

# UNIVERSITY LIBRARY

## MANAGEMENT

### BY:

- Menna Elminshawy 20217011
- Radwa Belal 20217005
- Nada Ahmed 20217013
- Ahmad Wael Abdelaziz 20216016
- Essam Alaa Nady 20216064

TA: Eng. Marwa Hussein.

## We have six entities as followed:

1. **ADMIN (Strong Entity)**
  - ADMIN\_ID: Identifier for the admin
  - EMAIL: Email address of the admin
  - PASSWORD: Password for admin authentication
2. **STUDENT (Strong Entity)**
  - STUDENT\_ID: Identifier for the student
  - EMAIL: Email address of the student
  - PASSWORD: Password for student authentication
3. **SUPPORT (Strong Entity)**
  - SUPPORT\_ID: Identifier for the support staff
  - EMAIL: Email address of the support staff
  - PASSWORD: Password for support staff authentication
4. **BOOK (Strong Entity)**
  - ISBN: Identifier for the book
  - AUTHOR\_ID: Identifier for the book's author
  - BOOK\_TITLE: Title of the book
  - PUBLICATION\_YEAR: Year of publication for the book
5. **AUTHOR (Strong Entity)**
  - AUTHOR\_ID: Identifier for the author
  - AUTHOR\_NAME: Name of the author
6. **BORROWED\_BOOKS (Weak Entity)**
  - BORROWED\_DATE: Date when the book was borrowed
  - ISBN: Identifier of the borrowed book
  - RETURN\_DATE: Date when the borrowed book is expected to be returned

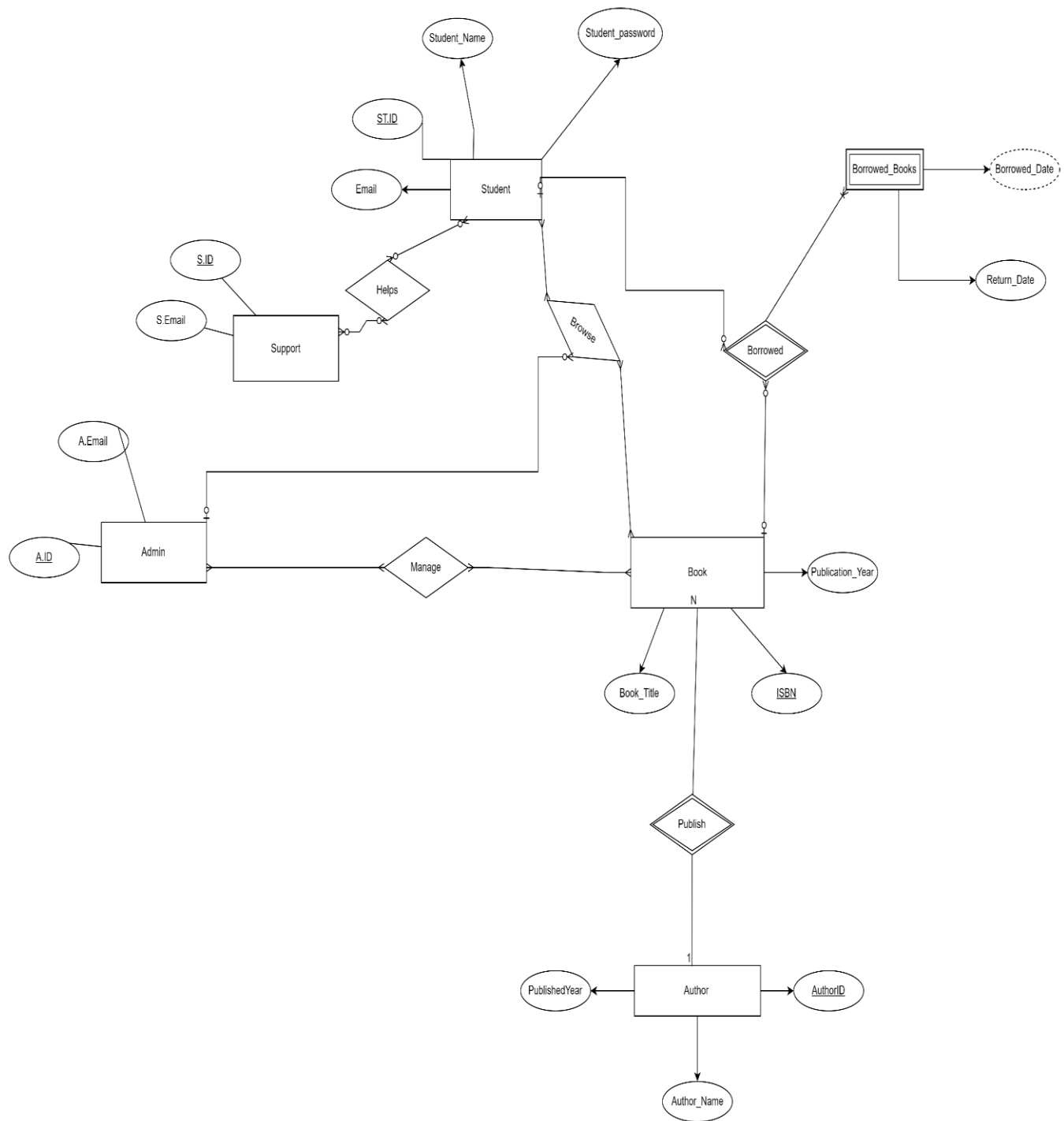
Entity Types	Strong or weak	Attributes
1. ADMIN	Strong	( <u>ADMIN_ID</u> , EMAIL, PASSWORD)
2. STUDENT	Strong	( <u>STUDENT_ID</u> , EMAIL, PASSWORD, EMAIL)
3. SUPPORT	Strong	( <u>SUPPORT_ID</u> , EMAIL, PASSWORD)
4. BOOK	Strong	( <u>ISBN</u> , AUTHOR_ID, BOOK_TITLE, PUBLICATION_YEAR)
5. AUTHOR	Strong	( <u>AUTHOR_ID</u> , AUTHOR_NAME)
6. BORROWED_BOOKS	Weak	( <u>ISBN, BORROWED_DATE, RETURN_DATE</u> )

## We have Five Relationship as followed:

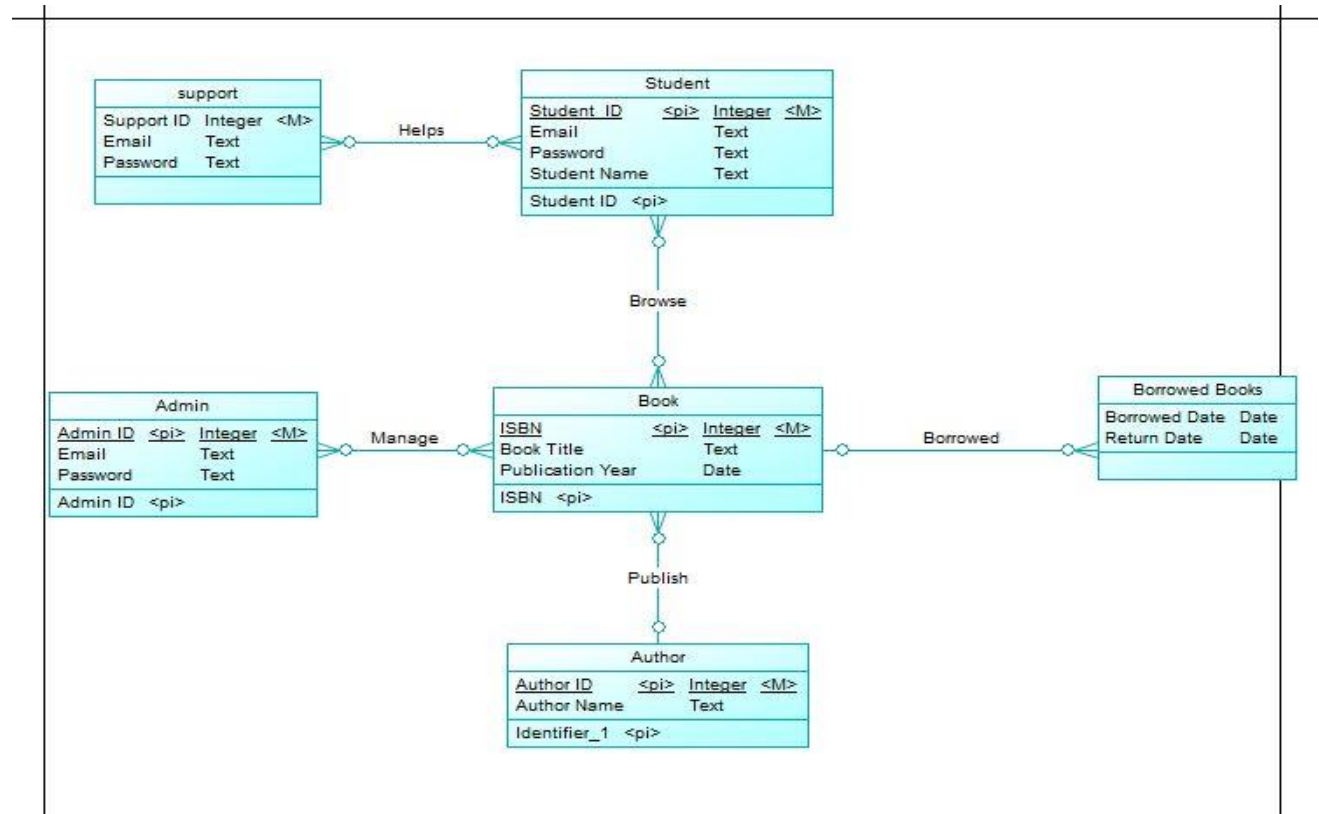
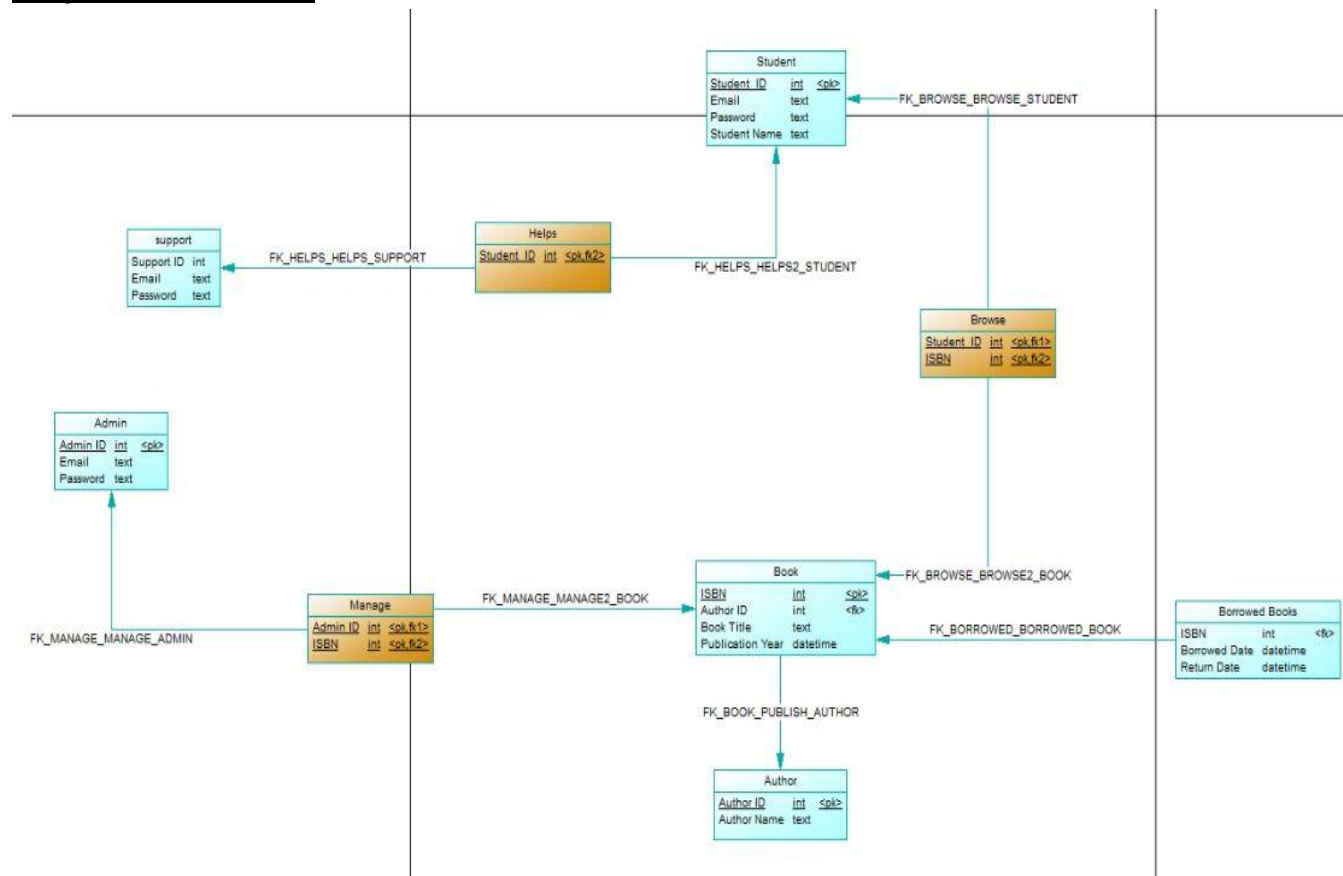
1. **Manage: Relationship between Admin and Book**
  - Cardinality: Many-to-Many (M: N)
  - Participation: Mandatory for Admin, Optional for Book
2. **Helps: Relationship between Support and Student**
  - Cardinality: Many-to-Many (M: N)
  - Participation: Mandatory for Support, Optional for Student
3. **Borrowed: Relationship between Book and Borrowed\_Books**
  - Cardinality: One-to-Many (1:M)
  - Participation: Mandatory for Borrowed\_Books, Optional for Book
4. **Publish: Relationship between Book and Author**
  - Cardinality: Many-to-One (M:1)
  - Participation: Mandatory for Book, Mandatory for Author (one author per book)
5. **Browse: Relationship between Book and Student**
  - Cardinality: Many-to-Many (M:N)
  - Participation: Mandatory for Book, Optional for Student

Relationship Type	Between	Cardinality	Participation
Mange	Admin-Book	M: N	Mandatory -optional
Helps	Support-student	M: N	Mandatory -optional
Borrowed	Book-Borrowed	1:M	Mandatory-optional
Publish	Book-Author	M:1	Mandatory -one
Browse	Book-student	M: N	Mandatory -optional

## Entity Relationship Diagram:



## Physical Model:



## **C# Program**

Our project is about UNIVERSITY Library Management.

First, the user will sign up based on specific criteria: student, admin, or someone who wants to search for a book.

If a student signs up, they will need to enter their ID, name, email, and password. If the admin signs up, they will need to enter their ID, password, and email. Once the admin signs in, they can manage the database.

The "Manage Database" button provides the admin with various functionalities. One of the key features is the ability to insert, update, or delete books from the database. These operations are accessible only to the admin.

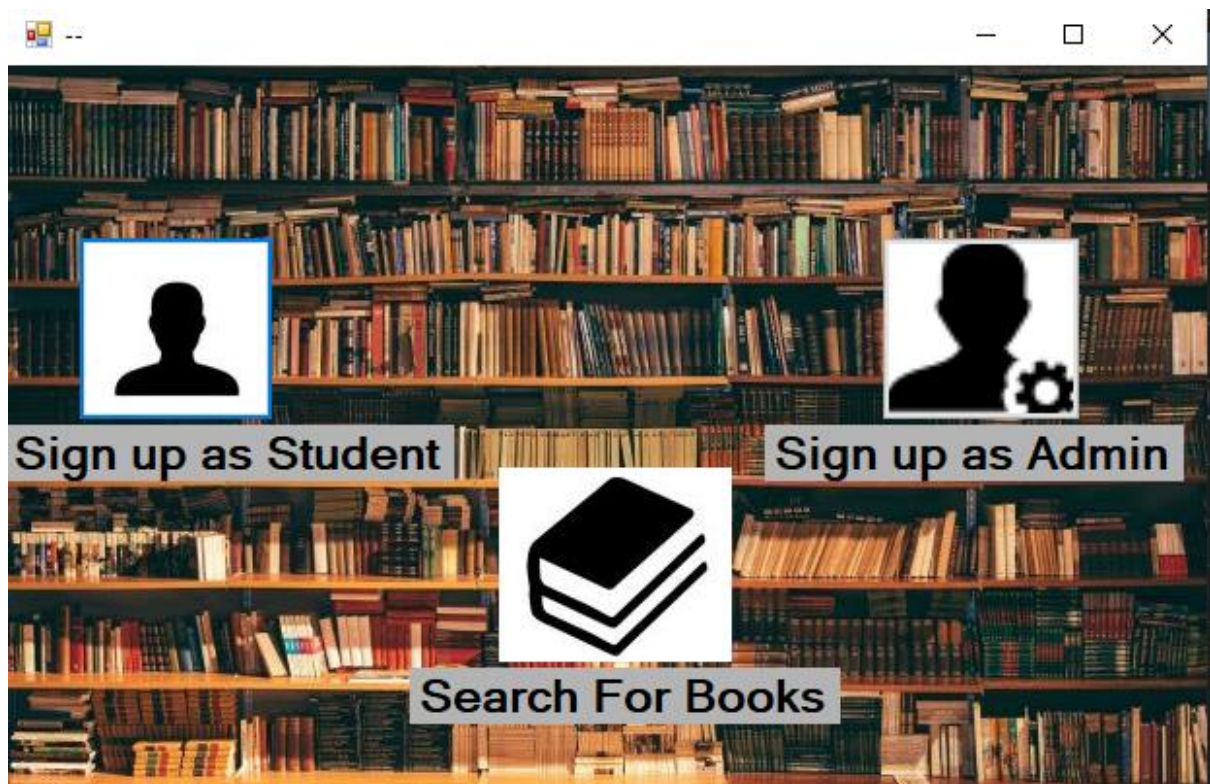
Additionally, the admin has the authority to manage users. This includes the ability to add new users directly or allow other users to register themselves.

Moreover, the system allows the admin to upload information about a borrowed book and mark it as "borrowed." This action will update the book's status in the database, indicating that it is currently unavailable due to being borrowed by a user.

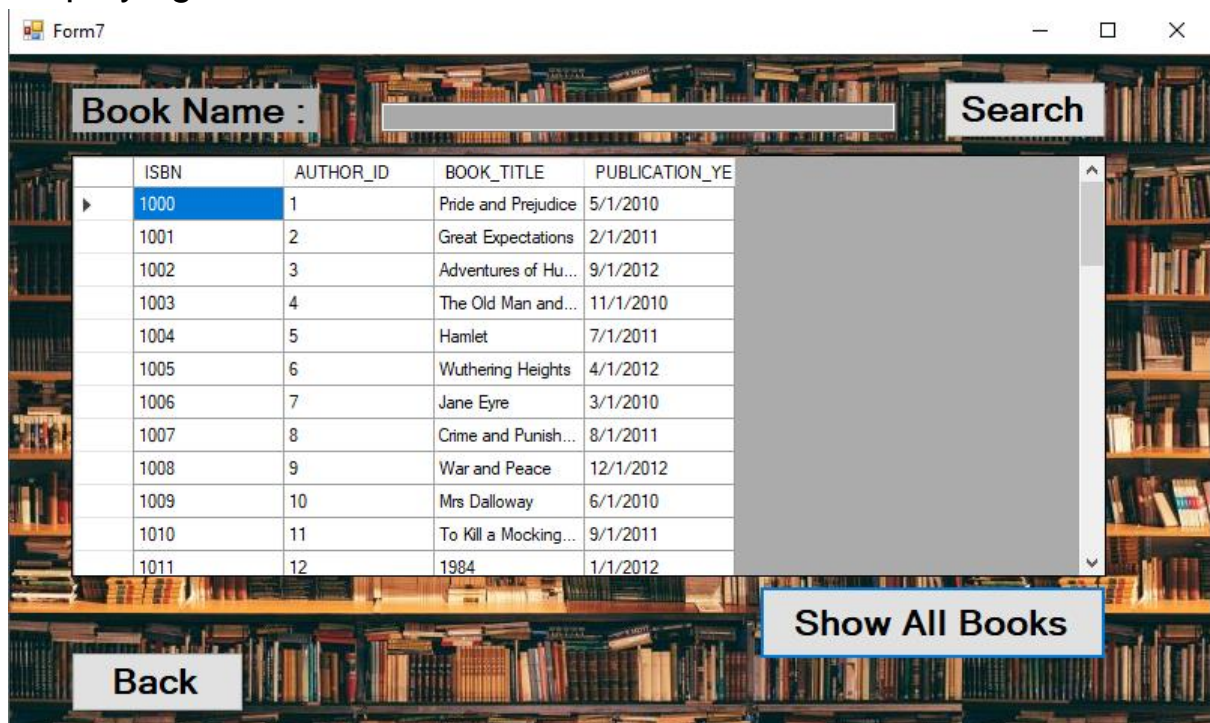
Furthermore, the admin can insert, delete, or update other items in the library using their ID and name.

If a user clicks the "Search for Book" button and enters the book title, the system will check its availability and display the result indicating whether the book is available or not. Users can also check if a book is currently borrowed by someone else.

Main Menu :



Displaying all books :





Searching for a specific book and checking availability:

Book Name :

	ISBN	AUTHOR_ID	BOOK_TITLE	PUBLICATION_YE	BORROWED_DAT	ISBN1	RETUR
▶	1000	1	Pride and Prejudice	5/1/2010			
*							

Student Signing up:



Admin Updating student info:

Form6

Student\_id

Student\_name

Student\_email

Student\_password

insert

update

delete

Back

Admin Management options:

Form4

insert/delete/update Book

insert/delete/update user

insert/delete/update author

Back