Library Management System

EECE 350 Project

Rayan Zein

Ali Yassine

Prof: Dr. Artail

MaktabaTech User Manual

MaktabaTech is a one of a kind library in the Middle East. It encourages all citizens to engage in reading and develop their languages. As we all know, reading is becoming an old and out of date trend. MaktabaTech wants to turn this idea upside down and develop an easy, yet important way for citizens all over the world to READ.

Though reading might seem to some that it is just for fun, but in fact, it can be helping your body and mind without you even realizing it. Not only does reading expand your knowledge, but it also sharpens your minds, imagination, and writing skills.

Books can hold and keep all kinds of information, stories, thoughts and feelings. They provide a timeless form of entertainment and at the same time knowledge, which is rare to find nowadays.

MaktabTech

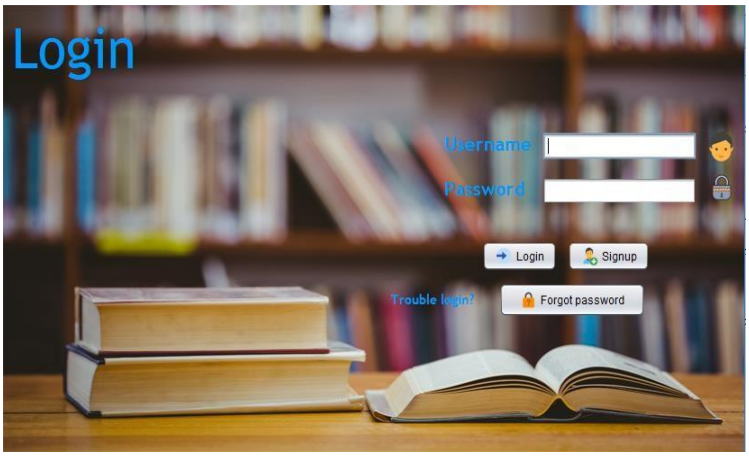
Home of Books

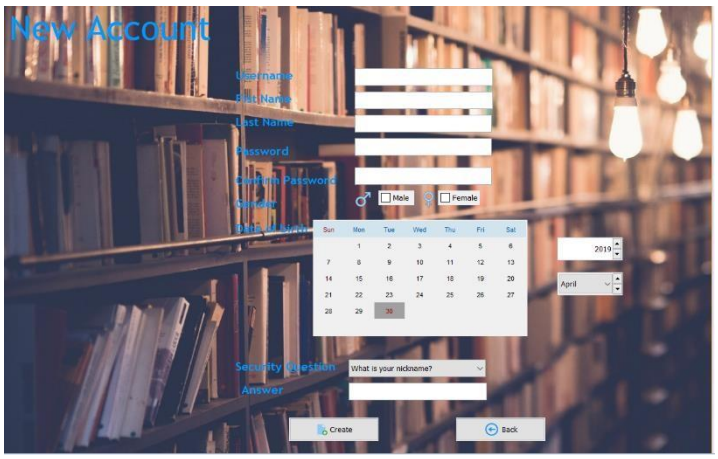
User Manual:

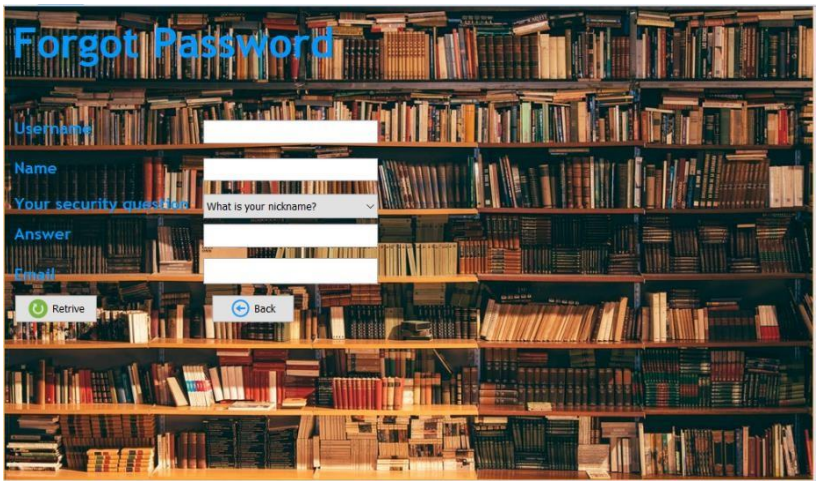
First of all, you will be greeted by our Login Page.

If you have an account already, you can immediately login in.

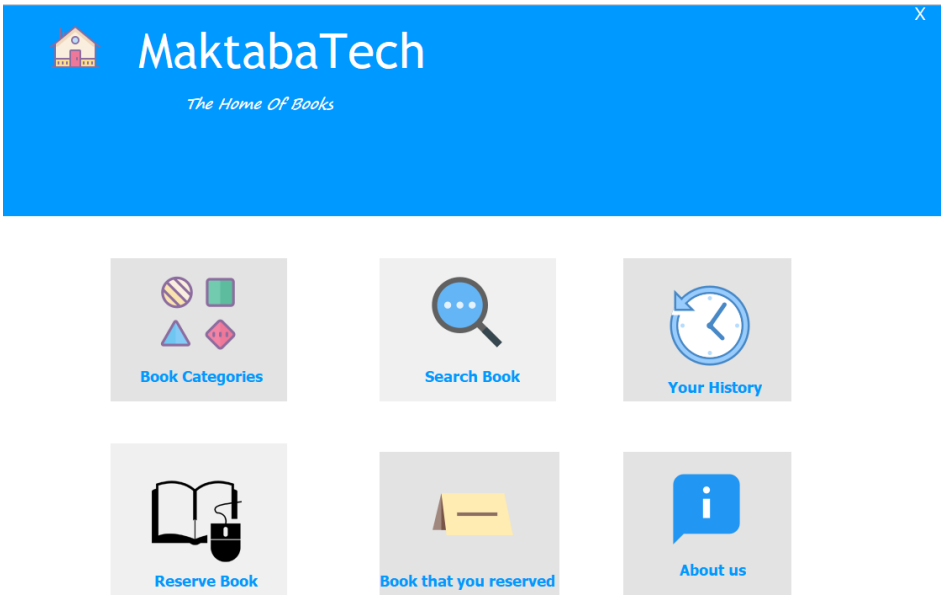
If not, you can sign up and join the family!





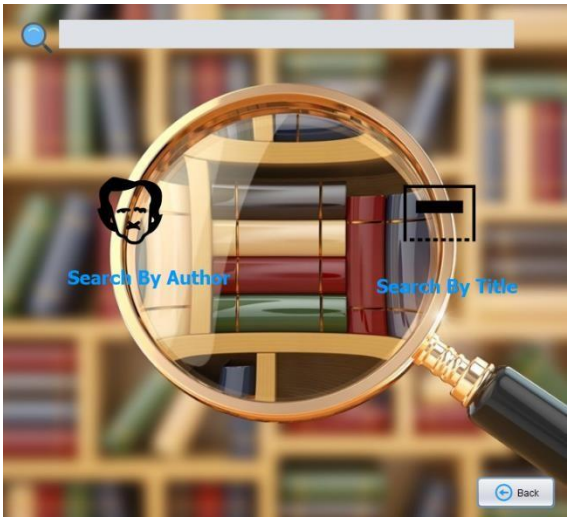


Once you log in, you will enter the E-library **Home Page** and can explore all the available services *MaktabaTech* can offer you.

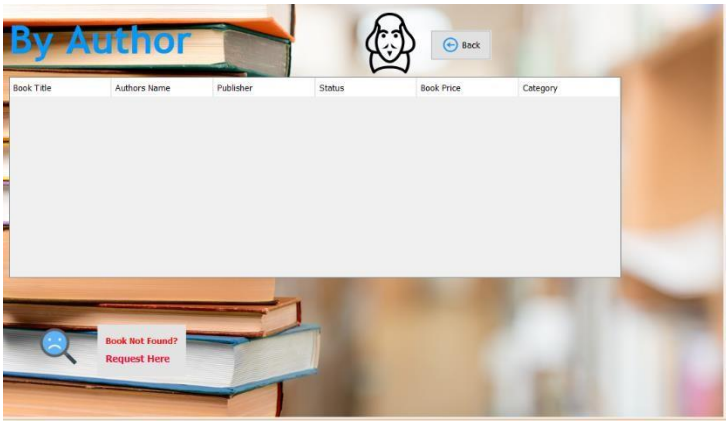


Once you are here in the Main Page, you can:

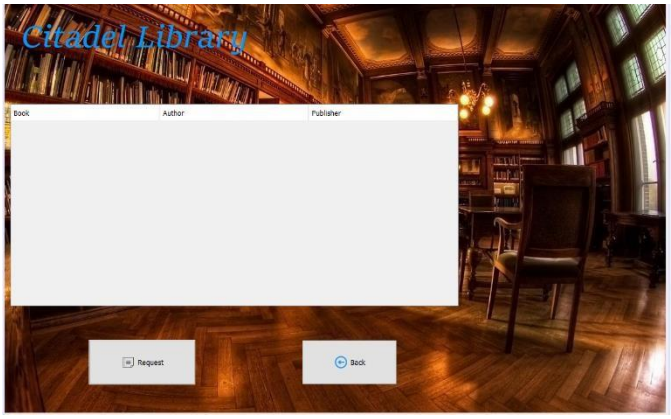
1) View different Categories of books available in the library and View the list of books available in each category
2) Search for a particular book
3) Reserve a book that you desire
4) View the books that you already reserved
5) View your interactions with MaktabaTech



You can search for the book that you want by Author or by Title to see if the book is available in our Library.

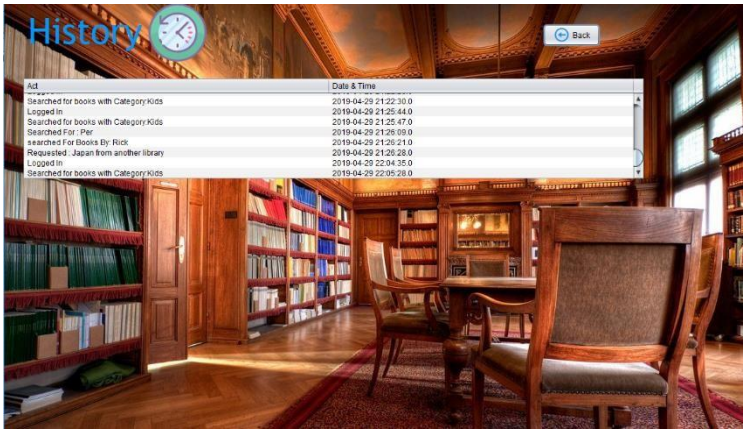


If the book is not available in our Library, no worries! You can request it from our partnering Library ***Citadel Library .***

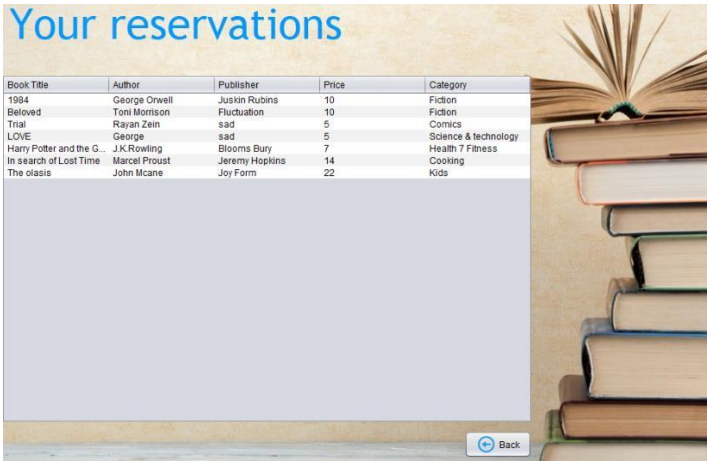


You can reserve your book that you desire by specifying its name and the return date.

MaktabaTech, ***the Home of Books*** *is also full of diverse books. You can search for books that fulfill your cravings and needs!*



You can also check you interactions with us and your reservations



Book Title	Author	Publisher	Price	Category
1984	George Orwell	Juskin Rubins	10	Fiction
Beloved	Toni Morrison	Fluctuation	10	Fiction
Trial	Rayan Zein	sad	5	Comics
LOVE	George	sad	5	Science & technology
Harry Potter and the G...	J.K.Rowling	Blooms Bury	7	Health 7 Fitness
In search of Lost Time	Marcel Proust	Jeremy Hopkins	14	Cooking
The olasis	John Mcane	Joy Form	22	Kids

Server/Client:

We created many Server functions to serve the purpose of our library.

The functions include:

- Login: checks if the username and password entered are valid and present in our database.
- Signup: Creates a new account and adds it to our database.
- DisplayCategory: View all books in our library with this category
- Reserve: Reserve a book from our library
- Request: Request book from another library
- SearchByAuthor : Search for books by Author
- SearchByTitle: search for books by title name
- ViewReserves: View all reservations by the user
- ViewHistory: Track the user's actions

We inserted, selected, and updated our database using queries.

We first tried all the functions on Eclipse (using the client code given by Dr. Matta).

We also did a JTable Function that displayed to us the info on the client side in Eclipse.

The following is the testing Jtable code for one of our functions:

```
JFrame jtabl;
jtabl = new JFrame();
int j = 0;
Vector<Vector<String>> data = new Vector<Vector<String>>();
while (!split[j].equals("FINISH_Answer")) {
    String splitj[] = split[j].split(",");
    Vector<String> entry = new Vector<String>();
    entry.add(splitj[0]);
    entry.add(splitj[1]);
    entry.add(splitj[2]);    //num of cols
    entry.add(splitj[3]);
    entry.add(splitj[4]);
    //entry.add(splitj[5]);


    data.add(entry);
    j = j + 1;
}
Vector<String> col = new Vector<String>();
col.add("Booktitle");
col.add("AuthorName");
col.add("Publisher");
col.add("price");
//col.add("price");
col.add("category");
JTable jr = new JTable (data,col);
jr.setBounds(300, 400, 200, 3000);
JScrollPane scroller = new JScrollPane(jr);
jtabl.add(scroller);
jtabl.setSize(3000, 4000);
jtabl.setVisible(true);
jtabl.setDefaultCloseOperation(2); */
```

After working out all the functions between the client and the server on Eclipse, we worked on connecting them using GUI.

Our database “library management system” includes 7 entities (tables) in total.

- 6 tables represent our library and the 7th table “library_2” represents a different library’s available books.

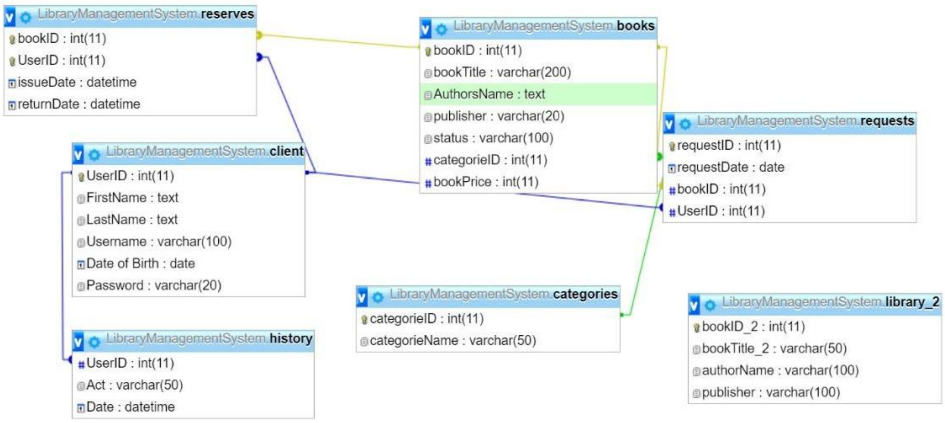


Table “categories” stores all available categories and each category is identified by a unique key and has the following attributes:

- categorieID = primary key. **Note: primary keys throughout this project are autoincremented.**
- categorieName

+ Options

				categorieID	categorieName
Edit	Copy	Delete		1	Kids
Edit	Copy	Delete		2	Biographies
Edit	Copy	Delete		3	lifestyle
Edit	Copy	Delete		4	Arts & Music
Edit	Copy	Delete		5	Comics
Edit	Copy	Delete		6	Cooking
Edit	Copy	Delete		7	Science & technology
Edit	Copy	Delete		8	Health 7 Fitness
Edit	Copy	Delete		9	Sports
Edit	Copy	Delete		10	Fiction
Edit	Copy	Delete		11	Medical
Edit	Copy	Delete		12	Religion
Edit	Copy	Delete		13	Romance
Edit	Copy	Delete		14	Travel

Table "books" stores all available books, each book is identified by a unique key and has the following attributes:

- bookID = primary key
- bookTitle
- AuthorsName
- Publisher
- Status: represents the status of the book available/reserved, this attribute is updated using a function reserveBook in our java server program.
- categorieID: is a foreign key that reference the primary key CategorieID in the categories table.
- bookPrice

+ Options

	bookID	bookTitle	AuthorsName	publisher	status	categorieID	bookPrice
Edit Copy Delete	1	1984	George Orwell	Juskin Rubins	Reserved	10	10
Edit Copy Delete	2	Beloved	Toni Morrison	Fluctuation	Reserved	10	10
Edit Copy Delete	3	Trial	Rayan Zein	sad	Reserved	5	5
Edit Copy Delete	4	LOVE	George	sad	Reserved	7	5
Edit Copy Delete	5	Harry Potter and the Goblet	J.K.Rowling	Blooms Bury	Available	8	7
Edit Copy Delete	6	In search of Lost Time	Marcel Proust	Jeremy Hopkins	Available	6	14
Edit Copy Delete	7	Harry Potter and the Goblet	J.K.Rowling	Blooms Bury	Reserved	8	7
Edit Copy Delete	8	In search of Lost Time	Marcel Proust	Jeremy Hopkins	Reserved	6	14
Edit Copy Delete	9	Don Quixote	Miguel de Cervantes	Jams Turb	Reserved	11	7
Edit Copy Delete	10	Ulysses	James Joyce	Universal Publishes	Reserved	7	12
Edit Copy Delete	11	The Great Gatsby	F.Scott Fitzgerald	Joy Form	Reserved	4	7
Edit Copy Delete	12	Moby Dick	Herman Melville	James Charlie	Reserved	12	16
Edit Copy Delete	13	Hamlet	William Shakespeare	Pearson	Reserved	14	15
Edit Copy Delete	14	War and Peace	Leo Tolstoy	RELX Group	Reserved	6	15

Console

Table "client" stores the client's usernames, passwords and personal information, each client is identified by a unique key and has the following attributes:

- UserID = primary key
- FirstName
- LastName
- UserName
- Date of Birth
- Password

UserID will be used throughout the program to reserve, request and keep track of the user's transactions as we will see shortly.

UserName, password and personal information will be inputted by the client using the signup function and stored in the database, then UserName and password will be used in the login function by the user to login to the library.

+ Options

	UserID	FirstName	LastName	Username	Date of Birth	Password
Edit Copy Delete	0	Rayan	Zein	riz06	2019-03-05	Nshallah
Edit Copy Delete	1	Ali	Yassine	amy19	2019-04-24	Maya1999
Edit Copy Delete	2	Sarah	El-Annan	sje18	2018-11-28	IloveHP
Edit Copy Delete	3	Dr. Jad	Matta	jm97	2019-04-30	IloveNetworking
Edit Copy Delete	4	Mariam	Hoteit	mmh27	2019-04-10	London

Console

Table "reserve" stores record of all reserved books and identifies each reservation by a unique key. It has the following attributes:

- BookID: is a foreign key that references bookID of table books
- UserID: is a foreign key that references UserID of table client

<u>BookID and UserID are a composite primary key for table reserves since each book and only be reserved by one Client.</u>

- IssueDate
- returnDate

+ Options

	bookID	UserID	issueDate	returnDate
Edit Copy Delete	1	0	2019-04-28 19:18:59	2019-07-17 00:00:00
Edit Copy Delete	2	0	2019-04-28 19:19:31	2019-06-17 00:00:00
Edit Copy Delete	3	0	2019-04-28 19:19:57	2019-08-25 00:00:00
Edit Copy Delete	4	0	2019-04-28 20:40:40	2019-07-11 00:00:00
Edit Copy Delete	7	0	2019-04-28 21:51:33	2019-08-22 00:00:00
Edit Copy Delete	8	0	2019-04-28 21:41:03	2019-05-10 00:00:00
Edit Copy Delete	34	0	2019-04-29 00:29:54	2019-07-25 00:00:00

Table "library_2" represents the table books of a different library from which we will be requesting non available books, similarly to table books, each is identified by a unique key. It has the following attributes:

- <u>BookID_2 = primary key</u>
- BookTitle_2
- AuthorsName
- Publisher

	bookID_2	bookTitle_2	authorName	publisher
Edit Copy Delete	1	Japan	James Arthur	Lovely Publishes
Edit Copy Delete	2	Senea	Simon Rolls	Jennings
Edit Copy Delete	3	The Adventures of Huckleberry Finn	Mark Twain	Pan Macmillan
Edit Copy Delete	4	The Iliad	Homer	Egmont
Edit Copy Delete	5	Lolita	Vladimir Nabokov	HarperCollins
Edit Copy Delete	6	Crime and Punishment	Fyodor Dostoyevsky	Faber Alliance
Edit Copy Delete	7	Alice's Adventures in Wonderland	Lewis Carroll	Disney

Table "requests" stores records of all requested books and identifies each request by a unique key. It has the following attributes:

- requestID: is a primary key
- RequestDate
- UserID: is a foreign key that references UserID of table client
- BookID is a foreign key that references bookID_2 of library_2

	requestID	requestDate	bookID	UserID
Edit Copy Delete	1	2019-02-02	3	6
Edit Copy Delete	2	2016-08-08	2	3
Edit Copy Delete	3	2019-04-29	1	1
Edit Copy Delete	4	2019-04-29	2	0
Edit Copy Delete	5	2019-04-29	2	1
Edit Copy Delete	6	2019-04-29	2	8
Edit Copy Delete	7	2019-04-29	1	0

Table "history" stores records of the interactions between client and server books and identifies each record by a unique key. It has the following attributes:

- UserID: is a primary key and a foreign key that references UserID of table client
- Act: description of the type of action the client took
- Date

UserID	Act	Date
4	Logged In	2019-04-17 11:17:08
4	Signed UP	2019-04-10 03:17:45
4	Reserved : Don Quixote	2019-02-02 00:00:00
4	Reserved : Trial	2019-02-02 00:00:00
4	Reserved : Ulysses	2019-02-17 00:00:00
8	Reserved : Ulysses	2019-02-17 00:00:00
8	Reserved : In search of Lost Time	2019-02-11 00:00:00
8	Reserved : Moby Dick	2019-02-11 00:00:00
7	Searched For : The	2017-12-12 00:00:00
7	Searched For : The	2017-12-12 00:00:00
7	Searched For : The	2017-12-12 00:00:00
5	searched For Books By: Rayan	2018-11-01 00:00:00
5	Searched for books with Category:Cooking	2018-07-07 00:00:00
2	Viewed his/her reservations	2017-11-06 00:00:00
2	Viewed his/her reservations	2017-11-06 00:00:00
2	Viewed his/her reservations	2017-11-06 00:00:00
11	First Became A Member	2016-07-07 00:00:00
3	Requested : Seneafrom another library	2016-08-08 00:00:00

Problems/Difficulties Encountered:

- At first, it was really tough to start. We didn't know from where to start.
- We faced issues in getting information from our database since it is connected using many IDs.
  However, we found online the INNER JOIN query which helped us a lot.
- We first did the GUI design of the pages as jFrames, which is a wrong GUI design. So we changed all our pages to dialogs and realized the positive difference.
- We didn't know where to put the Socket and connection initiation code at first in the client code.
- It was very difficult to find the way to display the information in jTables.
  In addition, we faced many errors regarding this issue.
  However, after one table worked well, the others passed by gently.

Final Note: This project really pushed us to our limits and took us out of our comfort zones. Other than being grouped with people that you don't know by random, but the project is difficult and needs a lot of work especially since we are not familiar with any of: Java, GUI and MYSQL.
It really improved our skills and knowledge in these 3 topics.