Software Requirements Specification Document

1 Abstract

This is the requirement specification document for the library management system to be built for the college library. This system will be focussing on automating the library management processes.

2 Introduction

2.1 Purpose

The purpose of this document is to describe the external requirements for a library management system. It also describes the interfaces for the system.

2.2 Scope

This document is the only one that describes the requirements of the system. It is meant for use by the developers and will be the basis for validating the final delivered system. Any changes made to the requirements in the future will have to go through a formal change approval process. The developer is responsible for asking for clarifications, where necessary, and will not make any alterations without the permission of the client.

2.3 Definitions, Acronyms, Abbreviations

Not applicable.

2.4 References

Not applicable.

2.5 Developer's Responsibilities

The developer is responsible for (a) developing the system, (b) installing the software on the client's hardware, (c) conducting any user training that might be needed for using the system, and (d) maintaining the system for a period of one year after installation.

3 General Description

3.1 Product Functions Overview

A college library management system is a piece of software created to effectively oversee library operations at a college or university. It lets librarians and employees manage duties unique to the college context, such as categorizing. This system assists with keeping track of user information, administering the college's collection of printed and digital resources, and handling student and faculty borrowing. It simplifies procedures such as reserves, and offers online catalog access. It also helps the college library allocate resources and make decisions.

3.2 User Characteristics

The main users of this system will be patrons, who can use the software to make their library experience easy.

The college library management system offers capabilities for searching, borrowing, administrative activities, and resource management. It also caters to students, instructors, librarians, and administrators. It guarantees the best possible user experience for each type of user.

3.3 General Constraints

The system should run on Sun 3/50 workstations running UNIX 4.2 BSD.

3.4 General Assumptions and Dependencies

Not applicable.

4 Specific Requirements

4.1 Inputs and Outputs

There are 3 input files and produces 4 types of outputs.

Input_form_1: This form is used to collect data about patrons registered with the library. The inputs contain information like patron_no, user_type, patron_name. The information will be stored as follows in the patron database.

Patron_no	user_type	patron_name
ptxxxx	student	abc
ptxxxx	faculty	xyz

Where ‘ptxxxx’ is a unique identification number for each patron and xxxx are any valid 4 digit number. The user_type should either be student or faculty. The name of the patron can be any valid string. An example of the Patron database is:

Patron_no	user_type	patron_name
pt0011	student	steve
pt1234	faculty	teresa
pt2453	student	kang
pt4918	student	kim

Input_form_2 : This form is used to collect the information about the books to create or update the library database. It contains the information about the book including the isbn(8 digit unique code for each book), book_code (is same for the duplicates of the same book), category, title, author, availability and self_no. The format of the library database is:

ISBN	book_code	category	title	author	availability	self_no
Xxxxxxxx	aadddd	in library use	book_name	author_name	avl	ax-xxx
Xxxxxxxx	aadddd	for instructors	book_name	author_name	lent	ax-xxx
Xxxxxxxx	aadddd	to lend	book_name	author_name	lent	ax-xxx

The category of the book can be either “in library use only”, these books cannot be issued to anyone, or “for instructors only” or “to lend”, meaning it can be lent to anyone. The availability of a book can either be ‘avl’ or ‘lent’. The self_no ‘ax-xxx’ is a combination of letters and digits (‘a’ is replaced with a letter and ‘x’ is replaced with digits). An example for the file is:

ISBN	book_code	category	title	author	availability	shelf_no
19385756	cs2003	in library use	computers	ram singh	avl	a5-123
19473528	en1846	for instructors	the journal	preety rao	lent	b6-452
29472648	mc2849	to lend	machines	pankaj pande	lent	s2-122

Input_form_3: This form is used to enter information about the book to be issued to the patron. It contains fields like patron_no, ISBN, date_of_issue. This information is stored in a separate table that relates patron_no to ISBN.The return date is initially the original return date of the book, it will be updated to the date the book was returned once the patron returns the book along with the fine, that will be automatically calculated and updated, once the book is returned. The issue_relation table has the following format:

patron_no	ISBN	date_of_issue	return_date	fine_incurred
Ptxxxx	Xxxxxxxx	dd-mm-yy	dd-mm-yy	0

Input_form_4: This is used to get information about the books to be returned to the library. It gets information about patron_no, ISBN, return_date. These inputs on processing will affect the issue_relation table as well as the library database. The return date of the book and fine_incurred of the issue_relation table are updated while returning.

Input_form_5: This is used to get input for performing search functions on the library database. The input fields are book_title, book_author. The user may input either one or both fields.

Input_form_6: This form is used to get librarian information for registration and login. The input fields include librarian_no, librarian_name, password. The librarian database used for librarian registration and login has the following format:

librarian _no	librarian_name	password
Lxxx	abc	********

Where the librarian_no is a unique number for each librarian and has a format of ”lxxx” (‘l’ followed by 3 digits).the password should have at least 8 characters.

Output_1 : This contains a report on the books borrowed by a particular patron. The report is a view from the databases and contains fields like ISBN, book_title, book_author, issue_date, return_date.

Output_2 : This displays to the user the overdue that has been calculated from the return_date.

Output_3 : It gives a list of book information that matches the search criteria.

Output_4 : This is a report on errors that occur on particular requests placed. They include:

- E1 - error in issuing the book
    - E1.1 - The user type is not in hand with the book type.
    - E1.2 - Maximum number of books issued.
    - E1.3 - Book is not to be lent.
    - E1.4 - Invalid patron information.
- E2 - error in returning the book
    - E2.1 - Fine not paid
    - E2.2 - Book can't be returned on the same day of issue.
    - E2.3 - Invalid patron information.
- E3 - error in searching
    - E3.1 - No such book_title or Author

Output_5: error in syntax of any input

- S-err1 - patron information syntax error
    - S-err1.1 - patron_no not as defined.
    - S-err1.2 - user_type not defined.
- S-err2 - book information syntax error
    - S-err2.1 - ISBN not in format.
    - S-err2.2 - shelf number not in format
- S-err3 - issue information syntax error
- S-err4 - return book information syntax error

4.2 Functional requirements

1) Determine if the book can be issued to the patron such that the following constraints are satisfied:
    a) If the patron is a student, they can borrow only 3 books at a time.
    b) If the patron is a faculty member, they can borrow only 6 books at a time.
    c) The book to be issued should not be of "In library use only".
    d) The books marked as "for instructors only" cannot be issued to the patrons other than the faculty members.

    Inputs: Input_form_3

    Output : output_1 , output_4

2) Determine if the book can be returned safely
    a) Calculate and collect the overdue fine if there are any:
        i) Patrons that are faculty can borrow the books for 3 months (including non-working and public holidays) from the date of issue. The fine is Rs.5 per day after this period.
        ii) Patrons that are non-faculty can borrow the books for 1 month (including non-working and public holidays) from the date of issue. The fine is Rs.5 per day after this period.
    b) Book cannot be returned on the same day it has been borrowed.

    Inputs : input_form_4

    Output : output_2, output_4

3) Produce a list of the books with their book_no, title, author, availability, shelf_no that match the search details from the input_form_5

    Inputs : input_form_5

    Output : output_3, output_4

4) The file formats of all three inputs are to be checked against their defined formats. The errors in the formats are to be reported.

    Inputs : input_form_3, input_form_4 and input_form_5

    Output: output_5

4.3 External Interface Requirements

User Interface: Only one user command is required. The file names can be specified in

the command line itself or the system should prompt for the input file names.

4.4 Performance Constraints

For the input_file_1 with 20 patrons, input_file_2 with 30 books and input_file_3 with 5 requests, the results should be generated within a minute.

4.5 Design Constraints

Software Constraints

The system is to be built using java and run under the UNIX/Windows operating system.

Hardware Constraints

The system will run on a Sun workstation with 1 GB RAM. It will be connected to an 8-page-per-minute printer.

Acceptance Criteria

Before accepting the system, the developer must demonstrate that the system works on a part of the library books database. The developer will have to show through test cases that all conditions are satisfied.

Team Members:
Shruti S (21z256)
Aishwarya R (22z431)