Final Report on
IIT Library

Submitted to

SPL II Evaluation Committee 2020-2021
Bachelor of Science in Software Engineering
Institute of Information Technology
Noakhali Science and Technology University

Submitted By

Prosanto Deb
prosanto2514@student.nstu.edu.bd

Abdullah Alif
alif2514@student.nstu.edu.bd

Ratna Kumer Tripura
ratna2514@student.nstu.edu.bd

Supervised By

Dipok Chandra Das
Assistant Professor
Institute of Information Technology
dipok.iit@nstu.edu.bd

Submission Date: 20th September, 2022

Final Report on IIT Library

By

Prosanto Deb
ASH1925005M
Year-03 Term-01
prosanto2514@student.nstu.edu.bd

Abdullah Alif
ASH1925009M
Year-03 Term-01
alif2514@student.nstu.edu.bd

Ratna Kumer Tripura
ASH1825042M
Year-03 Term-01
ratna2514@student.nstu.edu.bd

Approved By

Dipok Chandra Das
Assistant Professor
Institute of Information Technology
dipok.iit@nstu.edu.bd

Table of Contents

List of Figures

List of Tables

1. Project Description

1.1 Introduction

Introducing “IIT Library” provides a hassle-free system. Using this system, we can easily borrow books and return them within a specific period. To manage membership, the system has a unique registration process for each individual authentic member of the IIT Library. Using this system, the users can easily search the books in the library. Moreover, the system calculates the fine due for non-return book and also get electronic copies of the books if they are available. For more details, please view Appendix B.

1.1.1 Motivation

The Seminar Library of IIT is a very important part of every single student because it provides reading materials. Almost all of our library's operations need a well-organized system. A system that assists librarians in keeping a database of newly released books as well as books that users have borrowed and their due dates. Each and every activity in our library need to be fully automated by a system. That’s why our team came forward and by implementing our system is the greatest approach to keeping many books organized, maintained, and handled properly. For more details, please view Appendix B.

1.1.2 Aims and Objectives

The initial aim and objectives of our system were to provide these features,

1. Manage Membership: Our first objective was to maintain a detailed database of the members. The system will automatically authenticate each member of the IIT Library.
2. Borrow Books: We have to borrow existing books and remove them digitally. Also, keep track of when the book will be returned.
3. Searching: Our system will help the librarian to search for books in the library. The search functions can be filtered to the need of each user.
4. Fine Calculation: The system calculates the fine due for non-return or lost and damaged books. The members will be notified about the fines by the system.
5. Advanced Booking: Any member can make an advanced booking to take a book using our system. The member will get an online receipt which will be valid for a specific period of days. After that period the validity will be expired.
6. Electronic Copy of Books: Members can get electronic copies of our books too only if the copies are permitted to be distributed by the publisher.

For more details, please view Appendix B.

1.2 Target Users

- Customers who request, purchase, and/or pay for the software product – IIT, NSTU
- End-users who actually use the product directly or indirectly. For our system, we are providing library facilities to students, faculty members, officers and staff of IIT, NSTU.

For more details, please view Appendix B.

1.3 Requirements

We have collected these requirements from our SRS document. For more details, please view Appendix A.

Table 01 List of requirements

No.	Requirement	Completed
FR-1	User (student, faculty member, officer, staff) registration and login a registered account of IIT Library.	YES
FR-2	User (student, faculty member, officer, staff) can send a borrow request to IIT Library.	YES
FR-3	User (student, faculty member, officer, staff) can extend the time period of the borrowed book.	YES
FR-4	User (student, faculty member, officer, staff) can access the electronic copy of the book if it is available after login.	YES
FR-5	The system can calculate the fine and both the user and librarian will be able to see it.	YES
FR-6	The system will generate a daily report with all the required information.	YES
FR-7	The system will send an email with all required information as a response for some specific operations.	YES
DR-1	Searching books using the keyword or book name or author name.	YES
PR-1	Faster searching for books.	YES
MR-1	Develop maintainable code.	YES

1.4 Models, Tools and Resources

1.4.1 Model

SPL I has had a significant influence on our project. As we iteratively developed our project using the methodology we had chosen, we encountered a variety of issues. We want to implement something new this time, and after learning from the project from last year, we have decided to use the agile model to do it. Break tasks into smaller iterations using the agile model. The project risk is

reduced and the overall project delivery time requirements are reduced due to the project's breakdown into smaller components. For more details, please view Appendix B.

1.4.2 Tools and Resources

Table 02 List of tools

Category	Name or Description
Text Editor	Visual Studio Code
IDE	PhpStorm 2022 2.1
Server	Xampp
RDBMS	MySQL
Language	HTML, CSS, JavaScript, PHP, SQL
Learning Resource	1. Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages 2nd Edition by Elisabeth Robson 2. Head First JavaScript Programming: A Brain-Friendly Guide 1st Edition by Eric Freeman 3. PHP & MySQL: Server-side Web Development by Jon Duckett

For more details, please view Appendix B.

1.5 Project Members

Table 03 List of members

Team Members	Supervisor
Prosanto Deb (ASH1925005M) Abdullah Alif (ASH1925009M) Ratna Kumer Tripura (ASH1825042M)	Dipok Chandra Das Assistant Professor Institute of Information Technology

2. User Guide

2.1 User end (Student, Faculty Member, Officer, Staff)

3. Source Code Documentation

3.1 Project Structure

3.1.1 index.php

This file contains our homepage content, that is, the text and images that people see when they first go to our site.

3.1.2 images folder

This folder contains all basic images required for our project. Like button icons or in the navbar, header, footer.

3.1.3 inc folder

This folder contains all the necessary files that are needed to be included.

3.1.4 css ans js folder

This folder contains all the necessary .css and .js files.

3.1.5 assets folder

This folder contains two sub-folder called cover(contains all the cover pages of the library books) and another is pdf(contains all the pdf files of the library books).

3.1.6 File naming convention

In this project all files are named starting with the actor name.

Syntax = [actor name][functionality] like ibrarianUserDetails.php, librarianNewBookRequests.php

3.2 Lines of Code

Table 04 List of lines of code

SL	Class	LOC	NCLOC	CLOC	Density of Comments
1	bookInfo.php	111	102	9	8.1 %
2	directorLibrarian.php	125	110	15	12 %
3	directorRemoveLibrarian.php	94	89	5	5.31 %
4	directorUserDetails.php	155	138	17	10.96 %
5	directorUsers.php	85	80	5	6.25 %
6	index.php	123	114	9	7.31 %
7	librarianAddNewBooks.php	97	93	4	4.12 %
8	librarianNewBookRequests.php	139	124	15	10.79 %
9	librarianProfile.php	132	115	17	12.88 %
10	librarianUserBooking.php	125	112	13	10.4 %
11	librarianUserBorrowedBooks.php	142	126	16	11.26 %
12	librarianUserDetails.php	152	134	18	11.84 %
13	librarianUsers.php	77	63	14	18.18 %
14	login.php	162	153	9	5.56 %
15	register.php	317	299	18	5.67 %
16	userBookInfo.php	156	143	13	8.33 %
17	userBookings.php	97	85	12	12.37 %
18	userBorrowedBooks.php	142	124	18	12.67 %
19	userHome.php	135	121	14	10.37 %
20	userNewBookRequest.php	114	106	8	7.01 %
21	userProfile.php	146	127	19	13.01 %
22	userWatchNewBookRequests.php	90	82	8	8.89 %
	Total Project	2916	2640	276	9.47 %

3.3 API Documentation

Table 05 List of functions

SL	Function	Parameter	Return
1	outputMessage()	message, type	void
2	fileUpload()	file, filename	void
3	islogged()	N/A	True or false
4	logout()	N/A	void
5	dbConnect()	N/A	True or false
6	drawHeader()	True or false	void
7	printRegisterForm()	N/A	void

4. SRS & Development Mapping

Table 06 List of use cases

Use Case No	Use Case	Implementation
UC 01	Send Registration Request	register.php
UC 02	Borrow Request	userBookings.php
UC 03	Extend Book Period	userBorrowedBooks.php
UC 04	Access Electronic Copy	userBookInfo.php
UC 05	Request New Book	userNewBookRequest.php
UC 06	Check Fine	userBorrowedBooks.php(shows as a list with specific book)
UC 07	Search Book (By User)	userHomePage.php
UC 08	Login	login.php
UC 09	Admin Login	login.php
UC 10	Logout	logout.php
UC 11	Send Emails	sendMail(message, receiver)
UC 12	Validate Borrow ID	isValid(borrowID)
UC 13	Authenticate User and Librarian	canLogin()
UC 14	Calculate Fine	calculateFine(dueDate)

UC 15	Show Books	userHome.php
UC 16	Borrower Receipt Generate	generateBorrowReceipt()
UC 17	Report Generate	directorHome.php
UC 18	Watch new book requests	librarianNewBookRequests.php, userNewBookRequests.php
UC 19	Send Request for approval	register.php
UC 20	Search Book (By Librarian)	librarianHome.php
UC 21	Add New Books	librarianAddNewBooks.php
UC 22	give Book	librarianUserBooking.php
UC 23	Approve Request	librarianUsers.php, directorUsers.php
UC 24	Add Librarian	directorLibrarian.php
UC 25	Delete Librarian	directorRemoveLibrarian.php
UC 26	Update Password	userProfile.php, librarianProfile.php
New	Show Detailed information of a book	bookinfo.php

5. Challenges and Future Work

The main challenges we faced during developing this system are,

1. Differentiate between booking services and borrowing services.
2. Taking accurate information for validating each member and checking if he/she is an authentic member of IIT or not.
3. Providing electronic copies for almost all books in IIT, Library.
4. Calculating the fine by comparing dates.
5. Sending emails as a response to some specific operations.

The future work we are proposing,

1. Sending text messages as a response because text message attracts more attention.
2. Implementing a payment gateway for taking fines.
3. Tracking a book precisely and making the member responsible to keep the book safe and return before the due date.
4. Generating library cards as soon as possible after the application form is completed by the member.
5. Implement barcoding for faster services.

Appendix A: SRS Document

User registration and login

FR-1	User (student, faculty member, officer, staff) registration and login a registered account of IIT Library.		
Description	User should register his/her account for the first time and be able to login to the account which was registered once. Already registered users will not face this stage. At first the registration request is checked by the librarian and then it is finally approved by the director or IIT, NSTU.		
Stakeholders	User(student, faculty member, officer, staff), System	**Priority**	High

Borrow Book

FR-2	User(student, faculty member, officer, staff) can send a borrow request to IIT Library.		
Description	To borrow a book, the user first needs to send a request to the IIT Library. The system will provide a receipt with all required information. This receipt will be valid for 72 hours. After showing the receipts to the librarian, he will provide the book and remove the book from the system which means the book is not in the library.		
Stakeholders	User(student, faculty member, officer, staff), Librarian	**Priority**	High

Extend Book Period

FR-3	User (student, faculty member, officer, staff) can extend the time period of the borrowed book.		
Description	The user has to log in and goes to his/her profile. There will be a list of borrowed books. From there a user can select and extend the borrowing period for 7days. But it will be valid only two times for each book.		
Stakeholders	User(student, faculty member, officer, staff), System	**Priority**	High

Access Electronic Copy

FR-4	User (student, faculty member, officer, staff) can access the electronic copy of the book if it is available.		
Description	The user has to login and go to his/her profile. There will be an option for searching books and it will show a list of available books. From there a user can select and access the electronic copy of the book is it is available.		
Stakeholders	User(student, faculty member, officer, staff), System	**Priority**	High

Calculate and Check Fine

FR-5	The system can calculate the fine and the user will be able to see it.		
Description	The user can see the fine after entering his/her profile and the system will provide a detailed information of the fine.		
Stakeholders	User(student, faculty member, officer, staff), System	**Priority**	High

Report Generate

FR-6	The system will generate a daily report with all required information.		
Description	Sometimes, it is required to see a daily report of which book is returned and which book is borrowed in a single day. That's why the system will perform this operation.		
Stakeholders	System	**Priority**	High

Send Email

FR-7	The system will send an email with all required information as a response for some specific operations.
Description	Operations like sending registration request, book borrowing requests, extending borrow period, updating passwords, and requesting new book needs to be notified by an email. It increases user satisfaction and leaves a footprint for any critical operation.

Stakeholders	System	**Priority**	High

Searching Books

DR-1	Searching books using the keyword or book name or author name.		
Description	If Any authentic user of our system wants to search a for viewing or downloading. He can search the book by the keyword or book name or author name.		
Stakeholders	User, Librarian, Director	**Priority**	High

Speed and Latency Requirements

PR-1	Faster searching for books.		
Description	When any authentic user of our system wants to search for a book then the user can feel the fast searching.		
Stakeholders	User, Librarian, Director	**Priority**	High

Maintenance Requirements

MR-1	Develop maintainable code		
Description	Maintainability must be ensured so that it can be modified later and will be readable.		
Stakeholders	Developers (Team Twins)	**Priority**	High

Use Case Descriptions

Table 07 Send Registration Request

Use Case No.	01	
Use Case	Send Registration Request	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) issues a request using IIT Library, expects the request will be approved and will be notified of any response via mail.	
Preconditions <what we expect is already the state of the world>	Must be a student or faculty member or officer or staff of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	"User Registration Request" sent successfully.	
Failed End Condition <the state of the world if goal abandoned>	"User Registration request" sending process failed.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System	
Trigger <the action upon the system that starts use case>	The user clicks the "Register" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the first page of the IIT Library.
	2	The user clicks the "Register" button.
	3.1 3.2 3.3 3.4	The user selects the "Register as Student" option. The user selects the "Register as Faculty Member" option. The user selects the "Register as Officer" option. The user selects the "Register as Staff" option.
	4	The user provides the required information.
	5	The user clicks the "Send Registration Request" button
	6	The system sends an email as a response to the user with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The user did not select any option.
	3a1	The registration page remains unchanged.
	4a	The user provided any invalid data.
	4a1	The webpage shows "Invalid Data".
	4a2	The specific field gets cleared in which the user has given the wrong data.
Quality Requirements	**Step**	**Requirement**
	4	The user should give valid information.

Table 08 Borrow Request

Use Case No.	02	
Use Case	Borrow Request	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) issues a borrow request using IIT Library, expects the book will be reserved for him/her, gets a system-generated receipt, and will be notified of any response via mail.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully and must be a student or faculty member or officer or staff of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The user has sent borrow requests successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user could not send a borrow request.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System	
Trigger	The user clicks the “Borrow Book” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the IIT Library home page after login.
	2	The user clicks the “Borrow Book” button.
	3	The system shows a list of available books along with a search option.
	4	The user searched for the book from the list.
	5	The desired book is found.
	6	The user clicks the book and the system shows book details.
	7	The user clicks the “Borrow” button.
	8	The user provides the time period.
	9	The user clicks the “Send Borrow Request” button.
	10	The system generates a borrow receipt with all required information.
	11	The system sends an email as a response with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The user searched using the search option.
	4a1 4a2 4a3	The user selects the “Search by Keyword” option. The user selects the “Search by Book Name” option. The user selects the “Search by Author Name” option.
	5a	The desired book is not found.
	5a1	The system prints a “Sorry” message and shows a button called “Request New Book”.
	5a2	Use case “Request New Book”.
Quality Requirements	**Step**	**Requirement**
	4	Searching available books should be fast and accurate

Table 09 Extend Borrow Period

Use Case No.	03	
Use Case	Extend Borrow Period	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) extends the borrowing period for 7 days using IIT Library and will be notified any response via mail.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully and must have borrowed any books earlier.	
Success End Condition <the state of the world upon successful completion>	The user has extended the borrowing period successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user could not extend the borrowing period.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System	
Trigger <the action upon the system that starts use case>	The user clicks the “Extend Borrow Period” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the IIT Library home page after login.
	2	The user enters into his/her profile.
	3	The user clicks the “Borrowed Books” button.
	4	The system shows a list of borrowed books by the user.
	5	The user searched for the book from the list.
	6	The desired book is found.
	7	The user clicks the book and the system shows book details.
	8	The user clicks the “Extend Borrow Period” button.
	9	The system checks of the validation.
	10	The system shows a “Successful” message.
	11	The system updates all required information about that user.
	12	The system sends an email as a response with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The book list is empty.
	4a1	The user did not borrow any book earlier.
	9a	The user is not valid to extend the borrowing period.
	9a1	The system shows an “Error” message.
	9a2	The user can not extend the borrowing period.
Quality Requirements	**Step**	**Requirement**
	9	A user who has already extended for 2 times, cannot extend further.
	11	The system will mark that book as unavailable for 7 days.

Table 10 Access Electronic Copy

Use Case No.	04	
Use Case	Access Electronic Copy	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) can access (view and download) an electronic copy of the book if available.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully and must be a student or faculty member or officer or staff of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The user has accessed (viewed and downloaded) the book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user could not access (view and download) the book.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System	
Trigger <the action upon the system that starts use case>	The user clicks the "Open Electronic Copy" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the IIT Library home page after login.
	2	The user clicks the "Borrow Book" button.
	3	The system shows a list of available books along with search option.
	4	The user searched for the book from the list.
	5	The desired book is found.
	6	The user clicks the book and the system shows book details.
	7	The user clicks the "Open Electronic Copy" button.
	8	The system opens the electronic copy of the book in a new window.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The user searched using the search option
	4a1 4a2 4a3	The user selects the "Search by Keyword" option. The user selects the "Search by Book Name" option. The user selects the "Search by Author Name" option.
	5a	The desired book is not found.
	5a1	The system prints a "Sorry" message and shows a button called "Request New Book".
	5a2	Use case "Request New Book".
	7a	The "Open Electronic Copy" button is not available.
	7a1	An Electronic copy of that particular book is not available.
Quality Requirements	**Step**	**Requirement**
	4	Searching available books should be fast and accurate.
	7	An Electronic copy can be viewed and downloaded only.

Table 11 Request New Book

Use Case No.	05	
Use Case	Request New Book	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) can request for a new book to be available in the IIT Library.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully and must be a student or faculty member or officer or staff of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The user has requested the new book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user could not request any new book.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System, Librarian	
Trigger <the action upon the system that starts use case>	The user clicks the "Request New Book" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the IIT Library home page after login.
	2	The user clicks the "Request New Book" button.
	3	The system shows some fields to enter information about the book.
	4	The user provides information about the book.
	5	The user clicks the "Send Request for New Book" button.
	6	The system sends an email as a response with all required information.
	7	Wait until the librarian performs any operation.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The user has provided any invalid data.
	4a1	The webpage shows "Invalid Data".
	4a2	The specific field gets cleared in which the user has given the wrong data.
	7a	Librarian checks for new book requests.
	7a1	Use Case "Watch New Book Requests"
Quality Requirements	**Step**	**Requirement**
	4	The user should give valid information.
	7	The system should send an email to the librarian as a notification regarding the new book requests.

Table 12 Check Fine

Use Case No.	06	
Use Case	Check Fine	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) can check how much fine he/she has to pay.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully, must borrow a book, and exceed the return date.	
Success End Condition <the state of the world upon successful completion>	The user has checked the fine successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user could not check the fine.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System	
Trigger <the action upon the system that starts use case>	The user clicks the “Check Fine” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the IIT Library home page after login.
	2	The user enters into his/her profile.
	3	The user clicks the “Check Fine” button.
	4	The system calculates the fine.
	5	Use Case “Calculate Fine”.
	6	The system shows fine with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The user has not borrowed any book that the return date exceeded.
	4a1	The system shows the “No Fine To Pay” message.
Quality Requirements	**Step**	**Requirement**
	5	The fine calculation must be valid.

Table 13 Search Book (By User)

Use Case No.	07	
Use Case	Search Book (By User)	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) can search for any book.	
Preconditions <what we expect is already the state of the world>	Must be a student or faculty member or officer or staff of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The user has searched for any book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user could not search for any book.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff) System	
Trigger <the action upon the system that starts use case>	The user clicks the “Search” icon on the top of a button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user enters the first page of the IIT Library.
	2	The user enter some text in the input field
	3	The user clicks the “Search” icon on the top of a button.
	4	The system shows a list of all available books.
	5	The user searches for a book.
	6	The desired book is found.
	7	The system shows all required information regarding the book.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The user wants to search by a specific option.
	3a1 3a2 3a3	The user selects the “Search by Keyword” option. The user selects the “Search by Book Name” option. The user selects the “Search by Author Name” option.
	6a	The desired book is not found.
	6a1	The system shows a “Sorry” message
Quality Requirements	**Step**	**Requirement**
	3	Searching available books should be fast and accurate
	7	The user can see only some basic and required information.

Table 14 Login

Use Case No.	08	
Use Case	Login	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff) and a librarian can log in to the system.	
Preconditions <what we expect is already the state of the world>	Must be a student or faculty member or officer or staff or librarian of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The user or librarian has successfully logged in to the system.	
Failed End Condition <the state of the world if goal abandoned>	The user or librarian could not log in to the system.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff), Librarian System	
Trigger <the action upon the system that starts use case>	The primary actor clicks the "Login" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The primary actor enters the first page of the IIT Library.
	2.1 2.1 2.3 2.4 2.5	The primary actor clicks "Login as Student" option. The primary actor clicks "Login as Faculty Member" option. The primary actor clicks "Login as Officer" option. The primary actor clicks "Login as Staff" option. The primary actor clicks "Login as Librarian" option.
	3	The primary actor enters username and password.
	4	The system checks for validation. Use Case "Authenticate User".
	5	The primary actor successfully logs in to the system.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The primary actor provides invalid data.
	4a1	The system prints a "Error" message.
	4a2	The primary actor cannot log in to the system.
Quality Requirements	**Step**	**Requirement**
	3	The system will not allow incorrect passwords more than 5 times and if happens, keeps waiting for 15 minutes.
	4	The primary actor authentication must be accurate.

Table 15 Admin Login

Use Case No.	09	
Use Case	Admin Login	
Goal <a longer statement of the goal in context if needed>	An admin (Director) can log in to the system.	
Preconditions <what we expect is already the state of the world>	Must be the Director of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The admin has successfully logged in to the system.	
Failed End Condition <the state of the world if goal abandoned>	The admin could not log in to the system.	
Primary Actors: **Secondary Actors:**	Admin (Director) System	
Trigger <the action upon the system that starts use case>	The admin clicks the "Login" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The admin enters the first page of the IIT Library.
	2	The admin clicks "Login as Admin" option.
	3	The admin enters username and password.
	4	The system checks for validation. Use Case "Authenticate User".
	5	The admin successfully logs in to the system.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The admin provides invalid data.
	4a1	The system prints an "Error" message.
	4a2	The admin cannot log in to the system.
Quality Requirements	**Step**	**Requirement**
	3	The system will not allow incorrect passwords more than 3 times and if happens, keeps waiting for 10 minutes.
	4	The admin authentication must be accurate.

Table 16 Logout

Use Case No.	10	
Use Case	Logout	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff), a librarian, and the admin (Director) can log out from the system.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully and must be a user (student, faculty member, officer, staff) or a librarian, or the Director of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The user (student, faculty member, officer, staff) or the librarian or the admin has successfully logged out from the system.	
Failed End Condition <the state of the world if goal abandoned>	The admin could not log out from the system.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff), Librarian, Admin (Director) System	
Trigger <the action upon the system that starts use case>	The admin clicks the “Logout” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The primary actor enters the home page of IIT the Library after login.
	2	The primary actor clicks “Logout” button.
	3	The primary actor has successfully logged out from to the system.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	2a	The primary actor does not click “Log out” button.
	2a1	The system performs according to the instructions of primary actor.
	4a2	The primary actor cannot log in to the system.
Quality Requirements	**Step**	**Requirement**
	2	The system should ask “Are you Sure to Logout?” after clicking the logout button.

Table 17 Send Emails

Use Case No.	11	
Use Case	Send Emails	
Goal <a longer statement of the goal in context if needed>	A user (student, faculty member, officer, staff), a librarian, and the admin (Director) will be notified by email from the system for any specific purpose.	
Preconditions <what we expect is already the state of the world>	There should be any operation that requires a response via email. Use Cases (Send Registration Request, Borrow Request, Extend Borrow Period, Request New Book, Approve Request)	
Success End Condition <the state of the world upon successful completion>	A system-generated email has been sent to a specific destination with all the required information.	
Failed End Condition <the state of the world if goal abandoned>	A system-generated email has not been sent to a specific destination.	
Primary Actors: **Secondary Actors:**	System User (student, faculty member, officer, staff), Librarian, Admin (Director)	
Trigger <the action upon the system that starts use case>	An email request comes in.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The system sets the destination email address.
	2	The system sets the subject of the email.
	3	The system generates the body of the email.
	4	The system sends the email to the destination address.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	1a	The destination email address is not valid.
	1a1	The system shows an “Error” message.
	1a2	Email is not sent.
Quality Requirements	**Step**	**Requirement**
	3	The email body should be accurate and contain all the required information.

Table 18 Validate Borrow ID

Use Case No.	12	
Use Case	Validate Borrow ID	
Goal <a longer statement of the goal in context if needed>	A user's (student, faculty member, officer, staff) borrower ID is checked whether it is valid or not.	
Preconditions <what we expect is already the state of the world>	A user must borrow a book.	
Success End Condition <the state of the world upon successful completion>	A user's borrower ID is checked successfully.	
Failed End Condition <the state of the world if goal abandoned>	A user's borrower ID is not checked.	
Primary Actors: **Secondary Actors:**	System Librarian	
Trigger <the action upon the system that starts use case>	A borrower ID validation request comes in.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	The librarian clicks the "Check Borrower ID" button.
	3	The system checks the borrower ID.
	4	The borrower ID is valid.
	5	The system shows all information against the borrower ID.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The borrower ID is not valid.
	4a1	The system shows an "Error" message.
Quality Requirements	**Step**	**Requirement**
	4	The borrower ID validation must be accurate and the system shows all the required information against the borrower ID.

Table 19 Authenticate User and Librarian

Use Case No.	13	
Use Case	Authenticate User and Librarian	
Goal <a longer statement of the goal in context if needed>	A user's (student, faculty member, officer, staff) and librarian's login information borrower ID is checked whether it is authenticate or not.	
Preconditions <what we expect is already the state of the world>	A user's (student, faculty member, officer, staff) registration request must be approved by the Director of IIT, NSTU or a librarian must be added by the Director of IIT, NSTU	
Success End Condition <the state of the world upon successful completion>	A user's or librarian's login information is authenticated successfully.	
Failed End Condition <the state of the world if goal abandoned>	A user's or librarian's login information is not authenticated.	
Primary Actors: **Secondary Actors:**	System User (student, faculty member, officer, staff), Librarian	
Trigger <the action upon the system that starts use case>	An authentication request comes in while login.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user or librarian clicks the button required for login.
	2	The system checks the login information.
	3	The login information is valid.
	4	The system allows the secondary actors to log in and opens the home page of IIT, NSTU.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The login information is not valid or authenticated by the system.
	3a1	The system shows an "Error" message.
Quality Requirements	**Step**	**Requirement**
	2	The authentication must be accurate.

Table 20 Calculate Fine

Use Case No.	14	
Use Case	Calculate Fine	
Goal <a longer statement of the goal in context if needed>	The system calculates a user's (student, faculty member, officer, staff) fine.	
Preconditions <what we expect is already the state of the world>	A user (student, faculty member, officer, staff) must borrow a book, and exceed the return date.	
Success End Condition <the state of the world upon successful completion>	A user's fine is calculated successfully.	
Failed End Condition <the state of the world if goal abandoned>	A user's fine is not calculated.	
Primary Actors: **Secondary Actors:**	System User (student, faculty member, officer, staff)	
Trigger <the action upon the system that starts use case>	An fine calculation request comes in while checking the fine by the user.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user clicks the "Check Fine" button.
	2	The system checks whether the user has borrowed any book.
	3	The system checks whether the user has exceeded the return date.
	4	The system checks whether the user has any previous fine.
	5	The system calculates the fine.
	6	The system shows fine with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	2a	The user did not borrow any books.
	2a1	The system shows the "No Fine To Pay" message.
	3a	The user did not exceed the return date.
	3a1	The system shows the "No Fine To Pay" message.
	4a	The user does not have any previous fine.
	4a1	The system shows the current fine only.
Quality Requirements	**Step**	**Requirement**
	5	A fine of Taka 5 will be added for each day delay for a single book.

Table 21 Show Books

Use Case No.	15	
Use Case	Show Books	
Goal <a longer statement of the goal in context if needed>	The system shows the book's information to the user (student, faculty member, officer, staff), librarian, and Director of IIT, NSTU.	
Preconditions <what we expect is already the state of the world>	The librarian must have added at least one book information.	
Success End Condition <the state of the world upon successful completion>	The system shows all the required information of books.	
Failed End Condition <the state of the world if goal abandoned>	The system is unable to show all the required information of books.	
Primary Actors: **Secondary Actors:**	System User (student, faculty member, officer, staff), Librarian, Director	
Trigger <the action upon the system that starts use case>	Search book request comes in.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1.1 1.2	The secondary actor search for a book. The secondary actor clicks a book.
	2	The system finds the book.
	3	The system shows the book with all the required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	2a	The system does not find the book.
	2a1	The system shows the "Sorry" message.
Quality Requirements	**Step**	**Requirement**
	2	Book searching should take a keyword or book name or author name.

Table 22 Borrower Receipt Generate

Use Case No.	16	
Use Case	Borrower Receipt Generate	
Goal <a longer statement of the goal in context if needed>	The system generates a receipt with all required information of the borrower or the user (student, faculty member, officer, staff) along with book information and return date.	
Preconditions <what we expect is already the state of the world>	The user (student, faculty member, officer, staff) must borrow a book.	
Success End Condition <the state of the world upon successful completion>	The system has generated a borrower receipt successfully.	
Failed End Condition <the state of the world if goal abandoned>	The system could not generate a borrower receipt.	
Primary Actors: **Secondary Actors:**	System User (student, faculty member, officer, staff)	
Trigger <the action upon the system that starts use case>	Borrower receipt generate request comes in.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The user clicks the “Send Borrow Request” button.
	2	The system collects all required information.
	3	The system generates the receipt.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	2a	The system cannot collect all information.
	2a1	The system shows the “Unable to generate receipt” message.
Quality Requirements	**Step**	**Requirement**
	2	The receipt must contain the user information, book information and return date.

Table 23 Report Generate

Use Case No.	17	
Use Case	Report Generate	
Goal <a longer statement of the goal in context if needed>	The system generates a report with all required information of the transactions (borrowed and returned books) made in a single day.	
Preconditions <what we expect is already the state of the world>	There must have been at least one borrowed or returned book in a single day.	
Success End Condition <the state of the world upon successful completion>	The system has generated a report successfully.	
Failed End Condition <the state of the world if goal abandoned>	The system could not generate a report.	
Primary Actors: **Secondary Actors:**	System Librarian, Director	
Trigger <the action upon the system that starts use case>	Report generate request comes in.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1.1 1.2	The librarian clicks the “Report Generate” button. The director clicks the “Report Generate” button
	2	The system collects all required information.
	3	The system generates the report.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	2a	The system cannot collect all information.
	2a1	The system shows the “Unable to generate report” message.
Quality Requirements	**Step**	**Requirement**
	2	The receipt must contain the book information and user information related to the borrowing and the returning books.

Table 24 Watch New Book Requests

Use Case No.	18	
Use Case	Watch new book requests	
Goal <a longer statement of the goal in context if needed>	The librarian watches how many requests of a new book arrived in the system by the users (student, faculty member, officer, staff).	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian and there must have at least one request for a new book.	
Success End Condition <the state of the world upon successful completion>	The librarian watches for new book requests successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not watch for new book requests.	
Primary Actors: **Secondary Actors:**	Librarian System, User(student, faculty member, officer, staff)	
Trigger <the action upon the system that starts use case>	Report generate request comes in.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	The librarian clicks the “Watch New Book Requests” button.
	3	The system shows the information about the requests.
	4	The librarian analyses the validity and feasibility of the requested books.
	5	The librarian discuss with the director and take necessary steps.
	6	The librarian provides a response via email about the actions regarding the new book request.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The system cannot find any new book requests.
	3a1	The system shows the “No New Book Request” message.
	4a	The book request is not valid or feasible.
	4a1	The librarian provides a response via email that the request cannot be implemented.
Quality Requirements	**Step**	**Requirement**
	3	The system should give an email to the librarian’s account about the new book request.

Table 25 Send Request for Approval

Use Case No.	19	
Use Case	Send Request for approval	
Goal <a longer statement of the goal in context if needed>	The librarian sends registration requests made by the users (student, faculty member, officer, staff) to the director for approval.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian and there must have at least one registration request by the users (student, faculty member, officer, staff)	
Success End Condition <the state of the world upon successful completion>	The librarian sends the registration requests to the director for approval successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not send the registration requests to the director for approval.	
Primary Actors: **Secondary Actors:**	Librarian User (student, faculty member, officer, staff), Director	
Trigger <the action upon the system that starts use case>	Registration request comes in to the system.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	The librarian clicks the “Watch Requests for Approval” button.
	3	The system shows the information about available the requests.
	4	The librarian checks the validity of the registration requests.
	5	The librarian clicks the “Send Request for Approval” button. (Only the Valid Requests)
	6	The system sends an email as a response with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The system cannot find any new registration requests.
	3a1	The system shows the “No New Registration Request” message.
	4a	The registration request is not valid.
	4a1	The librarian provides a response via email that the request cannot be granted.
Quality Requirements	**Step**	**Requirement**
	3	The system should give an email to the librarian’s account about the new registration request.

Table 26 Search Book (By Librarian)

Use Case No.	20	
Use Case	Search Book (By Librarian)	
Goal <a longer statement of the goal in context if needed>	The librarian can search for any book.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The librarian has searched for any book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not search for any book.	
Primary Actors: **Secondary Actors:**	Librarian System	
Trigger <the action upon the system that starts use case>	The librarian clicks the "Search" icon on the top of a button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	The librarian enter some text in the input field.
	3	The librarian clicks the "Search" icon on the top of a button.
	4	The system shows a list of all available books.
	5	The librarian searches for a book.
	6	The desired book is found.
	7	The system shows all required information regarding the book.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The librarian wants to search by a specific option.
	3a1 3a2 3a3	The librarian selects the "Search by Keyword" option. The librarian selects the "Search by Book Name" option. The librarian selects the "Search by Author Name" option.
	6a	The desired book is not found.
	6a1	The system shows a "Sorry" message.
Quality Requirements	**Step**	**Requirement**
	3	Searching available books should be fast and accurate.

Table 27 Add New Book

Use Case No.	21	
Use Case	Add New Book	
Goal <a longer statement of the goal in context if needed>	The librarian can add all information of a new book.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian of IIT, NSTU and the information of the book must be unique.	
Success End Condition <the state of the world upon successful completion>	The librarian has added all information of a new book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not add information of a new book.	
Primary Actors: **Secondary Actors:**	Librarian System	
Trigger <the action upon the system that starts use case>	The librarian clicks the "Add New Book" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	The librarian clicks the "Add New Book" button.
	3	The librarian provides all necessary information of a new book.
	4	The librarian clicks the "Add Book" Button".
	5	The system checks whether the information is valid.
	6	The system checks whether the information is unique.
	7	The system updates the database.
	8	The system shows a "Successful" message.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	5a	The information is not valid.
	5a1	The system shows an "Error" message.
	6a	The information is not unique.
	6a1	The system shows an "Error" message.
Quality Requirements	**Step**	**Requirement**
	7	The database must be updated after every operation performed by the librarian.

Table 28 Update Book Information

Use Case No.	22	
Use Case	Update Book Information	
Goal <a longer statement of the goal in context if needed>	The librarian can update the information of an existing book.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian of IIT, NSTU and there must be at least one book in the system to update information.	
Success End Condition <the state of the world upon successful completion>	The librarian has updated the information of an existing book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not update the information of an existing book.	
Primary Actors: **Secondary Actors:**	Librarian System	
Trigger <the action upon the system that starts use case>	The librarian clicks the "Update Book Information" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	Use Case "Search Book".
	3	The librarian clicks the "Update Book Information" button.
	4	The librarian edits the information where required.
	5	The librarian clicks the "Update" button.
	6	The system checks whether the information is valid.
	7	The system checks whether the information is unique.
	8	The system updates the database.
	9	The system shows a "Successful" message.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	6a	The information is not valid.
	6a1	The system shows an "Error" message.
	7a	The information is not unique.
	7a1	The system shows an "Error" message.
Quality Requirements	**Step**	**Requirement**
	8	The database must be updated after every operation performed by the librarian.

Table 29 Delete Book

Use Case No.	23	
Use Case	Delete Book	
Goal <a longer statement of the goal in context if needed>	The librarian can delete the information of an existing book.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian of IIT, NSTU and there must be at least one book in the system to delete information.	
Success End Condition <the state of the world upon successful completion>	The librarian has deleted the information of an existing book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not delete the information of an existing book.	
Primary Actors: **Secondary Actors:**	Librarian System	
Trigger <the action upon the system that starts use case>	The librarian clicks the "Delete Book Information" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	Use Case "Search Book".
	3	The librarian clicks the "Delete Book" button.
	4	The system updates the database.
	5	The system shows a "Successful" message.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	2a	The book is not found.
	2a1	The system shows a "Sorry" message.
Quality Requirements	**Step**	**Requirement**
	4	The database must be updated after every operation performed by the librarian.

Table 30 Remove Book

Use Case No.	24	
Use Case	Remove Book	
Goal <a longer statement of the goal in context if needed>	The librarian can remove the book from the system which is borrowed by any user.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the librarian of IIT, NSTU and there must be at least one book in the system to remove.	
Success End Condition <the state of the world upon successful completion>	The librarian has removed the borrowed book successfully.	
Failed End Condition <the state of the world if goal abandoned>	The librarian could not remove the borrowed book.	
Primary Actors: **Secondary Actors:**	Librarian System, User (student, faculty member, officer, staff)	
Trigger <the action upon the system that starts use case>	The librarian clicks the "Remove Book" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The librarian enters the home page of IIT Library after login.
	2	The librarian clicks the "Remove Book" button.
	3	The librarian enters the borrower ID.
	4	Use Case "Validate Borrow ID".
	5	The librarian clicks the "Remove" button.
	6	The system updates the database.
	7	The system shows a "Successful" message.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	4a	The borrow id is not valid.
	4a1 4a2	The system shows an "Error" message. The book cannot be removed.
Quality Requirements	**Step**	**Requirement**
	6	The database must be updated after every operation performed by the librarian.

Table 31 Approve Request

Use Case No.	25	
Use Case	Approve Request	
Goal <a longer statement of the goal in context if needed>	The director can approve a registration request.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the Director of IIT, NSTU and there must be at least one registration request.	
Success End Condition <the state of the world upon successful completion>	The director has approved the registration request successfully.	
Failed End Condition <the state of the world if goal abandoned>	The director could not approve the registration request.	
Primary Actors: **Secondary Actors:**	Director System, Librarian	
Trigger <the action upon the system that starts use case>	The director clicks the "Pending Requests" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean-up after>	**Step**	**Action**
	1	The director enters the home page of IIT Library after login.
	2	The director clicks the "Pending Requests" button. (Send from Librarian)
	3	The director clicks the "Approve" button.
	4	The system updates the database.
	5	The system shows a "Successful" message.
	6	The system sends an email as a response with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The director clicks the "Reject" button.
	3a1	The system sends an email of rejection as a response to the user's email account.
Quality Requirements	**Step**	**Requirement**
	4	The database must be updated after every operation is performed.

Table 32 Add Librarian

Use Case No.	26	
Use Case	Add Librarian	
Goal <a longer statement of the goal in context if needed>	The director can add a librarian.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the Director of IIT, NSTU.	
Success End Condition <the state of the world upon successful completion>	The director has added a librarian successfully.	
Failed End Condition <the state of the world if goal abandoned>	The director could not add a librarian.	
Primary Actors: **Secondary Actors:**	Director System, Librarian	
Trigger <the action upon the system that starts use case>	The director clicks the "Add Librarian" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean-up after>	**Step**	**Action**
	1	The director enters the home page of IIT Library after login.
	2	The director clicks the "Add Librarian" button.
	3	The director enters all required information of a librarian.
	4	The director clicks the "Add" button.
	5	The information is valid and unique.
	6	The system updates the database.
	7	The system shows a "Successful" message.
	8	The system sends an email as a response to the librarian with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	5a	The information is not valid or unique.
	5a1	The system shows an "Error" message.
Quality Requirements	**Step**	**Requirement**
	6	The database must be updated after every operation is performed.

Table 33 Delete Librarian

Use Case No.	27	
Use Case	Delete Librarian	
Goal <a longer statement of the goal in context if needed>	The director can delete all information of an existing librarian.	
Preconditions <what we expect is already the state of the world>	Log in to the system by the Director of IIT, NSTU and there must be at least one librarian to delete.	
Success End Condition <the state of the world upon successful completion>	The director has deleted all the information of an existing librarian successfully.	
Failed End Condition <the state of the world if goal abandoned>	The director could not delete the information of an existing librarian.	
Primary Actors: **Secondary Actors:**	Director System	
Trigger <the action upon the system that starts use case>	The director clicks the "Delete Librarian" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean-up after>	**Step**	**Action**
	1	The director enters the home page of IIT Library after login.
	2	The director clicks the "Delete Librarian" button.
	3	The system shows the information of the librarian.
	4	The director clicks the "Delete" button.
	5	The system updates the database.
	6	The system shows a "Successful" message.
	7	The system sends an email as a response to the librarian with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The system does not find any information of a librarian.
	3a1	The system shows the "No Librarian to Delete" message.
Quality Requirements	**Step**	**Requirement**
	5	The database must be updated after every operation is performed.

Table 34 Delete User

Use Case No.	28	
Use Case	Delete User	
Goal <a longer statement of the goal in context if needed>	The director can delete all information of an existing user (student, faculty member, officer, staff).	
Preconditions <what we expect is already the state of the world>	Log in to the system by the Director of IIT, NSTU and there must be at least one user to delete.	
Success End Condition <the state of the world upon successful completion>	The director has deleted all the information of an existing user (student, faculty member, officer, staff) successfully.	
Failed End Condition <the state of the world if goal abandoned>	The director could not delete the information of an existing user (student, faculty member, officer, staff).	
Primary Actors: **Secondary Actors:**	Director System	
Trigger <the action upon the system that starts use case>	The director clicks the “Delete User” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean-up after>	**Step**	**Action**
	1	The director enters the home page of IIT Library after login.
	2	The director clicks the “Delete User” button.
	3	The system shows a list of the information of the users.
	4	The director clicks the “Delete” button
	5	The system updates the database.
	6	The system shows a “Successful” message.
	7	The system sends an email as a response to the user with all required information.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	3a	The system does not find any information of a user.
	3a1	The system shows the “No User to Delete” message
Quality Requirements	**Step**	**Requirement**
	7	The database must be updated after every operation is performed.

Table 35 Update Password

Use Case No.	29	
Use Case	Update Password	
Goal <a longer statement of the goal in context if needed>	The user (student, faculty member, officer, staff), and the librarian can update his/ her password.	
Preconditions <what we expect is already the state of the world>	Log in to the system successfully by the user (student, faculty member, officer, staff) and the librarian.	
Success End Condition <the state of the world upon successful completion>	The user and librarian has updated the password successfully.	
Failed End Condition <the state of the world if goal abandoned>	The user and librarian could not update the password.	
Primary Actors: **Secondary Actors:**	User (student, faculty member, officer, staff), Librarian System	
Trigger <the action upon the system that starts use case>	The primary actor clicks the "Update Password" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean-up after>	**Step**	**Action**
	1	The primary actor enters the home page of IIT Library after login.
	2	The primary actor clicks the "Update Password" button.
	3	The primary actor enters the old password.
	4	The primary actor enters the new password.
	5	The primary actor clicks the "Update" button.
	6	The system finds that the old password is matched.
	7	The system updates the database.
	8	The system shows a "Successful" message.
	9	The system sends an email as a response to the primary actor that the password is changed.
Alternative Flows <a: condition causing branching> <a1: action or name of sub-use case>	**Step**	**Branching Action**
	6a	The old password is not matched.
	6a1	The system shows a "Error" message.
Quality Requirements	**Step**	**Requirement**
	7	The database must be updated after every operation is performed.

Source: SRS (Full Document)

Appendix B: Project Proposal

Introduction

This is a proposal report for Software Project Lab II. This project will be developed during the 5th semester of BSSE, Institute of Information Technology, Noakhali Science and Technology University from 12 June 2022 by the proposal presentation, and the final project will be delivered on 08 September 2022. Our team will work along with our regular academic courses.

Motivation

A library is a collection of books that are accessible for use. We can borrow books and have to return them within a specific period. In our institute, we also have a seminar library. But it is a matter of regret that we don't have any automated and online library management system. Our existing library performs all operations manually and can't keep track of the books accurately. It's too difficult to find a book from the library. The solution is either we have to buy paid software or develop a web-based library system of our own.

For our team, it is the best time to utilize the period of SPL II and build a web-based library system that can save the external costs to buy expensive software and protect national resources. We will also implement a feature where a member can make a request for a book remotely. Our web-based system will show who is the last person to take a specific book and when will return the book. The system will also store electronic copies that are permitted by the publisher.

Objectives

Our system will fulfill the following objectives to provide a hassle-free system.

1. Manage Membership: Our first objective is to maintain a detailed database of the members. The system will generate a library card for each and individual authentic member of the IIT Library.
2. Book Tracing: Our second objective is to trace the movement of books. The location of any book at any point in time can be traced. Misplaced or missing books can be traced easily.
3. Borrow Books: We have to borrow new books and add them digitally. Irrelevant and outdated books are deleted.
4. Bar-coding and Scanning: Our system will give specific identification to each book. All books, old and new, are bar-coded based on title, author, topic, and date of publishing. The database is automatically updated when books are scanned while issuing or returning.
5. Searching: Our system will help both the office staff and the members to search the catalog of books in the library. The search functions can be filtered to the need of each user.

6. Fine Calculation: The system calculates the fine due for non-return or lost and damaged books. The members will be notified about the fines by the system.

Target Customers

The target customers for our system are students, faculty members denoted as members in the proposal. We have a role for permitting to get the library card which will be performed by the Director of IIT, NSTU. The work of checking in or checking out and adding new books to the system will be performed by one of the staff of the office.

Model

Our project is highly inspired by SPL I. We faced different problems with our selected model to develop our project which was an iterative process. This time, taking experience from the previous year's project, we want to implement something new and we have selected the agile model to develop this project. Agile Model break tasks into smaller iterations. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.

Tools and Resources

Category	**Name or Description**
Text Editor	Visual Studio Code
Server	Xampp
RDBMS	MySQL
Language	HTML, CSS, JavaScript, PHP, SQL
Learning Resource	1. Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages 2nd Edition by Elisabeth Robson 2. Head First JavaScript Programming: A Brain-Friendly Guide 1st Edition by Eric Freeman 3. PHP & MySQL: Server-side Web Development by Jon Duckett

Challenges

The main challenges to face during developing this system are,

1. Defining strict rules and regulations for each operation and making accountable each member for any problems created by them.

2. Implementing a payment gateway and developing a transaction system for taking due and membership amounts from the members of our system.
3. Solving the complexity of distributing a specific book when the quantity of the book is less than the number of requests made by the library members.
4. Validate each member and check if he/she is an authentic member of IIT or not.
5. Generating library cards as soon as possible after the application form is completed by the member.
6. Tracking a book precisely and making the member responsible to keep the book safe and return before the due date.

Source: Project Proposal (Full Document)

Appendix C: Database Design

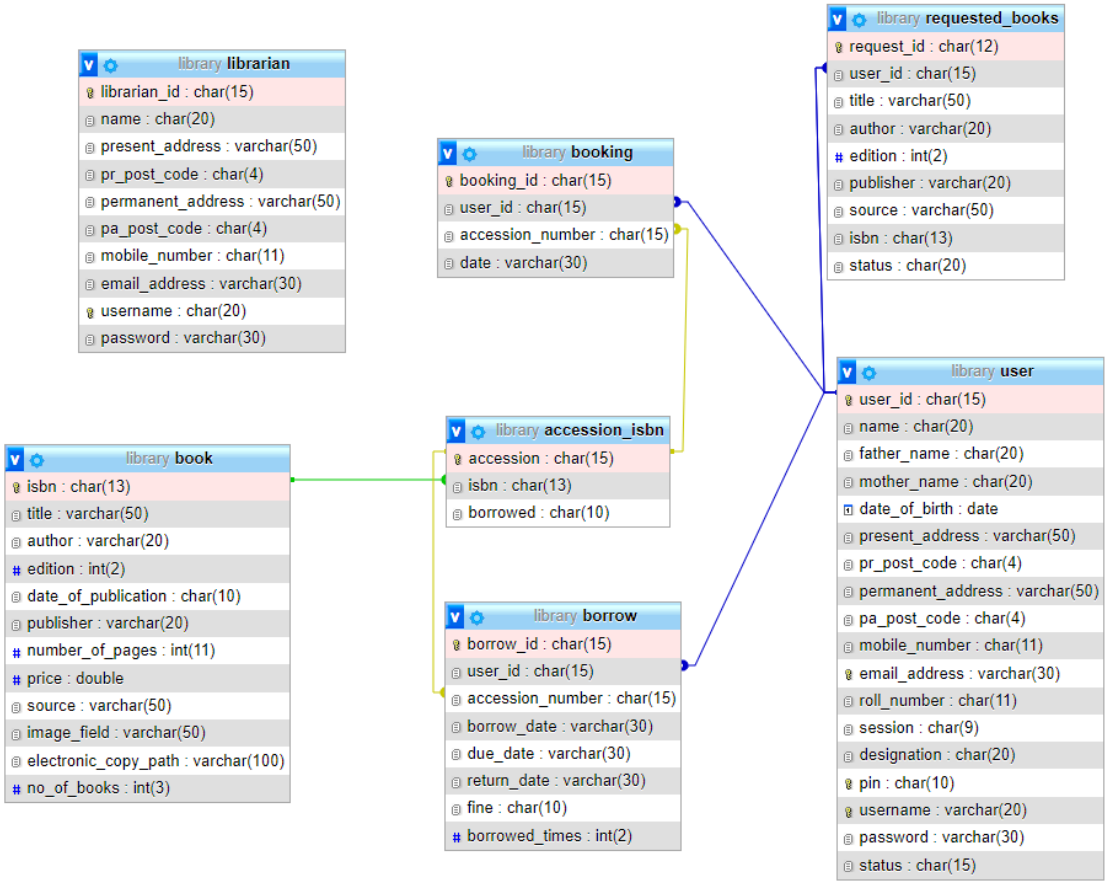

Figure 1 Schema Diagram of IIT Library