

# פרויקט מסכם מודול Node.js

מסמך הנחיות ודגשים לפרויקט

## מפרט אפיון הפרויקט

### מבוא ומטרות

בפרויקט זה נתנסה בניית **rest API** בצד שרת. ידע ויכולת בניית **rest API** נחשב למשימה מבוקשת ביותר בעולם הפיתוח בהייטק ולכן יש ליחס חשיבות גדולה מאוד לפרויקט זה ומומלץ להשקיע בקריאה ובהבנה של הקוד, המודולים, הקבצים, והתיקיות המרכיבים פרויקט זה!

### תיאור כללי:

פיתוח צד שרת לאפליקציה אינטרנטית הכוללת מערכת ניהול אתר המאפשרת למשתמשים עסקיים לפרסם תוכן לערוך ולמחוק אותו.

### דגשים מרכזיים:

- יש להשתמש במאגר המידע **mongoDB** בפרויקט זה
- יש להשתמש **express.js**
- יש להשתמש בספריות **bcryptjs joi jsonwebtoken config morgan cors chalk mongoose** לצורך בניית פרויקט זה.

### אופן הגשת הפרויקט:

את הפרויקט יש להעלות ל - git ללא תיקיות ה - `node_modules`, ולשלוח לינק אליו במשימה שתפתח בקמפוס. על הפרויקט לכלול קובץ `ReadMe.md` שמסביר על הפרויקט, תכולתו, הפונקציונאליות ודרך ההתממשקות עמו.

### דרישות כלליות

- יש לשמור על קוד נקי ומסודר: לנקות `console.log` וקטעי קוד שהפכו להערות.
- יש לבנות קוד שמספר סיפור ולתת לפונקציות ולמשתנים שמות משמעותיים.
- יש לחלק את הפרויקט למודולים לפי נושאים: לשים לב שלא יופיע בו קובץ ובו למעלה מ 200 שורות קוד
- יש להקפיד על המוסכמות לכתיבת קוד.

חשוב לזכור שפרויקט זה יהיה חלק מתיק העבודות שלכם וייצג אותך בכבוד מול מעסיקים פוטנציאליים ולכן יש לשמור על אסתטיקה של קוד ועיצוב

## דרישות טכנולוגיות צד שרת

### 1. Users end points צור יירוטים לכתובות הבאות:

No.	URL	method	Authorization	action	notice	return
1.	/users	POST	all	register user	Unique email	נספח 1
2.	/users/login	POST	all	login		Encrypt token
3.	/users	GET	admin	Get all users		Array of users
4.	/users/:id	GET	The registered user or admin	Get user		User
5.	/users/:id	PUT	The registered user	Edit user		user
6.	/users/:id	PATCH	The registered user	Change isBusiness status		user
7.	/users/:id	DELETE	The registered user or admin	Delete user		Delete d user

### 2. Cards end points צור יירוטים לכתובות הבאות:

No.	URL	method	Authorization	action	return
1.	/cards	GET	all	All cards	נספח 2
2.	/cards/my-cards	GET	The registered user	Get user cards	Array of cards
3.	/cards/:id	GET	all	Get card	card
4.	/cards	POST	Business user	Create new card	card
5.	/cards/:id	PUT	The user who created the card	Edit card	card
6.	/cards/:id	PATCH	A registered user	Like card	card
7.	/cards/:id	DELETE	The user who created the card or admin	Delete card	Deleted card

- 
3. **Logger** יש להשתמש בספריית morgan כדי ליצור logger לאפליקציה. ה- logger ידפיס בקונסול את כל הבקשות שנשלחות אל השרת ויפרט על כל בקשה באיזה שעה היא נשלחה, שיטת שליחת הבקשה, לאיזה כתוב URL היא נשלחה, סטאטוס התשובה, וכמה זמן לקח למתן תשובה.
  4. **Chalk** יש להשתמש בספרייה זאת על מנת לצבוע את ההדפסות בקונסול בצבע שיתאים לתוכן ההדפסה
  5. **Cors** יש להשתמש בספרייה זאת על מנת לאפשר שליחת בקשות HTTP מכתובות מורשות בלבד
  6. **JSON** יש לאפשר קבלת json בגוף בקשות HTTP
  7. **Public** יש לנתב בקשות שלא יורטו על ידי האפליקציה לתיקיית public בשרת ורק אם הקובץ לא נמצא גם בה לשלוח הודעת שגיאה 404 לגולש עם הכיתוב המתאים
  8. **Environments** יש לאפשר שני סביבות עבודה כאשר הראשונה היא לוקלית והשנייה על הענן של atlas
  9. **בקשות HTTP** יש להחזיר תשובה לגולש עם הנתונים שהוא ביקש בהתאם להרשאותיו. במידה שאין למשתמש הרשאות או שנשלחה מצד לקוח בקשה עם נתונים חסרים או לא תקינים יש להחזיר תשובה לגולש עם הסטאטוס המתאים והודעת השגיאה המתאימה.
  10. **Joi** יש להשתמש בספרייה זאת על מנת לבצע וולידציה על אובייקטים שמתקבלים מצד לקוח ובמידה והאובייקט לא עובר את הוולידציה יש לשלוח תשובה עם סטאטוס מתאים והודעת שגיאה מתאימה.
  11. **Bcryptjs** יש להצפין סיסמה של משתמש חדש לפני שמירתה במאגר המידע של mongoDB עם ספרייה זאת ובתהליך ההתחברות לוודא שהסיסמה שנשלחה מצד לקוח תואמת את הסיסמה השמורה במאגר המידע.
  12. **Json web token** יש להשתמש בספרייה זאת על מנת ליצור token מוצפן שיכיל בתוכו מידע על המשתמש שמנסה להתחבר. באובייקט של ה- payload צריכים להיות המפתחות הבאים: `_id, isBusiness, isAdmin`
  13. **Initial data** יש ליצור שלושה מופעים של כרטיסי ביקור ושלושה מופעים של משתמשים כאשר הראשון הוא משתמש רגיל השני עסקי והשלישי אדמיין.
  14. **Mongoose** השתמש בספרייה זאת על מנת ליצור models של Cards Users. הוסף את המפתחות המתאימים בצד שרת על מנת שהאובייקט שיישמר במאגר המידע יהיה כפי שמופיע בנספחים 1,2
-

## בונוסים

1. **bizNumber** יש לאפשר למשתמש מסוג אדמיין לשנות מספר של עסק בתנאי שהמספר החדש לא תפוס על ידי עסק אחר באפליקציה.
2. **File logger** צור file logger שעל כל בקשה עם סטאטוס קוד 400 ומעלה תיצור בתיקיית logs קובץ שהשם שלו יהיה התאריך של אותו היום (אלא אם כן הקובץ הזה כבר קיים) ובתוכו יירשמו: תאריך הבקשה, סטאטוס הקוד, והודעת השגיאה
3. **Google API** אפשר הרשמה והתחברות למשתמשים שאינם אדמיין ואינם עסקיים באמצעות google API
4. **חסימת משתמש** חסום משתמש שניסה להתחבר שלוש פעמים רצוף באמצעות אותו המייל אבל עם סיסמה לא נכונה ל - 24 שעות

## נספחים

1. **יצירת משתמש חדש**  
האובייקט שצריך להתקבל מצד לקוח לצד שרת

```
{
  "name": { "first": "first", "middle": "", "last": "user" },
  "isBusiness": true,
  "phone": "050-0000000",
  "email": "first@gmail.com",
  "password": "Aa1234!",
  "address": {
    "state": "",
    "country": "israel",
    "city": "tel-aviv",
    "street": "magnive",
    "houseNumber": "5"
  },
  "image": { "url": "", "alt": "" }
```

## האובייקט שצריך להישמר במאגר המידע

```
_id: ObjectId('6361a80583262d33d9d4294f')
name: Object
  first: "first"
  middle: ""
  last: "user"
  _id: ObjectId('6361a80583262d33d9d42950')
phone: "050-0000000"
email: "first@gmail.com"
password: "$2a$10$.Do1N9TnpJ4Qj6m7U48TD.Y7qULHi7OOD6w00BbapQPvagTvNQOIe"
image: Object
  url: "https://cdn.pixabay.com/photo/2016/04/01/10/11/avatar-1299805_960_720..."
  alt: "business card image"
  _id: ObjectId('6361a80583262d33d9d42951')
address: Object
  state: "not defined"
  country: "israel"
  city: "tel-aviv"
  street: "magnive"
  houseNumber: 5
  zip: 0
  _id: ObjectId('6361a80583262d33d9d42952')
isAdmin: false
isBusiness: true
createdAt: 2022-11-01T23:13:09.180+00:00
__v: 0
```

## 2. יצירת כרטיס ביקור חדש

האובייקט שצריך להתקבל מצד לקוח לצד שרת

```
1 {
2   ... "title": "test123",
3   ... "subtitle": "testing",
4   ... "description": "testing 123",
5   ... "phone": "050-0000000",
6   ... "email": "testing@gmail.com",
7   ... "web": "https://www.test.co.il",
8   ... "image": {
9     ... "url": "",
10    ... "alt": ""
11  },
12  ... "address": {
13    ... "state": "",
14    ... "country": "test",
15    ... "city": "test",
16    ... "street": "test",
17    ... "houseNumber": 3,
18    ... "zip": "0"
19  }
20 }
```

## האובייקט שצריך להישמר במאגר המידע

```
_id: ObjectId('636183b6cb0710446e7a7326')
title: "test123"
subtitle: "testing"
description: "testing 123"
phone: "050-0000000"
email: "testing@gmail.com"
web: "https://www.test.co.il"
✓ image: Object
  url: "https://cdn.pixabay.com/photo/2016/04/20/08/21/entrepreneur-1340649_96..."
  alt: "business card image"
  _id: ObjectId('636183b6cb0710446e7a7327')
✓ address: Object
  state: "not defined"
  country: "test"
  city: "test"
  street: "test"
  houseNumber: 3
  zip: 0
  _id: ObjectId('636183b6cb0710446e7a7328')
bizNumber: 6401563
✓ likes: Array
  user_id: ObjectId('636121fbdf5896f21017cba5')
  createdAt: 2022-11-01T20:38:14.216+00:00
  __v: 0
```