

## גרסה 2 - רכיב ייצוג, התראות, והגדרות מדיניות - 2022

בגרסה 2 נוסף רכיב ייצוג (presentation), נדרשת תמיכת השוק בהתראות זמן-אמת ומושהות, מוגדרים סוגים וכללים לקנייה ולהנחה, ונוספות מספר פעילויות לבעלי תפקידים בשוק.

תפקידי צוות: מנהל/ת גרסה המשמש גם כבודק מטעם הלקוח, ומפתחים.

### הדרישות לגרסה 2:

1. **דרישות רמת שירות:** מימוש רכיב ייצוג (presentation layer) שמותאם להצגה בדפדפנים. כדלהלן:
  - a. **דרישה 3 - אבטחה:** שימוש בפרוטוקולי תקשורת מאובטחים (HTTPS/WSS) בתקשורת בין רכיב הייצוג לבין מערכת המסחר. אפשר להנפיק TLS Certificate בחינם ב-<https://letsencrypt.org/>.
  - b. **דרישה 4.3 - אפיון ומימוש ממשק משתמש נוח:** הנחיות לדרישה זו מופיעות בהמשך.
  - c. **דרישה 4.4 - ממשק משתמש ייעודי עבור כל סוג משתמש:** מאפשר רק פעולות שהמשתמש רשאי לבצע, בהתאם לתפקידיו והרשאותיו.
2. **דרישות פונקציונליות:**
  - a. **דרישה II.4.2 -** הגדרה ועריכה של סוגי וכללי הנחה ורכישה בחנות על ידי בעל חנות. מידע נוסף על סוגי וכללי הנחה ורכישה מפורט בנספח.
  - b. **דרישה II.4.5 -** הסרת מינוי של בעל חנות על ידי הממנה שלו.
  - c. **פעילות חדשה של מנהל שוק המבקר בשוק (II.6.6) -** קבלת מידע על מנויים המבקרים בשוק ומנויים שאינם בשוק.
  - d. **דרישה II.6.2 -** ביטול מנוי של השוק על ידי מנהל שוק. הפעילות מוגבלת לביטול מנוי שאינו בעל תפקידים בשוק (מנהל חנות, מייסד/בעל חנות, מנהל שוק). כדאי לשים לב, שתרחיש זה כולל שני חלקים נפרדים: בדיקה של כשירות הפעולה, והסרה של מנוי (שמוטב שתבצע על ידי הרכיב שעוסק ברישום המנויים ובפרטיותם).
  - e. **דרישה I.5&6 של פעילות מערכת המסחר - התראות, זמן-אמת ומושהות.**  
**קבוצות של שישה סטודנטים/ות:** יש לממש גם התראות (זמן-אמת ומושהות) לבעלי חנות כאשר משתמש מנוי כותב ביקורת על מוצר בחנות.
3. **העשרה בתת-סוגים:** הגדרה של סוגים וכללים של הנחה ורכישה - כמפורט בנספח לגרסה 2.

### דגשים למימוש:

1. **ארכיטקטורה:** דרישות רמת השירות של הקיבול והזמינות, שכבר נכללו בגרסה 1, הכתיבו ארכיטקטורת שרת-לקוח, כאשר רכיב שכבת השירות (Service) משמש כמנשק לרכיב ה-domain. הארכיטקטורה העדכנית של המערכת תכלול את הרכיבים הבאים:
  - a. **רכיב לקוח (presentation layer):** Web Client שמנגיש את המערכת למשתמש.
  - b. **רכיב תקשורת:** העברת הפניות אל רכיב הלקוח וממנו. מספק תמיכה בשני אופני התקשורת אל המערכת: HTTPS ו-WSS.
  - c. **רכיב שירות (service layer):** עדכני.
  - d. **רכיב domain:** עדכני. כולל רכיב התראות זמן-אמת ומושהות (publisher) המאפשר תקשורת מה-domain ללקוח.
2. **ממשק-משתמש גרפי:**

- a. יש לממש את ממשק המשתמש כ- **Web client** המשמש כממשק גרפי (GUI).
  - b. **תרחישי שימוש ומודל מצבים היררכי**: פעולות הממשק מוכתבות על ידי הדרישות הפונקציונליות. עבור תרחישי השימוש: II.1.4 – login, ו-II.4.6 – מינוי מנהל חנות, מימוש הממשק צריך להיות תואם למודל המצבים ההיררכי שנבנה עבורם (בגרסה 1).
  - c. **תרחישי בדיקה**: יש לוודא שממשק המשתמש עובר בהצלחה תרחישי בדיקה על פי מבחני הקבלה.
  - d. הממשק תומך בקבלת **התראות זמן-אמת** באמצעות פרוטוקול WebSockets, ובהצגתן.
  - e. הממשק מאפשר הגדרה של סוגי הנחה וקנייה בחנות (מבין הסוגים הנתמכים על ידי מערכת המסחר), והגדרה ועריכה של כללי הנחה וקנייה בחנות והרכבות שלהם.
3. **בדיקות**: מורחבות עבור הדרישות החדשות, **כולל עבור התראות זמן-אמת ומושהות** (לא דרך המימשק).

## תוצרי גרסה 2

1. **מודלים עדכניים**: מחלקות, ארכיטקטורה, תרחישי שימוש, מילון.
2. **אפיון ממשק משתמש נוח**: מסמך דרישות עבור האספקט הגרפי של ממשק המשתמש (ראה/י נספח).
3. **מימוש תואם מודלים**:
  - a. שכבת הייצוג (presentation), תואמת למסמך אפיון הגרפיקה, ולתרחישי השימוש. ייבדק גם תיאום עבור שני תרחישי השימוש שתוארו באמצעות מודל מצבים היררכי בגרסה 1 – הזדהות במערכת (תרחיש II.1.4) ומינוי מנהל חנות (תרחיש II.4.6).
  - b. מימוש התראות זמן-אמת ומושהות, על פי תבנית העיצוב Observer.
  - c. לב המערכת תואם (כולל) למודל המחלקות; מבנה המערכת תואם את דיאגרמת הארכיטקטורה.
4. **בדיקות**: מורחבות עבור הדרישות החדשות, עבור התראות, ובדיקות לממשק הגרפי (לא ממוחשב). **לא יתקבל מימוש ללא רכיב בדיקות משמעותי!**

## מצגת כיתה: עקביות ושמירת נתונים לאורך זמן

- הגרסה הבאה של המערכת תאפשר גיבוי נתונים לאורך זמן באמצעות רכיב מסד נתונים.
1. בחר/י ספרייה המספקת **קישור ותרגום של מערכת לבסיס נתונים**. למשל, עבור מסד נתונים טבלאי, תתאים ספריית ORM שתקשר בין מערכת בגישה מונחית עצמים לבין מודל טבלאי. כלי כזה מאפשר מיפוי של ישויות של המערכת לישויות של בסיס הנתונים, ומטפל בשמירה ואחזור של מידע בבסיס הנתונים.
  2. הסבר/י והדגם/י כיצד הספרייה משמשת כגשר בין המערכת לבין בסיס הנתונים (Data access layer): **תאר/י את השינויים** שיש להכניס למערכת (אפליקציה).
  3. הדגם/י עבודה עם הכלי. בפרט, הצג/י את התמיכה והשימוש בטרנסאקציות.
  4. הסבר/י והדגם/י את אופן עבודת הספרייה עבור מצבים של אי-תאימות בין האפליקציה לבין הספרייה:
    - a. התמודדות עם היררכיית מחלקות.
    - b. התמודדות עם ההקשר (context) של אובייקטים: שמירת הקשרים של אובייקט לסביבתו.

## נספח לגרסה 2

### הרחבת הדרישות לסוגי וכללי הנחה ורכישה

#### 1. תמיכה בהנחות קנייה: סוגים וכללים

- a. הנחות מוגדרות באופן כפלי, באחוזים. ההנחה יכולה להיות ברמת מוצר, קטגוריה או חנות:
1. "הנחה של 50% על מוצרי חלב".
  2. "מכירת חיסול! 20% הנחה על כל החנות!".
- b. סוגי הנחה: פשוטה ומותנית.
1. הנחה פשוטה – כרוכה רק בחישוב מחיר.
  2. הנחה מותנית – מלווה בתנאי (פרדיקט) להפעלתה:
- "קנייה שערכה מעל 200 ש"ח בחנות מזכה ב-10% הנחה על עגבניות".
- c. כללי הנחה: כללים להרכבה של הנחות. כללי הנחה מייצרים, בזמן ריצה, הנחות המורכבות מהנחות פשוטות יותר. יש שני סוגי הרכבה:
1. הרכבה לוגית: הרכבה של הנחות באמצעות הפעולות הלוגיות **וגם**, **או**, **או-בררני**:  
(*and, or, xor*). לדוגמא:
- **xor**: "הנחה על מוצרי חלב **או** על מאפים, אבל לא שתיהן. בתוספת כלל להכרעה".
  - **and**: "יש הנחת מאפים של 5% על לחמניות ולחם רק אם הסל מכיל לפחות 5 לחמניות **וגם** לפחות 2 כיכרות לחם".
  - **or**: "יש 5% הנחה על מוצרי חלב אם הסל מכיל לפחות 3 גביעי קוטג' **או** לפחות 2 יוגורטים".
  - **הרכבה של תנאים**: אפשר גם לבנות הנחה מותנית על ידי הרכבת תנאים: "אם שווי הסל גבוה מ-100 ש"ח **וגם** הסל מכיל לפחות 3 חבילות פסטה, אז יש הנחה של 5% על מוצרי חלב".
2. הרכבה מספרית: הרכבת ערך ההנחה באמצעות הפעולות **מקסימום** ו**חיבור**. לדוגמא:
- **מקסימום**: "ההנחה הניתנת היא המקס' מבין הערך בש"ח של 5% מעלות כל חבילות הפסטה בסל לבין הערך בש"ח של 17% מעלות קרטוני החלב בסל".
  - **חיבור**: "יש 5% הנחה על מוצרי חלב, ובנוסף, יש 20% הנחה על כל חנות לכן על מוצרי חלב יש בסה"כ 25% הנחה".

#### 2. תמיכה באופני (כללים) קנייה שונים

- a. בשלב זה, סוג הקנייה הנתמך היחיד הוא **קנייה מיידית**.
- b. כללים לקנייה הם תנאים (פרדיקטים) שסל הקנייה צריך לקיים. כלל קנייה יכול להיות מוגדר ברמת מוצר, קטגוריה, סל קניות, או משתמש:
1. "סל קניות לא מכיל יותר מ-5 ק"ג עגבניות".
  - "משתמש מתחת לגיל 18 לא מורשה לקנות אלוהול".
  - "לא ניתן לקנות אלוהול אחרי שעה 23:00".
  - "בראש חודש אין מכירה של גלידות".
2. הרכבה של כללי קנייה: אפשר להרכיב כללי קנייה באמצעות פעולות לוגיות: **וגם**, **או** ו**התניה**: (*and, or, conditioning*). לדוגמא:
- **and**: "סל קניות לא מכיל יותר מ-5 ק"ג עגבניות **וגם** חייב להכיל לפחות 2 תירסים".
  - **or**: "לא ניתן לקנות אלוהול אחרי שעה 23:00 **או** שזה ערב חג".
  - **conditioning**: "ניתן לקנות 5 ק"ג עגבניות **רק אם** הסל כולל חצילים".

### 3. דגשים למימוש

- a. כדאי להשתמש בתבנית העיצוב **Composite** למימוש של הרכבה – מסוגים שונים.
- b. על פי התבנית, רצוי להגדיר **מחלקות הרכבה** עבור הסוגים השונים והאופרטורים השונים. למשל,
  - **הרכבת or של תנאים** – יכול לשמש עבור תנאים מורכבים של הנחות, ועבור כללי קנייה מורכבים מסוג or.
  - **הרכבת התניה של כללי קנייה**: "ניתן לקנות 5ק"ג עגבניות רק אם (הסל כולל לפחות 2 חצילים או שזה ערב חג)". שימו לב לקינון של הפעולות בהגדרה המורכבת. תבנית ה-composite תומכת באופן טבעי בדירוג של הרכבה.
  - **הרכבת xor של הנחות**

### אפיון ממשק משתמש נוח

צריך לספק מסמך אפיון (**ברמה הנדסית**) לדרישת רמת שירות **4.3** (ממשק משתמש נוח), למשל, הדרישה "המלל צריך להיות קריא" היא ערטילאית ולא ניתנת לבדיקה באופן הנדסי. המסמך צריך להגדיר את המושג **ממשק נוח** באמצעות אוסף מאפיינים הניתנים לבדיקה אוטומטית. לכל מאפיין צריך להגדיר את **המטרה, המדדים, ומידת הדרישה – SLO, SLI, SLA**, בהתאמה, בדרישות רמת שירות. לכל מדד צריכה להיות **בדיקה טכנית**. למשל:

- מטרה: "מסר (תוכן) ברור לכל אלמנט גרפי". מדד אפשרי הוא: "בכל רזולוציה נתמכת, אין שני אלמנטים גרפיים חופפים (למעט דיאלוגים מרחפים)". מידת הדרישה יכולה להיות 100%.
- מטרה: "נוחות שימוש". אחד המדדים יכול להיות: "רוחב התוכן מוגבל לרוחב המסך (אין גלילה ימינה/שמאלה)".
- מטרה: "טקסט קריא ללא עזרי קריאה". אחד המדדים יכול להיות: "גודל אותיות על המסך תואם את התקן של מבחני אופטומטריסטים לקריאה".