# XRCVC - eLibrary

## Project Report submitted to
## St. Xavier's College – Autonomous Mumbai

**For the partial fulfilment for the award of the degree of**
**Bachelor of Science (BSc) in**
**Information Technology**

**By**
**VARUN MANOJ KUMAR**
**(UID No. : 205118)**

**Under the Supervision of**
**Prof. LYDIA FERNANDES**

**INFORMATION TECHNOLOGY**
**ST. XAVIER'S COLLEGE (AUTONOMOUS),**
**MUMBAI-400001, INDIA**
**April 2023**

# PROJECT CERTIFICATE

This is to certify that the project entitled **"XRCVC - eLibrary"** undertaken at the Information Technology Department of St. Xavier's College – Autonomous Mumbai has been submitted by: **VARUN MANOJ KUMAR (UID No: 205118)** in partial fulfilment of Bachelor's in Information Technology degree (Semester VI) Examination. It is further certified that he has completed all required phases of the project.

| | | |
|---|---|---|
| Signature | Signature | Signature |
| (Internal Guide) | ( Internal Examiner) | ( External Examiner) |

Signature College Seal

(HOD – Information Technology department)

# Student Declaration

I, **VARUN MANOJ KUMAR (UID No: 205118)**, do hereby, certify that:

1) that the project report titled, **"XRCVC - eLibrary"** which is being submitted in partial fulfilment of the requirements for the Degree of Bachelor of Science with a specialization in Information Technology is the result of the **original work** carried out by me under the guidance of the Ms. Lydia Fernandes, faculty of Information Technology Department, St. Xavier's College, Mumbai-01.
2) This project has not previously formed the basis for the award of any degree, diploma, or certificate of this college or of any other college or university.
3) Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them by providing references of them in the project documentation.
4) From the plagiarism test, it is found that the similarity index of whole submission is within 10 %. [optional]

Date:

Place:

_____

VARUN MANOJ KUMAR

UID : 205118

Table of Contents


1 INTRODUCTION ..... 1

1.1 ORGANIZATIONAL PROFILE ..... 1

1.2 INTRODUCTION TO PROJECT ..... 3

2 SYSTEM STUDY ..... 4

2.1 OVERVIEW ..... 4

2.2 CURRENT SYSTEM & LIMITATIONS ..... 5

2.3 PROPOSED SYSTEM & ADVANTAGES ..... 5

3 SYSTEM ANALYSIS ..... 7

3.1 PROJECT PLAN & GANTT CHART ..... 7

3.2 USE CASE DIAGRAM ..... 8

3.3 ACTIVITY DIAGRAM ..... 10

3.4 SEQUENCE DIAGRAM ..... 11

3.5 DEPLOYMENT DIAGRAM ..... 12

3.6 MENU TREE ..... 13

4 SYSTEM DESIGN ..... 15

4.1 SDLC PROCESS USED ..... 15

4.2 FIGMA DESIGN – WIREFRAME DIAGRAM ..... 17

4.3 ACCESSIBLE DESIGN ..... 22

4.4 WHAT WE HAVE DONE TO MAKE SITE ACCESSIBLE ..... 23

5 FEASIBILITY STUDY ..... 26

5.1 TECHNICAL FEASIBILITY ..... 26

5.2 ECONOMIC FEASIBILITY ..... 27

6 TECHNOLOGY USED ..... 28

6.1 ARCHITECTURE DIAGRAM ..... 28

6.2 DOMAIN HOSTING ..... 28

6.3 WORDPRESS[2] ..... 29

6.4 JQUERY ..... 33

6.5 LiteSpeed WebServer ..... 34
6.6 Android with Java ..... 34
6.7 Firebase Backend Services ..... 35

7 TABLES AND DATA STRUCTURE ..... 36

7.1 WordPress Database ER Diagram [1] ..... 36
7.2 WordPress Database Table Structure ..... 37
7.3 Additional Custom Tables ..... 46

8 MODULES & SCREEN LAYOUTS ..... 49

8.1 Student Book Request Modules ..... 49
8.2 Admin Modules (Frontend) ..... 68
8.3 Admin Module (Backend -WordPress) ..... 75
8.4 Android App ..... 81
8.5 Validation ..... 96

9 REPORTS ..... 98

9.1 Total Book Requests ..... 98
9.2 Book Requests per Student ..... 99
9.3 Book Request Status ..... 100

10 SYSTEM IMPLEMENTATION ..... 101

10.1 Source Code ..... 101
10.2 Unit Testing ..... 150
10.3 Accessibility Test Reports (Lighthouse) ..... 153

11 CONCLUSION AND FUTURE ENHANCEMENT ..... 158

12 TESTIMONIALS ..... 159

13 REFERENCES ..... 161

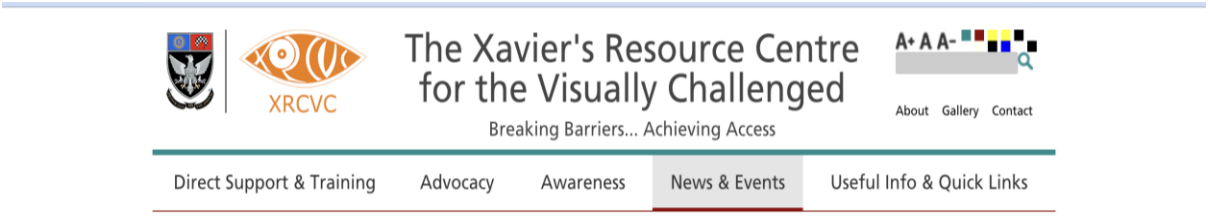
13.1 For Website ..... 161
13.2 For Android App ..... 161
13.3 For Accessibility ..... 161
13.4 For Documentation ..... 161

**14 BIBLIOGRAPHY ..... 161**

# 1 Introduction

## 1.1 Organizational Profile

### 1.1.1 About XRCVC




The Xavier's Resource Centre for the Visually Challenged (XRCVC) is an integral part and department of St. Xavier's College - Autonomous, Mumbai.

The XRCVC was started in 2003 as an effort to ensure an inclusive environment at St. Xavier's College, for its students with blindness and low vision. Having created an inclusive set-up for its own students, keeping with the college's long tradition of creating social impact within the larger community, the XRCVC has today become a national advocacy and support centre for the blind and low-vision across the city and the country. The XRCVC is committed to giving the best to the students and is always exploring possibilities of opening up more facilities for students.

Access to information for education, leisure or work is one of the biggest requirements for all students especially for persons with print disabilities. Persons with blindness, weak eyesight or any other disability cannot read the standard print books and they need books in different formats. The XRCVC has books in different accessible formats. These include DAISY audio, DAISY full text, EPUB, Word format (DOC/DOCX), HTML, BRF (Braille ready files), Text and RTF (Braille ready files).

The XRCVC through its years of work has an existing database of a rich resource bank. It has a catalogue of books in the e-Library. These books are made available to the students in collaboration with Sugamaya Pusthakalaya.

### 1.1.2 About Sugamaya Pusthakalaya

Sugamaya Pusthakalaya is India 's first and largest online library for collection of accessible books. If you are looking for books accessible to the blind, people with low vision or to persons with any other print disability, this library of libraries is your one-stop shop.

Sugamya Pustakalaya is a collaborative effort of TCS, Daisy Forum of India and NIEPVD(National Institute for Empowerment of Persons with Visual Disabilities) and Government of India to end the book famine faced by people with print disabilities.

Here, you can access books in diverse languages from various libraries across India. We have also partnered with international agencies like Bookshare and Accessible Books Consortium to provide you with accessible books from all over the world.

Users can access a collection of over 6,53,234 books, maintain their individual reading shelves online and also download books in chosen formats.

Sugamya Pustakalaya is a National Library for Persons with Disabilities.

## 1.2 Introduction to Project

Currently the XRCVC does not have its own system to manage their books and library. Our purpose of this project is to design and develop an e-library system for the XRCVC.

The XRCVC e-Library System project aims to design and develop a comprehensive and user-friendly digital library system for XRCVC. The project seeks to replace the current manual system of managing books with an automated system that will help both students and staff to easily manage and retrieve information about the books available in the library.

As the name suggests, the e-Library System project has been developed as an online digital library to store the information about books in the library electronically. Users would be able to request books from the XRCVC and download accessible books directly from Sugamya Pustakalaya. Along with the online website, an Android App has been developed for quick access to the library from the mobile phones.

The primary benefit of the e-Library System is that it will reduce manual work to a great extent and streamline library activities. Students will be able to search for and request books online, while staff members will be able to manage book requests and update the library catalogue in real-time. This will significantly improve the efficiency and effectiveness of the library operations.

# 2 System Study

## 2.1 Overview

An online website for the e-Library has been developed that could be accessed via the XRCVC website. All the book information would be maintained in a Library Database. Students would be able to browse the website and search for the required books. They can then fill in a request form and request for the books that they would like to read. Administrators would be able to view these book requests and provide the details. They would also be able Add/Edit/Delete Book Details using the website.

The entire website would be Accessible so that users with disabilities would be able to browse the site easily using assistive technologies like screen readers.

Additionally, an Android App also has been developed for students and staff to quickly access the website from their mobile phones.

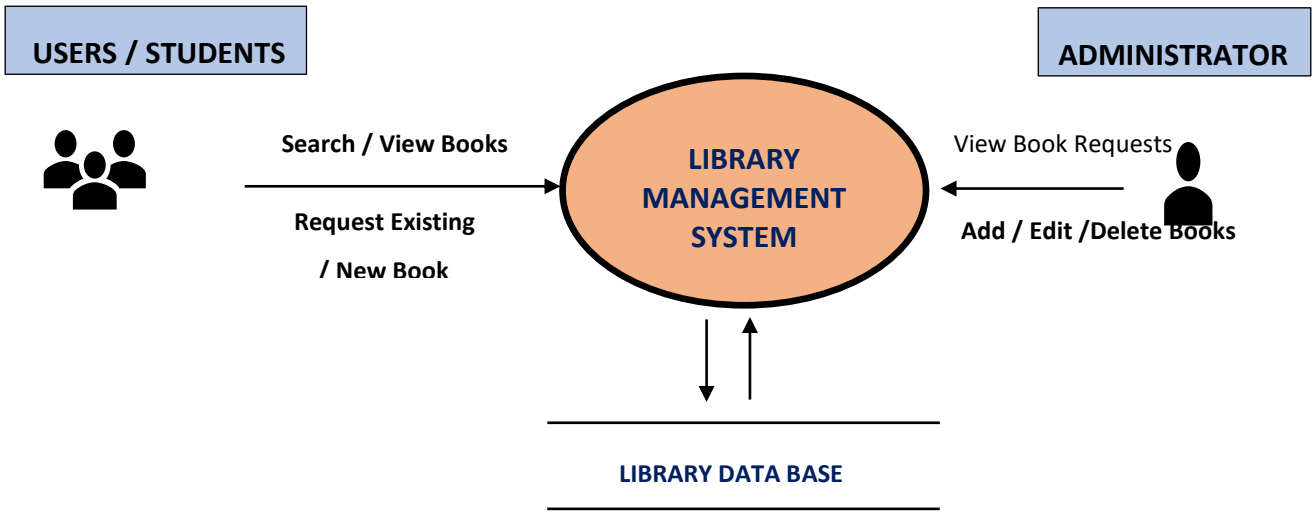



*Figure 2-1 Data Flow Diagram*

## 2.2 Current System & Limitations

At Present XRCVC uses an Excel based database containing list of all the books that XRCVC offers at the Centre. This Excel spreadsheet is uploaded and updated regularly on XRCVC's Website.

There are several limitations to the present system which are listed below:

- Every time users have to download the spreadsheet to get the updated list from the website to view and request books.
- Library catalogue does not get updated in real time so users may not be able to view the most up-to-date list of books.
- Users need to manually send emails to XRCVC and there is no automated system for managing book requests from users.
- This makes it very cumbersome and difficult for both users and staff members to manage the list of books and requests made by the user.
- Moreover, if students have any specific books that needs to be converted to accessible formats, the current system does not accommodate for such a requirement. Students must send emails personally to the XRCVC.
- An Excel based database does not Account for a robust, sorting, searching and filtering mechanism which makes it time consuming for users to get the desired book from XRCVC.

Overall, the current process is time-consuming and inefficient as there is no centralized system for managing requests. This causes significant inconvenience and delay for students seeking accessible books.

## 2.3 Proposed System & Advantages

The proposed centralized online platform for XRCVC's library of accessible books aims to address the limitations of the current system and provide a more efficient and user-friendly experience for both staff and users. This project implements an online website that allows for real-time updates, automated book requests, and efficient management of requests for converting specific new books to accessible formats. Additionally, a more user-friendly interface with robust search and filtering capabilities would make it easier for students to find the books they need.

Some of the key features of the platform include:

- A comprehensive database of available books with information on authors, publishers, ISBN, formats, and language.
- A search system for the e-library that allows users to quickly find the books they need.

- An easy-to-use administration system that allows the XRCVC staff to add, update, and delete information in the system as needed.
- Elimination of the need for users to download an Excel spreadsheet to view the book list.
- Real-time updates of the library catalogue, ensuring that users always have access to the most up-to-date information.
- An automated system for new book requests that eliminates the need for users to send emails to the XRCVC.
- The ability for users to download accessible books from the National Library for persons with disabilities, Sugamya Pustakalaya.

By implementing these features, the new platform would significantly reduce the inconvenience and delay experienced by students seeking accessible books. Users would be able to search and request books online, and the real-time updates would ensure that they always have access to the latest information. Overall, the new platform would provide a more efficient, user-friendly, and accessible experience for students with disabilities who rely on XRCVC's library services.

# 3 System Analysis

## 3.1 Project Plan & Gantt Chart

The project plan and timelines are as given in the Gantt Chart below.

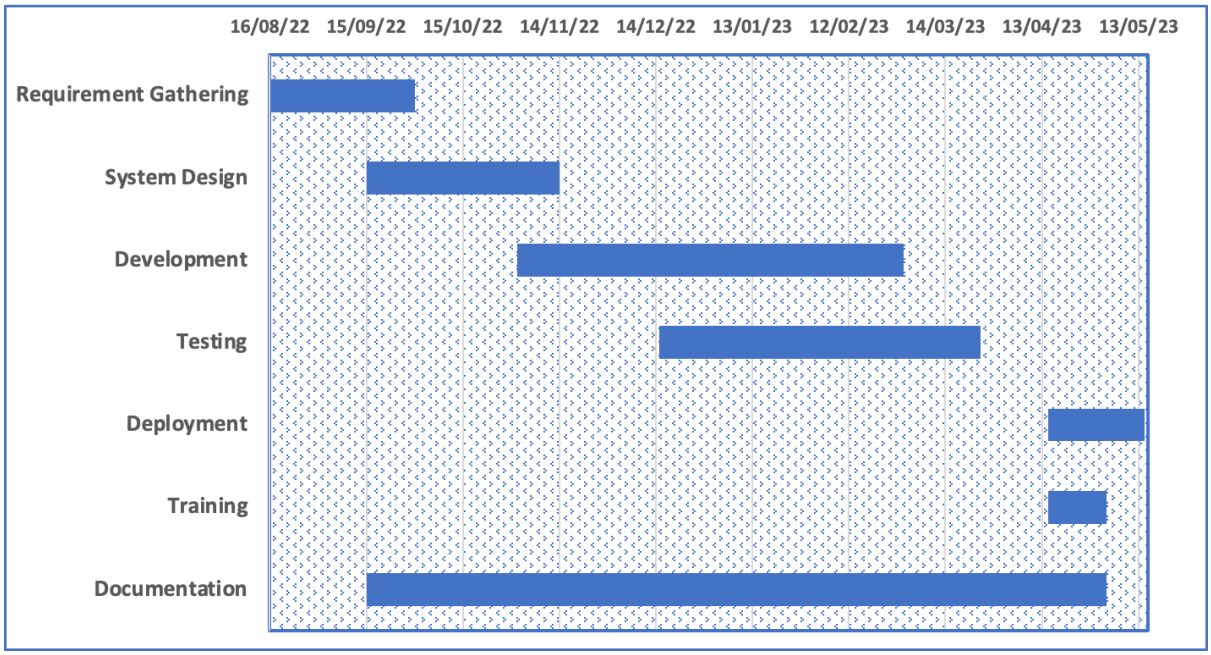



*Figure 3-1 Gantt Chart*

## 3.2 Use Case Diagram

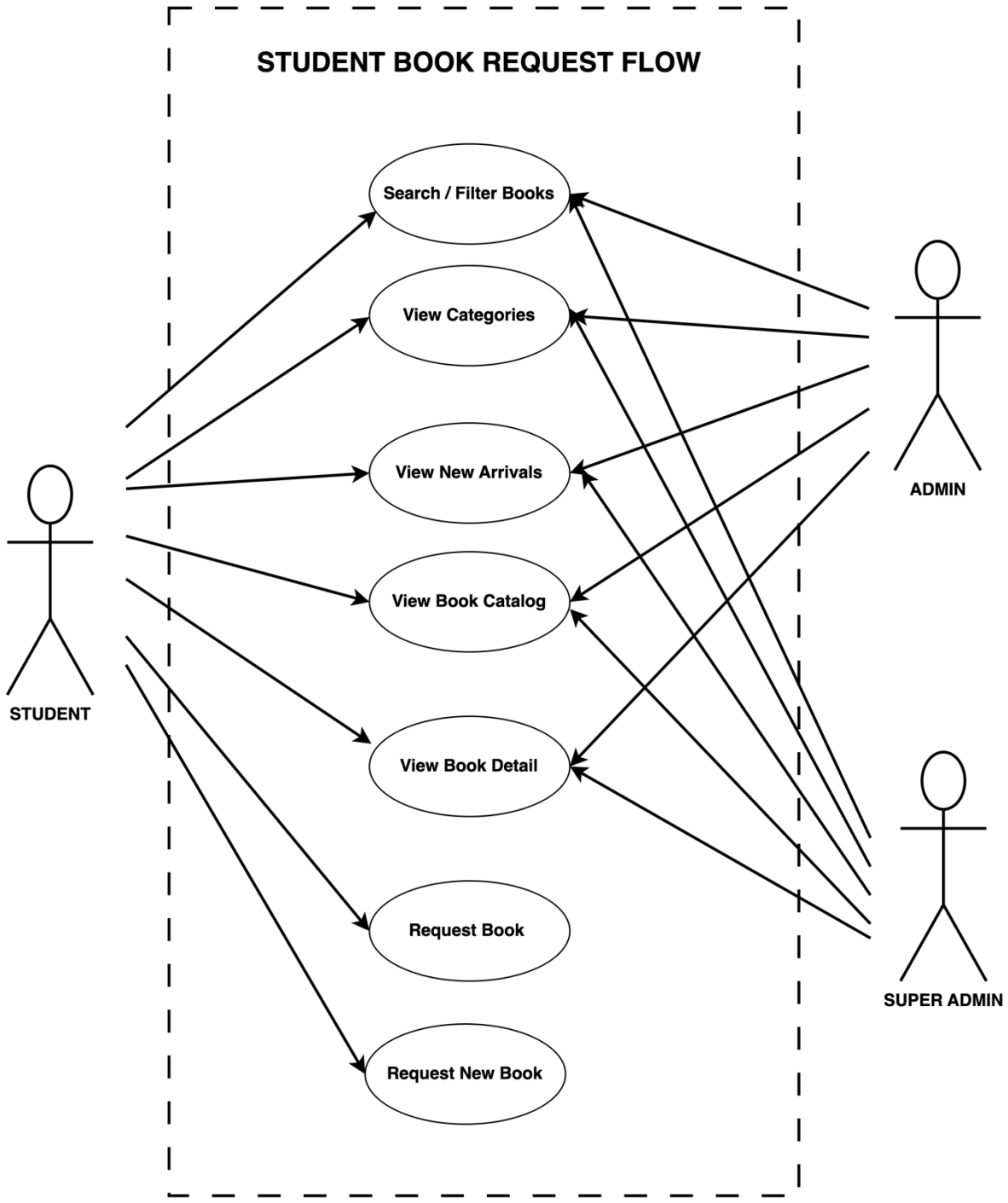



*Figure 3-2 Student Book Request Flow use case*

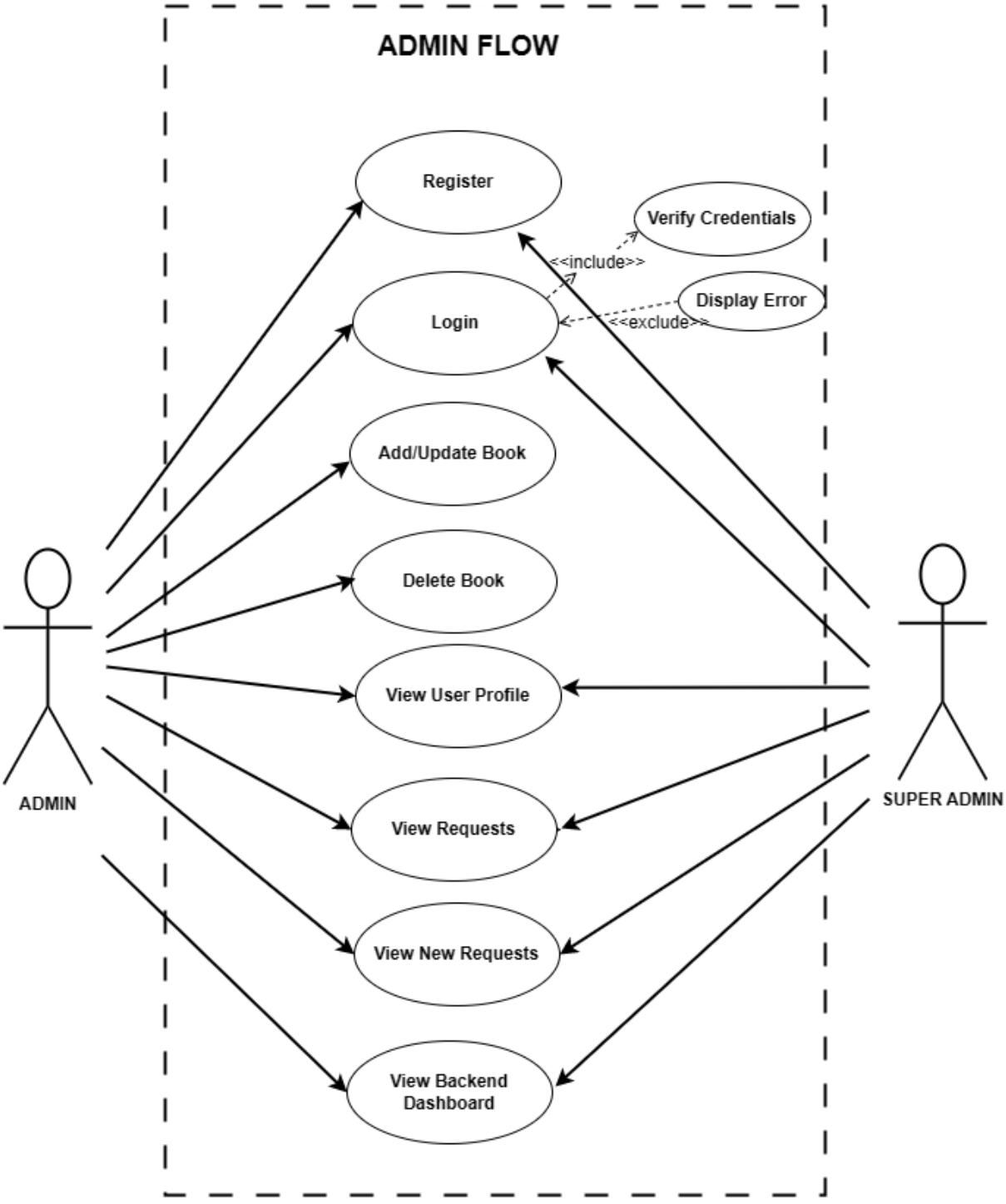



*Figure 3-3 Admin Flow Use Case*

## 3.3 Activity Diagram

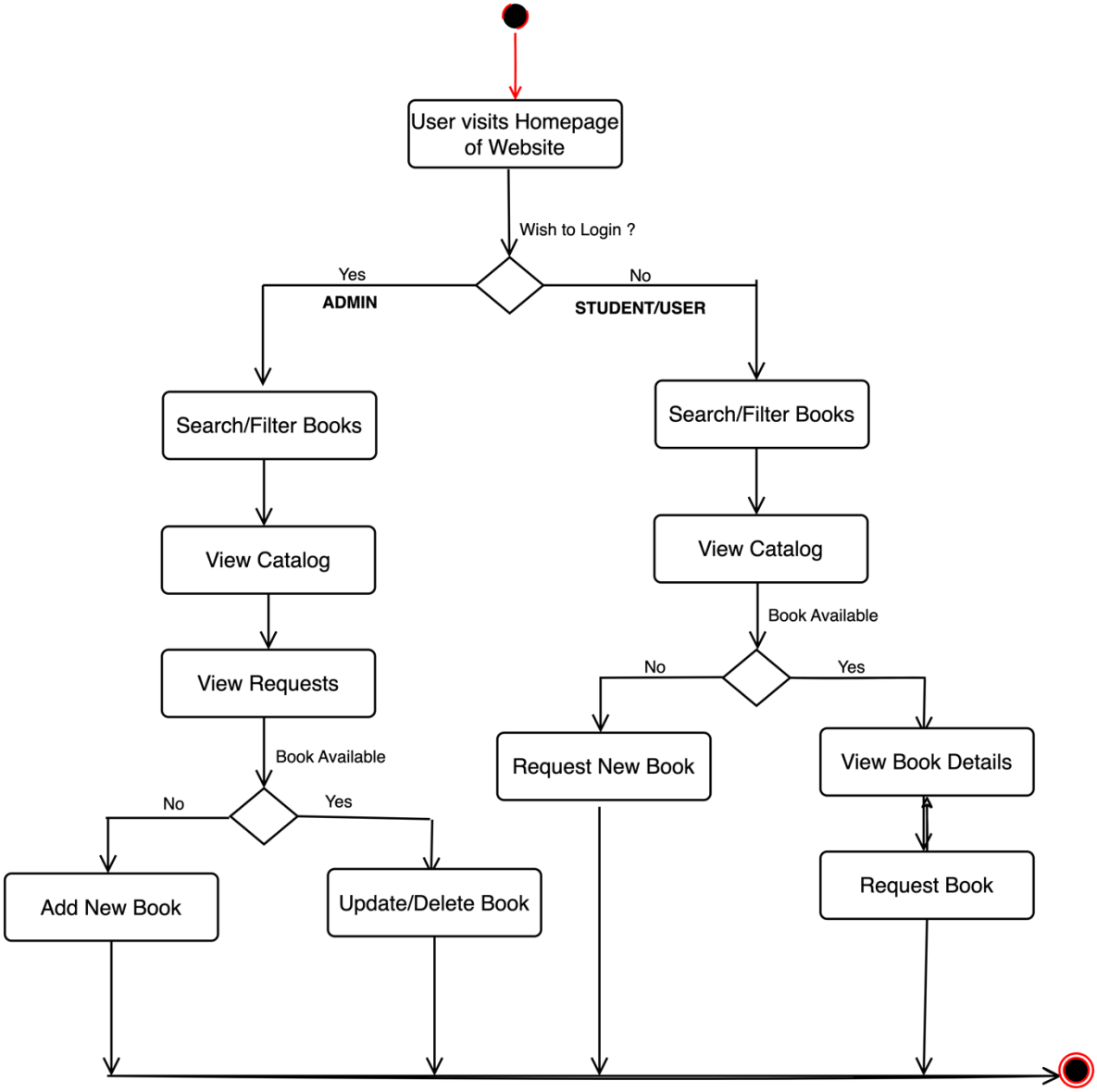



*Figure 3-4 Activity Diagram*

## 3.4 Sequence Diagram

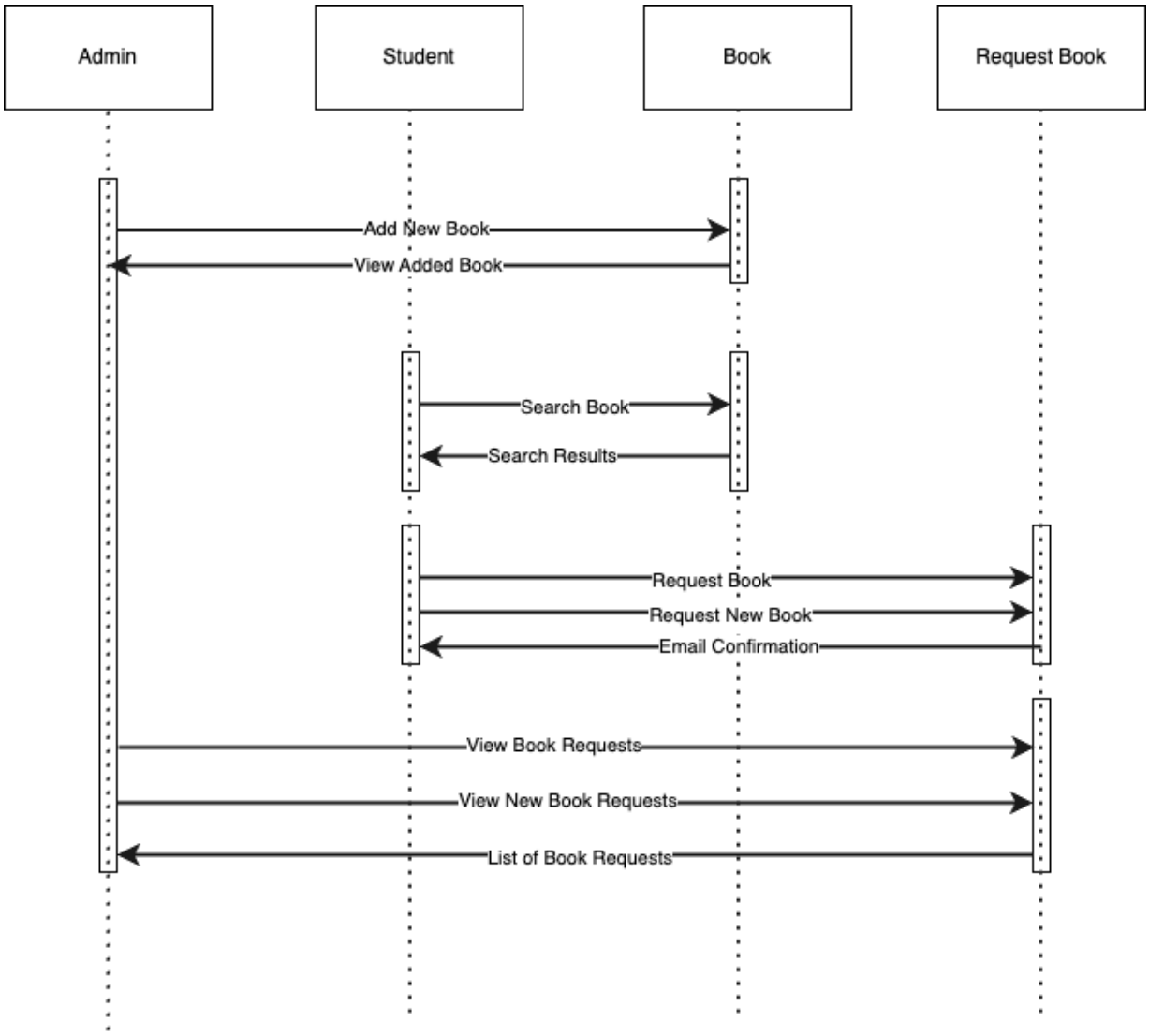



*Figure 3-5 Sequence Diagram*

## 3.5 Deployment Diagram

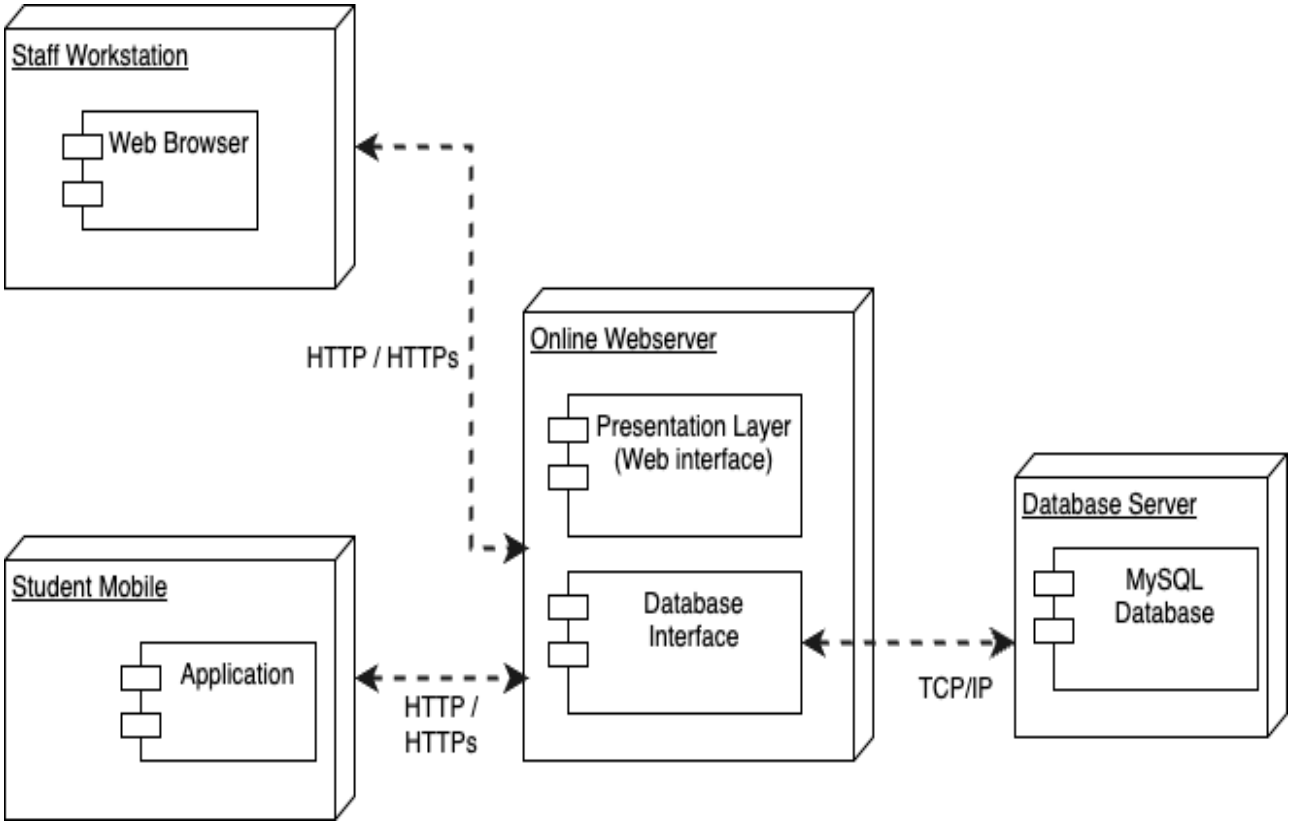



*Figure 3-6 Deployment Diagram*

## 3.6 Menu Tree

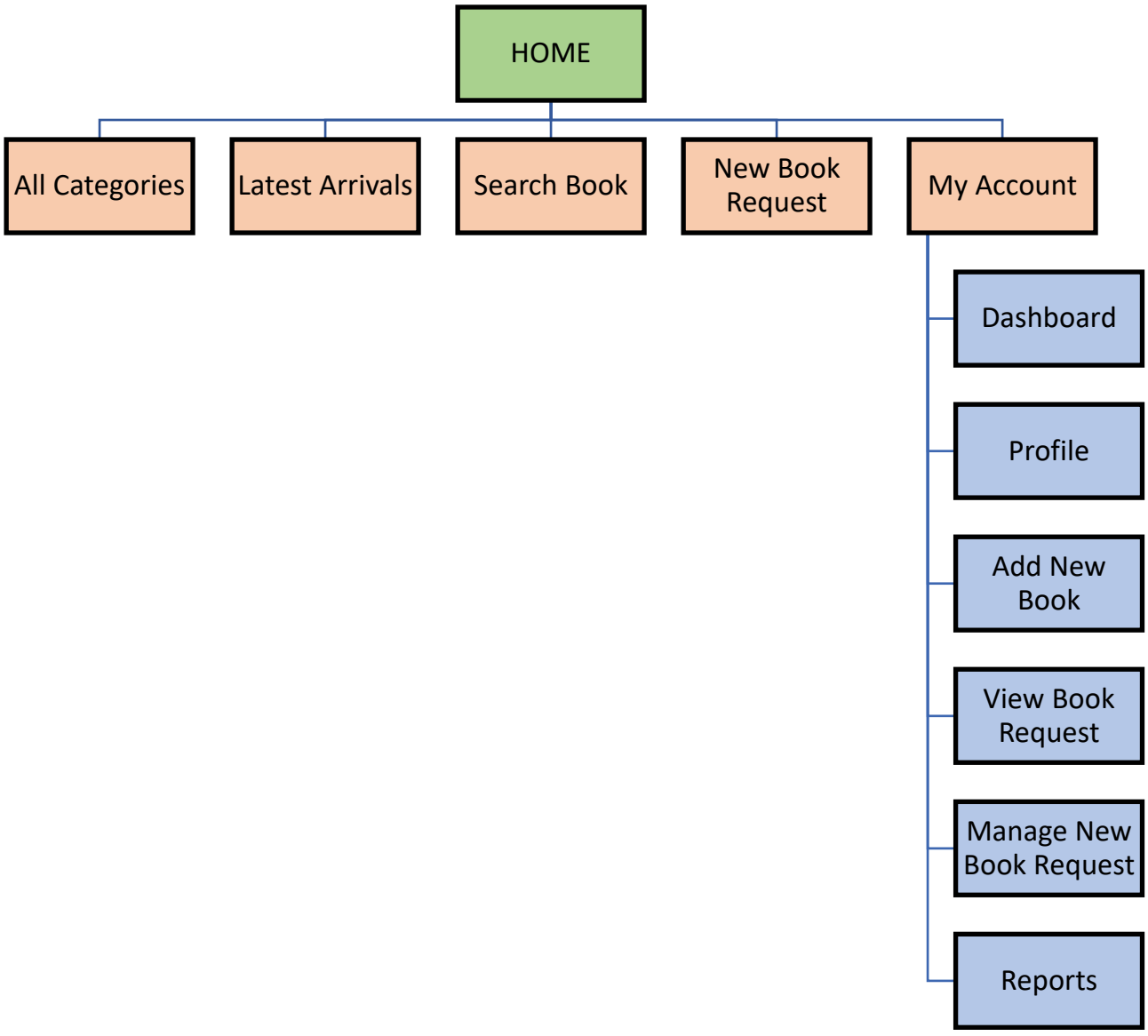
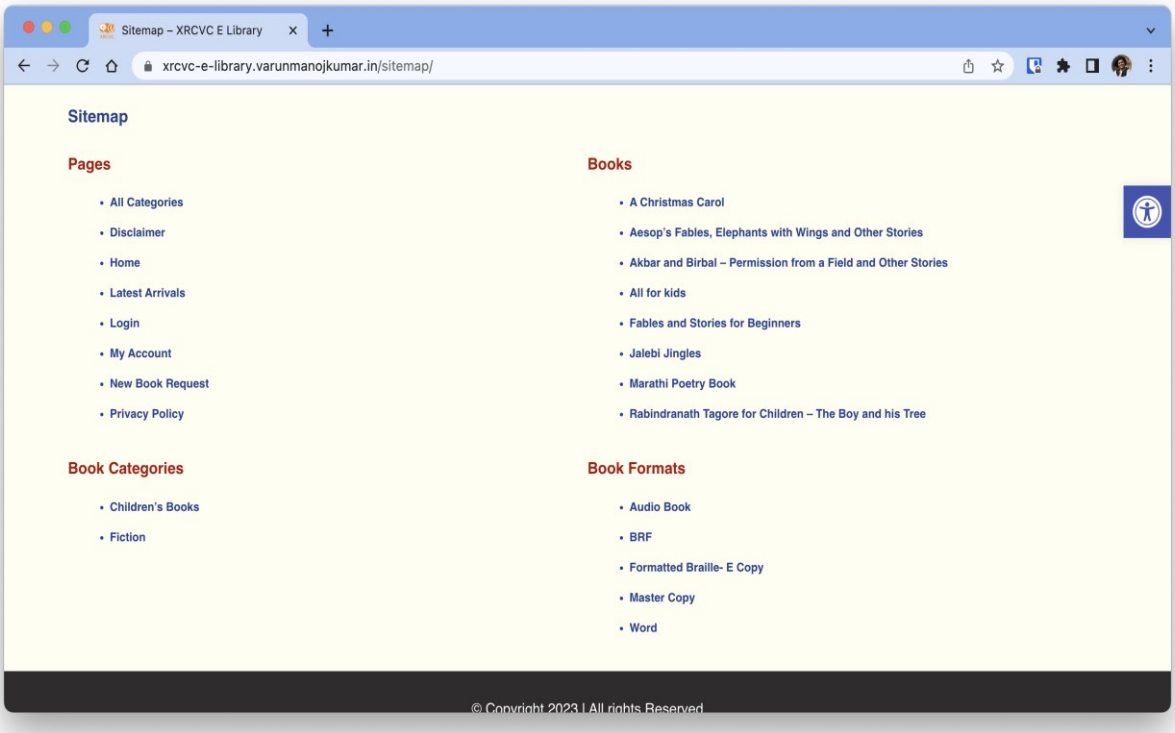



*Figure 3-7 Website Menu Tree*

Site Map


Sitemap – XRCVC E Library
xrcvc-e-library.varunmanojkumar.in/sitemap/
Sitemap
Pages
All Categories
Disclaimer
Home
Latest Arrivals
Login
My Account
New Book Request
Privacy Policy
Books
A Christmas Carol
Aesop's Fables, Elephants with Wings and Other Stories
Akbar and Birbal – Permission from a Field and Other Stories
All for kids
Fables and Stories for Beginners
Jalebi Jingles
Marathi Poetry Book
Rabindranath Tagore for Children – The Boy and his Tree
Book Categories
Children's Books
Fiction
Book Formats
Audio Book
BRF
Formatted Braille- E Copy
Master Copy
Word
© Copyright 2023 | All rights Reserved

# 4 System Design

## 4.1 SDLC Process Used

In this project, the SPIRAL MODEL of SDLC has been followed.

Originally proposed by Barry Boehm, the *spiral model* is an evolutionary software process model that couples the iterative nature of prototyping with the controlled and systematic aspects of the waterfall model. It provides the potential for rapid development of increasingly more complete versions of the software.

The spiral development model is a *risk*-driven *process model* generator that is used to guide multi-stakeholder concurrent engineering of software intensive systems. It has two main distinguishing features. One is a *cyclic* approach for incrementally growing a system's degree of definition and implementation while decreasing its degree of risk. The other is a set of *anchor point milestones* for ensuring stakeholder commitment to feasible and mutually satisfactory system solutions.

Using the spiral model, software is developed in a series of evolutionary releases. During early iterations, the release might be a model or prototype. During later iterations, increasingly more complete versions of the engineered system are produced.

Spiral Process Model

**Phases of Spiral Model**

In the spiral model each cycle or iteration has to pass through the four general steps which are as follows:

1. **Risk Analysis & Planning**

Developers identify risks involved in the current iteration and also corrective measures to minimize risk. Then they evaluate corrective measures against objectives and constraints. Thus the deadline is set for the next stage.

2. **Requirement Analysis**

This phase analyses the client's requirements. Also, the requirements for developing the product are also established.

3. **Coding & Testing**

Developers develop multiple programs and integrated them to form a software or prototype. Like, in the early cycles the product of this stage would be a prototype. In subsequent cycles, the product of this stage becomes **developed software**.

The product of this stage is tested to find any error in coding.

4. **Evaluation**

This stage evaluates whether the project is going as per planning or not. It evaluates whether the objective set at the first stage has been achieved or not. The evaluation phase also helps developers decide the number of cycles required to complete the project.

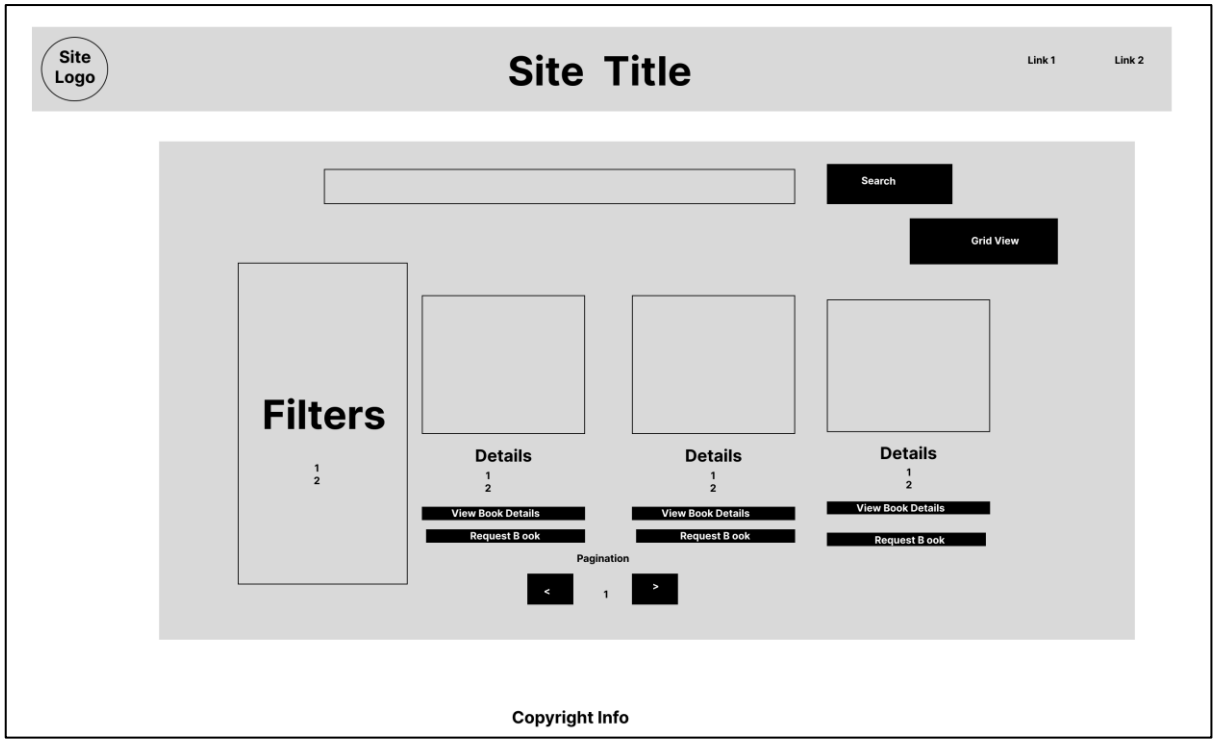
The spiral model allows using other process models in one or more of its cycles. This is either to reduce the risk at some stage or to get the requirements of the user clear or for resolving technical risks.

## 4.2 FIGMA Design – Wireframe Diagram

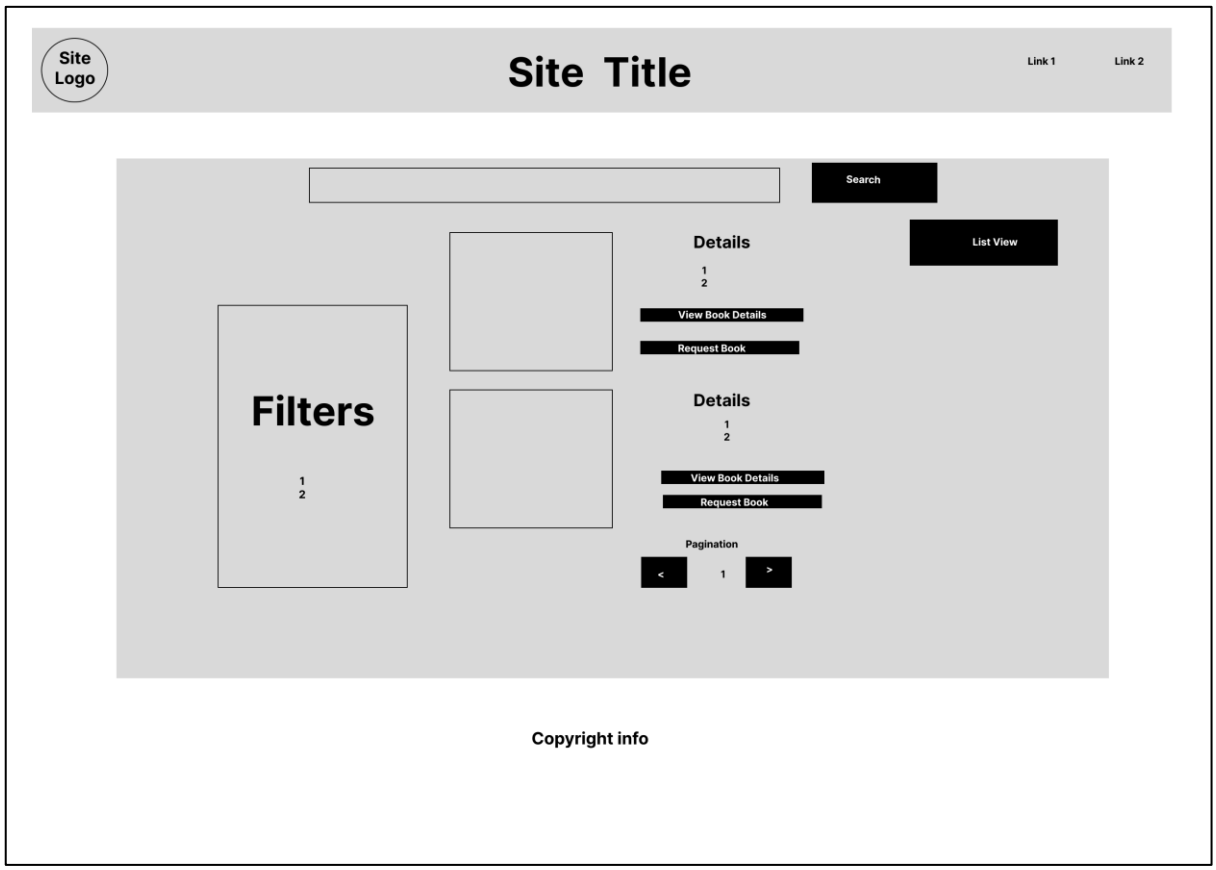
### 4.2.1 Student Book Request Flow

**Home Page**

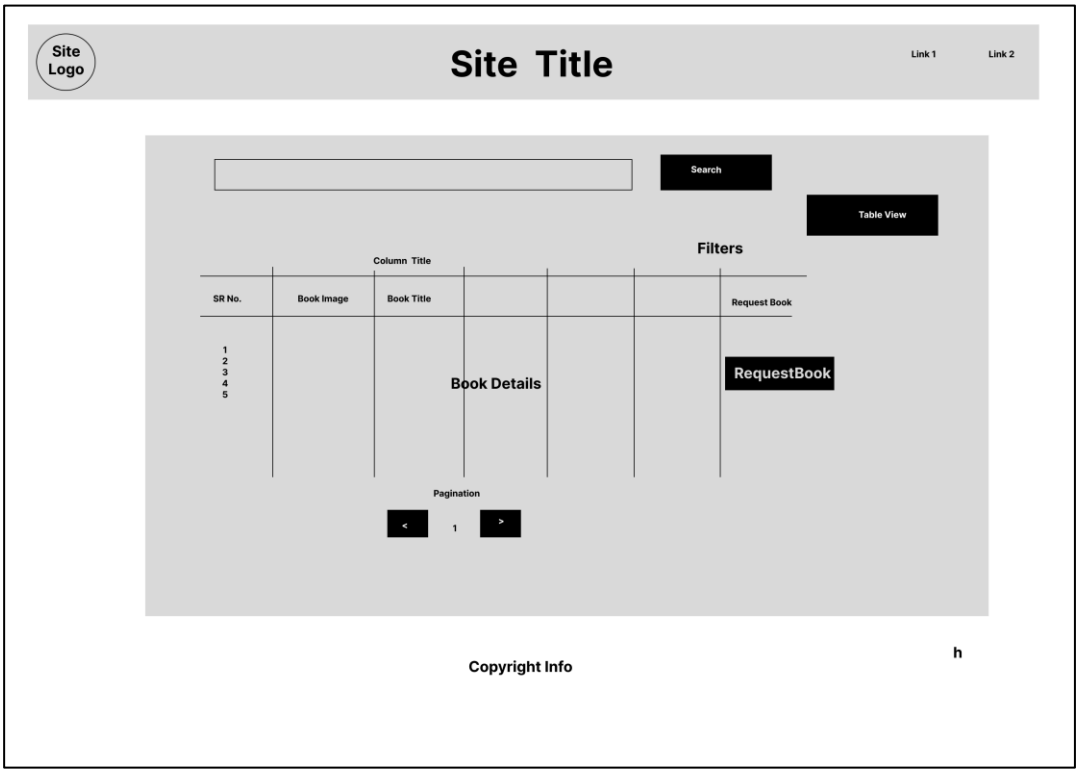
**Search results Grid View**



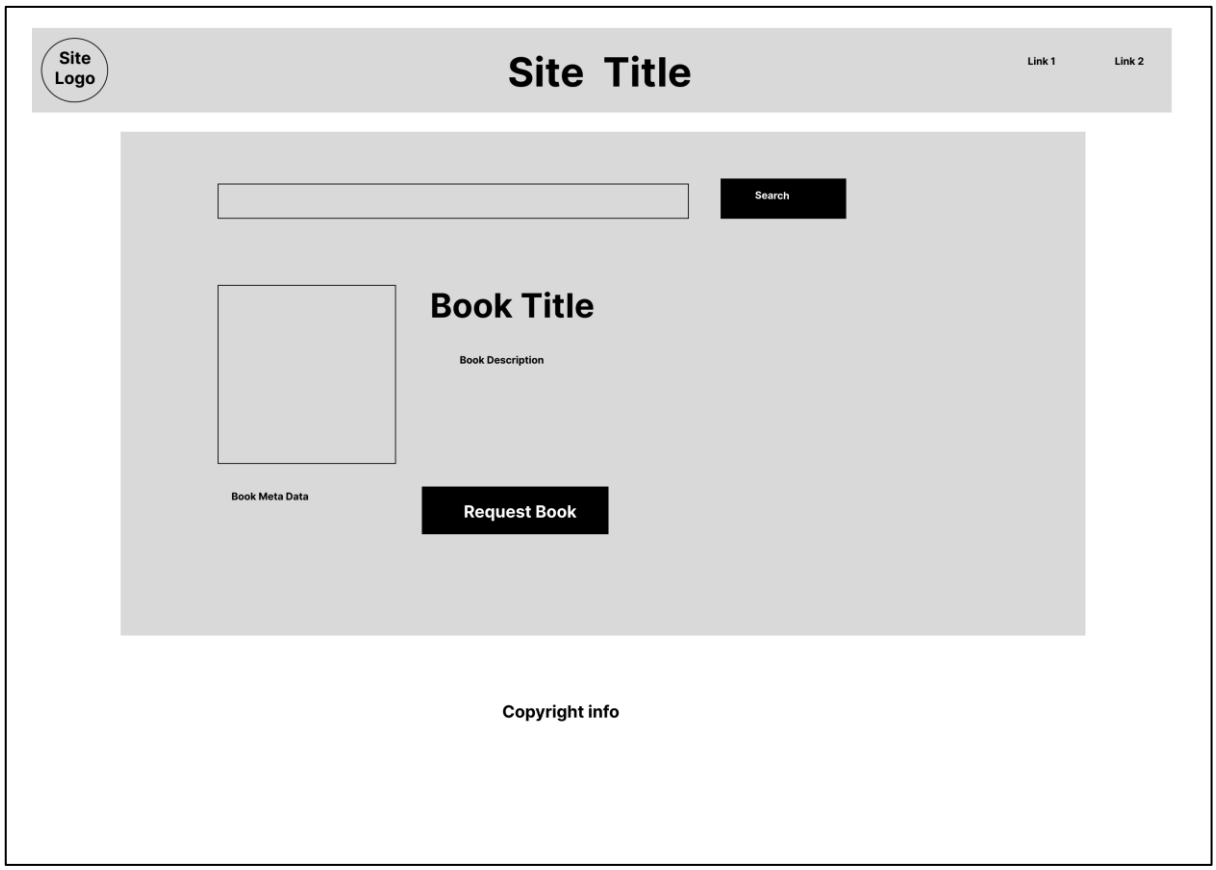
## Search Results List View

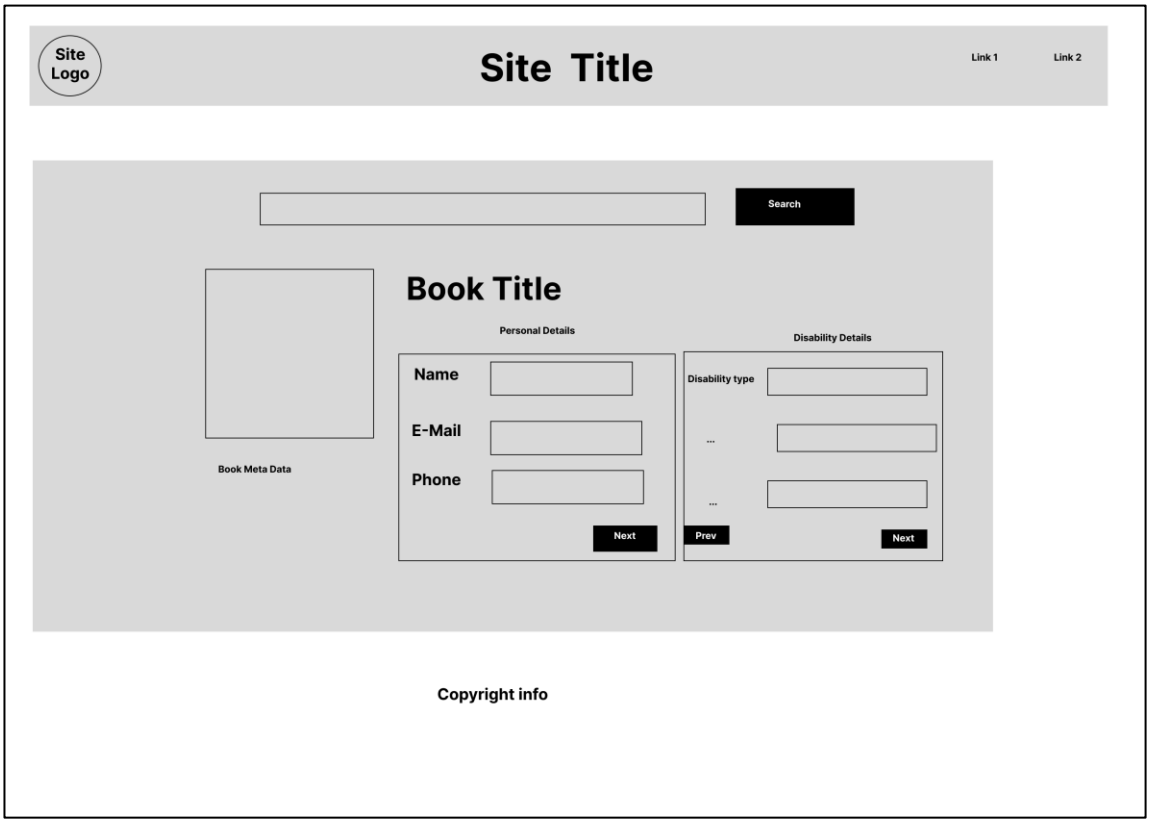



## Search Results Table View



## Book Details Page and Request Form

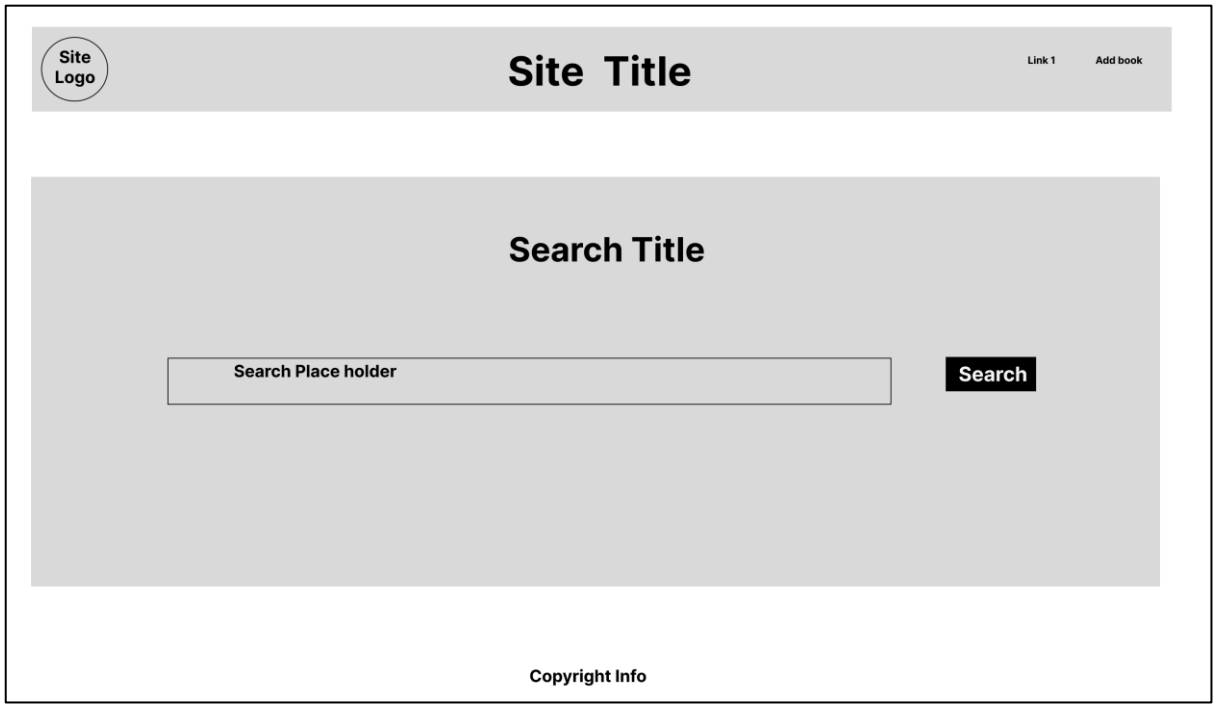


**Thank you Page.**

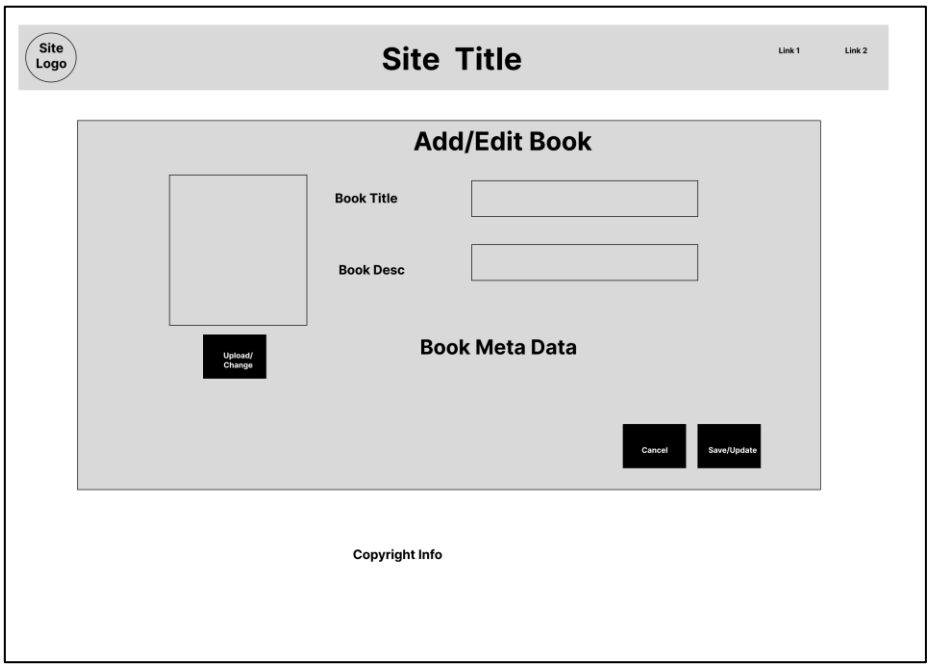
### 4.2.2 Admin Flow

**Home Page**



**Admin login Page**

**Add New Book Form**



## 4.3 Accessible Design

Accessible means a person with a disability is afforded the opportunity to acquire the same information, engage in the same interactions, and enjoy the same services as a person without a disability in an equally effective and equally integrated manner, with substantially equivalent ease of use. A person with a disability must be able to obtain the information as fully, equally, and independently as a person without a disability.

To design for accessibility means to be inclusive to the needs of all users. Accessible design is the idea that websites should be built so everybody can use them – no matter what their age, impairment, or situation.

It means simplifying your website so it's easily navigable by those using screen readers; removing links that use colour, to convey their clickability for those who are colour blind; ensuring you use a large enough font size for people with low vision.

And with over one billion people worldwide suffering from a disability that affects how they use the web, accessible web design is a necessity in today's world to ensure equal opportunities and access for all. Autism, colour blindness, paralysis; there are plenty of reasons why someone may struggle to use the internet, so ensuring the site has an accessible web design is key to creating inclusive web experiences.

Our project website has been developed for the XRCVC and hence we have taken special efforts to make it accessible specially for the visually challenged.

## 4.4 What we have done to make site accessible

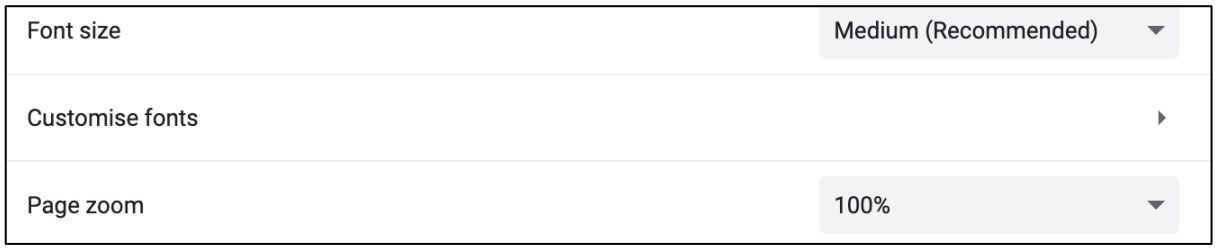
### 4.4.1 REM Units

We want to provide users with low vision a way to choose how fonts are displayed. They should have the ability to increase the size by say 200% while maintaining readability and avoiding content collisions and overlaps.

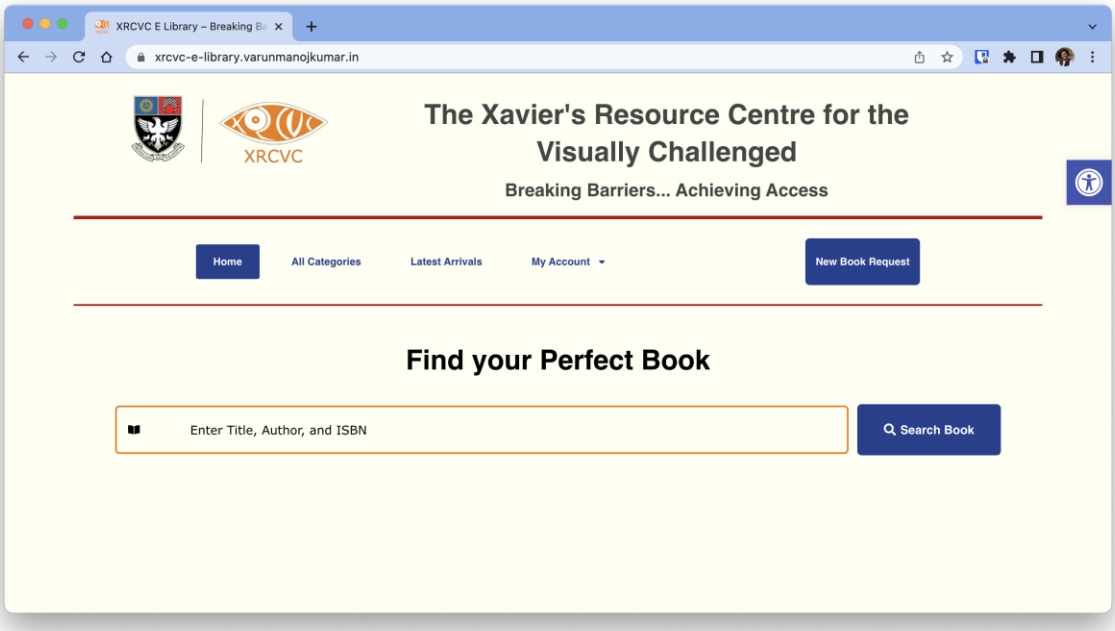
Of the many HTML styling options that CSS enables, font size is a crucial property — especially when it comes to display responsiveness. Moreover, there are several ways to dictate font size using CSS. Of these, rem units prove to be exceptionally flexible and relatively intuitive.

Rem units allow you to set sizes relative to the root element. Therefore, they are different from other sizing alternatives like em units — which depend on a parent element — or percentages, whose real values depend on the property on which they are used.

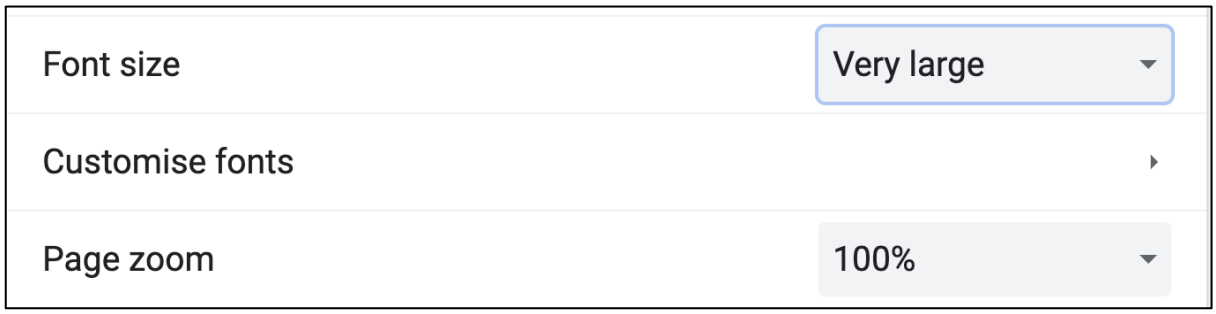
When building accessible websites, you should avoid using pixels to size font elements. Their absoluteness means they are not scalable. Instead, opt for rems and ems.

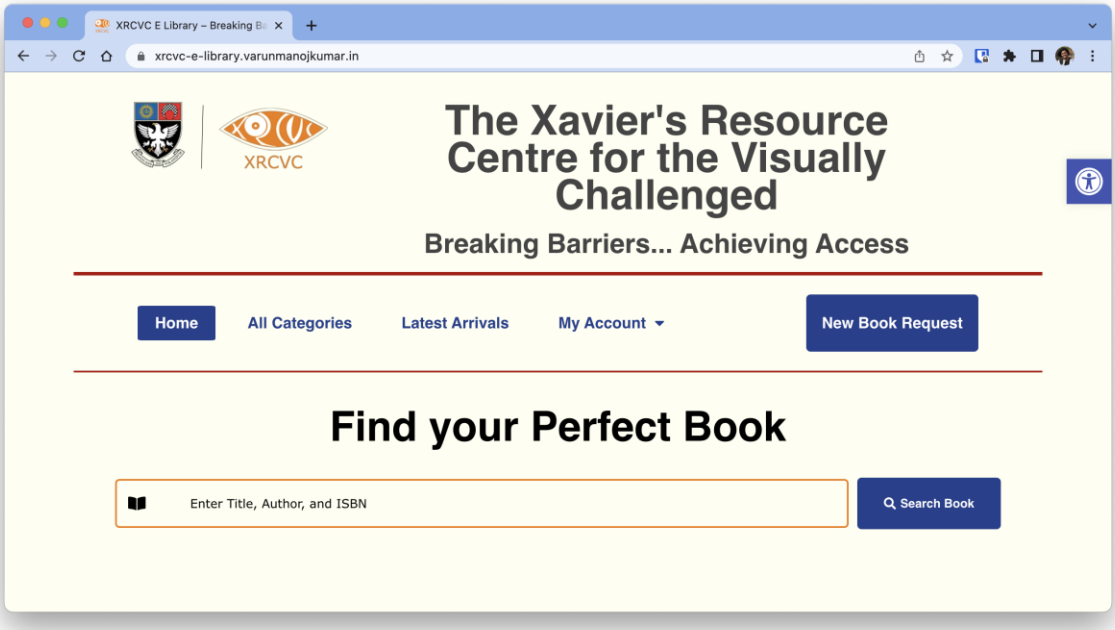



Default Font Size



Large Font Size



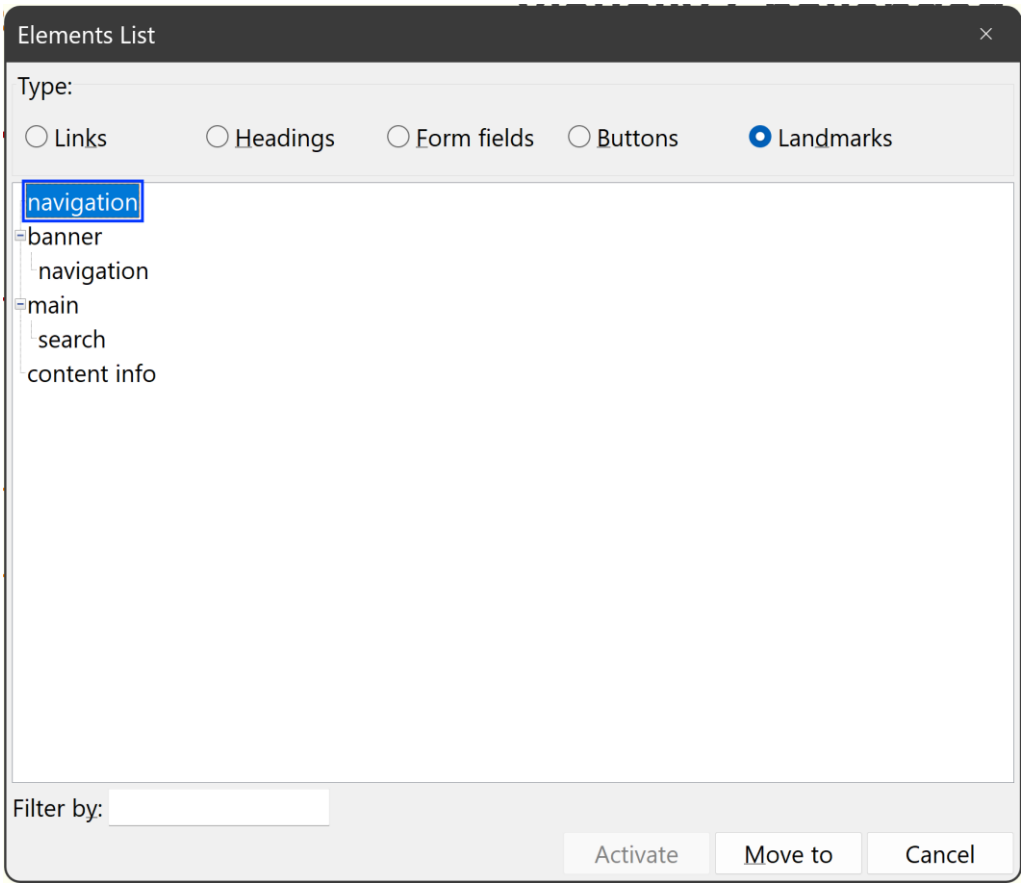




### 4.4.2 Semantic HTML and ARIA

Semantic HTML is considered the foundation of web accessibility. It greatly improves the experience of users that browse the web with assistive technology. This is because assistive technologies depend on the semantic structure and meaning of web pages to convey information to their users. Therefore, to ensure accessibility, semantic HTML should be used to define a meaningful structure that transcends what is visible through visual CSS styling. Although this may not make much difference to sighted users of the web, visually disabled users of the web rely on this to quickly navigate and understand the structure of web documents.

Like semantic HTML, ARIA is a W3 standard that helps make interfaces more accessible to people who use screen readers and other assistive technologies to consume content. ARIA attributes can be used to make unsemantic HTML more accessible to screen reader users.

We have used ARIA attributes for providing semantic markup to the website.




### 4.4.3 jQuery for Accessibility

jQuery has been used to add ARIA attributes to various elements on the website to make them accessible and screen reader friendly. It has also been used to make the website keyboard friendly.

# 5 Feasibility Study

## 5.1 Technical Feasibility

The main objective of this feasibility study is to find out the applicability and possibility of using WordPress, a content management system used mainly for online blogging, to build an eLibrary Management System / Website for the XRCVC.

Based on our study, we find that compared to traditional specialized software, using WordPress for building a website has its unique advantages and disadvantages.

**Advantages :** The first and most well-known advantage of WordPress is that it is ease of use and its ability to customize. With the help of some plug-ins, the blogging tool can build a completely functional website with all necessary components. There are also large number of themes, either free or not, made explicitly for commercial purpose. Users only need to find which plug-ins and theme fit their business the most. Depend on user's preference, extra functions can be added or removed by install/uninstall plug-ins. The vast number of plug-ins and themes found on WordPress points out the high demand of using WordPress. Secondly, the software is totally free, with open source code. Hence, WordPress programming code can be accessed by anyone, so users with coding skill can change it to better accommodate their needs. Besides, the user interface of WordPress is rather friendly, so that even users with basic computer skills can still manage a website by their own. The software is also search engine optimizable as there are many plug-ins and guidelines available on the Internet.

**Disadvantages :** As those plug-ins are from different developers, there is a risk of incompatible when updating them or upgrading WordPress. Another disadvantage of WordPress, when compared to mainstream software, is security. Because of its popularity and open source characteristic, WordPress continually undergo many kinds of attack form hackers. For online business, security is of utmost importance and this disadvantage is seen as one of the biggest barrier that prevents online vendors from choosing WordPress. Finally, since WordPress is created and distributed freely, there is no official support channel by developers. The only available way to find help is through documentations on their website or in forums where users help each other.

| **Advantages** | **Disadvantages** |
|---|---|
| Highly customizable | Compatibility when upgrade |
| Free and open-source | Security |
| Supported by most well-known hosting service provider | Help and support |

| | |
|---|---|
| User friendly and Search engine friendly | |

In conclusion, based on our analysis, even though there are other powerful and more specialized options, WordPress seems to be a beneficial choice for small organisations such as the XRCVC as the platform is highly customizable and powerful. Furthermore, it does not require advanced information technology knowledge to operate and the staff of XRCVC can manage the WordPress website easily.

## 5.2 Economic Feasibility

Building a website has its costs and the exact cost of a website would depend on many factors, like the domain and hosting fees, server configuration, plugins used and level of complexity.

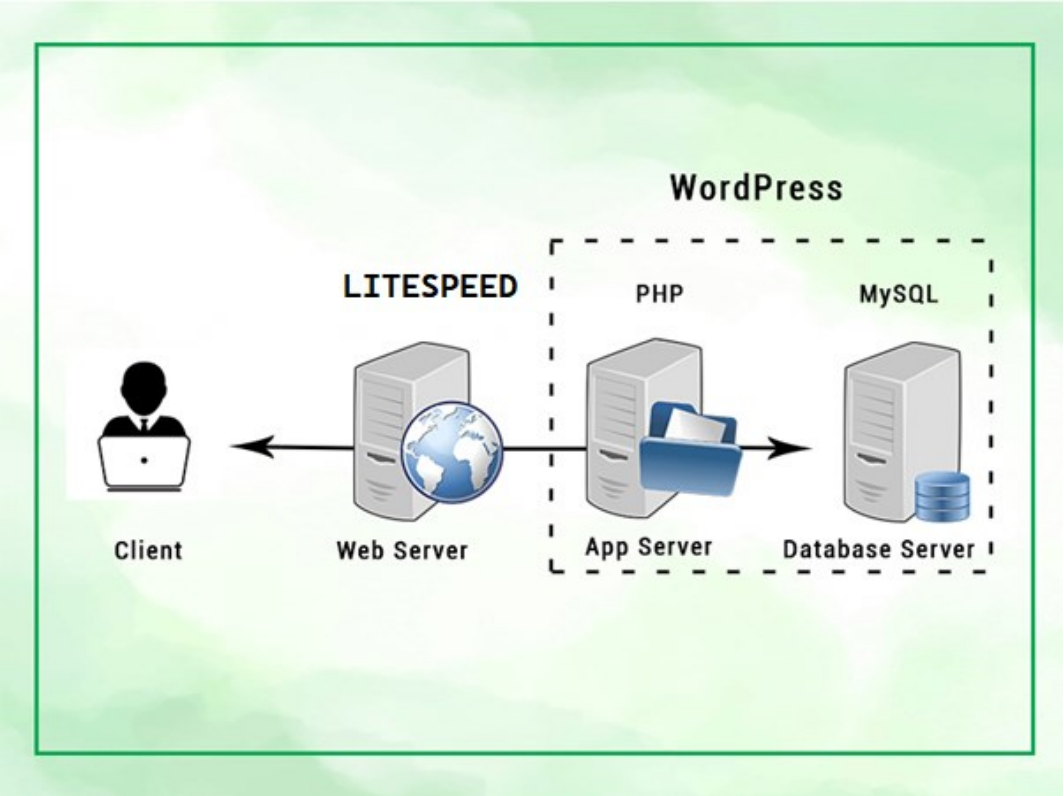
In this project, we have built the website using WordPress, a free content management system. While Wordpress is free, we need need to purchase a domain name and a web hosting plan. We have used Hostinger for the domain and hosting during the development phase in this project. Hostinger offers several plans for WordPress websites and we have used the “WordPress Starter” Plan.

WordPress Starter Plan cost – Rs. 149.00/month which includes free domain name for the first year.

For the deployment, XRCVC will host the website on their subdomain and we would migrate the website to their domain.

# 6 Technology Used

## 6.1 Architecture Diagram




## 6.2 Domain Hosting

Web hosting is a must-have when it comes to getting your website live.

A web hosting service is a type of Internet hosting service that hosts websites for clients, i.e. it offers the facilities required for them to create and maintain a site and makes it accessible on the World Wide Web. Companies providing web hosting services are sometimes called web hosts.

There are many types of hosting services available, such as shared hosting, WordPress hosting, and cloud hosting. The more popular a website is, the more resources it will need to function effectively.

We have hosted our website on hostinger.com. When we get a hosting plan, we rent an online space from a web host like hostinger.com to store all of your website files. This will allow people to access our site using a web browser.

## 6.3 WordPress[2]

WordPress (WP or WordPress.org) is a free and open-source content management system (CMS) written in hypertext preprocessor (PHP) language and paired with a MySQL or MariaDB database with supported HTTPS. Features include a plugin architecture and a template system, referred to within WordPress as "Themes". WordPress was originally created as a blog-publishing system but has evolved to support other web content types including more traditional mailing lists and Internet, media galleries, membership sites, learning management systems (LMS) and online stores. One of the most popular content management system solutions in use, WordPress is used by 42.8% of the top 10 million websites as of October 2021.

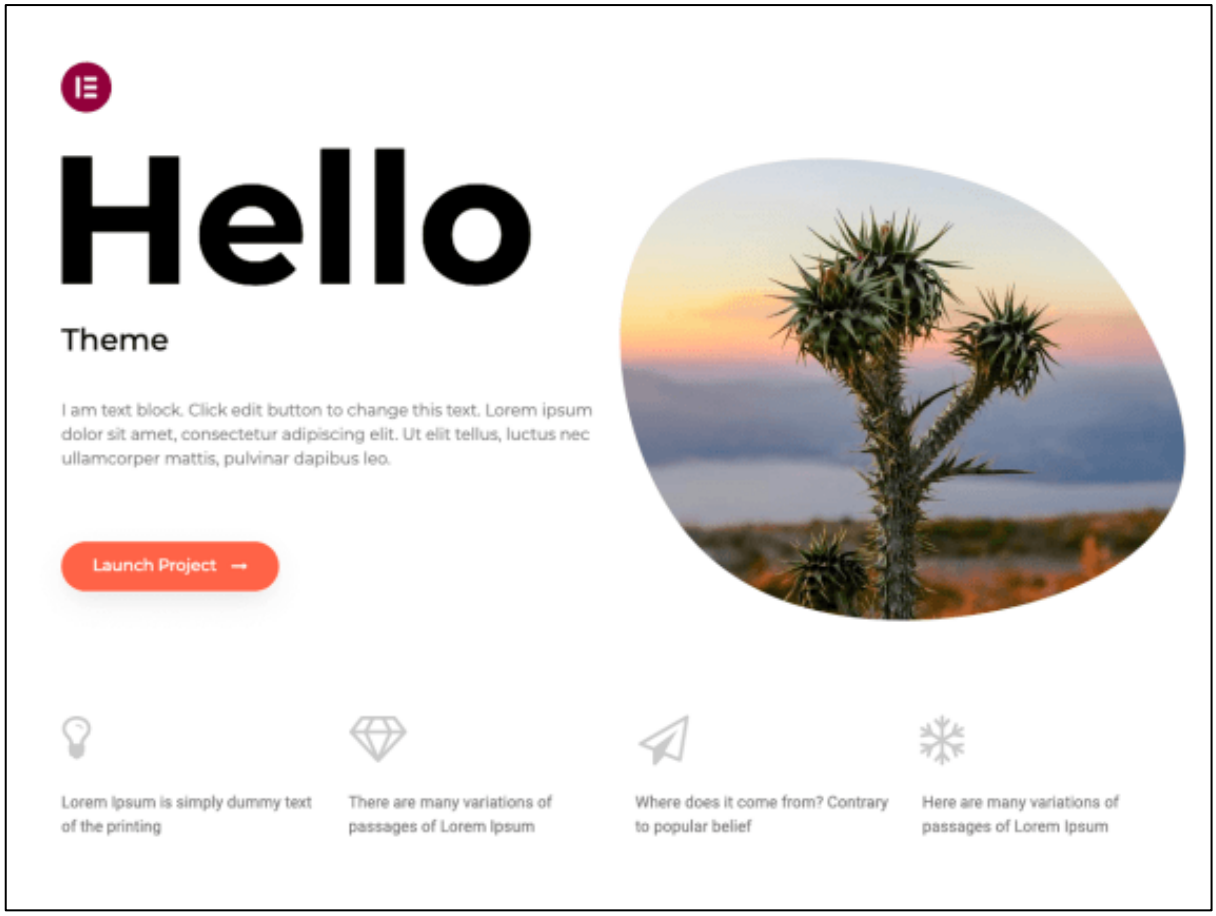
To function, WordPress has to be installed on a web server, either part of an Internet hosting service like Hostinger or a computer running the software locally.

### 6.3.1 Themes Used

A WordPress theme is a collection of files that make up the foundation of the site's appearance, including page layouts, style sheets, and sidebar positioning. These files will dictate how pages appear to website visitors. Themes play a crucial role in an overall website's appearance and help create a good-looking site without having to code. Themes are often used to change the look and feel of a wide range of things at once, which makes them much less granular than allowing the user to set each option individually.

A new WordPress website comes with an installed default theme that is simple and lightweight. We can either use it or change it to a free or premium theme.

**Hello Elementor Theme**




In this project, we have used the “Hello Elementor” theme. The Hello Theme is forever free and a great starting point to control and customize every part of any website you build with Elementor. Hello is free to download, loads quickly, and starts with minimal styling so you can customize it exactly the way you want.

Hello works in pure harmony with Elementor’s popular page builder and minimizes the need for other plugins. The theme loads in the blink of an eye, giving users a positive experience on the website. It offers a great responsive user experience across mobile, desktop, and tablet devices.

### 6.3.2 Plugins Used

WordPress' plugin architecture allows users to extend the features and functionality of a website or blog. As of December 2021, WordPress.org has 59,756 plugins available, each of

which offers custom functions and features enabling users to tailor their sites to their specific needs. However, this does not include the premium plugins that are available (approximately 1,500+), which may not be listed in the WordPress.org repository. These customizations range from search engine optimization (SEO) to client portals used to display private information to logged-in users, to content management systems, to content displaying features, such as the addition of widgets and navigation bars.

Most plugins are available through WordPress themselves, either via downloading them and installing the files manually via FTP or through the WordPress dashboard. However, many third parties offer plugins through their own websites, many of which are paid packages.

Plugins also represent a development strategy that can transform WordPress into all sorts of software systems and applications, limited only by the imagination and creativity of programmers. These are implemented using custom plugins to create non-website systems, such as headless WordPress applications and Software as a Service (SaaS) products.

#### *6.3.2.1 Elementor Page Builder*

Elementor is the WordPress website builder that more than five million WordPress sites use to create content and designs with a visual, drag-and-drop editor. It is the leading website building platform for WordPress, enabling web creators to build professional, pixel-perfect websites with an intuitive visual builder. This plugin helps create beautiful pages using a visual editor. It's designed to build dynamic websites quickly. Elevate your website's design with over 40 widgets and features like Box Shadow, Background Overlays, Hover Effects, Headline Effects, Animations, Gradient Backgrounds, CSS Transform, Mask Options, Shape Dividers, Coming Soon Mode, and many more.

#### *6.3.2.2 Crocoblock*

Crocoblock is a toolkit for building dynamic WordPress websites, using Elementor page builder and/or WordPress Block editor. Crocoblock JetPlugins is a collection of tools for building any type of website with WordPress and Elementor plugins.

#### 6.3.2.2.1 JetEngine [4]

JetEngine is a robust dynamic content plugin for WordPress and Elementor. It is a plugin that gives users the ability to append additional information to their WordPress website. It boasts a tremendous feature set, providing both basic functionality and more advanced.

#### 6.3.2.2.2 JetSmart Filter [5]

The JetSmartFilters plugin is a solution for creating advanced filters for the content of sites created in Wordpress. First, it allows you to customize any type of post in record time. In a few seconds and without reloading a page, users can filter products and publications.

#### 6.3.2.2.3 JetForm Builder

A functional yet easy-to-use Form Builder plugin lets you create, edit, and style advanced form types in the block editor (Gutenberg). No more supplementary drag-and-drop form builders. You can now use a one-stop interface to develop fully operative custom forms.

#### 6.3.2.3 One Click Accessibility [9]

The One Click Accessibility toolbar is the fastest plugin to help you make your WordPress website more accessible.

While most accessibility issues can't be addressed without directly changing your content, One Click Accessibility adds a number of helpful accessibility features with the minimum amount of setup and without the need for expert knowledge.

**Accessibility Toolbar:**

Add a toolbar toggling that allows you to set:

Resize font (increase/decrease)

Grayscale

Negative Contrast

High Contrast

Light Background

Links Underline

Readable Font

Link to Sitemap / Feedback / Help pages

**Accessibility Features:**

Enable skip to content

Add outline focus for focusable elements

Remove the target attribute from links

Add landmark roles to all links

Customizer for style adjustment

### 6.4 jQuery

jQuery is a JavaScript library designed to simplify HTML DOM tree traversal and manipulation, as well as event handling, CSS animation, and Ajax. It is free, open-source software using the permissive MIT License.

Also, jQuery has a number of built-in functions that can make it easier to add accessible features to a website, such as the ability to toggle ARIA attributes and add keyboard focus to elements.

This project uses jQuery to improve the accessibility of the website by manipulating the HTML DOM.

## 6.5 LiteSpeed WebServer

We have used LiteSpeed Webserver in our project.

LiteSpeed provides one-stop web-acceleration solutions that embrace advance cutting-edge technologies. LiteSpeed Web Server (LSWS) is a proprietary, lightweight web server software that provides fast performance and resource conservation without compromising server security. It is a popular web server offering high scalability, security, and load balancing. It is an Apache alternative that conserves resources without sacrificing performance, security, or convenience. It is designed to accelerate a website's performance. Built with event-driven architecture, the LiteSpeed server can handle higher traffic with fewer processes, reducing resource usage and improving page speed.

When hosting a WordPress website, there are several benefits of LiteSpeed compared to other server types. The performance and speed of LiteSpeed is much faster than other options like the Apache Server. LiteSpeed Web Server's cache engine is in-built into the server core and for WordPress purposes, it has its own LSCache plugin. LiteSpeed Web Serveral also provides built-in protection from large-scale WordPress Brute Force attacks.

Hence LiteSpeed is a perfect choice for WordPress Hosting.

## 6.6 Android with Java

Android apps are developed in the Java programming language using an IDE called Android Studio. Java is best suited for creating Android apps since it is platform-independent, and as a result, Java apps work on any platform. Java has its own runtime environment, Java Runtime Environment, as well as an API.

Android Studio is an IDE designed specifically for Android development. Android Studio provides a unified environment where you can build apps for Android phones, tablets, Android Wear, Android TV, and Android Auto. Structured code modules allow you to divide your project into units of functionality that you can independently build, test, and debug.

In this project we have developed an Android App using Java and Android Studio.

## 6.7 Firebase Backend Services

Firebase is an app development platform that helps you build and grow apps and games users love. Backed by Google and trusted by millions of businesses around the world.

### 6.7.1 Firebase Realtime Database [8]

We have used the Firebase Realtime Database to dynamically change the website URLs in the Android Application without having to update the android application. This makes it easier to update the application at the time of production.

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in realtime to every connected client. When you build cross-platform apps with our Apple platforms, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

### 6.7.2 Firebase App Check [7]

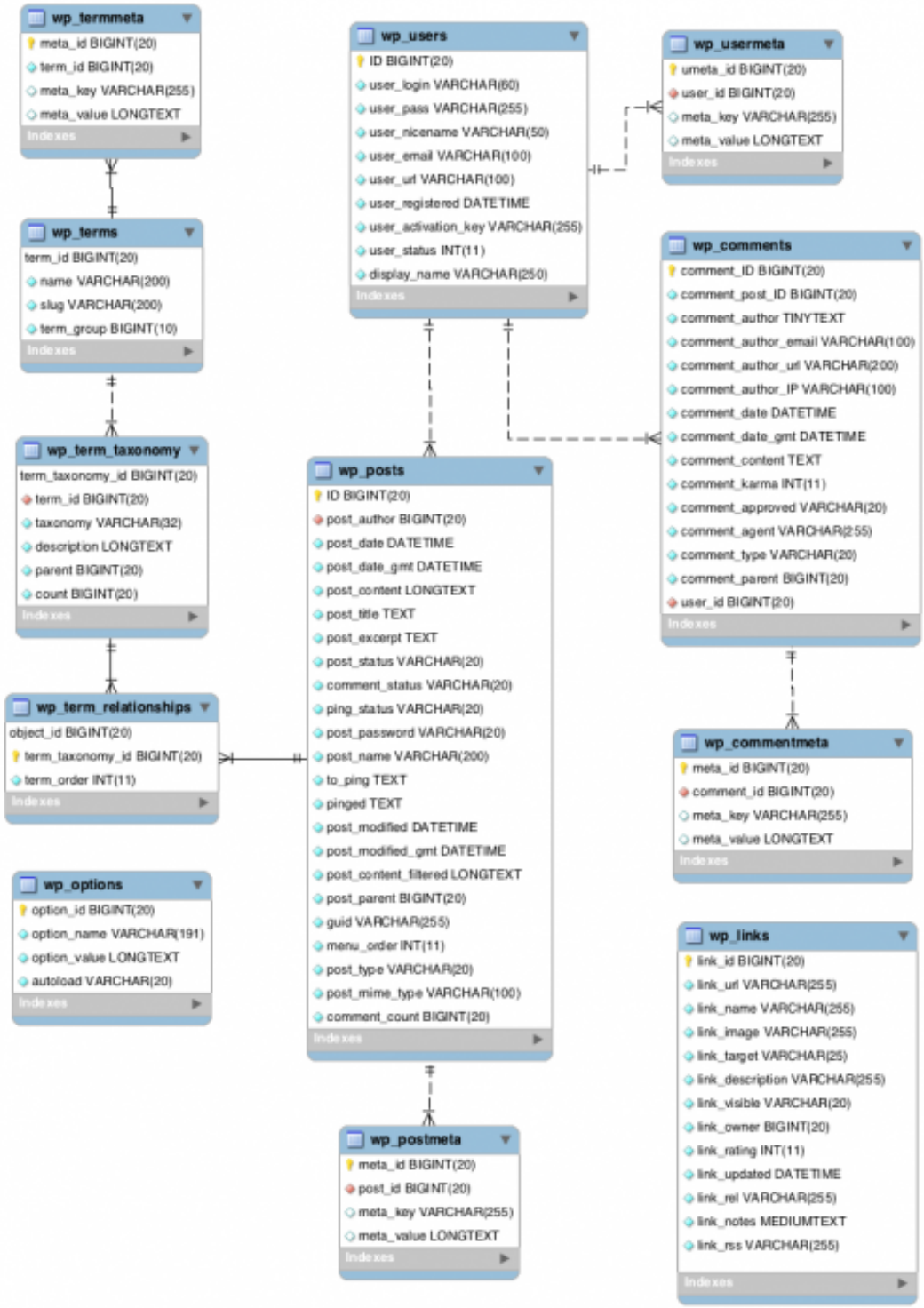
App Check helps protect your API resources from abuse by preventing unauthorized clients from accessing your backend resources. It works with both Firebase services, Google Cloud services, and your own APIs to keep your resources safe.

This project uses Firebase App Check in conjunction with type firebase, real-time database to protect the Real Time Database from unauthorized access.

# 7 Tables and Data Structure

## 7.1 WordPress Database ER Diagram [1]

The diagram below provides a visual overview of the WordPress database and the relations between the tables created during the WordPress installation.



## 7.2 WordPress Database Table Structure

| Table Name | Description |
|---|---|
| wp_commentmeta | Each **comment** features information called the **meta data** and it is stored in the `wp_commentmeta`. |
| wp_comments | The **comments** within WordPress are stored in the `wp_comments` table. |
| wp_links | The `wp_links` holds information related to the **links** entered into the Links feature of WordPress. |
| wp_options | The Options set under the **Administration >** Settings panel are stored in the `wp_options` table. |
| wp_postmeta | Each **post** features information called the **meta data** and it is stored in the `wp_postmeta`. Some plugins may add their own information to this table. |
| wp_posts | The core of the WordPress data is the **posts**. It is stored in the `wp_posts` table. Also Pages and navigation menu items are stored in this table. |
| wp_terms | The **categories** for both posts and links and the **tags** for posts are found within the `wp_terms` table. |
| wp_termmeta | Each **term** features information called the **meta data** and it is stored in `wp_termmeta`. |
| wp_term_relationships | **Posts** are associated with categories and tags from the `wp_terms` table and this association is maintained in the `wp_term_relationships` table. The association of **links** to their respective categories are also kept in this table. |
| wp_term_taxonomy | This table describes the taxonomy (**category**, **link**, or **tag**) for the entries in the `wp_terms` table. |
| wp_usermeta | Each **user** features information called the **meta data** and it is stored in `wp_usermeta`. |
| wp_users | The list of **users** is maintained in table `wp_users`. |

### 7.2.1 Table:wp_commentmeta

| Field | Type | Null | Key |
|---|---|---|---|
| meta_id | bigint(20) unsigned | | PRI |
| comment_id | bigint(20) unsigned | | IND |
| meta_key | varchar(255) | YES | IND |
| meta_value | Longtext | YES | |

**Indexes**

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | meta_id |
| comment_id | INDEX | comment_id |
| meta_key | INDEX | meta_key |

### 7.2.2Table: wp_comments

| Field | Type | Null | Key |
|---|---|---|---|
| comment_ID | bigint(20) unsigned | | PRI |
| comment_post_ID | bigint(20) unsigned | | IND |
| comment_author | tinytext | | |
| comment_author_email | varchar(100) | | IND |
| comment_author_url | varchar(200) | | |
| comment_author_IP | varchar(100) | | |
| comment_date | datetime | | |
| comment_date_gmt | datetime | | IND & IND Pt2 |

| | | | |
|---|---|---|---|
| comment_content | Text | | |
| comment_karma | int(11) | | |
| comment_approved | varchar(20) | | IND Pt1 |
| comment_agent | varchar(255) | | |
| comment_type | varchar(20) | | |
| comment_parent | bigint(20) unsigned | | IND |
| user_id | bigint(20) unsigned | | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | comment_ID |
| comment_post_ID | INDEX | comment_post_ID |
| comment_approved_date_gmt | INDEX | comment_approved<br>comment_date_gmt |
| comment_date_gmt | INDEX | comment_date_gmt |
| comment_parent | INDEX | comment_parent |
| comment_author_email | INDEX | comment_author_email |

## 7.2.3Table: wp_links

| Field | Type | Null | Key |
|---|---|---|---|
| link_id | bigint(20) unsigned | | PRI |
| link_url | varchar(255) | | |
| link_name | varchar(255) | | |
| link_image | varchar(255) | | |

| | | | |
|---|---|---|---|
| link_target | varchar(25) | | |
| link_description | varchar(255) | | |
| link_visible | varchar(20) | | IND |
| link_owner | bigint(20) unsigned | | |
| link_rating | int(11) | | |
| link_updated | Datetime | | |
| link_rel | varchar(255) | | |
| link_notes | mediumtext | | |
| link_rss | varchar(255) | | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | link_id |
| link_visible | INDEX | link_visible |

## 7.2.4 Table: wp_options

| Field | Type | Null | Key |
|---|---|---|---|
| option_id | bigint(20) unsigned | | PRI |
| option_name | varchar(64) | | UNI |
| option_value | Longtext | | |
| autoload | varchar(20) | | IND |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | option_id |
| option_name | UNIQUE | option_name |
| Autoload | INDEX | Autoload |

## 7.2.5Table: wp_postmeta

| Field | Type | Null | Key |
|---|---|---|---|
| meta_id | bigint(20) unsigned | | PRI |
| post_id | bigint(20) unsigned | | IND |
| meta_key | varchar(255) | YES | IND |
| meta_value | Longtext | YES | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | meta_id |
| post_id | INDEX | post_id |
| meta_key | INDEX | meta_key |

## 7.2.6 Table: wp_posts

| Field | Type | Null | Key |
|---|---|---|---|
| ID | bigint(20) unsigned | | PRI & IND Pt4 |
| post_author | bigint(20) unsigned | | IND |
| post_date | datetime | | IND Pt3 |
| post_date_gmt | datetime | | |

| | | | |
|---|---|---|---|
| post_content | longtext | | |
| post_title | text | | |
| post_excerpt | text | | |
| post_status | varchar(20) | | IND PT2 |
| comment_status | varchar(20) | | |
| ping_status | varchar(20) | | |
| post_password | varchar(20) | | |
| post_name | varchar(200) | | IND |
| to_ping | text | | |
| Pinged | text | | |
| post_modified | datetime | | |
| post_modified_gmt | datetime | | |
| post_content_filtered | longtext | | |
| post_parent | bigint(20) unsigned | | IND |
| Guid | varchar(255) | | |
| menu_order | int(11) | | |
| post_type | varchar(20) | | IND Pt1 |
| post_mime_type | varchar(100) | | |
| comment_count | bigint(20) | | |

## Indexes

| Keyname | Type | Field |
|---|---|---|

| | | |
|---|---|---|
| PRIMARY | PRIMARY | ID |
| post_name | INDEX | post_name |
| type_status_date | INDEX | post_type<br>post_status<br>post_date<br>ID |
| post_parent | INDEX | post_parent |
| post_author | INDEX | post_author |

### 7.2.7Table: wp_terms

| Field | Type | Null | Key |
|---|---|---|---|
| term_id | bigint(20) unsigned | | PRI |
| Name | varchar(200) | | IND |
| Slug | varchar(200) | | MUL |
| term_group | bigint(10) | | |

**Indexes**

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | term_id |
| slug | UNIQUE | slug |
| name | INDEX | name |

### 7.2.8 Table: wp_termmeta

| Field | Type | Null | Key |
|---|---|---|---|
| meta_id | bigint(20) unsigned | | PRI |
| term_id | bigint(20) unsigned | | IND |
| meta_key | varchar(255) | YES | IND |

| | | | |
|---|---|---|---|
| meta_value | longtext | YES | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | meta_id |
| term_id | INDEX | term_id |
| meta_key | INDEX | meta_key |

## 7.2.9 Table: wp_term_relationships

| Field | Type | Null | Key |
|---|---|---|---|
| object_id | bigint(20) unsigned | | PRI Pt1 |
| term_taxonomy_id | bigint(20) unsigned | | PRI Pt2 & IND |
| term_order | int(11) | | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | object_id<br>term_taxonomy_id |
| term_taxonomy_id | INDEX | term_taxonomy_id |

## 7.2.10 Table: wp_term_taxonomy

| Field | Type | Null | Key |
|---|---|---|---|
| term_taxonomy_id | bigint(20) unsigned | | PRI |
| term_id | bigint(20) unsigned | | UNI Pt1 |
| Taxonomy | varchar(32) | | UNI Pt2 & IND |
| Description | longtext | | |

| | | | |
|---|---|---|---|
| Parent | bigint(20) unsigned | | |
| Count | bigint(20) | | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | term_taxonomy_id |
| term_id_taxonomy | UNIQUE | term_id taxonomy |
| Taxonomy | INDEX | taxonomy |

## 7.2.11 Table: wp_usermeta

| Field | Type | Null | Key |
|---|---|---|---|
| umeta_id | bigint(20) unsigned | | PRI |
| user_id | bigint(20) unsigned | | IND |
| meta_key | varchar(255) | Yes | IND |
| meta_value | longtext | Yes | |

### Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | umeta_id |
| user_id | INDEX | user_id |
| meta_key | INDEX | meta_key |

## 7.2.12 Table: wp_users

| Field | Type | Null | Key |
|---|---|---|---|
| ID | bigint(20) unsigned | | PRI |
| user_login | varchar(60) | | IND |

| | | | |
|---|---|---|---|
| user_pass | varchar(64) | | |
| user_nicename | varchar(50) | | IND |
| user_email | varchar(100) | | |
| user_url | varchar(100) | | |
| user_registered | datetime | | |
| user_activation_key | varchar(60) | | |
| user_status | int(11) | | |
| display_name | varchar(250) | | |

## Indexes

| Keyname | Type | Field |
|---|---|---|
| PRIMARY | PRIMARY | ID |
| user_login_key | INDEX | user_login |
| user_nicename | INDEX | user_nicename |

### 7.3 Additional Custom Tables

#### 7.3.1 Table structure for table wp_jet_cct_book_requests

| Column | Type | Null | Default |
|---|---|---|---|
| ***_ID*** | bigint(20) | No | |
| cct_status | text | Yes | NULL |
| full_name | text | Yes | NULL |
| email | text | Yes | NULL |
| phone | bigint(20) | Yes | NULL |

| | | | |
|---|---|---|---|
| book_title | text | Yes | NULL |
| sp_link | text | Yes | NULL |
| cct_author_id | bigint(20) | Yes | NULL |
| cct_created | timestamp | No | current_timestamp() |
| cct_modified | timestamp | No | 0000-00-00 00:00:00 |
| disability_type | text | Yes | NULL |
| other_disability | text | Yes | NULL |
| additional_notes | longtext | Yes | NULL |

### 7.3.2 Table structure for table wp_jet_cct_new_book_requests

| Column | Type | Null | Default |
|---|---|---|---|
| _ID | bigint(20) | No | |
| cct_status | text | Yes | NULL |
| name | text | Yes | NULL |
| email | text | Yes | NULL |
| phone | bigint(20) | Yes | NULL |
| book_name | text | Yes | NULL |
| book_author_ | text | Yes | NULL |
| book_publisher | text | Yes | NULL |
| book_isbn | bigint(20) | Yes | NULL |
| cct_author_id | bigint(20) | Yes | NULL |
| cct_created | timestamp | No | current_timestamp() |

| | | | |
|---|---|---|---|
| cct_modified | timestamp | No | 0000-00-00 00:00:00 |
| disability_type | text | Yes | NULL |
| other_disability | text | Yes | NULL |
| language | text | Yes | NULL |
| book_formats | text | Yes | NULL |
| request_status_336 | text | Yes | NULL |
| request_status_details_ | text | Yes | NULL |
| additional_notes | text | Yes | NULL |

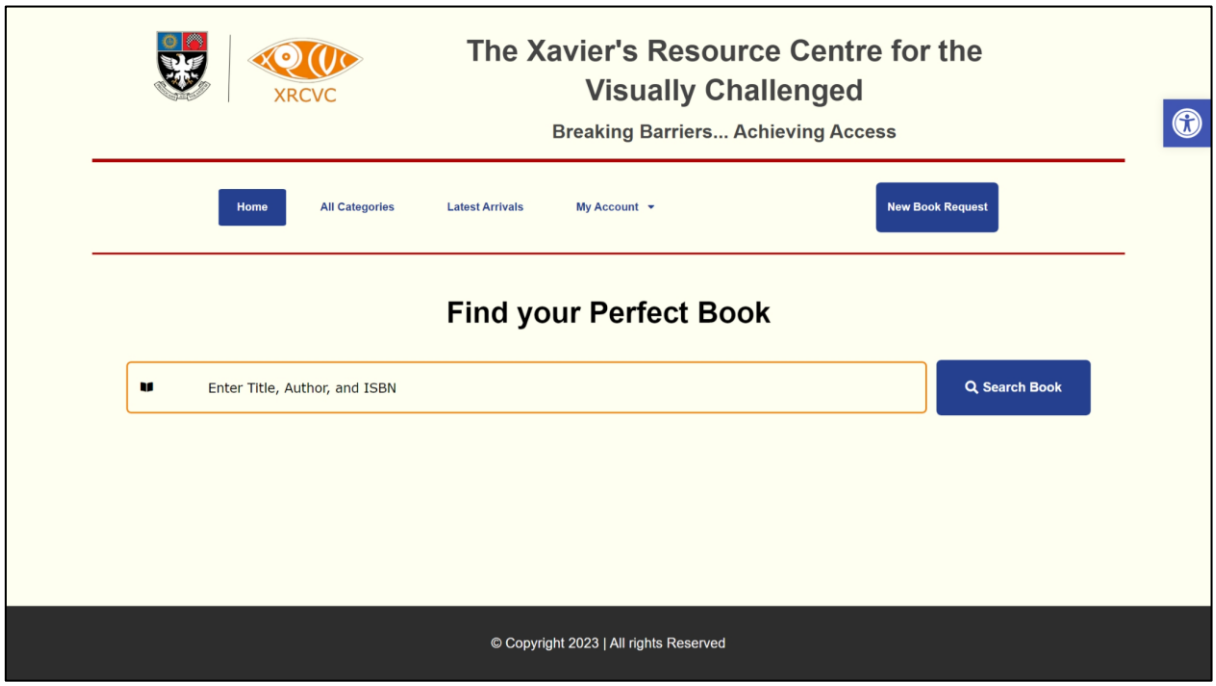
# 8 Modules & Screen Layouts

## 8.1 Student Book Request Modules

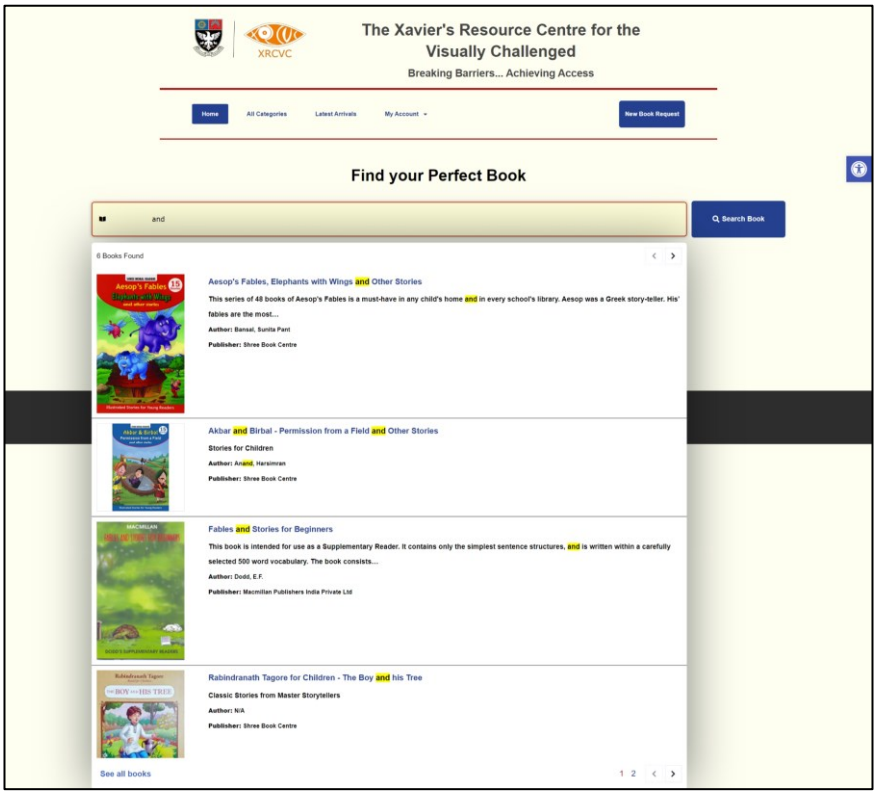
### 8.1.1 Search Book / Filter Book

The Search Book is displayed on the Home Page

Student may Enter 3 or more characters of the Book's Title, Author or ISBN Number to search the repository of accessible books in the E-library.



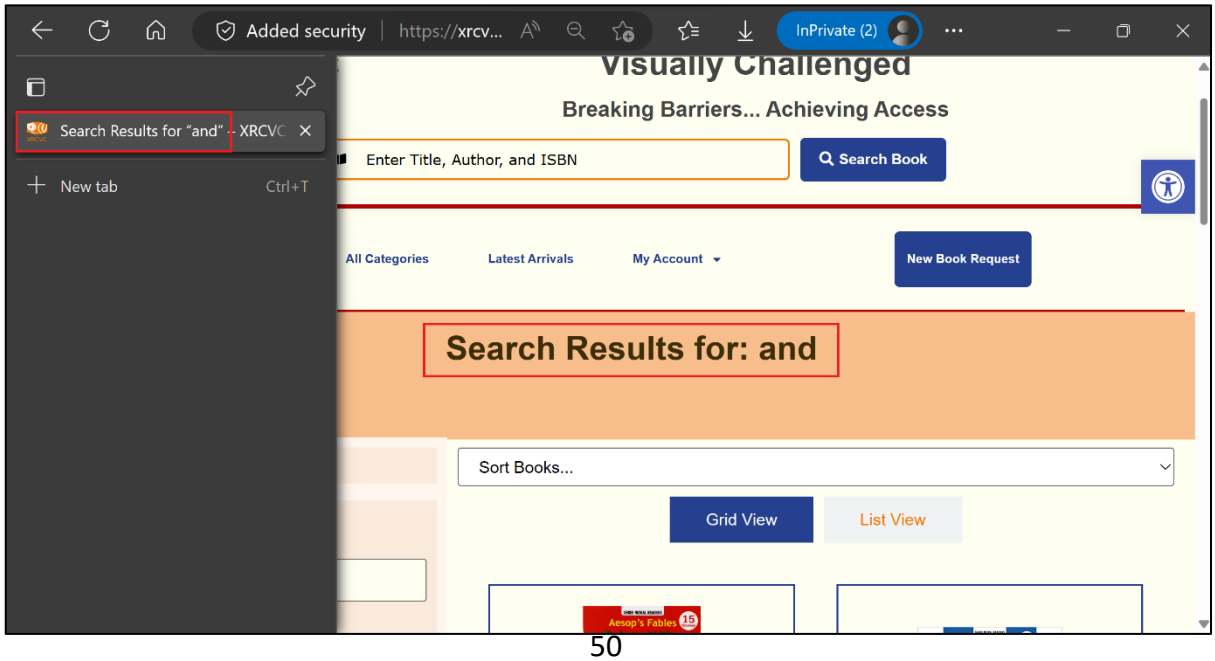

A quick Preview of the Book is displayed below the search box.



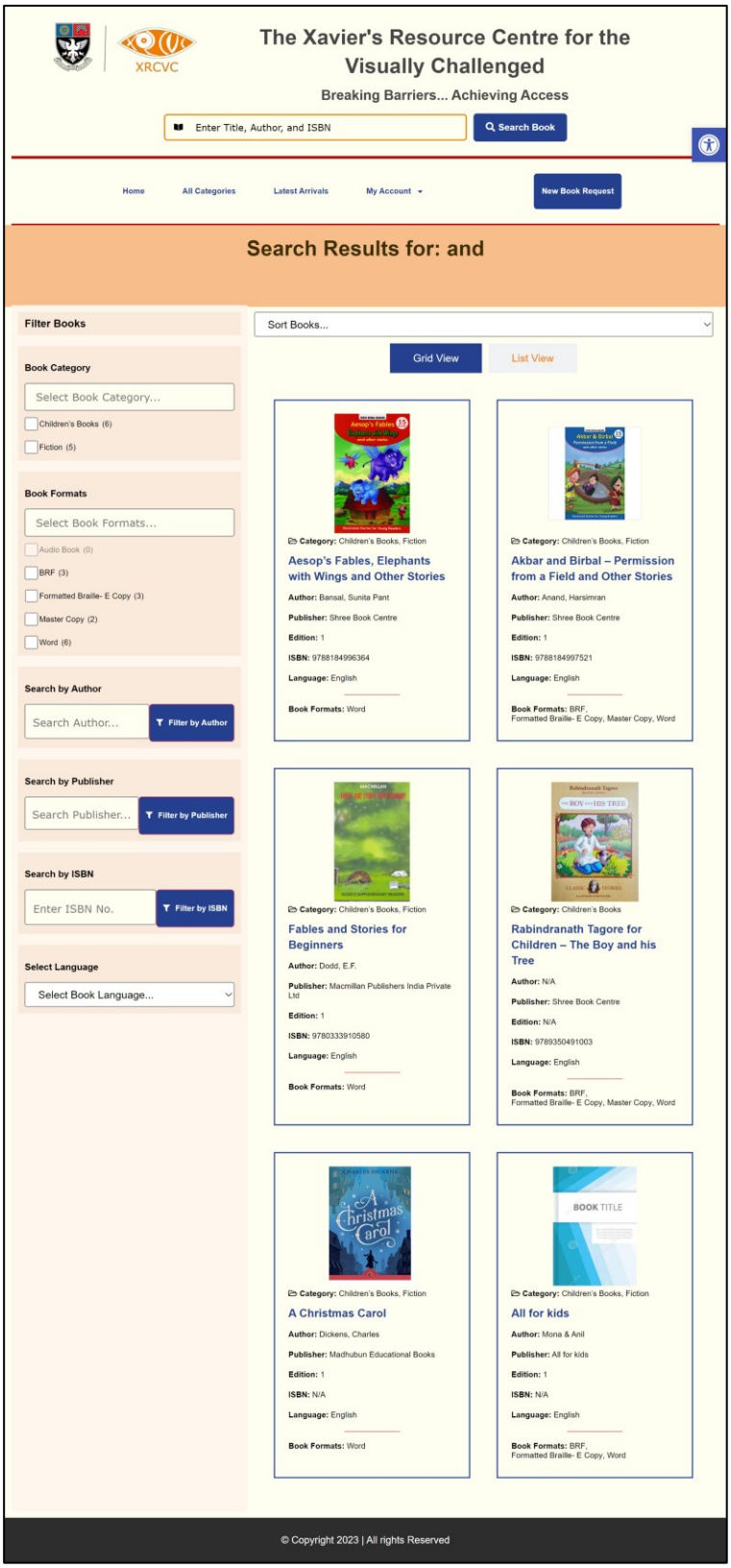

Students may navigate the Search results by using the arrow keys provided at the bottom of the search results or by using the **TAB Key** on the keyboard. Student may also click on the Button **'See all Results'** to view all results.

### 8.1.2 Search Results

The Page Title of the Website changes to indicate the search term.



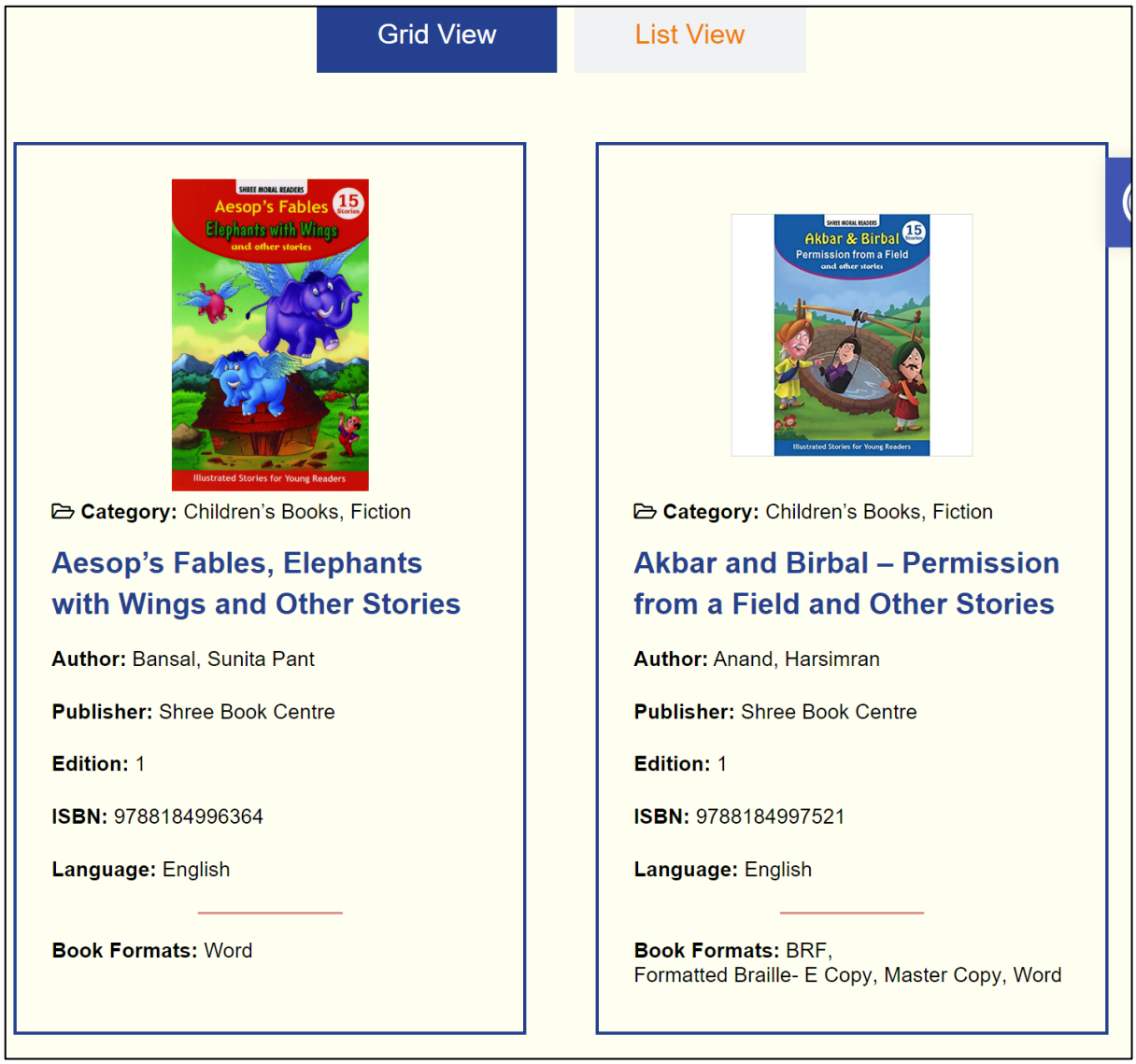
A page containing all books that match the search criteria are displayed.



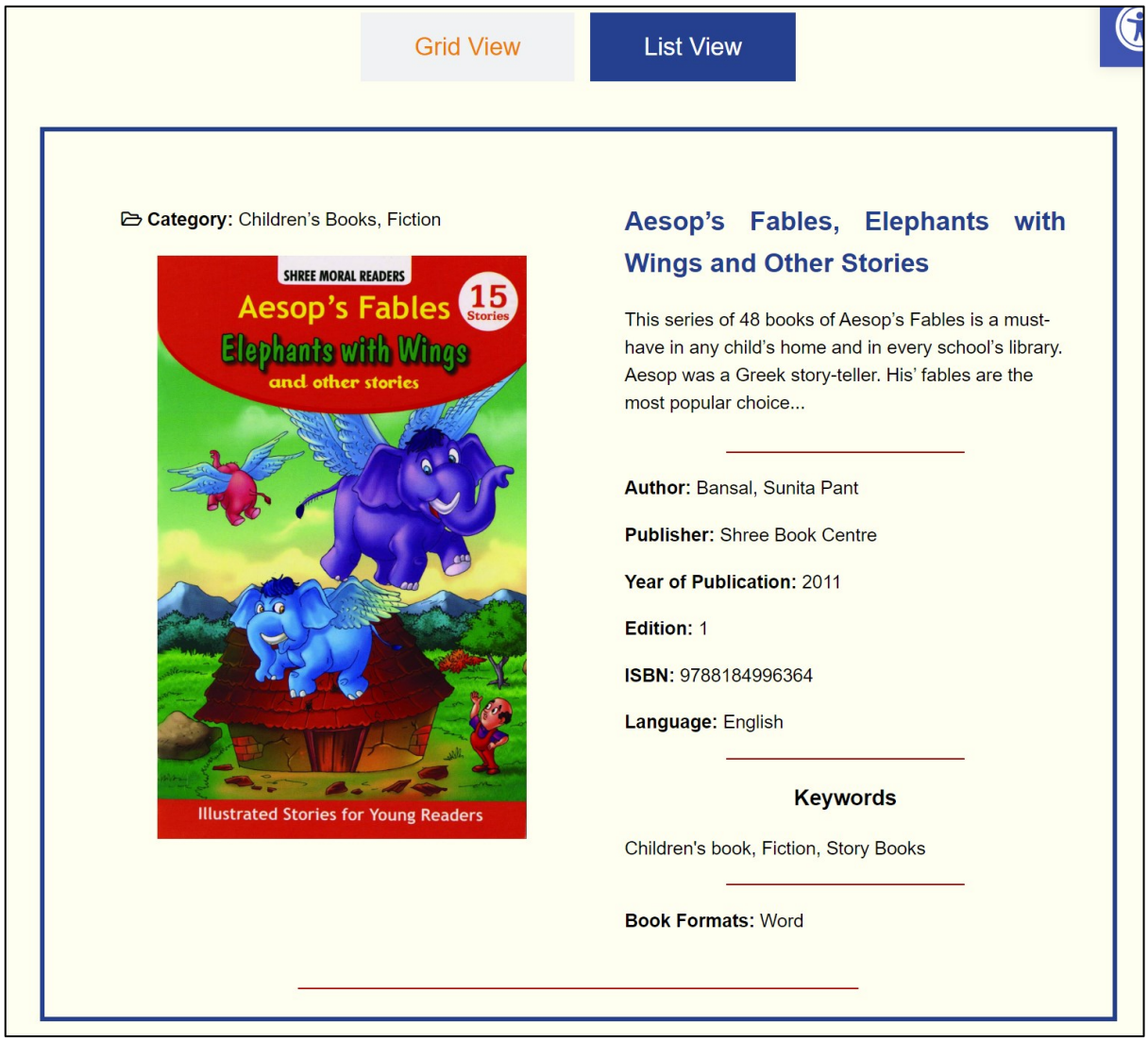
From the Search Results page, Students may

- Choose to view the search results in Grid and List view
- Sort the Results
- Filter the Books based on
    - Category
    - Formats
    - Author
    - Publisher
    - ISBN
    - Language

### *8.1.2.1 Grid View*



### 8.1.2.2 *List View*




### 8.1.2.3 *Sort books*

Students may also Sort the books in Alphabetical order or by Date. The Book will update automatically without refreshing the website.

### 8.1.2.4 *Filter Books*

Students may choose to filter books by using the filter side bar present at the left of the search results.

# The Xavier's Resource Centre for the Visually Challenged

XRCVC

Breaking Barriers... Achieving Access

Enter Title, Author, and ISBN

Search Book

Home | All Categories | Latest Arrivals | My Account | New Book Request

## Search Results for: and

### Filter Books

**Active filters:**

Book Category: Fiction X

Book Format: Formatted Braille- E Copy X

Book Language: English X

Remove all filters

**Book Category**

Select Book Category...

- Children's Books (2)
- Fiction (2)

**Book Formats**

Select Book Formats...

- Audio Book (0)
- BRF (2)
- Formatted Braille- E Copy (2)
- Master Copy (1)
- Word (2)

**Search by Author**

Search Author... Filter by Author

**Search by Publisher**

Search Publisher... Filter by Publisher

**Search by ISBN**

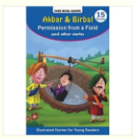
Enter ISBN No. Filter by ISBN

**Select Language**

English (2)

Sort Books...

Grid View | List View




**Category:** Children's Books, Fiction

**Akbar and Birbal – Permission from a Field and Other Stories**

**Author:** Anand, Harsimran

**Publisher:** Shree Book Centre

**Edition:** 1

**ISBN:** 9788184997521

**Language:** English

**Book Formats:** BRF, Formatted Braille- E Copy, Master Copy, Word

**Category:** Children's Books, Fiction

**All for kids**

**Author:** Mona & Anil

**Publisher:** All for kids

**Edition:** 1

**ISBN:** N/A

**Language:** English

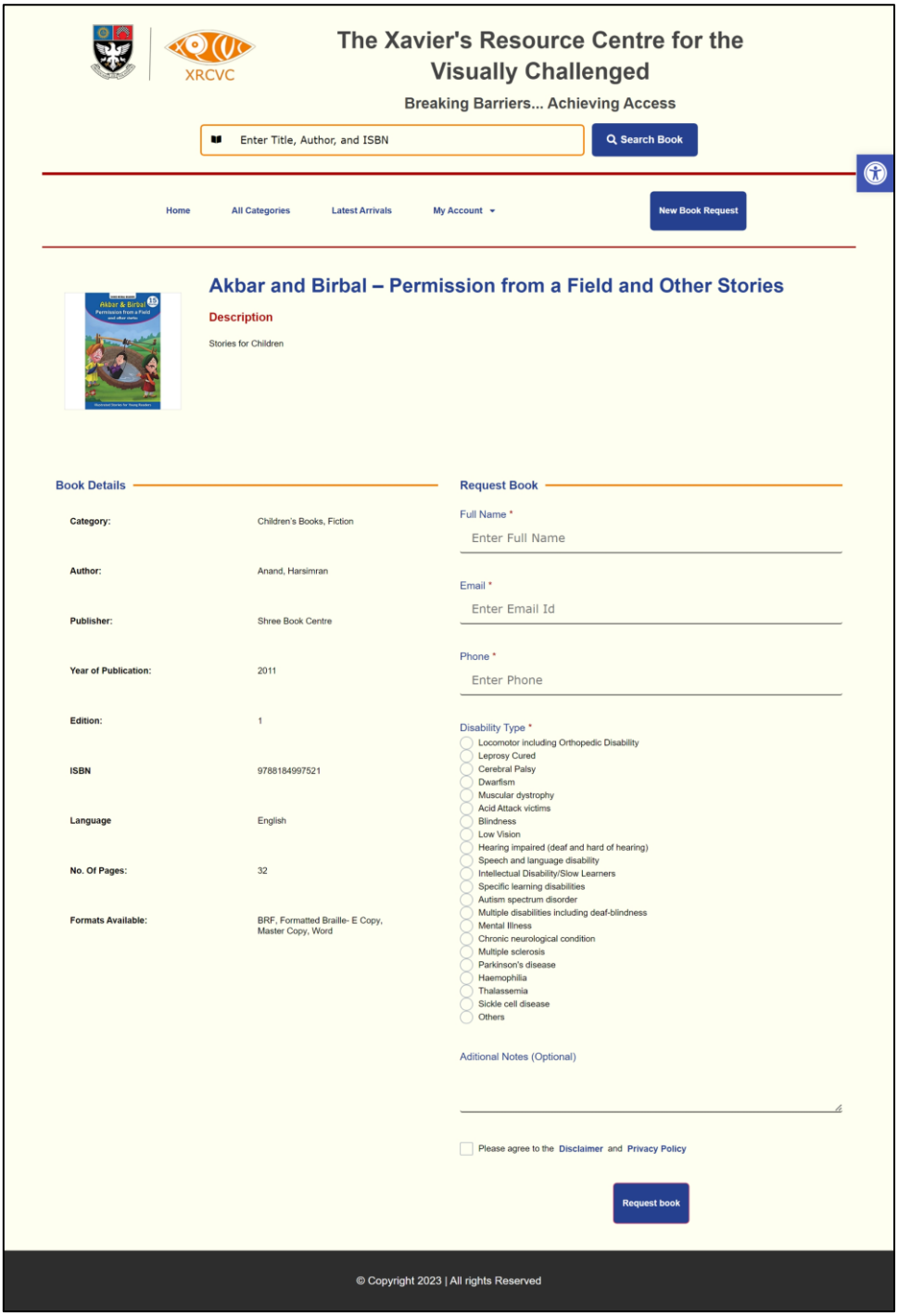
**Book Formats:** BRF, Formatted Braille- E Copy, Word

© Copyright 2023 | All rights Reserved

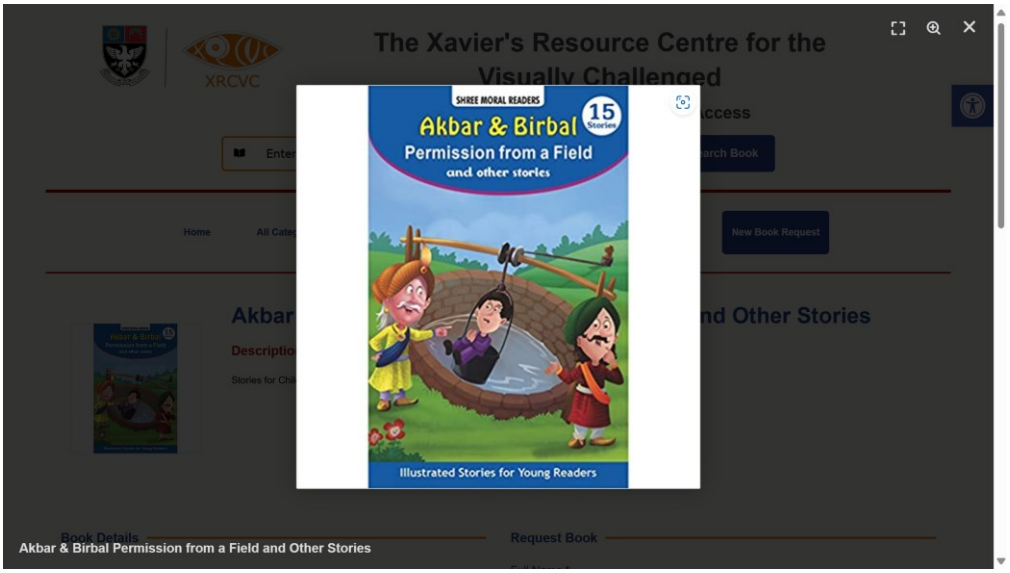
The Active Filters are displayed in the side bar along with the option to remove all filters or individually. The counter beside each filter shows the count of results that match the criteria.

### 8.1.3 Book Details Page

Once the student clicks on either the Book Cover Image or the Title, they are redirected to the Book Details Page.

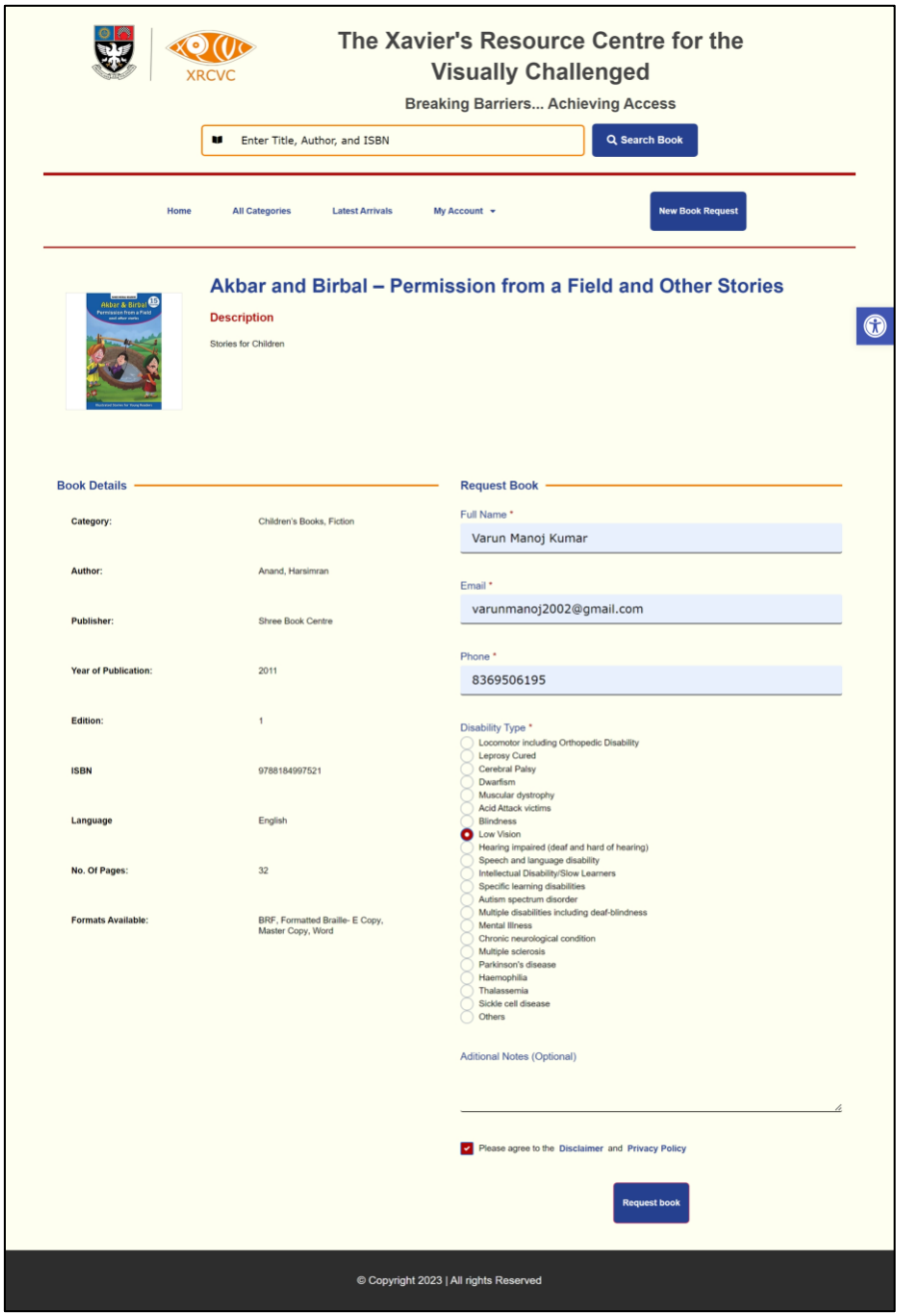



From the Book Details Page, the student can read more about the chosen book or request the same from XRCVC by filling the from. The student may also click on the image to view an enlarged version of the same (In a **Lightbox effect**).



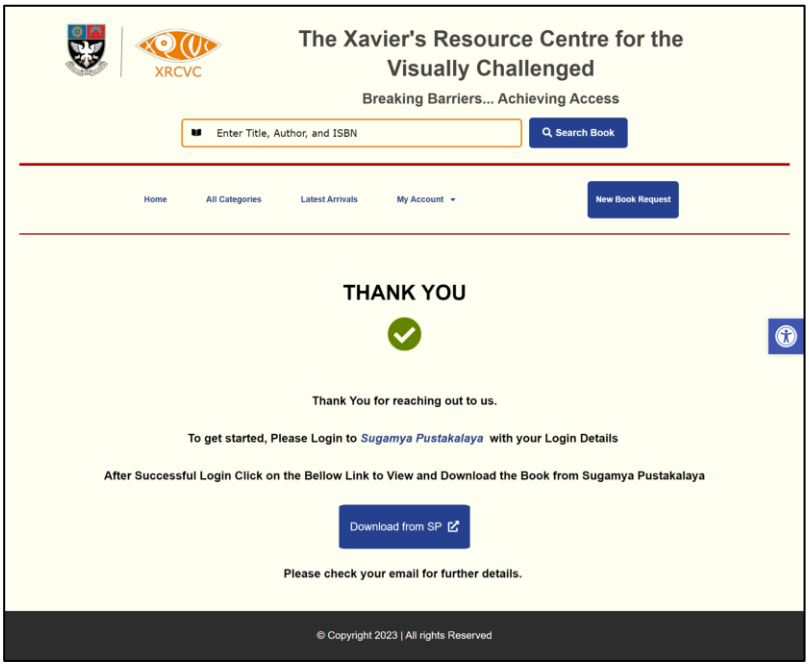

### 8.1.4 Request Book

Student may request the book from XRCVC by filling the Book Request Form available in the Book Details page. Student must fill out all mandatory fields and agree to the Disclaimer and Privacy policy before they can request the book from XRCVC. Students also have the option to mention any additional notes while making the request in the Additional Notes field.




### *8.1.4.1 Thank You Page*

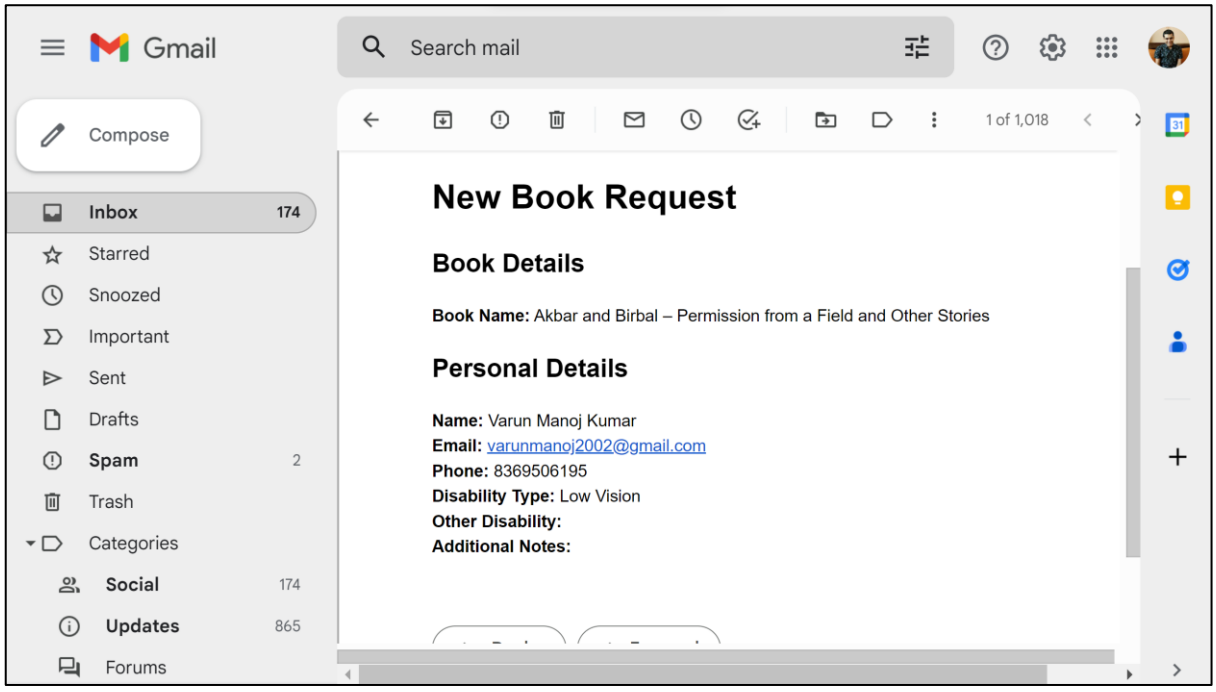
Once the student requests the book, they are redirected to a Thank You Page.




The Thank You page shows the instructions on how to download the book from Sugamya Pustakalaya. Students are requested to first Login to Sugamya Pustakalaya and then click on the **'Download From SP'** Link to redirect the student to the Book's Detail Page from where they can download the Book.

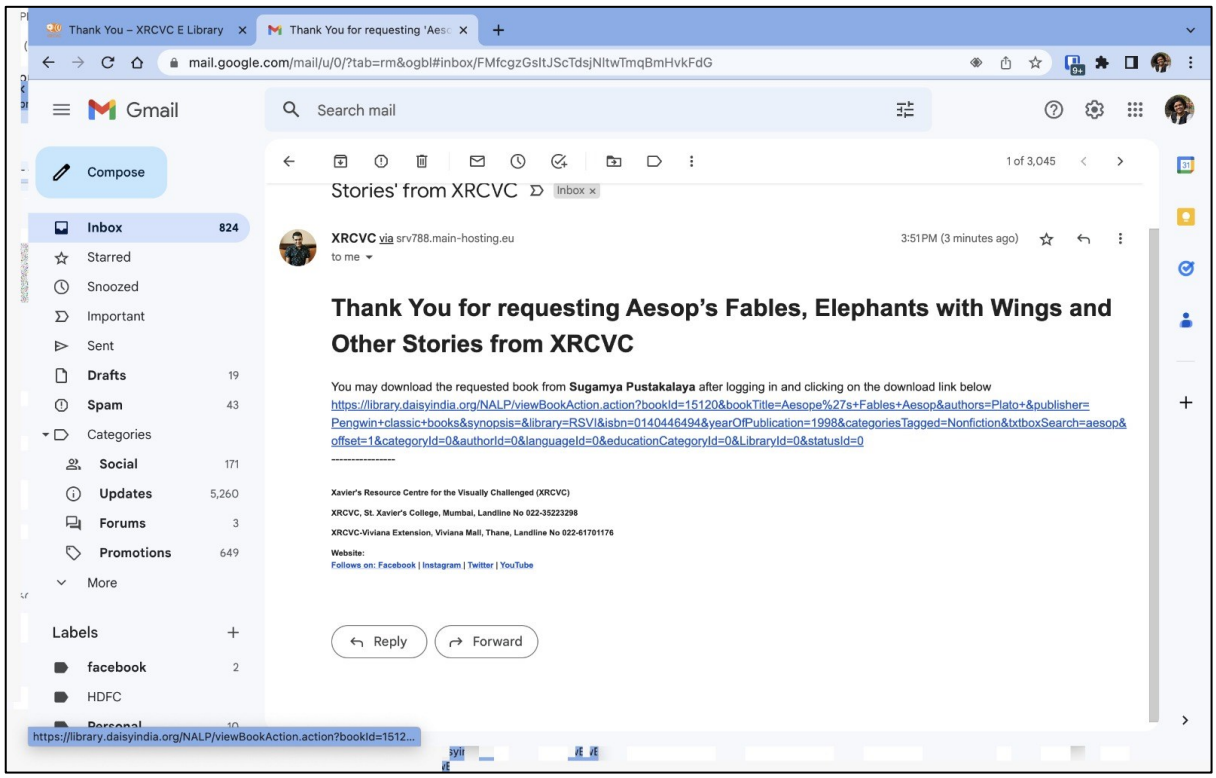
The student and the admin of the website is also notified by email.

### *8.1.4.2 Admin Email*




The Admin is provided with the information about the student's personal details like the name, email Id, phone and disability type and Additional notes if any.

### *8.1.4.3 Student Email*




The Student receives an automated email with instructions on how to access the book from Sugamya Pustakalaya and a direct Book Link.

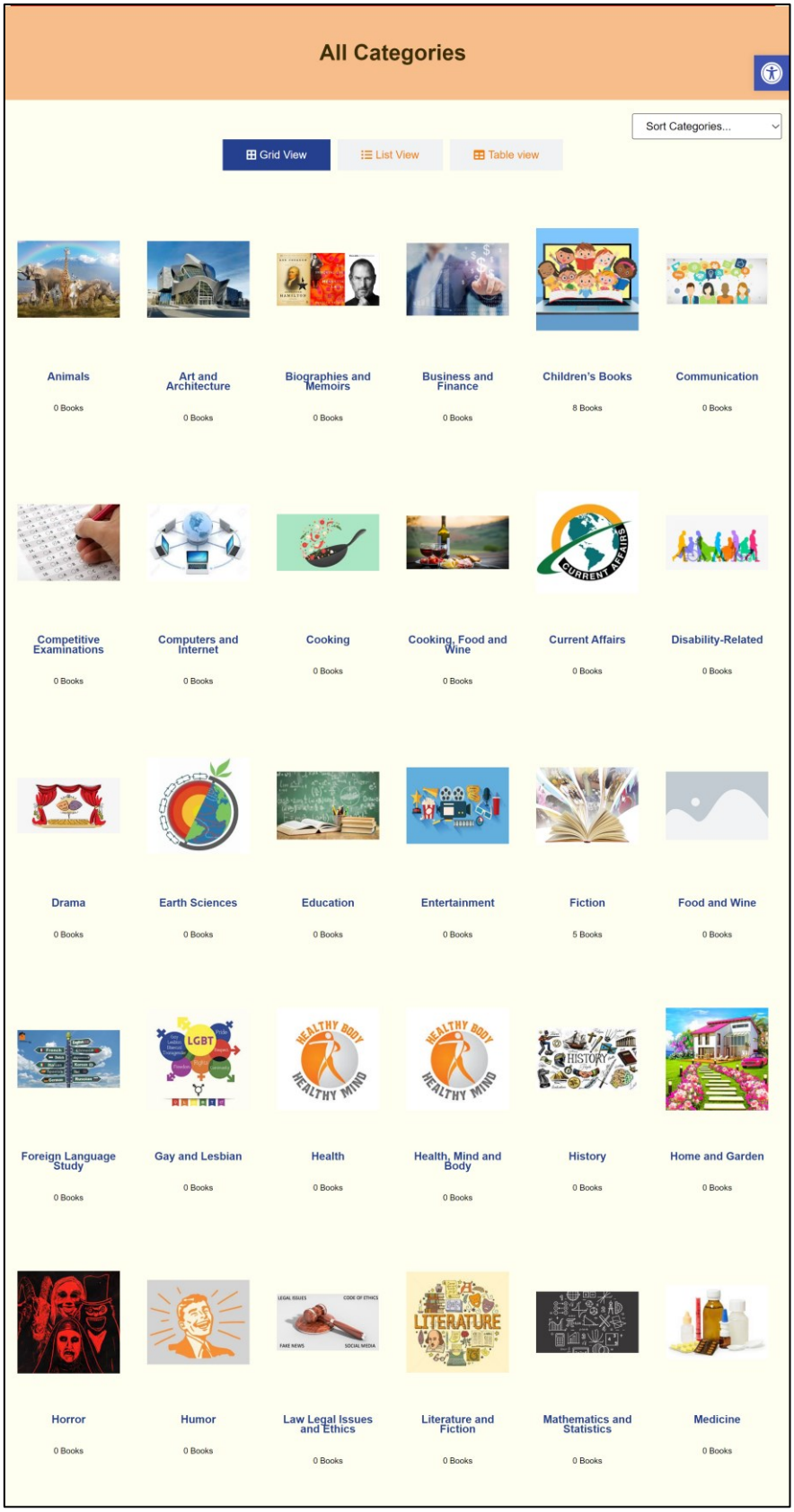
### 8.1.5 All Categories

The All-Categories Page show the list of all the Book categories with an Image and the number of books that are present in that category. The student may wish to view the same in

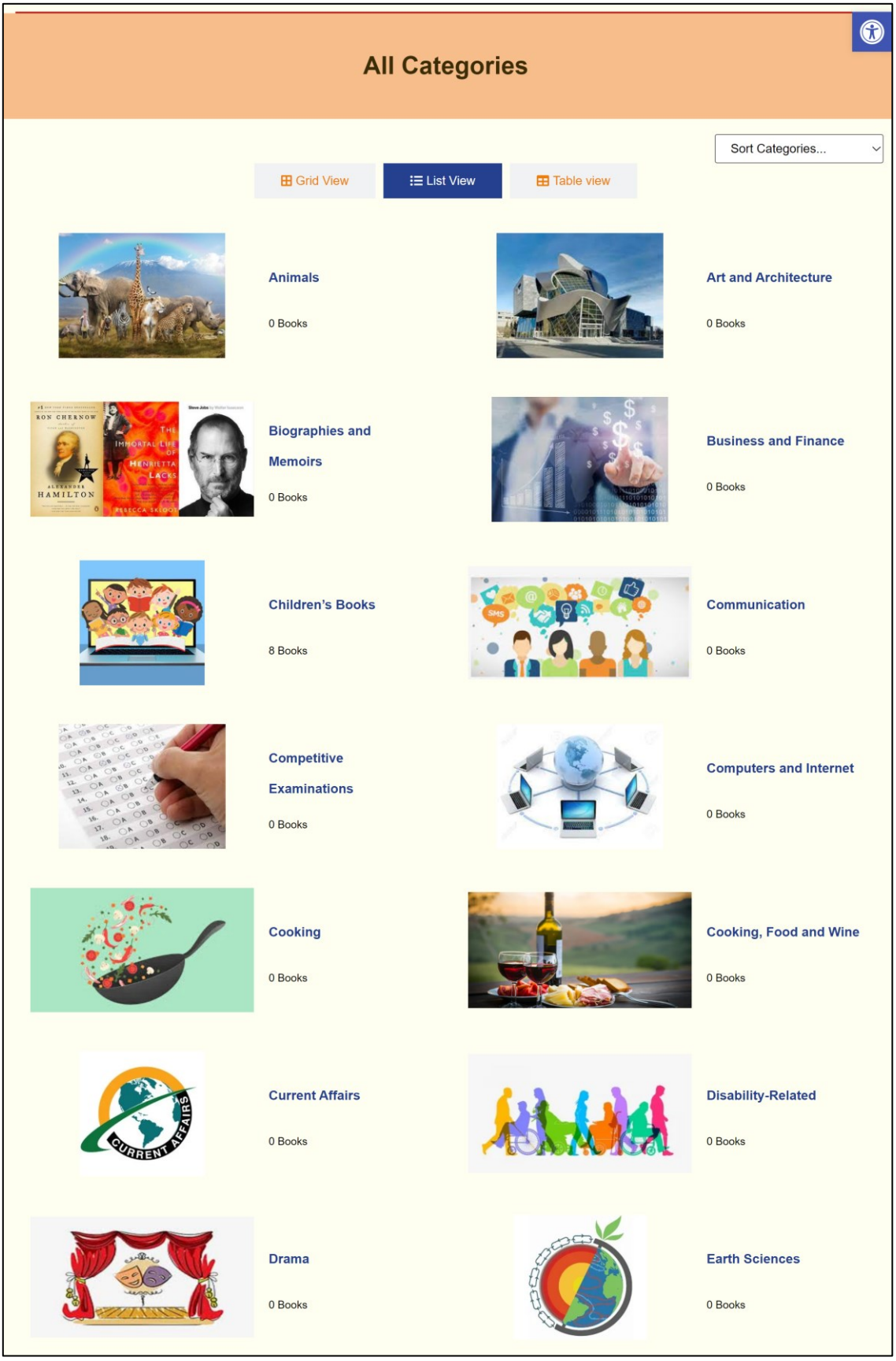
- Grid View (default)
- List View
- Table View

By Clicking on any of the Book Categories, they are redirected to a page with all the books that match that category.

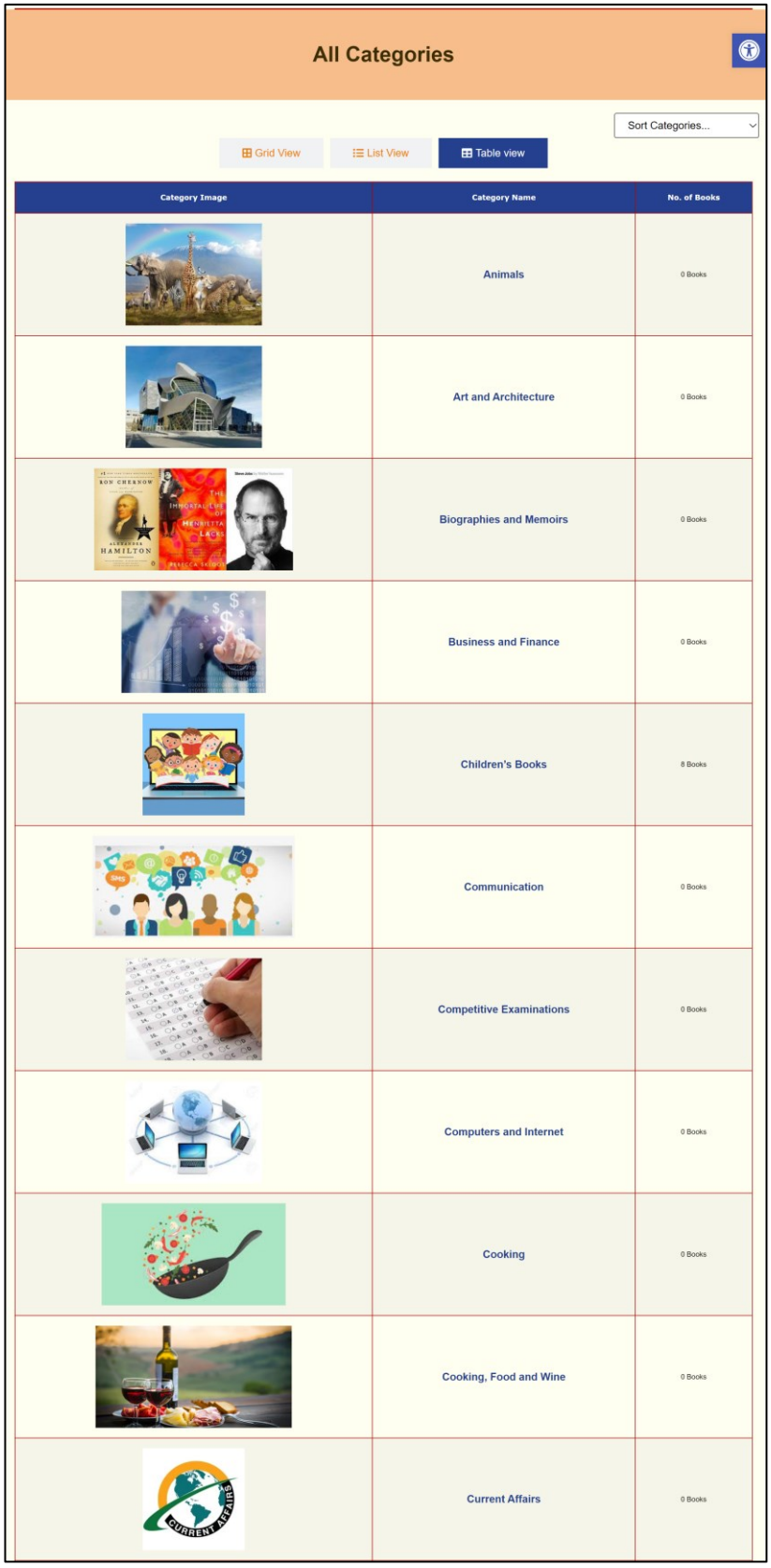
### 8.1.5.1 *Grid view*



### *8.1.5.2 List View*



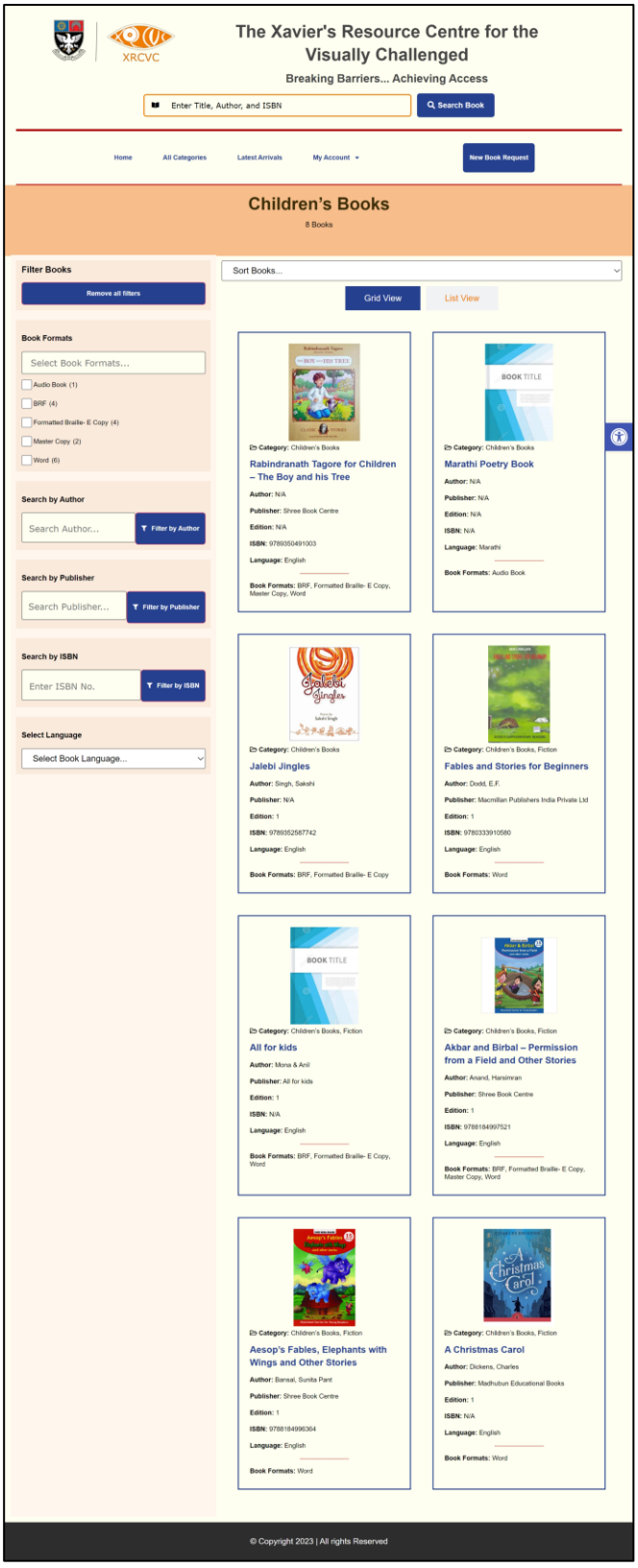
### *8.1.5.3 Table View*




The Student may also wish to Sort the Category by Alphabetically (Both Ascending and Descending).

### 8.1.6 Category Page

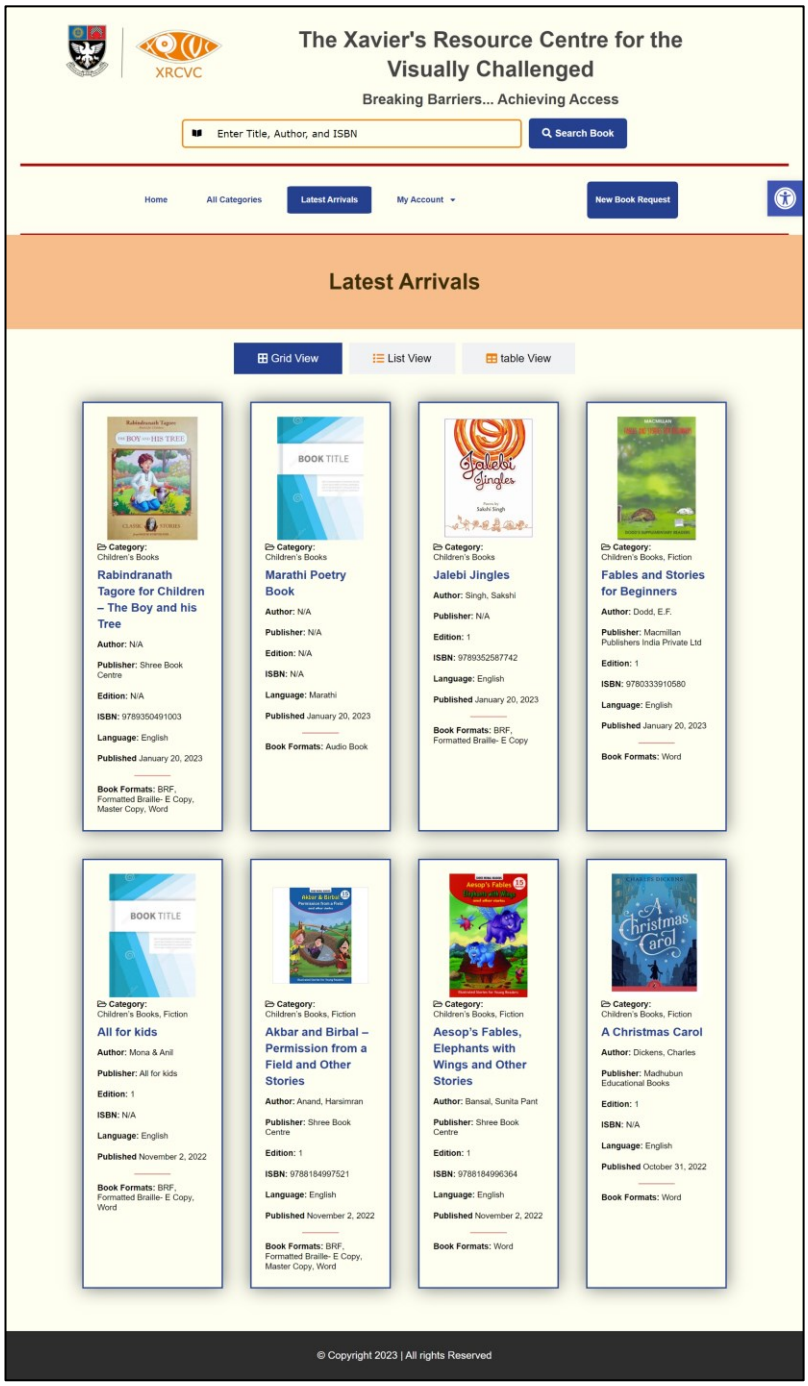
The Category Page shows the list of books that are present in that category with the option to narrow down the results by using the filter sidebar in the left. The Header shows the name of the category along with the Number of Books that are present in that category.



The student may choose to sort the Books in Alphabetical order or Date wise. The student may also choose to view the results in a Grid or List View.

### 8.1.7 Latest Arrivals

The Latest Arrivals Page shows the list of Books that have been recently added to the platform. The last 8 books are displayed in a Grid View with the option to view in a List or Table also.

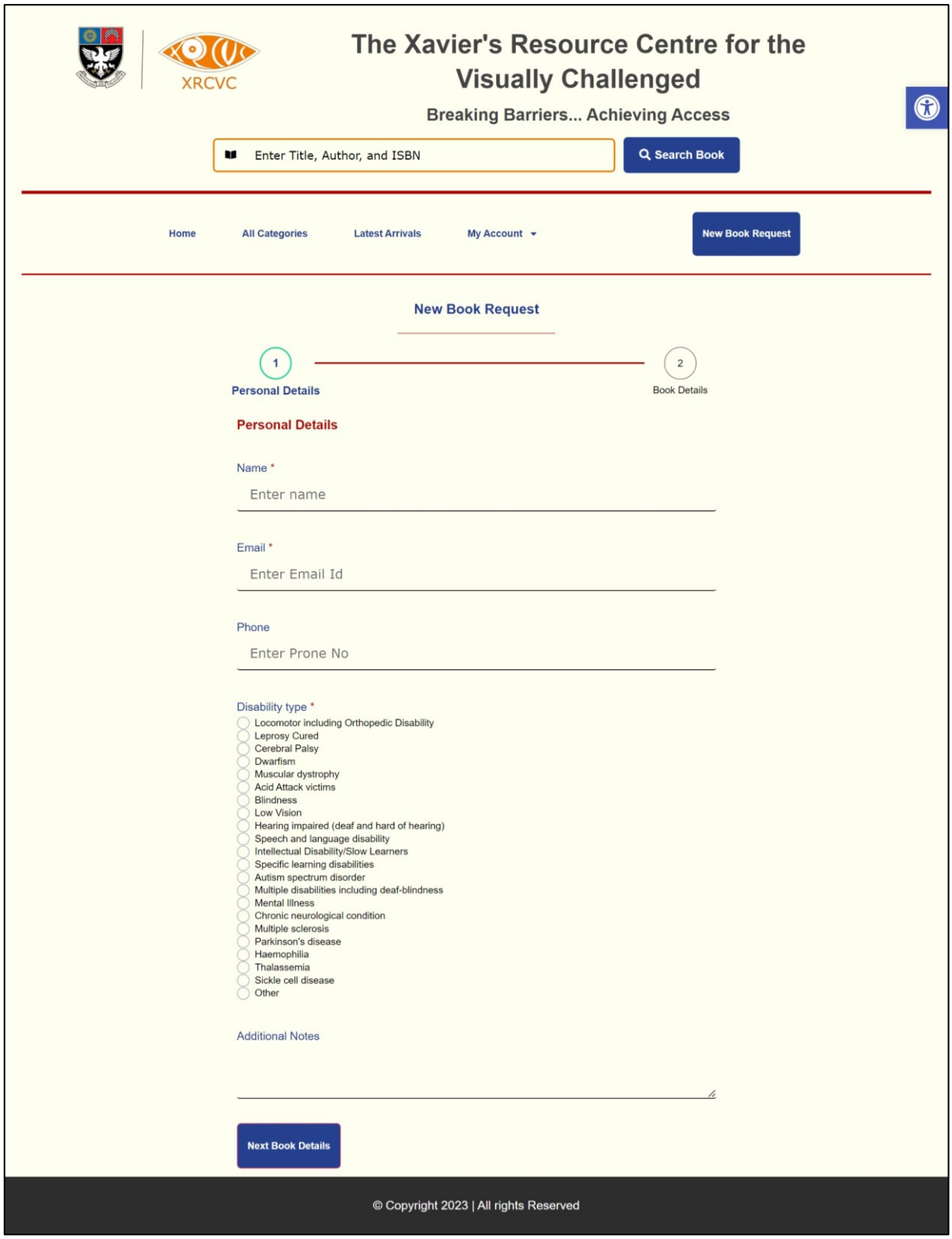



The Grid displays the Book's Cover Image, Title and basic details along with the published date for quick reference. When any book is clicked, the corresponding book's detail page would be shown.

### 8.1.8 New Book Request

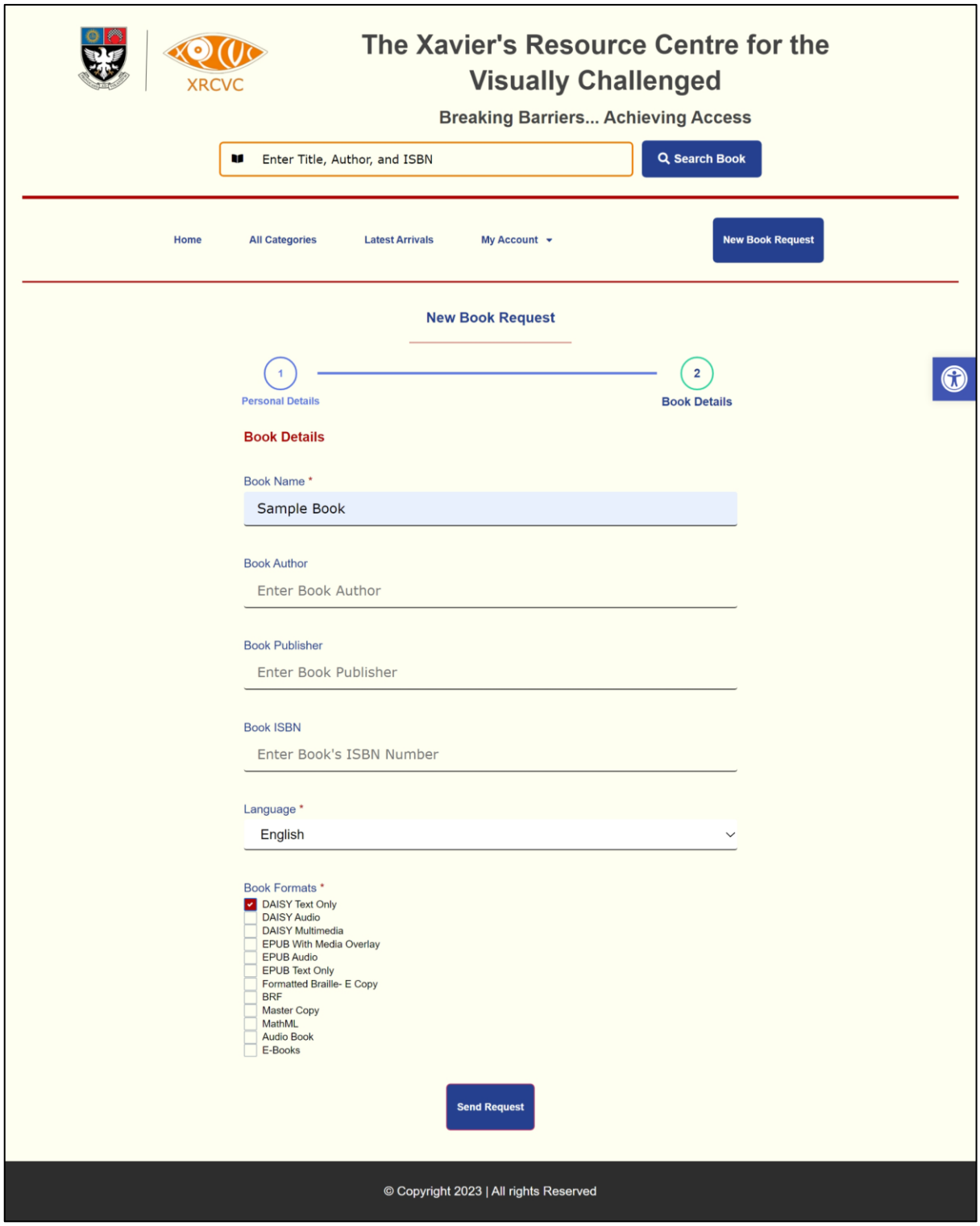
If a Student searches for a book and does not find the book that he/she is looking for, the student may opt to request the book to be uploaded on the platform by filling out a short form on the website.

#### *8.1.8.1 Personal Details*



The students must first enter their **Personal details** followed by the **Book Details**.

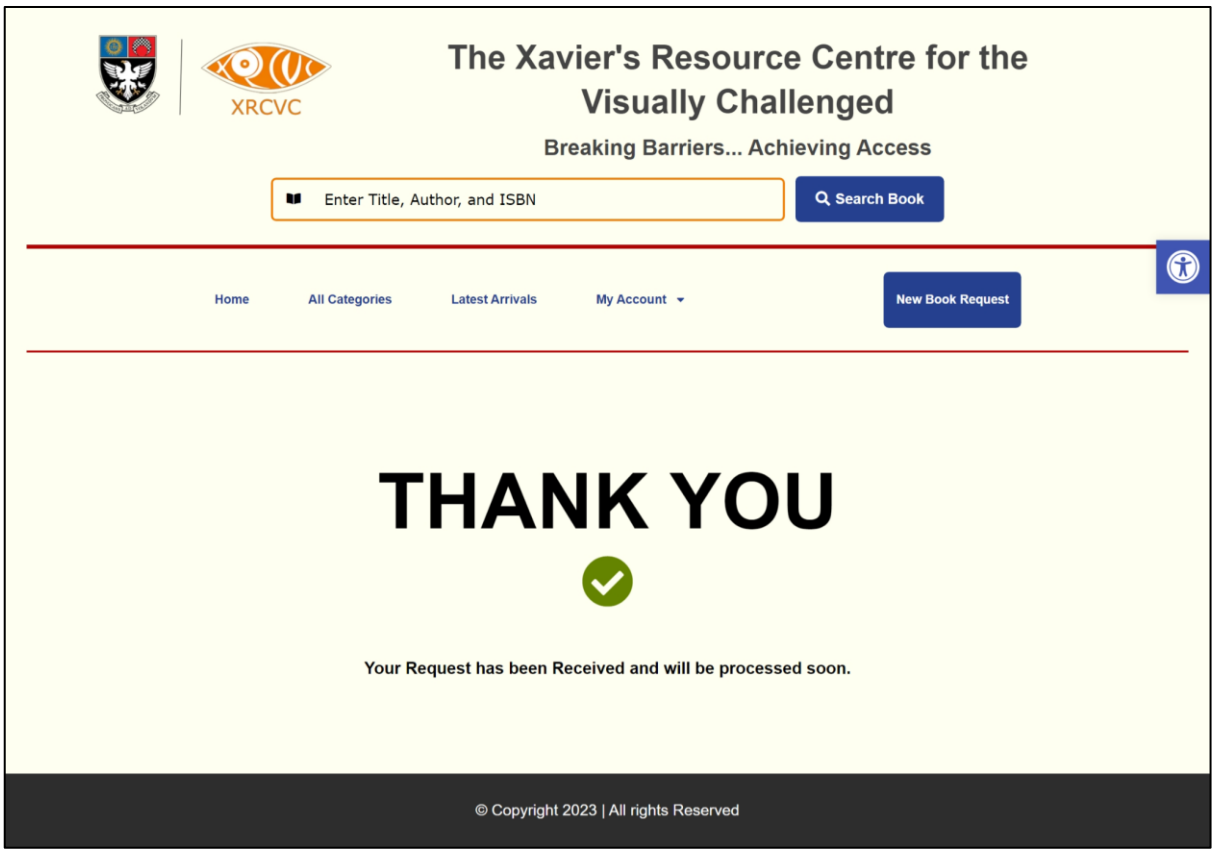
### *8.1.8.2 Book Details*




The student must enter the Book Title/ Name and the Formats in which the student wants the book to be uploaded, to successfully request the book from XRCVC. All other fields are optional.

### *8.1.8.3 Thank You Page*

After Requesting a new book to be uploaded on the platform, the student is redirected to a thank you page.



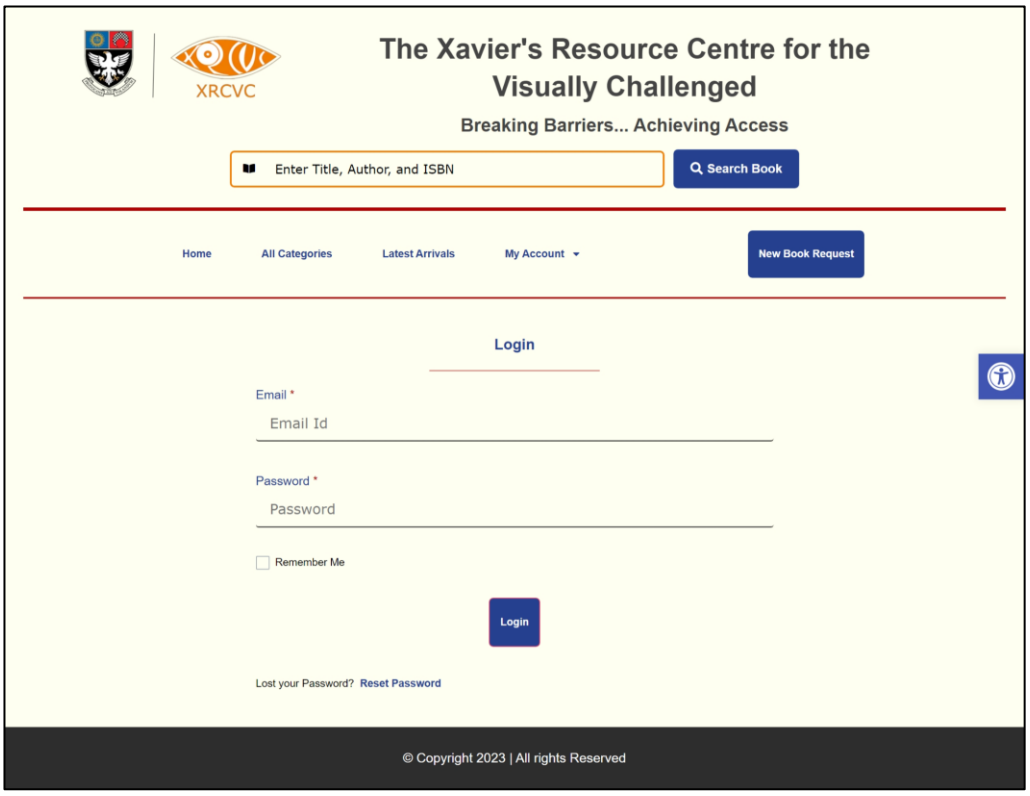

The admin of the website is also notified by email and a new Request is logged in the system.

## 8.2 Admin Modules (Frontend)

### 8.2.1 Login Page

#### *8.2.1.1 Custom Login Page*

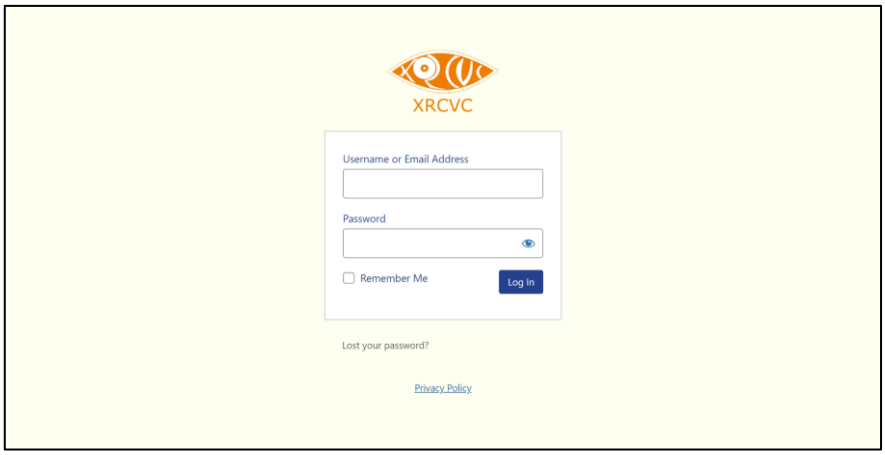
A site admin must login to the website to access his dashboard. From the Dashboard, the admin can update his Profile, Add New books, View Book Requests and Manage New Book Requests.




The Custom Designed Login page allows the users to login by just using their email ID and Password.

#### *8.2.1.2 Default WordPress Login Page*

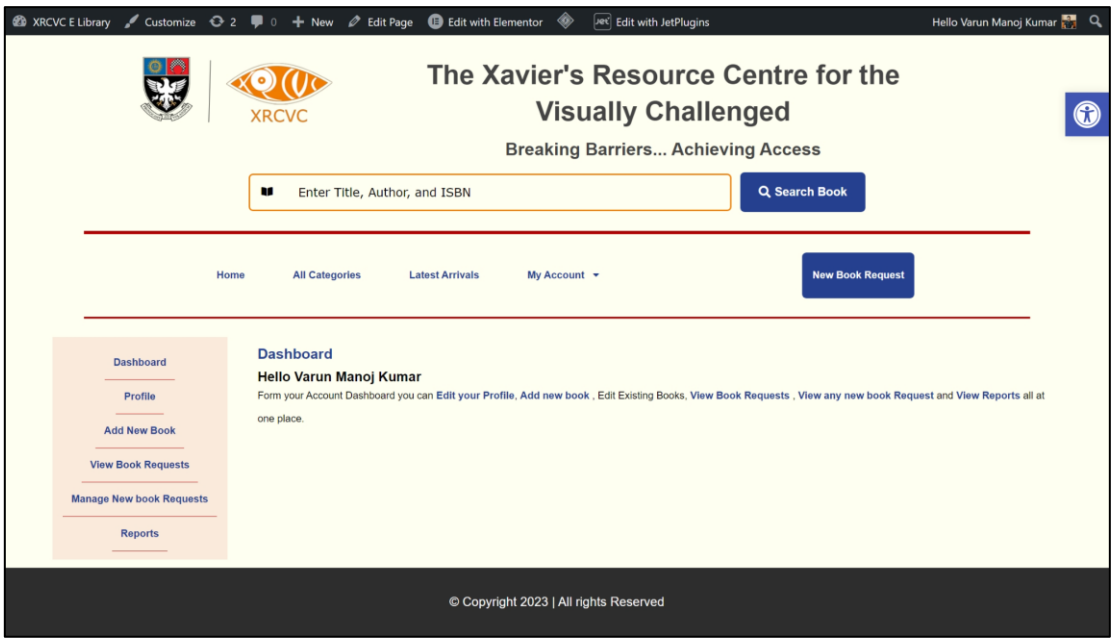
The WordPress's Default Login page is used in case of any error during logging in to the website.



The admin can login by using their username or email id and password.

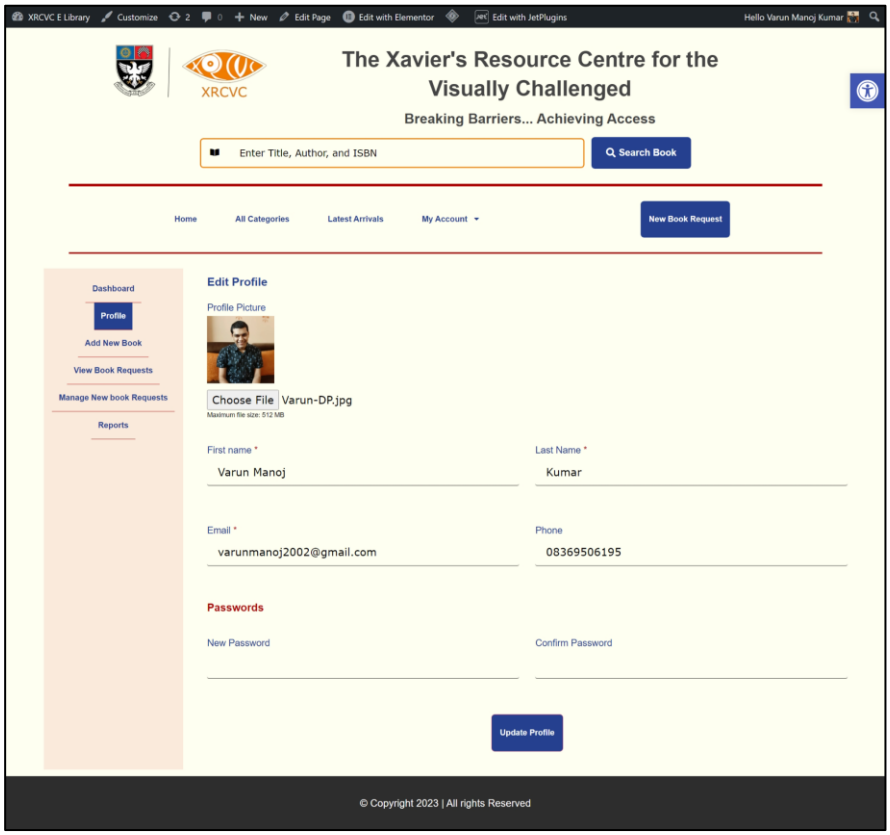
### 8.2.2 Dashboard

The Dashboard Provides quick links to all the major functions that the admin can perform once logged in.




### 8.2.3 Profile

Form the Account Profile, The admin can change their name, upload their profile photo, view his email Id, update his/her phone number and also change their account login password.

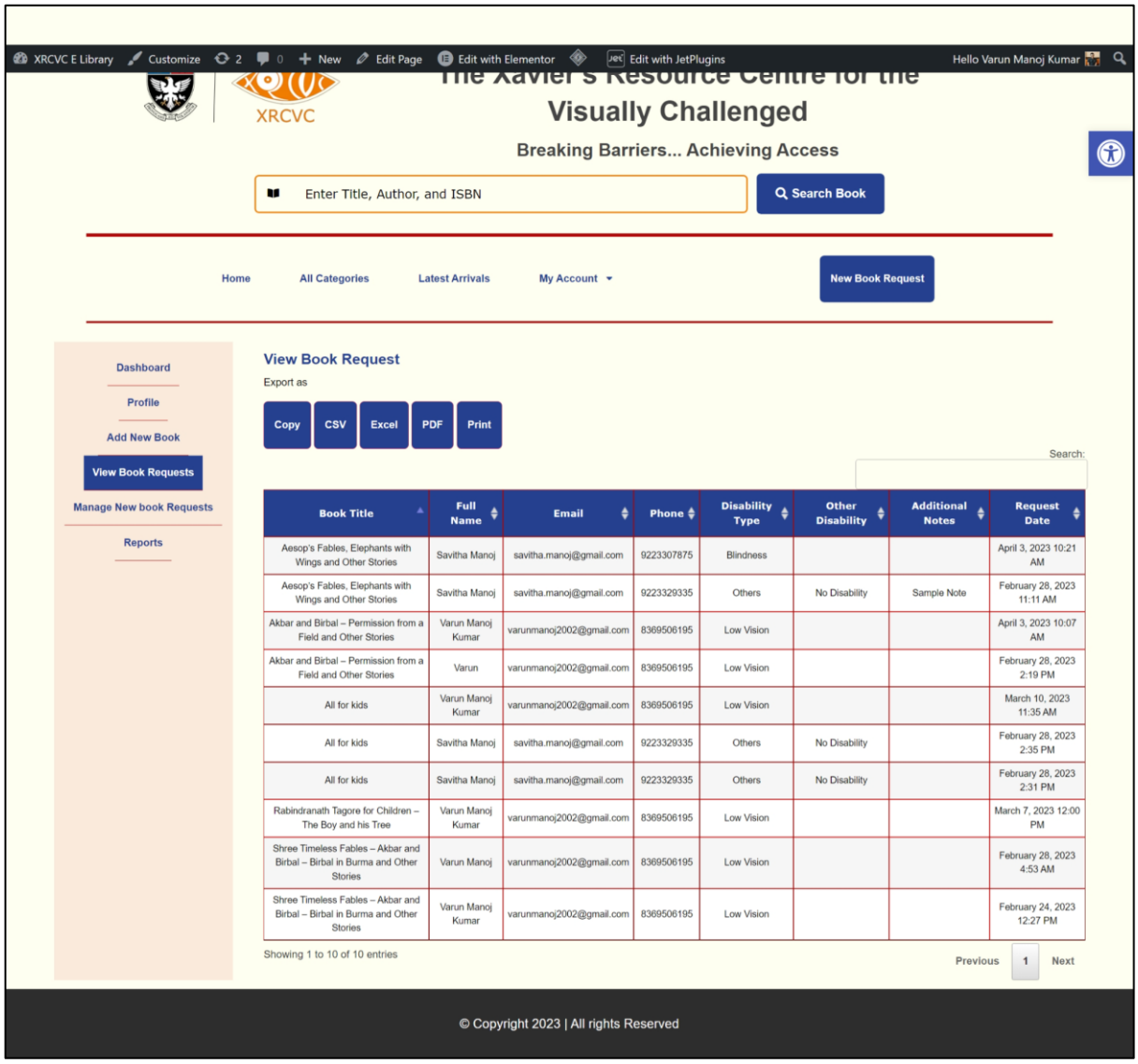



### 8.2.4 Add New Book

The admin can add a new book from the front end by filling out the form. The admin must fill out all mandatory fields to upload a new book to the platform.

### 8.2.5 View Book Request

The Admin can view the list of all books requested sorted by Book Name.

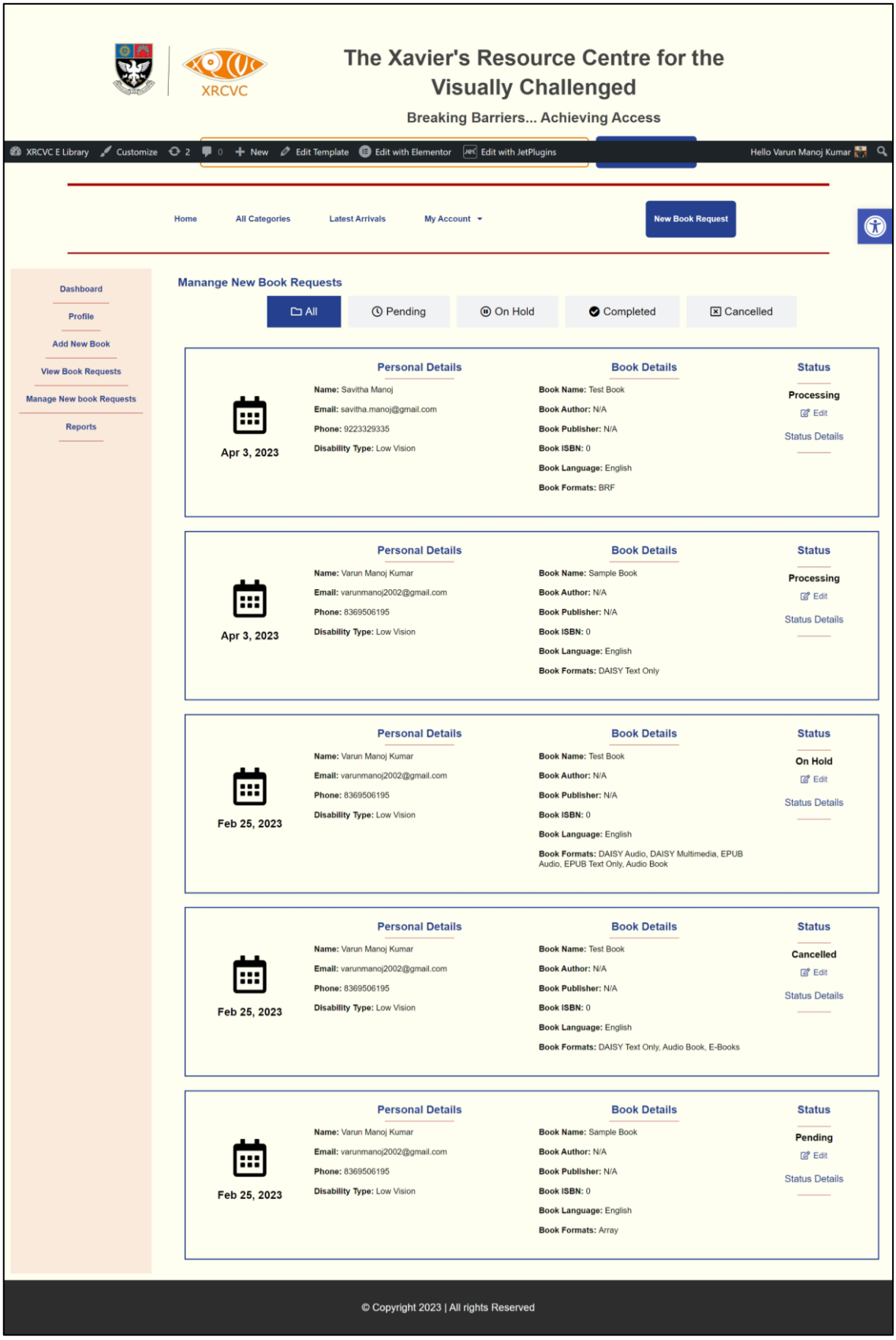



The admin can also sort the same table by Date or any other column. Moreover the admin can also export the same content in the form of an Excel workbook, CSV file or print it or just copy the contents to the clipboard.

### 8.2.6 Manage New Book Request

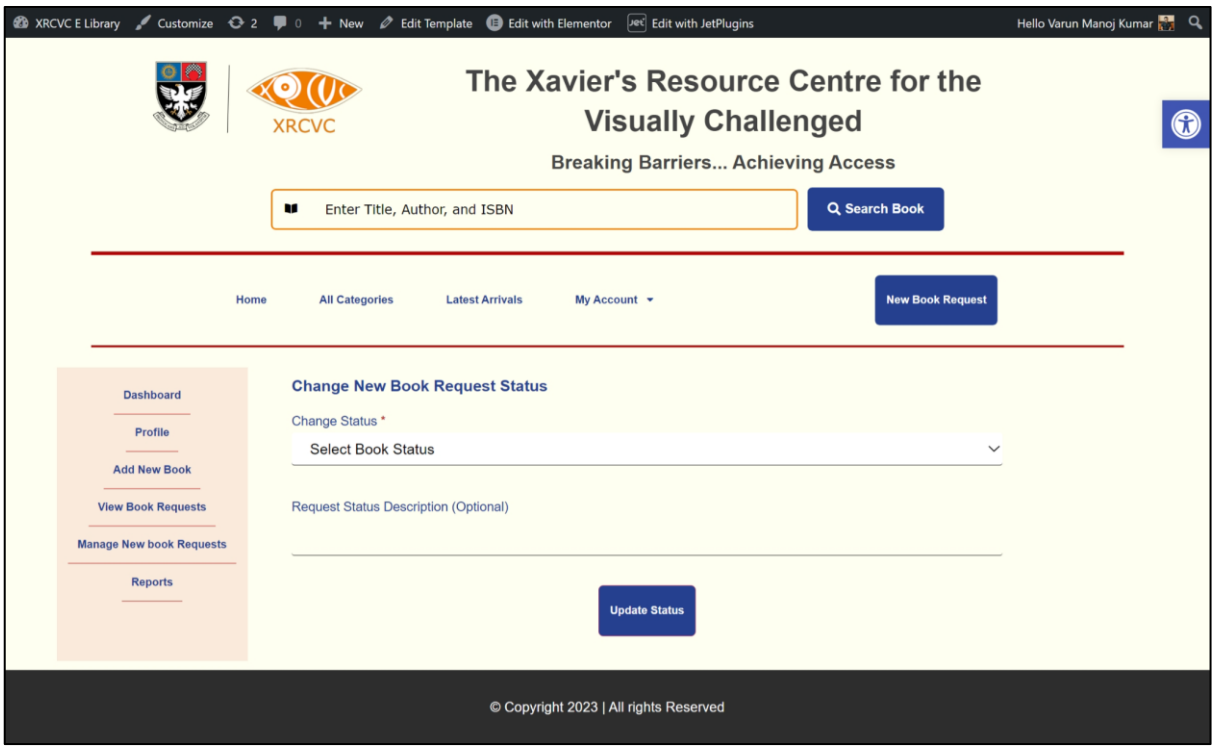
The admin can manage the new book requests from the dashboard. The admin can view the list of new book requests, organize them into various statuses and provide feedback on each of them.

### *8.2.6.1 View New Book Request*



### *8.2.6.2 Edit New Book Request Status*

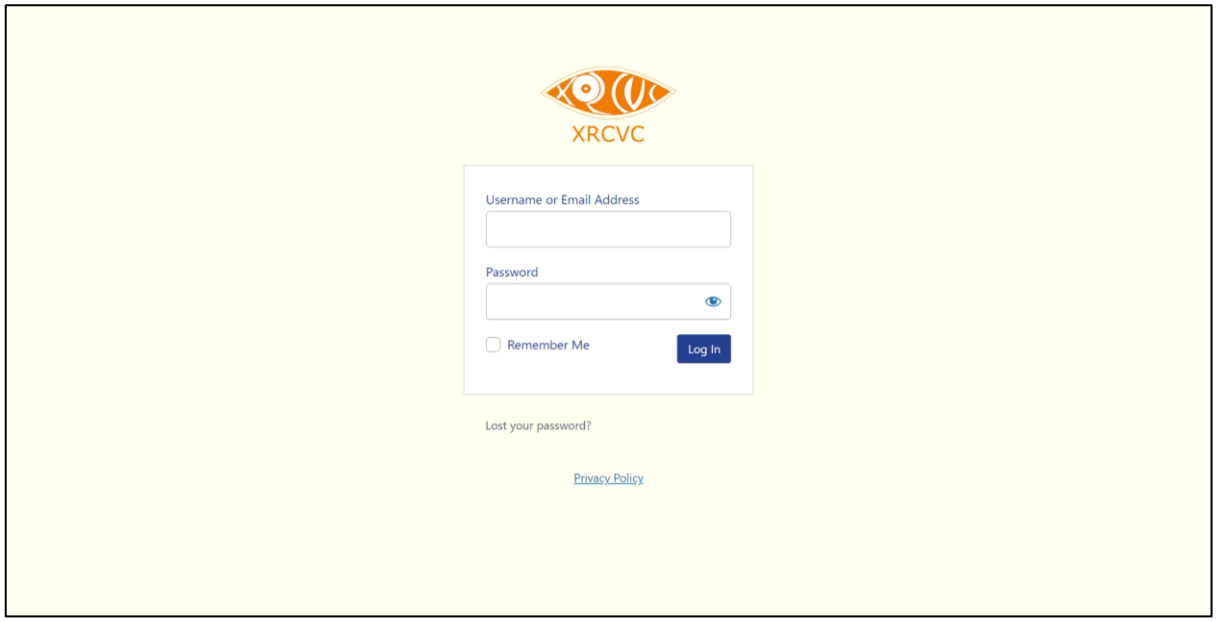
The admin can assign status to the New Book Request by filling the form. Moreover the admin can also add additional notes.



## 8.3 Admin Module (Backend -WordPress)

### 8.3.1 Login

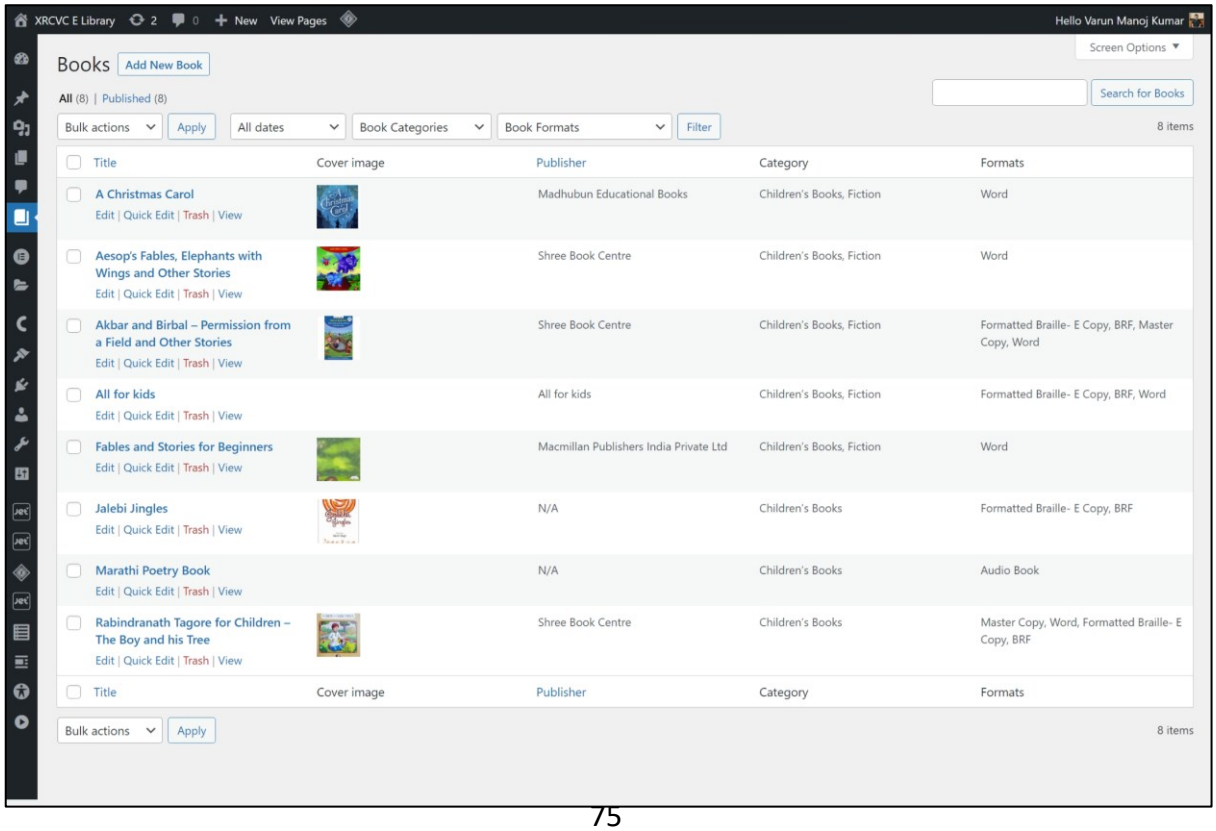
The Default WordPress Login Page




The Default Login Page has been customized in keeping with the XRCVC's Colour Scheme and minimalistic approach of the website.

### 8.3.2 All Books List

**'Books'** is a **Custom Post type** designed in WordPress which is used to store all the Books and relevant information all in one centralized location.

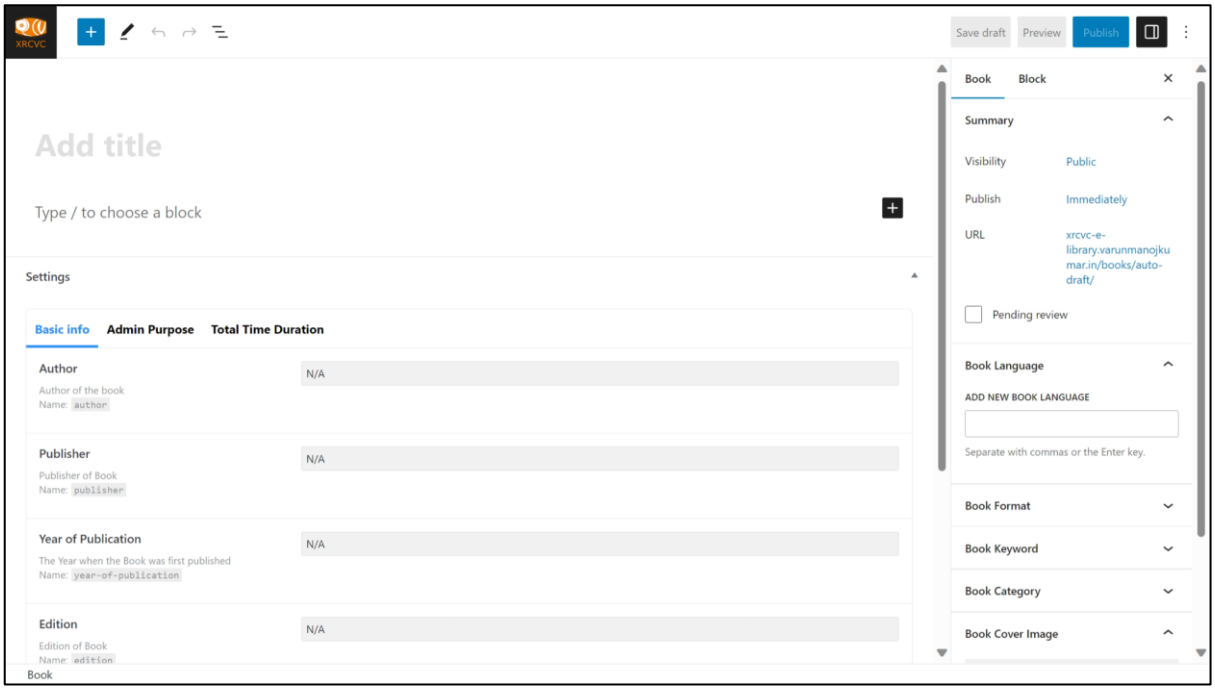


From the All-Books Page, the admin can

- View the list of all books on the website.
- View the details page of any of the books on the website.
- Add New Book
- Edit the Book details.
- Delete the Book (Move the Book to Trash).
- Search for a book
- Sort the Book by clicking on the corresponding column header name.
- Filter the Book by Category, Formats, Published Month and Year, etc.
- Perform Bulk Operations

### 8.3.3 Add New Book

The Add New Book Page can be used to add a new Book to the Website from the backend. Once all relevant Information is filled and published, the Book will be available to view in the frontend.



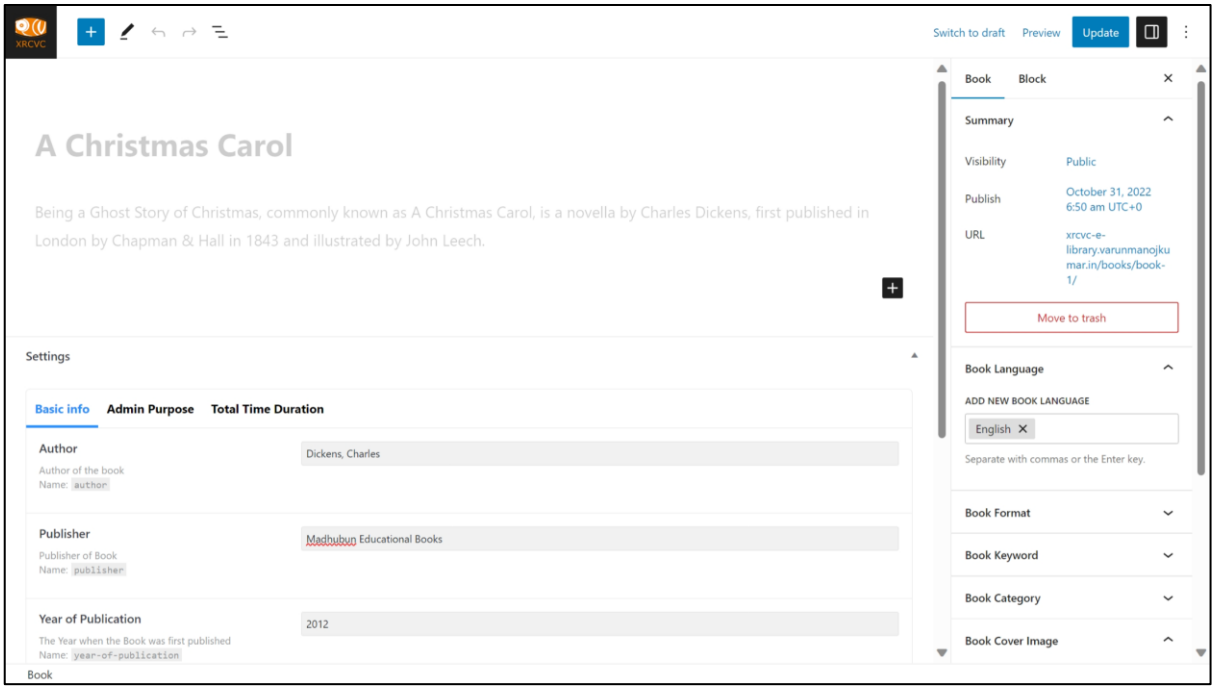

In the Add new Book Page, the admin must provide:

- Title of the Book.
- Description of the book
- All mandatory meta fields.
- Language Book
- Book Formats
- Book Category
- Book Cover Image

The Add New Book Page also shows the Published Status, the Book's public URL (Permalink) once the book is published.

### 8.3.4 Edit Existing Book

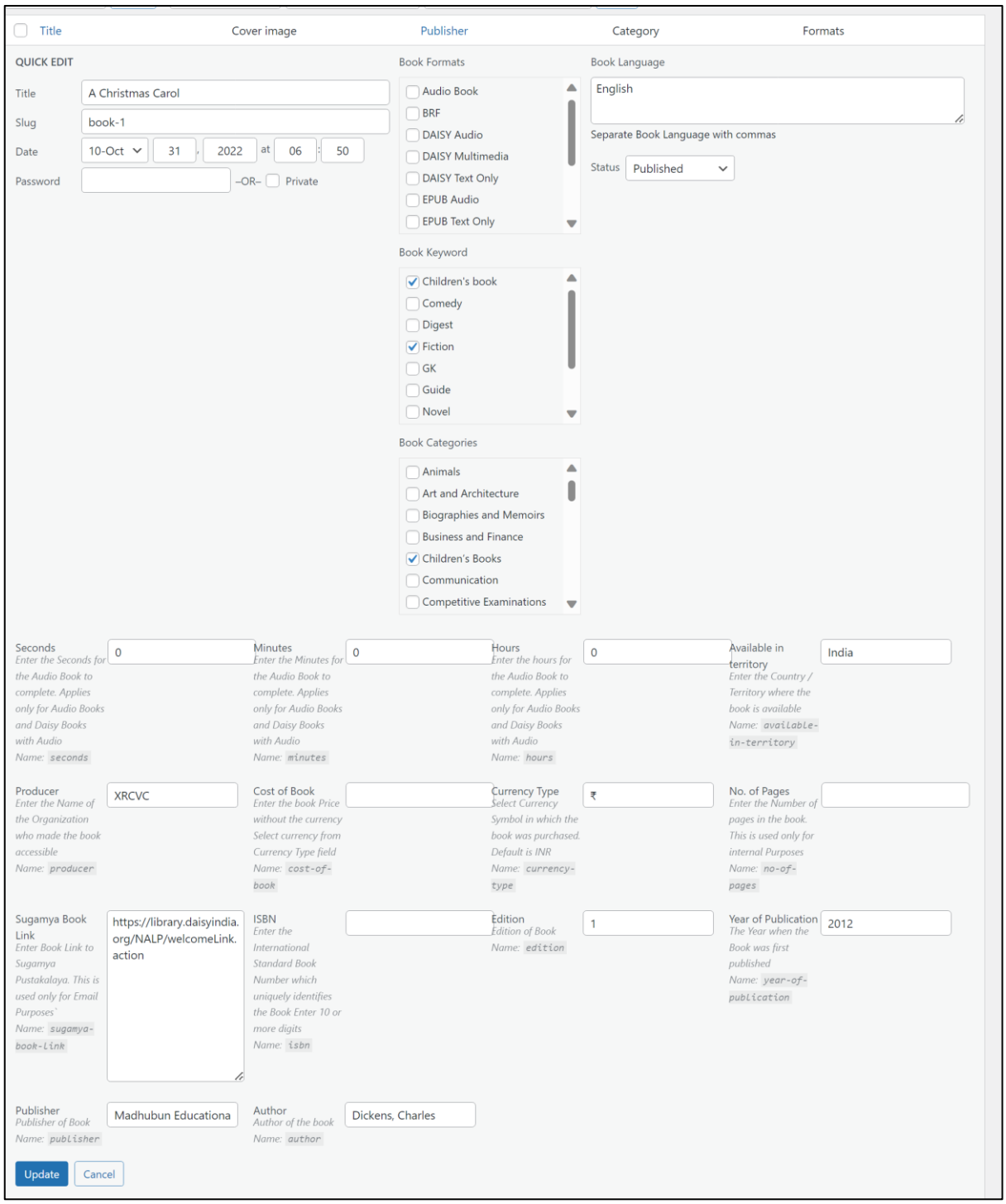
The admin can Click on Edit Book to Edit any existing Book's details and update the book at the frontend.




All the data can be modified from the backend and updated when done. The admin can also change the status of the Book from Published to draft, or even delete the book by moving it to the Trash. The admin can also preview the changes before publishing it on the website.

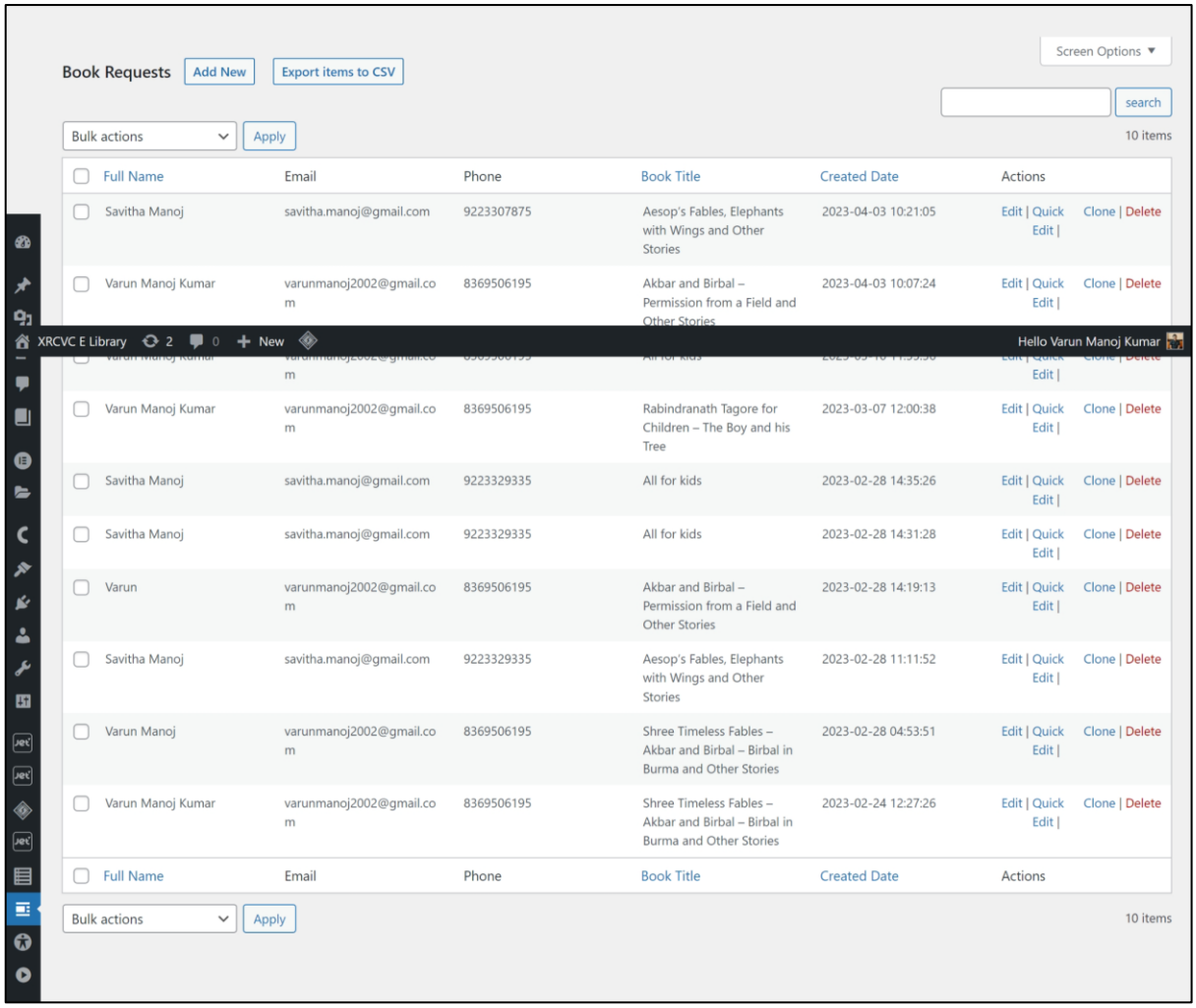
### 8.3.5 Quick Edit (Backend Exclusive)

From All Books page, the admin can also quickly view all important information quickly without entering the detail edit Page.



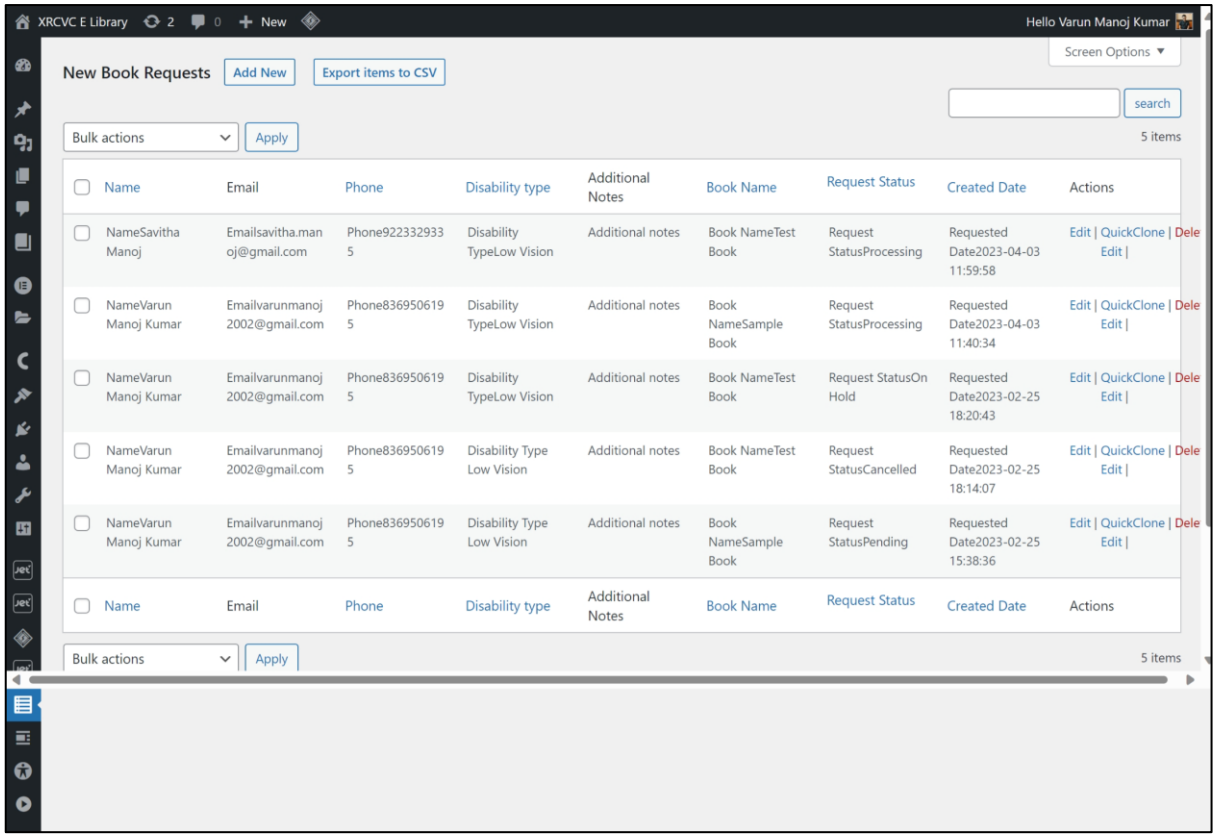
### 8.3.6 View Book Request

View Book Request is a **Custom Content Type** (CCT) defined in WordPress from the WordPress Backend, the admin can view the list of Book Requests in a table View and Export the information in a CSV file.



### 8.3.7 Manage New Book Requests

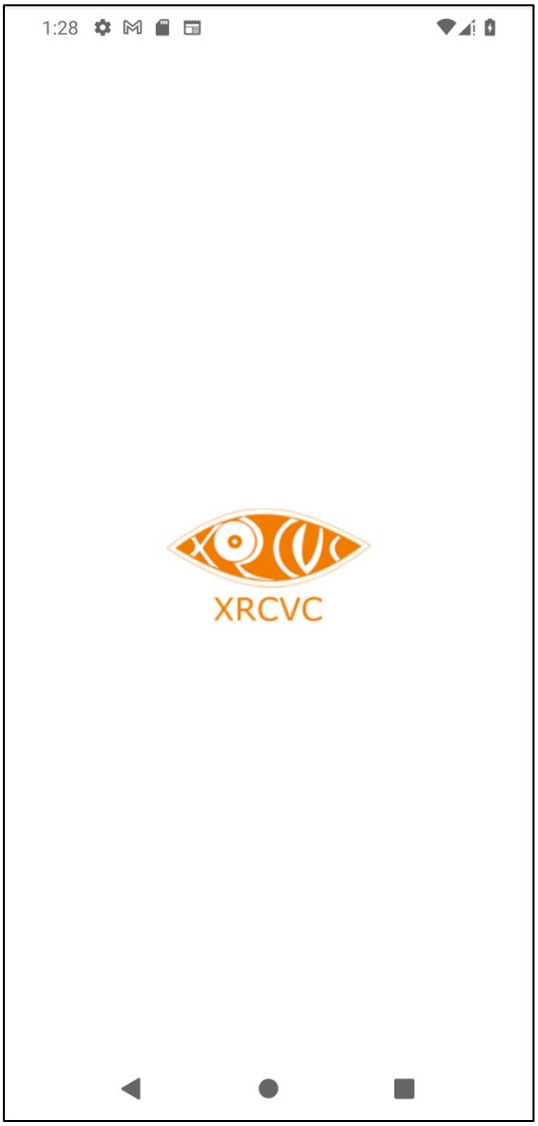
Manage New Book Request is a **Custom Content Type** (CCT) defined in WordPress from the WordPress Backend, the admin can view and manage the list of New Book Requests in a table View and Export the information in a CSV file.



## 8.4 Android App

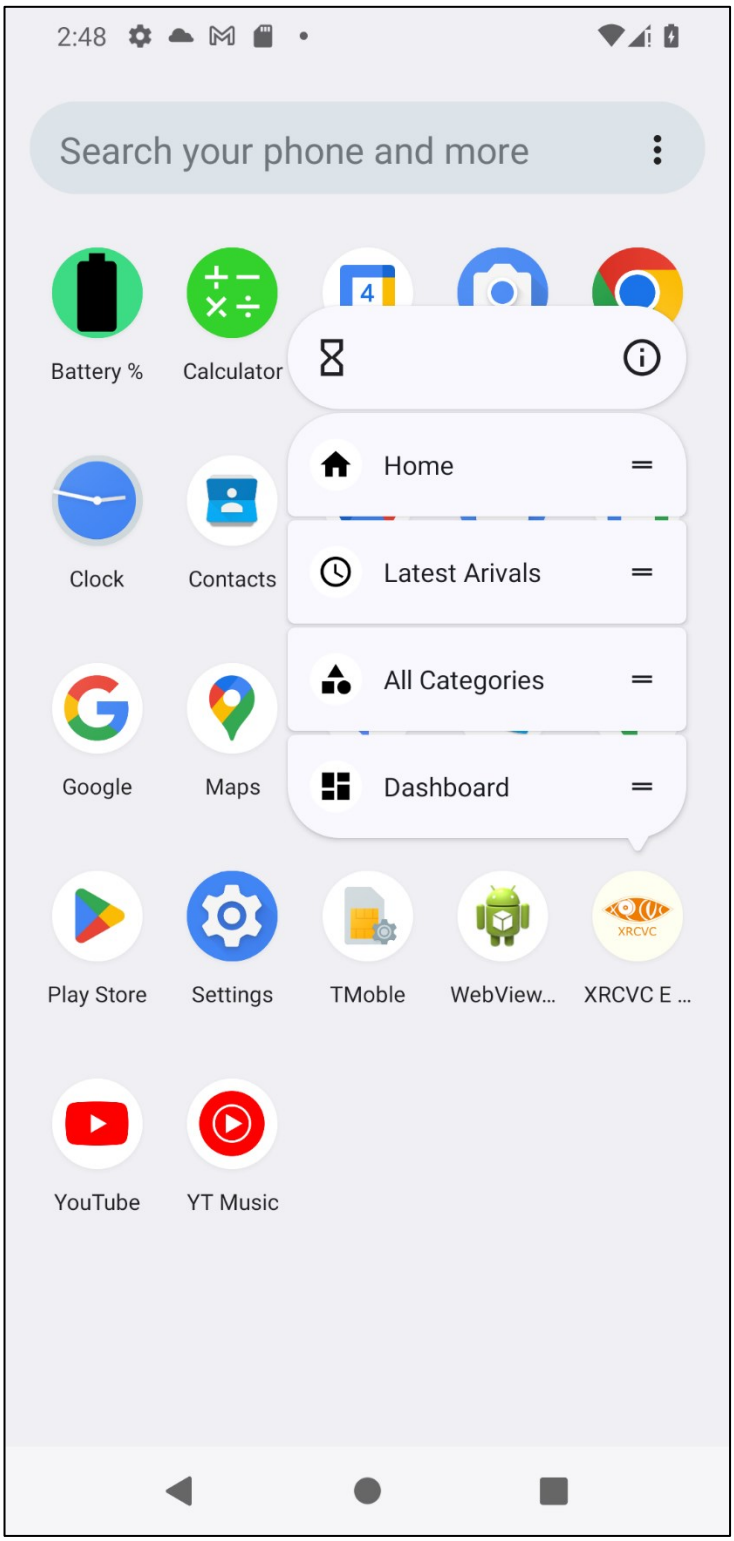
The XRCVC E Library is an android application that can be used to quickly access the website on the android devices.

### 8.4.1 Splash Screen (Default Android 12 and above)



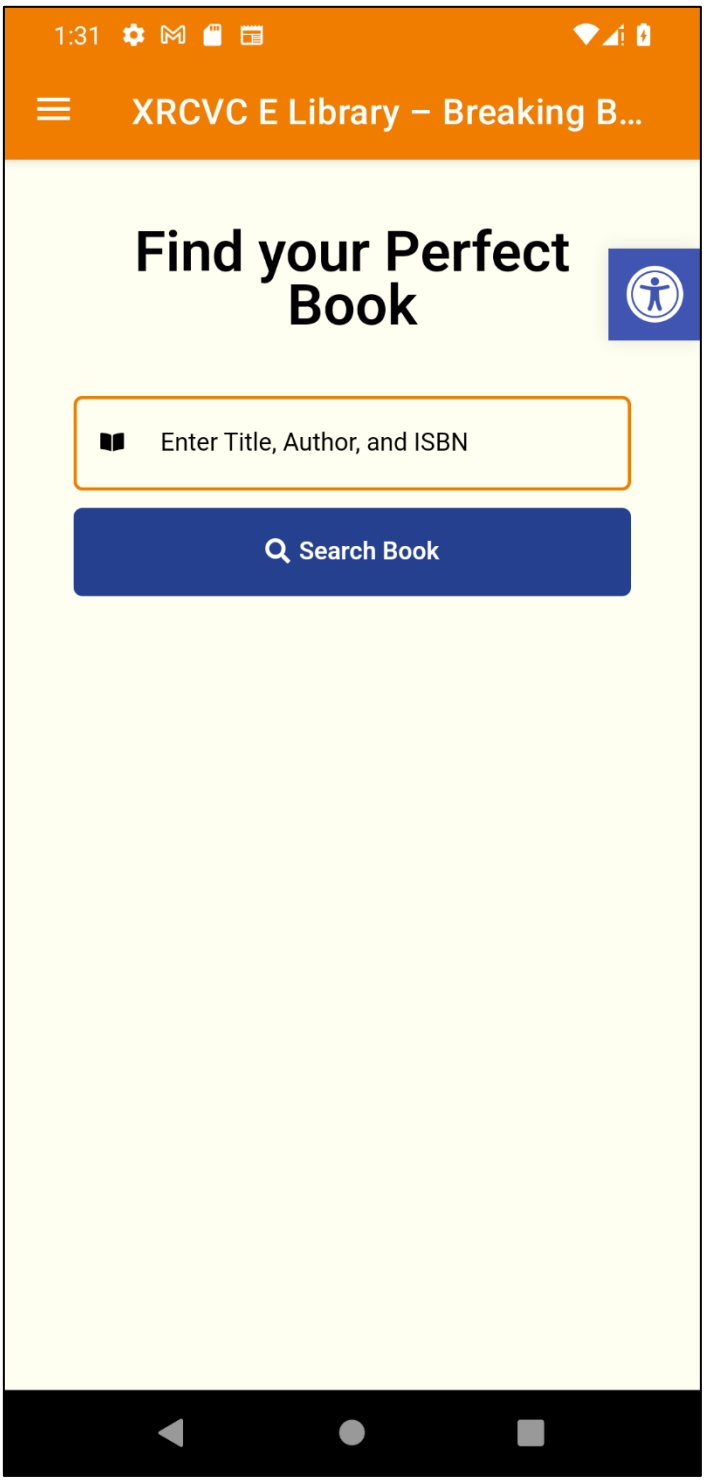

The Default Splash screen appears indicating that the application is launching.

### 8.4.2 App Shortcuts




One may long press on the app icon to reveal the shortcuts that allow the user to quickly and directly navigate to the desired page.

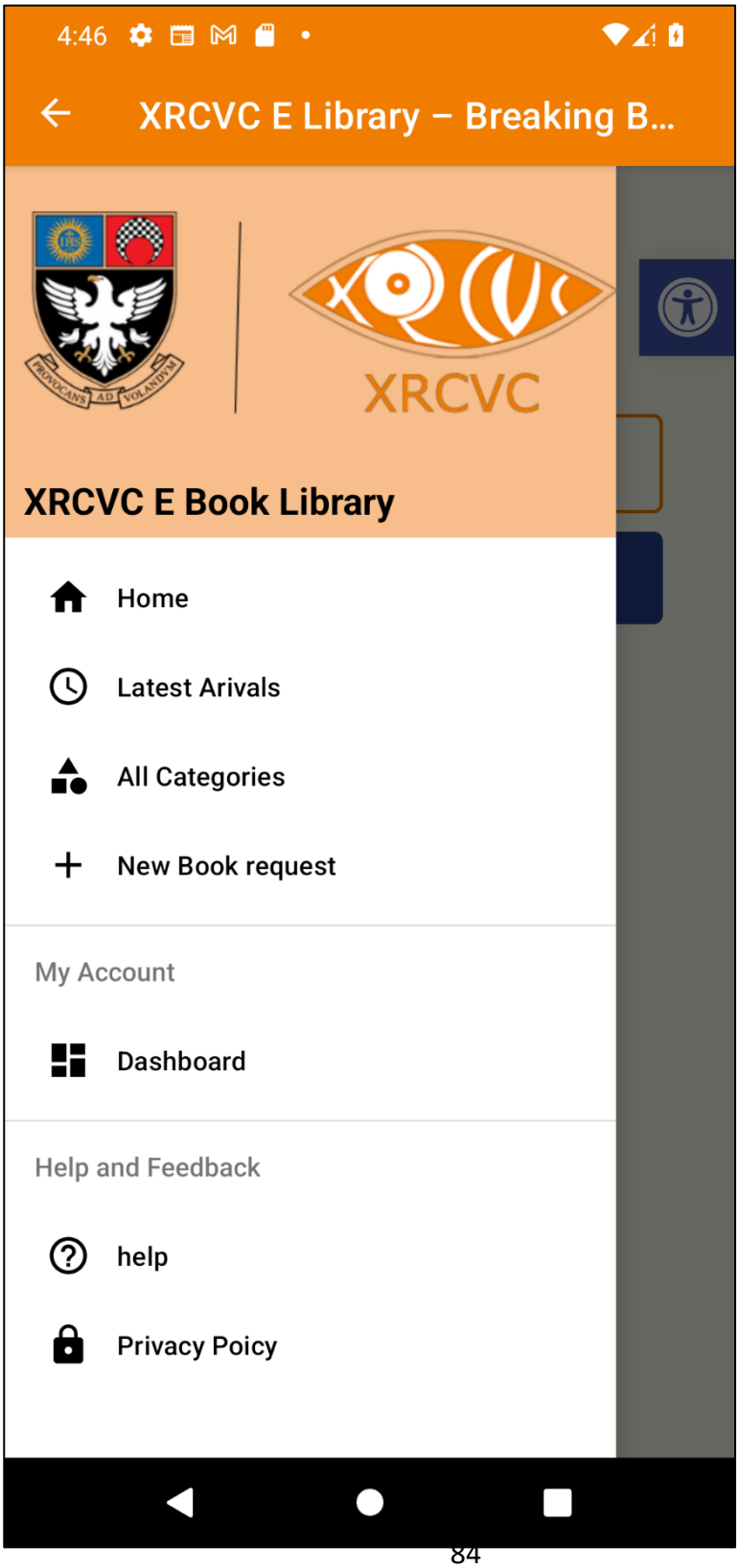
### 8.4.3 Home Page




Once the application is launched, the student is greeted with the Home Page. From the Home page the student can search for a book and request it from XRCVC.

### 8.4.4 Hamburger Menu

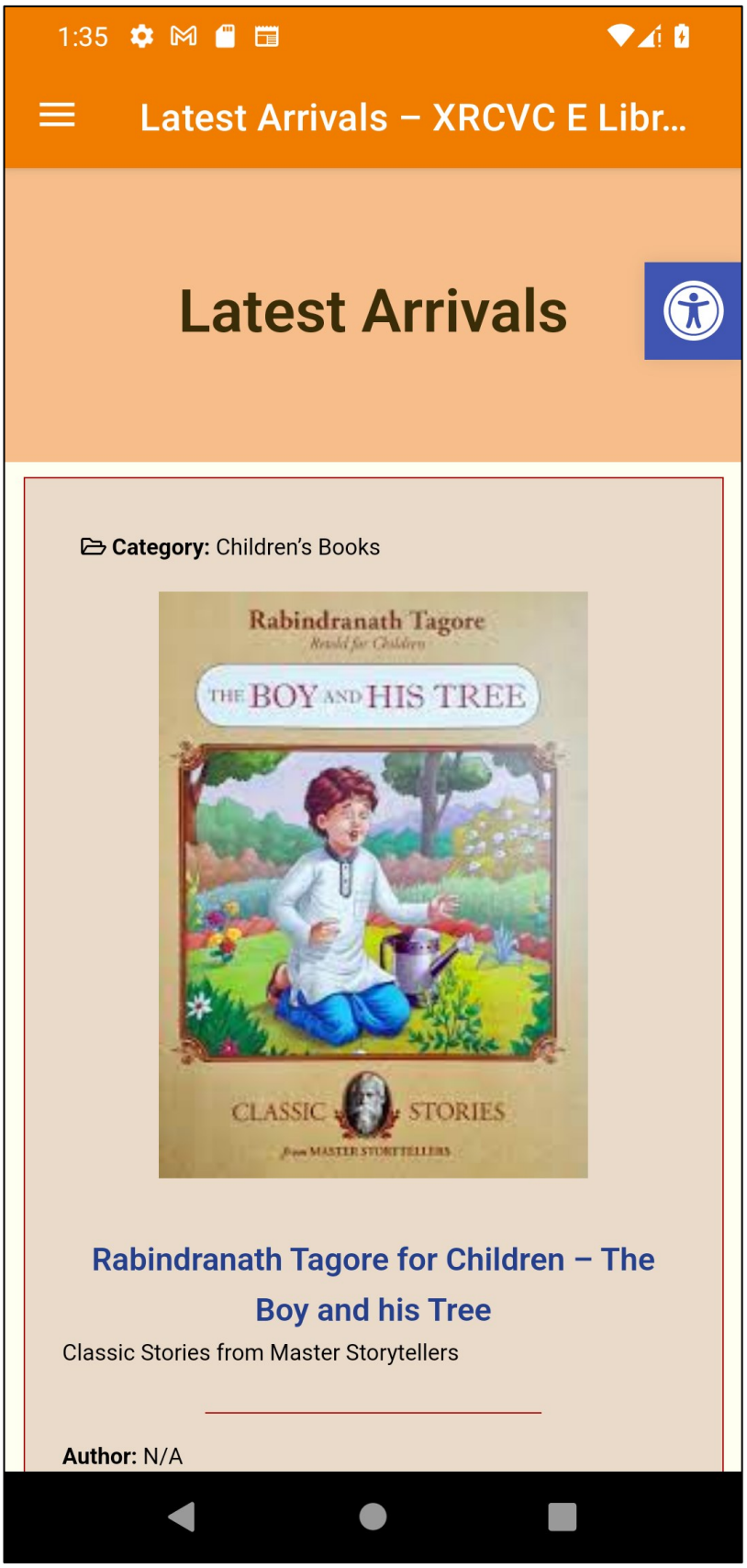
The Hamburger menu is used to navigate from One Screen to another in the android App. It provides Links to all other screens.



8.4.5 Search Result page

8.4.6 Book Details page

8.4.7 Latest Arrivals

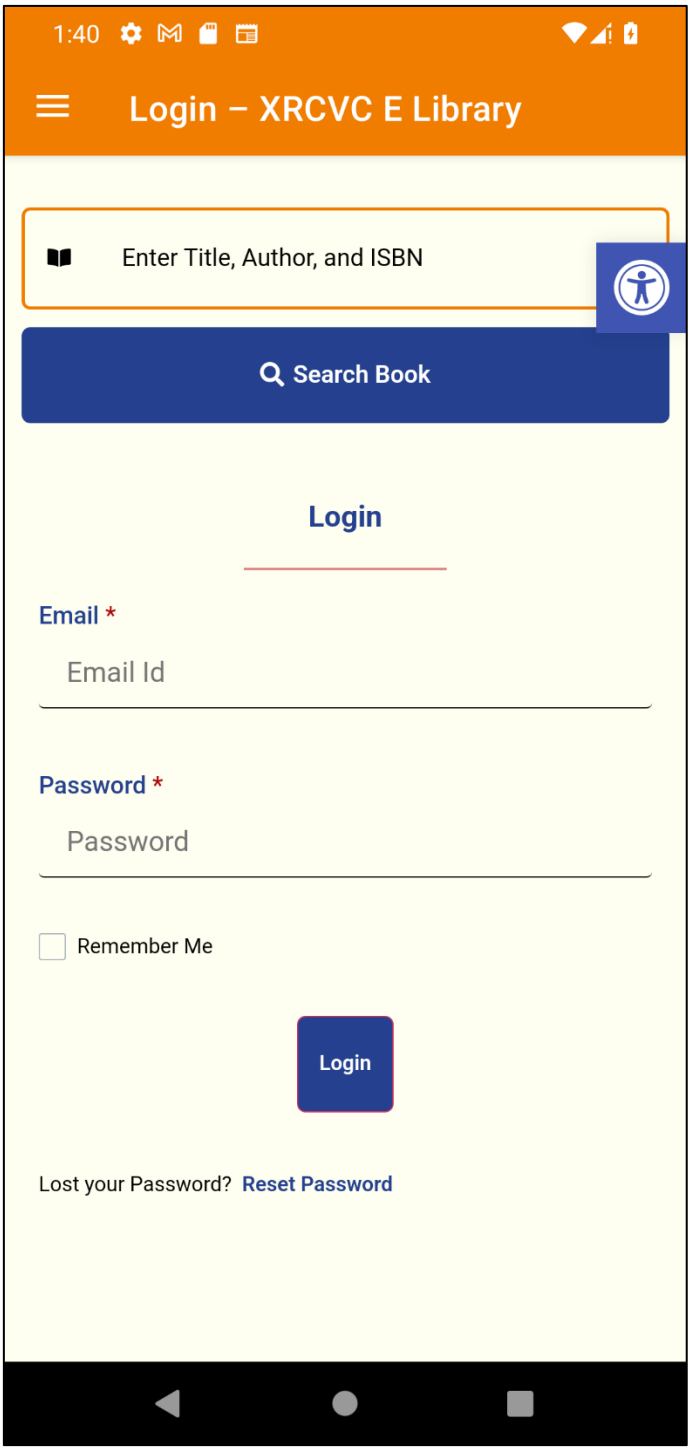



From the Latest Arrivals page, the student can view the top 8 most recent books that have been added to the app.

### 8.4.8 All Categories

The All-Categories Page shows the list of all the categories that are present in the application in a Table View. It also shows the number of books that are present in each category.

### 8.4.9 Admin Login

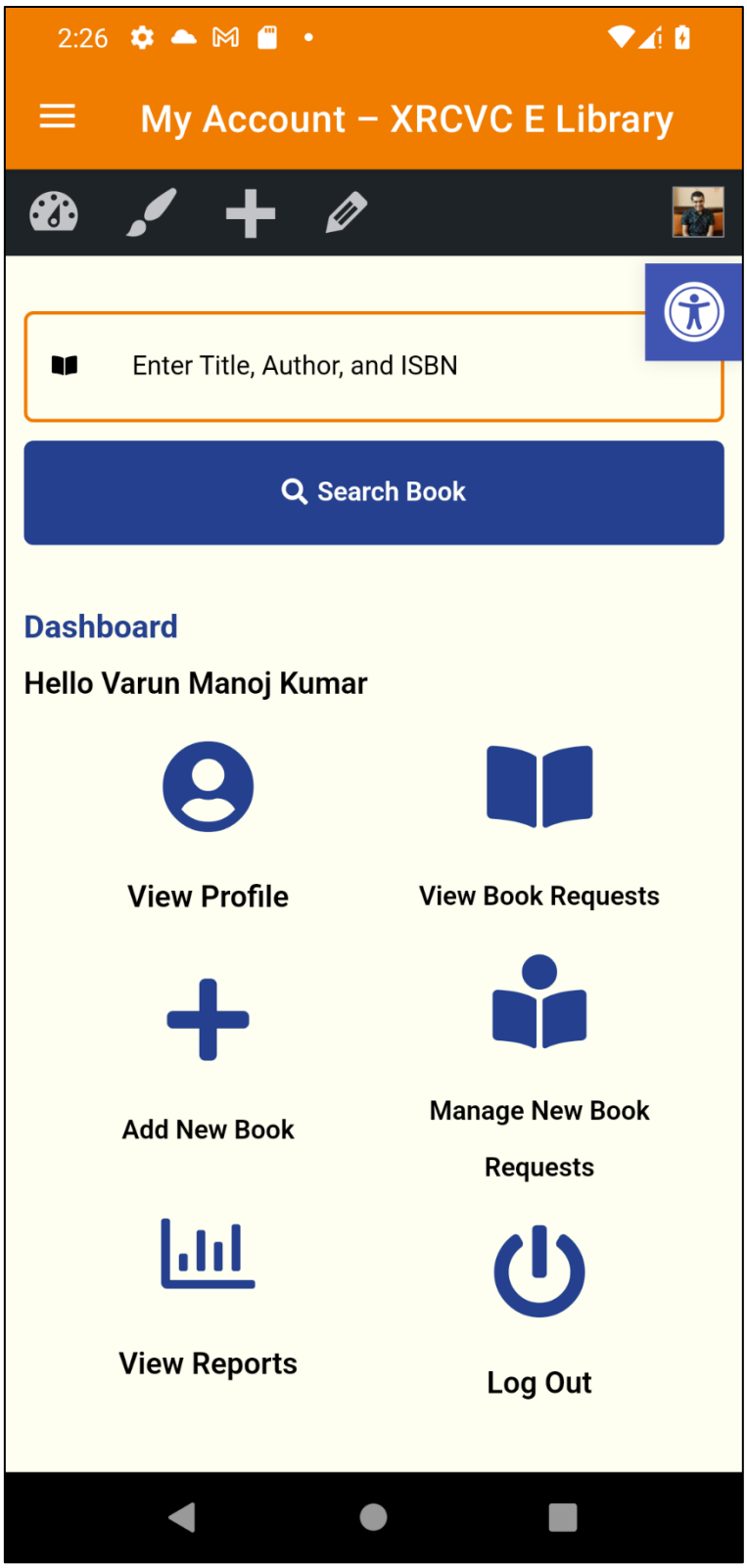



The admin can login to perform the admin functions like

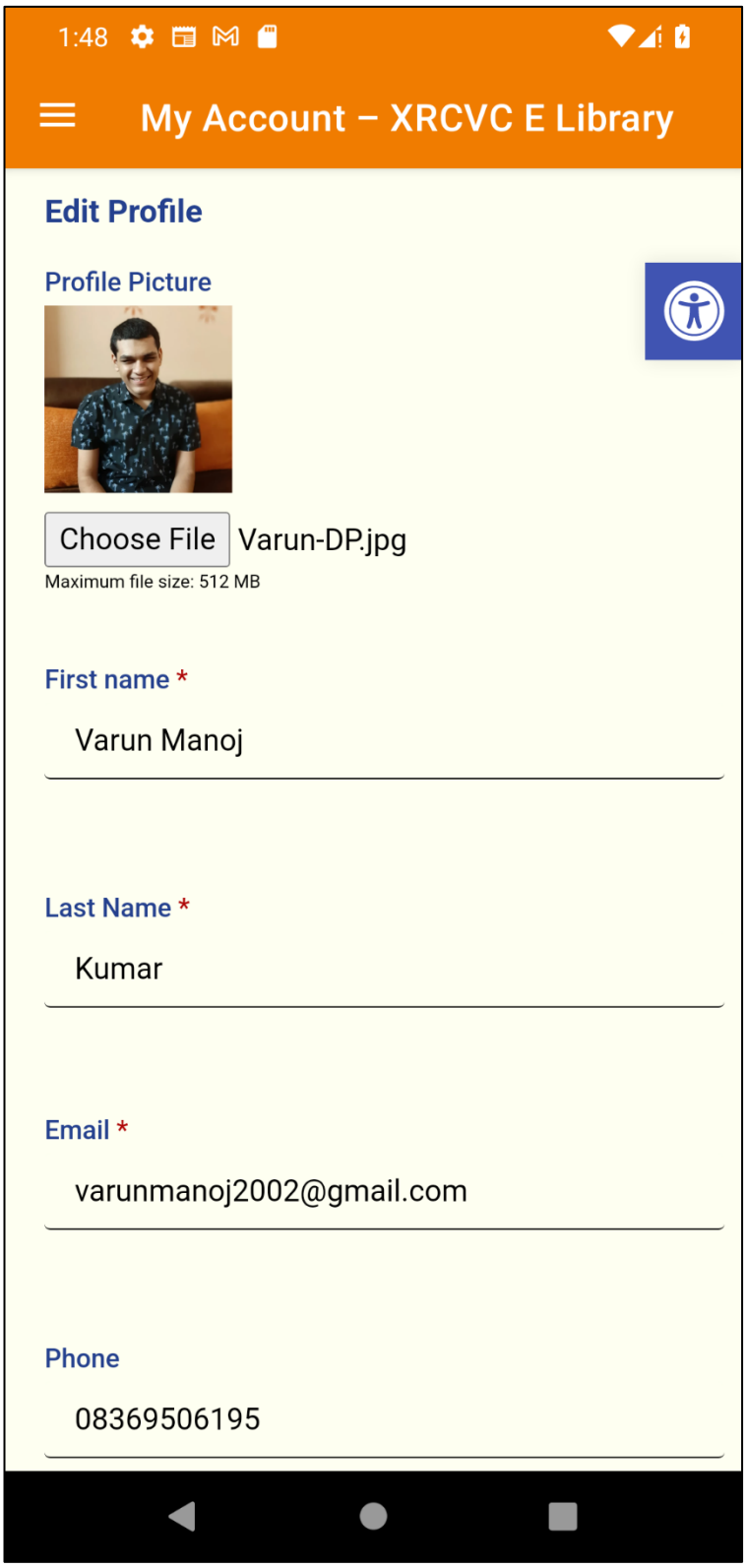
1. Add New book.
2. Update their profile.
3. View Book Requests
4. Manage New book requests.

From the Login page, the admin must enter the email Id and password to login. The admin is also given to option to reset his/her password.

### 8.4.10 Account Dashboard



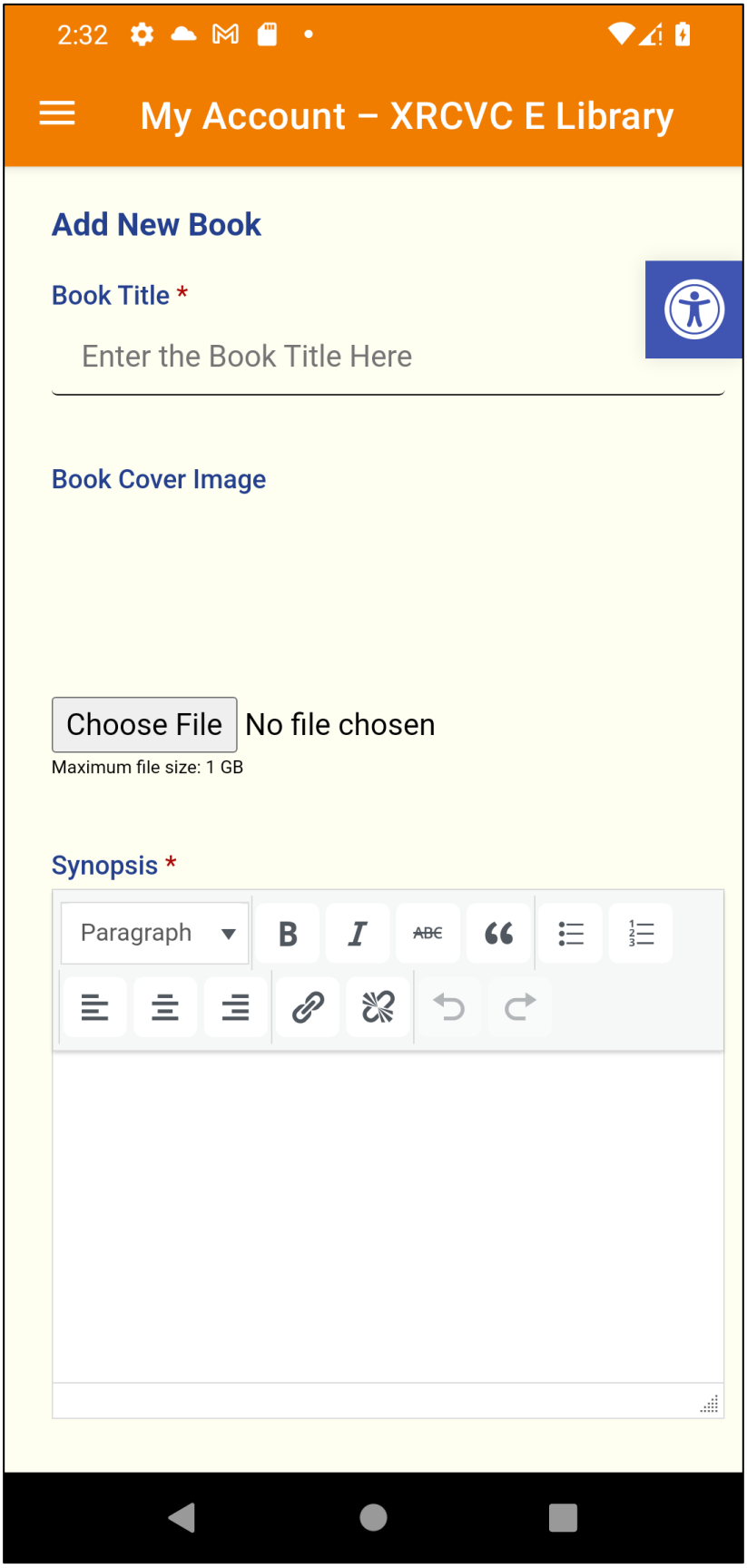
### 8.4.11 Account Profile



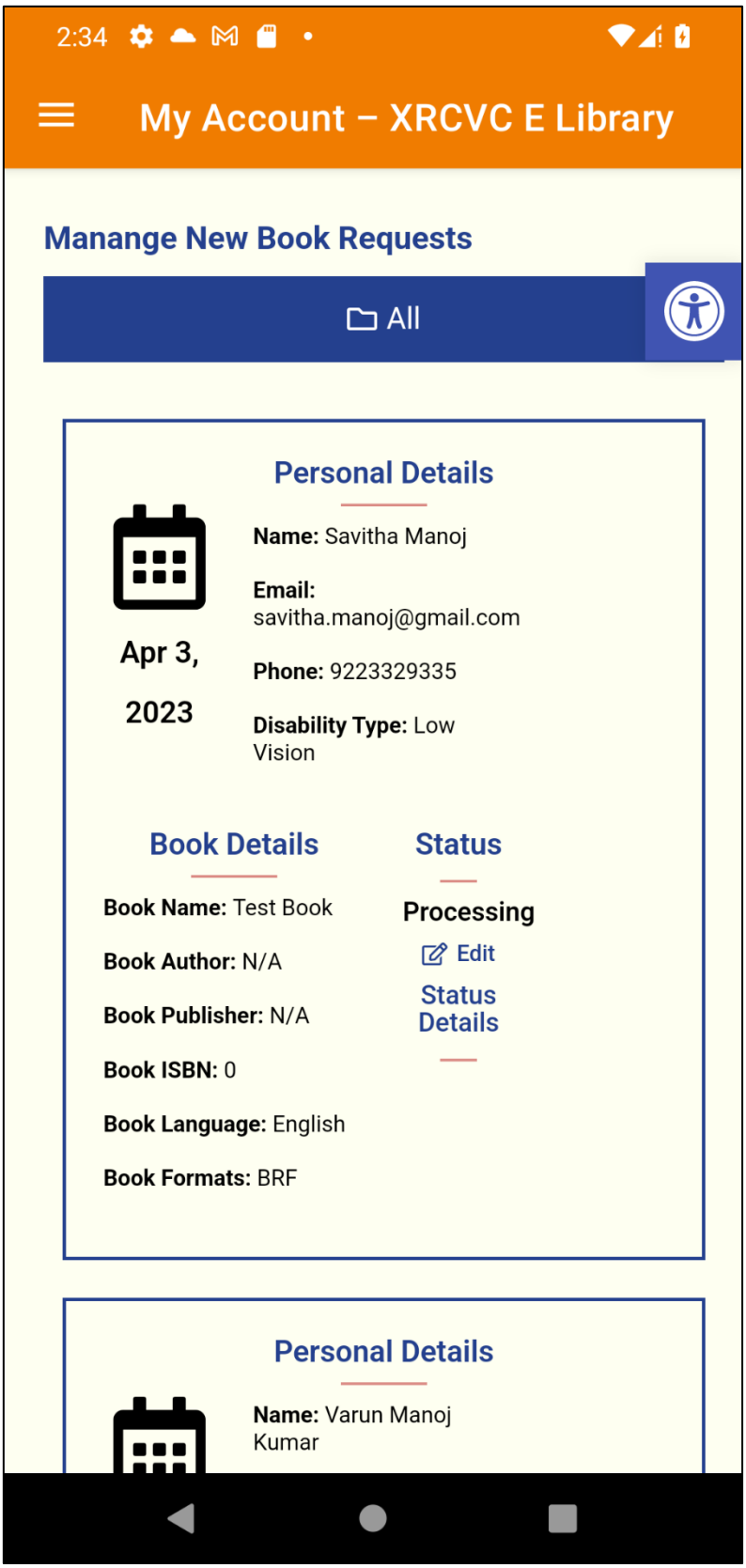

Once Logged in, the admin can view their profile. The admin can also update their profile photo, change the First and Last Name, update their phone Number, or even change the password.

8.4.12 View Book Requests

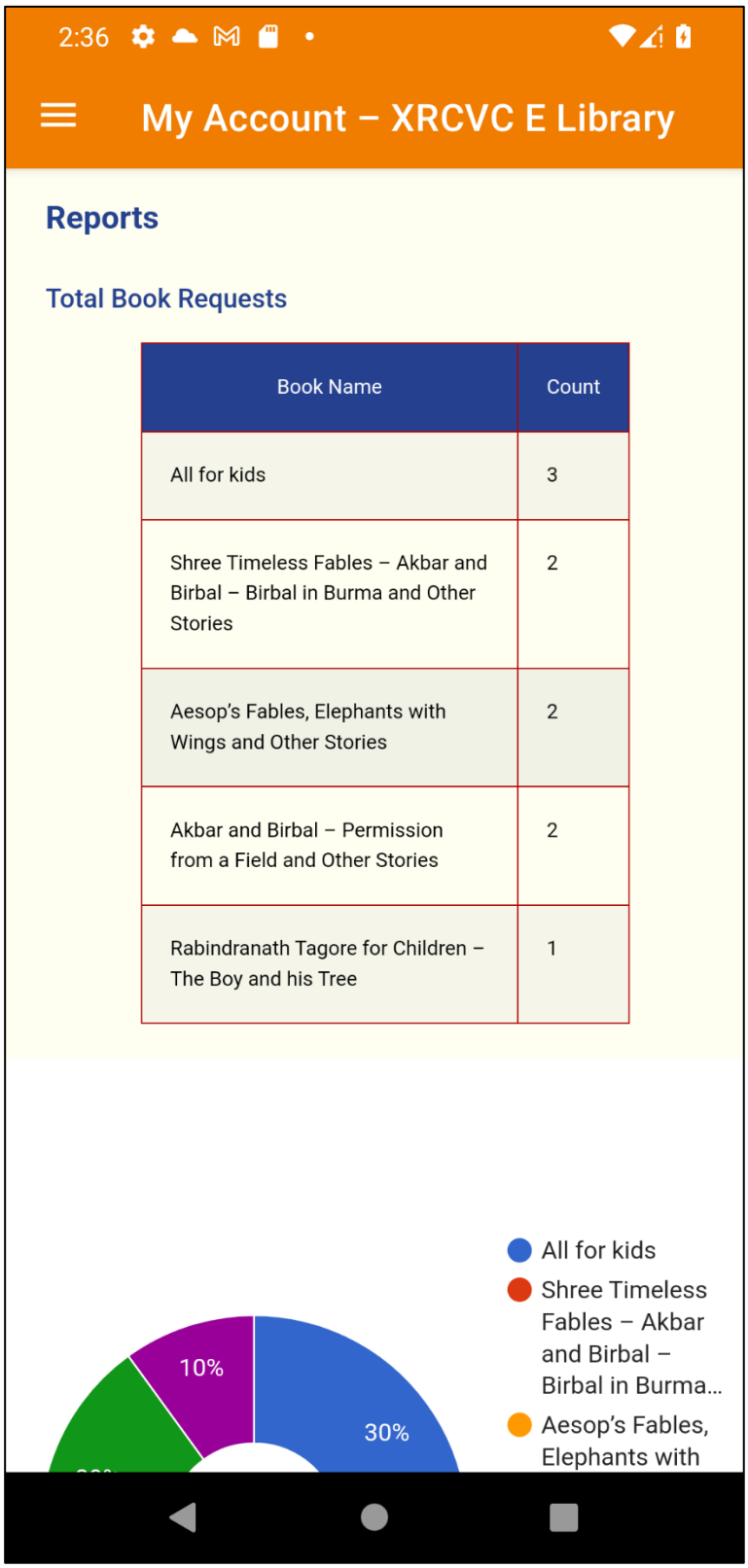
8.4.13 Add New Book



8.4.14 Manage New Book Requests

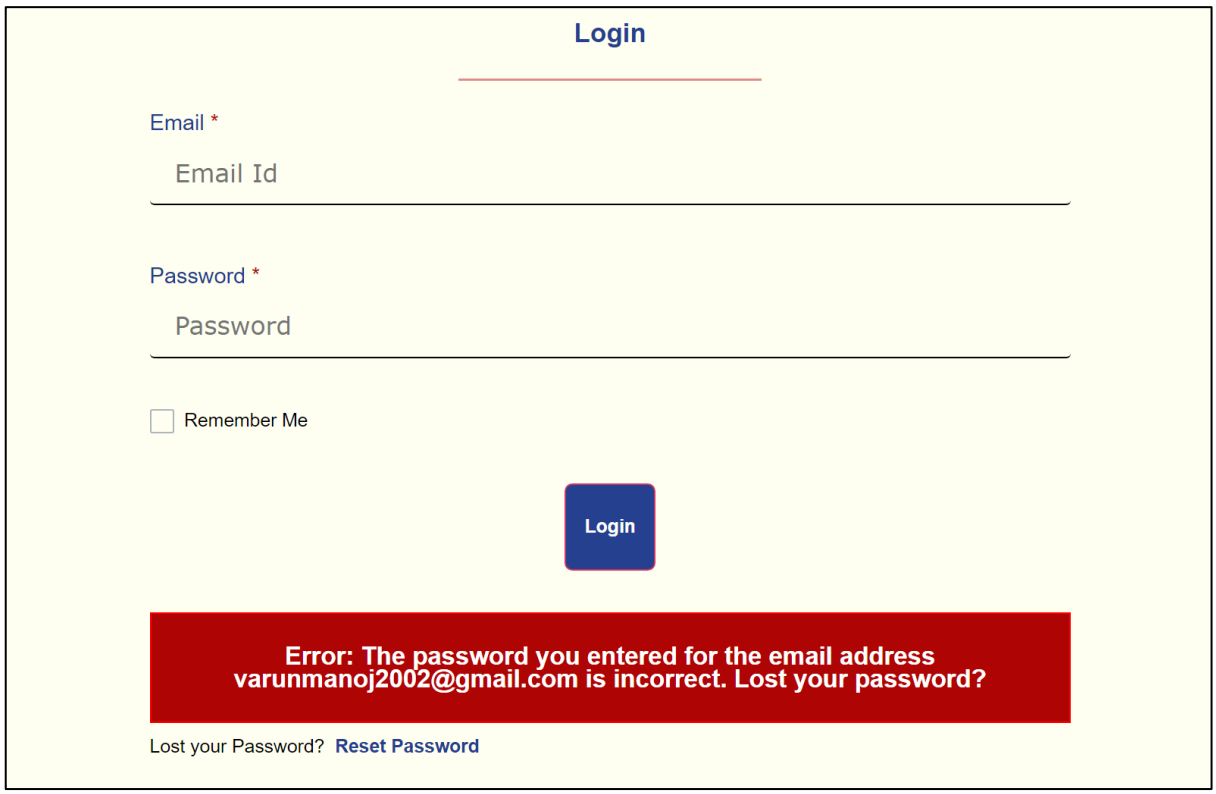


8.4.15 View Reports

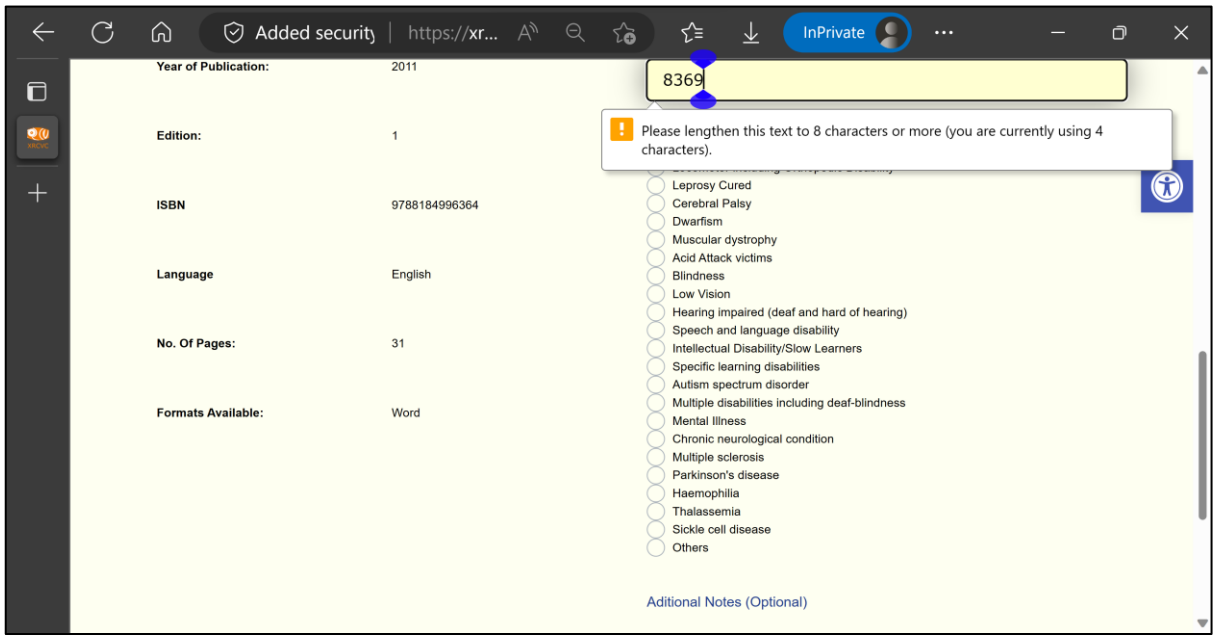


## 8.5 Validation

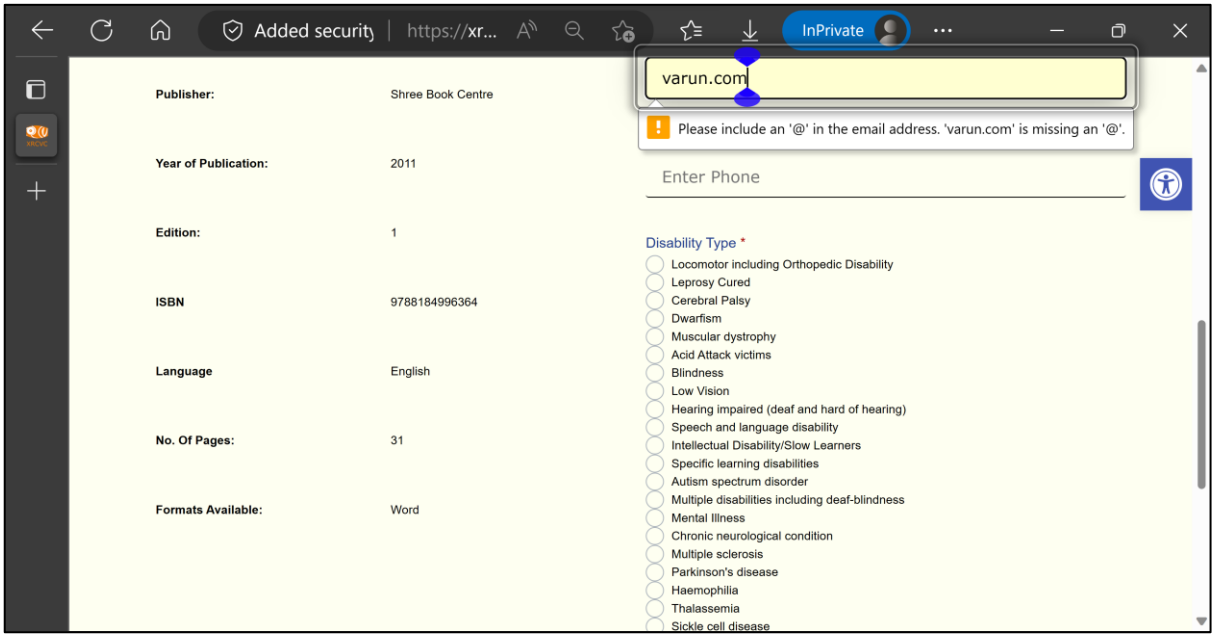
### 8.5.1 Invalid Login page




### 8.5.2 Invalid phone Filed in Book Request Form



### 8.5.3 Invalid Email Id in Book Request Form



# 9 Reports

The admin can view various reports from the dashboard regarding the book requests and new book requests. The admin can view the reports both in a table view and a graphical view.

## 9.1 Total Book Requests

### 9.1.1 Table View

*Table 9-1 Table showing Total Number of Books Requested*

Total Book Requests

| Book Name | Count |
|---|---|
| All for kids | 3 |
| Shree Timeless Fables – Akbar and Birbal – Birbal in Burma and Other Stories | 2 |
| Aesop's Fables, Elephants with Wings and Other Stories | 2 |
| Akbar and Birbal – Permission from a Field and Other Stories | 2 |
| Rabindranath Tagore for Children – The Boy and his Tree | 1 |

### 9.1.2 Graphical View

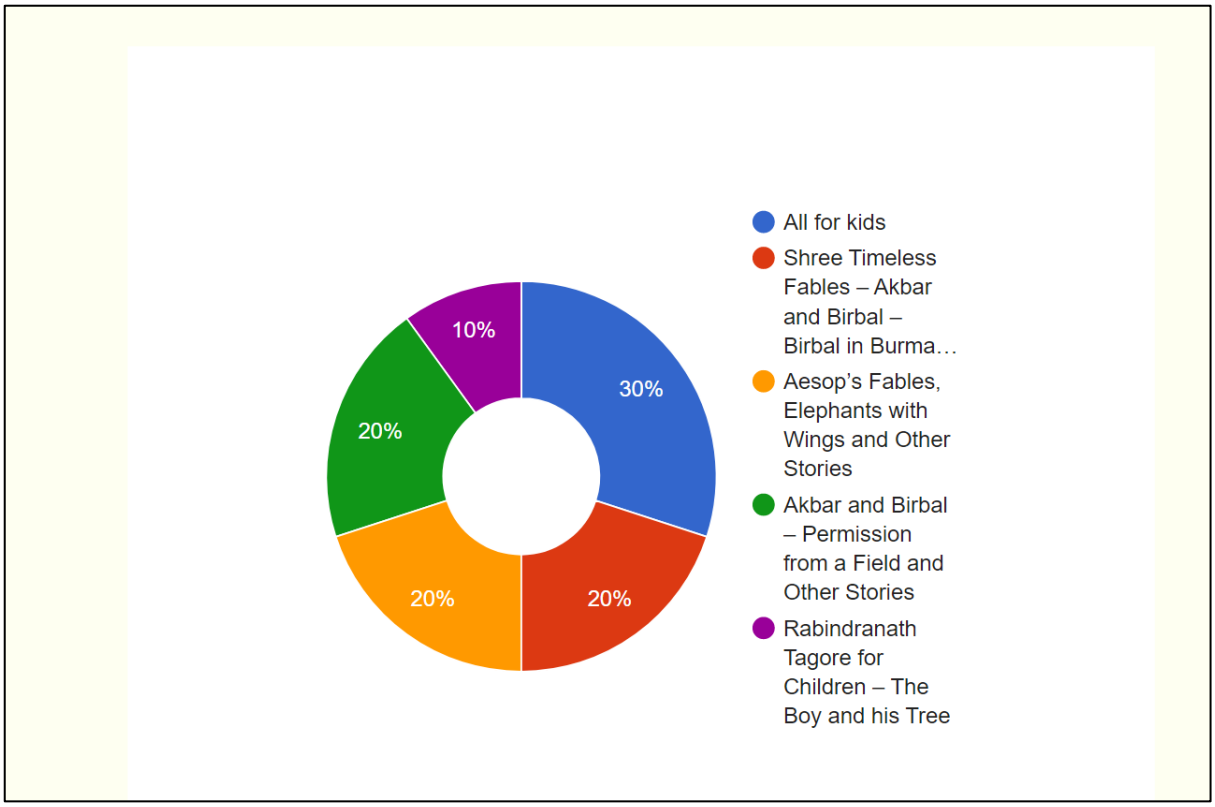



*Figure 9-1 Pie Chart showing the distribution of various books requested.*

## 9.2 Book Requests per Student

### 9.2.1 Table View

*Table 9-2 Table show Number of Books Requested by Each Student*

Book Requests per Student

| Full Name | Number of Requests |
|---|---|
| Varun Manoj Kumar | 4 |
| Savitha Manoj | 4 |
| Varun Manoj | 1 |
| Varun | 1 |

### 9.2.2 Graphical View

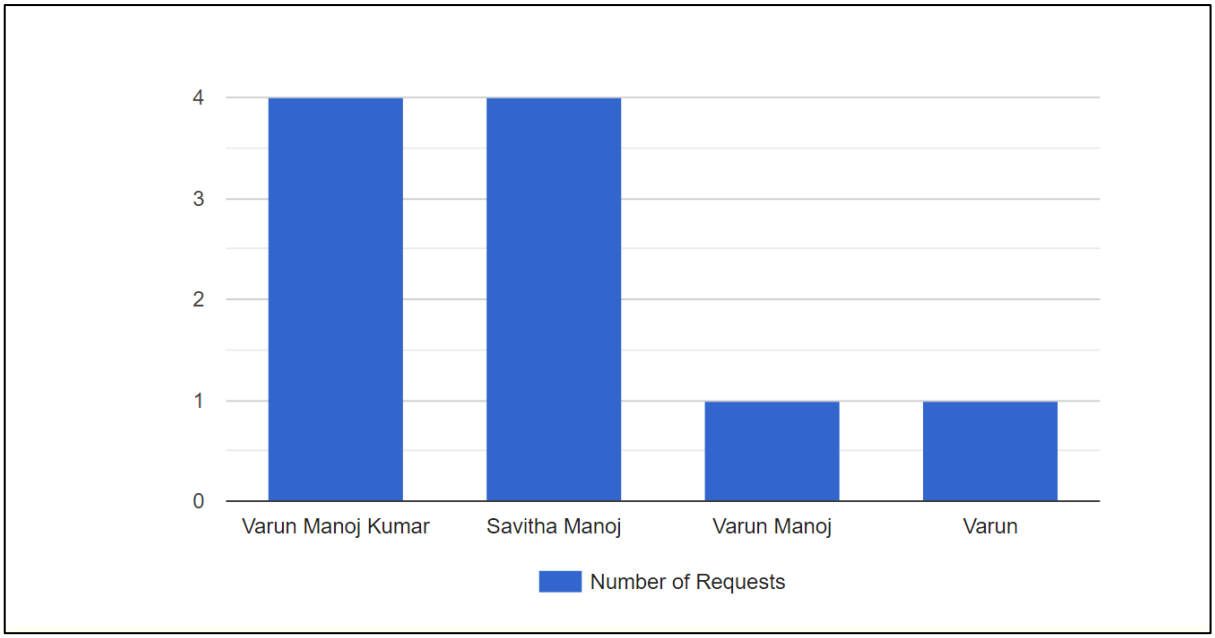



*Figure 9-2 Bar Graph Showing the number of Books requested by Each Student*

## 9.3 Book Request Status

### 9.3.1 Table View

*Table 9-3Showes book requests grouped by request status*

9.3 Book Request Status

| Request Status | Number of Requests |
|---|---|
| Processing | 2 |
| Pending | 1 |
| Cancelled | 1 |
| On Hold | 1 |

### 9.3.2 Graphical View

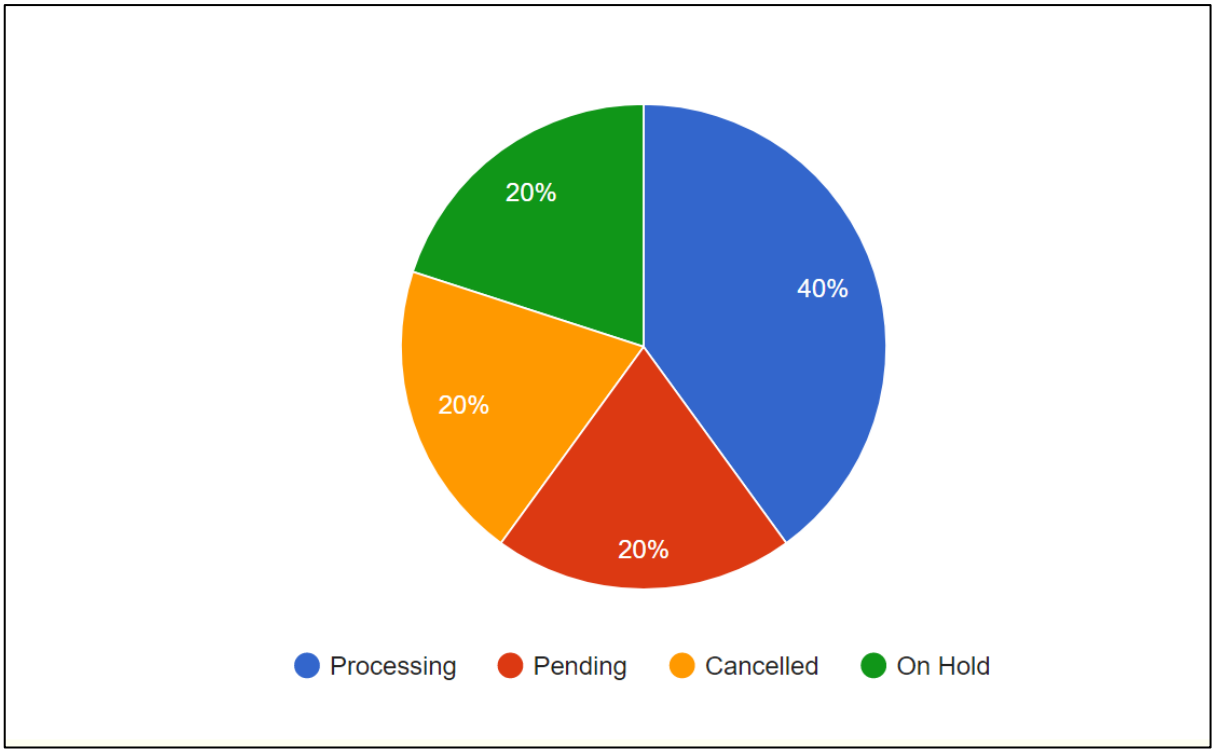



*Figure 9-3 Pie Chart showing the Book Request Status*

# 10 System Implementation

## 10.1 Source Code

### 10.1.1 jQuery Code for Accessibility

#### *10.1.1.1 Make Active Filters Accessible*

```
/* Make the Active Filter Section Accessible
Make the Active Filters keyboard focusable
Replace the Times symbol by default with a X symbol in each active filter */

jQuery( document ).ready( function ( $ )
{
    // Run the function only after 300 milliseconds
    setInterval( () =>
    {
        // Check wheather the div element is present on the page or not.
        if ( '.jet - active - filter' )
        {
            // Set the div to receave keyboard focus by using Keyboard Focus
            $( '.jet-active-filter' ).attr( 'tabindex', 0 );
            // Set the rol of the div to a button so that it can be pressed by the enter key.
            $( '.jet-active-filter' ).attr( 'role', 'button' );
        }
        // Replace the Default 'times' symbol with a X symbol which indicates the Close button.
        $( '.jet-active-filter__remove' ).text( 'X' );

    }, 300 );

} );
```

#### *10.1.1.2 Make Checkboxes accessible with screen readers in My Account- Add new Book*

```
/* Add semantic relationship between Checkboxes labels and the Group
Add IDS to all label tags and link each checkbox with the group label
Add aria - describe attribute to announce both the label of the checkbox and the group label */

jQuery( document ).ready( function ( $ )
{
    setTimeout( () =>
    {

        // Add ID to the label element so as to link the checkboxes with the label
        $( "label.jet-form-builder__label-text:contains('Book Category')" ).attr( "id", "Book-Category" );

        $( "label.jet-form-builder__label-text:contains('Book Formats')" ).attr( "id", "Book-Formats" );

        $( "label.jet-form-builder__label-text:contains('Book Keywords')" ).attr( "id", "Book-Keywords" );

        // Link the Checkboxes with the Label

        $( "input[name='book-category[]']" ).attr( 'aria-describedby', 'Book-Category' );

        $( "input[name='book_formats[]']" ).attr( 'aria-describedby', 'Book-Formats' );

        $( "input[name='book_keywords[]']" ).attr( 'aria-describedby', 'Book-Keywords' );

    }, 300 );

} );
```

#### *10.1.1.3 Make Search Box 's Search Button Text accessible.*

```
/* Make Search Box''s Search Button Text accessible
Change the Screen reader text to Search book
Also Change the Tooltip Text to Search book
 */
```

```
jQuery( document ).ready( function ( $ )
{
    setInterval( () =>
    {
        // Change the Search Box Button text to Search book for Screen Readers
        $( '.jet-ajax-search__submit' ).attr( 'aria-label', 'Search Book' );
        // XChange the Tooltip text of the search button to Search Books
        $( '.jet-ajax-search__submit' ).attr( 'title', 'Search Book' );
    }, 300 );
} );
```

### 10.1.2 CSS Code for Accessibility

#### *10.1.2.1 Change Active Filter Colour on focus*

```
/* Change the Background color , text colour and close button colour on focus along with the default hover
 */

/* Change the Background Colour of active filters to yell on receiving focus  */
.jet-active-filter:focus {
        background-color: #FFFFCC !important;
}

/* Change the Text colour to black on focus */
.jet-active-filter:focus > .jet-active-filter__label,
.jet-active-filter:focus > .jet-active-filter__val {
        color:#000 !important;
                background-color: #FFFFCC !important;

}


/* Change the Remove button 's colour to red on focus */
.jet-active-filter:focus > .jet-active-filter__remove {
        color:#AF0404 !important;
}
```

#### *10.1.2.2 Replace Fake Checkboxes with HTML 5 native Checkboxes*

```
/* Show Native Checkboxes and hide fake div's that act like Checkboxes in the filters'
sidebar
 */

/* Increase the size of the Checkboxes to 20px and set the visibility to true */
input.jet-checkboxes-list__input {
    width: 20px !important;
    height: 20px !important;
    visibility: visible !important;
    display:block !important;
}

.jet-checkboxes-list__decorator{
    display: none;
}

/* Increase Space between Each Checkbox to increase Touch target Size */
.jet-checkboxes-list__label{
    padding-left:2px;
}
```

#### *10.1.2.3 Make the Links in Headings Easier to see for low vision*

```
/* Add a visual indicator to Links in Headings whey they receive focus or when the mouse hovers over them */


h2.elementor-heading-title a:hover{
    /* Add an underline effect on hover */
    text-decoration: underline;
    /* Change the text colour to red on hover */
    color: #b4332d !important;
    /* Add a dotted Border on hover */
    border-style: dotted;

}
h2.elementor-heading-title a:focus{
    /* Add an underline effect on focus */
    text-decoration: underline;
    /* Change text Colour on focus */
    color: #b4332d !important;
    /* Add a Dotted Boerder on focus  */
    border-style: dotted;

}
```

#### *10.1.2.4 Make Search Box's Search Button Change Colour on Focus*

```
/* Make the Search Box 's Search Button change Background and Text colour on Focus*/

.jet-ajax-search__submit:focus {
    color: var(--e-global-color-text ) !important;
    background-color: var(--e-global-color-0483021 ) !important;
}
```

#### *10.1.2.5 Make the Search Results Container Change colour on focus*

```
/* Make the Search Results Container change background colour and text colour on focus  */

.jet-ajax-search__item-link:focus{
    background-color: #DEC8C8;
}

/* Change the Text colour of the custom fields in the search results to black  */ */
.jet-search-content-fields__item-value{
    color: black;
}
```

### 10.1.3 Jet Engine Dynamic Table's Data Table Module

```
<link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/v/dt/dt-1.11.2/datatables.min.css" />
<script type="text/javascript" src="https://cdn.datatables.net/v/dt/dt-1.11.2/datatables.min.js"></script>
```

```
<script>
    jQuery( document ).ready( function ( $ )
    {
        $( '.jet-dynamic-table' ).DataTable();
    } );
</script>
```

### 10.1.4 Jet Engine Dynamic table's Data Table Module -Add Export Functionality

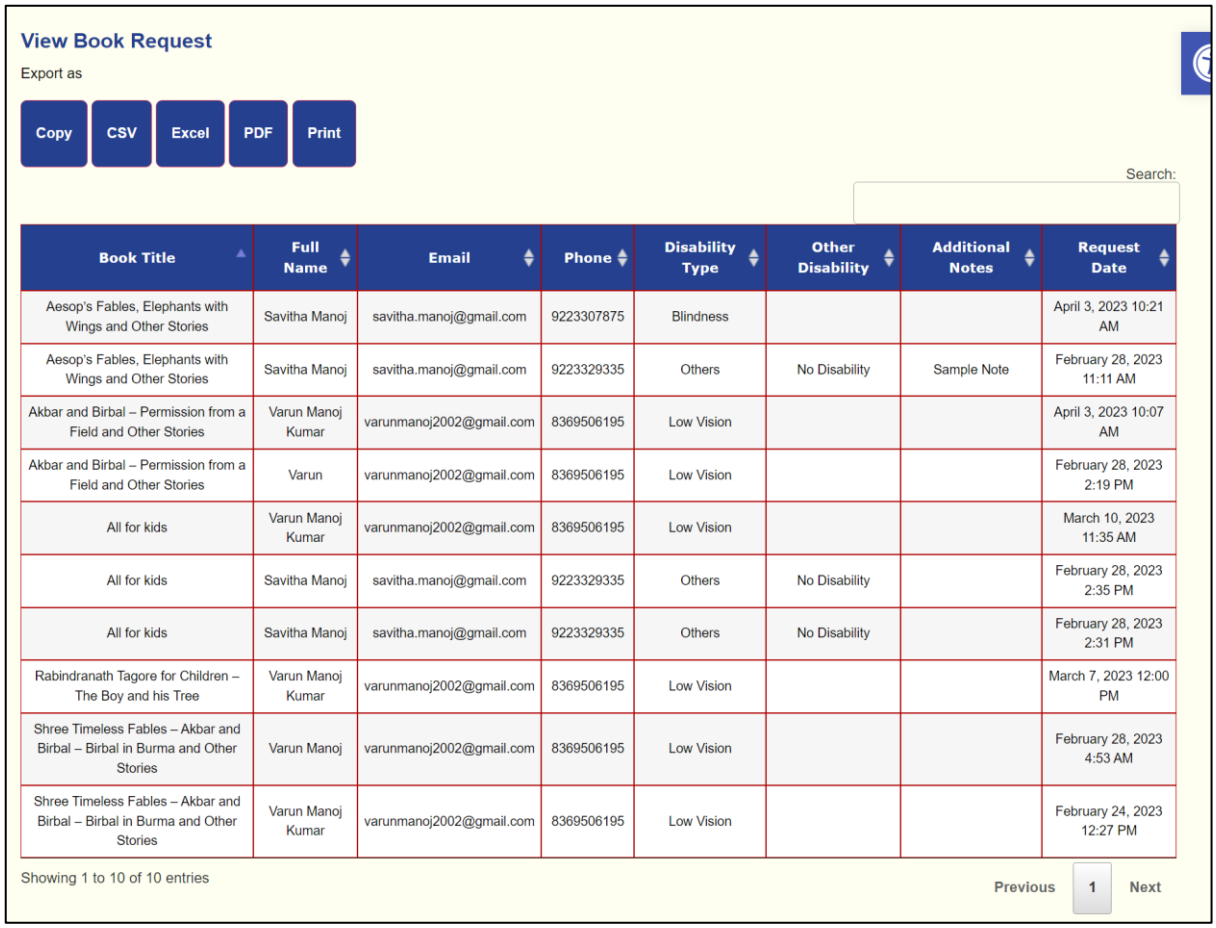
```
<link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/v/dt/dt-1.11.2/datatables.min.css" />

<script type="text/javascript" src="https://code.jquery.com/jquery-3.5.1.js"></script>
<script type="text/javascript" src="https://cdn.datatables.net/1.12.1/js/jquery.dataTables.min.js"></script>
<script type="text/javascript" src="https://cdn.datatables.net/buttons/2.2.3/js/dataTables.buttons.min.js"></script>
<script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/jszip/3.1.3/jszip.min.js"></script>
<script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.53/pdfmake.min.js"></script>
<script type="text/javascript" src="https://cdnjs.cloudflare.com/ajax/libs/pdfmake/0.1.53/vfs_fonts.js"></script>
<script type="text/javascript" src="https://cdn.datatables.net/buttons/2.2.3/js/buttons.html5.min.js"></script>
<script type="text/javascript" src="https://cdn.datatables.net/buttons/2.2.3/js/buttons.print.min.js"></script>

<script>
    jQuery( document ).ready( function ( $ )
    {
        $( '.jet-dynamic-table' ).DataTable( {
            dom: 'Bfrtip',
            buttons: [
                'copy', 'csv', 'excel', 'pdf', 'print'
            ]
        } );
    } );
</script>
```

The above code has been used to convert a Jet engine Dynamic Table in to a Data Table with the option to export the Data from the Data Table to Excel, CSV, PDF or Copy the data to the clipboard or Print the Data.

This data has been added to the '**View Book Request Page**' in the My Account Section.



View Book Request

Export as

Copy CSV Excel PDF Print

Search:

| Book Title | Full Name | Email | Phone | Disability Type | Other Disability | Additional Notes | Request Date |
|---|---|---|---|---|---|---|---|
| Aesop's Fables, Elephants with Wings and Other Stories | Savitha Manoj | savitha.manoj@gmail.com | 9223307875 | Blindness | | | April 3, 2023 10:21 AM |
| Aesop's Fables, Elephants with Wings and Other Stories | Savitha Manoj | savitha.manoj@gmail.com | 9223329335 | Others | No Disability | Sample Note | February 28, 2023 11:11 AM |
| Akbar and Birbal – Permission from a Field and Other Stories | Varun Manoj Kumar | varunmanoj2002@gmail.com | 8369506195 | Low Vision | | | April 3, 2023 10:07 AM |
| Akbar and Birbal – Permission from a Field and Other Stories | Varun | varunmanoj2002@gmail.com | 8369506195 | Low Vision | | | February 28, 2023 2:19 PM |
| All for kids | Varun Manoj Kumar | varunmanoj2002@gmail.com | 8369506195 | Low Vision | | | March 10, 2023 11:35 AM |
| All for kids | Savitha Manoj | savitha.manoj@gmail.com | 9223329335 | Others | No Disability | | February 28, 2023 2:35 PM |
| All for kids | Savitha Manoj | savitha.manoj@gmail.com | 9223329335 | Others | No Disability | | February 28, 2023 2:31 PM |
| Rabindranath Tagore for Children – The Boy and his Tree | Varun Manoj Kumar | varunmanoj2002@gmail.com | 8369506195 | Low Vision | | | March 7, 2023 12:00 PM |
| Shree Timeless Fables – Akbar and Birbal – Birbal in Burma and Other Stories | Varun Manoj | varunmanoj2002@gmail.com | 8369506195 | Low Vision | | | February 28, 2023 4:53 AM |
| Shree Timeless Fables – Akbar and Birbal – Birbal in Burma and Other Stories | Varun Manoj Kumar | varunmanoj2002@gmail.com | 8369506195 | Low Vision | | | February 24, 2023 12:27 PM |

Showing 1 to 10 of 10 entries

Previous 1 Next

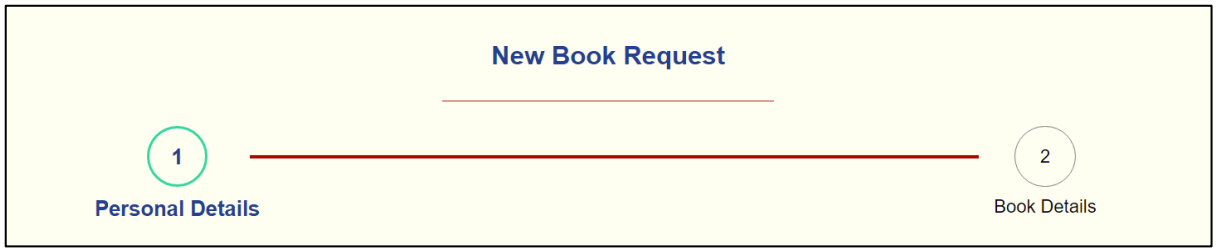
### 10.1.5 Change the Step Title in the Multi Step Form on the New Book Request Page

```
// Change the Step title in the Multi Step form on the New Book Request pae  using jQuery

jQuery( document ).ready( function ( $ )
{
    setInterval( () =>
    {
        // the eq(number ) argument is used to uniqualy identity each div and change the text in both div's
        // Use the text() method to change the text inside the div.
        $( '.jet-form-builder-progress-pages__item--label:eq(0)' ).text( "Personal Details" );

        $( '.jet-form-builder-progress-pages__item--label:eq(1)' ).text( "Book Details" );
    }, 300 );
} );
```

The above code has been used to change the title of the Multi Step form on the new Book Request Page.

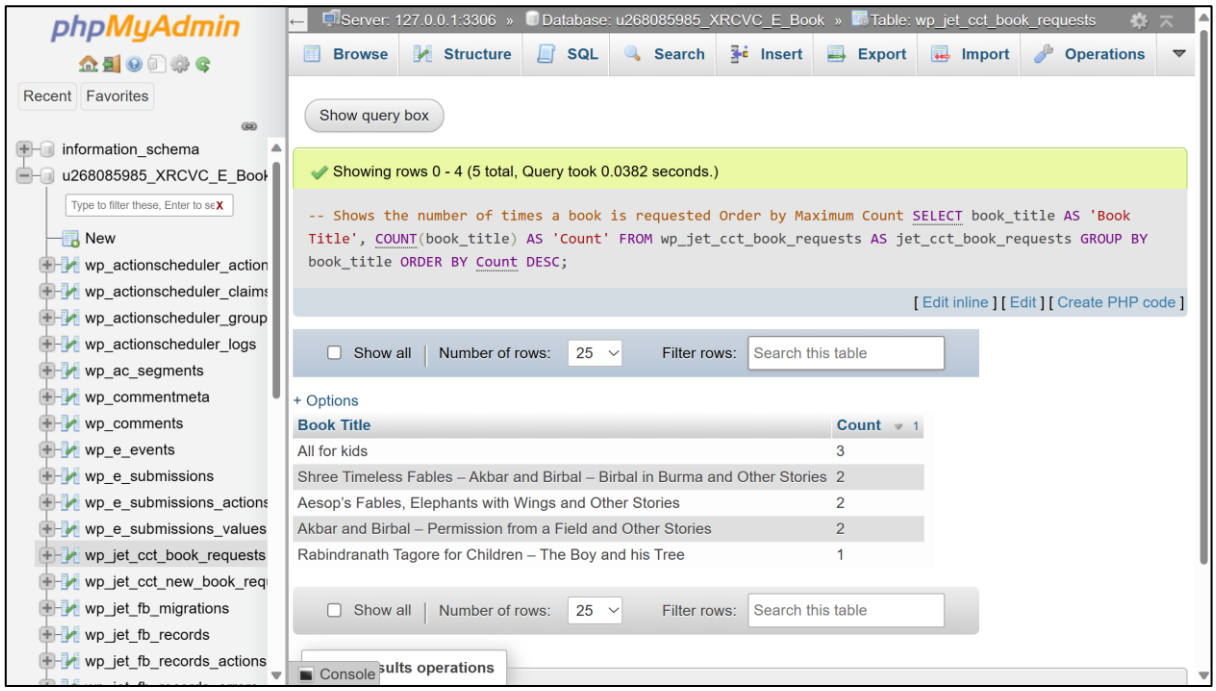



### 10.1.6 SQL Queries for Reports

#### *10.1.6.1 Total Book Requests*

```
-- Shows the number of times a book is requested  Order by Maximum Count
SELECT book_title AS 'Book Title', COUNT(book_title) AS 'Count' FROM
wp_jet_cct_book_requests AS jet_cct_book_requests GROUP BY book_title ORDER BY
Count DESC;
```

##### 10.1.6.1.1 Result in PHP MyAdmin



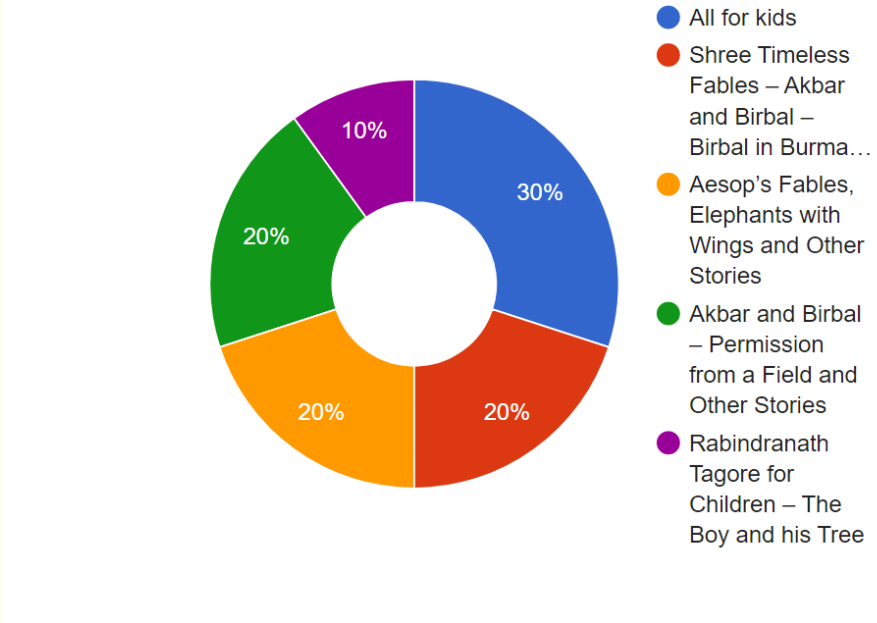
*10.1.6.2 Table View*

**Reports**

Total Book Requests

| Book Name | Count |
|---|---|
| All for kids | 3 |
| Shree Timeless Fables – Akbar and Birbal – Birbal in Burma and Other Stories | 2 |
| Aesop's Fables, Elephants with Wings and Other Stories | 2 |
| Akbar and Birbal – Permission from a Field and Other Stories | 2 |
| Rabindranath Tagore for Children – The Boy and his Tree | 1 |

*10.1.6.3 Graphical view*

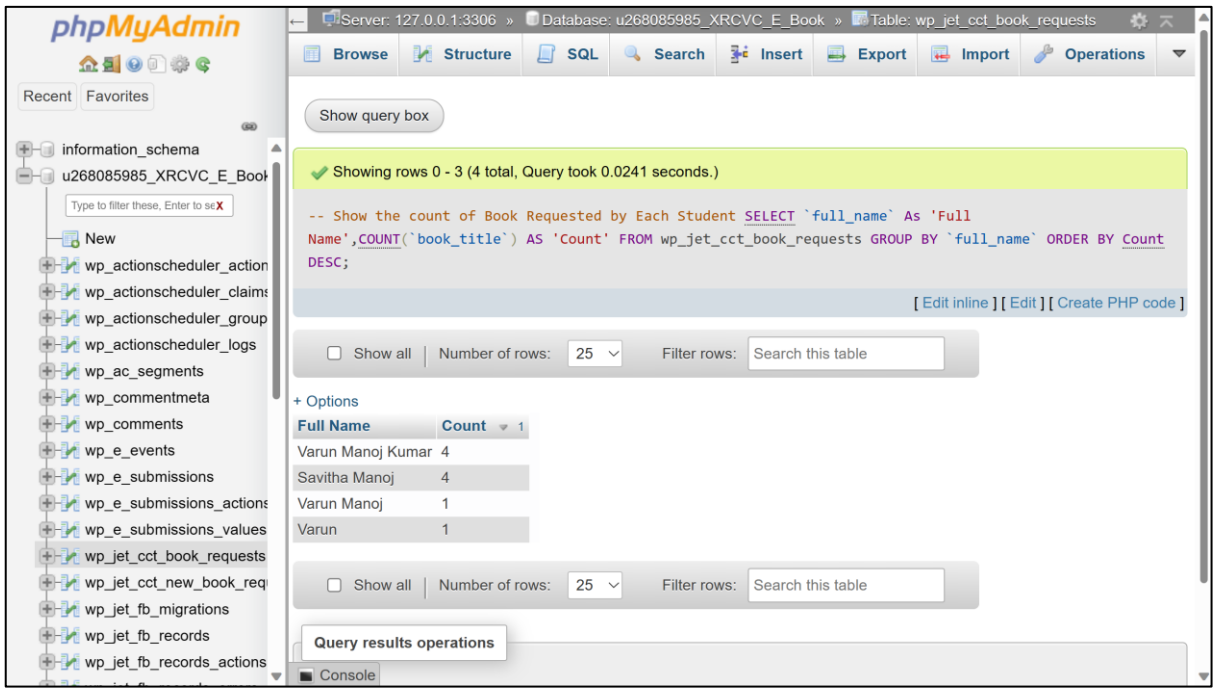



## 10.1.7 Book Requests per Student

```
-- Show the count of Book Requested by Each Student
```

```
SELECT `full_name` As 'Full Name',COUNT(`book_title`) AS 'Count' FROM
wp_jet_cct_book_requests GROUP BY `full_name` ORDER BY Count DESC;
```

*10.1.7.1 Result in PHP MyAdmin*

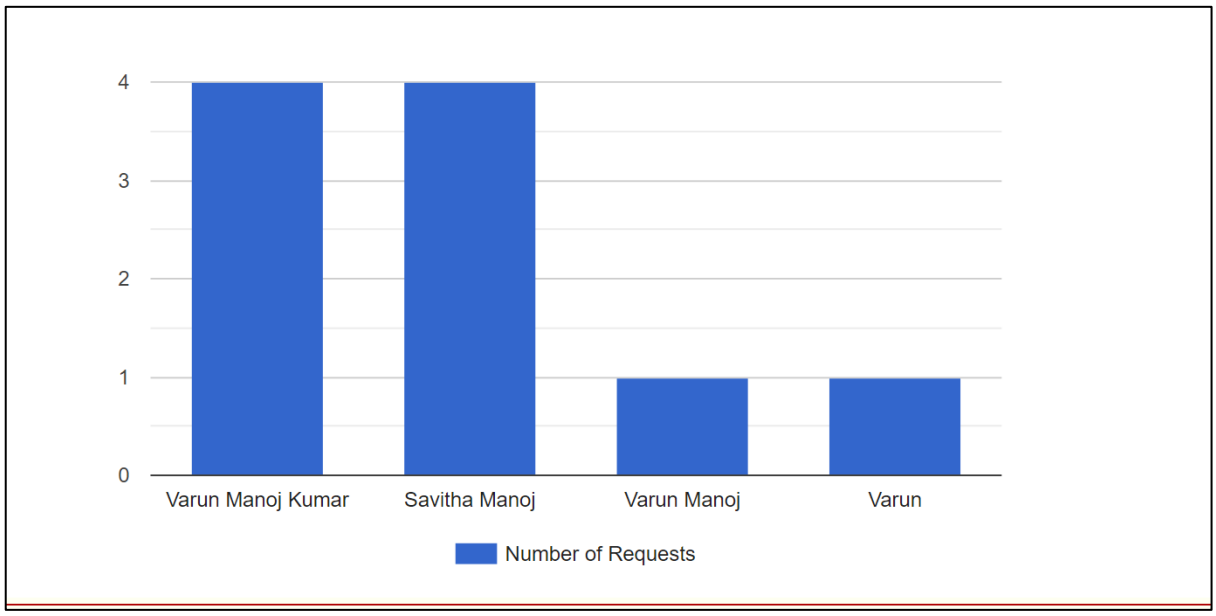



*10.1.7.2 Table View*

Book Requests per Student

| Full Name | Number of Requests |
|---|---|
| Varun Manoj Kumar | 4 |
| Savitha Manoj | 4 |
| Varun Manoj | 1 |
| Varun | 1 |

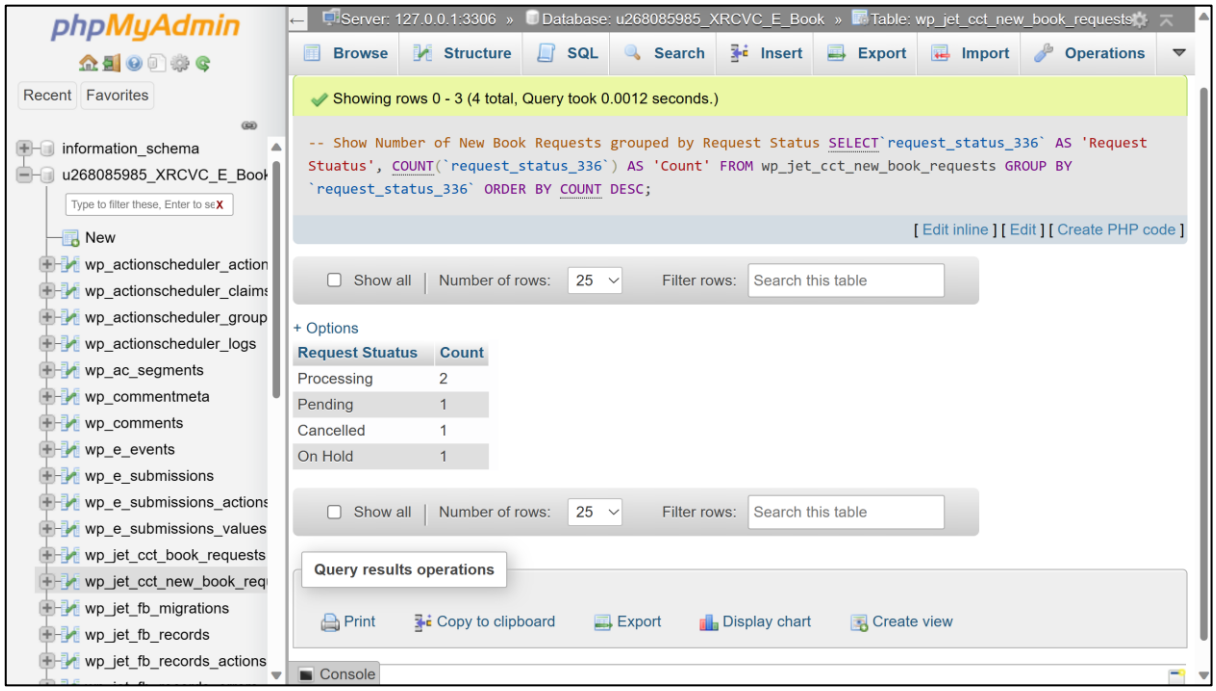
*10.1.7.3 Graphical View*




## 10.1.8 Book Request Status

```
-- Show Number of New Book Requests grouped by Request Status
SELECT`request_status_336` AS 'Request Stuatus', COUNT(`request_status_336`)
AS 'Count' FROM wp_jet_cct_new_book_requests GROUP BY `request_status_336`
ORDER BY COUNT DESC;
```

*10.1.8.1 Result in PHP MYAdmin*

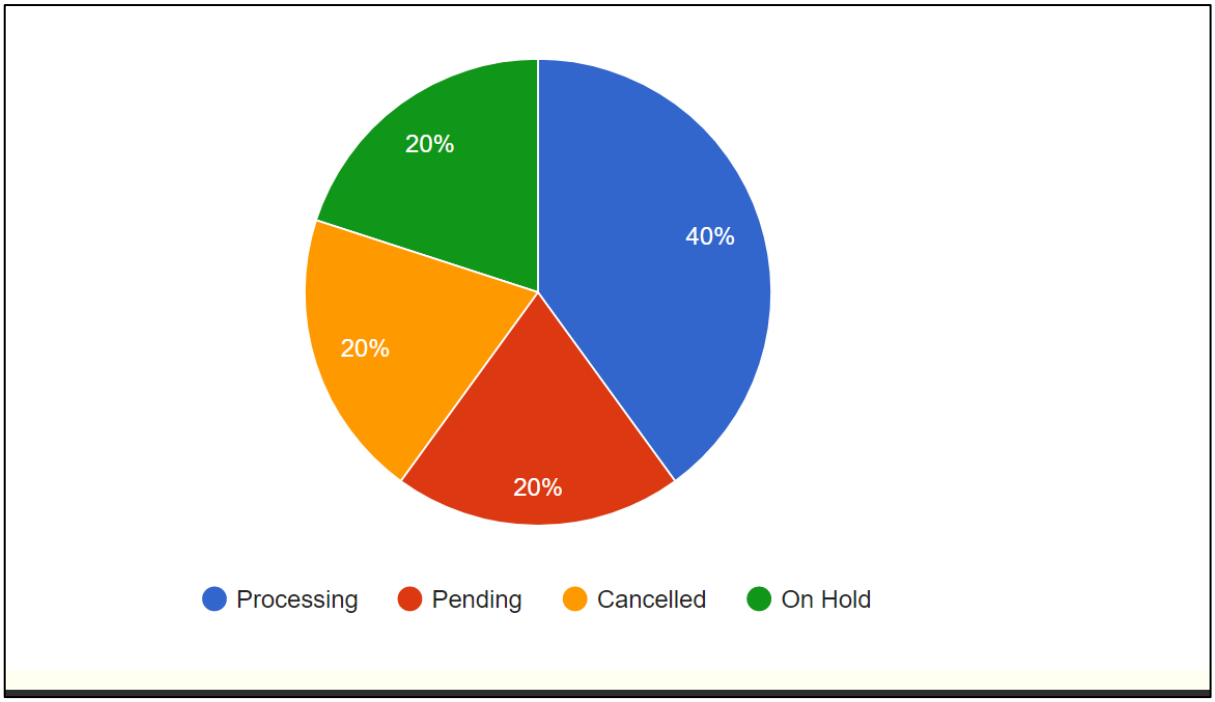


### *10.1.8.2 Table View*

Book Request Status

| Request Status | Number of Requests |
|---|---|
| Processing | 2 |
| Pending | 1 |
| Cancelled | 1 |
| On Hold | 1 |

### *10.1.8.3 Graphical View*



### 10.1.9 Android Application (Android Studio and Java)

#### *10.1.9.1 Layouts*

##### 10.1.9.1.1 Activity Main

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/draw_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:fitsSystemWindows="true"
    tools:context=".MainActivity" >

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <WebView
            android:id="@+id/webView"
            android:layout_width="0dp"
            android:layout_height="0dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent">

        </WebView>

        <ProgressBar
            android:id="@+id/progressBar"
            style="?android:attr/progressBarStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:max="100"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.5"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_bias="0.5" />

    </androidx.constraintlayout.widget.ConstraintLayout>

    <com.google.android.material.navigation.NavigationView
        android:id="@+id/navigation_view"
```

```
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:fitsSystemWindows="false"
        app:headerLayout="@layout/header_menu"
        app:menu="@menu/navigation_menu" />
</androidx.drawerlayout.widget.DrawerLayout>
```

10.1.9.1.2 No internet

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/draw_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/page_background_colour"
    android:fitsSystemWindows="true"

    tools:context=".NoInternent">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="177dp"
            android:layout_height="0dp"
            android:layout_marginTop="110dp"
            android:layout_marginBottom="42dp"
            app:layout_constraintBottom_toTopOf="@+id/nointernet"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:srcCompat="@drawable/wifi_off" />

        <TextView
            android:id="@+id/nointernet"
            android:layout_width="371dp"
            android:layout_height="0dp"
            android:layout_marginStart="18dp"
            android:layout_marginTop="285dp"
            android:layout_marginEnd="18dp"
            android:layout_marginBottom="9dp"
            android:selectAllOnFocus="true"
            android:text="@string/no_internet"
            android:textAlignment="center"
            android:textColor="@color/text"
            android:textIsSelectable="true"
            android:textSize="30dp"
            android:textStyle="bold"
            app:layout_constraintBottom_toTopOf="@+id/gohome"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
```

```
            app:layout_constraintTop_toTopOf="parent" />

        <Button
            android:id="@+id/gohome"
            android:layout_width="287dp"
            android:layout_height="82dp"
            android:layout_marginStart="38dp"
            android:layout_marginBottom="237dp"
            android:text="@string/goHomeBTN"
            app:icon="@drawable/home"
            app:iconPadding="10dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="@+id/nointernet"
            app:layout_constraintTop_toBottomOf="@+id/nointernet" />

    </androidx.constraintlayout.widget.ConstraintLayout>

    <com.google.android.material.navigation.NavigationView
        android:id="@+id/navigation_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:fitsSystemWindows="false"
        app:headerLayout="@layout/header_menu"
        app:menu="@menu/navigation_menu" />
</androidx.drawerlayout.widget.DrawerLayout>
```

10.1.9.1.3 Header Menu (navigation Header)

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:background="@color/secondary">

    <ImageView
        android:id="@+id/logo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:contentDescription="@string/xrcvc_and_st_logo"
        android:paddingStart="10dp"
        app:srcCompat="@drawable/xrcvc_and_st__logo" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:ellipsize="end"
        android:paddingLeft="10dp"
        android:text="@string/app_name"
        android:textColor="@color/text"
        android:textSize="20dp"
        android:textStyle="bold" />
</LinearLayout>
```

10.1.9.1.4 Error page

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.drawerlayout.widget.DrawerLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/draw_layout"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/page_background_colour"
    android:fitsSystemWindows="true"

    tools:context=".ErrorPage">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="177dp"
            android:layout_height="0dp"
            android:layout_marginTop="110dp"
            android:layout_marginBottom="42dp"
            app:layout_constraintBottom_toTopOf="@+id/nointernet"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:srcCompat="@drawable/info" />

        <TextView
            android:id="@+id/nointernet"
            android:layout_width="371dp"
            android:layout_height="0dp"
            android:layout_marginStart="18dp"
            android:layout_marginTop="285dp"
            android:layout_marginEnd="18dp"
            android:layout_marginBottom="9dp"
            android:selectAllOnFocus="true"
            android:text="@string/invalidURL"
            android:textAlignment="center"
            android:textColor="@color/text"
            android:textIsSelectable="true"
            android:textSize="30dp"
            android:textStyle="bold"
            app:layout_constraintBottom_toTopOf="@+id/gohome"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
```

```
            app:layout_constraintTop_toTopOf="parent" />

        <Button
            android:id="@+id/gohome"
            android:layout_width="287dp"
            android:layout_height="82dp"
            android:layout_marginStart="38dp"
            android:layout_marginBottom="237dp"
            android:text="@string/goHomeBTN"
            app:icon="@drawable/home"
            app:iconPadding="10dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintStart_toStartOf="@+id/nointernet"
            app:layout_constraintTop_toBottomOf="@+id/nointernet" />

    </androidx.constraintlayout.widget.ConstraintLayout>

    <com.google.android.material.navigation.NavigationView
        android:id="@+id/navigation_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="start"
        android:fitsSystemWindows="false"
        app:headerLayout="@layout/header_menu"
        app:menu="@menu/navigation_menu" />
</androidx.drawerlayout.widget.DrawerLayout>
```

*10.1.9.2 Java (Program logic)*

10.1.9.2.1 Main Activity

```
package com.xrcvc.e_library;

import android.content.Context;
import android.content.Intent;
import android.graphics.Bitmap;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.webkit.WebChromeClient;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.ProgressBar;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.FirebaseApp;
import com.google.firebase.analytics.FirebaseAnalytics;
import com.google.firebase.appcheck.FirebaseAppCheck;
import
com.google.firebase.appcheck.playintegrity.PlayIntegrityAppCheckProviderFactor
y;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.Objects;

public class MainActivity extends AppCompatActivity {
    //    Navigation Menu Items
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle toggle;

    //    Mai Web Page
    WebView webView;
```

```
    WebSettings webSettings;
    ProgressBar progressBar;

    //    Firebase
    FirebaseAnalytics firebaseAnalytics;
    DatabaseReference mDatabase;


    public void FetchURL() {
        // Retrieve the URL from the Firebase Realtime Database
        mDatabase.child("HomeURL").addValueEventListener(new
ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot dataSnapshot) {
                String url = dataSnapshot.getValue(String.class);
                if (url != null) {
                    // Load the URL into the WebView
                    webpageloadcontent(url);
                }
            }
            @Override
            public void onCancelled(@NonNull DatabaseError databaseError) {
                Log.w("No Data", "onCancelled", databaseError.toException());
            }

        });
    }

            private void webpageloadcontent(String url) {

//                Load webpage in the app and not in an external web browser
                webView.setWebViewClient(new WebViewClient() {
                    //            Show Error Page in case of any error
                    @Override
                    public void onReceivedError(WebView view, int errorCode,
String description, String failingUrl) {
                        startActivity(new Intent(getApplicationContext(),
ErrorPage.class));
                    }

                    @Override
                    public void onPageStarted(WebView view, String url, Bitmap
favicon) {
                        // Perform any actions you want when the page starts
loading
                        super.onPageStarted(view, url, favicon);
                        Objects.requireNonNull(getSupportActionBar()).setTitle
("Loading");
```

```
                    }

                    @Override
                    public void onPageFinished(WebView view, String url) {
                        // Retrieve the page title from the Web View
                        String pageTitle = view.getTitle();

                        // Set the title of the action bar
                        Objects.requireNonNull(getSupportActionBar()).setTitle
(pageTitle);
                    }

                });


//        Show a Progress Bar indicating the loading Status of the Website
till the Website is fully loaded
                webView.setWebChromeClient(new WebChromeClient() {
                    public void onProgressChanged(WebView view, int progress)
{
//                Show Progressbar till the loading has not reached 100%
                        if (progress < 100) {
                            progressBar.setVisibility(View.VISIBLE);
                        } else {
//                      Hide the Progress bar once the page is fully loaded
                            progressBar.setVisibility(View.GONE);
                        }

                    }
                });


//        Turn Javascript on in the web page
                webSettings = webView.getSettings();
                webSettings.setJavaScriptEnabled(true);
                if (CheckInternent()) {
                    webView.loadUrl(url);
                } else {
                    startActivity(new Intent(this, NoInternent.class));
                }
            }


            public boolean CheckInternent() {
                try {

//            Use the Connectivity Manager Class to get Network Info from
Android device
```

```
                    ConnectivityManager connectivityManager =
(ConnectivityManager) getSystemService(Context.CONNECTIVITY_SERVICE);
                    NetworkInfo info = null;
//            Check if the device is connected to internet through Wi-Fi or
Mobile Network
                    if (connectivityManager != null) {
                        info = connectivityManager.getActiveNetworkInfo();
                    }
                    return info != null && info.isConnected();
                } catch (NullPointerException e) {
                    return false;
                }
            }


            @Override
            protected void onCreate(Bundle savedInstanceState) {
                super.onCreate(savedInstanceState);
                setContentView(R.layout.activity_main);

//                Firebase App Check
                FirebaseApp.initializeApp(/*context=*/ this);
                FirebaseAppCheck firebaseAppCheck =
FirebaseAppCheck.getInstance();
                firebaseAppCheck.installAppCheckProviderFactory(
                        PlayIntegrityAppCheckProviderFactory.getInstance());

//        Link XML and Java

                drawerLayout = findViewById(R.id.draw_layout);
                navigationView = findViewById(R.id.navigation_view);
                webView = findViewById(R.id.webView);
                progressBar = findViewById(R.id.progressBar);
                firebaseAnalytics = FirebaseAnalytics.getInstance(this);

                // Initialize the Firebase Realtime Database
                mDatabase = FirebaseDatabase.getInstance().getReference();

//        Toggle Button for Navigation
//        Create the Toggle button to Show and Hide the handburger Menu
                toggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.open_menu, R.string.close_menu);

                //        Link the Drawer Layout from XML with he toggle
Button in the Action Bar
                drawerLayout.addDrawerListener(toggle);
```

```
                // pass the Open and Close toggle for the drawer layout
listener
                // to toggle the button
                toggle.syncState();

                // to make the Navigation drawer icon always appear on the
action bar// to make the Navigation drawer icon always appear on the action
bar
                Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAs
UpEnabled(true);


//        Set Item listener
                navigationView.setNavigationItemSelectedListener(item -> {
                    switch (item.getItemId()) {
//                Check which menu item is clicked and then open the
corresponding activity depending on wheather Internet Connection is present or
not
                        case R.id.home:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
MainActivity.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
//                    Close the Navigation Drawer once a particular item is
clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.latest_arrival:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
LatestArrival.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
//                    Close the Navigation Drawer once a particular item is
clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.all_categories:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
AllCategory.class));
                            } else {
```

```
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.dashboard:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
AccountDashboard.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.help:
                            if (CheckInternent()) {
                                startActivity(new Intent(this, help.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                        case R.id.privacy:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
PrivacyPolicy.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                    }


                    return true;
                });
```

```
        if (!CheckInternent()) {
            startActivity(new Intent(this, NoInternent.class));
        }


    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        if (toggle.onOptionsItemSelected(item)) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onBackPressed() {
        if (drawerLayout.isDrawerOpen(GravityCompat.START)) {
            drawerLayout.closeDrawer(GravityCompat.START);
        } else if (webView.canGoBack()) {
            webView.goBack();
        } else {
            super.onBackPressed();
        }
    }

    @Override
    protected void onStart() {
        super.onStart();
        FetchURL();
    }

    @Override
    protected void onResume() {
        super.onResume();
        FetchURL();
    }
}
```

10.1.9.2.2 Error Page

```
package com.xrcvc.e_library;

import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.media.AudioAttributes;
import android.media.SoundPool;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Button;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.analytics.FirebaseAnalytics;

import java.util.Objects;

public class ErrorPage extends AppCompatActivity {

    Button Gohome;

    //    Navigation Menu Items
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle toggle;

    //    Notification Sound using Sound Pool
    SoundPool soundPool;
    int notifysound;

    FirebaseAnalytics firebaseAnalytics;

    public boolean CheckInternent() {
        try {
            ConnectivityManager connectivityManager = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
            NetworkInfo info = null;
            if (connectivityManager != null) {
                info = connectivityManager.getActiveNetworkInfo();
```

```
            }
            return info != null && info.isConnected();
        } catch (NullPointerException e) {
            return false;
        }
    }

    public void NoInternetDialog() {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);

        // Set the message show for the Alert time
        builder.setMessage("Please Connect to the Internet.");

        // Set Alert Title
        builder.setTitle("No internet !");

        // Set Cancelable false for when the user clicks on the outside the
Dialog Box then it will remain show
        builder.setCancelable(false);

        // Set the positive button with yes name Lambda OnClickListener method
is use of DialogInterface interface.
        builder.setPositiveButton("Open Network Settings ",
(DialogInterface.OnClickListener) (dialog, which) -> {
            // When the user click yes button then app will close
            OpenNetworkSetings();
        });

        // Set the Negative button with No name Lambda OnClickListener method
is use of DialogInterface interface.
        builder.setNegativeButton("No", (DialogInterface.OnClickListener)
(dialog, which) -> {
            // If user click no then dialog box is canceled.
            dialog.cancel();
        });
        builder.setIcon(R.drawable.wifi_off);

        // Create the Alert dialog
        AlertDialog alertDialog = builder.create();
        // Show the Alert Dialog box
        alertDialog.show();
    }

    private void OpenNetworkSetings() {
//        OPen Android Network Settings
        startActivity(new
Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));
```

```
        startActivity(new
Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));
    }


    public void createsoundpool() {
//        Create the SoundPool
        AudioAttributes audioAttributes = new AudioAttributes.Builder()
//        Usage value to use when the usage is sonification, such as with user
interface sounds.
                .setUsage(AudioAttributes.USAGE_ASSISTANCE_SONIFICATION)
//        Content type value to use when the content type is a sound used to
accompany a user action, such as a beep or sound effect expressing a key
click, or event, such as the type of a sound for a bonus being received in a
game. These sounds are mostly synthesized or short Foley sounds.
                .setContentType(AudioAttributes.CONTENT_TYPE_SONIFICATION)
                .build();

        soundPool = new SoundPool.Builder()
//                Set the Maximum number of audio files can be loaded in
memory and played
                .setMaxStreams(1)
                .setAudioAttributes(audioAttributes)
                .build();

//        Create sounds Pool object with Audio Attributes object and load the
Wav File in memory. Call the audio file from the raw folder using it's id.
//        Set the priority of the audio file for compatibility with future
versions of Android.
        notifysound = soundPool.load(this, R.raw.windowsnotify, 1);
    }

    public void PlayNotifySound() {
//        Load the audio file , set the volume for the left , and right
channel of audio on the speaker, the priority, weather to loop or not and the
rate at which the audio file should be played.
        soundPool.play(notifysound, 1, 1, 0, 0, 1);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_error_page);

        firebaseAnalytics= FirebaseAnalytics.getInstance(this);

        Gohome = findViewById(R.id.gohome);
        Gohome.setOnClickListener(v -> {
```

```
            if (CheckInternent()) {
                startActivity(new Intent(ErrorPage.this, MainActivity.class));
            } else {
                PlayNotifySound();
                NoInternetDialog();
            }
        });

        //        Create and Load the Sound for Notifications
        createsoundpool();
        drawerLayout = findViewById(R.id.draw_layout);
        navigationView = findViewById(R.id.navigation_view);


        //        Toggle Button for Navigation
//        Create the Toggle button to Show and Hide the handburger Menu
        toggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.open_menu, R.string.close_menu);

        //        Link the Drawer Layout from XML witht he toggle Button in
the Action Bar
        drawerLayout.addDrawerListener(toggle);

        // pass the Open and Close toggle for the drawer layout listener
        // to toggle the button
        toggle.syncState();

        // to make the Navigation drawer icon always appear on the action
bar// to make the Navigation drawer icon always appear on the action bar
        Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnable
d(true);


//        Set Item listener
        navigationView.setNavigationItemSelectedListener(item -> {
                    switch (item.getItemId()) {
//                Check which menu item is clicked and then open the
corresponding activity depending on wheather Internet Connection is present or
not
                        case R.id.home:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
MainActivity.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
```

```
//                    Close the Navigation Drawer once a particular item is
clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.latest_arrival:
                        if (CheckInternent()) {
                            startActivity(new Intent(this,
LatestArrival.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
//                    Close the Navigation Drawer once a particular item is
clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.all_categories:
                        if (CheckInternent()) {
                            startActivity(new Intent(this,
AllCategory.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
                        //                    Close the Navigation Drawer
once a particular item is clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.dashboard:
                        if (CheckInternent()) {
                            startActivity(new Intent(this,
AccountDashboard.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
                        //                    Close the Navigation Drawer
once a particular item is clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.help:
                        if (CheckInternent()) {
                            startActivity(new Intent(this, help.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
```

```
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                        case R.id.privacy:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
PrivacyPolicy.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                    }
                    return true;
                }
        );
    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        if (toggle.onOptionsItemSelected(item)) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onBackPressed() {
        if (drawerLayout.isDrawerOpen(GravityCompat.START)) {
            drawerLayout.closeDrawer(GravityCompat.START);
        }
        else{
            startActivity(new Intent(this,MainActivity.class));
        }
    }
}
```

10.1.9.2.3 No Internet

```
package com.xrcvc.e_library;

import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.media.AudioAttributes;
import android.media.SoundPool;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Button;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.analytics.FirebaseAnalytics;

import java.util.Objects;

public class NoInternent extends AppCompatActivity {
    Button Gohome;

    //    Navigation Menu Items
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle toggle;

    //    Notification Sound using Sound Pool
    SoundPool soundPool;
    int notifysound;

    FirebaseAnalytics firebaseAnalytics;

    public boolean CheckInternent() {
        try {
            ConnectivityManager connectivityManager = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
            NetworkInfo info = null;
            if (connectivityManager != null) {
                info = connectivityManager.getActiveNetworkInfo();
            }
```

```
            return info != null && info.isConnected();
        } catch (NullPointerException e) {
            return false;
        }
    }

    public void NoInternetDialog() {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);

        // Set the message show for the Alert time
        builder.setMessage("Please Connect to the Internet.");

        // Set Alert Title
        builder.setTitle("No internet !");

        // Set Cancelable false for when the user clicks on the outside the
Dialog Box then it will remain show
        builder.setCancelable(false);

        // Set the positive button with yes name Lambda OnClickListener method
is use of DialogInterface interface.
        builder.setPositiveButton("Open Network Settings ",
(DialogInterface.OnClickListener) (dialog, which) -> {
            // When the user click yes button then app will close
            OpenNetworkSetings();
        });

        // Set the Negative button with No name Lambda OnClickListener method
is use of DialogInterface interface.
        builder.setNegativeButton("No", (DialogInterface.OnClickListener)
(dialog, which) -> {
            // If user click no then dialog box is canceled.
            dialog.cancel();
        });
        builder.setIcon(R.drawable.wifi_off);

        // Create the Alert dialog
        AlertDialog alertDialog = builder.create();
        // Show the Alert Dialog box
        alertDialog.show();
    }

    private void OpenNetworkSetings() {
//        OPen Android Network Settings
        startActivity(new
Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));
        startActivity(new
Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));
```

```
    }


    public void createsoundpool() {
//        Create the SoundPool
        AudioAttributes audioAttributes = new AudioAttributes.Builder()
//        Usage value to use when the usage is sonification, such as with user
interface sounds.
                .setUsage(AudioAttributes.USAGE_ASSISTANCE_SONIFICATION)
//        Content type value to use when the content type is a sound used to
accompany a user action, such as a beep or sound effect expressing a key
click, or event, such as the type of a sound for a bonus being received in a
game. These sounds are mostly synthesized or short Foley sounds.
                .setContentType(AudioAttributes.CONTENT_TYPE_SONIFICATION)
                .build();

        soundPool = new SoundPool.Builder()
//                Set the Maximum number of audio files can be loaded in
memory and played
                .setMaxStreams(1)
                .setAudioAttributes(audioAttributes)
                .build();

//        Create sounds Pool object with Audio Attributes object and load the
Wav File in memory. Call the audio file from the raw folder using it's id.
//        Set the priority of the audio file for compatibility with future
versions of Android.
        notifysound = soundPool.load(this, R.raw.windowsnotify, 1);
    }

    public void PlayNotifySound() {
//        Load the audio file , set the volume for the left , and right
channel of audio on the speaker, the priority, weather to loop or not and the
rate at which the audio file should be played.
        soundPool.play(notifysound, 1, 1, 0, 0, 1);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_no_internent);

        firebaseAnalytics= FirebaseAnalytics.getInstance(this);

        Gohome = findViewById(R.id.gohome);
        Gohome.setOnClickListener(v -> {
            if (CheckInternent()) {
```

```
                startActivity(new Intent(NoInternent.this,
MainActivity.class));
            } else {
                PlayNotifySound();
                NoInternetDialog();
            }
        });

        //        Create and Load the Sound for Notifications
        createsoundpool();
        drawerLayout = findViewById(R.id.draw_layout);
        navigationView = findViewById(R.id.navigation_view);


        //        Toggle Button for Navigation
//        Create the Toggle button to Show and Hide the handburger Menu
        toggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.open_menu, R.string.close_menu);

        //        Link the Drawer Layout from XML witht he toggle Button in
the Action Bar
        drawerLayout.addDrawerListener(toggle);

        // pass the Open and Close toggle for the drawer layout listener
        // to toggle the button
        toggle.syncState();

        // to make the Navigation drawer icon always appear on the action
bar// to make the Navigation drawer icon always appear on the action bar
        Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnable
d(true);


//        Set Item listener
        navigationView.setNavigationItemSelectedListener(item -> {
                    switch (item.getItemId()) {
//                Check which menu item is clicked and then open the
corresponding activity depending on wheather Internet Connection is present or
not
                        case R.id.home:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
MainActivity.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
```

```
//                    Close the Navigation Drawer once a particular item is
clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.latest_arrival:
                        if (CheckInternent()) {
                            startActivity(new Intent(this,
LatestArrival.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
//                    Close the Navigation Drawer once a particular item is
clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.all_categories:
                        if (CheckInternent()) {
                            startActivity(new Intent(this,
AllCategory.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
                        //                    Close the Navigation Drawer
once a particular item is clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.dashboard:
                        if (CheckInternent()) {
                            startActivity(new Intent(this,
AccountDashboard.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
                        //                    Close the Navigation Drawer
once a particular item is clicked
                        drawerLayout.closeDrawer(GravityCompat.START);
                        break;
                    case R.id.help:
                        if (CheckInternent()) {
                            startActivity(new Intent(this, help.class));
                        } else {
                            startActivity(new Intent(this,
NoInternent.class));
                        }
```

```
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                        case R.id.privacy:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
PrivacyPolicy.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                    }
                    return true;
                }
        );

    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        if (toggle.onOptionsItemSelected(item)) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onBackPressed() {
        if (drawerLayout.isDrawerOpen(GravityCompat.START)) {
            drawerLayout.closeDrawer(GravityCompat.START);
        } else if (CheckInternent()) {
            startActivity(new Intent(NoInternent.this, MainActivity.class));
        } else {
            PlayNotifySound();
            NoInternetDialog();
        }
    }
}

package com.xrcvc.e_library;
```

```
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.media.AudioAttributes;
import android.media.SoundPool;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.Bundle;
import android.view.MenuItem;
import android.widget.Button;

import androidx.annotation.NonNull;
import androidx.appcompat.app.ActionBarDrawerToggle;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.GravityCompat;
import androidx.drawerlayout.widget.DrawerLayout;

import com.google.android.material.navigation.NavigationView;
import com.google.firebase.analytics.FirebaseAnalytics;

import java.util.Objects;

public class NoInternent extends AppCompatActivity {
    Button Gohome;

    //    Navigation Menu Items
    DrawerLayout drawerLayout;
    NavigationView navigationView;
    ActionBarDrawerToggle toggle;

    //    Notification Sound using Sound Pool
    SoundPool soundPool;
    int notifysound;

    FirebaseAnalytics firebaseAnalytics;

    public boolean CheckInternent() {
        try {
            ConnectivityManager connectivityManager = (ConnectivityManager)
getSystemService(Context.CONNECTIVITY_SERVICE);
            NetworkInfo info = null;
            if (connectivityManager != null) {
                info = connectivityManager.getActiveNetworkInfo();
            }
            return info != null && info.isConnected();
        } catch (NullPointerException e) {
```

```
            return false;
        }
    }

    public void NoInternetDialog() {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);

        // Set the message show for the Alert time
        builder.setMessage("Please Connect to the Internet.");

        // Set Alert Title
        builder.setTitle("No internet !");

        // Set Cancelable false for when the user clicks on the outside the
Dialog Box then it will remain show
        builder.setCancelable(false);

        // Set the positive button with yes name Lambda OnClickListener method
is use of DialogInterface interface.
        builder.setPositiveButton("Open Network Settings ",
(DialogInterface.OnClickListener) (dialog, which) -> {
            // When the user click yes button then app will close
            OpenNetworkSetings();
        });

        // Set the Negative button with No name Lambda OnClickListener method
is use of DialogInterface interface.
        builder.setNegativeButton("No", (DialogInterface.OnClickListener)
(dialog, which) -> {
            // If user click no then dialog box is canceled.
            dialog.cancel();
        });
        builder.setIcon(R.drawable.wifi_off);

        // Create the Alert dialog
        AlertDialog alertDialog = builder.create();
        // Show the Alert Dialog box
        alertDialog.show();
    }

    private void OpenNetworkSetings() {
//        OPen Android Network Settings
        startActivity(new
Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));
        startActivity(new
Intent(android.provider.Settings.ACTION_WIRELESS_SETTINGS));
    }
```

```
    public void createsoundpool() {
//        Create the SoundPool
        AudioAttributes audioAttributes = new AudioAttributes.Builder()
//        Usage value to use when the usage is sonification, such as with user
interface sounds.
                .setUsage(AudioAttributes.USAGE_ASSISTANCE_SONIFICATION)
//        Content type value to use when the content type is a sound used to
accompany a user action, such as a beep or sound effect expressing a key
click, or event, such as the type of a sound for a bonus being received in a
game. These sounds are mostly synthesized or short Foley sounds.
                .setContentType(AudioAttributes.CONTENT_TYPE_SONIFICATION)
                .build();

        soundPool = new SoundPool.Builder()
//                Set the Maximum number of audio files can be loaded in
memory and played
                .setMaxStreams(1)
                .setAudioAttributes(audioAttributes)
                .build();

//        Create sounds Pool object with Audio Attributes object and load the
Wav File in memory. Call the audio file from the raw folder using it's id.
//        Set the priority of the audio file for compatibility with future
versions of Android.
        notifysound = soundPool.load(this, R.raw.windowsnotify, 1);
    }

    public void PlayNotifySound() {
//        Load the audio file , set the volume for the left , and right
channel of audio on the speaker, the priority, weather to loop or not and the
rate at which the audio file should be played.
        soundPool.play(notifysound, 1, 1, 0, 0, 1);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_no_internent);

        firebaseAnalytics= FirebaseAnalytics.getInstance(this);

        Gohome = findViewById(R.id.gohome);
        Gohome.setOnClickListener(v -> {
            if (CheckInternent()) {
                startActivity(new Intent(NoInternent.this,
MainActivity.class));
            } else {
                PlayNotifySound();
```

```
                NoInternetDialog();
            }
        });

        //        Create and Load the Sound for Notifications
        createsoundpool();
        drawerLayout = findViewById(R.id.draw_layout);
        navigationView = findViewById(R.id.navigation_view);


        //        Toggle Button for Navigation
//        Create the Toggle button to Show and Hide the handburger Menu
        toggle = new ActionBarDrawerToggle(this, drawerLayout,
R.string.open_menu, R.string.close_menu);

        //        Link the Drawer Layout from XML witht he toggle Button in
the Action Bar
        drawerLayout.addDrawerListener(toggle);

        // pass the Open and Close toggle for the drawer layout listener
        // to toggle the button
        toggle.syncState();

        // to make the Navigation drawer icon always appear on the action
bar// to make the Navigation drawer icon always appear on the action bar
        Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnable
d(true);


//        Set Item listener
        navigationView.setNavigationItemSelectedListener(item -> {
                    switch (item.getItemId()) {
//                Check which menu item is clicked and then open the
corresponding activity depending on wheather Internet Connection is present or
not
                        case R.id.home:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
MainActivity.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
//                    Close the Navigation Drawer once a particular item is
clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.latest_arrival:
```

```
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
LatestArrival.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
//                    Close the Navigation Drawer once a particular item is
clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.all_categories:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
AllCategory.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.dashboard:
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
AccountDashboard.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;
                        case R.id.help:
                            if (CheckInternent()) {
                                startActivity(new Intent(this, help.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                        case R.id.privacy:
```

```
                            if (CheckInternent()) {
                                startActivity(new Intent(this,
PrivacyPolicy.class));
                            } else {
                                startActivity(new Intent(this,
NoInternent.class));
                            }
                            //                    Close the Navigation Drawer
once a particular item is clicked
                            drawerLayout.closeDrawer(GravityCompat.START);
                            break;

                    }
                    return true;
                }
        );

    }

    @Override
    public boolean onOptionsItemSelected(@NonNull MenuItem item) {
        if (toggle.onOptionsItemSelected(item)) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onBackPressed() {
        if (drawerLayout.isDrawerOpen(GravityCompat.START)) {
            drawerLayout.closeDrawer(GravityCompat.START);
        } else if (CheckInternent()) {
            startActivity(new Intent(NoInternent.this, MainActivity.class));
        } else {
            PlayNotifySound();
            NoInternetDialog();
        }
    }
}
```

*10.1.9.3 Hamburger Menu (Menu File)*

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <group android:checkableBehavior="single">
        <item
            android:id="@+id/home"
            android:icon="@drawable/home"
            android:title="@string/home_menu_item" />

        <item
            android:id="@+id/latest_arrival"
            android:icon="@drawable/latest_arrivals"
            android:title="@string/latest_arrival" />

        <item
            android:id="@+id/all_categories"
            android:icon="@drawable/all_categories"
            android:title="@string/all_categories" />
    </group>

    <!--Admin Menu Section-->
    <group android:checkableBehavior="single">
        <item android:title="@string/Admin_menu_section_header">
            <menu>
                <item
                    android:id="@+id/dashboard"
                    android:icon="@drawable/dashboard"
                    android:title="@string/admin_dashboard_menu_item">

                </item>
            </menu>
        </item>
    </group>


    <!--    Help Section-->
    <group android:checkableBehavior="single">
        <item android:title="@string/help_and_feedback_menu_item">
            <menu>
                <item
                    android:id="@+id/help"
                    android:icon="@drawable/help"
                    android:title="@string/help_menu_item">

                </item>

                <item
                    android:id="@+id/privacy"
```

```
                    android:icon="@drawable/lock"
                    android:title="@string/privacy_poicy">

                </item>
            </menu>
        </item>
    </group>

    <!--    Exit App-->
    <group android:checkableBehavior="single"/>
</menu>
```

*10.1.9.4 Shortcuts XML File*

```
<?xml version ="1.0" encoding="utf-8"?>
<shortcuts xmlns:android="http://schemas.android.com/apk/res/android">

<!--    Home Activity-->
    <shortcut
        android:enabled="true"
        android:icon="@drawable/home"
        android:shortcutId="home"
        android:shortcutLongLabel="@string/home_menu_item"
        android:shortcutShortLabel="@string/home_menu_item">

        <intent
            android:action="android.intent.action.VIEW"
            android:targetClass="com.xrcvc.e_library.MainActivity"
            android:targetPackage="com.xrcvc.e_library" />
    </shortcut>

    <!--    latest Arrivals Activity-->
    <shortcut
        android:enabled="true"
        android:icon="@drawable/latest_arrivals"
        android:shortcutId="Latest Arrivals"
        android:shortcutLongLabel="@string/latest_arrival"
        android:shortcutShortLabel="@string/latest_arrival">

        <intent
            android:action="android.intent.action.VIEW"
            android:targetClass="com.xrcvc.e_library.LatestArrival"
            android:targetPackage="com.xrcvc.e_library" />
    </shortcut>

    <!--    All Categories  Activity-->
    <shortcut
        android:enabled="true"
        android:icon="@drawable/all_categories"
        android:shortcutId="All Categories"
        android:shortcutLongLabel="@string/all_categories"
        android:shortcutShortLabel="@string/all_categories">

        <intent
            android:action="android.intent.action.VIEW"
            android:targetClass="com.xrcvc.e_library.AllCategory"
            android:targetPackage="com.xrcvc.e_library" />
    </shortcut>

    <!--    Account Dashboard  Activity-->
```

```
    <shortcut
        android:enabled="true"
        android:icon="@drawable/dashboard"
        android:shortcutId="Dashboard"
        android:shortcutLongLabel="@string/admin_dashboard_menu_item"
        android:shortcutShortLabel="@string/admin_dashboard_menu_item">

        <intent
            android:action="android.intent.action.VIEW"
            android:targetClass="com.xrcvc.e_library.AccountDashboard"
            android:targetPackage="com.xrcvc.e_library" />
    </shortcut>
</shortcuts>
```

*10.1.9.5 Android Manifest*

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.CHANGE_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
    <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/app_logo"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/Theme.ELibrary"
        tools:targetApi="31">
        <activity
            android:name=".ErrorPage"
            android:exported="false" />
        <activity
            android:name=".help"
            android:exported="false"
            android:label="@string/ErrorPage"/>
        <activity
            android:name=".PrivacyPolicy"
            android:exported="false"
            android:label="@string/privacy_poicy" />
        <activity
            android:name=".AccountDashboard"
            android:exported="false"
            android:label="@string/admin_dashboard_menu_item" />
        <activity
            android:name=".AllCategory"
            android:exported="false"
            android:label="@string/all_categories" />
        <activity
            android:name=".LatestArrival"
            android:exported="false"
            android:label="@string/latest_arrival" />
        <activity
            android:name=".NoInternent"
            android:exported="false"
            android:label="@string/no_internet"
            android:noHistory="true" />
```

```
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
            <!-- App Shortcuts -->
            <meta-data
                android:name="android.app.shortcuts"
                android:resource="@xml/shortcuts" />
        </activity>
    </application>

</manifest>
```

#### *10.1.9.6 Strings XML File*

```
<resources>
    <string name="app_name" translatable="false">XRCVC E Book Library</string>
    <string name="xrcvc_and_st_logo">XRCVC and St. Xavier\'s Logo</string>
    <string name="open_menu">Open Menu</string>
    <string name="close_menu">close Menu</string>
    <string name="home_menu_item">Home</string>
    <string name="help_and_feedback_menu_item">Help and Feedback</string>
    <string name="Admin_menu_section_header">My Account</string>
    <string name="admin_dashboard_menu_item">Dashboard</string>
    <string name="help_menu_item">help</string>
    <string name="privacy_poicy">Privacy Poicy</string>
    <string name="disclaimer_menu_item">Disclaimer</string>
    <string name="latest_arrival">Latest Arivals</string>
    <string name="all_categories">All Categories</string>
    <string name="site_logo">XRCVC and St. Xaviers Logo</string>
    <string name="reload">Reload Site</string>
    <string name="no_internet">No Internet  Connection</string>
    <string name="goHomeBTN">Go Home</string>
    <string name="Exitapp">Exit</string>
    <string name="Disclaimer">Disclaimer</string>
    <string name="ErrorPage">Error Page</string>
    <string name="invalidURL">This Page is currently Unavailable</string>
</resources>
```

#### *10.1.9.7 Colour XML File*

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="primary">#F07C00</color>
    <color name="secondary">#F7BE8B</color>
    <color name="text">#000000</color>
    <color name="accent">#24408E</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="page_background_colour">#FEFFF1</color>
</resources>
```

## 10.2 Unit Testing

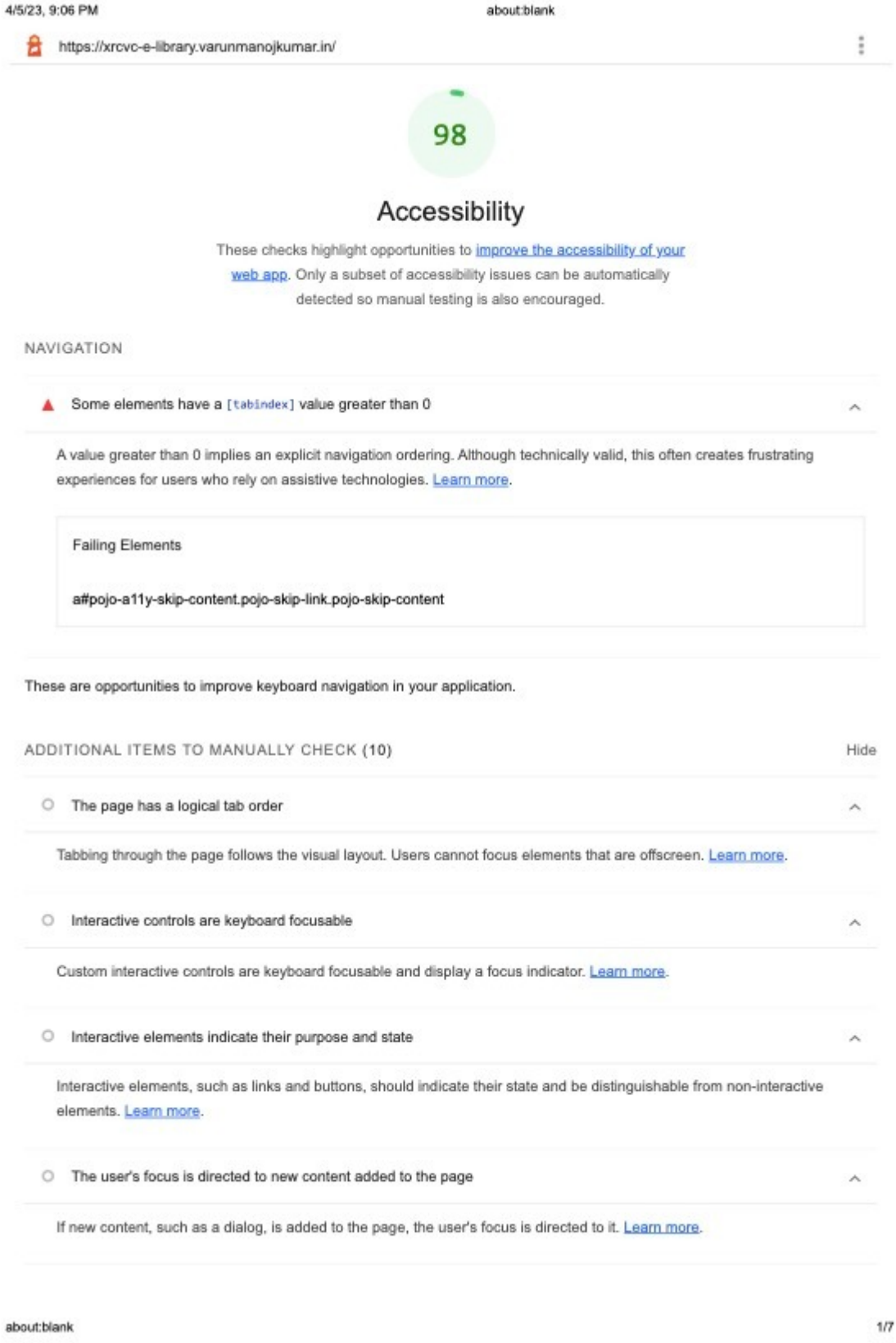
| Sr No | Module | Test Case | Expected Result | Actual Result |
|---|---|---|---|---|
| 1 | HOMEPAGE | Visit the XRCVC eLibrary website | Display homepage | Displays the homepage |
| | FIND YOUR PERFECT BOOK | Enter Any part/word of Title/Author or ISBN in Search Box | Display all books having that word in the Title.<br>Display Detail Search Page and list of Books in Grid View by default | Preview a list of books that match the search criteria. |
| | | Click on List View | Listing of books displayed in a List View | The Books are displayed in a List view with the option to view the Detail page of a Single Book. |
| 2 | SEARCH & FILTERS | Search Filters – CATEGORY - Select any one Category and click on APPLY FILTERS Button | Display books from selected Category | All Books that match the selected Category are displayed. |
| | | Select any Book Format, or enter Author, Publisher or ISBN no or Language and click on APPLY FILTERS button | Books from selected Filter criteria would be displayed | Books that match all the criteria are displayed in the view specified. |
| | | Select any SORT By criteria | Books would be sorted as per criteria selected | Books are sorted by the criteria |
| 3 | REQUEST BOOK FORM | Fill name, email address, phone and other details and click on REQUEST BOOK button | The request is made only once all mandatory fields are satisfied. | Once all mandatory fields are satisfied, the student is redirected to Thank you page. |

| | | | | |
|---|---|---|---|---|
| 4 | LATEST ARRIVALS | Click on LATEST ARRIVALS tab on top of the menu | The books would be displayed in order of book dates - Latest to earliest.<br><br>All three views – Grid View, List View and Table View would be available | The Last 8 books that are been uploaded on the website are displayed.<br>The Student has the option to view books in:<br>• Grid View<br>• List View<br>• Table View |
| 5 | ALL CATEGORIES | Click on ALL CATEGORIES tab on top of the menu | All the different categories of books would be displayed along with the number of books in each category. | List of All Book Categories are displayed and Student can click on either the Image or Title to view the list of books filtered by that category. |
| 6 | LOGIN (only for Admin ) | Enter Invalid Name or Password | Should display error | Error: The password you entered for the email address **'ENTERED EMAIL ID'** is incorrect. Lost your password? |
| | | Enter Valid Name and Password and click on Login | Should log into the Website as Admin User | Logs in user and redirected to Dashboard Page |
| 7 | ADD NEW BOOK | Navigate to Add New Book Tab. Enter details of book to be added | Should add new book to platform | Book is added and redirected to newly added book page. |
| 8 | VIEW BOOK REQUESTS | Navigate to View Book Requests Tab | Should display the list of books requested by students | Displays a table showing the list of bools requested with the option to Export Data |

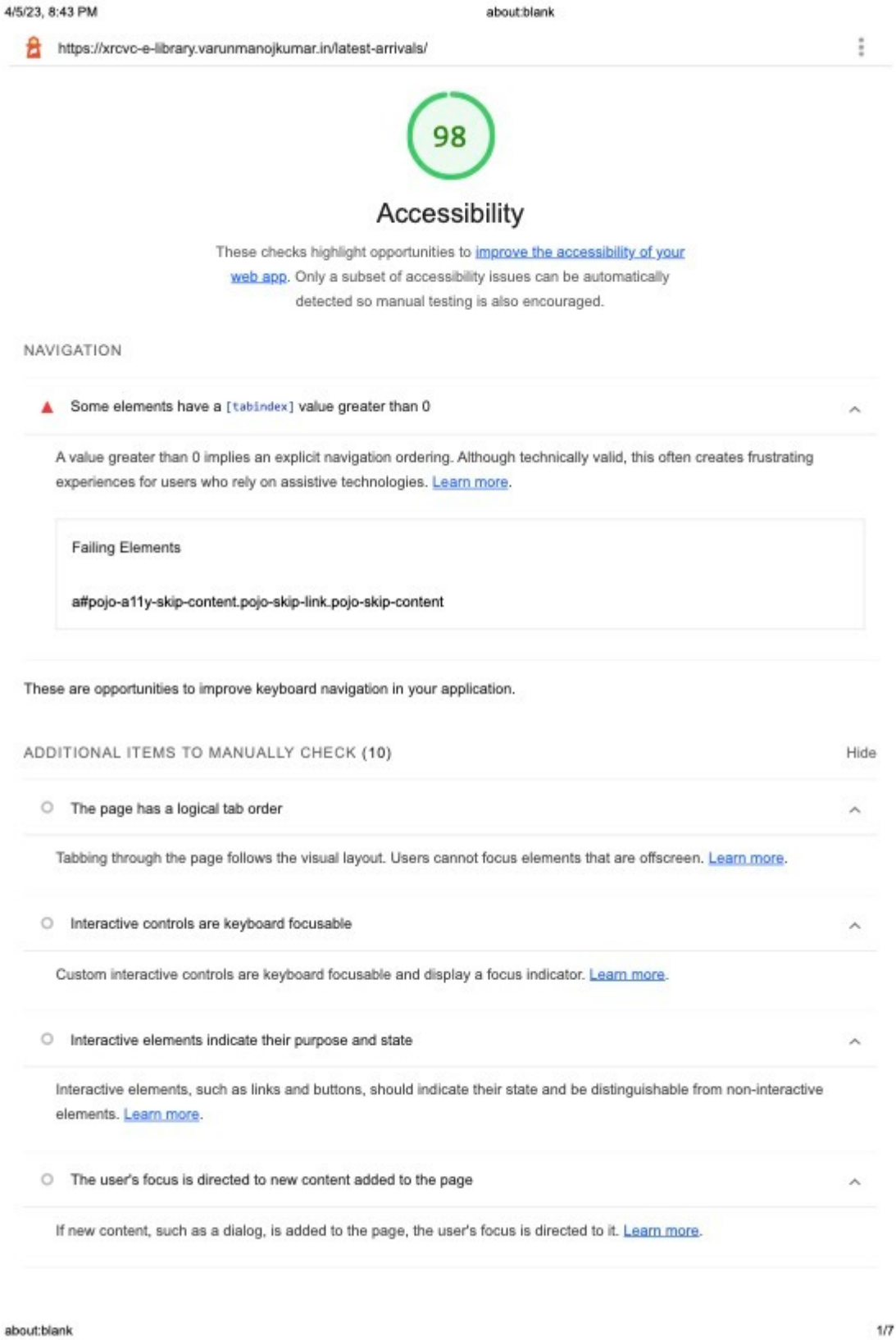
| | | | | |
|---|---|---|---|---|
| 9 | GENERATE REPORTS | Navigate to Reports Tab on Dashboard | Should display 3 reports | Displays 3 reports<br>1. Total Book Requests<br>2. Book Requests per Student<br>3. Book Request Status |

## 10.3 Accessibility Test Reports (Lighthouse)

### 10.3.1 Home page Report

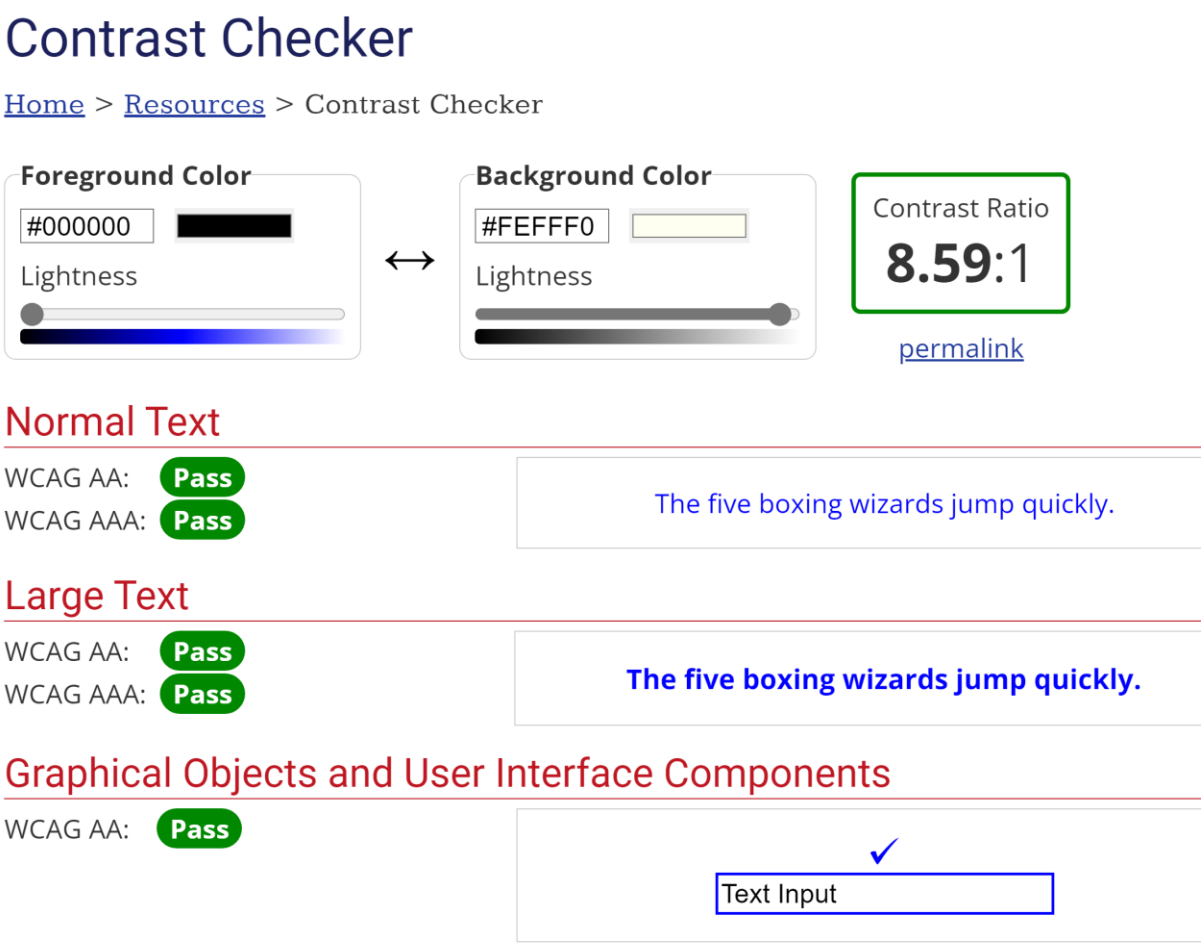


## 10.3.2 Latest Arrivals Accessibility Report

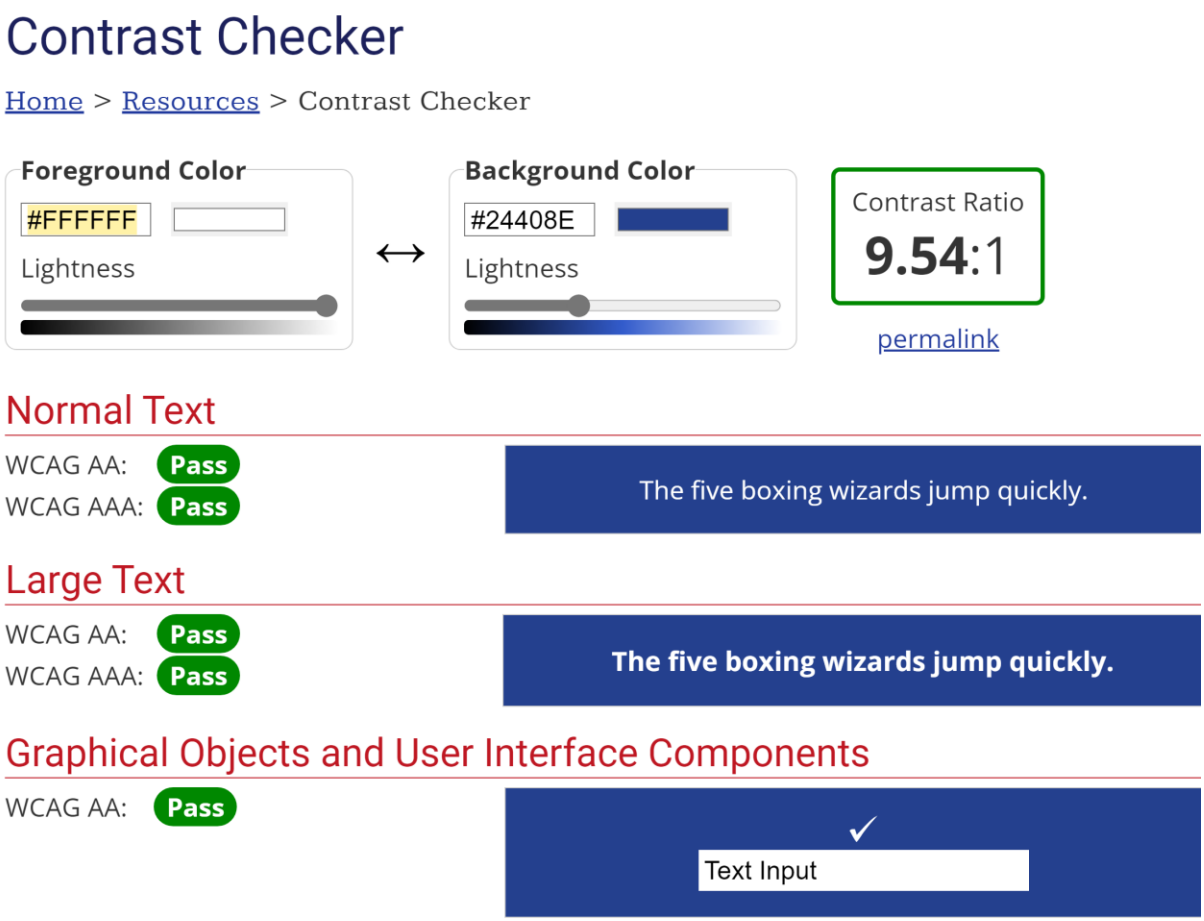


### 10.3.3 Colour Contrast Checker

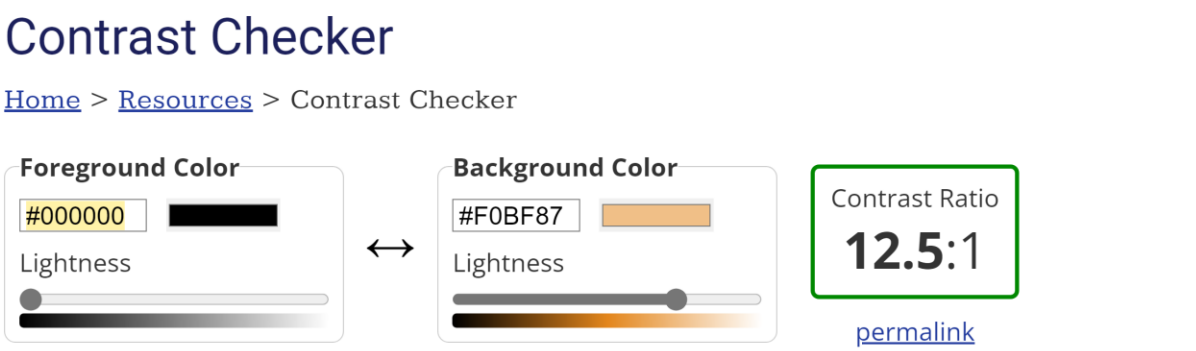
#### *10.3.3.1 Contrast between Page Background and Text Colour*



*10.3.3.2 Buttons Background Colour and Text Colour*



*10.3.3.3 Header Background Colour*



# 11 Conclusion and Future Enhancement

The XRCVC eLibrary has been developed to meet the requirements of the XRCVC faculty and for the benefit of visually challenged students who require accessible books. This system both online and on the mobile app would make it simple and easy for students to search and access any books that are available in the digital library.

The demo of the system has been given to the XRCVC faculty and they have appreciated the fact that this system would reduce their manual work.

The system was also given for testing to a few visually challenged students in the college to get their feedback from an accessibility and ease of use perspective. The students have appreciated the simplicity to use the system with assistive technology.

Few testimonials of the feedback have been given in a subsequent section.

The XRCVC may have a few future additional requirements and would reach out to me as and when required.

# 12 Testimonials

**1. Feedback from XRCVC Faculty**

Finally, thanks for the great job done by Varun Kumar. He kept the same visual design of the existing website. The good part was that he kept accessibility in mind and made it accessible with the screen reader and also maintained the color contrast for low-vision users. The primary benefit of the e-Library System is that it will reduce manual work to a great extent and streamline library activities. Students will be able to search for and request books online. At the same time, staff members will be able to manage book requests and update the library catalog in real time. It will significantly improve the efficiency and effectiveness of library operations. He was pretty understanding of the issue and gave his valuable suggestions. In the future, we will need to update the features and then will get in touch with you.

*Sushil Pandit*
*Sr. Consultant - Technology*
*Xavier's Resource Centre for the Visually Challenged*
*(XRCVC)*
*sushil@xrcvc.org*

**2. Feedback from visually challenged ex-student of St. Xavier's College**

User Feedback for XRCVC E library website made by Varun Kumar in collaboration with XRCVC:

Tools used for Testing: NVDA screen reader, Windows inbuilt magnifier and different inbuilt display theme settings.

The user interface of the website is accessible and distinguished users can access it with various customized options available on the page like color themes, text magnifier etc.

NVDA screen reader was able to read out all the headings and buttons on the webpage. Everything could be done easily by using assistive technology tools.

Screen magnification and invert colors could be implemented and a person with low vision would not have trouble accessing the website.

Overall, the design and the functionalities are easy to use and accessible and hence using this product will be a hassle free experience for the disabled.

*Tanmay Shinde*
*Operations Analyst,*
*Corporate & Investment bank,*
*JP Morgan Chase & Co.*

tanmayshinde820@gmail.com

### 3. Feedback from visually challenged student

As someone with low vision, I have found the XRCVC e-library to be a truly remarkable resource. The website is easy to access, with a well-structured layout that makes it simple to find the books and resources that I need.

One thing that I really appreciate about this e-library is the option to send a request for a book. Compared to other e-libraries I've used, this process is incredibly easy and straightforward. And the response time is impressively prompt.

Another feature that I like about the XRCVC e-library is the simple procedure for accessing the books. The color contrast and font size are carefully chosen to make the text as easy to read as possible. This makes a huge difference for me, and it's something that I don't always find in other e-libraries.

*Neelam Kukreja*
*TY-BA*
*TY-BA*
*St. Xavier's College*
neelam.kukreja@xaviers.edu.in

### 4. Feedback from visually challenged student

Personally, I feel that the website is so amazing. It is very easy to use and understand. Since it is so easy to navigate one can use it without any difficulties. The very last website which I used for reading books was Sugamyapustakalaya long back ago, but again, personally I find this website slightly easier for the users. Since I have not been using Sugamyapustakalaya website these days, I can’t comment more about it but over all this website is so good.

**Shreyas More**
**TY-BA**
**St. Xavier’s College**
**shreyas.more@xaviers.edu.in**

# 13 References

## 13.1 For Website


1. https://codex.wordpress.org/
2. https://wordpress.org/
3. https://datatables.net/
4. JetEngine Plugin for Elementor | Crocoblock
5. JetSmartFilters Plugin for Elementor | Crocoblock
6. Crocoblock JetTable Builder Export to CSV, PDF, Excel (github.com)
7. DataTables example - File export
8. Customize date and time format – WordPress.org Documentation
9. One Click Accessibility – WordPress plugin | WordPress.org
10. WordPress Hosting | Lightning-Fast Performance | Hostinger India


## 13.2 For Android App


5. Android WebView progress bar - Stack Overflow
6. Create shortcuts | Android Developers
7. Firebase App Check (google.com)
8. Firebase Realtime Database (google.com)


## 13.3 For Accessibility


9. WebAIM: Contrast Checker
10. Lighthouse overview - Chrome Developers


## 13.4 For Documentation


11. Xavier's Resource Center for the Visually Challenged (xrcvc.org)
12. Home|Sugamya Pustakalaya (daisyindia.org)
13. Hostinger Academy


# 14 Bibliography


Software Engineering – A Practitioners Approach