**דרישות המערכת**

המערכת תנהל ספרים, וכתבי עט בספריה העירונית

המערכת תאפשר:

חלק ראשון (נדרש לפני שאר ההרחבות)

* הצגת פריט (5 נק')
* הוספת פריט (10 נק')
* עריכת פריט (10 נק')
* מחיקת פריט (5 נק')
* פילטור פריטים על פי תנאי בודד (10 נק') והצגת רשימה של שמות של ספרים (10 נק') (סרגל פריטים חיפוש מהיר חיפוש בודד)
  + על פי שם פריט
  + על פי מחבר
  + על פי ז'אנר
  + על פי הוצאת ספרים

חלק שני

* השאלת פריט (5 נק') (נדרש עבור חלק 3 ו-4)
* החזרת פריט (5 נק') (נדרש עבור חלק 3 ו-4)
* הצגת הערה בפרטי הספר על איחור בהחזרת ספר מושאל (לאחר יותר משבועיים) (5 נק')
* דוח יומי על פריטים
  + כמות המוצרים סך הכל (5 נק')
    - כמות ספרים
    - כמות כתבי עט
  + כמות המוצרים המושאלים (5 נק')

חלק שלישי

* הוספת מבצע על פי פרמטר בודד (15 נק')
  + לפי הז'אנר
  + לפי מחבר
  + לפי תאריך ההוצאה
  + לפי הוצאת הספרים
  + בחנות מונהגת שיטת אין כפל הנחות - ההנחה הגבוהה קובעת
* סיום מבצע (5 נק') – אין להתייחס למצב של כפל מבצעים על ספר מסויים.

חלק רביעי (20 נק')

* יצירת 2 סוגי משתמשים: ספרן (יכול הכל) ומשאיל, כולל שם משתמש וסיסמא
* משאיל יכול רק:
  + לעשות חיפוש
  + להציג פריט
  + להשאיל ספר
  + להחזיר ספר.

ניתן לשמור את המידע לתוך קבצים דרך Serialization, ולטעון אותו יותר בעליית המערכת (10 נק').

ניתן להשתמש MVVM (15 נק' תוספת)

ניתן לבצע Unit Tests במידה ולמדתם (10 נק').

**מידע נדרש על כל פריט:**

* ISBN (ID לספר – Guid)
* מחבר (רק בספר)
* הוצאת ספרים/ירחונים (שם החברה)
* תאריך הוצאה
* ז'אנר (יכול להיות יותר מ-1)
* מחיר השכרה כולל חישוב לאחר הנחה
* הנחה במחיר השכרה (במידה ועשיתם את חלק 3)
* תאריך השכרה (במידה ועשיתם חלק 2)

## **מסמך אפיון**

1. להגשת הפרויקט יש לצרף מסמך אפיון קצר הכולל
   1. תיאור הפרויקט (5 נק')
   2. ציור קשרים ועץ מחלקות בפרויקט (5 נק')
   3. תיאור קצר על פקדים ומסכים בפרויקט (5 נק')

**הנחיות כלליות**

1. יש לבצע את הלוגיקה ב-Class Library
2. החלק הגרפי צריך להתבצע ב-UWP או WPF
3. יש להקפיד על התמודדות עם שגיאות ותפיסת שגיאות.
4. אין לכלול בחלון אחד מספר תפקודים שונים עבור צרכים שונים.

**הכוונה לכתיבת קוד עבור המערכת**

* ספר, וכתב עט יורשים מ-Abstract Class.
* השתמש בפרמטרים ב- Constructors במידת האפשר.
* הוסף Class בשם ItemsCollection המכיל Collection<AbstractItem> ומחזיק רשימה של Items.
* כלול ב ItemsCollection מתודות המאפשרות הוספת מוצר לרשימה, וחיפוש על פי תנאים.
* ניהול Exceptions תקין ובדיקת שגיאות (במיוחד של פרטים מהמשתמש)

## **שלביי עבודה בפרוייקט בסיסי**

יש לעבוד על כל חלק, לסיים אותו ולראות שהוא עובד, ואז לעבור על החלק הבא.

1. אפיון מפורט ותכנון עץ Classes (איטרציות של כל חלק על גבי אותו מסמך)
2. יצירת ה GUI (עבור כל חלון/פיצ'ר)
3. כתיבת Class Library (איטרציות של כל חלק על גבי אותו Project)
4. הרצה ו debug. (עבור כל פיצ'ר)
5. בדיקות. (אחרי כל חלון)
6. תיעוד. (עבור כל החלק הראשון, ואז עבור כל חלק נוסף)

**חישוב ניקוד**

החלק הראשון של הרכיבים בעמוד הראשון של ההנחיות הוא המרכזי ומחייב, וכולל כ-50 נקודות מהציון.

הפרויקט מאפשר ניקוד רב בהרבה מ-100.

מערכת סלע מאפשרת להגיע רק עד 100 כך שלא ניתן לעבור אותו, אך במידה של הורדת ציון ניתן לכפר על כך בכתיבת עוד חלקים.

בונוסים אפשריים במערכת:

* כתיבת חלקי רכיבים נוספים לפרויקט מהרשימה בעמוד הראשון.
* כתיבת מסמך אפיון.
* כתיבת Serialization.
* כתיבת Unit Tests.
* שימוש בMVVM. (לא חובה !)

ציון החלק הראשון כולל את התכנים של הכותרות הבאות הנמצאים בעמוד השני של ההנחיות:

* מידע נדרש על כל פריט
* הנחיות כלליות
* הכוונה לכתיבת קוד עבור המערכת

במידה וחלק מתכני הלוגיקה חסרים זה ישוקף בתכנים של החלק הראשון, ובציון, כולל קוד שפועל אך כתוב לא נכון.

מסמך אפיון עבודת ספריה כללי שם הספר סופר ז'אנר כמות דפים הוצאת ספרים IBMS תאריך הוצאת הספר כמות ספרים זמינה במלאי האם ספר זמין להשכרה מחיר/הנחה או בלי הנחה כניסה בתור ספרן(שם משתמש וסיסמא) 1.1 עריכה 1.2 מחיקה 1.3 הוספה 1.4 הצגת פריט 1.5 הוספת מבצעים והנחות 1.6הצגת רשימת ספרים(פילטור) כניסה בתור לקוח(שם משתמש וסיסמא) 1.1 הצגת רשימת הספרים 1.2 יכולת לעשות חיפוש(פילטור) 1.3 השבת ספר 1.4 השאלת ספר