**פרוייקט NODEJS MONGO**

יצירת מערכת .צעצועים

בקצרה על המערכת:

במערכת כל אדם (רשום) יוכל להוסיף צעצוע עם שם , תיאור , קטגוריה , תמונה ,מחיר   
בנוסף יהיה ניתן לשלוף מידע שונה על צעצועים שנמצאים במסד נתונים

**חלק 1**

יש לייצר קולקשיין שנראה כך ב DB חדש עדיף בשם TOYS:

|  |  |
| --- | --- |
|  | \_ID |
| שם המוצר | name |
| תיאור המוצר | info |
| קטגוריית המוצר | category |
| תמונת המוצר ב URL (ניתן להביא מאתר פיקסילס) | img\_url |
| מחיר | price |
| ,תאריך שנוסף בו המוצר (לא חובה) | date\_created |
| ישמש בחלק 2 של המשימה לקישור בין משתמש למוצר (לפי TOKEN) | user\_id |

תפקידכם במשימה זו לייצר את הצד שרת בלבד ולא צד לקוח ובעצם לספק API לצד לקוח שבו המתכנת צד לקוח יוכל לבצע את הפעולות הרשומות מעלה,

להלן הכתובת הרצויות לביצוע פעולות ב API:

|  |  |
| --- | --- |
| בקשה כללית לצעצועים במערכת | domain/toys ?s -> חיפושים  החיפוש יתבצע במאפיין  Name או info ?page -> מספר עמוד לפי 10 צעצועים בדף |
| בקשת צעצוע לפי קטגוריה | domain/toys/cat/:catid ?page -> מספר עמוד לפי 10 צעצועים בדף |
| \* מחייב בחלק 2 שהפעולה תתבצע עם TOKEN |  |
| \*הוספת צעצוע- POST | domain/toys/ |
| \*עריכת צעצוע- PUT | domain/toys/:editId |
| \*מחיקת צעצוע- DELETE | domain/toys/:delId |
| בונוס - יכולת לקבל מוצרים לפי טווח מחירים מנימום ומקסימום | domain/toys/prices  ?page -> מספר עמוד לפי 10 מוצרים בדף  ?min - מחיר מנימום של מוצר  ?max - מחיר מקסימלי של מוצר דוגמא להצגת מוצרים שהמחיר שלהם בין 10 ל40 localhost:3000/toys/prices/?min=10&max=40 |

**חלק 2 המשך שכולל את** כל חוקי האבטחה של AUTH (עם טוקן ו BCRYPT)

* צור קולקשן של USERS ואת שבנוי בצורה הבאה:  
  Name,Email,Password,date\_created,role

\*המאפיין EMAIL צריך להיות UNIQUE

* צור ROUTE ב USERS של POST ל "כתובת <http://localhost:3000/users> שייצר משתמש חדש לפי הסכמה של הקולקשן (לשים לב שה PASSWORD מוצפן עם BCRYPT ושמחזירים את המידע לצד לקוח לשים לב לכל חוקי האבטחה שלמדנו)
* בנוסף לכל יהיה כתובת שתאפשר למשתמש להתחבר בבקשת POST : <http://localhost:3000/users>/login

שבו ישלח ב BODY רק המאפיניים EMAIL ו PASSWORD

במידה והאימייל והסיסמאות תקניות (שים לב לשמור את ה PASSWORD מוצפן עם מודול BCRYPT וגם לבדוק אותו מוצפן מול ה BCRYPT) המשתמש יקבל בחזרה TOKEN ,

יש להוסיף לסכמה של הקולקשן TOYS מאפיין/קיי בשם USER\_ID (שבעצם יציג לאיזה משתמש הצעצוע שייך)

* בראוט של ה TOYS כל בקשות ה POST , DELETE ו PUT יחייבו שהמשתמש ישגר TOKEN תקני ב HEADER עם קיי בשם "auth-token" ובמידה והטוקן לא תקין המשתמש יקבל הודעת שגיאה ב JSON שמתאימה לבעיה (יש לבצע את הבדיקה עם פונקציית MIDDLEWARE כמו שלמדנו).

\* יש לדאוג בפרוייקט להשתמש ב JOI , BCRYPT במקומות הנכונים , וכמובן MIDDLEWARE של AUTH לכתובות שיש צורך בהם ב TOKEN

מומלץ שבמערכת יהיו קיימים כ 12 צעצועים עם 3 קטגוריות שונות.

יש לספק דוקומנטיציה שמסבירה כיצד להשתמש ב API אפשר להסתכל בכתובת הבאה לכתיבת דוקימנציה נכונה (במיוחד כיצד לגשת לכתובות השונות )

<https://jsonplaceholder.typicode.com/>

ובנוסף לעלות את הפרוייקט לשרת HEROKU ככה שהפרוייקט יעבוד לפי החוקיות למעלה , וכמובן לעלות ל GITHUB , יש להגיש את 2 הכתובות של הגיט והירקו למייל  
ofer.shelli@gmail.com