# L I B R A R Y  M A I N T E N A N C E  S Y S T E M

## DESIGN DOCUMENT

SUBMITTED BY

LAVANYA SITA TEKUMALLA


- INTRODUCTION 2
- PROBLEM DEFINITION 2
- DATA DICTIONARY 3
- DATA FLOW DIAGRAMS 4
- E R DIAGRAMS 10
- SOFTWARE REQUIREMENT SPECIFICATION
  - INTRODUCTION: 11
  - GENERAL DESCRIPTION: 11
  - FUNCTIONAL REQUIREMENTS 12
  - EXTERNAL INTERFACES 15
  - EXCEPTION HANDLING 15
  - PERFORMANCE CONSTRAINTS 15
  - DESIGN CONSTRAINTS 15
  - FORSEEABLE MODIFICATIONS 15

## INTRODUCTION

This report is the design document for a library maintenance system.

## PROBLEM DEFINITION

Management of books in a Library involves several tasks such as locating a book, issuing of books, renewal of books, receipt of books, adding and removing books from the library and adding and removing users from the records.

These functions are performed based on certain norms of the library.

- *Client* : Head of the Library.
- *End-Users* : Students (to locate a book); Librarians( to keep track of students and books).

The entire process can be divided into six tasks which have been broadly outlined below.

- *Locating a book*: The students can use this facility to find the physical location of the book if they know the author or the title of the book (Catalogue function).
- *Issue of books*: This process involves first checking if the student is authorized to borrow a book. If he is permitted to take the book and if the book is available for issue, the corresponding records are updated.
- *Renewal of books*: This involves generating fine if necessary and updating the TABLE to change the issue date.
- *Receipt of books*: This process involves updating the necessary records after taking the book back. This task also involves calculating the fine in the case of late return.
- *Managing books*: This feature allows addition and removal of books from the library.
- *Student record updation*: This feature provides for accommodating new users and striking specified names off the list.

*Managing books*: To keep track of books, every book is assigned an accession number (in serial order) which uniquely identifies the book in the library.

Further a call number is allotted (conforming to the Dewey Decimal system) to aid easy subject-wise physical retrieval of books .

*Managing students*: To keep track of the students, every student in the university is assigned a unique identification number. (The roll number given to the student can also serve this purpose)

The system interacts with the librarians and the students through two interfaces.

1. *Query interface*: This serves the purpose of the conventional catalogue. The student can directly interact with the system by typing in the name of the author or the title of the book. The system displays a list of books as a result of the query along with their call numbers.

2. *Maintenance interface* : This deals with issue of books, return of books, renewals, adding and removing books from the library and adding and deleting names of students from the library TABLE when their eligibility commences or ceases. A Library staff member must be placed here to receive and issue the books and to interact with the system. The person in charge also needs to check the authentication of the student (by inspecting his identity card. Adding a book is done whenever a new book is purchased for the library (and a bill passed). Deleting a book must follow the decision of a special committee formed for the purpose. A user can be added by checking admission details from the department. A user can be removed by application or on instructions from the department.

***DATA DICTIONARY***

**1.** *Name* : **Books TABLE**
*Usage* : query processor (input).
Issue unit (input and output).
Receipts unit (input and output).
Books addition unit (input and output).
Book removal unit (output)
*Description* : **acc no(Primary key)+date of issue+ call no+ authors name + book title, +availability+ student id**
(if the book is not available, student id gives us the id of the borrower. Date of issue arises only when the book is not available)

**2.** *Name* : **Students TABLE**
*Usage* : Issue unit (input)
Receipts unit (output)
Students addition unit (output)
student removal unit (output and input)
*Description* : **stu ID (Primary key)**
**name+ authentication+ number of books+ acc no of**

*3.* *Name* : **Deleted books TABLE**
*Usage* : Book removal unit (output)
*Description* : **acc no(Primary key)+**
**Athor+ title+acc no of book1+acc no of book 2+ acc no of book3+ acc no of book 4.**

(the last 4 fields are relavent depending on the number of books borrowed. A student can borrow a maximum of four books.)

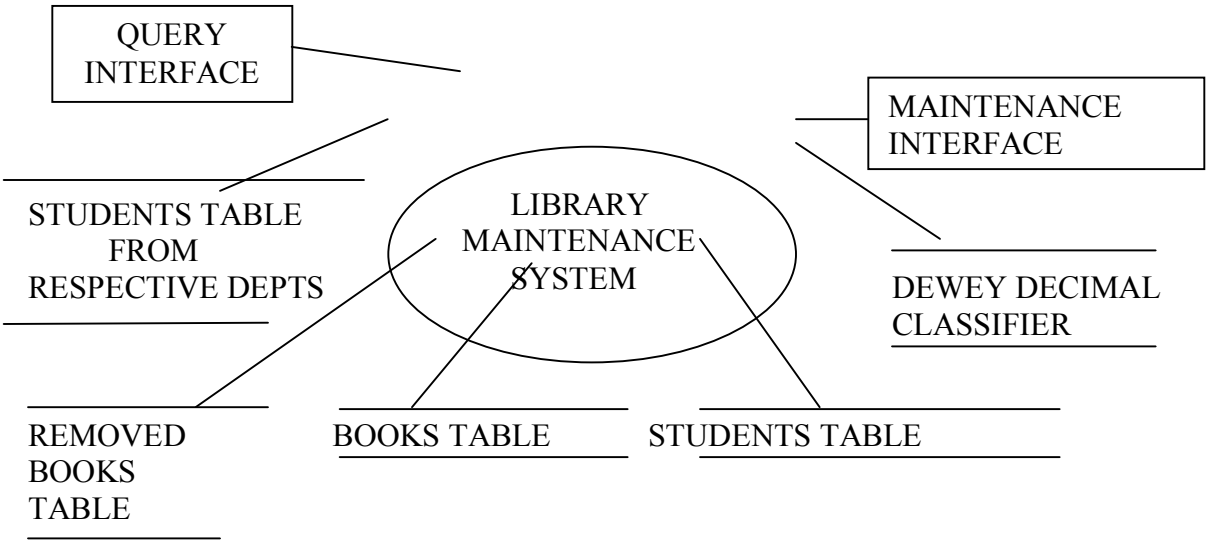
**3.** *Name* : stu TABLE from other depts
*Usage* : Issue unit(input)
*Description* : **stu ID(Primary key )**
**name+ age+ address+ Dept. No**.

**4.** *Name* : **Dewey Decimal classifier**
*Usage* : Books addition unit(input)
*Description* :**Subject(Primary key )+call no.**

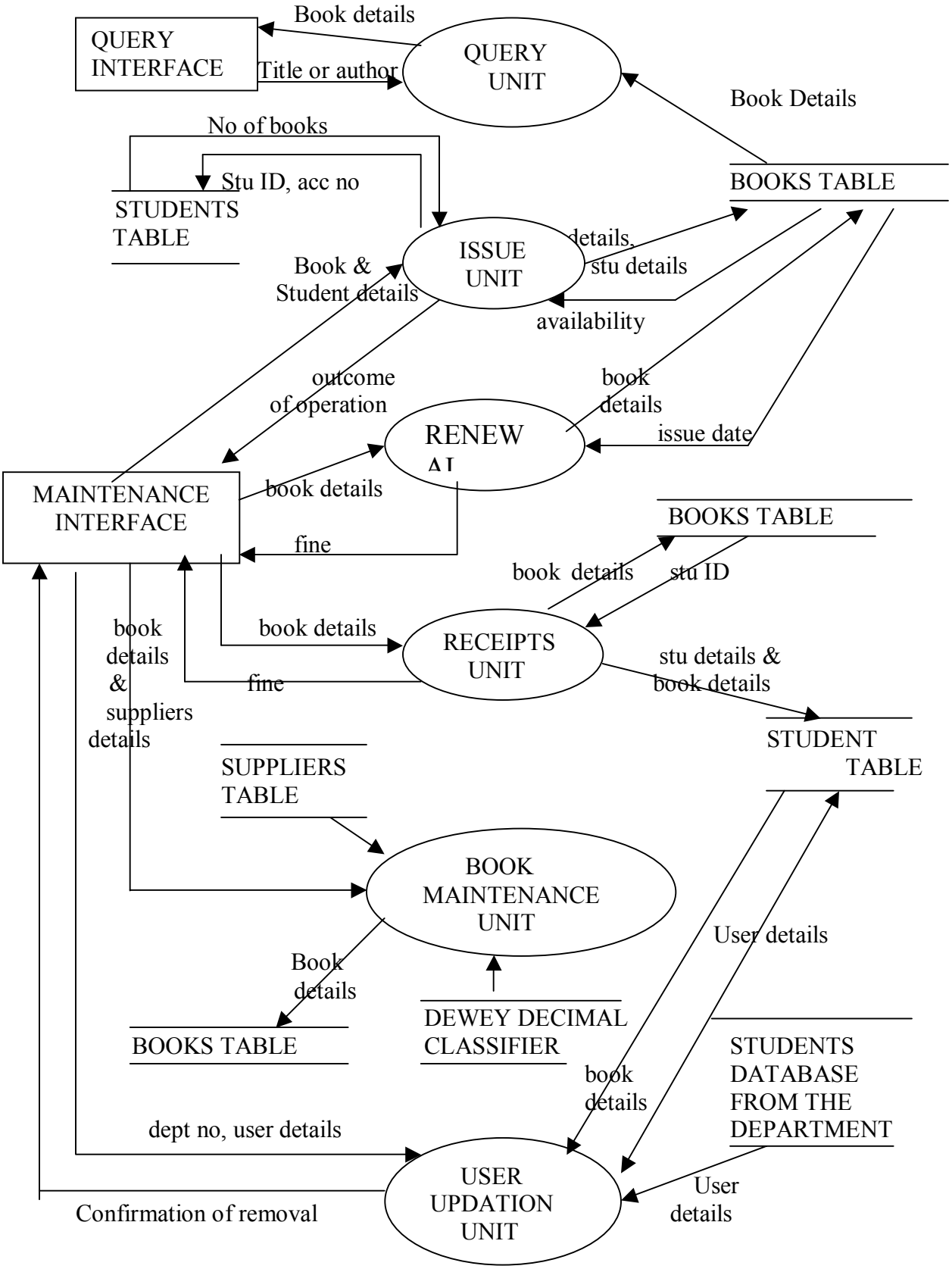
## DATA FLOW DIAGRAMS

The data flow has been specified in three levels.

### Level 0: Context diagram

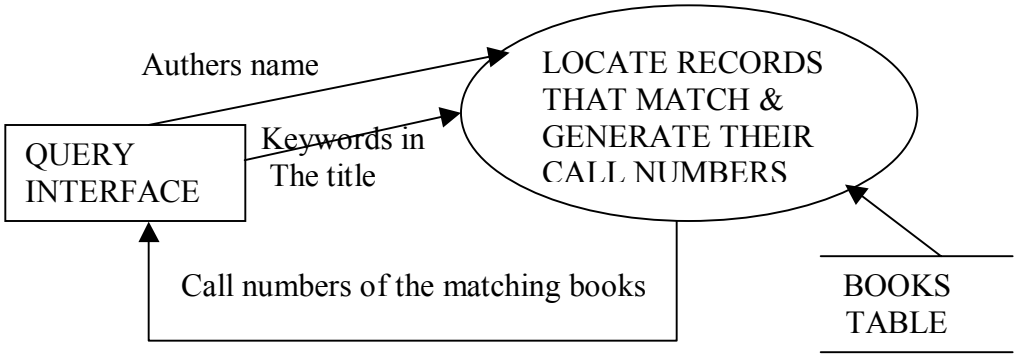


**Level 1:**

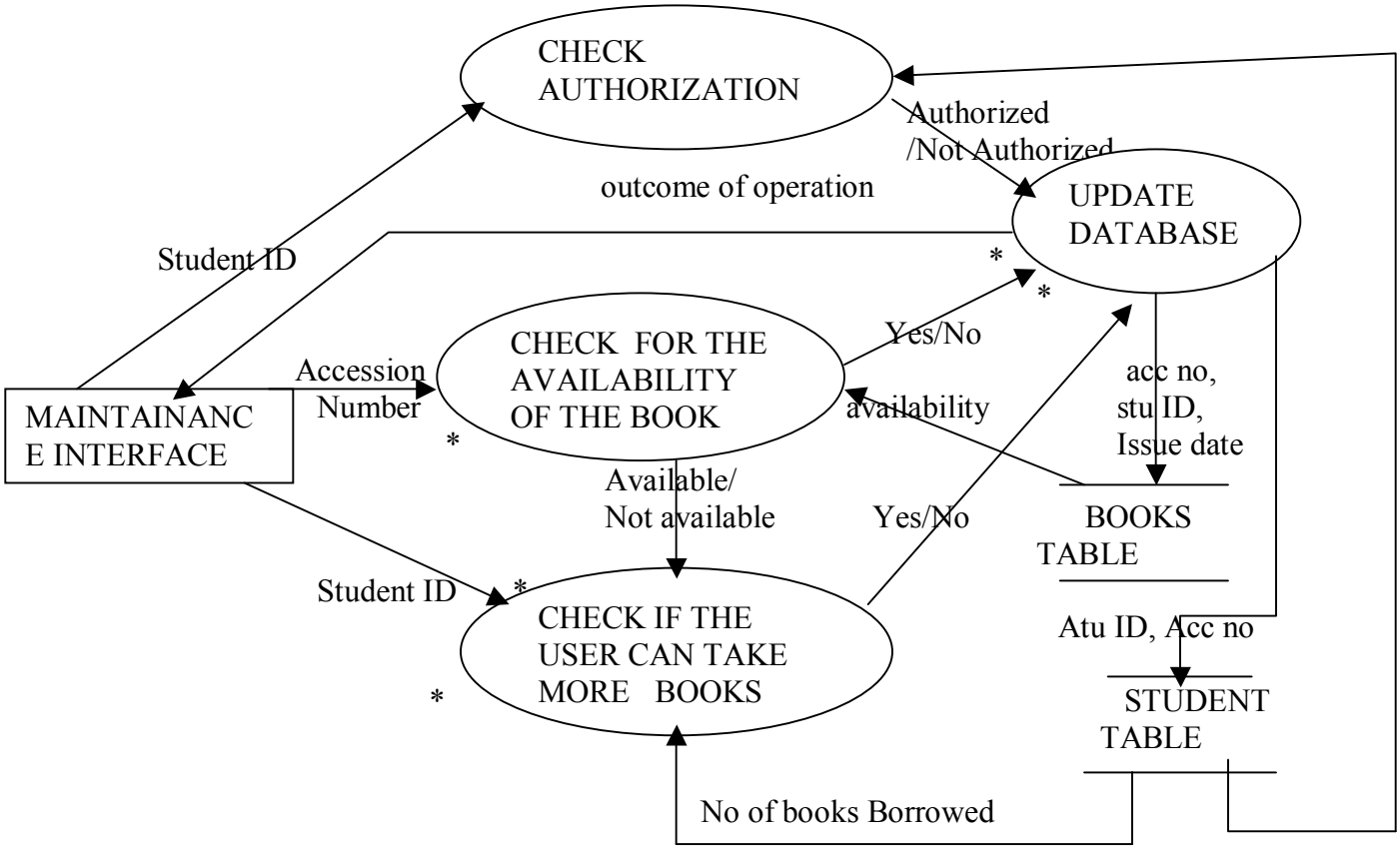

QUERY INTERFACE
Book details
Title or author
QUERY UNIT
Book Details
No of books
Stu ID, acc no
STUDENTS TABLE
BOOKS TABLE
ISSUE UNIT
details,
stu details
Book & Student details
availability
outcome of operation
book details
issue date
RENEW AL
MAINTENANCE INTERFACE
book details
fine
BOOKS TABLE
book details
stu ID
book details
book details & suppliers details
fine
RECEIPTS UNIT
stu details & book details
STUDENT TABLE
SUPPLIERS TABLE
BOOK MAINTENANCE UNIT
User details
Book details
DEWEY DECIMAL CLASSIFIER
BOOKS TABLE
STUDENTS DATABASE FROM THE DEPARTMENT
book details
dept no, user details
USER UPDATION UNIT
User details
Confirmation of removal

**LEVEL 2:**

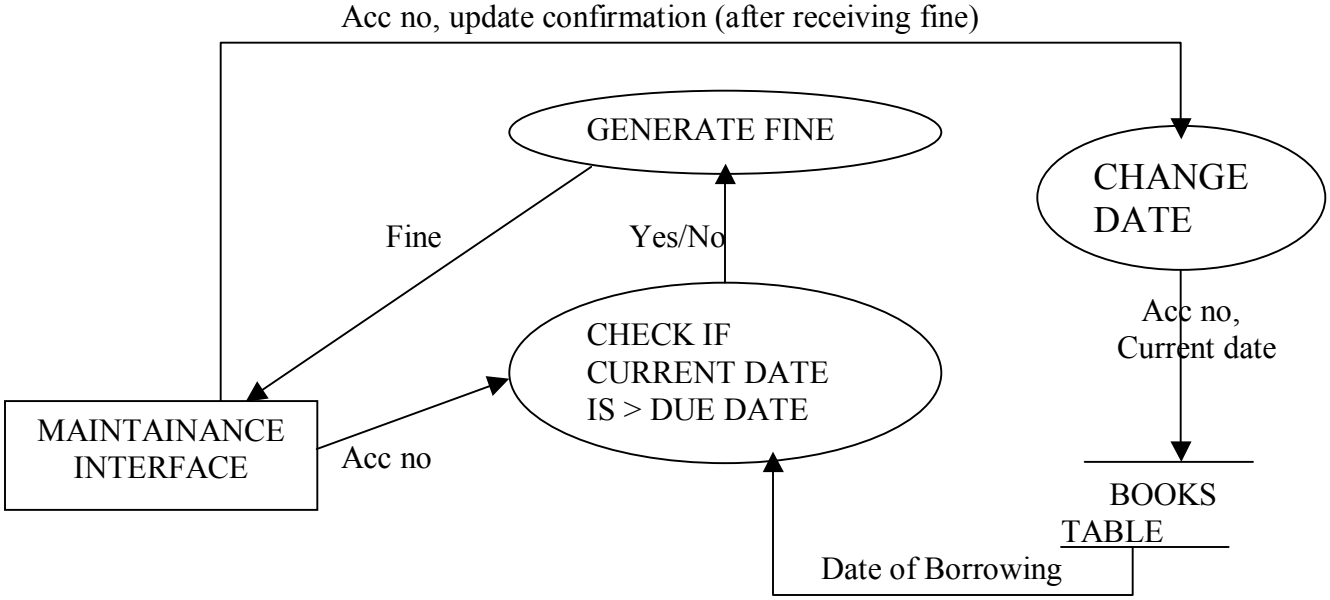
**QUERY UNIT**



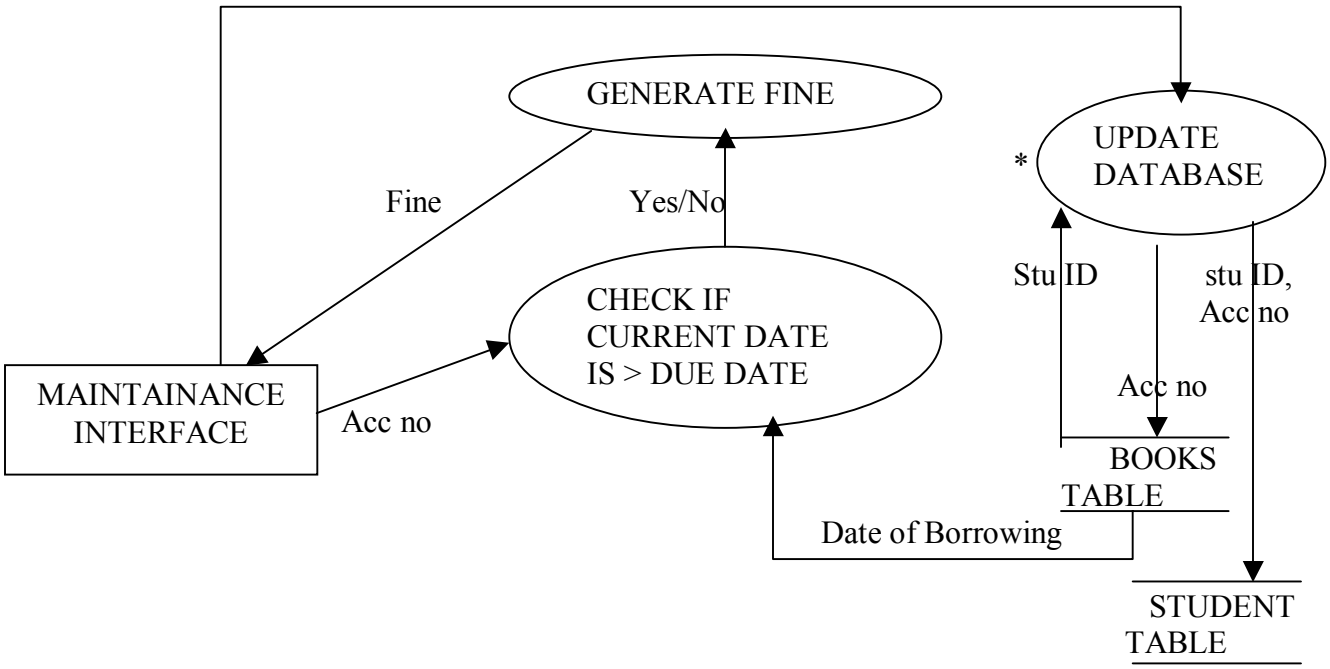

ISSUE UNIT



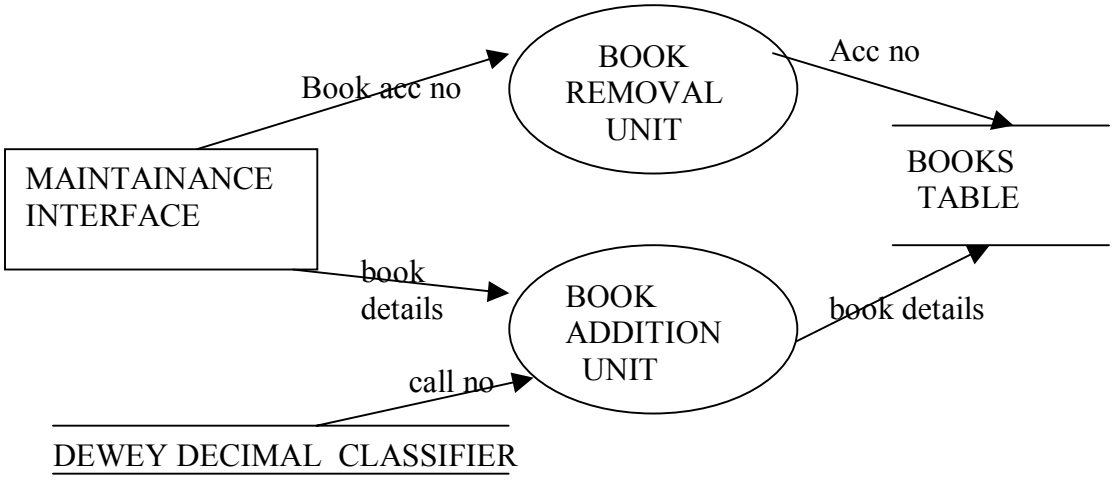
RENEWAL UNIT



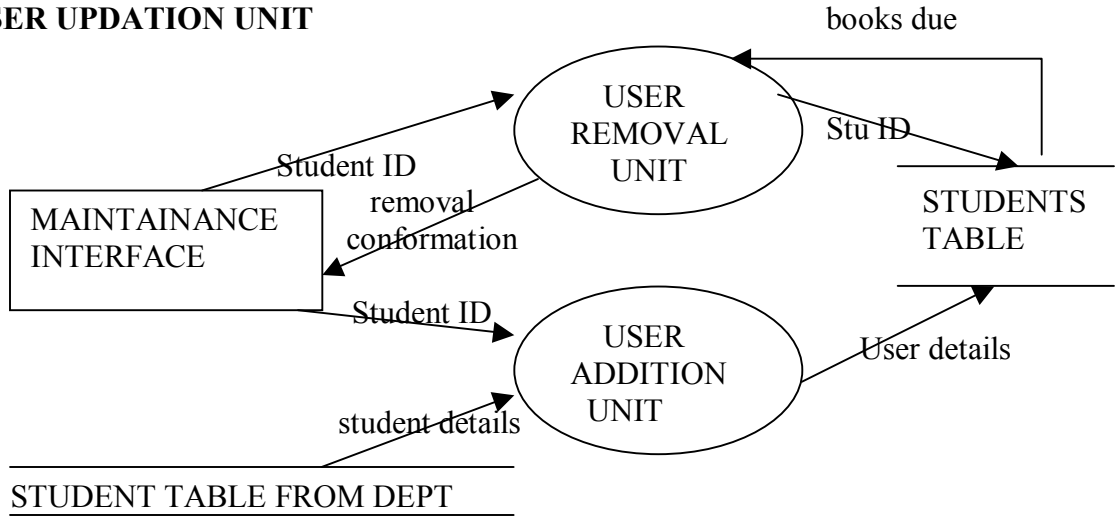

**RECEIPTS UNIT**



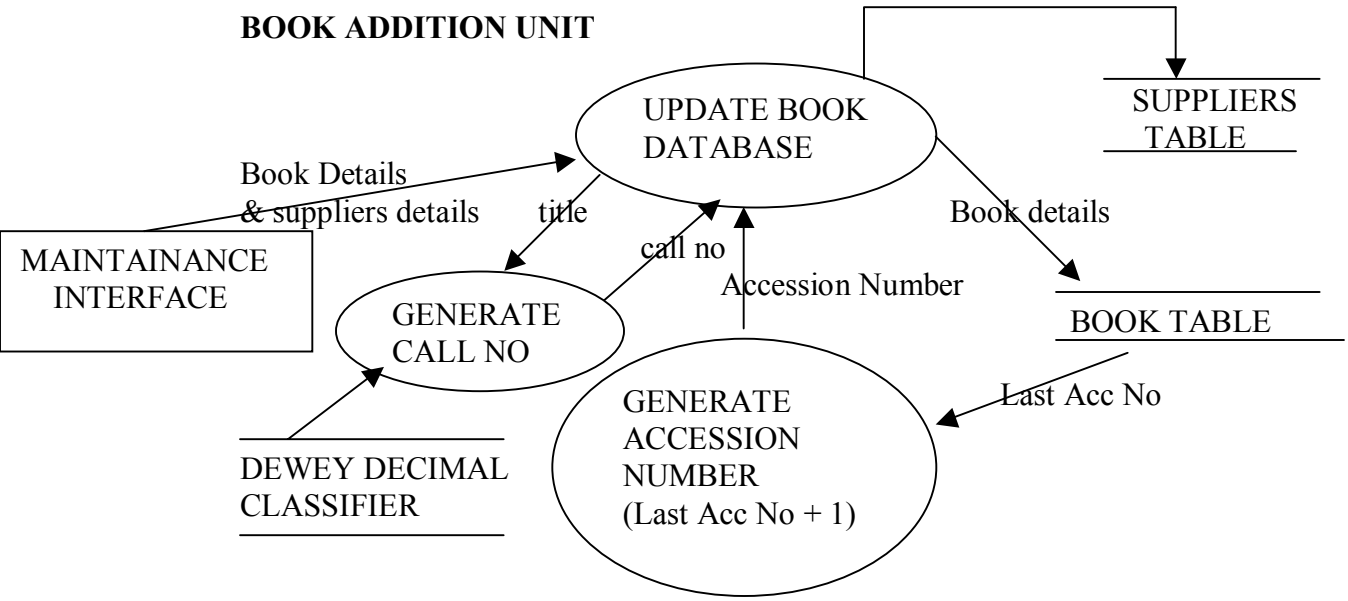
**BOOK MAINTENACE UNIT**



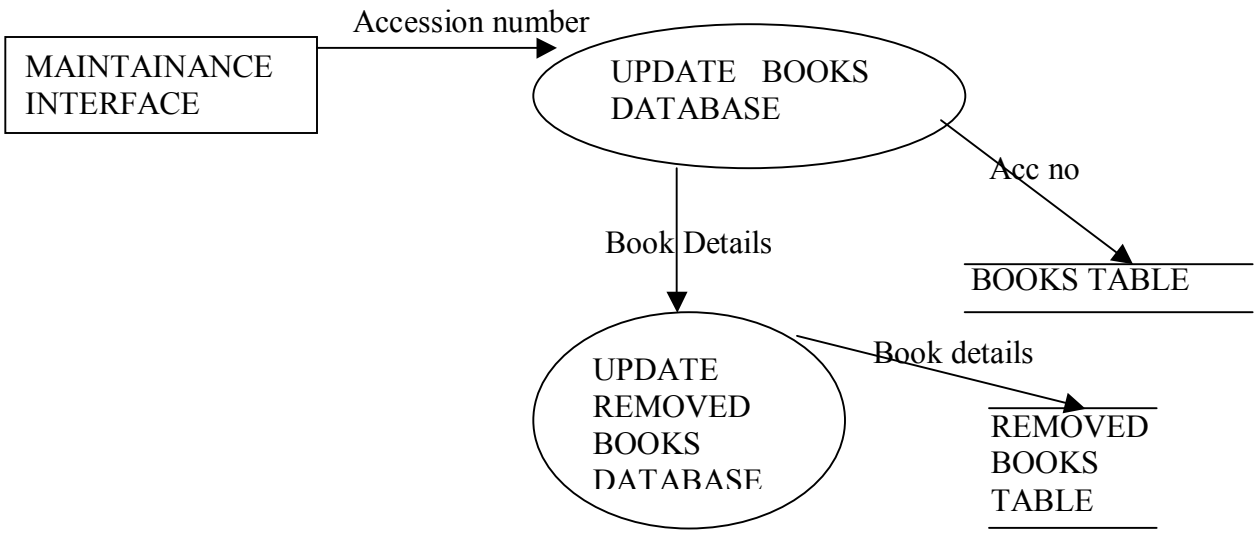

**USER UPDATION UNIT**



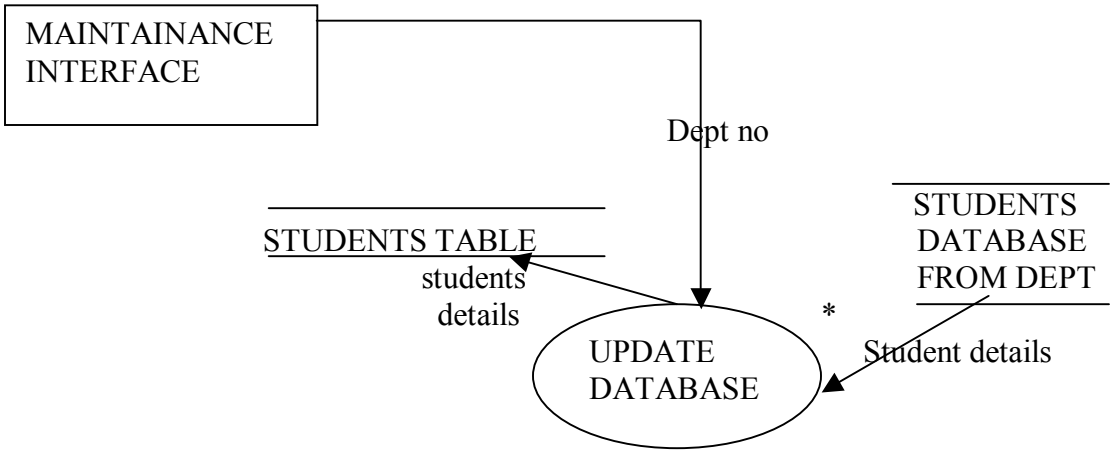

**BOOK ADDITION UNIT**



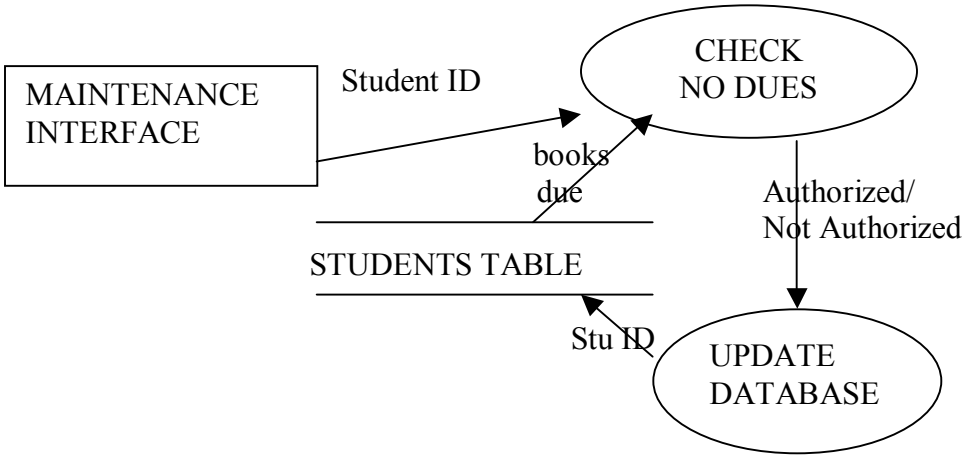
**BOOK REMOVAL UNIT**



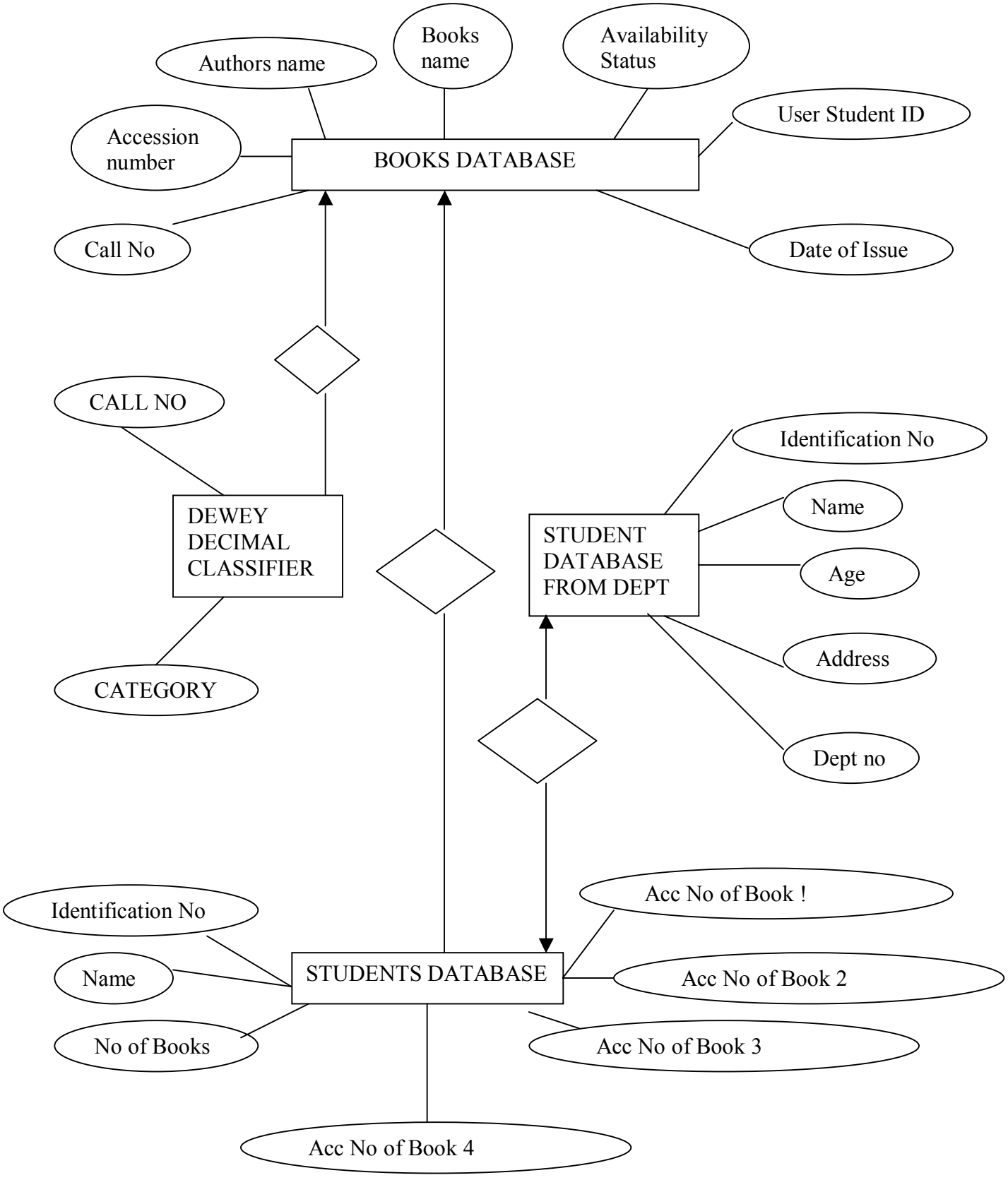

**USER ADDITION UNIT**




**USER REMOVAL UNIT**



E R DIAGRAMS


Authors name
Books name
Availability Status
User Student ID
Accession number
BOOKS DATABASE
Call No
Date of Issue
CALL NO
DEWEY DECIMAL CLASSIFIER
CATEGORY
Identification No
Name
STUDENT DATABASE FROM DEPT
Age
Address
Dept no
Identification No
Name
No of Books
STUDENTS DATABASE
Acc No of Book !
Acc No of Book 2
Acc No of Book 3
Acc No of Book 4

# SOFTWARE REQUIREMENT SPECIFICATION

## 1. INTRODUCTION:

**a) PURPOSE**: This document is meant for the developers to validate the final delivered system.

Any future changes will have to be approved by a change approval process.

**b) SCOPE**: This document is meant to describe the requirements for a library maintenance system.

It also describes the various interfaces for the system.

**c) DEFINITIONS, ACRONYMS AND ABBREVIATIONS:**

**i) *DEFINITIONS*:**

- *Accession Number*: It is the unique number that completely identifies any given book in the library. No two books can have the same accession number. It is given in serial order.
- *Call Number*: It is used to group books subject-wise for their easy retrieval. Books concerning the same topic have the same call number. The call number can be derived from the accession number.

**ii) *ACRONYMS AND ABBREVIATIONS*:**

- *Acc no* : Accession Number
- *Call no* : Call Number
- *Stu ID* : Student Identification Number

**iii) *REFERENCES*:**

Mrs. B. Mangala, OU Coll of Law.

**iv) *DEVELOPE R'S RESPONSIBILITY OVERVIEW*:**

- Developing the software
- Installing the software on the clients hardware
- Training the users
- Maintenance of the product till the end of two years after installation

## 2. GENERAL DESCRIPTION:

a) **PRODUCT FUNCTIONS OVERVIEW**:

The system performs four major operations.

- Locating books
- Receipt of books
- Issue of books
- Renewal of books
- Removing and adding users
- Adding new books and removing specified books

In each case it updates the relevant TABLEs.

b**) USER CHARACTERISTICS**:The user must be comfortable with the computer and must be conversant with the user interface software provided.

**c) GENERAL CONSTRAINTS:**

The operation of the product should not be limited by the hardware or operating system on the available computer.

## 3. FUNCTIONAL REQUIREMENTS

**GENERAL OVERVIEW OF INPUTS AND OUTPUTS**: The system is designed to perform six tasks (broadly). Each module has its own inputs and outputs. During every operation, one of the four modules must be chosen.

**a) INPUTS:**

- *MODULE 1*: QUERY UNIT
  Book author or title

- *MODULE 2*: ISSUE UNIT
  Stu ID
  Acc no of the book

- *MODULE 3*: RENEWAL OF BOOKS
  Acc no
  Update confirmation(after taking the fine)

- *MODULE 4*: RECEIPTS UNIT
  Acc no
  Update confirmation(after taking the fine)

- *MODULE 5*: BOOK MAINTAINANCE UNIT
  Book Name
  Author Name
  Operation Performed: Add book/ Remove book

- *MQDULE 6*: USER UPDATION UNIT
  Dept no (for getting a list of students for addition)
  Stu ID (for removal)

**b) OUTPUTS:**

- *MODULE 1*: QUERY UNIT
  List of books with acc number, call no, title and authors

- *MODULE 2*: ISSUE UNIT
  Outcome of the operation

- *MODULE 3*: RENEWAL OF BOOKS
  Fine

- *MODULE 4*: RECEIPTS UNIT
  Fine

- *MODULE 5*: BOOK MAINTENANCE UNIT
  None

- *MODULE 6*: USER UPDATION UNIT
  Outcome of operation (for removal)

**c) PROCESSING:**

The main function of the product is to see if the student is authorized and to update the TABLE in a suitable way depending on the operation requested.

- ***QUERY UNIT***

i) This unit takes the authors name or the title of the book as the input, queries the TABLE and displays a list of accession numbers followed by the authors and titles.

- ***ISSUE MODULE***

i) The system must check whether the user is authorized by comparing his identification.
ii) I t must check whether the book is available.
iii) It must determine if the student can take more books (depending on the number of outstanding books).
iv) It must modify the student TABLE and the books TABLE to reflect the changes.
v) It must signal successful operation in the case of issue and failure in the case of unavailability of the book and the student not being authenticated.

- ***RENEWAL MODULE***

i) The system is given the accession number of the book. It locates the due date of the book from the book TABLE.
ii) It computes fine if due date is lower than the present rate.
iii) It must update the book TABLE by changing the date of borrowing to the current date if the fine is paid.

- ***RECEIPT MODULE***

i) The system is given the accession number of the book. It locates the due date of the book from the book TABLE.
ii) It computes fine if due date is lower than the present rate.
iii) It must update the book TABLE and the student TABLE.

- (during the updation of student TABLE, it decreases the number of books borrowed by one and marks the corresponding book with an invalid accession number.
- While updating books TABLE, it marks the book available.

- ***BOOK MAINTAINANCE UNIT***

*BOOK ADDITION*

i) Generate an accession number by adding 1 to the last number in the TABLE.
ii) Generate the call number for the new book by referring to the Dewey Decimal Classifier.
iii) Add the new entry in the TABLE and enter all the details of the book

*REMOVING A BOOK*

i) add the details of the removed book to the removed books TABLE
ii) update books TABLE by removing the record corresponding to the book.

- ***USER UPDATION UNIT***

*USER ADDITION UNIT*

i) Get the department no of the dept from which students are to be added.
ii) Get the list of students from the concerned department from the department number.
iii) Update the students TABLE by adding records for every student.(set the no of books field to zero initially).

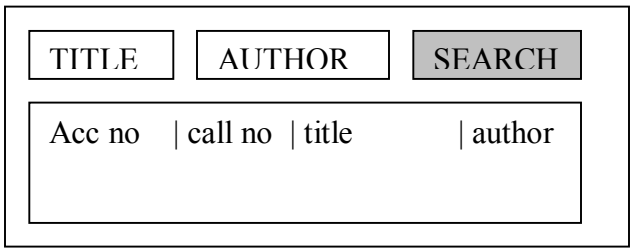
*USER REMOVAL UNIT*

i) get the stu ID of the user.
ii) Check if there are any borrowed books not returned by the user.(this can be checked by looking at the number of books field in the students TABLE).
iii) If there are no borrowed books unreturned remove the name of the user.

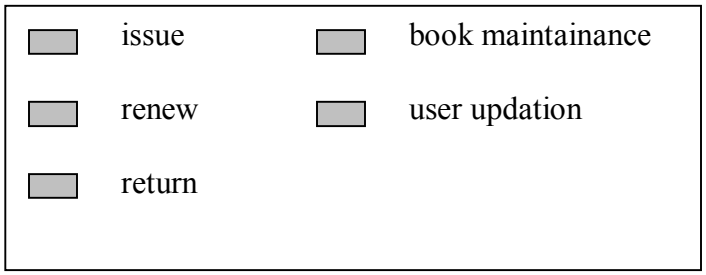
iv) Report the outcome of the operation ( whether it is successful or unsuccessful).

## 4. EXTERNAL INTERFACES

### QUERY INTERFACE




### MAINTENANCE INTERFACE




## 5. EXCEPTION HANDLING

- The program should handle situations where a wrong query is posed (with non existent author/title)
- The program should not loop infinitely when an erroneous acc no is given as input.
- The program should take care when a non existent dept no is presented at the user updation unit.

## 6. PERFORMANCE CONSTRAINTS

The system should have a maximum response time of 20 seconds for query and 30 seconds for all other operations. The bills for fine should be generated within 40 seconds.

## 7. DESIGN CONSTRAINTS

The software must be able to run on the specified hardware and operating system. Further it must be tested for all cases.

## 8. FORSEEABLE MODIFICATIONS

The system can be modified so that it can keep track of the purchase details and details about the orders placed of the books, magazines and journals present in the reference and issue sections.