**אפליקציית טריוויה**

תרגיל מתגלגל

# תיאור

אפליקציית הטריוויה היא משחק רשת בו השחקנים רשומים במערכת וצוברים נקודות לפי רמת הידע הכללי שלהם. שחקן שנכנס למשחק מקבל שאלות טריוויה בנושאים שונים באופן רנדומלי. על כל תשובה נכונה, השחקן מקבל 10 נקודות. על כל תשובה לא נכונה, יורדים לו 5 נקודות. כאשר שחקן מגיע ל100 נקודות הוא רשאי להוסיף שאלה למערכת. השאלה נכנסת למערכת לאחר אישור של שחקן בכיר ממנו (כפי שיוסבר בהמשך).

שחקן מזוהה בעזרת המייל שלו ויש לו שם. לכל שחקן במשחק נקבעת דרגה:

* **מנהל** הוא שחקן שרשאי להוסיף שאלות בלי קשר להצלחה שלו. מנהל ממונה על ידי מנהל אחר. מנהל יכול לאשר / לפסול / לבטל שאלות וגם להוסיף נושאי טריוויה חדשים.
* **מאסטר** הוא שחקן שהוסיף 10 שאלות חדשות ומאושרות למשחק. מאסטר רשאי לאשר או לפסול שאלות חדשות ששחקנים אחרים הוסיפו.
* **טירון** הוא שחקן חדש שלא הוסיף שאלות.

שחקן לא יורד בדרגתו לעולם! גם אם הניקוד שלו יורד. הניקוד הוא תמיד בין 0 ל – 100. הניקוד מתאפס בכל פעם ששאלה מתווספת למשחק (בין אם אושרה או נפסלה).

לשאלה המבנה הקשיח הבא:

* **נושא:** נושא השאלה (לכל שאלה נושא אחד בלבד) כגון: ספורט, פוליטיקה, היסטוריה, מדע, וכו'.
* טקסט
* **תשובה נכונה**
* **תשובה שגויה 1**
* **תשובה שגויה 2**
* **תשובה שגויה 3**

**מהלך המשחק**: בכל פעם מופיעה שאלה בודדת ולאחר מתן תשובה, נקבע הניקוד. במידה והשחקן הגיע ל 100 נקודות, הוא רשאי להוסיף שאלה למערכת, ובמידת הצורך מתעדכנת דרגתו של השחקן.

פעולות נוספות באפליקציה:

* **אישור / פסילה של שאלות חדשות**: אופציה זו לא מתאפשרת לשחקנים בדרגה של טירון. בחירה באופציה זו מציגה את רשימת השאלות החדשות שטעונות אישור. בחירה של שאלה מציגה אותה על המסך עם אופציה לאשר את השאלה או לפסול אותה.
* **ניהול משתמשים:** אופציה זו מתאפשרת רק למנהלים. בחירה באופציה זו מציגה רשימה של שחקנים. עבור כל שחקן, ניתן לראות את הפרטים שלו, לשנות את הדרגה שלו, את הניקוד שלו ואת הסיסמה שלו.
* **פרטים אישיים:** אופציה זו פתוחה לכל השחקנים. בחירה בה מציגה את פרטי השחקן וניתן לעדכן אותם (שם, מייל וסיסמה). לא ניתן לעדכן דרגה וניקוד כמובן.
* **לוג אין והרשמה:** בכניסה לאפליקציה מתבצע מסך לוג אין או הרשמה לשחקנים חדשים.
* **ניהול שאלות:** אופציה זו מאפשרת רק למנהלים. בחירה באופציה זו מציגה את כל השאלות. בחירה בשאלה מראה את כל פרטי השאלה עם אפשרות לעדכן אותה. בנוסף לרשימת השאלות יש לחצן הוספה המאפשר להוסיף שאלה למאגר השאלות. במסך הוספת שאלה יש לחצן להוספה של נושא חדש למערכת.

# שלב א' – סיום שלב בסיסי נתונים ו Entity Framework

המטרה של התרגיל היא לפתח אפליקציית Console Application בשכבות כפי שנלמד בכיתה המאפשרת לשחק את משחק הטריוויה לפי התיאור לעיל תחת המגבלות הבאות:

* בשלב הזה המשחק ישוחק על מחשב בודד ולא כמשחק רשת.
* לא כל המסכים יפותחו בשלב זה. המסכים שיפותחו יכללו:
  + מסך Log in - השחקן יתבקש להכניס שם מייל וסיסמה ובמידה והם מתאימים יכנס למערכת. הפעולה של מסך זה מחזירה true אם פעולת הכניסה הצליחה או false אחרת.
  + מסך Signup – השחקן יתבקש להכניס מייל, שם, וסיסמה. במידה וההרשמה הצליחה יש להחזיר true או false אחרת.
  + מסך המשחק המציג שאלה אחרי שאלה ומאפשר לענות או לסיים.
  + מסך הוספת שאלה חדשה – מסך זה יתאפשר רק לשחקנים שיש להם כמות נקודות נדרשת או למנהלים. במידה ואין הרשאות מתאימות תינתן הודעה מתאימה.
  + מסך בדיקת שאלות שדורשות אישור. במסך הזה תופיע כל פעם שאלה אחת ויהיה אפשר לאשר או לפסול אותה (בכל שלב ניתן לסיים או לעבור לשאלה הבאה). כמו המסך הקודם – גם מסך זה צריך לפעול לפי הרשאות מתאימות.
  + מסך פרופיל שמציג את פרטי השחקן (מייל, שם, סיסמה, דרגה וניקוד) ומאפשר לעדכן מייל, שם וסיסמה בלבד.

על מנת שהאפליקציה תעבוד כמשחק, בנינו עבורכם פרוייקט המכיל תפריטים המפעילים את המסכים השונים שתכתבו כפי שיוסבר בהמשך.

**יש לעבוד על פי השלבים הבאים:**

1. יש לקרוא ולהבין את הדרישות ולתכנן בסיס נתונים מנורמל שיאפשר את פיתוח האפליקציה כפי שהיא מתוארת. תכנון בסיס הנתונים יתואר בתרשים ERD כפי שנלמד בכיתה.
2. לאחר אישור התכנון על ידי טל / עופר, יש לכתוב סקריפט בשפת SQL שבונה את בסיס הנתונים ומכניס אליו לפחות את הנתונים הבאים:
   * משתמש שמוגדר כמנהל מערכת
   * 5 נושאי טריוויה (ספורט, פוליטיקה, היסטוריה, מדע ותיכון רמון)
   * לפחות שאלה אחת בכל נושא
3. יש להעתיק מ Github את הפרוייקט הבא שיהווה נקודת התחלה לפרוייקט שלכם:  
   <https://github.com/stma1one/Trivia_Stage1>
4. יש להוסיף לפרויקט ספריות לכל שכבה כפי שנלמד בכיתה ולבנות את מחלקות המודל בעזרת Entity Framework. (יש להוסיף ספריה ובה הסקריפט לבניית בסיס הנתונים - DBScripts, ספריה למחלקות המודל – Models וספריה להרחבה של מחלקות אלו - ModelsExt)
5. יש לתכנן את שכבת ה UI על ידי למידה של המחלקה TriviaScreensImp והפעולות בה. יש לתכנן איזו אינפורמציה תרצו לשמור במחלקה זו כדי שתלווה אתכם לאורך זמן ריצת התוכנית ולעדכן את המחלקה בהתאם (לדוגמא, בוודאי תרצו לשמור את אובייקט השחקן שמשחק כרגע לאחר שביצע Login).
6. יש לחלק את פיתוח המסכים בין חברי הצוות. לפני פיתוח של מסך יש לתכנן אילו פעולות יכתבו בשכבת ה Business Logic ואילו בשכבת ה UI.
7. בתום שלב הפיתוח יש לבדוק את המערכת.
8. יש להגיש את הפרוייקט בקלאס רום על ידי שיתוף לינק ל GitHub המכיל את כל הקוד של הפרוייקט (כולל תרשים ERD וסקריפט ה SQL שישמרו בספריית ה DBScripts)