



**Bilkent University**

---

**Department of Computer Engineering**

**CS 353**

**Database Management Systems**

**Term Project**

**Design Report**

**BookHub**

**Project Group Members**

Adil Meriç	21802660
Arda Akça Büyük	21802835
Emin Adem Buran	21703279
Onur Oruç	21702381

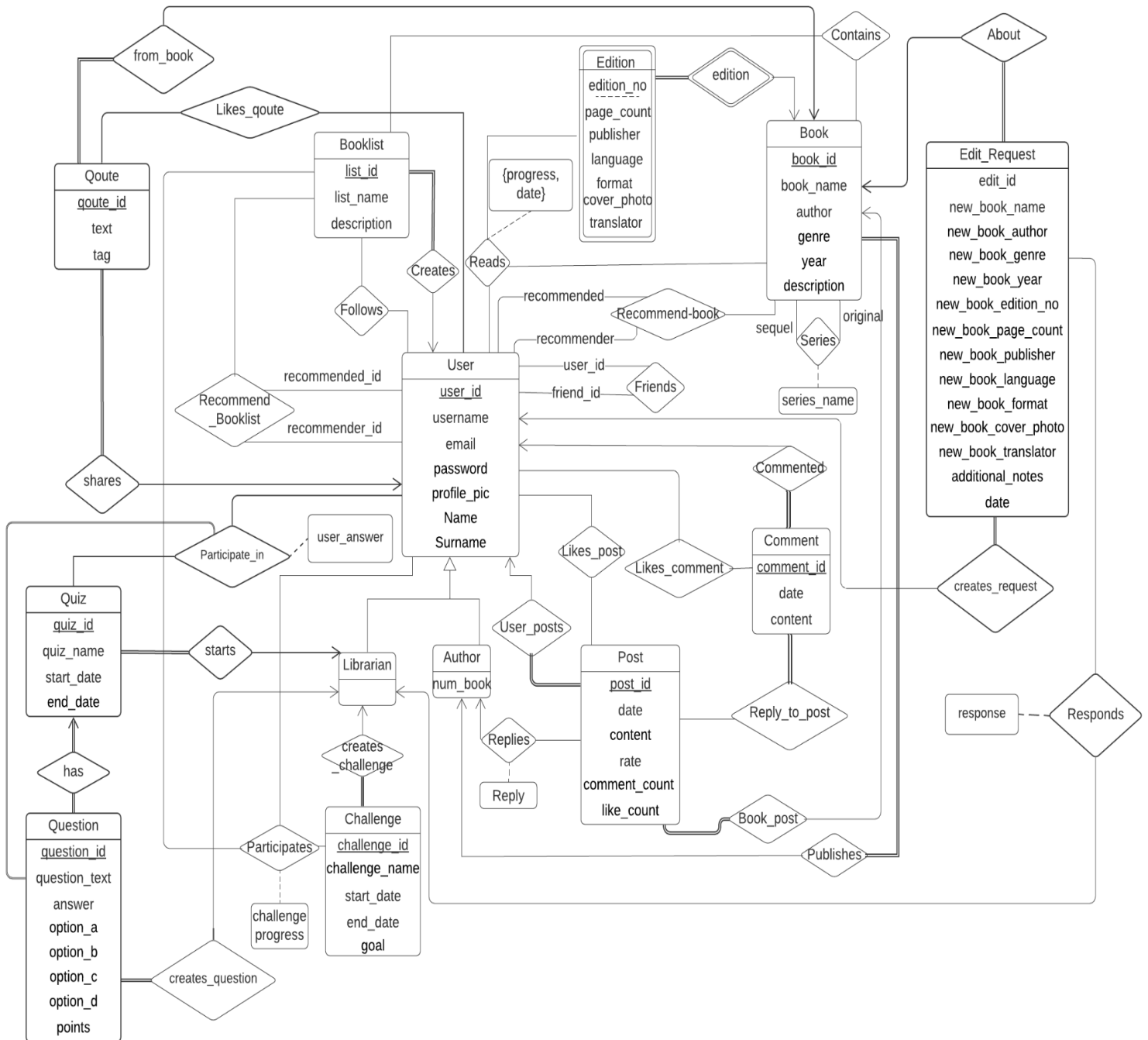
**April 2, 2021**

# Table of Contents

<b>1. Revised ER</b>	<b>4</b>
<b>2. Table Schemas:</b>	<b>5</b>
<b>2.1 Quote</b>	<b>5</b>
<b>2.2 Book</b>	<b>6</b>
<b>2.3 Edition</b>	<b>7</b>
<b>2.4 Quiz</b>	<b>8</b>
<b>2.5 Question</b>	<b>9</b>
<b>2.6 Series</b>	<b>10</b>
<b>2.7 Participate_in</b>	<b>11</b>
<b>2.8 Edit_Request</b>	<b>12</b>
<b>2.9 Booklist</b>	<b>13</b>
<b>2.10 Contains</b>	<b>14</b>
<b>2.11 Follows</b>	<b>15</b>
<b>2.12 Recommend_Booklist</b>	<b>16</b>
<b>2.13 Participate</b>	<b>17</b>
<b>2.14 Challenge</b>	<b>18</b>
<b>2.15 Post</b>	<b>19</b>
<b>2.16 Likes_post</b>	<b>20</b>
<b>2.17 Recommend_Book</b>	<b>21</b>
<b>2.18 Replies</b>	<b>22</b>
<b>2.19 Friends</b>	<b>23</b>
<b>2.20 Comment</b>	<b>24</b>
<b>2.21 Likes_quote</b>	<b>25</b>
<b>2.22 Likes_comment</b>	<b>26</b>
<b>2.23 Reads</b>	<b>27</b>
<b>2.24 User</b>	<b>28</b>
<b>2.25 Librarian</b>	<b>29</b>
<b>2.26 Author</b>	<b>30</b>
<b>2.27 Responds</b>	<b>31</b>
<b>3. User Interface</b>	<b>32</b>

<b>3.1 Login Page</b>	<b>32</b>
<b>3.2 Register Page</b>	<b>33</b>
<b>3.3 Main Page</b>	<b>34</b>
<b>3.4 Challenge Notification</b>	<b>35</b>
<b>3.5 Search Books</b>	<b>36</b>
<b>3.6 Search User</b>	<b>37</b>
<b>3.7 Search Author</b>	<b>38</b>
<b>3.8 Book Page</b>	<b>39</b>
<b>3.9 Add Progress</b>	<b>40</b>
<b>3.10 User Profile</b>	<b>41</b>
<b>3.11 Create Book List</b>	<b>44</b>
<b>3.12 Book List</b>	<b>46</b>
<b>3.13 Author Profile</b>	<b>47</b>
<b>3.14 Create Edition of Book</b>	<b>48</b>
<b>3.15 Publish Book Pop-up</b>	<b>49</b>
<b>3.16 Librarian Profile Page</b>	<b>50</b>
<b>3.17 Request Details Pop-up</b>	<b>51</b>
<b>3.18 Create Challenge Pop-up</b>	<b>53</b>
<b>3.19 Challenge Information Page</b>	<b>54</b>
<b>3.20 Quiz Pop-up</b>	<b>55</b>
<b>3.21 Prepare Quiz</b>	<b>56</b>
<b>3.22 Erroneous Information Request Pop-up</b>	<b>57</b>
<b>3.23 Challenges Page</b>	<b>58</b>
<b>3.24 Post Page</b>	<b>59</b>
<b>3.25 Book Review Page</b>	<b>61</b>
<b>3.26 Quotes Page</b>	<b>62</b>
<b>3.27 Post Review Pop-up</b>	<b>63</b>
<b>3.28 Reply Post</b>	<b>64</b>
<b>3.29 Post Quote Pop-up</b>	<b>65</b>
<b>3.30 Recommend Book</b>	<b>66</b>

# 1. Revised ER



## 2. Table Schemas:

### 2.1 Quote

#### Relational Model

Quote( quote\_id, text, tag, book\_id, user\_id)

**Foreign Key:** book\_id references to Book

user\_id references to User

**Candidate Key:** {quote\_id}

#### Table Definition

**create table Quote(**

quote\_id                    **int** AUTO INCREMENT,  
text                        **varchar(300)** NOT NULL,  
tag                         **varchar(50)** NOT NULL,  
book\_id                    **int** NOT NULL,  
user\_id                    **int** NOT NULL,

**primary key( quote\_id ),**

**foreign key( book\_id )** references Book,

**foreign key( user\_id )** references User

**);**

## 2.2 Book

### Relational Model

Book( book\_id, book\_name, description, author, genre, year, author\_id)

**Foreign Key:** author\_id references to Author

**Candidate Key:** {book\_id}

### Table Definition

**create table Book(**

```
book_id          int AUTO INCREMENT,  
book_name       varchar(75) NOT NULL,  
author          varchar(50) NOT NULL,  
genre           varchar(40) NOT NULL,  
year            int NOT NULL,  
description     varchar(600) NOT NULL,  
author_id       int NOT NULL,  
primary key( book_id ),  
foreign key( author_id ) references Author );
```

## 2.3 Edition

### Relational Model

Edition( edition\_no, book\_id, page\_count, publisher, language, format, cover\_photo, translator)

**Foreign Key:** book\_id references to Book

**Candidate Key:** {(edition\_no, book\_id)}

### Table Definition

**create table Edition(**

```
edition_no          int AUTO INCREMENT,  
book_id             int NOT NULL,  
page_count          int NOT NULL,  
publisher           varchar(40) NOT NULL,  
language            varchar(15) NOT NULL,  
format              varchar(30) NOT NULL,  
cover_photo         varbinary(50) NOT NULL,  
translator          varchar(30),  
primary key( edition_no, book_id ),  
foreign key( book_id ) references Book  
);
```

## 2.4 Quiz

### Relational Model

Quiz( quiz\_id, quiz\_name, start\_date, end\_date, librarian\_id)

**Foreign Key:** librarian\_id references to Librarian

**Candidate Key:** {quiz\_id}

### Table Definition

```
create table Quiz(  
quiz_id          int AUTO INCREMENT,  
quiz_name        varchar(300) NOT NULL,  
librarian_id     int NOT NULL,  
start_date       date NOT NULL,  
end_date         date NOT NULL,  
primary key( quiz_id ),  
foreign key( librarian_id ) references Librarian);
```



## 2.5 Question

### Relational Model

**Question**( question\_id, question\_text, answer, option\_a, option\_b, option\_c, option\_d, quiz\_id, librarian\_id, points)

**Foreign Key:** quiz\_id references to Quiz

librarian\_id references to Librarian

**Candidate Key:** {question\_id}

### Table Definition

**create table** Question(  
question\_id

**int** AUTO INCREMENT,

quiz\_id **int** NOT NULL,

librarian\_id **int** NOT NULL,

question\_text **varchar(300)** NOT NULL,

answer **char(1)** NOT NULL,

option\_a **varchar(200)** NOT NULL,

option\_b **varchar(200)** NOT NULL,

option\_c **varchar(200)** NOT NULL,

option\_d **varchar(200)** NOT NULL,

points **int** NOT NULL,

**primary key**( question\_id),

**foreign key**( quiz\_id ) **references** Quiz,

**foreign key**( librarian\_id ) **references** Librarian);

## 2.6 Series

### Relational Model

series( sequel\_id, original\_id, series\_name)

**Foreign Key:** sequel\_id references to Book

original\_id references to Book

**Candidate Key:** {(sequel\_id, original\_id)}

### Table Definition

```
create table Series(  
sequel_id          int NOT NULL,  
original_id       int NOT NULL,  
series_name       varchar(30) NOT NULL,  
primary key( sequel_id, original_id ),  
foreign key( original_id ) references Book,  
foreign key( sequel_id ) references Book,  
);
```

## 2.7 Participate\_in

### Relational Model

Participate\_in ( quiz\_id, user\_id, question\_id, user\_answer)

**Foreign Key:** quiz\_id references to Quiz

user\_id references to User

question\_id references to Question

**Candidate Key:** {(quiz\_id, user\_id, question\_id)}

### Table Definition

```
create table Participate_in(  
quiz_id      int NOT NULL,  
user_id      int NOT NULL,  
question_id  int NOT NULL,  
user_answer  char(1),  
primary key( quiz_id, user_id, question_id),  
foreign key( quiz_id) references Quiz,  
foreign key( user_id) references User,  
foreign key( question_id) references Question);
```

## 2.8 Edit\_Request

### Relational Model

Edit\_Request( edit\_id, new\_book\_name, new\_book\_author, new\_book\_genre, new\_book\_year, new\_book\_edition\_no, new\_book\_page\_count, new\_book\_publisher, new\_book\_language, new\_book\_format, new\_book\_cover\_photo, new\_book\_translator, additional\_notes, user\_id, book\_id, date)

**Foreign Key:** book\_id references to Book

(new\_book\_edition\_no, book\_id) references to Edition

user\_id references to User

**Candidate Key:** {edit\_id}

### Table Definition

```
create table Edit_Request(  
edit_id                int AUTO INCREMENT,  
new_book_name          varchar(75) ,  
new_book_author        varchar(50) ,  
new_book_genre         varchar(40) ,  
new_book_year          int,  
new_book_edition_no    int NOT NULL,  
new_book_page_count    int,  
new_book_publisher     varchar(40),  
new_book_language      varchar(40),  
new_book_format        varchar(30),  
new_book_cover_photo   varbinary(40),  
new_book_translator    varchar(30),  
additional_notes       varchar(300),  
user_id                int NOT NULL,  
book_id                int NOT NULL,  
date                   date NOT NULL,  
primary key( edit_id ),  
foreign key( new_book_edition_no, book_id ) references Edition,  
foreign key( book_id ) references Book,  
foreign key( user_id ) references User);
```

## 2.9 Booklist

### Relational Model

Booklist (list\_id, list\_name, description, user\_id)

**Foreign Key:** user\_id references to User

**Candidate Key:** {list\_id}

### Table Definition

```
create table Booklist(  
list_id          int AUTO INCREMENT,  
list_name       varchar(30) DEFAULT 'Madness',  
description     varchar(300) DEFAULT 'Description',  
user_id        int NOT NULL,  
primary key( list_id ),  
foreign key( user_id ) references User);
```

## 2.10 Contains

### Relational Model

Contains (list\_id, book\_id)

**Foreign Key:** list\_id references to Booklist

book\_id references to Book

**Candidate Key:** {(list\_id, book\_id)}

### Table Definition

```
create table Contains (  
list_id          int NOT NULL,  
book_id         int NOT NULL,  
primary key( list_id, book_id ),  
foreign key( list_id ) references Booklist,  
foreign key( book_id ) references Book );
```

## 2.11 Follows

### Relational Model

Follows (list\_id, user\_id)

**Foreign Key:** list\_id references to Booklist  
user\_id references to User

**Candidate Key:** {(list\_id, user\_id)}

### Table Definition

```
create table Follows (  
list_id          int NOT NULL,  
user_id         int NOT NULL,  
primary key( list_id, user_id),  
foreign key( list_id ) references Booklist,  
foreign key( user_id) references User);
```

## 2.12 Recommend\_Booklist

### Relational Model

Recommend\_Booklist( list\_id, recommended\_id, recommender\_id )

**Foreign Key:** list\_id references to Booklist

recommended\_id references to User

recommender\_id references to User

**Candidate Key:** {(list\_id, recommended\_id, recommender\_id)}

### Table Definition

```
create table Recommend_Booklist (  
list_id                int NOT NULL,  
recommended_id        int NOT NULL,  
recommender_id        int NOT NULL,  
primary key( list_id, recommended_id, recommender_id ),  
foreign key( list_id ) references Booklist,  
foreign key( recommended_id ) references User,  
foreign key( recommender_id ) references User);
```



## 2.13 Participate

### Relational Model

Participate (list\_id, challenge\_id, user\_id, challenge\_progress)

**Foreign Key:** list\_id references to Booklist

challenge\_id references to Challenge

user\_id references to User

**Candidate Key:** {(list\_id, challenge\_id, user\_id)}

### Table Definition

**create table** Participate(  
list\_id

**int** NOT NULL,

challenge\_id

**int** NOT NULL,

user\_id

**int** NOT NULL,

challenge\_progress

**int** NOT NULL,

**primary key**( list\_id, challenge\_id, user\_id ),

**foreign key**( list\_id ) references Booklist,

**foreign key**( challenge\_id ) references Challenge,

**foreign key**( list\_id ) references Booklist,

);

## 2.14 Challenge

### Relational Model

Challenge (challenge\_id, challenge\_name, start\_date, end\_date, goal, user\_id)

**Foreign Key:** user\_id references to Librarians

**Candidate Key:** {challenge\_id}

### Table Definition

```
create table Challenge(  
challenge_id          int AUTO INCREMENT,  
challenge_name       varchar(40) NOT NULL,  
start_date           date NOT NULL,  
end_date             date NOT NULL,  
goal                 int NOT NULL,  
user id              int NOT NULL,  
primary key( challenge_id ),  
foreign key( user_id ) references Librarian );
```

## 2.15 Post

### Relational Model

Post( post\_id, book\_id, date, content, rate, user\_id, like\_count, comment\_count )

**Foreign Key:** user\_id references to User

book\_id references to Book

**Candidate Key:** {post\_id}

### Table Definition

**create table** Post (

post_id	<b>int</b> AUTO INCREMENT,
book_id	<b>int</b> NOT NULL,
date	<b>date</b> NOT NULL,
content	<b>varchar(300)</b> NOT NULL,
rate	<b>int</b> NOT NULL,
user_id	<b>int</b> NOT NULL,
like_count	<b>int</b> NOT NULL,
comment_count	<b>int</b> NOT NULL,

**primary key**( post\_id ),

**foreign key**( user\_id ) references User,

**foreign key**( book\_id ) references Book);

## 2.16 Likes\_post

### Relational Model

Likes\_post( user\_id, post\_id )

**Foreign Key:** user\_id references to User

post\_id references to Post

**Candidate Key:** {(user\_id, post\_id)}

### Table Definition

```
create table Likes_post(  
user_id                int NOT NULL,  
post_id                int NOT NULL,  
primary key( user_id, post_id ),  
foreign key( user_id ) references User,  
foreign key( post_id ) references Post,  
);
```



## 2.18 Replies

### Relational Model

Replies( post\_id, user\_id, reply)

**Foreign Key:** post\_id references to Post

user\_id references to Author

**Candidate Key:** {post\_id}

### Table Definition

```
create table Replies(  
post_id          int NOT NULL,  
user_id         int NOT NULL,  
reply           varchar(300) NOT NULL,  
primary key( post_id ),  
foreign key( post_id ) references Post),  
foreign key( user_id ) references Author);
```

## 2.19 Friends

### Relational Model

Friends( user\_id, friend\_id)

**Foreign Key:** user\_id references to User

friend\_id references to User

**Candidate Key:** {(user\_id, friend\_id)}

### Table Definition

**create table** Friends(  
user\_id

**int** NOT NULL,

friend\_id

**int** NOT NULL,

**primary key**( user\_id, friend\_id ),

**foreign key**( user\_id ) **references** User),

**foreign key**( friend\_id) **references** User);

## 2.20 Comment

### Relational Model

Comment ( comment\_id, date, content, user\_id, post\_id )

**Foreign Key:** user\_id references to User

post\_id references to Post

**Candidate Key:** {comment\_id}

### Table Definition

```
create table Comment (  
comment_id      int AUTO INCREMENT,  
date            date NOT NULL,  
content         varchar(300) NOT NULL,  
user_id         int NOT NULL,  
post_id        int NOT NULL,  
primary key ( comment_id ),  
foreign key ( user_id ) references User,  
foreign key ( post_id ) references Post );
```



## 2.21 Likes\_qoute

### Relational Model

Likes\_qoute( user\_id, qoute\_id )

**Foreign Key:** user\_id references to User

qoute\_id references to Qoute

**Candidate Key:** {(user\_id, qoute\_id)}

### Table Definition

```
create table Likes_qoute(  
user_id          int NOT NULL,  
qoute_id        int NOT NULL,  
primary key( user_id, qoute_id ),  
foreign key( user_id ) references User,  
foreign key( qoute_id ) references Qoute,  
);
```

## 2.22 Likes\_comment

### Relational Model

Likes\_comment( user\_id, comment\_id )

**Foreign Key:** user\_id references to User

comment\_id references to Comment

**Candidate Key:** {(user\_id, comment\_id)}

### Table Definition

```
create table Likes_comment(  
user_id          int NOT NULL,  
comment_id      int NOT NULL,  
primary key( user_id, comment_id ),  
foreign key( user_id ) references User,  
foreign key( comment_id ) references Comment);
```

## 2.23 Reads

### Relational Model

Reads ( book\_id, user\_id, edition\_no, date, progress)

**Foreign Key:** (book\_id, edition no) references to Edition

user\_id references to User

book\_id references to Book

**Candidate Key:** {(book\_id, user\_id, edition\_no, date)}

### Table Definition

**create table** Reads (

book\_id                           **int** NOT NULL,

edition\_no                       **int** NOT NULL,

user\_id                           **int** NOT NULL,

progress                         **int** NOT NULL,

date                               **timestamp** NOT NULL,

**primary key**( book\_id, user\_id, edition\_no, date),

**foreign key**( book\_id, edition\_no ) references Edition,

**foreign key**( user\_id ) references User,

**foreign key**( book\_id ) references Book

);

## 2.24 User

### Relational Model

User ( user\_id, username, email, password, profile\_pic, name, surname)

**Candidate Key:** {user\_id}

### Table Definition

```
create table User(  
user_id                int AUTO INCREMENT,  
username               varchar(20) NOT NULL,  
name                   varchar(20) NOT NULL,  
surname                varchar(20) NOT NULL,  
email                  varchar(30) NOT NULL,  
password               varchar(30) NOT NULL,  
profile_pic            varbinary(50) NOT NULL,  
primary key( user_id ),  
unique( username)  
);
```

## 2.25 Librarian

### Relational Model

Librarian ( librarian\_id )

**Foreign Key:** librarian\_id references to User

**Candidate Key:** {librarian\_id}

### Table Definition

```
create table Librarian (  
librarian_id          int AUTO INCREMENT,  
primary key( librarian_id ),  
foreign key( librarian_id ) references User);
```

## 2.26 Author

### Relational Model

Author ( author\_id, num\_book)

**Foreign Key:** author\_id references to User

**Candidate Key:** {author\_id}

### Table Definition

```
create table Author (  
author_id                int AUTO INCREMENT,  
num_book                 int NOT NULL DEFAULT 0,  
primary key( author_id ),  
foreign key( author_id ) references User);
```

## 2.27 Responds

### Relational Model

Responds( edit\_id, librarian\_id, response)

**Foreign Key:** edit\_id references to Edit\_request

librarian\_id references to Librarian

**Candidate Key:** {edit\_id}

### Table Definition

```
create table Responds(  
edit_id          int NOT NULL,  
librarian_id    int NOT NULL,  
response        char(1) NOT NULL,  
primary key( edit_id ),  
foreign key( edit_id ) references User,  
foreign key( librarian_id ) references Librarian);
```

## 3. User Interface

### 3.1 Login Page

# BookHub

Login

Username

Password

Show Password

Figure 1: Login page

#### **Login query:**

Select \*

From User

Where username = @username and password = @password



## 3.2 Register Page

# BookHub

## Register

Username

Password  
  
 Show Password

User  Author

Figure 2: Register page

**Sign up query:**

Insert into User (username, email, password, profile\_pic, name, surname)

Values (@username, @email, @password, @profile\_pic, @name, @surname)

### 3.3 Main Page

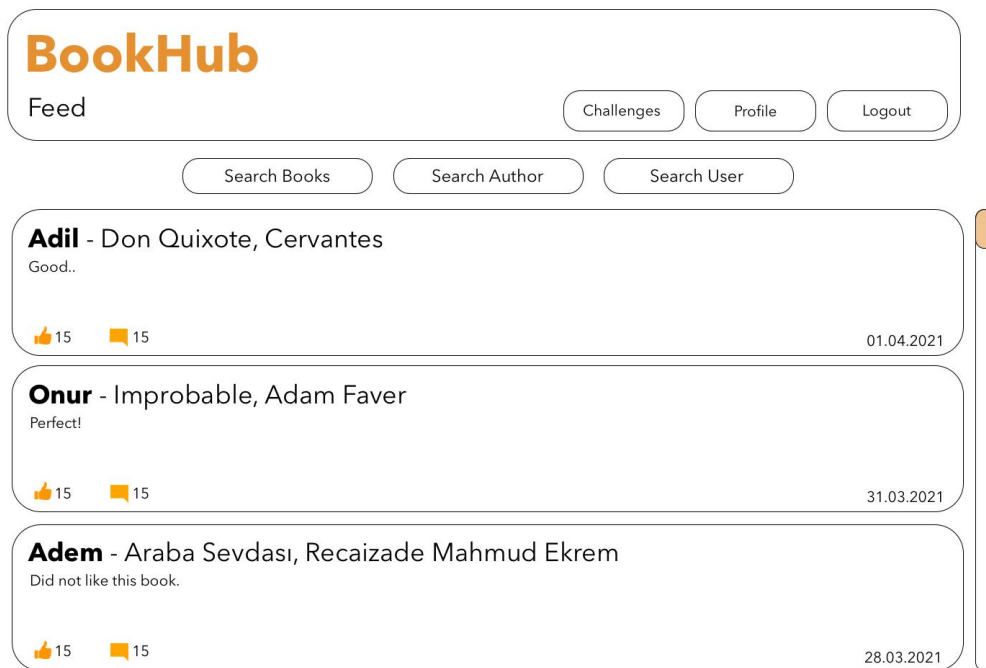


Figure 3: Home page

#### Retrieve friends' posts for feed:

Select \*

From Post P, Friends F

Where F.friend\_id = P.user\_id and F.user\_id = @user\_id

Order by date DESC

### 3.4 Challenge Notification

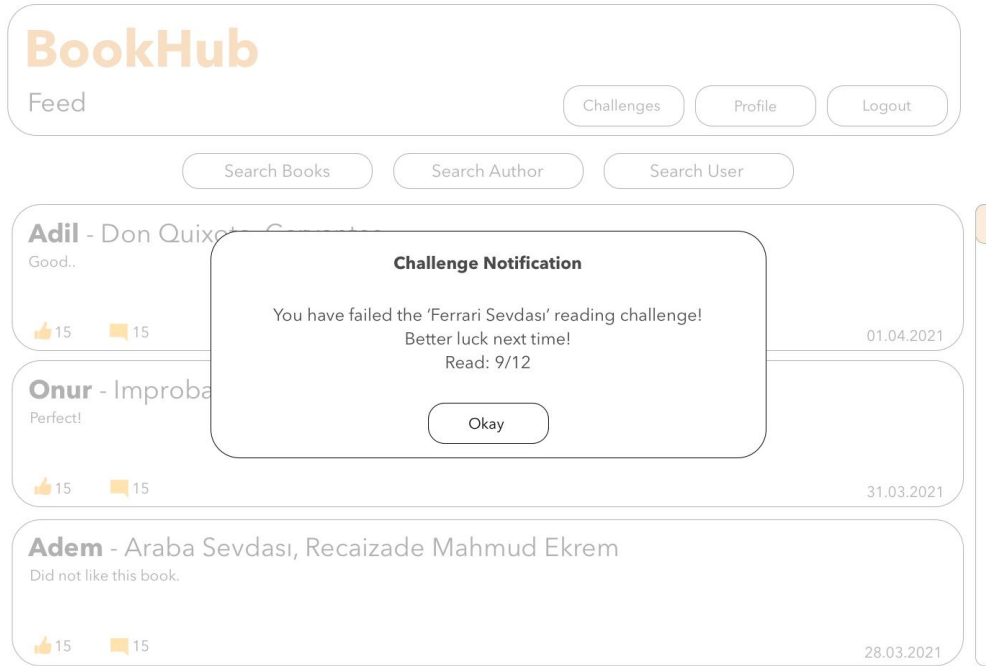


Figure 4: Challenge notification pop-up

#### **Challenge information for participant:**

Select challenge\_id, challenge\_name, challenge\_progress

From Challenge natural join Participate

Where user\_id = @user\_id AND challenge\_id = @challenge\_id

## 3.5 Search Books

# BookHub

Search Books

Profile Logout

Search by Title

Search Clear

---

Search Results

**The Curious Incident of the Dog in the Night-Time**  
by **Mark Haddon**, 2003 - 222 editions  
★★★★☆ Average rating  
★★★★☆ Rate this book

Track

**Night (The Night Trilogy, #1)**  
by **Elie Wiesel, Marion Wiesel**, 1956 - 151 editions  
★★★★☆ Average rating  
★★★★☆ Rate this book

Track

**The Night Circus**  
by **Erin Morgenstern**, 2011 - 150 editions  
★★★★☆ Average rating  
★★★★☆ Rate this book

Track

Figure 5: Search screen that allows users to search for books by title, author, genre, year

### Search book:

Select \*

From Book

Where book\_name like '@search%'

## 3.6 Search User

**BookHub**  
Search User Profile Logout

Search by Name  
  
Search Clear

Search Results

- Adil Meriç**  
Read: 51 books  
Lists: 13  
Ratings: 27  
Posts: 2 ✓ Friends
- Adil Bayramoğlu**  
Read: 2 books  
Lists: 0  
Ratings: 1  
Posts: 0 Add Friend
- Adile Kafa**  
Read: 122 books  
Lists: 23  
Ratings: 78  
Posts: 35 Add Friend

Figure 6: Search screen that allows users to search for users by name

### Search user:

Select \*

From User

Where user\_id <> @user\_id AND (name like '@search%' OR surname like '@search%')

### 3.7 Search Author

**BookHub**  
Search Author Profile Logout

Search by Name  
  
Search Clear

Search Results

- George R.R. Martin**  
Books: 12  
Replies: 3
- George Orwell**  
Books: 21  
Replies: 0
- Jim George**  
Books: 37  
Replies: 10

Figure 7: Search author screen that allows users to search for authors by name, book name, genre, and year

#### Search Author:

Select \*

From Author A, User U

Where  $A.author\_id = U.user\_id$  and  $(U.name \text{ like } '@search\%' \text{ OR } U.surname \text{ like } '@search\%')$

## 3.8 Book Page

### BookHub

Book Profile Logout

**1984** by **George Orwell** Your progress: 217


Genre: **Dystopia** Prepare Quiz Quiz Recommend

1949

Among the seminal texts of the 20th century, Nineteen Eighty-Four is a rare work that grows more haunting as its futuristic purgatory becomes more real. Published in 1949, the book offers political satirist George Orwell's nightmarish vision of a totalitarian, bureaucratic world and one poor stiff's attempt to find individuality. The brilliance of the novel is Orwell's prescience of modern life—the ubiquity of television, the distortion of the language—and his ability to construct such a thorough version of hell. Required reading for students since it was published, it ranks among the most terrifying novels ever written. Erroneous Info Request Quotes Reviews

Editions

Edition	Publisher	Page Count	Format	Language
250	<b>Plume</b>	339	Mass Market Paperback	EN
249	<b>Penguin Books</b>	311	Kindle Edition	EN
248	<b>Houghton Mifflin Harcourt</b>	237	Paperback	EN
247	<b>New American Library</b>	328	Paperback	EN

Progress Steps 

Date	Edition	Progress
10.02.2021	249	217
09.02.2021	249	156
23.01.2021	249	97
20.01.2021	249	45

Figure 8: Book screen that lists the book description, information about its edition, user progress.

### For book information:

Select \*

From Book

Where book\_id = @book\_id

### For edition information:

Select \*

From Edition

Where book\_id = @book\_id

### For progress:

Select \*

From Reads

Where book\_id = @book\_id and user\_id = @user\_id

### 3.9 Add Progress

The screenshot shows the BookHub app interface. At the top, the 'BookHub' logo is in orange. Below it, the word 'Book' is displayed. On the right, there are 'Profile' and 'Logout' buttons. The main content area features the book title '1984 by George Orwell' and the genre 'Dystopia'. A progress indicator shows 'Your progress: 217' and a 'Recommend' button. A descriptive paragraph about the book is visible. Below the text is an 'Editions' table with columns for Edition, Publisher, and Language. A pop-up window titled 'Add Progress' is overlaid on the page, containing a form with fields for 'Date', 'Edition', and 'I am now at page' (with a 'Page No' sub-field). A 'Done' button is at the bottom of the pop-up. Below the editions table is a 'Progress Steps' section with a pencil icon and a table listing previous progress entries.

Edition	Publisher	Language
250	Plume	Paperback EN
249	Penguin Books	EN
248	Houghton Mifflin	EN
247	New American Library	Paperback EN

Date	Edition	Progress
10.02.2021	249	217
09.02.2021	249	156
23.01.2021	249	97
20.01.2021	249	45

Figure 9: Pop-up window for a user to add his/her progress about a specific book

#### **Progress Insertion:**

Insert into Reads

Values(@book\_id, @user\_id, @edition\_no, @progress, @date)



## 3.10 User Profile

# BookHub

Profile Profile Logout

### Arda Akça Büyük - User

#### Currently Reading

Last Update	Title, Author	Edition	Page	
01.04.2021	<b>1984, Geore Orwell</b>	174	54	
25.03.2021	<b>Oliver Twist, Charles Dickens</b>	45	126	
23.03.2021	<b>Fahrenheit 451, Ray Bradbury</b>	7	143	

#### Read

Finished	Title, Author		
10.02.2021	<b>Heidi, Johanna Spyri</b>		<span>Post Review</span>
25.01.2021	<b>Animal Farm, George Orwell</b>		<span>Post Review</span>
23.01.2021	<b>To Kill a Mockingbird, Harper Lee</b>		<span>Post Review</span>

#### Posts

Last Update	Title, Author	Likes	Comments	Rating
10.02.2021	<b>Heidi, Johanna Spyri</b>	27	3	3/5
25.01.2021	<b>Animal Farm, George Orwell</b>	233	10	4/5
23.01.2021	<b>To Kill a Mockingbird, Harper Lee</b>	101	14	5/5

#### Book Lists

Last Update	Name	Books	
01.04.2021	<b>Classics</b>	21	
25.03.2021	<b>Special</b>	3	
23.03.2021	<b>Readlist</b>	55	<span>Create</span>

#### Friends

Add

Name
<b>Adil Meriç</b>
<b>Emin Adem Buran</b>
<b>Onur Oruç</b>

#### Challenges Succeeded

Title	Book Count
Bozcaada'nın En Hızlı Üzümü	35

Figure 10: User profile page listing currently reading book(s), read books, user's ratings for books, his/her posts, friends challenges, and booklists

**For friends:**

Select \*

From Friends

Where user\_id = @user\_id

**For successful challenges:**

Select \*

From Challenge natural join Participate

Where user\_id = @user\_id AND goal <= challenge\_progress

**Currently reading:**

Select \*

From Reads R

Where R.user\_id = @user\_id AND progress < ( Select page\_count

From Edition

Where edition\_no = R. edition\_no

AND book\_id = R.book\_id)

**Finished Books:**

From Reads R

Where R.user\_id = @user\_id AND progress = ( Select page\_count

From Edition

Where edition\_no = R. edition\_no

AND book\_id = R.book\_id)

**For Posts:**

Select \*

From Post

Where user\_id = @user\_id

Order by date DESC;

**For Booklist:**

Select \*

From Booklist

Where user\_id = @user\_id

## 3.11 Create Book List

Create Book List

List Name

Select Books

Search by Title  Search by Author  Search by Genre  Search by Year

Search Results

**The Curious Incident of the Dog in the Night-Time**  
by **Mark Haddon**, 2003 - 222 editions  
★★★★☆ Average rating  
☆☆☆☆☆ Rate this book

**Night (The Night Trilogy, #1)**  
by **Elie Wiesel, Marion Wiesel**, 1956 - 151 editions  
★★★★☆ Average rating  
☆☆☆☆☆ Rate this book

Figure 11: The screen that allows a user to search for books and create a book list with those books

### **Create book list:**

Insert into Booklist (list\_name, description, user\_id)

Values (@list\_name, @description, @user\_id)

### **Add book to book list:**

Insert into Contains

Values (@list\_id, @book\_id)

### **Search by Title:**

select \*

from Book

where title like '@Title%'

**Search by Author:**

select \*  
from Book  
where author like '@Author%'

**Search by Genre:**

select \*  
from Book  
where genre like '@Genre%'

**Search by Year:**

select \*  
from Book  
where year = @Year

## 3.12 Book List

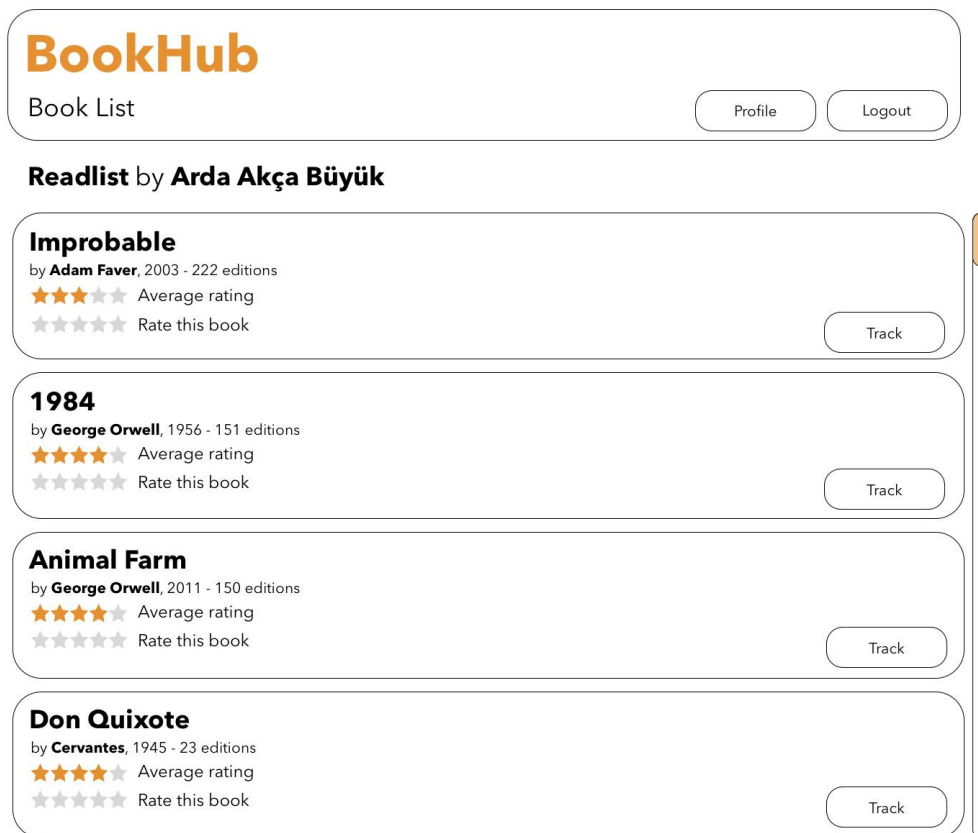


Figure 12: The page that lists books in a user's booklist

### Information about the books in the list:

Select \*

From Contains C, Book B, Edition E

Where C.list\_id = @list\_id AND C.book\_id = B.book\_id

AND e.edition\_no = B.edition\_no

### 3.13 Author Profile

The screenshot shows the 'BookHub' author profile for George R.R. Martin. At the top, there is a navigation bar with the 'BookHub' logo, a 'Profile' link, and buttons for 'Switch to User', 'Profile', and 'Logout'. Below this, the author's name 'George R.R. Martin - Author' is displayed. A 'Published Books' section includes a 'Publish' button and a table of books. The table has columns for 'Year' and 'Title'. Three books are listed: 'A Dance with Dragons' (2011), 'A Feast for Crows' (2005), and 'Game of Thrones' (1996). Each book entry has 'List Reviews' and 'Create Edition' buttons.

Year	Title	List Reviews	Create Edition
2011	<b>A Dance with Dragons</b>	List Reviews	Create Edition
2005	<b>A Feast for Crows</b>	List Reviews	Create Edition
1996	<b>Game of Thrones</b>	List Reviews	Create Edition

Figure 13: Author profile page listing the books published by the author

#### Retrieve Author's books:

Select \*

From Book

Where author\_id = @author\_id

### 3.14 Create Edition of Book

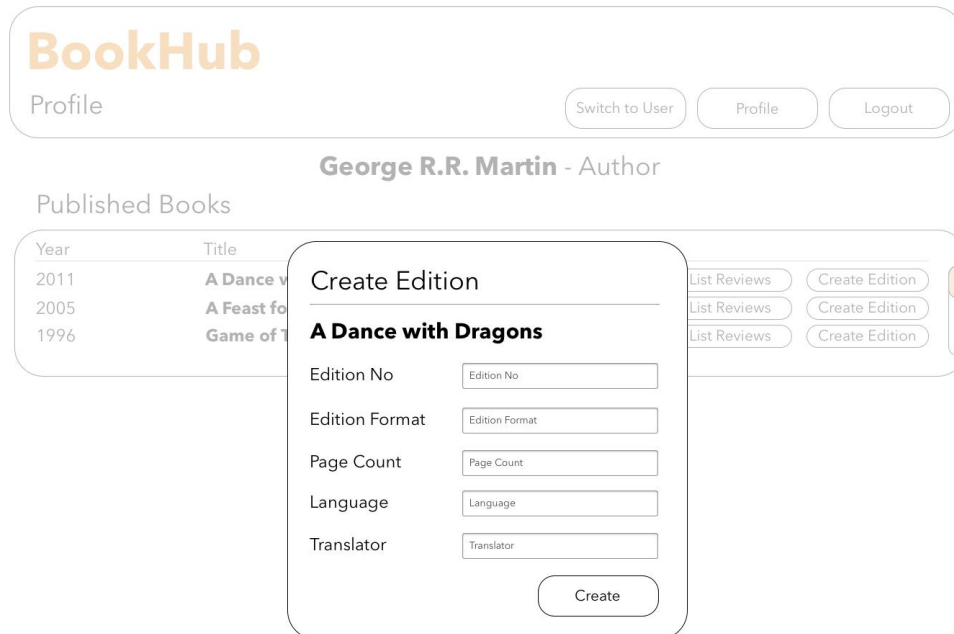


Figure 14: Pop-up window that allows librarians to create an edition of a book

#### **Create Edition:**

Insert into Edition

Values (@edition\_no, @book\_id, @page\_count, @publisher, @language, @format, @cover\_photo, @translator);



### 3.15 Publish Book Pop-up

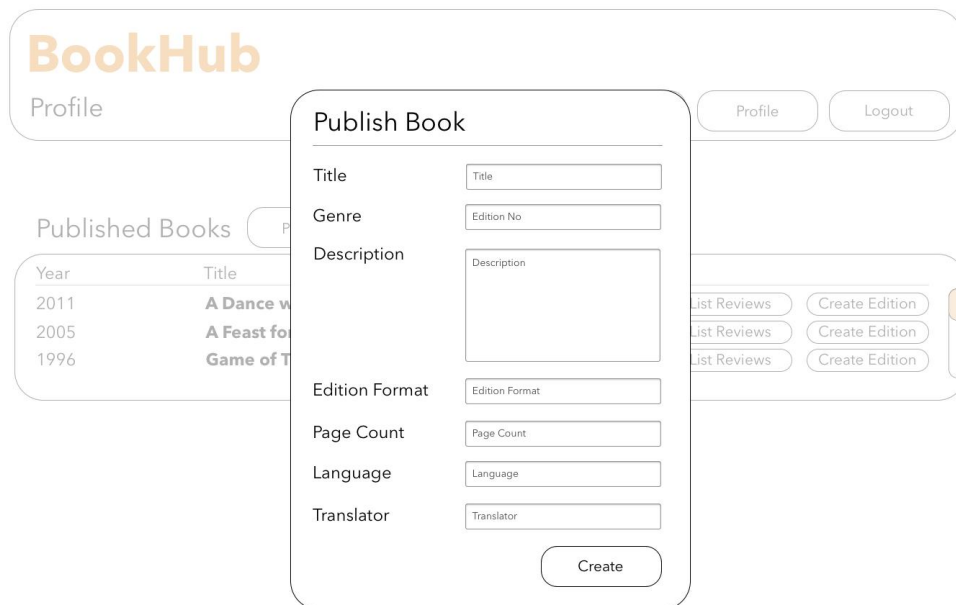


Figure 15: The screen that allows authors to publish new books

#### **Publish a book:**

Insert into Book (book\_name, author, description, genre, year, author\_id)

Values (@book\_name, @author, @description, @genre, @year, @author\_id)

Update Author

Set num\_book = num\_book + 1

Where author\_id = @author\_id

## Publish a book:

Update Author

```
set book_count = book_count +1
```

```
where author_id = ( select author_id
```

```
from Book
```

```
Where book_id = @book_id)
```

## 3.16 Librarian Profile Page

### BookHub

Profile

Switch to User

Profile

Logout

**İlyas Alperen Büyük** - Librarian

Erroneous Information Correction Requests

Date	User	Book	
07.03.2021	Arda Akça Büyük	A Dance with Dragons	<a href="#">Details</a>
07.02.2021	Adil Meriç	Night (The Night Trilogy, #1)	<a href="#">Details</a>
07.01.2021	Elif Özer	1984	<a href="#">Details</a>

Reading Challenges

Create

Start Date	Deadline	Title	Type	Book Count	
07.01.2021	01.04.2021	Minik Kitap Kurtları	Booksss	30	<a href="#">Details</a>
07.01.2021	23.02.2021	Kitap Canavarları	Booksss	10	<a href="#">Details</a>
07.01.2021	14.01.2021	Küçük Dostoyevksiler	Author / Speedy	3	<a href="#">Details</a>

Figure 16: Librarian profile page that allows librarians to review erroneous information correction requests and to list and create reading challenges

### **For edit requests:**

Select \*

From (Select \*

From Edit\_Request E, Respond R

Where E.edit\_id = R.edit\_id AND R.response <> "A") as ER,

User U, Book B

Where U.user\_id = ER.user\_id AND B.book\_id = ER.book\_id

### **Retrieve created challenges:**

Select \*

From Challenge

Where user\_id = @librarian\_id

## **3.17 Request Details Pop-up**

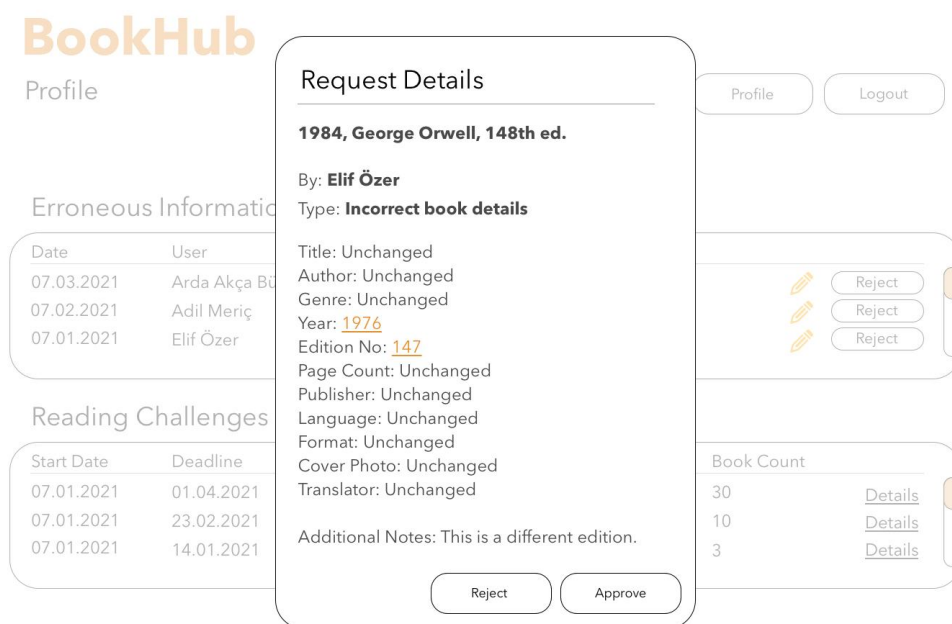


Figure 17: Pop-up window that allows librarians to review details about correction requests

**For information:**

Select \*

From (Select \*

From Edit\_Request E, Respond R

Where E.edit\_id = R.edit\_id AND R.response <> "A") as ER,

User U, Book B

Where U.user\_id = ER.user\_id AND B.book\_id = ER.book\_id

**If approve:**

Update Book

Set book\_name = ER.new\_book\_name, genre = ER.new\_book\_genre, year =  
ER.new\_book\_year

From Edit\_Request ER

Where ER.book\_id = @book\_id AND Book.book\_id = @book\_id

Update Edition

Set edition\_no = ER.new\_book\_edition\_no, page\_count = ER.new\_book\_page\_count, publisher  
= ER.new\_book\_publisher, language = ER.new\_book\_language, format = ER.new\_book.format,  
cover\_photo = ER.new\_book\_cover\_photo, translator = ER.new\_book\_translator

From Edit\_Request ER

Where Edition.book\_id = @book\_id AND ER.book\_id = @book\_id

**If reject:**

Insert Into Respond (edit\_id, librarian\_id, respond)

Values(@edit\_id, @librarian\_id, 'R')

## 3.18 Create Challenge Pop-up

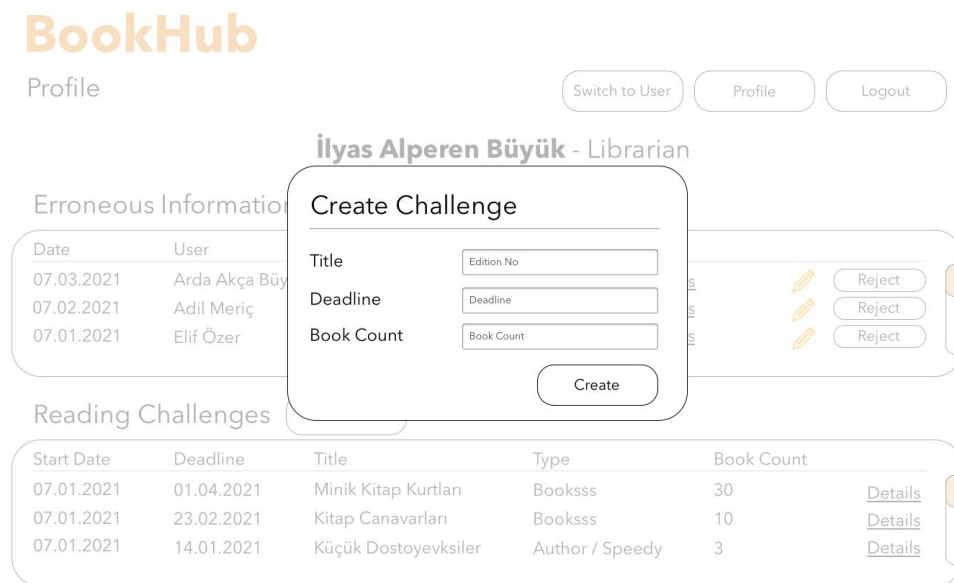


Figure 18: Pop-up screen that allows librarians to create challenges

### **Create Challenge:**

Insert into Challenge (challenge\_name, start\_date, end\_date, goal, user\_id)

Values (@challenge\_name, @start\_date, @end\_date, @goal, @user\_id)

### 3.19 Challenge Information Page

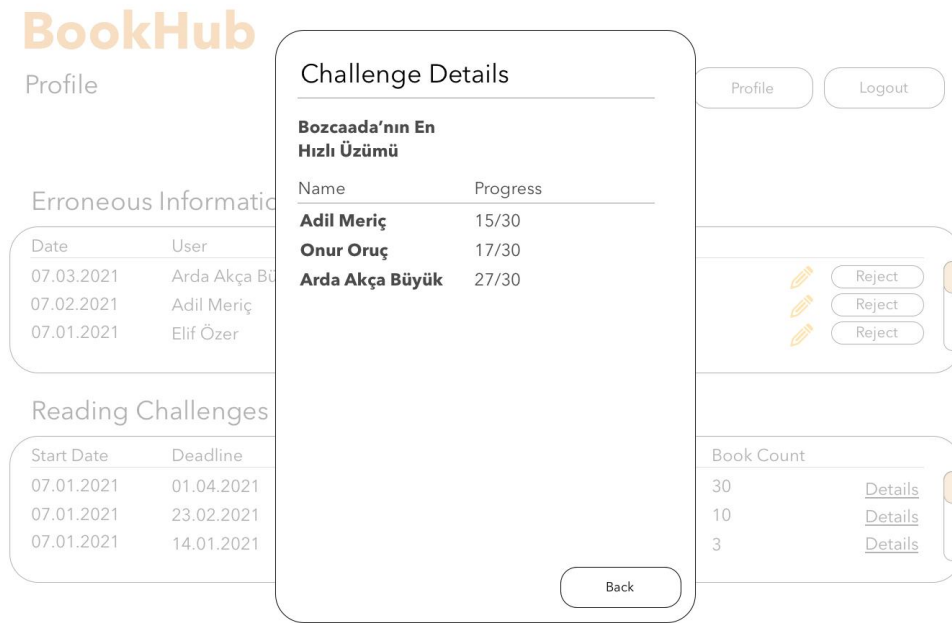


Figure 19: Pop-up screen listing a challenge details

#### **Retrieve participants and their names:**

Select \*

From Challenge C, Participate P, User U

Where C.challenge\_id = P.challenge\_id AND P.user\_id = U.user\_id

## 3.20 Quiz Pop-up

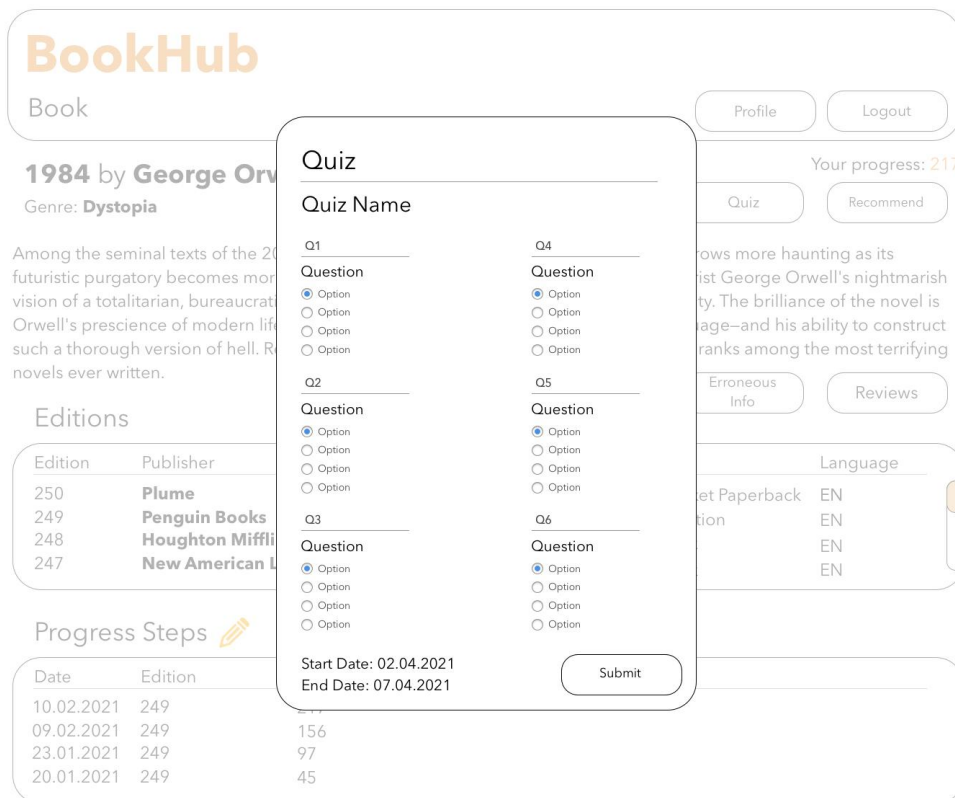


Figure 20: Quiz screen listing the questions related to a book

### Retrieving questions:

Select \*

From Question Que, Quiz Qui

Where Que.quiz\_id = Qui.quiz\_id

### Submit Answer:

Insert into Participate\_in

Values(@quiz\_id, @user\_id, @question\_id1, @user\_answer1)

## 3.21 Prepare Quiz

**BookHub**

Book

Profile Logout

1984 by George Orwell  
Genre: Dystopia

Your progress: 217

Quiz Recommend

Among the seminal texts of the 20th century, George Orwell's dystopian novel 1984 is a futuristic purgatory becomes more of a totalitarian, bureaucratic vision of a totalitarian, bureaucratic Orwell's prescience of modern life, such a thorough version of hell. Ranks among the most terrifying novels ever written.

Erroneous Info Reviews

Language

Edition	Publisher
250	Plume
249	Penguin Books
248	Houghton Mifflin
247	New American Library

Progress Steps

Date	Edition	...
10.02.2021	249	...
09.02.2021	249	156
23.01.2021	249	97
20.01.2021	249	45

**Prepare Quiz**

Quiz Name

**Q1**

Question 1

Option 1

Option 2

Option 3

Option 4

**Q2**

Question 2

Option 1

Option 2

Option 3

Option 4

**Q3**

Question 3

Option 1

Option 2

Option 3

Option 4

**Q4**

Question 4

Option 1

Option 2

Option 3

Option 4

**Q5**

Question 5

Option 1

Option 2

Option 3

Option 4

**Q6**

Question 1

Option 1

Option 2

Option 3

Option 4

End Date

Done

Figure 21: Pop-up window that allows librarians to create quizzes

### **Create Quiz:**

Insert into Quiz (quiz\_name, start\_date, end\_date, librarian\_id)

Values( @quiz\_name, @start\_date, @end\_date, @librarian\_id)

### **Create Question:**

Insert into Question ( question\_text, answer, option\_a, option\_b, option\_c, option\_d, quiz\_id, librarian\_id)

Values( @question\_text, @answer, @option\_a, @option\_b, @option\_c, @option\_d, @quiz\_id, @librarian\_id)



## 3.22 Erroneous Information Request Pop-up

**BookHub**

Book Profile Logout

**1984** by **George Orwell**  
Genre: **Dysto**

Among the sent  
futuristic purgat  
vision of a total  
Orwell's prescie  
such a thorough  
novels ever writ

**Erroneous Info Request**

**1984** by **George Orwell**

Type

Title

Author

Genre

Year

Page Count

Edition No

Publisher

Language

Format

Cover Photo

Translator

Additional Notes

Done

Progress: 217

Recommend

Reviews

language

Date	Edition	Progress
10.02.2021	249	217
09.02.2021	249	156
23.01.2021	249	97
20.01.2021	249	45

Figure 22: Pop-up screen that allows users to create erroneous information requests to be reviewed by librarians about books

### Create edit request:

Insert into edit\_request (new\_book\_name, new\_book\_author, new\_book\_genre, new\_book\_year, new\_book\_edition\_no, new\_book\_page\_count, new\_book\_publisher, new\_book\_language, new\_book\_format, new\_book\_cover\_photo, new\_book\_translator, additional\_notes, user\_id, book\_id, date)

Values (@new\_book\_name, @new\_book\_author, @new\_book\_genre, @new\_book\_year, @new\_book\_edition\_no, @new\_book\_page\_count, @new\_book\_publisher, @new\_book\_language, @new\_book\_format, @new\_book\_cover\_photo, @new\_book\_translator, @additional\_notes, @user\_id, @book\_id, date)

## 3.23 Challenges Page

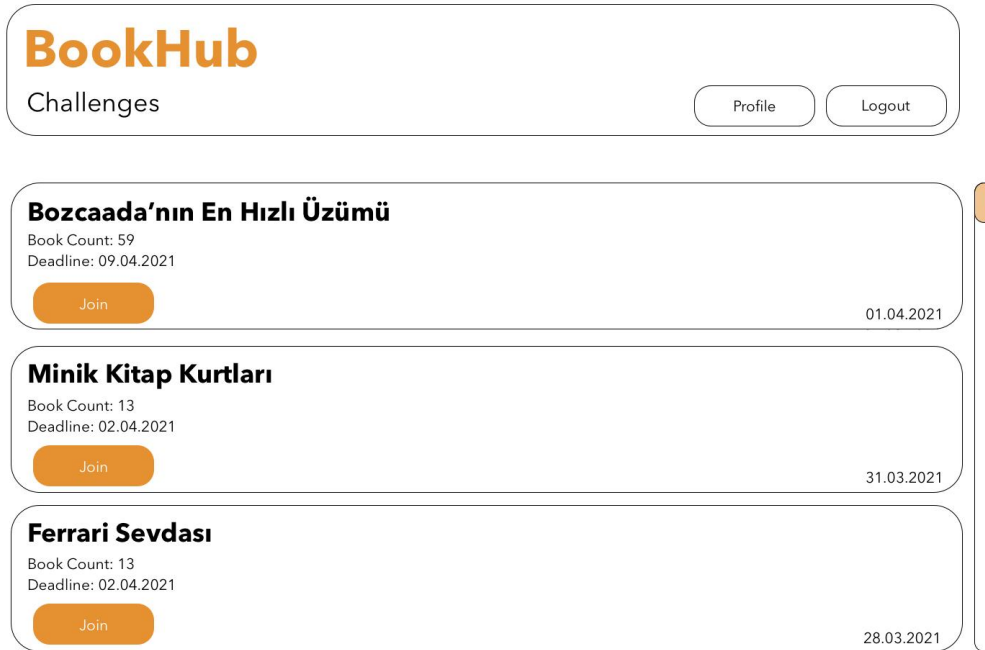


Figure 23: The page listing ongoing challenges and allowing users to join challenges

### Retrieve active challenges:

Select \*

From Challenge

Where end\_date > dateadd(day, 90, getdate())

## 3.24 Post Page

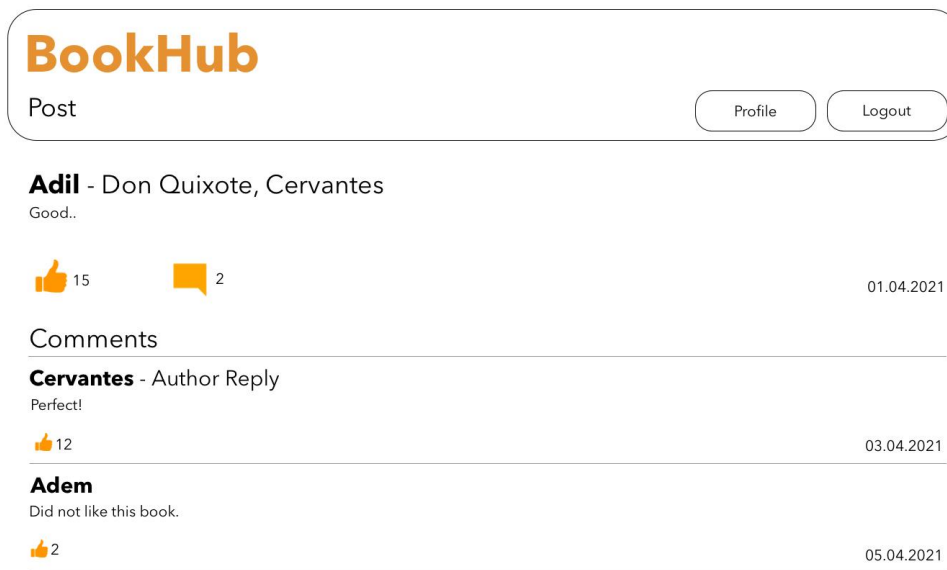


Figure 24: Post screen that lists a post, its rate, and comments

### **Retrieve author's reply:**

Select \*

From Replies

Where post\_id = @post\_id

### **Retrieve specific post's comments:**

Select \*

From Comment C, Post P, (Select user\_id, username

From User

where C.user\_id = user\_id) as CU

Where C.post\_id = P.post\_id AND P.post\_id = @post\_id

**Like post:**

Insert Into Likes\_post (user\_id, post\_id)

Values(@user\_id, @post\_id)

Update Post

Set like\_count = like\_count + 1

Where post\_id = @post\_id

**Comment post:**

Insert Into Comment ( comment\_id, date, content, user\_id, post\_id )

Values (@comment\_id, @date, @content, @user\_id, @post\_id)

Update Post

Set comment\_count = comment\_count + 1

Where post\_id = @post\_id

## 3.25 Book Review Page

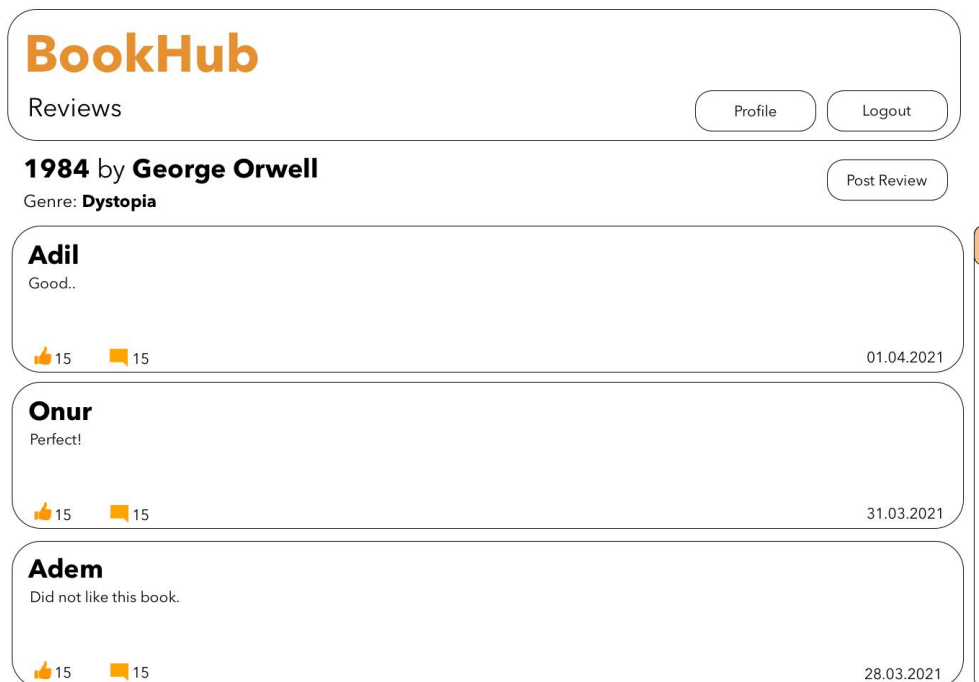


Figure 25: The page that lists reviews posted by users about a specific book

### Retrieve all posts about the book:

Select \*

From Post P, (Select user\_id, name

from User) as U

Where P.book\_id = @book\_id AND U.user\_id = P.user\_id

Order By date DESC

## 3.26 Quotes Page

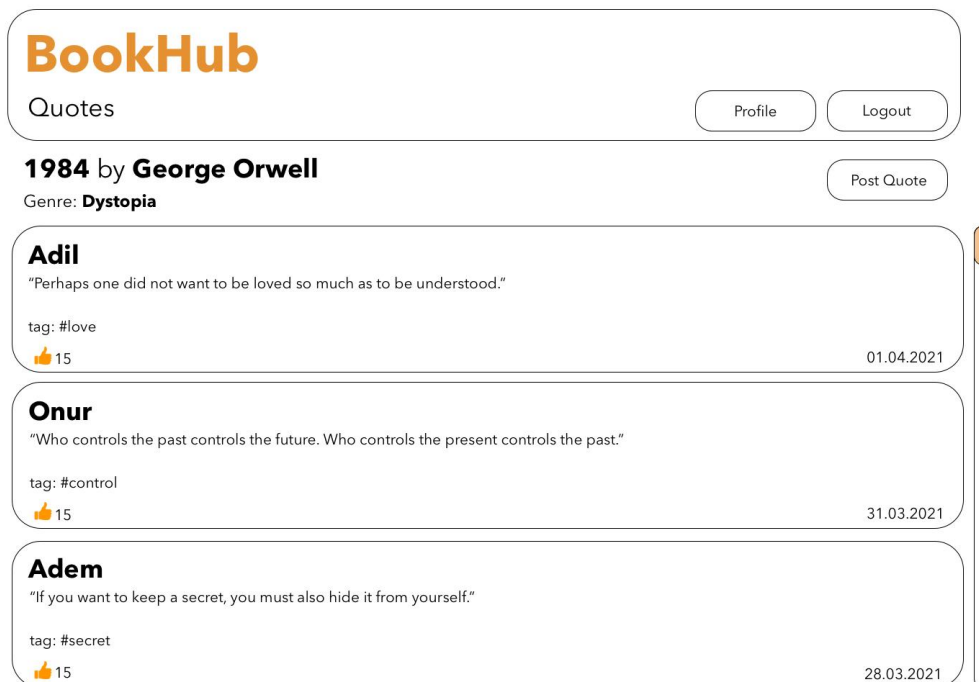


Figure 26: Quotes page that lists quotes posted by librarians

### Retrieve all quotes about the book:

Select \*

From Quote Q, (Select user\_id, name

from User) as U

Where Q.book\_id = @book\_id AND U.user\_id = Q.user\_id

Order By date DESC

### 3.27 Post Review Pop-up



Figure 27: Pop-up screen that allows users to type their reviews about a book and rate the book

#### **Create a review post:**

Insert into Post (book\_id, date, content, rate, user\_id, like\_count, comment\_count )

Values(@book\_id, @date, @content, @rate, @user\_id, 0, 0)

## 3.28 Reply Post



Figure 28: Pop-up screen that allows users to reply to a post

### Create an author reply to post:

Insert into Replies

Values( @post\_id, @user\_id, @reply)



## 3.29 Post Quote Pop-up

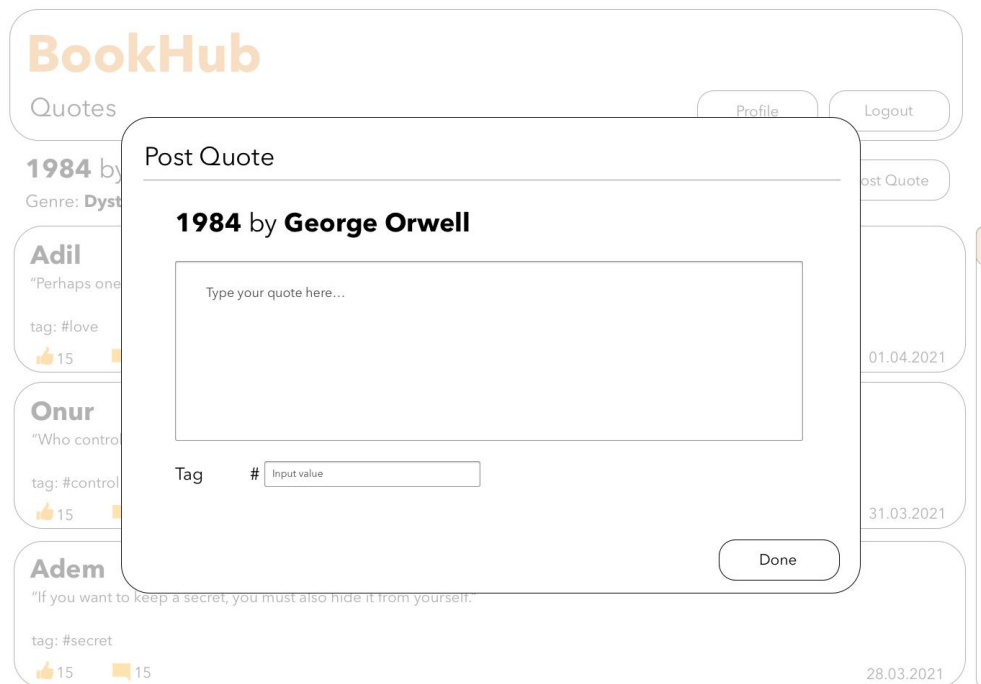


Figure 29: Pop-up screen that allows librarians to post a quote about a book

### **Create a quote:**

Insert into Quote (text, tag, book\_id, user\_id )

Values(@text, @tag, @book\_id, @user\_id)

### 3.30 Recommend Book

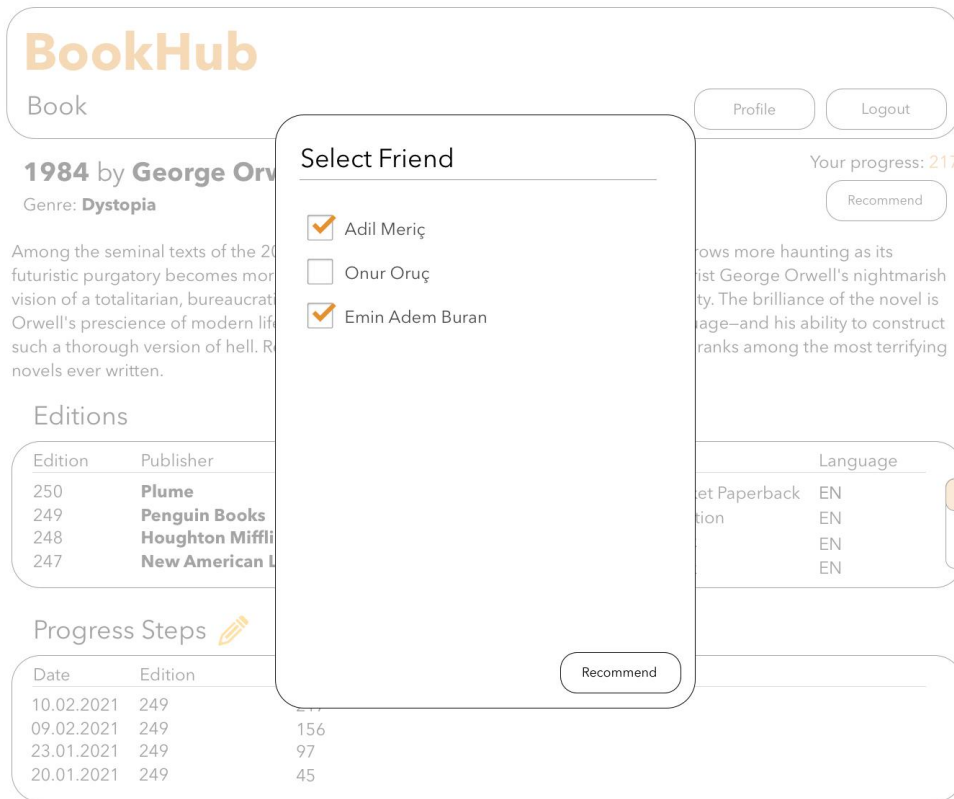


Figure 30: Pop-up screen that allows users to recommend a book to his/her friends

#### **For information retrieval:**

Select U.name, U.surname

From User U, Friends F

Where U.user\_id = F.friend\_id AND F.user\_id = @user\_id

AND U.user\_id NOT IN ( Select R.user\_id

From Reads R

Where R.book\_id = @book\_id)

#### **For recommend:**

Insert into Recommend\_Book

Values( @book\_id, @reccomended\_id, @recommender\_id)

## 4. Implementation Details

In this project we will use Java Spring Framework as backend, and MySQL as the database. For front-end, JavaScript and React will be used.

## 5. Website

Reports can be found at: <https://mercadil.github.io/>