RV COLLEGE OF ENGINEERING®
BENGALURU – 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

“Library Management System With Face Recognition And Email Notification”

PROJECT REPORT
DATABASE DESIGN LAB(18CS53)
V SEMESTER

2020-2021

Submitted by

Sumith Kumar S	1RV18CS177
Sreedhar S Dhulkhed	1RV18CS171

Under the Guidance of

Dr. Poonam G
Associate Professor
Department of CSE, RVCE,
Bengaluru - 560059

RV COLLEGE OF ENGINEERING®, BENGALURU - 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the **Mini-**Project work titled "**Library Management System With Face Recognition And Email Notification**" has been carried out by **Sumith Kumar S (1RV18CS177), Sreedhar S Dhulkhed (1RV18CS171),** bonafide students of RV College of Engineering, Bengaluru, have submitted in partial fulfillment for the **Assessment of Course: DATABASE DESIGN PROJECT (18CS53)** during the year 2020-2021. It is certified that all corrections/suggestions indicated for the internal assessment have been incorporated in the report.

Faculty Incharge
Department of CSE,
RVCE., Bengaluru –59

Head of Department
Department of CSE,
RVCE, Bengaluru–59

RV COLLEGE OF ENGINEERING, BENGALURU® - 560059
(Autonomous Institution Affiliated to VTU)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **Sumith Kumar S (1RV18CS177), Sreedhar S Dhulkhed(1RV18CS171)** the students of 5th Semester B.E., Department of Computer Science and Engineering, R.V. College of Engineering, Bengaluru hereby declare that the mini-project titled **"Library Management System With Face Recognition And Email Notification"** has been carried out by us and submitted in partial fulfillment for the **Assessment of Course: DATABASE DESIGN PROJECT (18CS53)** during the year 2020-2021.

Place: Bengaluru	**Sumith Kumar S**	**Signature:**
Date:	**Sreedhar S Dhulkhed**	**Signature:**

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project work. I would like to take this opportunity to thank them all.

I deeply express my sincere gratitude to my guide **Dr. Poonam G, Associate Professor,** Department of CSE, RVCE, Bengaluru, for his able guidance, regular source of encouragement and assistance throughout this project

I would like to thank **Dr.Ramakanth Kumar P**, Head of Department, Computer Science & Engineering, R.V.C.E, Bengaluru, for his valuable suggestions and expert advice.

First and foremost I would like to thank **Dr. Subramanya K N**, Principal, R.V.C.E, Bengaluru, for his moral support towards completing my project work.

I thank my Parents, and all the Faculty members of Department of Computer Science & Engineering for their constant support and encouragement.

Last, but not the least, I would like to thank my peers and friends who provided me with valuable suggestions to improve my project.

Abstract

The purpose of this project is to make an automated library management system and also to emulate most of the functionality a normal one. Existing library management systems do provide most of the required functionality of adding / removing / issuing books and adding users to the library. Apart from the functionality listed above ,the proposed system allows for seamless login with the help of face recognition and also notifies users to return books based on the return date by sending automated emails. Also, a person can review books and view books which have already been reviewed .This will help him make a sound decision as to which book to borrow later on.

The Django framework of python is used to create the data models and code the functionality. Sqlite3 is used as the relational database and MonogoDB as the nosql database. Majority of the data is stored in the sql database while the reviews are stored in the MongoDB database. HTTP POST/GET/PUT act as the apis and help to manipulate the data in the database. Most of the tasks are sequential in nature i.e only after the completion of one task does the other task start. Email notification runs as a background process and runs asynchronously which doesn’t hinder the normal flow of the program.

Key findings of this project is that the existing library management system can be upgraded to a better version as depicted here with minimal amount of hardware and software requirements. If inculcated into the existing system then it can ease the need for id cards which are traditionally used for issuing a particular book. Also it will keep the borrowers on their toes as they will be constantly notified of as when they have to return their book back.

Table of Contents

List of Figures

GLOSSARY

DFD	:	Data Flow Diagram
ER	:	Entity-Relationship
SRS	:	Software Requirement Specification
SQL	:	Structured Query Language

Chapter 1

Introduction

Libraries are essential parts of institutions. They are the driving force of research and one of the most critical ways to obtain knowledge. Libraries have gone from being ledger based issuing and return to digitized platforms. But fully autonomous library systems are yet to become commonplace. With this project we would like to endeavour to make an effort towards that direction i.e of automation.

1.1 Objective

The main aim of the Library Management Software is to handle the entire activity of a library. The software keeps track of all the information about the books in the library, their cost,their complete details and total number of books available in the Library. The user will find it easy in this automated system rather than using the manual writing system. The system contains a database where all the information will be stored safely. The system is user-friendly and error free. The “Library Management Software" has been developed to override the problems prevailing in the practicing manual system.

1.2 Scope

Goal is to build an automated library management system. Objectives of the library management system are:

1. A system for library staff to be able to add and remove books, issue books and impose fines to readers.

2. Readers are divided into faculty and students and on the basis of the category a reader belongs to he/she may attain higher privileges like no fines and extended period of issue of book.

3. A face recognition system which will be able to recognise a particular reader and help him/her in login. In this way there will be no need for an ID card.

The benefits of the library system are :

1. It will be very easy to understand the functionalities and operate upon them.

2. Consistency will be guaranteed for most of the operations apart from book review.

3. The facial recognition system will mitigate the need for ID cards.

1.3 Proposed System

Project has a facility where students after logging in to their accounts can see a list of books issued and its issue date and return date and also the students can request the librarian to add new books by filling the book request form.

Extra security is provided by adding a face recognition system to login to the student/admin account. So even if the account is hacked still the student/admin won't lose their account because login is possible only when username,password and face all three of them match. Project has a facility to send notifications to the student in case the student has crossed the deadline to return the book to the library.

Existing system does not have any facility for book requests and suggestions where as in the proposed system after logging in to their accounts students can request books as well as provide suggestions to improve the library. Much of human intervention can be eliminated.

1.4 Societal Concern and Innovative Component

Libraries are equivalent to banks in terms of the number of transactions (borrow and return) which take place everyday. With a computer interface it will be much easier to keep track of the happenings in the library and will also require less manpower for maintenance. To further speed up and provide more security to the process of people borrowing and returning books , a computer vision system can be installed which'll identify a person and the category he/she belongs to staff, student or administrator . Later the person can enter the id of the book and be able to borrow/return a book. Based on the date of borrow/return appropriate fines can also be levied. With respect to the recommender system it works solely on nosql queries but that can be taken to another level by introducing a model which can use previous statistics to identify trends in the borrowing/returning capacities and make a decision.

Chapter 2

Requirement Specification

A software requirements specification (SRS) is a detailed description of a software system to be developed with its functional and non-functional requirements. The SRS is developed based on the agreement between customers and contractors. It may include the use cases of how the user is going to interact with the software system. The software requirement specification document is consistent with all necessary requirements required for project development. To develop the software system we should have a clear understanding of Software system. To achieve this we need continuous communication with potential customers to gather all requirements.

2.1 Hardware Requirements

Processor	: Octa-core Exynos 7884 processor
Storage	: 32GB
RAM	: 2GB or more
OS	: Android 6.1 or higher
Camera	: 2 MP Web Camera
Display	: 6.40-inch (720x1560) or more

2.2 Software Requirements

1. SQL database - SQLite
2. NoSQL - MongoDB
3. Backend: Django
4. Frontend: HTML, CSS
5. Web Browser: Microsoft Internet Explorer, Mozilla, Google Chrome.
6. Operating System: Windows 10

2.3 Functional Requirements

A functional requirement defines the functionality of a system or one of its subsystems. It also depends upon the type of software, expected users and the type of system where the software is used. The functional requirements of the proposed work are as follows:

1.Reader Module

1.1 User Login
Readers should be able to login using face recognition. System must only allow users with valid id and password to enter the system else an error must be displayed. The user must be able to logout after they finish using the system.

1.2 Register New User
Any reader should be able to register and create an account in the database. If improper details are entered then error must be shown.

1.3 View Issued Book

Readers should be able to borrow books and see which all books are issued to him/her and should also be able to see the deadline to return the books.

1.4 Extra Privileges for Teachers

Teachers have same facilities as other readers with some extra privileges which are as follows:
1. Keep books for a longer period of time as compared to students.
2. They are exempted from fines.

2. Staff Module

2.1 Staff Login

Staff will have a separate login section with additional privileges through face detection. If the credentials are incorrect or the student tries to login then an error must be shown.

2.2 Add New Book

1. Staff should be able to add ,view and issue books
2. Staff should be able to impose a fine on students who return books after the due date.
3. Staff can view which book is issued to which student and can also view all the students.

3. Email Notification Module

The system should be able to send an email to the readers to return the books back to the library when the deadline is approaching.

4.Review System Module

Readers should be able to submit reviews about the books which will be helpful for the library Staff to have more such books in the library.

Chapter 3

ER Diagram

An entity–relationship model describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities. An ER model is commonly formed to represent things a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model that defines a data or information structure which can be implemented in a database, typically a relational database. An E-R model is usually the result of systematic analysis to define and describe what is important to processes in an area of a business. It does not define the business processes; it only presents a business data schema in graphical form. It is usually drawn in a graphical form as boxes (*entities*) that are connected by lines (*relationships*) which express the associations and dependencies between entities.

The ER-diagram below shows four strong entities which are Customer, Store, Product and Bill. It also consists of a weak entity which is represented by a double lined box called Cart. The ER diagram represents the relations between the different entities and also the attributes of these entities. Let us consider one of the relationships, the customer relation is connected to the bill entity by a 1:N relationship as one customer can have multiple bills. By using the ER diagram the database schema is proposed which is later normalized to produce an efficient database architecture.

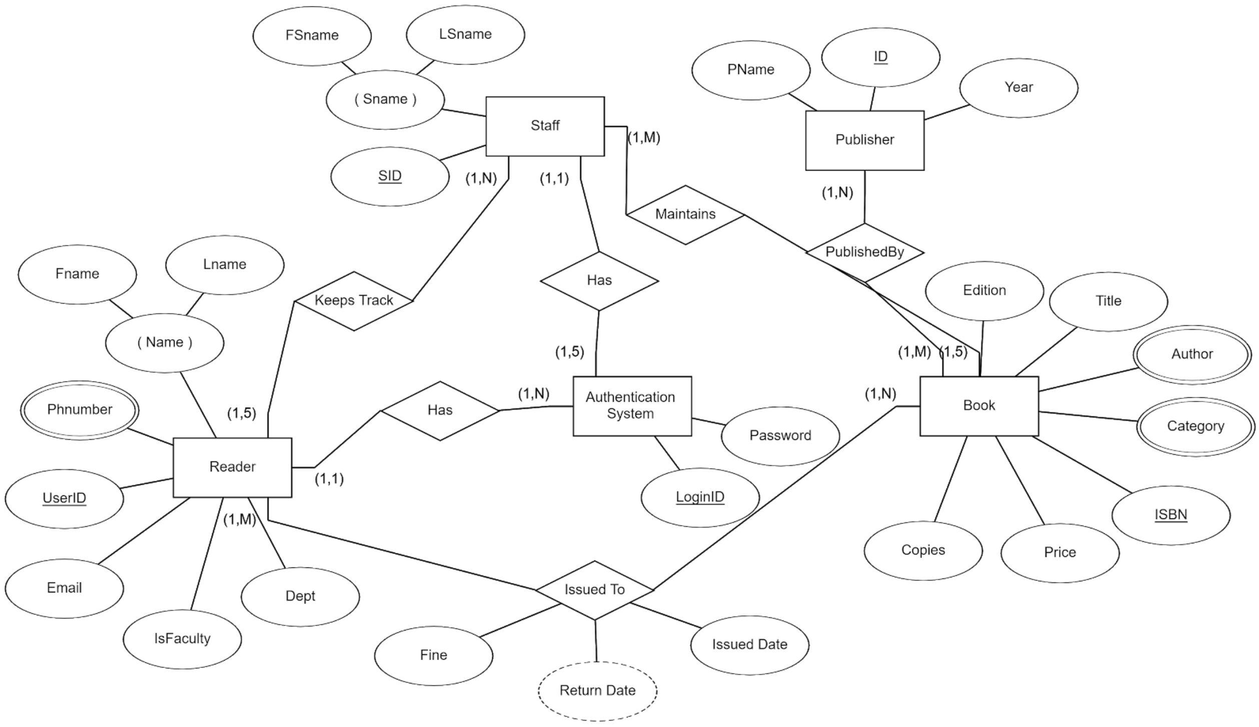


Fig 3.1: ER Diagram

Figure 1.1 shows the entity-relationship diagram for our project. The entities (described in rectangles) are the Reader, Staff, Book, Publisher and Authentication System. All these entities have attributes (described in ovals), with one attribute being the key attribute. The relationships between these entities are described using diamonds, and the multiplicity of these relationships is also shown.

The attributes for all the entity types are as follows:

- Reader: fname, lname, phnumber, userid, email, isfaculty, dept
- Staff : fsname, lsname, sid
- Publisher: pname, pid, year
- Book: edition, title, author, category, isbn, price, copies

Chapter 4

Detailed Design

A data-flow diagram (DFD) is a way of representing a flow of data of a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops. Specific operations based on the data can be represented by a flowchart.

DFD consists of processes, flows, warehouses, and terminators. There are several ways to view these DFD components.

Process

The process is part of a system that transforms inputs to outputs. The symbol of a process is a circle, an oval, a rectangle or a rectangle with rounded corners. The process is named in one word, a short sentence, or a phrase that is clearly to express its essence.

Data Flow

Data flow shows the transfer of information from one part of the system to another. The symbol of the flow is the arrow. The flow should have a name that determines what information is being moved. Exceptions are flows where it is clear what information is transferred through the entities that are linked to these flows. Material shifts are modeled in systems that are not merely informative. Flow should only transmit one type of information. The arrow shows the flow direction. Flows link processes, warehouses and terminators.

Warehouse

The warehouse is used to store data for later use. The symbol of the store is two horizontal lines, the other way of view is shown in the DFD Notation. The warehouse does not have to be just a data file, for example, a folder with documents, a filing cabinet, and optical discs. Therefore, viewing the warehouse in DFD is independent of implementation. The flow from the warehouse usually represents the reading of the data stored in the warehouse, and the flow to the warehouse usually expresses data entry or updating . Warehouse is represented by two parallel lines between which the memory name is located.

Terminator

The Terminator is an external entity that communicates with the system and stands outside of the system. It can be, for example, various organizations, groups of people , authorities or a department of the same organization, which does not belong to the model system. The terminator may be another system with which the modeled system communicates.

4.1 DFD level 0 of Library Management System (Description and Diagram)

The figure below shows a Data Flow Diagram that is drawn for the Library Management System. It contains a process (shape) that represents the system to model. Here Readers and Staff are the entities that will interact with the system. In between the process and the external entities, there is data flow (connectors) that indicate the existence of information exchange between the entities and the system.

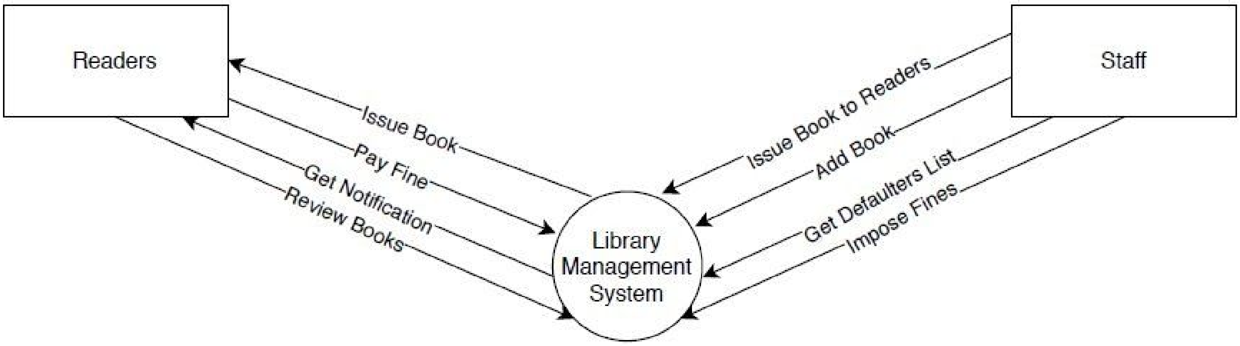


Fig 4.1: DFD Level 0

The Readers can Borrow books from the library, return the books after using them. Readers need to pay the fine if they do not return the books on time. Readers get email notifications reminding them to return the books and they can also provide the review of the books so that it will be useful to others.

Staff can issue the books to the users, they can add books into the library. Staff can also see which book is issued to whom and they can impose fine on those readers who do not return the books on time.

4.2 DFD level 1 of Library Management System (Description and Diagram)

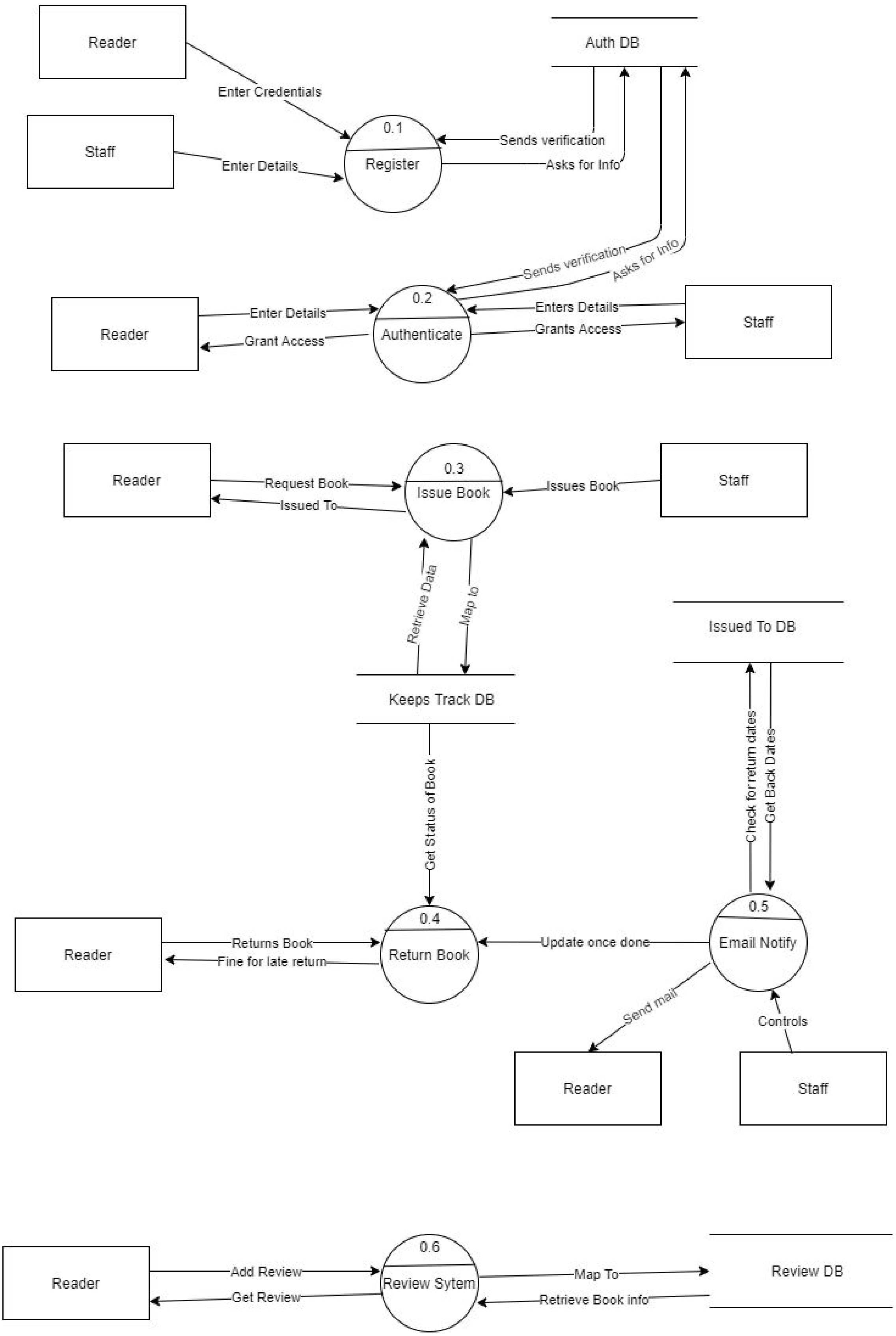


Fig 4.2: DFD Level 1

Figure 4.1.2 shows the level 1 DFD, which is the decomposition (i.e. break down) of our project Library Management System

Readers and Staff can register by providing valid credentials and it will be stored in the database.

Readers and Staff need to provide valid credentials in order to login and access the system.

Staff can issue books to the reader which will be stored in the database and this information will be used to send email notification to the readers to remind them to return the books on time and this information will also be used when the reader returns the book to impose fine if he is late.

The Readers can provide reviews about the books which are also stored in the database. This information will be helpful for other readers.

Chapter 5

Relational Schema and Normalization

From the ER diagrams to the data flow diagrams we have come to the point such that we can decompose the ER diagrams to tables. There is no single database structure that is always more efficient than any other. This is because the type and amount of data stored changes the optimal structure of the database. Using a relational schema, we can create a database that is optimized for data volume. This is ideal in situations where the cost of storing, processing, and memory is prohibitively expensive. The schema provides us foreign keys and primary keys and tells how one table depends on the other. While designing Relational Models, some conditions need to be defined which must hold for data present in the database. These conditions are called Constraints. These constraints are checked before performing any operation (insertion, deletion and updation) in the database. If there is a violation in any of constraints, operation will fail.

One of the main points to note in a relational schema, in contrast to an ER schema, is that relationship types are not represented explicitly; instead, they are represented by having two attributes A and B, one a primary key and a foreign key (over the same domain) included in two relations S and T. Two tuples in S and T are related when they have the same value for A and B. By using the EQUI JOIN operation (or NATURAL JOIN if the two join attributes have the same name) over S.A and T.B, we can combine all pairs of related tuples from S and T and materialize the relationship. When a binary 1:1 or 1:N relationship type is involved, a single join operation is usually needed. For a binary M:N relationship type, two join operations are needed, whereas for n-ary relationship types, n joins are needed to fully materialize the relationship instances.

In a relational schema we create a separate relation for each multivalued attribute. For a particular entity with a set of values for the multivalued attribute, the key attribute value of the entity is repeated once for each value of the multivalued attribute in a separate tuple because the basic relational model does not allow multiple values for an attribute in a single tuple.In the following Schema Diagram the tables are represented by rectangular boxes where each portion of the box represents an attribute of that entity. The relationship between the entities is represented by using directed arrows where the direction of the arrow represents the direction of the referenced key. The specifications between the relationships ensures the tables can be joined to produce meaningful information when required. The use of Foreign key constraints ensures that different entities do not contain different values of the same attribute leading to false results.

5.1 Relational Schema

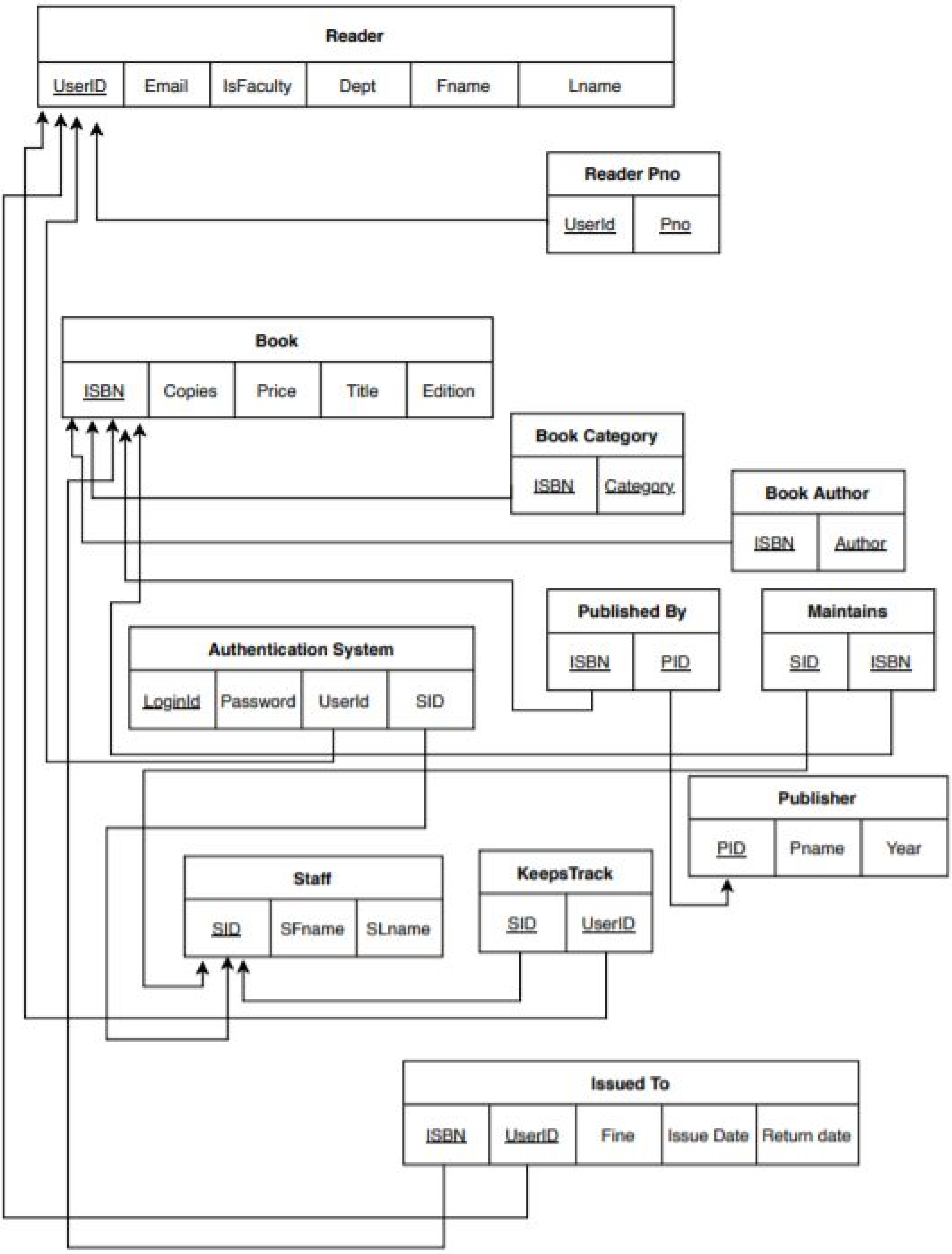


Fig 5.1 : Relational Schema

5.2 Normalization

Here we are going to use normalization to accomplish the following i.e. :

1. minimize the redundancy from a relation or set of relations
2. eliminate the undesirable characteristics like Insertion, Update and Deletion Anomalies

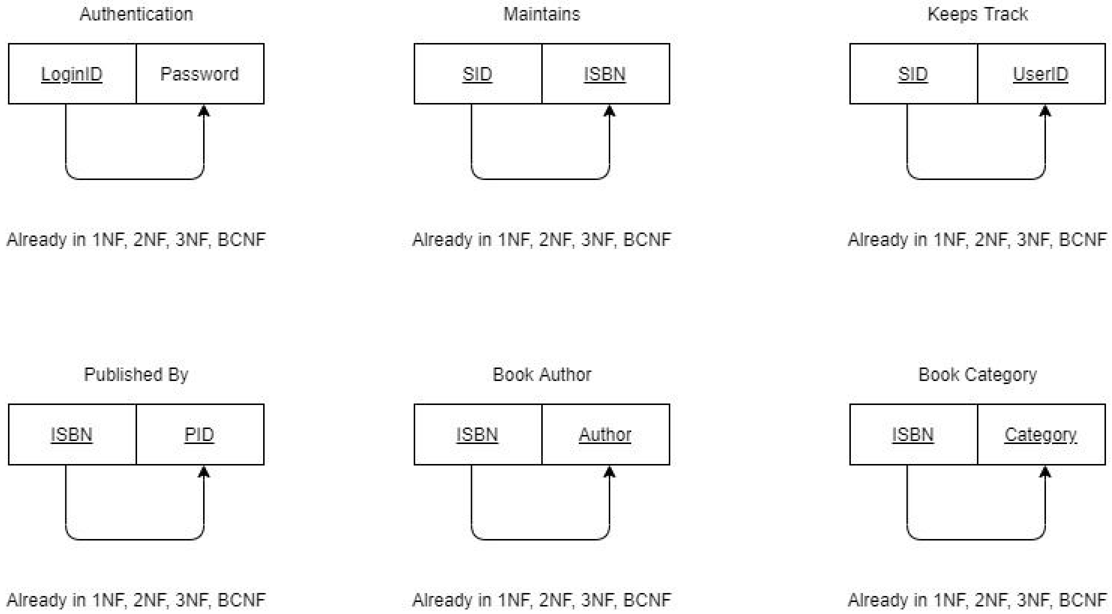

Authentication
LoginID
Password
Already in 1NF, 2NF, 3NF, BCNF
Maintains
SID
ISBN
Already in 1NF, 2NF, 3NF, BCNF
Keeps Track
SID
UserID
Already in 1NF, 2NF, 3NF, BCNF
Published By
ISBN
PID
Already in 1NF, 2NF, 3NF, BCNF
Book Author
ISBN
Author
Already in 1NF, 2NF, 3NF, BCNF
Book Category
ISBN
Category
Already in 1NF, 2NF, 3NF, BCNF

Fig 5.2 : Normalized Tables Authentication , Maintains, Keeps Track , Published By, Book Author, Book Category

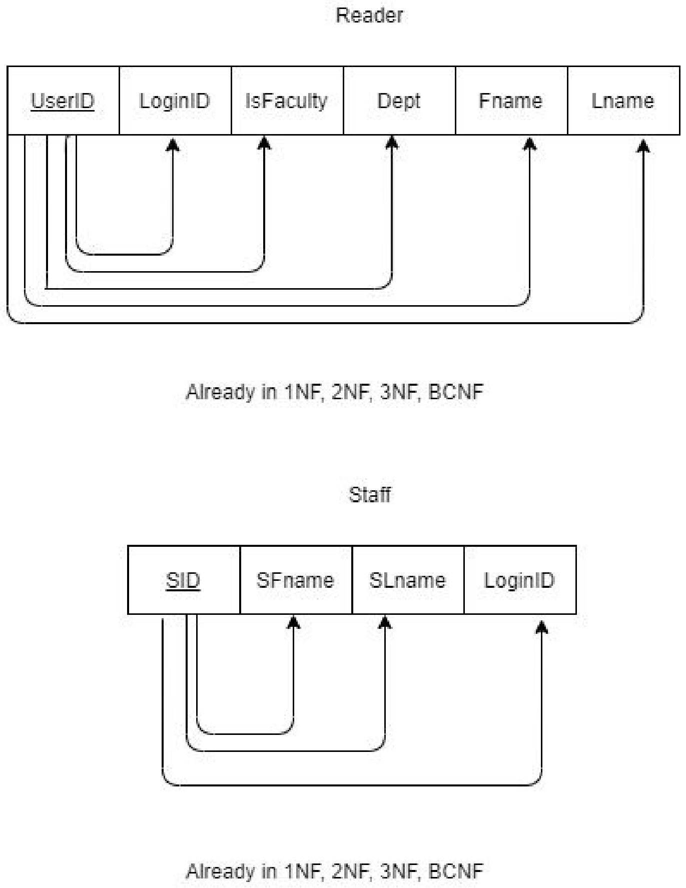


Fig 5.3 : Normalized Tables Reader, Staff

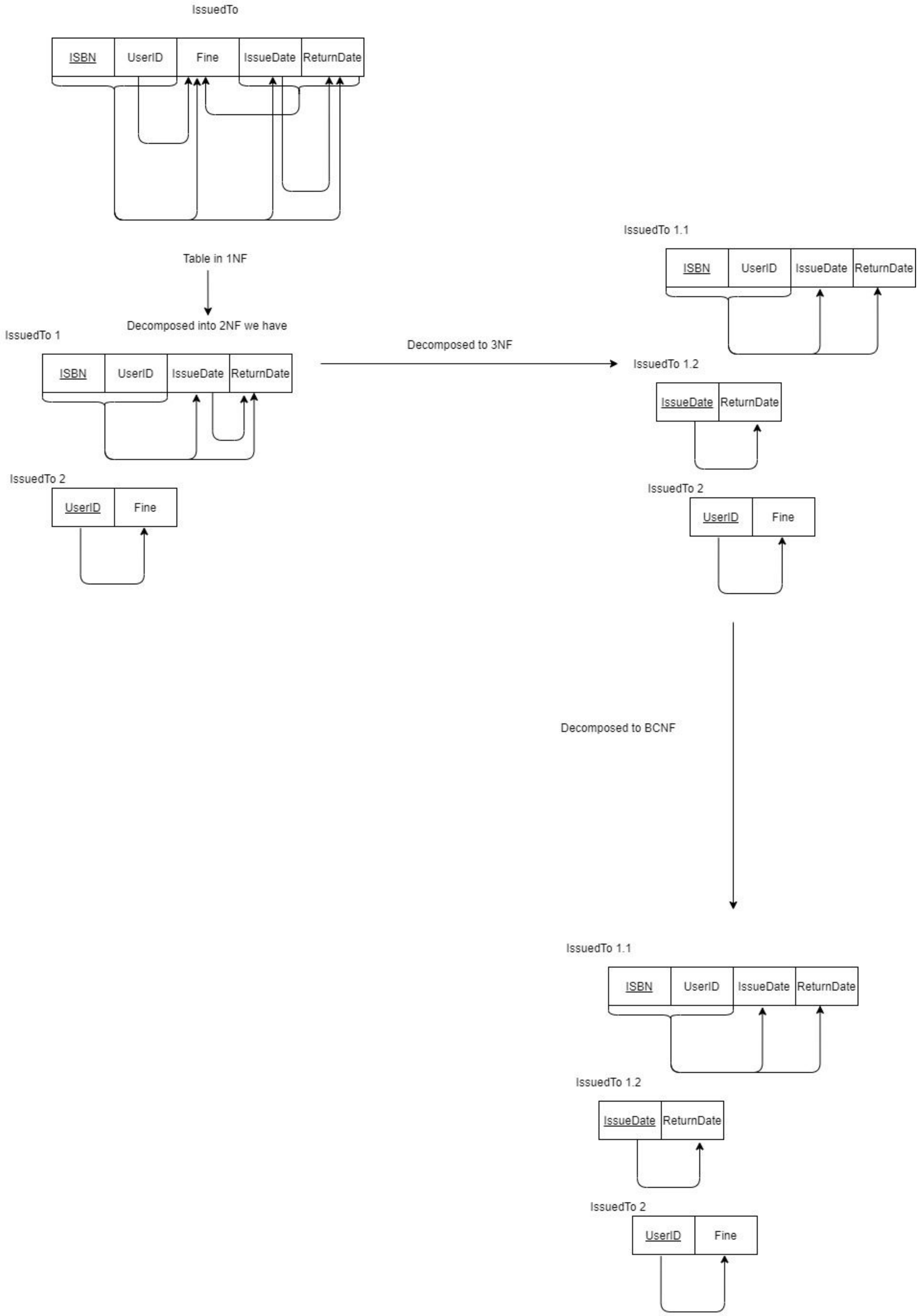


Fig 5.4 : Normalized Table IssuedTo

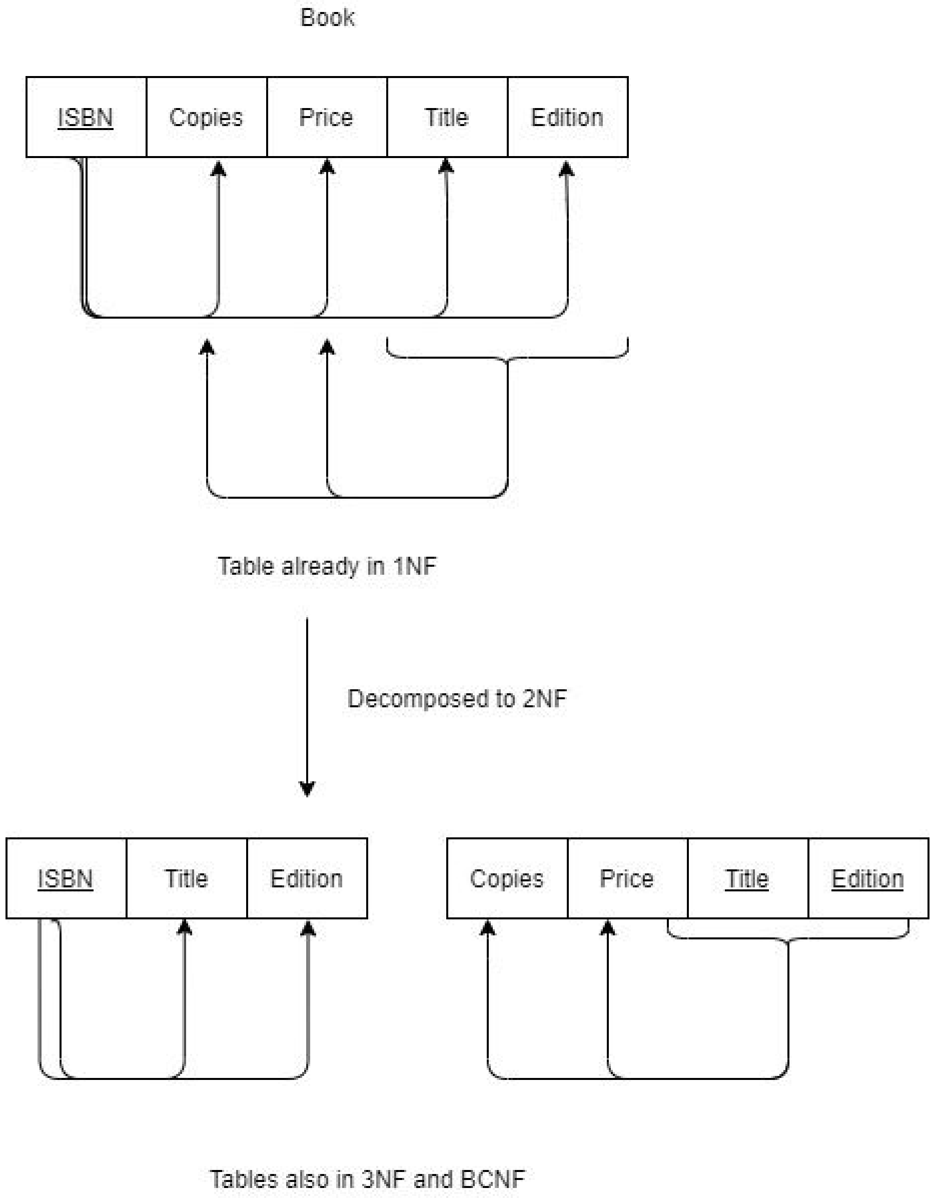


Fig 5.5 : Normalized Table Book

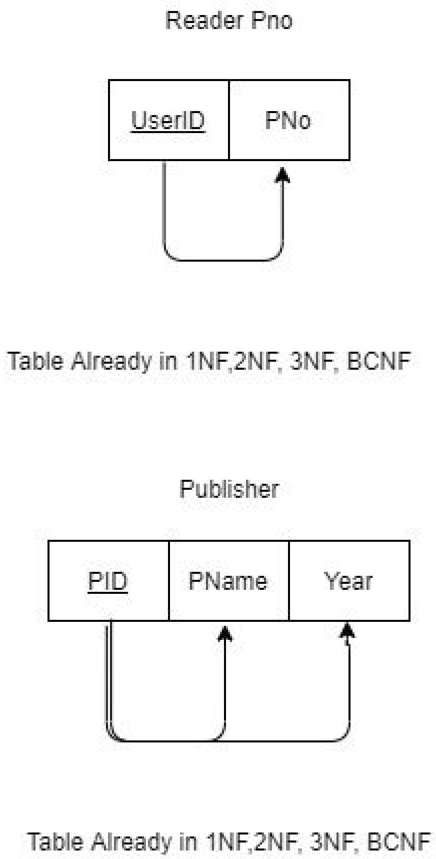


Table Already in 1NF,2NF, 3NF, BCNF

Table Already in 1NF,2NF, 3NF, BCNF

Fig 5.6 : Normalized Tables Reader_Pno , Publisher

Chapter 6

NoSQL Component

A NoSQL originally referring to non SQL or non relational is a database that provides a mechanism for storage and retrieval of data. This data is modeled in means other than the tabular relations used in relational databases. A NoSQL database includes simplicity of design, simpler horizontal scaling to clusters of machines and finer control over availability. The data structures used by NoSQL databases are different from those used by default in relational databases which makes some operations faster in NoSQL. The suitability of a given NoSQL database depends on the problem it should solve.

6.1. Integrating NOSQL database to SQL with complete system architecture

The SQL Database stores the reader information, library staff information and the library books information. The NoSQL database stores the book reviews and feedback. The application first starts with the Face Recognition authentication and later enters the home screen. The SQL database is used to extract reader and book information which is then displayed on the screen. For most of the library transactions the application uses the SQL database. The NoSQL database is used to manage the book reviews and feedback of the books in the library. Any review about a book that is given by the reader is updated in the NoSQL database. The two database architectures are integrated to provide a seamless experience to the application users.

Chapter 7

Conclusion and Future Enhancement

Library Management System allows the user to store the book details and the customer details. This software package allows storing the details of all the data related to the library. The system is strong enough to withstand regressive yearly operations under conditions where the database is maintained and cleared over a certain time of span. The implementation of the system in the organization will considerably reduce data entry, time and also provide readily calculated reports.

Some of the future enhancements which can be done are to inculcate a speech assistant which can help the person navigate through the digital library to find a particular book. A book recommender system can be inculcated in which the person can be advised to take a particular book based on his/her previous book issues. This can make the library management system a completely autonomous system on its own.

References

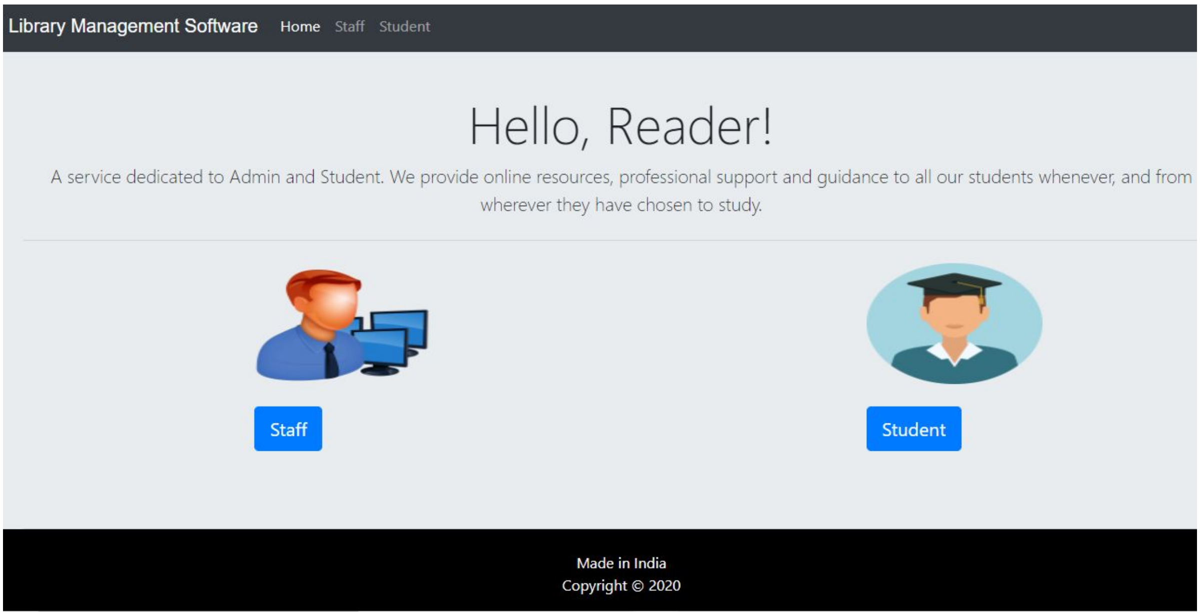
1. C. Győrödi, R. Győrödi, G. Pecherle and A. Olah, "A comparative study: MongoDB vs. MySQL," *2015 13th International Conference on Engineering of Modern Electric Systems (EMES)*, Oradea, 2015, pp. 1-6, doi: 10.1109/EMES.2015.7158433.
2. Shuo Xu, Liyuan Hao, Guancan Yang, Kun Lu, Xin An, A topic models based framework for detecting and forecasting emerging technologies, Technological Forecasting and Social Change, 10.1016/j.techfore.2020.120366, 162, (120366), (2021).
3. X. Liu, X. Fu and G. Sun, "Recovery of Deleted Record for SQLite3 Database," 2016 8th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, 2016, pp. 183-187, doi: 10.1109/IHMSC.2016.173.
4. NoSQL-Distilled_Book https://bigdata-ir.com/wp-content/uploads/2017/04/NoSQL-Distilled.pdf
5. https://www.w3schools.com
6. https://www.visual-paradigm.com/guide/data-flow-diagram/what-is-data-flow-diagram/#:~:text=Also%20known%20as%20DFD%2C%20Data,divided%20into%20logical%20and%20physical.
7. https://beginnersbook.com/2015/04/e-r-model-in-dbms/
8. https://www.import.io/post/what-is-data-normalization-and-why-is-it-important/#:~:text=The%20Importance%20of%20Data%20Normalization&text=Data%20normalization%20gets%20rid%20of,information%2C%20or%20updating%20existing%20information

Appendix

Source Code

Available on our github repository: https://github.com/Sreedhar-S-D/Library/tree/return-book

Snapshots:

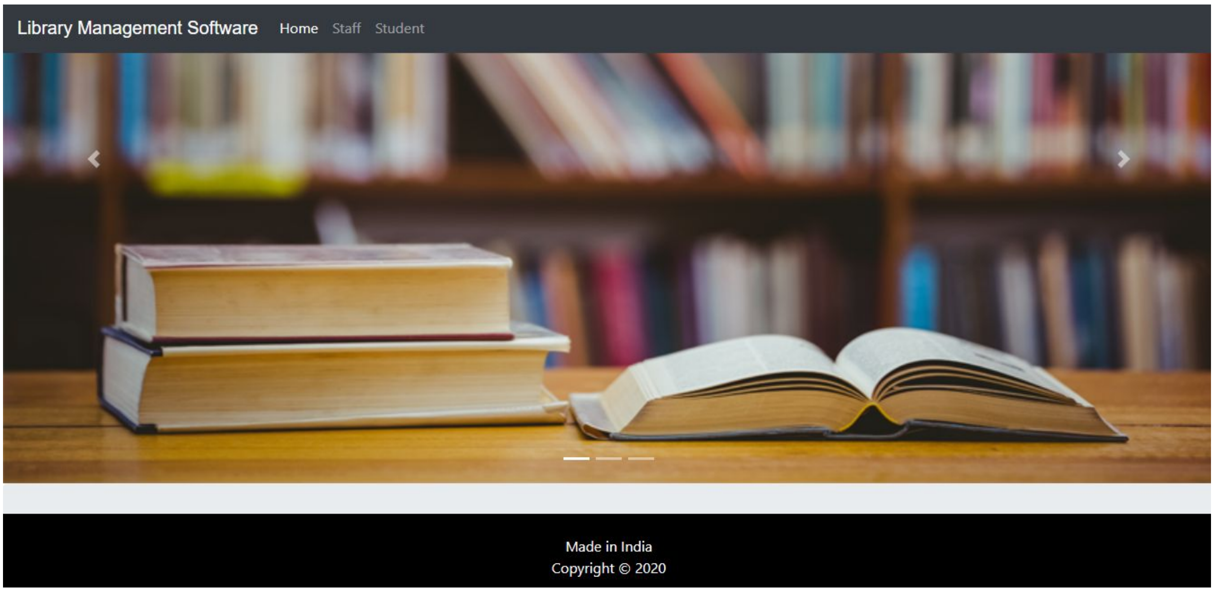


Hello, Reader!

A service dedicated to Admin and Student. We provide online resources, professional support and guidance to all our students whenever, and from wherever they have chosen to study.

Made in India
Copyright © 2020

Fig 6.1 :Landing page with student and staff login



Made in India
Copyright © 2020

Fig 6.2: Landing Page with slideshow of pictures of books

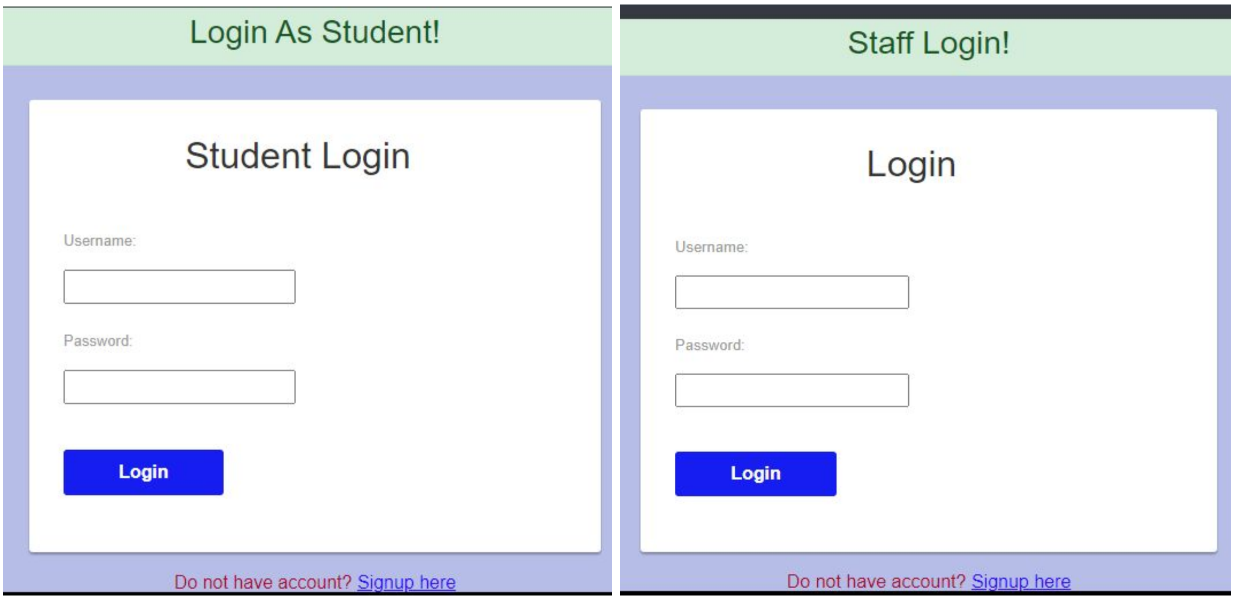


Fig 6.3 : Student and Staff Login Pages

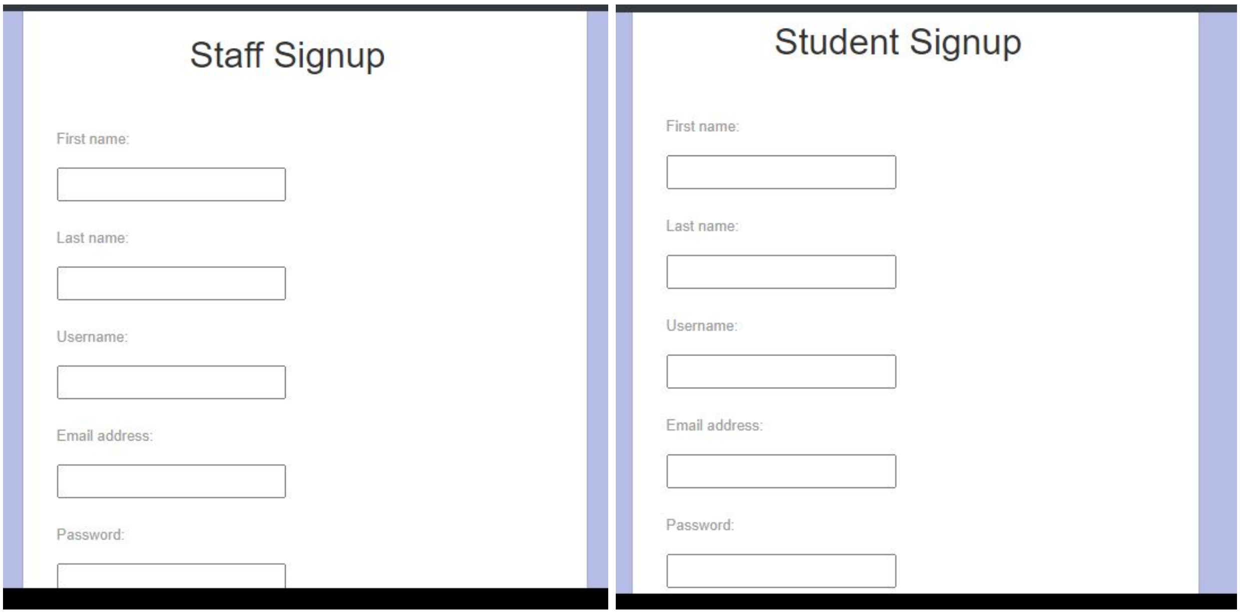
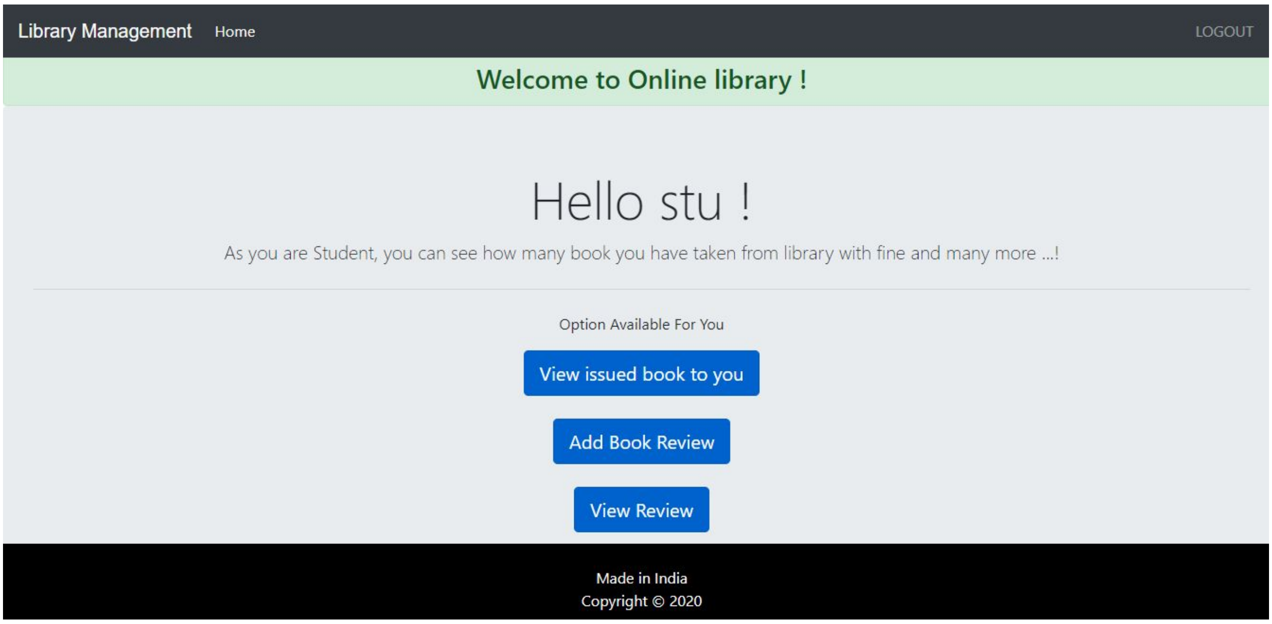


Fig 6.4 : Student and Staff Signup Pages



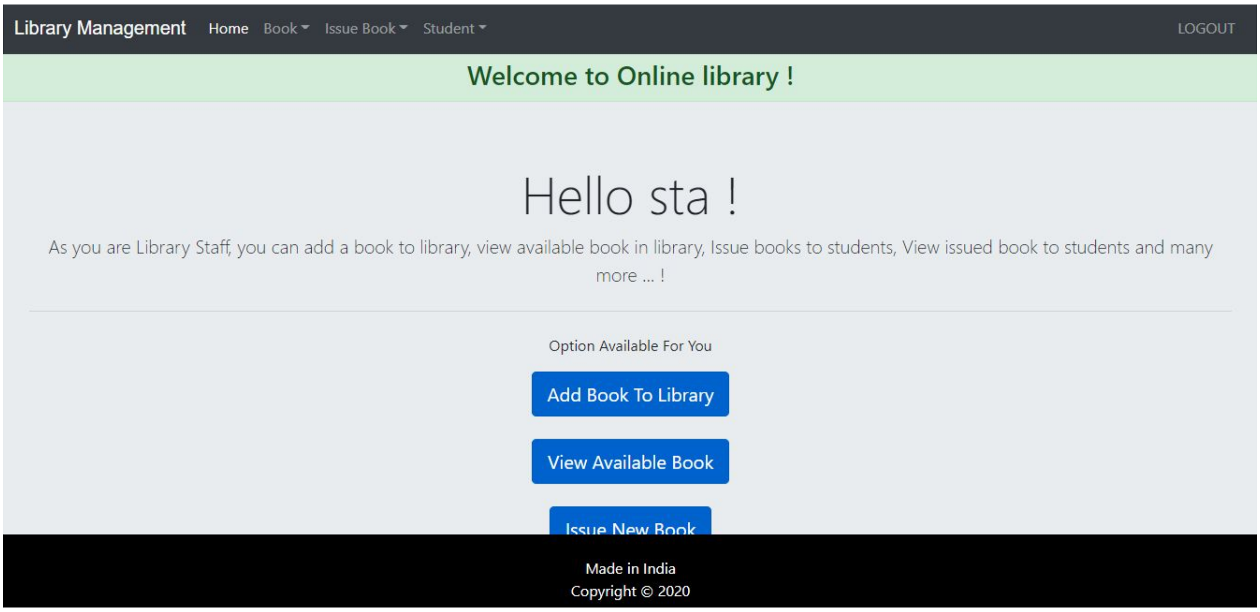
Welcome to Online library !

Hello stu !

As you are Student, you can see how many book you have taken from library with fine and many more ...!

Option Available For You

Fig 6.5 : Landing Page of Student



Welcome to Online library !

Hello sta !

As you are Library Staff, you can add a book to library, view available book in library, Issue books to students, View issued book to students and many more ... !

Option Available For You

Fig 6.6 : Landing Page for Staff

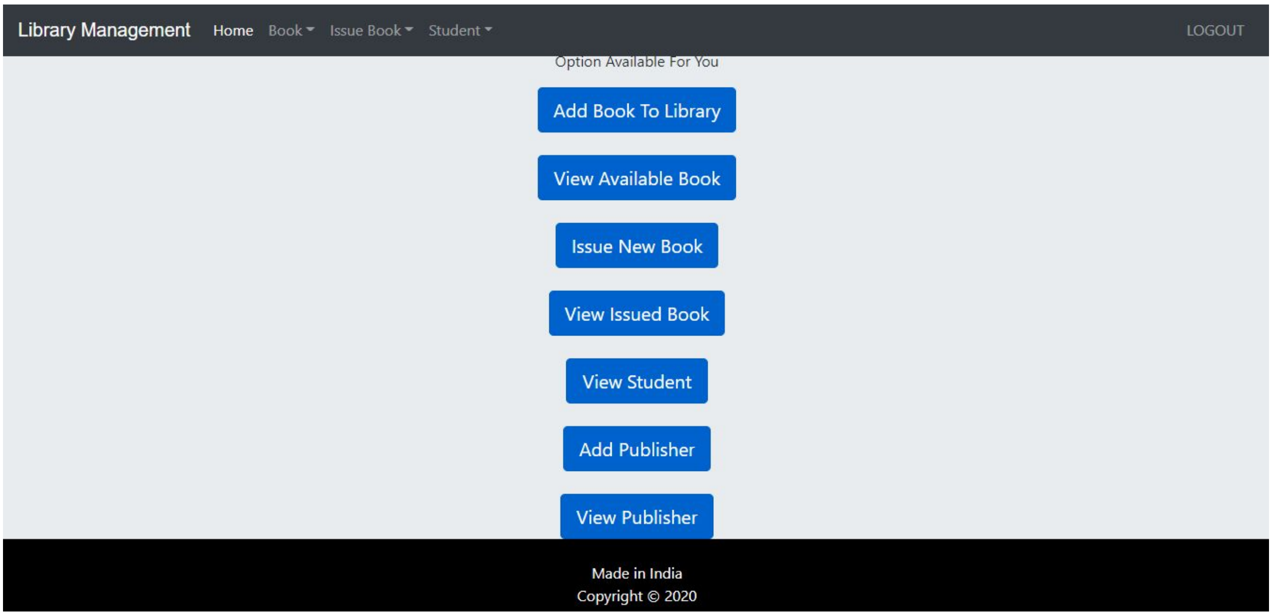
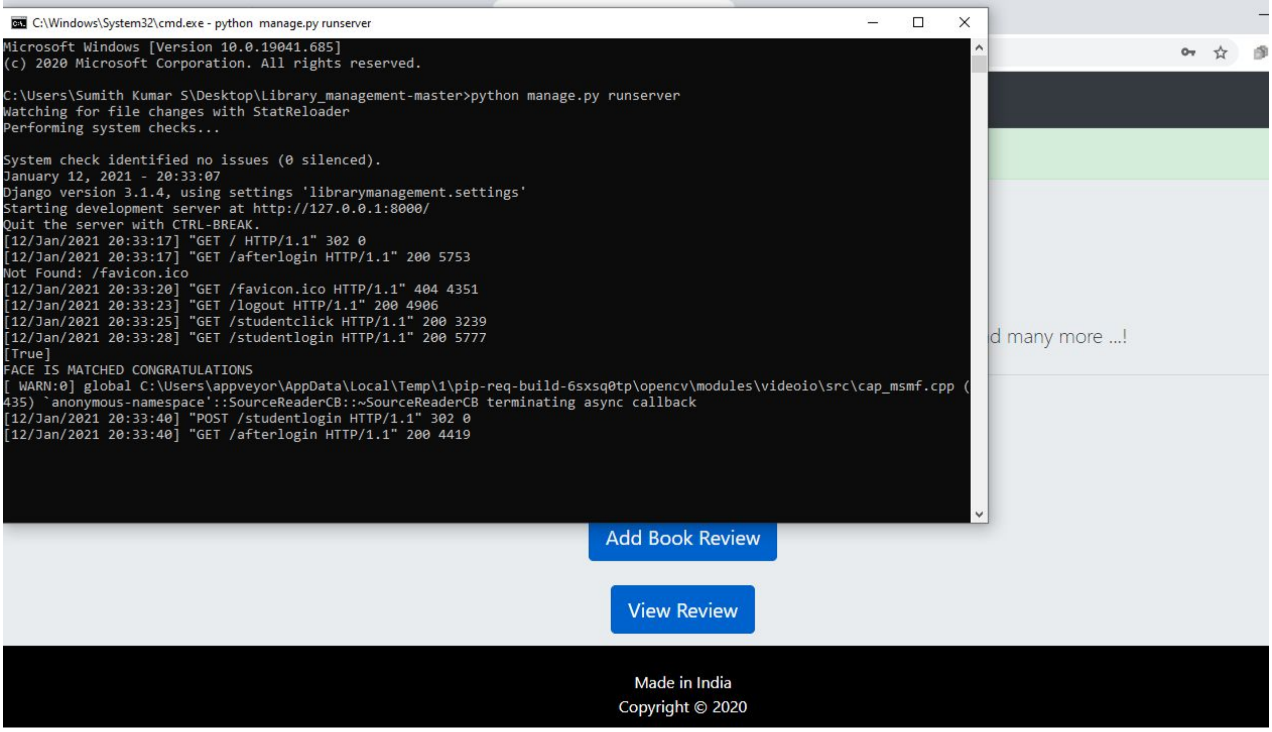


Fig 6.7 : List of Options for the Staff

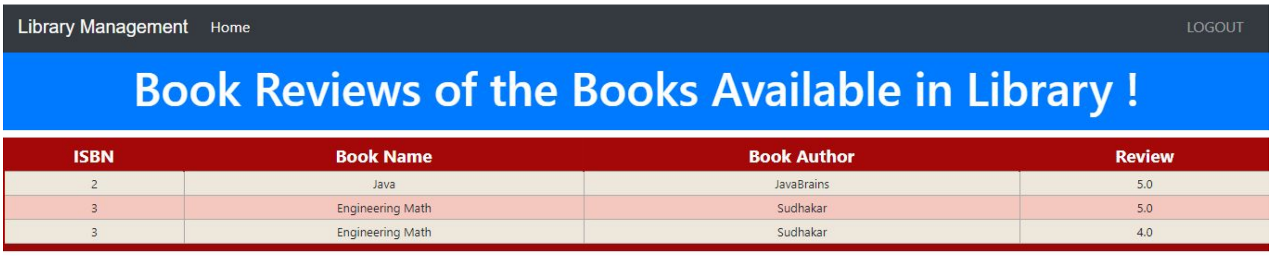


```
Microsoft Windows [Version 10.0.19041.685]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\Sumith Kumar S\Desktop\Library_management-master>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
January 12, 2021 - 20:33:07
Django version 3.1.4, using settings 'librarymanagement.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
[12/Jan/2021 20:33:17] "GET / HTTP/1.1" 302 0
[12/Jan/2021 20:33:17] "GET /afterlogin HTTP/1.1" 200 5753
Not Found: /favicon.ico
[12/Jan/2021 20:33:20] "GET /favicon.ico HTTP/1.1" 404 4351
[12/Jan/2021 20:33:23] "GET /logout HTTP/1.1" 200 4906
[12/Jan/2021 20:33:25] "GET /studentclick HTTP/1.1" 200 3239
[12/Jan/2021 20:33:28] "GET /studentlogin HTTP/1.1" 200 5777
[True]
FACE IS MATCHED CONGRATULATIONS
[ WARN:0] global C:\Users\appveyor\AppData\Local\Temp\1\pip-req-build-6sxsq0tp\opencv\modules\videoio\src\cap_msmf.cpp (
435) `anonymous-namespace'::SourceReaderCB::~SourceReaderCB terminating async callback
[12/Jan/2021 20:33:40] "POST /studentlogin HTTP/1.1" 302 0
[12/Jan/2021 20:33:40] "GET /afterlogin HTTP/1.1" 200 4419
```

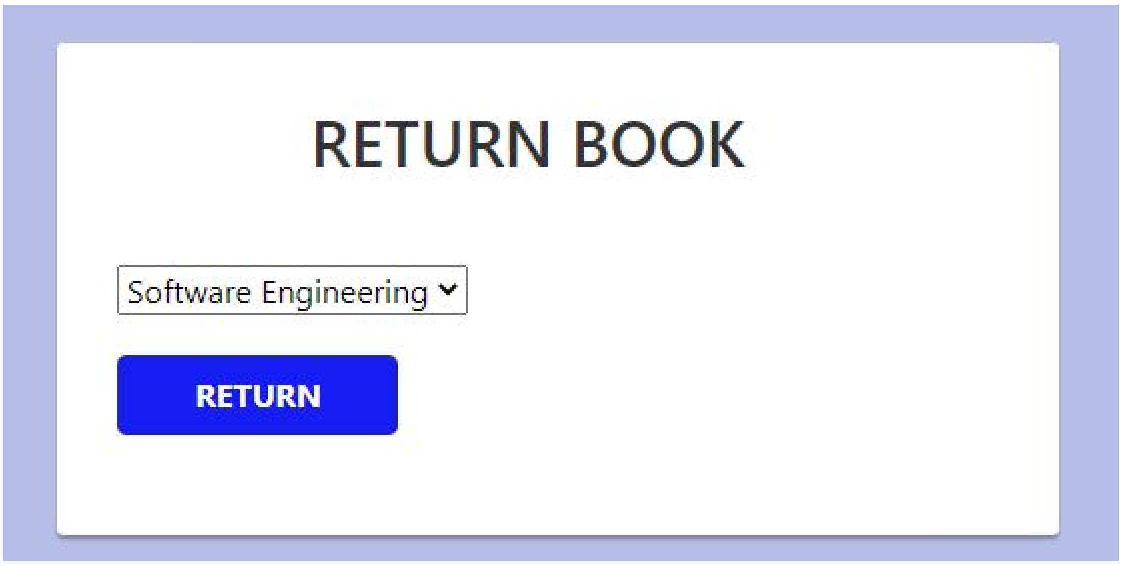
Fig 6.8 : Face Recognition upon Login Success Alert



Book Reviews of the Books Available in Library !

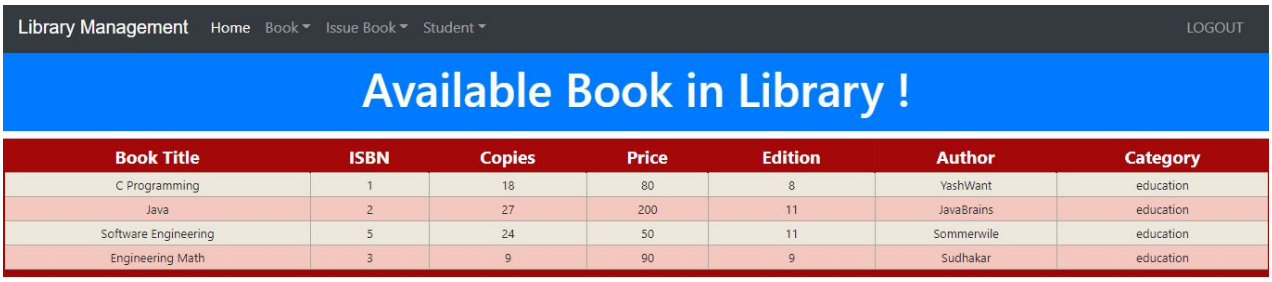
ISBN	Book Name	Book Author	Review
2	Java	JavaBrains	5.0
3	Engineering Math	Sudhakar	5.0
3	Engineering Math	Sudhakar	4.0

Fig 6.9 : Review of the Books(reader functionality)



RETURN BOOK

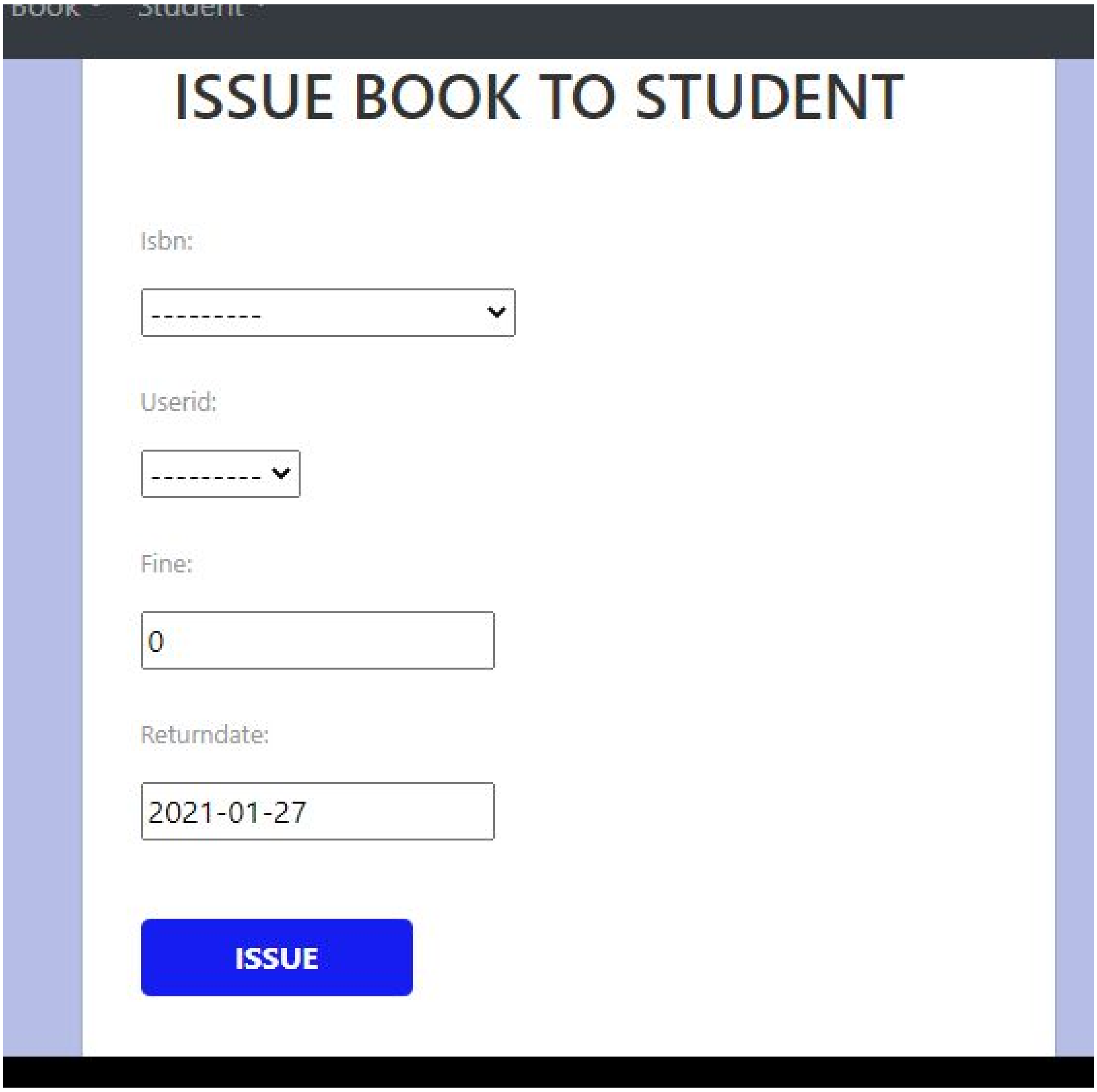
Fig 6.10 : Return Book Status(reader functionality)



Available Book in Library !

Book Title	ISBN	Copies	Price	Edition	Author	Category
C Programming	1	18	80	8	YashWant	education
Java	2	27	200	11	JavaBrains	education
Software Engineering	5	24	50	11	Sommerwile	education
Engineering Math	3	9	90	9	Sudhakar	education

Fig 6.11 : Check available books (staff functionality)



ISSUE BOOK TO STUDENT

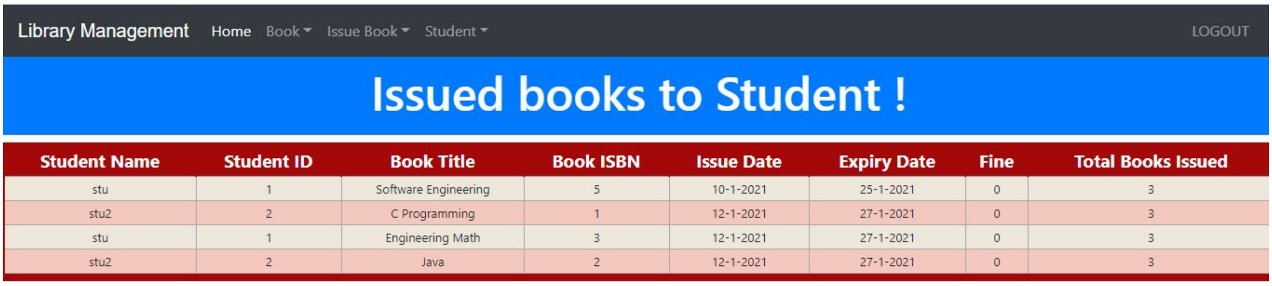
Isbn: ---------

Userid: ---------

Fine: 0

Returndate: 2021-01-27

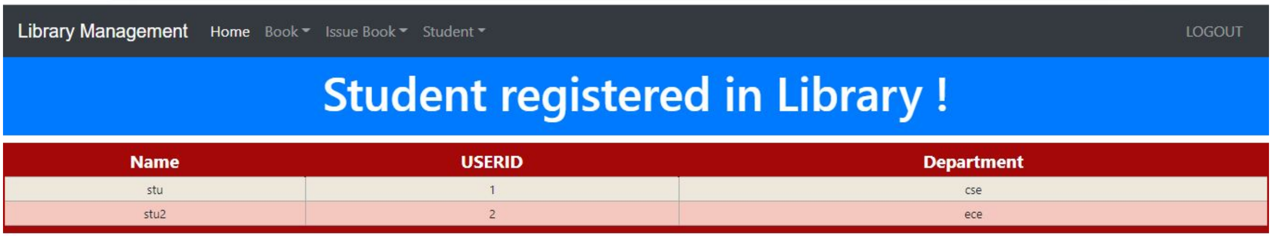
Fig 6.12 : Issue Books (staff functionality)



Issued books to Student !

Student Name	Student ID	Book Title	Book ISBN	Issue Date	Expiry Date	Fine	Total Books Issued
stu	1	Software Engineering	5	10-1-2021	25-1-2021	0	3
stu2	2	C Programming	1	12-1-2021	27-1-2021	0	3
stu	1	Engineering Math	3	12-1-2021	27-1-2021	0	3
stu2	2	Java	2	12-1-2021	27-1-2021	0	3

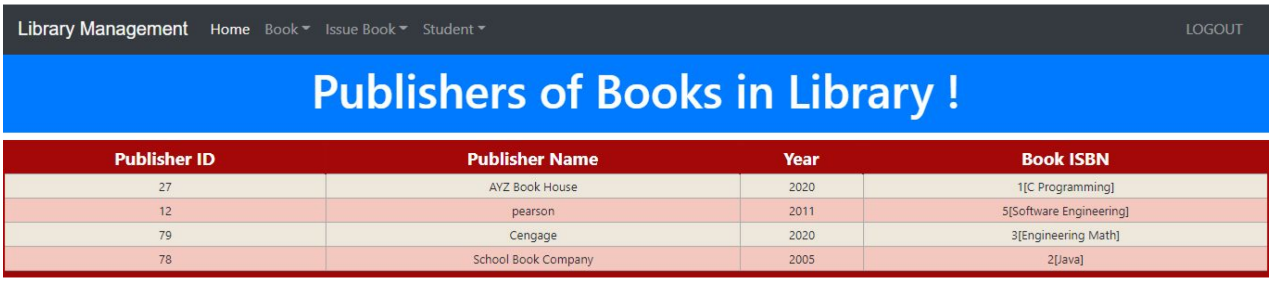
Fig 6.13 : Check issued books (staff functionality)



Student registered in Library !

Name	USERID	Department
stu	1	cse
stu2	2	ece

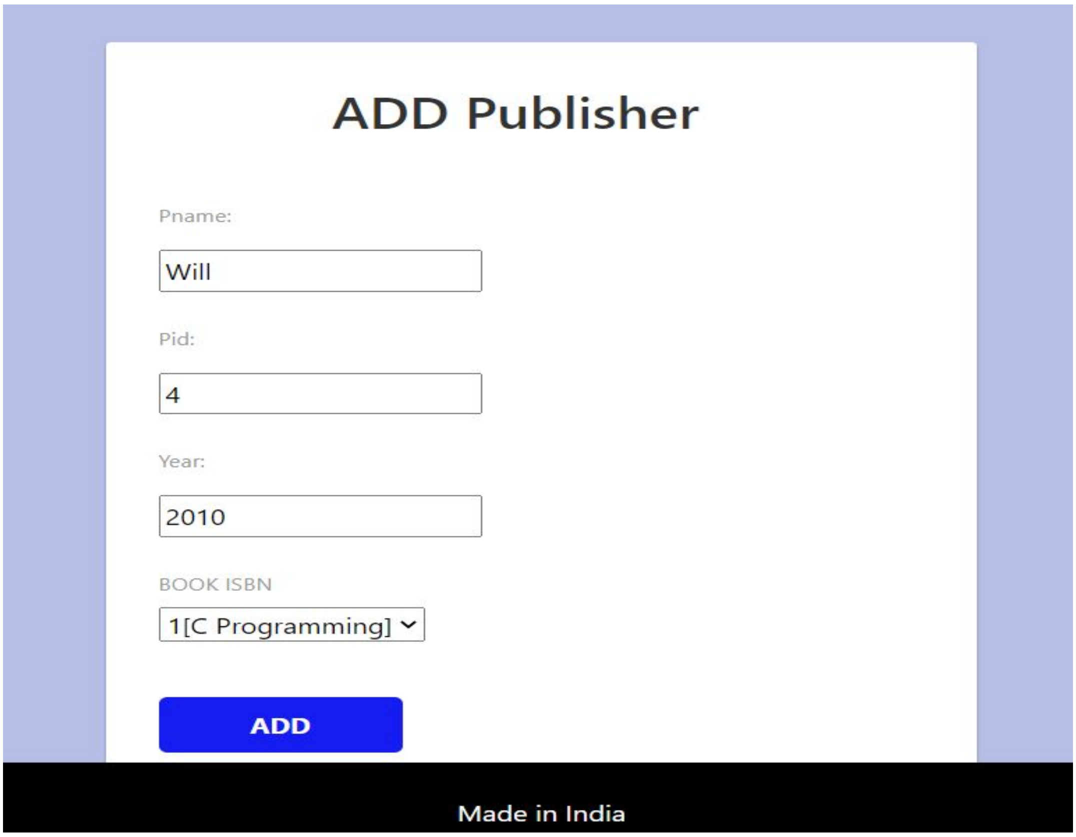
Fig 6.14 : Check registered Students (staff functionality)



Publishers of Books in Library !

Publisher ID	Publisher Name	Year	Book ISBN
27	AYZ Book House	2020	1[C Programming]
12	pearson	2011	5[Software Engineering]
79	Cengage	2020	3[Engineering Math]
78	School Book Company	2005	2[Java]

Fig 6.15 : Check book publishers (staff functionality)



ADD Publisher

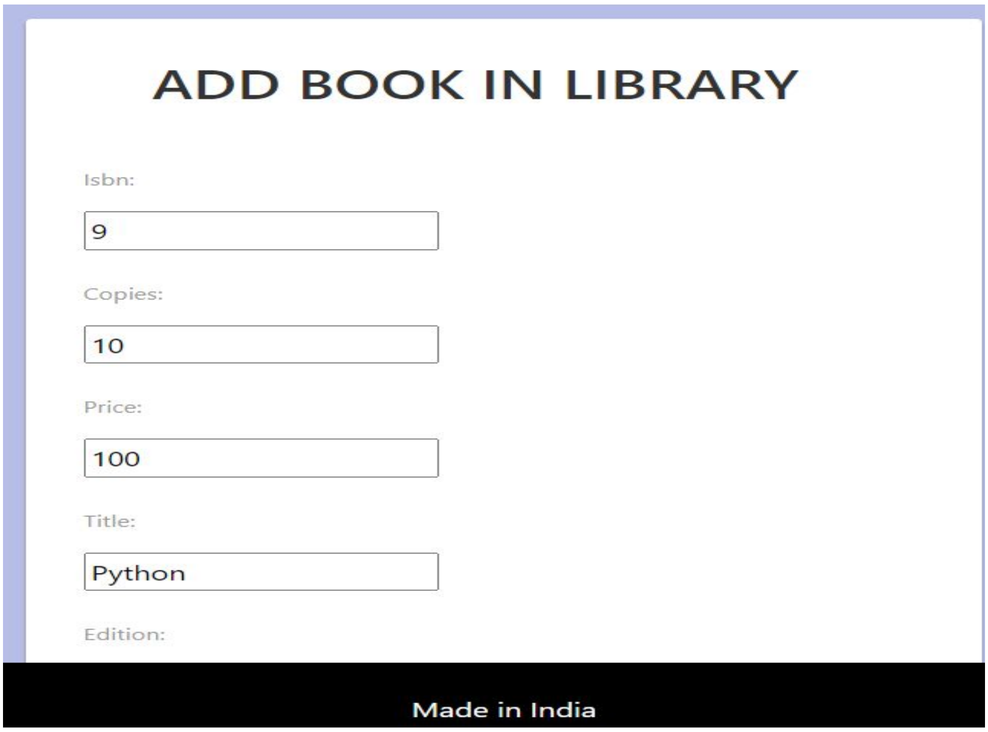
Pname: Will

Pid: 4

Year: 2010

BOOK ISBN: 1[C Programming]

Fig 6.16 : Add book publishers (staff functionality)



ADD BOOK IN LIBRARY

Isbn:

9

Copies:

10

Price:

100

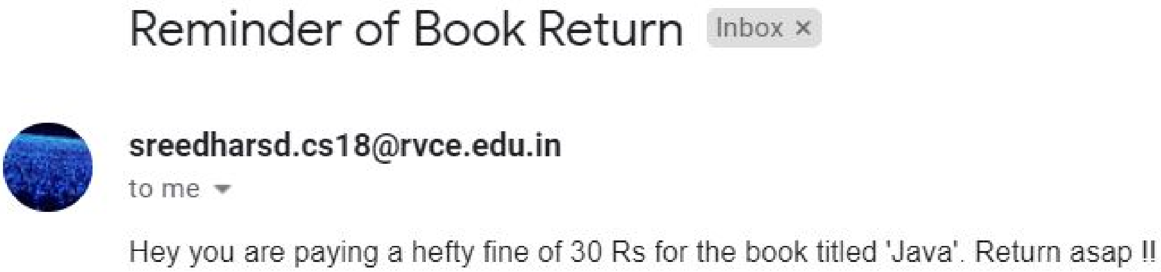
Title:

Python

Edition:

Made in India

Fig 6.17 : Add book to library (staff functionality)



Reminder of Book Return

sreedharsd.cs18@rvce.edu.in

to me

Hey you are paying a hefty fine of 30 Rs for the book titled 'Java'. Return asap !!

Fig 6.18 : Book Alerts with the help of email