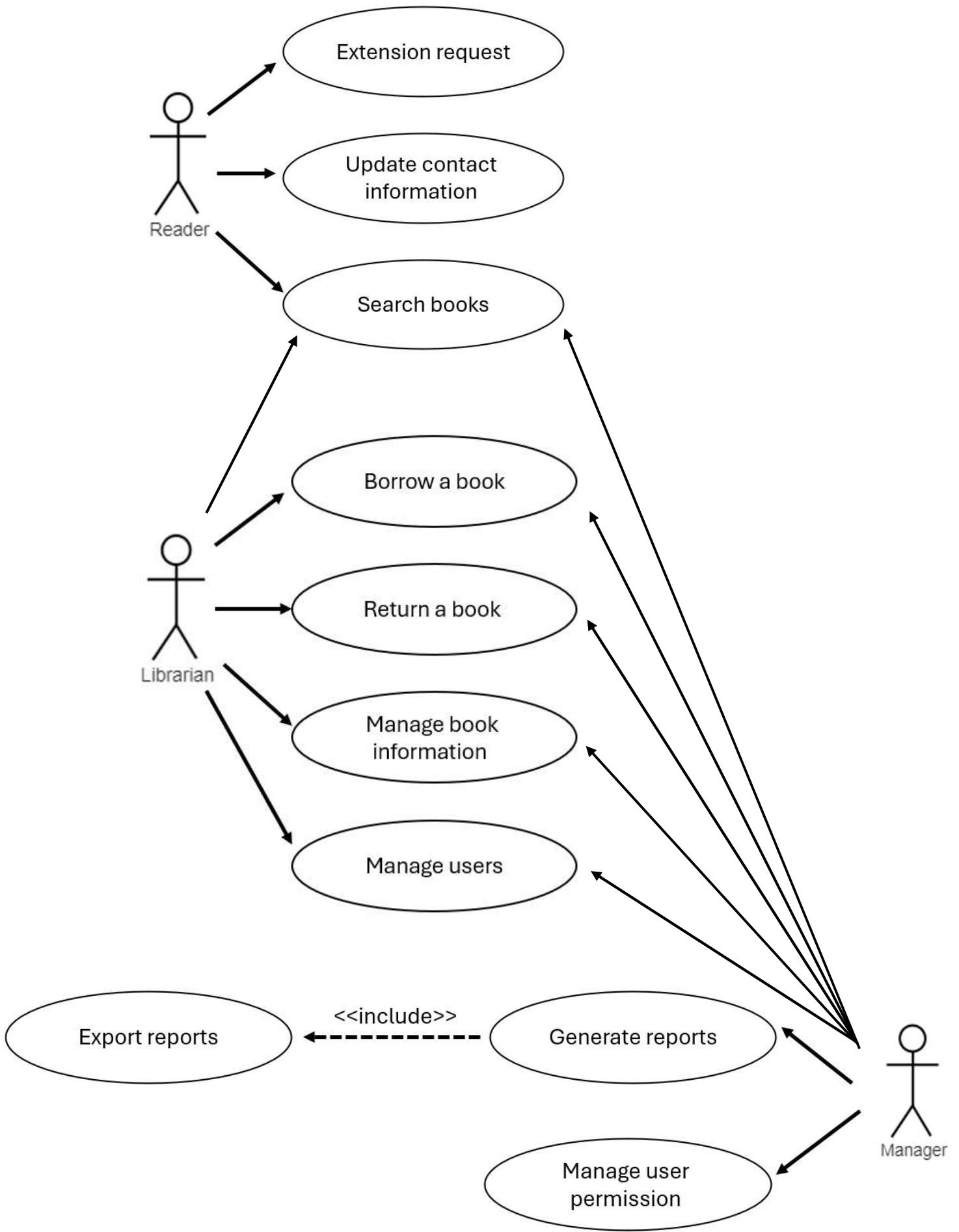
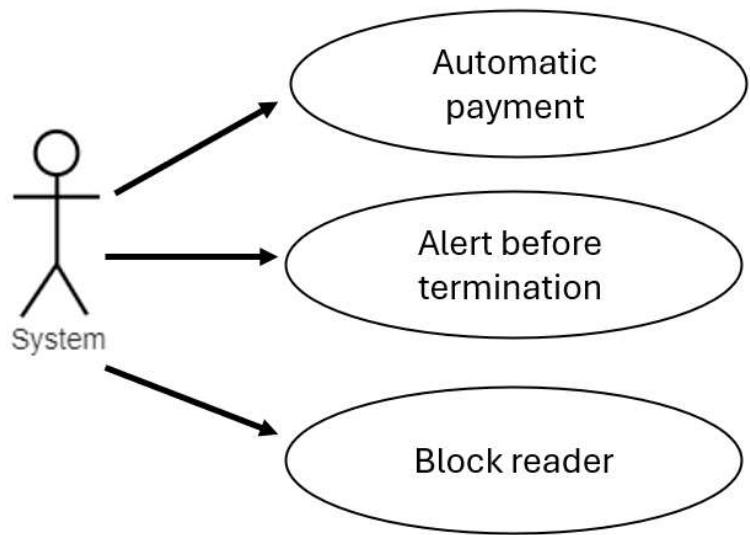


## : USE CASE DIAGRAM





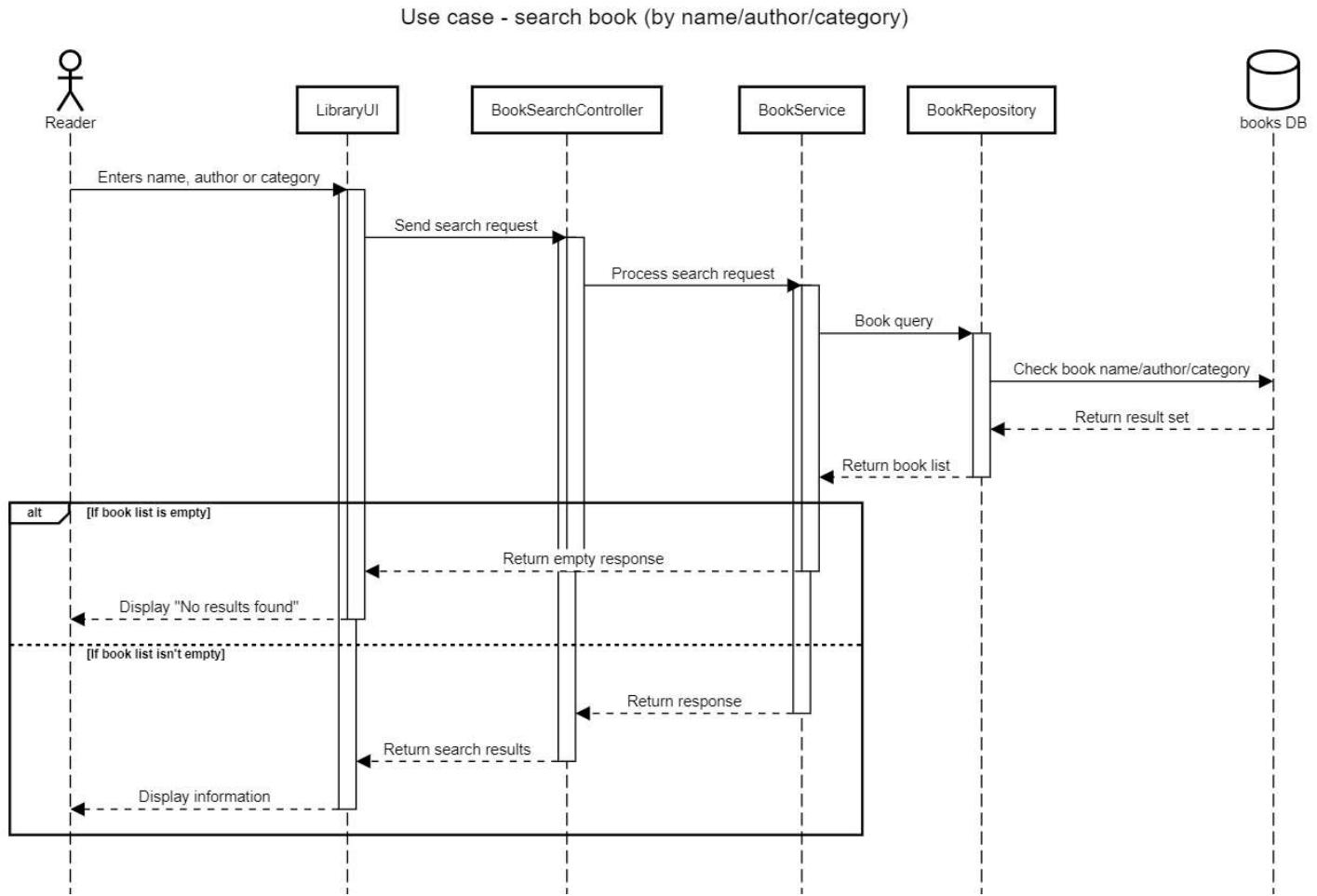
## תיאור USE CASE DIAGRAM

### : ACTORS •

- Reader: הקורא יכול לחפש ספרים ולצפות בפרטי הספר, לבקש להאריך את משך השאלה ולעדכן את פרטי ההתקשרות עמו.
- Librarian: הספרן יכול לחפש ספרים ולצפות בפרטי הספר, לבצע פעולה השאלת והחזרת ספר, לנוהל את פרטי הספרים ואת פרטי המשתמשים.
- Manager: בנוסף ליכולותיו של הספרן, המנהל יוכל להפיק דוחות סטטיסטיים ולשנות את הרשות הגישה של הספרנים.
- System: המערכת תתריע ללקוח שימוש בשאלת השאלה עומדת לפוג, תבצע חיבור אוטומטי במידה ולא החזיר את הספר ותחסום קורא שלא החזיר את הספר בזמן מעל 3 פעמים.

### : USE CASES •

- Extension request: הגשת בקשה להארכת משך השאלה.
- Update contact information: עדכון פרטי ההתקשרות.
- Search books: חיפוש ספרים לפי בותרת, מחבר או קטgorיה.
- Borrow a book: ביצוע פעולה השאלת ספר לקורא.
- Return a book: ביצוע פעולה החזרת ספר שקורא החזיר.
- Manage book information: עריכת ומחיקת מידע על ספרים (שם, סופר, מק"ט, קטgorיה, סטטוס (זמן/מושאל) וכו').
- Manage users: עריכת ומחיקת מידע על קוראים (שם, מספר קורא, היסטורית השאלות).
- Generate reports: הפקת דוחות סטטיסטיים למנהלים על פעילות הספרייה.
- Export reports: ייצוא הדוחות הסטטיסטיים לקובץ EXCEL או PDF.
- Manage user permission: שינוי הרשות גישה לכל ספרן.
- Alert before termination: התראה שהזמן המוקצב להחזרת הספר עומד לפוג והתרעה טרם חיבור.
- Block reader: חסימת ל��וח שלא החזיר ספר בזמן מעל 3 פעמים.
- Automatic payment: חיבור אוטומטי לקורא שלא החזיר את הספר בזמן.



.1. Reader :Actors – מבצע חיפוש ספר לפי שם, מחבר או קטגוריה דרך המערכת.

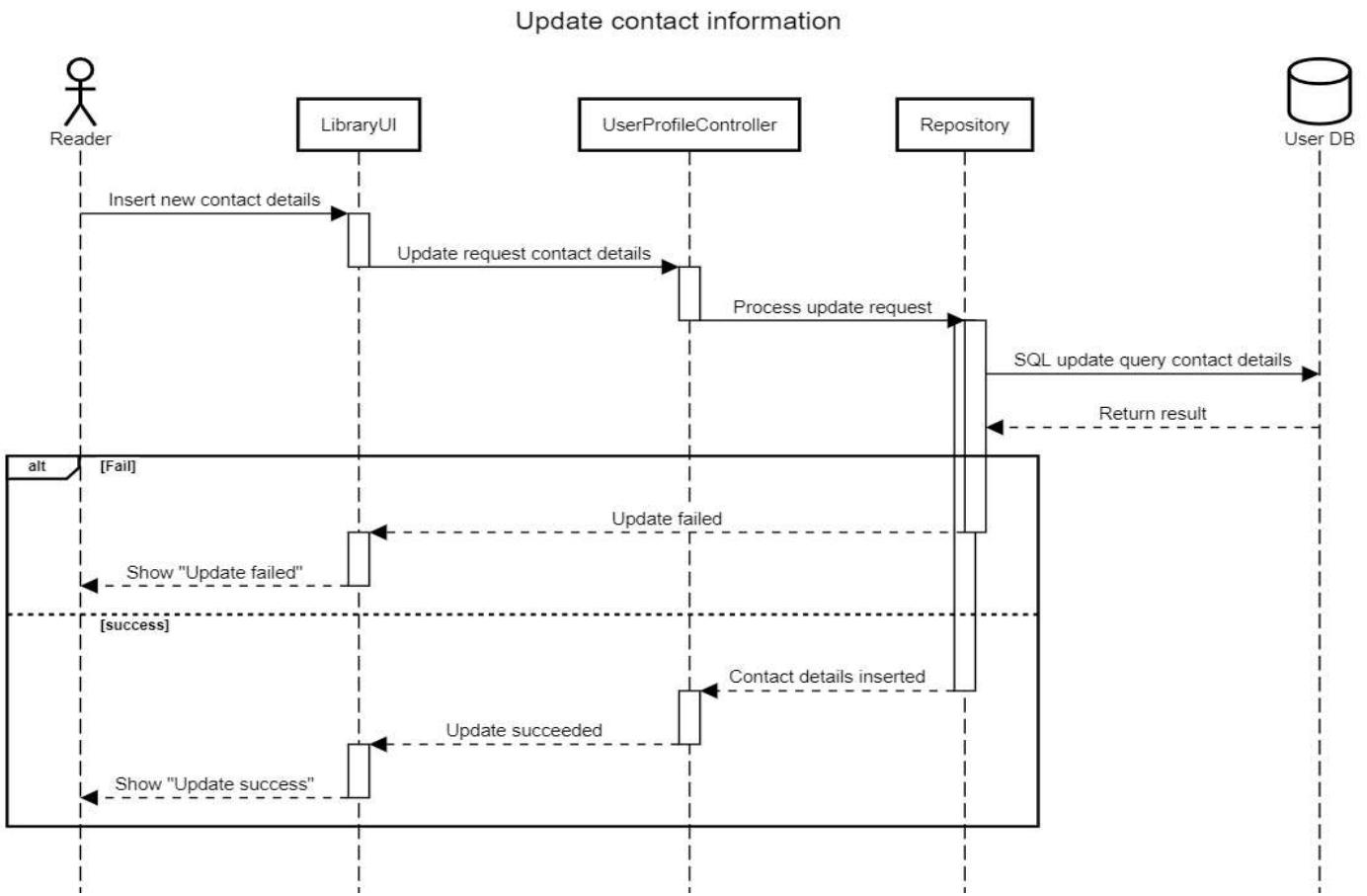
.2. System components :System components

- a. LibraryUI – ממשק המשתמש, מקבל את הקלט ו מעביר ל- BookSearchController .
- b. BookSearchController – מתווך בין ה-UI לבין השירותים.
- c. BookService – מטפל בלוגיקת החיפוש.
- d. BookRepository – מבצע את השאלה בפועל.
- e. Books DB – מסד נתונים שמכיל את כל המידע בנוגע בספרים.

.3. Process :

- a. הקורא מזין את פרטי החיפוש.
- b. המערכת שולחת את בקשת החיפוש עד מסד הנתונים.
- c. מבצע חיפוש במסד הנתונים.

d. במידה וחזרה רשימה ריקה : מוחזרת הודעה "No results found"  
e. במידה והרשימה שחזרה אינה ריקה היא תוצג לקורא.

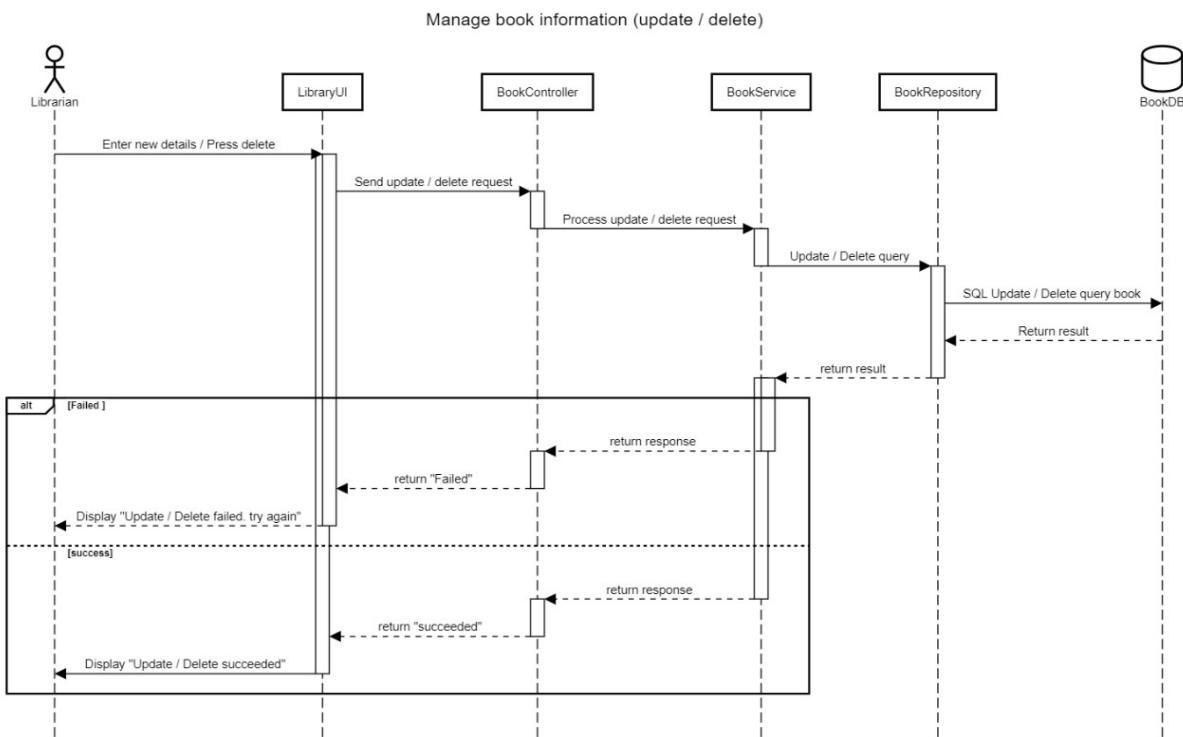


### Reader :Actors .1 :System components .2

- a. IU – ממשק המשתמש, מקבל את הקלט ו מעביר ל- .Controller
- b. UserProfileController – מתווך בין ה-IU לבין המשרת.
- c. Repository – מבצע את השאלה בפועל למסד הנתונים.
- f. User DB – מסד נתונים ש מכיל את כל המידע בנוגע לקוראים.

### :Process .3

- a. הקורא מזין פרטי התקשרות חדשים.
- b. ה-IU שלוח ל-Controller בקשה עדכון.
- c. ה-Controller מגלה את הבקשה ל-Repository שמבצע את השאלה ה-SQL למסד הנתונים.
- d. במידה ומסד הנתונים עדכן בהצלחה – תוחזר הודעה מתאימה לקורא.
- e. במידה ומסד הנתונים נכשל בעדכון – תוחזר הודעה כישלון לקורא.



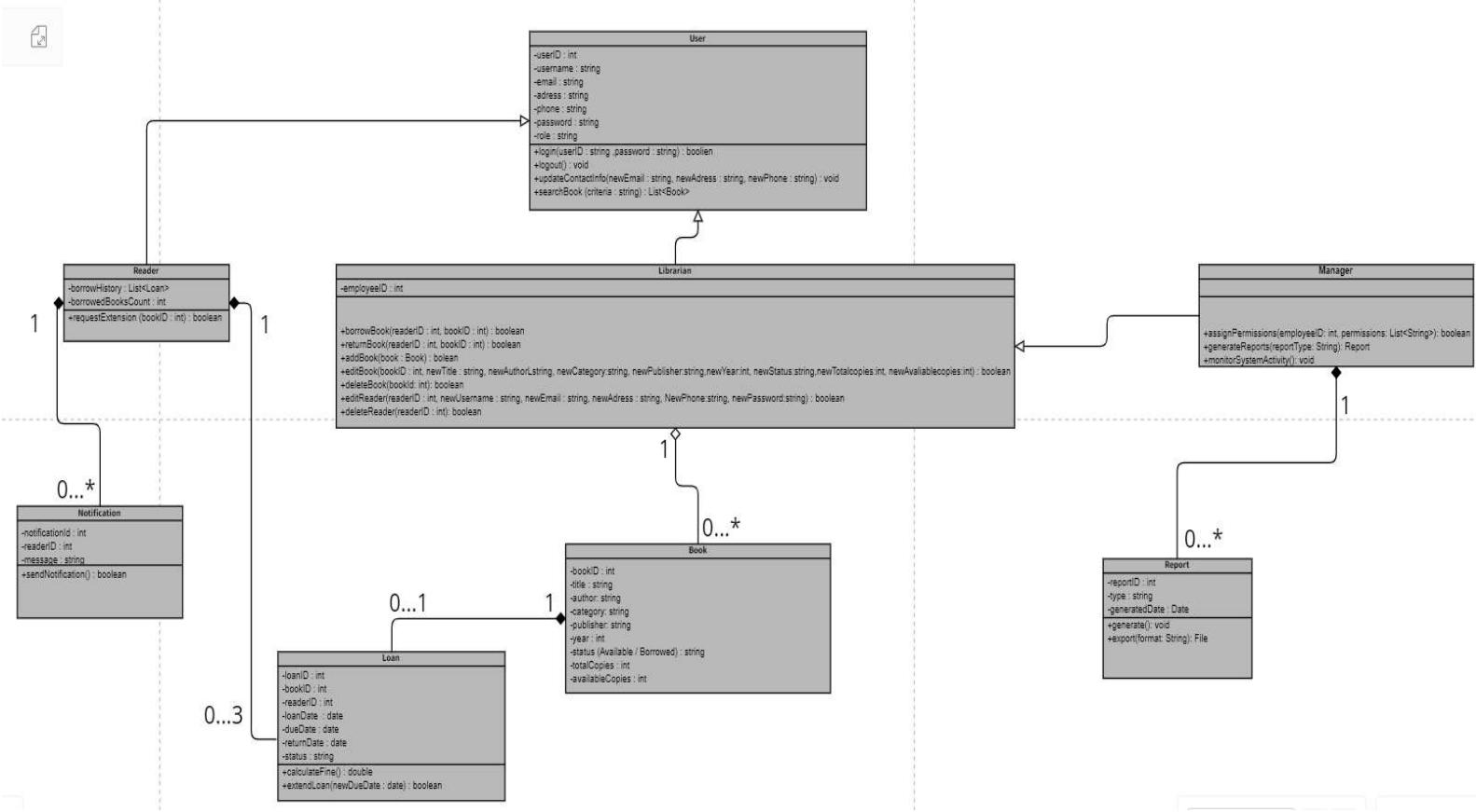
**1. Librarian :Actors** – מזין פרטים חדשים לעדכון הספר / מבקש למחוק ספר.

## **2. :System components**

- a. LibraryUI – מסמך המשתמש, מקבל את הקלט ו מעביר ל-Controller.
- b. BookController – מתווך בין ה-UI לבין הממשק.
- c. BookService – מטפל בלוגיקת העדכון / המחיקה.
- d. BookRepository - מבצע את השאלה בהתאם לפועל למסד הנתונים.
- e. BookDB – מסד נתונים ש מכיל את כל המידע בנוגע בספרים.

## **3. :Process**

- a. הספרן מזין פרטי ספר חדשים / מבקש למחוק ספר.
- b. ה-UI שלוח ל-Controller בקשה עדכון / מחיקה.
- c. ה-Controller מגלה את הבקשה עד ל-Repository שמבצע את שאלהת SQL למסד הנתונים.
- d. במידה ומסד הנתונים ביצע את הפעולה בהצלחה – תוחזר הודעה מתאימה בספרן.
- e. במידה ומסד הנתונים נכשל בעדכון – תוחזר הודעה כישלון בספרן.
- f. חשוב להזכיר – כישלון יכול להיווצר כי הבקשה לא הצליחה / הספר לא נמצא / הספר נמצא בהשאלה ולא ניתן למחוק אותו וכך ..



## : ARCHITECTURAL DIAGRAM

### • **מערכת LIBPRO מעוצבת לפי התבנית הארכיטקטונית MVC**

בחרנו בתבנית זו מכיוון שעיקר העיסוק של המערכת הוא הנגשה של מיגע מתוך מסד נתונים (Model), בצורה נוחה דרך ממשך המשמש (View).

ארכיטקטורת MVC מאפשר הפרדה בין שני חלקים המערכת. באמצעות שכבת ה-controller נוכל לוודא ניהול נכון של בקשות המשתמשים והלוגיקה הפנימית ותאפשר מודולריות נוחה בتحقוקת הפלטפורמה.

#### • : Model

רכיבי ה-Model מייצגים את הנתונים במערכת (לדוג': מחלקת User, Reader, Book וכו').

כל מחלוקת מטפלת במידע הקשור לתחומי אחריותה ומנהלת את הפעולות הרלוונטיות כמו חיפוש ספרים, עדכון קוראים, הפקת דוחות ...

#### • : View

ה-View יציג את המידע מה-Model למשתמשים.  
דוגמאות מוגelijkות :

- ה-View יוכל את הדפים בהם ספרן יוכל לבצע עדכון/מחיקת ספר.
- ה-View יוכל לקבל התראות טרם סיום זמן השאלה.

#### • : Controller

يقبل בקשות המשתמשים וישלח אותן ל-Model על מנת לבצע את הפעולה (לדוג' – בקשה לעדכון פרטי התקשורת).

לאחר הטיפול נתונים על ידי ה-Model, ה-Controller יעדכן את ה-View בהתאם והשינויים יוצגו למשתמש (לדוג' – אישור על עדכון פרטי התקשורת).

