



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

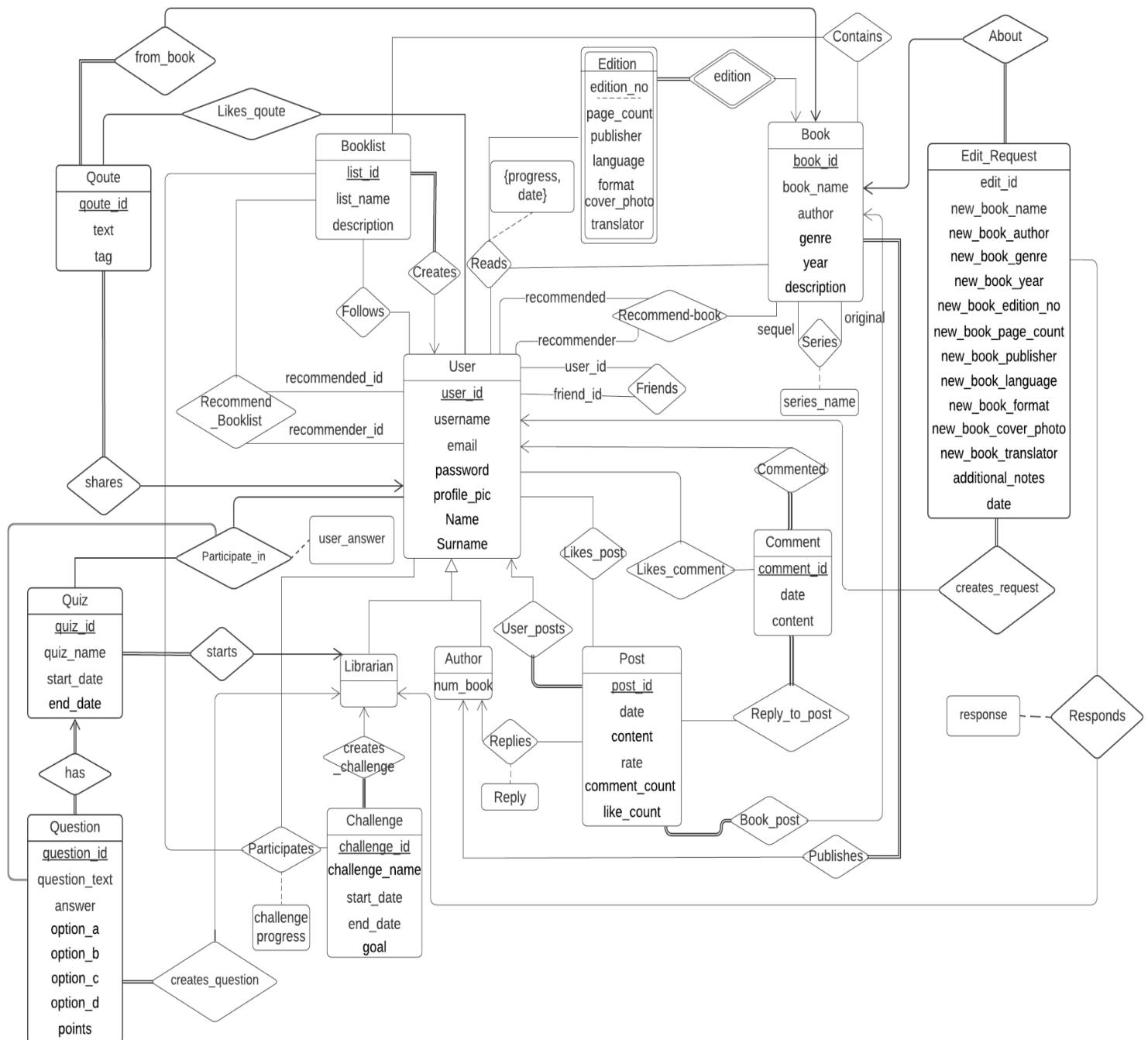
1. Revised ER	4
2. Table Schemas:	5
2.1 Quote	5
2.2 Book	6
2.3 Edition	7
2.4 Quiz	8
2.5 Question	9
2.6 Series	10
2.7 Participate_in	11
2.8 Edit_Request	12
2.9 Booklist	13
2.10 Contains	14
2.11 Follows	15
2.12 Recommend_Booklist	16
2.13 Participate	17
2.14 Challenge	18
2.15 Post	19
2.16 Likes_post	20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

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

67

67

67



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	int AUTO INCREMENT,
book_id	int NOT NULL,
date	date NOT NULL,
content	varchar(300) NOT NULL,
rate	int NOT NULL,
user_id	int NOT NULL,
like_count	int NOT NULL,
comment_count	int 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,  
);
```

2.17 Recommend_Book

Relational Model

Recommend_Booklist(book_id, recommended_id, recommender_id)

Foreign Key: book_id references to Book

recommended_id references to User

recommender_id references to User

Candidate Key: {(book_id, recommended_id, recommender_id)}

Table Definition

```
create table Recommend_Book (
```

book_id int NOT NULL,

<u>recommended id</u>	int NOT NULL,
-----------------------	----------------------

recommender_id **int** NOT NULL,

primary key(book_id, recommended_id, recommender_id),

```
foreign key( book_id ) references Book,
```

```
foreign key( recommended_id ) references User),
```

```
foreign key( recommender_id ) references User);
```

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

Login

Clear

Register

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

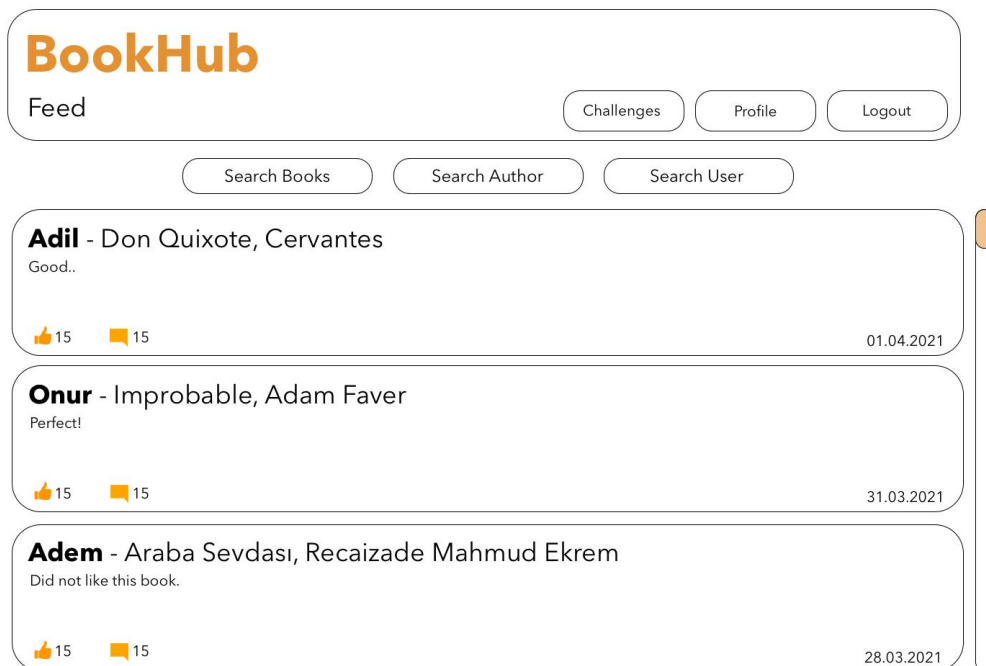


Figure 3: Home page

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

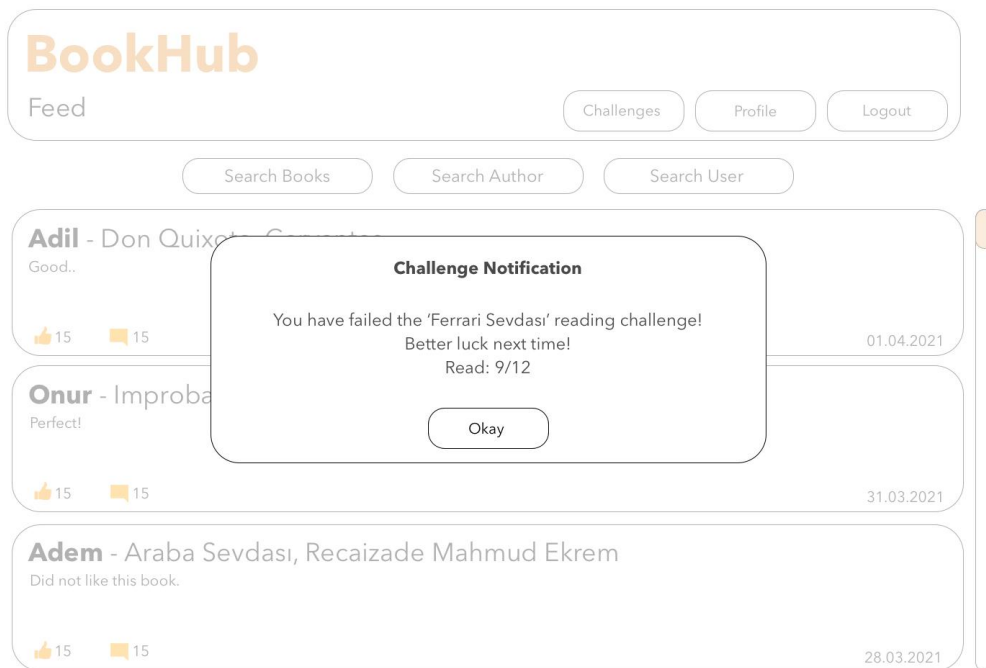


Figure 4: Challenge notification pop-up

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

Night

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

Adil

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

George

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

ProfileLogout

1984 by **George Orwell**
Genre: **Dystopia**
1949

Your progress: 217

Prepare QuizQuizRecommend

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 RequestQuotesReviews

Editions

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

BookHub

Book

Profile

Logout

1984 by George Orwell

Your progress: 217

Genre: Dystopia

Recommend

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. Orwell's prescience of modern life—and his ability to construct such a thorough version of hell—makes this novel one of the most terrifying novels ever written.

George Orwell's nightmarish vision of a totalitarian, bureaucratic world—and his ability to construct among the most terrifying and prescient fictional worlds—makes this novel one of the most terrifying and prescient novels ever written.

Add Progress

1984 by George Orwell

Date

Date

Edition

Edition

I am now at page

Page No

Done

Editions

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

ProfileLogout

Arda Akça Büyük - User

Currently Reading

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

Read

Finished	Title, Author		
10.02.2021	Heidi, Johanna Spyri	★★★★☆	Post Review
25.01.2021	Animal Farm, George Orwell	★★★★☆	Post Review
23.01.2021	To Kill a Mockingbird, Harper Lee	★★★★☆	Post Review

Posts

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

Book Lists

Last Update	Name	Books
01.04.2021	Classics	21
25.03.2021	Special	3
23.03.2021	Readlist	55

Create

Friends

Add

Name

Adil Meriç
Emin Adem Buran
Onur Oruç

Challenges Succeeded

Title	Book Count
Bozcaada'nın En Hızlı Üzüümü	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

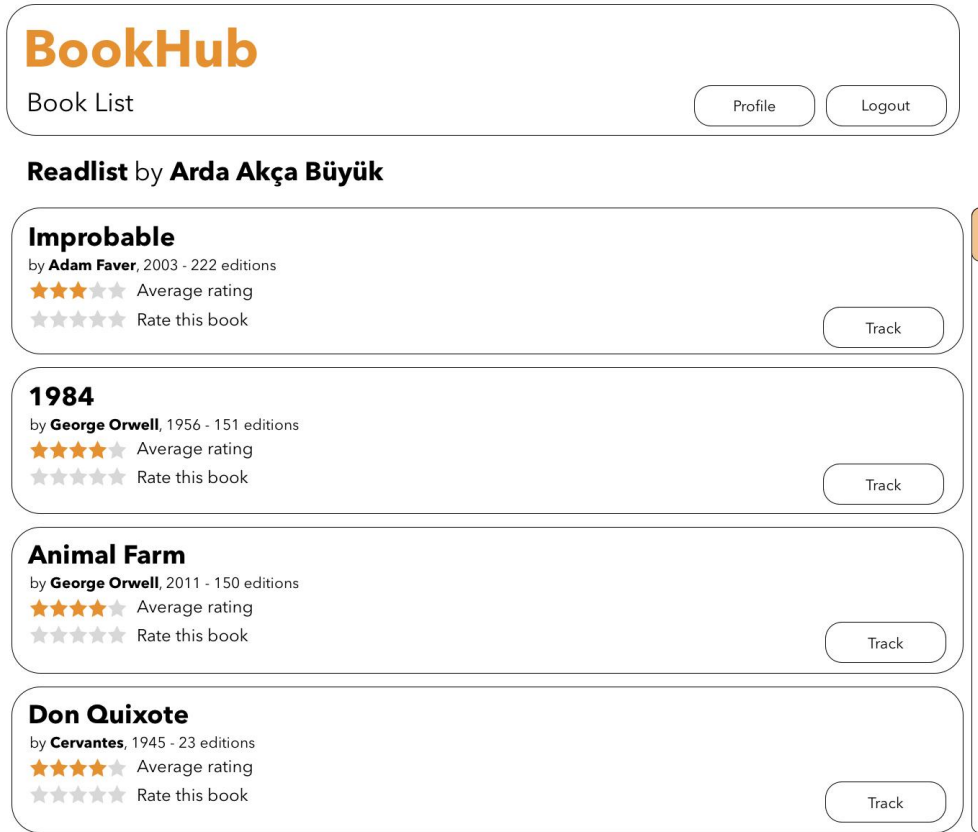


Figure 12: The page that lists books in a user's booklist

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

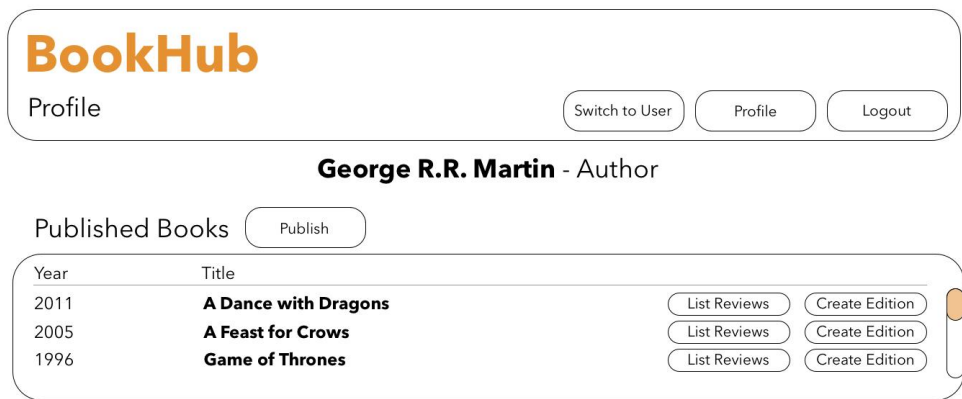


Figure 13: Author profile page listing the books published by the author

Retrieve Author's books:

Select *

From Book

Where author_id = @author_id

3.14 Create Edition of Book

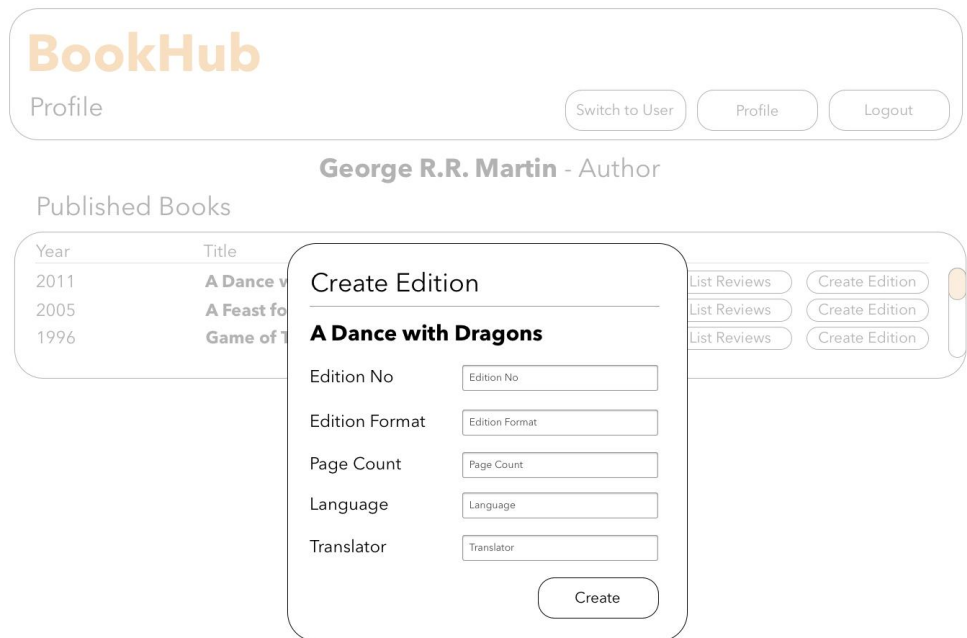


Figure 14: Pop-up window that allows librarians to create an edition of a book

Create Edition:

Insert into Edition

Values (@edition_no, @book_id, @page_count, @publisher, @language, @format, @cover_photo, @translator);

3.15 Publish Book Pop-up

The image shows a wireframe of a web application named 'BookHub'. In the background, there is a user profile section with a 'Profile' button and a 'Logout' button. Below the profile, there is a 'Published Books' section with a table listing books by year and title. The table has two columns: 'Year' and 'Title'. The rows show books published in 2011, 2005, and 1996. Overlaid on this is a 'Publish Book' pop-up form. The form has the following fields: 'Title' (text input), 'Genre' (text input), 'Description' (text area), 'Edition Format' (text input), 'Page Count' (text input), 'Language' (text input), and 'Translator' (text input). There is a 'Create' button at the bottom of the form. To the right of the form, there are buttons for 'List Reviews' and 'Create Edition' for each book in the 'Published Books' list.

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	Details
07.02.2021	Adil Meriç	Night (The Night Trilogy, #1)	Details
07.01.2021	Elif Özer	1984	Details

Reading Challenges

Create

Start Date	Deadline	Title	Type	Book Count	
07.01.2021	01.04.2021	Minik Kitap Kurtları	Booksss	30	Details
07.01.2021	23.02.2021	Kitap Canavarları	Booksss	10	Details
07.01.2021	14.01.2021	Küçük Dostoyevksiler	Author / Speedy	3	Details

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

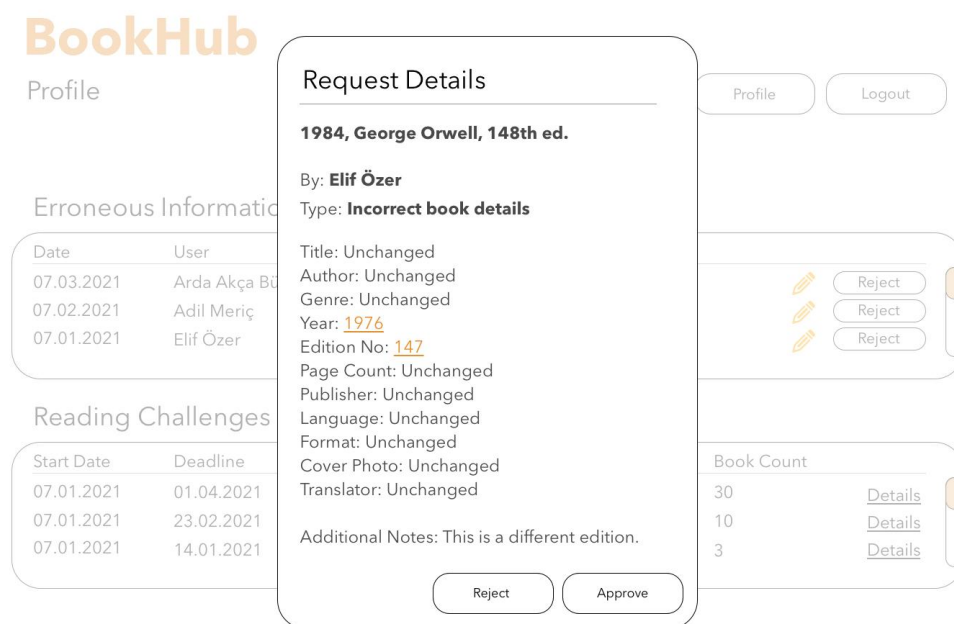


Figure 17: Pop-up window that allows librarians to review details about correction requests

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

BookHub

Profile Switch to User Profile Logout

İlyas Alperen Büyük - Librarian

Erroneous Information

Date	User
07.03.2021	Arda Akça Büyü
07.02.2021	Adil Meriç
07.01.2021	Elif Özer

Create Challenge

Title Edition No

Deadline Deadline

Book Count Book Count

Reading Challenges

Start Date	Deadline	Title	Type	Book Count	
07.01.2021	01.04.2021	Minik Kitap Kurtları	Booksss	30	Details
07.01.2021	23.02.2021	Kitap Canavarları	Booksss	10	Details
07.01.2021	14.01.2021	Küçük Dostoyevksiler	Author / Speedy	3	Details

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

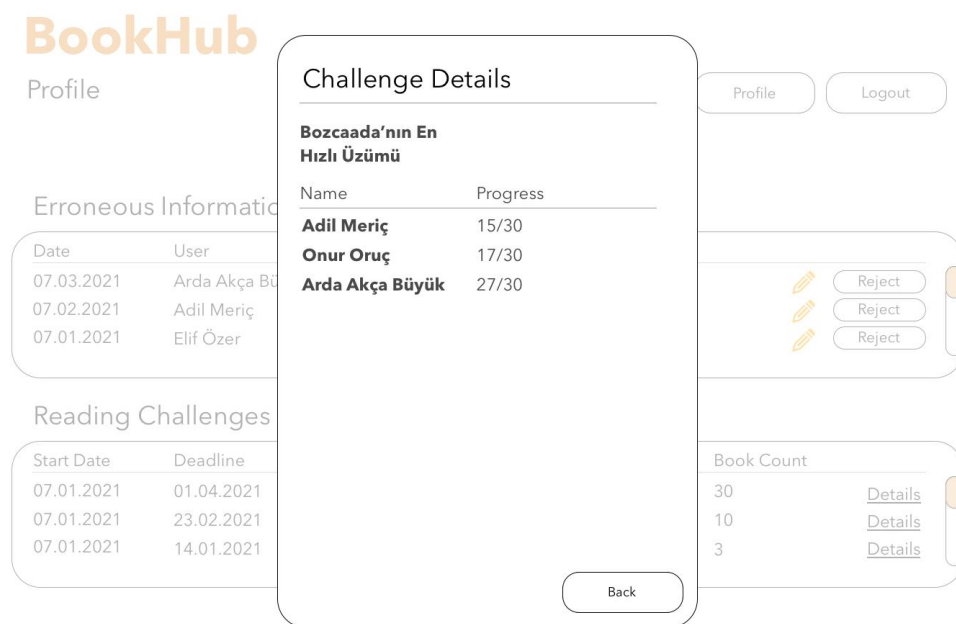


Figure 19: Pop-up screen listing a challenge details

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

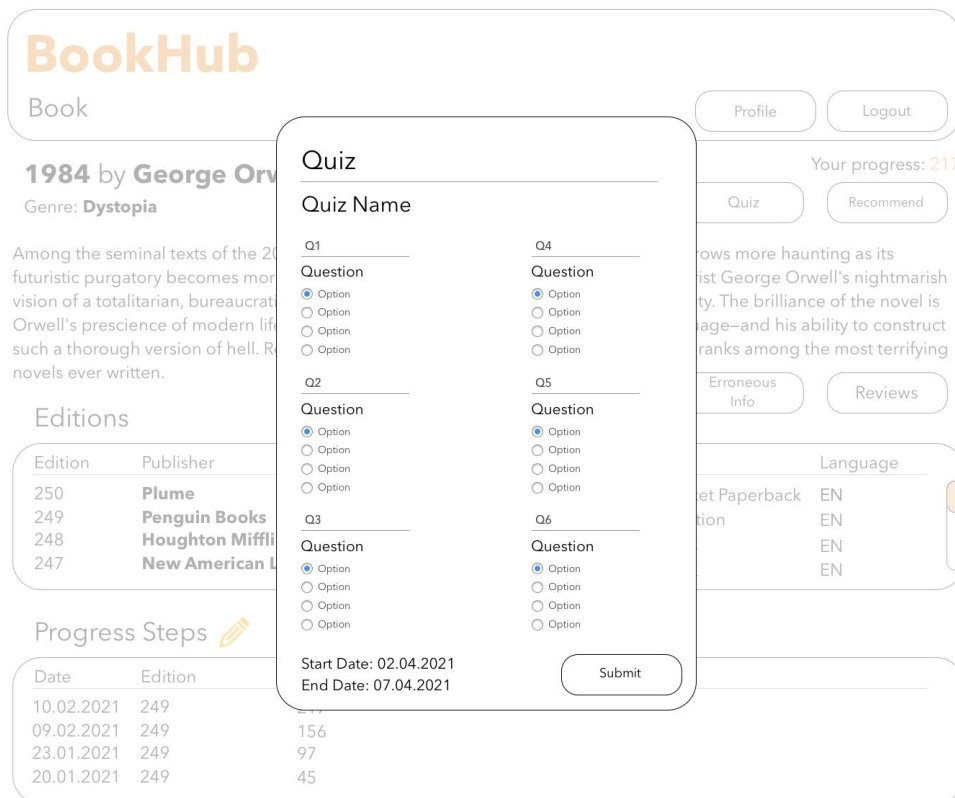


Figure 20: Quiz screen listing the questions related to a book

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**

Among the seminal texts of the 20th century, *1984* depicts a futuristic purgatory becomes more and more a vision of a totalitarian, bureaucratic, and dehumanized world. Orwell's prescience of modern life and the dangers of such a thorough version of hell. *1984* is one of the novels ever written.

Editions

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

Progress Steps

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

Prepare Quiz

Quiz Name

End Date

Done

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

Your progress: 217

Quiz Recommend

Erroneous Info Reviews

Language

et Paperback EN

tion EN

EN

EN

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: **Dystopian**

Among the series of futuristic purgatorial vision of a totalitarian society, Orwell's precise and such a thorough novels ever written.

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

Editions

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

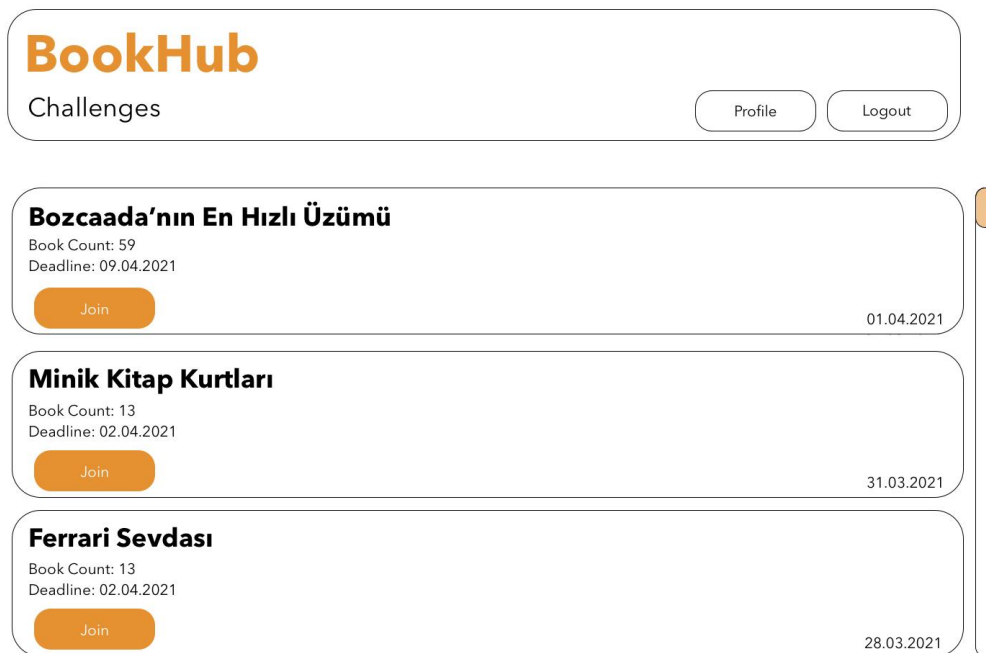


Figure 23: The page listing ongoing challenges and allowing users to join challenges

Retrieve active challenges:

Select *

From Challenge

Where end_date > dateadd(day, 90, getdate())

3.24 Post Page

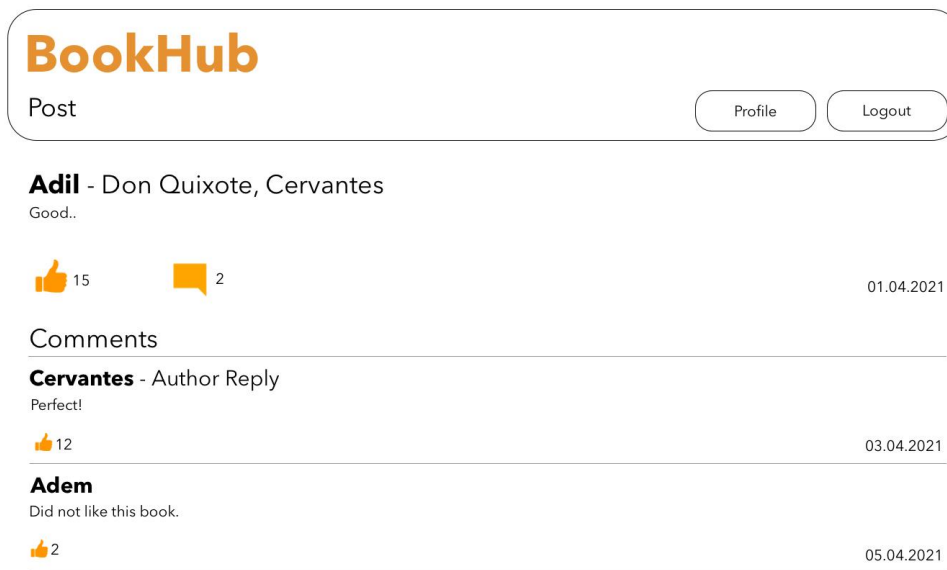


Figure 24: Post screen that lists a post, its rate, and comments

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

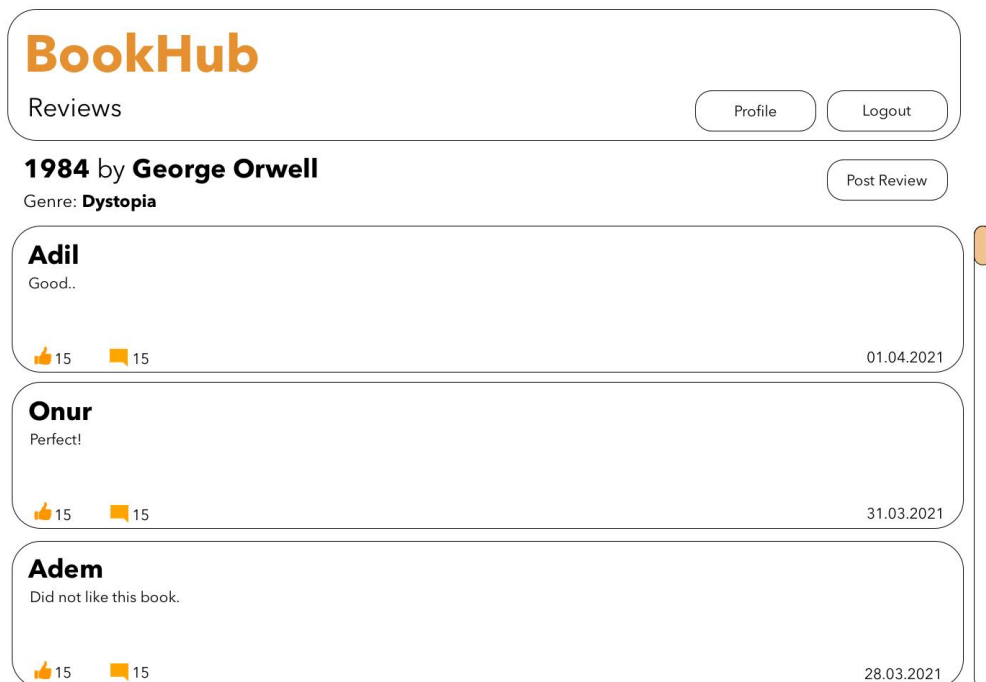


Figure 25: The page that lists reviews posted by users about a specific book

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

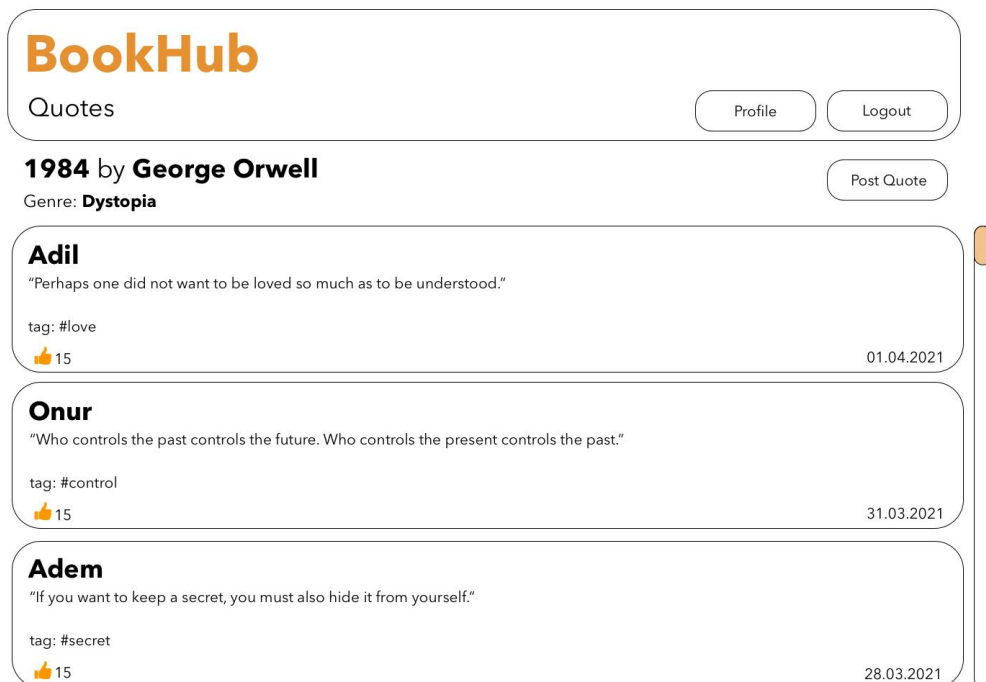


Figure 26: Quotes page that lists quotes posted by librarians

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

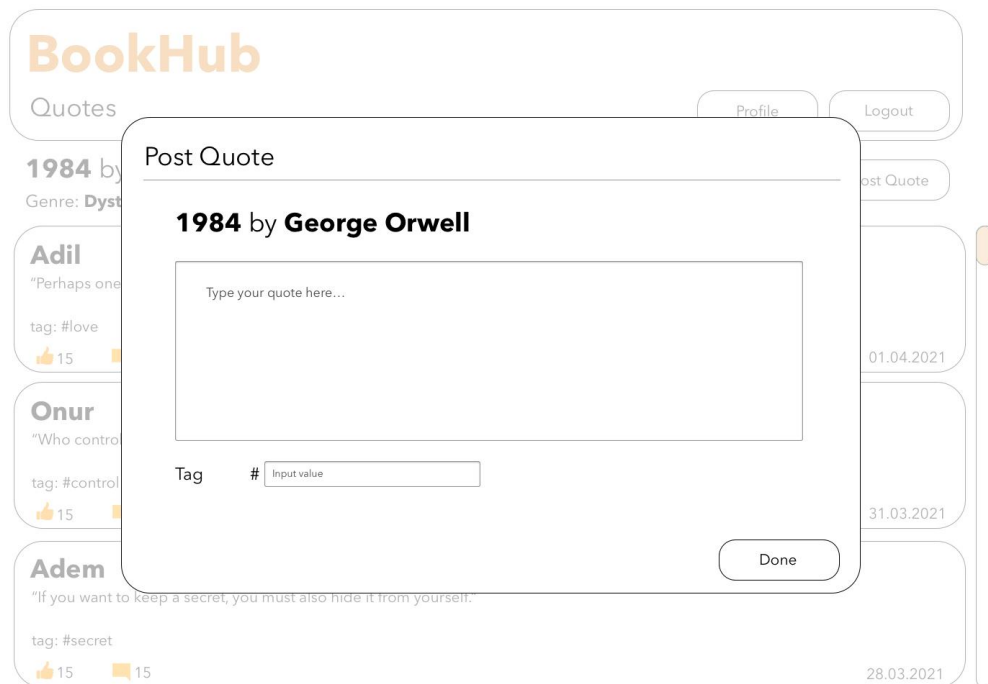


Figure 29: Pop-up screen that allows librarians to post a quote about a book

Create a quote:

Insert into Quote (text, tag, book_id, user_id)

Values(@text, @tag, @book_id, @user_id)

3.30 Recommend Book

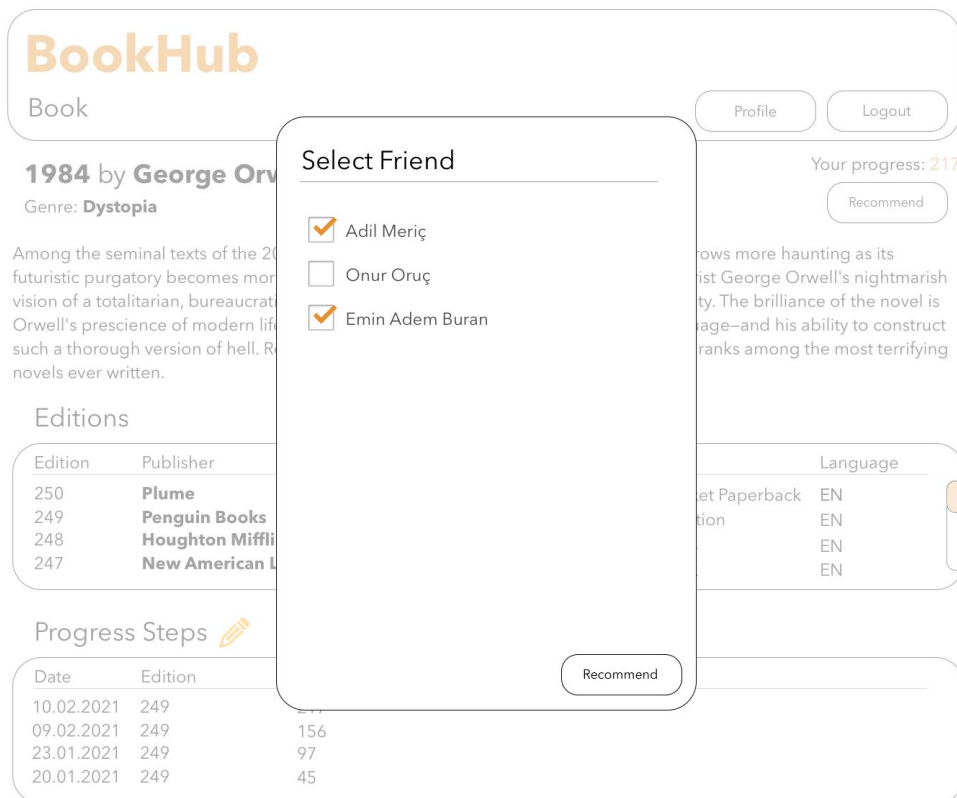


Figure 30: Pop-up screen that allows users to recommend a book to his/her friends

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, @recommender_id, @recommended_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://mericadil.github.io/>