

# מסמך הגשת קוד מודול מלאי

מגישים:

גיל חטאב ת.ז. 206226581

אבי פרדמן ת.ז. 316420132

הוראות הפעלה :

תחילה נתאר את הפונקציות שמערכת המלאי שלנו מאפשרת.

חילקנו את מודול המלאי ל-5 תתי חלקים שכל אחד מהם אחראי על פעולות ששייכות אליו :

Inventory	(1
Location	(2
Transaction	(3
Defect	(4
Report	(5

i. כעת נפרט מה הפונקציות הקיימות בכל אחד מתתי החלקים ונפרט את הפרמטרים שמצפות לקבל :

## Inventory section

(1.1) Add new 'Sale product' – מגדירה פרטים על מכירת מוצר עפ"י קביעת המחסנאי, **פרמטרים:**

- מספר האובייקטים (sale product) החדשים שהשתמש מעוניין להוסיף. (number)
- ברקוד המוצר שהשתמש מעוניין ליצור עבורו את האובייקט. (number)
- שם המוצר שהשתמש מעוניין ליצור עבורו את האובייקט. (string)
- מחיר המוצר שהשתמש מעוניין לקבוע עבור המוצר. (number)
- גודל ההנחה שהשתמש מעוניין לקבוע עבור המוצר (number)
- גודל ההנחה שהשתמש מעוניין לקבוע עבור המוצר (number)

(1.2) Edit 'Sale product' – עורכת פרטים על מכירת מוצר קיימת, עפ"י קביעת המחסנאי. תחילה המחסנאי בוחר בין 2 אופציות, האם הוא מעוניין לערוך :

- את המחיר ← לחץ 1
- את ההנחה ← לחץ 2

עבור כל בחירה כזו :

### פרמטרים:

- מספר האובייקטים (sale product) שהשתמש מעוניין לערוך. (number)
- עבור כל אובייקט שהשתמש מעוניין לערוך עליו לספק את הפרמטרים :

- ברקוד המוצר שהשתמש מעוניין ליצור עבורו את האובייקט. (number)
- מחיר / ההנחה החדשה (number)

(1.3) Alter product information – עורכת פרטים על מוצר קיים, עפ"י קביעת המחסנאי. תחילה המחסנאי בוחר בין 3 אופציות, האם הוא מעוניין לערוך :

- את הכמות המינימאלית שכאשר המוצר מגיעה לכמות זו הוא יקבל התראה ← לחץ 1
- את הספק ← לחץ 2
- את זמן האספקה הבא של המוצר ← לחץ 3

עבור כל בחירה כזו :

### פרמטרים:

- מספר המוצרים שהשתמש מעוניין לערוך. (number)

עבור כל אובייקט מוצר מעוניין לערוך עליו לספק את הפרמטרים :

- ברקוד המוצר שהמשתמש מעוניין ליצור עבורו את האובייקט. (number)
- הכמות / שם הספק / זמן האספקה הבא, בהתאמה (date) (string) (number)

1.4 Add category – מוסיפה קטגוריה חדשה, עפ"י קביעת המחסנאי. תחילה המחסנאי בוחר בין 2 אופציות, האם הוא מעוניין לערוך :

- להוסיף קטגוריה ראשית ← לחץ 1
- להוסיף קטגוריה משנית ← לחץ 2

עבור כל בחירה כזו :

#### פרמטרים:

- מספר הקטגוריות שהמשתמש מעוניין להוסיף. (number)
- עבור כל קטגוריה שהמשתמש מעוניין לערוך עליו לספק את הפרמטרים :

- שם הקטגוריה החדשה. (string)
- אילו בחר בהוספת תת קטגוריה יש לספק גם :
- שם קטגוריית האב שאליה הקטגוריה תקושר (string)

1.5 Edit category – עורכת קטגוריה קיימת, עפ"י קביעת המחסנאי. תחילה המחסנאי בוחר בין 2 אופציות, האם הוא מעוניין לערוך :

- לערוך שם קטגוריה קיימת ← לחץ 1
- למחוק קטגוריה קיימת ← לחץ 2

עבור כל בחירה כזו :

#### פרמטרים:

- מספר הקטגוריות שהמשתמש מעוניין לערוך. (number)
- עבור כל קטגוריה שהמשתמש מעוניין לערוך עליו לספק את הפרמטרים :

אם הוא מעוניין בעריכה :

- שם הקטגוריה הנוכחי. (string)
- שם הקטגוריה החדש. (string)
- אם הוא מעוניין במחיקה :
- שם הקטגוריה (string)

- הערה בנוגע למחיקה : תתבצע אך ורק אם לא קיימות קטגוריות תחת הקטגוריה שאותה מעוניינים למחוק וגם לא קיימים מוצרים תחת קטגוריה זו.

1.6 Connect product to category – מעבירה מוצר מסויים להיות תחת קטגוריה מסויימת :

#### פרמטרים:

- מספר המוצרים שהמשתמש מעוניין לקשר לקטגוריות. (number)
- עבור כל קישור יש לספק :

- ברקוד המוצר שאותו מעוניינים לקשר. (int)
- שם הקטגוריה שאליה הוא יקושר. (string)

1.7 Product information – מדפיסה מידע על מוצר, עפ"י קביעת המחסנאי. תחילה המחסנאי בוחר בין 2 אופציות, האם הוא מעוניין לקבל :

- מידע כללי ← לחץ 1
- מידע על פרטי מכירת המוצר ← לחץ 2

עבור כל בחירה כזו :

### פרמטרים:

- מספר המוצרים שהמשתמש מעוניין לקבל מידע עליהם. (number)  
עבור כל מוצר שהמשתמש מעוניין לקבל מידע עליו:

○ ברקוד המוצר. (number)

- (1.8) Print all categories name in store – מדפיסה את שמות כל הקטגוריות בחנות.  
פרמטרים: אין

## Location section

- (2.1) Move product – מעבירה מוצר ממיקום למיקום עפ"י קביעת המחסנאי,  
פרמטרים:

- מספר המוצרים השונים שהמשתמש מעוניין להעביר. (number)  
עבור כל אובייקט שהמשתמש מעוניין לערוך עליו לספק את הפרמטרים:

○ ברקוד המוצר (number)

○ תאריך תפוגת המוצר (date)

○ כמות שאותה רוצה להעביר (number)

○ מיקום נוכחי (number)

○ מיקום יעד (number)

- (2.2) Show locations by barcode – מציגה מיקומים של מוצר מסויים, עפ"י קביעת המחסנאי.  
פרמטרים:

- מספר המוצרים שהמשתמש מעוניין לראות את מיקומם. (number)  
עבור כל מוצר שהמשתמש מעוניין לראות את מיקומיו עליו לספק את הפרמטרים:

○ ברקוד המוצר שהמשתמש מעוניין לראות את מיקומיו. (number)

## Transaction section

- (3.1) Sell product – ביצוע מכירת מוצר עפ"י קביעת המחסנאי,  
פרמטרים:

- מספר המוצרים השונים שהמשתמש מעוניין לבצע מכירה עבורם. (number)  
עבור כל אובייקט שהמשתמש מעוניין לערוך עליו לספק את הפרמטרים:

○ ברקוד המוצר (number)

○ תאריך תפוגת המוצר (date)

○ כמות שאותה מוכר (number)

● הערה: ההנחה היא שהמוצר על המדף (location = 2)

- (3.2) Purchase barcode – קניית מוצר מספק, עפ"י קביעת המחסנאי.  
פרמטרים:

- מספר המוצרים שהמשתמש מעוניין לראות את מיקומם. (number)  
עבור כל מוצר שהמשתמש מעוניין לראות את מיקומיו עליו לספק את הפרמטרים:

○ שם הספק (string)

○ ברקוד המוצר (number)

○ שם המוצר (string)

○ מחיר קנייתו (number)

○ הנחה על קנייתו (number)

- תאריך תפוגת המוצר (date)
- כמות שאותה מוכר (number)
- מיקום לאחסן את האספקה של המוצר (number)

## Defect section

- (4.1) Add defect – עדכון מוצג פג עפ"י קביעת המחסנאי, **פרמטרים:**
- מספר המוצרים הפגים שהמשתמש מעוניין לעדכן. (number)
  - עבור כל אובייקט שהמשתמש מעוניין לערוך עליו לספק את הפרמטרים:
    - ברקוד המוצר (number)
    - כמות פגומה (number)
    - סיבה מדוע זה הפג פגומים (string)
    - מזין העדכון (string)
    - מיקום המוצרים (number)
    - תאריך תפוגת המוצר (date)

## Report section

- (5.1) Inventory report – הצגת דוח מלאי עדכני **פרמטרים: אין**
- (5.2) Report by category – הצגת דוח מלאי עדכני עפ"י קטגוריות שנקבעות ע"י המחסנאי. **פרמטרים:**
- מספר הקטגוריות שהמשתמש מעוניין לראות עבורם דוח מלאי. (number)
  - עבור כל מוצר קטגוריה שהמשתמש מעוניין לראות עבורה דוח מלאי עליו לספק את **הפרמטרים:**
    - שם הקטגוריה (string)
- (5.3) Defect report – הצגת דוח מוצרים פגים עדכני עפ"י תאריך שנקבע ע"י המחסנאי. **פרמטרים:**
- תאריך שממנו והלאה המשתמש מעוניין לראות את המוצרים הפגים.
- ii. כעת נפרט את המצב ההתחלתי של המערכת : (לאחר ביצוע פונ' initialize ב-main) :

### קיימות 3 קטגוריות ראשיות:

- Dairy (1)
- Canning (2)
- Personal care (3)

### קישרנו תת קטגוריות:

- Milk תחת Dairy (1)
- Large תחת Milk (2)
- Shower תחת Personal care (3)

קנינו את המוצרים הבאים עם הפירוט הבא :

*Milk* (1)

Milk barcode: 1, supplier: tnuva, price: 10, discount: 0, expiration date: 17/05/2020,  
amount: 200, supply time: 10/05/2020, location: SHELF (2)

קישרנו את *Milk* לקטגוריה: *Large*

יצרנו לו אובייקט מכירה:

Price: 6.5, discount: 0

העברנו 50 יחידות מ- *SHELF* ל- *MINOR\_STORAGE*

*Tuna* (2)

Tuna barcode: 2, supplier: StarKist, price: 6.5, discount: 2, expiration date: 17/05/2020,  
amount: 800, supply time: 10/05/2020, location: STORAGE (1)

קישרנו את *Tuna* לקטגוריה: *Canning*

יצרנו לו אובייקט מכירה:

Price: 12.5, discount: 2.5

העברנו 600 יחידות מ- *STORAFE* ל- *SHELF*

*Shampoo* (3)

Tuna barcode: 3, supplier: Dove, price: 20, discount: 5.5, expiration date: 17/05/2020,  
amount: 900, supply time: 10/05/2020, location: STORAGE (1)

קישרנו את *Shampoo* לקטגוריה: *Shower*

יצרנו לו אובייקט מכירה:

Price: 25, discount: 3

העברנו 600 יחידות מ- *STORAFE* ל- *SHELF*

*Cheese* (4)

Tuna barcode: 4, supplier: tnuva, price: 8, discount: 3, expiration date: 17/05/2020,  
amount: 1000, supply time: 10/05/2020, location: STORAGE (1)

קישרנו את *Cheese* לקטגוריה: *Dairy*

יצרנו לו אובייקט מכירה:

Price: 6, discount: 0

העברנו 450 יחידות מ- *STORAFE* ל- *SHELF*

*Shocko* (5)

Tuna barcode: 5, supplier: tnuva, price: 10, discount: 1.5, expiration date: 17/05/2020,  
amount: 1000, supply time: 10/05/2020, location: STORAGE (1)

קישורנו את *Shocko* לקטגוריה: *Milk*

יצרנו לו אובייקט מכירה:

Price: 14, discount: 0

העברנו 555 יחידות מ- *STORAFE* ל- *SHELF*

#### יצרנו מוצרים פגים:

- (1) תאריך: 10/05/2020, ברקוד: 5, כמות: 200, סיבה: נפילה מהמשאית, יוצר: אבי פרדמן, מיקום: 1, תאריך תפוגה: 17/05/2020
- (2) תאריך: 10/05/2020, ברקוד: 4, כמות: 50, סיבה: נפילה מהמשאית, יוצר: אבי פרדמן, מיקום: 1, תאריך תפוגה: 17/05/2020
- (3) תאריך: 10/05/2020, ברקוד: 3, כמות: 155, סיבה: נפילה מהמשאית, יוצר: אבי פרדמן, מיקום: 1, תאריך תפוגה: 17/05/2020
- (4) תאריך: 10/05/2020, ברקוד: 2, כמות: 40, סיבה: נפילה מהמשאית, יוצר: אבי פרדמן, מיקום: 1, תאריך תפוגה: 17/05/2020
- (5) תאריך: 10/05/2020, ברקוד: 2, כמות: 5, סיבה: לקוח הפיל מהמדף, יוצר: אבי פרדמן, מיקום: 2, תאריך תפוגה: 17/05/2020

#### מכרנו מוצרים:

- (1) בתאריך: 15/01/2020, ברקוד: 1, כמות: 3, תאריך תפוגה: 17/05/2020
- (2) בתאריך: 15/01/2020, ברקוד: 1, כמות: 2, תאריך תפוגה: 17/05/2020
- (3) בתאריך: 15/01/2020, ברקוד: 1, כמות: 7, תאריך תפוגה: 17/05/2020
- (4) בתאריך: 15/01/2020, ברקוד: 2, כמות: 9, תאריך תפוגה: 17/05/2020
- (5) בתאריך: 15/01/2020, ברקוד: 2, כמות: 12, תאריך תפוגה: 17/05/2020
- (6) בתאריך: 15/01/2020, ברקוד: 5, כמות: 14, תאריך תפוגה: 17/05/2020
- (7) בתאריך: 15/01/2020, ברקוד: 3, כמות: 5, תאריך תפוגה: 17/05/2020
- (8) בתאריך: 15/01/2020, ברקוד: 3, כמות: 5, תאריך תפוגה: 17/05/2020
- (9) בתאריך: 15/01/2020, ברקוד: 2, כמות: 5, תאריך תפוגה: 17/05/2020

#### הערות כלליות על המערכת:

- אחת ההנחות היא שניתן לבצע בכל יום לכל היותר דוח אחד מכל סוג, ולכן אם כבר בוצע דוח ואז מתבצעים שינויים והמשתמש מבקש עוד דוח באותו היום הוא לא יראה שקרו שינויים מהדוח הקודם של אותו היום.
- המיקומים נקבעים עפי הקידוד:
  - DEFECTS ← 0
  - STORAGE ← 1
  - SHELF ← 2
  - MINOR\_STORAGE ← 3

#### דיאגנוסת המחלקות לאחר המימוש:

- נציין כי שכבת ה- *Business* שלנו לא השתנתה כלל מהתכנון המקורי שהגשנו. עם זאת לא הצגנו את המחלקות בשכבת ה- *Presentation* (בתרגיל הגשת התרשימים)

מכיוון שחשבנו שיש להתמקד בשכבת הביזנס בתרשים. לכן כעת הוספנו את שכבת הפרזנטיישן.

- בנוסף, אנו מודעים לכך שאין צורך בתרגיל זה לשים דגש על שכבת ה-DAL אבל יצרנו אותה לצורך נוחות בכדי להכין את התשתית לשילוב של *database* למערכת. לכן נציג שכבה זו בתרשים המחלקות לא באופן מפורט.

תרשים המחלקות המעודכן יופיע בעמוד הבא :



