutorial groups

Other information is so limited that I used to go to the UG office to get further student details or module details from there.

Q: Could you give me some details of how the information of the coursework marks from marking to be presented to students?
M: Yes. But as the detailed process is confidential, so what I can tell you is quite general. I will put the results into the spreadsheet set up by the UG staff. Then they are responsible for implement the results into the school's student database. After all those have been done, the system will print coursework marks for each student.

N: same as Militza said.

Q: What else you can do with the system?
M: Apart from what I told you just now, I use the e-mail system mostly while I need to get contact with my student. The electronic resources available from Banner can also be access from the system here.

N: nothing other than that I think.

Q: What is your main manual wok currently?
M: My current manual work is mainly coursework marking. It takes a lot of time and not very efficient due to the different marking scheme. If the coursework can be submitted electronically, it will be greatly help plagiarism checking as well as share marking. By which I mean a piece of coursework can be shared by all the markers.

N: marking coursework

Q: How do u feel like the system, (it is effective, efficient enough)?
M: It is not very convenient to use the marking system some times. And if I want to get further detail information about a student in the business school but not under my teaching list. I have to go to the UG office and ask for it.

N: not really helpful, limited functionality and resources. I don't really like use that all the time.

Q: There is a new system under construction, what do u expect mostly from the new system?
M, N: The integration of those separated system is necessary, however, security is an important issue either in the systems integration or user interaction. Many activities are undertaken separately due to the various systems each are dedicated for particular functions, which also made the information system within business school quite complicated. I think a more integrated system is needed either for students and staffs
Secondly, timetabling is very important in teaching. The current system is lack of those sorts of functionalities. I hope it can be improved in the future.

4. To define the requirements of the information system modelling in the Leeds University Business School, (the internal systems of LUBS) the following questions were answered after the interview with Tony Jones (Director of UG studies)

Q: What system do staff use in LUBS?
T: The department use the ODBC link to Banner, the UG office staffs use Banner directly for example to update student records. The other academic staffs can only get access to the Web for Faculty, which is, provide by the university to retrieve the required information. Therefore the information system available to them is not from the LUBS local system but the Banner database.

Q: Is the system web-based?
T: Well, no actually. The systems available to the UG office staff are directly database access.

Q: How many systems are there within the LUBS?
T: Since the separated database systems, there are mainly 4 systems with the school
It is distinguished by Levels:

- Research student database system
- MBA database system
- Taught postgraduate database system
- Undergraduate database system, which can be further divided into exam system and admission system.

Q: What activities can those systems cover?
T: The users of the system are only those office staff. They set the data either in Banner or our local system. The data that belongs to Banner should always be entered into Banner, however, some local information such as tutorial group allocation and exam results are always stored in our local database.

Secondly, comes the undergraduate application admission, while the university receives the UCAS form, they will record some basic information in the central database, which will then be sent to LUBS through the ODBC link. The paper form of the application comes at the same time as well. Then we will put that information into our local admission database and inform the applicant when he/she will receive an offer. And we send open day invitation letter as well.

Third one is the admission system, there is no enquiry database in admission system, the local admi database pulls down the enrolment information from central Banner database, then the system use those information for assessment as well as attendance.
The system will generate a unique script number for each student and use those scripts as identifiers in the exam system. The admission system also provide information to personal tutor, module manager about the degree subject area the student are in. Moreover, the admission system also responsible for the module enrolment, it can be used to undertake the class allocation process e.g. Operations Management tutorial class.

The exam system consists of using Excel and Access database. The formatted Excel spreadsheet is called the mark workbook. The coursework marks are entered by module manager and then transferred to the exam system by office staff. The exam system holds all the student records in LUBS and some basic progression or classification detail.

Q: What platform do those systems run on?
T: We have Windows XP operating system installed every PC in LUBS, Microsoft Office Access is the main soft ware we are using for out internal systems

Q: How many people can directly access the system?
T: Only staff from UG office can get access to those internal local systems, there are 7 or 8 people can actually do so.

Q: What other functions you would like to see in the future development?
T: Currently, a new school web site is under construction for potential applicants as well as our existing students. This will be another mediate to provide updated information for our students. The staff from the UG office will have the ability to write into the database for the web site.

In terms of Banner database and our local database, the university will be moving from Client-Server based to Internet based. So the front-end will be very much web based. This is also the direction that we are moving to.

It will really good that if some useful system functions can be introduced on top of our existing system such as personalised timetabling, electronic submitting etc.

5. To define the requirements of the information system modelling for Nathan Bodington System, (university system facility) the following questions were answered after the interview with Paul Wheatley.

Q: What is your role in terms of relation with the Bodington System? What is your responsibility of the System?
P: I am the Virtual Learning Environment Service Team Leader. I have overall responsibility for running the Nathan Bodington system at the University of Leeds. I manage the team that runs the service and provides support to users via the Bodington Helpdesk.

Q: What is overall aim of Bodington System?
P: The Bodington system provides support for delivering online learning to the students of the University of Leeds. The overall aim of the service is to enhance teaching at the University by providing the teachers with the mechanisms they need to effectively deliver e Learning.

Q: What activities are covered by BS?
P: Bodington focuses specifically on supporting teaching and enhancing learning.

Q: What functions does Bodington System provided?
P: Bodington provides a structured and secure environment for teaching materials, peer to peer and peer to tutor discussion (bulletin boards), online submission of coursework, online testing using multiple choice questionnaires, etc.

Q: What development languages does Bodington System use? Technology?
P: Bodington is a JAVA based application which delivers a web based (HTML) Virtual Learning Environment system to the users.

Q: How does Bodington System work? Is there any Interaction with Banner? Interaction with individual school's system?
P: Bodington is frequently updated with new staff and student information from Banner. Interaction with school's systems is typically at the level of links from Bodington to other online teaching materials.

Q: Who is responsible for the development of Bodington System? What are the current developments?
P: The Bodington system is developed by the recently formed "Bodington.org". The software is open source and originates at the University of Leeds. It is now used and developed at other institutions like Oxford, Manchester and the Uni of the Highlands and Islands.

Q: What is the future development direction of Bodington System?
P: The future of the software will be dictated very much by the users of the system. The broad aim is to further enhance the range and detail of functionality of the system. As an example, one
area of possible development might be to add functionality for online "Progress Files".

6. To define the requirements of the information system modelling for Nathan Bodington System, (university system facility) the following questions were answered after the interview with John Stell.

Q: What is your role in terms of relation with the Bodington System? What is your responsibility?
J: I am the floor manager, who is responsible for the student placement in School of Computing.

Q: What is overall aim of Bodington System?
J: Communicating with students. SoC does not use it very often since SIS does the most of the work.

Q: What functions does Bodington System provided?
J: It provides bulleting board that enables student and staff write comments. It also provides online log work that students can fill in and then the lectures can read about. Finally, it provides general-purpose service web, for example, I post information about how placement assessed on the SoC page. As I mentioned previously, we don't actually use the system very often. Normally, the department who doesn't have fully functional information system use it as a main communication media.

7. To define the requirements of the information system modelling for Shanghai University of Finance and Economics the following questions were answered after a phone call with Haiying Zhao (senior lecturer).

Q: Do you use the system very often? What do you do with the system?
H: yes, I do. It is involved in my daily work. Since I am a senior lecture, a part of my job is dealt with the student admission work. So the efficient student records access is very important. The information that I can obtain from the system also includes all student module results, study plan, timetable, etc.

Q: Is the system that you can get access different from the general student interface?
H: yes, it is completely different. There are actually two separated systems sharing the same database. One is student web based interface, which use java script and html. The admission system that all staff are using are different. There are two standard version of admission system currently used across most of Chinese Universities. One is developed and published by Zhejiang University, the other one is developed by GuangZhou Jinan University. They are both derived from the previous systems used in those universities. They are both written in VB.

Appendix C Requirement Specification for New Student Information System

Vision Document

1. Introduction

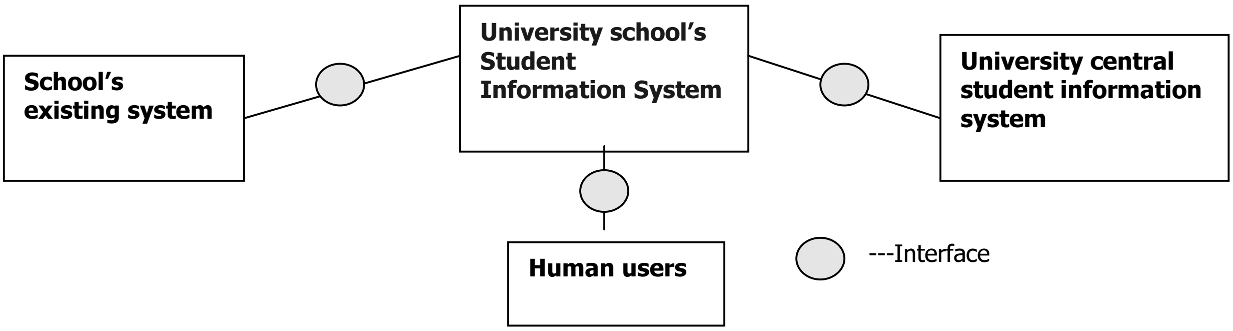
This document describes the specification for Student Information System within potential university schools. By introducing this system, all members of a school have the access via its main interface (web-based) intranet web pages by providing their username and password. The value proposal to the user is the effective and efficient way of broader range of information access. Moreover, from the student's prospect, it also means convenient coursework submitting and results checking whereas, for the staff, it helps them generate workload report and ease administration process.

1.1 Purpose

The purpose of this document is to describe the high-level requirements of a university school's Web-based Student Information System from the users perspective, using Unified Modelling Language (UML) diagrams for illustration.

1.2 Scope

This Vision Document applies to the university school's Student Information System, which is going to be developed by Aquilo. The system will be installed and maintained at the school level rather than university. The web based Student Information System will be available for both staff and students of that individual school to browse, and will provide them with different information authorizations from any Internet enabled computer.


University school's Student Information System
School's existing system
University central student information system
Human users
---Interface

The Student Information Database and Dynamic Web page Generation System will interface with a number of existing systems, including university central student information management system (like Banner), and school's local existing system.

1.3 Definitions, Acronyms and Abbreviations

Please see the Glossary.

1.4 References

Simon Bennett, John Skelton, Ken Lunn. 'Schaum's outline of UML'. 2001.UK
School of computing, University of Leeds, 'IN21-Topic1 course Notes' 2002. UK

1.5 Overview

This document details the requirements and specification for the Student Information System for University Schools.

Part 1 - Vision	- General overview of the project.
Part 2 - Business Modelling	- Description of business context of system.
Part 3 - System Scope	- Description of elements to be developed by software team.
Part 4 - Requirements	- System layout, and description of processes that will occur at each stage.
Part 5 - Design	- Detailed model of system from a technical perspective.

2. Positioning - Problem Statement

The aim of the system is to enable the university schools to provide a level of services to both staff and students from which they would benefit. This includes an extensive student information access, which provides student personal details, coursework schedules and results, module details, timetables etc. Ensuring that students and staff are well informed. The system also stores and maintains the student information as well as coursework, which can directly reduce a lot workforce involved.

Efficiency and quality of web based service; the student information system will provide a reliable information service to the students and staff within that university school based on a central, powerful relational database with data updated frequently by other systems such as university's central student records system (e.g. Banner).

3. Stakeholder and User Descriptions

This section describes the users and the stakeholders of the student information system.

3.2 Stakeholder Summary

Name	Represents	Role
Head of the School	University School	Responsible for project funding approval, and monitoring progress of the project.
Database Administration and Software development Manager	Database Admin. And Software development Team	Ensures that the system is integrated with the database effectively, upholding data integrity. Also involved with providing content, and developing and updating the System, as well as site maintenance.
Director of UG studies	UG teaching staff	Responsible for all the under graduation studies affairs including the student records approval
Project Manager	Aquilo system develop Team	Primary role is integration of the different facets of the development project.

3.3 User Summary

Name	Description	Stakeholder
Student of School	Uses the system via a web browser to access student Info.	Head of University School
Director of UG studies	Uses the system to manage coursework results and generate staff workload report	Self-represented
General Administration	Use the system to help acceptance and clearance process. Coursework schedule and timetable updating	Director of UG studies
General staff	Access any student records	Director of UG studies
Database Administration	Involved with data integrity and database performance issues, allowing seamless use of system.	Database Administration Manager

3.4 User Environment

The user will be accessing the system via a connection to the Internet. This is most likely to be from home, which is an excellent baseline, as this category includes the slowest types of Internet connection. However, the staff may normally use the system at work from the university's PCs and network.

In order to achieve flexible system access and broader range of information at the same time, web based system interface was chosen. However, issues of compatibility between different computer types still exist, although testing of the web site will have to be carried out over a number of different web-browsers. (e.g. Netscape, IE)

3.5 Key Stakeholder / User Needs

Need	Priority	Concerns	Current Solution	Proposed Solutions
Effective and Efficient student information Administration	**High**	Mass Information management is difficult and time consuming	Require intensive data entry by staff and paper work.	User 'Self Service' approach using Internet based central database information distribution system.
Database Integrity, and efficient accessibility.	**High**	Data can be lost due to security breaches, improper database design and hardware failure.	Firewall and virus detection software.	Firewall and virus software, coupled with regular data backup, use of mySQL and sound internal password management schemes.
Information accuracy	**High**	The wrong information may result in serious mistake (e.g. wrong classification)	Human data input and paper record system involves certain human mistake.	Update daily from Banner No body can change them while they are in system except certain position staff.
24 Hour Internet Access to student information system	**Medium**	Flexible access whenever and wherever you are	Hand book or go to the school reception	24/7 system access provide by the system

3.6 Alternatives and Competition

- *Get personal information (e.g. coursework results, coursework schedule and timetable etc) from the school reception in paper version*
- *Submit coursework in paper version to the certain submission box*

System Strengths: Flexible accessibility and much broader range of information can be attained at a time.
Historical records can be retrieved easily.

System Weaknesses: Poor human interaction, limited service (some query still need to go through help desk).
Technical fault of server and database will lead to bad outcomes.
Lose of password may lead to unauthorized access.

4. Product Overview

4.1 Product Perspective

The system will reside on a web server, which will send data to users on request via the World Wide Web. This will be integrated with a Database Server, which has two databases. One for storing information on taught students; research students, research grants, staff, admissions enquiries, and module etc. the other one is used to stored electronic coursework and coursework marks. They are interlinked, however, only the first one provides information to the student through dynamic web page generation whereas the second one is confidential and staff only.

Users will be required to input their username and password in order to access the system. The university central student record system will update student records regularly. Therefore the student records in school student information system are monitored and refreshed which any change has been made.

4.2 Summary of Capabilities

Customer Benefit	Supporting Features
24 Hour system access	Student and staff of the school can access the web-server of system anytime from any place. Server request central relational database to retrieve user requested information.
Great range of Information	Centralized relational database holds information on almost all aspect of student and school.
Fast, Efficient Submit	Fast intranet upload to the database through web interface
Latest Results	Coursework results update as soon as it is recorded.
Economic	Cost saving in human resource management (it normally take a lot of workforce to manage the student records manually)

4.3 Assumptions and Dependencies

- The accuracy of the information will be highly relying on either system interconnection (central student information system and school student information system) or the database maintain. Incorrect information will result a big inconveniency to the user.

- Dynamic Web-page Creation System progress will depend on student records Database completion timescale being met.

Business Modelling

Business Context

School student information system is developed to provide high-level information service to students and staff of that particular university school.

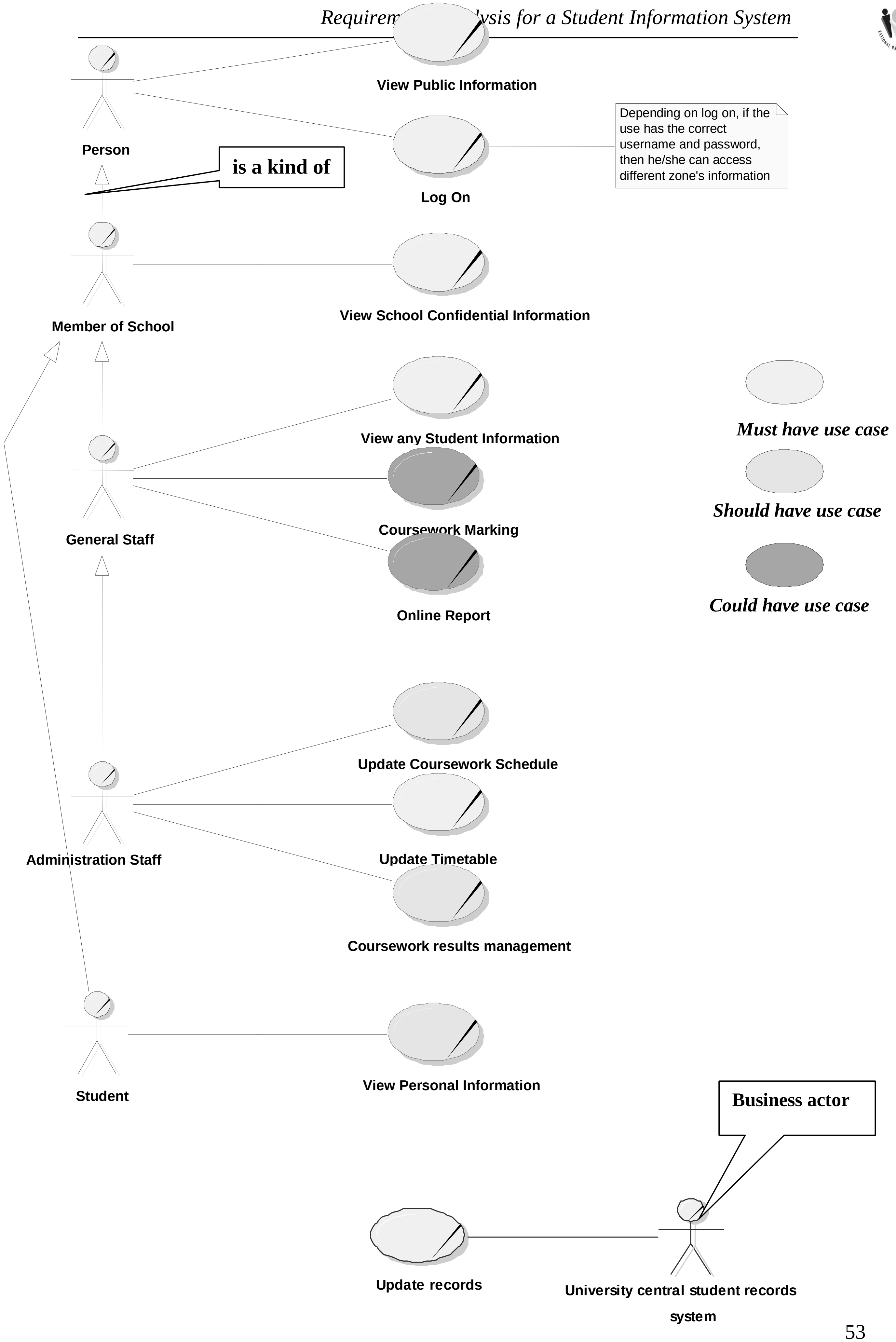
Great amount of information available	Centralized relational database stores great amount of information
Accurate information	Update and checked by module leader, system update script to ensure the data consistency with university central database
Flexible information access	24/7 any time any place.
Reduce formalized workload	System generates workload report and help admission process.
Access from the comfort of home.	Via connection to World Wide Web.

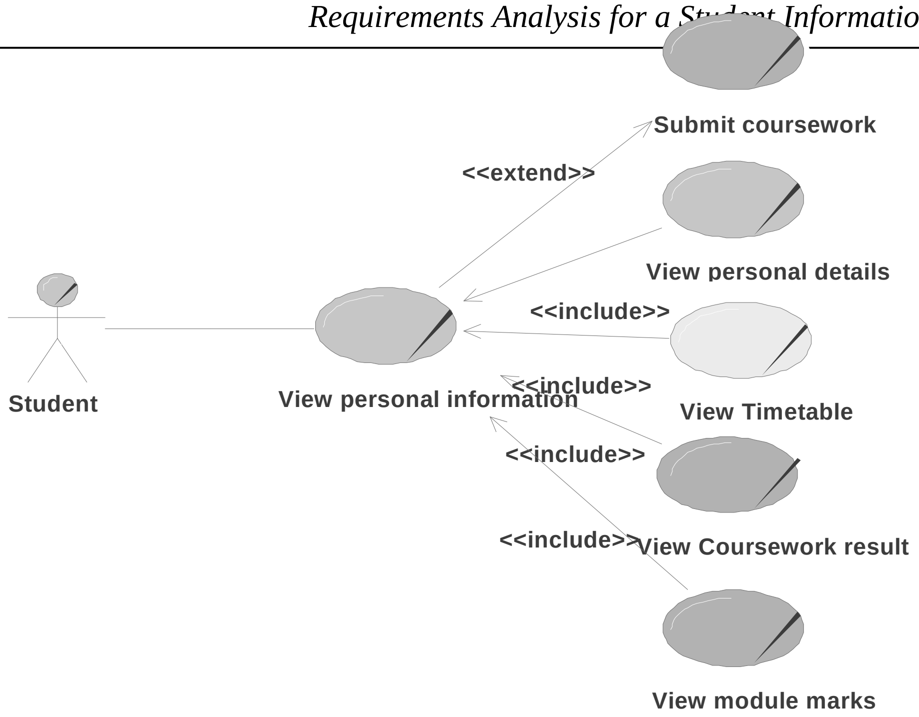
- Costs associated with staff required in help desk or reception can be reduced, moreover, cost of paper work will be significantly reduced.
- Information administration and paperwork will be minimized, with details on users stored in computer databases.
- Integration of the existing system (e.g. university central student admission system -- Banner) will make sure the information is updated as soon as any changes occur.

In the following user requirements defining part, I will adopt the MoSCoW rules (Stapleton, 1997) to distinguish between what the users 'needs' in a system and what the users 'would like'. User requirements are divided into four categories depending on their relative importance and practical use in the proposed system. The first category is 'must have' requirements which states "are fundamental to the system". The second category is the 'should have' requirements, which are important requirements, but the system would still be useful and usable without them. The third category is the 'could have' requirements, which would add value to a proposed system, which could be left out of the basic development process. The fourth category is 'want to have' but will not have this time around. This is less vital to this particular project because the size of the final system would mean that all requirements probably could be met which has been demonstrated in the first three categories.

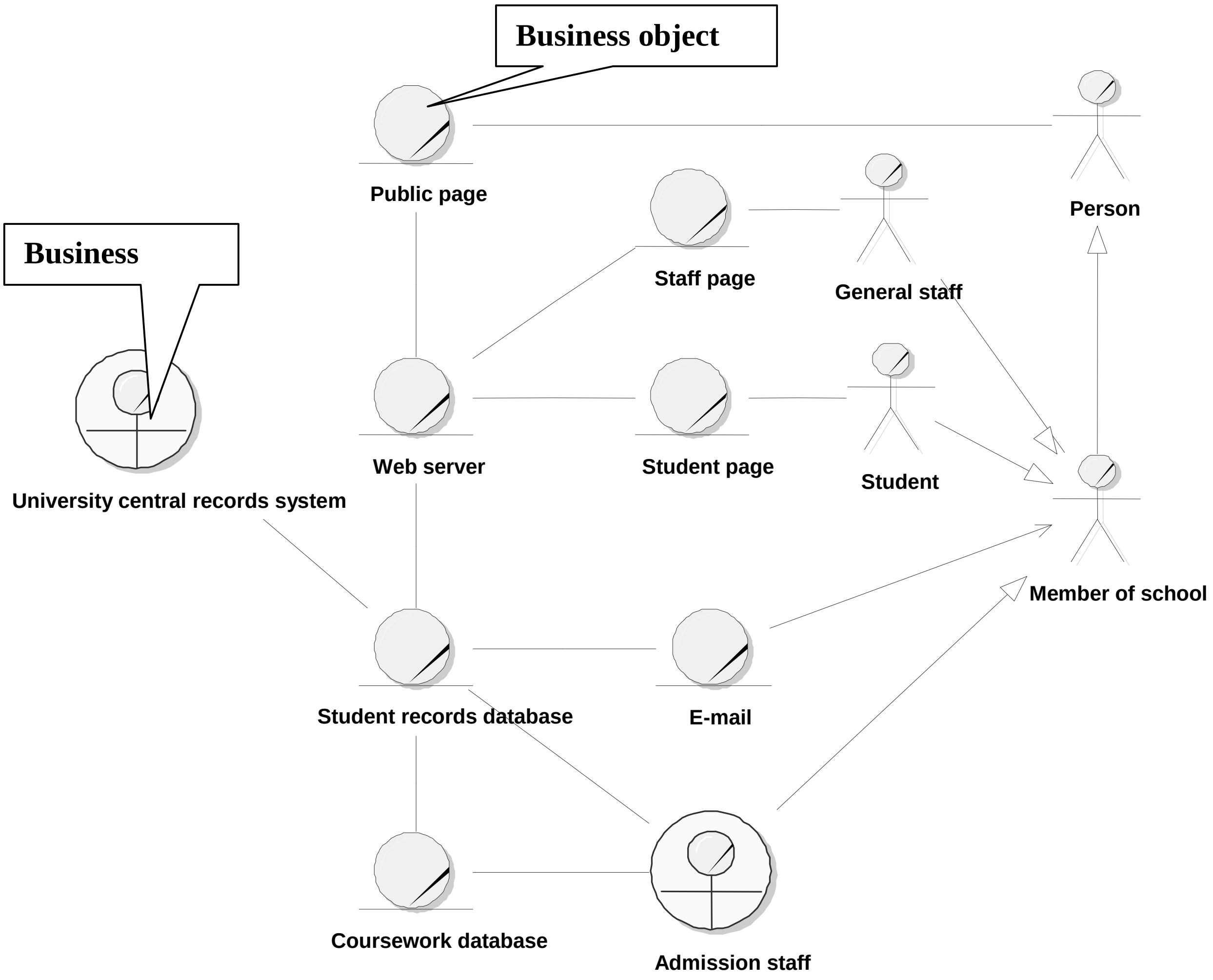
Business Use Case Diagram

The following diagram shows the main users involved in the system and the major activities carried out by them.


View Public Information
Person
Log On
Depending on log on, if the use has the correct username and password, then he/she can access different zone's information
is a kind of
Member of School
View School Confidential Information
View any Student Information
Must have use case
Coursework Marking
Should have use case
General Staff
Online Report
Could have use case
Update Coursework Schedule
Update Timetable
Administration Staff
Coursework results management
View Personal Information
Student
Business actor
Update records
University central student records system

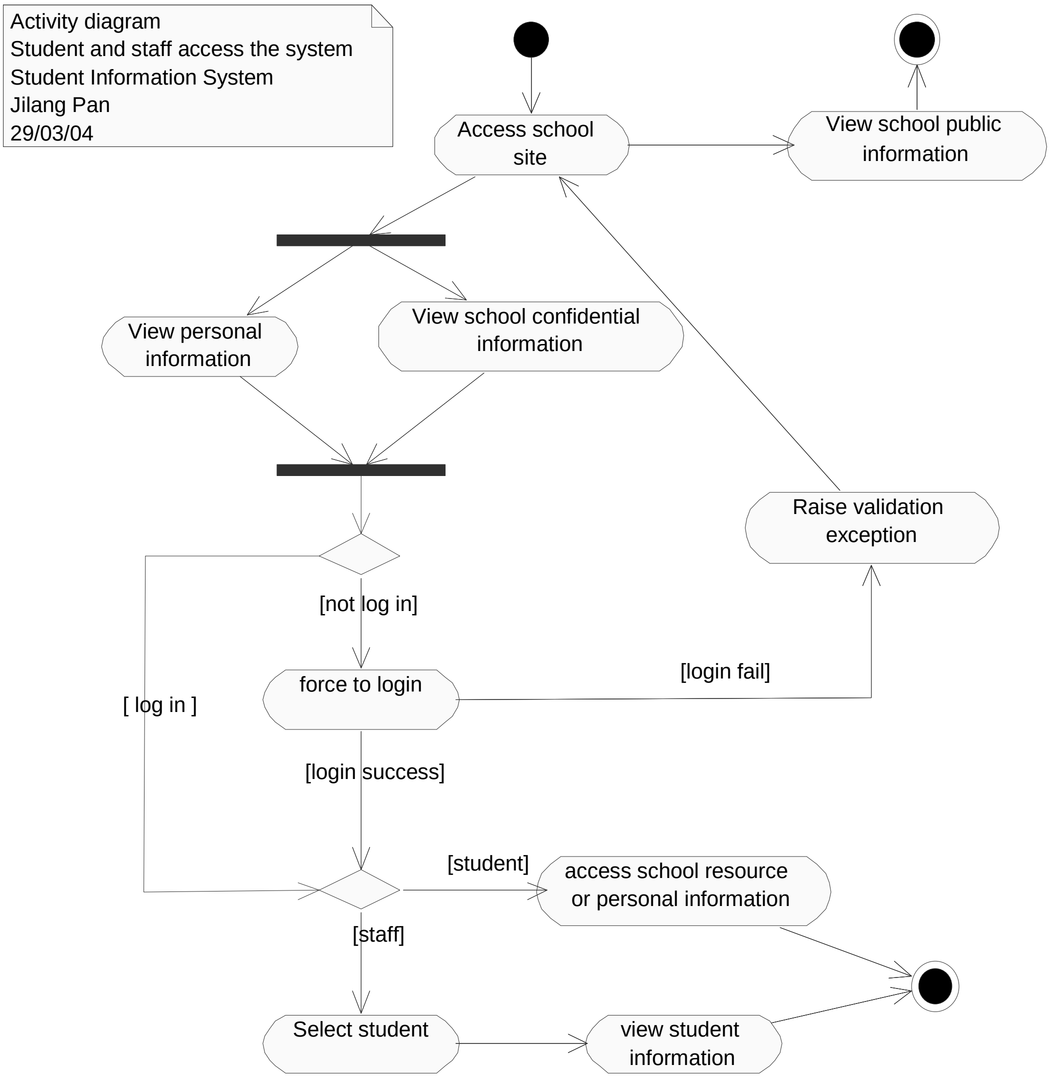


Business Object Diagram



Activity Diagram

This diagram describes the general process that happen when a user access school student information system service. (Alternative and Exception paths are also displayed).


Activity diagram
Student and staff access the system
Student Information System
Jilang Pan
29/03/04
Access school site
View school public information
View personal information
View school confidential information
Raise validation exception
[not log in]
force to login
[login fail]
[log in]
[login success]
[student]
access school resource or personal information
[staff]
Select student
view student information

System Scope

Elements In Scope

School public information (school homepage)

School internal resources (student and staff)
Coursework submit (data transfer)
Coursework management (marking and results management)
Database maintenance (timetable and coursework schedule management)
Online report

Elements Out of Scope

Module selection
E-mail system

Elements within the scope of this system include those listed, which deal with aspects of the services provides by student information system. This includes coursework submitting, view student personal details, timetable management and coursework management. Advanced functionality online report, which can be implemented, but will only be modelled at the abstract level in this specification.

Interface	Interface Type	Interface Method	Description
Any visitor to System	Human – System	Web browser	Any people can view school public information via a web browser using an Internet connection.
Student to System	Human – System	Web browser and login	Students will have a password to log into the system, then they can access the school internal student resources
Staff to System	Human – System	Web browser and login	Staff will have a password to log into the system, then they can access confidential staff only resources and services
Administrator to System	Human – System	Web browser and login or direct login	Administrators will have the high privilege for student coursework database maintenance as well as timetable or coursework schedule update.
Database to dynamic web-page	System – System	(e.g. perl script)	Unique web-pages can be created from the database providing relevant information to individual user
School student records database to university central database	System – System	(e.g. ODBC link)	University database regularly updates the local database ensure student data consistency and information integrity
General student information database to student coursework database	System - System		The reason to have two-separated databases is for security concern as well as ease of database maintenance. The electronic version of coursework will be stored in the coursework database before marking

Requirements

No1.

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**View public Information**
*	Primary Actor: *(Actor that initiates Use Case)*	Any people who has access to the Internet.
R	Other Actors:	
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The user will be able to get general information about the school and the courses and programs it provides.
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a person wants to view the school's information. The user needs to type in the exact URL of the school's home page or directed by a hyperlink from the university's homepage.
	Alternative Paths: *(Other paths through the use case which result in a successful outcome – typically variations to the basic course of events, determined by the actor and their needs).*	The user could be directed by the search engine such as google.com The user could also order a university prospect from the university homepage and get a paper version of school information. He/she could also get a copy by phone call or personal visit.
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user gave a wrong URL in the first try. The Web server of the system is Down. Those hyperlinks are incorrect.
	Assumptions:	The user gave a correct URL and the Web server is not down
	Pre-conditions:	The user must have the correct URL of the school's homepage or he must click on the correct link.
	Post-conditions:	The website would prompt a interface showing the various school information on the screen
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B05
	Related Non-Functional requirements – Usability, Performance, Security:	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No.2

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**Log in**
*	Primary Actor: *(Actor that initiates Use Case)*	Member of the school
R	Other Actors:	Visitors
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	Provide the authorized user the accessibility of further and detailed school's confidential information
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a member of the School wants to log in to the system. He/she first enters in the correct username and password through the user interface of the system then press enter.
	Alternative Paths: *(Other paths through the use case which result in a successful outcome – typically variations to the basic course of events, determined by the actor and their needs).*	The user could have logged in already and then close the Webpage. The user need not type in the username and password again in order to enter the site. The user could then be entered into the member confidential information zone directly.
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user gave a wrong password in the first try. The Web server of the system is Down.
	Assumptions:	The user gave a correct password and the Web server is not down
	Pre-conditions:	The user must have the correct username and password and he must click on the correct link
	Post-conditions:	The website would prompt a interface showing the various detailed school confidential information
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security: *(Any non-functional requirements that are specific to this Use Case rather than the system as a whole)*	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No.3

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**View any student information**
*	Primary Actor: *(Actor that initiates Use Case)*	Staff of the school
R	Other Actors:	
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The user will be able to get all the detailed information of each individual student within the school. Information includes: personal details, contact information, modules taken, module marks, coursework results and timetable.
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a staff wants to access and view the student information. The user first enters his username and password through the user interface of the student information system website and click on view student information link. He/she needs to select student before view his/her personal information
	Alternative Paths: *(Other paths through the use case which result in a successful outcome – typically variations to the basic course of events, determined by the actor and their needs).*	The user could have logged in already and then close the Webpage. The user need not type in the username and password again in order to enter the site. The user could then click on the required link and view the information The other way is request student information through the school student office.
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user gave a wrong password in the first try. The Web server of the system is Down.
	Assumptions:	The user gave a correct password and the Web server is not down
	Pre-conditions:	The user must have the correct username and password and he must click on the correct link
	Post-conditions:	The website would prompt a interface showing the all the student information on the screen
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security: *(Any non-functional requirements that are specific to this Use Case rather than the system as a whole)*	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**Update Timetable**
*	Primary Actor: *(Actor that initiates Use Case)*	Administration staff of the school
R	Other Actors:	
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The administrators will be able to modify the timetable information in the school system database. Therefore the other lower level users will be able to notice the change with the help of dynamic web sites
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when the administrators notice the lecture conflict then decide to make a change to the timetable. They simply rearrange the lectures slots in the school student information system database. The server side (or client side) script will dynamically update the information to the other users.
	Alternative Paths:	They could make the timetable changes notice in the student common room (on board) and also in the paper version timetables.
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user gave a wrong password in the first try. The Web server of the system is Down. The database is not linked with the system.
	Assumptions:	The user gave a correct password and the Web server is not down and the user has the privilege to modify the timetable details in database
	Pre-conditions:	The user must have the correct username and password and he must click on the correct link
	Post-conditions:	The website would prompt a screen confirms the modification
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security: *(Any non-functional requirements that are specific to this Use Case rather than the system as a whole)*	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No.5

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**View student personal information**
*	Primary Actor: *(Actor that initiates Use Case)*	All students of the school
R	Other Actors:	All staff of the school
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The user will be able to get all the information of himself on the screen includes: personal details, modules taken, module marks, coursework results, dissertation or final year project information as well as all the confidential student resources within the school such as lecture slides, notes etc.
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a student wants to view his/her personal information as well as the school confidential resources. The user needs to provide valid username and password in the log in screen and click the right links afterwards
	Alternative Paths:	He can also get those information from the school UG office (or reception) or from the school hand book
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user doesn't have the correct username and password. The web server is down.
	Assumptions:	The user gave a correct username and password and the Web server is not down
	Pre-conditions:	The user must provide the correct username and password in the log in screen and click the right link afterwards
	Post-conditions:	The website would prompt a interface showing the various school and student information on the screen
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security: *(Any non-functional requirements that are specific to this Use Case rather than the system as a whole)*	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No.6

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**Update coursework schedule**
*	Primary Actor: *(Actor that initiates Use Case)*	Only the administration staff in the school
R	Other Actors:	
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The administrators will be able to modify the coursework schedule information in the school system database. Therefore the other lower level users will be able to notice the change with the help of dynamic web sites
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when the administrators notice the coursework conflicts then decide to make a change to their schedule. They simply rearrange the coursework deadlines in the school student information system database. The server side (or client side) script will dynamically update the information to the other users.
	Alternative Paths: *(Other paths through the use case which result in a successful outcome – typically variations to the basic course of events, determined by the actor and their needs).*	They could make the coursework schedule changes notice in the student common room (on board) and also in the paper version timetables.
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user gave a wrong password in the first try. The Web server of the system is Down. The database is not linked with the system.
	Assumptions:	The user gave a correct password and the Web server is not down and the user has the privilege to modify the timetable details in database
	Pre-conditions:	The user must provide the correct username and password in the log in screen and click the right link afterwards
	Post-conditions:	The website would prompt a screen confirms the modification
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security:	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No.7

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**Coursework results management**
*	Primary Actor: *(Actor that initiates Use Case)*	Administration staff only
R	Other Actors:	
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The administrators will be able to access coursework results database and make any changes if needed.
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a administrator wants to access the coursework results database or make any modification to the records. He/she first log on the system and directed by the certain link to view the data or make changes.
	Alternative Paths: *(Other paths through the use case which result in a successful outcome – typically variations to the basic course of events, determined by the actor and their needs).*	If student coursework results are stored as paper version files, then the modification of the records will simply be the changes in the files
	Exception Paths: *(Other paths through the use case which result in an unsuccessful outcome – typically when something goes wrong)*	The user doesn't have the correct username and password. He/she doesn't have the privilege to access the coursework database. The web server is down.
	Assumptions:	The user gave a correct username and password and the Web server is not down the user have the privilege to access the coursework database.
	Pre-conditions:	The user must provide the correct username and password in the log in screen and click the right link afterwards
	Post-conditions:	The web site either prompt a screen shows various coursework results information or a confirmation screen for a modification
	Related Business Rules: *(Reference to your Business Rules list)*	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security:	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No. 8

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**Coursework marking**
*	Primary Actor:	Academic staff of the school
R	Other Actors:	Administration staff
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The staff will mark all the student coursework and submit the results to the provisional database waiting for the module leader's final confirmation. Then the results will be upload to the student coursework database
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a staff marked student's coursework (either electronic version or paper version). He/she then input the results in to the provisional database from the system interface. The module leader will get a full view of the coursework marks then after his final check; the results will be uploaded to the coursework database.
	Alternative Paths:	If student coursework results are stored as paper version. The staff can put down individual marks in paper and send it to the module leader. With the recheck and confirmation from the module leader, the results can be finally input into the database or stored as paper version files.
	Exception Paths:	The user doesn't have the correct username and password. He/she doesn't have the privilege to access the marking process. The web server is down. The results cannot be uploading to the database.
	Assumptions:	The user gave a correct username and password and the Web server is not down the user have the privilege to access the coursework marking process.
	Pre-conditions:	The user must provide the correct username and password in the log in screen and click the right link afterwards. The module leader should have the privilege to access both provisional database and write the results to the coursework database.
	Post-conditions:	The website either prompt a screen for staff to submit coursework marks to the provisional database and the module leader will have a screen of all the student results and make the final confirmation to them. Then the results will be finally written to the database.
	Related Business Rules:	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security:	NF01 NF02 NF03 NF04 NF05
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

No.9

*	Use Case Name: *(The name as it appears in the Use Case Model)*	**Submit coursework**
*	Primary Actor: *(Actor that initiates Use Case)*	Student of the school
R	Other Actors:	
R	Value Proposal to Actor(s) *(the goal of the Use Case from the Actor's perspective)*	The coursework can be submitted electronically to the student coursework database by the student which eases the marker to do the coursework marking as well as plagiarism checking
R	Basic Course of Events: *(The Normal Flow)*	This use case begins when a student is required to submit the coursework before the deadline. He first log in the system and click the right link to the submitting interface. By specifying the exact directory path in the interface, then he simply press submit button and got a submitting confirmation screen afterwards
	Alternative Paths:	Student can also submit the coursework in the paper version to the coursework collection box in the school student office.
	Exception Paths:	The user doesn't have the correct username and password. He/she doesn't have the right coursework file path. The web server is down. The coursework cannot be submitted to the database.
	Assumptions:	The user gave a correct username and password and the Web server is not down the user have the coursework files ready in the computer and know the directory path.
	Pre-conditions:	The user must provide the correct username and password in the log in screen and click the right link afterwards. Student should have the coursework file ready interlinked with the computer.
	Post-conditions:	The website either prompt a confirmation screen for the successful submit or an exception error message for submit failure. The successful submit will result the coursework file stored into the coursework database with student ID and time associated with it
	Related Business Rules:	B01, B02, B03, B07
	Related Non-Functional requirements – Usability, Performance, Security: *)*	NF01 NF02 NF03 NF04 NF06
*	Project:	Modelling of Student Information System
*	Author:	Jilang Pan
*	Date:	30/04/04 15:37

Business Rules Reference List

B01 - be processed in accordance with the data subject's rights

B02 - be kept safe from unauthorized access, accidental loss or destruction

B03 - the right of access for staff and students to personal data that relate to them held in computerized systems also includes 'organized' manual filing systems within departments, services and the centre

B04 - Students must ensure that all personal data provided to the University are accurate and up to date.

B05 - ensure that any information that provided by system is accurate and up to date

B06 – secure of student data transaction such as coursework submitting

B07 - All staff (and where appropriate, students) must adhere to University policy and guidelines on data security, and specifically must ensure that:

- any personal data which they hold are kept securely
- personal information is not disclosed either orally or in writing, or in any other way, intentionally or otherwise to any unauthorized third party

Non-Functional Requirements Reference List

NF01 – Web-site display
Information provided on the system web site must be unambiguous.

NF02 – Easy to use
The system should be easy to use and fulfil different staff and students' needs.

NF03 - Efficiency
The reasonable quality access speed provide by school system web server

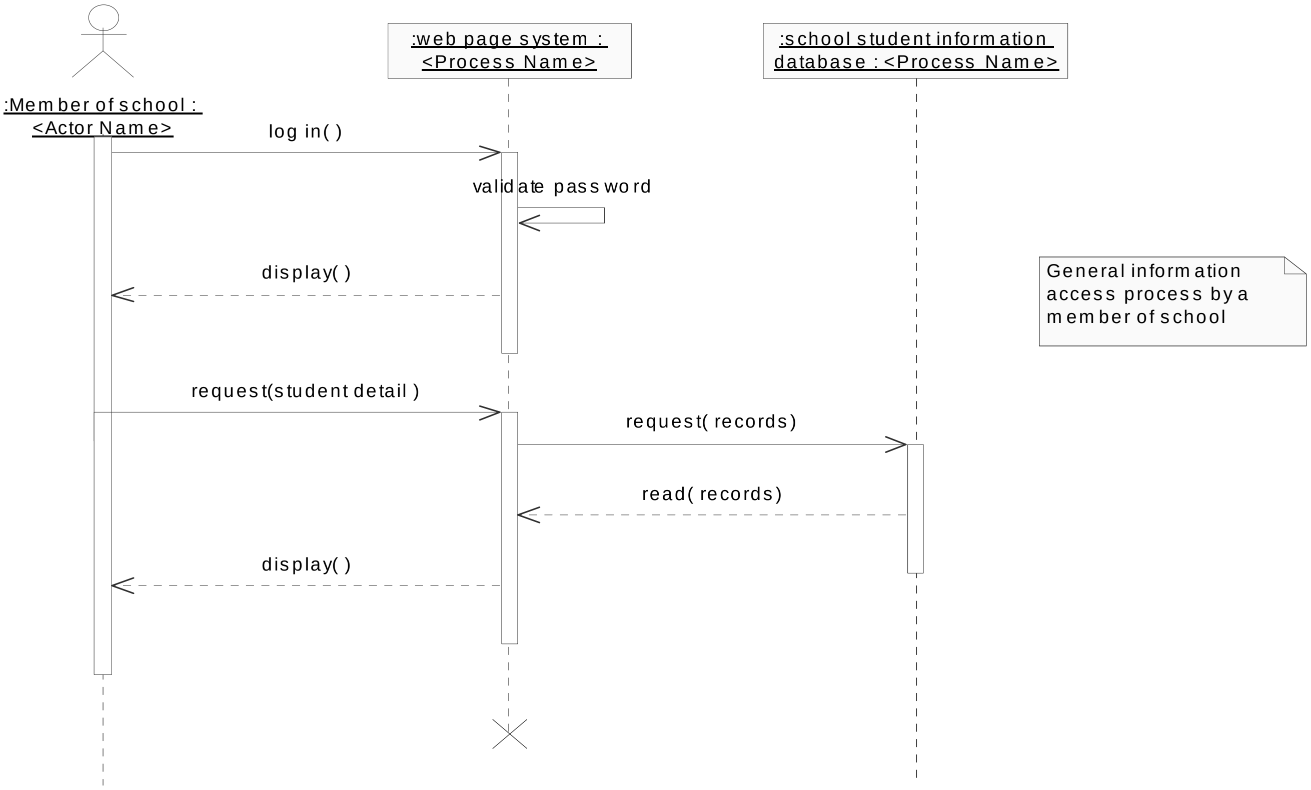
NF04 - Effective
The Information provide by the system should be useful to the user.

NF05 – up to date
The information for either coursework marks or other data should be updated as soon as possible to provide useful information

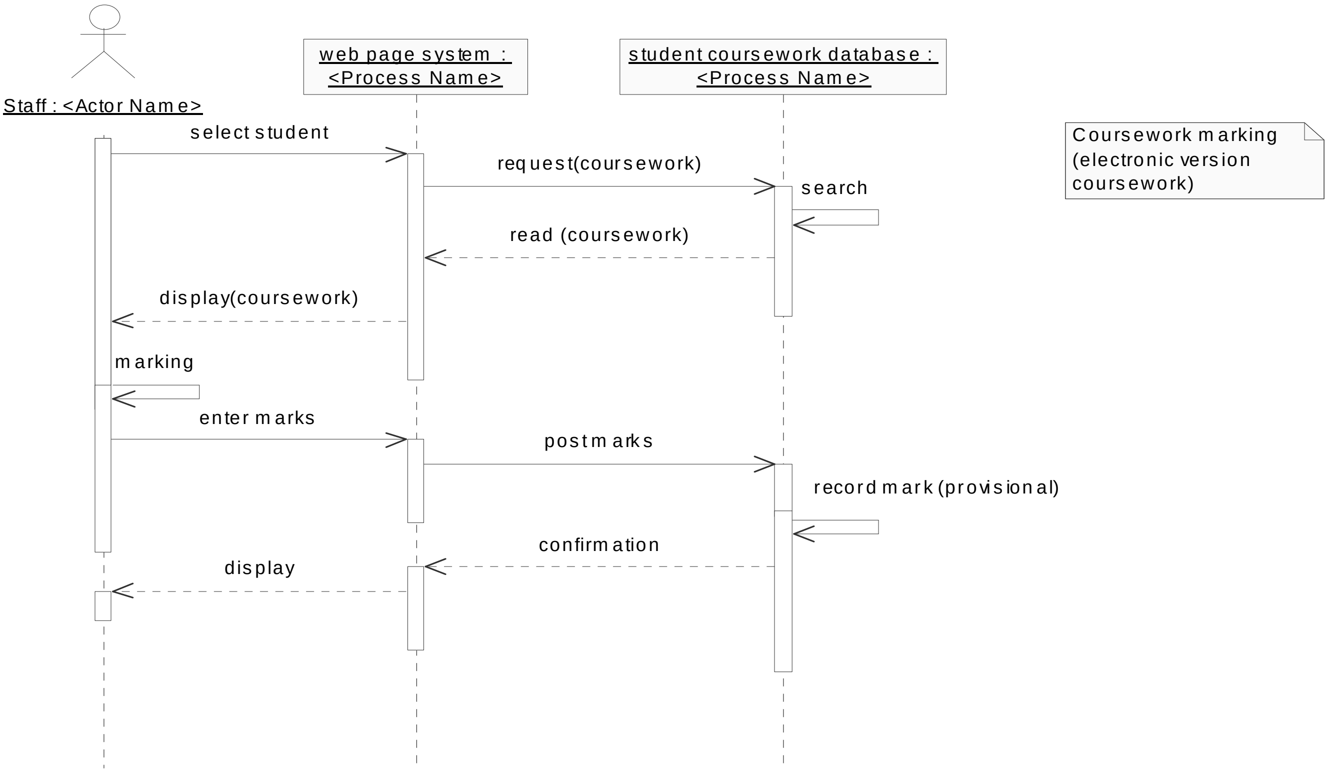
NF06–Security
The security concern either for information access and data uploading

Interaction Diagrams

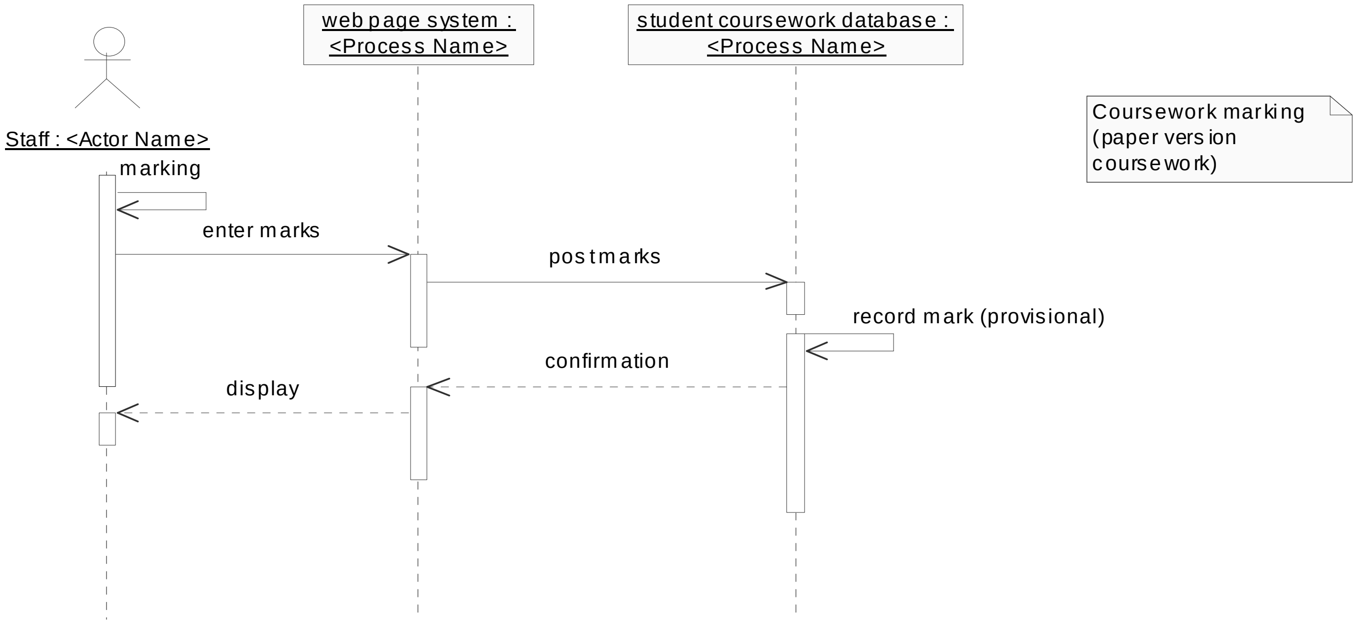
General information process of the system (when a member of school accesses the system)


:Member of school : <Actor Name>
:web page system : <Process Name>
:school student information database : <Process Name>
log in( )
validate password
display( )
General information access process by a member of school
request(student detail)
request(records)
read(records)
display( )

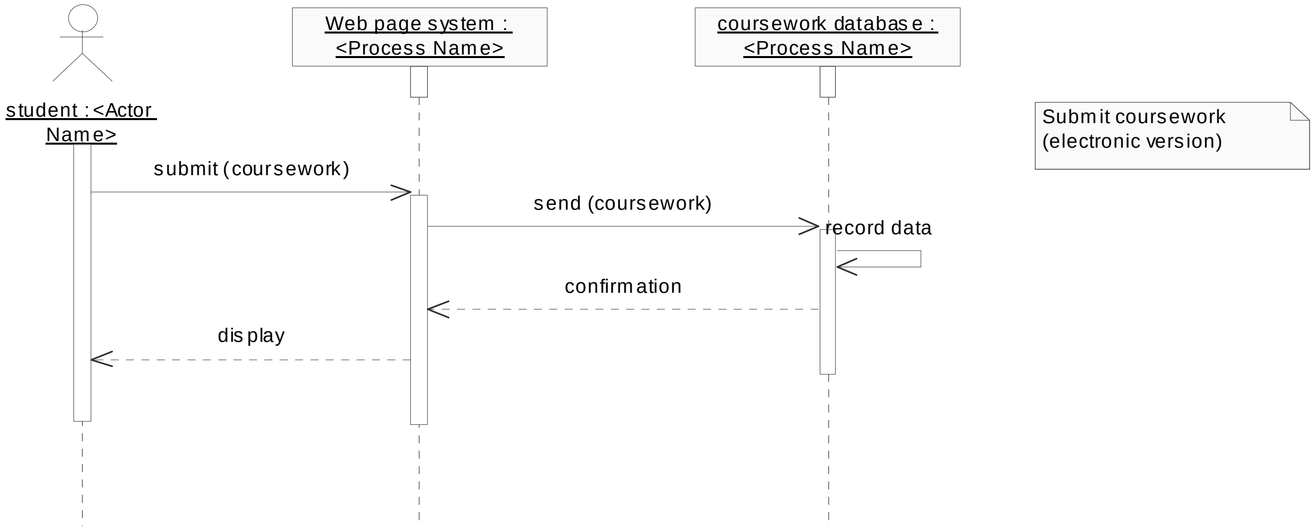
Coursework marking (electronic version)


Staff : <Actor Name>
web page system : <Process Name>
student coursework database : <Process Name>
select student
request(coursework)
search
Coursework marking (electronic version coursework)
read (coursework)
display(coursework)
marking
enter marks
post marks
record mark (provisional)
confirmation
display

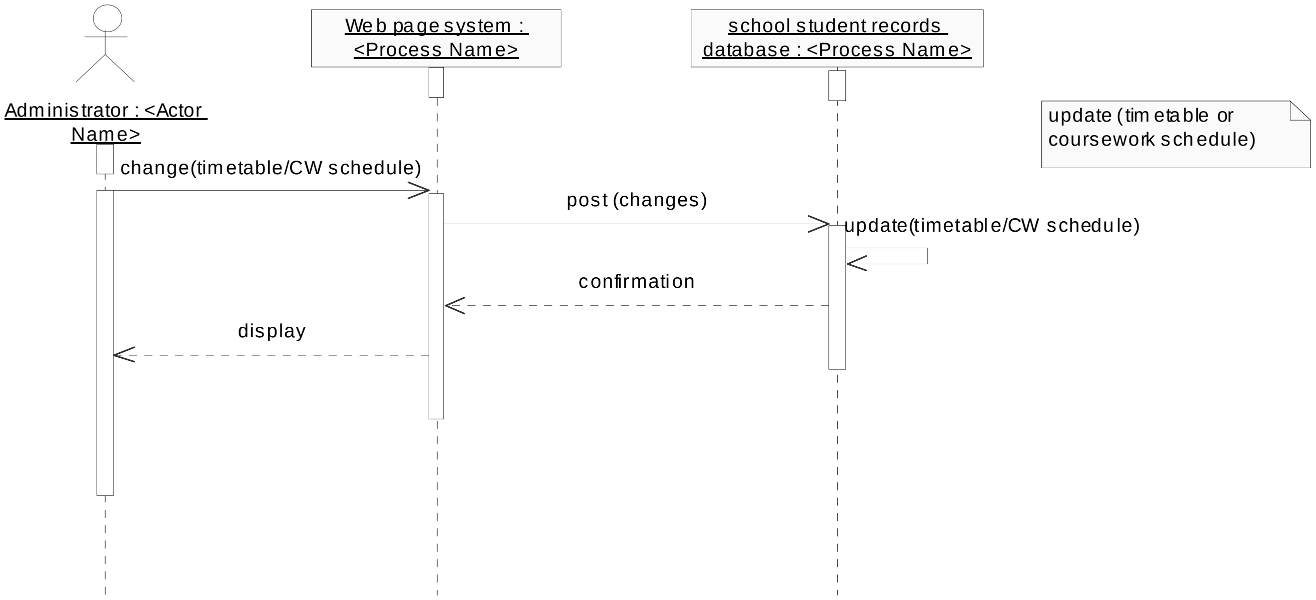
Coursework marking (paper version)



Submit coursework (electronic version)



Update timetable or coursework schedule


Administrator : <Actor Name>
Web page system : <Process Name>
school student records database : <Process Name>
update (timetable or coursework schedule)
change(timetable/CW schedule)
post (changes)
update(timetable/CW schedule)
confirmation
display

Those sequence diagrams shows the information flow of a single activity respectively, which would give the developer a general idea of how request flows through the system and how a request is satisfied.

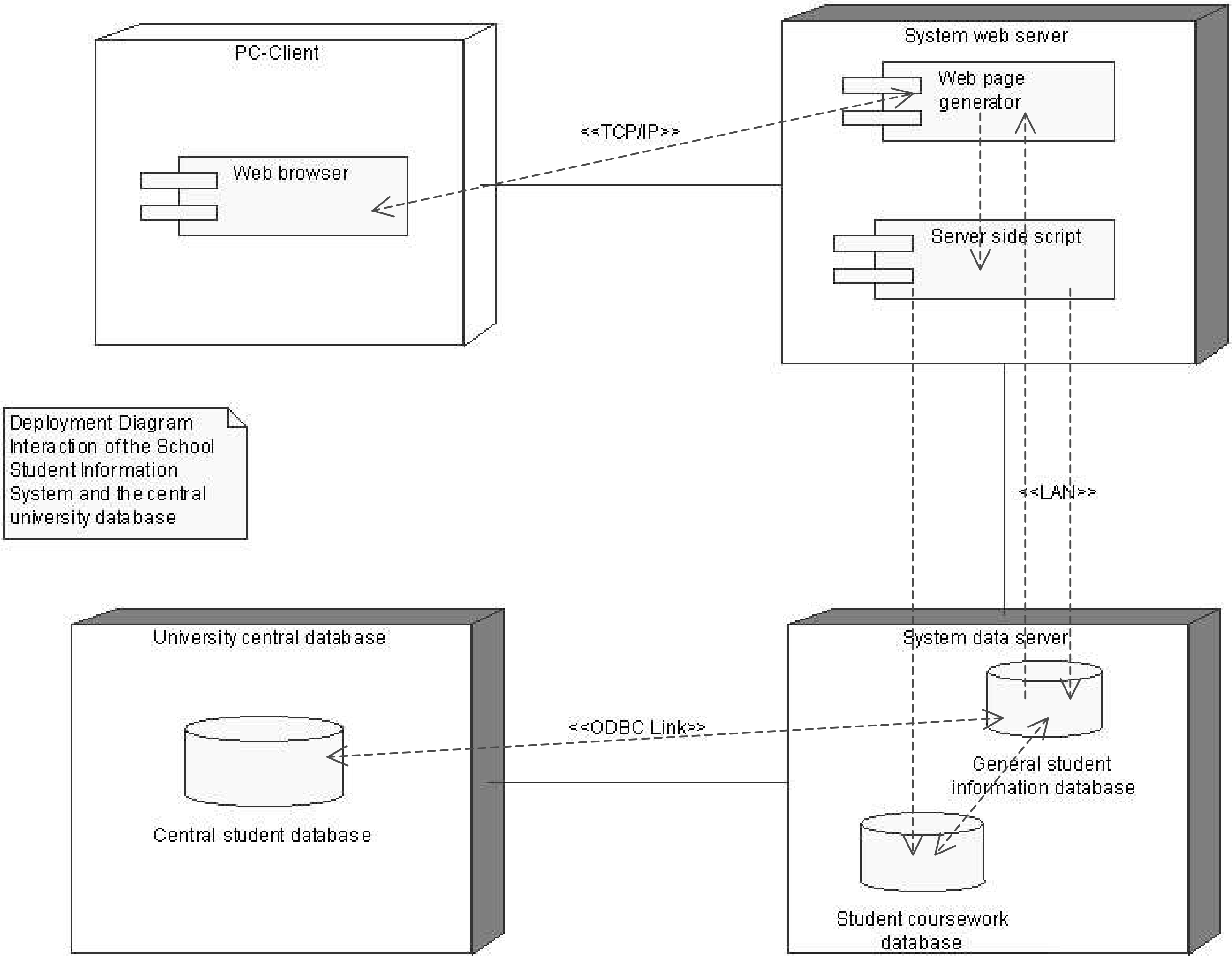
Design

System Description

The Student Information System will allow all members of a school to access the school information. For student, it can provide student personal details, timetable, coursework schedule, coursework results, coursework submitting, module results and other module related resources. For general staff, it can be used to monitor the progress of the individual student by accessing all student records. The system is also used to mark coursework as well as manage the results.

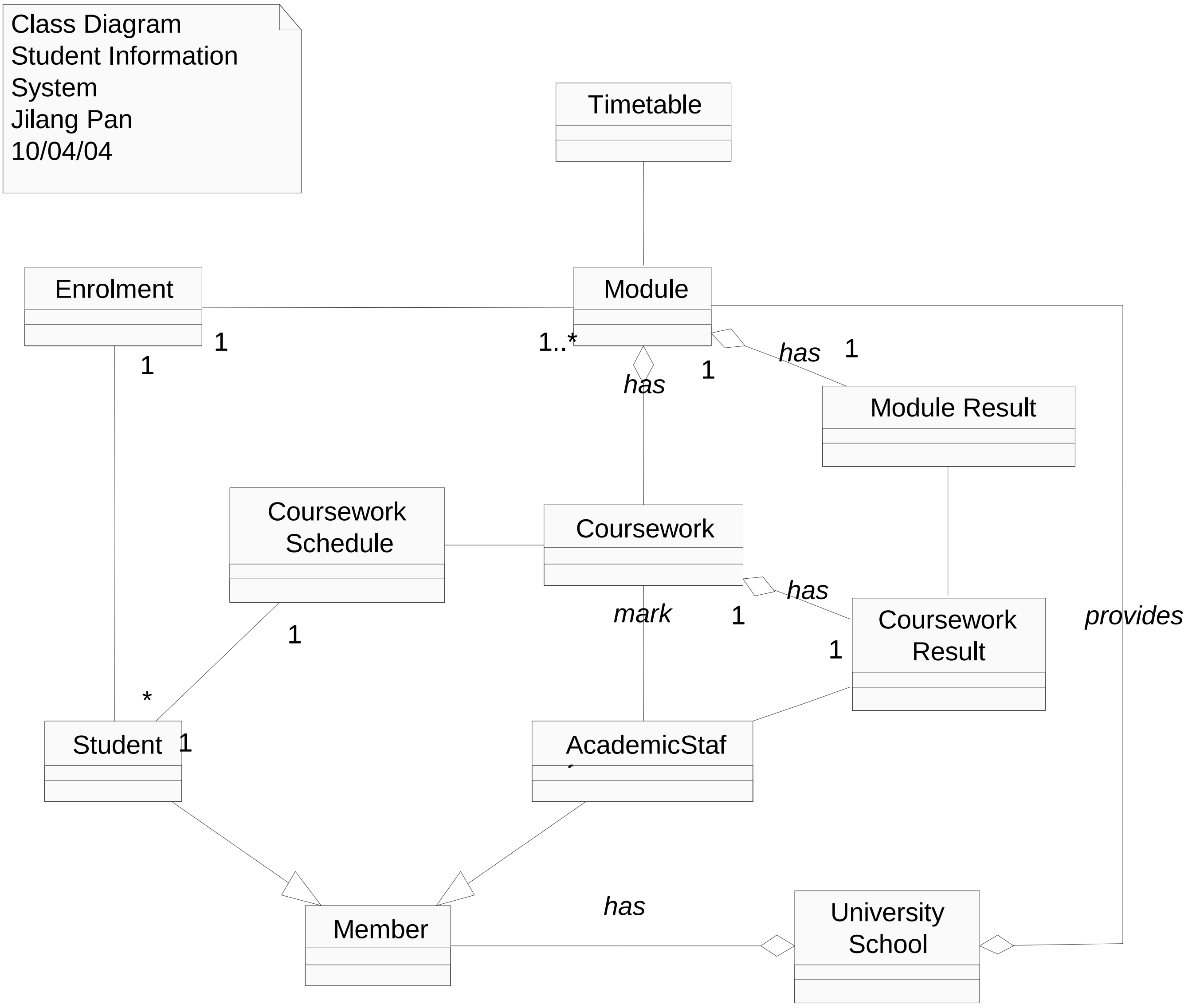
The diagrams below show parts of the system from different aspects. The Deployment Diagram shows the physical units within the system, with connections. Super-imposed on top of these are the individual components, and their dependencies on one another.

Systems Deployment Diagram


PC-Client
Web browser
<<TCP/IP>>
System web server
Web page generator
Server side script
Deployment Diagram Interaction of the School Student Information System and the central university database
<<LAN>>
University central database
Central student database
<<ODBC Link>>
System data server
General student information database
Student coursework database

Class Diagram

The Class Diagram shows the system from a technical perspective, allowing software engineers to start developing the system.



Glossary

SIS
School Information System. Which means the intranet of School of Computing.

SoC
School of Computing in university of Leeds

Edass
The coursework marking system in school of computing

Banner
University of Leeds' central student administration system

PC - Personal Computer
Computer running windows, originally based on IBM technology.

Username and Password
The username and password provide by the School for its member.

Dynamic Web Page Generation
A web page will be generated when requested by a user, from information taken from the database.

LAN
Local Area Network – Implemented within constraints of a building or across a campus (limited geographic area).

Server
High powered computer that stores data for access across a network.

Database
The data storage for maintaining and recording student information as well as the other data of school.

TCP/IP
Software language that computers use to communicate across the Internet.

UML
Unified Modelling Language.

WWW
World Wide Web.

Appendix D Requirement Specification of SIS
Appendix E Requirement Specification of System in LUBS
Appendix F Requirement Specification of System in SUFE

(They are stored in the attached floppy disk.)