חוזר מנהל מה"ט 11-4-51 – נספח מס' 1 (הצעה לפרויקט גמר(

תאריך: \_\_\_\_\_\_\_\_\_\_

### לכבוד

### יחידת הפרויקטים

מה"ט

### **הצעה לפרויקט גמר**

\* יש להדפיס את כל הנתונים הנדרשים

1. **פרטי הסטודנטים**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם הסטודנט | ת.ז. 9 ספרות | כתובת | טלפון נייד | תאריך סיום הלימודים |
| אלעד אור | 318689627 | לוי אשכול 63 ד | 0523447227 |  |
| מישל פלאם | 337791610 | תל חי 50 רמת גן | 0584006678 |  |

שם המכללה שנקר הנדסאים סמל המכללה: 72424

מסלול ההכשרה: הנדסאים מגמת לימוד: תוכנה

מקום ביצוע הפרויקט: שנקר הנדסאים חיים לבנון 30 ת״א

1. **פרטי המנחה האישי**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| שם המנחה \* | כתובת | טלפון נייד | תואר | מקום עבודה/תפקיד |
| שי אברהם | כפר יונה | 054-5726662 | MBA | מרצה |

A picture containing text

Description automatically generated

\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
חתימת הסטודנט חתימת המנחה האישי חתימת הגורם המקצועי מטעם מה"ט

**דרך מנחם בגין 86 תל אביב ת.ד. 36049 מיקוד 67138**

**טלפון : 03-7347521 פקס: 03-7347644**

1. **שם הפרוייקט**
2. **רקע**
   1. **תיאור ורקע כללי**
   2. **מטרות המערכת**
3. **סקירת מצב קיים בשוק, אילו בעיות קיימות**
4. **מה הפרוייקט אמור לחדש או לשפר**
5. **דרישות מערכת ופו+נקציונליות**

**5.1. דרישות מערכת,**

**סביבת הטמעה ושימוש. שרידות, ביצועים/התמודדות עם עומסים.**

**5.2. דרישות פונקציונליות**

**רשימת דרישות המשתמש מהמערכת, מהן הפעולות בהן נדרשת המערכת לתמוך.**

1. **בעיות צפויות במהלך הפיתוח ופתרונות (תפעוליות, טכנולוגיות, עומס ועוד)**

**6.1. תיאור הבעיות – כפועל יוצא של דרישות המשתמש מהתוכנה.**

**6.2. פתרונות אפשריים**

1. **פתרון טכנולוגי נבחר:**

**7.1 טופולוגית הפתרון כלומר: - פרישת המערכת , היכן יתבצע יישום המערכת ) deployment ,) מרכיבי הפרישה. הנ"ל ברמת מערכת )לדוג, פרויקט פיתוח אתר אינטרנט: המערכת מורכבת משרת, ממשק משתמש בצד הלקוח,, DB’s , טווח תקשורתי אינטרנט, המערכת -תיושם ברשת האינטרנט , יש להציג את דיאגרמת המערכת וכו'(**

**7.2 טכנולוגיות בשימוש .)איזה ומדוע בכמה מילים(**

**7.3 שפות הפיתוח )איזה שפות ומדוע בכמה מילים?(**

**7.4 תיאור הארכיטקטורה הנבחרת הסבר בכמה מילים מדוע -**

**7.5 חלוקה לתכניות ומודולים**

**7.6 סביבת השרת )מקומי, וירטואלי, ענן, שירות אירוח(**

**7.7ממשק המשתמש/לקוח - GUI**

**7.8 ממשקים למערכות אחרות/ :API**

**7.9 שימוש בחבילות תוכנה:**

1. **שימוש במבני נתונים וארגון קבצים**

**8.1 נא פרט את מבני הנתונים.**

**8.2 נא פרט את שיטת האיחסון )מאגר, קבצים ובאיזה טכנולוגיה(**

**8.3 נא ציין מנגנוני התאוששות מנפילה\קריסה\תמיכה בטראנזקציות.**

1. **תרשימי מערכת מרכזיים**

**9.1 . Use Case**

**9.2 - . Sequence diagram רצף קריאות פונקציות מרכזיות בלוגיקה העיסקית המרכזית של הפרוייקט**

**9.3 Data flow**

**10. תיאור המרכיב האלגוריתמי – חישובי**

**10.1 איזה בעיה בא לפתור, איך יפתור?**

**10.2 איסוף מידע וניתוחים סטטיסטיים**

**11. תיאור/ התייחסות לנושאי אבטחת מידע**

**נא לציין אזורים הדורשים אבטחה כגון: שרת, בקרת גישה לאתר, חשבונות משתמשים.**

**נר לציין מס מקרים ותגובות להן ניתן מענה אבטחתי.**

**12. משאבים הנדרשים לפרוייקט:**

**12.1 מספר שעות המוקדש לפרוייקט, חלוקת עבודה בין חברי הצוות.**

**12.2 ציוד נדרש**

**12.3 תוכנות נדרשות**

**12.4 ידע חדש שנדרש ללמוד לצורך ביצוע הפרוייקט**

**12.5 ספרות ומקורות מידע**

**13. תכנית עבודה ושלבים למימוש הפרוייקט**

**14. תכנון הבדיקות שיבוצעו**

**14.1 נא לפרט בטבלה, בדיקות תהליכיות ברמת המשתמש בהן נדרשת המערכת לעמוד (full flow)**

**14.2 נא לפרט בטבלה, מס מייצג של בדיקות יחידה למודולים המרכזיים בהן נדרשת המערכת לעמוד. (unit test)**

**15. בקרת גרסאות (control version)**

**חתימת הסטודנט חתימת המנחה**

1. **הערות ראש המגמה במכללה**
2. **אישור ראש המגמה**

**שם: חתימה: תאריך:**

1. **הערות הגורם המקצועי מטעם מה"ט**
2. **אישור הגורם המקצועי מטעם מה"ט**

**שם: חתימה: תאריך:**

1. שם הפרויקט

שם הפרויקט הוא ''"Makolot.org  
  
**2. רקע**הפרויקט קיבל תנופה משיתוף פעולה שנרקם בין מכללת שנקר הנדסאים לבין בעלי עסקים שנפגעו בתקופת הקורונה או מעוניינים בעזרה על מנת למנף ולמתג את עצמם.  
הצוותים שנמצאו מתאימים להשתתף בפרויקט קיבלו עסק לעבוד איתו לפי שיקולים של ראש המגמה והמרצים הרלוונטיים לפרויקט.  
  
אנחנו נבחרנו לעבוד עם איש בשם אלי סטוי, הוא יועץ בכיר ברמי לוי ובעל ניסיון עשיר ורב שנים בתחום הקמעונאות ובקמעונאות הישראלית בפרט.  
  
מדובר בפרויקט מטעם עמותת ''פורום מזון ומכולת לביטחון תזונתי בישראל'' של אלי סטוי.  
המטרה שלנו היא ליצור פלטפורמה נוחה ונגישה שתאפשר לקמעונאים הקטנים כגון מכולות שכונתיות את האפשרות להציע שירות משלוחים נוח שרוב הניהול שלו, גם בהיבט הכספי, יורד מהכתפיים של בעלי המכולות ומבוצע בעזרת הפלטפורמה של העמותה.  
  
מדובר בחנות דיגיטלית שתהיה זמינה גם בצורת אפליקציה, דרכה יוכלו להזמין מוצרי מכולת כמו ברשתות הגדולות אך לפי עקרון פארטו, משמע החנות תחזיק מלאי קבוע של 20% מהמוצרים אך הם יענו על 80% מהצרכים השכיחים.  
  
כאשר תתבצע הזמנה, החנות תעבירה לביצוע המכולת השכונתית שהכי קרובה ללקוח באופן גאוגרפי או לחילופין מכולת שנמצאת ברדיוס הזמין ללקוח ונבחרה ספציפית על ידו.  
  
  
  
  
  
  
  
  
  
  
  
  
  
3.סקירת המצב הקיים בשוק  
  
אתר העמותה (נכון לכרגע כמובן אין חנות וגם העיצוב נתון לשינוי)

https://www.makolot.org.il/

ישנן מספר פלטפורמות למשלוחים של מוצרי סופר/מכולת אונליין, מהבולטות שבהן:

https://quik.co.il

https://www.ampm.co.il/

https://www.shufersal.co.il/

כולן מציעות משלוחים בזמנים בסה''כ אטרקטיביים וממשק נוח למדיי, אך ההבדל הניכר הוא הגופים שהן עובדות איתם, המשלוחים מבוצעים בעזרת אספקה של חברות הענק בישראל ואין כאן תמיכה או עירוב של המכולות והמרכולים השכונתיים ואנשים פרטיים.  
  
במהלך שנת הקורונה, כמות הקניות אונליין בישראל זינקה פי 3 בקירוב ולכן מדובר בשוק שחשוב ומוטב להשתלב בו כמה שיותר מוקדם כדי לתפוס את המומנטום, אמנם רוב הקניות בסופרים עדיין מתבצעות פיזית, ישנה מגמה גוברת של שינוי.  
  
4. מה הפרויקט יחדש או ישפר  
  
השירות יהיה פשוט ומהיר מבחינת ממשק וזמני משלוח ובעצם ייווצר מצב בו המשתמש יוכל לבחור גם לקבל מוצר מהר שיישלח אליו מהמכולת הקרובה אליו וגם יהיה לכך ערך מוסף של תמיכה בבעלי עסקים קטנים וחיזוק מעמדם במקום גופי ענק כמו שופרסל וגם קבלת שירות מהיר ואיכותי ללא פשרות.  
  
5. דרישות מערכת:

**סביבת הטמעה ושימוש, שרידות, ביצועים/התמודדות עם עומסים.**הפרויקט מבוסס על צד שרת שנכתבת בשפת SQL, משתמשים ב-Express ו- Node JS על מנת לכתוב Routes ולבצע בעזרתם את החיבור לסביבת צד לקוח באמצעות API, הפקודות נבדקות ב-ThunderClient שזה כלי לשליחת ושמירת פקודות API בצורת JSON, דומה למקביליו כגון Postman.

נקבל מאגר תמונות חיצוני של כל המוצרים ונשתמש במאגר כתובות חיצוני ונחבר אותם למסד הנתונים שלנו.

סביבת צד הלקוח בנויה ב-React ו-CSS, ייתכן בהמשך מעבר ל-SCSS.

יהיו במערכת:  
עדכון אוטומטי של מסד הנתונים (או ידני במידת הצורך).  
עדכון המערכת לאחר שינויים.  
שימוש בשירותי מיקום על מנת לזהות את מיקום המשתמש בהשוואה למכולות הקרובות אליו.  
שימוש במערכת תשלומים על מנת שיוכלו לבצע קניות באתר.  
5.2 דרישות פונקציונאליות:  
**רשימת דרישות המשתמש מהמערכת, מהן הפעולות בהן נדרשת המערכת לתמוך.**

* **הרשמה לאתר**
* **התחברות (כניסה עם שם משתמש וסיסמה קיימים)**
* **עדכון פרטים אישיים**
* **הוספת פריטים לסל הקניות**
* **ביצוע הזמנה של הפריטים בסל הקניות ממכולת לבחירת המשתמש בשעה הרצויה (כל עוד היא במרחק מתאים).**
* **חיפוש מוצרים**
* **בדיקת היסטוריית הזמנות**
* **הרשמה לחדשות**
* **הצגת פריטים לפי קטגוריה**
* **פונקציות מיוחדות למשתמש אדמין כמו ניהול משתמשים אחרים ואישור הצטרפות של עסקים חדשים לשירות.**
* **הוספת/ביטול הנחות, עדכון ומחיקת פריטים מטבלאות.  
    
  6. בעיות צפויות במהלך הפיתוח ופתרונות (תפעוליות, טכנולוגיות, עומס ועוד):**6.1 תיאור הבעיות הללו כפועל יוצא של דרישות המשתמש מהתוכנה:
* **שכחתי שם משתמש/סיסמה**
* **מייל לא תקין או קיים כבר (קיים כבר - בהרשמה)**
* **שם משתמש לא תקין**
* **סיסמה שגויה**
* **המשתמש לא נותן גישה למיקום (במידה ולא רשום עם הכתובת שלו)**
* **המשתמש בוחר מכולת רחוקה מדי לביצוע המשלוח**
* **המשתמש מבקש להשתמש בשירות בשעות בהן הוא סגור.**

# פתרונות אפשריים. (נא ציין פתרונות אפשריים וחלופות ארכיטקטוניות)

* **שחזור נתונים באמצעות הזדהות למייל שרשום מראש**
* **הזנת סיסמה פעמיים בעת הרשמה למניעת טעויות**
* **לבקש את כתיבת כתובת האימייל פעמיים ולשלוח מייל לבדיקה שהמשתמש יאשר את המייל ורק אז להכניס למאגר נתונים או שליחת סמס לטלפון (אישור מייל בעת הרשמה)**
* **הצגת הודעה מתאימה למשתמש שאומרת במפורש איזה סוג שם משתמש ניתן להכניס (קופסת הערה בצד)**
* **שליחת מייל/הודעה מתאימה עם הסיסמא (ואם הוחלפה – שליחת מייל עם סיסמא עדכנית)**
* **הקפצת התראה שמבקשת להפעיל שירותי מיקום למען שירות אופטימלי.**
* **הצעת מכולת חלופית אם המשתמש בחר אחת לא תקינה.**
* **הקפצת הודעה ראשית ובולטת למסך שהשירות סגור כעת וציון השעות בהן הוא פעיל.**

# 7. פתרון טכנולוגי נבחר:

# 7.1טופולוגית הפתרון- כלומר: פרישת המערכת , היכן יתבצע יישום המערכת (deployment), מרכיבי הפרישה. הנ"ל ברמת מערכת (לדוג' פרויקט פיתוח אתר אינטרנט: המערכת מורכבת משרת, ממשק משתמש בצד הלקוח,, DB’s, טווח תקשורתי-אינטרנט, המערכת תיושם ברשת האינטרנט , יש להציג את דיאגרמת המערכת וכו')

פרישת המערכת:

מסד נתונים שנכתב בSQL ויישמר על שרת ענן ושרת ששומר על האתר באוויר.

REST API ב-Node.JS לקישור השרת לצד לקוח.

צד לקוח שכתוב ב-React ו-CSS.  
**המערכת תיושם ברשת האינטרנט על אתר שיכול לרוץ על כל מכשיר עם דפדפן וגישה לאינטרנט.**7.2 טכנולוגיות בשימוש:

7.3 שפות הפיתוח:

SQL – צד שרת  
Node.JS - API

React – צד לקוח  
CSS – עיצוב  
  
7.4 תיאור הארכיטקטורה הנבחרת – הסבר בכמה מילים מדוע

SQL שפה שלמדנו בלימודים עבור צד שרת, פשוט למדיי לבנות בה procs ולקשר אותם ל-API ברגע שמבינים את שיטת העבודה אז בחרנו בה עבור הפרויקט.

Node.JS שפה נוחה מאוד עבור בניית REST API ובאופן כללי טוב ללמוד אותה עבור פרויקטים עתידיים וגם למדנו לבנות בעזרתה API במהלך הלימודים אז לכן בחרנו אותה.  
React שפה נוחה ורלוונטית מאוד כיום לבניית צד לקוח, בנוסף למדנו לעבוד בה בלימודים ולכן בחרנו בה לצד לקוח.

7.5 חלוקה לתוכניות ומודולים  
שרת שמורכב מ-Data ו-API

7.6 סביבת השרת  
השרת יהיה שרת וירטואלי שמאוחסן על הענן.

7.7 ממשק המשתמש/לקוח - GUI   
  
7.8 ממשקים למערכות אחרות / API

משתמשים ב-RESTFUL API והפקודות backend שלו נשמרות נבדקות ומבוצעות בעזרת ThunderClient בשלב הפיתוח.

7.9 שימוש בחבילות תוכנה:  
SQL ב- Azure Date Studio עבור בניית מסד הנתונים הרלוונטי.  
Visual Studio Code + extentions בשביל שאר הפרויקט, שם כותבים בReact, Node.JS, CSS וכל השאר.

8. שימוש במבני נתונים וארגון קבצים  
8.1 נא פרט את מבני הנתונים.

טבלת כתובות:

* ID כתובת
* עיר
* רחוב
* מידע נוסף (כניסה, דירה וכו')
* מיקוד
* כתובת פעילה (זמינה כאופציה במערכת)

טבלת כרטיסי אשראי: (נוצרה אך הצורך בקיומה תלוי בפתרון התשלום שבעל העסק יבחר)

* ID כרטיס
* מספר כרטיס
* תאריך כרטיס
* CVV
* שם בעל הכרטיס

טבלת לקוחות:

* ID לקוח
* שם פרטי
* שם משפחה
* אימייל
* מס' פלאפון
* תאריך לידה
* סיסמה
* עיר
* ID אשראי

טבלת קישור בין כתובות ללקוחות:

* ID לקוח
* ID כתובת

שניהם מפתח ראשי מוצמד

טבלת מנהלי חנות:

* ID מנהל
* שם פרטי
* שם משפחה
* אימייל
* מס' פלאפון
* תאריך לידה
* סיסמה
* עיר
* ID אשראי

טבלת קטגוריות:

* ID קטגוריה
* שם קטגוריה
* אינפורמציה/הסבר על קטגוריה
* תמונה
* קטגוריה פעילה (כן או לא)

תת-קטגוריה:

* ID תת קטגוריה
* ID קטגוריה
* שם תת קטגוריה
* אינפורמציה/הסבר על תת קטגוריה
* תמונה
* תת קטגוריה פעילה (כן או לא)

מוצרים:

* ID מוצר
* ID קטגוריה
* ID תת קטגוריה
* שם מוצר
* מחיר ברירת מחדל
* מחיר סופי
* מידע בסיסי (לדוג' ערכים תזונתיים)
* הסבר – לא חובה
* תמונה
* ספקים
* מוצר פעיל (כן או לא)

חנויות:

* ID חנות
* שם חנות
* ID ספק
* עיר
* ID כתובת
* זמני פתיחה
* רדיוס (מרחק שהם מוכנים לשלוח אליו)
* מספר טלפון
* שם איש קשר
* חנות פעילה (כן או לא)

הזמנות:

* ID הזמנה
* ID סטטוס הזמנה
* הנחה בהזמנה (כמה כסף נחסך)
* סכום תשלום על ההזמנה
* פרטי ההזמנה
* תאריך ההזמנה
* ID לקוח
* זמן הגעה מועדף
* ID חנות שולחת

טרנזקציות:

* ID טרנזקציה
* ID לקוח
* סכום תשלום
* תאריך תשלום
* ID הזמנה
* סטטוס תשלום
* ID כרטיס אשראי  
    
  ---- טבלת פרטי הזמנה – האם צריך את זה?

**8.2 נא פרט את שיטת האחסון (מאגר, קבצים ובאיזה טכנולוגיה)**  
מסד נתונים SQL בסביבת בענן.

**8.3 נא ציין מנגנוני התאוששות מנפילה/קריסה/תמיכה בטראנזקציות**

בעל העסק ישתמש בפלטפורמה חיצונית לתמיכה בתשלומים וטראנזקציות לבחירתו (לדוגמא PayPal)

1. **תרשימי מערכת מרכזיים**

# תיאור המרכיב האלגוריתמי – חישובי

# 10.1איזה בעיה בא לפתור, איך יפתור?

ניהול יעיל של הנחות על מוצרים או קטגוריות באתר.

הבעיות יפתרו על ידי procs שיצרנו ב-SQL שיודעים לקבל מספר שיהווה את אחוזי ההנחה ושינוי ערך ה-''מחיר סופי'' של המוצר המדובר, או לחילופין תת קטגוריה או קטגוריה שלמה.  
לכל proc כזה יצרנו גם הליך הפוך על מנת לבטל את ההנחה בסקאלה של מוצר אחד, תת קטגוריה או קטגוריה שלמה.

# תיאור/התייחסות לנושאי אבטחת מידע

# נא לציין אזורים הדורשים אבטחה, כגון: שרת, בקרת גישה לאתר, חשבונות משתמשים, מאגרי מידע וכיצד ניתן מענה.

# נא ציין מס' מקרים ותגובות להם ניתן מענה אבטחתי.

חשבונות משתמשים:  
לכל לקוח ובעל עסק יהיו שם משתמש וסיסמה לבחירתו שיאוחסנו במערכת, את הסיסמה יהיה ניתן לשנות בהמשך.  
משתמש לא יוכל לגשת לפרטיו של משתמש אחר.

מסד הנתונים כמובן יוגן על ידי שם משתמש וסיסמה ולא יוכלו לגשת אליו, לכתוב אליו ללא רשות או לדעת בכלל היכן מאוחסן ומה פרטיו.

# משאבים הנדרשים לפרויקט:

# 12.1מספר שעות המוקדש לפרויקט, חלוקת עבודה בין חברי הצוות

350+ שעות

# 12.2ציוד נדרש

מחשב, טלפון סלולרי

# 12.3תוכנות נדרשות

Visual Studio Code + Thunder Client, Azure Data Manager, Chromium browser, Adobe XD, Photoshop.

# ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט

* הרחבת הידע ב React
* Node.JS וחבילות כגון Express, Joi, Cors, Http, Nodemon
* עבודה עם API, בדיקת פקודות באמצעות ThunderClient, חיבורו לצד שרת SQL וצד לקוח בריאקט.
* עבודה ב-Adobe XD
* חיזוק משמעותי בידע בSQL, כתיבת procs ושימוש במפתחות.
* שיפור בCSS ולמידת SCSS.

# ספרות ומקורות מידע

* חיפושים בגוגל על כל גווניהם
* Stack Overflow
* שי אברהם, המנחה והמרצה שלנו
* YouTube
* Docs של חלק מהשפות.