

[אתר](#)

[מטרה](#)

[פונקציונאליות](#)

[משימות](#)

[Definitions](#)

[מאגר ספרים](#)

[ממשק משתמש](#)

[מסכים](#)

[מסך פתיחה](#)

[מסך Register](#)

[מסך Login](#)

[מסך Reading List](#)

[מסך Completed List](#)

[מסך Details](#)

[מסך Search](#)

[נוחות משתמש](#)

[הנחיות כלליות](#)

[הנחיות ספציפיות לפרויקט](#)

[תכן](#)

[מימוש](#)

[אתגר](#)

אתר

הספריה שלי

מטרה

תרגול react hooks + routing + authentication using firebase + local storage

פונקציונליות

משימות

- כניסת משתמש : לפי מייל , סיסמה
- רישום משתמש : לפי מייל , סיסמה
- יציאת משתמש
- הצג - books list , reading list , completed list
- העבר ספר - מ books list אל reading list
- מחק ספר - מ reading list
- העבר ספר - מ reading list אל completed list
- מחק ספר - מ completed list
- הוסף הערת משתמש - עבור ספר מ reading list או completed list
- הוסף ציון - עבור ספר מ completed list
- ולידצית טופס : אין לאפשר שליחת טופס עם מידע הכרחי ריק. יש לתת הודעה מתאימה

מאגר ספרים

- ספר מוגדר על ידי :
 - מזהה יחודי - id, שם הספר , שם מחבר / ים , תמונת הספר , תיאור הספר (לפחות 500 תווים) אשר נמצאים במאגר הספרים הכללי
 - הערת משתמש , ציון משתמש (1-5) אשר יכולים להינתן על ידי המשתמשים באתר (אין צורך להוסיף זאת למאגר הספרים הכללי)
- יש להכין מאגר כללי של 30 ספרים עם המאפיינים הבאים : שם הספר , שם מחבר / מחברים , תמונת ספר , תקציר הספר עם לפחות 500 תווים
- המאגר יהיה לוקלית בקובץ json בספריה data תחת public (זה מצריך פנית (AJAX) . אופציה נוספת היא שימוש ב books api - ראה אתגר

ממשק משתמש

מסכים

מסך פתיחה

- יוצג תפריט המאפשר לעשות login \ register

מסך Register

- רישום בעזרת email,password,confirm password

מסך Login

- כניסה בעזרת email \ password

מסך Reading List

- מציג הספרים שנמצאים בשלב קריאה
- user note , name , author , description , picture
- לא להציג description מלא , להציג חלקי לדוגמא במקביל לתמונה
- כל המידע מופיע כ read only ז"א ללא יכולת עריכה
- לחיצת עכבר על ספר מעבירה למסך Details

מסך Completed List

- מציג ספרים ששלב הקריאה שלהם הסתיים
- כל ספר מופיע עם המאפיינים הבאים :
- user note, user grade , name , author , description , picture
- לא להציג description מלא , להציג חלקי לדוגמא במקביל לתמונה
- כל המידע מופיע כ read only ז"א ללא יכולת עריכה
- לחיצת עכבר על ספר מעבירה למסך Details

מסך Details

- מציג ספר על כל מאפייניו במלואם (description מלא) ומאפשר לעדכן user note , user grade

מסך Search

- מציג בעליה 10 ספרים ראשונים במאגר
- מציג את Books list
- לא להציג description מלא , להציג חלקי לדוגמא במקביל לתמונה
- מאפשר לחפש ספרים לפי מחרוזת. מחפש את המחרוזת בשם הספר או שם המחבר או תיאור - description ומציג 10 ראשונים תואמים
- כל ספר מופיע עם המאפיינים הבאים :
name , author , description , image

נוחות משתמש

- יש להציג spinner בכל גישה לשרת
- יש להציג שגיאה במידה ולא ניתן לגשת לשרת
- שמירת auth ב localStorage . לאחר השמירה המשתמש יכול לסגור את הדפדפן , לפתוח מחדש בלי שיצטרך לעשות Login
- הוספה ומחיקה יסומנו בעזרת icon
- עיצוב היא לא המשימה בעדיפות עליונה בפרויקט זה אבל לתת לזה תשומת לב אם נשאר זמן. חשוב להשקיע בעיצוב במיוחד לרגע בו תעלו את הפרויקט לדוגמא ל heroku ותתנו משם לינק ללינקדאין כי זה ימשוך הרבה תגובות !

זמנים

- הפרויקט יוצג בתאריך 4.1 ויש להתחיל לעבוד עליו מידית
- המפגשים של 10.1, 11.1 יוקדשו לעבודה על הפרויקט
- להעלות את הפרויקט לגיט שלכם לספרייה בשם my-library עד 16.1 בשעה 0900 הצגת הפרויקט בתאריך 17.1 0900-1230

הנחיות כלליות

- לחלק את הפרויקט לרשימה של משימות קטנות לתת עדיפות לכל משימה ולעבוד לפי רשימת העדיפויות. לדוגמה הלוגיקה בפרויקט זה בעדיפות יותר גבוהה בפרויקט זה מעיצוב
- להתחיל לממש את הדברים היותר קלים
- לכתוב קוד JavaScript בצורה אינקרמנטלית ז"א לכתוב משהו קטן וישר לבדוק שעובד ורק אז להוסיף עוד משהו, לבדוק וכך הלאה
- לא להתעכב על שום בעיה יותר מחצי שעה, אחרי חצי שעה להתייעץ עם מישהו ולפתור את הבעיה או לעקוף אותה
- להשתמש במצגות והתרגילים שפתרנו בכיתה כ reference

הנחיות ספציפיות לפרויקט

- מי שמסתבך עם ה openlibrary.org שיגדיר 30 ספרים במערך ויעבוד עליו. ה description חייב להיות כמה משפטים ולפחות 500 תווים
- סמנו לעצמם את ה CRUD ביחס לספרים בפעולות המוגדרות ב [Tasks](#)
- לחשוב דבר ראשון על ה state ואיך לארגן אותו ביחס לספרים במצבים השונים (כמה מצבים יש ??)

תכן

- צד קליינט ימומש בעזרת ריאקט

- שימוש ב firebase auth בתור authentication server
- לשקול שימוש ב context api לטובת נתוני auth שמתקבלים מ firebase
- לשקול להשתמש ב useFetch עבור גישת AJAX

מימוש

- רצוי לתכנן לפני שכותבים קוד :
 - Components
 - State
 - props
- בשלב הראשון אפשר לשים הכל בתוך קומפוננטה אחת . בהמשך אפשר להשתמש ב components
- מומלץ להשתמש בספריה [react-icons](https://react-icons.github.io/react-icons/) לצורך שימוש ב icons

אתגר

- שמירת העדפות משתמש ב localStorage כך ש login חדש יחזור למצב שהיה לפני logout
- שימוש ב books api. יש דוגמא שעשיתי <https://github.com/NathanKr/openlibrary-playground> אבל היא לא מציגה description וחלק מן התמונות חסר. אז או לחקור את זה או להשתמש במשהו אחר
- הצגת tooltip לכל אייקון
- להשתמש ב [dialog](#) במקום מסך Login
- להשתמש ב [dialog](#) במקום מסך Register
- עבור ספר ב Completed יש להציג תאריך התחלת קריאה ותאריך סיום קריאה
- הצג grade בעזרת 5 כוכביות שהמשתמש יכול לסמן

- שמירת auth ב localStorage תוגבל ל 20 דקות לכל היותר. בתוך 20 דקות המשתמש יכול לסגור את הדפדפן , לפתוח מחדש בלי שיצטרך לעשות Login
- לממש sorting ז"א שאפשר לסדר בסדר אלפביתי עולה או יורד את הופעת הספרים על פי לדוגמא שם הספר
- תמיכה ב route לא קיים
- לשקול ניהול מצב המשתמש עם useReducer ראה לדוגמא [כאן](#)
- מימוש רספונסיביות