NAME: SUDHANSU SEKHAR SWAIN

ROLL: 118CS0689

SOFTWARE REQUIREMENT SPECIFICATION

STUDENT COURSE REGISTRATION

TABLE OF CONTENTS

1.INTRODUCTION

This software requirement specification is made with the purpose of outlining the software architecture and design of the Student Registration System in detail. The document will provide developers an insight in meeting client’s needs efficiently and effectively. Moreover the document facilitates communication and understanding of the system by providing several views of the system design. The online student course registration system is automated system where the user can register the student for various courses. This is proposed for automating the student’s course registration. While the student joins any educational institution his admission is made on the basis of his previous records. The students who wish to the institution must be given with the available course details. The student is allotted with a seat in the institution based on the marks that he scored in the institution he studied previously. After the confirmation of his joining the student must be given with new identity and records as per the institution.

1. PURPOSE

The purpose of this document is to present a detailed description of the Online Course Registration System. It will explain the purpose and features of the system, the interfaces of the system will do, the constraints under which it must operate and how the system will react to external factors. This document is intended for both stakeholders and developers of the system.

2. SCOPE

The Online Course Registration System will enable students to register for courses on-line. The Online Course Registration System allows professors to select their teaching courses and to maintain student grades. Course Registration System will help the student to gather information about a particular course and then they can easily register them self in a particular course.The management of the institution can easily see the records of the students and course and fees.

3. GLOSSARY

OS: Operating System

GB: Gigabyte

TB: Terabyte

RAM: Random Access Memory

NIC: Network Interface Card

4. REFERENCES

www.google.com

www.scribd.com

www.app.diagram.net

5. OVERVIEW

The second section, the General Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.
The third chapter, Specific Requirements section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

2.OVERALL DESCRIPTION

This is proposed for automating the student's course registration. While the student joins any educational institution his admission is made on the basis of his previous records. The students who wish to the institution must be given with the available course details.

2.1 PRODUCT PERSPECTIVE

Hardware interfaces

Hard disk: The database connectivity requires a hardware configuration that is on-line. This makes it necessary to have a fast database system running on high rpm hard disk permitting complete data redundancy and back-up systems to support the primary goal of reliability

The system must interface with the standard output device, keyboard and mouse to interact with this software.

Software interfaces

Back End: MS-Access

Front End: Microsoft Visual Basic 6.0

No specific constraints on memory.

2.2 PRODUCT FUNCTIONS

2.2.1 The system shall be able to access courses. Students can select course and instructors can update course.

2.2.2 The system shall support group registration features. It will have the functional capabilities to access course, select course, do payment, create account.

2.2.3 The system allows admins to modify course.

2.3 USER CHARACTERISTICS

2.3.1 The student expected to be Internet literate Once he/she can log in the system and navigate between WebPages he/she can use basic functionality of the system.
2.3.2 Instructor expected to be internet literate and be able use more complex functionality of the system.

2.4 CONSTRAINTS

2.4.1 The system shall use mysql database for all data management tasks.

2.4.2 A student can register a course if: He/She comes among the required number of students with highest marks in previous college.

2.4.3 The system must run in windows OS environment.

3 SPECIFIC REQUIREMENTS

3.1 EXTERNAL INTERFACE REQUIREMENTS: The system must interfaces icons or wizard. The GUI screen makes the user friendly.

3.1.1 HARDWARE INTERFACES: Intel i5 processor, 8 GB RAM, 1 TB storage, 1 NIC.

3.1.2 SOFTWARE INTERFACES: Windows 10, Google chrome/Mozilla firefox, Netbeans, MS SQL Server 2005, Language: C++.

3.1.3 COMMUNICATION INTERFACES: The system is user interface and not command line prompt.

3.2 FUNCTIONS

3.2.1 LOGIN: Students, and the Course Registrar shall provide a valid ID and password for entry to the Registration System. Users are assigned their ID and a temporary password at the time they apply for admission to the College. The system shall enable a user to change their temporary password afetr they get a seat in the college.

Input: ID and password to login.

Output: Takes into the home page of the respective user for correct information else displays an error message.

3.2.2 Register for Courses
The system shall display available courses to the student upon request. The student shall be able to query based upon course name, course code, and department. The system shall accept course registrations from students and shall validate based upon course availability, schedule conflicts, and completed pre-requisite courses. The system shall notify the student immediately if the course registration does not succeed. The system shall allow the student to change course selections prior to the end of the registration period.

Input: Choice of course to apply for.

Output: Registration successful message upon successful application else error message.

3.2.3 View Course Catalog Information
The course catalog information maintained in the Course Catalog Database shall be displayed to the user upon request. Users shall be able to query for information based upon course name, course code, professor name, and department.

Input: Request to view course catalog information.

Output: Course catalog.

3.2.4 Apply for course

Students can apply for the course using this functionality.

Input: The courses student wants to apply for.

Output: Student join the list of usser who applied for the course.

3.2.5 Student Billings
The system shall send notifications to the Billing System following closure of the Registration period. These notifications shall include student name, address, course selections, and payment due. The student if gets selected for the course is further notified.

Input: Auto generated for those students who haven't paid for the registration and for those who gets selected into the college.

Output: Message to respective students.

3.2.6 Enter, Update, and View Student Information
The system shall accept and update student information, including student ID, name, address, phone number, and email address. Student information shall be available to the Professors and Course Registrar for viewing. The system shall ensure that a student only has access to his or her own student information. The Registrar maintains student information.

Input: Choice from user whether to enter/update/view student information.

Output: User taken to the page depending on the choice they make. System further takes input from user and strores in database if user wants to enter the details.

3.2.7 Maintain student information: This use case is started by the registrar. It provides the capability to create, review, modify, and delete student information.

USE CASE DIAGRAM

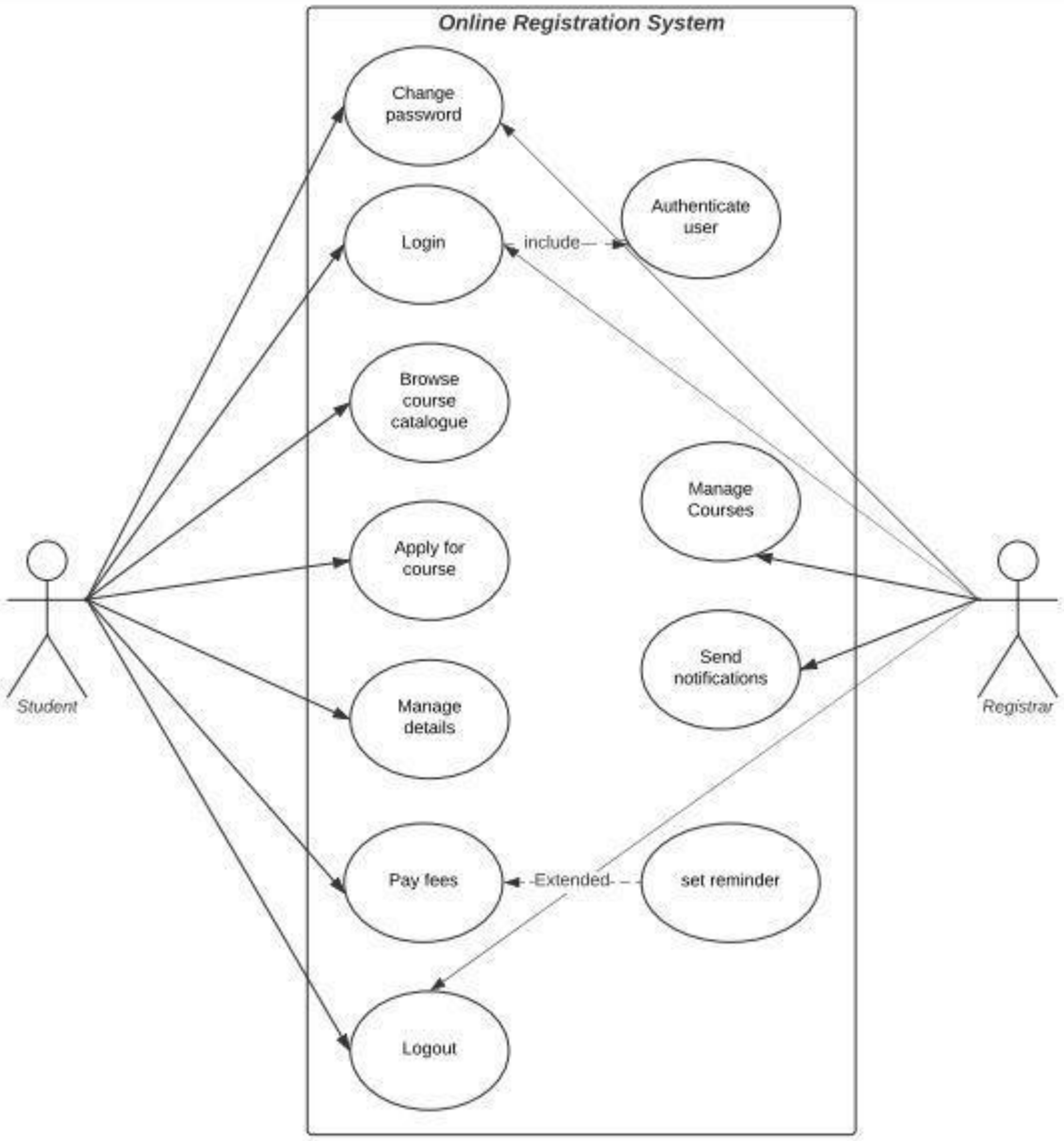


Figure: Use case diagram for online registration system

3.3 PERFORMANCE REQUIREMENT

3.3.1 The system shall support up to 2000 simultaneous users against the central database at any given time, and up to 500 simultaneous users against the local servers at any one time.
3.3.2 The system shall provide access to the legacy course catalog database with no more than a 10 second latency.

3.3.3 RELIABILITY: The system shall be available 24 hours a day and 7 days a week.User can access anytime within the deadline.

3.3.4 SECURITY: 1. The system must prevent students from changing any schedules other than their own, and professors from modifying assigned course offerings for other professors.
2. Only the Registrar is allowed to change any student information.

3.4 LOGICAL DATABASE DESIGN

3.4.1 The system should contain databases that include all necessary information for the product to function according to the requirements. These include relations such as student information and course details.

3.4.2 The student information includes information of the student such as name, address, date of birth, result in previous college.

3.4.3 The course details includes information such as the details of professor who will teach it, the details of the course and the application of the course.

3.5 DESIGN CONSTRAINTS

3.5.1 This product requires a total of 2 tables, the student details, the course details.

3.5.2 The student details have the details of student alongside the courses they apply for and if they are selected for the course or not.

3.5.3 The course details have the details of the courses.

Software Requirements Specification

Automated Banking System

(Version 1.0)

Table of Contents:

2. Overall Description

- Product Perspective
- Product Functions
- User Characteristics
- Constraints
- Assumptions and Dependencies

3. Specific Requirements

- External Interfaces
- Functions
- Performance Requirements
- Logical Database Requirements
- Design Constraints
- Software System Quality Attributes
- Object Oriented Models

1. Introduction:

The software **ABSXclr 4.8**^{AM} **Version1.0** is to be developed for Public Sector Banks. A Public Sector Banks (PSBs) are a major type of bank in India, where a majority stake (i.e. more than 50%) is held by the government. Through **ABSXclr 4.8**^{AM}, customers interact with a user-friendly interface that enables them to access their bank accounts and perform various transactions.

1.1 Purpose:

This SRS defines External Interface, Performance and Software System Attributes requirements of **ABSXclr 4.8**^{AM}. This document is intended for the following group of people:-

Ø Developers for the purpose of maintenance and new releases of the software.

Ø Management of the Public Sector Banks.

Ø Documentation Writers.

Ø Testers.

1.2 Scope:

This document applies to Automated Banking System ABSXclr 4.8^{AM}. This software facilitates the user to perform various transaction in his account without going to bank. This software offers benefits such cash withdrawals, balance transfers, deposits, inquiries and other banking related operations for customers.

The software takes as input the Card Number and Pin Number of the user for login purposes. The outputs then comprise of an interactive display that lets the user select the desirable function that he wants to perform.

The software is expected to complete in duration of **5 months** and the estimated cost is **Rs10 lakhs**.

1.3 Definitions, Acronyms and Abbreviation:

ASB	Automated Banking System
PSBs	Public Sector Banks where a majority stake is held by the government
BMS	Bank Management Software developed by KPM Bank.
CMS	Card Management Software developed by KPM Bank.
Smart Card	Card without hardware which stores the user‘s private keys within a tamper proof software guard.
SRS	Software Requirements Specification.
Internet	An interconnected system of networks that connects computers around the world via the TCP/IP protocol.

1.4 References:

The references for the above software are as follows:-

i. www.google.co.in

ii. www.wikipedia.com

iii. IEEE. Software Requirements Specification Std. 830-1993.

iv. Private Sector Banks, UMBC Branch.

v. Godfather of SRS by Michael Jackson.

1.5 Overview:

The SRS contains an analysis of the requirements necessary to help easy design.

Section 1.0 discusses the purpose and scope of the software.

Section 2.0 describes the overall functionalities and constraints of the software and user characteristics.

Section 3.0 details all the requirements needed to design the software.

2. Overall Description:

The overall description provides interface requirements for the Banking system, product perspective, hardware interfaces, software interfaces, communication interface, memory constraints, product functions, user characteristics and other constraints.

2.1 Product Perspective:

Hardware Interfaces: The database connectivity requires a hardware configuration that is on-line. This makes it necessary to have a fast database system (such as any RDBMS).

Software Interfaces: Back End: MS Access 2007

Front End: Microsoft Visual Basic 6.0

Operations:

The user can create a new account.

The existing user can access his account and view his balance by entering his details.

The user can deposit and withdraw money from his account.

2.2 Product Functions:

Creating a New Account: The user should provide his personal details to facilitate the bank clerk to create a new account.

The user should provide: *Customer Name, Customer Address, Required Account Type, Pin Number* and *Initial Deposit*.

Operating with Created Account: The user should be able to operate with his new account after entering the *card number, pin number, account type, transaction type* and *amount involved in the transaction*.

2.3 User Characteristics:

The intended users of this software need not have specific knowledge as to what is the internal operation of the system. Thus the end user is at a high level of abstraction that allows easier, faster operation and reduces the knowledge requirement of end user.

The Product is absolutely user friendly, so the intended users can be the naïve users. Any person who knows to use the mouse and the keyboard can successfully use this product.

2.4 Constraints:

At the time of creating the new account, each user gives a pin number and is provided with a unique card number that must be used for further transactions. Hence the user is required to remember or store these numbers carefully.

At the time of creating the new account, the initial deposit should not be less than the specified amount.

2.5 Assumptions and Dependencies:

The user cannot forget the pin number and the unique card number.

The Initial deposit is not less than the specified amount.

3. Specific Requirements:

Software Requirements specified in sufficient detail so that designers can design the system and testers can verify whether requirements meet. Requirements are stated that are externally perceivable by users, operators, or externally connected systems.

3.1 External Interfaces:

User Interface: The interface provided to the user should be a very user-friendly one. A login screen is provided in the beginning for entering the required username/pin no. and account number. An unsuccessful login leads to a reattempt (maximum three) screen for again entering the same information.

After the login, a screen with a number of options is then shown to the user. It contains all the options along with their brief description to enable the user to understand their functioning and select the proper option.

Hardware Interface: There are various hardware components with which the machine is required to interact. The card reader shall be a magnetic stripe reader and have Smart card option. There shall be a 40 column dot matrix receipt and statement printer. Screen resolution of at least 800X600-required for proper and complete viewing of screens. Higher resolution would not be a problem.

Software Interface: In order to perform various different functions, this software needs to interact with various other software.

The transaction management software used to manage the transaction and keep track of resources shall be BMS version 2.0. The card management software used to verify pin no and login shall be CMS version 3.0.

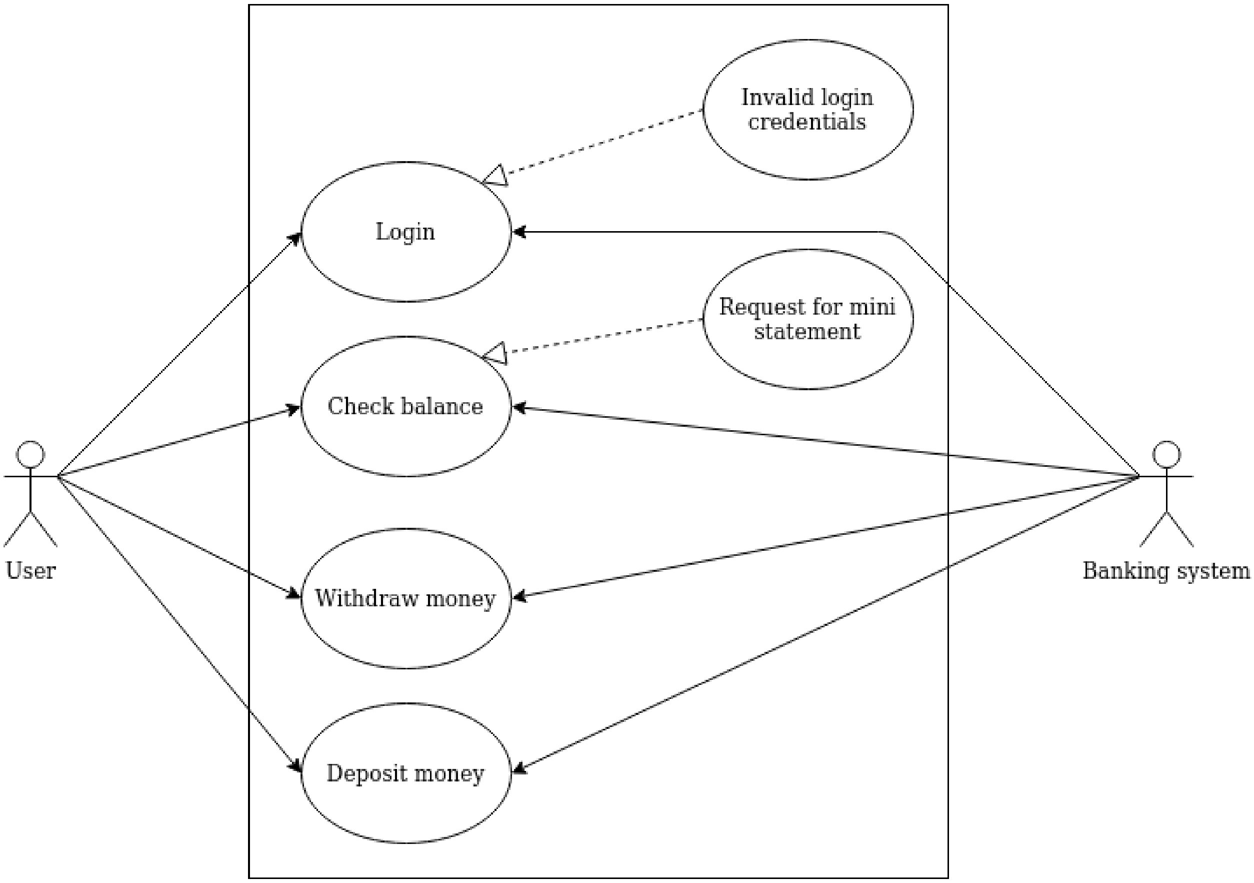
3.2 Functions:

Create Account: User Details are taken as Input and confirmation message is shown as output regarding the successful creation of account.

Login: While Login take Input as Card Number and pin Number and Output will be shown as the Home Page of the software in successful login.

Check Balance: No Input and Output will be the current balance in the account.

Withdraw Money: Take Input as Money to be withdrawn and Output a confirmation message after withdrawal is completed successfully.



Use case diagram for automated banking system

Deposit Money: Take Input as Money to be deposited and Output a confirmation message after deposition is completed successfully.

3.3 Performance Requirements:

Capacity: The Software shall provide customers a 24 hour service.

Dynamic Requirements: The card verification time and pin verification time must not exceed 0.8 sec. under normal server workload and 1 sec. under peak server workload.

Cash withdrawal transaction time must not exceed 4 sec. under normal server workload and 5 sec. under peak server workload.

Receipt printing time after must not exceed 3 sec. under normal server and peak server workload. Touch screen and button response time must not exceed 5000ms.

Quality: The primary objective is to produce quality software. This is achieved through consistency of all code and all functionality thoroughly tested.

3.4 Logical Database Requirements:

The system should contain databases that include all the necessary information for the product to function according to the requirements. These include relations such as Customer Details and Account Details.

Customer details refer to the customer's name and address. Account details of the customer include the card number, account type, transaction type and the pin number given by the user to be used at the time of the transaction at the bank.

3.5 Design Constraints:

Cost: The Software cost should be less than 12 lakhs.

Time: The Software must be completed within 6 Months.

Size: The Software Size must not exceed 1GB.

3.6 Software System Quality Attributes:

Reliability: The data communication protocol shall be such that it ensures reliability and quality of data. The memory system shall be of non-volatile type.

Availability: The product will have a backup power supply in case of power failures. Any abnormal operations shall result in the shutting down of the system. There should be no inconsistency introduced in the account during transaction when the system is shut down.

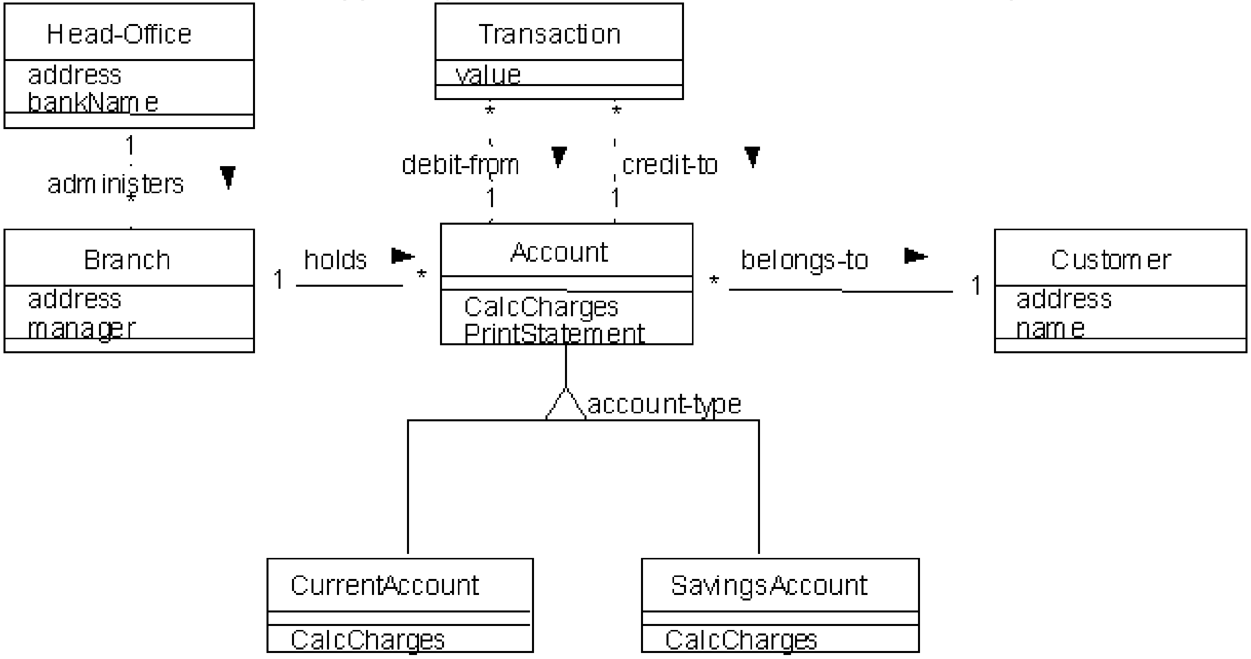
Security: The system shall have two levels of security i.e. Card and Pin verification both authenticated by the CMS software. Passwords shall not contain name of customers as they are easy to be hacked. User should be provided with only three attempts for login failing which his card needs to be blocked.

Maintainability: The system should have the mechanism of self-monitoring periodically in order to detect any fault. The system should inform the main branch automatically as soon as it detects any error. The kind of fault and the problem being encountered should also be mentioned by the system automatically.

3.7 Object Oriented Models:

Object-oriented modelling (OOM) is an approach to modelling an application that is used at the beginning of the software life cycle when using an object-oriented approach to Software Development.

It is the construction of objects using a collection of objects that contain stored values of the instance variables found within an object. It allows for object identification and communication while supporting data abstraction, inheritance and encapsulation.



The object model visualizes the elements in a software application in terms of objects.An object is a real-world element in an object–oriented environment that may have a physical or a conceptual existence. For eg:- Account, Customer, etc.

Software Requirement Specification
Library Management System

Table of Contents

1. INTRODUCTION

a. Purpose

i. The purpose of this SRS is to define the requirements in the development of a system of library management.

ii. Any individual who wants to inquire, borrow the books and return them. is the intended audience.

b. Scope

. The product is titled Library Management System.

i. The product performs the tasks below:

1. Enquire about the availability of books.
2. Borrow books if available.
3. Return the borrowed books.

c. Definitions, Acronyms and Abbreviations

. DBMS – Database Management System.

d. References

. IEEE standard 830-1998 recommended practice for SoftwareRequirements Specifications-Description.

e. Overview

. The SRS includes an analysis of the requirements required to facilitate easy design.

i. Interface requirements for the library management system, product perspective, hardware interfaces, software interfaces, communication interface, memory constraints, product functions, user characteristics and other constraints are provided in the overall description.

ii. Successive pages illustrate the features and legal and functional constraints enforced by typical naive users accessing the system that affect the library management system in any way.

2. THE OVERALL DESCRIPTION

. Product Perspective

. Hardware interfaces

1. Hard disk: A hardware configuration that is online is required for database connectivity. To support the primary objective of reliability, it is necessary to have a fast database system running on a high rpm hard disk that allows complete data redundancy and back-up systems.
2. The system must interface with the standard output device, keyboard and mouse to interact with this software.

i. Software interfaces

1. Back End: MS-Access 2007
2. Front End: Microsoft Visual Basic 6.0

ii. Memory Constraints

1. No specific constraints on memory

iii. Operations

1. The software allows three modes of operations

a. Enquire about the availability and status of books.

b. By extracting the username and password the software allows the user to borrow a maximum of three books.

c. By extracting the username and password the software allows the user to return the borrowed books.

a. Product Functions

. Enquire about the availability and status of books.

i. Search the availability of books by entering the title of the book.

ii. Search the availability of book by entering the author of the book

iii. The software validates the authentic user by extracting their username and password.

iv. After the validation of the user software allows the user to borrow a maximum of three books based on the number of books which were already borrowed.

v. After the validation of the user software allows the user to return the books, which were borrowed.

b. User characteristics

. The intended users of this software do not need to have particular knowledge of what the system's internal operation is. The end user is therefore at a high level of abstraction, allowing for easier, faster operation and reducing the end user's knowledge requirements.

i. The product is entirely user-friendly, so naive users can be the intended users.

ii. The product does not expect any technical background to be owned by the user. This product can be used successfully by any individual who knows how to use the mouse and the keyboard.

c. Constraints

. There are no options for retrieving a password or username if it is forgotten or lost, so the user is required to remember or store the username and password. The user has a unique username and password.

3. SPECIFIC REQUIREMENTS

. External Interfaces

. The system takes in the name of the author or the title of the book and checks the database to return the availability of the book.

i. The system takes in the username and card number of a user and authenticates the user into the system. Following this the system checks the number of books already borrowed by the user and returns if the user can borrow the book or not.

ii. The system takes in the user details and the book details and allows the return of borrowed books.

a. Functions

. Book entry: In this module we can store the details of the books.

i. Register student: in this module we can keep the details of the new student.

ii. Book issue: This module is used to keep a track of book issue details.

iii. Book return: This module enables to keep a track of return the books.

b. Performance Requirements

. The system should handle a reasonable number of users without break or inconsistency.

c. Logical Database Requirements

. The system should contain databases that include all necessary information for the product to function according to the requirements. These include relations such as user details and book details

i. The user details refer to the information such as name, card number, no. of books borrowed, the title and the name of the author of the books that were borrowed.

ii. The book details refer to the information such as the title of the book, author availability status and the number of copies that is available.

d. Design Constraints

. Simultaneous requests to borrow same book must be handled by prioritising the user who sent request to the system first

e. Software System Quality Attributes

. Reliability

1. The memory of the system should be non volatile type

i. Availability

1. The system is expected to be available during the time the library is open i.e 14 hours per day.
2. Any power outage must trigger a backup of the ongoing requests to issue or return books so that no inconsistency is introduced.

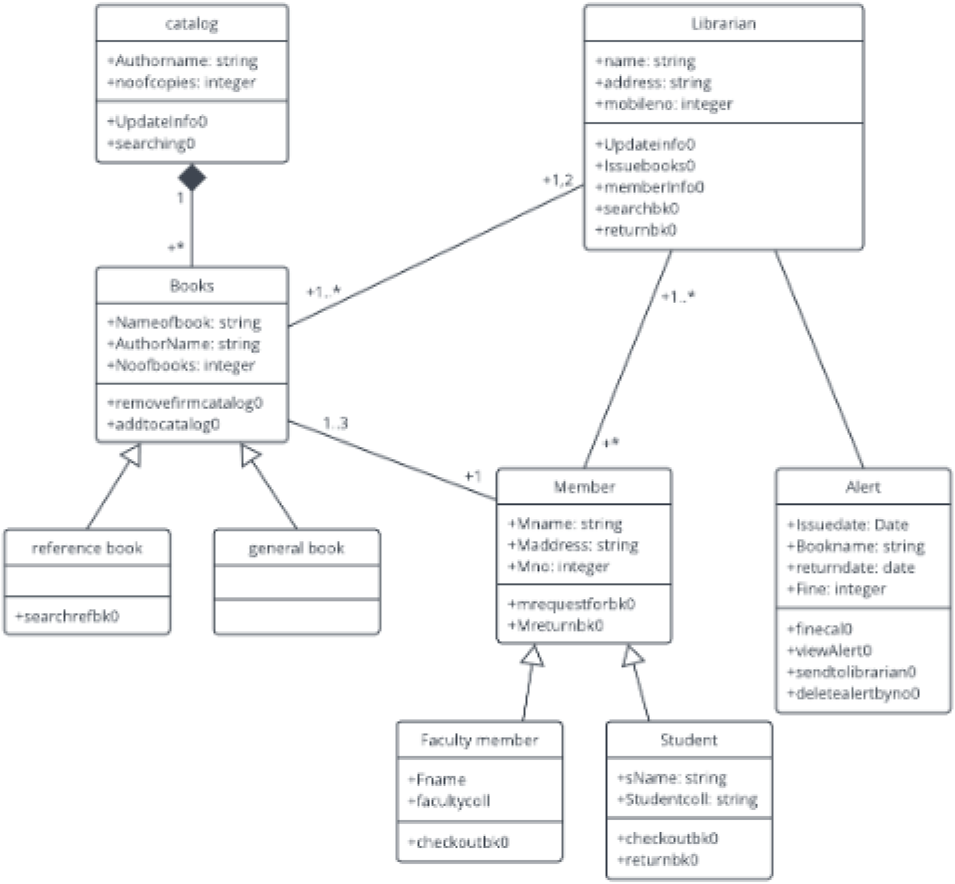
ii. Security

1. Card number must be of 10 digits.
2. Card number must be alphanumeric.
3. Username can contain digit, hyphen and underscore

iii. Maintainability

1. The system components should be robust and easily serviceable.
2. The system should have the mechanism to self-monitor periodically in order to detect any discrepancies in the database.

f. Object Oriented Models

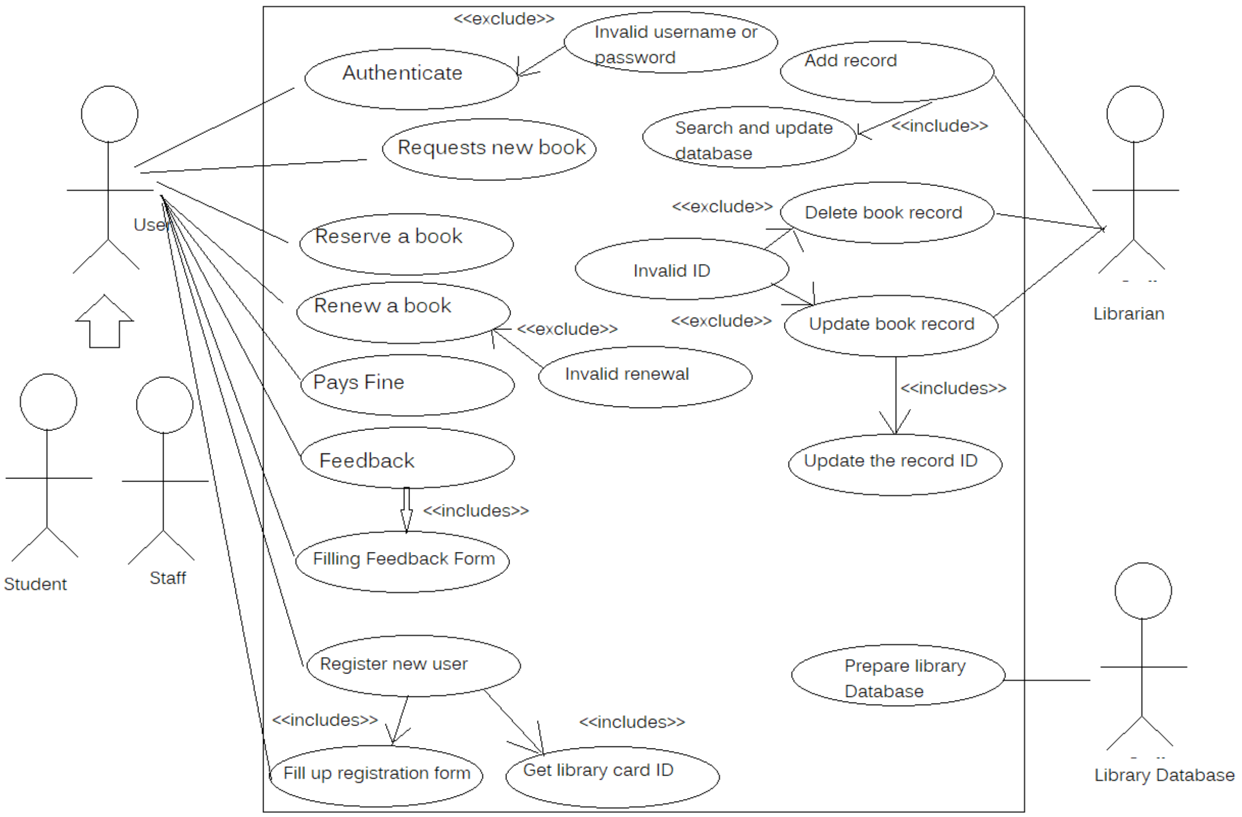


4. APPENDICES

a. COCOMO : The Constructive Cost Model (COCOMO) is a procedural software cost estimation model developed by Barry W. Boehm. The model parameters are derived from fitting a regression formula using data from historical projects (63 projects for COCOMO 81 and 163 projects for COCOMO II).

Reference: COCOMO - Wikipedia

5. USE CASE DIAGRAM


<<exclude>>
Invalid username or password
Authenticate
Add record
Requests new book
Search and update database
<<include>>
User
<<exclude>>
Delete book record
Reserve a book
Invalid ID
Renew a book
<<exclude>>
<<exclude>>
Update book record
Librarian
Invalid renewal
Pays Fine
<<includes>>
Feedback
Update the record ID
<<includes>>
Student
Staff
Filling Feedback Form
Register new user
Prepare library Database
<<includes>>
<<includes>>
Fill up registration form
Get library card ID
Library Database