**סקירה כללית על מבנה האתר:**

* לאתר ניתן להיכנס כאורח ולצפות בעמוד הראשי, להשתמש בשורת החיפוש או לצפות בתיאור ספר מסוים, כל פעולה נוספת (העלאת ספר, בקשת השאלה) תעביר את המשתמש לדף ההזדהות/הרשמה.
* בתור לקוח רשום יפתחו בפני המשתמש האפשרות להשאיל, להעלות ספרים ולצפות בהיסטוריית הפעולות והבקשות שלו בדף הפרופיל. המשאיל יכול לקבל את פרטי המבקש אך לא להיפך, אחרי יצירת הקשר יש באפשרות המשאיל לאשר או לדחות את הבקשה, הספר יעודכן כמוזמן ולא תתאפשר השאלה שניה עד לשינוי הסטטוס שלו בחזרה.

\* ניתן להגיש בקשה לספר עם סטטוס "מושאל"(צד שואל) אך לא תתאפשר השאלה כפולה(צד משאיל)

* לבדיקה ניתן לפתוח משתמש חדש או להשתמש במשתמש שלי שרוב הספרים רשומים על שמו: [avimen@post.bgu.ac.il](mailto:avimen@post.bgu.ac.il) סיסמא: 12345

**התייחסות לסעיפים:**

1. מבנה תיקיות- בviews נמצאים עמודי הPUG בstatic נמצאים תמונות, CSS וJS של צד לקוח
2. לכל עמוד יש פונקצית GET ולחלקם יש גם שיטת POST אחת או יותר.
3. יש כל מיני שאילתות באתר:

CREATE : יש יצירה של לקוח בעמוד signUp , יצירה של ספר חדש בעמוד addBook , יצירת הזמנה חדשה ויצירת השאלה חדשה.

UPDATE : בעת יצירת השאלה חדשה אנחנו נדאג לעדכן את סטטוס הספר ל"לא זמין"

SELECT: נצפה בנתונים בעת כניסה לאתר כדי לאמת את המשתמש, בנוסף את הספרים שיוצגו בקרוסלת עמוד הבית אנחנו נמשוך את ה20 הספרים האחרונים שהועלו לאתר.

כמו כן גם שורת החיפוש תשלוף לנו תוצאות שיתאימו לשם הספר או שם הסופר.

DELETE : בגדול בקורס בסיסי נתונים למדנו שאין הרבה תרחישים בהם נרצה למחוק נתונים אז אין כאן שאילתות מחיקה, לא כי זה קשה ליישום פשוט לא מצאתי הצדקה.

\*בעת ביטול הזמנה במקום מחיקה השתמשתי בשינוי סטטוס ל"מבוטל"

4. מפורט ב3

5. לאתר יש שורת חיפוש שתחזיר תוצאות אם יש התאמה לשם ספר או סופר שקיים בבסיס הנתונים.

בנוסף האתר משתמש בsession כדי להחזיק את זהות המשתמש הרשום, ככה האתר יודע להציג את דף הפרופיל האישי ולנהל את הבקשות של המשתמש הספציפי

6. במימוש דפי הPUG יש שימוש בinclude של layout אחיד לכל הדפים באתר, בנוסף יש שימוש בלולאות each כדי להציג מערכים של אובייקטים שנשלחים מבסיס הנתונים