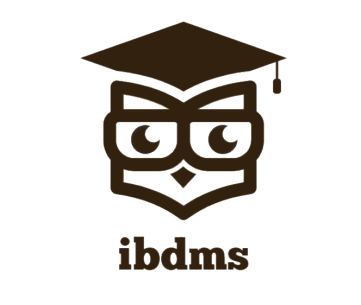
|  |  |
| --- | --- |
|  | **DOKUZ EYLÜL UNIVERSITY**  **ENGINEERING FACULTY**  **DEPARTMENT OF COMPUTER ENGINEERING** |



<International Book Database Management System (IBDMS)>

# CME 3201 Database Management Systems

# Term Project Report

Phase III FINAL REPORT

2020-2021 FALL

|  |  |  |
| --- | --- | --- |
| 2017510084 | Burcu ÖLMEZ | burcu.olmez@ceng.deu.edu.tr |
| 2017510055 | Can KIZILÖZ | can.kiziloz@ceng.deu.edu.tr |
| 2018510053 | İsmail Ege OĞUZ | Ismail.oguz@ceng.deu.edu.tr |

**Abstract**

The web application is a online book recommendation database system. Aim of the application is that to develop the consumers reading habit. The application will be useful for potential user group who likes to read book and discuss about it. The consumer can create a profile then search for books by publisher, author, name and add them to their to-read list. Also they can create special lists to recommend other users. Also, application stores the record of the books which user has read once and user can visit that list too. In time, Application will store the users data which their favorite publishers, authors then create suggested books list. So users won’t be stable to find a great book for them.

After all, when user read the book, they can add comments which will be inspected by application to create safe environment for users, and rate the book from 0-5. User can add spoiler included comment by using the spoiler alert button so other users not going to affect by that comment. Database administrators will be inspect the comment either to provide insulting.

After the team spent first 2 weeks for searching requirements and discussing about purpose of this program, the team achieved requirements segment. Then, the team has started to plan how we build International Book Database Management System. The team had all significant diagrams such as ER Diagram and UML Diagrams which are Sequence, Class, Activity, Use-Case diagrams and Site-map diagram. The team has necessary set ups and created database with suitable technologies.

The team has accomplished all significant backend tasks such as database design and implementation part, all queries and necessary functions in few weeks. Then started to accomplish user friendly and advanced frontend phase.

## Completion Report

The team has finished code phase of the project and project has a few tasks to accomplish but the most of the significant tasks has been accomplished . First of all, team couldn’t fill database with Goodreads API because Goodreads no longer issues new developer keys for public developer API and plans to retire the current version of these tools at 8th of December. So team had decided to find another API to fill database but couldn’t find free one in a short time, Team had found a csv with books information than filled database book entity with that and the pictures of the books had taken from Google Books API. Projects timeline has limped because of API problem.

Users page and the most of the features has been accomplished successfully. User can create personal lists with books which has chosen by the user, see the top lists, search books by name, author, publisher and all book attributes. User can get recommendation with AI system.

Blog segment has been created by admin and the team has planned to get advertisement of the other web sites for our blog. And also user can see the other contracted web sites. This segment will be rebuild at certain intervals.

Admins page is not available in the system but the team had recorded an admin in to the database and admins duty has been restricted. Now, admin can manage the appearance of the web site such as updating the top lists, pictures and checking the comments manually. Nevertheless, curse detector and spoiler system are working correctly so the web site has user friendly environment. Rating system has accomplished with triggers and necessary tables.

Top lists has created manually, if the team would have more time, the AI algorithm which gets the most rated books inside of the top lists ,would be implemented to the project. Now user can see his/her personal lists and top lists.

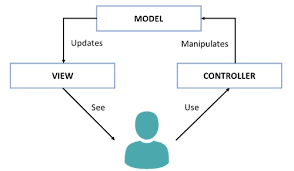
The frontend phase has been accomplished but the team spent more time than assumed. However, the web site looks as expected. The web sites appearance can be developed at a later stage.

Testing phase has been accomplished and several bugs has handled but some error checks can be detected and improved as the web site is used in the future with users feedbacks.

## Functional Decomposition

Our team used a MVC structure for this project. On the other hand we have a database. First of all Code Igniter was downloaded and all the components' are implemented into the project. Models are used to reach the database properly. All the functions in the models are created to accomplish the functionality of the project for the part of the database. Controllers are used to get the datum from the database, handle with them properly and send them to the views. Views are created to show the data to user and make it understandable for everybody.

.

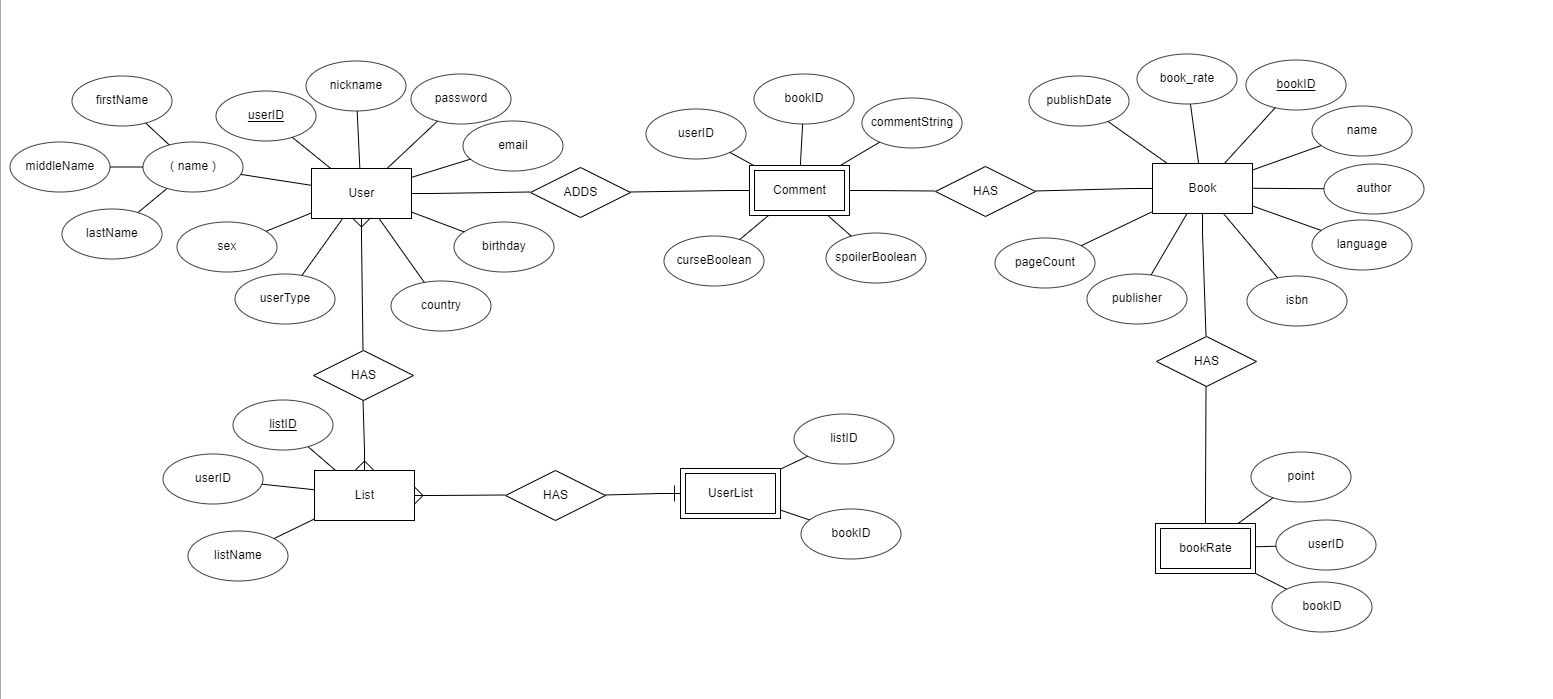


Mainly, the data get, insert; update or delete functions which are database related are all in the model. In the controllers, we mainly have functions to comment, recommendation, search, rate, list, curse detector functions. Recommendation function stand for recommending books to user with his/her related authors which he/she rated higher than 3 any of the author's book. Search function does the job of searching the book according to given string. Rate function does let the users to rate the books. List function does create lists for the user and add books into them. Also comment function does the functionality of adding comment and checks whether curse or not, also it handles with the spoiler ones.

In view section we have html documents. The team used script function to make the html pages more dynamic and more understandable for the user. As the search result does not return just a book, all the books that it returns, showed smoothly and regularly on the page such in the "destination.html".

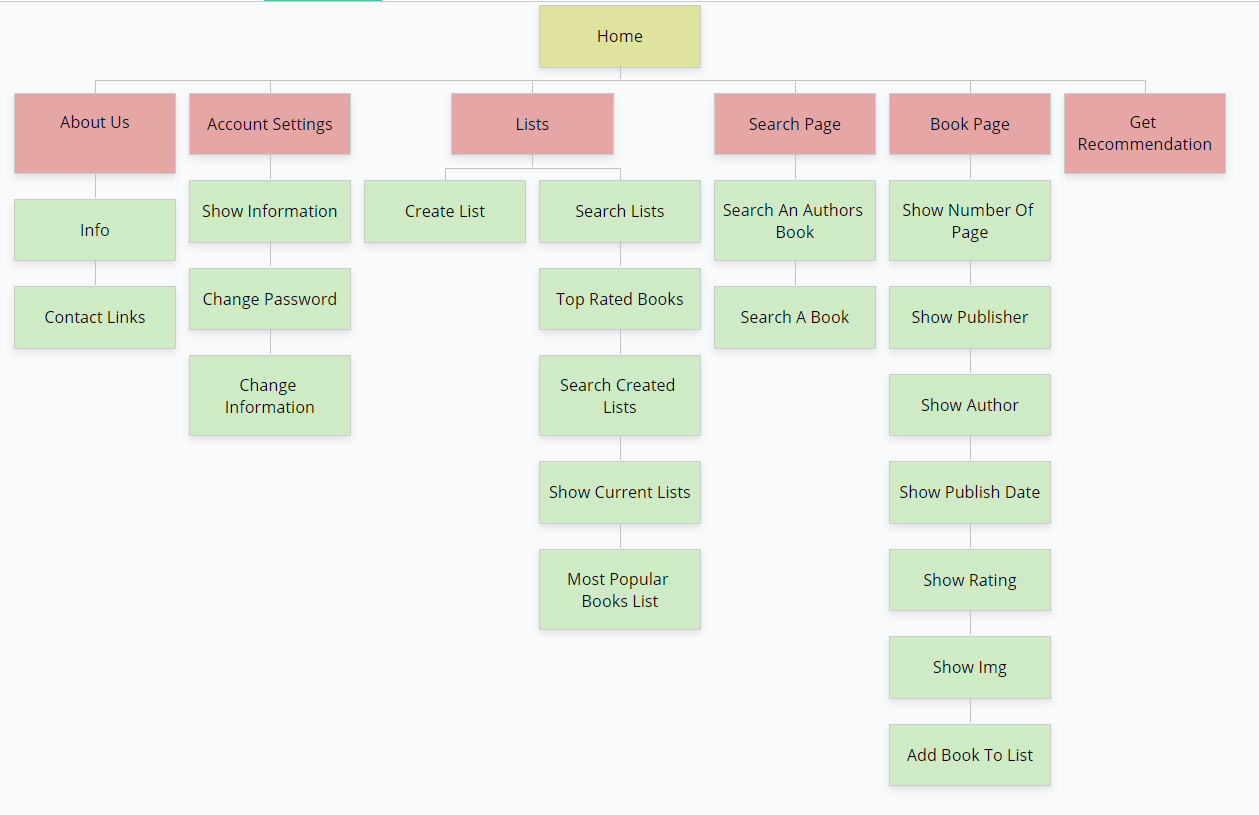
The database has the main information about users, books, user's comments, user's list, user's rate and rates of book. In addition we have a view and two triggers. The view has information about the comment and user as joined. Triggers update the book's rate automatically whenever a user rate a book.

The design of the entities, their attributes , relationships and cardinalities has been considered as shown at the Entity Relationship Diagram according to the functionalities.



**3.High Level Organization**

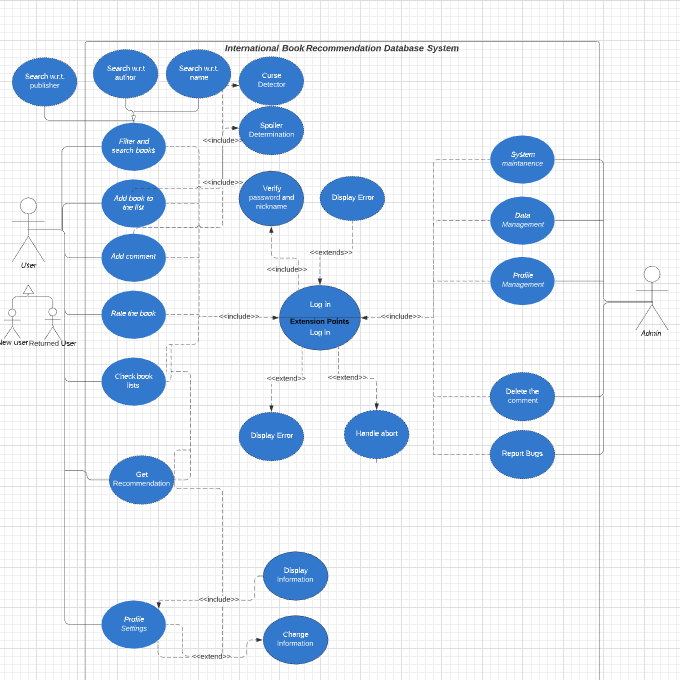
To explain the high level organization of the projects team decided to draw some diagrams to make it easier. Firstly the site-map diagram shows the main functions of the system with respect to team’s expectations.



Also to show the difference between the normal user and the admin a use case-diagram is drawn by the team. This diagram also shows the expected use-case possibilities from the application.

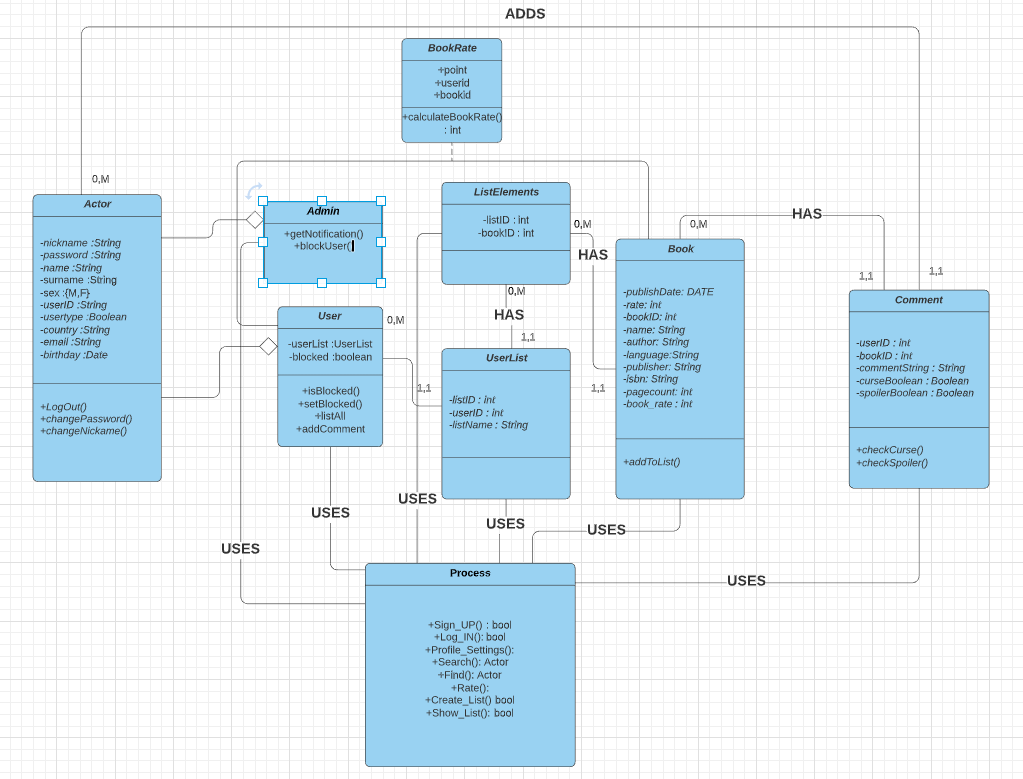
**3.Clickstreams**

As seen in the diagram there are 6 main functions and there are other sub-functions. These are the functions the team expects from the site in general hierarchically.

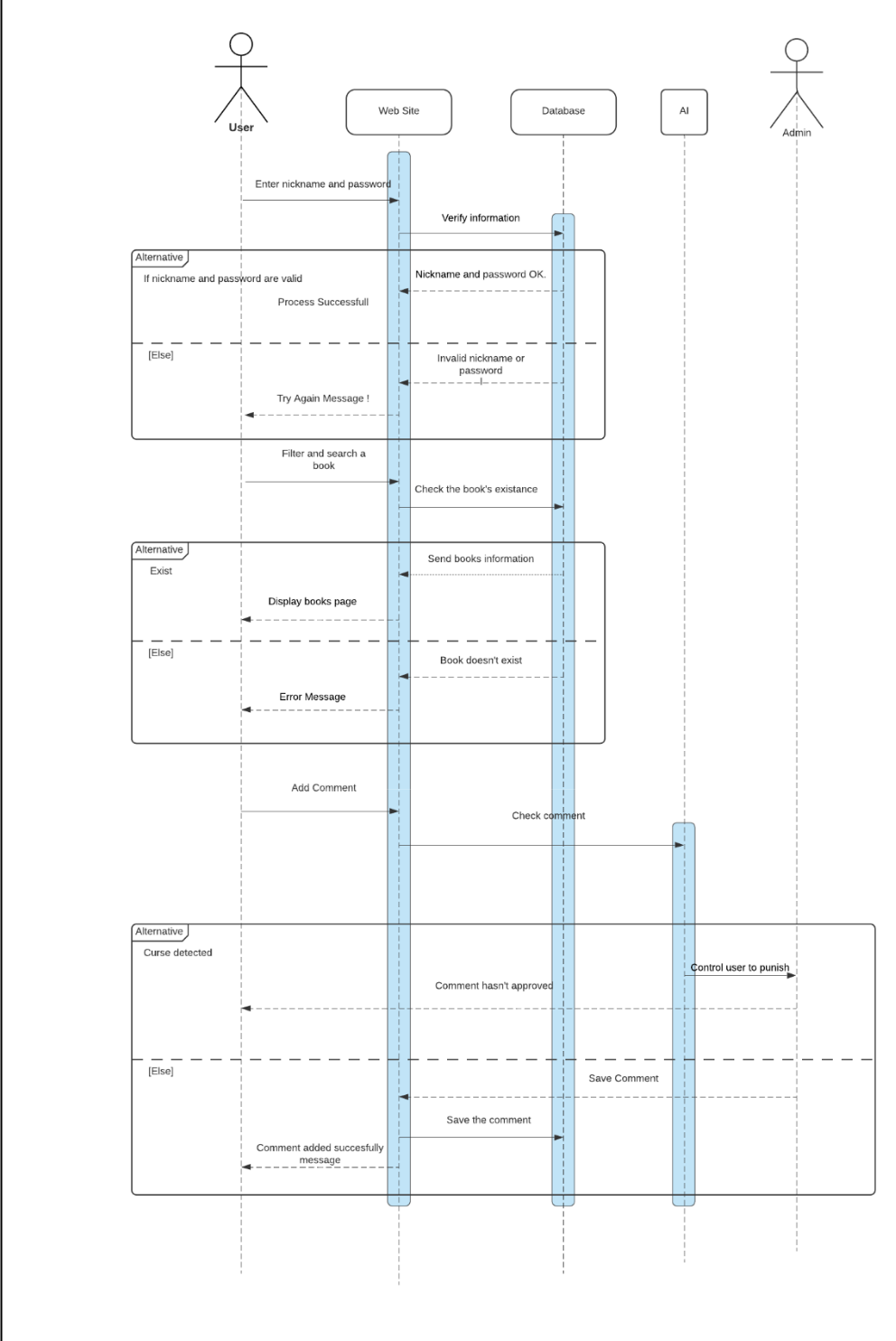


**4.Layouts**

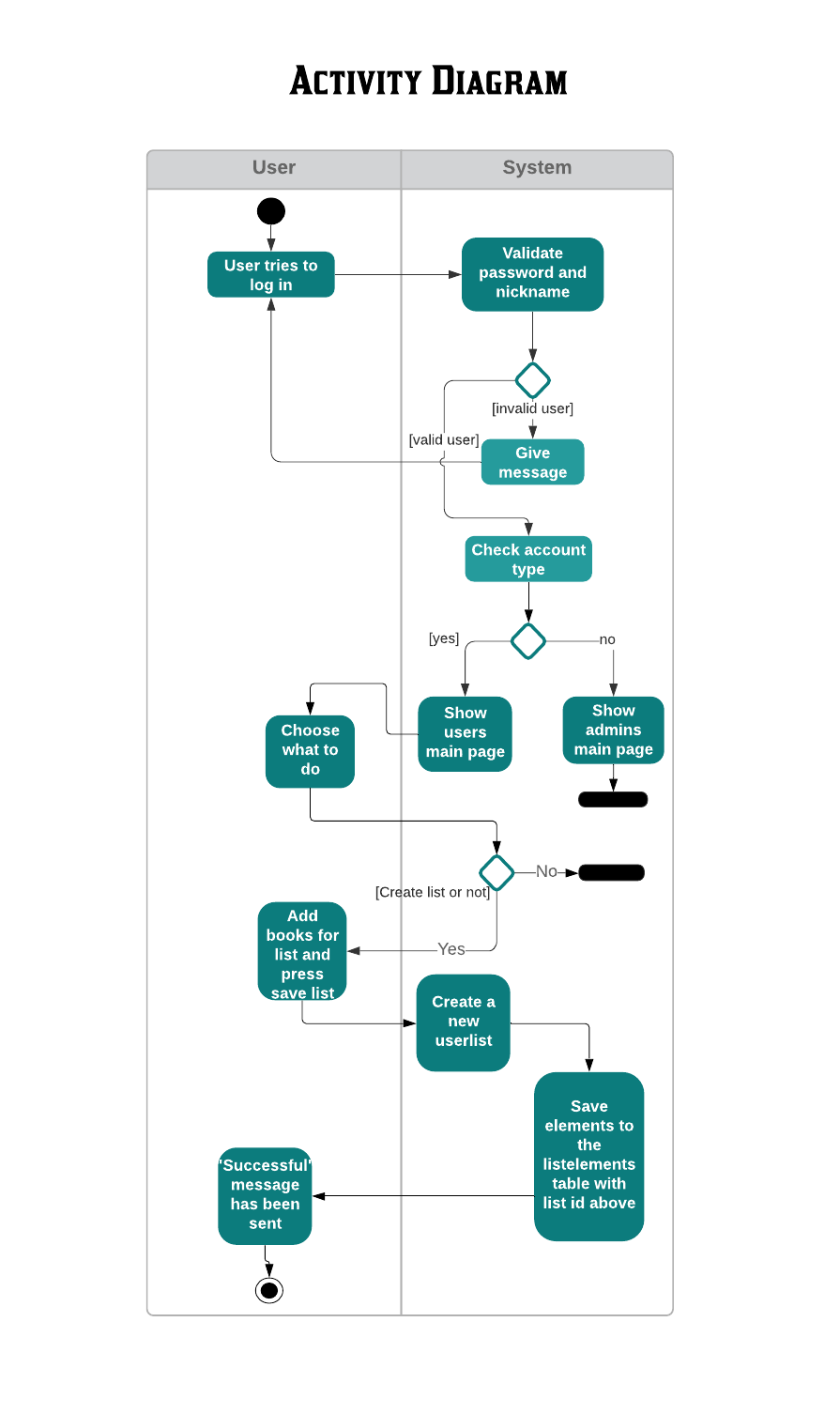
This diagram, Shows the fundamental structure of the database management system according to teams considerations and expectations.



In this example a user tries to enter the system and tries to make a comment to a book. The admin also tries to evaluate the comment. These processes and background processes are shown in the sequence diagram to make the understanding easier.

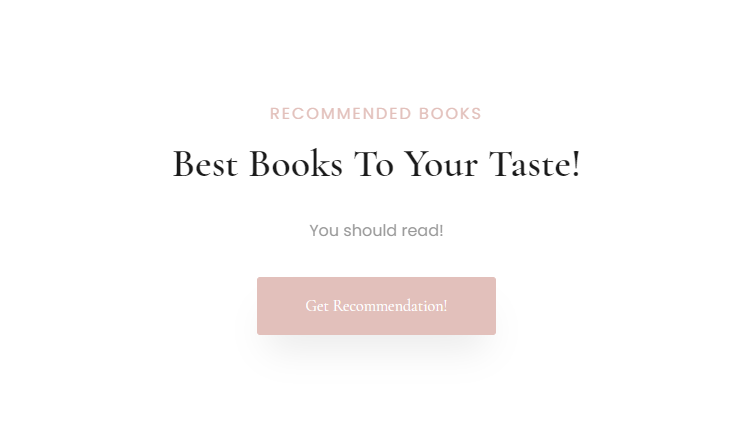


This design shows the log-in and creating a list process. First the user try to login and the system checks input each time. The menu page differs to the user type. If any user wants to add a list, at the backend a list element tuple gets created and get a location in the user list table. The whole process's success is sent to the user as a massage at the end.

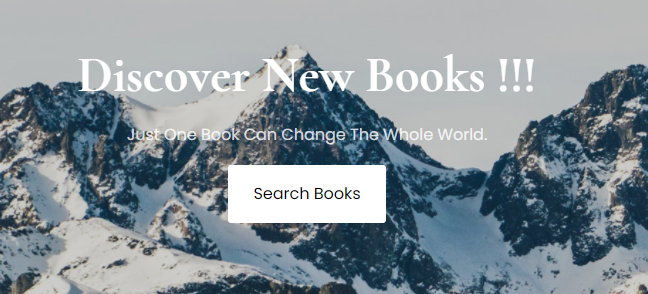


## 5. Implementation

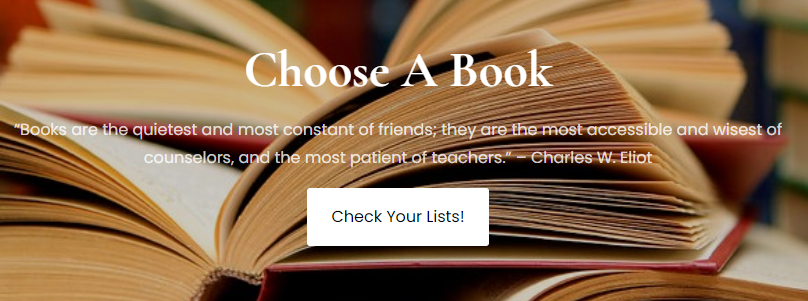
* Index.html

This page is designed for the users. They can see their recommendations, search for a book, check their lists, see the top lists and to the profile settings.

*When “Get Recommendation!” gets clicked, user can see his/her recommended books by the system.*



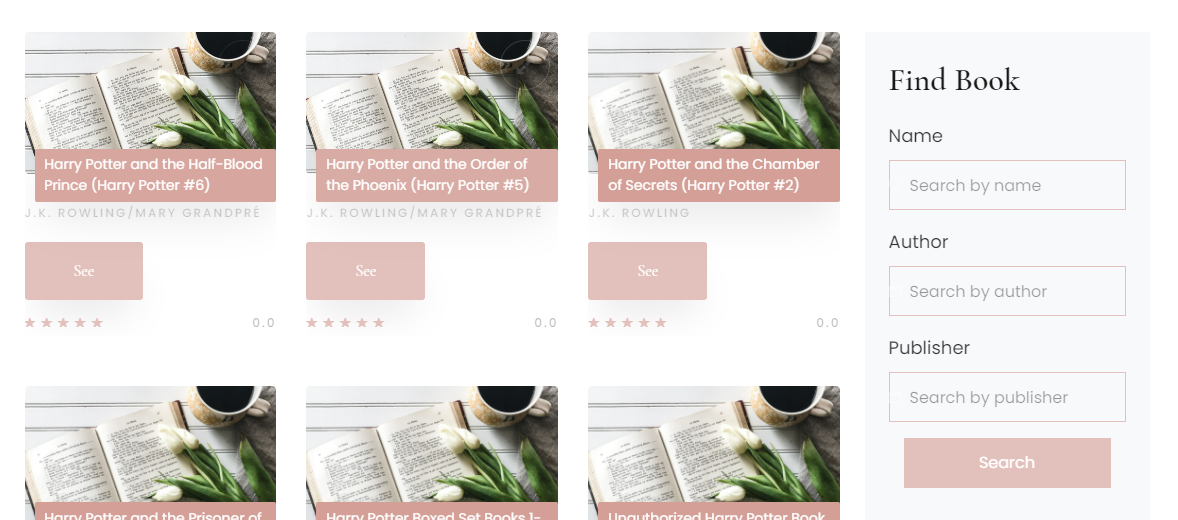
*When “Search Books” gets clicked, users being directed to page where they can give input according to their books that thet trying to find.*



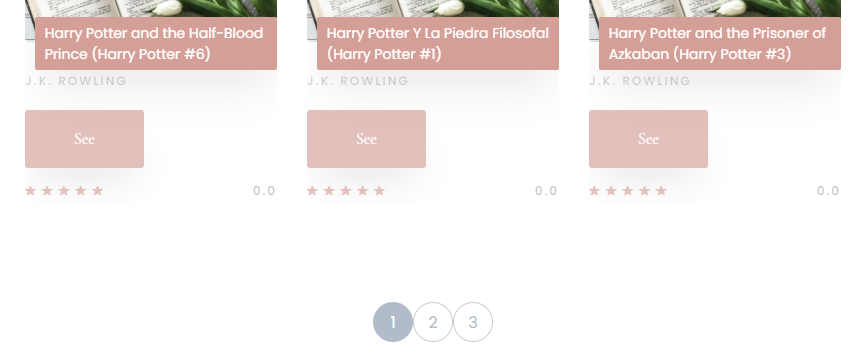
*Button “Check Your Lists!” directs user to the page where they can see their lists.*

* *Destination.html*

This page is developed to show the books properly. Also users can make the listed books more detailed and reach the detailed informations about a chosen book.



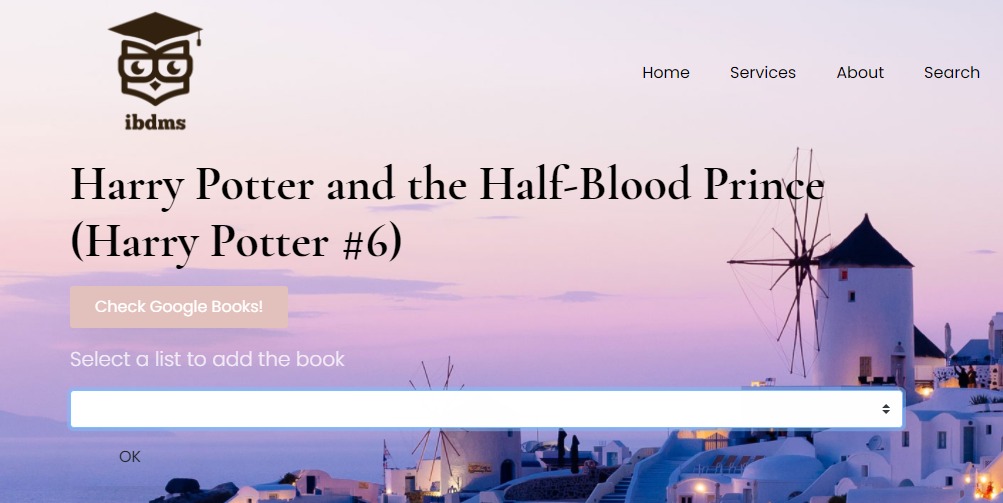
*“Find Book” is for detailed search if there is too many books on the page. “See” button below the each books are for to see the all information and reach the other functionalities of the system.*



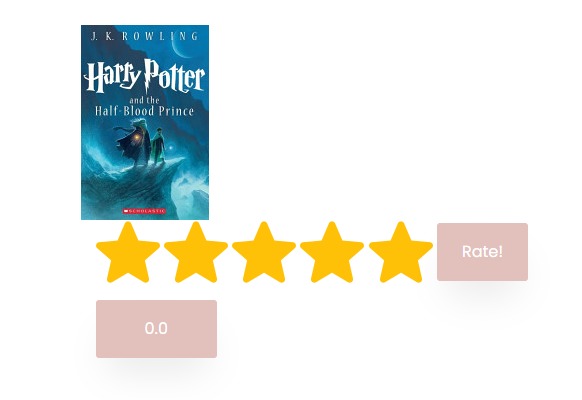
*This page designed dynamicly. User can go to the other pages to see other results by the buttons on the each page at the below.*

* *Destination-single.html*

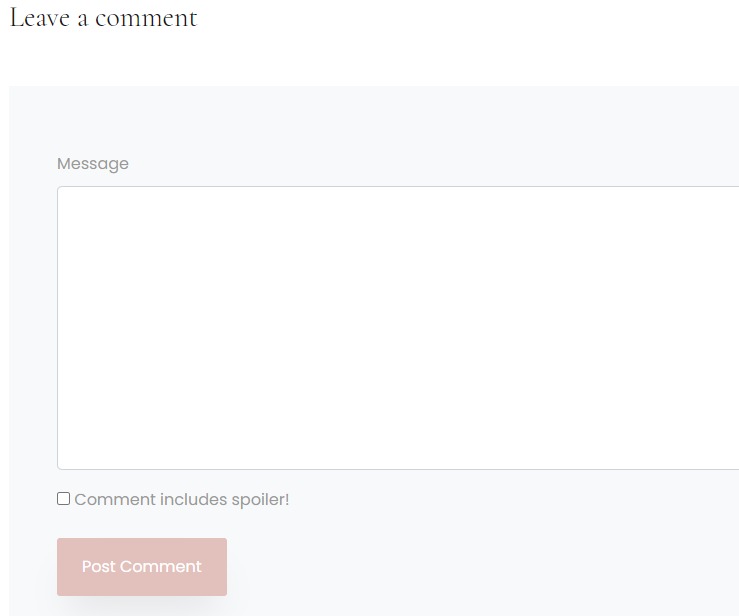
This page has developed to show books information like author, name, publisher, page count, publish date, comments, rates and book cover.



*Users can select their personal list to add the book.*



*User can rate the book and see the books average rate.*



*User can add comment and can see the other users comments if it exists.*

## 6. Future Work

The team would be build exact AI system to create top lists from users rates. The system needs more proper API which updates itself not like csv file. Users interactions could be implemented to the project such as adding friend, add reply to each others comment. Frontend could be improved.