



**POLITECHNIKA
RZESZOWSKA**
im. IGNACEGO ŁUKASIEWICZA



**WYDZIAŁ
ELEKTROTECHNIKI
I INFORMATYKI**
POLITECHNIKI RZESZOWSKIEJ

Praca projektowa z przedmiotu Bazy Danych

Opracowanie strony internetowej do wymiany książek

Wykonali: Martyna Bednarska
Bartosz Sobina
Grzegorz Rzeszut
Dariusz Wiącek
Mariia Rybak
Yuliia Paziuk
Veronika Vanivska
Łukasz Skórka

Rzeszów, 2024

Spis treści

1. Założenia projektowe	2
1.1. Analiza wymagań i planowanie projektu	2
1.2. Zakres prac	2
2. Opis technologii	4
3. Opis struktury bazy danych	5
4. Diagramy	8
5. Wygląd strony internetowej	10

1. Założenia projektowe

1.1. Analiza wymagań i planowanie projektu

a. Nazwa projektu:

Serwis do wymiany książek

b. Opis aplikacji/platformy:

Serwis do wymiany książek jak sama nazwa wskazuje będzie umożliwiał zalogowanym użytkownikom wymianę książek. Każdy użytkownik będzie mógł szukać upragnionych książek a także dzielić się swoimi zasobami z innymi. Transakcja wymiany uzgadniana będzie między użytkownikami na czacie, gdzie użytkownicy w dowolny i wygodny dla siebie sposób informować się będą o warunkach wymiany. Ciekawa grafika i możliwość przeglądania książek, których użytkownik szuka bądź posiada pozwolą na wygodne i komfortowe korzystanie z serwisu.

c. Cele projektu:

Stworzenie serwisu do wymiany książek.

d. Zespół projektowy:

Imię Nazwisko	Rola w zespole
Martyna Bednarska	Scrum Master, Product Owner
Bartosz Sobina	Frontend Developer
Dariusz Wiącek	Frontend Developer
Grzegorz Rzeszut	Backend Developer
Mariia Rybak	QA Engineer
Yuliia Paziuk	UI/UX Designer
Veronika Vanivska	Fullstack Developer
Łukasz Skórka	Database Specialist

1.2. Zakres prac

1. Rejestracja użytkownika:

- o Implementacja formularza rejestracyjnego umożliwiającego wprowadzenie imienia, nazwiska, adresu e-mail, hasła oraz dodanie zdjęcia profilowego.

- Weryfikacja poprawności wprowadzonych danych.
- Zapisanie danych użytkownika w bazie danych.

2. Logowanie użytkownika:

- Stworzenie formularza logowania umożliwiającego użytkownikowi wprowadzenie adresu e-mail i hasła.
- Weryfikacja poprawności danych logowania.

3. Zabezpieczenie dostępu:

- Implementacja mechanizmu zabezpieczającego dostęp do funkcji serwisu tylko dla zalogowanych użytkowników.
- Zapewnienie bezpiecznej komunikacji między przeglądarką a serwerem (HTTPS).

4. Zarządzanie kontem użytkownika:

- Umożliwienie użytkownikowi zmiany hasła i zdjęcia profilowego.
- Implementacja funkcji usuwania konta użytkownika.

5. Dodawanie książek do profilu:

- Stworzenie interfejsu umożliwiającego dodanie książki do kategorii "Szukam" lub "Wymieniam".
- Zbieranie informacji o książce: tytuł, autor, rok wydania, wydawnictwo, opcjonalnie zdjęcie okładki.
- Zapisanie danych książki w profilu użytkownika w bazie danych.

6. Aktualizacja ofert wymiany:

- Implementacja możliwości dodawania i usuwania książek z profilu użytkownika.
- Aktualizacja danych książek na profilu po dokonaniu zmian.

7. Czat między użytkownikami:

- Stworzenie interfejsu czatu umożliwiającego komunikację między użytkownikami.
- Powiadomienie użytkownika o nowych wiadomościach w czacie.
- Możliwość wyszukiwania użytkowników oraz konwersacji z nimi.

8. Filtrowanie i wyszukiwanie książek:

- Implementacja funkcji wyszukiwania książek po autorze, tytule, kategorii oraz osobie.
- Możliwość filtrowania książek alfabetycznie na stronie głównej.

9. Interakcja ze strony głównej i profilu użytkownika:

- Dodanie przycisku umożliwiającego wejście w czat z profilu użytkownika.
- Implementacja możliwości przeglądania profilu oraz ustawień użytkownika ze strony głównej.
- Stworzenie przycisku umożliwiającego kontynuację czatu z poziomu strony głównej.

2. Opis technologii

1. **TypeScript** to nadzbiór języka JavaScript, który dodaje statyczną typizację oraz inne zaawansowane funkcje do JavaScript.
2. **React** to biblioteka JavaScript, która umożliwia tworzenie interfejsów użytkownika w sposób modułowy. Dzięki komponentowej architekturze, React pozwala na budowanie aplikacji, w których interfejs jest podzielony na niezależne części zwane komponentami.
3. **Next.js** to meta-framework Reacta, który zapewnia wydajne narzędzia do budowania aplikacji internetowych. Oferuje wiele zaawansowanych funkcji, takich jak renderowanie po stronie serwera (SSR), statyczne generowanie (SSG) oraz obsługa routingu.
4. **Tailwind CSS** to narzędzie do tworzenia stylów w aplikacjach internetowych. Zamiast korzystać z tradycyjnych arkuszy stylów, Tailwind CSS stosuje podejście oparte na klasach, co pozwala szybko tworzyć interfejsy użytkownika poprzez zastosowanie predefiniowanych klas CSS.
5. **Supabase** to platforma do budowania aplikacji serwerowych, która zapewnia gotowe narzędzia do obsługi baz danych, autentykacji użytkowników oraz hostingu plików. Jest oparta na bazie danych PostgreSQL i oferuje wygodne API do pracy z danymi.
6. **Prisma** to narzędzie ORM (Object-Relational Mapping), które ułatwia pracę z bazami danych w aplikacjach backendowych. Prisma automatycznie generuje kod do obsługi bazy danych na podstawie zdefiniowanych modeli danych.

7. **Auth.js** to biblioteka do obsługi autentykacji i zarządzania sesjami użytkowników w aplikacjach internetowych. Zapewnia funkcje takie jak uwierzytelnianie, zarządzanie sesjami oraz integrację z różnymi dostawcami usług autentykacji (np. Google, Github).
8. **Vercel** to platforma do hostowania aplikacji webowych, która zapewnia łatwe i szybkie wdrażanie aplikacji zbudowanych w frameworkach takich jak Next.js, React czy Vue.js.

3. Opis struktury bazy danych

Tabele i ich pola:

Tabela User:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- name: String (opcjonalne)
- email: String (opcjonalne, unikalne)
- emailVerified: DateTime (opcjonalne)
- description: String (wymagane, domyślnie "")
- enabledNotifications: Boolean (wymagane, domyślnie true)
- image: String (opcjonalne)
- password: String (opcjonalne)

Tabela Account:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- userId: String (wymagane)
- type: String (wymagane)
- provider: String (wymagane)
- providerAccountId: String (wymagane)
- refresh_token: String (opcjonalne)
- access_token: String (opcjonalne)
- expires_at: Int (opcjonalne)
- token_type: String (opcjonalne)
- scope: String (opcjonalne)
- session_state: String (opcjonalne)

Relacja z tabelą User (pole userId, referencja do id, z zasadą kaskadowego usuwania). Kombinacja pól provider i providerAccountId musi być unikalna.

Tabela VerificationToken:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- email: String (wymagane)
- token: String (wymagane, unikalne)
- expires: DateTime (wymagane)

Kombinacja pól email i token musi być unikalna.

Tabela PasswordResetToken:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- email: String (wymagane)
- token: String (wymagane, unikalne)
- expires: DateTime (wymagane)

Kombinacja pól email i token musi być unikalna.

Tabela Book:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- title: String (wymagane)
- author: String (wymagane)
- description: String (opcjonalne)
- publicationYear: Int (opcjonalne)
- publisher: String (opcjonalne)
- coverImage: String (opcjonalne)
- category: Enum (wymagane, BookCategory)
- userId: String (wymagane)

Relacja z tabelą User (pole userId, referencja do id, z zasadą kaskadowego usuwania).

Tabela Chat:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- user1Id: String (wymagane)
- user2Id: String (wymagane)

Relacja z tabelą User.

Relacja z tabelą Message.

Kombinacja pól user1Id i user2Id musi być unikalna.

Tabela Message:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())

- content: String (wymagane)
- timestamp: DateTime (wymagane, domyślnie now())
- senderId: String (wymagane)
- chatId: String (wymagane)

Relacja z tabelą Chat (pole chatId, referencja do id, z zasadą kaskadowego usuwania).

Relacja z tabelą User (pole senderId, referencja do id, z zasadą kaskadowego usuwania).

Tabela Notification:

- id: String (wymagane, unikalne, domyślnie generowane przez cuid())
- type: Enum (wymagane, NotificationType)
- isReaded: Boolean (wymagane, domyślnie false)
- timestamp: DateTime (wymagane, domyślnie now())
- senderId: String (wymagane)
- reciverId: String (wymagane)

Relacja z tabelą User (pole senderId, referencja do id, z zasadą kaskadowego usuwania, relacja o nazwie Sender).

Relacja z tabelą User (pole reciverId, referencja do id, z zasadą kaskadowego usuwania, relacja o nazwie Reciver).

Kombinacja pól senderId, reciverId i type musi być unikalna.

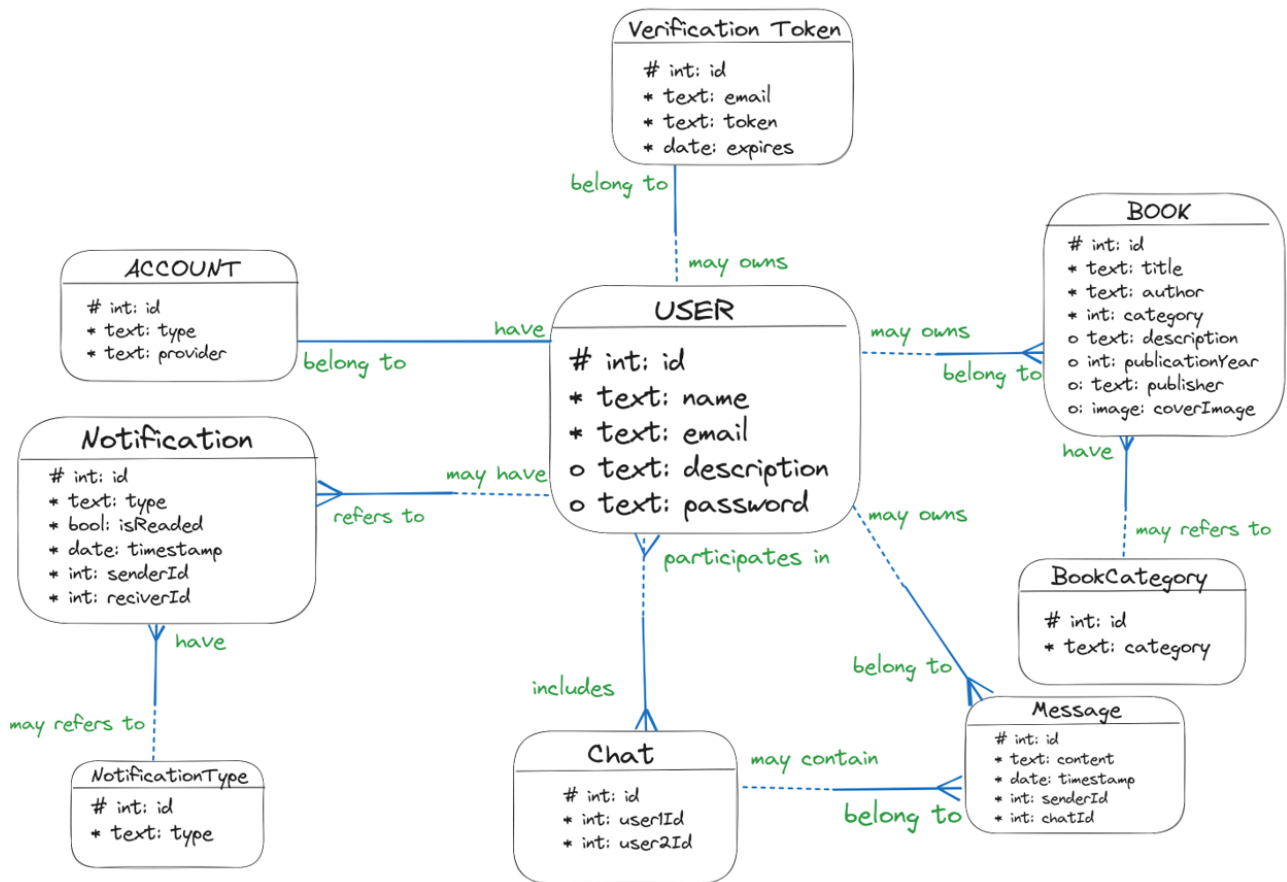
Relacje:

- User 1:1 Account
- User 1:N Book
- User 1:N Message
- User 1:N Notification
- User 1:N Chat
- Chat 1:N Message

Enumy:

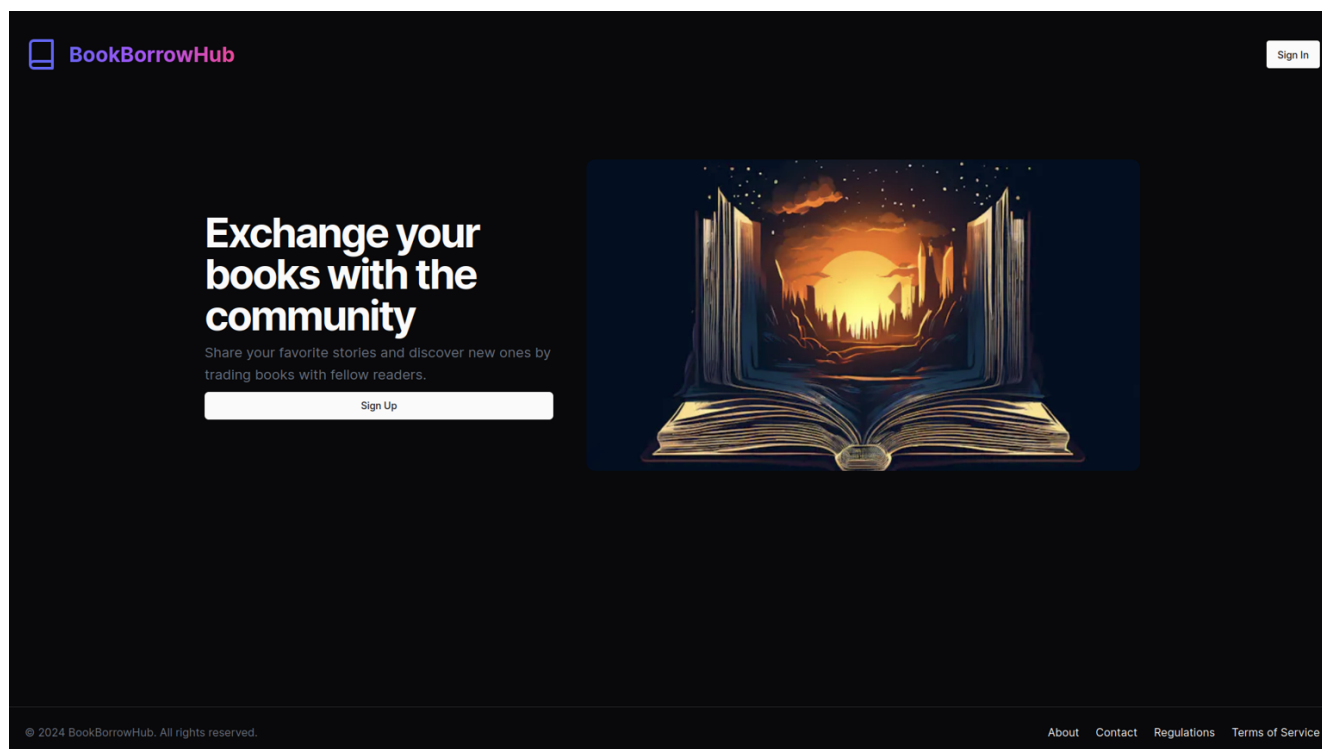
- NotificationType:
 - NEW_CHAT
 - NEW_MESSAGE
- BookCategory:
 - LOOKING_FOR
 - EXCHANGING

4. Diagramy



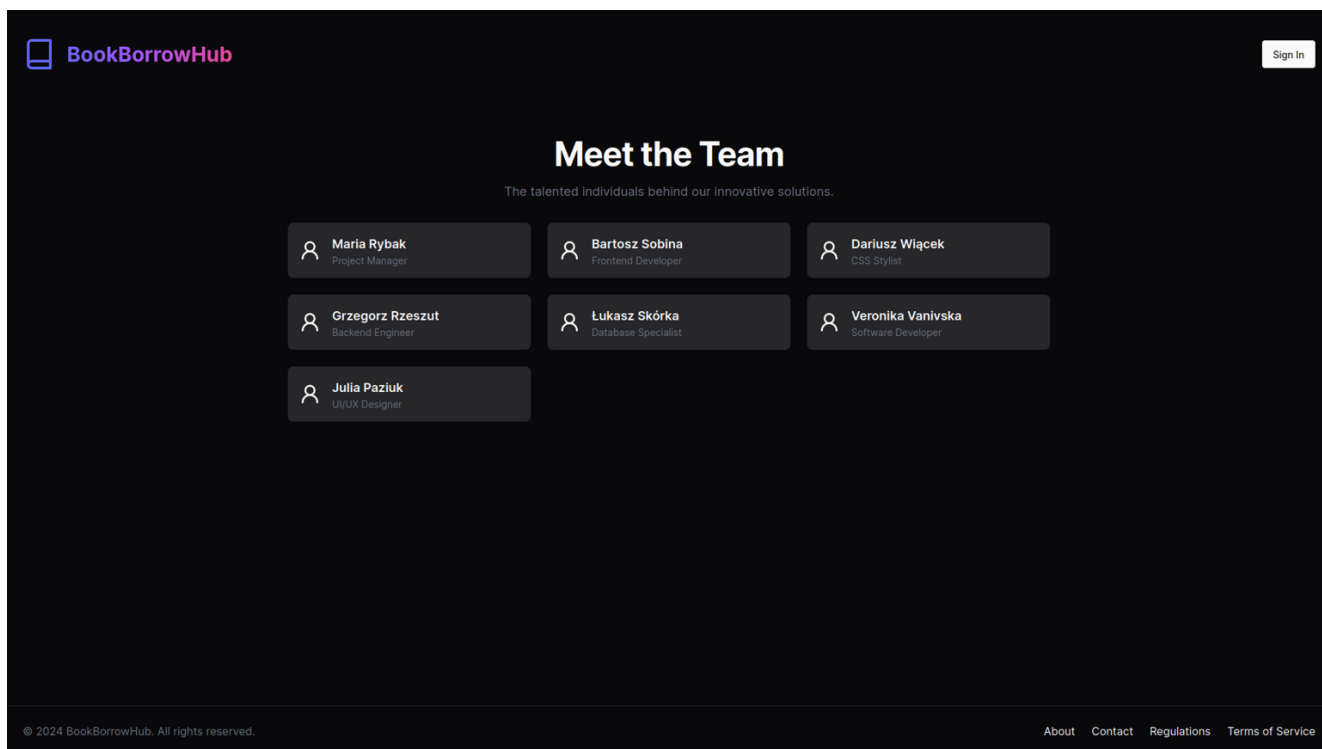
Rysunek 4.1: Diagram ERD

5. Wygląd strony internetowej



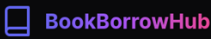
Rysunek 5.4: Strona główna

Strona główna "BookBorrowHub" prezentuje minimalistyczne, ciemne tło, na którym widoczne są jasne elementy tekstowe i graficzne. Centralnym elementem jest dynamiczna ilustracja otwartej książki, która ma zachęcać użytkowników do korzystania ze strony oraz przycisk "Sign Up" który służy do rejestracji. W górnej części strony znajduje się przycisk "Sign In", który służy do logowania. Na dole strony umieszczone są linki do dodatkowych informacji: "About", "Contact", "Regulations" i "Terms of Service". Całość projektu jest nowoczesna i funkcjonalna, zachęcając użytkownika do interakcji z platformą.



Rysunek 5.5: Panel informacji o zespole developerów

Widzimy tutaj zawartość zakładki 'About'. Przedstawia ona zespół developerów, który był odpowiedzialny za stworzenie strony wraz ze wszystkimi jej funkcjonalnościami.



Sign In

Get in Touch

Have a question or want to work together? Fill out the form below and we'll get back to you as soon as possible.

Name
Enter your name

Email
Enter your email

Message
Enter your message

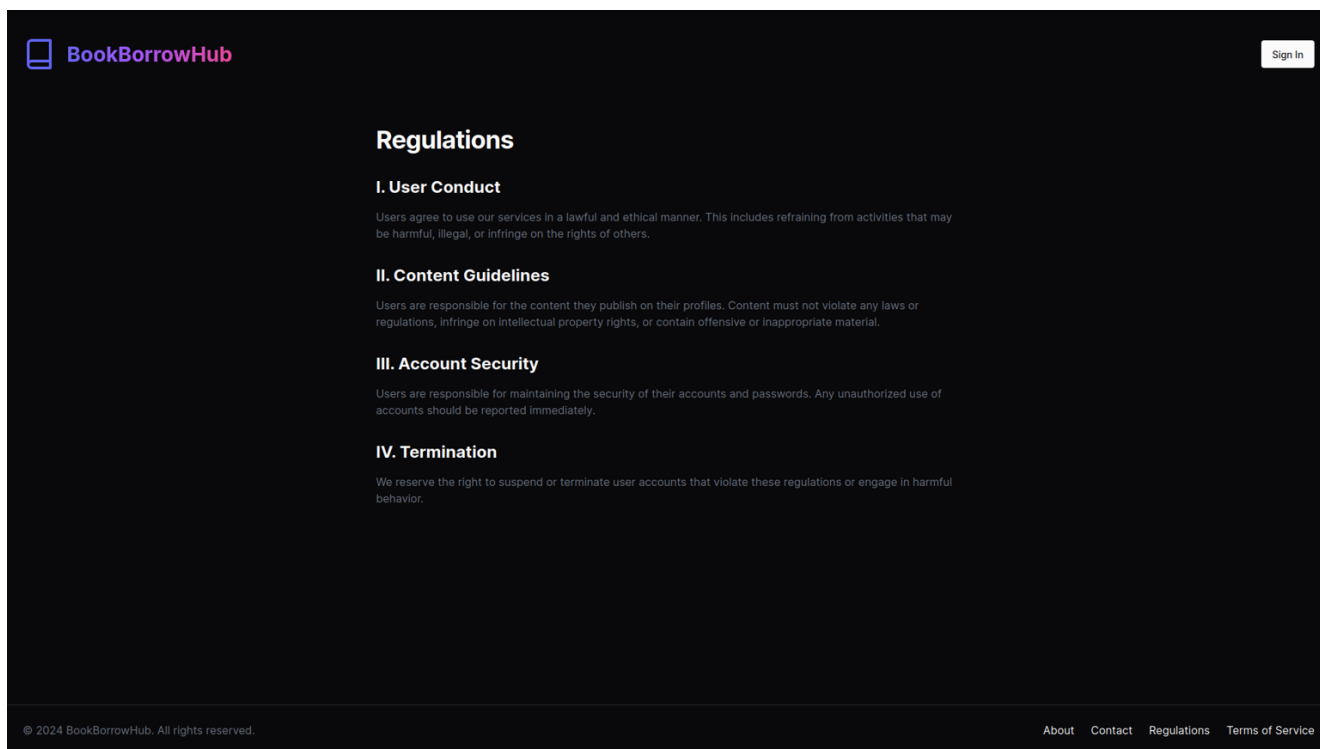
Submit

© 2024 BookBorrowHub. All rights reserved.

[About](#) [Contact](#) [Regulations](#) [Terms of Service](#)

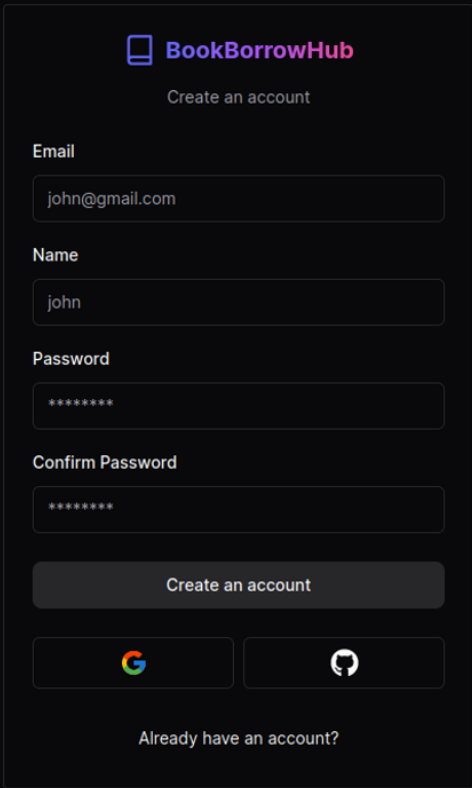
Rysunek 5.6: Panel kontaktu z administracją strony

Na obrazie pokazany jest panel, dzięki któremu użytkownicy mogą skontaktować się z administracją strony w razie problemów z jej działaniem lub pytań. Aby się do niego dostać, trzeba kliknąć zakładkę 'Contact'. Aby wysłać wiadomość, trzeba podać swoją nazwę oraz e-mail.



Rysunek 5.7: Regulamin strony

Zrzut ekranu przedstawia regulamin, który obowiązuje wszystkich użytkowników strony. Osoba zakładająca konto na platformie automatycznie akceptuje warunki korzystania z niej. Aby wyświetlić tę stronę, należy kliknąć zakładkę 'Regulations'.

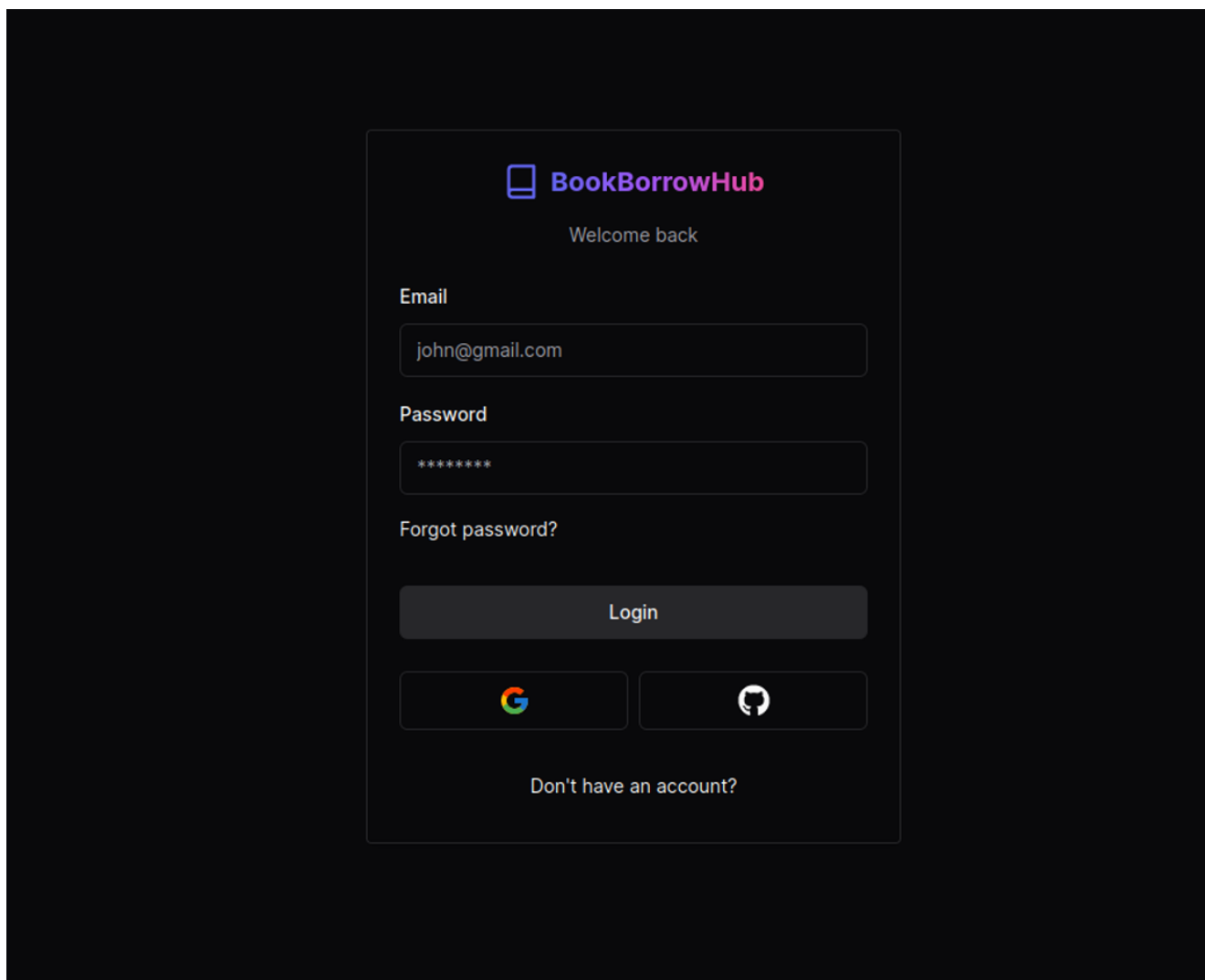


The image shows a registration form for 'BookBorrowHub' on a dark background. The form is centered and contains the following elements:

- Logo:** A book icon followed by the text 'BookBorrowHub' in a light blue and pink font.
- Title:** 'Create an account' in a small, light blue font.
- Fields:**
 - Email:** A text input field containing 'john@gmail.com'.
 - Name:** A text input field containing 'john'.
 - Password:** A password input field with masked characters '*****'.
 - Confirm Password:** A password input field with masked characters '*****'.
- Buttons:**
 - A large, rounded rectangular button labeled 'Create an account' in light blue text.
 - Two smaller buttons below it: one with the Google logo and another with the GitHub logo.
- Link:** A link labeled 'Already have an account?' at the bottom of the form.

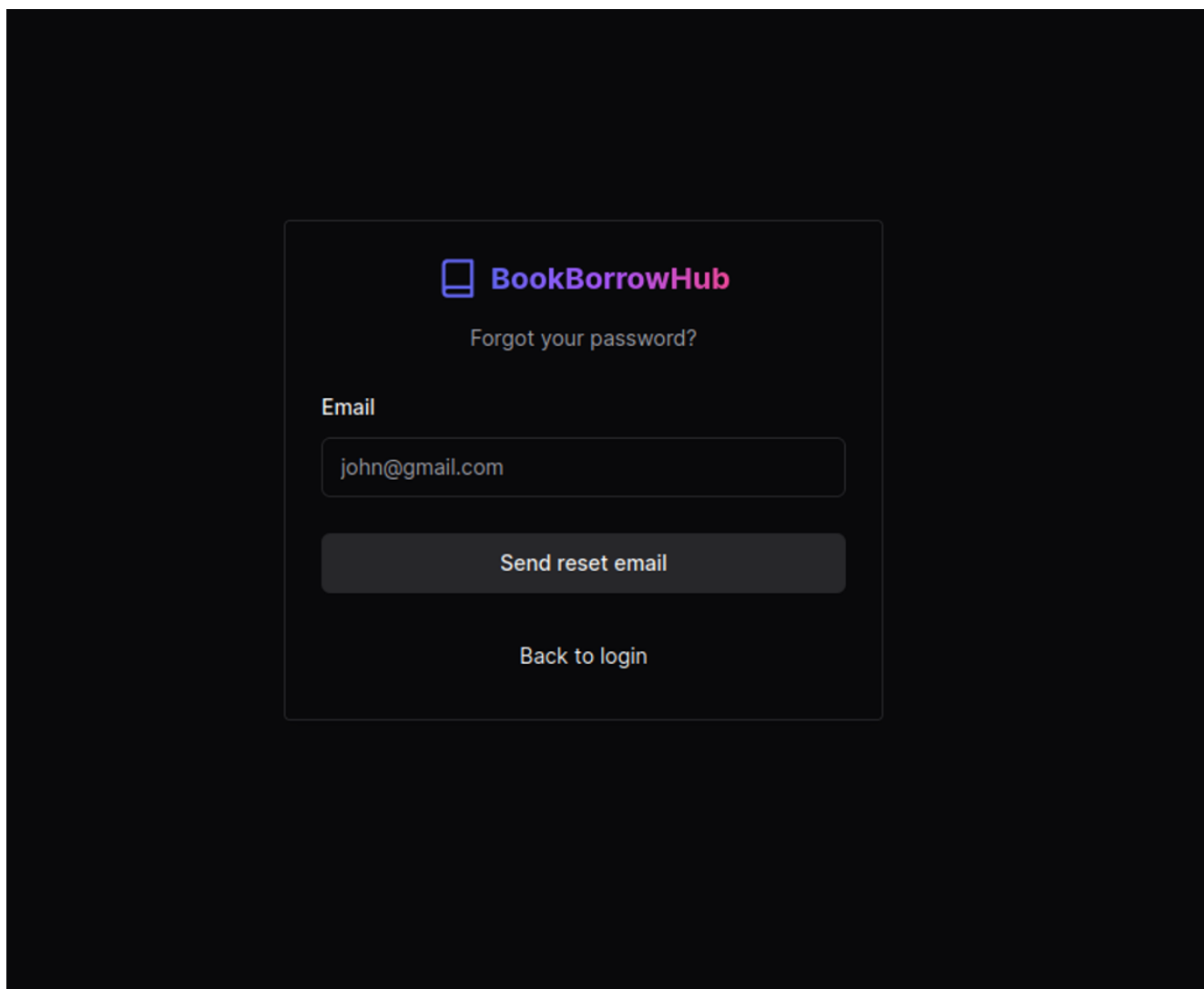
Rysunek 5.8: Panel rejestracji

Aby założyć konto, wymagane jest podanie: e-mailu, nazwy użytkownika oraz hasła. Możliwe jest założenie konta poprzez konto Google lub GitHub. Jeśli użytkownik posiada już konto, może przejść do strony logowania, klikając przycisk 'Already have an account?'.



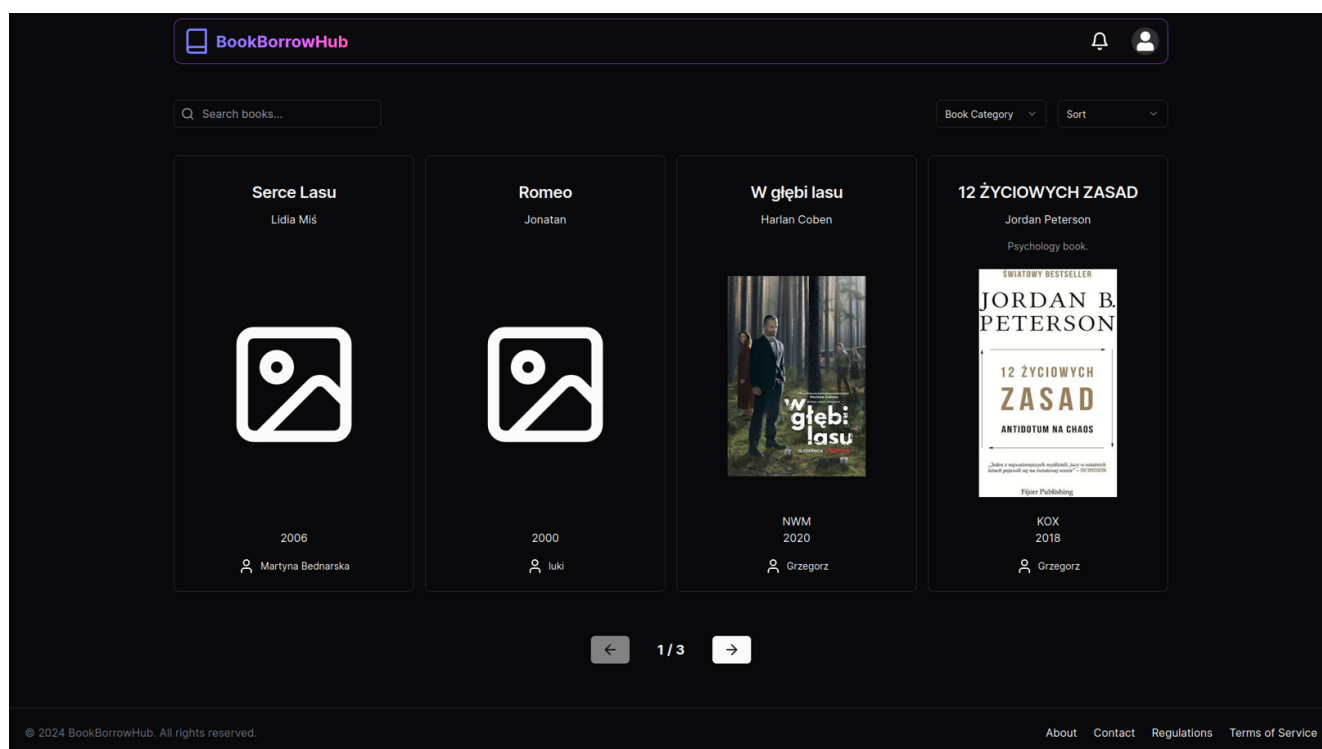
Rysunek 5.9: Panel logowania

Aby się zalogować, trzeba podać e-mail oraz hasło. Istnieje możliwość zresetowania hasła i odzyskania dostępu do konta, klikając w opcję 'Forgot password?'. Można również przejść do panelu rejestracji, klikając przycisk 'Don't have an account?'.



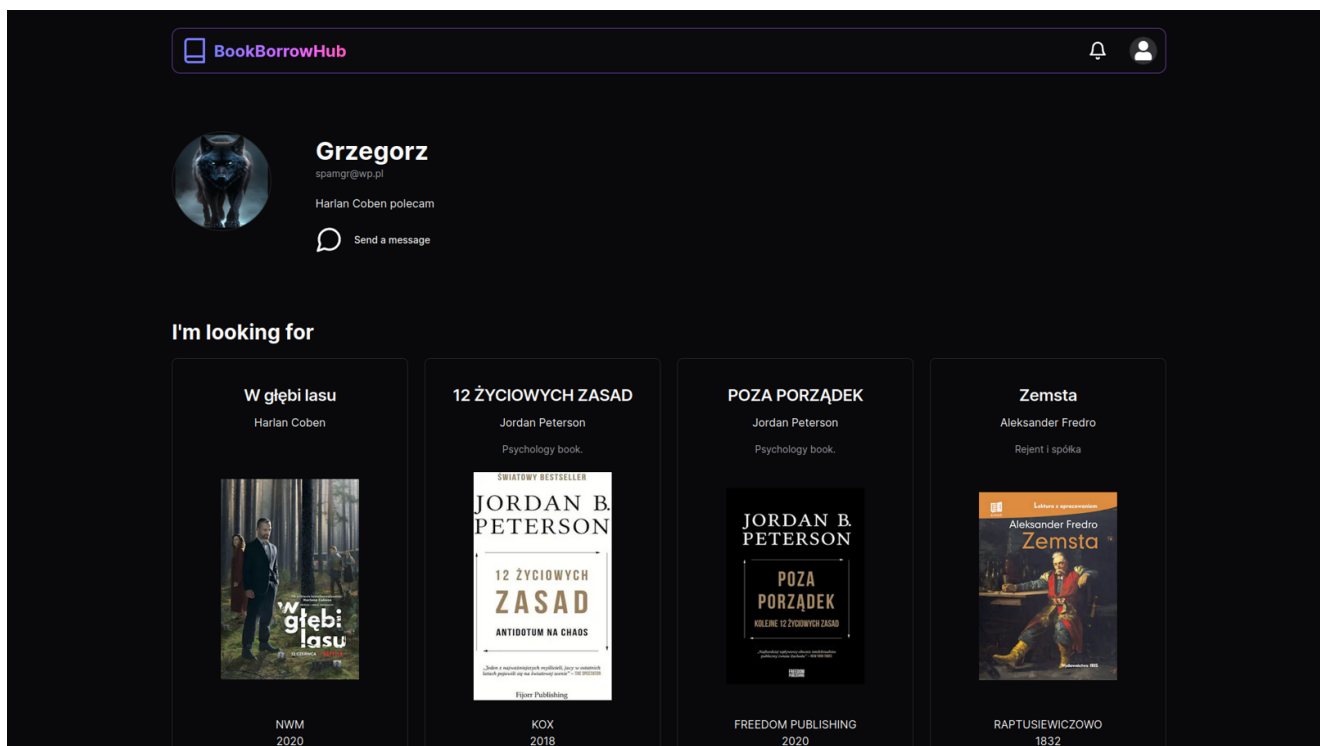
Rysunek 5.10: Panel zmiany hasła

Aby zresetować hasło, należy podać e-mail, którym logujemy się do strony. Po zatwierdzeniu na skrzynkę pocztową zostanie wysłany link umożliwiający ustawienie nowego hasła.



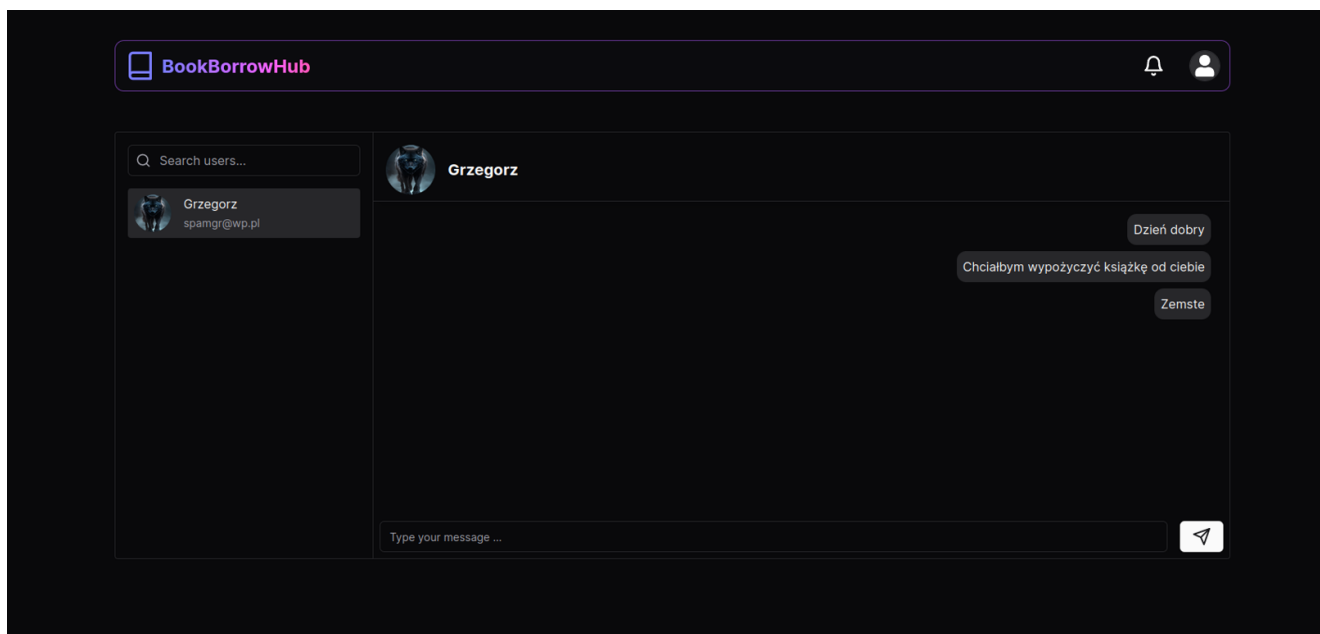
Rysunek 5.11: Panel ofert wymiany użytkowników strony

Strona wyświetla oferty wymiany książek wszystkich użytkowników. Można posortować wyświetlane pozycje oraz wybrać kategorię książek, która nas interesuje. Możliwe jest również wyszukanie książki po nazwie.



Rysunek 5.12: Panel ofert wymiany innego użytkownika

Strona domowa użytkownika wyświetla jego zdjęcie profilowe, e-mail oraz informację, jaką użytkownik chce podać o sobie. Jest również możliwość rozpoczęcia z nim konwersacji. Pod danymi użytkownika wyświetlane są pozycje, które chce on wymienić, oraz pozycje, których szuka.



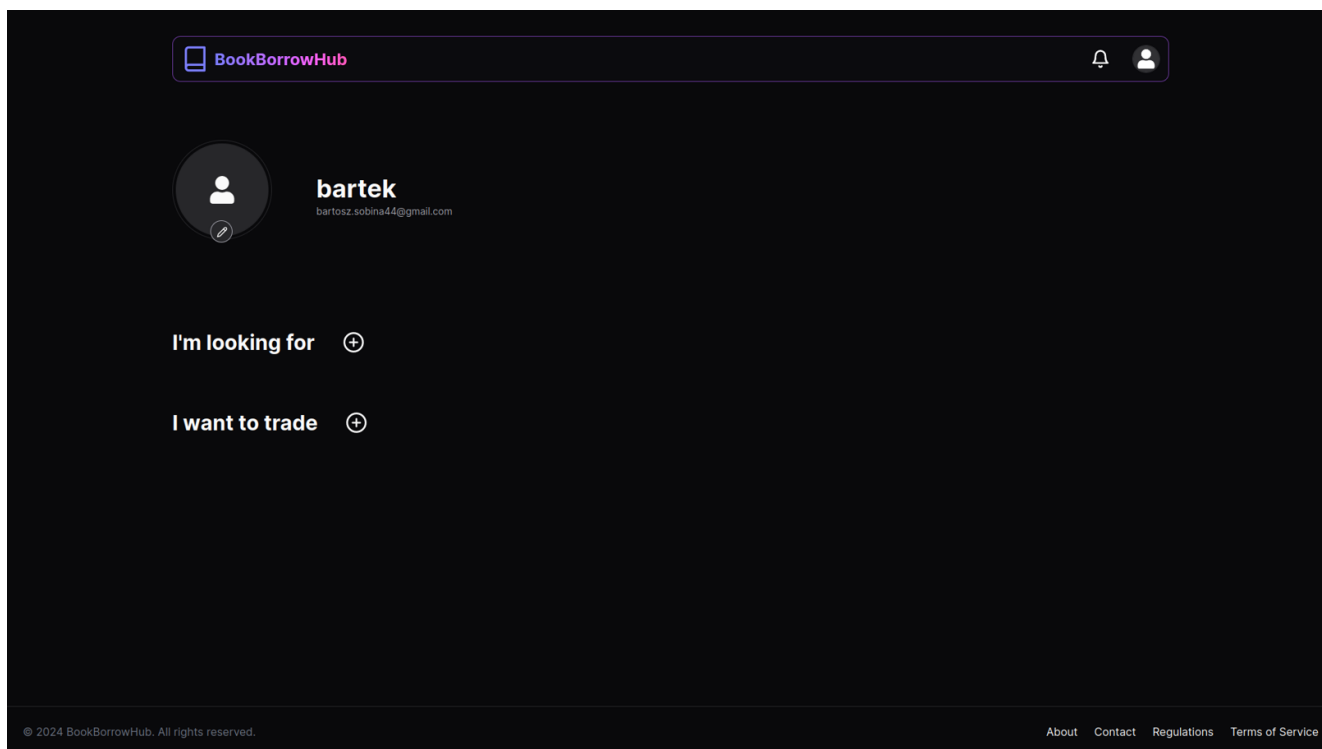
Rysunek 5.13: Panel czatów z innymi użytkownikami

Panel rozmów daje możliwość znalezienia użytkownika, z którym chcemy rozpocząć konwersację lub z którym już prowadziliśmy rozmowę. Wybieramy osobę i możemy przejrzeć historię konwersacji lub wysłać nową wiadomość.

The screenshot shows a web application interface for 'BookBorrowHub'. At the top, there is a header bar with the logo on the left and a notification bell and user profile icon on the right. The main content area is titled 'Settings' and contains several sections: 'Name' with a text input field containing 'bartek'; 'Bio' with a text area containing 'Tell us a little bit about yourself'; 'Notifications' with a toggle switch for 'Message notifications' (currently on) and a description 'Receive a notification when a new user sends you a message.'; 'Email' with a text input field containing 'bartosz.sobina44@gmail.com'; 'Password' with a text input field containing '*****'; and 'New password' with a text input field containing '*****'. At the bottom left of the settings panel is a 'Delete account' button, and at the bottom right is a 'Save' button. The footer of the page contains a row of links: 'Home', 'About Us', 'BookBorrowHub', 'Terms of Service', and 'Privacy Policy'.

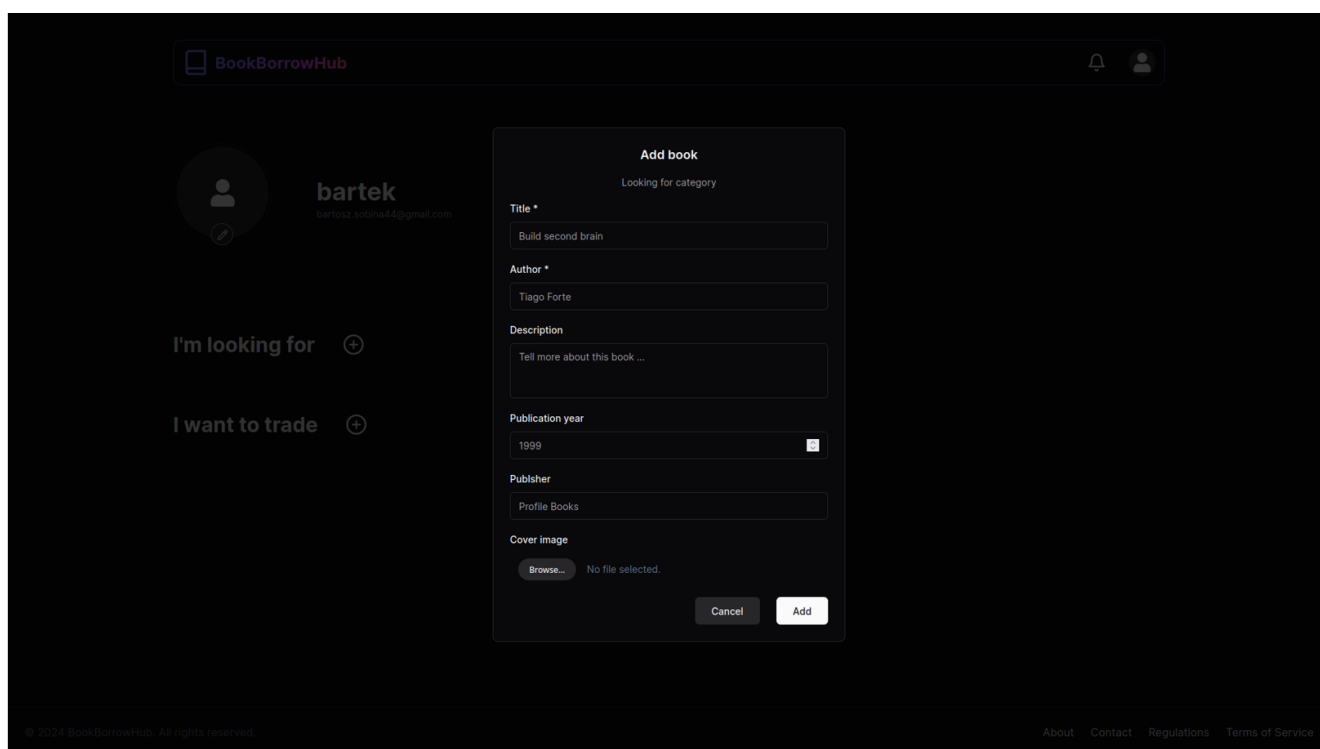
Rysunek 5.14: Panel ustawień użytkownika

W panelu ustawień użytkownika można ustawić informacje takie jak nazwa i bio, a także wyłączyć powiadomienia, zmienić adres e-mail oraz hasło. Jest możliwość usunięcia swojego konta na platformie.



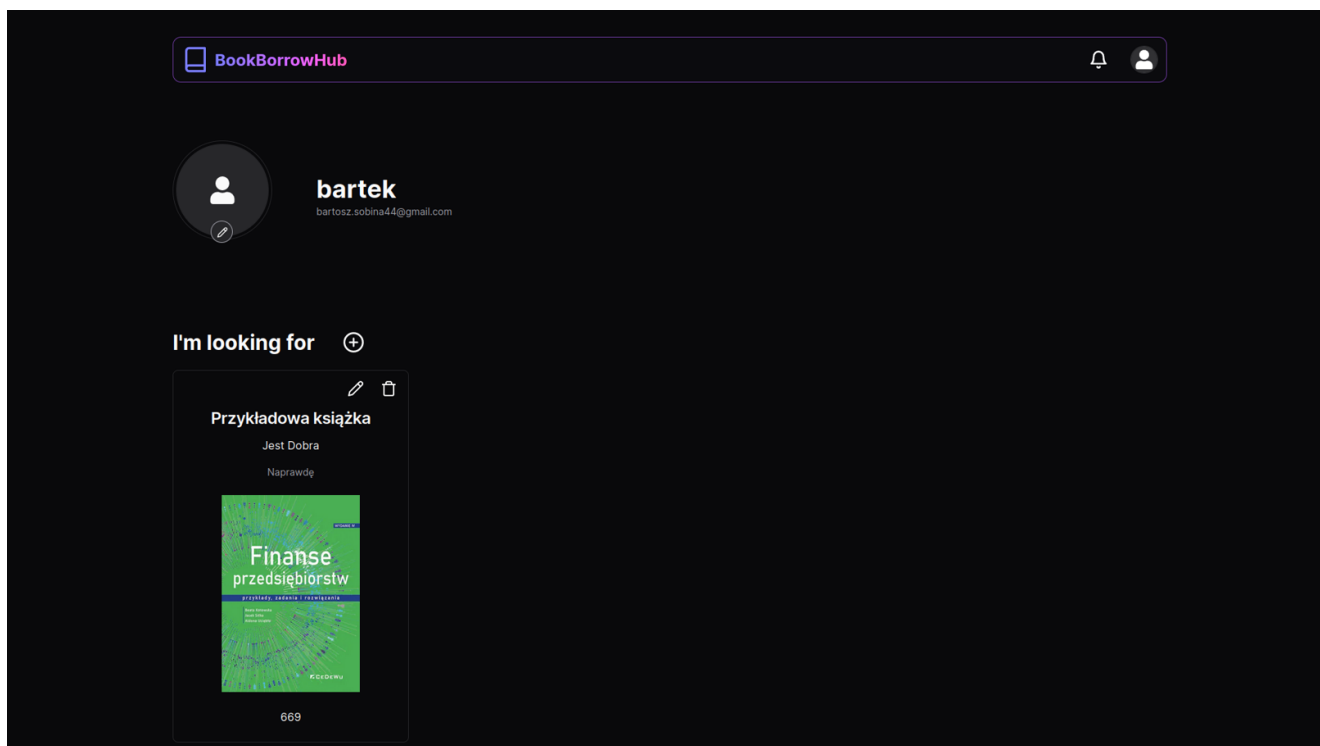
Rysunek 5.15: Panel profilu użytkownika

Tutaj widzimy panel profilu, w którym mamy możliwość zmiany zdjęcia profilowego oraz dodania, edytowania i usunięcia ofert wymiany zalogowanego użytkownika.



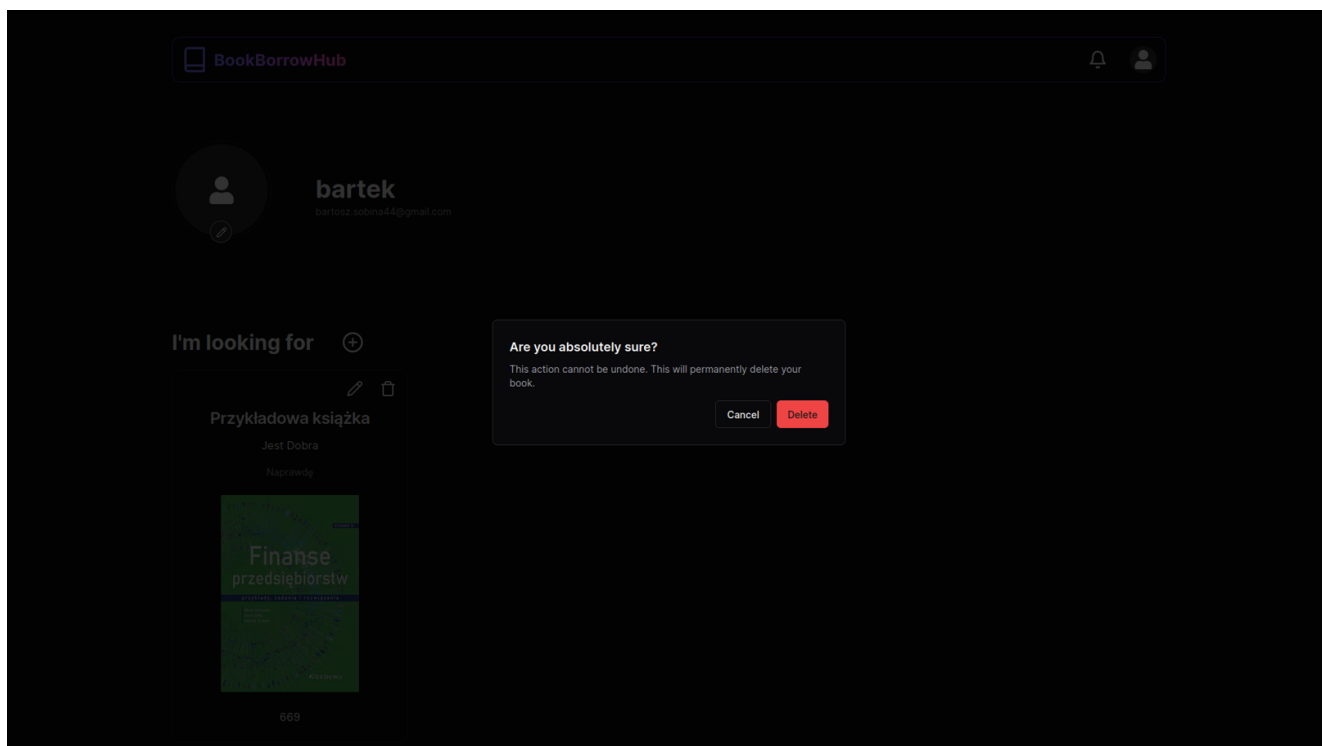
Rysunek 5.16: Panel dodania ogłoszenia/książki

Chcąc dodać ofertę wymiany, trzeba podać informacje takie jak: tytuł, autor, opis, rok wydania, wydawnictwo oraz zdjęcie. Po dodaniu ogłoszenia będzie ono widoczne na stronie ze wszystkimi ofertami oraz na profilu użytkownika.



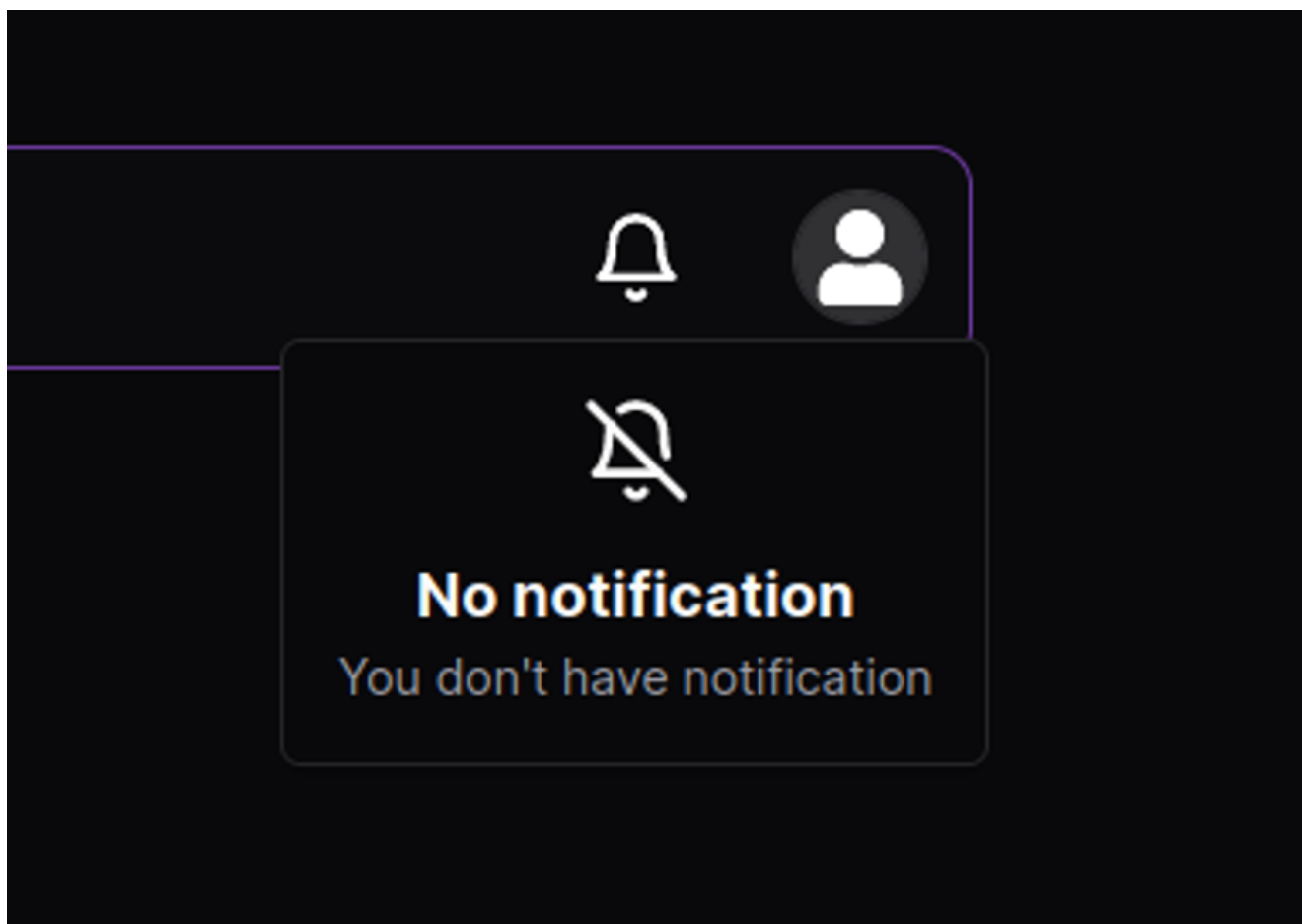
Rysunek 5.17: Panel profilu użytkownika z dodanymi książkami

Dodane książki można na swoim profilu edytować lub usunąć.

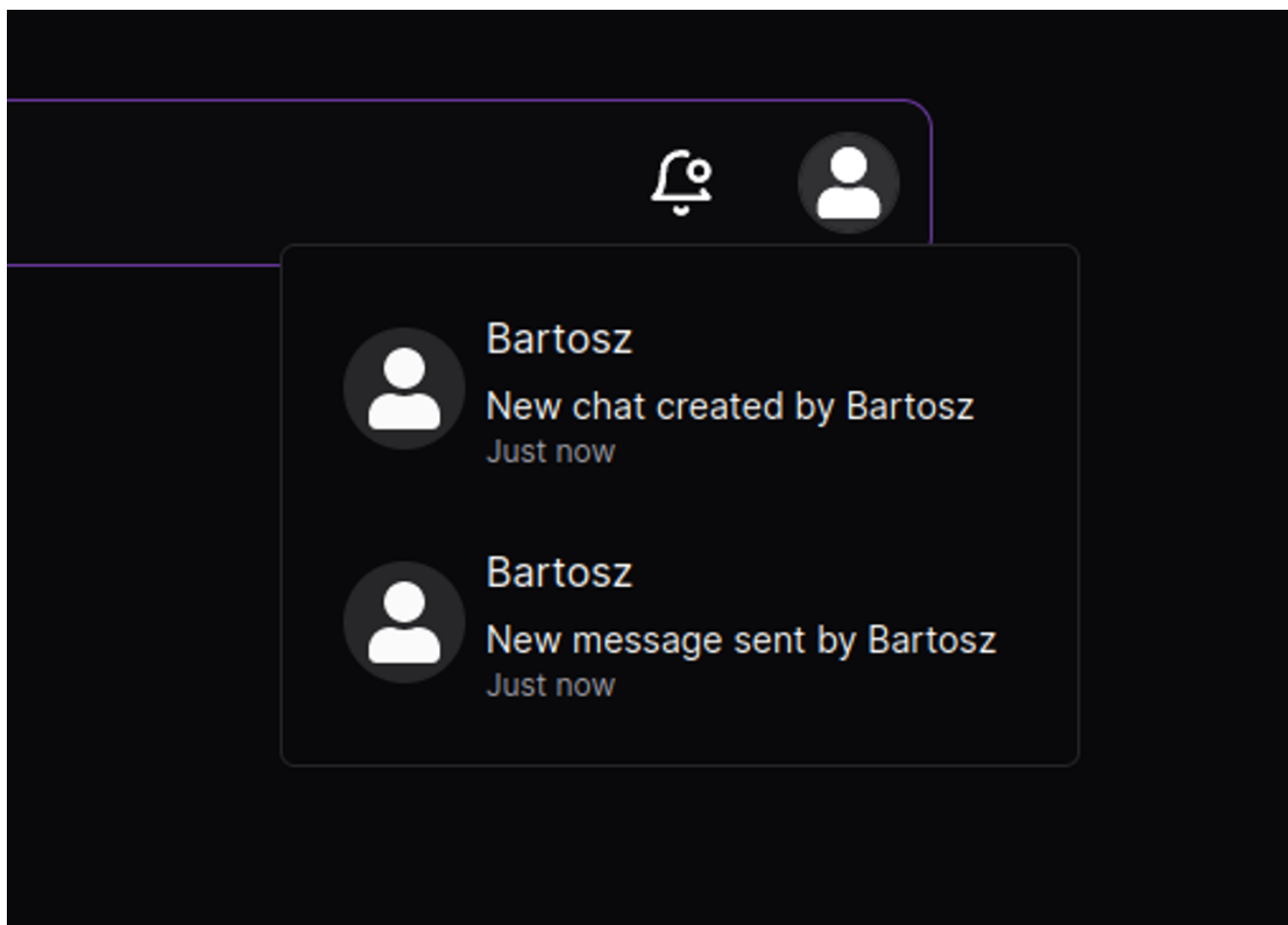


Rysunek 5.18: Okienko usunięcia pozycji ze swoich ofert wymiany

Usunięcie swojej pozycji należy dodatkowo potwierdzić w oknie dialogowym.



Rysunek 5.19: Widget powiadomień (brak powiadomień)



Rysunek 5.20: Wideget powiadomień (kilka powidomień)

Powiadomienia o nowych konwersacjach wyświetlane są w prawym górnym rogu strony. Są przedstawiane w taki sposób, jak widać na obrazie.