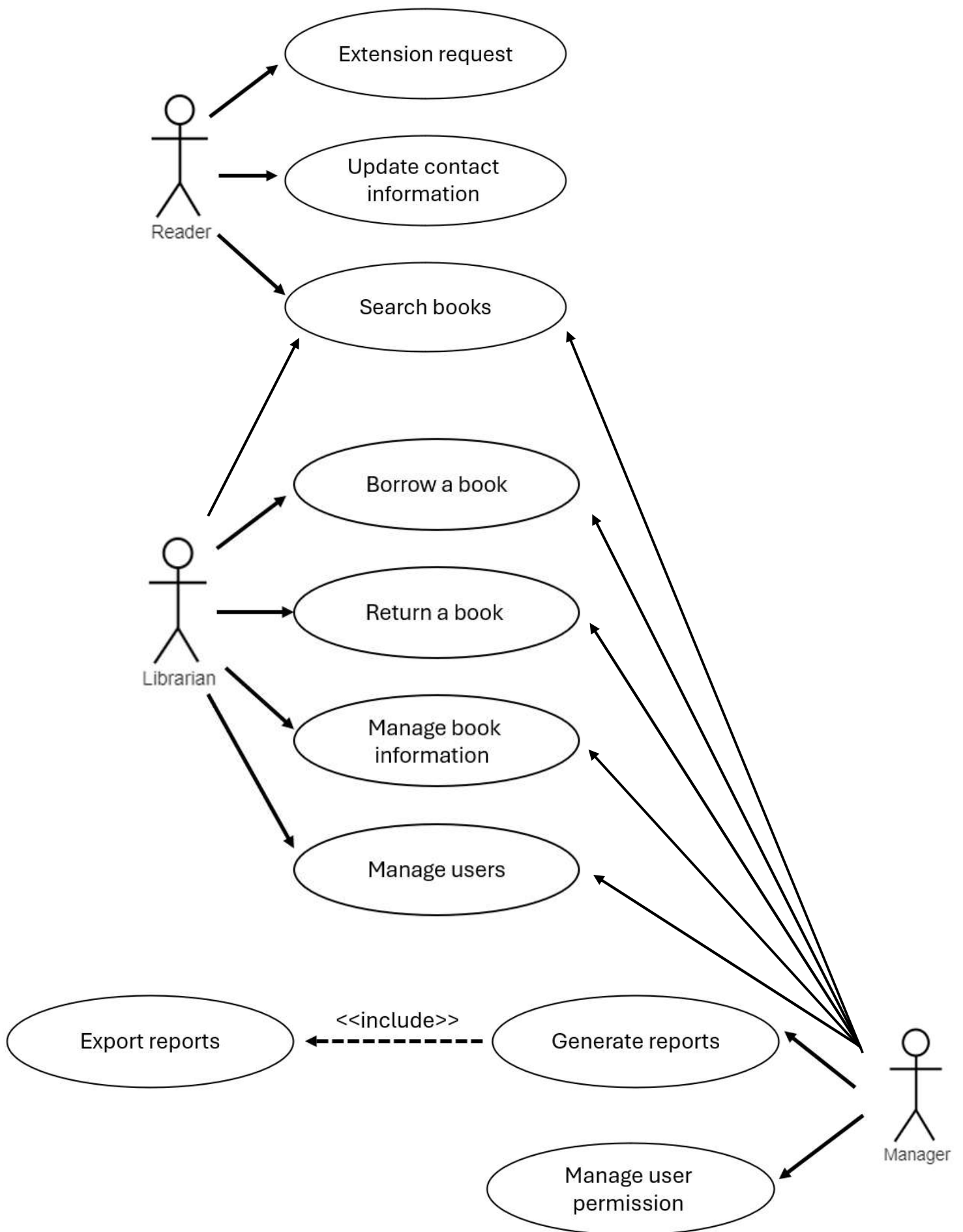
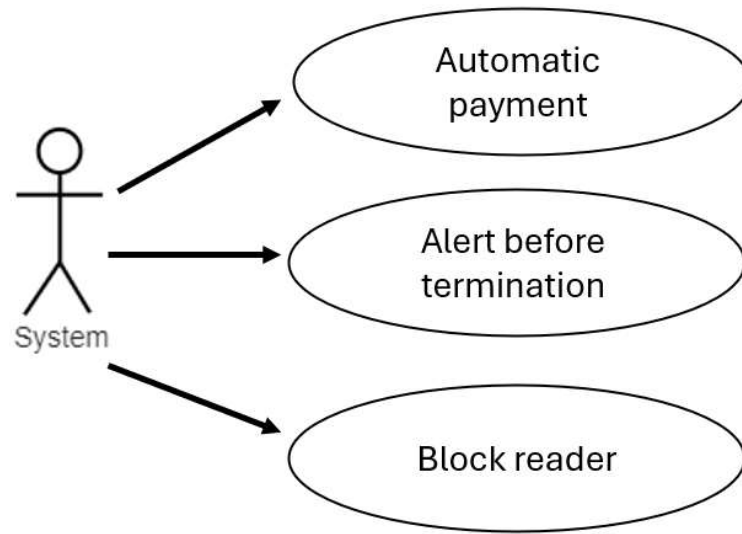


**: USE CASE DIAGRAM**





## תיאור USE CASE DIAGRAM :

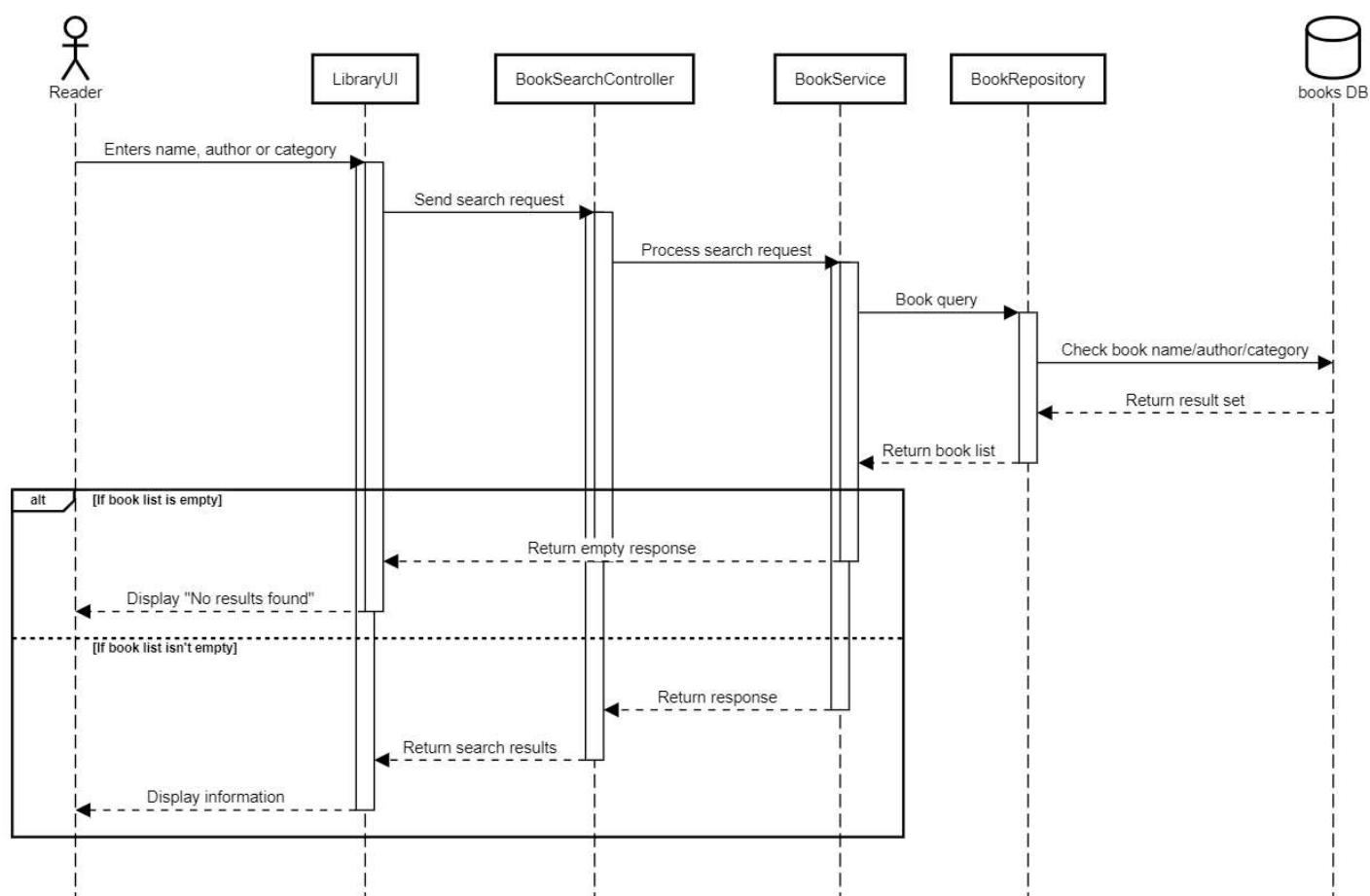
### : ACTORS •

- Reader: הקורא יוכל לחפש ספרים ולצפות בפרטי הספר, לבקש להאריך את משך ההשאלה ולעדכן את פרטי ההתקשרות עמו.
- Librarian: הספרן יוכל לחפש ספרים ולצפות בפרטי הספר, לבצע פעולת השאלת והחזרת ספר, לנהל את פרטי הספרים ואת פרטי המשתמשים.
- Manager: בנוסף ליכולותיו של הספרן, המנהל יוכל להפיק דוחות סטטיסטיים ולשנות את הרשאות הגישה של הספרנים.
- System: המערכת תתריע ללקוח שמשך ההשאלה עומד לפוג, תבצע חיוב אוטומטי במידה ולא החזיר את הספר ותחסום קורא שלא החזיר את הספר בזמן מעל 3 פעמים.

### :USE CASES •

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

## Use case - search book (by name/author/category)



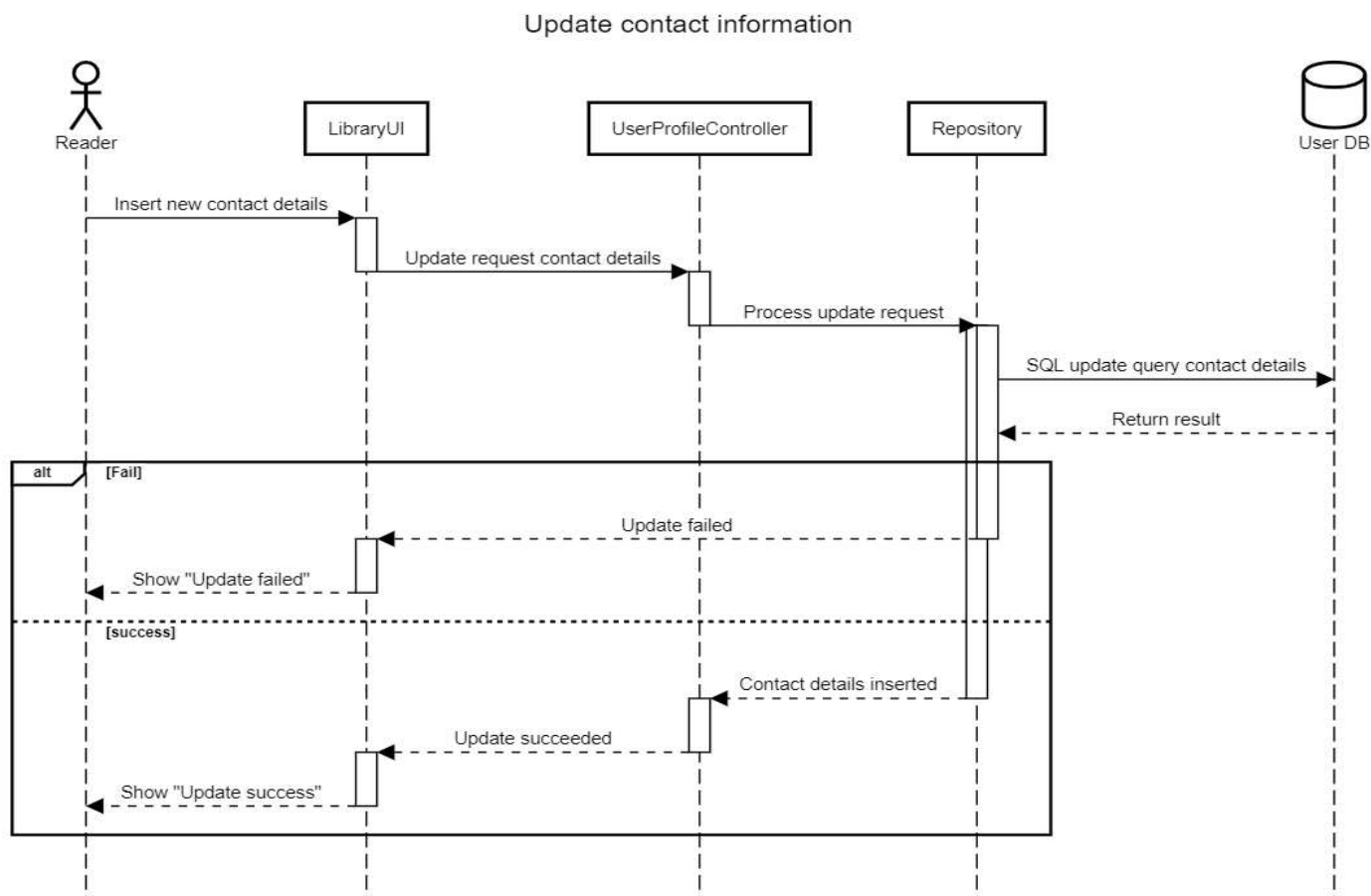
**1. Actors:** Reader - מבצע חיפוש ספר לפי שם, מחבר או קטגוריה דרך המערכת.

## 2. System components

- LibraryUI – ממשק המשתמש, מקבל את הקלט ומעביר ל-Controller.
- BookSearchController – מתווך בין ה-UI לבין השירותים.
- BookService – מטפל בלוגיקת החיפוש.
- BookRepository – מבצע את השאילתה בפועל.
- Books DB – מסד נתונים שמכיל את כל המידע בנוגע לספרים.

## 3. Process

- הקורא מזין את פרטי החיפוש.
- המערכת שולחת את בקשת החיפוש עד מסד הנתונים.
- מתבצע חיפוש במסד הנתונים.
- במידה וחזרה רשימה ריקה: מוחזרת הודעת "No results found"
- במידה והרשימה שחזרה אינה ריקה היא תוצג לקורא.



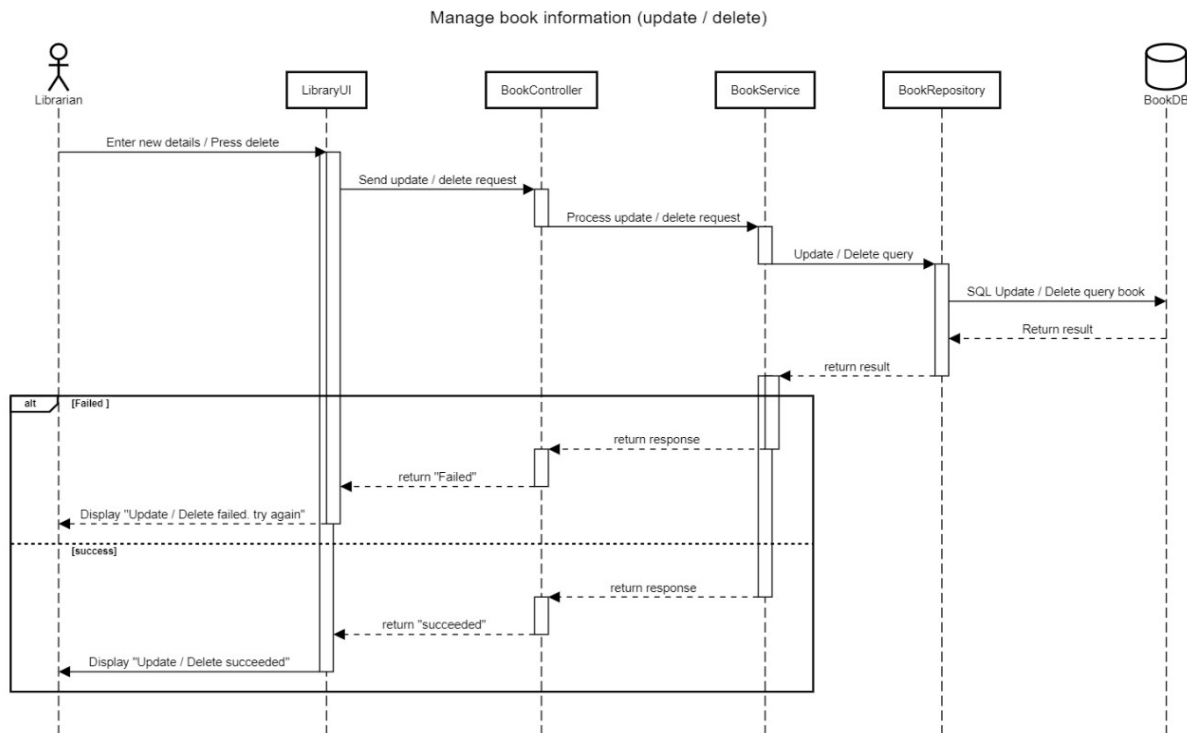
**1. Actors:** Reader – מזין פרטי התקשרות חדשים לעדכון.

**2. System components**

- a. LibraryUI – ממשק המשתמש, מקבל את הקלט ומעביר ל-Controller.
- b. UserProfileController – מתווך בין ה-UI לבין ההמשך.
- c. Repository – מבצע את השאילתה בפועל למסד הנתונים.
- f. User DB – מסד נתונים שמכיל את כל המידע בנוגע לקוראים.

**3. Process**

- a. הקורא מזין פרטי התקשרות חדשים.
- b. ה-UI שולח ל-Controller בקשת עדכון.
- c. ה-Controller מגלגל את הבקשה ל-Repository שמבצע את שאילתת ה-SQL למסד הנתונים.
- d. במידה ומסד הנתונים עדכן בהצלחה – תוחזר הודעה מתאימה לקורא.
- e. במידה ומסד הנתונים נכשל בעדכון – תוחזר הודעת כישלון לקורא.



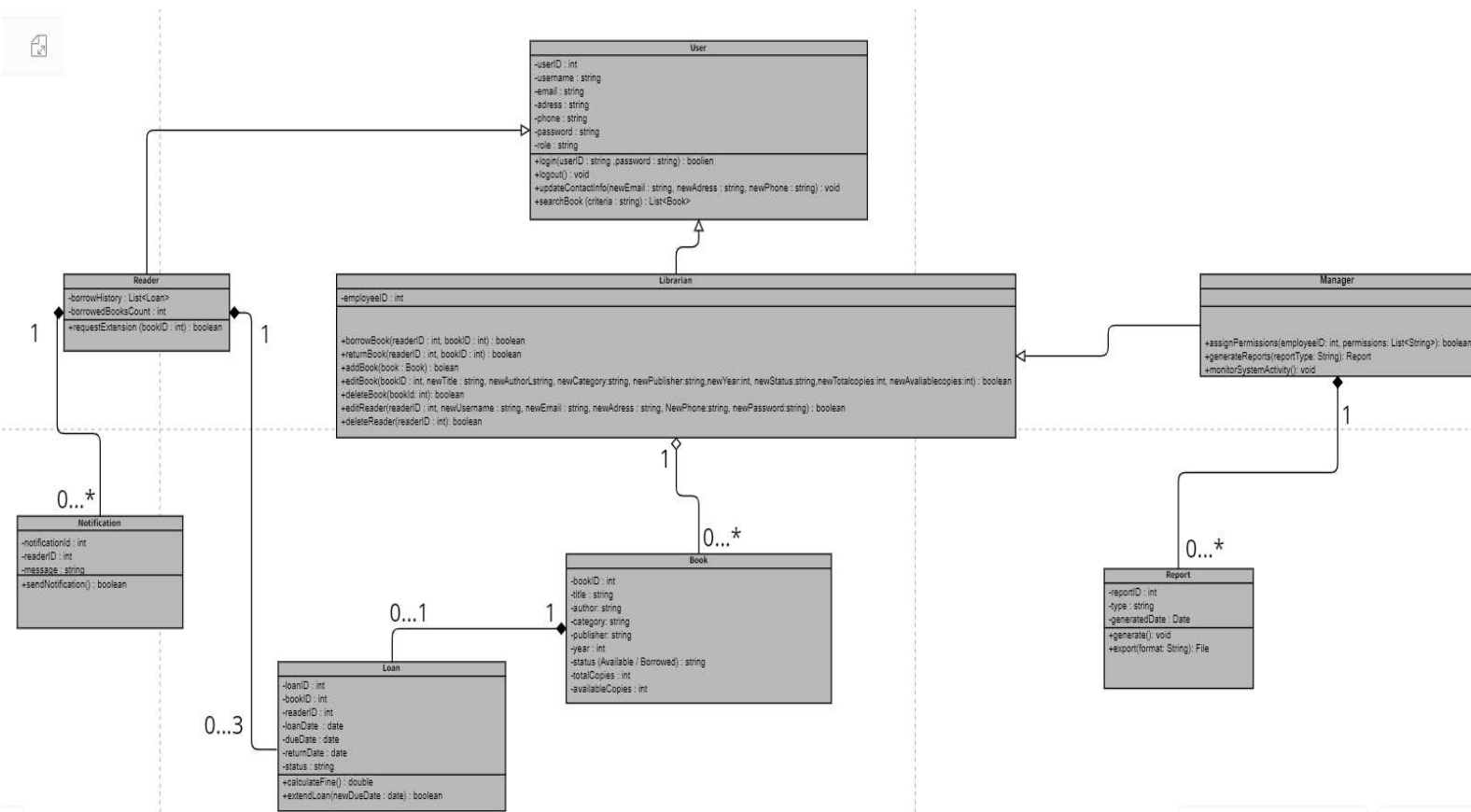
**1. Actors:** Librarian – מזין פרטים חדשים לעדכון הספר / מבקש למחוק ספר.

## 2. System components

- a. LibraryUI – ממשק המשתמש, מקבל את הקלט ומעביר ל-Controller.
- b. BookController – מתווך בין ה-UI לבין ההמשך.
- c. BookService – מטפל בלוגיקת העדכון / המחיקה.
- d. BookRepository – מבצע את השאילתה בפועל למסד הנתונים.
- g. BookDB – מסד נתונים שמכיל את כל המידע בנוגע לספרים.

## 3. Process

- a. הספרן מזין פרטי ספר חדשים / מבקש למחוק ספר.
- b. ה-UI שולח ל-Controller בקשת עדכון / מחיקה.
- c. ה-Controller מגלגל את הבקשה עד ל-Repository שמבצע את שאילתת ה-SQL למסד הנתונים.
- d. במידה ומסד הנתונים ביצע את הפעולה בהצלחה – תוחזר הודעה מתאימה לספרן.
- e. במידה ומסד הנתונים נכשל בעדכון – תוחזר הודעת כישלון לספרן.
- f. חשוב להדגיש – כישלון יכול להיווצר כי הבקשה לא הצליחה / הספר לא נמצא / הספר נמצא בהשאלה ולא ניתן למחוק אותו וכד'..



## : ARCHTITECTURAL 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 ממשק המשתמש (View).
- ארכיטקטורת MVC תאפשר הפרדה בין שני חלקי המערכת. באמצעות שכבת ה-controller נוכל לוודא ניהול נכון של בקשות המשתמשים והלוגיקה הפנימית ותתאפשר מודולריות נוחה בתחזוקת הפלטפורמה.
- Model :

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

כל מחלקה מטפלת במידע הקשור לתחום אחריותה ומנהלת את הפעולות הרלוונטיות כמו חיפוש ספרים, עדכון קוראים, הפקת דוחות ...
- View :

ה-View יציג את המידע מה-Model למשתמשים.

דוגמאות מדגמיות :

  - ה-View יכול את הדפים בהם ספרן יוכל לבצע עדכון/מחיקת ספר.
  - ה-View יכלול קבלת התראות טרם סיום זמן ההשאלה.
- Controller :

יקבל בקשות מהמשתמשים וישלח אותן ל-Model על מנת לבצע את הפעולה (לדוג' – בקשה לעדכון פרטי התקשרות).

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



