Airport(Malir) Campus

Final Year Project Report

BS (COMPUTER SCIENCE)

For

DonoBook: A Cognitive Platform for the Intelligent System of Collaborative Sharing and Exchange of Educational Resources

Repository: https://github.com/AdeemAmir/DonoBook

Submitted by

1.	Name:	**Adeem Amir**
	Reg #:	025-23-16050 (**IU05-0123-0228**)
	CGPA:	3.7
	Email:	adeem.16050.ac@iqa.edu.pk
2.	Name:	**Chaudhry Sajjawal Ali**
	Reg #:	025-23-16059 (**IU05-0123-0237**)
	CGPA:	3.34
	Email:	chaudhry.16059.ac@iqra.edu.pk

Supervisor

Dr. Affan Alim

Coordinator

Dr. Muhammad Naveed

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and the accompanied report entirely based on our personal efforts. If any part of this project is proved to be copied out from any source or found to be the reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Adeem Amir

Sajjawal Ali

Certificate of Approval

It is to certify that the final year project of BS (CS) **"DonoBook: A Cognitive Platform for the Intelligent System of Collaborative Sharing and Exchange of Educational Resources"** was developed by *Adeem Amir (IU05-0123-0228)* and *Chaudhry Sajjawal Ali (IU05-0123-0237)* under the supervision of *"Dr. Affan Alim"* and that in his opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Science.

Dr. Affan Alim

(Supervisor)

Dr. Muhammad Naveed

(Coordinator)

Dr. Affan Alim

(Head of Department - CS)

Abstract

DonoBook is a web-based cognitive platform utilizing Progressive Web Application (PWA) technique, that enables parents, students, and trusted vendors to share, exchange, lend, and sell educational resources (physical and digital) easily and conveniently. Built with modern web technologies (Vite, Tailwind, React), Supabase as backend (database, auth, realtime), and geospatial features, DonoBook incorporates assistive AI features such as OCR-based metadata extraction, basic recommendations, and a conversational assistant to support user interaction within a Progressive Web Application. This report describes the requirements, architecture, design, implementation details, testing, deployment, and evaluation for the project.

Acknowledgement

All praise and gratitude are due to Allah, rabbul-izza, whose countless blessings and guidance have enabled us to embark on and complete this project successfully.

We extend our heartfelt thanks to our academic supervisor and Head of the Computer Science Department, Dr. Affan Alim, for his invaluable guidance and support throughout this journey. Our sincere gratitude for his encouragement and insights that played a pivotal role in shaping this project and our future.

We are deeply grateful to the faculty and administration of Iqra University for their continuous encouragement and for providing us with a conducive environment for learning and growth. Special thanks are extended to our peers and the development community for their valuable feedback and contributions.

We also express our profound gratitude to our families for their unwavering support, patience, and prayers, which have been a source of strength and motivation throughout this endeavor.

Table of Contents_Toc219077040

Chapter 1: Project Introduction

1.1 Project Background

Introduction:

Education remains one of the most significant recurring expenses for families, particularly in developing economies where inflation, rising printing costs, and frequent curriculum revisions continuously increase the price of school textbooks and learning materials. Each academic year, parents are required to purchase new or revised editions of these resources, even though previously used books often remain academically valid and physically usable. As a result, a large volume of educational resources is underutilized or discarded, leading to unnecessary financial burden and material waste.

At the same time, the digital transformation of society has reshaped how communities interact, share resources, and build trust. Online platforms have successfully enabled peer-to-peer exchanges in domains such as transportation, accommodation, and general e-commerce. However, educational resource exchange, especially at the school level, has not seen equivalent structured, intelligent, and localized solutions. Existing marketplaces are either too generic, lack trust mechanisms, or are not designed to align with academic progression (for example, matching students moving from Class X to Class X+1 with those entering Class X).

Furthermore, the manual process of finding second-hand resources through informal channels, school WhatsApp groups, local vendors, or word-of-mouth, is inefficient, fragmented, and unreliable. Parents often remain unaware that suitable resources are available within close geographic proximity. Even when such resources exist, the lack of standardized metadata (title, edition, ISBN, condition), absence of verification, and poor communication channels create friction and discourage reuse.

Recent advancements in web technologies, geospatial systems, artificial intelligence, and Progressive Web Applications (PWA) provide an opportunity to address these challenges holistically. Optical Character Recognition (OCR) can automate the extraction of book details from images, while recommendation systems can intelligently suggest relevant resources based on class, subject, and location. Real-time communication and authentication frameworks further enable trust-based interactions within digital communities.

DonoBook is proposed within this context as a cognitive, AI-assisted platform that bridges the gap between surplus and demand of educational resources. By combining community-based sharing with intelligent automation, DonoBook aims to modernize textbook exchange into a structured, accessible, and sustainable digital ecosystem.

1.2 Problem Statement

Parents frequently incur unnecessary expenses by purchasing despite the availability of reusable second-hand resources within proxoimity but, remain unknown. DonoBook is here to become the centralized, trusted, and location-aware platform that will reduce resource wasteage and highlight the existence of these educational materials.

1.3 Motivation

Even with modern technology and online marketplaces available today, many tools still fail to help parents and students easily reuse educational resources. Most parents end up buying brand-new textbooks every academic year because they are unaware that usable second-hand books are available nearby. This results in unnecessary financial burden, wasted pages, and discarded educational materials. The goal of this project is to create a simple and intelligent platform that helps families discover, share, and reuse educational resources within their local communities.

Here are the main reasons for developing the **DonoBook** platform:

Reducing Unnecessary Expenses:
Parents often spend large amounts of money on new textbooks even when suitable second-hand books already exist nearby. DonoBook helps reduce these costs by making existing resources visible and accessible.

Avoiding Resource Waste:
Many textbooks remain in good condition but are thrown away or stored unused after one academic year. DonoBook encourages reuse, reducing paper waste and promoting sustainability.

Easy Discovery of Educational Resources:
Finding second-hand books through informal channels is difficult and unreliable. DonoBook provides a centralized system where users can easily search and discover books based on class, subject, and location.

Making Technology Easy to Use:
By using features such as image-based book listing, automatic data extraction, and simple navigation, the platform minimizes effort for parents and non-technical users.

Using Modern Technology:
The platform utilizes modern web technologies, artificial intelligence, and location-based services to recommend relevant books and simplify the exchange process.

Built for the Community:
DonoBook is designed with parents, students, and trusted welfare organizations in mind, aiming to create a safe, community-based environment for sharing educational resources.

By addressing these issues, DonoBook aims to reduce financial stress on families, prevent educational resource waste, and provide an intelligent and user-friendly solution for textbook sharing and exchange. The platform promotes affordability, sustainability, and independence in accessing educational materials.

1.4 Aim and Objectives

Aim

To develop an intelligent, community-based digital platform that enables parents, students, and trusted welfare organizations to easily share, exchange, lend, and reuse educational resources, reducing financial burden and minimizing educational material waste.

Objectives

The key objectives of this project are to:

1. Design and develop a mobile-first Progressive Web Application using React and modern frontend tooling.
2. Implement a secure backend using Supabase for authentication, database management, storage, and real-time communication.
3. Develop an OCR-assisted metadata extraction pipeline to minimize manual effort during listing creation.
4. Integrate AI-assisted normalization and recommendation mechanisms to enhance listing accuracy and discovery relevance.
5. Enable real-time chat and negotiation features supported by intelligent message suggestions.
6. Incorporate geospatial search and map-based discovery for location bounding and better matches.
7. Enforce strong security measures through role-based access control and row-level security policies.
8. Evaluate the system based on functionality, usability, performance, and accuracy.

Benefits:

a) **Reduces Educational Costs:**
The platform helps parents save money by enabling access to nearby second-hand textbooks instead of purchasing new ones every academic year.

b) **Prevents Resource Waste:**
Reusable books and learning materials are reused rather than discarded, reducing paper waste and promoting sustainability.

c) **Easy to Use:**
The system provides a simple and user-friendly interface, allowing non-technical users to list and find books with minimal effort.

d) **Intelligent Discovery:**
By using AI-based recommendations and location-based search, users can quickly find relevant resources based on class, subject, and proximity.

e) **Safe Community Exchange:**
The platform supports verified users, ratings, and direct communication to build trust within the community.

Deliverables:

In-scope:

a) **Web-Based Application:**
A responsive Progressive Web Application (PWA) accessible on mobile phones, tablets, and desktops.

b) **Automated Book Listing:**
OCR-assisted extraction of book and resource details such as title, author, and edition, for books and type and category from images.

c) **Intelligent Recommendation System:**
An AI-powered module that suggests relevant educational resources based on user preferences and academic level.

d) **Geolocation-Based Discovery:**
Map-based and nearby search functionality to locate resources within a specific region.

e) **Testing and Evaluation Results:**
Test cases and performance results to validate system accuracy, usability, and reliability.

f) **Deployment-Ready System:**
A fully functional system ready for real-world use with plans for monitoring, maintenance, and future enhancements.

Out-of-Scope:

a) Full marketplace economics (commissioning, escrow).

b) Complete payment gateway integration (planned beyond the scope of FYP).

c) Native mobile app (PWA only).

d) Advanced and/or custom ML research\experiments\api model integration (demo project will use pre-built embedder/models).

Key Features:

a) **Fast and Convenient Listing:** photo upload → OCR → LLM normalization → pre-filled form → publish

b) **Map Discovery:** a map with clustered markers and a geo search radius

c) **Chat & Negotiation:** Supabase Realtime with AI suggested counters and template messages

d) **Recommendations:** semantic + geo ranking

e) **QR Flow:** QR per listing opens detail & chat

f) **Welfare Verification & Ratings** for trust

g) **PWA:** offline-capable shell, installable on mobile

1.5 Problem Solution for Proposed System

a) A web-based application that enables parents and students to list, discover, and exchange educational resources easily.

b) An intelligent system that recommends relevant textbooks based on class, subject, and location.

c) Automated extraction of book details using image-based OCR to reduce manual effort.

d) A simple and intuitive user interface designed for non-technical users.

e) A trusted, community-driven platform supported by user verification, ratings, and real-time communication.

f) Documentation and guidelines for users and developers to ensure ease of use and future scalability.

Chapter 2: Related System Analysis/Literature Review

2.1 Review

2.1.1 DonoBook

2.1.1.1 Introduction:

Several systems and platforms have been developed to support the buying, purchasing and excahnge of books and educational materials. Most of these systems, however, focus on general e-commerce or informal community exchanges rather than structured educational reuse. Individuals still often rely on social media groups, local vendors, or general online marketplaces to find second-hand textbooks, which makes the process unorganized and inefficient.

Existing platforms usually lack academic-level categorization such as class, subject, or curriculum alignment. In addition, users are required to manually enter book details, which can be time-consuming and error-prone. The absence of location-based discovery and trust mechanisms further limits the effectiveness of these systems for school-level textbook exchange.

The DonoBook project is inspired by these existing solutions but focuses specifically on school-oriented educational resource sharing. The system aims to simplify discovery, listing, and exchange of textbooks by combining structured data, basic intelligent features, and localized access.

In conclusion, while previous systems highlight the demand for educational resource sharing, there is a clear need for a simple, intelligent, and community-focused platform that reduces cost, effort, and waste. DonoBook aims to address these limitations by providing an accessible and practical solution for parents and students.

2.1.1.2 Working Model:

User-Friendly Web Interface:
The system is developed as a web-based Progressive Web Application (PWA) with a simple and intuitive interface suitable for non-technical users.

Book Listing through Images:
Users can upload images of textbooks, allowing the system to assist in extracting basic book information instead of requiring full manual entry.

Automated Metadata Assistance:
The platform uses OCR-based assistance to identify book details such as title, author, and edition, which users can review and edit if needed.

Search and Discovery:
Users can search for educational resources based on class, subject, and location to find nearby available books.

Communication between Users:
The system allows direct communication between users to discuss exchange, lending, or pricing of educational resources.

Performance and Reliability:
The platform is designed and tested to ensure responsive interaction, accurate data handling, and stable performance under normal usage conditions.

Error Handling and Guidance:
Clear system messages and guidance are provided if incorrect input is detected or required information is missing.

2.1.1.3 Advantages:

- **Cost Reduction:**
Helps parents save money by enabling access to second-hand textbooks.

- **Resource Reuse:**
Encourages reuse of educational materials, reducing waste.

- **Ease of Use:**
Designed for parents and students with minimal technical knowledge.

- **Localized Discovery:**
Allows users to find books available within nearby locations.

- **Time Efficient:**
Reduces the time spent searching for books through informal channels.

- **Community-Based:**
Supports interaction within a trusted and shared educational community.

- **Scalable Design:**
The system can be extended with additional features in the future.

2.1.1.4 Disadvantages:

• Dependence on Internet Access:
The platform requires a stable internet connection for full functionality.

• OCR Limitations:
Automated book detail extraction may not always be accurate and may require manual correction.

• User Adoption:
Effectiveness depends on user participation and willingness to share resources.

• Limited Initial Coverage:
In early stages, the number of available books may be limited.

• Privacy Concerns:
Users may have concerns about sharing location or information.

• Device Compatibility:
Although web-based, performance may vary across older devices or browsers.

2.1.1.5 Summary:

The **DonoBook** platform is developed to assist individuals and educational communities by providing a simple digital system for sharing and reusing educational resources. The platform focuses on reducing unnecessary textbook purchases by making nearby second-hand books visible and easily accessible.

Upon accessing the system, users are guided through a straightforward interface that allows them to search, list, and communicate about educational materials. Book listings can be created using images, and the system assists users by automatically extracting basic book information, reducing manual effort. Users can then review, update, or proceed with the listing.

The platform allows users to interact directly through built-in communication features to discuss exchange, lending, or pricing. Location-based discovery helps users find relevant books within nearby areas, improving efficiency and convenience.

Overall, DonoBook emphasizes simplicity, accessibility, and practicality. By combining structured listings with basic intelligent assistance, the system helps users reuse educational resources independently without relying on informal or unreliable channels.

2.1.2 Core Research Area A - Modern web architecture & Progressive Web Applications (PWA): building maintainable, scalable, high-engagement web apps:

Overview & Importance:

PWA adoption & impact studies (2023–2024): Empirical studies and engineering reports show that PWA adoption improves mobile retention and conversions when combined with service workers, app manifest, and careful caching strategies. The literature highlights three high-impact engineering practices:

1) server-side rendering (or hybrid) for first paint,

2) service-worker-driven offline UX (caching listing drafts and recent searches), and

3) push notifications for timely transactional updates. See Industry case studies (e.g., web.dev) PWA case studies for concrete evidence and recommended patterns.

Performance & perceptual speed: Research from 2022–2024 emphasizes that perceived performance (time-to-first-interaction, skeleton UIs, progressive image loading) is more important than raw throughput. For DonoBook, measure and tune Lighthouse metrics (performance, accessibility, best practices) and use progressive image loading for listing images (client-side resizing, WebP/AVIF where possible).

Scalability & maintainability for full web apps: Recent engineering literature (2022–2024) stresses modular architecture (component-based frontend, edge functions for server-side tasks, BaaS like Supabase to reduce backend maintenance) and automated pipelines (CI/CD with preview environments). This is important for your FYP: recommending Supabase + Edge Functions (for OCR orchestration) and Vite + React (for a modular PWA frontend) reduces implementation time while keeping the architecture production-appropriate.

Key Technical Takeaways

1. **PWA basic**s: The literature consistently recommends the use of application manifests and service workers to enable offline functionality, background synchronization, and installability, particularly for transactional PWAs.
2. **Progressive Enhancement**: Keep the app fully usable without JavaScript where possible for critical flows (viewing listings), but provide app-like features in JS for a full experience.
3. **Image Pipeline**: Perform client-side resizing and compress to WebP/AVIF before upload to lower storage and speed up OCR. Consider using background uploads so users can keep using the UI.

4. **Edge Compute for Heavy Tasks**: Run OCR orchestration and LLM calls from server/edge functions (Supabase Edge or Cloudflare Workers) to protect service keys and centralize caching/rate-limiting.
5. **Measure & Iterate**: Instrument Lighthouse and user-experience metrics (First Input Delay, Largest Contentful Paint) in your test plan.

2.1.3 Core Research Area B – User psychology, trust, and engagement in peer-to-peer marketplaces and web apps

Overview & Importance to DonoBook

A platform for used textbooks depends on rapid user adoption and trust: parents must feel comfortable transacting with neighbors or welfare organizations. The literature since 2022 emphasizes that **trust signals**, **reputation design**, **visual quality** (images), and **usable, interpretable UI** matter more than raw feature lists. Additionally, user motivation and retention are boosted by social proof, easy onboarding, incentives, and low-friction listing flows.

Recent research and findings

Reducing perceived risk in sharing platforms (Marth et al., 2022): platform signals such as verification badges, clear return/transaction policies, and dispute-resolution workflows reduce perceived risk and increase willingness to transact. For DonoBook, welfare verification and an admin-moderated badge program can materially improve adoption.

Trust, privacy & safety factors (ACM 2024): peer-to-peer systems must explicitly surface safety mechanisms — for example, identity verification, safety guidelines, and complaint handling — and surface them at the moment of decision (listing detail pages). The study suggests combining technical safeguards with human-centric policies for best effects.

Visual quality & conversion: image quality correlates with trust and conversion rates. Practical implication: enforce minimum image quality (resolution/clarity), provide in-app tips for good photos (lighting, flat surface), and auto-enhance or guide the user with overlays during capture.

UX & engagement research

Engagement drivers: research shows that interactive elements (quick filters, saved searches, push notifications for matches, and QR-based sharing) increase return visits and conversion. Minimal friction on core tasks (create listing, contact exchangers) increases adoption.

Simplicity beats feature clutter: academic evidence and industry reports both recommend focusing on a few high-quality features rather than a long laundry list. For DonoBook, prioritize OCR-assisted listing, map discovery, and chat/negotiation — each polished — before adding extras.

Design & Persuasion Techniques

Onboarding flow: short progressive onboarding that collects only necessary info (email + grade) - minimize friction.

Trust-first listing page: highlight verification badge, welfare rating, recent transaction count, and a clear pickup/meeting guideline.

A/B testing & analytics: run small A/B experiments to measure whether green "verified" badges or textual trust statements produce better conversion.

2.1.4 Platforms for Textbook Exchange & Community Sharing

Small-scale and campus initiatives for second-hand textbook exchange continue to appear in the literature and in practice, showing demand for localised solutions that reduce cost and waste (campus exchange stations, university programs). Platform work typically combines simple listing mechanics with trust-building features to increase adoption. Examples of campus initiatives and small platform implementations (2022–2024) demonstrate feasibility and community uptake, and are useful precedents for DonoBook's school-focused scope.

Implication for DonoBook: local, school-level onboarding (parents + verified welfare organizations) and a simple UX for rapid listing/claiming improves adoption over general-purpose marketplaces.

2.1.5 Trust, Reputation, and User Behavior in Peer-to-Peer Marketplaces

Trust plays a critical role in peer-to-peer platforms, particularly where transactions occur between individuals who do not know each other. Features such as visible user ratings, transaction history, and clear communication channels help reduce uncertainty and improve user confidence during exchanges.

In community-driven systems, reputation indicators and verification mechanisms are especially important, as they encourage participation while discouraging misuse.

Implication for DonoBook: The platform incorporates ratings, welfare verification, image guidance, and in-app messaging to improve trust and reduce perceived risk during exchanges.

2.1.6 Optical Character Recognition (OCR) & LLM Post-processing for Book Metadata

Optical Character Recognition has reached a level of maturity where it can reliably assist in extracting basic textual information from printed materials such as book covers. However, OCR performance may degrade when images contain stylized fonts, low contrast, or poor lighting conditions. As a result, fully automated extraction is not always reliable in practical, real-world usage.

To address these limitations, many modern applications adopt a hybrid approach in which OCR is used to extract raw text, followed by automated post-processing or rule-based normalization to structure the extracted information. This approach reduces user effort while still allowing manual correction where necessary.

Implication for DonoBook: DonoBook adopts OCR as an assistive mechanism rather than a fully autonomous system. Extracted metadata is presented to users in an editable form, ensuring accuracy while significantly reducing manual data entry effort.

2.1.7 Recommender Systems for Educational Resources (2021–2025 literature)

Recommendation systems are increasingly used in educational platforms to assist users in discovering relevant learning resources. Rather than relying on complex predictive models, many practical systems prioritize contextual relevance, such as academic level, subject alignment, and user interaction history.

For platforms focused on resource reuse, simple hybrid approaches combining content similarity with basic interaction signals are often sufficient and easier to maintain. Transparency in recommendations, such as explaining why an item is suggested, further improves user trust and acceptance.

Implication for DonoBook: DonoBook employs a lightweight recommendation strategy that considers class or grade relevance, subject similarity, and geographic proximity. The system emphasizes simplicity and explainability over complex predictive modeling.

2.1.8 Real-time Communication (Supabase Realtime) & Authentication (Supabase Auth + JWT + RLS)

Supabase is positioned as a full-stack backend for web apps and offers: Auth (JWT tokens), Postgres DB, Storage, Realtime (broadcast via Postgres changes), and Edge Functions. Supabase's documentation is explicit that JWTs are used for authentication and for enabling Row Level Security (RLS) policies in Postgres, and that tokens expose claims usable by RLS. Use of Supabase also simplifies implementing authenticated realtime chat and RLS-protected access to messages and listings.

Implication for DonoBook: Supabase is an appropriate choice for rapid development: use Supabase Auth (JWT) + RLS policies for secure per-row access; use Supabase Realtime for chat; use Edge Functions for OCR orchestration and notification logic.

2.2 Traditional Method Comparison

Table: Related System Analysis with Proposed Project Solution

Application	**Weakness**	**Proposed Project Solution**
General Online Marketplaces	- Not designed specifically for educational resources - No academic class or subject alignment	- DonoBook is focused on school-level textbooks - Books are categorized by class and subject
Social Media Groups	- Unstructured posts - Difficult to search or verify availability	- Centralized platform with structured listings - Search and filter options
Local Book Vendors	- Limited stock visibility - Higher prices	- Community-driven sharing and exchange - Promotes reuse at lower or no cost
Existing Exchange Platforms	- Manual data entry - No intelligent assistance	- Image-based listing with OCR assistance - Simplified discovery and communication

2.3 Key Benefits and Beneficiaries

1. Centralized Book Discovery:
The platform allows users to search and discover educational resources from a single system instead of relying on multiple informal sources.
Beneficiaries: Parents, students

2. Automated Listing Assistance:
By allowing users to upload book images and receive assisted metadata extraction, the system reduces effort and improves consistency.
Beneficiaries: Parents, welfares, students

3. Location-Based Access:
Users can find books available within nearby locations, making exchanges faster and more convenient.
Beneficiaries: Parents and students within local communities

4. Simple and Accessible Design:
The platform is designed to be easy to use for non-technical users, ensuring smooth navigation and interaction.
Beneficiaries: Parents, guardians, general users

2.4 Summary

A review of existing educational resource sharing methods reveals several limitations, including lack of structure, poor discoverability, and absence of academic alignment. Informal platforms such as social media groups are difficult to manage, while general marketplaces do not cater specifically to school-level educational needs.

Some systems also require excessive manual effort for listing resources and provide limited support for finding nearby books. These challenges often result in parents purchasing new textbooks despite the availability of reusable materials.

The DonoBook project addresses these issues by offering a centralized, community-focused platform designed specifically for educational resource sharing. The system provides structured listings, assisted data entry, and localized discovery while maintaining a simple and user-friendly interface.

By focusing on affordability, reuse, and accessibility, DonoBook aims to reduce educational costs, minimize resource waste, and improve access to learning materials. The platform provides a practical and reliable solution that better meets the needs of parents and students compared to existing approaches.

Chapter 3: Research Approach

3.1 Development / Research Methodology

Introduction

The goal of the DonoBook project is to design and develop a web-based intelligent platform that assists parents, students, and trusted welfares in sharing, exchanging, and reusing educational resources, particularly school textbooks. The system focuses on reducing unnecessary educational expenses and minimizing resource waste by making nearby second-hand books visible and accessible.

The project follows a **practical, system-oriented development approach**, integrating modern web technologies with basic artificial intelligence techniques to simplify book discovery, listing, and interaction. Rather than relying on theoretical experimentation, the research emphasizes applied development, usability, and feasibility in real-world scenarios.

Detailed feasibility analysis and wireframes are provided in Appendices.

Technologies Used

Frontend Framework (React + Vite):
Used to build a responsive and interactive user interface. React enables component-based development, while Vite provides fast builds and efficient development workflow.

Tailwind CSS:
Used for designing a clean, responsive, and user-friendly interface suitable for non-technical users.

Supabase:
Acts as the backend service, providing database management, authentication, file storage, and real-time communication features.

PostgreSQL (Supabase Database):
Used for structured storage of users, book listings, transactions, and related data.

OCR Assistance (Image-Based Extraction):
Used to assist users by extracting basic book details (e.g., title, author) from uploaded images, reducing manual data entry.

• AI-Based Recommendation Logic:
Basic recommendation techniques are used to suggest relevant books based on class, subject, and user interaction patterns.

• Geolocation Services:
Used to support location-based discovery of educational resources within nearby regions.

Development Process

• Requirement Analysis:
Initial analysis was conducted to understand the problems faced by parents and students in obtaining affordable educational resources.

• System Design:
High-level architecture and data models were designed to ensure modularity, scalability, and ease of use.

• Frontend Development:
User interfaces were implemented with a focus on simplicity, accessibility, and responsiveness across devices.

• Backend Integration:
Authentication, database operations, file uploads, and real-time communication were integrated using Supabase services.

• AI Feature Integration:
OCR-based assistance and basic recommendation logic were added to improve usability and discovery efficiency.

• Testing and Refinement:
The system was tested to ensure correct functionality, responsiveness, and stable performance under typical usage conditions.

High-Level Algorithm

1. **User Access:**
   Users access DonoBook through a web browser on a mobile phone, tablet, or desktop device.

2. **Authentication:**
   Users register or log in using secure authentication services.

3. **Book Listing Creation:**
   Users upload images of textbooks and enter minimal information. The system assists by extracting basic details automatically.

4. **Data Processing:**
   Extracted metadata is validated and stored in the database for search and recommendation purposes.

5. **Search and Discovery:**
   Users search for books based on class, subject, and location. The system retrieves and ranks relevant listings.

6. **Recommendation Assistance:**
   The platform suggests related educational resources based on user preferences and academic level.

7. **User Interaction:**
   Users communicate directly through the platform to discuss exchange, lending, or pricing.

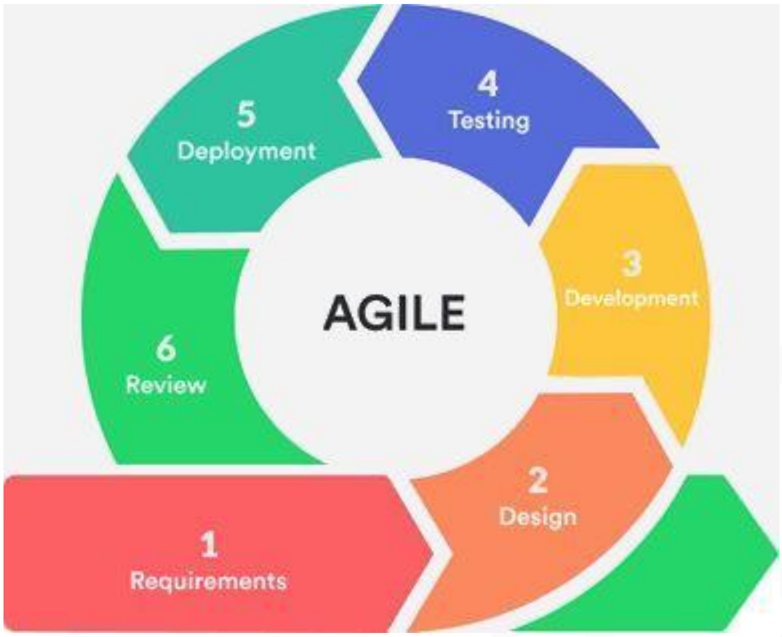
8. **Completion:**
   Transactions are finalized, and records are updated for future reference.

3.2 Development Model

The **Incremental-Iterative Agile Development Model** is used for the development of the **DonoBook** platform. This model supports releasing a usable version of the system early and then gradually enhancing it by adding new features over time. The approach is well suited for this project as it allows the core promised features of the Final Year Project to be delivered first, followed by continuous improvements and feature extensions.

In this model, **DonoBook** is initially deployed with essential functionality such as book listing, search, and user interaction. After deployment, additional features, such as intelligent recommendations, enhanced automation, and usability improvements, can be introduced in later iterations. User feedback will play an important role in guiding these enhancements.

The Incremental Agile model ensures flexibility, controlled development, and scalability while keeping the project achievable within academic constraints.



3.2.1. Requirement Gathering

User Research

During the requirements gathering phase, research was conducted to understand the challenges faced by parents and students in acquiring educational resources. Informal discussions, online research, and observation of existing practices helped identify pain points such as high textbook costs, lack of awareness of second-hand resources, and inefficient discovery methods.

Surveying the Community

Basic surveys and informal feedback were collected from parents, students, and peers to understand their expectations from a textbook sharing platform. This helped identify the need for simplicity, trust, and localized access rather than complex marketplace features.

Analyzing Existing Solutions

Existing solutions such as online marketplaces, social media groups, and local book vendors were analyzed. This analysis highlighted limitations including lack of academic categorization, unstructured listings, and absence of intelligent assistance.

Identifying Core Features

Based on the gathered information, the essential features for the first release were identified. These include user registration, book listing, search and filtering, basic communication, and location-based discovery. Advanced features were planned for later iterations.

Creating User Scenarios

Simple user scenarios were created to represent how parents and students would interact with the system, from listing a book to finding and exchanging it. These scenarios helped ensure that the system remains easy to use and practical.

3.2.2. Design Requirements

Converting Needs into System Features

The collected requirements were translated into functional and non-functional system requirements. This step defined how the platform should behave and what technologies would be used to implement it.

Designing a Simple and Accessible Interface

The user interface was designed to be clean, minimal, and easy to navigate for non-technical users. Emphasis was placed on clarity, readability, and straightforward interaction flows.

Creating Low-Fidelity Designs

Basic wireframes and screen layouts were created to visualize user interaction. These designs were reviewed and refined before implementation to avoid usability issues.

Initial Prototype Development

A basic prototype of DonoBook was developed that included core features such as book listing and browsing. This early version helped validate the system concept and guided further development.

3.2.3 Development \ Iteration

Incremental Feature Development

The system was developed in small increments. Each iteration focused on implementing or improving a specific feature, such as search, listing management, or user communication. This ensured steady progress and manageable complexity.

Core Deployment First

The initial deployment focused on FYP-promised features. Once a stable version was achieved, the system design allowed additional features to be added without major restructuring.

Continuous User Feedback

Feedback from users was collected after each iteration. Suggestions and usability issues were reviewed and used to refine the platform in subsequent updates.

Gradual Enhancement of Intelligent Features
Basic intelligent components, such as assisted metadata extraction and simple recommendations, were introduced carefully and improved over time based on system performance and feedback.

Testing and Quality Assurance
Each increment was tested for functionality, performance, and usability. Testing ensured that newly added features did not affect existing system stability and that the platform remained reliable across devices.

3.2.4 Justification

The **Incremental Agile Development Model** is chosen for the DonoBook project because it allows the system to be developed, deployed, and improved in manageable stages. This approach is suitable for a Final Year Project where core functionality must be delivered within limited time and resources, while still allowing room for future enhancement.

By releasing a stable version of the platform with essential features first, the project ensures that all promised requirements are implemented and tested effectively. Additional features can then be added gradually based on user feedback and system performance, without disrupting existing functionality.

This model also supports flexibility, as requirements may evolve during development. Incremental development reduces risk, improves quality through continuous testing, and ensures better alignment with user needs. Therefore, the **Incremental Agile model** provides a practical, scalable, and academically viable approach for the successful production and development of the **DonoBook** project.

3.3 Project Schedule / Gantt Chart

Weekly Expanded Gantt Chart (Nov 2025 – Apr 2026: 24 Week Project Timeline)

Total duration: ~24 weeks
Legend

🟦 = Main work
🟩 = Overlap / Partial
🟧 = Review / Testing
🟥 = Critical path
= Not active

3.3.1 Planning, Requirements & Research (Weeks 1–3)

Task	**W1**	**W2**	**W3**
Requirement refinement & scope freeze	🟦	🟦	
Literature review finalization	🟩	🟩	🟦
Technology validation (Supabase, OCR, Maps)	🟩	🟩	🟦
Initial architecture & data model	🟦	🟦	🟩
Project scheduling & milestone approval	🟦	🟦	🟩

3.3.2 System Design, UI/UX & Environment Setup (Weeks 4–6)

Task	**W4**	**W5**	**W6**
UI/UX wireframes (mobile-first)	🟦	🟦	
ERD, DFD, UML diagrams	🟦	🟦	
Frontend scaffolding (React PWA)	🟩	🟦	🟧
Backend setup (Supabase, Auth, RLS)	🟩	🟦	🟧
Storage & image pipeline setup	🟩	🟦	🟧

3.3.3 Core MVP Development (Weeks 7–13)

Task	W7	W8	W9	W10	W11	W12	W13
User authentication & profiles	🟥	🟥	🟥	🟧	🟧	🟧	
Listing creation & management	🟦	🟦	🟦	🟩	🟩	🟧	
Image upload & storage	🟩	🟦	🟦	🟦	🟩	🟩	🟧
Search & basic filters	🟩	🟦	🟦	🟦	🟦	🟧	🟧
Map integration		🟩	🟩	🟦	🟦	🟧	🟧
Realtime chat			🟩	🟦	🟦	🟧	🟧

3.3.4 Advanced Features & AI Integration (Weeks 14–20)

Task	W14	W15	W16	W17	W18	W19	W20
OCR pipeline (Tesseract / Cloud OCR)	🟩	🟩	🟦	🟥	🟥	🟥	🟧
AI metadata normalization (LLM)	🟩	🟦	🟦	🟩	🟥	🟥	🟧
Map clustering & geo-search	🟦	🟦	🟦	🟦	🟧	🟧	
Donation & Exchange flow	🟦	🟦	🟦	🟦	🟧	🟧	
Reviews & Ratings		🟩	🟩	🟦	🟦	🟧	
Chatbot Sssistant			🟩	🟦	🟦	🟧	

3.3.5 Testing, Optimization & Security (Weeks 21–22)

Task	W21	W22
Unit testing (frontend & backend)	🟩	🟩
Integration testing (OCR, chat, search)	🟦	
Performance testing	🟩	🟧
Security testing (RLS, auth, APIs)	🟥	🟥
Bug fixing & UI polish	🟩	🟧

3.3.6 Documentation, Deployment & Viva Preparation (Weeks 23–34)

Task	W23	W24
Final report & annexures	🟥	🟥
User & admin manuals	🟦	🟦
Final deployment	🟦	
Presentation slides & demo	🟩	🟦
Viva rehearsal & internal review	🟩	🟦

Summary (High-Level Timeline)

Phase	Weeks
Planning & Research	01–03
Design & Setup	04-06
Core MVP Development	07–13
Advanced Features & AI	14–20
Testing & Optimization	21–22
Finalization & Presentation	23–24

3.4 Risk Assessment & Mitigations

Risk	Likelihood	Impact	Mitigation
OCR inaccuracies on poor photos	High	Medium	LLM post-processing + UI for manual corrections + image-quality guidance
AI Misuse	High	Medium(PR)	Input filtering, rate limiting, and moderation safeguards.
Database Leak	Medium	High	Better security and strict endpoints.
Nominatim rate-limits / geocode failures	Medium	Medium	Cache geocode results; throttle; consider paid geocoding
Supabase pgvector not available	Medium	Medium	Fall back to external DB, easy port to MongoDB
Time overrun for ML tuning	Medium	High	Focus on a simple, robust recommender prototype (embedding + heuristic rank)
Security misconfiguration (RLS)	Low	High	Thorough RLS testing & staging; least privilege keys; code reviews

Chapter 4: Requirement Analysis

4.1 Product Perspective

DonoBook is a standalone platform built as a **PWA**, accessible on both mobile and desktop browsers. It integrates with:

- Large Language Model (LLM) service for metadata normalization and suggestions.
- OCR engine for extracting metadata from book cover images.
- Geolocation and geocoding services for map visualization.
- Secured backend for storing listings, user data, messages, reviews.
- Real-time communication service for chat and negotiation.

The system maintains extensibility to support future features such as payment integration, mobile native apps, and advanced recommendation models.

4.2 Product Functions

Key functions include:

User & Profile

- Create and authenticate accounts (individuals, welfares, admin)
- Set personal information and individual's status
- Upload verification documents (Welfare Organizations)

Listing Management

- Upload multiple images of books
- OCR extraction of metadata
- AI-assisted metadata refinement
- Edit and publish listing
- Update listing status (available / exchanging / donated)

Search & Discovery

- Text-based search with multiple filters
- Map-based discovery with clustered pins
- Radius or map-bounds search

Communication & Transactions

- Real-time chat between exchangers, donator and receiver.
- AI-assisted message suggestions
- Offer creation and response
- Transaction lifecycle: Ongoing → Completed

Recommendations

- Personalized recommendations based on:
  - class grade,
  - subject relevance,
  - past interactions,
  - geographic proximity.

Trust & Moderation

- Ratings and reviews.
- Admin moderation tools.

4.3 User Types

Parent/Student/Individuals

- Non-technical.
- Requires simple listing flows.
- Needs easy access to nearby textbook availability.

Admin

- Has full control over verification, moderation, and reports.
- Requires dashboards for review and action.

4.4 Operating Environment

- **Browsers:** Chrome, Firefox, Safari, Edge, etc.
- **Devices:** Android/iPhone (via PWA), Tablets.
- **Backend:** Cloud-hosted database, storage, and services.
- **Network:** Requires stable internet for OCR, geolocation, and chat.

4.5 Potential Design and Implementation Constraints

- OCR performance depends on lighting, camera quality.
- Nominatim/geocoding rate limits.
- PWA offline mode is limited to cached pages.
- Real-time chat performance depends on WebSocket stability.
- AI calls must be optimized to reduce latency and cost.

4.6 Assumptions and Dependencies

- Users will provide correct class/grade info.
- Third-party AI/OCR APIs will remain available.
- Map data (OSM) will remain accessible.
- An internet connection is available during listing creation.

4.7. Functional Requirements

User Authentication (FR-01)

a) The system shall allow users to register as one of the following roles: individuals, welfare organizations, and/or administrator.

b) The system shall allow users to authenticate using email and password credentials.

c) The system shall issue and manage secure auth-tokens for logged-in users.

d) The system shall restrict access to features based on authenticated user roles.

e) The system shall allow users to securely log out of the system.

User Profile Management (FR-02)

a) The system shall allow users to create and update personal profile information.

b) The system shall allow users to specify a individual's educational level.

c) The system shall allow users to view their own profile information.

d) The system shall protect profile data from unauthorized access.

e) The system shall allow welfare organizations, users to upload organiziation details and verification documents.

Book Listing Creation and Management (FR-03)

a) The system shall allow users to create a new book listing.

b) The system shall allow users to upload one or more images of a book.

c) The system shall extract textual information from uploaded images using OCR.

d) The system shall apply AI-assisted normalization to extracted metadata.

e) The system shall allow users to manually edit extracted metadata before publishing.

f) The system shall update the status of a listing (available, exchanging, donated).

g) The system shall allow users to delete or deactivate their own listings.

Search and Filtering (FR-04)

a) The system shall allow users to search for listings using textual queries.

b) The system shall allow filtering of listings by title, author, ISBN, subject, class/grade, condition, and ranged location.

c) The system shall allow users to view only free or exchange-only listings.

d) The system shall allow users to sort search results by relevance, distance, distance, or recency.

e) The system shall display only listings that match the applied filters.

Map-Based Discovery (FR-05)

a) The system shall display available listings on an interactive map.

b) The system shall show listings as location-based markers.

c) The system shall cluster markers in high-density areas.

d) The system shall update visible listings when the map view or radius changes.

e) The system shall allow users to select a marker to view listing details.

f) The system shall allow users to enable or disable location-based discovery.

Listing Details View (FR-06)

a) The system shall display detailed information for a selected listing.

b) The system shall display book images, title, author, edition, and description.

c) The system shall display the class or grade level associated with the book.

d) The system shall display exchangers profile information and rating.

e) The system shall display the approximate location of the listing on a map.

f) The system shall generate a QR code for sharing the listing.

g) The system shall provide options to initiate chat or make an offer.

Real-Time Chat and Negotiation (FR-07)

a) The system shall allow exchangers and donators to message each other in real time.

b) The system shall store chat history securely for each transaction.

c) The system shall allow users to send text-based messages.

d) The system shall notify users of new messages.

e) The system shall restrict chat access to only the involved parties.

Exchanges and Donations (FR-08)

a) The system shall allow welfare organizations to set up accounts.

b) The system shall allow exchangers to accept, reject, and exchange.

c) The system shall update listing status when an offer is accepted.

d) The system shall mark a transaction as completed after confirmation by users.

e) The system shall maintain a transaction history for each user.

Recommendation System (FR-09)

a) The system shall provide personalized book recommendations to users.

b) The system shall consider the user’s class or educational level in recommendations.

c) The system shall consider subject relevance and prior user interactions.

d) The system shall consider geographic proximity when ranking recommendations.

e) The system shall display recommended listings on the home or search interface.

Reviews and Ratings (FR-10)

a) The system shall allow users to rate each other after a completed transaction.

b) The system shall allow users to submit optional textual reviews.

c) The system shall calculate and display average ratings on user profiles.

d) The system shall restrict reviews to users involved in a completed transaction.

e) The system shall prevent modification of reviews after submission.

Administrative Moderation (FR-11)

a) The system shall allow administrators to verify welfare organization accounts.

b) The system shall allow administrators to review and remove inappropriate listings.

c) The system shall allow administrators to suspend or ban users for policy violations.

d) The system shall allow administrators to view system activity logs.

e) The system shall restrict administrative functions to authorized admin users only.

4.8 Non-Functional Requirements

Performance Requirements

1. Standard search results return within ~500ms
2. Map-bounded search <1000ms
3. OCR pipeline completes within 5–8 seconds
4. Real-time chat latency <500ms

Security Requirements

1. All communication over HTTPS
2. JWT authentication
3. File type restrictions on uploads
4. Access control by user role
5. Sanitize all user inputs (prevent XSS/SQL injection)
6. RLS or equivalent row-level access restrictions

Usability Requirements

1. Mobile-first interface
2. Simple step-by-step listing creation
3. Accessible colors, fonts, and controls
4. Minimal manual data entry (OCR auto-fill)

Availability

1. The system should remain usable 95% of the time
2. Auto-reconnection for chat when the network resumes

Scalability

1. Support up to 10,000 user records initially
2. Storage scalable to thousands of images
3. Search optimized via indexing

Maintainability

1. Modular architecture
2. Clear separation of services and components
3. API documentation maintained

4.9 External Interface Requirements

User Interface Requirements

Main screens include:

a) Login / Registration

b) Home

c) Search page

d) Map view

e) Listing creator (with OCR)

f) Chat window

g) Profile and settings

h) Admin dashboard

Hardware Interface Requirements

a) A working device or handheld

b) Any camera for OCR captures

Software Interface Requirements

a) OCR engine

b) AI metadata extraction service

c) Geolocation & geocoding API

d) Real-time messaging service

e) Notification system

Communication Interface Requirements

a) REST APIs

b) WebSocket for chat

c) Push notifications via service worker

Chapter 5: Software Design Analysis

5.1 Use Case

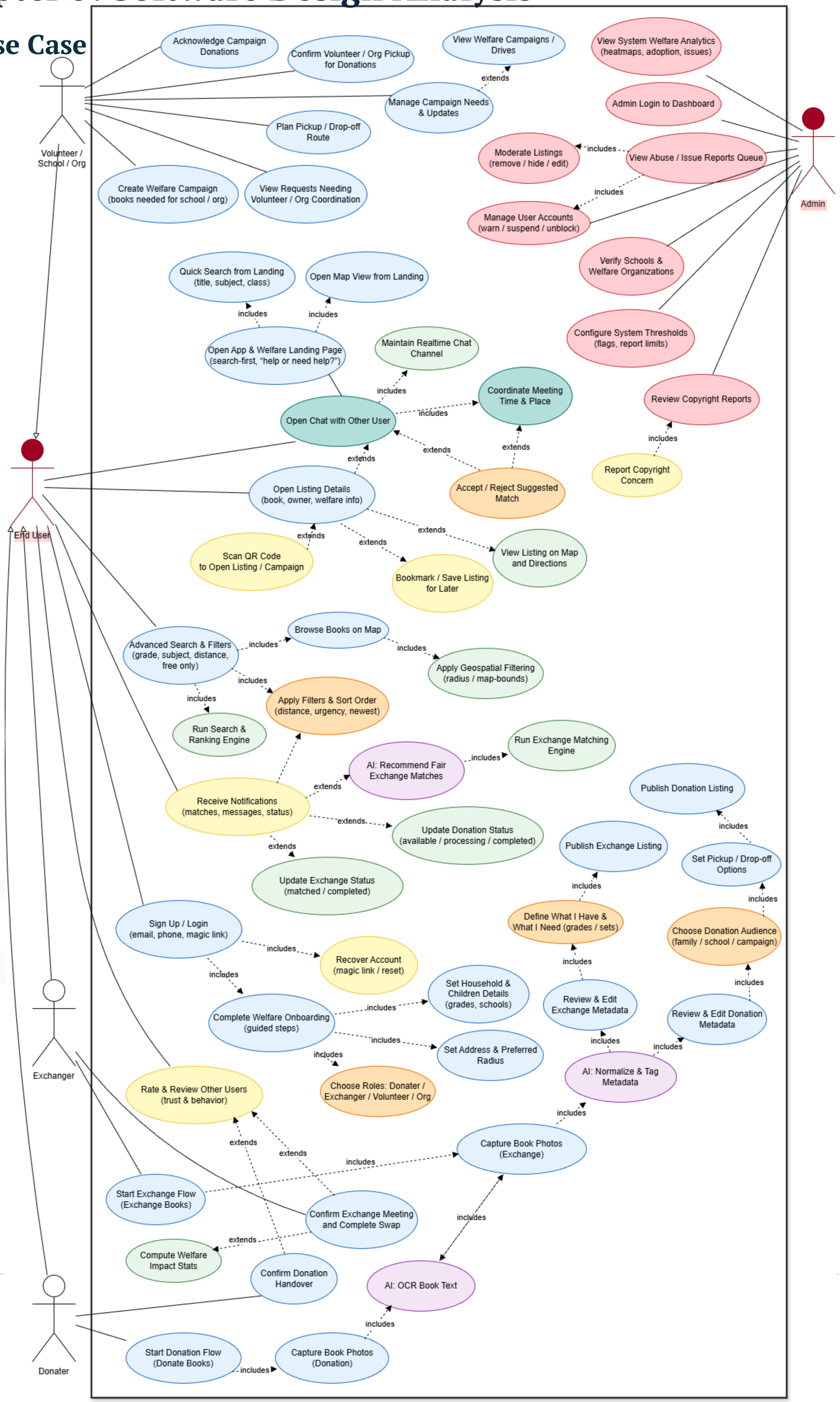

Volunteer / School / Org
Acknowledge Campaign Donations
Confirm Volunteer / Org Pickup for Donations
View Welfare Campaigns / Drives
extends
Manage Campaign Needs & Updates
Plan Pickup / Drop-off Route
Create Welfare Campaign (books needed for school / org)
View Requests Needing Volunteer / Org Coordination
View System Welfare Analytics (heatmaps, adoption, issues)
Admin Login to Dashboard
Moderate Listings (remove / hide / edit)
includes
View Abuse / Issue Reports Queue
includes
Manage User Accounts (warn / suspend / unblock)
Verify Schools & Welfare Organizations
Configure System Thresholds (flags, report limits)
Review Copyright Reports
includes
Report Copyright Concern
Admin
Quick Search from Landing (title, subject, class)
Open Map View from Landing
includes
includes
Open App & Welfare Landing Page (search-first, "help or need help?")
Maintain Realtime Chat Channel
includes
Coordinate Meeting Time & Place
includes
Open Chat with Other User
extends
extends
extends
Accept / Reject Suggested Match
End User
Open Listing Details (book, owner, welfare info)
extends
extends
extends
Scan QR Code to Open Listing / Campaign
Bookmark / Save Listing for Later
View Listing on Map and Directions
Advanced Search & Filters (grade, subject, distance, free only)
includes
Browse Books on Map
includes
Apply Geospatial Filtering (radius / map-bounds)
includes
Apply Filters & Sort Order (distance, urgency, newest)
includes
Run Search & Ranking Engine
AI: Recommend Fair Exchange Matches
includes
Run Exchange Matching Engine
extends
Receive Notifications (matches, messages, status)
extends
Update Donation Status (available / processing / completed)
extends
Update Exchange Status (matched / completed)
Publish Donation Listing
includes
Publish Exchange Listing
Set Pickup / Drop-off Options
includes
includes
Define What I Have & What I Need (grades / sets)
Choose Donation Audience (family / school / campaign)
Sign Up / Login (email, phone, magic link)
includes
Recover Account (magic link / reset)
includes
includes
includes
Complete Welfare Onboarding (guided steps)
includes
Set Household & Children Details (grades, schools)
includes
Set Address & Preferred Radius
includes
Choose Roles: Donater / Exchanger / Volunteer / Org
Review & Edit Exchange Metadata
Review & Edit Donation Metadata
includes
includes
AI: Normalize & Tag Metadata
Exchanger
Rate & Review Other Users (trust & behavior)
extends
extends
includes
includes
Capture Book Photos (Exchange)
Start Exchange Flow (Exchange Books)
Confirm Exchange Meeting and Complete Swap
includes
extends
Compute Welfare Impact Stats
Confirm Donation Handover
AI: OCR Book Text
includes
Donater
Start Donation Flow (Donate Books)
includes
Capture Book Photos (Donation)

Fig 5.1: This figure presents the Use Case Diagram of the DonoBook system, showing how different actors interact with the system's core functionalities at a high level.

Actor-Based Responsibilities

Parent / Student Actors

- Browse and search educational resources.
- Create listings for donation or exchange.
- Chat, negotiate, and coordinate exchanges.
- Complete donation or exchange processes and submit reviews.

Donor / Exchanger Actors

- Participate in donation and exchange workflows.
- Confirm handover and completion of transactions.
- Negotiate terms and finalize swaps.

Welfare Organization Actors

- Manage inventory and publish resource listings.
- Create and manage donation campaigns.
- Communicate with users and receive ratings.

Volunteer / Organization Actors

- Coordinate pickups and drop-offs.
- Manage campaign needs and updates.
- Track donations and assist with logistics.

Administrator Actors

- Verify schools and welfare organizations.
- Moderate listings and user accounts.
- Review abuse reports and system issues.
- Monitor analytics and configure system settings.

System Capabilities Highlighted

- Users can search, browse, and view educational resources.
- Donation and exchange workflows are initiated and managed within the system.
- Map-based browsing and geospatial filtering support localized discovery.
- Real-time chat enables negotiation and meetup coordination.
- AI assists with metadata normalization and fair exchange matching.
- Notifications inform users about messages, matches, and status updates.

5.2 Workflow Diagram

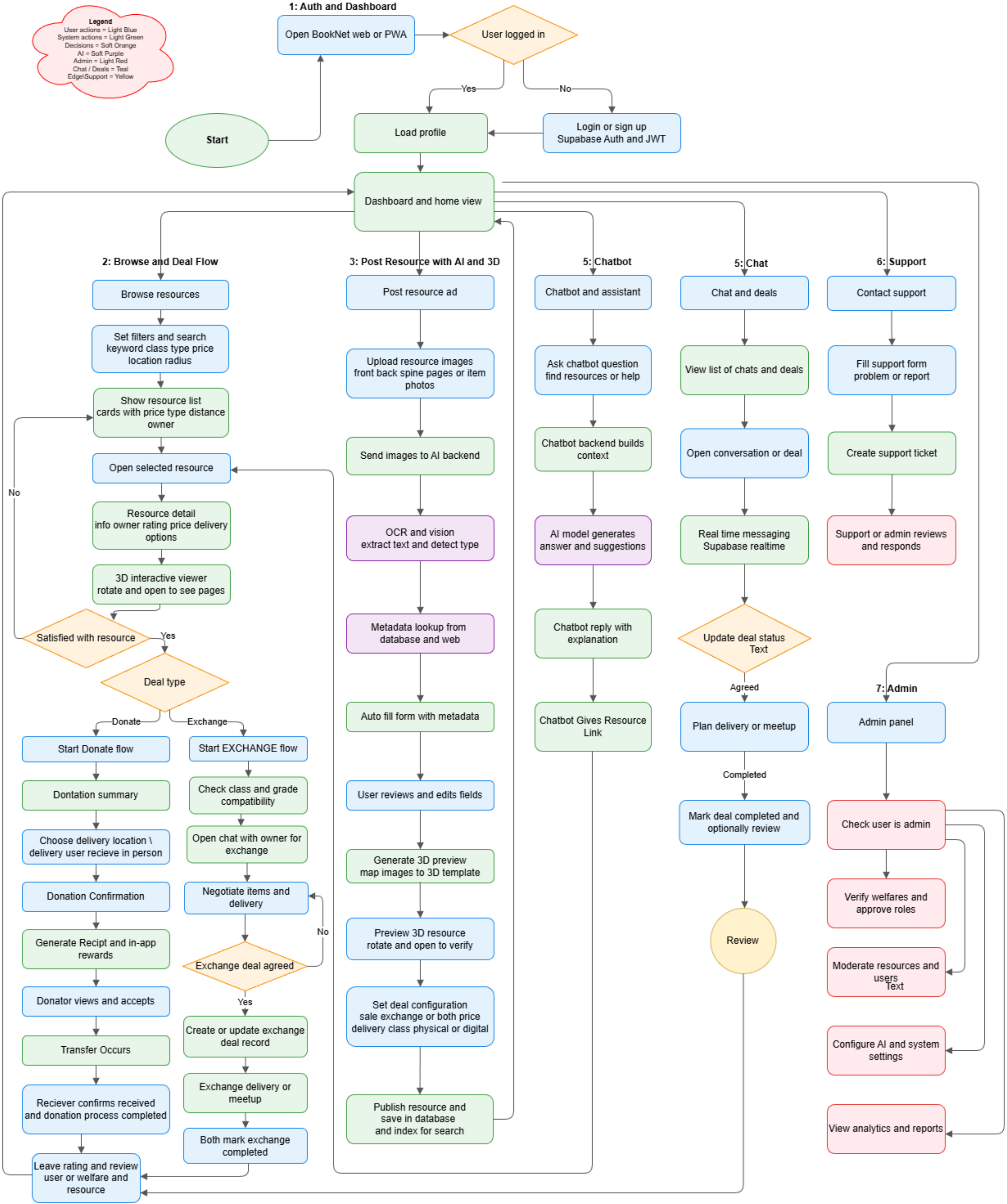


Fig 5.2: This figure depicts the end-to-end operational Entity Relationship Diagram (ERD), defining the core data structure of the DonoBook system and explains how users, resources, interactions, and AI-assisted components are connected.

- Users access DonoBook via a web browser or Progressive Web Application (PWA).
- Authentication is handled using Supabase Auth with JWT-based secure sessions.
- After successful login, the user profile and dashboard are loaded.
- Users can browse available resources using search and map-based discovery.
- Advanced filters support class, subject, condition, price, and distance-based search.
- Users may initiate a new listing for donation or exchange.
- Listing creation involves image upload and OCR-based text extraction.
- AI-assisted metadata normalization processes extracted information.
- Users review, edit, and publish listings to the searchable catalogue.
- Discovery workflows lead to detailed listing views.
- Real-time chat enables communication and negotiation between users.
- Chatbot assistance supports discovery, guidance, and user queries.
- Accepted deals progress through reservation and meetup or delivery coordination.
- Users complete donation or exchange through real-world interaction.
- Post-transaction steps include confirmation, rating, and review submission.
- Parallel workflows support donations, support requests, chatbot interactions, and administrative moderation.
- Admin and support flows handle verification, issue reporting, and system oversight.

5.3 Entity Relational Diagram

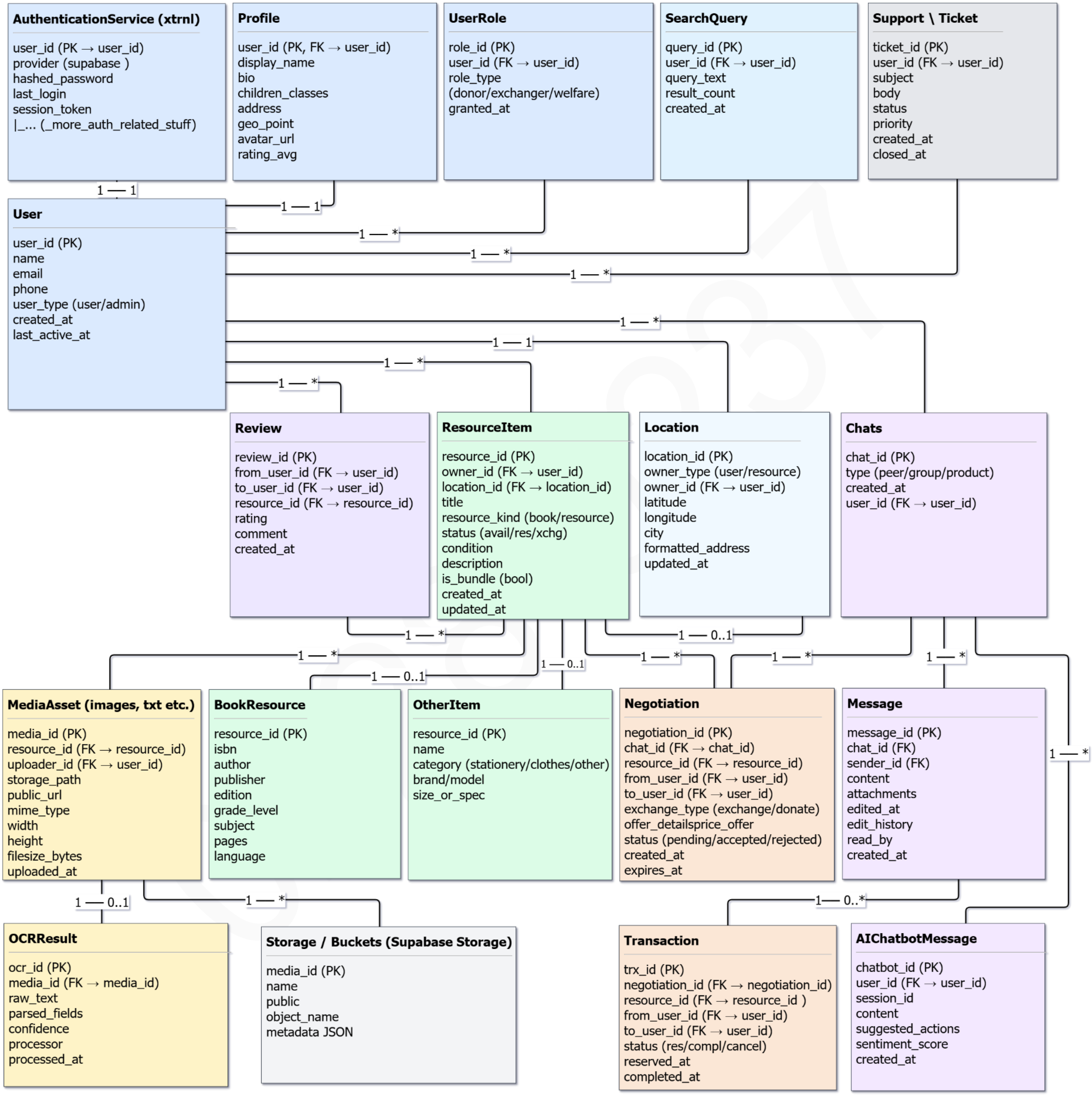


Fig 5.3: This figure depicts the end-to-end operational Entity Relationship Diagram (ERD), defining the core data structure of the DonoBook system and explains how users, resources, interactions, and AI-assisted components are connected.

User Core User Structure

- User is the central entity representing all system participants, including parents, donors, exchangers, welfare organizations, and administrators.
- AuthenticationService manages user authentication using Supabase services.
- Each user is linked to a Profile, which stores personal information, geographic location, and average ratings.
- UserRole defines the role of a user (donor, exchanger, welfare organization, or admin) and supports role-based access control.

Resource & Content Management

- ResourceItem represents all shared items in the system, including books and other educational resources.
- BookResource extends ResourceItem with academic metadata such as ISBN, subject, grade level, and language.
- OtherItem supports non-book educational resources.
- Each resource may contain multiple MediaAsset records, storing uploaded images or documents.
- OCRResult stores extracted and processed text from resource images to support AI-assisted metadata extraction.

Location & Discovery

- Location stores geographic coordinates and address data for map-based discovery and proximity filtering.
- SearchQuery tracks user search activity to support discovery optimization and analytics.

Communication & Interaction

- Chat and Message entities enable secure real-time communication between users.
- Negotiation manages exchange or donation offers, including offer details and deal status.
- Transaction records the completion of donation or exchange processes.

5.4 Presentation Layer (Client / PWA)

- React + Tailwind UI
- Map, listing forms, chat, OCR results
- QR code generation & scanning

5.5 Application Layer (Supabase + Edge Functions)

- Authentication
- Database operations
- Storage (images, PDFs)
- Realtime messaging
- AI / OCR pipelines (Serverless)

5.6 External Services Layer

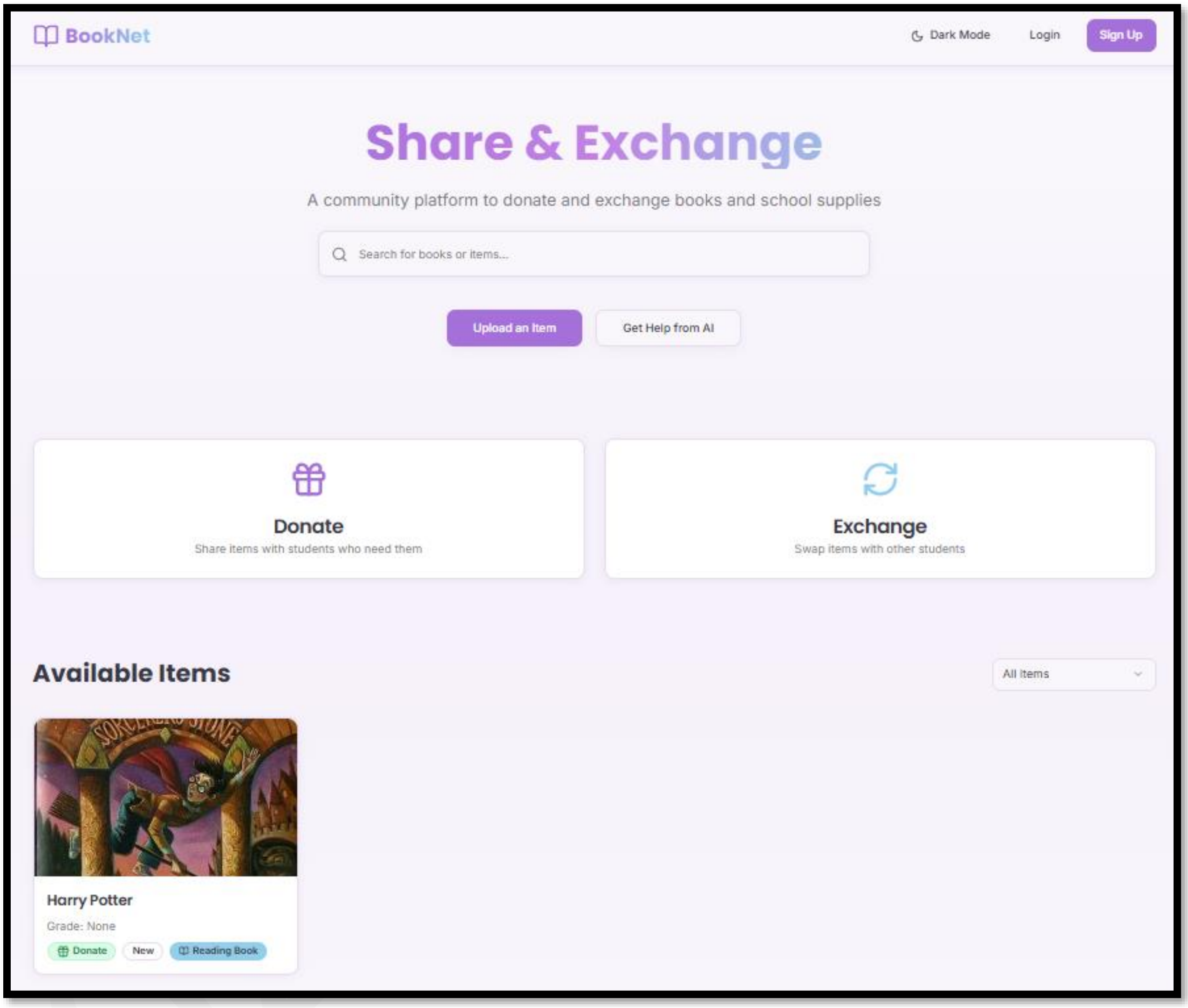
- OCR API (Tesseract/Cloud OCR)
- LLM API (Metadata extraction, Recommendations, Negotiation assistance)
- Map tile servers (OpenStreetMap)

This layered architecture separates concerns between presentation, application logic, and external services. Such separation improves maintainability, enables independent scaling of components, and reduces coupling between user interfaces and backend services. The use of managed cloud services further minimizes operational overhead and accelerates development.

Chapter 6: Prototype

DESKTOP

Landing Page:

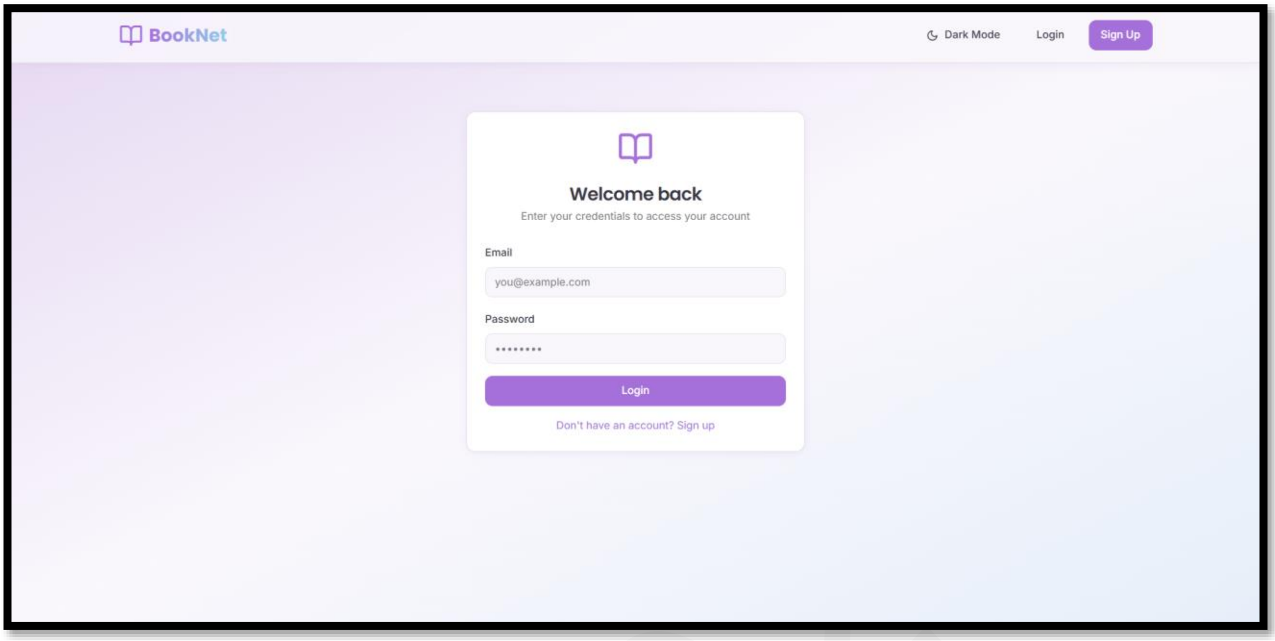


A community platform to donate and exchange books and school supplies

Share items with students who need them

Swap items with other students

Login:



Welcome back

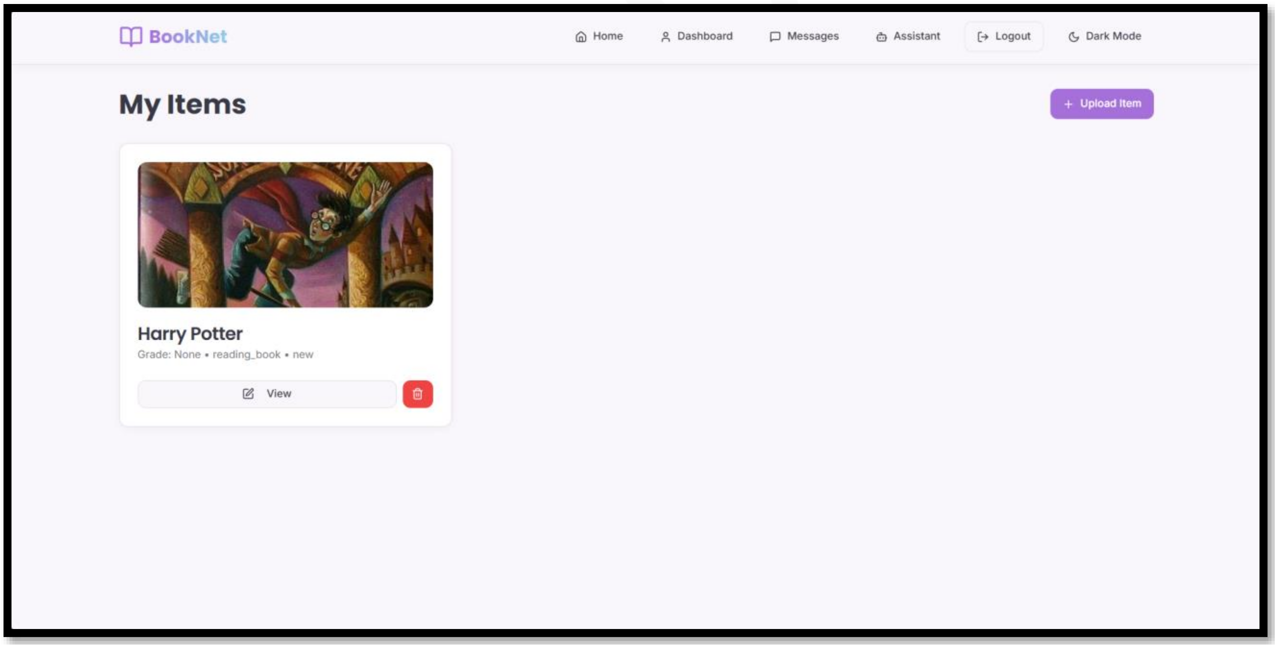
Enter your credentials to access your account

Email: you@example.com

Password: ••••••••

Don't have an account? Sign up

Dashboard:

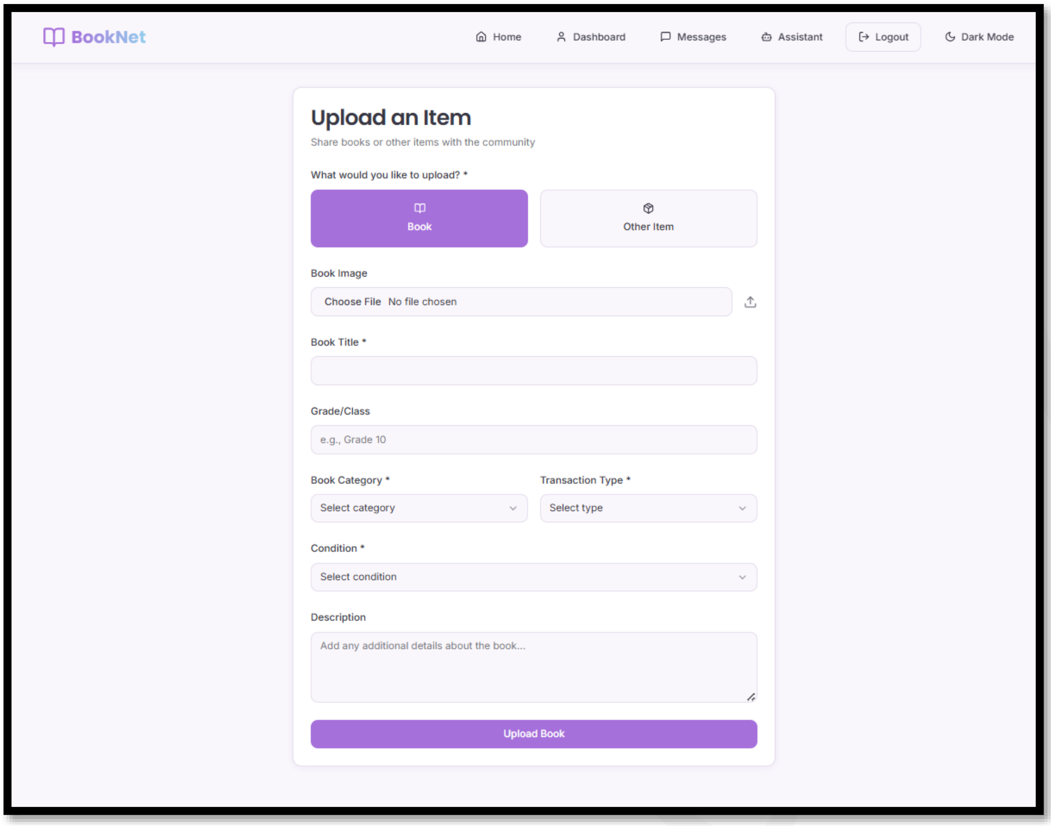


My Items

Harry Potter

Grade: None • reading_book • new

Upload \ Donate:



Upload an Item

Share books or other items with the community

What would you like to upload? *

Book | Other Item

Book Image

Choose File No file chosen

Book Title *

Grade/Class

e.g., Grade 10

Book Category *

Select category

Transaction Type *

Select type

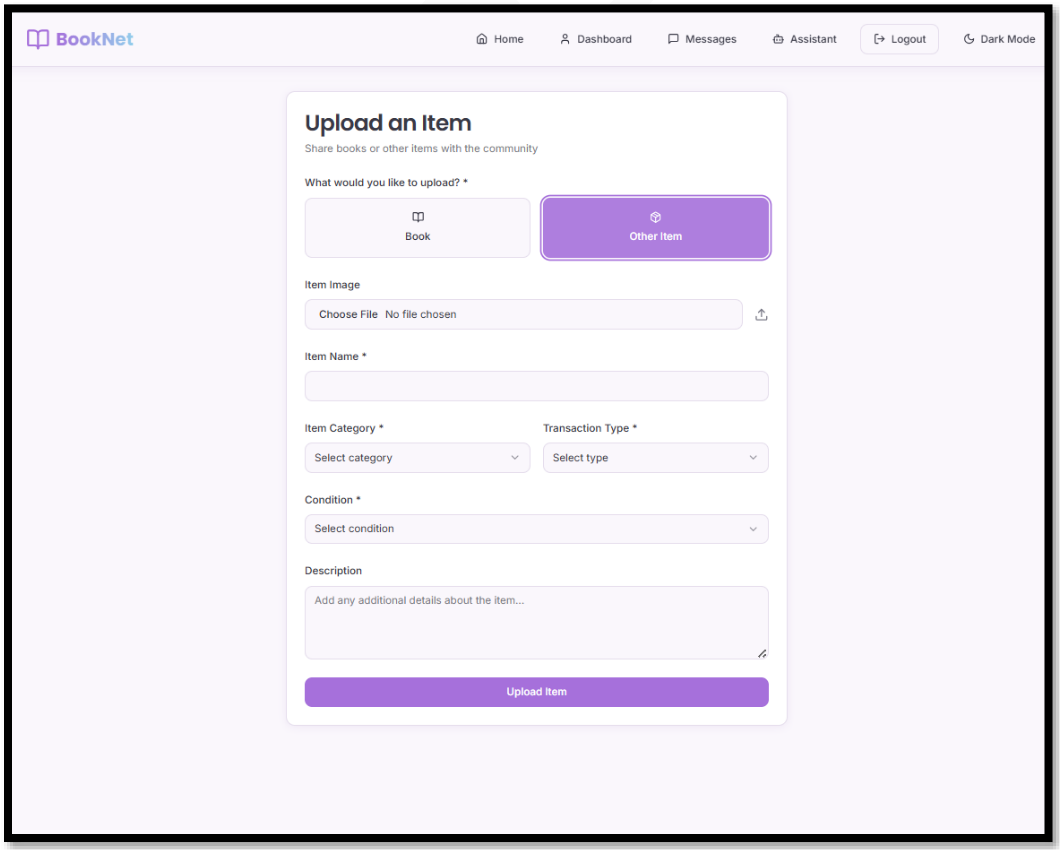
Condition *

Select condition

Description

Add any additional details about the book...

Upload Book



Upload an Item

Share books or other items with the community

What would you like to upload? *

Book | Other Item

Item Image

Choose File No file chosen

Item Name *

Item Category *

Select category

Transaction Type *

Select type

Condition *

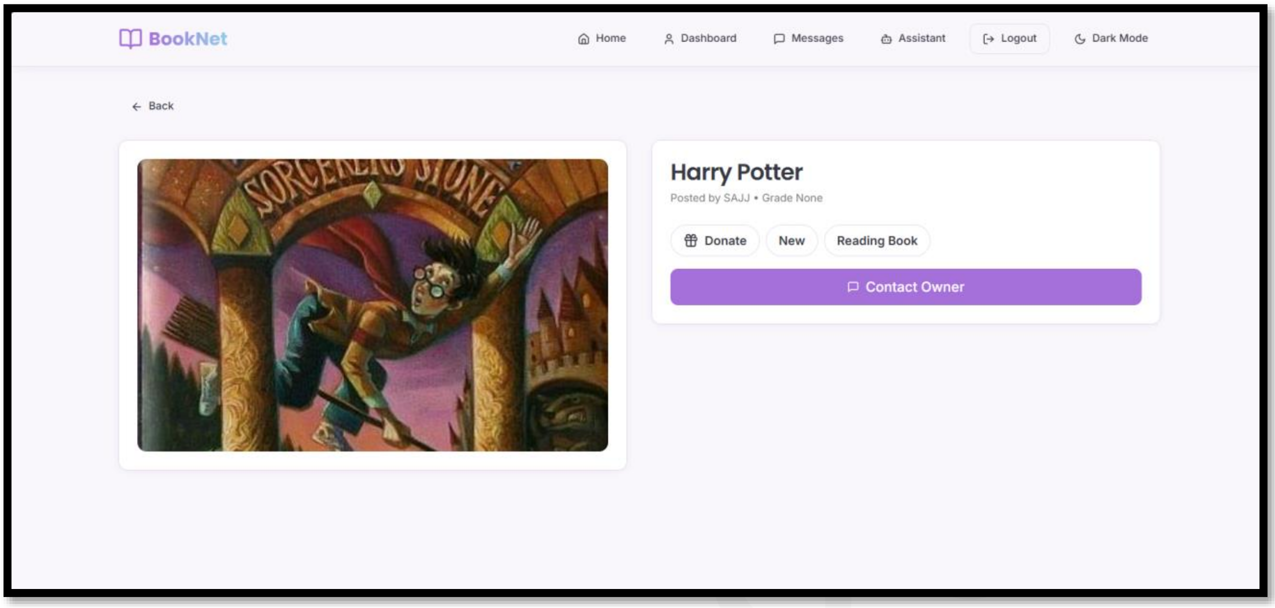
Select condition

Description

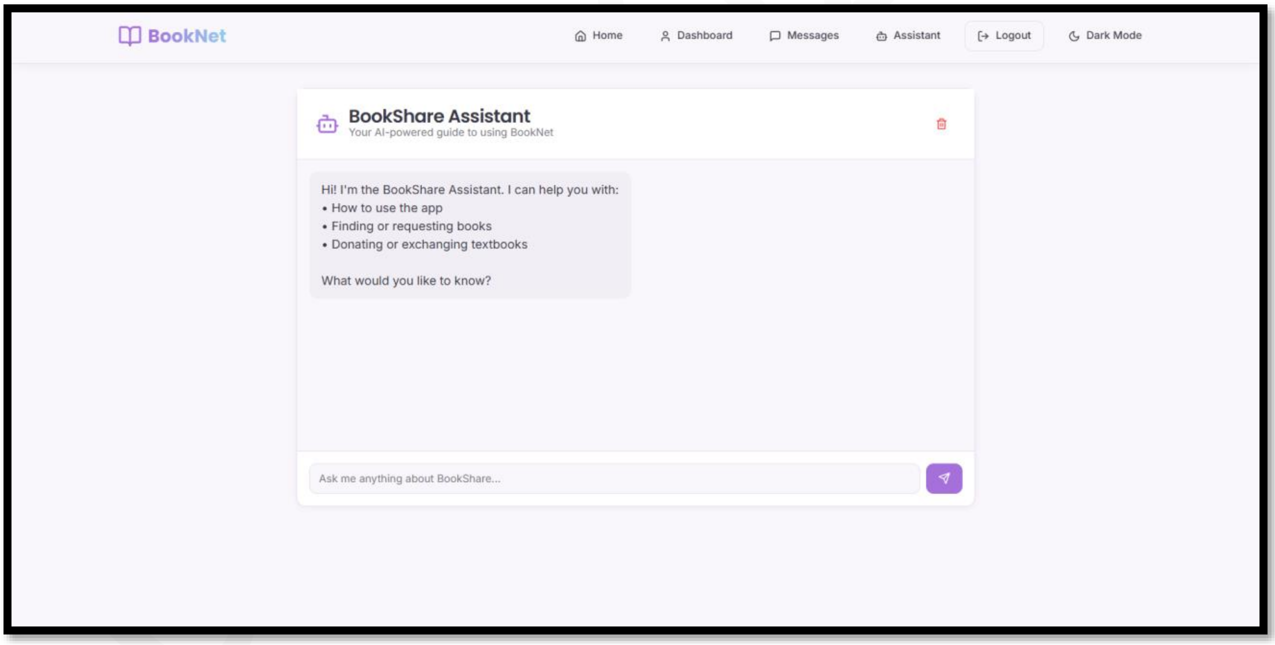
Add any additional details about the item...

Upload Item

View Product:



Chatbot:


BookNet
Home
Dashboard
Messages
Assistant
Logout
Dark Mode
BookShare Assistant
Your AI-powered guide to using BookNet

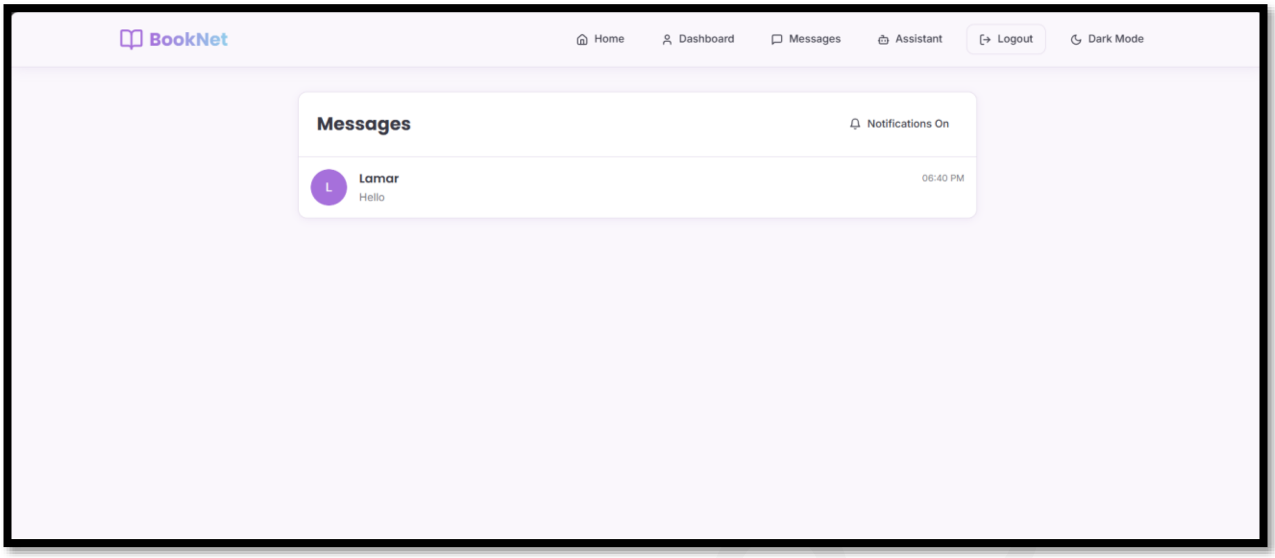
Hi! I'm the BookShare Assistant. I can help you with:

- How to use the app
- Finding or requesting books
- Donating or exchanging textbooks

What would you like to know?

Ask me anything about BookShare...

Messages:

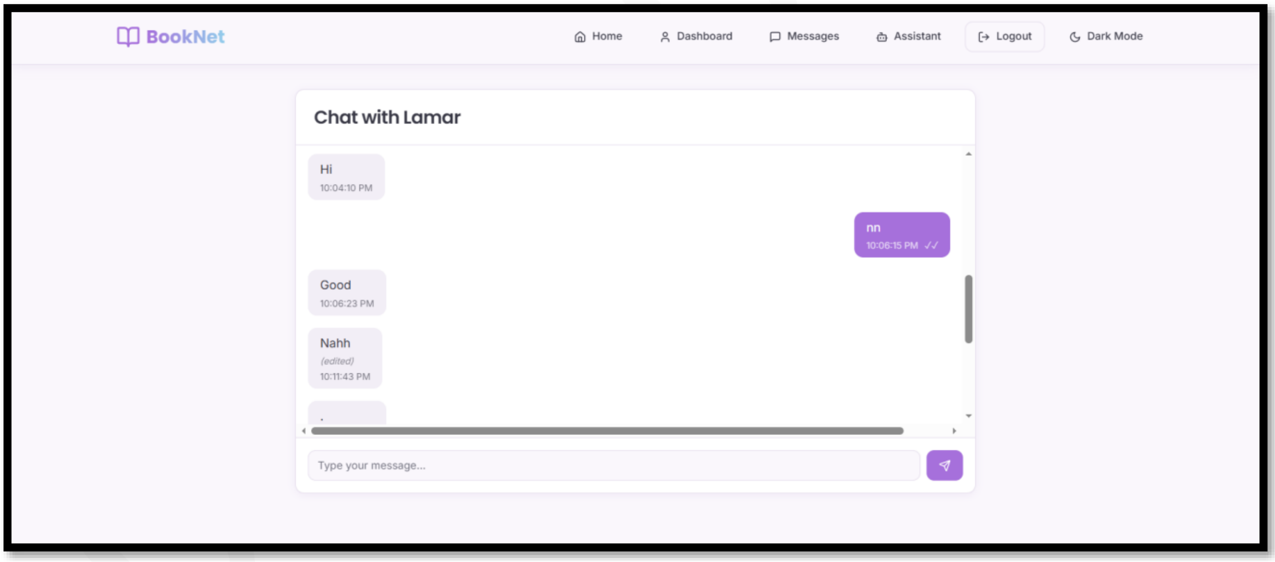


Messages

Notifications On

L Lamar
Hello
06:40 PM

Chat:



Chat with Lamar

Hi
10:04:10 PM

nn
10:06:15 PM ✓✓

Good
10:06:23 PM

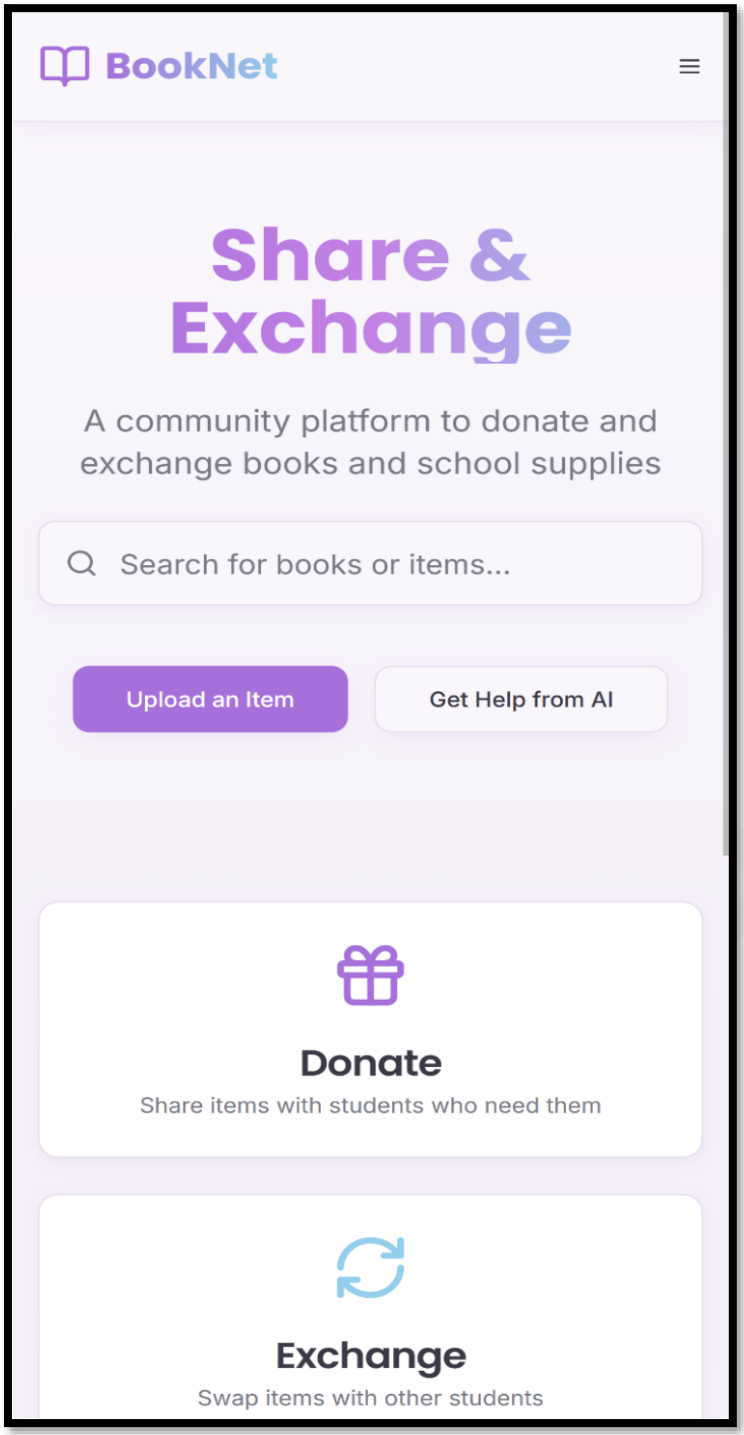
Nahh
(edited)
10:11:43 PM

.

Type your message...

MOBILE

Landing Page:



Share & Exchange

A community platform to donate and exchange books and school supplies

Search for books or items...

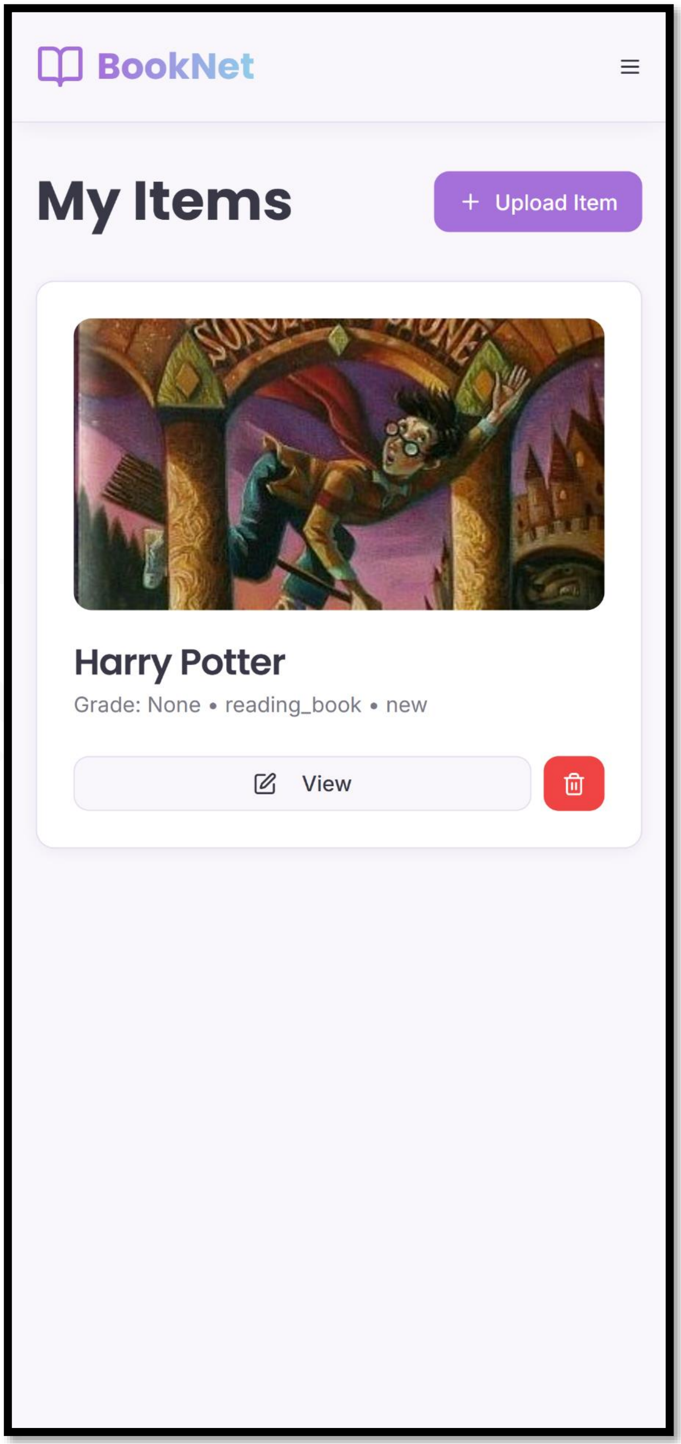
Donate

Share items with students who need them

Exchange

Swap items with other students

Dashboard:

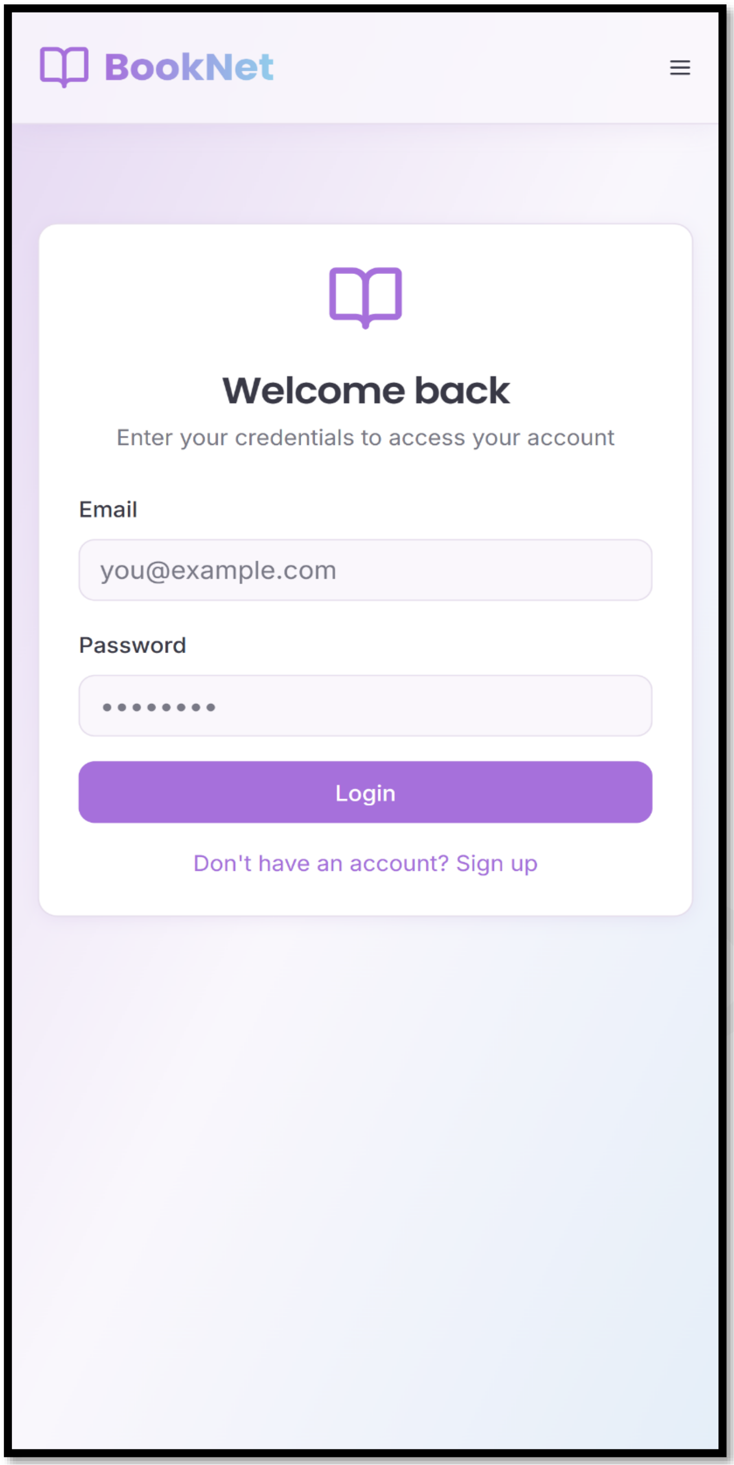


My Items

Harry Potter

Grade: None • reading_book • new

Login \ Signup:



BookNet

Welcome back

Enter your credentials to access your account

Email

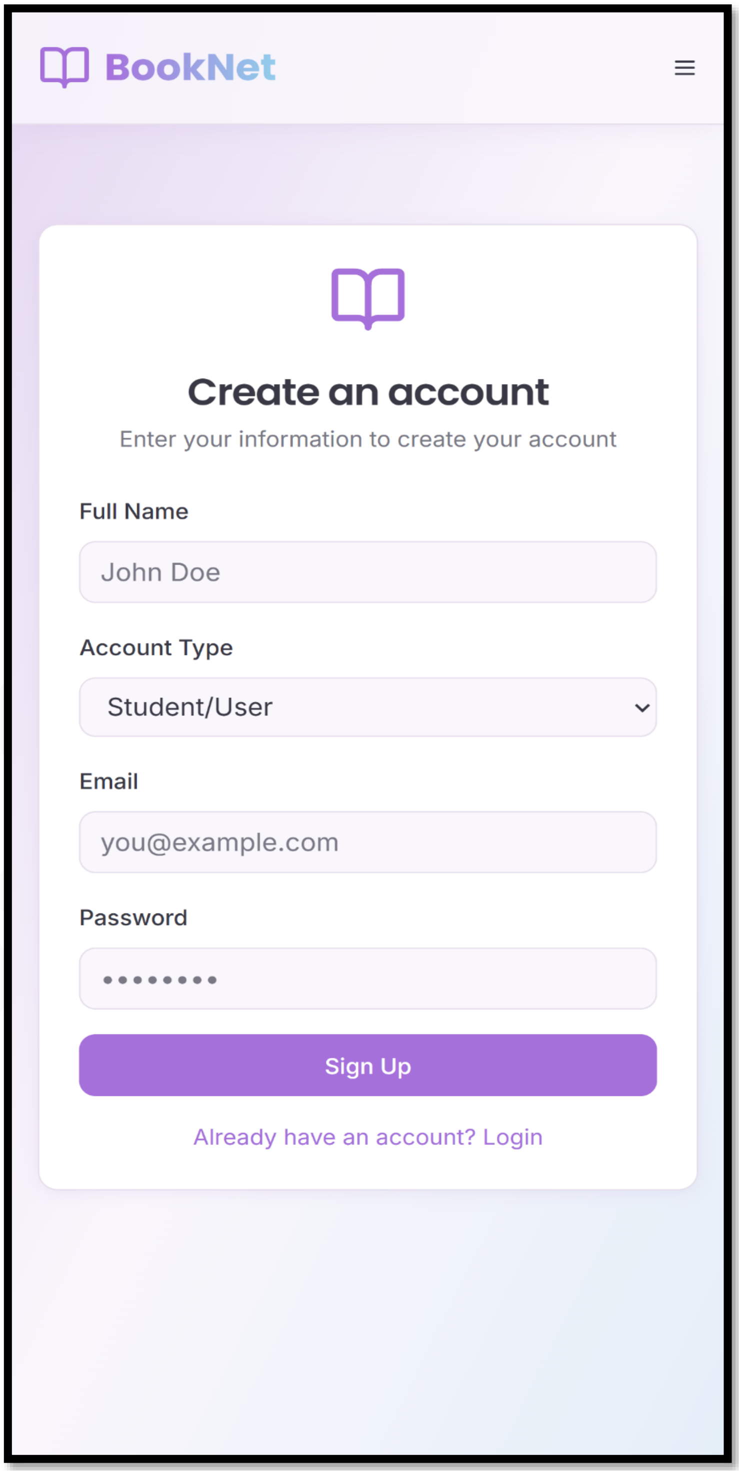
you@example.com

Password

••••••••

Login

Don't have an account? Sign up



BookNet

Create an account

Enter your information to create your account

Full Name

John Doe

Account Type

Student/User

Email

you@example.com

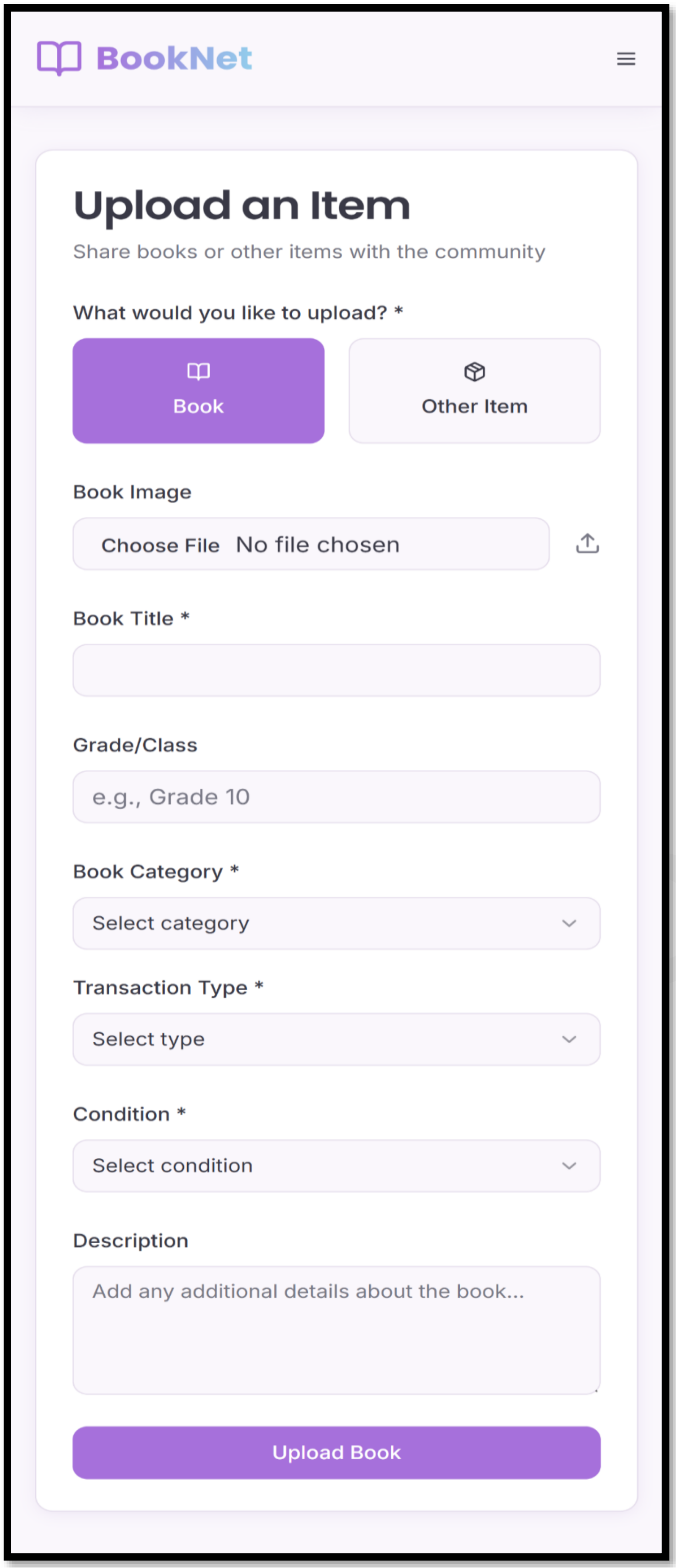
Password

••••••••

Sign Up

Already have an account? Login

Upload \ Donate:



BookNet

Upload an Item

Share books or other items with the community

What would you like to upload? *

Book

Other Item

Book Image

Choose File No file chosen

Book Title *

Grade/Class

e.g., Grade 10

Book Category *

Select category

Transaction Type *

Select type

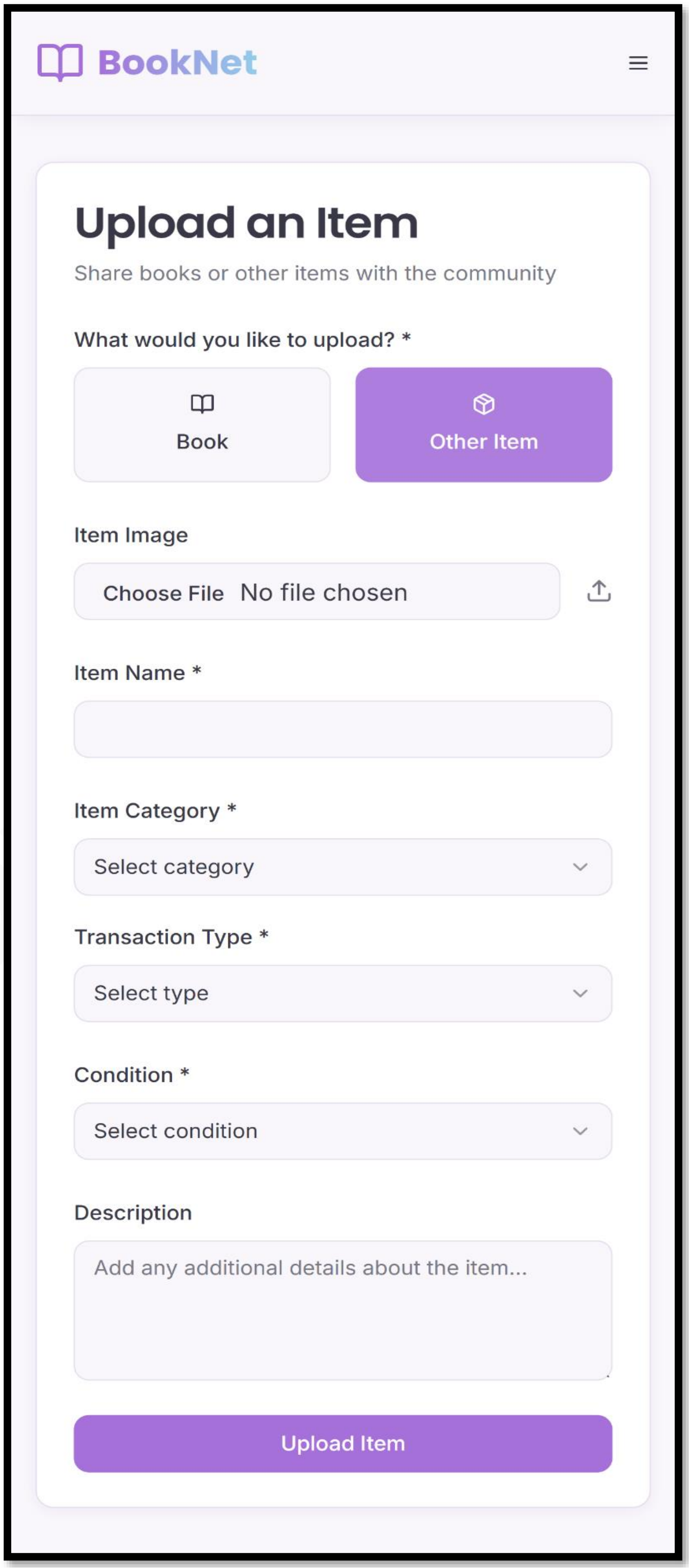
Condition *

Select condition

Description

Add any additional details about the book...

Upload Book



BookNet

Upload an Item

Share books or other items with the community

What would you like to upload? *

Book

Other Item

Item Image

Choose File No file chosen

Item Name *

Item Category *

Select category

Transaction Type *

Select type

Condition *

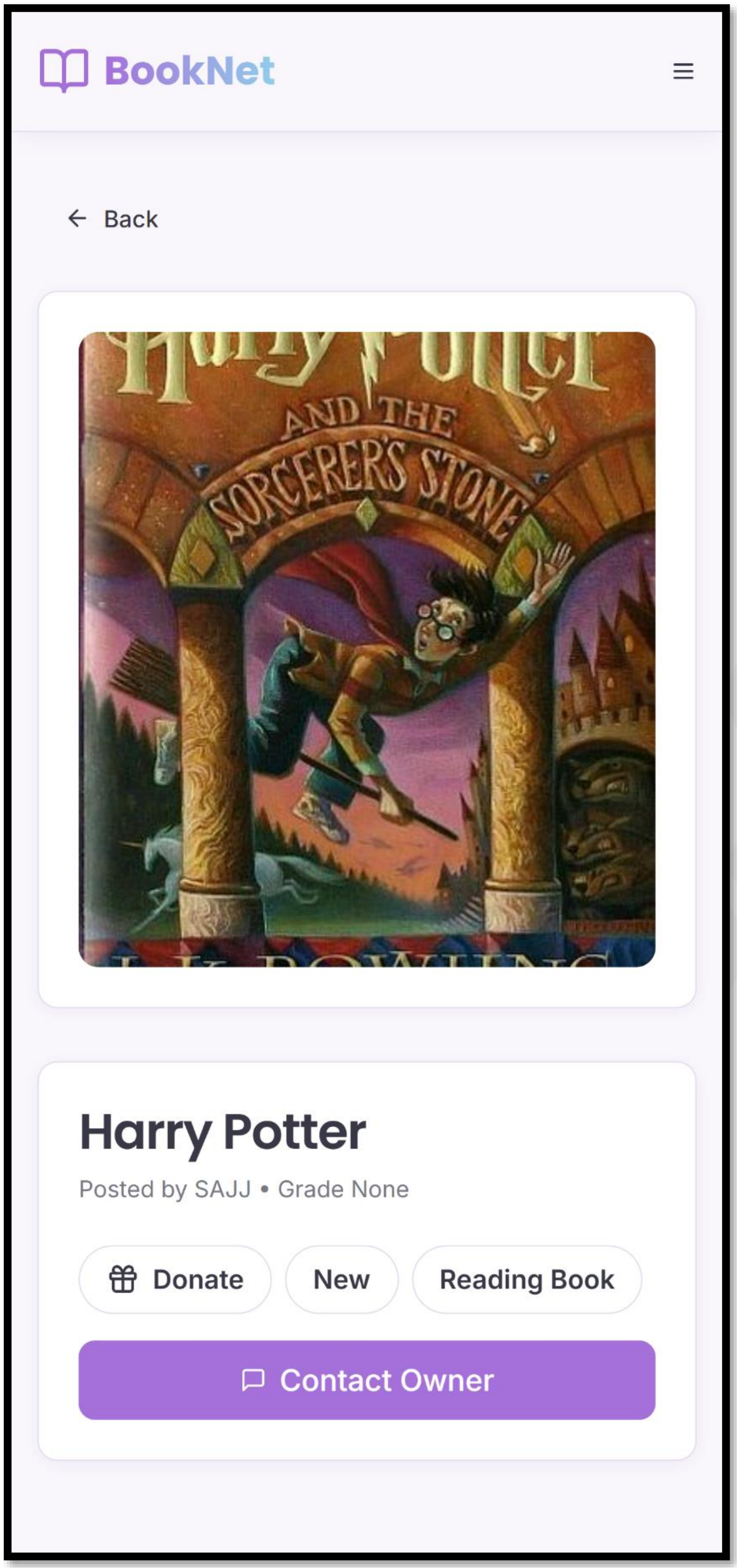
Select condition

Description

Add any additional details about the item...

Upload Item

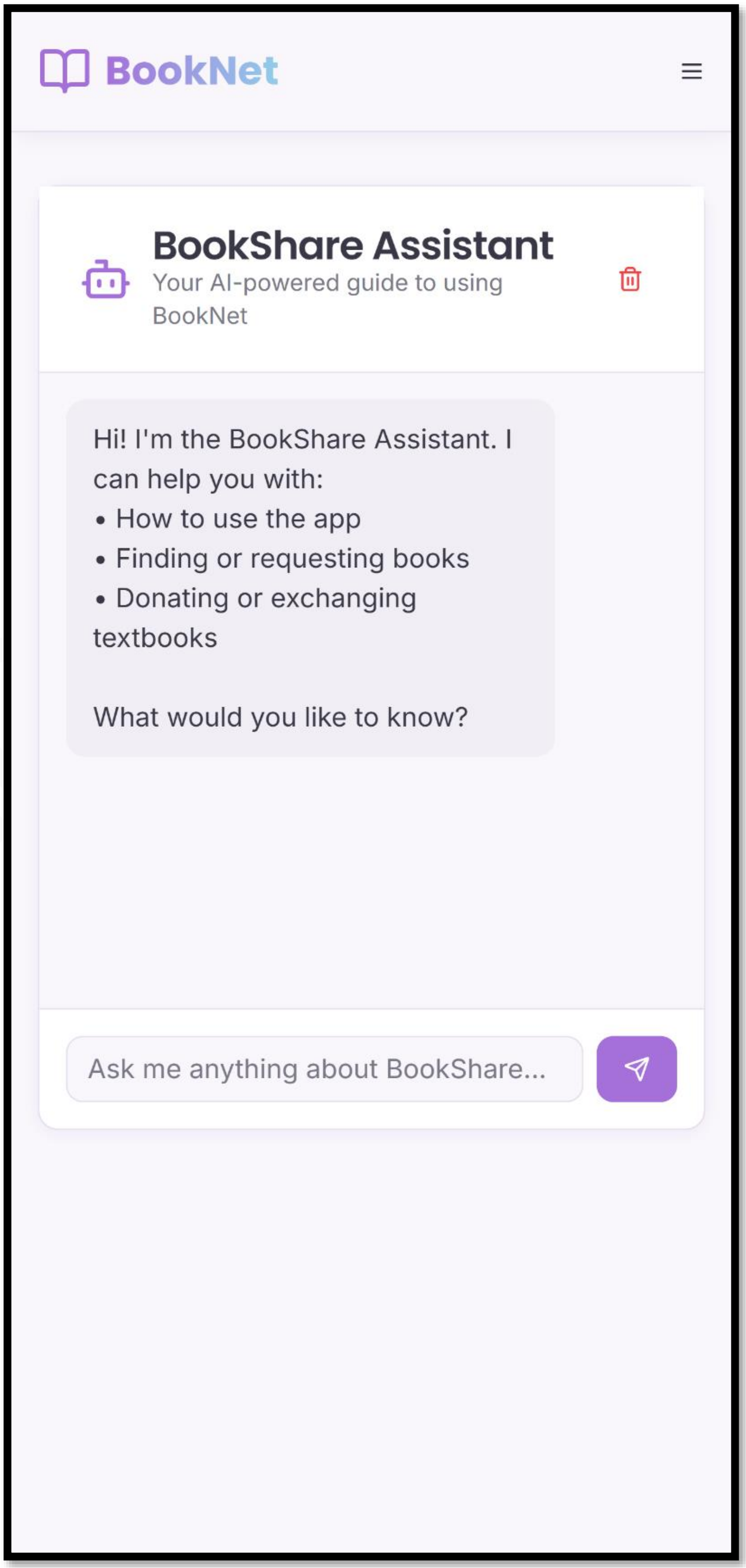
View Product:



Harry Potter

Posted by SAJJ • Grade None

Chatbot:



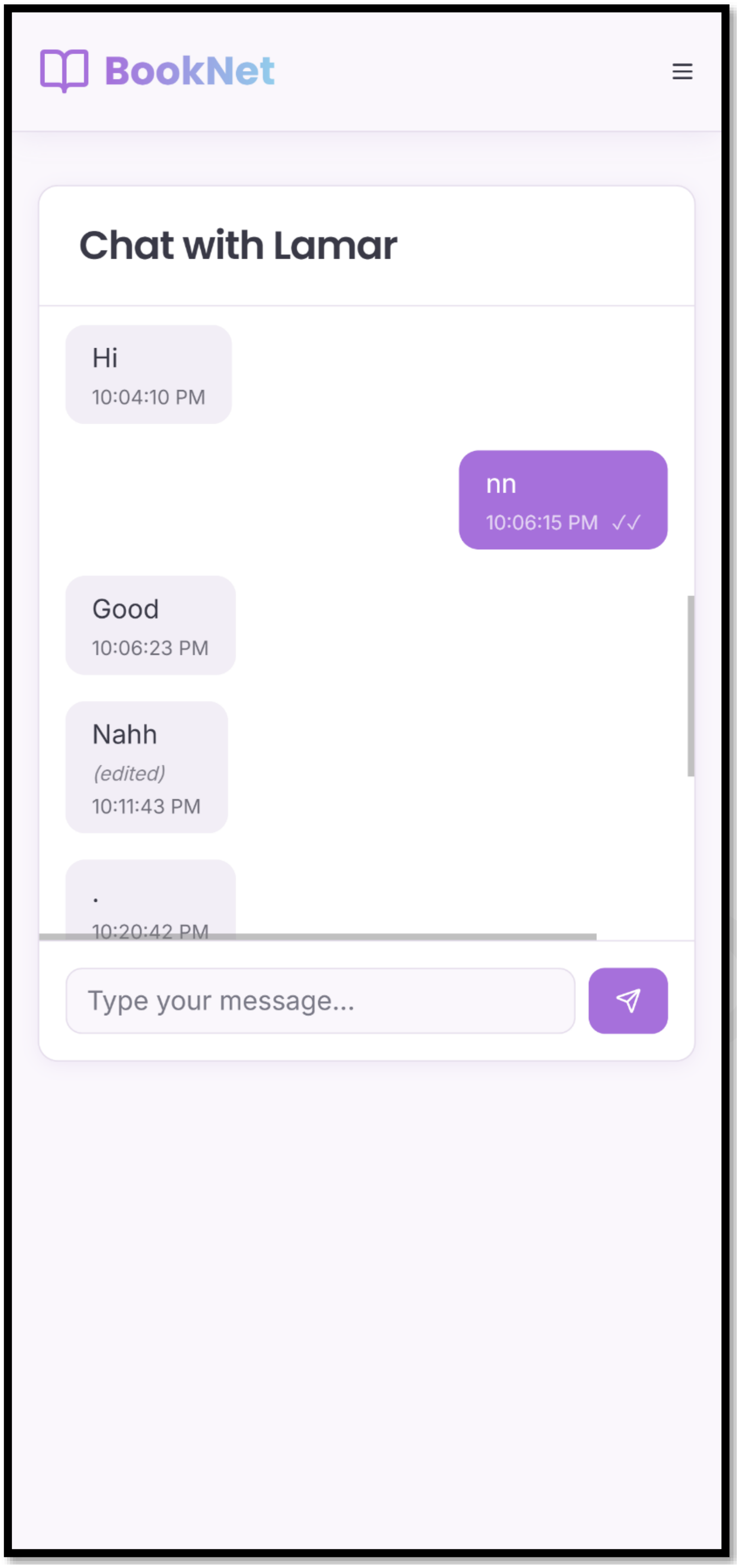
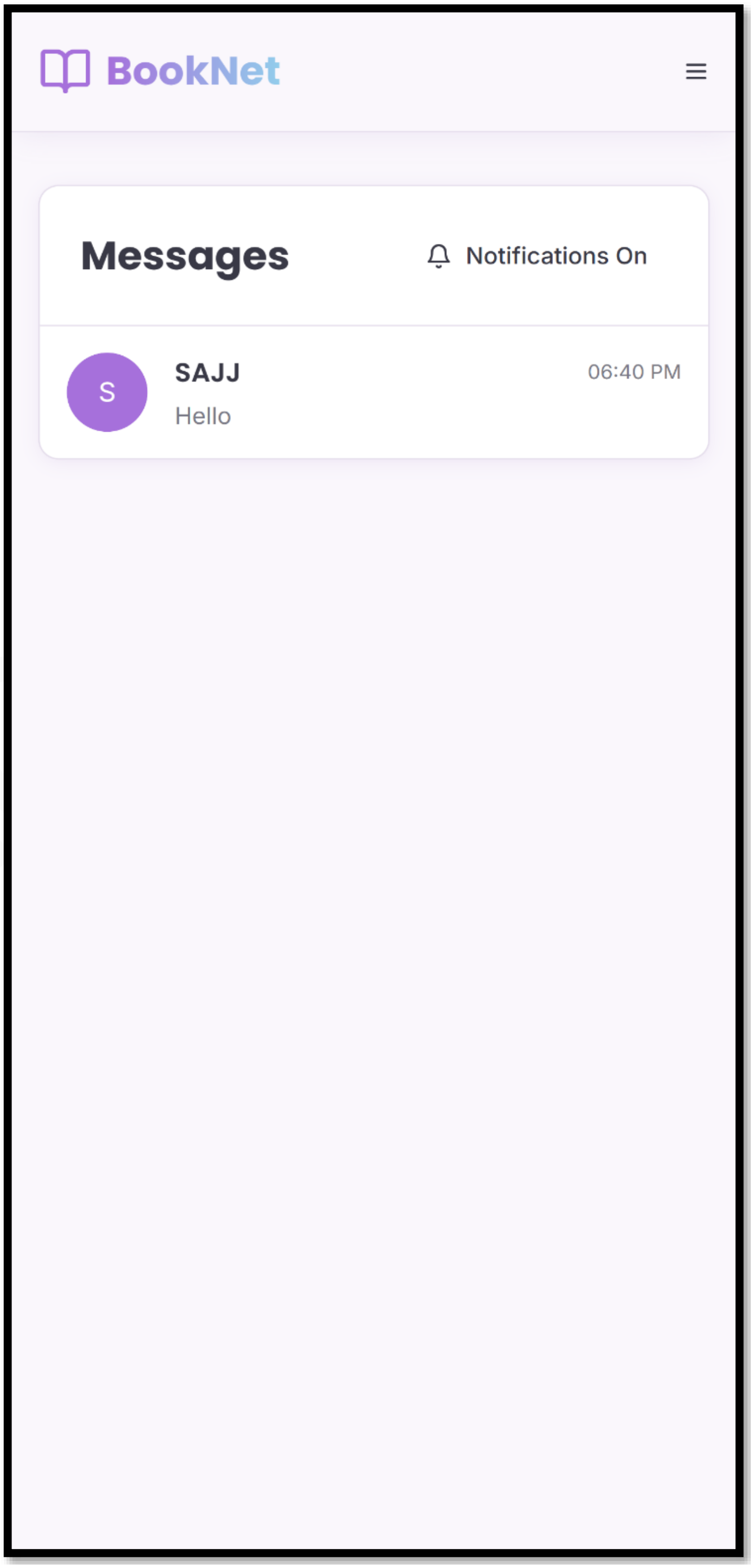
BookShare Assistant

Your AI-powered guide to using BookNet

Hi! I'm the BookShare Assistant. I can help you with:
- How to use the app
- Finding or requesting books
- Donating or exchanging textbooks

What would you like to know?

Ask me anything about BookShare...

Chat:	**Messages:**
 Chat with Lamar Hi 10:04:10 PM nn 10:06:15 PM ✓✓ Good 10:06:23 PM Nahh *(edited)* 10:11:43 PM . 10:20:42 PM Type your message...	 Messages Notifications On S SAJJ 06:40 PM Hello

Chapter 7: Evaluation & Conclusion

7.1 Evaluation & Success Criteria

- **Functional Evaluation**
  The system will be considered functionally successful if users can create listings with OCR-assisted metadata extraction, discover relevant resources within defined geographic bounds, engage in real-time chat, and receive meaningful AI-based recommendations.

- **Performance Evaluation**
  System responsiveness will be evaluated based on search latency, with a target of under 500 milliseconds for typical queries during pilot testing. OCR and AI-assisted processes should complete within acceptable timeframes to maintain a smooth user experience.

- **Usability Evaluation**
  Usability will be assessed using the System Usability Scale (SUS), with a target score of above 75, indicating good overall user satisfaction.

- **Accuracy Evaluation**
  The accuracy of OCR and AI-based metadata extraction will be measured by comparing extracted fields against ground truth data, with a target accuracy under typical usage conditions.

7.2 Conclusion

This proposal introduced DonoBook, a Progressive Web Application designed to facilitate the local sharing and exchange of educational resources. It outlined the problem of rising textbook costs and inefficient access to second-hand materials, and proposed a practical, technology-driven solution integrating modern web architecture, geospatial discovery, and AI-assisted automation.

The defined objectives, scope, and high-level design demonstrate that the project is feasible, relevant, and suitable as a Final Year Project. This proposal therefore, provides a clear foundation for proceeding to detailed requirement analysis, system design, and implementation.

Chapter 8: References & Documentations

Salunke, V., & Patil, S. (2023). Online book exchange system with chatbot "Bookish." *Proceedings of the International Conference on Artificial Intelligence and Education*, *1*(1), 27–34. https://www.researchgate.net/publication/376232510_Online_Book_Exchange_System_with_Chatbot_Bookish

Salunke, V., & Deshmukh, S. (2023). Design and development of a digital platform for peer-to-peer book exchange. *International Journal of Educational Technology in Higher Education*, *20*(1), Article 14. https://www.researchgate.net/publication/388681274_Design_and_Development_of_a_Digital_Platform_for_Peer_-to_-Peer_Book_Exchange

Siafis, V., Rangoussi, M., & Psaromiligkos, Y. (2024). Recommender systems for teachers: A systematic literature review of recent research (2011–2023). *Education Sciences*, *14*(7), 723. https://doi.org/10.3390/educsci14070723

Urdaneta-Ponte, M. C., Mendez-Zorrilla, A., & Oleagordia-Ruiz, I. (2021). Recommendation systems for education: A systematic review. *Electronics*, *10*(14), 1611. https://doi.org/10.3390/electronics10141611

Towards the adoption of recommender systems in online education: A framework and implementation. (2025). *Computers*, *9*(10), 259. https://www.mdpi.com/2504-2289/9/10/259

Design of a conversational recommender system in education. (2024). *User Modeling and User-Adapted Interaction*. https://doi.org/10.1007/s11257-024-09397-y

Jannach, D., Jugovac, M., & Lerche, L. (2023). Evaluating conversational recommender systems. *Artificial Intelligence Review*, *56*, 3271–3315. https://doi.org/10.1007/s10462-022-10229-x

Baek, J., Kim, G., Lee, J., Park, S., Han, D., Yun, S., Oh, S. J., & Lee, H. (2019). What is wrong with scene text recognition model comparisons? Dataset and model analysis. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. https://openaccess.thecvf.com/content_ICCV_2019/html/Baek_What_Is_Wrong_With_Scene_Text_Recognition_Model_Comparisons_Dataset_ICCV_2019_paper.html

Marth, M., Füller, J., & Hutter, K. (2022). Reducing perceived risk in peer-to-peer platforms. *Psychology & Marketing*, *39*(4), 723–739. https://doi.org/10.1002/cb.2075

Huang, Y., Lv, T., Cui, L., Lu, Y., & Wei, F. (2022). LayoutLMv3: Pre-training for document AI with unified text and image masking. *Proceedings of the ACM International Conference on Multimedia*.
https://arxiv.org/abs/2204.08387

Trust, privacy, and safety factors in peer-to-peer marketplaces. (2024). *Proceedings of the ACM Conference on Human Factors in Computing Systems*.
https://doi.org/10.1145/3613904.3641966

Ma, X., Mezghani, L., Wilber, K., Hong, H., Piramuthu, R., Naaman, M., & Belongie, S. (2019). Understanding image quality and trust in peer-to-peer marketplaces. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)*.
https://doi.org/10.1109/WACV.2019.00060

Moderators of reputation effects in peer-to-peer online markets: A meta-analytic model selection approach. (2022). *Journal of Computational Social Science*, *5*, 1063–1089.
https://doi.org/10.1007/s42001-022-00160-0

Buyers' trust and mistrust in e-commerce platforms: A synthesizing literature review. (2021). *Information Systems and E-Business Management*, *19*, 1115–1147.
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8581124/

Campus second-hand textbook trading platform based on Vue 3 and Spring Boot. (2024). *International Core Journal of Engineering*.
https://www.airitilibrary.com/Article/Detail/P20190813001-N202406040016-00007

Kim, G., Hong, T., Yim, M., Nam, J., Park, J., Yoon, S., & Han, J. (2022). Donut: OCR-free document understanding transformer. In *Proceedings of the European Conference on Computer Vision (ECCV)*.
https://www.ecva.net/papers/eccv_2022/papers_ECCV/papers/136880493.pdf
https://github.com/clovaai/donut

Leaflet. (2025). *Leaflet: An open-source JavaScript library for interactive maps*.
https://leafletjs.com

Supabase. (2025). *Authentication and JWT documentation*.
https://supabase.com/docs/guides/auth

pgvector. (2024). *pgvector: Vector similarity search for PostgreSQL*.
https://github.com/pgvector/pgvector

Google Developers. (2025). *Progressive web apps (PWA): Guides and case studies*.
https://web.dev/explore/progressive-web-apps

Chapter 9: Appendixes

APPENDIX A: Feasibility Analysis

1. Technical Feasibility

Technical Feasibility covers our tech stack and what our project will bring to the table.

Technology Stack

Component	**Chosen Technology**	**Feasibility Verdict**
Frontend PWA	Vite + React + Tailwind	Fully feasible, modern, lightweight
Backend	Supabase (Auth, DB, Storage, Realtime, Edge Functions)	Fully feasible; removes backend maintenance workload
OCR	Tesseract or Cloud OCR + Edge Function	Feasible via a serverless function, accuracy is manageable
LLM Metadata Extraction	Hosted LLM API via Edge Function	Feasible; low latency with processed text
AI Chatbot	LLM-based assistant via API	Feasible; simple prompt engineering + safety guardrails
Recommendation System	Chatbot-based suggestions + embeddings	Feasible
Geospatial	Leaflet + OSM + Nominatim	Fully feasible; must implement caching

Search Bound to Map	Postgres + PostGIS/earthdistance	Feasible and efficient for radius search
Realtime Chat	Supabase Realtime	Native feature; trivial to integrate
QR Flow	Edge Function signing + QR generator	Simple to implement
Admin Dashboard	Simple React + Supabase Policy Management	Feasible
Deployment	Supabase hosting + Vercel	Straightforward

The selected technology stack consists primarily of widely adopted, well-documented, and production-ready tools. By relying on Supabase as a Backend-as-a-Service, the project avoids the complexity of managing custom servers while still supporting advanced features such as authentication, real-time communication, and serverless functions. The use of established OCR and LLM APIs further enhances feasibility by minimizing the need for custom machine learning model development.

Core Feasibility Notes

- **Supabase eliminates the need for a custom Express backend**, significantly increasing feasibility.
- **OCR accuracy may fluctuate**, but LLM post-processing stabilizes metadata extraction.
- **Real-time chat infrastructure is already provided**, reducing development time.
- **Geocoding via Nominatim must be cached** to avoid rate limits.
  Recommendation engine complexity is minimized by using chatbot-based suggestions instead of heavy ML.

Technical Risks

- OCR accuracy drops on low-quality images.
- Nominatim free tier rate limitations.
- RLS misconfiguration could break critical flows.
- Heavy AI usage may cause latency if not cached and properly handled.

2. Operational Feasibility

End-User Feasibility

The system is designed for parents, students, and verified welfare organizations.
Operational usage is simple:

1. Open the PWA
2. Browse the map or search
3. Create a listing with auto-filled OCR
4. Chat/Negotiate
5. Pick-up/Exchange/Lend

Because DonoBook targets non-technical users, the OCR auto-fill and map-centric UI reduce friction.

The operational workflow is intentionally designed to minimize cognitive and technical burden on end users, many of whom may not be technologically proficient. Automated metadata extraction, map-based browsing, and guided interaction flows significantly reduce user effort, making the system operationally feasible for parents, students, and welfare organizations.

Maintainability

Operation is feasible long-term due to:

- Supabase manages database, auth, storage, and real-time
- PWA eliminates mobile app publishing and update overhead
- Easy update cycle via Vite/React rebuilds

Data Management

Supabase handles:

- User access control via RLS
- Secure storage for images and documents
- Row-level permissions for listings, messages, transactions

3. Economic Feasibility

DonoBook is financially feasible for an FYP.

Cloud / Platform / API Costs

Resource / Service	Estimated Monthly Cost	Notes
Supabase Pro Plan	**8,000 – 12,000 PKR**	Enough for serious usage, logs, compute, RLS
Extra Supabase Storage (20–40GB)	**1,500 – 3,000 PKR**	For images/document uploads
pgvector / PostGIS compute	**1,000 – 3,000 PKR**	Reduced overhead (moderate usage)
OCR API (Google Vision / AWS Textract Light Usage)	**5,000 – 12,000 PKR**	Reliable OCR, medium-tier plan
LLM API usage (metadata + chatbot)	**6,000 – 20,000 PKR**	Realistic for FYP-level but still premium
Embedding API	**1,500 – 5,000 PKR**	If fallback to embeddings is used
Cloud Functions (Supabase Edge + Cloudflare Worker)	**2,000 – 5,000 PKR**	For OCR pipeline + AI routing
Frontend Hosting (Vercel Pro)	**6,000 – 12,000 PKR**	Traffic analytics + edge caching
Monitoring (Sentry / LogRocket)	**1,500 – 3,000 PKR**	Good for production-level debugging

Map API (Mapbox / Geoapify Pro Tier Lite)	**4,000 – 8,000 PKR**	Higher tile/geocode limits
CDN & WAF (Cloudflare Pro)	**3,000 – 7,000 PKR**	Protection + caching
Domain	**4,000 PKR (annual)**	.app or .com

Estimated Services Total: 39,500 – 90,000 PKR/month

Although the estimated monthly operational costs appear substantial, these figures represent upper-bound estimates based on premium or pro-tier services. For this FYP, the system can operate within free and low-tier plans with controlled usage, a limited user base, and throttled AI requests. Therefore, the economic feasibility remains strong for academic implementation and controlled pilot testing.

All recurring service costs are calculated on a **monthly basis** and assume a **6 month project duration**, consistent with the overall project schedule.

Human Resource Cost

- 2 developers
- 5 days/week
- Project duration: 6 months
- Hourly Rate (per developer): **300 – 600 PKR/hour**

Cost Component	Calculation	Total
Dev 1 Monthly Cost	300–600 PKR * 160 hours	48,000 – 96,000 PKR
Dev 2 Monthly Cost	300–600 PKR * 160 hours	96,000 – 192,000 PKR
Total Dev Cost Per Month	—	**96,000 – 192,000 PKR**
Total for 4 Months	*6 × monthly*	**576,000 – 1,152,000 PKR**

Estimated HR Total: 1,152,000 – 2,304,000 PKR

Hardware & Equipment Costs

Item	Cost
Developer Laptops (2 mid-range machines)	160,000 – 220,000 PKR total
Test Devices (Android + iPhone used models)	35,000 – 70,000 PKR
Backup Drive / NAS	10,000 – 15,000 PKR
Lighting/mini photography box for OCR testing	5,000 – 8,000 PKR
High-speed Internet (6-month period)	12,000 – 18,000 PKR

Estimated Hardware Total: 222,000 – 331,000 PKR

Economical Benefits

- Reduces Wastage
- Reduces textbook costs for families
- Reduces waste via reuse
- Practical for school communities
- Welfare organizations can list verified stock to donate

The system is beneficial and has potential for real business use.

---

4. Schedule Feasibility

Given the Supabase-led architecture, development time is realistic.

Phase	Duration
Documentation & Planning (Docs 01–03)	3 weeks
System Design (SDD, ERD, UML)	3 weeks
Supabase Schema & Auth Setup	2 week

Frontend Base + Core UI	2 weeks
Map Integration & Search	3 weeks
OCR & Metadata Pipeline	3 weeks
Chat, Message System	2 weeks
Recommendation Engine	2 week
Testing, QA & Security Hardening	2 week
Final Docs, Deployment & Viva Prep	2 week
Total	**24 weeks (6 months)**

The proposed schedule is feasible due to the modular nature of the system architecture and the use of managed backend services. Parallel development of frontend components and backend configuration further reduces time risk. Additionally, the use of existing APIs and libraries allows the team to focus on integration rather than low-level implementation.

Feasibility Verdict Summary

Feasibility Type	**Verdict**	**Notes**
Technical	Feasible	Supabase simplifies backend; AI is manageable
Operational	Feasible	Easy user workflow; low maintenance
Economic	Feasible	Mostly free; OCR/LLM minimal cost
Schedule	Feasible	Achievable within the semester timeline

Constraints & Dependencies

Dependencies

- Stable internet access (Supabase is cloud-hosted)
- External OCR/LLM APIs uptime
- Nominatim geocoding reliability
- Browser support for PWA installation

Constraints

- OCR accuracy depends heavily on input quality
- Free-tier limitations (rate limits, storage limits)

APPENDIX B: Technology Analysis and Significance

Technology Transfer

The DonoBook project will be developed as a **web-based Progressive Web Application (PWA)** that can be accessed through standard web browsers on smartphones, tablets, and desktop devices. This approach ensures ease of access without requiring complex installation procedures, making the system usable for a wider range of users.

The platform will allow parents, students, and trusted welfare organizations to list, discover, and exchange educational resources such as textbooks and learning materials. Users will be able to upload images of books, search listings based on class and subject, and communicate directly through the application. Basic intelligent features such as automated book detail extraction and simple recommendations will assist users during the listing and discovery process.

To create awareness of the platform, DonoBook will be shared through **online channels**, including social media posts, university platforms, and community groups related to education. Demonstrations and walkthroughs may be conducted informally to explain how the system works and how it can benefit parents and students.

Project findings, design decisions, and implementation experiences will be documented in the final project report and may be shared in academic settings such as seminars or departmental presentations. The source code and basic documentation may be made available through an online repository to support learning and future academic use.

This approach focuses on **practical adoption, ease of access, and knowledge sharing**, without relying on complex infrastructure or commercial deployment.

Significance of the Project

DonoBook holds significance on both societal and technical levels. Socially, it addresses the growing financial burden on families caused by rising educational costs by enabling resource reuse, thereby promoting affordability, sustainability, and community collaboration. Environmentally, the platform contributes to waste reduction by extending the lifecycle of physical educational materials.

From a technical perspective, the project demonstrates the practical integration of modern web technologies with artificial intelligence capabilities in a real-world domain. It showcases how OCR, large language models, recommender systems, and geospatial search can be combined within a Progressive Web Application package to solve a tangible social problem. Academically, DonoBook serves as a comprehensive case study in full-stack system design, secure backend development, AI-assisted user interaction, and scalable architecture suitable for future expansion.

APPENDIX C: Wireframe

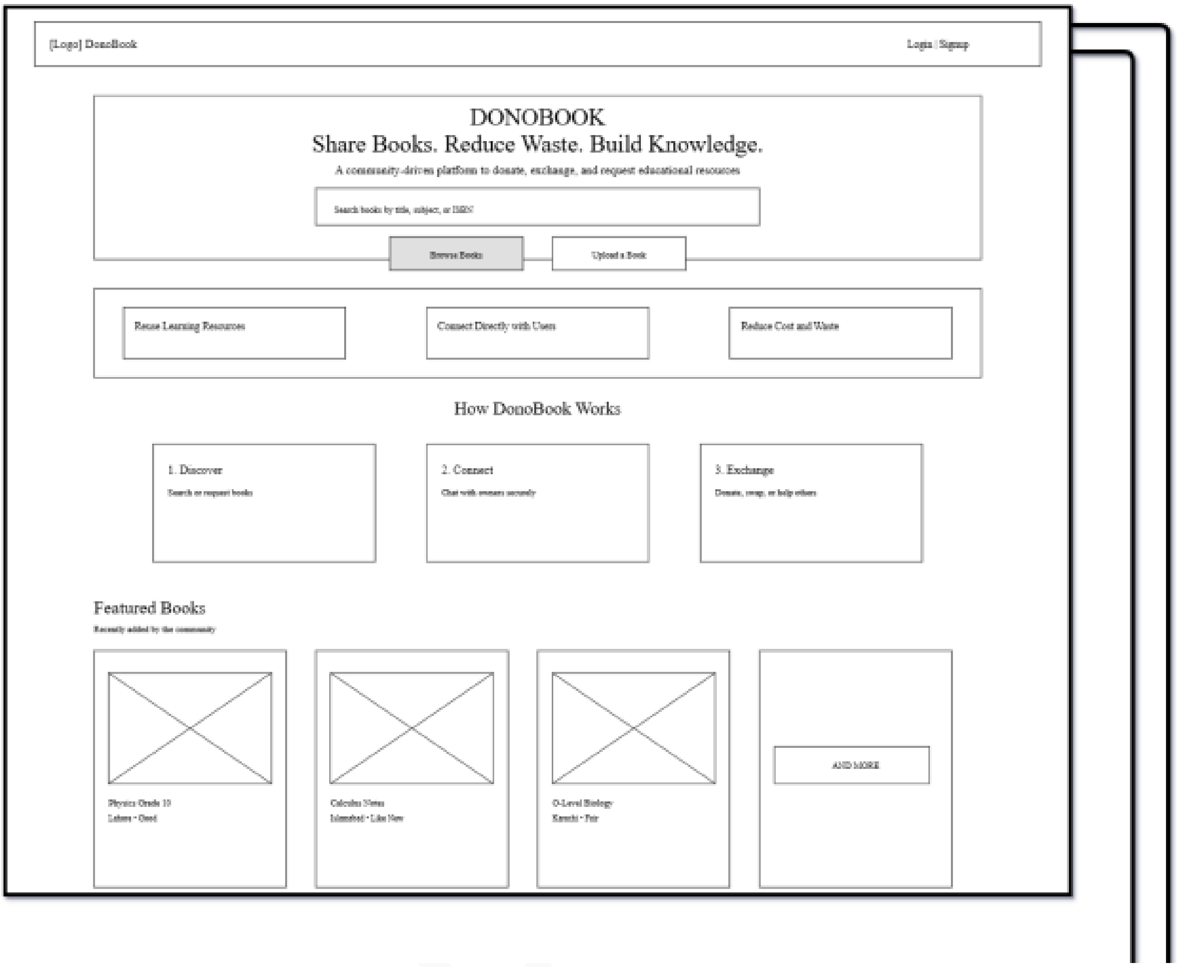
Tech: React + Tailwind + Vite. Mobile-first responsive design (PWA).

Main screens

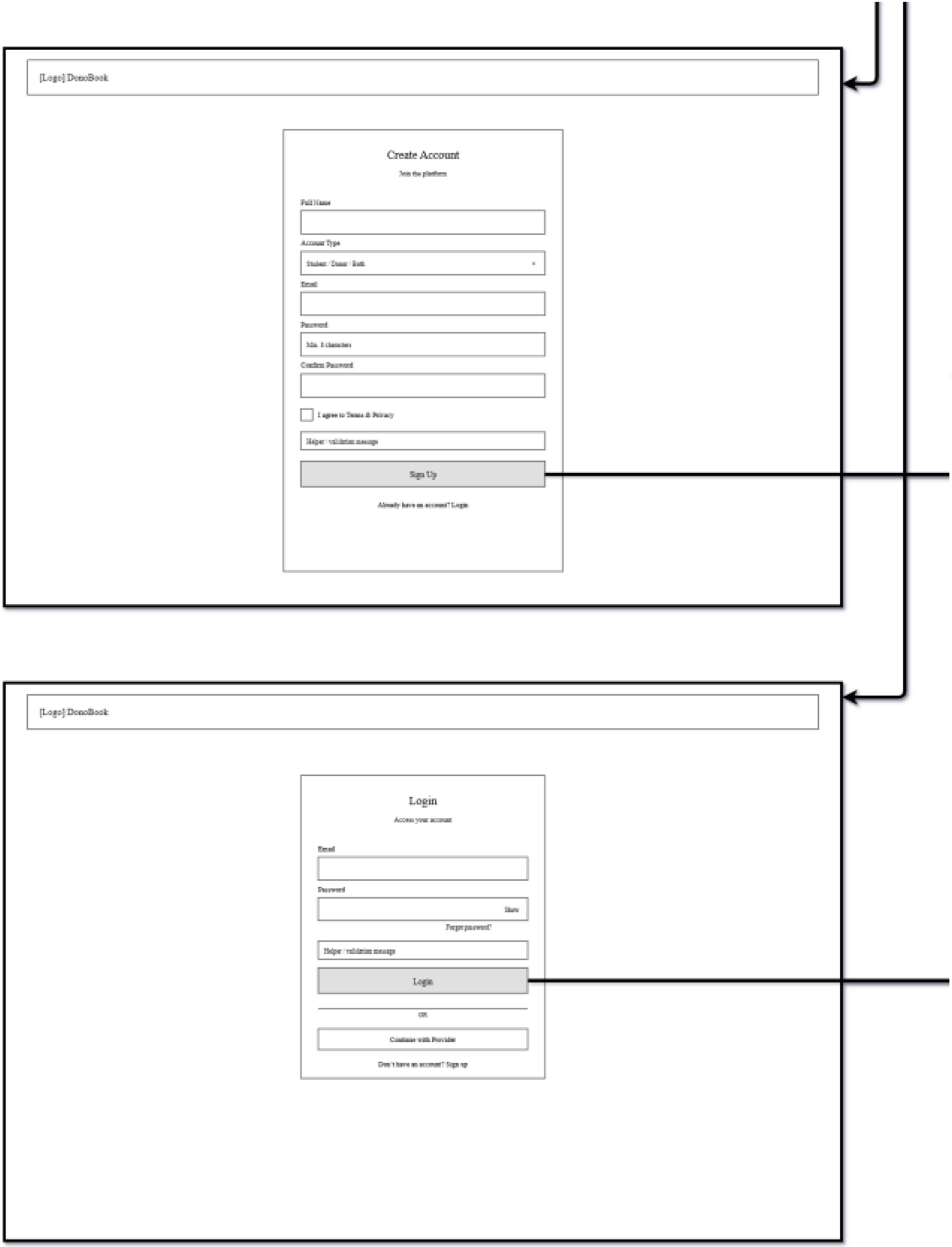
1. Landing / Search
2. Map View (Leaflet) with clustered markers
3. Listing details + QR button
4. Create Listing (photo upload + OCR metadata form)
5. Chat & Negotiation
6. Profile & History
7. Admin Dashboard

Desktop:

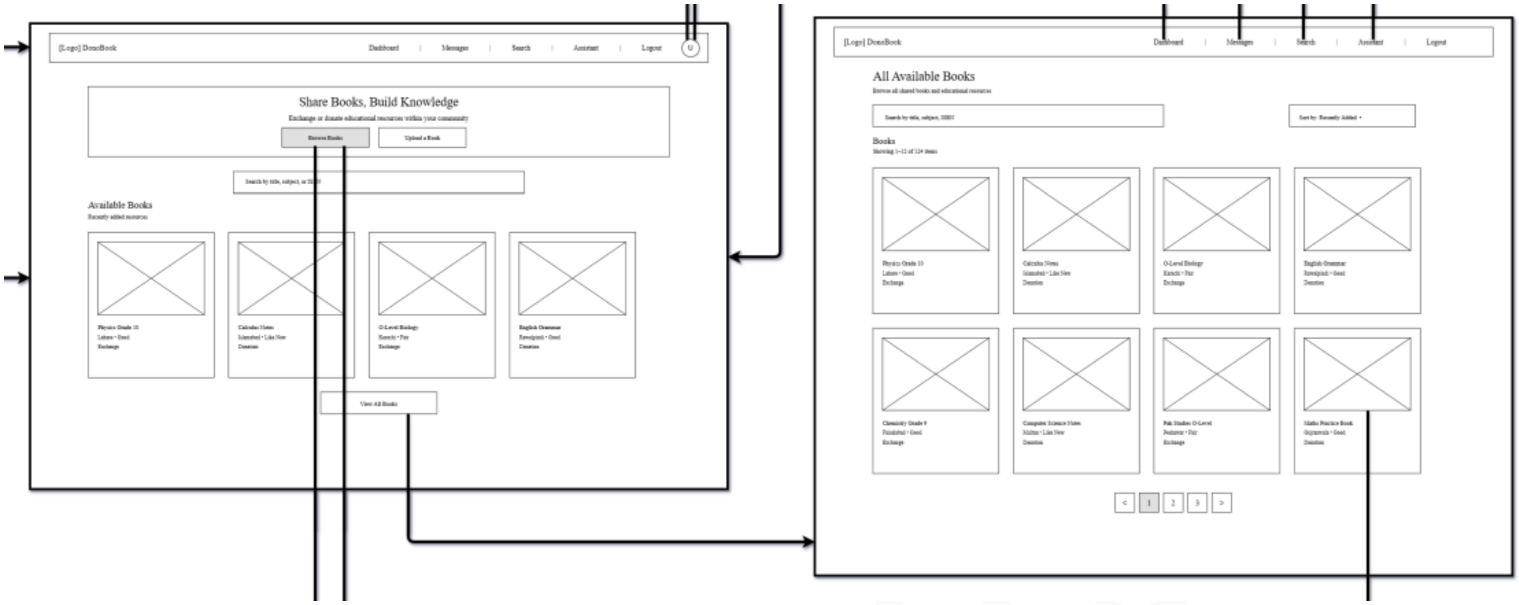
Landing Page:



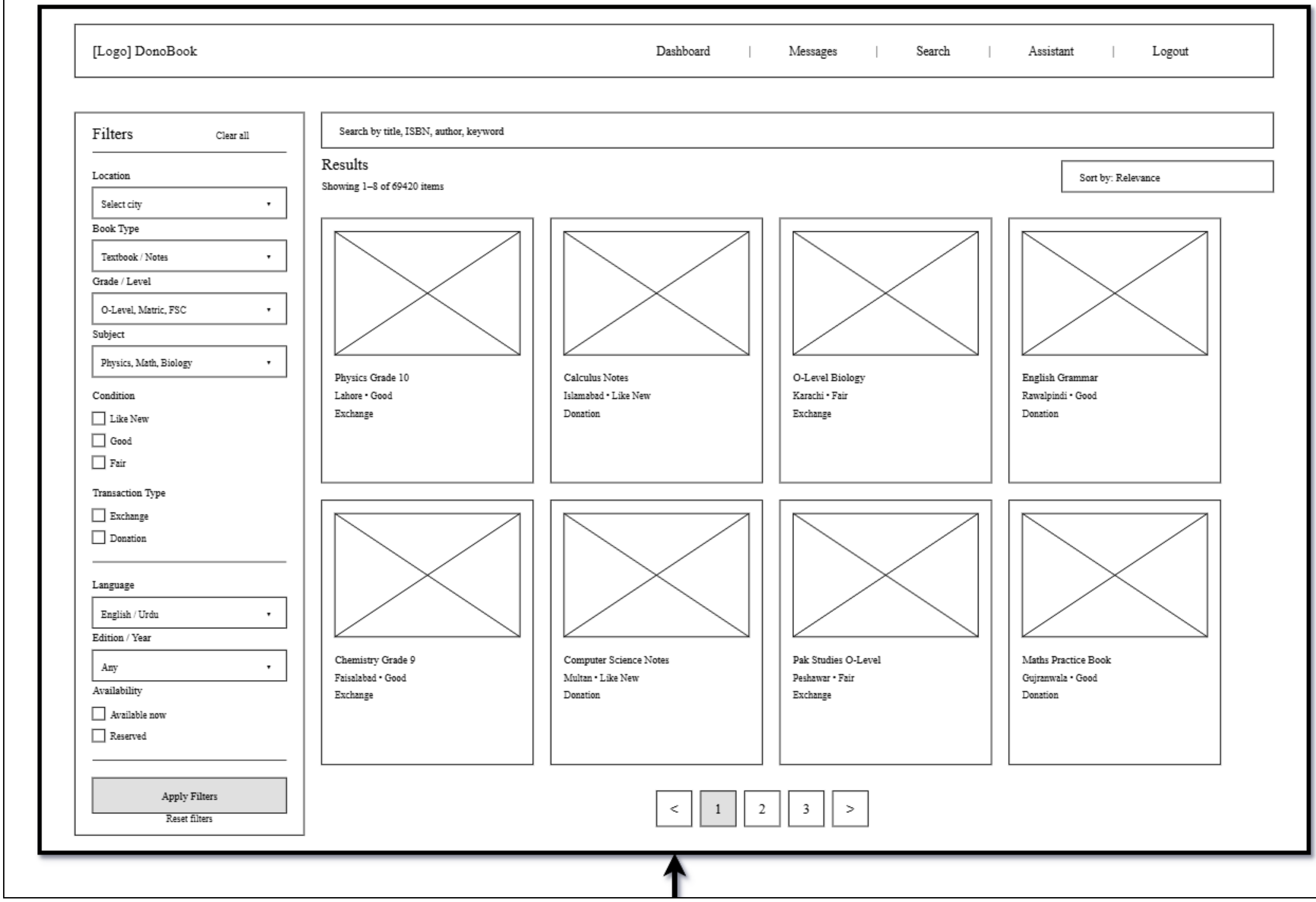
Login \ Signup:



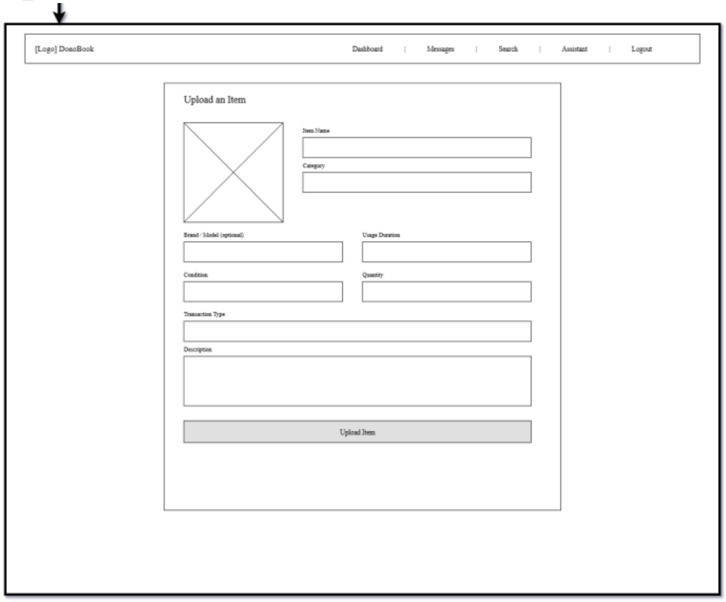
Dashboard:

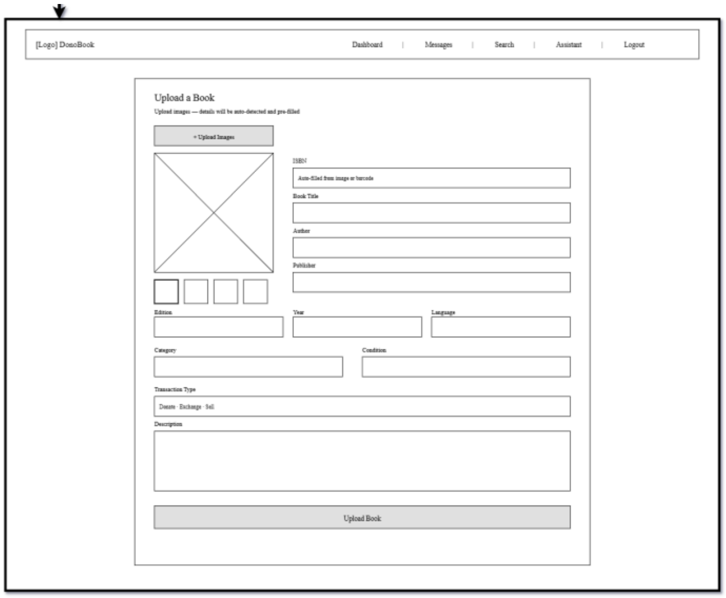


Search \ Filters:



Upload \ Donate:



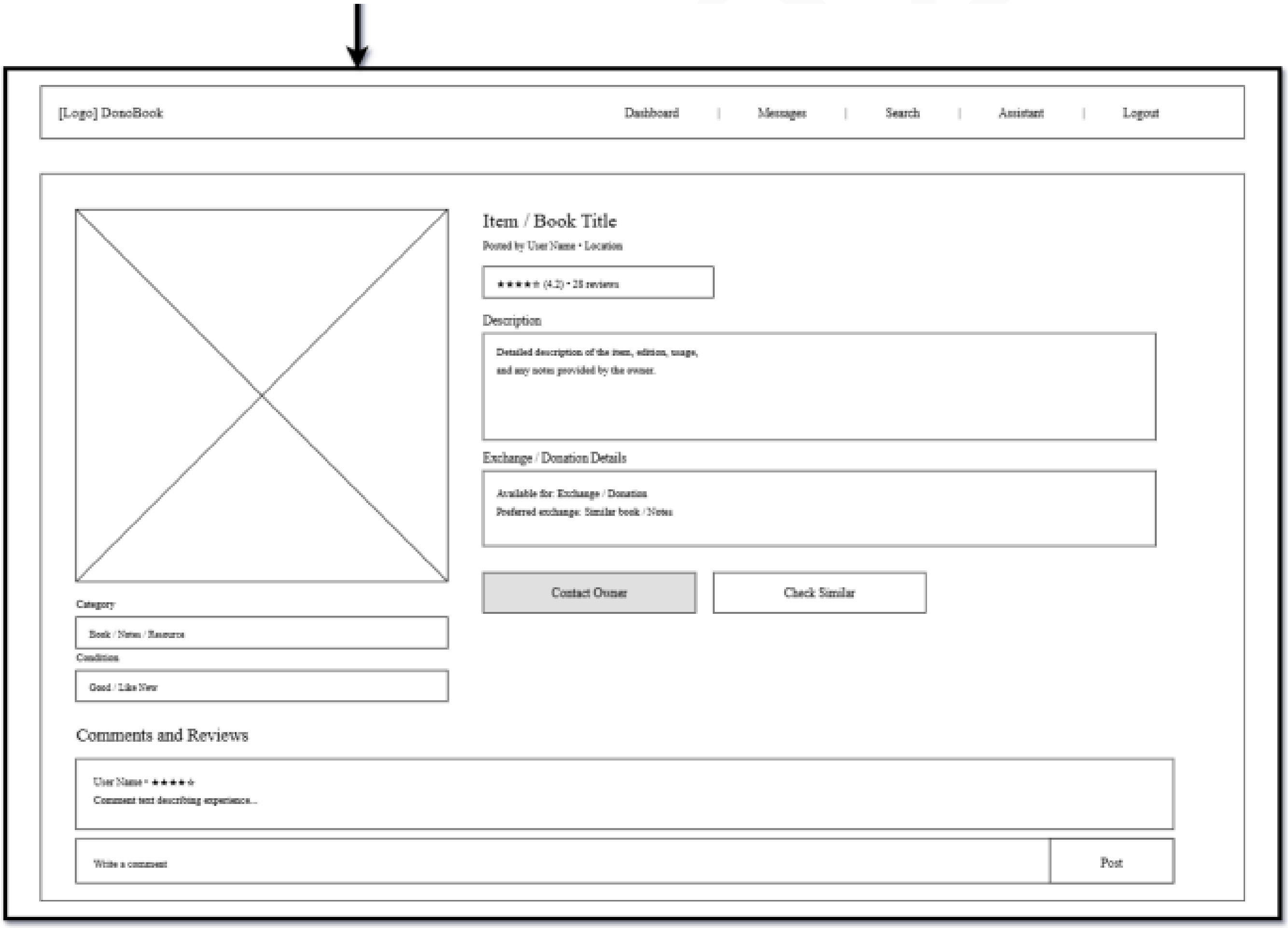


Upload images — details will be auto-detected and pre-filled

Auto-filled from image or barcode

Donate · Exchange · Sell

View Product:



Posted by User Name • Location

★★★★☆ (4.2) • 28 reviews

Detailed description of the item, edition, usage,
and any notes provided by the owner.

Available for: Exchange / Donation
Preferred exchange: Similar book / Notes

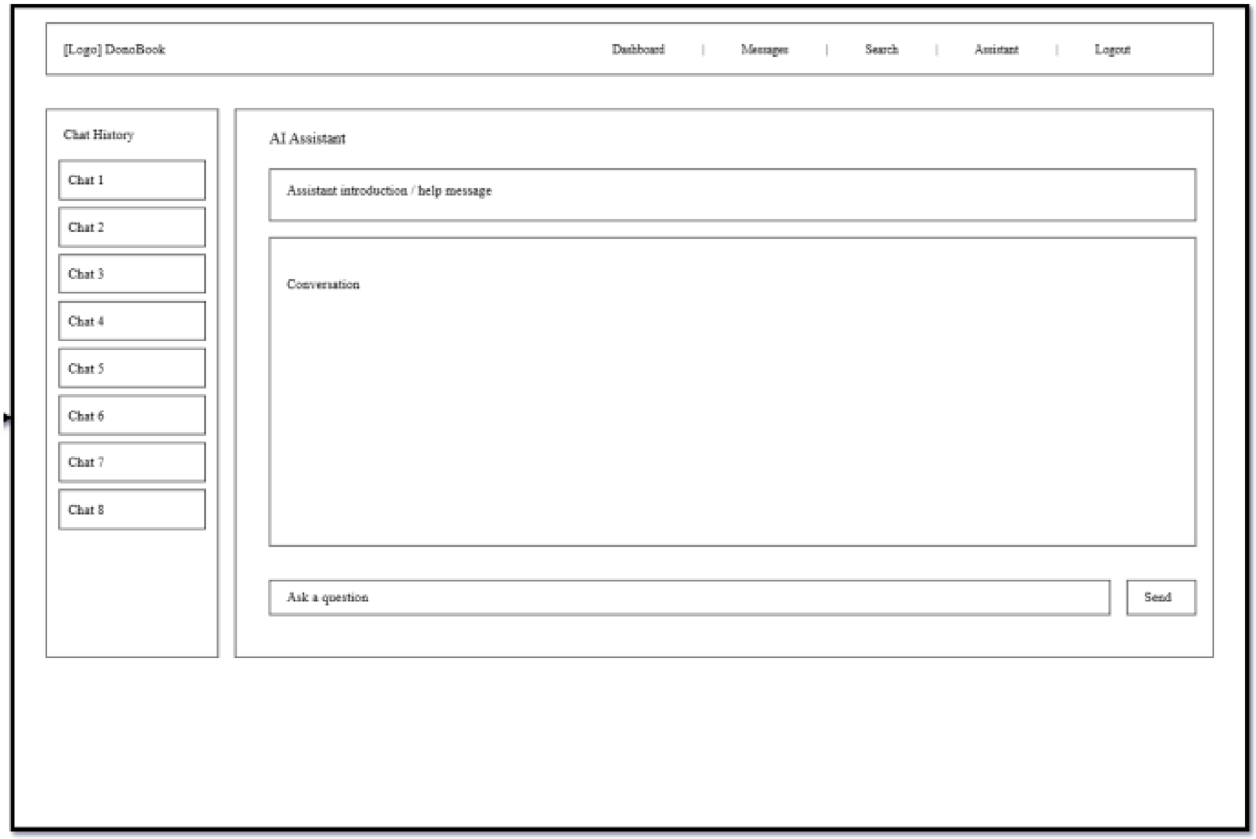
Book / Notes / Resource

Good / Like New

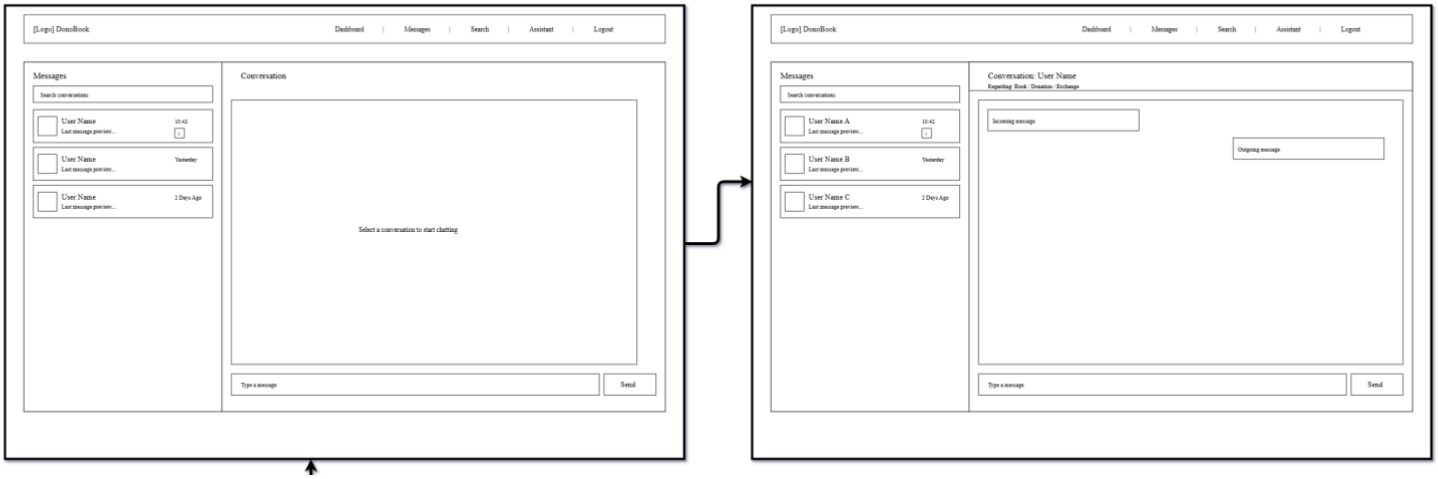
User Name • ★★★★★
Comment text describing experience...

Write a comment

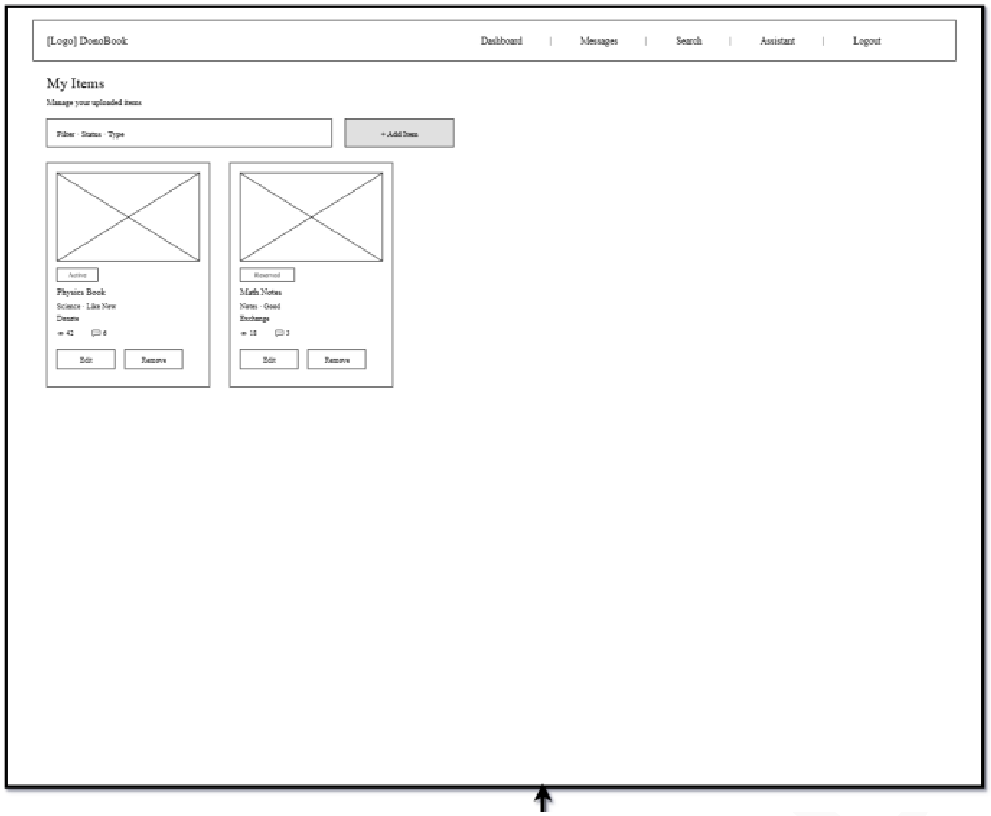
Chatbot:



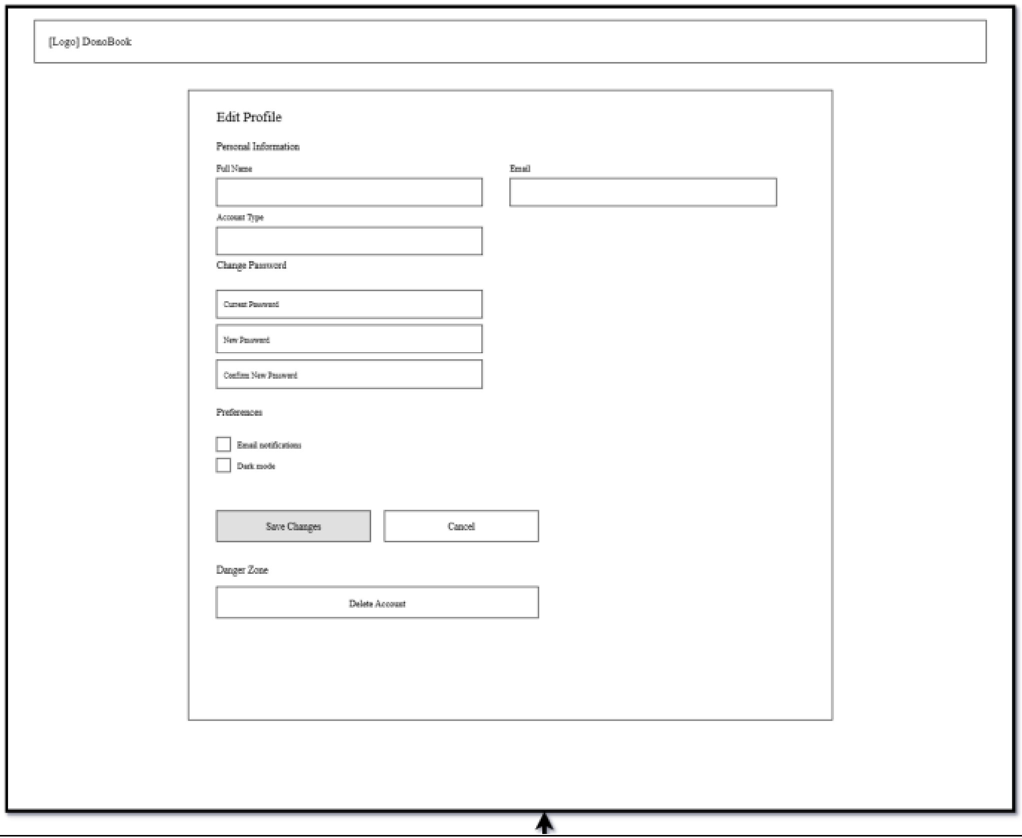
Messages:



My Profile:

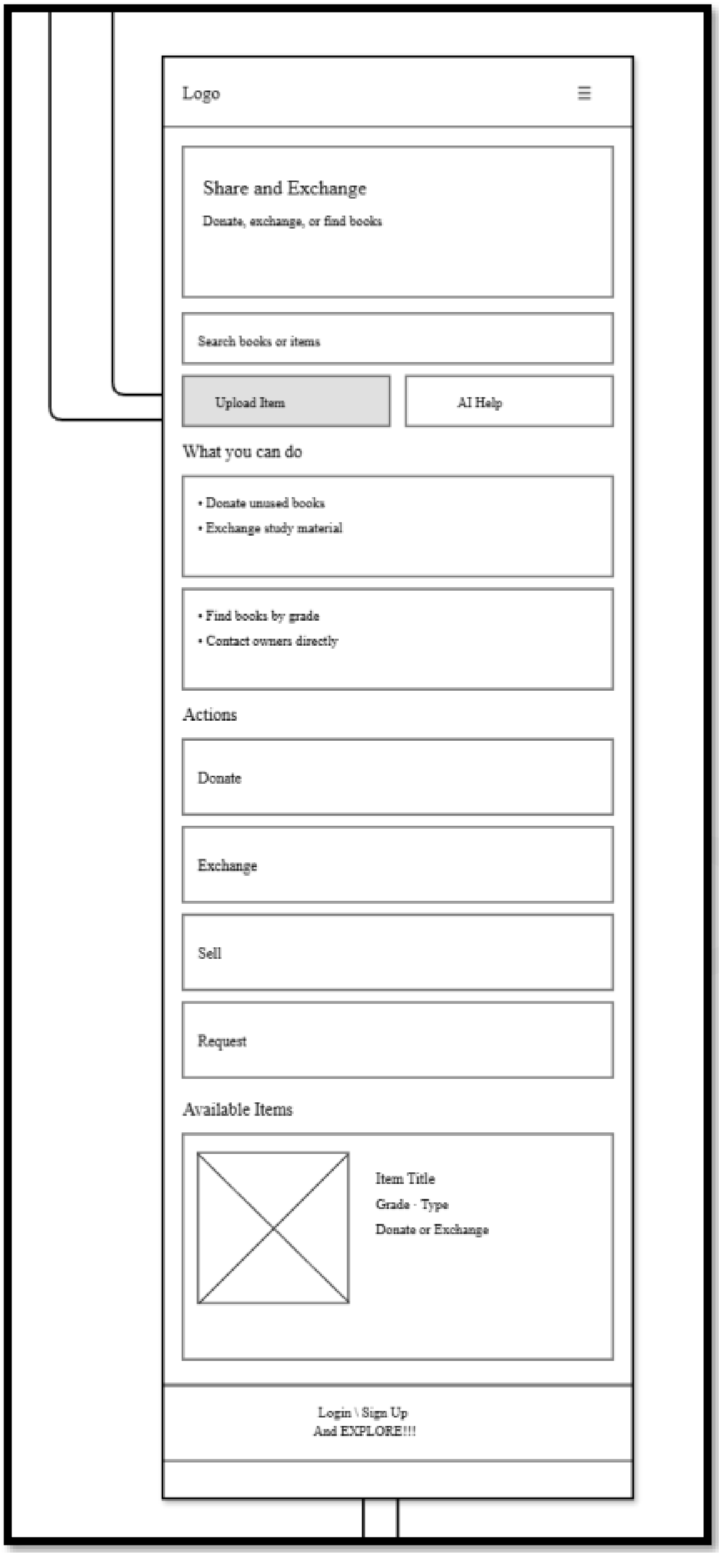


Edit Profile:

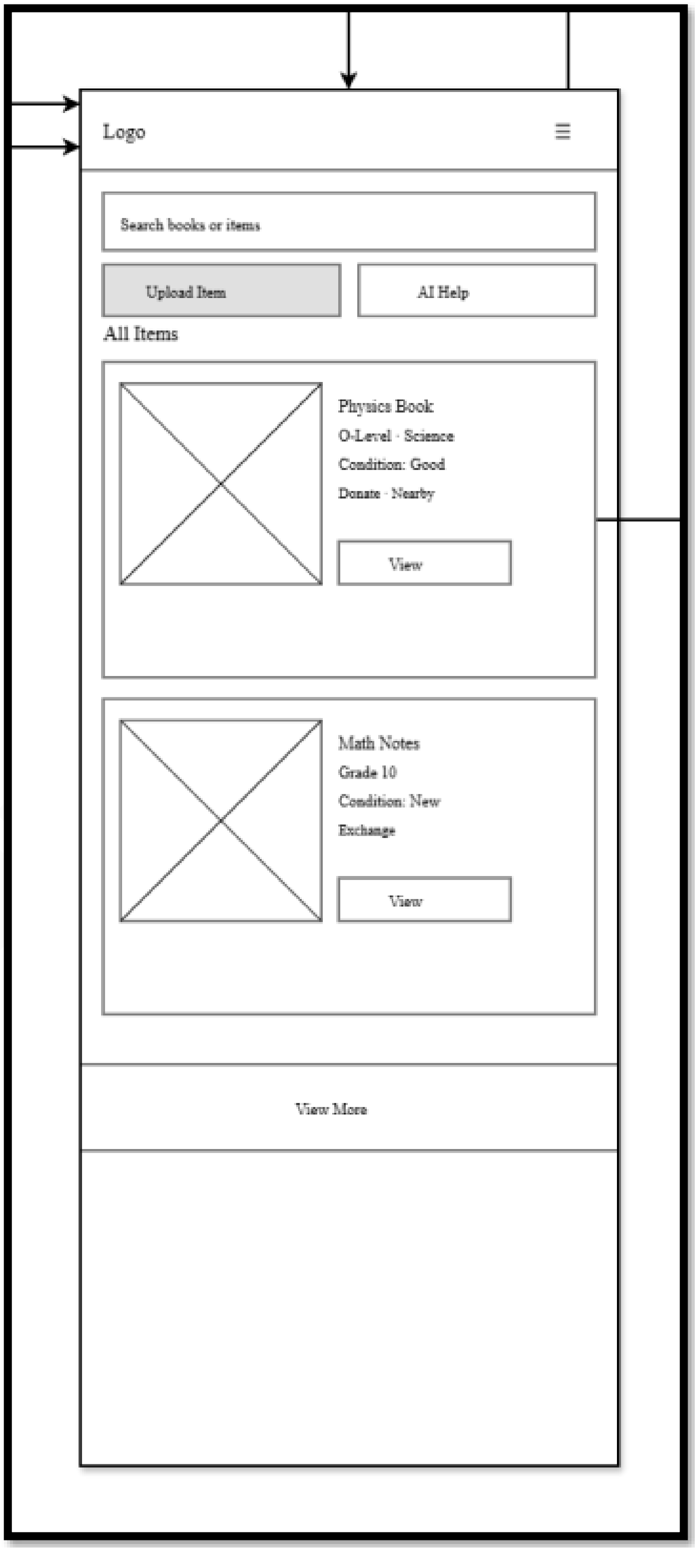


Mobile:

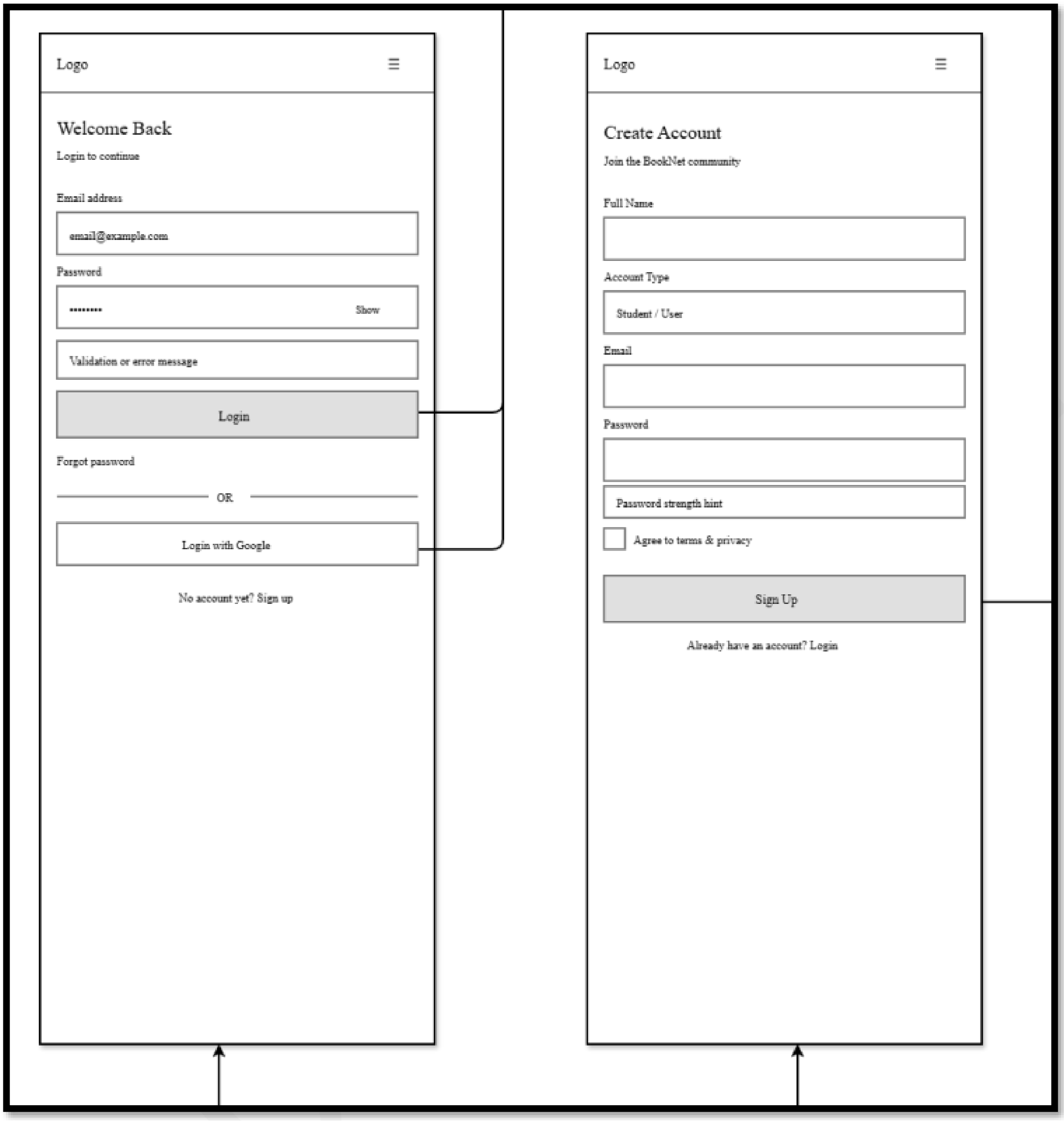
Landing Page:



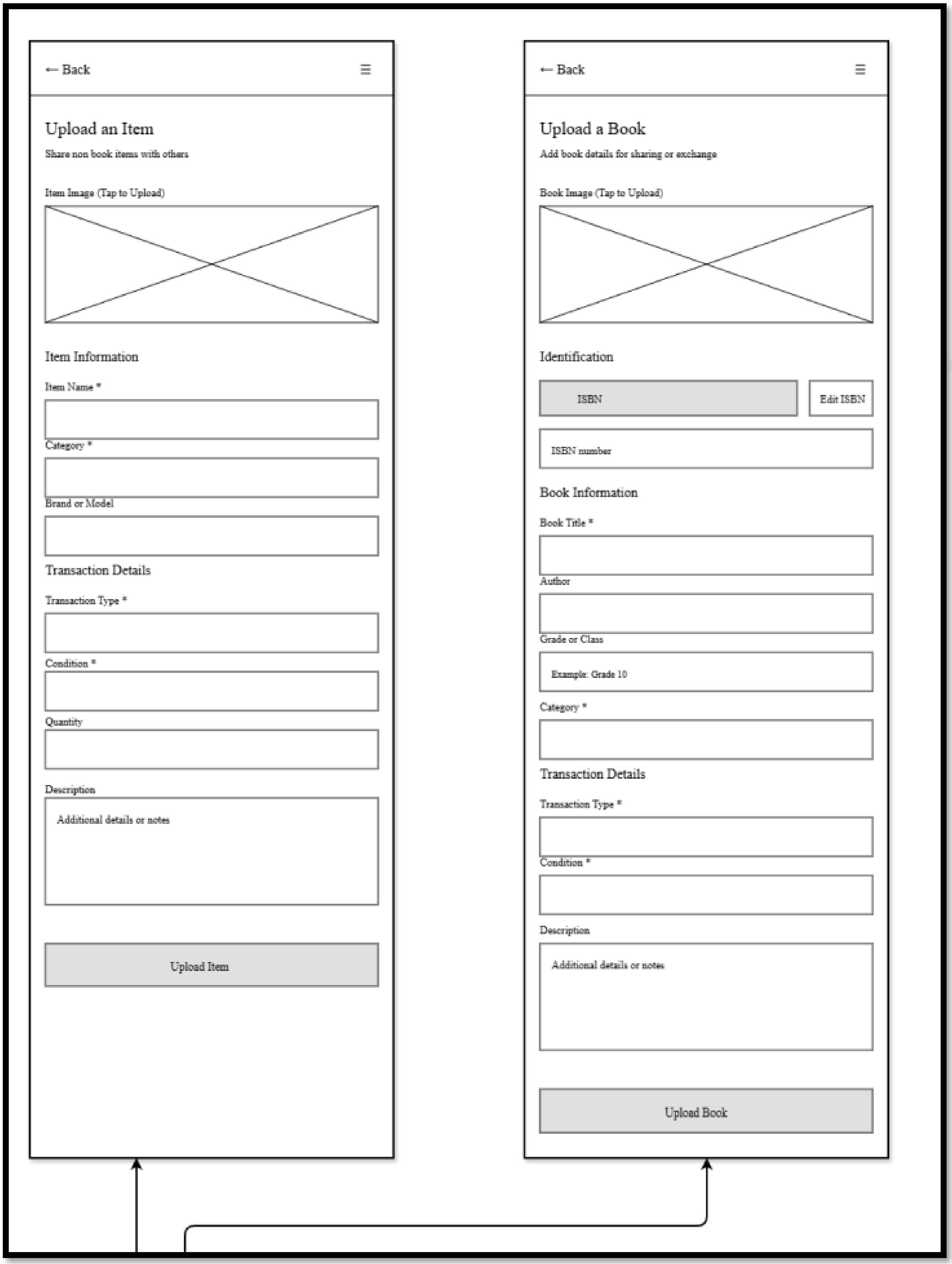
Dashboard:



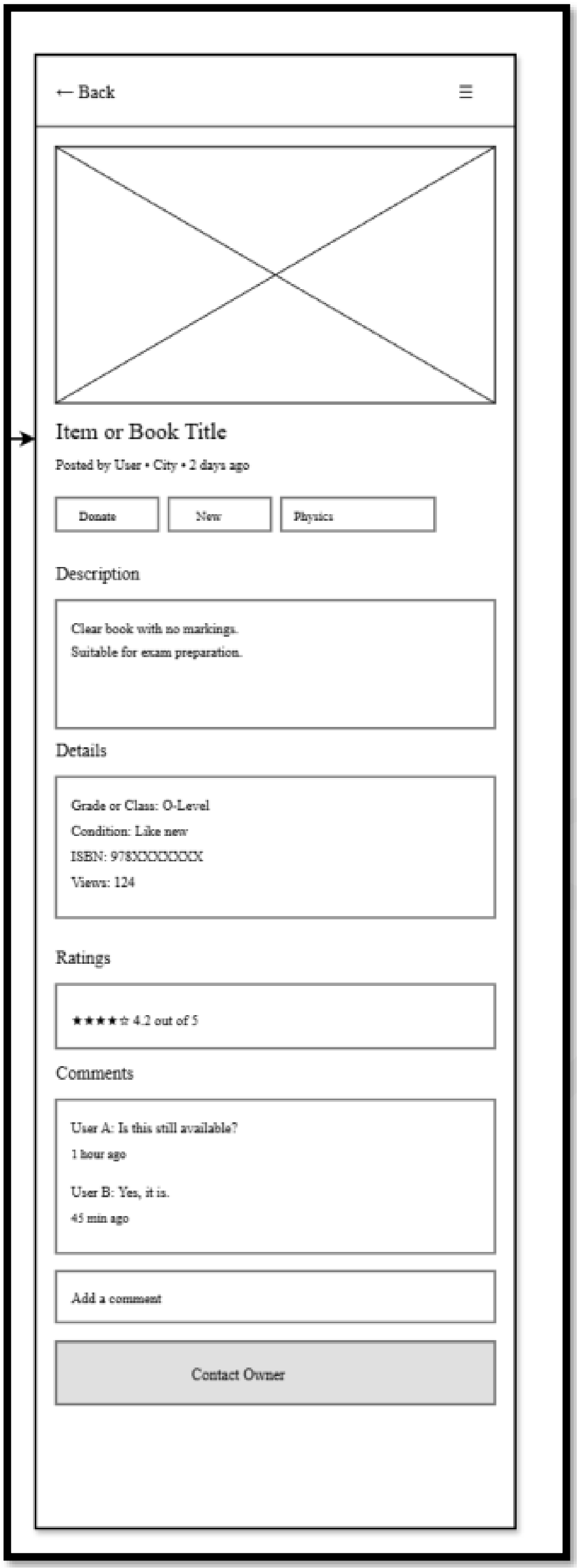
Login \ Signup:



Upload \ Donate:



View Product:



← Back ☰

Item or Book Title

Posted by User • City • 2 days ago

Description

Clear book with no markings.
Suitable for exam preparation.

Details

Grade or Class: O-Level
Condition: Like new
ISBN: 978XXXXXXXX
Views: 124

Ratings

★★★★☆ 4.2 out of 5

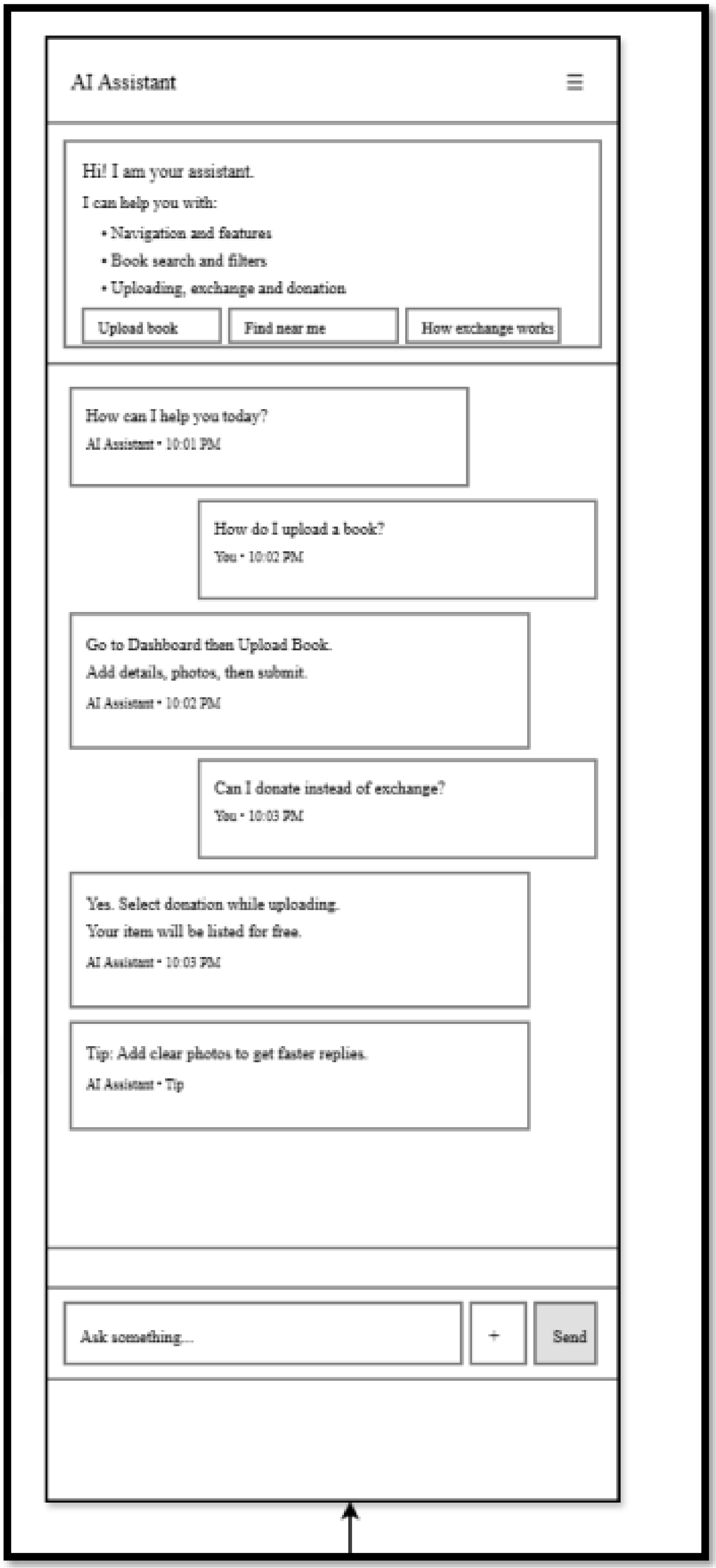
Comments

User A: Is this still available?
1 hour ago

User B: Yes, it is.
45 min ago

Add a comment

Chatbot:



AI Assistant ☰

Hi! I am your assistant.
I can help you with:

- Navigation and features
- Book search and filters
- Uploading, exchange and donation

How can I help you today?
AI Assistant • 10:01 PM

How do I upload a book?
You • 10:02 PM

Go to Dashboard then Upload Book.
Add details, photos, then submit.
AI Assistant • 10:02 PM

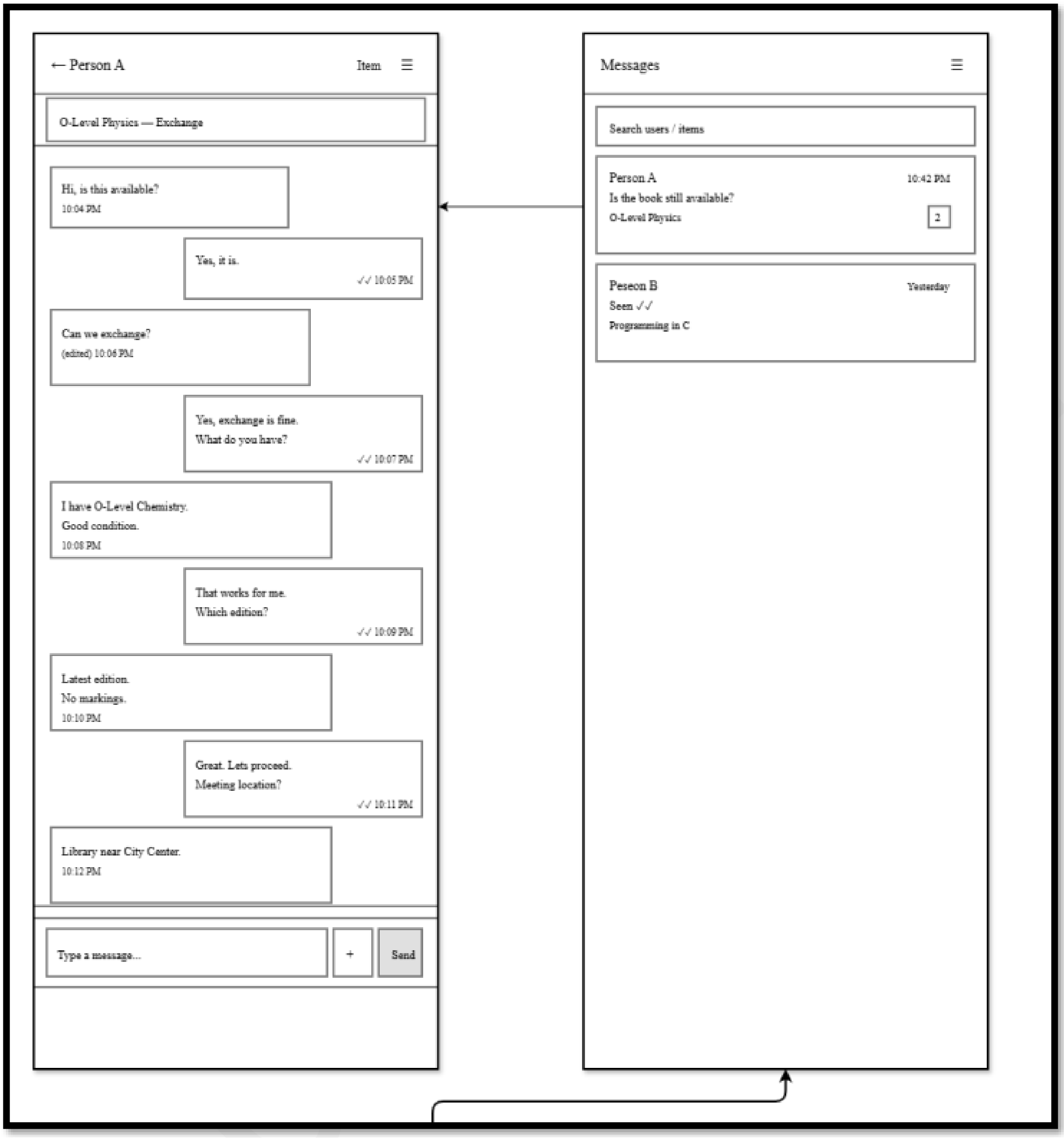
Can I donate instead of exchange?
You • 10:03 PM

Yes. Select donation while uploading.
Your item will be listed for free.
AI Assistant • 10:03 PM

Tip: Add clear photos to get faster replies.
AI Assistant • Tip

Ask something...

Messages:



Hi, is this available?
10:04 PM

Yes, it is.
✓✓ 10:05 PM

Can we exchange?
(edited) 10:06 PM

Yes, exchange is fine.
What do you have?
✓✓ 10:07 PM

I have O-Level Chemistry.
Good condition.
10:08 PM

That works for me.
Which edition?
✓✓ 10:09 PM

Latest edition.
No markings.
10:10 PM

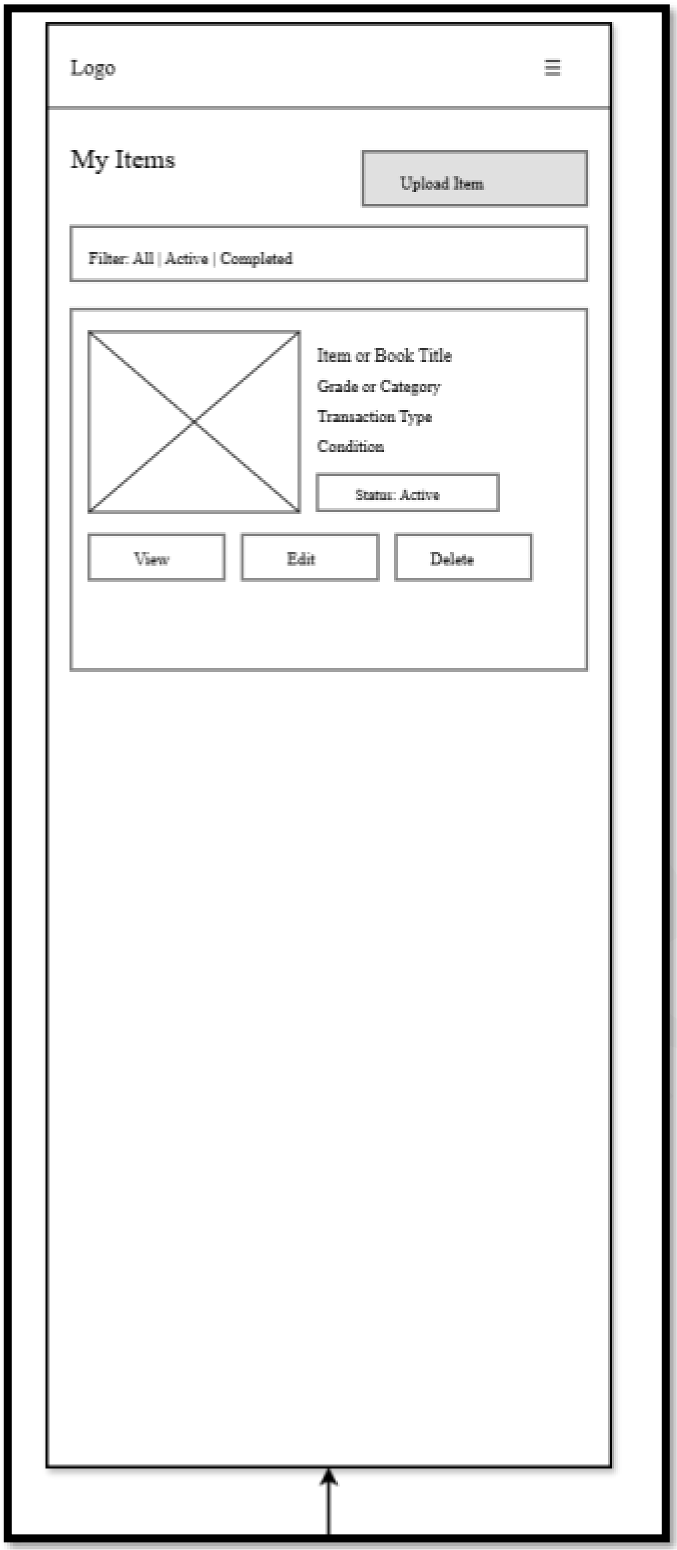
Great. Lets proceed.
Meeting location?
✓✓ 10:11 PM

Library near City Center.
10:12 PM

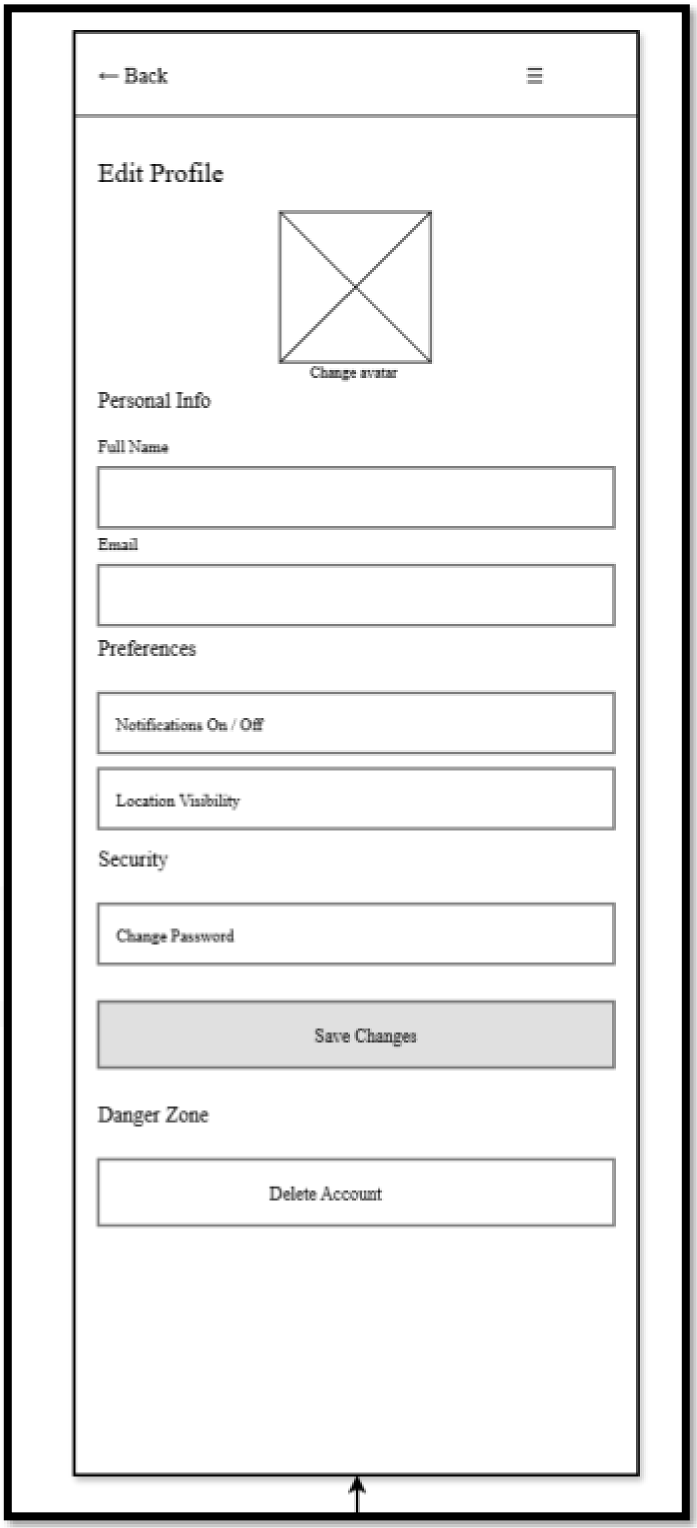
Person A 10:42 PM
Is the book still available?
O-Level Physics 2

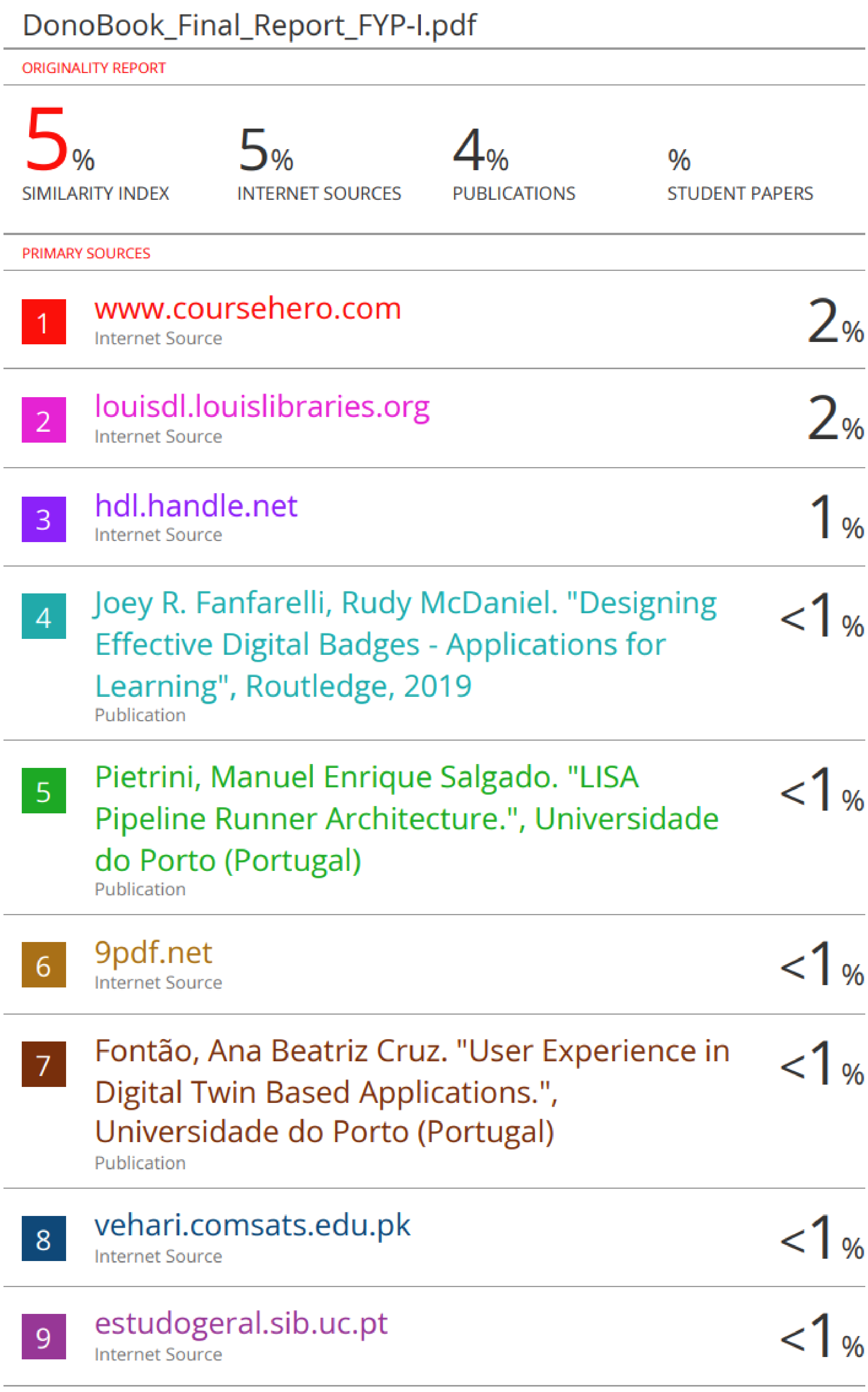
Peseon B Yesterday
Seen ✓✓
Programming in C

My Profile:



Edit Profile:





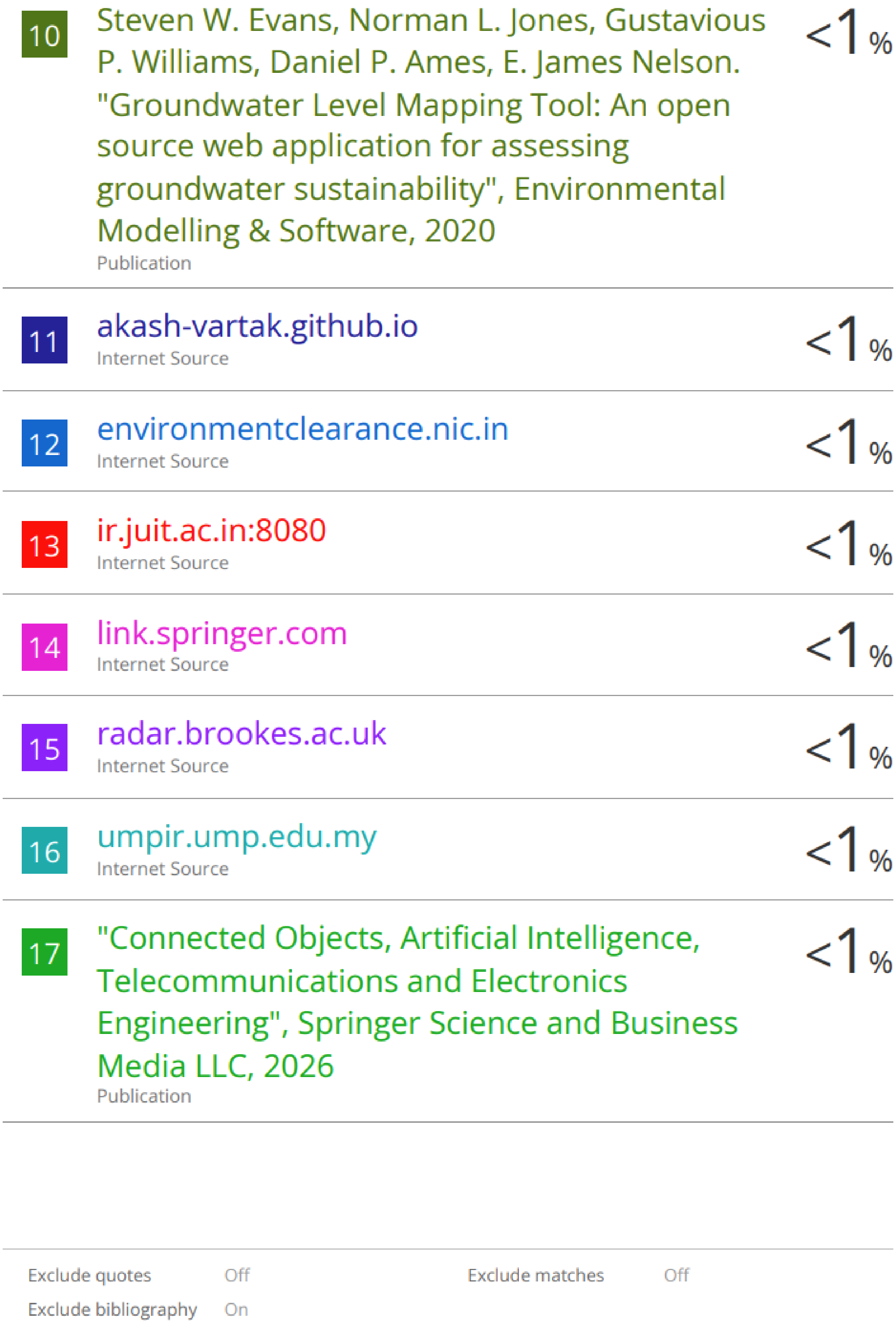
DonoBook_Final_Report_FYP-I.pdf

ORIGINALITY REPORT

5%	5%	4%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.coursehero.com Internet Source	2%
2	louisdl.louislibraries.org Internet Source	2%
3	hdl.handle.net Internet Source	1%
4	Joey R. Fanfarelli, Rudy McDaniel. "Designing Effective Digital Badges - Applications for Learning", Routledge, 2019 Publication	<1%
5	Pietrini, Manuel Enrique Salgado. "LISA Pipeline Runner Architecture.", Universidade do Porto (Portugal) Publication	<1%
6	9pdf.net Internet Source	<1%
7	Fontão, Ana Beatriz Cruz. "User Experience in Digital Twin Based Applications.", Universidade do Porto (Portugal) Publication	<1%
8	vehari.comsats.edu.pk Internet Source	<1%
9	estudogeral.sib.uc.pt Internet Source	<1%



10	Steven W. Evans, Norman L. Jones, Gustavious P. Williams, Daniel P. Ames, E. James Nelson. "Groundwater Level Mapping Tool: An open source web application for assessing groundwater sustainability", Environmental Modelling & Software, 2020 Publication	<1 %
11	akash-vartak.github.io Internet Source	<1 %
12	environmentclearance.nic.in Internet Source	<1 %
13	ir.juit.ac.in:8080 Internet Source	<1 %
14	link.springer.com Internet Source	<1 %
15	radar.brookes.ac.uk Internet Source	<1 %
16	umpir.ump.edu.my Internet Source	<1 %
17	"Connected Objects, Artificial Intelligence, Telecommunications and Electronics Engineering", Springer Science and Business Media LLC, 2026 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		