פרויקט גמר

מגישה: תהלה חדי טואיטו

ת.ז:212554117

סמינר: גברא

מנחה: המורה שושי טעפ והמורה יעל עמר

תאריך הגשה: יולי

סמל

תוכן

[1. הצעת פרויקט 3](#_Toc102417030)

[2. מבוא / תקציר 3](#_Toc102417031)

[2.1. הרקע לפרויקט 3](#_Toc102417032)

[2.2 תהליך המחקר 3](#_Toc102417033)

[2.3 סקירת ספרות 3](#_Toc102417034)

[3. מטרות ויעדים 4](#_Toc102417035)

[4. אתגרים 4](#_Toc102417036)

[5. מדדי הצלחה 4](#_Toc102417037)

[6. תיאור המצב הקיים 4](#_Toc102417038)

[7. רקע תאורטי 4](#_Toc102417039)

[8. ניתוח חלופות מערכתי 4](#_Toc102417040)

[9. תיאור החלופה הנבחרת והנימוקים לבחירה 4](#_Toc102417041)

[10. אפיון המעכת 4](#_Toc102417042)

[10.1. ניתוח דרישות המערכת 4](#_Toc102417043)

[10.2. מודול המערכת 4](#_Toc102417044)

[10.3. אפיון פונקציונאלי 4](#_Toc102417045)

[10.4. ביצועים עיקריים 4](#_Toc102417046)

[10.5. אילוצים 4](#_Toc102417047)

[11. תיאור הארכיטקטורה 4](#_Toc102417048)

[11.1. הארכיטקטורה של הפיתרון המוצע בפורמט של Design level Down-Top 4](#_Toc102417049)

[11.2. תיאור הרכיבים בפיתרון 4](#_Toc102417050)

[11.3. ארכיטקטורת רשת (לא רלוונטי ) 4](#_Toc102417051)

[11.4. תיאור פרוטוקולי התקשורת (HTTP) 4](#_Toc102417052)

[11.5. שרת – לקוח 4](#_Toc102417053)

[11.6. תיאור הצפנות (לא רלוונטי) 4](#_Toc102417054)

[12. ניתוח ותרשים use case של המערכת המוצעת 4](#_Toc102417055)

[12.1. רשימת use case 4](#_Toc102417056)

[12.2. תיאור ה-use case העיקריים של המערכת 4](#_Toc102417057)

[12.3. מבני נתונים בהם משתמשים בפרויקט 4](#_Toc102417058)

[12.4. תרשים מחלקות 4](#_Toc102417059)

[12.5. תיאור המחלקות 4](#_Toc102417060)

[13. תיאור התוכנה 4](#_Toc102417061)

[14. אלגוריתמים מרכזיים 4](#_Toc102417062)

[14.1. חלק מהאלגוריתם... 4](#_Toc102417063)

[14.2. חלק אחר מהאלגוריתם... 5](#_Toc102417064)

[14.3. החלק העיקרי באלגוריתם 5](#_Toc102417065)

[15. קוד האלגוריתם 5](#_Toc102417066)

[16. תיאור מסד הנתונים 5](#_Toc102417067)

[16.1. פירוט הטבלאות ב- Data Base 5](#_Toc102417068)

[17. מדריך למשתמש 5](#_Toc102417069)

[17.1. תיאור המסכים 5](#_Toc102417070)

[17.2. מדריך למשתמש 5](#_Toc102417071)

[17.3. צילומי מסכים 5](#_Toc102417072)

[18. בדיקות והערכה 5](#_Toc102417073)

[19. ניתוח יעילות 5](#_Toc102417074)

[20. אבטחת מידע 5](#_Toc102417075)

[21. מסקנות 5](#_Toc102417076)

[22. פיתוח עתידי 5](#_Toc102417077)

[23. ביבליוגרפיה 5](#_Toc102417078)

[24. מטרות ויעדים 5](#_Toc102417079)

# הצעת פרויקט

**סמל מוסד:**

**שם מכללה:**

**שם הסטודנט:** תהלה חדי טואיטו

**ת.ז. הסטודנט:**   
**שם הפרויקט:**

**תיאור הפרויקט:** בכניסה למערכת יהיה אפשרות לכל מנהל מוסד לימודים לפתוח חשבון ולהשתמש במערכת. המנהל יכנס למערכת על ידי סיסמא וימלא את רשימת התלמידים שהוא רוצה לשבץ בכיתות, הוא יבחר מרשימת עדיפויות מוכנה את החשיבות שהוא רוצה לתת לכל עדיפות מהרשימה על ידי בחירה מ0-10 בסרגל בחירה לדוגמא לדרג מ0-10 כמה חשיבות הוא נותן למעורבות חברתית, מקום מגורים, רמה לימודית ועוד, תהיה לו גם אפשרות להוסיף עדיפות משלו ולדרג את החשיבות שלה. תהייה כניסה למורה שכל מורה יוכל למלא פרטים על כל אחד מהתלמידים שלו בלבד לפי האילוצים שהמנהל החליט. ותהיה כניסה לתלמיד או להורה התלמיד שיוכל למלא מספר בקשות שהמנהל החליט לתת אפשרות לתלמידים/להורי התלמיד לבקש. לדוגמא בחירת 3 חברים שהוא רוצה שילמדו אתו בכיתה, עדיפות למחנך ועוד. המערכת תקבץ את כל הנתונים של המורה והתלמידים, האילוצים השונים ותחשב לפי דרוג עדיפויות המנהל את שיבוץ התלמידים בכיתות כך שיגיע לשיבוץ הטוב ביותר שקרוב לפתרון הסופי.

**הגדרת הבעיה האלגוריתמית:** בעיית שיבוץ תלמידים בכיתה בבית ספר לפי אילוצים שונים היא בעיה לא פתירה ורק האלגוריתם הגנטי שהוא משמש לפתירת בעיות מורכבות יוכל להגיע לפתרון הקרוב ביותר לפתרון האמתי .האלגוריתם הגנטי פותר בעיות מורכבות בשונה מהאלגוריתם ההונגרי שפותר בעיות שיבוץ קטנות יותר ולכן האלגוריתם הגנטי מתאים לפתרון בעיית השיבוץ תלמידים שיש בו מלא אילוצים שונים.

**רקע תאורטי בתחום הפרויקט**: שיבוץ תלמידים הוא עבודה מאוד קשה שנמצאת בהרבה בתי ספר מפני שהמורים צריכים לדווח על כל תלמיד מכיתתו פרטים מסוימים שהמנהל דורש, הוא צריך קודם לשאול את המנהל מה הוא רוצה שימלא על כל תלמיד ולפעמים בחטיבה התלמידים הולכים לבתי ספר שונים ולכל מנהל יש את הדרישות שלו שהם שונות ממקום לימודים אחר וזה לוקח לו הרבה זמן לכתוב את הפרטים של כל תלמיד לפי דרישות ההנהלה של בת ספרו ולשלוח לבתי ספר השונים

**תהליכים עיקריים בפרויקט:**

1. המערכת תקבל מהמנהל את רשימת התלמידים שצריך לשבץ ואת דרוג העדיפויות.
2. המערכת תקבל מהמורה את הפרטים על כל התלמידים שלו בלבד לפי עדיפויות המנהל.
3. המערכת תקבל מהתלמיד את הבקשות שהמנהל נותן אפשרות לתלמידים לבחור.
4. המערכת תקבץ את כל הנתונים ותפעיל את אלגוריתם השיבוץ שישבץ את התלמידים בכיתות בצורה הקרובה ביותר לפתרון הסופי.

**תיאור הטכנולוגיה:**

**צד שרת:**

שפת תכנות בצד השרת: C#

**צד לקוח:**

שפת תכנות בצד לקוח: angular

**מסד נתונים:** SQL

**פרוטוקלי תקשורת**: **.**

**לוחות זמנים:**

1. חקר המצב הקיים – ספטמבר
2. הגדרת הדרישות – ספטמבר
3. אפיון המערכת – אוקטובר
4. אפיון בסיס הנתונים – נובמבר
5. עיצוב המערכת – דצמבר
6. בנית התוכנה – ינואר, פברואר
7. בדיקות – מרץ
8. הכנת תיק פרויקט – אפריל
9. הטמעת המערכת – מאי
10. הגשת פרויקט סופי - מאי

**חתימת הסטודנט:** אפרת ביטון

**חתימת הרכז המגמה:**

**אישור משרד החינוך:**

# מבוא / תקציר

## הרקע לפרויקט

כשהתחלנו בכיתה לחשוב על רעיון לפרויקט והעלנו כל מיני נושאים, חיפשתי רעיון חדש שיוכל להועיל ולהפחית קושי ,מאמץ והשקעה לאנשים בעולם ואז עלה לי רעיון של שיבוץ תלמידים בכיתות א' שזאת עבודה קשה ומאומצת ולוקחת הרבה מחשבה ורציתי לייעל את משימת השיבוץ בכך שאפתח תוכנה קלה לשימוש שהיא תעשה במקום מנהלי בית הספר את העבודה הזאת ותקל מעליהם את המשימה הקשה והמעייפת הזאתי.

בחרתי את הרעיון הזה כי אני מכירה מקרוב מנהל בית ספר שכל שנה הם מסתבך עם השיבוץ לכיתות של תלמידי כיתה א' וזה עבודה קשה ומורכבת שלוקחת הרבה זמן, חשיבה, מאמץ והשקעה שגם אחריהם לא כולם מרוצים מהשיבוץ

אני חושבת שהרעיון שלי טוב לפרויקט כי אני יודעת ששיבוץ תלמידים בכיתות זה דבר מאוד מסובך וקשה ולוקח הרבה מאוד זמן ולכן חשבתי על הפרויקט הזה שיוכל לעזור למנהלים לשבץ בצורה הרבה יותר קלה ויעילה ובלי לשבת על זה הרבה זמן.

הפרויקט יכול להיות עם תועלת רבה בכך שהוא יעזור למנהלי בית הספר בשיבוץ תלמידי כיתות א' בקלות וביעלות ולהוריד את הקושי של השיבוץ שהיה עד היום

מומלץ לפתח את נושא שיבוץ תלמידים בכיתות כי עד היום לא ראיתי מנהלים שיש להם תוכנה קלה לשיבוץ תלמידים שיוכלו להשתמש בה בקלות וביעילות.

כיום המנהלים יושבים עם רשימת כל התלמידים שרוצים לשבץ ומנסים לשבץ בכל מיני אפשרויות לפי אילוצים שונים ומידע שהכניסו על כל תלמיד כמו מה ממוצע הציונים שלו מה מקום המגורים שלו מהי המעורבות החברתית שלו ועוד נתונים חשובים שלפיהם מנסים לשבץ ולחלק את כל התלמידים בכיתות כך שכל הכיתות יהיו בממוצע אותה רמה ולא יהיה כיתה אחת של חלשים, כיתה של חזקים כיתה של היפראקטיביים וכו' ובנוסף להתחשב בבקשת כל תלמיד עם מי הוא רוצה להיות בכיתה.

לאחר פיתוח הפרויקט משימת השיבוץ תהיה הרבה יותר קלה, מהירה ויעילה כי המנהל יצטרך רק להכניס את כל רשימת התלמידים ועל כל תלמיד ימלא מספר שאלות כמו ממוצע הציונים שלו מה מקום המגורים שלו מהי המעורבות החברתית שלו ובאיזה מוסד לימודים קודם למד כדי שנוכל לדעת מי למד אתו ולשים אותם ביחד כדי שיהיה להם קל יותר בהסתגלות למקום הלימודים החדש. ואז האלגוריתם עושה הרבה שיבוצים ועל כל שיבוץ נותן ציון לפי האילוצים וכך עד שמגיע לשיבוץ הטוב ביותר.

לאחר שבחרתי בנושא השיבוץ חיפשתי אלגוריתם טוב שאני יוכל להשתמש בו לפרויקט שלי ומצאתי את האלגוריתם הגנטי שמאוד יעיל לשיבוץ תלמידים בכיתות ששיבוץ זה בעיה מורכבת והאלגוריתם הגנטי משמש לפתירת בעיות מורכבות והוא נתן לי את הפתרון הכי טוב ויעיל.

בחרתי בשם לפרויקט?????????????????

אני מקווה שהפרויקט שלי יועיל ויעזור

## תהליך המחקר

הדבר הראשון שהתחלתי לעשות כשרציתי להתחיל את הפרויקט זה לחקור על נושא השיבוץ בכל מיני נושאים כדי שאני יבין טוב את נושאי השיבוץ לאחר מכן חקרתי על אלגוריתמים לשיבוץ

חיפשתי חומר לקריאה על נושא של שיבוץ תלמידים בכיתות ואז חיפשתי אלגוריתמיים לשיבוץ

לאחר הרבה חיפושים על אלגוריתמיים מצאתי את האלגוריתם הגנטי ואת האלגוריתם ההונגרי קראתי על שניהם והגעתי למסקנה שהאלגוריתם הגנטי הכי טוב לפרויקט שלי

מצאתי את האלגוריתם הגנטי שעובד בהשראת תורת האבולוציה של דרווין. האלגוריתם יוצר אוכלוסייה של N פריטים אקראיים. כל פרט באוכלוסייה מקבל הערכת התאמה -ציון לפי העמידה באילוצים של כל התלמידים. שבעזרת הציון בוחרים את הפרטים שמתאימים להיות הורים לדור הבא ההתרבות נעשית על ידי הצלבה של הגנים של שני פריטים, חלק מהצאצאים עוברים מוטציה כדי להוסיף מגוון. מוטציה פרושה החלפת ערך של תכונה באובייקט בצורה אקראית. התהליך הזה של הבחירה והתרבות מופעל שוב ושוב N פעמים עד שיש לנו אוכלוסייה של N פריטים שהיא מהווה את הדור הבא. התהליך מופעל על מספר דורות גדול עד לקבלת התוצאה הרצויה

בחרתי דווקא את האלגוריתם הגנטי שהכי מתאים לשיבוץ תלמידים בכיתות שהיא בעיה מורכבת שניתנת לפתירה בצורה הטובה ביותר על ידי האלגוריתם הגנטי.

## סקירת ספרות

באיזה אתרים נעזרת ל- (תיאור במשפטים)

נעזרתי בהרבה אתרי אינטרנט לקרא על אלגוריתם לשיבוץ

* אלגוריתם
* שפות, שרת, לקוח
* שפות:
* שרת- gituhb
* לקוח- angular io
* עיצוב: השתמשתי בספריה של bootstrap

# מטרות ויעדים

חלקי למטרות ויעדים.

מה הביא אותך לרצות לעשות את הפרויקט ( מטרות אנושיות, מטרות של ידע, מטרות של התנסות מקצועית)?

ת'אמת שכלום לא הביא אותי ועשיתי את זה בלית ברירה

רציתי לעשות את הפרויקט כדי לעזור לאנשים בעולם שיוכלו להשתמש בתוכנה קלה ויעילה שתוכל להקל ולהועיל וגם היה מטרה לדעת יותר על אלגוריתמיים ומטרות של התנסות מקצועית

מה מטרה המרכזית/מטרת העל?

מטרת העל שלי בעשיית הפרויקט היא מטרה של התנסות מקצועית, לפתח את החשיבה, לנתח את הפרויקט שלי בצורה טובה וברורה ומובנת, להתנסות בכתיבת תוכנה וכך אני ידע יותר והתמקצע.

מה המטרות הנוספות? (לתת מידע בנוחות, לאפשר גישה ל... ליצור עיצוב נח...לאפשר, להקל על..)המטרות הנוספות של הפרויקט זה לאפשר שימוש נוח בתוכנה קלה לשימוש, שמעוצבת יפה ונוח לעין ובמיוחד להקל על מנהלי בית הספר בשיבוץ תלמידי כיתות א' על ידי שימוש בתוכנה ויגיעו לתוצאות טובות ומרשימות

מה היעדים שאליהם שאפת להגיע? (המערכת תעשה, המשתמש יעשה, החולים יוכלו, הפרטים ישמרו)היעדים שלי שהתוכנה

# אתגרים

(מקומות שבהן הסתכבת, היית צריכה להשקיע מחשבה, להתייעץ, לשנות כיוון.)

במהלך הפרויקט נתקלתי באתגרים....  
במהלך עבודתי נהניתי לראות שהדברים אינם פשוטים כמו שנראו בתחילה. אפרט כמה נקודות....

# מדדי הצלחה

(האפליקציה שלי הצליחה אם...... 1, 2, 3  
השיבוץ אכן עובד / יש זיהוי של לפחות 70 אחוז מהמקרים / המחשב מזהה את האותיות / כניסה לבעלי הרשאה בלבד)

# תיאור המצב הקיים

כיום המצב הוא.....

לאחר בירור אצל אנשים שונים....

אחרי חיפוש של תוכנות המספקות את השירות הנ''ל גיליתי ש...

# רקע תאורטי

(חלק חשוב מאד!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!! פירוט של האלגוריתם)

האלגוריתם המרכזי בפרוייקט הוא...... האלגוריתם שמאפשר את ביצוע ה..... הוא  
האלגוריתם שבו השתמשתי לצורך ה..... הוא. אפרט את האלגוריתם העיקרי בפרויקט.

1. מה האלגוריתם (החלק הכי מורכב) העיקרי בפרויקט? (3 שורות)
2. תזכירי, מה הייתה הבעיה שהאלגוריתם פותר? (6 שורות)
3. איזה סוג של אלגוריתם הכי מתאים לפירוט הבעיה שתיארת?  
   אל תתנו שם של אלגוריתם אלא תסבירו איזה סוג של אלגוריתם צריך.

# ניתוח חלופות מערכתי

1. אפשרויות שונות לפתרון הבעיה (לדוגמא, שיבוץ – באופן ידני, לנסות את כל האפשרויות, להמציא אלגוריתם, לבחור אלגוריתם הונגרי, לבחור אלגוריתם גנטי)
2. אפשר לפתור את בעיית השיבוץ בכמה דרכים שונות: לשבץ באופן ידני, אפשר להמציא אלגוריתם, לבחור באלגוריתם הונגרי, לבחור אלגוריתם גנטי
3. מה בחרת? (משפט 1)
4. בחרתי באלגוריתם הגנטי שהוא הכי יעיל לפתירת בעיית שיבוץ תלמידים בכיתות.

# תיאור החלופה הנבחרת והנימוקים לבחירה

1. שוב, מה המערכת שפתחת?
2. מה עם הרעיונות האחרים שהצעת לגבי המערכת שלך?
3. לגבי האלגוריתם שנבחר. כיצד הוא עובד? (מהם הם השלבים שלו, החלקים שלו, ממה מורכב?) (תוכלי לתת דוגמאות והמחשה ע''י תמונות)

# אפיון המערכת

**סביבת פיתוח :**

חומרה: מעבד RAM 32GB i7 עמדת פיתוח: מחשב Lenovo מערכת ההפעל ה: 10 Windows

שפות תוכנה: #C , תוך שימוש בטכנולוגי ת WebApi, אנגולר .

כלי תוכנה לפיתוח המערכת: .Microsoft Visual Studio2019, vs code:

מסד נתונים:.SQL Server עמדת משתמש מינימאלי ת :

* + חומרה: מעבדRAM 4GB i5 .

o מערכת ההפעל ה: Windows 7ומעלה.

* + חיבור לרש ת: נדרש .
  + . chrome :תוכנות

## ניתוח דרישות המערכת

**דרישות בהן המערכת צריכה לעמוד:**

* + כתיבה בסטנדרטים מקצועיים.
  + מחשוב השרות ללקוח.
  + כתיבת הקוד בסיבוכיות היעילה ביותר.
  + ממשק נוח וידידותי למשתמש.
  + תגובה מהירה ככל שניתן למשתמש.

## מודול המערכת

* העלאת קובץ Excel המכיל את רשימת התלמידים לשיבוץ עם פרטים טכני ים עליהם.
* קריאת הקובץ שהועלה והזנת הנתונים בדאטה בייס, וכן הצגתם ללקוח.
* הזנת נתונים נוספים לתלמידים ע"י המשתמש.
* קריאה למערכת לביצוע שיבוץ.
* מציאת השיבוץ האופטימלי ככל שניתן ע"י האלגוריתם.
* הזנת פרטי השיבוץ בדאטה ביי ס
* הצגת פרטי השיבוץ ללקוח.
* שליחת מייל ללקוח עם השיבוץ הסופי

## אפיון פונקציונאלי

(פירוט פונקציות עיקריות ותפקידן)

MyFunc1 – הפונקציה מבצעת....

MyFunc2 – הפונקציה מסננת את...

## ביצועים עיקריים

המנהל יכול להוסיף....

המשתמש מכניס את פרטיו ומקבל רשימת....

למשתמש מוצגת מפה של....

למשתמש נשלחת הודעה עם....

## אילוצים

המערכת יכולה להכיל עד 300 חנויות

המערכת מתבססת על קובץ EXCELL לקבלת המידע על

המערכת פועלת ע''פ נתונםי ... ואינה יכולה לנתח את ה... בלעדיהם.

# תיאור הארכיטקטורה

## הארכיטקטורה של הפיתרון המוצע בפורמט של Design level Down-Top

צד השרת - server side פותח במודל 3 השכבות ומתחלק ל-4 פרויקטי ם

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.

API – שכבת ה Controller – חיבור בין צד השרת והלקוח.

BL – הלוגיקה של המערכת.

DAL – מכיל את הפונקציונאליות הנדרשת לכל התקשורת עם הData Base.

Models – מכילה מחלקות המתארות את הנתונים ובמבנה זה מעבירים את הנתונים בין השכבות.

מטרת שכבה זו היא למנוע תלות של שכבת הBL במבנה בסיס הנתונים. שכבת הBL מכילה פונקציות המרה מטיפוס הנתונים של בסיס הנתונים לטיפוס הנתונים של שכבת הModels ולהיפך, וכך מיוצגים הנתונים בכל הפרויקט.

## תיאור הרכיבים בפיתרון

הפרויקט מחולק ל-2 חלקים:

* + - צד שרת - הנכתב בשפת #C ובטכנולוגיית WebApi.
    - צד לקוח - נכתב בשפת Angular ובטכנולוגיית Html, TypeScript.

בחרתי לכתוב צד לקוח ב - אנגולר שהינה שפה מתקדמת ועדכנית בעלת מאפייניAngular8 חדשניים ופונקציונאלית ביות ר.

אנגולר הינה סביבת עבודה שפותחה על ידי גוגל. מאפשרת לפתח אפליקציותFramework אינטרנט בקלות ומהירות. במקור היא באה לתת מענה לבניי תApplications Page Single בצורה מושלמת ומהירה. מהיתרונות הבולטים והעיקריים של אנגולר אפשר למנות: חיסכון במשאבים, מהירות ביצוע, קוד קצר יותר, רוב העבודה מתבצעת בצד הלקוח ופחות בשרת ויכולת התמודדות טובה )סינון מהיר ופשוט לביצוע( של תוכן המתקבל מהשרת לפי מספר רב של פרמטרים.

צד שרת בחרתי לכתוב בC# .C# היא שפת תכנות עילית מרוב ת-פרדיגמות, מונחית עצמים בעיקרה המשלבת רעיונו ת כמו טיפוסיות חזקה, אימפרטיביות, הצהרתיו ת, פונקציונאליו ת פרוצדוראליות וגנריות .

C# היא שפה מעניינ ת, נוחה ומלאה פונקציונאליות למתכנת. שימוש בשפה זו נפוץ כיום , וכתוצאה מכך, ניתן היה למצוא בה קודים שונים שנדרשו לפיתו ח.

בנוסף ,בחרתי להשתמש ב - EntityFramework טכנולוגית עבודה מתקדמת של מיקרוסופט.

הEntityFramework מאפשר לטעון את הנתונים מהDB -ולעשות להם השמה בצורה ישירה ואוטומטית לתוך אובייקטים בקוד הממפים את מאגר הנתונים בצורה מידי ת.

.Sql Server שנכתב בשפת DataBaseקורא נתונים מ ה EntityFrameworkה

למסד הנתונים של ה-SQL Server יש כלים נרחבים לגיבוי כל המידע של המערכת ,כולל מערכת ההפעלה, חשבונות המשתמשים והרשאותיהם, הגדרות ההתקנים, תוכניות וכן של שאר הרכיבים המסופקים עם השרת ואובייקטי המשתמש.

*דוגמא לזרימת מידע במערכת*

שליפת כל התלמידי ם

ברצוננו לקבל את כל התלמידים של משתמש מסוים מ הDB ולכן יתבצעו השלבים הנ"ל:

* המשתמש יחפוץ לראות את כל התלמידים שלו, הוא ילחץ על כפתור מסוים בתצוגה) html( ובקשתו תפנה לTypeScript.
* .servicesאשר תפנה ל GetStudentByiD תתבצע קריאה לפונקציה- script Type ב .לשרת url תתבצע בקשת services -ב o
* השרת מקבל את הבקשה ומנווט ל Controller שנמצא בAPI.
* הController יזמן את הפונקציה GetStudentByiD שנמצאת studentBL. o BL מעוניין לקבל נתונים מהDB ולכן הוא פונה לDAL- דרך הframework Entity o ה-DAL שואב את הנתונים הרצויים ממסד הנתונים וכעת מתבצע שלב החזרה.
* ה DAL מחזיר את רשימת התלמידים לשכבת הBL בה מתבצעת פונקצית הסינון של הבאת התלמידים של משתמש מסוים.
* .BLמה controllerמחזירה את הנתונים ל GetStudentByiD הפונקציה
* .serviceהנתונים מוחזרים ל controller -מה
* מהservice חוזרת הרשימה לtypeScript. o הרשימה מוצגת בHTML.

*איור:*

DB

Sql Server

Dal

Entity Framework

Bl

L

ayer

Controllers

Web Api

Client

Models

1. מסד הנתונים הבנוי מטבלאות וקשרי גומלין ביניה ם.
2. שכבת הגישה לנתונים באמצעות Entity Framework.
3. שכבת הישויות.
4. שכבת ה - BL בה כתובים האלגוריתמים.
5. Web Api פרוטוקול התקשורת בי ן צד הלקוח וצד השר ת.
6. .angular, TypeScript צד לקו ח

## ארכיטקטורת רשת (לא רלוונטי )

## תיאור פרוטוקולי התקשורת

http – תמצאו משפט הסבר

## שרת – לקוח

צד השרת נכת ב בטכנולוגית WebApi ובשפת c#.

. angular בטכנולוגית -Html, css ,typescript צד הלקוח נכתב בשפות

## תיאור הצפנות (לא רלוונטי)

# ניתוח ותרשים use case של המערכת המוצעת

תרשים כפי הנלמד בשנה שעברה. (איש לכל סוג משתמש וחץ עם הפעולות שהוא מבצע)

## רשימת use case

רשימת הפעולות המבוצעות ע''י המשתמשים.  
(המשתמש מתחבר ע''י הקלדת שם משתמש וסיסמא,  
 המנהל צופה בהודעות שהתקבלו,  
המשתמש מקבל הודעה לתיבת המייל...)

## תיאור ה-use case העיקריים של המערכת

Identifier , Name, Description , Actors, Frequency , pre-condition ,   
תחפשו כיצד מנסחים תיאור של use case.

## מבני נתונים בהם משתמשים בפרויקט

(מילון / רשימה / תור / עץ / מחסנית / ערימה והסבר איפה ולמה השתמשת במבנה זה)

## תרשים מחלקות

לכל שכבה תיאור המחלקות שלה.

שכבת ה-DAL  
צילום של פירוט המחלקות:  


שכבת ה-BL ...... וכו'.

## תיאור המחלקות

לכל שכבה בפרויקט– הסבר מילולי על השכבה. רשימת כל שמות המחלקות ולכל מחלקה הסבר מילולי. (לפי הסדר – DAL, BL, DTO, WEB-API , צד הלקוח הוא השכבה הגבוהה ביותר)

**DAL – השכבה ................**

מחלקת User – המחלקה משמשת ל....

# תיאור התוכנה

* + סביבת עבודה:

Visual Studio Codeו Visual Studio

* + שפות תכנות:

צד השרת נכתב בטכנולוגית WebApi ובשפת c#.

**.** angular בטכנולוגית -Html, css ,typescript צד הלקוח נכתב בשפות

# אלגוריתמים מרכזיים

כאן תפרטי את הפעולות העיקריות בפרויקט.

## חלק מהאלגוריתם... הפיכת התמונה לשחור לבן

## חלק אחר מהאלגוריתם... פירוק התמונה לאותיות

## החלק העיקרי באלגוריתם למידת מכונה – וזיהוי האות

# קוד האלגוריתם

לכאן תעתיקי את הפונצקיות העיקריות בפרויקט

# תיאור מסד הנתונים

תצלמי את הדיאגרמה מה-SQL

## פירוט הטבלאות ב- Data Base

יש לפרט שם כל טבלה, הסבר על תפקידה ועל עמודות חשובות (מפתח זר או קוד מסוים ) ולמלא:

טבלת ??????? – משמשת ל......

U\_userid – תפקידו הוא.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מפתח | שם שדה | תיאו ר | טיפו ס | שדה שאינו חובה |
| pk | u\_UserId | קוד משתמש | int |  |
|  | u\_FirstName | שם פרט י | string |  |
|  | u\_LastName | שם משפח ה | String |  |

# מדריך למשתמש

## תיאור המסכים

תרשים זרימה של (כל)המסכים:  


## מדריך למשתמש

תכתבי הדרכה למישהו שלא מכיר את האתר שלך ( בכניסה לאתר יש.... בפרופיל האישי ניתן לבחור...)

## צילומי מסכים

לכל מסך: שם המסך ותמונה

# בדיקות והערכה

(דוגמא: כל אחת צריכה לכתוב בניסוח משלה!!!  
לאחר הרצת האלגוריתם נבחנו כל האילוצים שדרושים כדי להביא לשיבוץ אפשרי ואופטימל י. כאשר הופיעו טעויות ובאגים בביצוע של האלגוריתם נבדק הקוד שוב עד שתוקנו הבעיות . לאח ר בדיקות רבות אחר כל מקרי הקצה שעלו בדעתי ,והרצת האלגוריתם מספר פעמים על נתוני ם שונים, האלגוריתם הגיע לקירוב האפשרי ביותר בכלים העומדים לרשות י. )

# ניתוח יעילות

(תסבירי מה זה יעילות למה היא הייתה חשובה לך וכמה הסיבוכיות של האלגוריתם שלך)

# אבטחת מידע

(תכתבתי על כניסה באמצעות סיסמא – אם יש)

# מסקנות

(מסקנות חיוביות ....... ; ) בנוגע לצורת העבודה, לידע שצברתן, איך לכתוב קוד, איך לבדוק, מה חשוב לעשות, מה חשוב לא לעשות, מסקנות לעתיד, מלא אופטימיות וסיפוק.....ופרגון לפרויקט)

# פיתוח עתידי

מה תוסיפי לפרויקט אם יהיה זמן ומשאבים..........

# ביבליוגרפיה

רשימת האתרים שבהם נעזרת במהלך העבודה.