The logo features a stylized illustration of two books. The top book is blue with white horizontal lines representing pages. The bottom book is purple with a yellow bookmark sticking out from its right side. To the right of the books, the words "Smart Book" are written in a purple, rounded, sans-serif font.

Smart Book

מגישה:

תהילה בוסקילה ת.ז. 212384499

המנחה: המורה נעמה בטיט תח"י

סמינר "אוהל חיה" נתיבות

שנת תשפ"א 07/2021

תוכן העניינים

1	תוכן העניינים
3	שלמי תודה
4	הצעת פרויקט
6	תקציר/מבוא
6	הרקע לפרויקט
6	תהליך המחקר
6	סקירת ספרות
7	אתגרים מרכזיים
8	מטרות/יעדים
8	מטרות המערכת:
8	יעדי המערכת:
8	אתגרים
8	מדדי הצלחה למערכת
9	רקע תיאורטי/ ספרות מקצועית
9	רקע תיאורטי
9	ספרות מקצועית
10	תיאור מצב קיים
10	ניתוח חלופות מערכתי
10	תיאור החלופה הנבחרת
10	אפיון המערכת שהוגדרה
10	ניתוח דרישות המערכת
10	סביבת פיתוח:
11	עמדת משתמש מינימלית:
11	מודול המערכת
11	נושאים באחריות המערכת:
11	נושאים שאינם באחריות המערכת:
11	אפיון פונקציונלי
12	ביצועים עיקריים
12	אילוצים
12	תיאור הארכיטקטורה
12	תיאור צד השרת:
13	תיאור הרכיבים בפיתרון
13	תיאור פרוטוקולי התקשורת
13	שרת-לקוח
15	ניתוח ותרשים UML של המערכת המוצעת
17	מבנה נתונים בהם השתמשתי
18	תרשים מחלקות
19	תיאור המחלקות המוצעות

20	רכיבי ממשק
20	תיכון המערכת
20	ארכיטקטורת המערכת
20	תיכון מפורט
20	חלופות לתיכון המערכת
20	תיאור התוכנה
20	סביבת עבודה
20	שפות תכנות
21	תיאור מסכים
21	תרשים מסכים
22	פירוט המסכים
25	קוד התוכנית+תיעוד
25	קלט:
25	פלט:
25	פונקציות:
25	צד שרת:
27	צד לקוח:
28	פונקציות חשובות:
29	תיאור מסד הנתונים
32	מדריך למשתמש
32	מסך הספרים שלנו:
32	מסך במיוחד בשבילך:
32	מסך השאלות והחזרות:
32	חלון הרשמה/ התחברות:
32	בדיקות והערכה
32	ניתוח יעילות
32	אבטחת מידע
33	מסקנות
33	פיתוחים עתידיים
33	בבליוגרפיה

שלמי תודה

בשבח ובהודיה לבורא עולם אני מתכבדת להגיש ספר זה המאגד בתוכו את פרויקט הגמר בהנדסת תוכנה : BookSmart עליו עמלתי רבות.

בנוסף אודה לכל אלו שנתנו רבות מכוחם, מרצם ומתבונתם במהלך העבודה :

- לגב' בת שבע לוי מרכזת המגמה שבלעדיה לא היינו מגיעות לשלב זה .
- למנחה המורה נעמה בטיט שסייעה לי רבות לאורך כל הדרך בהכנת הפרויקט, החל משלב התכנון ועד לשלב הביצוע. ייעצה, בדקה ונתנה חוות דעת על מנת להגיע להישגים הטובים ביותר במסגרת הזמן והאמצעים שעמדו לרשותי.
- למורה דינה וידיסלוסקי אשר לימדה אותי להשתמש בצורה יעילה בתוכנת אנגולר וסייעה לי רבות בבניית האתר ובטיפול בבעיות.
- לצוות המורות המסור, על הנכונות הרבה והסבלנות האינסופית לייעץ לבדוק לתמוך וללמד והכל מתוך רוח טובה ונעימה.
- לסמינר "מאיר" נתיבות ולעומדת בראשו המנהלת רחל ארליך על ההשקעה והמקצועיות.
- לצוות חדרי המחשבים.
- להורי היקרים ולמשפחתי על התמיכה והעידוד לאורך כל הדרך החל מתקופת הלימודים.

הצעת פרויקט

מל מוסד: 770503

שם מכללה: סמינר "מאיר" נתיבות

שם הסטודנט: תהילה בוסקילה

ת.ז הסטודנט: 212384499

שם הפרויקט: ספריה עירונית חכמה

תיאור הפרויקט: המשתמש נכנס לאתר ובאפשרותו לצפות בקטלוג הספרים, לצפות בספרים שהשאל, לקבל הצעות לספרים שמתאימים לו לפי הפרופיל שלו, וכן באפשרותו לתת ביקורות על ספרים שקרא ובכך המערכת מעדכנת את פרופיל הספר.

הגדרת הבעיה האלגוריתמית: המערכת אוספת מידע על המשתמש ע"י פרוט הספרים שהשאל בעבר ובונה פרופיל משתמש ומתעדכנת כל הזמן לפי העדפות המשתמש.

רקע תיאורטי בתחום הפרויקט: רקע בתחום קטלוג הספרים, מערכת מתלמדת-בינה מלאכותית.

תהליכים עיקריים בפרויקט: בניית פרופיל משתמש עפ"י עבר השאלת הספרים.

תיאור הטכנולוגיה: Web Api

צד שרת: Visual Studio

שפת תכנות בצד השרת: #C

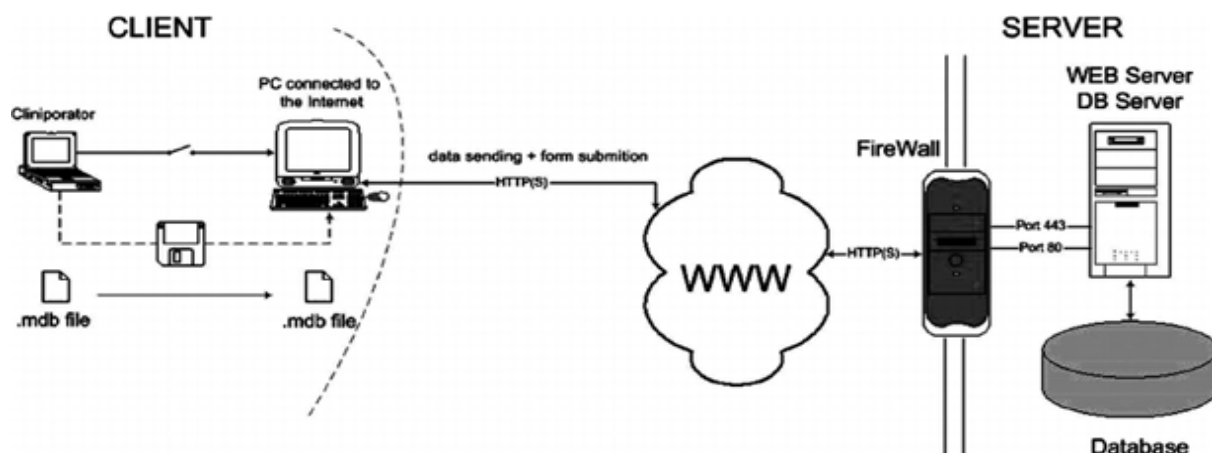
צד לקוח: Angular, Visual Studio Code

שפת תכנות בצד הלקוח: TypeScript, JavaScript, Html

מסד נתונים: SQL Server

פרוטוקולי תקשורת: המערכת מורכבת משרת IIS המריץ את האתר בסביבת ה-server,

מסד נתונים - database של sql-server, ממשק משתמש בצד הלקוח: דפדפן אינטרנט כלשהו: internet explorer, firefox, chrome,



לוחות זמנים:

1. חקר מצב קיים: נובמבר
2. הגדרת הדרישות: נובמבר
3. אפיון המערכת: דצמבר
4. אפיון בסיס הנתונים: דצמבר
5. אלגוריתמים: דצמבר-ינואר
6. עיצוב המערכת: ינואר
7. בניית התוכנה: ינואר-מרץ
8. בדיקות: אפריל
9. הכנת תיק פרויקט: מאי
10. הטמעת המערכת: מאי
11. הגשת פרויקט סופי: יולי

חתימת הסטודנט:



חתימת רכז המגמה:



אישור משרד החינוך:

תקציר/מבוא

הרקע לפרויקט

הרעיון לפיתוח אתר של ספריה עלה בראשי בתקופת הקורונה, כאשר לא היתה אפשרות ללכת לספריה ולשהות בה זמן רב על מנת לעיין בספרים. בתור בחורה שאוהבת לקרוא - הדבר היה מאתגר יותר. בנוסף לכך אנשים שהיו רגילים לצאת לספריה מידי שבוע ולחפש ספר מעניין לשבוע הקרוב - לא יכלו להרשות לעצמם זמן שהות ממושך עקב המצב. לאור עובדה זו חשבתי על בניית אתר שיתן למשתמש חווית חיפוש ספרים וסינון מאגר ספרי הספריה עפ"י רצונות המשתמש. כמו כן האתר שומר את היסטוריית הספרים אליהם נכנס המשתמש ובונה ע"י נתונים אלו פרופיל דינמי המאפשר המלצה על ספרים שמתאימים לסגנון המשתמש ללא מאמץ מצידו. אמנם, ישנם אתרים המספקים שירות של ספריה - אבל רובם ככולם מציעים חומרים, אך לא במיוחד עבור המשתמש ולפי צרכיו.

תהליך המחקר

תחילה התחלתי להתעניין ולברר האם הרעיון שלי כבר בוצע וגיליתי שאכן קיימים אתרי ספריות אך אלו מציעים חומר קריאה ללא כל התייחסות לחווית החיפוש המתאימה את עצמה למשתמש לפי העדפותיו וצרכיו.

סקירת ספרות

האתרים שבהם נעזרתי:

[/https://www.w3schools.com](https://www.w3schools.com)

[/https://freshdesignweb.com](https://freshdesignweb.com)

[/https://freefrontend.com](https://freefrontend.com)

[/https://material.angular.io](https://material.angular.io)

[/https://codepen.io](https://codepen.io)

[/https://www.codebrain.co.il](https://www.codebrain.co.il)

[/https://reshetech.co.il](https://reshetech.co.il)<https://www.tutorialspoint.com/index.htm>[/https://angular.io](https://angular.io)

אתגרים מרכזיים

בחרתי לבנות אתר ספריה כיוון שאני מאוד אוהבת ספרים ותמיד רציתי שיהיה משהו יעיל שיעזור למשתמש למצוא בקלות ובחוויתיות ספר מתאים עבורו. ובמיוחד שזו היתה תקופת הקורונה שלא היה ניתן להגיע לספריה ולשבת לחפש מבין הספרים ספר שמתאים לי.

כשהתחלתי לחשוב איך האתר שלי ילמד את נתוני המשתמש והספרים שיתאימו לו, עלה בראשי לעשות מבנה נתונים מסוג Lists. כשנכנסתי למחשבה עמוקה יותר הבנתי שזה יהיה עם סיבוכיות לא יעילה היות ולצורך ביצוע האלגוריתם צריך לעבור שוב ושוב על הרשימה, לכן חיפשתי עוד דרכים לביצוע הרעיון ובאמת מצאתי פיתרון בעל סיבוכיות נמוכה ויעיל מאוד שהוא ה-Map.

מטרות/יעדים

מטרות המערכת:

המטרה העיקרית שהפרויקט דרש ממני היא רכישת ידע בתחום "הבינה המלאכותית", הכרה נרחבת

בשימוש בטכנולוגית Angular ולימוד מעמיק ויסודי בשפת C#.

- מערכת חוויתית ונוחה למשתמש תוך התאמה אישית ומדויקת לצרכיו ורצונותיו.
- חיסכון גדול ומשמעותי בזמן מחוץ לבית.

יעדי המערכת:

- שליפת רשימת הספרים שנמצאים במאגר והצגתם למשתמש.
- בניית פרופיל משתמש ע"י חישוב העדפות עפ"י היסטוריית חיפוש ושימוש של המשתמש.
- אתר מונגש ונח למשתמש.
- קבלת פרמטרים והצגת הנתונים עפ"י הפרמטרים שהתקבלו ע"י סינון המאגר הקיים.

אתגרים

כמו בכל דבר שמשקיעים בו ועובדים עליו - חייבים להיות אתגרים. במהלך הפרויקט עלו אתגרים רבים שבסופו של דבר נפתרו לאחר זמן של מחשבה ועבודה. העלאת תמונות הספרים - כחלק מהצגת הספרים למשתמש הצגתי גם תמונה של כל ספר, תחילה לא הצלחתי בשום דרך להציג את התמונות מהדטה בייס, לאחר שעות של ניסויים ועבודה רבה הבעיה נפתרה ולבסוף הצלחתי להציג את התמונות. שליפת נתונים - כחלק מהצגת נתוני הספרים הייתי צריכה להציג מידע מתוך הדטה בייס על כל ספר וזה פשוט לא עבד בצד של ה-angular, בלי שום סיבה! וכמובן אחרי ניסויים רבים - הם הוצגו.

מדדי הצלחה למערכת

לאחר כל כתיבת קוד אני מבצעת בדיקה אם אכן הקוד תאם למטרתי וביצע מה שצריך ובכך תוצאות המערכת עונות על מטרותיה ביעילות ובנוחות.

רקע תיאורטי/ ספרות מקצועית**רקע תיאורטי**

האלגוריתם המרכזי בפרויקט הוא פענוח פרופיל המשתמש והצגת ספרים המתאימים לו. האלגוריתם בנוי בצורה של "אלגוריתם מתלמד" כלומר: ככל שהמשתמש יכנס באתר למספר גדול יותר של ספרים כך ביצועי האלגוריתם ישתפרו ויגיעו להישגים מדויקים יותר.

האלגוריתם שומר את פרטי הספרים שנקלטו בהסטוריית הספרים או בספרים שאהב ואז מנסה למצוא מה משותף לכל הספרים בפרמטרים כגון: מחבר, סוג ספר, וכו' ובהתאם לזה מבצע סינון במאגר הספרים הקיים ומציג למשתמש את הספרים העונים לרצונותיו.

ספרות מקצועית**מהי בינה מלאכותית?**

אחד הנושאים הלוהטים ביותר כיום בתחום מדעי המחשב הוא בינה מלאכותית, שנקראת גם Artificial Intelligence או בקצרה – AI. סביר להניח שאין אחד שלא שמע עד היום על בינה מלאכותית, בין אם זה בחדשות בטלוויזיה, בעיתונים, ברחבי האינטרנט או אפילו בסרטי מדע בדיוני. מצד אחד, בינה מלאכותית נשמע כמו מושג מוכר וידוע אבל אם תיקחו רגע ותחשבו, תוכלו להגדיר בצורה מדויקת את המושג בינה מלאכותית?

נכון, מצד אחד מדובר בביטוי שהצליח להגיע אל כולנו, אין אחד שלא יצא לו לחשוב או לדבר על בינה מלאכותית בשלב כזה או אחר, או אפילו לחקור את הנושא אבל מה זה בעצם בינה מלאכותית? מה זה AI? ולמה בעצם כולם מדברים על זה ולמה דווקא עכשיו? האם בינה באמת יכולה להיות מלאכותית?

עד עכשיו, העלנו שאלות רבות ועל כולן ואפילו יותר אענה במאמר הזה. חשוב לציין כי המטרה של המאמר הזה הינה לתת רקע תאורטי בכל הקשור לבינה מלאכותית, למידת מכונות ולמידה עמוקה. במהלך המאמר אסביר בעיקר בצורה תאורטית על המושגים ולא אכנס למושגים טכניים או כתיבת קוד כלל.

בינה מלאכותית זהו מושג די כללי, הוא כולל בתוכו המון תחומים וגישות שונות ומגוונות שהתחילו להתפתח תאמינו או לא כבר בסביבות שנות ה-50. בינה מלאכותית זהו מושג המתייחס לכל שיטה או דרך הגורמות למחשב או למכונה להציג רמה כלשהי של חכמה.

במילים אחרות, בינה מלאכותית היא מעטפת הכוללת בתוכה שורה ארוכה של כלים, שיטות ודרכים הנועדו להעניק למחשב יכולות הדומות לאלו של בני האדם, למשל: זיהוי פנים. בינה מלאכותית כפי שאמרנו, התחילה כבר בשנות החמישים ובשנותיה המוקדמות כללה בעיקר אלגוריתמים שנבנו לפי מאגרי נתונים מסוימים, כלומר בכל שינוי של מידע נדרש שינוי של המערכת, כביכול "לימוד מחדש". בעצם, יכולת מערכת הבינה מלאכותית היא לקבל החלטות באופן עצמאי על בסיס עיבוד של מידע ונתונים ללא עירוב אדם בתהליך.

תיאור מצב קיים

אני אכן מודעת לכך שקיימים אתרי ספריות למיניהם, אומנם אתרים אלו נותנים מידע על ספרים ולא מציעים למשתמש ספרים המתאימים לו, הם אינם בעלי למידה על כל משתמש ורצונותיו. והאלגוריתם שלי עוסק בדיוק בדבר הנצרך הזה של הלמידה העצמית עבור המשתמש.

ניתוח חלופות מערכת

אלגוריתם מתלמד בעל מבנה נתונים של רשימות הינו אלגוריתם שיכול היה לשמור הרבה נתונים אך הוא אינו אלגוריתם יעיל בכלל. אלגוריתם מתלמד שמכיל מיפוי ג'נרי הינו אלגוריתם שיכול להכיל הרבה נתונים בצורה מסודרת ומאד יעילה.

תיאור החלופה הנבחרת

בחרתי לבצע את האתר באמצעות מפה.

מתוך נתוני ההיסטוריה של המשתמש אני יוצרת רשימה המורכבת ממספרי הזיהוי של הספרים שניצפו בשלושה חודשים האחרונים. כשהמשתמש נכנס לאתר מתבצעת פעולה ששולפת את כל הספרים מהמאגר הכללי. ומסננת רק את אלו שהפרמטרים שלהם מופיעים במפה לדוג': ספר מסוג מתח ייבדק רק אם קיים בפרמטר: סוג ספר את קוד הסוג: מתח וכן הלאה. לאחר מכן אעבור על הרשימה ואנסה ליצור מבנה מצומצם יותר המורכב לפי רמת האחוזים של כל תת פרמטר. לדוג': גם ספר שכל התכונות שלו מופיעות במילון יכול להיות שלא יוצגו למשתמש כי כמות הספרים בסוג זה של הספר יותר נמוך ולכן נציג פחות ממנו ויותר מספרים אשר הפרמטרים שלהם יותר גבוהים באחוזים.

אפיון המערכת שהוגדרה ניתוח דרישות המערכת

סביבת פיתוח:

חומרה: מעבד i5 RAM 8GB

עמדת פיתוח: מחשב Intel

מערכת ההפעלה: Windows 10

שפות תוכנה: C# , תוך שימוש בטכנולוגיית WebApi , אנגולר.

כלי תוכנה לפיתוח המערכת: Microsoft Visual Studio 2017.

מסד נתונים: SQL Server.

עמדת משתמש מינימלית:

- חומרה: מעבד i5 RAM 4GB.
- מערכת ההפעלה: Windows 7 ומעלה.
- חיבור לרשת: נדרש.
- תוכנות: Internet Explorer 7 or higher or chrome.

מודול המערכת

נושאים באחריות המערכת:

האתר עוסק בהצגת ספרים למשתמש לפי נתוני סינון שמכניס.

האתר יאפשר למשתמש שמחובר למערכת לצפות בספרים שאהב או בהצעות ספרים עבורו.

האתר יאפשר למשתמש לא מחובר לצפות במאגר הספרים.

נושאים שאינם באחריות המערכת:

האתר לא יאפשר למשתמש לא מחובר לצפות בספרים שאהב או בהצעות ספרים עבורו.

האתר לא מכיל ספרי לימוד.

האתר לא יתעסק בהשאלות ובהחזרות

האתר לא יתעסק בהוספת ספרים ומחיקתם

אפיון פונקציונלי

האתר יציע למשתמש את הספרים המתאימים לו עפ"י הפרופיל שלו, החיפושים שלו בעבר והסינונים שביצע.

ההתאמה תתבצע בצורה אוטומטית באופן שתוצאה את כל משתמשי האתר עפ"י צרכיהם.

ביצועים עיקריים

1. קליטת פרופיל המשתמש(בעת ההרשמה לאתר המשתמש החדש נשאל כמה שאלות שעזרות לבנות פרופיל זמני), קליטת חיפוש המשתמש וקליטת נתונים שסימן כ-"אהב" ומתוך כל זה האתר מבצע התאמה של ספרים עבור המשתמש.
2. מיפוי כל הספרים לפי מפתחות וערכים וע"י כך בחירת הספרים עם הערכים הכי גבוהים להצעה למשתמש.

אילוצים

האתר צריך להתחשב באילוצי המשתמש הכוללים: הכנסת פרופיל מסוים ומנגד- חיפושים שונים שלא כפרופיל שהכניס.
בנוסף, התחשבות בספרים שעייין בהם מלפני זמן, שכל הנראה הוא כבר לא מתעניין בהם.

תיאור הארכיטקטורה

תיאור צד השרת:

צד השרת מחולק כמקובל לשכבות:
שכבת ה-DAL
שכבת ה-Common
שכבת ה-BL
החלוקה לשכבות נועדה להפריד באופן מוחלט בין הלוגיקה של הפרויקט לבין הנתונים עצמם.
הפרדה זו מאפשרת לבצע שינויים בכל אחת מהשכבות בלי תלות ובלי זעזועים בשכבות האחרות.

פירוט:

DAL: שכבת ה-DAL היא השכבה דרכה ניגשים לנתונים היושבים ב-DB. היא מכילה מחלקות המייצגות את בסיס הנתונים וכן פונקציות נחוצות לתפעול. פעולות ההתקשרות עם בסיס הנתונים נעשו בטכנולוגיית Entity Framework. טכנולוגיה זו היא המקובלת לטיפול בבסיס הנתונים כאשר היא מנתחת את בסיס הנתונים, בונה מחלקות לייצוג הטבלאות ומספקת רשימות מלאות בנתוני בסיס הנתונים.

BL: שכבה שבה כתובה כל הלוגיקה של הפרויקט.

Common: שיתוף. שכבה זו מכילה מחלקות המתארות את הנתונים ובמבנה זה מעבירים את הנתונים בין השכבות. מטרת שכבה זו היא למנוע תלות של שכבת ה- BL במבנה בסיס הנתונים. שכבת ה- DAL מכילה פונקציות המרה מטיפוס הנתונים של בסיס הנתונים לטיפוס הנתונים של ה- Common ולהיפך, וכך מיוצגים הנתונים בכל הפרויקט. בנוסף קיימות מתודות השרת המציינות את הפעולות שניתן לבצע בשרת. מתודות אלו משתמשות ב- BL ומופעלות ע"י שכבת ה- GUI בצד הלקוח. מודל זה אמנם גורם טרחה טכנית לא מעטה בכתיבת הקוד ובתכנון הפונקציות אבל מספק קוד נקי, קל להבנה ונוח לשינויים ולשדרוגים. בפרויקט הושקעה עבודה רבה בכתיבה נכונה ויעילה כמקובל בעולם התכנות, דבר שאמנם דרש זמן רב למדי אך הקנה לנו ניסיון אמיתי מעין כמוהו.

תיאור הרכיבים בפיתרון

הפרויקט פותח בסביבת עבודה .NET.

צד השרת נכתב בטכנולוגיית Web Api ובשפת C#.

צד הלקוח נכתב בשפות Html5, CSS, TypeScript בטכנולוגיית Angular.

שילוב זה מאפשר בניית תוכנה של שרת - לקוח בצורה מובנית וברורה.

התוכנה נעזרת בפונקציות 'צד לקוח' רבות ומורכבות לצורך יצירת חווית משתמש גבוהה, הן מצד מהירות השירות, והן מצד נוחות השימוש.

באופן כללי הושקעה בתוכנה מחשבה רבה, הן לצורך יצירת תוכנה מתקדמת ונאה לשימוש והן לצורך תפעול פרקטי, נוח ופשוט.

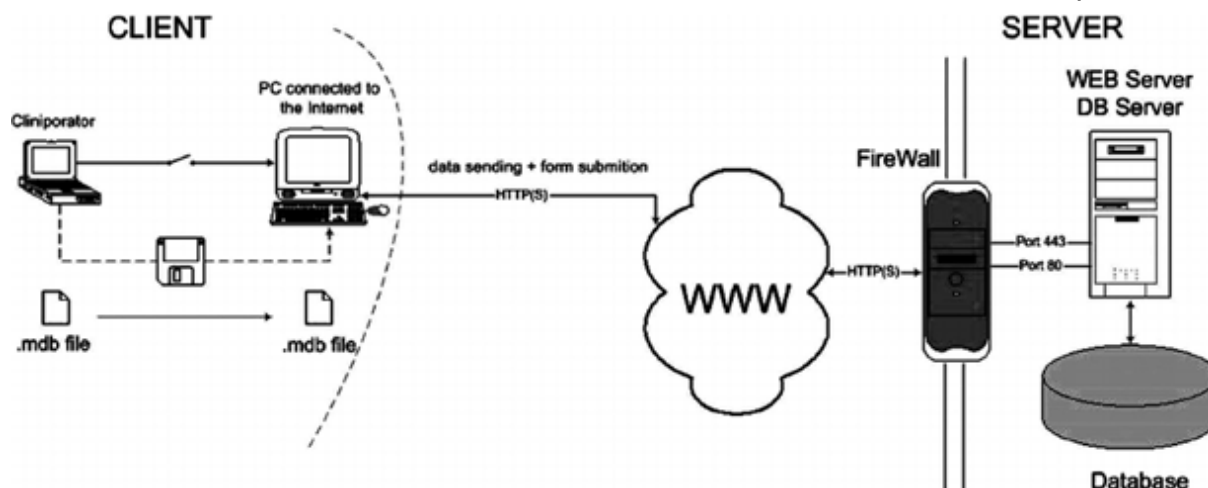
פונקציות צד השרת נכתבו תוך חלוקה נכונה ומסודרת של הקוד.

תיאור פרוטוקולי התקשורת

המערכת מורכבת משרת IIS המריץ את האתר בסביבת ה- server.

מסד נתונים - database של Sql-server, ממשק משתמש בצד הלקוח: דפדפן אינטרנט כלשהו:

internet explorer, chrome, firefox



שרת-לקוח

בניית התוכנה נעשתה תוך שימוש בהפרדת רשויות מלאה ובניית היררכיה ברורה, על מנת למנוע שיבושים בנתונים ובצורת התצוגה. מה גם שרצינו לאפשר שינוי תכנותי בעתיד, הן בצורת הגישה לנתונים, והן בצורת התצוגה ללא הצורך במגע עם רשויות נוספות. המערכת מתחלקת לשני תחומים: הראשון- ממשק המשתמש הבנוי בצורת אתר המתקשר לשרת לצורך ניהול האתר תוך שליפת נתונים ממסד הנתונים והצגתם למשתמש בשכבת ה-GUI. והשני שרת המבצע את הפעולות הנדרשות תוך התקשרות למסד הנתונים.

האתר מחולק למספר שכבות:

שכבת מסד הנתונים (SQL)

שכבת הישויות (Entity Framework)

הגישה לבסיס הנתונים (DAL)

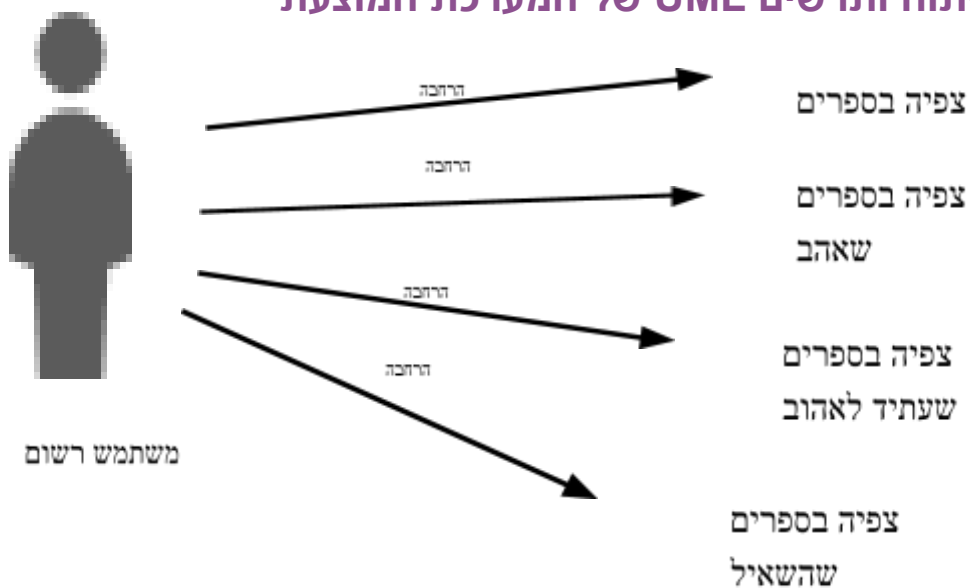
שכבת הקוד (BL)

שכבת הקוד נעזרת בספריית ה- Common שמכילה את מבנה הנתונים כדי לתקשר עם שכבת ה-

DAL וה- WebApi.

שכבת הקישור (Web Api)

ניתוח ותרשים UML של המערכת המוצעת



פירוט משתמש:

שם ה- use case	צפיה בספרים
תיאור קצר	המשתמש יכול לצפות במאגר הספרים בספריה
תנאי קודם	קיימים ספרים
שחקנים מעורבים	משתמש ומערכת
הזנק	נכנס לאתר לעמוד "הספרים שלנו"
תנאי סיום	סגירת האתר / מעבר לעמוד אחר
תדירות	כל רגע נתון

שם ה- use case	צפיה בספרים שאהב
תיאור קצר	המשתמש יכול לצפות בספרים שלחץ עליהם "אהבתי"
תנאי קודם	קיימים ספרים ברשימת הספרים שאהב
שחקנים מעורבים	משתמש ומערכת
הזנק	לחץ "אהבתי" על הספר
תנאי סיום	סגירת האתר / מעבר לעמוד אחר
תדירות	כל רגע נתון

שם ה- use case	צפיה בספרים שהשאל
תיאור קצר	המשתמש יכול לצפות בספרים שהשאל
תנאי קודם	קיימים ספרים ברשימת הספרים שהשאל
שחקנים מעורבים	משתמש ומערכת
הזנק	לחץ על עמוד "השאלות והחזרות"
תנאי סיום	סגירת האתר / מעבר לעמוד אחר
תדירות	כל רגע נתון

שם ה- use case	צפיה בספרים שאולי יאהב
תיאור קצר	המשתמש יכול לצפות בספרים שלחץ עליהם "אהבתי" / לחץ על "מידע"
תנאי קודם	קיימים ספרים ברשימת הספרים שאולי יאהב
שחקנים מעורבים	משתמש ומערכת
הזנק	לחץ "אהבתי" / על "מידע" בספר
תנאי סיום	סגירת האתר / מעבר לעמוד אחר
תדירות	כל רגע נתון



משתמש שאינו רשום

צפיה בספרים

הרחבה

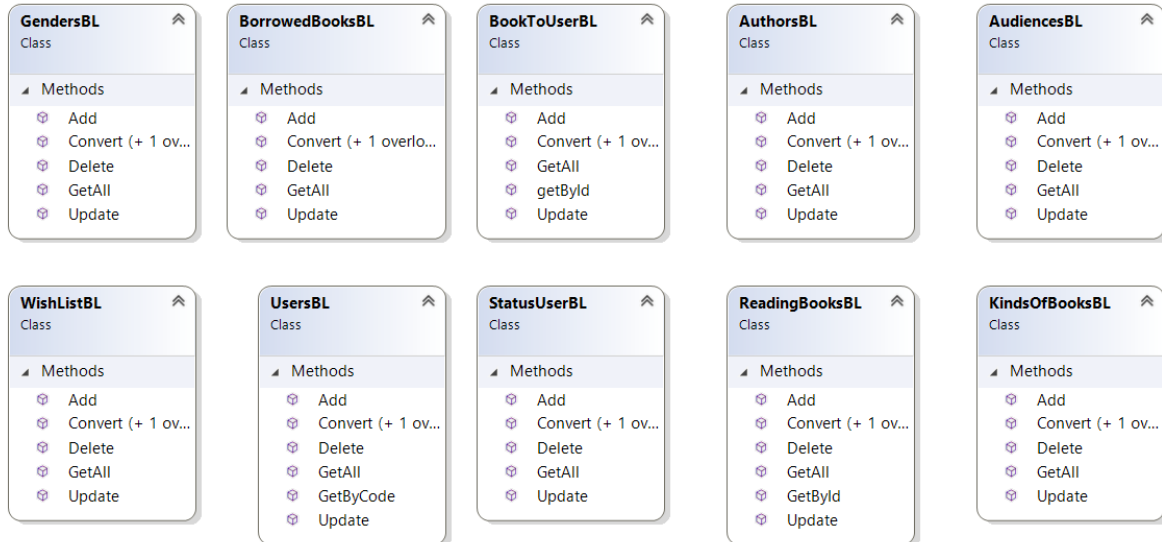
שם ה- use case	צפיה בספרים
תיאור קצר	המשתמש יכול לצפות במאגר הספרים בספריה
תנאי קודם	קיימים ספרים
שחקנים מעורבים	משתמש ומערכת
הזנק	נכנס לאתר לעמוד "הספרים שלנו"
תנאי סיום	סגירת האתר / מעבר לעמוד אחר
תדירות	כל רגע נתון

מבנה נתונים בהם השתמשתי

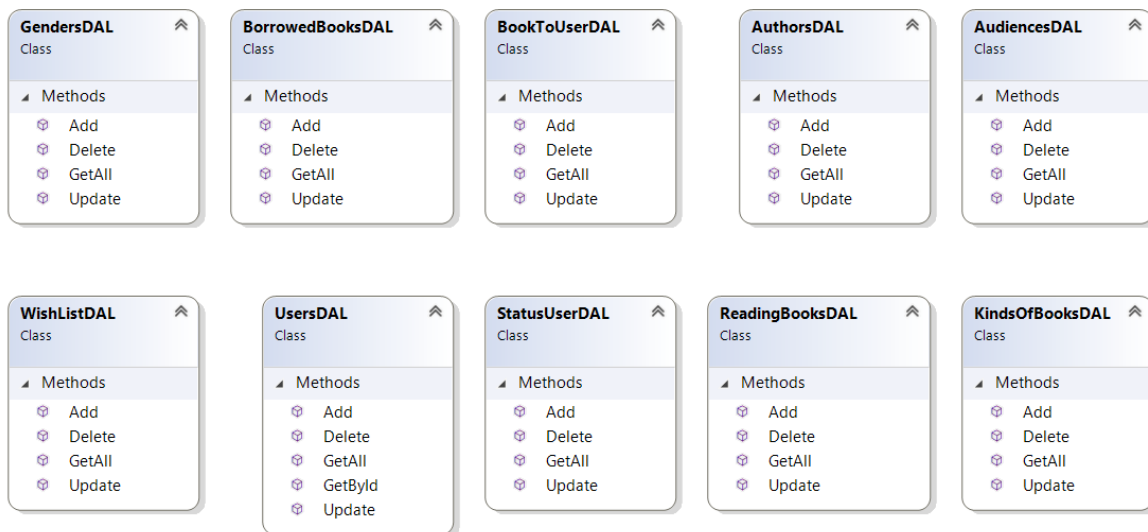
מבנה הנתונים בו השתמשתי הוא Map. שהוא הנכון לשמירת הנתונים שלי בעבור האלגוריתם בגלל שקיימות לי הרבה שליפות והחזרות של נתונים והוא בעל סיבוכיות יעילה.

תרשים מחלקות

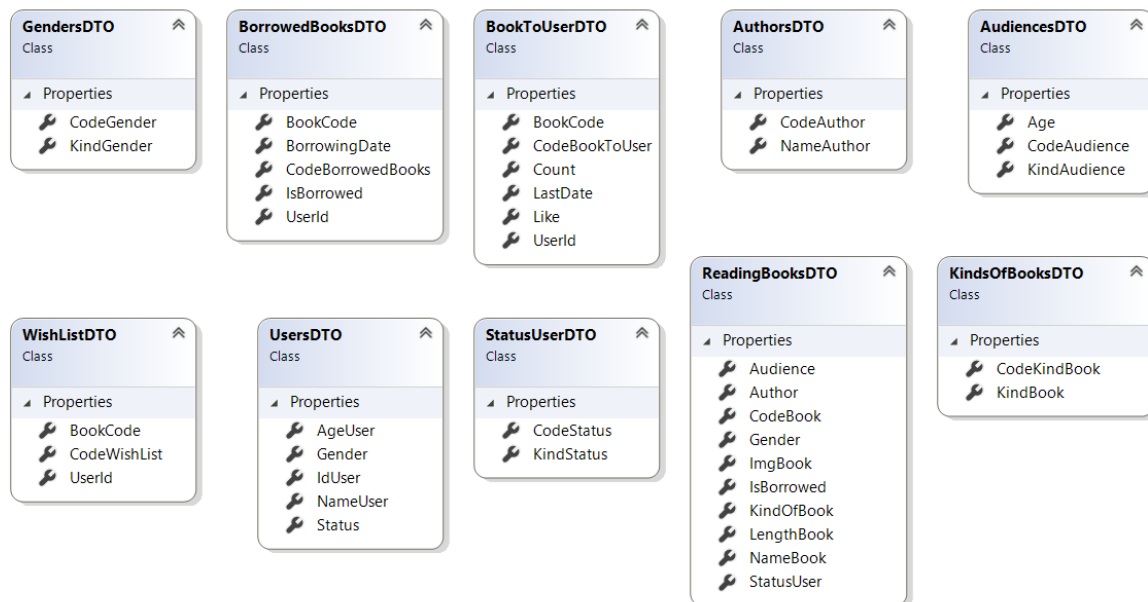
BL:



DAL:



DTO:



תיאור המחלקות המוצעות :BL

מחלקות אלו מקשרות בין בקשות בצד הלקוח לבין מסד הנתונים, עם כל השליפות שנמצאות ב-DAL. כל אחת מן המחלקות האלו מממשת ממשק שבו כתובות כל פעולות המחלקה, כשמגיעה הקריאה מהקונטרולר, הקריאה פונה לפעולה מהממשק ואז פונה למחלקה לצורך ביצוע הפעולה, הפעולות בשכבה זו פונות לפעולות שבשכבת ה-DAL וממירות את הערך המוחזר לערך חדש שיכול לחזור ללקוח.

:DAL

מחלקות אלו מתקשרות עם מסד הנתונים לצורך שליפה, עדכון, הוספה, מחיקה. מגיעות בקשות מה-BL שהיא מקבלת את הבקשה מהלקוח. כל אחת מן המחלקות האלו מממשת ממשק שבו כתובות כל פעולות המחלקה, כשמגיעה קריאה מה-BL, הקריאה פונה לפעולה מהממשק ואז פונה למחלקה לצורך ביצוע הפעולה, הפעולות במחלקה מבצעות שליפות, עדכונים וכו' במסד הנתונים, האובייקטים המוחזרים חוזרים כאובייקטים שמוכרים רק בקוד ולכן שכבת ה-BL מבצעת את ההמרה לאובייקט שהלקוח מכיר.

:Data Object

מחלקות אלו מיועדות לתצוגה, אלו המחלקות שהלקוח מכיר, ואליהם מתבצעת ההמרה בשכבת ה-BL.

רכיבי ממשק

בדפדפנים שונים כגון: Chrome, Microsoft Edge

תיכון המערכת ארכיטקטורת המערכת

.Web-API

ארכיטקטורת שרת-לקוח שבה קיימות 3 שכבות שדרך עוברים הנתונים בצורה מסודרת וממוסדת ומאפשרת סדר פעילות קבוע להעברת נתונים בין הלקוח והשרת.

תיכון מפורט

צד שרת: #C.

צד לקוח: Angular.

מסד נתונים: SQL.

חלופות לתיכון המערכת

קיימות חלופות רבות, בתוכן: Python, C, Java וכו'..

תיאור התוכנה סביבת עבודה

הפרויקט פותח בסביבת עבודה - .NET.

שפות תכנות

צד השרת נכתב בטכנולוגיית Web Api - בטכנולוגיית C#. צד הלקוח נכתב בשפות TypeScript, CSS, HTML5 בטכנולוגיית Angular.

תיאור מסכים

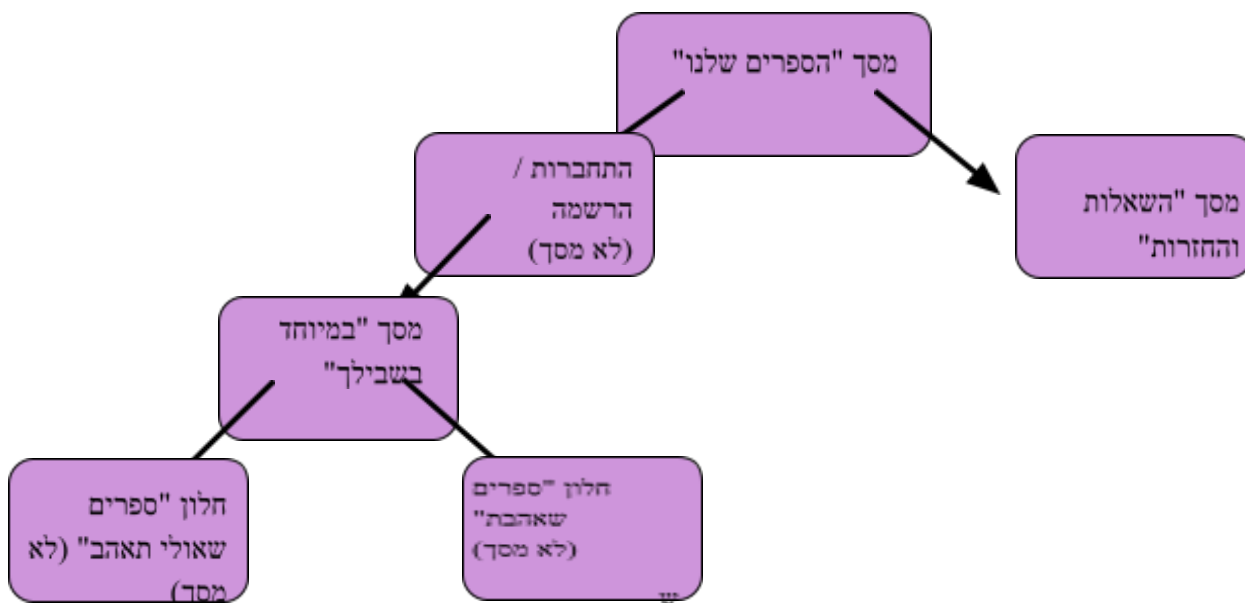
מסך התחברות והרשמה.

מסך השאלות והחזרות- המשתמש יכול לצפות בספרים שהשאל ולציין לעצמו מתי השאל ומתי החזיר ולהוסיף את הספר לרשימת הספרים שאהב.

מסך במיוחד בשבילך- המשתמש יכול לצפות ברשימת ספרים שאהב וברשימת ספרים שאולי הוא יאהב.

מסך הספרים שלנו- המשתמש יכול לצפות בספרים שבמאגר ולחפש ספר ע"י סינון לפי קטגוריות.

תרשים מסכים



פירוט המסכים

הרשמה:
הרשמה לאתר - משתמש המעוניין בהתאמת ספרים עבורו חייב בפעם הראשונה להרשם לאתר -למלא פרטים.

Smart Book

הספרים שלנו

במיוחד בשבילך

השאלות והחזרות

הרשמה/התחברות

חפש מחב.

חפש ספר.

סוג ספר

קודש

בישול

מתח

השקפה

קומיקס

מדע

מצוירים

קהל יעד

התחבר הרשם

שם משתמש

תהילה בוסקילה

סיסמא

גיל

20

מגדר

זכר

נקבה

סטטוס

נשוי/אה

רווק/ה

הרשם

כבר חבר?

התחברות:
אחרי שכבר נרשם למערכת. המשתמש יכול בכל פעם שנכנס להתחבר עם"נ שהמערכת תזהה אותו וכך תציע לו על פי רצונותיו.
לאחר התחברותו יוצג במסך "במיוחד בשבילך" את שמו של המשתמש.

התחבר הרשם

שם משתמש

תהילה בוסקילה

סיסמא

השאר אותי מחובר

התחבר

שכחת את הסיסמא?

חפש מחב.

חפש ספר.

סוג ספר

קודש

בישול

מתח

השקפה

קומיקס

מדע

מצוירים

קהל יעד

מסך הבית:
מראה האתר כולל ניווט . וכן סינונים עפ"י קטגוריה וחיפושים.

ספרים בספריה

חפש מחב.

חפש ספר.

סוג ספר

קודש

בישול

מתח

השקפה

קומיקס

מדע

מצוירים

קהל יעד





מסך במיוחד בשבילך:

במסך זה המשתמש יכול להיכנס אך ורק אם התחבר למערכת. בו מוצגים לו הספרים שאהב והספרים שרולי הוא יאהב.



יצאה

השאלות והחזרות

במיוחד בשבילך

הספרים שלנו

Smart Book

שלום, תהילה בוסקילה



ספרים שאהבת



ספרים שאולי תאהב

מסך השאלות והחזרות:
באפשרותו ללחוץ על השאלת ספר ופה תוצג לו רשימה של הספרים שהשאל.



יצאה

השאלות והחזרות

במיוחד בשבילך

הספרים שלנו

Smart Book

כאן תוכל לצפות בספרים שהשאלת

שם ספר	תאריך השאלה	תאריך החזרה	אהבתי
ילקוט יוסף	2021-01-20T00:00:00	21/01/2021	<input type="checkbox"/>
ארגז הכלים	2020-01-15T00:00:00	21/01/2021	<input type="checkbox"/>
צוואה	2020-07-10T00:00:00	21/01/2021	<input type="checkbox"/>
המצולע	2021-01-22T00:00:00	21/01/2021	<input type="checkbox"/>
מאין יבוא	2020-12-03T00:00:00	21/01/2021	<input type="checkbox"/>
ומתוך האור	2020-11-04T00:00:00	21/01/2021	<input type="checkbox"/>
שתול	2020-10-05T00:00:00	21/01/2021	<input type="checkbox"/>
חבורת תרי"ג	2020-12-22T00:00:00	21/01/2021	<input type="checkbox"/>
חרב מרגליות	2021-01-02T00:00:00	21/01/2021	<input type="checkbox"/>

קוד התוכנית+תיעוד

קלט:

1. התוכנית קולטת מהמשתמש פרטים אישיים עמ"נ שתוכל להתאים את עצמה להיסטוריית החיפוש של המשתמש. ולכן האתר יציע למשתמש דברים רק אם יבצע כניסה-התחברות.
2. האתר קולט מהמשתמש פרמטרים לסינון הספרים באתר לדוג': סוג ספר, קהל יעד וכו'.
3. קלט זה הינו קלט נסתר כי המערכת (בלי שהמשתמש יודע) אוספת עבורו את היסטוריית החיפושים שלו באתר ושומרת אותם במסד הנתונים על שמו.
4. האתר קולט מהמשתמש ספרים שאהב וספרים שאולי יאהב(ע"י לחיצה על אהבתי) ושומר אותם במסד הנתונים. כמו כן האתר קולט מהמשתמש ספרים שאולי יאהב (ע"י לחיצה על מידע) ושומר אותם במסד הנתונים.
5. האתר קולט מהמשתמש ספר להשאלה (ע"י לחיצה על "השאלה" בתוך חלון "ספרים שאהבת" ושם את הספר בתוך מסך "השאלות והחזרות").

פלט:

1. האתר מציג את שם המשתמש שמתחבר לאתר. כמו כן האתר יציע למשתמש ספרים שמתאימים לו לפי היסטוריית החיפושים.
2. האתר יציג למשתמש את הספרים העומדים בפרמטרים שסימן.
3. הפלט שכאן הוא בעצם קלט שנשמר כתוצאה מסעיף 2. ואז האתר יציג למשתמש את היסטוריית החיפושים שלו.
4. האתר יציג למשתמש בעמוד "במיוחד בשבילך" בתוך החלון של "ספרים שאהבת" ספרים שלחץ עליהם- אהבתי. ובתוך החלון של "ספרים שאולי תאהב" יהיו ספרים שלחץ עליהם- מידע.
5. האתר יציג למשתמש בתוך מסך "השאלות והחזרות" את הספרים שלחץ עליהם "השאלה".

פונקציות: צד שרת:

בשכבת הקונטרולר מוחזר מילון אשר מורכב ממפתח וערך עבור המפתח.

מפתח: שם של פרמטר לדוג': מחבר, סוג ספר וכו'.

ערך: מילון נוסף:

מפתח: קוד של פרמטר לדוג': קוד 1 שהוא סוג ספר- מתח. קוד 2 שהוא סוג ספר- קודש וכו'...

ערך: רמת החוזק של הופעת מפתח זה. לדוג': עבור 1 הערך -10.

דוגמא:

מתוך נתוני ההיסטוריה עולה שבמפתח סוג ספר נמצאים מפתחות בשם 8,9,10

מפתח 8 (קודש) ערכו הוא 5.

מפתח 9 (בישול) ערכו הוא 10.

מפתח 10 (מתח) ערכו הוא 15.

כלומר : המשתמש נכנס לספר מסוג קודש – 5 פעמים.

מסוג בישול-10 פעמים.

מסוג מתח-15 פעמים.

המערכת תציג יותר ספרים מסוג בישול ומתח מתוך הנחה שאלו הספרים שבהם מתעניין המשתמש.

Pseudo code

Controller:

GetAll()

Return *dictionary of book to user;

*dictionary key:parameter, value: dictionary key: code kind of parameter, value:

strength level

בשכבת הBL

יוצרים רשימה של כל היסטוריית השימוש המורכבת ממספרי הזיהוי של הספרים שנצפו בשלושה חודשים האחרונים.

כל ספר שהופיע בספרים שאהב הוספתי חוזק של 1 כדי להדגיש יותר את הפרמטרים של ספר זה.

עבור כל ספר הוספתי למילון תוספת של כמות הפעמים שספר זה נצפה ע"י המשתמש בתוך הערך של הפרמטר. לדוג': עבור ספר מסוג מתח שנפתח 2 פעמים, במילון מפתח סוגי הספרים – סוג מתח- יכנס הערך 2.

לאחר מכן עברתי על המילון המתקבל ומחקתי את תתי הפרמטר (הסוג הספציפי של הפרמטר) שחזרו ע"ע מעט מאד פעמים, וכן הגדלתי את ערכי תתי הפרמטרים כדי ליצור 100% מכל פרמטר. לדוג':

סוג ספר- מתח שהיה לו 5 <-50%.

סוג ספר-בישול שהיה לו 3 <-30%.

Pseudo code

BookToUserBL:

GetAll()

Retrieve from dal the book to user list and filter item from the last 3 months

Bild dictionary of the book to user

Put on all the keys value of parameter

Loop over the book to user list

If this readingBook was in the "wish list"

Add 1 to count

Loop on the readingBook

If the attribute==parameter

If the dictionary has a parameter with same code of parameter at the attribute

Add the count to the value of this key

Else

Add code kind of parameter with value=count

Loop on the dictionary

Find the biggest value of this parameter

Delete from the dictionary the keys where the value is very low than the biggest value

Convert to percent out of the 100% of the values in the parameter
Return the dictionary book to user

צד לקוח:

בעת לחיצה על קבלת מידע על הספר , נשלח ל-service מבנה המכיל את קוד הספר, קוד המשתמש שבאתר כרגע, תאריך והאם הספר סומן כאהבתי.

Pseudo code

On click on the information of the readingBook. Go to function:

Send to controller BookToUser the code of readingBook and the id of user and if the readingBook in the wish list and the date today.

בעת כניסה לאתר מופעלת פעולה ששולפת את כל הספרים מהמאגר.
מסנן רק את אלו שהפרמטרים שלהם מופיעים במפה, לדוג': ספר מסוג מתח ייבדק אם קיים בפרמטר -סוגי ספר קוד של סוג -מתח וכן הלאה על כל הפרמטרים.
לאחר מכן אעבור על הרשימה ואנסה ליצור מבנה מצומצם יותר המורכב לפי רמת האחוזים של כל תת פרמטר. לדוג': גם ספר שכל התכונות שלו מופיעות במילון יכול להיות שלא יוצגו למשתמש כי כמות הספרים בסוג זה של הספר יותר נמוך ולכן נציג פחות ממנו ויותר מספרים אשר הפרמטרים שלהם יותר גבוהים באחוזים.

Pseudo code

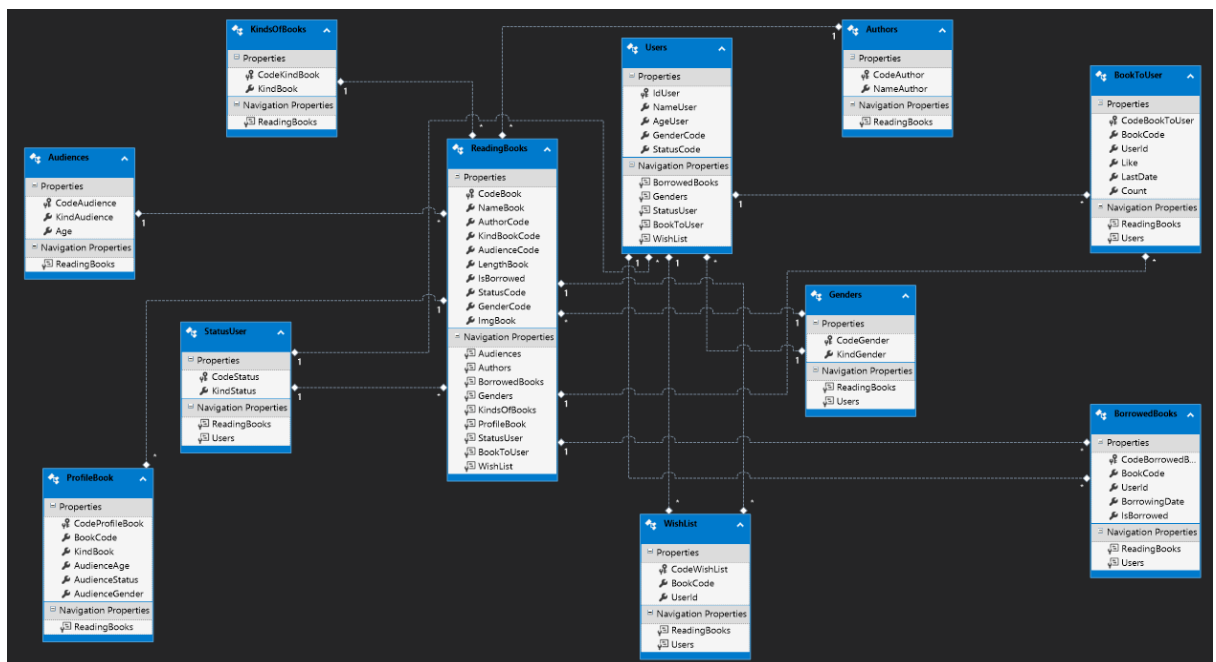
On the user signIn:

1. Retrieve from the service list of readingBook
2. Retrieve from the service map of bookToUser
3. Filter the list readingBook that have all the parameter from the map of bookToUser
4. Loop over the list readingBook
 1. If this code of parameter still exists
 1. Sub the percent
 2. Else
 2. Delete this readingBook from the readingBook list

פונקציות חשובות:

- סינון ספרים ע"י לחיצה על פרמטרים.
- חישוב ומיפוי לפי אחוזים של כל ערך במפה.

תיאור מסד הנתונים



טבלאות ה- Data Base - ספרי קריאה: ReadingBooks

שם	סוג	הסבר	חובה / לא חובה
CodeBook	int-primary key	קוד ספר 🔑	Not null
NameBook	varchar	שם ספר	Not null

Not null	קוד מחבר ספר	int-foreign key	AuthorCode
Not null	קוד סוג ספר	int-foreign key	KindBookCode
Not null	קוד קהל יעד	int-foreign key	AudienceCode
Not null	קוד סטטוס	int-foreign key	StatusCode
Not null	קוד מגדר	Int-foreign key	GenderCode
Not null	אורך ספר	int	LengthBook
Not null	האם מושאל	bit	IsBorrowed
Not null	תמונת הספר	Image	ImgBook

KindsOfBooks – סוגי ספרים:

שם	סוג	הסבר	חובה/ לא חובה
CodeKindBook	Int-primary key	קוד סוג ספר	Not null
KindBook	Varchar	סוג ספר	Not null


Users – משתמשים:

שם	סוג	הסבר	חובה / לא חובה
NameUser	Varchar	שם משתמש	Not null
IdUser	Varchar-primary key	ת.ז משתמש	Not null
AgeUser	Int	גיל משתמש	Not null
GenderCode	Int-foreign key	קוד מגדר	Not null
StatusCode	Int-foreing key	קוד סטטוס	Not null


Authors – מחברים:

שם	סוג	הסבר	חובה/לא חובה
NameAuthor	varchar	שם מחבר	Not null
CodeAuthor	int-primary key	קוד מחבר	Not null


Audiences – קהל יעד:

שם	סוג	הסבר	חובה/ לא חובה
CodeAudience	int-primary key	קוד קהל יעד 	Not null
KindAudience	varchar	סוג קהל יעד	Not null
Age	int	גיל קהל יעד	Not null




Genders – מגדרים:

שם	סוג	הסבר	חובה/ לא חובה
CodeGender	int-primary key	קוד מגדר 	Not null
KindGender	varchar	סוג מגדר	Not null

StatusUser – מעמד משתמש:




שם	סוג	הסבר	חובה/ לא חובה
CodeStatus	int-primary key	קוד מעמד 	Not null
KindStatus	varchar	סוג מעמד	Not null

BorrowedBooks – ספרים שהושאלו:

שם	סוג	הסבר	חובה/ לא חובה
CodeBorrowedBooks	int-primary key	קוד ספרים שהושאלו 	Not null
BookCode	int-foreign key	קוד ספר 	Not null
UserId	varchar-foreign key	קוד משתמש 	Not null
BorrowingDate	date	תאריך השאלה	Not null
IsBorrowed	bit	האם מושאל	Not null

BookToUser – הסטוריית החיפוש:

שם	סוג	הסבר	חובה/ לא חובה
----	-----	------	---------------

Not null	קוד הסטוריית משתמש 	int-primary key	CodeBookToUser
Not null	קוד ספר 	int-foreign key	BookCode
Not null	קוד משתמש 	varchar-foreign key	UserId
Not null	אהבתי/לא אהבתי	bit	Like
Not null	תאריך פתיחה אחרון	date	LastDate
Not null	מספר הפעמים באחוזים של הספר	int	Count

wishList – הספרים שאהב:

שם	סוג	הסבר	חובה/ לא חובה
CodeWishList	Int-primary key	קוד ספרים שאהב 	Not null
BookCode	Int-foreign key	קוד ספר 	Not null
UserId	Varchar-foreign key	קוד משתמש 	Not null

מדריך למשתמש

מסך הספרים שלנו:

המשתמש יכול לצפות בספרי האתר, לבצע סינון ספרים כפי שירצה.

מסך במיוחד בשבילך:

המשתמש יוכל להיכנס למסך זה ע"י התחברות למערכת. במסך זה יוכל לצפות בספרים שסימן כאהב, וכן לצפות בהצעות שהמערכת מציעה לו לפי נתוניו.

מסך השאלות והחזרות:

המשתמש יוכל לצפות בספרים שלחץ עליהם "השאלה". ושם יהיה כתוב את פרטי הספר וכו'..

חלון הרשמה/ התחברות:

עמ"נ להיכנס ל"במיוחד בשבילך" המשתמש חייב לבצע הרשמה או התחברות.

בדיקות והערכה

שליפת והצגת הספרים מהמאגר מתבצעת בהצלחה.

סינון לפי פרמטרים כפי הסימון של המשתמש מתבצע בהצלחה.

קליטת הנתונים על כל הספרים שהמשתמש צפה בהם או סימנם כאהב עובדים כראוי.

המערכת מציעה למשתמש ספרים לפי החיפושים שביצע, הצפיות במידע על הספרים ולפי ספרים שאהב.

ניתוח יעילות

בתחילה חשבתי לעבוד עם רשימות ואז גיליתי שמפה ומילון הרבה יותר יעילים, ולמרות הזמן הקצר שהיה לי, החלטתי ללמוד את הנושים האלו עמ"נ להשיג יעילות נמוכה יותר.
זמן ריצה של רשימה $O(n)$ ושל מפה $O(1)$

אבטחת מידע

על מנת שהמשתמש יוכל לצפות בחלון של "במיוחד בשבילך" כדי לקבל הצעות ספרים וכדי לצפות בספרים שאהב – הוא חייב להתחבר לאתר ע"י שם משתמש וסיסמא.
וכן בכל הכנסת נתונים ישנם בדיקות תקינות על מנת לבדוק שהנתונים שהוזנו מתאימים ונכונים.

מסקנות

ניתן לומר כי הפרויקט תרם לי רבות, היכולת לדעת לבצע עבודת מחקר באופן מעמיק, למצוא חומרים, לעיין, לחפש, ללמוד באופן עצמאי ולדעת ליישם- כמובן.
פעמים רבות עבודת מחקר ארוכה מסתכמת בשורת קוד קצרה, למדתי לתמצת מחקר רב למס' מועט של שורות קוד פשוט, קריא וקצר.
למדתי ורכשתי כלים רבים במהלך העבודה על הפרויקט: ידע נרחב בשפת C#, ידע נרחב בטכנולוגית Web Api ובספריות Angular ו- Bootstrap, מיומנות בבניית אלגוריתם, ניתוח יעילות ומבט רחב על מערכת ממוחשבת. יש לציין כי כתיבת ספר זה תרמה רבות לביסוס הידע שצברתי במהלך הפרויקט.
פרויקט זה אכן היווה צומת רחב ומשמעותי בהתמקצעות בתחום הנדסת תוכנה.

פיתוחים עתידיים

- מטרתי לשדרג את האתר ברמת הביצועים:
- אפשרות של שמע באתר-הנגשה.

- הקפצת הודעות למשתמש לאחר שבוע של השאלת ספר.

בבליוגרפיה

[/https://www.w3schools.com](https://www.w3schools.com)

[/https://freshdesignweb.com](https://freshdesignweb.com)

[/https://freefrontend.com](https://freefrontend.com)

[/https://material.angular.io](https://material.angular.io)

[/https://codepen.io](https://codepen.io)

[/https://www.codebrain.co.il](https://www.codebrain.co.il)

[/https://reshetech.co.il](https://reshetech.co.il)

<https://www.tutorialspoint.com/index.htm>

[/https://angular.io](https://angular.io)

[/https://stackoverflow.com](https://stackoverflow.com)