

עבודת בית - תחקור

נבנה העתק של האתר) [ZipHub](#) לינק בשם.

אפשר לשחק ולהשתמש באתר, רעיון האתר הוא כזה:

ZipHub הינו מיני גרסא של GitHub, לאתר מעלים קבצי zip שלאחר מכן הוא מציג את הקבצים שלהם כמו בגיטהאב. אפשר להיכנס לתוך התיקיות של ה ZIP ולהתסכל בקבצים. את הקבצים ניתן לערוך ישירות מהאתר ולשמור. אם לקובץ יש README.md ב root של הפרוייקט, אז אנחנו מציגים אותו מפורמט (כלומר בגרסת preview) דפולטיבית לפני בחירת הקובץ.

דברים טכניים:

1. יש להשתמש ב react (ב frontend גרסת vite ולא Next.js לדוגמא)
2. יש להשתמש בקומפוננטות של shadcn ובעיצוב css בעזרת tailwindcss.
3. יש לממש מנגנון authentication לבד, לא להשתמש בסיפרייה.
a. לבחור בין JWT לבין db sessions ולהיות מוכנים להסביר מדוע נבחרה הדרך הנ"ל.
- b. יש לשים לב שבאתר דוגמא אין כרגע מנגנון כניסה אמיתי, הוא שומר פשוט ב local storage אם מישהו מחובר או לא.
4. יש להשתמש ב mongodb בתור db.
5. יש לשמור את קבצי ה zip בתוך S3 הרגיל עולה כסף, אז אפשר להשתמש בפתרון לוקלאי כמו LocalStack.

משימות:

1. להעתיק את העמוד נחיתה כמו שהוא ועמודי התחברות והרשמה.
2. יש לממש תהליך כניסה כמו שמתואר בדברים טכניים.
3. לאחר שיוזר מחובר הוא מגיע לעמוד dashboard בו הוא יכול להוסיף repositories חדשים.
4. בעת הוספת repository יש להביא שם, תיאור (אופציונאלי), ולעלות קובץ zip.
a. יש לשים לב כי להרשאות לעלות רק קבצי zip.
b. יש לא לאשר יצירת repo אם אין שם תקין וקובץ תקין.
5. לאחר מכן לכל repository יהיה קישור ייחודי /{repo_id}/ repository
6. אם יש קובץ Readme.md ב root של הפרוייקט יש להציגו לבד אוטומטית.

7. יש אפשרות לעבור בין התיקיות והקבצים של הפרוייקט ולראות אותם.
8. אפשר לערוך את הקבצים ישירות באתר ולשמור אותם.
9. יש לדאוג למסמך ארכיטקטורה שמסביר איך הלוגיקות הראשיות באתר מומשו, לדוגמא:
- a. הצגת ושמירת קבצים.
 - b. עריכת קבצים.
10. לבסוף יש לשמור את הפרוייקט ב github ולשלוח קישור לפרוייקט לצורך בקרה של הקוד ולהתייחסותם ברעיון הבא.