



**National University of Computer and Emerging Sciences**



## **Book Reviewing and Investment Platform**

### **FYP Team**

Zaviyaar Bin Irfan 19L-2225

Usama Iqbal 19L-2232

Muhammad Ahmad 19L-2300

### **Supervised by**

**Dr. Rana Asif Rehman**

**FAST School of Computing**

**National University of Computer and Emerging Sciences**

**Lahore, Pakistan**

**April 2023**



## **Anti-Plagiarism Declaration**

This is to declare that the above publication produced under the:

**Title: Book Reviewing and Investment Platform**

is the sole contribution of the author(s) and no part hereof has been reproduced on **as it is** basis (cut and paste) which can be considered as **Plagiarism**. All referenced parts have been used to argue the idea and have been cited properly. I/We will be responsible and liable for any consequence if violation of this declaration is determined.

Date: October 12<sup>th</sup>, 2022

Student 1

Name: Muhammad Ahmad

Signature: 

Student 2

Name: Usama Iqbal

Signature: 

Student 3

Name: Zaviyaar Bin Irfan

Signature: 

---

### **Authors' Declaration**

This states Authors' declaration that the work presented in the report is their own, and has not been submitted/presented previously to any other institution or organization.

## **Abstract**

The aim of this project is to provide the writers with a platform to help them find an investment for their books. Moreover, this platform will help investors in finding a suitable investment opportunity. The project also provides a facility for reviewing books, which assists writers in the improvement of their books as they receive feedback from reviewers. Neither, Investors have to read the book themselves nor they have to decide whether the book is promising or not. They can rely on reviewer's review for that purpose. Lastly, our website will also provide the publishing facility for writers to publish their books easily.

## **Executive Summary**

This report is about our development based final year project, named as Book Reviewing and Investment Platform. The reason we are doing this is that there are many talented writers who can write well but they do not have the resources to get their work published. Resultantly, their work never sees the light of day. We are providing writers with a platform where they can upload their books and get an investment to publish their work. There will be reviewers who will review the books uploaded by the writers. To prevent false or fake reviews we will get each book reviewed by at least three reviewers. Based on the reviews given by the reviewers, investors will choose which book they want to invest in. Every module will run through our system and we will handle the publishing end as well. It will be our responsibility from the beginning till the end, from getting the book reviewed to getting it published. The main goal of this project is to build a platform where writers can get an investment for their books to get published and provide investors with some potential books to invest in.

The problem we have is that there are people who cannot publish their work just for the reason they do not have enough capital to publish their work and their hard work is all wasted. We will have reviewers to review the books to judge whether the book is capable or not. As a reviewer can also be a writer, they might try to steal the poor writer's work by publishing the same thing under their name. Our project will devise ways to overcome their problems. The main objective of our project is to provide writers with a platform where they can get an investment and to get their book published without any hassle. Also, we provide investment opportunities for people who are strong financially and want to earn more and also who want to help these talented authors.

The project scope will consist of writers, reviewers and investors all at one platform. They will all work together to give life to a book. The writer will upload a book on our portal. Our system will assign the best reviewers to review it. If the reviews are positive an investor will provide capital for publishing of the books. We will have our partnered publishers who will be responsible for publishing the book. Furthermore, the Sustainable Development Goals (SDG) for this project are Decent Work and Economic Growth. As from our platform, many writers will get an investment, reviewers will get paid to give reviews, investors will also be able to invest their money in books and publishing houses will also make some revenue in publishing the books. Moreover, the second SDG is Quality Education. As the books will be reviewed by three of our reviewers, they will ensure the best quality whether they are educational books or novels, they will be the best of quality.

In addition to this, the constraints that this project has are that the investors will not be cross checked for their financial background. However, they will be blocked from the website if they are found to be fraud. Apart from this, the business opportunities that this project offers are numerous. Firstly, a writer can get an investment for his book. Secondly, reviewers will be paid for performing their task. Thirdly, investors will get their capital back with profits. Lastly, the publishing houses will also earn their share by publishing the books. Moving on, the high-level goals of this project are to provide writers with reviewers and investors with potential books to invest in. The problems of stakeholders will include the scammers. They might sign up as investors and may not have any capital to invest. There might be people who sign up as reviewers but don't know how to review books. Also, there can be reviewers who try to steal the intellectual property of writers and publish it under their name. Our system will cater all these problems by adding watermarks, checking plagiarism and blocking scam users.

The relevant research and the related work have also been discussed. There was less research available for our topic, however, there was a lot of related work. There are a number of websites that provide reviewing services for books. In addition to that, the print-on-demand services are

also available. Booklist and IngramSpark are both the working examples of these two aspects of our project. The reviewing and publishing are going on for quite a while now, but there exists no platform that reviews the book and then publishes it. This project combines both these aspects and adds the investment part as well.

The software requirement specifications are also given in this report. This also includes the list of features. This website has the ability to check for plagiarism of the books uploaded automatically. A watermark of our website will also be added to each page of the book preventing theft of intellectual property. Moreover, our system will recommend reviewers based on their interests and previous work. In addition to this, the functionalities of all the users have also been discussed. They include the login/sign up and other main functionalities of all the users. The non-functional requirements include the usability, availability, extensibility and performance. Assumptions include that the investors have enough capital to invest, and every user has some basic knowledge on how to use a computer and internet. Also, there are use cases, providing every function by the users. The database design gives a brief idea about the relationship between different users. Moreover, risk analysis provides the potential technical and business risks of this project.

Moving on, this document provides overview of the system, consideration that we considered while making design, constraints due to which design was altered, goals and objectives design that the design achieved. It also provides system architecture diagram and architecture of components of system. Furthermore, it also mentions architectural strategies of system, class diagram of the system and sequence diagrams of different operations that different users will perform is also given. Lastly, it also covers policies and tactics that changed the interface of the system.

Moving on, implementation details of different components such as data storage and retrieval, authentication, and watermarking of books are provided in the next chapter of this project. Future work that is to be done and the details on how this project can be enhanced and conclusion is also provided in the next chapter.

## Table of Contents

Table of Contents .....	i
List of Tables .....	iv
List of Figures .....	v
Chapter 1: Introduction .....	1
1.1 Purpose of this Document .....	1
1.2 Intended Audience .....	1
1.3 Definitions, Acronyms, and Abbreviations .....	1
Chapter 2: Project Vision.....	3
2.1 Problem Domain Overview .....	3
2.2 Problem Statement .....	3
2.3 Problem Elaboration .....	3
2.4 Goals and Objectives .....	3
2.5 Project Scope .....	4
2.6 Sustainable Development Goal (SDG) .....	4
2.7 Constraints .....	4
2.8 Business Opportunity.....	4
2.9 Stakeholders Description/ User Characteristics.....	4
2.9.1 Stakeholders Summary .....	5
2.9.2 Key High-Level Goals and Problems of Stakeholders .....	5
Chapter 3: Literature Review / Related Work .....	6
3.1 Definitions, Acronyms, and Abbreviations .....	6
3.2 Detailed Literature Review .....	6
3.2.1 Related Research Work 1.....	6
3.2.2 Related Research Work 2.....	7
3.3 Literature Review Summary Table .....	7
3.4 Conclusion .....	8
Chapter 4: Software Requirement Specifications .....	9
4.1 List of Features .....	9
4.1 Functional Requirements .....	9
4.2 Quality Attributes.....	10
4.3 Non-Functional Requirements .....	11
4.4 Assumptions.....	11
4.5 Hardware and Software Requirements .....	11
4.5.1 Hardware Requirements.....	11
4.5.2 Software Requirements .....	11
4.6 Use Cases .....	12
4.6.1 Login .....	12
4.6.2 Sign-Up .....	12
4.6.3 Sign-Up for Reviewer .....	13
4.6.4 Reset password.....	13
4.6.5 Change password .....	14
4.6.6 Edit Profile .....	15
4.6.7 Upload Book .....	15
4.6.8 Resubmit Book.....	16
4.6.9 View List of Pending Books .....	16
4.6.10 View Comments of Reviewer .....	17
4.6.11 Rate Reviewer .....	17
4.6.12 Download an Assigned Book.....	18

4.6.13 Fill the Review Form .....	18
4.6.14 Invest in a Book .....	19
4.6.15 View Complete Review .....	19
4.6.16 View Invested Books .....	19
4.6.17 Rate Reviewer.....	20
4.6.18 View Comments by Reviewer .....	20
4.6.19 Reject a Book.....	21
4.6.20 Block a User.....	21
4.6.21 Assign Book to Reviewers.....	22
4.6.22 Notify the Reviewer.....	22
4.6.23 Extend Deadline.....	23
4.7 Graphical User Interface .....	24
4.8 Database Design.....	35
4.8.1 ER Diagram .....	35
4.8.2 Data Dictionary .....	35
4.9 Risk Analysis .....	37
Chapter 5: High-Level and Low-Level Design .....	38
5.1 System Overview .....	38
5.2 Design Considerations .....	38
5.2.1 Assumptions and Dependencies .....	38
5.2.2 General Constraints.....	38
5.2.3 Goals and Guidelines .....	39
5.2.4 Development Methods .....	40
5.3 System Architecture .....	41
5.3.1 Subsystem Architecture .....	42
5.4 Architectural Strategies.....	44
5.4.1 Reuse of existing product.....	44
5.4.2 Use of Programming language.....	44
5.4.3 User-Interface Paradigm .....	44
5.4.4 Extension and Enhancement of software .....	44
5.4.5 Error detection and Recovery .....	44
5.4.6 Concurrency and Synchronization.....	44
5.5 Domain Model/Class Diagram .....	45
5.6 Sequence Diagrams.....	46
5.7 Policies and Tactics.....	54
5.7.1 Conventions .....	54
5.7.2 Testing the Software .....	54
5.7.3 Maintenance of Software .....	54
5.7.4 Vulnerability Check .....	54
5.7.5 End User Interface .....	54
5.7.6 Tools .....	54
Chapter 6: Implementation and Test Cases .....	55
6.1 Implementation .....	55
6.1.1 Prototype Description .....	55
6.1.2 Authentication.....	55
6.1.3 Saving and retrieval of data .....	56
6.1.4 Upload a book .....	56
6.1.5 Rate Reviewers .....	56
6.1.6 Resubmit a book .....	56
6.1.7 Watermarking a book.....	56

6.1.8 Plagiarism Checker .....	56
6.1.9 Recommendation System.....	57
6.2 Test case Design and description.....	57
6.2.1 User Login Test Case.....	57
6.2.2 Sign-Up Test Case .....	58
6.2.3 Reset Password Test Case.....	58
6.2.4 Change Password Test Case .....	59
6.2.5 Edit Profile Test Case .....	59
6.2.6 Upload Book Test Case .....	60
6.2.7 Resubmit Book Test Case.....	60
6.2.8 View List of Pending Books Test Case .....	61
6.2.9 View comments of Reviewer Test Case .....	61
6.2.10 Rate Reviewer Test Case .....	62
6.2.11 Download an Assigned Book Test Case.....	62
6.2.12 Fill Review Form Test Case .....	63
6.2.13 Invest in Book Test Case .....	63
6.2.14 View Review Test Case .....	64
6.2.15 View Invested Books Test Case .....	64
6.2.16 Rate Reviewer Test Case .....	65
6.2.17 View comments of Reviewer Test Case .....	65
6.2.18 Reject Book Test Case .....	66
6.2.19 Block User Test Case.....	66
6.2.20 Assign Book Test Case .....	67
6.2.21 Notify a Reviewer Test Case .....	67
6.2.22 Extend the Deadline Test Case .....	68
6.3 Test Metrics .....	68
6.3.1 Test case Matric.No.1 .....	68
Chapter 7: User Manual .....	69
7.1 Login .....	69
7.2 Sign-Up .....	69
7.3 Edit Profile .....	69
7.4 Upload a Book .....	69
7.5 Resubmit a Book.....	70
7.6 Review a Book .....	70
7.7 Invest in a Book .....	71
Chapter 8: Conclusion and Future Work .....	72
References.....	73

**List of Tables**

Table 1: Summary Table.....	7
Table 2: Data Dictionary Table.....	35

## List of Figures

Figure 1: Login Page.....	24
Figure 2: Signup Page.....	24
Figure 3: Signup Page for Reviewers .....	25
Figure 4: Writer Homepage .....	26
Figure 5: Writer's pending books .....	27
Figure 6: Reviewer's Homepage .....	28
Figure 7: Review page .....	29
Figure 8: Investor's Homepage.....	30
Figure 9: Investment Page .....	31
Figure 10: Admin's Homepage.....	32
Figure 11: Under-Review books Page .....	32
Figure 12: Reviewed books page.....	33
Figure 13: Manage Writer page .....	33
Figure 14: Manage Reviewer page .....	34
Figure 15: Manage investor page.....	34
Figure 16: ER Diagram.....	35
Figure 17: Architecture Diagram .....	41
Figure 18: Admin component Diagram .....	42
Figure 19: Writer component Diagram.....	42
Figure 20: Reviewer component Diagram .....	43
Figure 21: Investor component Diagram .....	43
Figure 22: Class Diagram .....	45
Figure 23: Assign Reviewer.....	46
Figure 24: Block a user .....	46
Figure 25: Change password.....	47
Figure 26: Download a book.....	47
Figure 27: Edit profile.....	48
Figure 28: Extend deadline .....	48
Figure 29: Invest in a book .....	49
Figure 30: Login .....	49
Figure 31: Notify a reviewer.....	50
Figure 32: Rate a reviewer.....	50
Figure 33: Reset Password.....	51
Figure 34: See a review.....	51
Figure 35: Sign Up.....	52
Figure 36: Upload Book.....	53
Figure 37: View Books .....	53

## Chapter 1: Introduction

Many writers lack the resources to publish their first book. Despite this, they keep writing in the hope that they may someday earn enough money to finance the publishing of it. With our web-based development project, we hope to give writers a platform to get investment to publish their books. Our website will be a straightforward gateway where various writers can register and receive investments based on how our esteemed reviewers rate their publications. Our system will ask our registered investors to invest in the book if the reviews are favorable. Reviews can only be given by registered reviewers, who will get paid for giving reviews. Additionally, writers will be able to avail themselves of the facility of approaching reviewers to get suggestions on their ongoing work. Moreover, we will also provide the facility to have the invested books published by our renowned publishing partners, so that both the writer and investor are saved from the hustle of finding a publisher. Lastly, our intelligent system will recommend the best reviewers using artificial intelligence techniques.

### 1.1 Purpose of this Document

The purpose of this project is to provide writers with a platform to get their books reviewed and then published without having to spend huge sums of money. In addition to this, providing investors with a platform for finding good books to invest in is also a purpose that this project will serve.

### 1.2 Intended Audience

The intended audience for this project is the writers who wants to get their books reviewed or the ones who can write but are not able to publish them due to financial issues. In addition to this, the people who have some publications or are fond of reading can become reviewers to earn in their free time. Lastly, the intended audience is the investors who are looking for a potential investment. They can invest in the books which are reviewed positive by our reviewers.

### 1.3 Definitions, Acronyms, and Abbreviations

List all important definitions, the acronyms and abbreviations used in this document.

**IDE:** Integrated Development Environment

**JS:** JavaScript

**DB:** Database

**ERD:** Entity Relationship Diagram

**API:** Application Programming Interface

**GUI:** Graphical User Interface

**SDG:** Sustainable Development Goal

The first chapter of this report gives the introduction regarding the Book Reviewing and Investment Platform being implemented and describes the purpose of this document, the intended audience, and the definitions and abbreviations of complex terms. The second chapter illustrates the vision of the project, the problem and its solution along with goals and objectives of the project. The project scope and sustainable development goals are also discussed in this

chapter. Moreover, the constraints of this project and the business opportunities are also key points in this chapter. Lastly, we have the stakeholder's description and the high-level goals in this chapter. A literature review of the previously conducted research can be found in the third chapter. The third chapter also mentions all the detailed literature review, the research work, and the summary tables of the literature review as well. The fourth chapter comprises of all the software requirement specifications, the list of features it has. It also highlights the functional and non-functional requirements along with the quality attributes and assumptions. Moreover, it contains the hardware and software requirements. Most importantly, it consists of User Cases with GUI, the database design which mainly contains ER diagram and data dictionary. In addition to that, this chapter includes the Risk analysis. Moving on, in fifth chapter, this document provides system overview, consideration that we did while making design, constraints on which design was alter, goals and objectives of design. It also provides system architecture and architecture of components of system. Furthermore, it also mentions architectural strategies of system, class diagram and sequence diagrams of different operations. Lastly, this chapter covers policies and tactics that changed the interface of the system.

## Chapter 2: Project Vision

The project vision is to provide writers a pathway to get their books published without spending a huge amount of money. It also aims to provide readers or writers a platform to earn money by reviewing books. On the other hand, it also visions to provide investors an easy solution to find books on which they can put their trust in and invest in.

### 2.1 Problem Domain Overview

Our website is basically a platform where a user can register himself as a writer, reviewer and investor. The writer can upload a book to get it reviewed by a reviewer. A reviewer is a user who will review books uploaded by writers and get paid. The reviewers can give positive as well as negative reviews. The reviews given by the reviewer will decide whether the book is good for investing or not. Based on the reviews given by the reviewers, an investor will invest some capital for the publication of the book. Lastly, our platform will provide the investors with our partnered publishers to publish the books that are invested in through our web-based platform.

### 2.2 Problem Statement

The major problem is that there are writers who can write well but they don't have the capital to publish these books. In addition to that, investors who want to invest in books do not have any platform to look out for books having great potential.

### 2.3 Problem Elaboration

The main problem is that there are writers who can write well but they do not have the capital to get their books in publishing phase, resultantly, their hard work is not rewarded. The first problem we will solve is by providing capable writers with investors who are willing to provide capital for publishing their books. This will create a sub-problem that how the investors would know in which book they should invest in. We will solve this problem by providing reviewers who will review the books uploaded by writers and based on their reviews the investors will choose, in which book they are willing to invest their money in.

This will create many secondary problems. One of which is what if the reviewers try to steal the intellectual property of writers and tries to publish it by their name. We will overcome this problem by taking a consent by the reviewers that they cannot publish anything being reviewed. Secondly, our system will automatically add watermark on every page of the book which is sent for review. Another problem we might face is what if a reviewer provides us with a false or fake review without even reading the book. To ensure the quality of reviews we will have each book reviewed by three different reviewers and after the review is finished, we will collect their feedback in which we will ask different questions about the book.

### 2.4 Goals and Objectives

The goal of this project is to facilitate writers with an online space to find potential investors for publishing their book. Secondly, to give authors an online facility where they can solicit feedback from reviewers and use it to enhance the content of their books. Moreover, an online portal for reviewers will also be established where they can earn money by reviewing books. In addition to this, to establish an interface where investors can look for books to invest in. Lastly, our system will enforce the contract on its end through publishing it via partnered publishing houses.

## 2.5 Project Scope

As discussed earlier, the project will consist of a website where different writers, reviewers, and investors can work together to give life to a book. The website will have a submission process in which writers can submit their books for the purpose of seeking an investment. The writer can also ask the reviewer to help him plot a certain link in the book. The website will also have a review process in which people can give feedback on the book and the author. They can also ask the writer to resubmit the book after making some changes. In addition to this, the website will have an investment process in which investors will have books that have been reviewed and they can invest money in the books. Lastly, there will be our partnered publication houses which will publish the books saving writers from the hustle of printing and publications.

## 2.6 Sustainable Development Goal (SDG)

The main sustainable development goal (SDG) in this project is “Decent work and Economic Growth”. People invest in a lot of things but no one has ever heard of investing in a book. There are a lot of writers whose work has never seen the light of day. By our platform, many such writers will be able to get an investment and from the same platform they will get their books published. This would result in providing growth to the writers as they can get their books published and can earn money without spending a huge sum of money. This would also grow reviewers financially as they will get paid for reviewing books. Moreover, the investors can grow by finding good books for investment which have a great potential for earning money. Moreover, second sustainable development goal of this project is “Quality Education”. When books are published by our platform, they will be quality books which are properly reviewed by our renowned reviewers and then funded by our investors. They can either be educational books, novels, short stories or essays, all will be of good quality only.

## 2.7 Constraints

The main constraints of this project are that the user will not be crossed check the first time they sign up. Anyone can sign up as an investor, writer or a reviewer. They can all be blocked from the website if they are involved in any scam. For example, we cannot check if the investor has enough capital to invest in anything or not. We will come to know about it at the time of investment. Secondly, our system will limit the writers to get their books published from our partnered publishing houses. They will be restricted to publish their books from our partnered publishers only.

## 2.8 Business Opportunity

This project provides a great business opportunity for everyone involved in this. Firstly, the writer can get an investment if his book is good enough to be published. Secondly, the reviewer will get a fee for reviewing the book. Thirdly, the investor who is investing in the book will get his investment with profit once the book is published and is in the market. Moreover, the publishing houses will also earn a good amount by publishing these books. Lastly, everything will be handled by our end, so our website would also have a margin in everything.

## 2.9 Stakeholders Description/ User Characteristics

The first type of user will be a writer who will upload a book for reviewing purposes. He can also get an investment to publish the book if the reviews of the book are positive. The second type of user will be the reviewers who will review the books. They can either be writers with a few or many publications as well as the people who are fond of reading. Thirdly, we will have

investors, who will provide an investment for a book based on its reviews. A writer can also be a reviewer and vice versa. Lastly, we will have an admin who will be responsible for keeping a check and balance on everything. He will block a writer if he uploads a plagiarized books more than three times. He will also notify reviewers about the deadlines and will also be responsible for checking the automated plagiarism reports. Most importantly, he will also be able to block any false/scam user.

### **2.9.1 Stakeholders Summary**

Our system will have four main stakeholders. The first will be the writers who will upload their book on our website for reviewing purposes. The second type will be of the reviewers who will review the books uploaded by the writers. A book will be reviewed by three reviewers to ensure the best quality of reviews. Thirdly, we will have investors who will provide an investment for a book keeping in the view the reviews provided by the reviewers. Lastly, we will have an admin who will keep a track of everything and will perform several tasks which includes checking automated plagiarism reports, reminding about the deadlines and blocking scam users.

### **2.9.2 Key High-Level Goals and Problems of Stakeholders**

The first and foremost goal of our website is to provide writer with an investor so that his book can be published. The high-level goal on the other hand is by categorizing the reviewers based on the reviews they have given. Moreover, checking the plagiarism of the book automatically and watermarking every page of the book will be a tough challenge.

The problems that the stakeholders might face are:

If a writer uploads a book but is not able to get a reviewer to review his book, or, a book is reviewed successfully and it cannot find an investor. Moreover, there can be scam writers, reviewers or investors on our website which can only be blocked after at least one experience.

To conclude this chapter, the main problem is to provide writers with investors who do not have enough money to publish their work. This process will be done by reviewing the book uploaded by the writer, and then if the review is favorable, the book might get an investment from our investors. There are many sub-problems which this project will cater.

## Chapter 3: Literature Review / Related Work

This chapter examines some of the previous work done related to our project and explores their methods. There are a number of sites that provide paid and unpaid reviews for books. There are many corporations working on the print-on-demand facility. Also, there are many platforms which provide an investment plan for books but there is no website that reviews the books, then finds an investor and then publishes it all at one place.

### 3.1 Definitions, Acronyms, and Abbreviations

**Haiku:** Haiku is a type of short form poetry

**POD:** print on demand

### 3.2 Detailed Literature Review

Following is the detailed literature review relevant to our project.

#### 3.2.1 Related Research Work 1

Booklist is an American publication library that provides reviews of books for people of all ages. Their main reviewers are educators, book lovers and book sellers. The audience can get a printed magazine or in online form. Booklist is published 22 times per year, and reviews over 7,500 titles annually.

##### 3.2.1.1 Summary of the research item

Booklist was established in 1905 to meet the need of issuing a buying list of rescind books with brief notes structured to assist librarians. It started with an annual fee of 50 cents. With the passage of time the reviews of the booklist were limited to 150 words. Reviews were handwritten, edited and typed up for the printer in 1950-60. As for today, booklist reviews are said to be "the haiku of book reviewing" [1]. They basically include a brief summary and the most successful element of style. Commonly, the reviews fall between 175 and 225 words.

Booklist believes that their audience is school and public library workers so they get their books reviewed by people who are familiar with books as well as libraries. They do not require a degree in book reviewing to perform this task. They just let the users (reviewers) fill out an application form. They have been performing this task for over a century now and are one of the best books reviewing platforms on the planet.

##### 3.2.1.2 Critical analysis of the research item

The strength of Booklist is that they have been performing the task of reviewing books for over a century and they are one of the best in their field. They have a huge community of reviewers who review over 7,500 books per year. Moreover, they publish almost 22 times a year.

The weakness of Booklist is that the reviewers who are reviewing books only for a time pass and have no interest in reading or reviewing books and are not even able to review books. This can result in false reviews which can ultimately lead to unfairness.

##### 3.2.1.3 Relationship to the proposed research work

Our project consists of two main parts. The first part i.e., the review of books is related directly with the above-mentioned website. We are doing almost the same thing, but our smart system will rate reviewers on the basis of the number and quality of reviews given. In this way, we will have different categories of reviewers based on their past experience.

### 3.2.2 Related Research Work 2

Lightning Source is a printing and print-on-demand service which works under the Ingram Content Group. They cater the needs of independent publishers and authors. IngramSpark has registered more than 7 million books, with 4,000 new books added each day [2]. They also have a global distribution to over 40,000 retailers and libraries [3].

#### 3.2.2.1 Summary of the research item

Lightning Source started in 1996 as Lightning Print Inc. They were based in the United Kingdom with operations in France, Melbourne and Australia. IngramSpark is operated by Lightning Source and has registered more than 7 million books since its launch in 2013. They basically provide print-on-demand service for independent authors and publishers. It can be a single copy of a book or large orders depending on the demand.

They have also implemented these Espresso Book Machines which is a compact version of a book printing machine which can be installed in book shops [4]. It can print, bind and cut books on demand while the customer waits. They have different packages depending on the demand of the customers.

#### 3.2.2.2 Critical analysis of the research item

The best thing about Lightning Source is that they can print-on-demand. The only thing required is a manuscript and a few ideas. From textbooks to graphic novels to children's books, any genre can be published with the print-on-demand technology. Moreover, with the pod service a publisher can never go out of stock, he can always re-order the stock again and again.

The weakness of this system is that when printing in less quantities the cost is much higher. The distributor has to sell it for higher prices to meet his profits. Secondly, the books using print-on-demand service are not returnable and should only be handled by reputable distributors [5].

#### 3.2.2.3 Relationship to the proposed research work

The publishing part of our project is linked with the given related work. The given related work is performing on-demand-printing for books from independent authors and other publishing houses. Our project is also providing the publishing facility with the modification that the book will first be reviewed and if it is worthy to be published, we will seek an investment from our investors and then get the book published by our partnered publishers.

## 3.3 Literature Review Summary Table

**Table 1: Summary Table**  
*The table of summary of literature review.*

No	Name, reference	Inventor	Year	Input	Output	Description
.						

1.	Booklist, <a href="https://www.booklistonline.com/writing-for-booklist">https://www.booklistonline.com/writing-for-booklist</a>	American Library Association	1905	Books	Reviews	The book uploaded will be reviewed by our reviewers in the said time.
2.	IngramSpark, <a href="https://www.ingramspark.com">https://www.ingramspark.com</a>	Lightning Source	1996	Soft copy of Books	Printed hard form of book	The book will be printed on demand depending upon the requirement of client

### 3.4 Conclusion

To conclude, Booklist has been providing book reviews for over 100 years. They have used librarians and other people associated with books for this purpose. However, our reviewers can be the people who are interested in reading or the people who love to write. Our writers can also be the reviewers. One part of our project is almost the same as what Booklist is doing. But Booklist is only providing the reviews, on the other hand, our project will also provide writers with investors and publishing services. Furthermore, IngramSpark has been providing an on-demand-printing facility for quite a while now. They even have these remote printing machines for individual orders. They can publish a single copy of a book or of large orders depending upon the demand of their customers. They basically save authors and publishing houses from the hassle of book printing and do it for a fee.

## Chapter 4: Software Requirement Specifications

There are many requirements that our software must fulfill such as a writer should be able to upload a book, a writer should be able to view pending books, a reviewer should be able to give review to a book, an investor should be able to see books with their complete review, and an investor should be able to invest in books, etc. Moreover, there are many features of our projects such as automatic plagiarism checker etc. All the requirements, features and design are given in the sub-sections below.

### 4.1 List of Features

The major features of our website are given below.

- Our web-based system will check plagiarism of the book each time a book is published and it will show percentage plagiarism.
- The system will watermark each page of the book that gets uploaded.
- The system will recommend the reviewers to admin for a certain book based on genre, type and availability using artificial intelligence.
- Our system will automatically send an email to writer, investor and publisher when the deal is done.

### 4.1 Functional Requirements

The functional requirements of system are given below.

- The system shall allow writer to create an account with a valid email address and maintain it.
- The system shall allow writer to login into already registered account using valid email.
- The system shall allow writer to edit his profile and update his personal information.
- The system shall allow writer to change their password.
- The system shall allow writer to upload a book in the required format given in the instruction section.
- The system shall allow writer to resubmit a book if it is conditionally rejected by a reviewer.
- The system shall allow writer to select type of reviewer from Beginner, Intermediate, Professional and Expert.
- The system shall allow writer to view the list of books along with their status which he uploaded but have not yet found any investor.
- The system shall allow writer to view comments made by reviewers on uploaded book.
- The system shall allow writer to rate reviewers out of 10 based on the explanation provided by the reviewers.
- The system shall allow writer to switch to reviewer.
- The system shall allow reviewer to create an account with a valid email address and maintain it.
- The system shall allow reviewer to login into already registered account using valid email.
- The system shall allow reviewer to make changes like password, publications and contact details.
- The system shall allow reviewers to change their password.
- The system shall allow reviewer to accept an assignment through email sent by admin.
- The system shall allow reviewer to reject an assignment through email sent by admin.

- The system shall allow a reviewer to download the watermarked version of an assigned book.
- The system shall allow reviewer to give review about an assigned book by filling form.
- The system shall allow reviewer to approve the book for investment.
- The system shall allow reviewer to approve the book with minor changes.
- The system shall allow reviewer to approve the book with major changes.
- The system shall allow reviewer to reject the book for investment.
- The system shall allow reviewer to switch to writer.
- The system shall allow investor to create an account with a valid email address and maintain it.
- The system shall allow investor to login into already registered account using valid email.
- The system shall allow investor to make changes like password and contact details.
- The system shall allow investors to change their password.
- The system shall allow investor to view the complete review of an approved book.
- The system shall allow investor to invest in book.
- The system shall allow investor to select the amount of copies of a certain book he wants to invest in.
- The system shall allow investor to see list of books he has invested in.
- The system shall allow investor to select the publisher from whom he wants to get the book published.
- The system shall allow investor to rate reviewers.
- The system shall allow admin to login into already registered account using valid email.
- The system shall allow admin to change their password.
- The system shall allow admin to view comments made by reviewers for admin for a certain book.
- The system shall allow admin to reject a newly uploaded book.
- The system shall allow admin to block a writer.
- The system shall allow admin to block an investor.
- The system shall allow admin to view the recommended reviewers.
- The system shall allow admin to send an email to reviewers.
- The system shall allow admin to view the reviewers whose deadlines are near and notify them.
- The system shall allow admin to block a reviewer.
- The system shall allow admin to extend the deadline of a pending assignment of a reviewer.

## 4.2 Quality Attributes

The quality attributes of system define the measures on which the performance of software is judged. It is a checklist of the things which helps in assessing the quality of a software. The quality attributes of our Book Reviewing and Investment platform are as follows:

- The system shall be easy to use.
- The system shall be functionally correct and complete and free from errors.
- The system shall be testable and will be made in such a way that finding bugs will be easy.
- The system will be easily maintainable as it will be constructed in modules and will be easily modifiable.

- The system shall be reusable and will be made in modules which can be reused.
- The system shall be made available and it will be designed in a way that every request gets handled.

### 4.3 Non-Functional Requirements

The non-functional requirements of our system are provided below.

- The website will be easy to use and will be made user friendly and interactive. It will guide users in intuitive way.
- The website will be made reusable by developing it in modules so that their code can favor reusability.
- The website will be available all the time and it will perform its functions when they are required.
- The website will be made in such a way that it is extensible and if we want to add new modules such as that of publisher, we can add it.
- At least 100 users shall be able to simultaneously access our website and perform functioning without a performance degrade.

### 4.4 Assumptions

The assumptions we make to develop our system are given in sub-section provided below.

- As we are not background checking an investor, we are assuming that investors have the financial power to finance the investment.
- We are assuming that all the users such as writers, investors and reviewers have basic knowledge of how to use a computer and website.

### 4.5 Hardware and Software Requirements

Hardware and software requirements that need to meet in order for project to get developed and deployed are provided in the sub-sections given below.

#### 4.5.1 Hardware Requirements

- A computer system will be required to develop this website. Core i5 or later processors will be ideal for development of this web-based project.

#### 4.5.2 Software Requirements

Following are the software requirements of our project:

- React.js Framework
- Node.js Framework
- Express.js Framework
- Stable browser: Chrome or Firefox
- An IDE we will use preferably Visual Studio Code
- Mongo DB
- Plagiarism API
- Python
- Mongo DB compass

## 4.6 Use Cases

Following are the use cases of this project:

### 4.6.1 Login

<b>Name</b>	Login		
<b>Actors</b>	Admin, Writer, Reviewer, Investor		
<b>Summary</b>	The user shall provide their email and password and select their role on the login form and after successful verification, redirect the user to the relevant home page.		
<b>Pre-Conditions</b>	The user must be in the database records. The user must not already be logged in.		
<b>Post-Conditions</b>	The user's session is successfully established and shall be redirected to the respective home page.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The user opens the login page.	2	The login page is displayed asking for email and password.
3	The user enters valid email, password and selects their role.	4	The system verifies the email and password, establishes a session for the user and redirects the user to the relevant home page.
<b>Alternative Flow</b>			
3	The user enters invalid email or password.	4-A	The system responds with an error message: <i>Incorrect email or password entered.</i>

### 4.6.2 Sign-Up

<b>Name</b>	Sign-Up		
<b>Actors</b>	Writer, Investor		
<b>Summary</b>	The user shall provide their valid email, name, CNIC, phone no. and password and select their role on the sign-up form.		
<b>Pre-Conditions</b>	The system should be working properly. The user must not already be signed up.		
<b>Post-Conditions</b>	The user's account would be created and the information will be recorded in database.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	

1	The user opens the sign-up page.	2	The sign-up page is displayed asking for user's information.
3	The user enters valid email, name, CNIC, phone no., password, and select their role from drop down.	4	The system verifies the details and user account would be created and will be saved in database.
<b>Alternative Flow</b>			
3	The user enters invalid details.	4-A	The system responds with an error message: <i>Invalid details</i>

#### 4.6.3 Sign-Up for Reviewer

Name	Sign-Up for Reviewer		
Actors	Reviewer		
Summary	The reviewer shall provide their valid email, name, CNIC, phone no., password and they can provide their publications on the sign-up form.		
Pre-Conditions	The system should be working properly. The reviewer must not already be signed up.		
Post-Conditions	The reviewer's account would be created and the information will be recorded in database.		
Special Requirements	None		
<b>Basic Flow</b>			
Actor Action		System Response	
1	The reviewer opens the sign-up page.	2	The sign-up page is displayed asking for user's information.
3	The reviewer enters valid email, name, CNIC, phone no., password, select reviewer from drop down and can enter their publications.	4	The system verifies the details and reviewer's account would be created and will be saved in database.
<b>Alternative Flow</b>			
3	The user enters invalid details.	4-A	The system responds with an error message: <i>Invalid details</i>

#### 4.6.4 Reset password

Name	Reset password		
Actors	Writer, Reviewer, Investor		
Summary	The user shall provide verification code sent on email and their new password and confirm their new password.		
Pre-Conditions	The system should be working properly. The user must be in the database record.		
Post-Conditions	The user's account password would be changed to the new password and the information will be recorded in database.		

<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The user opens the password reset page by clicking on the forgot password button.	2	The password reset page is displayed asking for user's email.
3	The user enters their registered email.	4	A verification code will be sent to user's email account.
5	The user will enter the verification code received in email	6	System verifies the verification code and asks the user for new password.
7	The user will enter the new password and confirm their new password.	8	The password will be changed and prompt will be shown: Password changed successfully
<b>Alternative Flow</b>			
3	The user enters an unregistered email.	4-A	The system responds with an error message: <i>The given email is not registered</i>
5	The user enters the incorrect verification code.	6-A	Message will be displayed: <i>Incorrect Code, Enter the code again</i>

#### 4.6.5 Change password

<b>Name</b>	Change password		
<b>Actors</b>	Admin, Writer, Reviewer, Investor		
<b>Summary</b>	The user shall provide their old, new password and confirm their new password.		
<b>Pre-Conditions</b>	The user must be logged in. The system should be working properly.		
<b>Post-Conditions</b>	The user's account password would be changed to the new password and the information will be recorded in database.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The user opens the change password page by clicking on the change password button.	2	The change password page is displayed.
3	The user enters their old password.	4	System verifies the old password and asks the user for new password.
5	The user will enter new password and confirm their new password.	6	The password will be changed and prompt will be shown: <i>Password changed successfully</i>
<b>Alternative Flow</b>			
3	The user enters incorrect password.	4-	The system responds with an error

		A	message: <i>Incorrect Password</i>
--	--	---	------------------------------------

#### 4.6.6 Edit Profile

<b>Name</b>	Edit Profile		
<b>Actors</b>	Writer, Reviewer, Investor		
<b>Summary</b>	The user will be allowed to edit their profile.		
<b>Pre-Conditions</b>	The user must be logged in. The system should be working properly.		
<b>Post-Conditions</b>	The user's account information would be changed to new information and will be stored in database.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The user opens the edit profile page by clicking on the edit profile button.	2	The edit profile page is displayed.
3	The user will edit the desired section and click the submit button.	4	The profile will be edited.
<b>No Alternative Flow</b>			

#### 4.6.7 Upload Book

<b>Name</b>	Upload Book		
<b>Actors</b>	Writer		
<b>Summary</b>	The system shall allow writer to upload a book in the required format given in the instruction section select type of the reviewer.		
<b>Pre-Conditions</b>	The writer must be logged in. The writer is on the upload book page. The system should be working properly.		
<b>Post-Conditions</b>	The book will be uploaded to the system and a notification will be sent to the admin.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The writer selects the type of reviewer.	2	Type of reviewer is selected.
3	The writer selects upload a book.	4	Systems shows file explorer.
5	The writer selects the book from their computer in pdf form and selects upload.	6	The book is uploaded to the system and a message will be displayed: <i>Book uploaded successfully</i>

<b>Alternative Flow</b>			
5	The writer uploads a book in wrong format.	6-A	The system responds with an error message: <i>Invalid Format</i>

#### 4.6.8 Resubmit Book

<b>Name</b>	Resubmit Book		
<b>Actors</b>	Writer		
<b>Summary</b>	The system shall allow writer to resubmit a book.		
<b>Pre-Conditions</b>	The writer must be logged in. The writer is on the list of books pages. The book is returned by the reviewer for changes.		
<b>Post-Conditions</b>	The book will be uploaded to the system and a notification will be sent to the admin.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The writer selects resubmit button against the book he wants to resubmit.	2	Systems shows file explorer.
3	The writer selects the book from their computer in pdf form and selects upload.	4	The book is uploaded to the system and a message will be displayed: <i>Book uploaded successfully</i>
<b>Alternative Flow</b>			
3	The writer uploads a book in wrong format.	4-A	The system responds with an error message: <i>Invalid Format</i>

#### 4.6.9 View List of Pending Books

<b>Name</b>	View list of pending books		
<b>Actors</b>	Writer		
<b>Summary</b>	The system shall allow writer to view the list of pending books.		
<b>Pre-Conditions</b>	Writer is logged in and is on the home page of writer.		
<b>Post-Conditions</b>	None		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	Writer selects pending books button.	2	List of pending books is displayed.

<b>No Alternative Flow</b>
----------------------------

#### **4.6.10 View Comments of Reviewer**

<b>Name</b>	View Comments of Reviewer		
<b>Actors</b>	Writer		
<b>Summary</b>	The system shall allow writer to view comments made by reviewers on uploaded book.		
<b>Pre-Conditions</b>	Writer is logged in and is on the list of pending books. The book has been reviewed by the reviewer.		
<b>Post-Conditions</b>	None		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	Writer selects the view comments buttons against the reviewed book.	2	Comments by the reviewer are displayed.
<b>No Alternative Flow</b>			

#### **4.6.11 Rate Reviewer**

<b>Name</b>	Rate reviewer		
<b>Actors</b>	Writer		
<b>Summary</b>	The system shall allow writer to rate reviewers out of 10 based on the explanation provided by the reviewers.		
<b>Pre-Conditions</b>	Writer is logged in and is on the list of pending books. The book has been reviewed by the reviewer.		
<b>Post-Conditions</b>	Rating is stored against the reviewer.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	Writer selects rate reviewer.	2	A list of reviewers is showed.
3	Writer selects the reviewer he wants rate.	4	Option for rating is displayed
5	Writer selects the rating.	6	A message is displayed: <i>Rated Successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.12 Download an Assigned Book

<b>Name</b>	Download an Assigned Book		
<b>Actors</b>	Reviewer		
<b>Summary</b>	The system shall allow the reviewer to download an assigned book.		
<b>Pre-Conditions</b>	The reviewer is logged in and is on the pending assignments page. The system should be working properly.		
<b>Post-Conditions</b>	The book will be downloaded to reviewer's system.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The user selects download from the list of assigned books.	2	Systems shows file explorer to select the directory to download the book.
3	The user selects the book from their computer in pdf form and selects upload.	4	The book is downloaded to the system and a message will be displayed: <i>Book downloaded successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.13 Fill the Review Form

<b>Name</b>	Fill the review form		
<b>Actors</b>	Reviewer		
<b>Summary</b>	The system shall allow reviewer to give review about an assigned book by filling form.		
<b>Pre-Conditions</b>	The reviewer is logged in and is on the assignments page. The system should be working properly.		
<b>Post-Conditions</b>	The book review is uploaded to the system and sent to the admin and reviewer.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The reviewer selects review book from the list of books	2	A review form is displayed.
3	Reviewer fills the form.	4	Filled form is displayed on the screen.
5	Reviewer makes the decision for the book i.e., accept, reject.	6	The review is uploaded to the system and a message is displayed: <i>Review uploaded successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.14 Invest in a Book

<b>Name</b>	Invest in a book		
<b>Actors</b>	Investor		
<b>Summary</b>	The system shall allow investor to select the number of copies of a certain book he wants to invest in and the publisher he wants to get it published from.		
<b>Pre-Conditions</b>	Investor is logged in and a list of approved books is displayed. The system should be working properly.		
<b>Post-Conditions</b>	An email will be sent to writer, investor and publisher and the status of the book will be changed to investor found.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The Investor selects a book to invest from list of approved books.	2	An investment form is displayed.
3	Investor selects the publisher.	4	The publisher is selected.
5	Investor selects the number of copies to print and submits.	6	A message will be displayed: <i>Investment Successful</i>
<b>No Alternative Flow</b>			

#### 4.6.15 View Complete Review

<b>Name</b>	View complete review		
<b>Actors</b>	Investor		
<b>Summary</b>	The system shall allow investor to view the complete review of an approved book.		
<b>Pre-Conditions</b>	Investor is logged in and a list of approved books is displayed.		
<b>Post-Conditions</b>	None		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The Investor selects open review from list of approved books.	2	Complete review will be displayed.
<b>No Alternative Flow</b>			

#### 4.6.16 View Invested Books

<b>Name</b>	View invested books		
-------------	---------------------	--	--

<b>Actors</b>	Investor		
<b>Summary</b>	The system shall allow investor to view list of books in which he has invested.		
<b>Pre-Conditions</b>	Investor is logged in. The system should be working properly.		
<b>Post-Conditions</b>	None		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	The Investor selects invested books button.	2	List of invested books is displayed.
<b>No Alternative Flow</b>			

#### 4.6.17 Rate Reviewer

<b>Name</b>	Rate reviewer		
<b>Actors</b>	Investor		
<b>Summary</b>	The system shall allow investor to rate reviewers out of 10 based on the explanation provided by the reviewers.		
<b>Pre-Conditions</b>	Investor is logged in and is on the list of invested books. The system should be working properly.		
<b>Post-Conditions</b>	Rating is stored against the reviewer.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	Investor selects rate reviewer.	2	A list of reviewers is showed.
3	Investor selects the reviewer he wants rate.	4	Option for rating is displayed
5	Investor selects the rating.	6	A message is displayed: <i>Rated Successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.18 View Comments by Reviewer

<b>Name</b>	View comments by reviewer		
<b>Actors</b>	Admin		
<b>Summary</b>	The system shall allow admin to view comments made by reviewers for admin for a certain book.		

<b>Pre-Conditions</b>	The admin must be logged in and is on the homepage of the admin. The system should be working properly.		
<b>Post-Conditions</b>	None		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	From manage books section admin selects reviewed option.	2	Reviewed books are displayed.
3	Admin selects view comments for the respective book.	4	Comments by the reviewer are displayed.
<b>No Alternative Flow</b>			

#### 4.6.19 Reject a Book

<b>Name</b>	Reject a book		
<b>Actors</b>	Admin		
<b>Summary</b>	The system shall allow admin to reject a newly uploaded book.		
<b>Pre-Conditions</b>	The admin must be logged in and is on the homepage of the admin. The system should be working properly.		
<b>Post-Conditions</b>	Book is rejected		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	From manage books section admin selects newly uploaded option.	2	Newly uploaded books are displayed along with their plagiarism.
3	Admin selects reject book for the respective book.	4	Message is displayed: <i>Book Rejected Successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.20 Block a User

<b>Name</b>	Block a user		
<b>Actors</b>	Admin		
<b>Summary</b>	The system shall allow admin to block a user.		
<b>Pre-Conditions</b>	The admin must be logged in and is on the homepage of the admin. The system should be working properly.		
<b>Post-Conditions</b>	The user is blocked and the record is saved in the database.		

<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	Form manage users section admin selects the type of user he wants to block.	2	A list of users will be displayed.
3	Admin can search the user and press block button for the respective user.	4	A message is displayed: <i>User Blocked Successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.21 Assign Book to Reviewers

<b>Name</b>	Assign book to reviews		
<b>Actors</b>	Admin		
<b>Summary</b>	Admin will assign books to reviews for reviews.		
<b>Pre-Conditions</b>	The admin must be logged in and is on the homepage of the admin. The system should be working properly.		
<b>Post-Conditions</b>	An email will be sent to the reviewers.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	From manage books section admin selects newly uploaded option.	2	Newly uploaded books are displayed along with their plagiarism.
3	Admin press assign reviewers for the respective book.	4	A list of reviewers is shown.
5	Admin selects the reviewers and selects assign.	6	A message is displayed: <i>Email Sent</i>
<b>No Alternative Flow</b>			

#### 4.6.22 Notify the Reviewer

<b>Name</b>	Notify the reviewer		
<b>Actors</b>	Admin		
<b>Summary</b>	The system shall allow admin to view the reviewers whose deadlines are near and notify them.		
<b>Pre-Conditions</b>	The admin must be logged in and is on the homepage of the admin. The system should be working properly.		
<b>Post-Conditions</b>	A reminder email is sent to reviewer.		
<b>Special</b>	None		

<b>Requirements</b>			
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	From manage books section admin selects under review option.	2	List of under review books is shown.
3	Admin selects send reminder against the respective book.	4	A message is displayed: <i>Reminder sent successfully</i>
<b>No Alternative Flow</b>			

#### 4.6.23 Extend Deadline

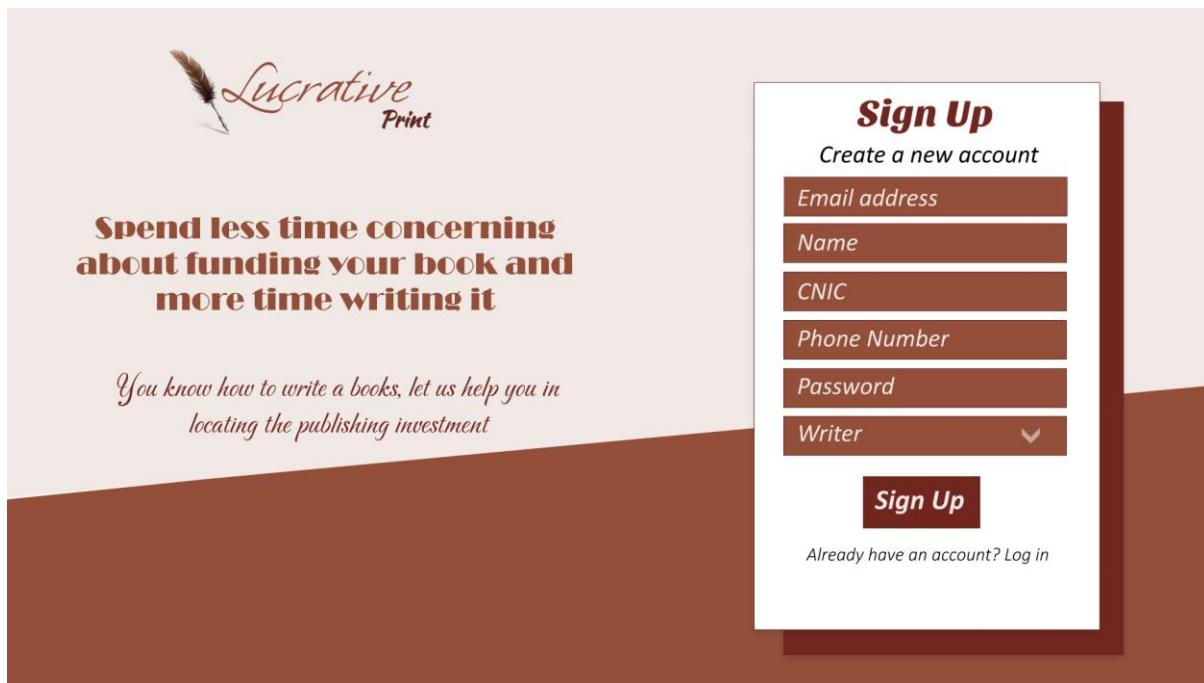
<b>Name</b>	Extend deadline		
<b>Actors</b>	Admin		
<b>Summary</b>	The system shall allow admin to extend the deadline of a pending assignment of a reviewer.		
<b>Pre-Conditions</b>	The admin must be logged in and is on the homepage of the admin.		
<b>Post-Conditions</b>	Deadline for the book is extended.		
<b>Special Requirements</b>	None		
<b>Basic Flow</b>			
<b>Actor Action</b>		<b>System Response</b>	
1	From manage books section admin selects under review option.	2	List of under review books is shown.
3	Admin selects the new date for the respective book.	4	A message is displayed: <i>Deadline Extended Successfully</i>
<b>No Alternative Flow</b>			

## 4.7 Graphical User Interface

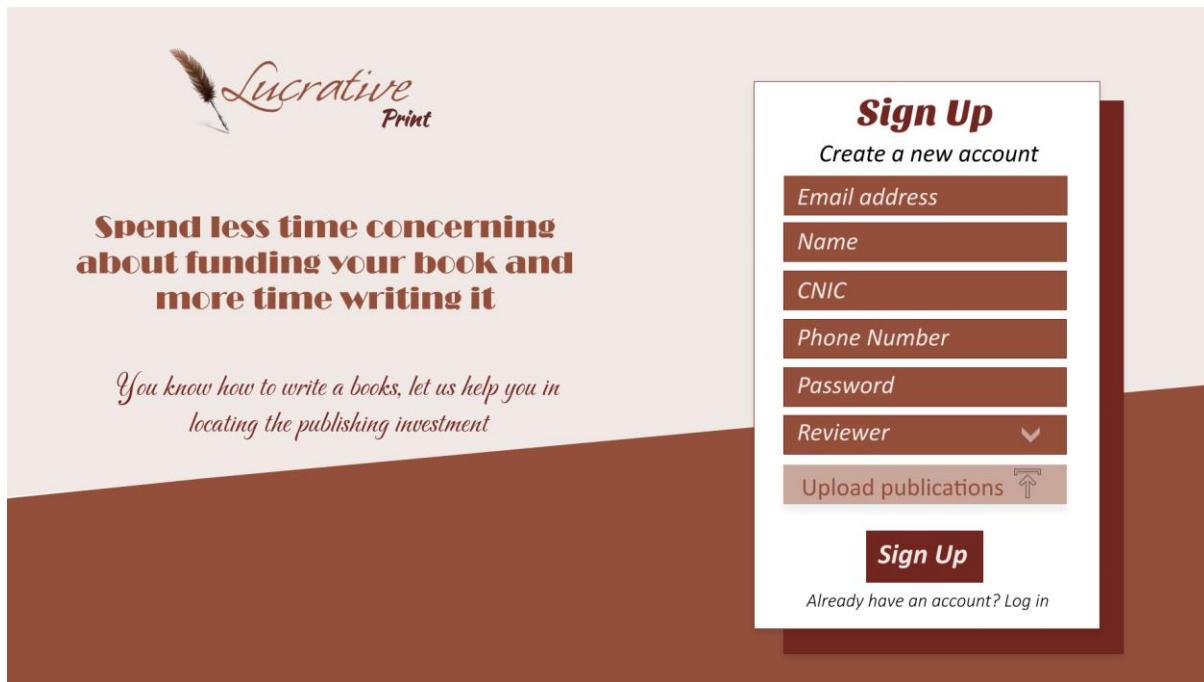
GUI dumps of each screen, with reference to the users and functionality are given below.



**Figure 1: Login Page**  
The users can login to system by providing email and password.



**Figure 2: Signup Page**  
The users can sign up to system by providing details.



**Figure 3: Signup Page for Reviewers**  
The reviewer can sign up to system by providing details.

The screenshot shows the Writer homepage for the platform "Lucrative Print". At the top right, there is a profile icon for "Zaviyaar Bin Irfan" and a toggle switch between "Reviewer" and "Writer" modes, which is currently set to "Writer". Below this, the title "Select your Reviewer Type" is displayed. Three options are shown in boxes numbered 1, 2, and 3:

- 1 Beginner**: US \$30 per book. Description: "Beginner Reviewer", "Detail Review", "Low investment rate". Status: "Selected".
- 2 Intermediate**: US \$65 per book. Description: "5 Stars+ Reviewer", "Atleast 1 publication", "Detailed Review", "Good investment rate". Status: "Select".
- 3 Professional**: US \$100 per book. Description: "8 Stars+ Reviewer", "Atleast 3 publications", "Detailed Review", "Excellent investment rate". Status: "Select".

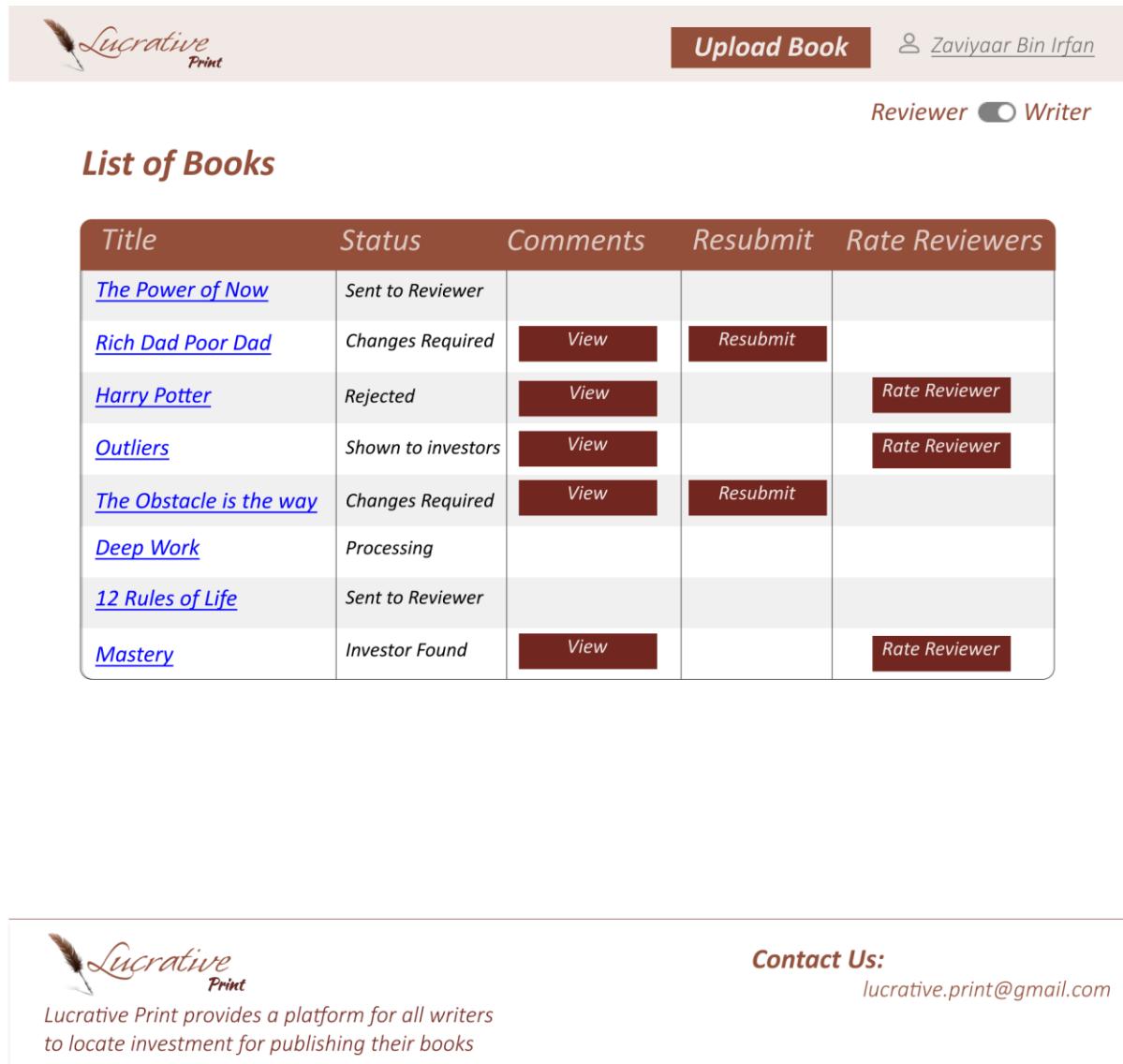
Below this section is a "Upload Your Book" form with fields for "Enter the title book", "Enter Description of book", and "Select your File" (with a file icon). A note states: "Format should be according to given Sample". A "Submit" button is at the bottom of the form.

---

**Contact Us:** lucrative.print@gmail.com

**Lucrative Print** provides a platform for all writers to locate investment for publishing their books

**Figure 4: Writer Homepage**  
The writer can upload book by selecting reviewer and can also see pending books.



The screenshot shows the Lucrative Print platform interface. At the top, there is a logo with a feather icon and the text "Lucrative Print". To the right of the logo are buttons for "Upload Book" and "Reviewer" (which is currently selected, indicated by a checked toggle switch). Below these buttons is the text "Zaviyaar Bin Irfan". Further down, there is a "Writer" button with an unchecked toggle switch.

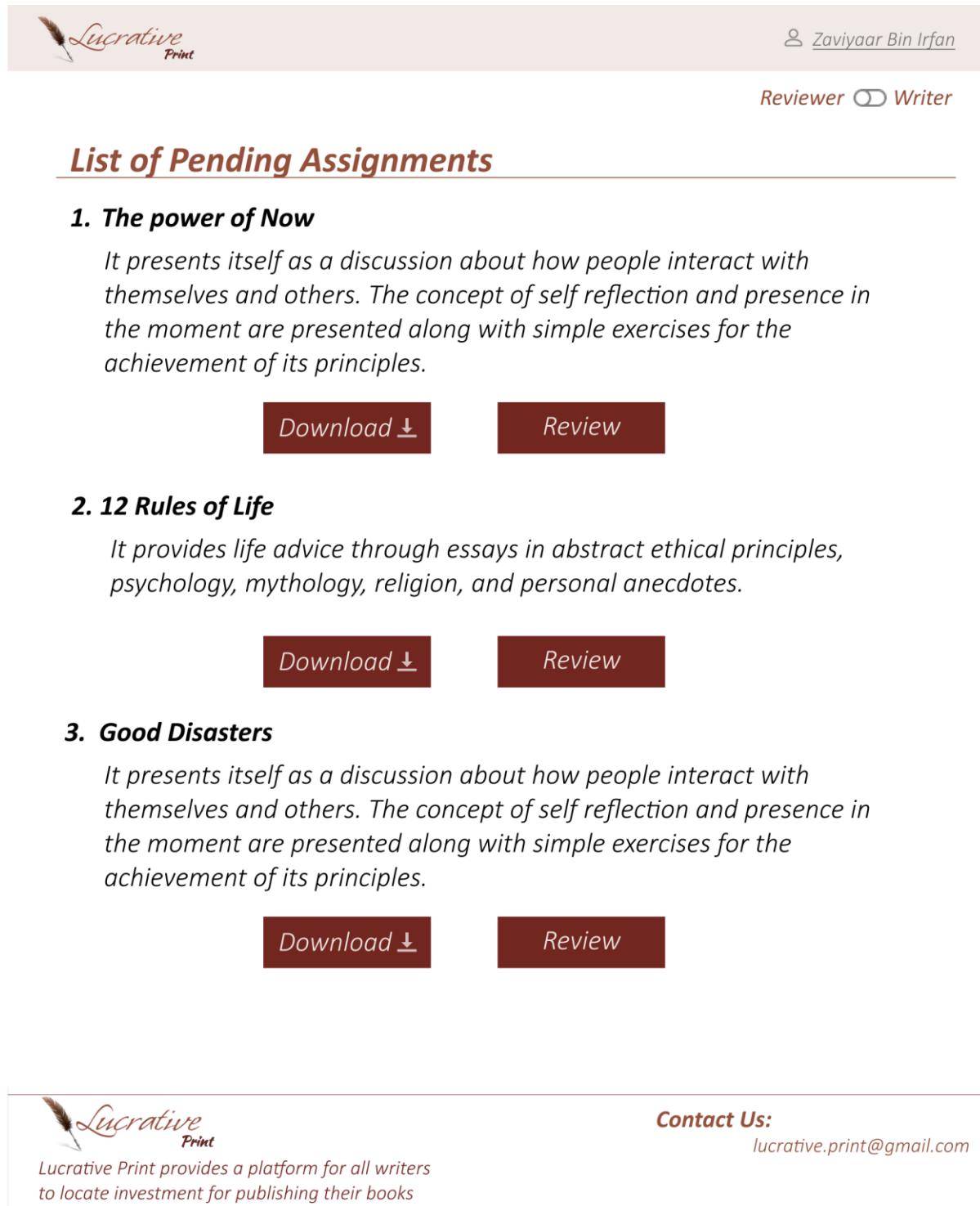
**List of Books**

Title	Status	Comments	Resubmit	Rate Reviewers
<a href="#">The Power of Now</a>	Sent to Reviewer			
<a href="#">Rich Dad Poor Dad</a>	Changes Required	<a href="#">View</a>	<a href="#">Resubmit</a>	
<a href="#">Harry Potter</a>	Rejected	<a href="#">View</a>		<a href="#">Rate Reviewer</a>
<a href="#">Outliers</a>	Shown to investors	<a href="#">View</a>		<a href="#">Rate Reviewer</a>
<a href="#">The Obstacle is the way</a>	Changes Required	<a href="#">View</a>	<a href="#">Resubmit</a>	
<a href="#">Deep Work</a>	Processing			
<a href="#">12 Rules of Life</a>	Sent to Reviewer			
<a href="#">Mastery</a>	Investor Found	<a href="#">View</a>		<a href="#">Rate Reviewer</a>

**Contact Us:**  
lucrative.print@gmail.com

Lucrative Print provides a platform for all writers to locate investment for publishing their books

**Figure 5: Writer's pending books**  
*The writer can view comments by reviewer, resubmit a book and can rate reviewers.*



The screenshot shows the homepage of a website for 'Lucrative Print'. At the top left is the logo 'Lucrative Print' with a quill pen icon. At the top right is the user profile 'Zaviyaar Bin Irfan' with a person icon. Below the header is a navigation bar with the text 'Reviewer  Writer'. The main content area features a section titled 'List of Pending Assignments' with three numbered sections: 1. The power of Now, 2. 12 Rules of Life, and 3. Good Disasters. Each section includes a 'Download' button with a downward arrow icon and a 'Review' button.

**List of Pending Assignments**

**1. The power of Now**

*It presents itself as a discussion about how people interact with themselves and others. The concept of self reflection and presence in the moment are presented along with simple exercises for the achievement of its principles.*

[Download ↓](#)      [Review](#)

**2. 12 Rules of Life**

*It provides life advice through essays in abstract ethical principles, psychology, mythology, religion, and personal anecdotes.*

[Download ↓](#)      [Review](#)

**3. Good Disasters**

*It presents itself as a discussion about how people interact with themselves and others. The concept of self reflection and presence in the moment are presented along with simple exercises for the achievement of its principles.*

[Download ↓](#)      [Review](#)

**Contact Us:**  
[lucrative.print@gmail.com](mailto:lucrative.print@gmail.com)

*Lucrative Print provides a platform for all writers to locate investment for publishing their books*

**Figure 6: Reviewer's Homepage**  
The reviewer can download a book and can opt to give review.



Zaviyaar Bin Irfan

Reviewer  Writer

## Review

**What is the most important part of book?**

Enter Text

**What is type of book?**

Story-Based  
 Facts-Based

**How you rate the delivery style of book?**

5 ▾

**How you rate the content of book?**

1 ▾

**How you rate the writing quality of book?**

3 ▾

**Comments for admin:**

Detailed Comments for admin

**Comments for writer:**

Detailed Comments for writer

**Submit**

### Decision on book

Book is correctly written and has a great potential

**Accept**

Book has grammatical errors. \*Change has to be provided in comments for writers.

**Minor change**

Some part of book can get better. \*Change has to be suggested in comments for writers.

**Major change**

Book is very poorly written. \*Explained reasoning is provided in comments for writers.

**Reject**

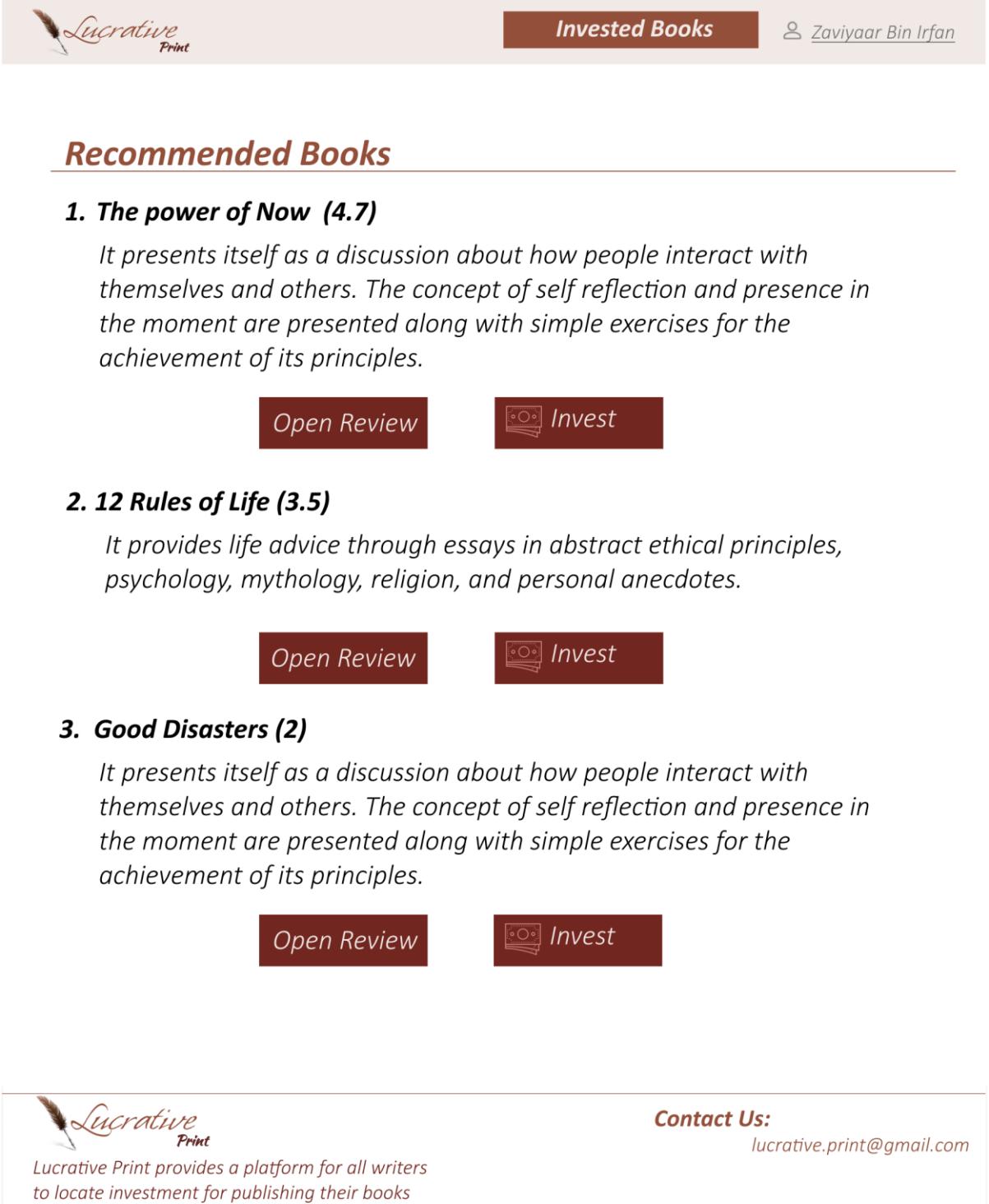


Lucrative Print provides a platform for all writers to locate investment for publishing their books

**Contact Us:**

lucrative.print@gmail.com

**Figure 7: Review page**  
*The reviewer can fill review form and then can make a decision on it.*



The screenshot shows the homepage of Lucrative Print. At the top left is the logo 'Lucrative Print' with a quill pen icon. At the top right is the user profile 'Invested Books' and 'Zaviyaar Bin Irfan'. Below the header is a section titled 'Recommended Books' with three items listed:

- 1. The power of Now (4.7)**  
*It presents itself as a discussion about how people interact with themselves and others. The concept of self reflection and presence in the moment are presented along with simple exercises for the achievement of its principles.*

 Open Review
 Invest
- 2. 12 Rules of Life (3.5)**  
*It provides life advice through essays in abstract ethical principles, psychology, mythology, religion, and personal anecdotes.*

 Open Review
 Invest
- 3. Good Disasters (2)**  
*It presents itself as a discussion about how people interact with themselves and others. The concept of self reflection and presence in the moment are presented along with simple exercises for the achievement of its principles.*

 Open Review
 Invest

At the bottom left is the 'Lucrative Print' logo. At the bottom right is the contact information 'Contact Us: lucrative.print@gmail.com'.

**Figure 8: Investor's Homepage**  
*The investor can view complete review and opt to invest in book.*

**Invested Books**

**Select your Publisher**

**1** **Ilmi Publishers**

- US\$ 1 for each page for books over 500 pages
- US\$ 1.5 for each page for books under 500 pages
- Fine page

**Selected**

**2** **Sang-meel**

- US\$ 1.25 for each page for books over 400 pages
- US\$ 1.5 for each page for books under 400 pages
- Fine page

**Select**

**3** **Humdard Kutab**

- US\$ 0.75 for each page for books over 700 pages
- US\$ 1.5 for each page for books under 700 pages
- Fine page

**Select**

**Amount of copies:**

100 ▾

**Submit**

**Contact Us:**  
lucrative.print@gmail.com

**Lucrative Print**  
Lucrative Print provides a platform for all writers to locate investment for publishing their books

**Figure 9: Investment Page**  
The investor can invest in a book by selecting publisher and by selecting number of books.

The screenshot shows the Admin's homepage for 'Lucrative Print'. The left sidebar has sections for 'Manage books' (Newly uploaded, Under review, Reviewed) and 'Manage users' (Writers, Reviewers, Investors). The main content area displays a table for books under review:

Title	Plagiarism	Writer	Assign Reviewers	Reject
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>
Deep Work	45%	Cal Newport	<b>Assign</b>	<b>Reject</b>

**Figure 10: Admin's Homepage**  
The admin can assign reviewers to books and also can reject a book.

The screenshot shows the Under-Review books page for 'Lucrative Print'. The left sidebar has sections for 'Manage books' (Newly uploaded, Under review, Reviewed) and 'Manage users' (Writers, Reviewers, Investors). The main content area displays a table for books under review:

Title	Reviewers	Deadline	Notify	Extend
Deep Work	<u>irfan.raja@gmail.com</u> <u>ahmad.khan@gmail.com</u> <u>usama.iqbal@gmail.com</u>	13/10/2022	<b>Notify</b>	<b>Extend</b>
Shallow	<u>irfan123@gmail.com</u> <u>ak_123@gmail.com</u> <u>usama.098@gmail.com</u>	17/10/2022	<b>Notify</b>	<b>Extend</b>

**Figure 11: Under-Review books Page**  
The admin can notify the reviewer and also can extend the deadline of a review.

**Manage books ▲**

- [Newly uploaded](#)
- [Under review](#)
- Reviewed**

**Manage users ▲**

- [Writers](#)
- [Reviewers](#)
- [Investors](#)

Title	Reviewers	Deadline	
Deep Work	<a href="#">irfan.raja@gmail.com</a> <a href="#">ahmad.khan@gmail.com</a> <a href="#">usama.iqbal@gmail.com</a>	13/10/2022	<a href="#">View Comments</a>
Shallow	<a href="#">irfan123@gmail.com</a> <a href="#">ak_123@gmail.com</a> <a href="#">usama.098@gmail.com</a>	17/10/2022	<a href="#">View Comments</a>

**Figure 12: Reviewed books page**  
The admin can view comments added by reviewer for admin.

**Manage books ▲**

- [Newly uploaded](#)
- [Under review](#)
- Reviewed**

**Manage users ▲**

- Writers**
- [Reviewers](#)
- [Investors](#)

Name	Email	Phone No	
Zaviyaar Bin Irfan	<a href="#">zaviyaar128@gmail.com</a>	03123456789	<a href="#">Block</a>
Zaviyaar Bin Irfan	<a href="#">zaviyaar128@gmail.com</a>	03123456789	<a href="#">Block</a>
Zaviyaar Bin Irfan	<a href="#">zaviyaar128@gmail.com</a>	03123456789	<a href="#">Block</a>
Zaviyaar Bin Irfan	<a href="#">zaviyaar128@gmail.com</a>	03123456789	<a href="#">Block</a>
Zaviyaar Bin Irfan	<a href="#">zaviyaar128@gmail.com</a>	03123456789	<a href="#">Block</a>

**Figure 13: Manage Writer page**  
The admin can block a writer.

**Manage books**

- Newly uploaded
- Under review
- Reviewed

**Manage users**

- Writers
- Reviewers
- Investors

Name	Email	Type	Action
Usama Iqbal	usama2232@gmail.com	Beginner	Block
Kamran	kamran.12@gmail.com	Intermediate	Block
Iqbal Khan	iqbal.2@gmail.com	Professional	Block
Kamran	kamran.12@gmail.com	Intermediate	Block

**Figure 14: Manage Reviewer page**  
The admin can block a reviewer.

**Manage books**

- Newly uploaded
- Under review
- Reviewed

**Manage users**

- Writers
- Reviewers
- Investors

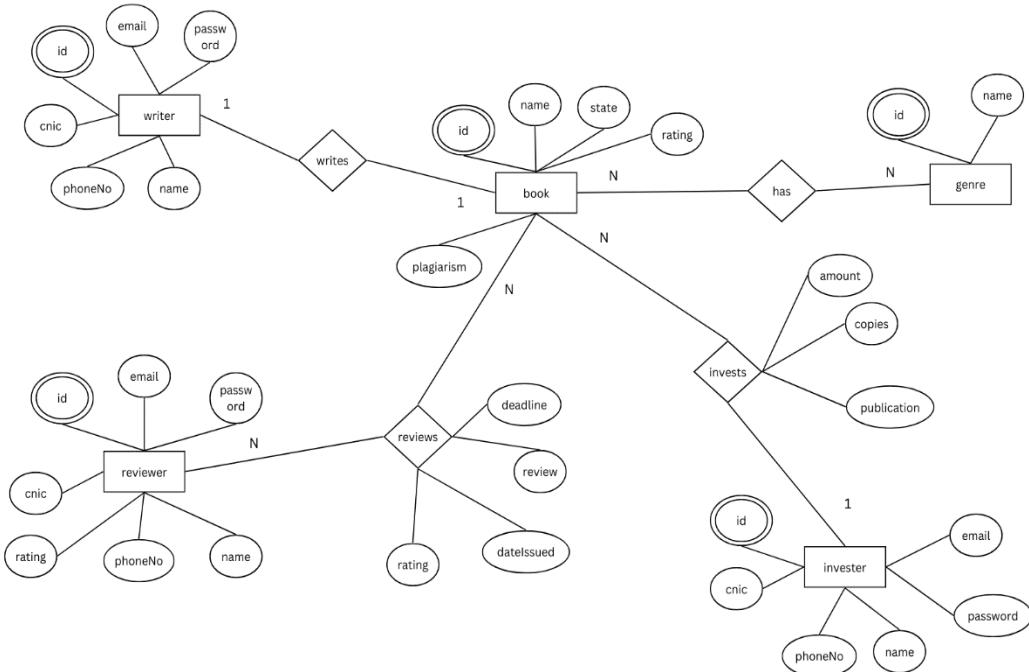
Name	Email	Phone No	Action
Zaviyaar Bin Irfan	zaviyaar128@gmail.com	03123456789	Block
Zaviyaar Bin Irfan	zaviyaar128@gmail.com	03123456789	Block
Zaviyaar Bin Irfan	zaviyaar128@gmail.com	03123456789	Block
Zaviyaar Bin Irfan	zaviyaar128@gmail.com	03123456789	Block
Zaviyaar Bin Irfan	zaviyaar128@gmail.com	03123456789	Block

**Figure 15: Manage investor page**  
The admin can block an investor.

## 4.8 Database Design

The database design of the system is provided in the sub-sections given below.

### 4.8.1 ER Diagram



**Figure 16: ER Diagram**  
*Above is the entity relationship diagram of our database.*

### 4.8.2 Data Dictionary

**Table 2: Data Dictionary Table**  
*The given table gives the details of the tables of database*

Entity	Attribute	Data type	Nullable	Relation to	Relation type	Description
Writer	Id	Int	No			Id of the writer (PK)
	Email	String	No			Email of the writer
	Password	String	No			Password of the writer
	Name	String	No			Name of the writer
	CNIC	String	No			CNIC of the writer
	Phone no.	String	No			Phone number of the writer

Reviewer	Id	Int	No			Id of the Reviewer (PK)
	Email	String	No			
	Password	String	No			
	Name	String	No			
	CNIC	String	No			
	Phone no.	String	No			
	Rating	Float	Yes			
Investor	Id	Int	No			Id of the Investor (PK)
	Email	String	No			
	Password	String	No			
	Name	String	No			
	CNIC	String	No			
	Phone no.	String	No			
Book	Id	Int	No			Id of the book (PK)
	WriterId	Int	No	Writer	1 to 1	
	Name	String	No			
	Plagiarism	Int	Yes			
	Rating	Float	Yes			
	State	Int	No			
Assignments	DateIssued	Date	No			Date the review is assigned to reviewer

	BookId	Int	No	Book	1 to *	Id of the book (FK)
	ReviewerId	Int	No	Reviewer	1 to 1	Id of the reviewer (FK)
	Deadline	Date	No			Deadline for the reviewer
	Rating	Float	Yes			Rating of the book
	Review	File	Yes			Review by reviewer
Investments	Amount	Float	No			Amount invested by investor
	InvestorId	Int	No	Investor	1 to 1	Id of the investor (FK)
	BookId	Int	No	Book	1 to 1	Id of the book (FK)
	Copies	Int	No			The number of books
	Publisher	String	Yes			Name of the publisher
Genre	Id	Int	No			Id of genre (PK)
	Name	String	No			Name of genre
Genres	BookId	Int	No	Book	1 to *	Id of the book (FK)
	GenreId	Int	No	Genre	1 to *	Id of the genre (FK)

## 4.9 Risk Analysis

The business risks that can be encountered during this project are the competitors who will copy this idea and implement it on their own with reduced cost. In addition to that, the technical risks that can be encountered are the vulnerabilities of the software. The sensitive information of the users will be at a risk of cyber-attacks. There can be password thefts. Lastly, there can be service outages due to non-availability of internet or server malfunction.

As a conclusion to this chapter, the software requirement specifications are given. This chapter highlights the important features of this project such as checking automatic plagiarism checker, adding watermark to pages to prevent theft, suggesting reviewers, etc. In addition to this, functional and non-functional requirements of project are also discussed in this chapter. Moreover, assumptions include that investor has enough capital to invest the amount he agreed upon and that every user knows how to use a computer. Moreover, there are GUI dumps of screens present in the system. The use-cases of system are also given and database design is also given. Lastly, risk analysis of system is also discussed in this chapter.

## Chapter 5: High-Level and Low-Level Design

The complete details of high-level and low-level design of the system we are building is provided in this chapter.

### 5.1 System Overview

In the system under construction, users will request login through UI and this request will be transferred to application layer and after that the credential requests will be matched by retrieving details from database. The writer will request upload book thorough user interface whose request will be transferred to application layer and after that this book will be stored in database. Reviewer will request download book through user interface whose request will be transferred to application layer and this book will be downloaded after retrieval from database. Admin will give different requests like blocking a user, etc. through user interface and these requests will be transferred to application layer and which will transfer the request to database layer. Investor will make request to show books through user interface and after that these requests will be transferred to application layer and then this request will be transferred to database layer which will fetch the books and show it to investor.

### 5.2 Design Considerations

The details of issues which need to be addressed or resolved before attempting to devise a complete design solution are provided in sub-sections below.

#### 5.2.1 Assumptions and Dependencies

The following are some assumptions that we are making while designing this system:

- We are assuming that the user has a computer or smart phone for accessing the website.
- User has knowledge of Basic English for using this website.
- For accessing this website, user has an internet connection.
- User has a basic knowledge on how to access general websites.
- All the users have modern browsers

#### 5.2.2 General Constraints

The global limitations or constraints that have a significant impact on the design of the system's software are provided in the sub-sections below.

##### 5.2.2.1 Hardware or software environment

The project assumes that the user has an operating system equivalent or stronger than windows 7 and with a RAM greater or equal to 2GB. The system should also have a stable internet connection for a smooth experience.

##### 5.2.2.2 End-user environment

- The end users of this application will run on feasible browsers like Chrome, Firefox, Safari and Edge.
- The end users should have some basic knowledge on how to use a website.

### 5.2.2.3 Availability or volatility of resources

- This project assumes that the end user will have all the resources to run the application.
- User must have an internet connection to open and use this website.

### 5.2.2.4 Standards compliance

- VS code for development
- Mongo DB for database

### 5.2.2.5 Interface/protocol requirements

- Interface will be reactive by using the React.JS.
- The interface will be able to run on any browser that supports JavaScript.
- Https will be used for connection between user and server.

### 5.2.2.6 Data repository and distribution requirements

- Mongo DB will be used for data repository and distribution requirements.

### 5.2.2.7 Security requirements (or other such regulations)

- The system will ensure to make the passwords of users strong.
- All the data will be backed up regularly.

### 5.2.2.8 Memory and other capacity limitations

Following are the minimum memory requirements for optimized performance:

- 2 GB RAM
- 32 GB ROM (SSD preferred)

Other requirements:

- Core i3 or more
- Decent processor
- Graphic Card (not necessary)

### 5.2.2.9 Performance requirements

- There should be stable internet for smooth performance.
- The code should be optimized for best performance.

### 5.2.2.10 Network communications

Hypertext transfer protocol secure (https) will be used for network communications

## 5.2.3 Goals and Guidelines

The goals which our design will be built upon are given below along with the guidelines to achieve these goals and desirability of each goal.

### 5.2.3.1 Modifiability

The development of this system will be based on the KISS principle (“Keep It Simple Stupid”) which will make things simpler and avoid the complexity as much as possible which further will make sure that modifiability and readability of code easy.

### **5.2.3.2 Performance**

The system will be designed in a way that it operates at high speed. Moreover, the system will be designed so that it does not additional space in memory and it will not ask user to download documents which are not useful.

### **5.2.3.3 User-Friendly**

The system will be designed using colors which are friendly to the human eye to make it more user-friendly and interactive. Moreover, it will be designed so that it supports English language speakers which makes it globally usable.

## **5.2.4 Development Methods**

The software methods or approaches that were considered while developing this project include waterfall model and Scrum method. There were many factors that were taken into consideration while selecting the right method for this project and it turns out to be Scrum methodology which was most suitable.

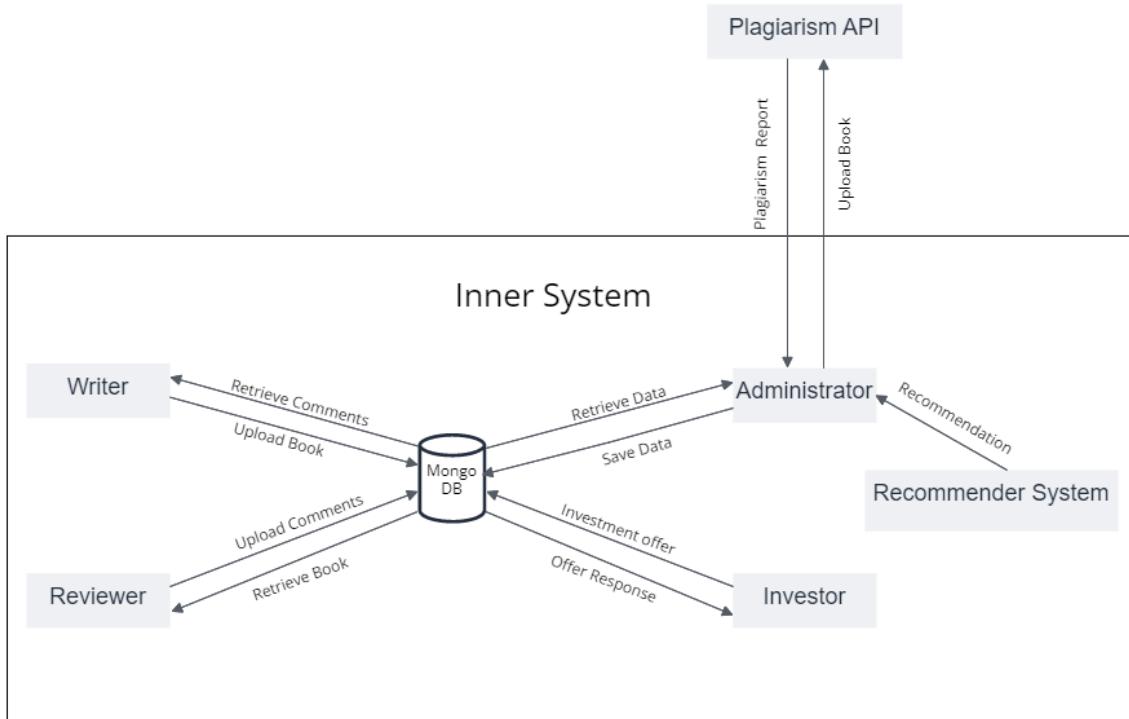
First, if we talk about waterfall model, it was not used here because it is not flexible and changes are very difficult to make. Since, we have to develop this system based on the feedback from our advisor, it is obvious that many changes might be required. So, waterfall model was not suitable in this case.

The method used for the development of this project is given in sub-section below.

### **5.2.4.1 Scrum Methodology**

Scrum Methodology was used for this project because it has somewhat same philosophy as Agile where things are done in incremental fashion. This allows the system to be tested and developed easily. Moreover, this design has some flexibility, which might increase its extensibility and allow us to enhance it by adding new features. Additionally, another advantage of this methodology is that we can easily incorporate the feedback given by our supervisor.

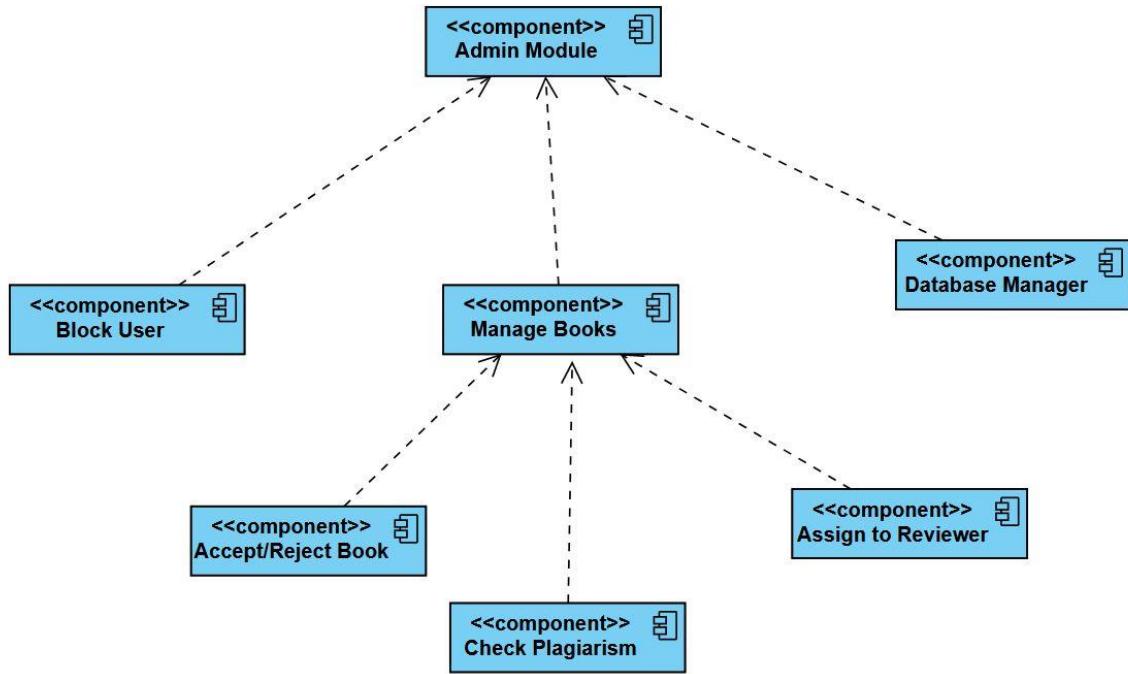
### 5.3 System Architecture



**Figure 17: Architecture Diagram**  
*The high-level diagram of the system*

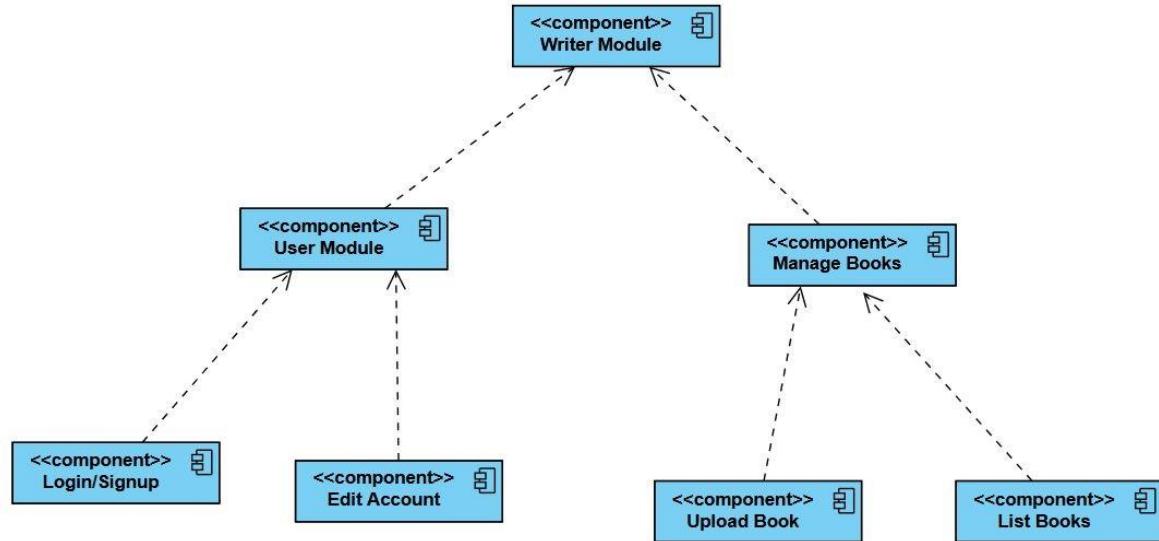
As shown in the above figure, our system was broken down into five major components. The system has four main clients which are investors, admin, writers and reviewers. Each of these clients will sign up and login to the system by uploading data to database or by retrieving data from database. Writer will upload the book to database. Writer can also see comments made by reviewers about the book. Admin will retrieve the book from database. Administrator can also send the book to plagiarism API. Admin can take recommendations for reviewers from recommender system and then he will assign the reviewers to the book. Recommender system will provide the recommendation about reviewers to admin module. Reviewer will retrieve the book from database. Reviewer can upload comments about book to database after reviewing the book. Admin can see the comments about book and if any illegality is found, he will block the writer or reviewer. Investors will retrieve the books that are recommended by reviewers and will then make an offer about book he wants to invest in. He can also see the response he got from the offer he made to the writer. Plagiarism API is an external system to which our system will communicate and from which it will collect the plagiarism report.

### 5.3.1 Subsystem Architecture



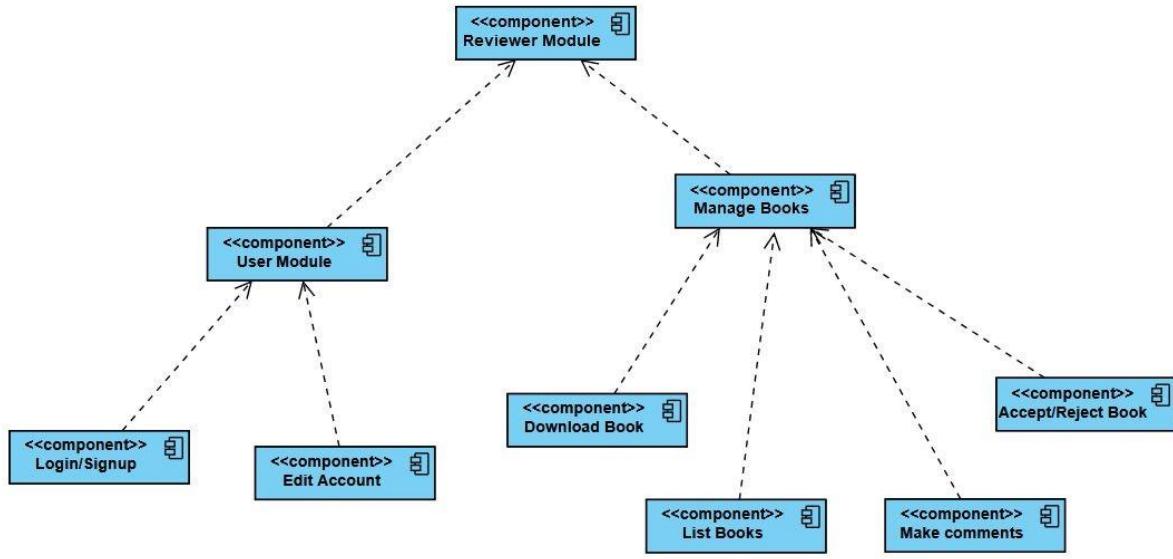
**Figure 18: Admin component Diagram**  
*The internal view of admin component is given above*

The admin can block a user. When a new book is uploaded it will send to plagiarism checker and upon arrival of reports, it will assign reviewers to book. It will accept or reject a book. It will also maintain lists of all users and publishers in the database.



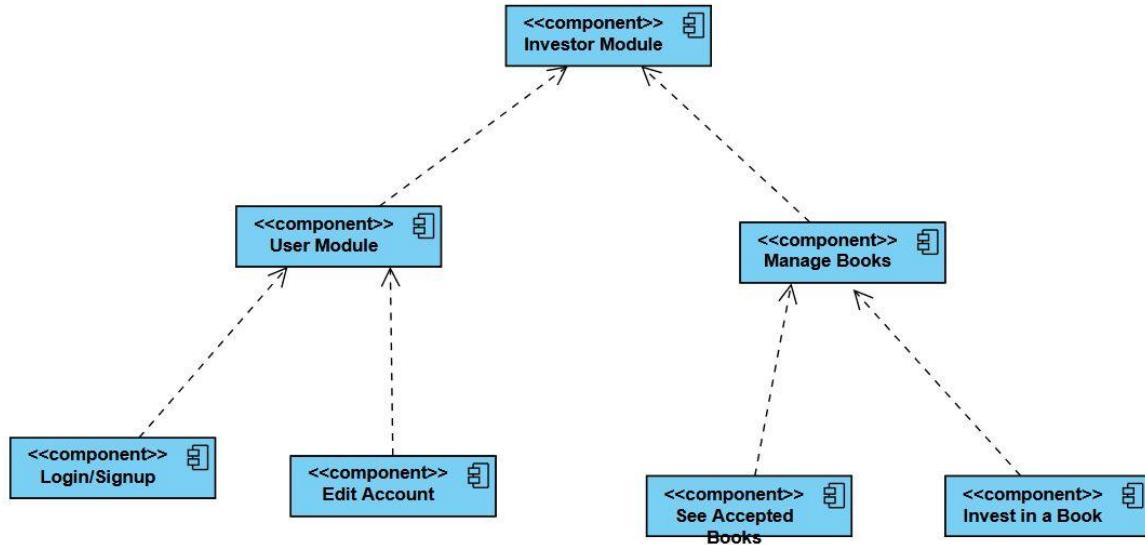
**Figure 19: Writer component Diagram**  
*The internal view of writer component is given above*

The writer will login and sign up to the system. It will edit the account details such as changing password, etc. Moreover, it will be able to upload a book and see the list of books he has already uploaded along with the status of these books.



**Figure 20: Reviewer component Diagram**  
The internal view of reviewer component is given above

Reviewer will login or sign up to the system. He can change his details. Moreover, it will be able to download the book, see the list of already uploaded books, make comment on these books and he can give decision on books.



**Figure 21: Investor component Diagram**  
The internal view of investor component is given above

Investor will login and sign up to the system. Moreover, it will edit account details. Moving on, it will see accepted books by the reviewer and it can also invest in a book.

## 5.4 Architectural Strategies

The architectural strategies that we used for design of this architecture are provided in the subsections below.

### 5.4.1 Reuse of existing product

The system will use existing products such as an API for checking the plagiarism of books such as iThenticate which is provided by “TurnItIn”. This API will send plagiarism reports to Administrator.

### 5.4.2 Use of Programming language

The system shall be built in MERN stack technology. Front-end of the system will be built in React JS, back-end in Node JS. For data storage, we will use Mongo DB which reduces the time to setup the database and modelling. The choice to adopt the MERN stack is based on the knowledge that it offers robust frameworks and libraries, as well as a component-based design structure, which is considerably superior to alternative approaches to creating online applications. Moreover, we will also use Python for recommendation system creation and for watermarking.

### 5.4.3 User-Interface Paradigm

The Schneider man's eight golden rules for interface design which serve as cornerstone of any UI design, will guide the styling of our front end. No extra training will be required for the usage of our website

### 5.4.4 Extension and Enhancement of software

In future, we will add a complete module for publishers where anyone with a publishing house will get registered and our investor can also publish the books themselves. Moreover, we will also automate the verification of books and blocking of clients and reduce the administrator involvement as much as possible.

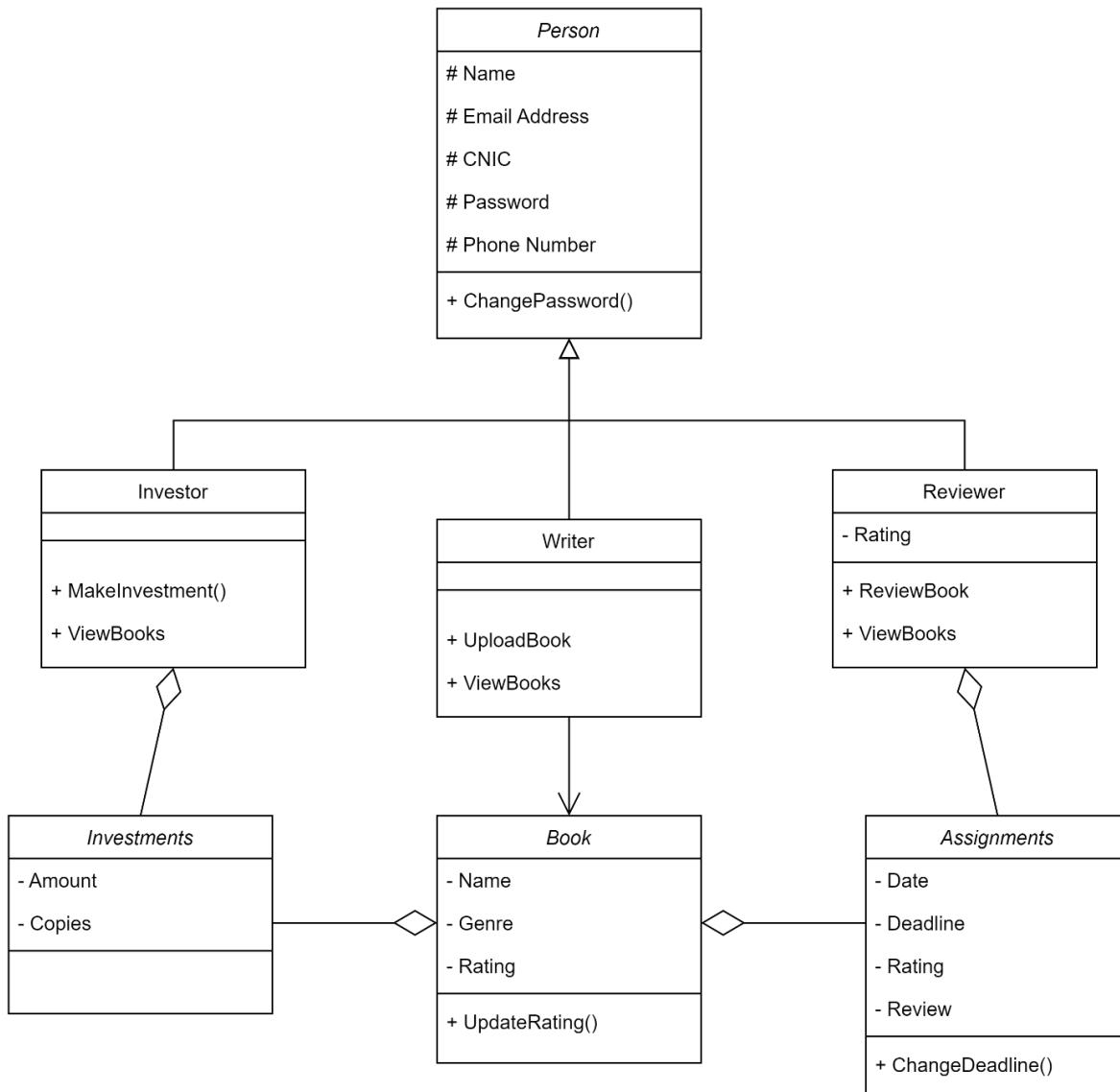
### 5.4.5 Error detection and Recovery

When the writer will upload a book in wrong format then an error message will be displayed and writer will be asked to upload the book again.

### 5.4.6 Concurrency and Synchronization

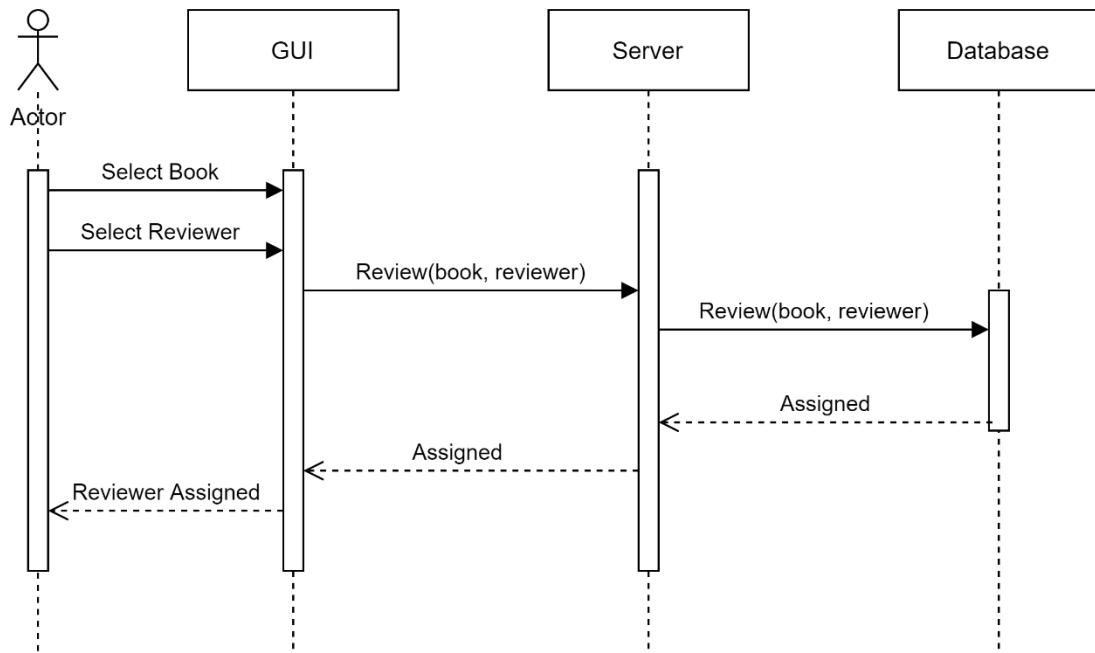
In order to allow users to communicate with the web application while the system is working on any time-consuming activities like web API queries, we will be employing async operations on the frontend so that system does not hinders the things that user wants to do on our system.

## 5.5 Domain Model/Class Diagram

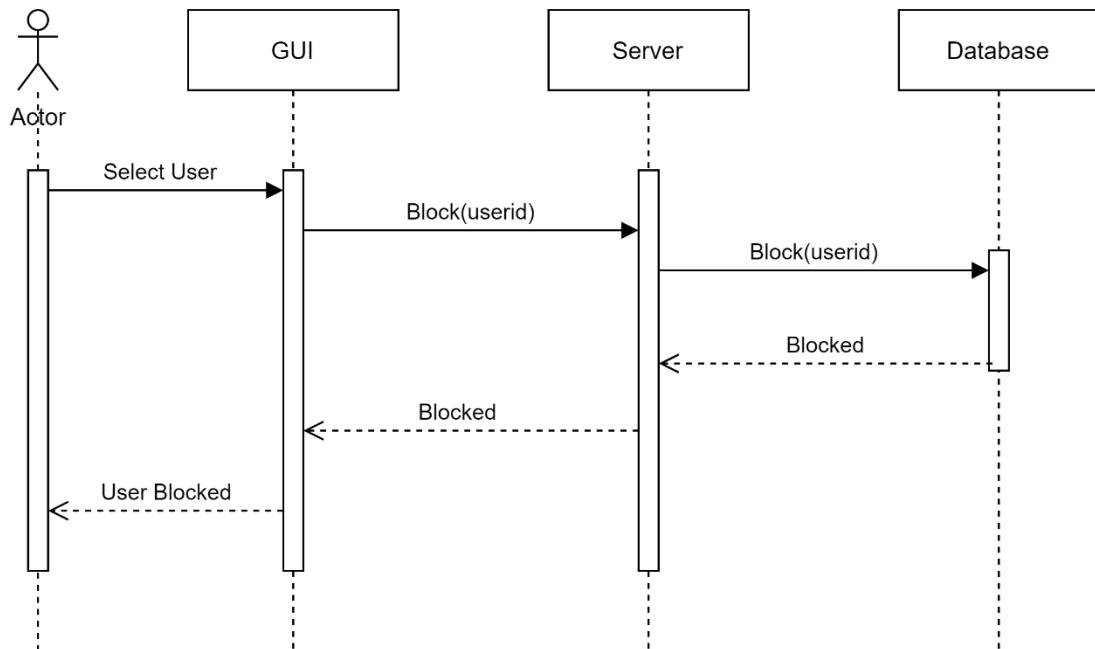


**Figure 22: Class Diagram**  
*The class diagram of whole system is given above*

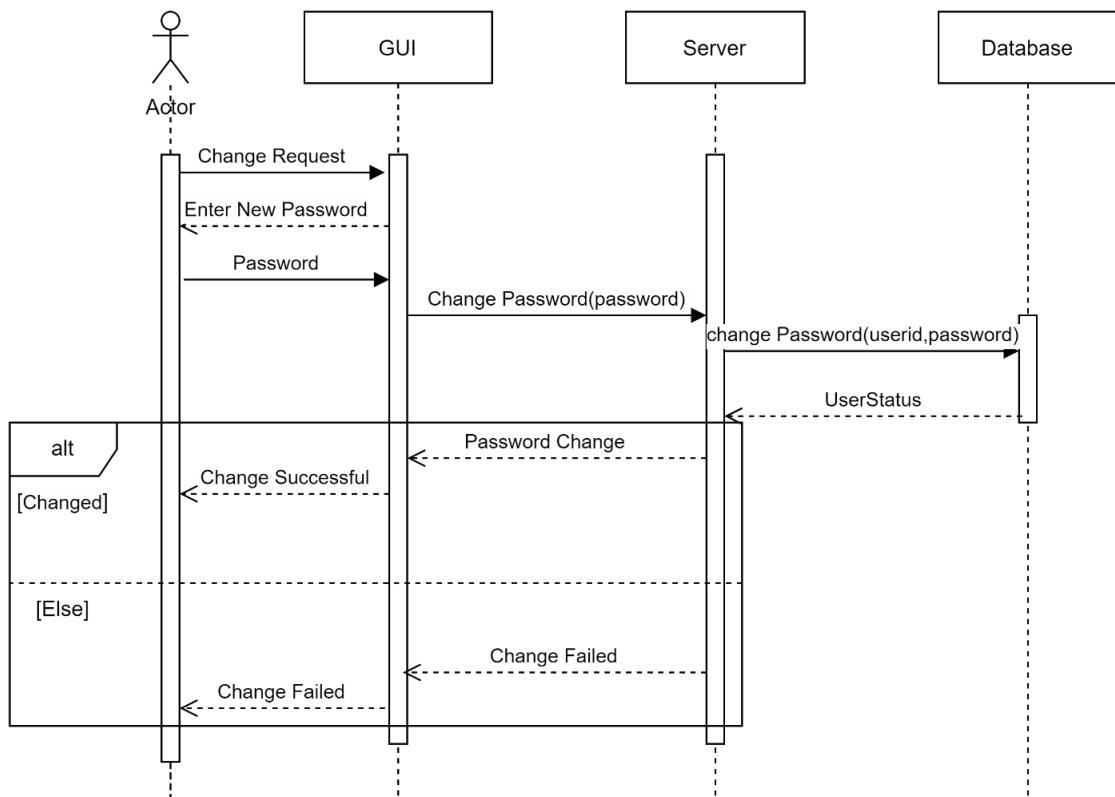
## 5.6 Sequence Diagrams



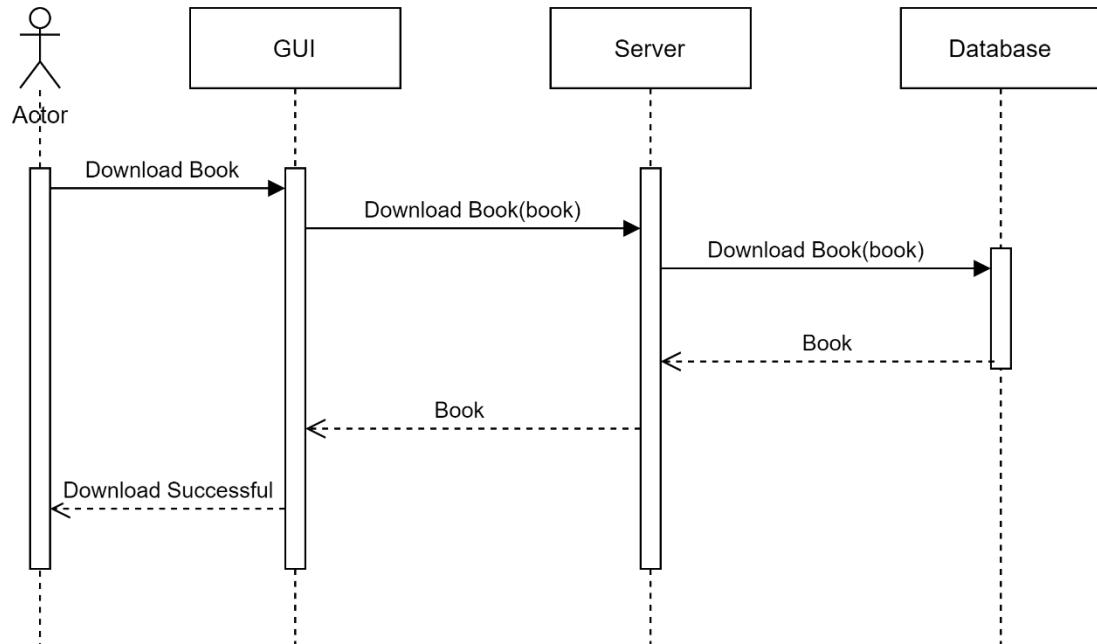
**Figure 23: Assign Reviewer**  
*The sequence diagram of assign a reviewer is given above*



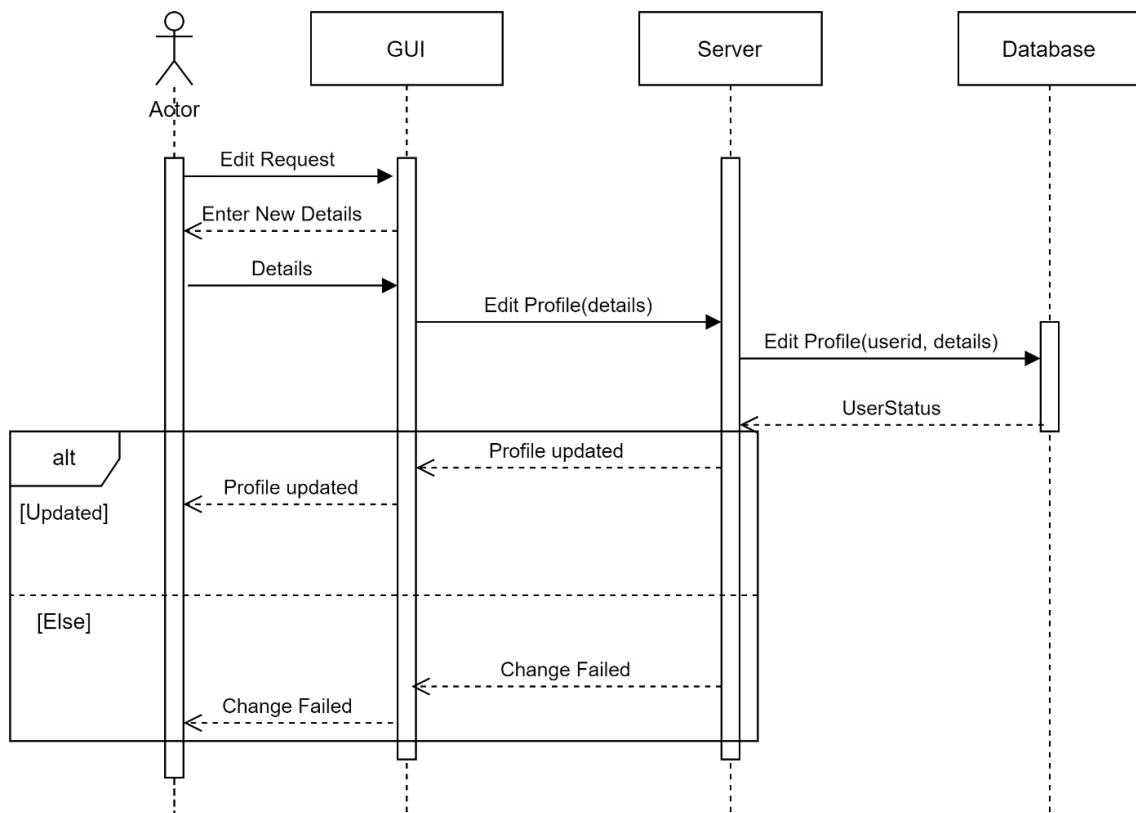
**Figure 24: Block a user**  
*The sequence diagram of block a user is given above*



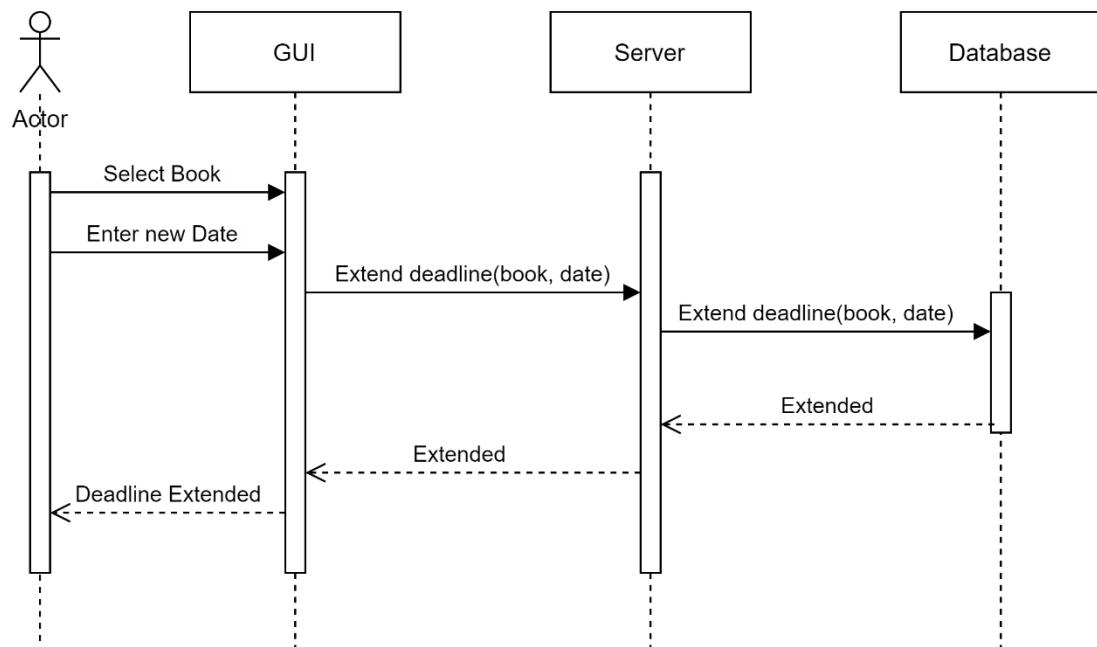
**Figure 25: Change password**  
The sequence diagram of change password of user is given above



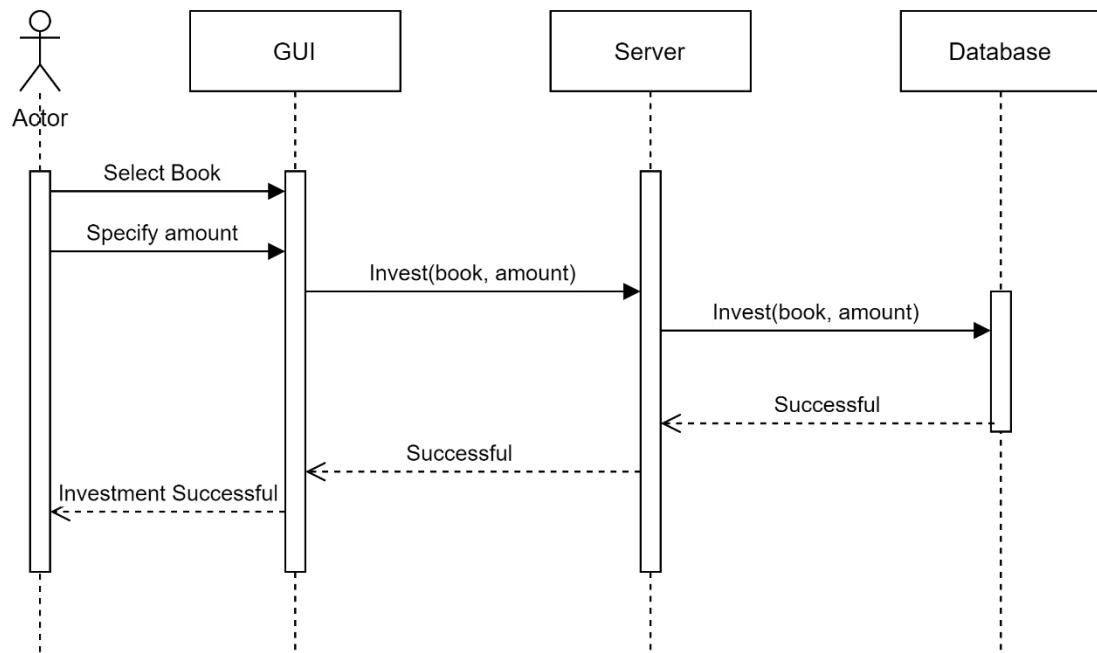
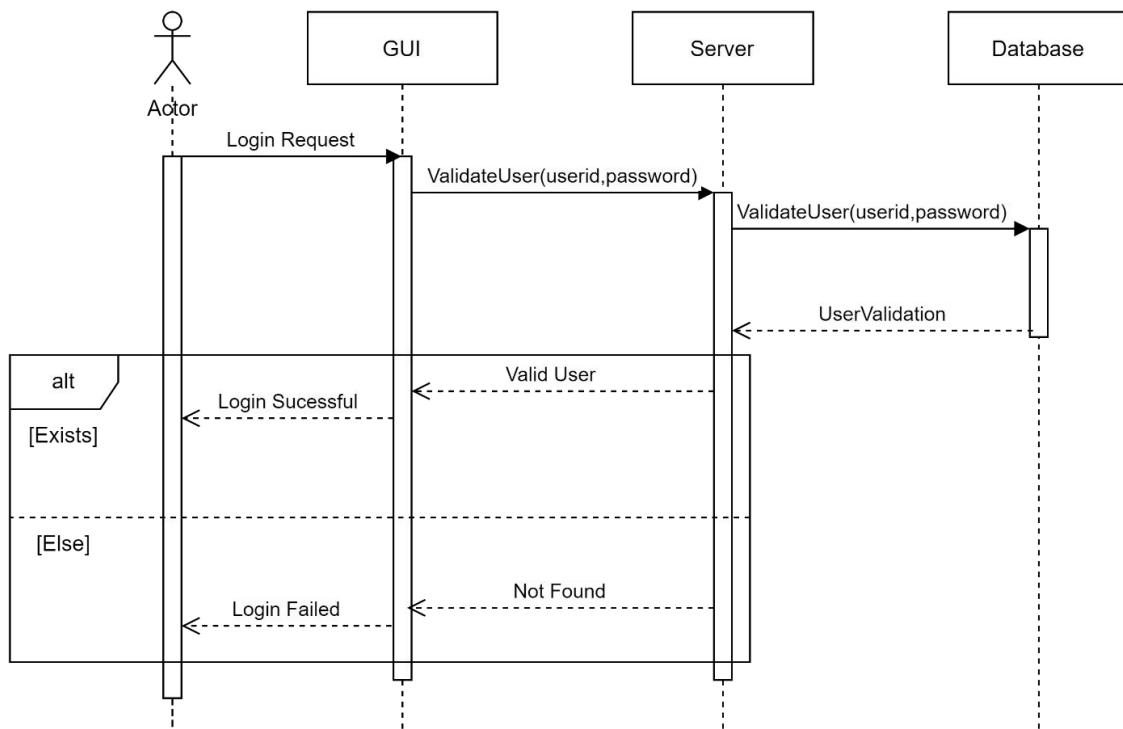
**Figure 26: Download a book**  
The sequence diagram of download a book is given above

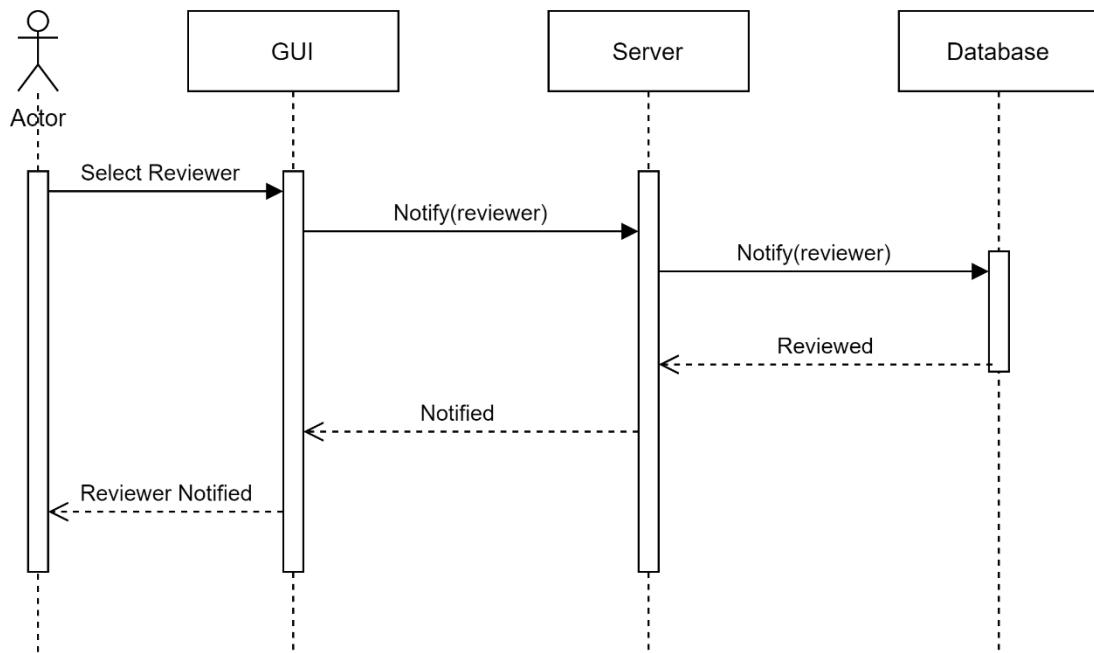


**Figure 27: Edit profile**  
*The sequence diagram of edit profile is given above*

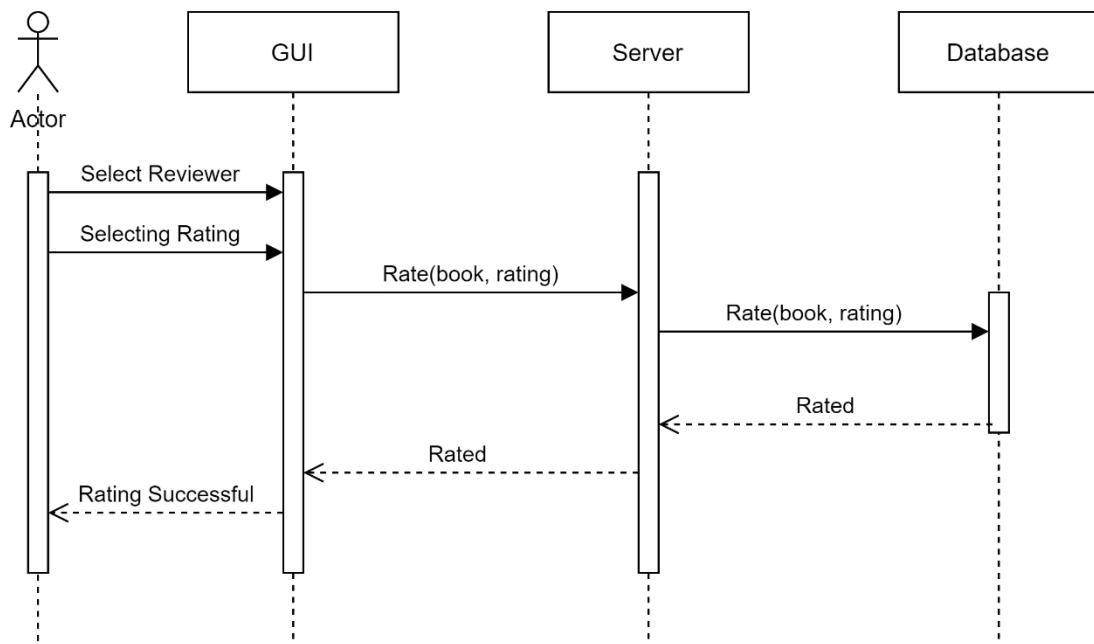


**Figure 28: Extend deadline**  
*The sequence diagram of extend deadline is given above*

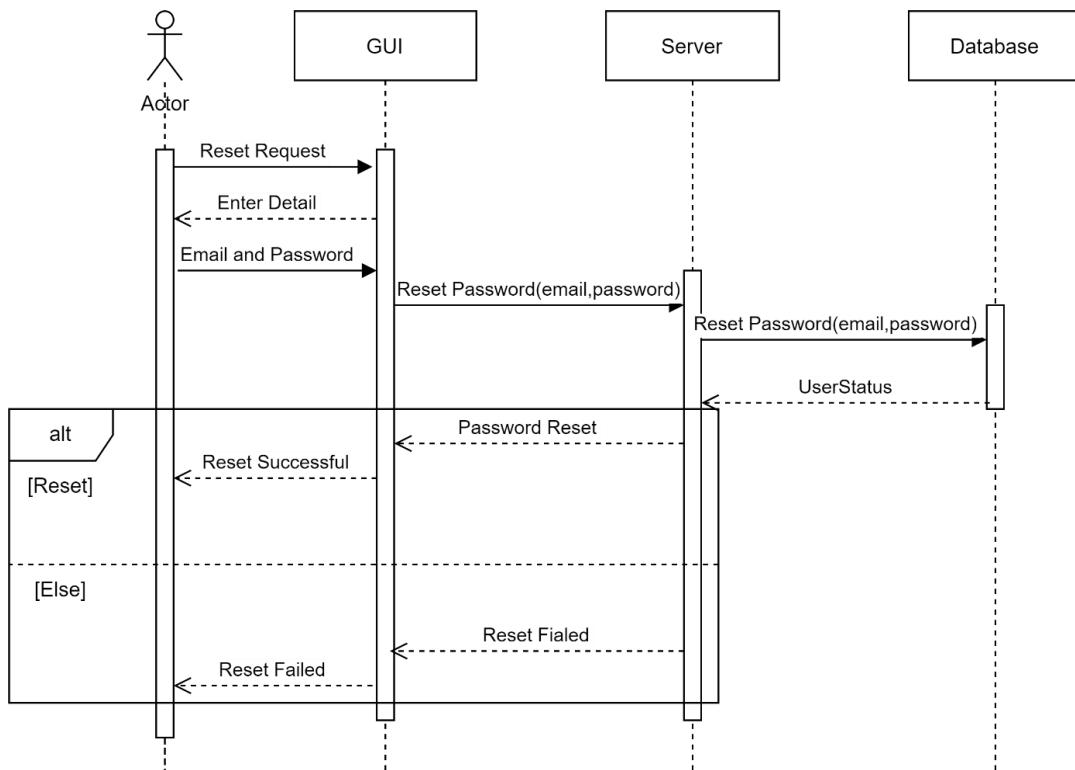
**Figure 29: Invest in a book***The sequence diagram of block a user is given above***Figure 30: Login***The sequence diagram of Login is given above*



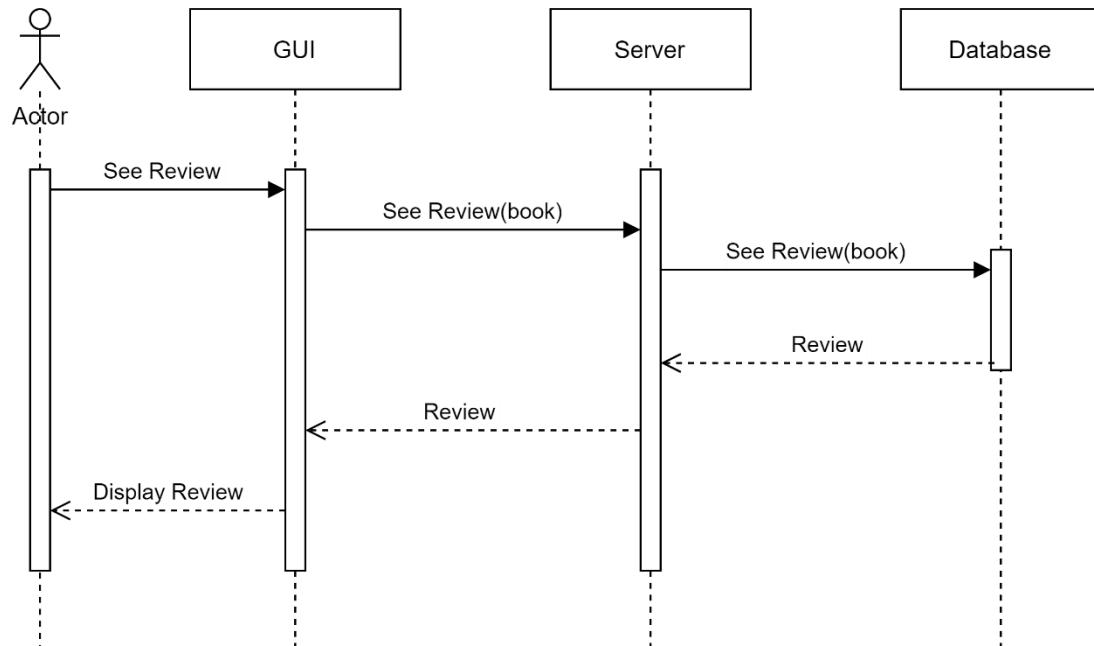
**Figure 31: Notify a reviewer**  
*The sequence diagram of notify a reviewer is given above*



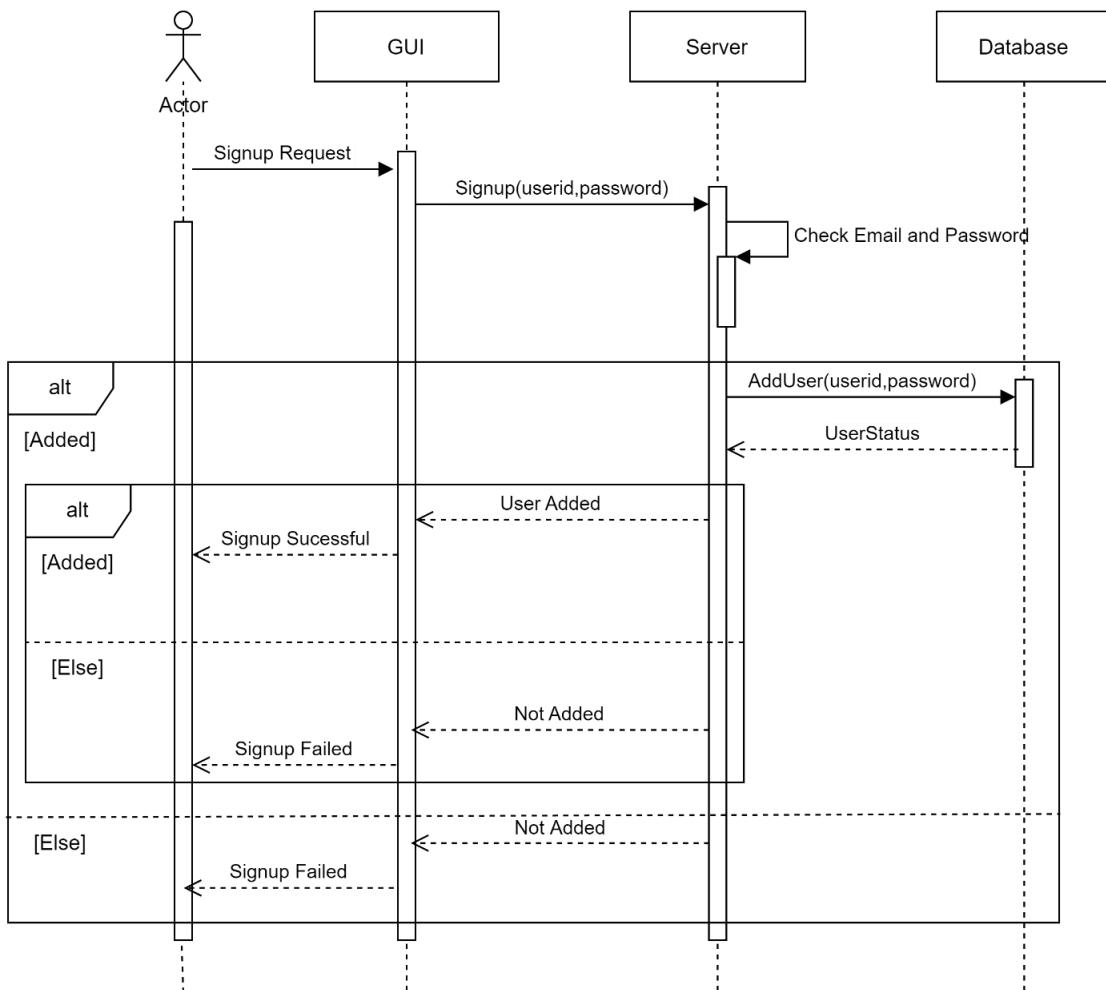
**Figure 32: Rate a reviewer**  
*The sequence diagram of rate a reviewer is given above*



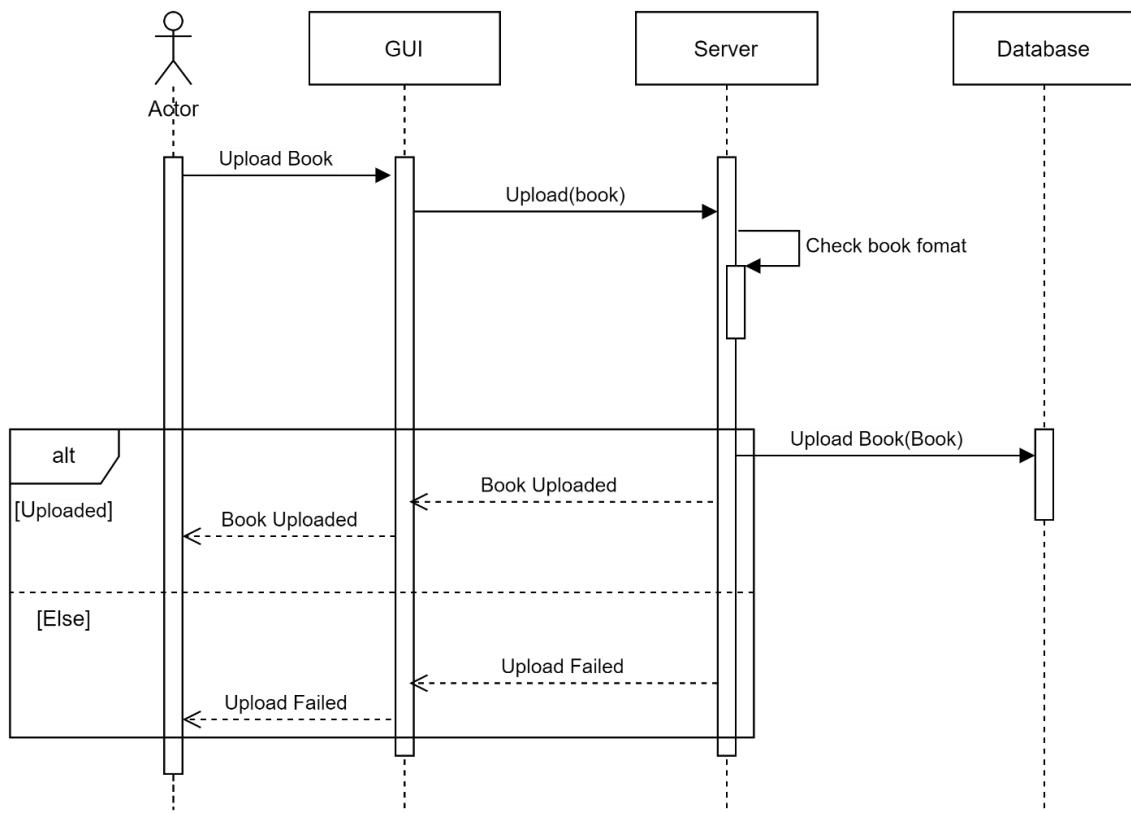
**Figure 33: Reset Password**  
*The sequence diagram of reset password is given above*



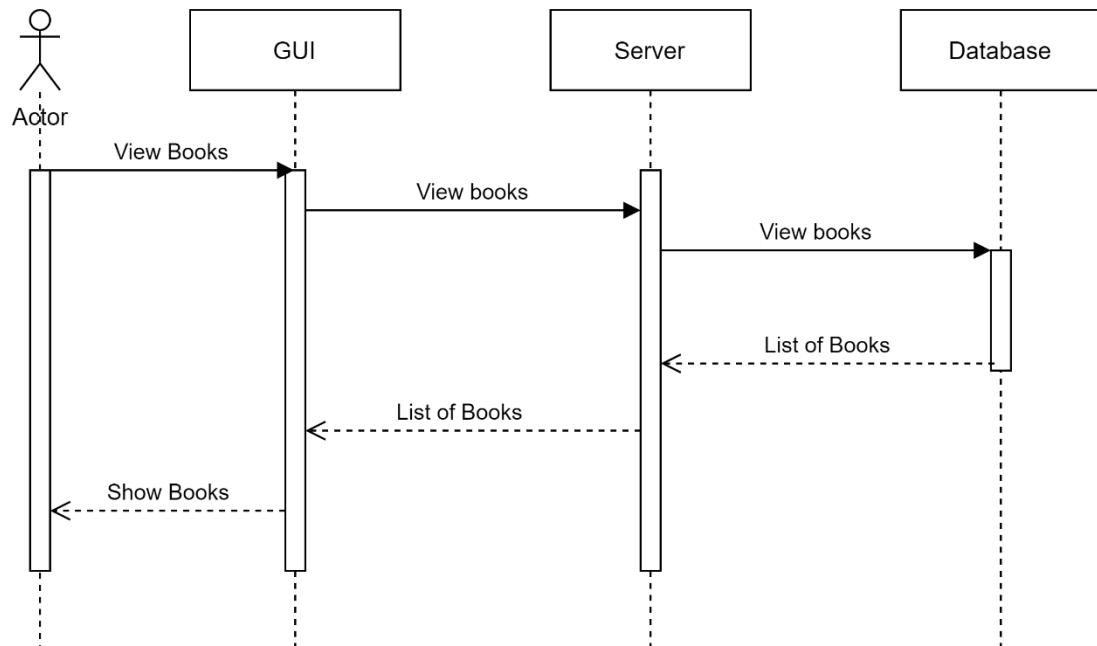
**Figure 34: See a review**  
*The sequence diagram of see a review is given above*



**Figure 35: Sign Up**  
The sequence diagram of sign up is given above



**Figure 36: Upload Book**  
The sequence diagram of upload a book is given above



**Figure 37: View Books**  
The sequence diagram of view books is given above

## 5.7 Policies and Tactics

Following are the policies and tactics that will be used in this project:

### 5.7.1 Conventions

We will be using MERN which basically works on JavaScript for building our website. Every aspect of the code will be tested using testing techniques like selenium. We will be using visual studio code as our IDE as it is lightweight, can be customized and supports many extensions.

### 5.7.2 Testing the Software

Both white box and black box testing techniques will be used for testing different modules of the software. Each and every aspect will be tested thoroughly to avoid any error. Lastly, the whole system will be tested to assure the best quality.

### 5.7.3 Maintenance of Software

The software will be updated regularly and more functionalities can be added at the time of need. Any reported issues or bugs can also be resolved. Lastly, some functionalities can also be added to ensure the best performance.

### 5.7.4 Vulnerability Check

All the vulnerabilities can be checked from npm feature called audit. It checks the vulnerabilities and notifies if the current version of dependencies has any vulnerability. It can be fixed by using the fix command which updates the version to the nearest one that is patched up.

### 5.7.5 End User Interface

This application will be a web-based application and will be responsive to any device it is accessed on. It can be viewed from a mobile, laptop or even tablets without any difficulty.

### 5.7.6 Tools

Visual studio code will be mainly used for the development phase as it is lightweight, can be customized and also supports a ton of extensions. We will also use Node and Express JS along with Mongo DB.

As a conclusion to this chapter, our design is based on assumptions such as internet availability, etc. Moving on, constraints such as network communication affected the design of our system. Moreover, goals like modifiability, etc. are achieved through our design. The development method which we used in our system is SCRUM. It also provides system architecture diagram and architecture of components of system. Furthermore, it also mentions architectural strategies of system, class diagram of the system and sequence diagrams of different operations that different users will perform is also given. Lastly, it also covers policies and tactics that changed the interface of the system.

## Chapter 6: Implementation and Test Cases

All the details related to implementation of designed prototype is provided in the sub-section given below. The implementation details of the website created and interaction between the modules as well as recommendation system used for recommending the reviewers are described here.

### 6.1 Implementation

The implementation details of the components and algorithms of prototype designed is given below.

#### 6.1.1 Prototype Description

The prototype developed will describe all the functionalities of the writer module. This website will be developed in MERN stack. When a user visits our website. He will land on login page. If he does not have an account, then he will sign up through sign up component. Sign up component will send him an email through which he will verify the account and will be added to users list along with their role.

When he logs himself as a writer, he will be placed on the home page for writers. Here, he can upload a book. For uploading a book, he will have to select a reviewer type. After that, he will have to enter the name of book, description of book, and after that he will have to select the book file which he wants to upload. He can also see the list of pending books on Homepage and he will be redirected to “Pending books page”.

In “Pending books page”, the writer can view the comments made by reviewer and he can resubmit the book that was asked to resubmit by reviewer on this page. Moreover, he can rate reviewers. Writer can also make comments to writers on this page. Moreover, he will also be able to view the investment offers made on the books.

#### 6.1.2 Authentication

The basic purpose of building this website is to provide ease to users and for this purpose, it is necessary for the users to access the website. Users can interact with it just like they would with any other application or website thanks to the implementation of the fundamental functionalities like Login, Sign up and Sign Out. We have implemented these functionalities by creating APIs in Node JS for Login, Sign Up and Sign Out respectively. The flow for Login module will be such that user will first fill Login form and after clicking on login button, an API call will be induced and this API will match the email and password entered by user with the email and password stored in Mongo DB documents. It will first calculate the hash of the password entered by user using the salt and will then match the hashed password to password stored in Mongo DB documents using JWT match password. If the passwords match, the API will create a token and this token will be stored in local storage of the user. This will enable the user to forget about logging in each time and will be redirected directly without logging in if he was previously not logged out. For Sign Up module, the user will first enter his details using the front end form and then these details will be sent to the API which is created for Sign Up. This API will calculate the hash of the password and will then create a JWT token using that password and email. For Sign Out module, the user will click on the Log Out and then the token stored in local storage will be cleared.

### **6.1.3 Saving and retrieval of data**

We will use Mongo DB to store the information about users. Mongo DB provides dynamic structure which makes it having less down time. Moreover, it will store data in collection and documents. In Mongo DB, you can easily alter a document's structure by adding new fields or removing others. In our case, it will be used to store the information such as users, books, reviews and contracts, etc.

### **6.1.4 Upload a book**

Writer will upload the book using this component. For this purpose, he will have to select a reviewer type. After that, he will have to enter the name of book, description of book, and after that he will have to select the book file which he wants to upload. This book will then be saved to Mongo DB.

### **6.1.5 Rate Reviewers**

Writer will be able to rate reviewer through this component. He will have to select the book for which he wants to rate reviewer and then he will select the review for which he wants to give rating. He will give reviewer rating and this rating will be saved on database.

### **6.1.6 Resubmit a book**

Writer will select the book from “pending books page”. Comments will be retrieved for the book he selected and will be displayed to writer. He will select the file he wants to resubmit and when he resubmit it, this book will be saved to database.

### **6.1.7 Watermarking a book**

The website under construction should enable the writer to upload a book. For this purpose, we have created a functionality of Upload Book. This will use the API created in Node JS. The user will first give the book name and book description and will select the type of reviewer and will select a file which will be in pdf format. When user will click on upload book button, it will call the API created by us and this API will store the file in folder of Files which will contain all the books and will act as a server. Moreover, it will store the document name along with all its details in Mongo DB documents. This API call will also induce another API call for plagiarism checker which will be discussed in the next sub chapter below. Moreover, this API will call the API of watermarking which will watermark the pdf uploaded by the writer by merging each page with our logo. This watermarking algorithm will use Python library and we will use spawn to run the code of python in JavaScript. Now, this watermarked edition of the book will be stored in folder of our computer which will act as a server. The writer will then be redirected by the API to a page where they can view the uploaded book's specifics, such as its name, description, type of reviewer, and status.

### **6.1.8 Plagiarism Checker**

There is a major problem of plagiarism in the field of literature and books are no exception to it. So this website will be provided with the functionality of plagiarism detector which will detect plagiarism of newly uploaded books. Whenever a user will upload a book, the API of plagiarism checker will be called. This API will be provided by iThenticate which will allow us to integrate the plagiarism detection functionality of iThenticate into the system under construction. After plagiarism is detected by this API, it will send the results to Admin which

will see whether the results are according to requirements. The Admin will decide whether or not the book can be published on the website based on the report. The book will be approved for publication if the similarity rate falls below the acceptable cutoff, and the API will update the status of the book in the Mongo DB documents. The writer will be shown in the status that their book has been rejected due to plagiarism if the similarity percentage exceeds the acceptable limit, in which case the Admin may reject the submission. In these situations, the author might be given the chance to edit their work and submit it again for plagiarism checking by clicking on resubmit book button provided to the writer in front of book.

## 6.1.9 Recommendation System

The system will have the ability to recommend best reviewers for a certain book and will rank these reviewers in front of admin so that admin can easily approach these reviewers. The model will take gender, preferences and age of reviewer and will also take genre of book as an input and will rank the best reviewers as described above. For training and testing of the model, data such as that of books was taken from Kaggle good reads data set. Moreover, as there was no system found for data gathering of reviewer, we use synthetic data which was formed on the basis of reviews of such books from Kindle by Amazon. We used this data to train our artificial neural network. After training, we used this network to review the input data and produce a list of suggested reviewers, ranked according to how well suited they are to the particular book. Moreover, the metrics for calculating the accuracy of recommendation system was based on that whether the reviewer accepted the offering and whether the review provided by the reviewer was rated well by the writer and investor. In the start, the system will not have a way to find out the accuracy of recommendation system as it will require a reviewer to review some books, so that it can measure accuracy. But once website gets in running state for some time, then recommendation system will gather data registered on website's server and will then use that data to find the accuracy. Moreover, once the website has reached a time of maturity, then we will again train our model according to the data stored in website's server so that the real world scenario will only be considered. We anticipate that as the system continues to collect data and hone its algorithms, its accuracy will increase even further, making it a crucial component of the website's architecture.

## 6.2 Test case Design and description

All the test cases performed during the testing phase of the system are elaborated in the sub chapters below.

### 6.2.1 User Login Test Case

Login for Users			
<b>Test Case ID:</b>	<i>1</i>	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case Reference(s):</b>	<i>Login use case (1)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if user is able to successfully login to the website</i>		
<b>Product/Ver/Module:</b>	<i>Login Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		
<b>Assumptions:</b>	<i>Login button is visible to the user</i>		

<b>Pre-Requisite:</b>	<i>The user is already stored in the database and registered as a valid user</i>	
<b>Step No.</b>	<b>Execution description</b>	<b>Procedure result</b>
<b>1</b>	<i>User clicks on login button.</i>	<i>System displays login page.</i>
<b>2</b>	<i>User enters his/her username and password and clicks login.</i>	<i>System greets the user to the home screen.</i>
<b>Comments:</b> The test case is passed. Our system is working according to our need.		
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed		

### 6.2.2 Sign-Up Test Case

Sign-Up for Users					
<b>Test Case ID:</b>	2	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>		
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>		
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case Reference(s):</b>	<i>Sign-Up use case (2)</i>		
<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if the user can sign up using their email address</i>				
<b>Product/Ver/Module:</b>	<i>Sign-Up Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Sign-Up button is visible, and user can locate it</i>				
<b>Pre-Requisite:</b>	<i>The user has a valid email address</i>				
<b>Step No.</b>	<b>Execution description</b>	<b>Procedure result</b>			
<b>1</b>	<i>User clicks on sign up button</i>	<i>The system displays the sign-up page</i>			
<b>2</b>	<i>User fills their information on the sign up page and clicks sign up</i>	<i>The system stores all user information and registers that user as a valid user</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.3 Reset Password Test Case

Reset Password			
<b>Test Case ID:</b>	3	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case Reference(s):</b>	<i>Reset Password use case (4)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if user can reset their password</i>		
<b>Product/Ver/Module:</b>	<i>Reset Password Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		

<b>Assumptions:</b>	<i>Forget password button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The user is already stored in the database and registered as a valid user</i>				
Step No.	<b>Execution description</b>		<b>Procedure result</b>		
<b>1</b>	<i>User clicks on reset password button.</i>		<i>System displays reset password page, asking for user's email</i>		
<b>2</b>	<i>User enters the registered email.</i>		<i>A verification code is sent to the user.</i>		
<b>3</b>	<i>User enters the verification code.</i>		<i>System verifies the code and asks user of a new password.</i>		
<b>4</b>	<i>User enters new password and clicks on reset password button.</i>		<i>System will change the password and store the new password in the database.</i>		
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

#### 6.2.4 Change Password Test Case

<b>Change Password</b>					
<b>Test Case ID:</b>	4	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>		
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>		
<b>Test Date:</b>	14-04-2023	<b>Use Case Reference(s):</b>	<i>Change Password use case (5)</i>		
<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if user can change their password</i>				
<b>Product/Ver/Module:</b>	<i>Edit profile Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Edit profile button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The user is already logged in to their account</i>				
Step No.	<b>Execution description</b>		<b>Procedure result</b>		
<b>1</b>	<i>User clicks on edit profile button.</i>		<i>System shows edit profile page.</i>		
<b>2</b>	<i>User enters their old and new password.</i>		<i>System verifies their old password and set their password to the new one.</i>		
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

#### 6.2.5 Edit Profile Test Case

<b>Edit Profile</b>			
<b>Test Case ID:</b>	5	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	14-04-2023	<b>Use Case Reference(s):</b>	<i>Edit Profile use case (6)</i>

<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if user can edit their profile</i>				
<b>Product/Ver/Module:</b>	<i>Edit profile Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Edit profile button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>				
Step No.	Execution description	Procedure result			
<b>1</b>	<i>User clicks on edit profile button</i>	<i>System displays edit profile page.</i>			
<b>2</b>	<i>User enters their details.</i>	<i>System stores their new information.</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

## 6.2.6 Upload Book Test Case

<b>Upload a Book</b>					
<b>Test Case ID:</b>	<i>6</i>	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>		
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>		
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case Reference(s):</b>	<i>Upload book use case (7)</i>		
<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if a writer can upload a book</i>				
<b>Product/Ver/Module:</b>	<i>Writer Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Upload book button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The writer is already signed into their account</i>				
Step No.	Execution description	Procedure result			
<b>1</b>	<i>User clicks on the upload book button.</i>	<i>Upload book page is displayed.</i>			
<b>2</b>	<i>User enters the details of the book and selects the book from their local directory.</i>	<i>The book is uploaded to the system.</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

## 6.2.7 Resubmit Book Test Case

<b>Resubmit Book</b>			
<b>Test Case ID:</b>	<i>7</i>	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case</b>	<i>Resubmit Book use case (8)</i>

		<b>Reference(s):</b>			
<b>Revision History:</b>		<i>None</i>			
<b>Objective</b>		<i>To check if writer can resubmit a book</i>			
<b>Product/Ver/Module:</b>		<i>Writer Module of website</i>			
<b>Environment:</b>		<i>Website is up and running</i>			
<b>Assumptions:</b>		<i>Resubmit button is visible to the user</i>			
<b>Pre-Requisite:</b>		<i>User is already logged into their account</i>			
<b>Step No.</b>	<b>Execution description</b>	<b>Procedure result</b>			
<b>1</b>	<i>User clicks on the resubmit book button.</i>	<i>Resubmit book page is displayed.</i>			
	<i>User selects the book from their local directory.</i>	<i>The book is uploaded to the system.</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.8 View List of Pending Books Test Case

<b>Pending Books</b>					
<b>Test Case ID:</b>	8	<b>QA Test Engineer:</b>	<i>Usama Iqbal</i>		
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>		
<b>Test Date:</b>	14-04-2023	<b>Use Case Reference(s):</b>	<i>View list of pending books use case (9)</i>		
<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if a writer can see the list of pending books</i>				
<b>Product/Ver/Module:</b>	<i>Writer Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>View pending list button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>Writer is already logged into their account</i>				
<b>Step No.</b>	<b>Execution description</b>	<b>Procedure result</b>			
<b>1</b>	<i>User clicks on view pending books button.</i>	<i>System shows a list of pending books.</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.9 View comments of Reviewer Test Case

<b>View Comments for Writer</b>			
<b>Test Case ID:</b>	9	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	14-04-2023	<b>Use Case Reference(s):</b>	<i>View comments of reviewers use case (10)</i>

<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if writer can see the comments of reviewer</i>				
<b>Product/Ver/Module:</b>	<i>Writer Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>View comments button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>				
Step No.	Execution description	Procedure result			
<b>1</b>	<i>User clicks on view comments button.</i>	<i>System displays the comments of reviewer.</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.10 Rate Reviewer Test Case

<b>Rate a Reviewer</b>					
<b>Test Case ID:</b>	<i>10</i>	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>		
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>		
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case Reference(s):</b>	<i>Rate reviewer use case (11)</i>		
<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if a writer can rate a reviewer</i>				
<b>Product/Ver/Module:</b>	<i>Writer Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Rate reviewer button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>				
Step No.	Execution description	Procedure result			
<b>1</b>	<i>User clicks on rate reviewer button and selects rating.</i>	<i>The rating is recorded in the system against the reviewer.</i>			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.11 Download an Assigned Book Test Case

<b>Download Book</b>			
<b>Test Case ID:</b>	<i>11</i>	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	<i>14-04-2023</i>	<b>Use Case Reference(s):</b>	<i>Download an Assigned book use case (12)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if reviewer can download a book</i>		
<b>Product/Ver/Module:</b>	<i>Reviewer Module of website</i>		

<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Download button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The writer is already logged into their account</i>				
Step No.	<b>Execution description</b>		<b>Procedure result</b>		
1	<i>User clicks on download button and selects the directory to save the book.</i>		<i>System saves the book the selected directory.</i>		
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.12 Fill Review Form Test Case

<b>Review a Book</b>					
<b>Test Case ID:</b>	12	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>		
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>		
<b>Test Date:</b>	14-04-2023	<b>Use Case Reference(s):</b>	<i>Fill review form use case (13)</i>		
<b>Revision History:</b>	<i>None</i>				
<b>Objective</b>	<i>To check if a reviewer can fill review form</i>				
<b>Product/Ver/Module:</b>	<i>Reviewer Module of website</i>				
<b>Environment:</b>	<i>Website is up and running</i>				
<b>Assumptions:</b>	<i>Review button is visible to the user</i>				
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>				
Step No.	<b>Execution description</b>		<b>Procedure result</b>		
1	<i>User clicks on review button.</i>		<i>System shows a review form.</i>		
2	<i>Reviewer fills the form and selects a decision about the book.</i>		<i>System stores the review.</i>		
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

### 6.2.13 Invest in Book Test Case

<b>Invest in a Book</b>			
<b>Test Case ID:</b>	13	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	15-04-2023	<b>Use Case Reference(s):</b>	<i>Invest in a Book use case (14)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if investor can invest in a book</i>		
<b>Product/Ver/Module:</b>	<i>Investor Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		

<b>Assumptions:</b>	<i>Invest button is visible to the user</i>	
<b>Pre-Requisite:</b>	<i>The investor is already logged into their account</i>	
Step No.	Execution description	Procedure result
<b>1</b>	<i>Investor clicks on invest button</i>	<i>An investment page is displayed.</i>
<b>2</b>	<i>Investor selects the publisher and the amount of copies and clicks on save button.</i>	<i>The details of the investment is recorded in the system against the book.</i>
<b>Comments:</b> The test case is passed. Our system is working according to our need.		
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed		

## 6.2.14 View Review Test Case

View Review			
<b>Test Case ID:</b>	<i>14</i>	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	<i>15-04-2023</i>	<b>Use Case Reference(s):</b>	<i>View review use case (15)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if an investor can see the complete review of a book</i>		
<b>Product/Ver/Module:</b>	<i>Investor Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		
<b>Assumptions:</b>	<i>View Review button is visible to the user</i>		
<b>Pre-Requisite:</b>	<i>The writer is already logged into their account</i>		
Step No.	Execution description	Procedure result	
<b>1</b>	<i>User clicks on view review button.</i>	<i>System shows the complete review of the book.</i>	
<b>Comments:</b> The test case is passed. Our system is working according to our need.			
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed			

## 6.2.15 View Invested Books Test Case

View Invested Books			
<b>Test Case ID:</b>	<i>15</i>	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b>	<i>Zaviyaar Bin Irfan</i>
<b>Test Date:</b>	<i>15-04-2023</i>	<b>Use Case Reference(s):</b>	<i>View invested books use case (16)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if investor can see the list of invested books</i>		
<b>Product/Ver/Module:</b>	<i>Investor Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		

<b>Assumptions:</b>	<i>View invested books button is visible to the user</i>	
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>	
<b>Step No.</b>	<b>Execution description</b>	<b>Procedure result</b>
<b>1</b>	<i>User clicks on view invested books button.</i>	<i>System displays a list of books.</i>
<b>Comments:</b> The test case is passed. Our system is working according to our need.		
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed		

## 6.2.16 Rate Reviewer Test Case

<b>Rate Reviewer</b>		
<b>Test Case ID:</b>	<i>16</i>	<b>QA Test Engineer:</b> <i>Muhammad Ahmad</i>
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b> <i>Usama Iqbal</i>
<b>Test Date:</b>	<i>18-04-2023</i>	<b>Use Case Reference(s):</b> <i>Rate reviewer use case (17)</i>
<b>Revision History:</b>	<i>None</i>	
<b>Objective</b>	<i>To check if a investor can rate a reviewer</i>	
<b>Product/Ver/Module:</b>	<i>Investor Module of website</i>	
<b>Environment:</b>	<i>Website is up and running</i>	
<b>Assumptions:</b>	<i>Rate button is visible to the user</i>	
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>	
<b>Step No.</b>	<b>Execution description</b>	<b>Procedure result</b>
<b>1</b>	<i>User clicks on rate reviewer button and selects a rating for the reviewer.</i>	<i>Rating is stored in the system against the reviewer.</i>
<b>Comments:</b> The test case is passed. Our system is working according to our need.		
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed		

## 6.2.17 View comments of Reviewer Test Case

<b>View Comments for Admin</b>			
<b>Test Case ID:</b>	<i>17</i>	<b>QA Test Engineer:</b> <i>Muhammad Ahmad</i>	
<b>Test case Version:</b>	<i>1</i>	<b>Reviewed By:</b> <i>Usama Iqbal</i>	
<b>Test Date:</b>	<i>18-04-2023</i>	<b>Use Case Reference(s):</b> <i>View comments of reviewers use case (18)</i>	
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if admin can see the comments of reviewer</i>		
<b>Product/Ver/Module:</b>	<i>Writer Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		
<b>Assumptions:</b>	<i>View comments button is visible to the user</i>		
<b>Pre-Requisite:</b>	<i>The user is already logged into their account</i>		

Step No.	Execution description	Procedure result
1	User clicks on view comments button.	System displays the comments of reviewer.
<b>Comments:</b> The test case is passed. Our system is working according to our need.		
		<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed

### 6.2.18 Reject Book Test Case

Book Rejection			
Test Case ID:	18	QA Test Engineer:	Muhammad Ahmad
Test case Version:	1	Reviewed By:	Usama Iqbal
Test Date:	18-04-2023	Use Case Reference(s):	Reject book use case (19)
<b>Revision History:</b>	None		
<b>Objective</b>	To check if admin can reject a book		
<b>Product/Ver/Module:</b>	Admin Module of website		
<b>Environment:</b>	Website is up and running		
<b>Assumptions:</b>	Reject button is visible to the user		
<b>Pre-Requisite:</b>	The admin is already logged into their account and on the manage books page		
Step No.	Execution description	Procedure result	
1	User clicks on reject book button.	System stores the information against the book.	
<b>Comments:</b> The test case is passed. Our system is working according to our need.			
		<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed	

### 6.2.19 Block User Test Case

Block User			
Test Case ID:	19	QA Test Engineer:	Muhammad Ahmad
Test case Version:	1	Reviewed By:	Usama Iqbal
Test Date:	18-04-2023	Use Case Reference(s):	Block user use case (20)
<b>Revision History:</b>	None		
<b>Objective</b>	To check if admin can block a user		
<b>Product/Ver/Module:</b>	Admin Module of website		
<b>Environment:</b>	Website is up and running		
<b>Assumptions:</b>	Block button is visible to the user		
<b>Pre-Requisite:</b>	The admin is already logged into their account and one the manage users page		
Step No.	Execution description	Procedure result	

<b>1</b>	<i>User clicks in block button.</i>	<i>System block the user and stores the information.</i>	
<b>Comments:</b> The test case is passed. Our system is working according to our need.			
	<input checked="" type="checkbox"/> Passed	<input type="checkbox"/> Failed	<input type="checkbox"/> Not Executed

### 6.2.20 Assign Book Test Case

Assign Book to Reviewer			
<b>Test Case ID:</b>	20	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Usama Iqbal</i>
<b>Test Date:</b>	18-04-2023	<b>Use Case Reference(s):</b>	<i>Assign book to reviewer use case (21)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if admin assign a book to a reviewer</i>		
<b>Product/Ver/Module:</b>	<i>Admin Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		
<b>Assumptions:</b>	<i>Assign reviewer button is visible to the user</i>		
<b>Pre-Requisite:</b>	<i>The admin is already logged into their account and on manage books page</i>		
Step No.	Execution description	Procedure result	
1	<i>User clicks on assign reviewer button.</i>	<i>System shows a list of reviewers.</i>	
2	<i>User selects the reviewer and clicks on save.</i>	<i>System assigns the book to the reviewer.</i>	
<b>Comments:</b> The test case is passed. Our system is working according to our need.			
	<input checked="" type="checkbox"/> Passed	<input type="checkbox"/> Failed	<input type="checkbox"/> Not Executed

### 6.2.21 Notify a Reviewer Test Case

Notify a Reviewer			
<b>Test Case ID:</b>	21	<b>QA Test Engineer:</b>	<i>Muhammad Ahmad</i>
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	<i>Usama Iqbal</i>
<b>Test Date:</b>	18-04-2023	<b>Use Case Reference(s):</b>	<i>Notify a reviewer use case (22)</i>
<b>Revision History:</b>	<i>None</i>		
<b>Objective</b>	<i>To check if admin can notify a reviewer when deadline is near</i>		
<b>Product/Ver/Module:</b>	<i>Admin Module of website</i>		
<b>Environment:</b>	<i>Website is up and running</i>		
<b>Assumptions:</b>	<i>Notify Reviewer button is visible to the user</i>		
<b>Pre-Requisite:</b>	<i>The admin is already logged into their account and on manage books page</i>		

Step No.	Execution description	Procedure result
1	User clicks on view list of reviewers	System shows a list of reviewers.
2	User selects notify button	System notifies the reviewer.
<b>Comments:</b> The test case is passed. Our system is working according to our need.		
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed		

## 6.2.22 Extend the Deadline Test Case

Extend the Deadline					
<b>Test Case ID:</b>	22	<b>QA Test Engineer:</b>	Muhammad Ahmad		
<b>Test case Version:</b>	1	<b>Reviewed By:</b>	Usama Iqbal		
<b>Test Date:</b>	18-04-2023	<b>Use Case Reference(s):</b>	Extend the deadline use case (23)		
<b>Revision History:</b>	None				
<b>Objective</b>	To check if admin can extend the deadline for an assignment				
<b>Product/Ver/Module:</b>	Admin Module of website				
<b>Environment:</b>	Website is up and running				
<b>Assumptions:</b>	Extend the Deadline button is visible to the user				
<b>Pre-Requisite:</b>	The admin is already logged into their account and on manage books page				
Step No.	Execution description	Procedure result			
1	User clicks on view list of pending books	System shows list of books.			
2	User selects extend the deadline button in front of book	System extends the deadline.			
<b>Comments:</b> The test case is passed. Our system is working according to our need.					
<input checked="" type="checkbox"/> Passed <input type="checkbox"/> Failed <input type="checkbox"/> Not Executed					

## 6.3 Test Metrics

The summary of the common ground of attributes of test case metrics is given below.

### 6.3.1 Test case Metric.No.1

Metric	Purpose
<b>Number of Test Cases</b>	20
<b>Number of Test Cases Passed</b>	20
<b>Number of Test Cases Failed</b>	0
<b>Test Case Defect Density</b>	0/10
<b>Test Case Effectiveness</b>	10/10
<b>Traceability Matrix</b>	It is provided in a separate file.

## Chapter 7: User Manual

### 7.1 Login

1. Find the Login button on the website by visiting it. This is usually found in the page's upper right corner.
2. To open the login form, click the Login button.
3. Fill up the boxes with your email address and password. To reset your password if you've forgotten it, click the "Forgot password" link.
4. To access your account after entering your login information, click the "Login" button. You can examine your account information, edit your profile, and change your preferences on your personalized dashboard after logging in.
5. Make sure you log out by clicking the "Logout" button in the top-right corner of the page once you've finished using the website. By doing this, you can make sure that your account is secure and that no one else has access to your personal data.

### 7.2 Sign-Up

1. Find the Signup button on the website. This is usually found in the page's upper right corner.
2. To open the signup form, click the Signup button.
3. Include all necessary details, including your name, email address, role and password, in the registration form. Make sure your password is strong and consists of a combination of upper- and lowercase letters, digits, and symbols.
4. After you've completed all the required fields, click "Signup" to start the account-creation process.
5. Your browser will redirect you to the login page after you create your account. For account login, enter your email address and password.
6. You may examine your account information, edit your profile, and adjust your preferences on your personalized dashboard once you've logged in.
7. Make sure you log out by clicking the "Logout" button in the top-right corner of the page once you've finished using the website. By doing this, you can make sure that your account is secure and that no one else has access to your personal data.

### 7.3 Edit Profile

1. Make use of your email address and password to log in to your account.
2. Find the "Edit Profile" button after logging in. This is usually found in the page's upper right corner.
3. To access your profile settings, click the "Edit Profile" button.
4. Here, you can change your name and password among other personal data. Additionally, you might be able to update your password or add a profile image.
5. When you've finished the required updates, click "Save" or "Update" to keep your changes.

### 7.4 Upload a Book

1. Make use of your email address and password to log in to your account. Check to see if you have created a writer account on our website.
2. You'll be taken to your writer's home page once you check in.
3. Find the "Upload Book" button on the writer's profile page.
4. To access the book upload form, click the "Upload Book" button.

5. Enter the required data in the upload form, including the book's title, author, and description. To increase the likelihood that readers will find your book, please be sure to include accurate and thorough information.
6. By choosing the "Choose File" or "Attach File" option, you can attach your book's PDF file. You will be asked to choose the file using the file explorer on your PC.
7. After attaching your book's PDF file, click the "Upload" button to upload it to our website.
8. Watch for the upload to be finished. Depending on the size of your book file and the speed of your internet connection, this could take a while.
9. You'll see a confirmation message on the screen letting you know that your book has been successfully uploaded after the upload procedure is finished.

## 7.5 Resubmit a Book

1. Make use of your email address and password to log in to your account. Ensure that the book you want to resubmit has already been uploaded.
2. Locate the book you want to resubmit after logging in. On your writer home page, go to the "My Books" or "My Uploads" section to accomplish this.
3. To access the book details page for the book you want to resubmit, click on it.
4. Find the "Resubmit" button on the page with the book details. This is often found on the page in a prominent place or close to the book cover image. The new book title, author, book description, and any other pertinent information should be filled out in the resubmit form. To increase the likelihood that readers will find your book, please be sure to include accurate and thorough information.
5. By selecting the "Attach File" or "Choose File" button, you can attach the most recent PDF version of your book. You'll be asked to choose the updated file from the file explorer on your PC.
6. Click "Resubmit" to upload your modified book to our website after attaching the new PDF version of your book. Watch for the resubmission procedure to finish. Depending on the size of your updated book file and the speed of your internet connection, this could take a while.
7. You'll see a confirmation notice on the screen after the resubmit procedure is finished letting you know that your revised book was successfully resubmitted.

## 7.6 Review a Book

1. Make use of your email address and password to log in to your account. Make sure you've created an account on our website as a reviewer.
2. Your reviewer home page will be displayed once you have successfully checked in.
3. Find the "Book List" section on the reviewer's home page.
4. To see the list of books that are available for review, click on the "Book list" section.
5. By selecting the "review" button, choose the book you wish to review from the list.
6. You may discover the book description, author name, and other pertinent details on the book details page.
7. To get the book in PDF format, click the "Read Book" button.
8. Carefully read the book, making notes on any points you believe should be covered in your review.
9. After finishing the book, go back to the page with the book's details and look for the "Review" button.
10. To access the book review form, click the "Review" button.
11. The book title, author name, and your review are all required fields in the review form.

12. Please make sure your review is truthful, balanced, and based on your own judgment of the book.
13. To submit a review to our website, click the "Submit Review" button.

## 7.7 Invest in a Book

1. Make use of your email address and password to log in to your account. Verify that you have created a user account on our website for investors.
2. Your investor home page will be displayed once you have successfully logged in.
3. Find the "Invest in Books" section on the investor home page.
4. To view the list of books that are up for investment, select the "Invest" section.
5. Click on a book's title or cover image on the list to choose the one you wish to purchase.
6. You may discover the book description, author name, and other pertinent details on the book details page. To decide whether you want to purchase the book, carefully consider the rating and reviews.
7. To access the investment form if you choose to invest in the book, click the "Invest" button.
8. Fill out the investment form with the essential details, including the amount invested, payment information, and any other pertinent information.
9. Please be sure to fill out the investment form completely and accurately.
10. To submit your investment to our website, click the "Submit Investment" button.

## Chapter 8: Conclusion and Future Work

The scope of our project was fully achieved. During implementation, the connectivity of front-end to back-end generated errors which were resolved by us. Moreover, watermarking the books was a hard part to do for large sized books. The watermarks were not shown at first and after that these watermarks were hiding the text below. Moreover, the recommendation system was incorporated. The data for training of model of recommendation system was not found and therefore we made that data synthetically. Moreover, finding the API for plagiarism detector was a hectic process but this issue was resolved at last. All the issues mentioned above were resolved by us and these issues are no more present in our project.

The first chapter of this report gives the introduction regarding the Book Reviewing and Investment Platform being implemented and describes the purpose of this document, the intended audience, and the definitions and abbreviations of complex terms. The second chapter discusses the vision of the project, along with the problem statement and solution to it. It also highlights the project and sustainable development goals of the project. In addition to that, constraints, business opportunities, stakeholder's description and high-level goals are also discussed in this chapter. In the third chapter, literature review and summary tables of literature review are discussed. Moreover, all of the software requirements specification and a list of features are included in the fourth chapter. Along with the quality attributes and assumptions, it also addresses the functional and non-functional requirements. It also includes the software and hardware requirements. The database design, which primarily includes an ER diagram and a data dictionary, and user cases with GUI make up the majority of what it consists of. The Risk analysis is also included in this chapter. Furthermore, in fifth chapter, system overview is provided along with the factors we have to keep in mind while making the design of this project. The goals and objectives of the design are also discussed in this chapter. Lastly, this chapter provides the system architecture and its components along with class and sequence diagrams, and policies and tactics that changed the interface of the system. In the sixth chapter of this report, i.e., the implementation part provides all the implementation details we have done so far such as that of Authentication, Recommendation system, Watermarking and Plagiarism detector. Moreover, it also provide test cases and test metrics details of each test that we performed for ensuring the quality of our website. All the test case scenarios and fulfilment criteria is mentioned there. After that this report provides user manual for users of our application. It provides them the steps on how to do certain things such as Login, Sign Up, Edit Profile, Upload a Book, Invest in a Book and Resubmit a Book, etc.

This project can be further enhanced by automating the work as much as possible and by minimizing the involvement of administrator. The new remote publishers can be added to system through sign up and login and can publish a book. Moreover, the reviewers can be automatically assigned which has not been done this time. Moreover, the authentication methods such as Google sign in and Facebook Sign in will be incorporated. Moreover, the payment by the investor can be given to writer without involvement of administrator and all this process will be done without any human involvement. Moreover, this system can also be enhanced into an application to increase the mobility of usage and for ease of users which will further increase the usage of the system and will achieve its goals hassle free.

## References

- [1] “Booklist,” <https://www.booklistonline.com/writing-for-booklist>, 2022.
- [2] Kate Stead, “IngramSpark interview: Answers to your self-publishing questions,” <https://oldmatemedia.com/ingramspark-interview/>, September 7, 2021.
- [3] “Self-Publish Like a Pro,” <https://www.ingramspark.com/how-it-works?hsCtaTracking=a8727cce-a303-4fb2-901e-4301a1846ca1%7C605fc5ea-6452-41a6-b806-ba352ab811e2>.
- [4] Dane Neller, “First Espresso Book Machine Installed and Demonstrated at New York Public Library's Science, Industry and Business Library,” <http://www.prweb.com/releases/2007/06/prweb534914.htm>, October 12, 2022.
- [5] Casey, “Pros and Cons of Print on Demand,” <https://lucidbooks.com/pros-and-cons-of-print-on-demand/>, August 22, 2018.