**פרויקט ספרייה – תיאור הפרויקט - בר עובד :**

1.המערכת מנהלת ספרים וכתבי עת בחנות ספרים.

2.המערכת שומרת מידע לגבי הפריטים, מאפשרת הוספה ועריכת פריטים וכן מציגה אותם.

3.קיימים הנחות על הפריטים לפי מחבר, כתב ספר, לפי ג'אנר או לפי סוג המוצר.

4.בחנות אין כפל מבצעים , ההנחה הגבוהה ביותר היא הנבחרת.

5.השתמשתי ב INTERFACE בכל מקום שנדרש ממני כדי לאפיין שייכות ולצורך הצהרה על מילוי ממשק ספציפי.

6.קיימים בשרת ובכל מקום שנדרשתי לפנות לקובץ חיצוני(התחברות לשרת או התחברות לבסיס הנתוניםEXCEPTIONS ובנוסף כתיבה ללוגר חיצוני המודיע על מיקום נפילת השגיאה.

7.קיימים UNIT TESTING על המודלים בפרויקט.

8.קיימים תיעודים עבור כל פונקציה כדי להבין מה נעשה.

9.את הפרויקט כתבתי ב WEB API CORE צד שרת ,ו UWP צד קליינט.

10.הפרויקט מחולק ל 5 פרויקטים: client :UI , clientLib , web api , shared ,unit testing.

ClientLib-לשם ביצוע קריאות HTTP לשרת.

Server-WEB API ASP.NET CORE שם אני מנהל את כל הקונטרולרים וסרוויסים תוך שימוש ב dependency injection.

Shared-שם נמצא כל המודלים.

Client with mvvm light

Unit testing-שם מתנהל עבור כל יחידה בדיקות .

הרחבות שבצעתי בפרויקט:

1.הרשאות לפי עובד ומנהל.

\* רק מנהל יכול להוסיף פריט.

\* מנהל יכול לעדכן פריט.

\* מנהל יכול לצפות בהיסטוריית הקניות המלאה שבוצעה בחנות עד היום.

\* מנהל יכול לצפות בהיסטוריית הקניות לפי לקוח שבוצעה עד היום.

\* מנהל יכול לצפות בהיסטוריית המבצעים.

\* עובד רגיל יכול רק לבצע מכירה בלבד.

2.לא יתאפשר לקנות מוצר שנגמר בחנות עד שהמנהל לא מעדכן את הכמות של אותו מוצר.

3.ביצוע פילטרים.

\* קיים חיפוש מוצר לפי הסוג, לפי השם או לפי הנחה בכלל.

\* קיים חיפוש מוצר לפי הנחת הוצאה לאור , הנחת כותב ספר או הנחה לפי ג'אנר .

\* קיים חיפוש מוצר בהנחה לפי סוג.

4.קיימת התחברות לחנות וכן רישום של עובד חדש.