

המכללה
הטכנולוגית
באר שבע

בשיתוף WORLD ORT
קדימה מדע



בטוח
מהיר
נח

EaSElect

מוגש ע"י: יעל עמר וכרמל אבדר

מגמת הנדסאות תוכנה

שם המנחה: גב' מ. שמעוניוביץ

ינואר, 2020

המכללה
הטכנולוגית
באר שבע

בשיתוף WORLD ORT
קדימה מדע



בטוח מהיר נח EaSElect

שמות המגישות: יעל עמר וכרמל אבדר

חתימה: _____

חתימה: _____

שם המנחה: גב' מ. שמעונוביץ

חתימת המנחה: _____

שם רכז המגמה: שם המרכזת במכללה

חתימה: _____

ינואר, 2020

תוכן עניינים

III.....	תודות.....
IV.....	תקציר.....
V	הצהרת סטודנטים.....
VII	הצעת פרוייקט.....
XX	אישור הצעת פרוייקט.....
1.....	גוף העבודה
1	1. מבוא
2	2. מדריך למתכנת
2	2.1 אסטרטגיות טכנולוגיות
4	2.2 תיאור מבנה הפרויקט
6	2.3 עקרונות התכנון/ הבניה/ הניתוח
6	2.3.1 עקרונות תיאורטיים
7	2.3.2 עקרונות תכנותיים
8	2.4 תרשימים
8	2.4.1 תרשים UML
9	2.4.2 תרשים מראה המחלקות
12	2.5 מבנה נתונים מאוכסנים
12	2.5.1 תרשים SQL
13.....	2.5.2 תיאור הטבלאות.....
14	2.5.3 מבנה קבצים ותיקיות
15	2.6 תוכן הפרויקט
15	2.6.1 תיאור המחלקות
20	2.6.2 תיאור הפונקציות
21	3. מדריך למשתמש
21	3.1 הוראות כלליות לשימוש באתר
23	3.2 צילומי מסכים
27	4. סיכום ומסקנות
28	5. ביבליוגרפיה

תודות

רצינו להביע את תודתנו לכל מי שסייע לנו במהלך העבודה על הפרויקט. לגב' שמעונוביץ, מנחת הפרויקט שהדריכה אותנו במהלך העבודה ולימדה אותנו את עקרונות השפה והטכנולוגיה שבה עבדנו.

לגב' לוי, רכזת מגמת מחשבים אשר דאגה שנקבל את כל התנאים שהיינו צריכות בשביל הפרויקט ונתנה לנו עצות בעת הצורך.

להורים שלנו שעמדו לצדינו, תמכו ועודדו אותנו לאורך הפרויקט.

וכמובן, אין אנחנו מספיקין להודות... לריבונו של עולם שראינו הרבה סיעתא דשמיא במהלך העבודה.

תקציר

תהליך של ביצוע סקר כלשהו, או בחירות הוא תהליך ארוך ומורכב. הוא גוזל זמן, משאבים ודורש מנגנוני הגנה רבים כדי שיהיה חשאי ואמין.

כיום, הקולות נספרים בסיום הבחירות בכל קלפי בנפרד בצורה ידנית: לכל מפלגה יש שיפוד, כל פתק כשר משופד למפלגה שלו, ובסיום התהליך סופרים כמה פתקים לכל שיפוד ואלו הקולות מקלפי זה למפלגה זו. תהליך זה ארוך זמן רב, ואין מן הנמנע שיפלו בו טעויות רבות הנובעות מטעויות אנוש או טעויות מכוונות. כמו"כ כאשר יודעים את הנתונים ע"פ קלפיות, יש בכך פגיעה בחשאיות ההצבעה, שכן תושבי אותו אזור המצביעים בקלפי זה, ידעו לנחש עבור הקולות הבודדים מי הצביע אותם.

נוסף על כך, בצורת הבחירות הנוכחית כל אדם חייב להצביע במקום מגוריו, מה שמחייב את המדינה להודיע על יום שבתון, כדי לאפשר לאנשים להגיע למקום ההצבעה שלהם. אנשים הגרים בחו"ל אינם יכולים להצביע, וסטודנטים או אנשים הנמצאים בדרך קבע רחוק ממקום מגוריהם נאלצים לגמוע ק"מ רבים כדי לממש את זכותם.

תוצאות אמת לבחירות מופיעות רק לאחר כשבוע מזמן הבחירות, ועד אז ישנם רק סקרים, ניחושים וקונספירציות.

מטרת הפרויקט היא לאפשר ביצוע של בחירות בחשאיות, בקלות ובצורה ממוחשבת. הפרויקט מותאם לכל גודל של בחירות, החל מבחירות במפעל או בי"ס הרוצים לבדוק מהי הארוחה המועדפת עליהם וכלה בבחירות לכנסת. הצבעה בבחירות בצורה כזאת תפתור בעיות רבות הקיימות בצורת הבחירות הנוכחית. ראשית, חסכון כלכלי. עלות בחירות בישראל מסתכמת בכ- מילארד ₪ לכל מערכת בחירות (בחישוב ההפסד הכלכלי למשק מיום שבתון, עובדי הבחירות בקלפיות וקמפייני המפלגות). שנית, אמין יותר: אין צורך לסמוך על אף אדם שיספור נכון את הקולות ולא יעלים כמה קולות שאינם מהמפלגה אליה הוא מצביע, שלישית, אין צורך לנסוע כדי להצביע, מה שיחסוך את יום השבתון הכללי. בעיה אחרונה שתיפתר היא החשאיות: רק הבוחר מחליט אלו נתונים אישיים עליו הוא מסכים למסור למערכת הבחירות.

הצהרת סטודנטים

מה"ט
המכון הממשלתי להכשרה בטכנולוגיה ובמדע
יחידת הפרויקטים
חוזר מנהל מה"ט 11-4-52 – נספח מס' 3

הצהרת סטודנט

שם הסטודנט: כרמל 37/א ת.ז. 31215 9321

שם המכללה בה לומד הסטודנט: המכללה הטכנולוגית - קריית גת

אני הח"מ, מצהיר בזאת כי פרויקט הגמר וספר הפרויקט המצ"ב נעשו על ידי בלבד.
פרויקט הגמר נעשה על סמך הנושאים שלמדתי במכללה ובאופן עצמאי.
פרויקט הגמר וספר הפרויקט נעשו על בסיס הנחייתו של המנחה האישי.
מקורות המידע בהם השתמשתי לביצוע פרויקט הגמר מצוינים ברשימת המקורות המצוינים בספר הפרויקט.
אני מודע לאחריות שהנני מקבל על עצמי על ידי חתימתי על הצהרה זו שכל הנאמר בה אמת ורק אמת.

חתימת הסטודנט: כרמל 37/א תאריך: 27.12.03

אישור המנחה האישי

הריני מאשר שהפרויקט בוצע בהנחייתי, קראתי את ספר הפרויקט ומצאתי כי הוא מוכן לצורך חגשת הסטודנט להגנה על פרויקט גמר.

שם המנחה: _____ חתימה: _____ תאריך: _____

אישור ראש המגמה

הריני מאשר שספר הפרויקט מוכן לצורך חגשת הסטודנט להגנה על פרויקט הגמר.

שם ראש המגמה: _____ חתימה: _____ תאריך: _____

מחנ
המכון הממשלתי להכשרה בטכנולוגיה ובמדע
יחידת הפרויקטים
 חוזר מנהל מה"ט 11-4-52 – נספח מס' 3

הצהרת סטודנט

שם הסטודנט: יזרעאל ת.ז. 207944943

שם המכללה בה לומד הסטודנט: המכללה הטכנולוגית סני שף

אני הח"מ, מצהיר בזאת כי פרויקט הגמר וספר הפרויקט המצ"ב נעשו על ידי בלבד.
 פרויקט הגמר נעשה על סמך הנושאים שלמדתי במכללה ובאופן עצמאי.
 פרויקט הגמר וספר הפרויקט נעשו על בסיס הנחייתו של המנחה האישי.
 מקורות המידע בהם השתמשתי לביצוע פרויקט הגמר מצוינים ברשימת המקורות המצוינים בספר הפרויקט.
 אני מודע לאחריות שהנני מקבל על עצמי על ידי חתימתי על הצהרה זו שכל הנאמר בה אמת ורק אמת.

חתימת הסטודנט: יזרעאל תאריך: 16.04.20

אישור המנחה האישי

הריני מאשר שהפרויקט בוצע בהנחייתי, קראתי את ספר הפרויקט ומצאתי כי הוא מוכן לצורך הגשת הסטודנט להגנה על פרויקט גמר.

שם המנחה: _____ חתימה: _____ תאריך: _____

אישור ראש המגמה

הריני מאשר שספר הפרויקט מוכן לצורך הגשת הסטודנט להגנה על פרויקט הגמר.

שם ראש המגמה: _____ חתימה: _____ תאריך: _____

הצעת פרויקט



תאריך: 3/6/2019

לכבוד: יחידת הפרויקטים מה"ט

הצעה לפרויקט גמר

א. פרטי הסטודנטים

שם הסטודנט	ת.ז. 9 ספרות	כתובת	טלפון נייד	שנת סיום הלימודים
יעל עמר	207944943	דקל 58 רכסים	052-7664160	תשע"ט
כרמל אבדר	315157321	הורד 8 אליכין	052-7131197	תשע"ט

שם המכללה: סמינר אופקים טכנולוגית ב"ש

סמל המכללה: 72266

מסלול ההכשרה: הנדסאים

מגמות לימוד: תכנות מחשבים

מקום ביצוע הפרויקט: בבית ובסמינר

ב. פרטי המנחה האישי

שם המנחה	כתובת	טלפון נייד	תואר	מקום עבודה/תפקיד

חתימת הגורם המקצועי מטעם מה"ט

חתימת המנחה האישי

חתימת הסטודנט

1. שם הפרויקט :

EaSElect .

2. רקע**2.1. תיאור ורקע כללי**

הפרויקט הינו אפליקציה לבחירות מסוגים שונים, המאפשרת לבחור בטביעת אצבע. כל מנהל שרוצה לקיים משאל בארגון שלו יוכל לבצע אותם בצורה אמינה, נוחה ועם אחוזי הצבעה גבוהים. מבנה האפליקציה מותאם לכל סוג בחירות בכל סדר גודל, בשאיפה שאף הבחירות הכלליות בכנסת יוכלו להתבצע בו (כמובן לאחר אבטחה מתאימה).

2.2. מטרות המערכת:

- לאפשר לבחור להצביע בצורה נוחה, נגישה ואמינה והכי חשוב - חשאית.
- לאפשר למנהל להגדיר רשימת בוחרים ואופציות לבחירה.
- לאפשר למנהל לצפות בתוצאות הבחירות בזמן אמת כולל פילוח נתונים לפי קבוצות באוכלוסייה.
- לאפשר לבחור להצביע מכל מקום בו הוא נמצא.
- להעלות את אחוז ההצבעה ואת המעורבות של הקהל המצביע בעניין המוצבע.

3. סקירת מצב קיים

נכון להיום הצבעות נעשות ע"י קלפיות, מול קהל משגיחים שיכולים להוות לחץ עבור הבחור. יתרה מזאת, האדם צריך לפנות זמן מסדר יומו העמוס, לגשת לקלפי הקרובה לביתו ואולי להתמודד עם ויכוחים על תורות וכו', מה שמוריד את אחוז ההצבעה.

ספירת הקולות נעשית ע"י פתקים ומעטפות, בצורה מסורבלת וארוכה, מה גם שיכולה להיות שגויה לפעמים.

במקומות אחרים בעולם קיים מצב מעורב – האנשים באים לקלפי בעצמם, אך שם מצביעים באמצעות תכנה.

כמו כן, כאשר מנהל ארגון רוצה לבדוק את ההעדפות של עובדיו הוא עושה זאת ע"י סקרים או כל דרך מסורבלת אחרת.

4. מה הפרויקט אמור לחדש או לשפר

הפרויקט שלנו מאפשר הצבעה מהירה ונוחה, ללא קשר לגודל האוכלוסייה המצביעה. זוהי דרך אמינה וקלה, מה שיעלה מאוד את אחוז ההצבעה. הפרויקט יחסוך כמובן את אחזקת כל העובדים האחראיים על הקלפיות, יעלה את רמת האמינות של ההצבעה, ספירת הקולות לא תהיה ידנית,

מה שימנע בברור טעויות וספירה לא נכונה. התוצאות יתקבלו תוך שניות מזמן סיום ההצבעה, והן יהיו מפולחות לפי קריטריונים קבועים מראש שלא יפגעו בחשאיות הבחירה.

5. דרישות מערכת ופונקציונאליות

5.1. דרישות מערכת, סביבת הטמעה ושימוש

לכתיבת הפרויקט נדרש:

- מחשב נייד עם professional 8 windows
- visual studio 2017
- Visual studio code
- server-Sql
- SSMS

לסביבת ההרצה נדרש:

בצד שרת:

- שרת אינטרנט המריץ IIS
- שרת sql-server

בצד הלקוח:

- דפדפן אינטרנט
- סמארטפון אנדרואיד עם קורא טביעת אצבע.

5.2. שרידות, ביצועים, התמודדות עם עומסים

צד השרת מריץ IIS Express המסוגל להתמודד עם מספר קריאות של קריאות בו זמנית. גם עומס על שרת הSQL אינו צפוי בסדר גודל כזה של אתר מכיוון שהוא בנוי להתמודדות בהצלחה עם עומסים כבדים בהרבה.

5.3. דרישות פונקציונאליות

מנהל:

5.3.1 מאפשר למנהל לייבא את טביעות האצבע של הבוחרים לתוך מאגר הבוחרים מתוך מאגר קיים.

5.3.2 מאפשר למנהל להכניס טביעות אצבע חדשות באופן ישיר למאגר הבוחרים.

5.3.3 מאפשר למנהל לקבוע את אופציות הבחירה, כולל תאריך ושעה, אופציות בחירה וסינון בוחרים ע"י סימון מסווג מרשימה.

5.3.4 מאפשר למנהל לשלוח קישור לאפליקציה לכל בוחר.

5.3.5 מאפשר למנהל לצפות בתוצאות הבחירות בזמן אמת גם ע"פ פילוחים שונים.

בוחר:

- 5.3.6 הבוחר יזדהה במערכת ע"י הזנת טביעת אצבע.
- 5.3.6.1 המערכת תבדוק אם טביעת האצבע קיימת במאגר הבוחרים, במקרה אינו קיים, תציג הודעה מתאימה ותסגור את האפליקציה אוטומטית.
- 5.3.7 המערכת תציג לבוחר את פרטי הבחירה שלו הכוללים את אופציות הבחירה והמנהל.
- 5.3.8 ביום הבחירות, בשעות הקבועות המערכת תאפשר לבוחר להיכנס למערכת לבצע בחירה.
- 5.3.8.1 המערכת תבדוק שהבוחר לא הצביע כבר.
- 5.3.8.1.1 במקרה שכבר הצביע, המערכת תציג הודעה מתאימה ותסגור את האפליקציה אוטומטית.
- 5.3.8.2 המערכת תאפשר לבוחר לבחור אופציית בחירה מסוימת.
- המערכת תבדוק שתאריך הבחירה זהה לתאריך יום הבחירות, במקרה אחר, תציג הודעה עם התאריך הנדרש.
- 5.3.8.3 המערכת תזהה את הבחירה שנקלטה, תיתן לבוחר אפשרות לאשר את בחירתו.
- 5.3.8.3.1 במקרה שאישר-תיתן לו הודעה שהבחירה נקלטה בהצלחה.
- 5.3.8.3.2 במקרה שלא- תינתן לבוחר אפשרות לבחור שנית, עד לאישורו הסופי.
- 5.3.8.4 במקרה שהבוחר אישר את בחירתו, התוצאה נשמרת, ומתווספת למאגר התוצאות.
- 5.3.9 בסוף יום הבחירות תשלח המערכת הודעה על תוצאת הבחירות לכלל הבוחרים.

6. בעיות צפויות במהלך הפיתוח**6.1. הבעיות:**

- 6.1.1. בעיה 1: כיצד נוודא שאדם לא יוכל לבחור יותר מפעם אחת?
- 6.1.2. בעיה 2: איך נוודא שהבחירות תהיינה חשאיות?

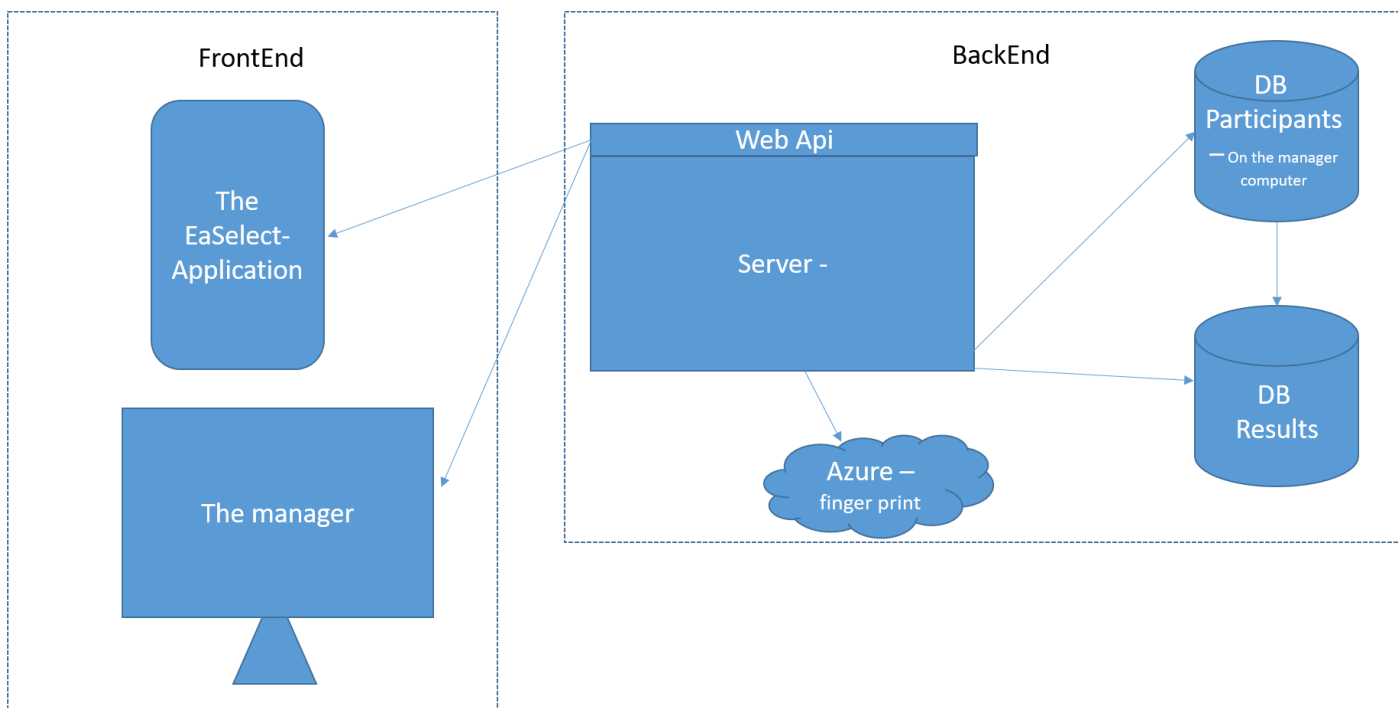
6.2. פתרונות אפשריים:**6.2.1. לבעיה 1:**

פתרון 1: להוסיף במסד הנתונים עבור כל רשומה מאפיין האם הצביע/לא.

- פתרון 2: להשוות מול מסד הנתונים של התוצאות, האם טביעת האצבע
 כבר קיימת במאגר כאדם שהצביע.
- 6.2.2. לבעיה 2: פתרון 1: לא נשמור פרטים מסגירים על הבוחר (כמו שם, ת"ז או
 כתובת), אלא רק מאפיינים כלליים.
 פתרון 2: נצפין את פרטי הבוחר.
- 6.3. הפתרון הנבחר עבור כל אחת מהבעיות:
- 6.3.1. לבעיה 1: פתרון 2, כיוון שבמאגר התוצאות אין לנו שום עניין לשמור
 פרטים של אדם שלא הצביע.
- 6.3.2. לבעיה 2: פתרון אחד, כיוון שכל הצפנה ניתן לפרוץ, ולכן עדיף לא לשמור
 כלל.

7. פתרון טכנולוגי נבחר

7.1. טופולוגית הפתרון



המערכת מורכבת משרת IIS המריץ את ממשק המנהל (אתר), מסדי נתונים של SQL Server, ממשק משתמש בצד הלקוח: מנהל: דפדפן אינטרנט (chrome), בוחר: סמארטפון android עם קורא טביעת אצבע.

7.2. טכנולוגיות בשימוש:

Angular, טכנולוגיה זו מאפשרת עיצוב נח, רספונסיבי וכתיבה גם ל android.

7.3. שפות הפיתוח:

בצד השרת: C# שפה זו היא שפת תכנות פופולארית שמיועדת לפיתוח כללי של מגוון אפליקציות בכל התחומים. התחביר והעקרונות שלה פשוטים מצד אחד, אך עשירים ביכולות מצד שני, C# היא תנאי הכרחי לשימוש במגוון טכנולוגיות. SQL, כדי לפתח מסד נתונים.

בצד הלקוח:

Angular, טכנולוגיה זו מאפשרת לפתח אתרים בצורה נוחה, רספונסיבית ומתאימה לכמה מערכות הפעלה. Angular מכילה כמובן HTML & CSS, ופיתוח קוד בשפת Type-Script.

7.4. תיאור הארכיטקטורה הנבחרת

הארכיטקטורה הנבחרת היא חלוקה ל2 פרויקטים , צד שרת וצד לקוח. צד שרת מחולק ל3 שכבות: DAL(data access layer) , BLL(Business Logic Layer) , Controllers(קישור לממשק המשתמש). צד לקוח: מחולק ע"פ התצורה המקובלת של Angular.

7.5. חלוקה לתכניות ומודולים:

7.5.1. בצד השרת: 3 שכבות, DAL, BLL, Controller

7.5.2. בצד הלקוח: חלוקה לקומפוננטות

סביבת השרת

7.6. ממשק המשתמש/לקוח – GUI מורכב מדפי HTML שמוצגים למשתמש דרך הדפדפן, וכן ממשק תואם לאנדרואיד.

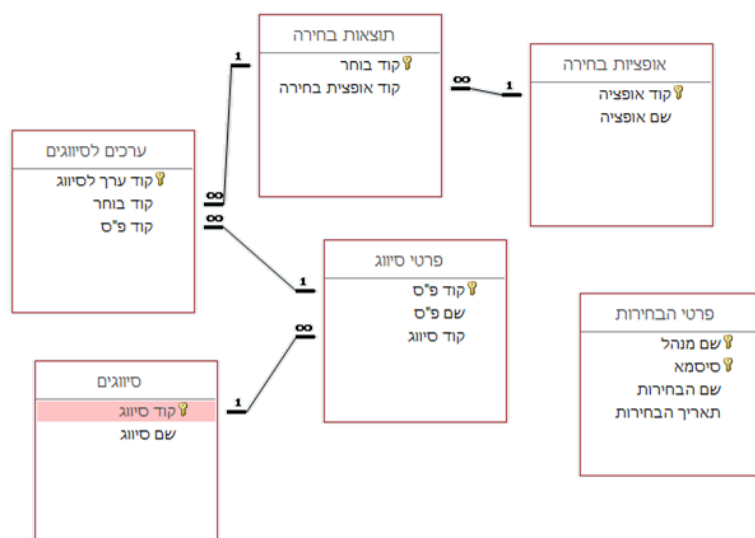
7.7. ממשקים למערכות אחרות / API

7.8. שימוש בחבילות תוכנה-JQuery, Bootstrap, CSS, Entity Framework

8. מבני נתונים וארגון קבצים

8.1. שיטת האחסון: המידע ישמר ב-Sql Server, טביעות האצבע ישמרו ב-Azure.

8.2. מבני הנתונים:

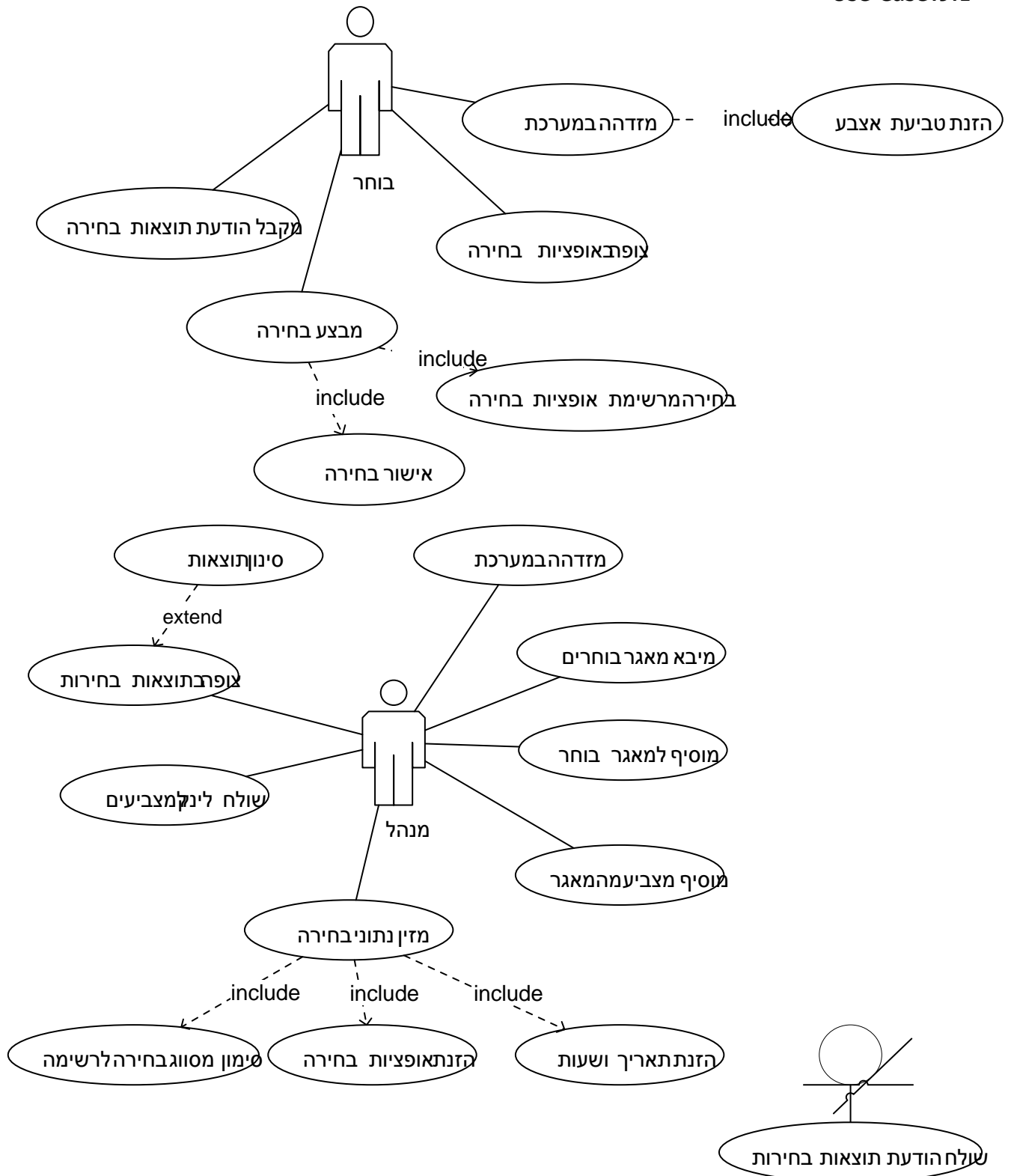


תוצאות בחירה:

8.3. מנגנוני התאוששות מפילה/ קריסה/ תמיכה בטרנזקציות.

9. תרשימי מערכת מרכזיים

Use Case 9.1



10. תיאור המרכיב האלגוריתמי – חישובי

10.1. הבעיה: לא לכל מנהל יש את טביעות האצבע של הבוחרים. הפתרון: בעזרת קורא טביעת אצבע נאפשר להוסיף טביעות אצבע חדשות. הבעיה: כיצד יכנסו המשתתפים לאפליקציית הבחירות המסוימת של המנהל שלהם. הפתרון: המנהל ישלח לינק להתקנה של האפליקציה עם פרטי ההצבעה הרצויה.

10.2. איסוף מידע וניתוחים סטטיסטיים (אנליטיקות). ניתן לבצע סטטיסטיקה למנהל על פי פילוחים מוגדרים באוכלוסייה. כמובן, פילוחים שאינם כוללים מידע פרטי על האדם או כל מזהה אחר. גם בזמן הבחירות וגם לאחריהם.

11. תיאור/התייחסות לנושאי אבטחת מידע

משתמש הגולש באתר מריץ אותה באופן אבסולוטי. קוד המשתמש ישלח לשרת ויצטרף לכל רשומה הנשמרת עבורו. לא יתכן מצב בו משתמש מריץ אפליקציה בתור משתמש אחר.

- במקרה שמנהל חדש מנסה להיכנס כמנהל רשום, המערכת תפנה אותו לדף ההרשמה לאתר.
- במקרה שבעת כניסת מנהל הסיסמא אינה תואמת לשם המשתמש שהקיש, המערכת תציג לו הודעת שגיאה ולא תאפשר כניסה.
- הסיסמא תהיה מוסתרת.
- במאגר הנתונים של תוצאות הבחירות, לא שמור שום פרט מזהה לגבי עובד כמו ת"ז, כתובת, פלאפון וכד'. מאגר טביעות האצבע של המצביעים יהיה שמור ב-Azure, ושם תתנהל ההשוואה. אין קשר בין מאגר המצביעים הקיים למאגר התוצאות כדי לוודא את חשאיות הבחירה. ייבוא נתוני הבוחרים יהיה מקובץ אקסל. מנהל הבחירות יצטרך להעביר את נתוני הבוחרים לאקסל.

12. משאבים הנדרשים לפרויקט:

- 12.1. מספר שעות המוקדש לפרויקט: 720
- 12.2. חלוקת עבודה בין חברי הצוות- עבודה ביחד
- 12.3. ציוד נדרש- מחשב הכולל חיבור לאינטרנט, CPU i5, 16GB RAM, HD SSD
- 12.4. תוכנות נדרשות- דפדפנים, IIS Express, SSMS, Visual Studio Code, Visual

Studio 2017, SQL Server

12.5. ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט-כתיבה לאנדרואיד, קריאת טביעת אצבע, התממשקות למייל.

12.6. ספרות ומקורות מידע

- stackoverflow.com
- codeproject.com
- msdn.microsoft.com
- getbootstrap.com
- [Angular.com](https://angular.com)
- w3schools.com

תכנית עבודה ושלבים למימוש הפרויקט:

<u>שלב</u>	<u>משך זמן</u>	<u>תאריכים משוערים</u>
○ ייזום הרעיון	שבועיים	ניסן
○ ניתוח מערכת	שבועיים	אייר
○ ניתוח מבנה נתונים	שבועיים	אייר
○ איפיון UI - UX	שבועיים	סיון
○ כתיבת הלוגיקה העסקית	חודשיים	עד סוף אלול
○ כתיבת ממשק המשתמש	חודש	עד סוף חשוון
○ עיצוב	שבועיים	כסלו

13. תכנון הבדיקות שיבוצעו:

מספר בדיקה	מס' דרישה במסמך אפיון	מקרי הבדיקה	ידנית / אוטומטית	חשיבות	הערות
1	5.3.8.2	הצבעה רק בתאריך הבחירות	ידני	קריטית	

	קריטית	ידני	הצבעה חד פעמית	5.3.8.1	2
	קריטית	ידני	הצבעה לרשומים בלבד	5.3.6.1	
	גבוהה	ידני	כתובת נמען חוקית	5.3.4	3
	גבוהה	ידני	שם משתמש חוקי	5.3.1	4
	גבוהה	ידני	סיסמא נכונה	5.3.1	5
	גבוהה	ידני	תוצאה התווספה למאגר בחירות	5.3.8.4	6
	גבוהה	ידני	קביעת נתוני בחירה	5.3.3	7
	בינונית	ידני	הכנסת טביעות אצבע חדשות	5.3.2	8
בינונית	ידני	ייבוא טביעות אצבע קיימות		5.3.1	9

14. בדיקות יחידה (unit test)

לא רלוונטי

15. בקרת גרסאות (version control)

GIT

חתימת המנחה האישי

חתימת הסטודנט

הגדל

הגדל

הערות ראש המגמה במכללה:

הגדל

אישור ראש המגמה במכללה:

שם: הגדל חתימה: הגדל תאריך: 21/7/19

הערות הגורם המקצועי מטעם מה"ס:

הגדל

אישור הגורם המקצועי מטעם מה"ס:

שם: הגדל חתימה: הגדל תאריך: 21/7/19

ביקור שטורקמן

חתימת הגורם המקצועי מטעם מה"ס

חתימת המנחה האישי

חתימת הסטודנט

הגדל

אישור הצעת פרויקט

המכללה
הטכנולוגית
באר שבע

בשיתוף WORLD ORT
קדימה מדע



10/10/2019

לכבוד
עמר יעל ת.ג. 207944943
צהל 13
אופקים
א.ג.נ.,

הנדון: אישור נושא הפרויקט

הנני להודיע כי הצעת הפרויקט בנושא "EASELECT"
בהנחיית - גב שימונוביץ מרים, אושרה ע"י המפקח בתאריך 27/08/2019.

הערות המפקח:

יפה! ממליץ להרחיב על צורת קריאת טביעת האצבע.

הנחיות לפרויקט ולמבנה פרויקט הגמר ניתן למצוא באתר המכללה באינטרנט,
מגמות – מדור פרויקטים.



מדור פרויקטים

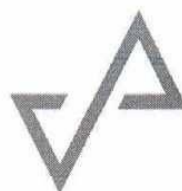
העתק : מנחה
תיק אישי

מדור חשבונות סטודנטים ופרויקטים
רחוב בזל 71, ת.ד. 45 באר-שבע 8410001
טל: 08-6462502 | פקס: 08-6462501
gabrielak@tcb.ac.il

1954-ב | www.tcb.ac.il

המכללה
הטכנולוגית
באר שבע

בשיתוף WORLD ORT
קדימה מדע



10/10/2019

לכבוד

אבדר כרמל ת.ז. 315157321

צהל 13

אופקים

א.ג.נ.

הנדון: אישור נושא הפרויקט

הנני להודיעך כי הצעת הפרויקט בנושא "EASELECT"

בהנחית - גב שימונוביץ מרים, אושרה ע"י המפקח בתאריך 27/08/2019.

הערות המפקח:

יפה! ממליץ להרחיב על צורת קריאת טביעת האצבע.

הנחיות לפרויקט ולמבנה פרויקט הגמר ניתן למצוא באתר המכללה באינטרנט,
מגמות - מדור פרויקטים.

העתק: מנחה

תיק אישי

בכבוד רב,

מדור פרויקטים

עדכונים ששינינו מההצעה בפועל

בתחילה חשבנו שהחלק שמתבצע בסמארט-פון יהיה אפליקציה, אך לאחר מחשבה נוספת, החלטנו שאין צורך להתקין אפליקציה לשימוש חד-פעמי כמו בחירות, ואפשר להשתמש באתר web רגיל.

כמו כן, ביצענו שינויים במסד הנתונים כדי שהבחירות תהינה חשאיות.

למשל, הוספת טבלה שתכיל את כתובות הדוא"ל של הבוחרים ללא קשר אליהם, אלא רק בקשר לקד הבחירה, כך שלא יהיה ניתן לגשת לכתובת הבוחר ולדעת מה הוא הצביע.

גוף העבודה

1. מבוא

הפרויקט שלנו מציע פלטפורמה נוחה לביצוע בחירות בכל גודל של ארגון.

מנהל הבחירות מגדיר בחירות, נותן להם שם, אופציות בחירה, תאריך התחלה ותאריך סיום. הוא טוען את פרטי כל המשתתפים בבחירות לתוכנה מתוך טבלת אקסל, שם שמורים כל טביעות האצבע של המשתתפים, פרטיהם, ופרטים אישיים עבור כל בוחר שלפיהם ירצה לערוך סיווגים אחרי הבחירות. בסיום התהליך הוא שולח מייל לכל המשתתפים בבחירות עם לינק לאתר הבחירות. האתר יפעל רק ביום הבחירות ולא יתן להצביע קודם הזמן, לא יתן להצביע פעמיים וכמו"כ לא יתן להצביע למי שלא רשאי להצביע. (במקרה של בחירות לכנסת ניתן לפרסם את הלינק גם באמצעי תקשורת שונים עבור מי שאין לו מייל)

מנהל הבחירות יוכל לצפות בנתוני הבחירות ע"פ סיווגים מסויימים, כל עוד הבוחר אישר לו לראות אותם. כך מתקבלים פילוחים סטטיסטיים מול אופציות הבחירה. הבוחר מקבל מייל עם לינק השולח אותו לאתר הבחירות. באתר הבחירות הוא מזדהה ע"פ טביעת אצבע, בקורא טביעת האצבע שע"ג הסמארט-פון שלו, נכנס למערכת, בוחר את בחירתו ובוחר איזה נתונים אודותיו הוא מסכים לשלוח למנהל, מתוך הנתונים אותם המנהל הגדיר שהוא רוצה לראות.

במהלך יום הבחירות המנהל יכול לצפות בזמן אמת בפרטי הבוחרים שהצביעו ע"פ סיווגים, אך לא בבחירה שלהם. כאשר נגמר זמן הבחירות, הוא יצפה גם בתוצאות הבחירות ע"פ סיווגים והמערכת תשלח מייל עם תוצאות הבחירות לכלל הבוחרים.

הבוחר נכנס לאתר ע"פ הלינק שקיבל, מזדהה עם טביעת אצבע, בוחר את בחירתו ולאח"מ מחליט אילו פרטים אישיים מתוך הפרטים שביקש המנהל הוא רוצה למסור. תהליך הבחירה הוא חד פעמי והוא לא יוכל לבחור שוב.

בסיום יום הבחירות הבוחר יקבל את תוצאות הבחירות לתיבת המייל שלו.

2. מדריך למתכנת

2.1 אסטרטגיות טכנולוגיות

צד השרת פותח בסביבת Visual Studio 2019/2017 ובטכנולוגית WebApi, בשיטות המתקדמות ביותר ש.NET מציעה.

צד הלקוח פותח בAngular 7 בסביבת Visual Studio Code.

#:C שפת C# היא שפת תכנות שפותחה ע"י מיקרוסופט ונחשבת לאחת משפות התכנות הפופולריות בעולם. היא מיועדת לפיתוח כללי של מגוון אפליקציות בכל התחומים- מאתרי Web, דרך משחקים, מאפליקציות למכשירי מובייל וטאבלטים ועד לשירותי ענן. התחביר והעקרונות שלה הם פשוטים מצד אחד אך עשירים ביכולות מצד שני.

Entity Framework : טכנולוגית עבודה מתקדמת מול DB. יצירת אובייקטים ומיפוי מול טבלאות ה-DB.

HTML : זוהי השפה הטבעית ליצירת דפי אינטרנט ברשת. זו שפה פשוטה, אוניברסלית, המאפשרת לעורכי אתרים ליצור דפים מורכבים שמכילים טקסט ותמונות, שיכולים להראות בידי כל המשתמשים ברשת האינטרנט ללא תלות בסוג המחשב או בסוג הדפדפן.

TypeScript : TypeScript היא שפת תכנות חנימית ומבוססת קוד פתוח המפותחת ומתוחזקת על ידי מיקרוסופט. היא ערכת-על של JavaScript הפופולרית ומוסיפה עליה טיפוסים סטטיים ותכנות מונחה עצמים מבוסס מחלקות. מאז השקתה חוותה הצלחה רבה בעיקר עקב היכולת לכתוב אפליקציות מורכבות על בסיס JavaScript.

CSS : CSS היא שפת עיצוב שמגדירה את תבנית העיצוב של מסמכי HTML למשל. CSS מטפלת בגופנים, צבעים, גבולות, שורות, גובה, רוחב, תמונות רקע, מיקום מתקדם ודברים רבים נוספים.

Angular : Angular הוא Framework של JavaScript שמשמש ליצירת אפליקציות מבוססות דף יחיד-SPA (Single Page Application). אפליקציות SPA מתנהגות כמו אפליקציות שרצות על מכשירים ניידים כי ניתן לעבור בין הדפים בלי צורך לטעון את כל הדף. כתוצאה מכך, אתר האינטרנט מציג התנהגות חלקה וחויית משתמש מעולה.

WebApi : ממשק אפליקציה למפתחים. כל מערכת אחרת יכולה להתחבר ל-API ולקבל ממנו מידע. על ידי התחברות ל-API דרך האינטרנט אנו חושפים מידע החוצה, משתמש הקצה לא יודע שהוא משתמש ב-SERVICE רק המתכנת עובד איתו.

Angular Material : ספריה המכילה פקדים מעוצבים בשפת עיצוב המכונה "material" מותאמת לשימוש בסביבת פיתוח Angular 2 ומעלה.

השתמשנו בסביבות העבודה הבאות:

:Visual Studio

סביבת הפיתוח המרכזית בעולם המייקרוסופטי נקראת Visual Studio. סביבה זאת מכילה מאות אפשרויות ותומכת במספר שפות תכנות. בנוסף, יש לציין כי רוב שפות התכנות והטכנולוגיות הנתמכות על ידי סביבת פיתוח זו הן חלק מתשתית הפיתוח של חברת Microsoft .NET Framework. תשתית זו מכילה מספר עצום של ספריות קוד וטכנולוגיות, המקלות עלינו בתהליך יצירת התוכנה.

:Visual Studio Code

ויז'ואל סטודיו קוד הוא שמה של פלטפורמה מבוססת קוד פתוח לעריכת קוד ופיתוח תוכנה הפועלת על מערכות ההפעלה, Windows, לינוקס ו OS X-העורך תומך בניפוי שגיאות, בקרת תצורה של גיט, המחשה סינטקטית של קטעי קוד, השלמת קוד חכמה, קטעי קוד אוטומטיים, ושינויי קוד רוחביים.

:SQL Server

Microsoft SQL Server : היא פלטפורמת שרת מידע ומבנה נתונים המיועדת למפתחים, ארגונים קטנים וגדולים כאחד אשר מציעה חבילה טכנולוגית שלמה של כלים ארגוניים המאפשרים להשיג את הערך המירבי מתוך המידע שלנו במחיר ובעלות הכוללת הנמוכים ביותר. תוכנת ניהול השרת מיועדת למשתמשים הזקוקים לפתרונות ניהול מידע באמצעות בסיסי נתונים בצורה יעילה ומאובטחת. בנוסף ניתן לפתח ולנהל תוכנות למחשבים שולחניים ותוכנות מבוססות רשת העושות שימוש בבסיס נתונים המנוהל תחת שרת SQL Server במהלך העבודה עימן.

שיתוף קוד וניהול גרסאות:

:Git

Git היא מערכת ניהול גרסאות מבוססת קוד פתוח, שמטרתה לסייע למפתחים בניהול קוד, תיאום עבודה צוותית ומעקב אחר שינויים בקובצי תוכנה. מטרותיה העיקריות הן לספק מהירות, שלמות מידע ותמיכה בתהליכים מבוצרים ולא ליניאריים.

:GitHub

GitHub היא תוכנת ניהול גרסאות ושירות אחסון, מבוסס רשת, עבור מיזמי פיתוח תוכנה, שבהם משתמשים בתוכנת GitHub. GitHub מספק שירות זה בתשלום למאגרים פרטיים ושירות חינמי למיזמי קוד פתוח.

2.2 תיאור מבנה הפרויקט

האתר שלנו מתחלק לשני פרויקטים, פרויקט עבור ה client ופרויקט עבור ה-server .

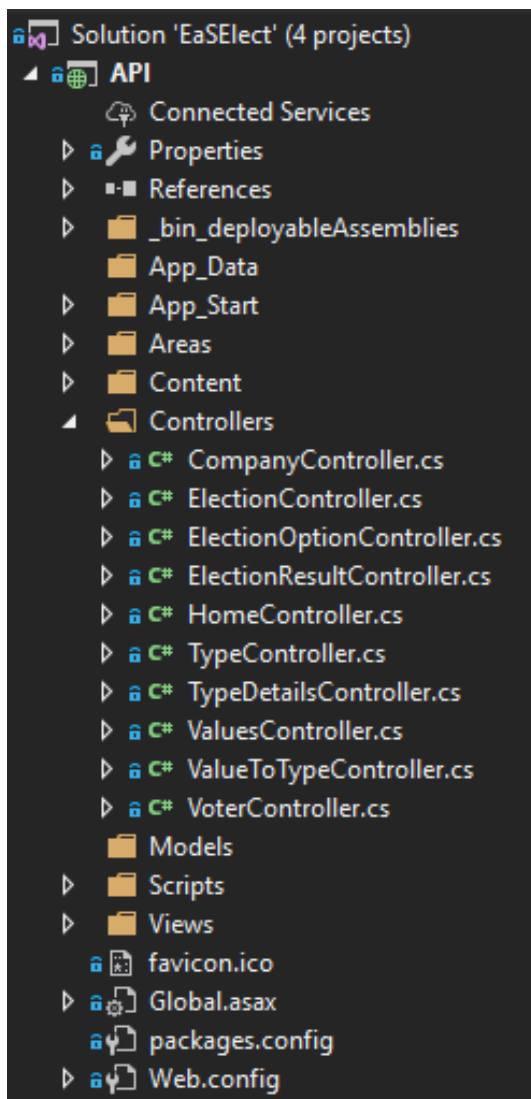
Server - Side

חלוקת המערכת לשכבות.

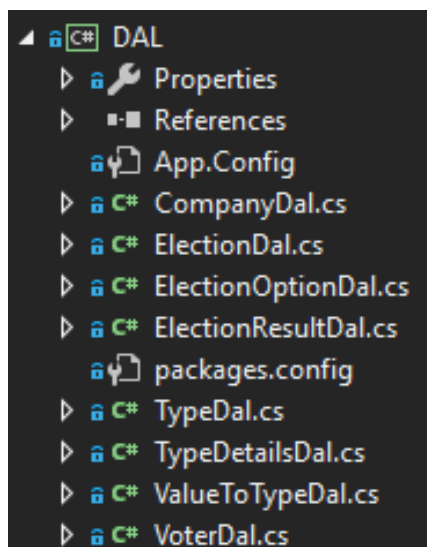
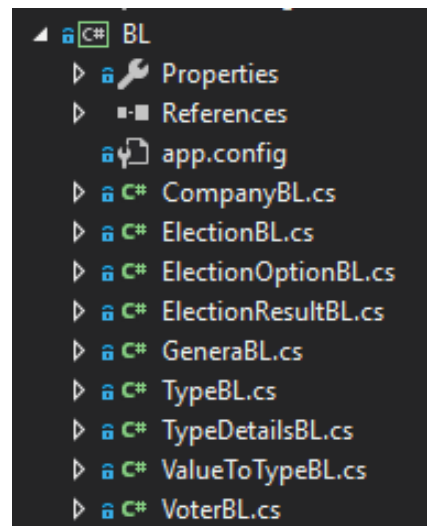
השרת נבנה בהתאם למודל השכבות הכולל את הפרויקטים הבאים:

- א. **Models** - פרויקט זה מתקשר למסד הנתונים הנדרש לפרויקט בשיטת DB.First, מכיל את המודלים המקבילים לישויות מסד הנתונים.
- ב. **DAL** - פרויקט זה משמש כשמו כאחראי על המידע, Layer Access Data כלומר על מסד הנתונים.
- ג. **BL** - פרויקט זה מכיל את עיקר הלוגיקה של הפרויקט כגון שליפת נתונים מ-DB הוספה, עדכון ומחיקה מה-DB.
- ד. **WebApi** - חשיפת API עמו מתממשקת האפליקציה.

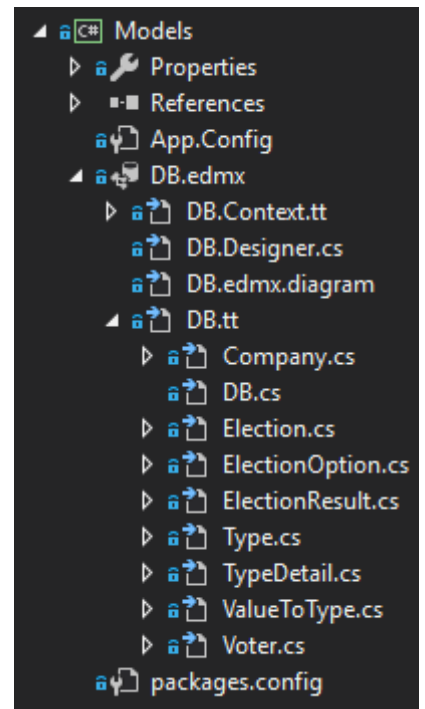
WebAPI



BL



Models

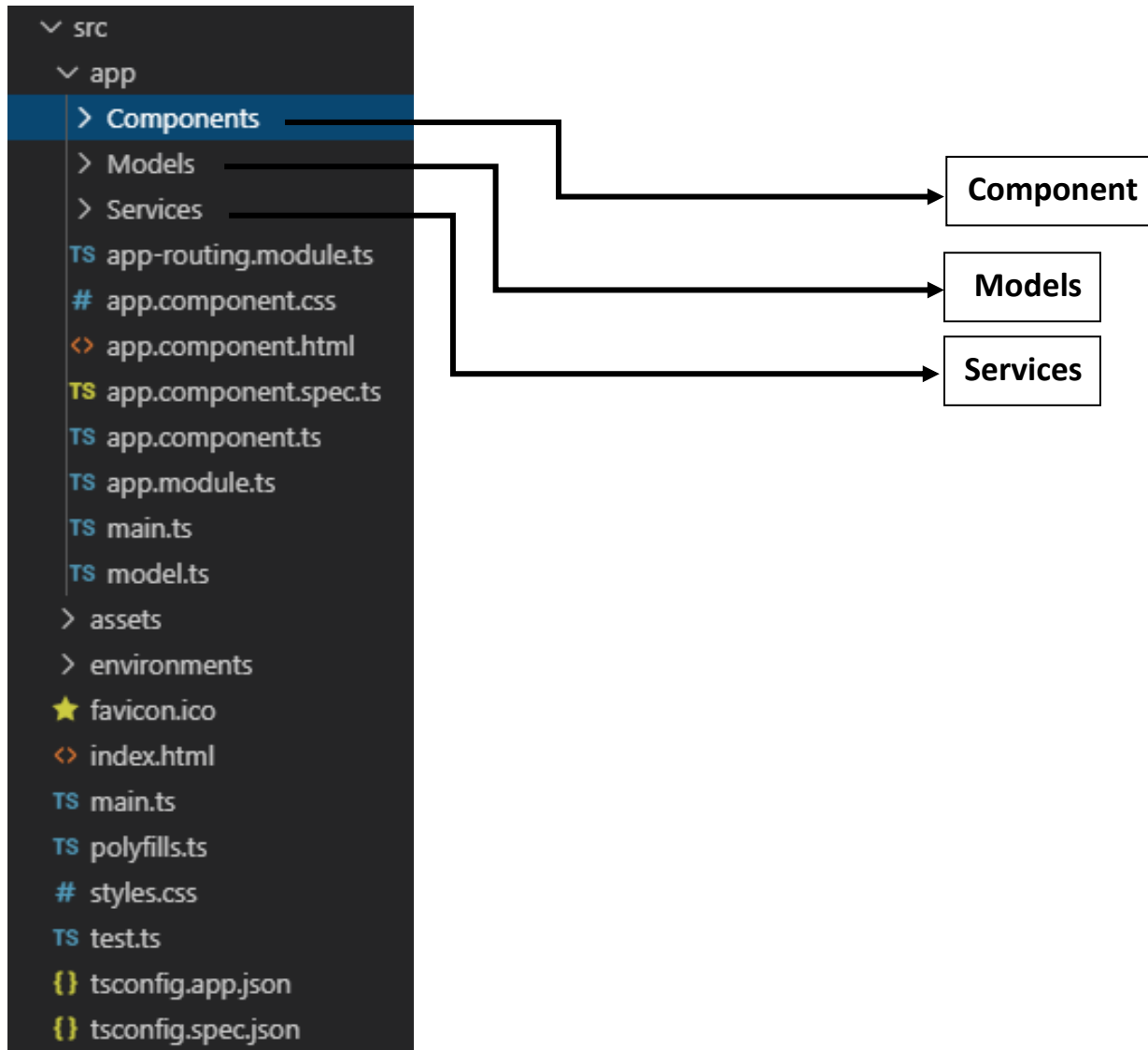


DAL

Client-Side

הפרויקט נכתב ב-Angular 7

1. Component - כל חל בתצוגה כולל את החלק הניהולי- הפונקציות והמשתנים שלו.
2. Services - מכילים את הפונקציות האחראיות על הגישה לפרויקט ה-Server.
3. Models - מבנה המחלקות.



2.3 עקרונות התכנון/הבניה/הניתוח

2.3.1 עקרונות תאורטיים

הפרדת שכבות

ישנם אתרים המבוססים על ארכיטקטורת שכבות הנקראת Tree Tire Architecture. בארכיטקטורה זו קיימת הפרדה בין השכבות במבנה של BL – GUI - DAL. זוהי תבנית עיצוב בסיסית שמגדירה את הפרדת האפליקציה לשכבת נתונים, שכבת לוגיקה, ושכבת משק משתמש. לתבנית עיצוב זו יתרונות רבים:

- תחזוקה:
ניתן להחליף או לתקן מימוש פנימי של שכבה אחת בארכיטקטורה בלי לשנות שכבה אחרת.
- נוחות פיתוח:
אדם אחד עובד על רכיב בתוכנה, אדם אחר עובד על רכיב אחר. כל עוד החתימות זהות ניתן לשלב כוחות ולייעל את מהירות הפיתוח.
- בדיקות:
כאשר נוצרת תקלה היא מבודדת בשכבה אליה היא שייכת. לדוגמה אם לא קיבלנו רשימת נתונים לתצוגה, נבדוק קודם בשכבת הנתונים (DAL) האם הנתונים שם תקינים, במידה והם אכן תקינים נעבור לבדוק את שכבת הביניים (BL), שכבת הלוגיקה, ואם גם שם הנתונים תקינים נדע בוודאות שהתקלה בשכבת התצוגה (GUI), ונפתור אותה שם.
- שימוש חוזר:
במידה ששכבות ה-DAL וה-BL עובדות היטב, וברצוננו להחליף את פלטפורמת ה-GUI, ניתן לעשות זאת במינימום מאמץ.

כיום הגישה לגבי אפליקציות web הוא כמה שיותר כתיבה בצד ה-client, מכיוון שכל פניה לשרת מכבידה על המערכת וגורמת לה לחכות עוד זמן, מה שלא כדאי שיקרה היות ולמשתמשים אין סבלנות לחכות עד לתוצאות הרצויות. בעקבות זאת השתדלנו שרוב הנתונים שלא צריכים עיבוד בשרת ימומשו בצד ה-Client וכן השתמשנו בספריית Angular 7 כדי לחסוך פניות לשרת.

הפרויקט נכתב בשפת C# ומחולק לשלוש שכבות:

- DAL - שכבת מסד הנתונים.
- BL - שכבת הלוגיקה.
- GUI - שכבת התצוגה למשתמש.

הורשה

בהורשה משתמשים בתכנות מונחה-עצמים, ירושה היא דרך לבסס יחס "סוג-של" בין עצמים. המימוש הנפוץ של הגדרה זו היא בעזרת מחלקות. מחלקה יכולה לקבל בירושה תכונות והתנהגות של מחלקת-האם שלה.

ירושה מאפשרת להשיג מספר מטרות:

1. שימוש חוזר בקוד. בעזרת ירושה ניתן לכתוב מחלקות בעלות טווח התנהגות 1 נרחב, תוך כתיבת קטעי קוד קטנים יחסית שמרחיבים מחלקה קיימת.
2. מידול של העולם האמיתי באופן דומה לזה המוגדר על ידי האדם. למשל, ניתן 2 לכתוב היררכיה של מחלקות המקבילה לטקסונומיה המקובלת בביולוגיה (יען היא סוג של ציפור, האדם הוא סוג של יונק, וכן הלאה).
3. פולימורפיזם (רב צורתיות) של זמן-ריצה. משתנה מטיפוס (סטטי) מסוים יכול להתנהג בצורות שונות, לפי הטיפוס של האובייקט שהוא מתייחס אליו בזמן ריצה. ניתן לבצע זאת גם ללא ירושה (בעזרת פקודות תנאי), אך ירושה מאפשרת מימוש מודולרי, גמיש וקל להרחבה.

בפרויקט השתמשנו בהורשה במקרים הבאים:

כל controller יורש ממחלקת ApiController. לדוגמא מחלקת VoterController.

```
public class VoterController : ApiController
```

מחלקת המקושרת למאגר DbContext יורשת ממחלקת lookItDBEntities הנתונים.

```
public partial class ElectionsDBEntities : DbContext
```

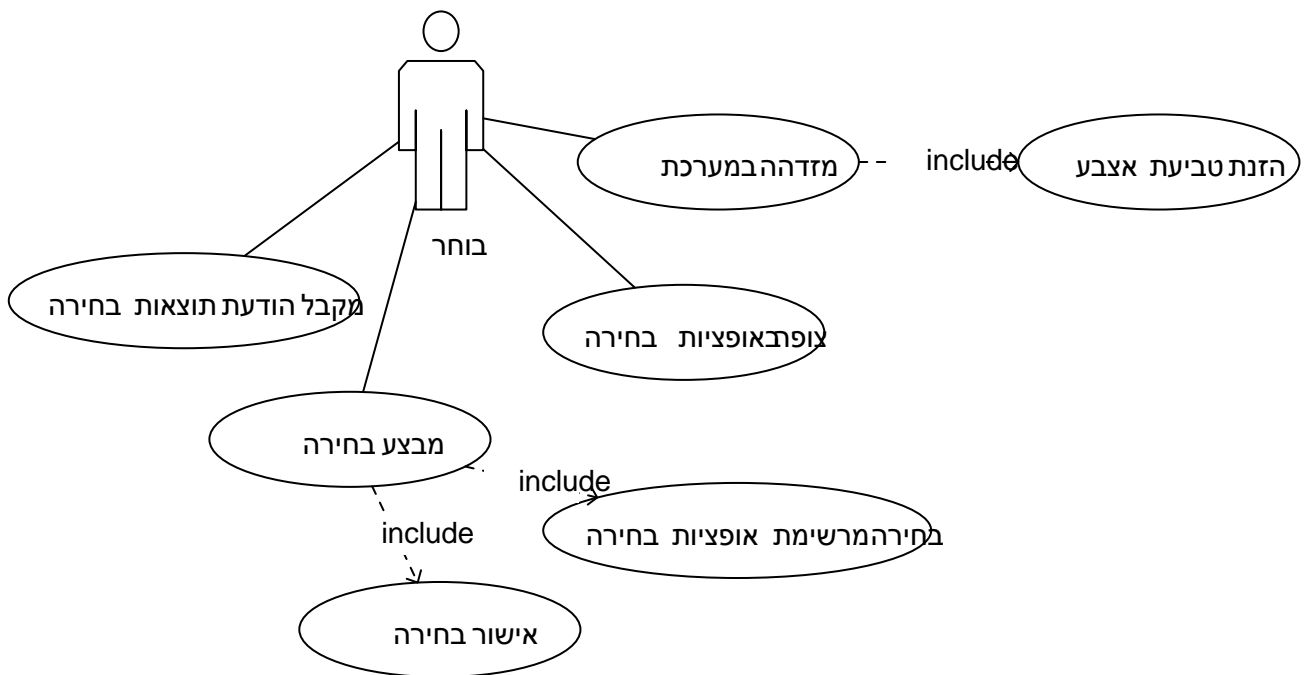
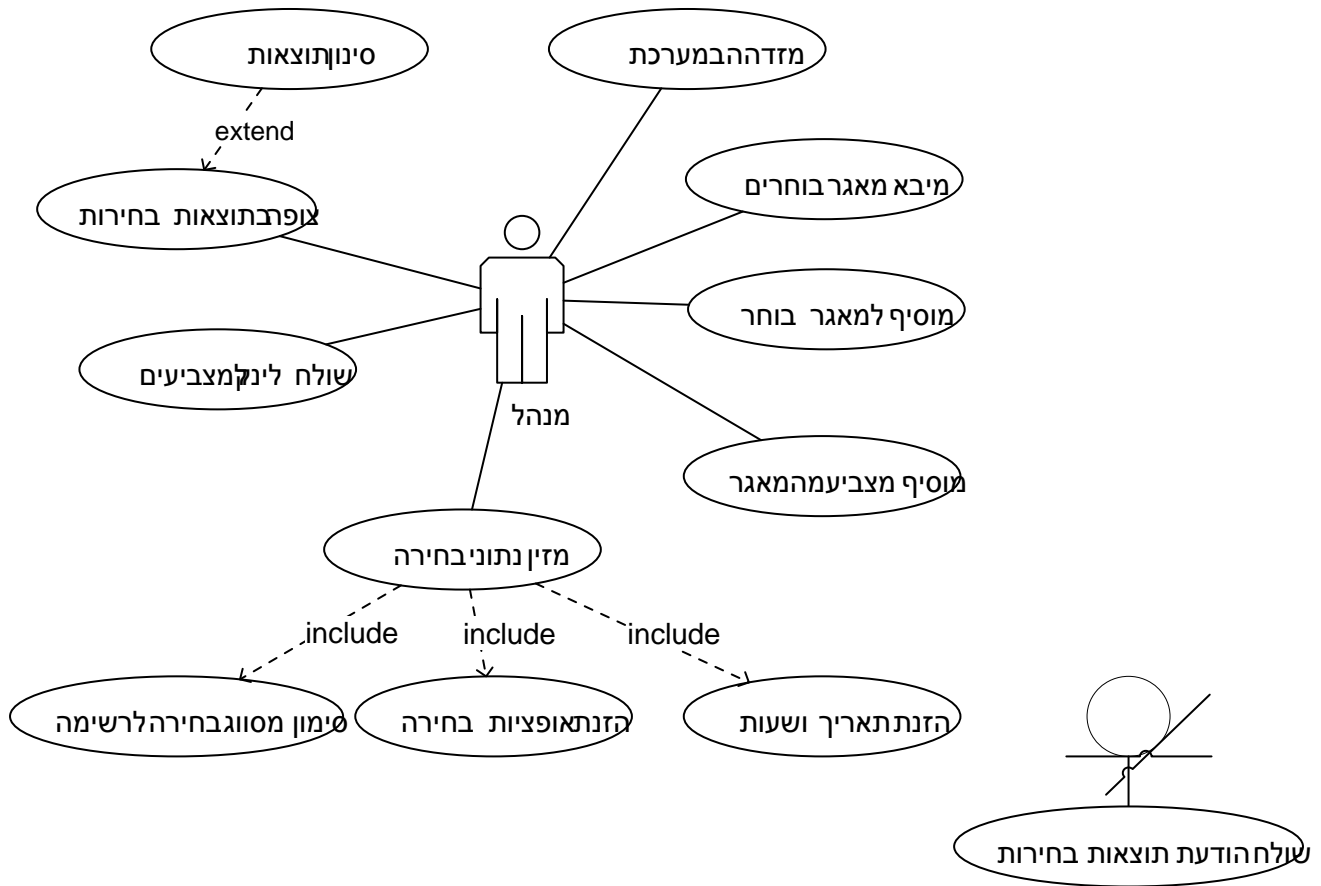
2.3.3 עקרונות תכנותיים

Session

שמירת מידע בצד שרת הוא משאב אישי ללקוח, המידע אינו משותף ואינו נגיש ללקוחות האחרים. כאשר הלקוח מבקר באתר, השרת מקצה לו אובייקט Session ושומר אותו בזיכרון. האובייקט נשמר בזיכרון כל עוד המשתמש נשאר באתר. אם במשך זמן (מוגדר מראש) האתר ננטש- האובייקט Session נמחק. הוא נמחק גם כאשר סוגרים את הדפדפן או כאשר נותנים הוראה למחיקתו. בעת כניסה לאתר ע"י שם משתמש וסיסמה נשמר המשתמש ב-Session. כך במעבר בין דפים נשמרים הנתונים למשתמש הנוכחי. בעת יציאה המשתמש ב-Session נמחק. במהלך ריצת האתר ניתנת למשתמש השמור ב-Session הרשאת גישה לדפים המיוחדים לו. בדוגמא זו רואים כי אם משתמש שאינו רשום מנסה להיכנס לדף שאינו רשאי לגשת, התוכנה תמנע ממנו זאת ויראה לו את דף הבית של משתמש לא רשום.

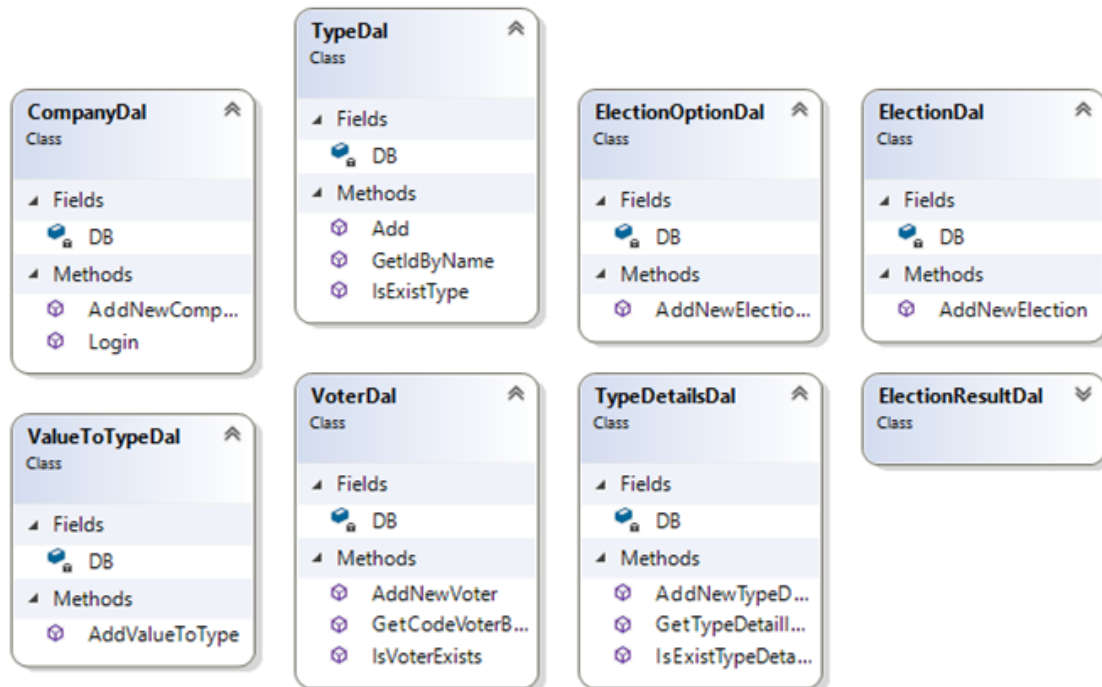
2.4 תרשימים

2.4.1 תרשים UML

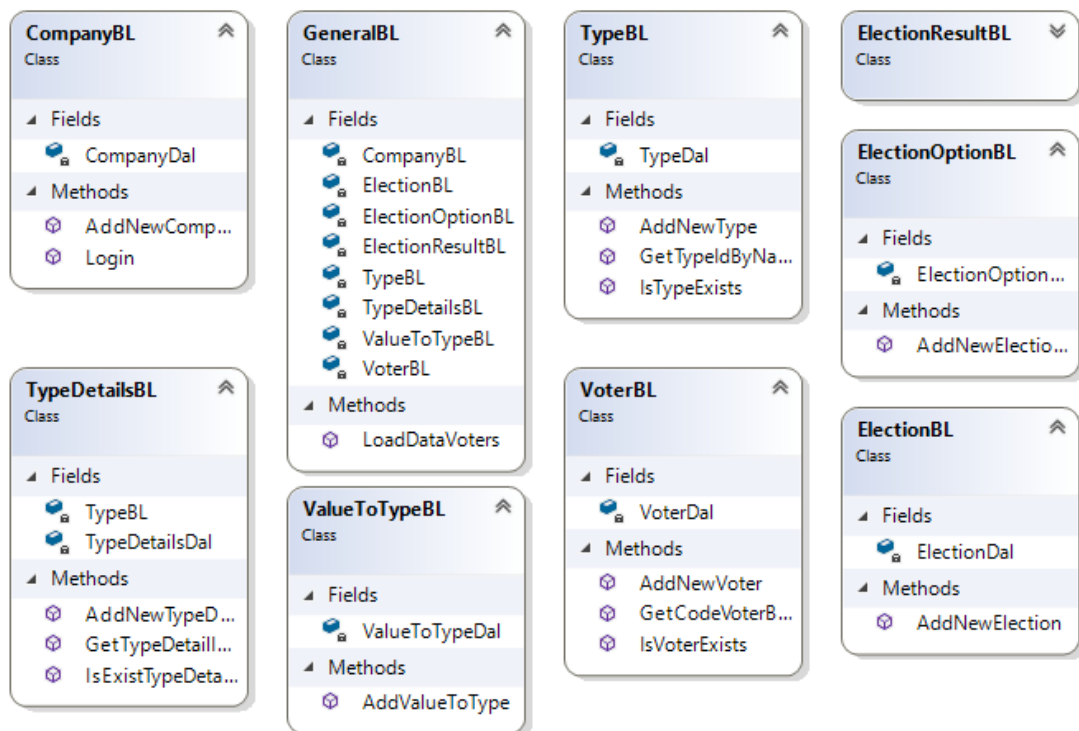


2.4.2 תרשים מראה המחלקות

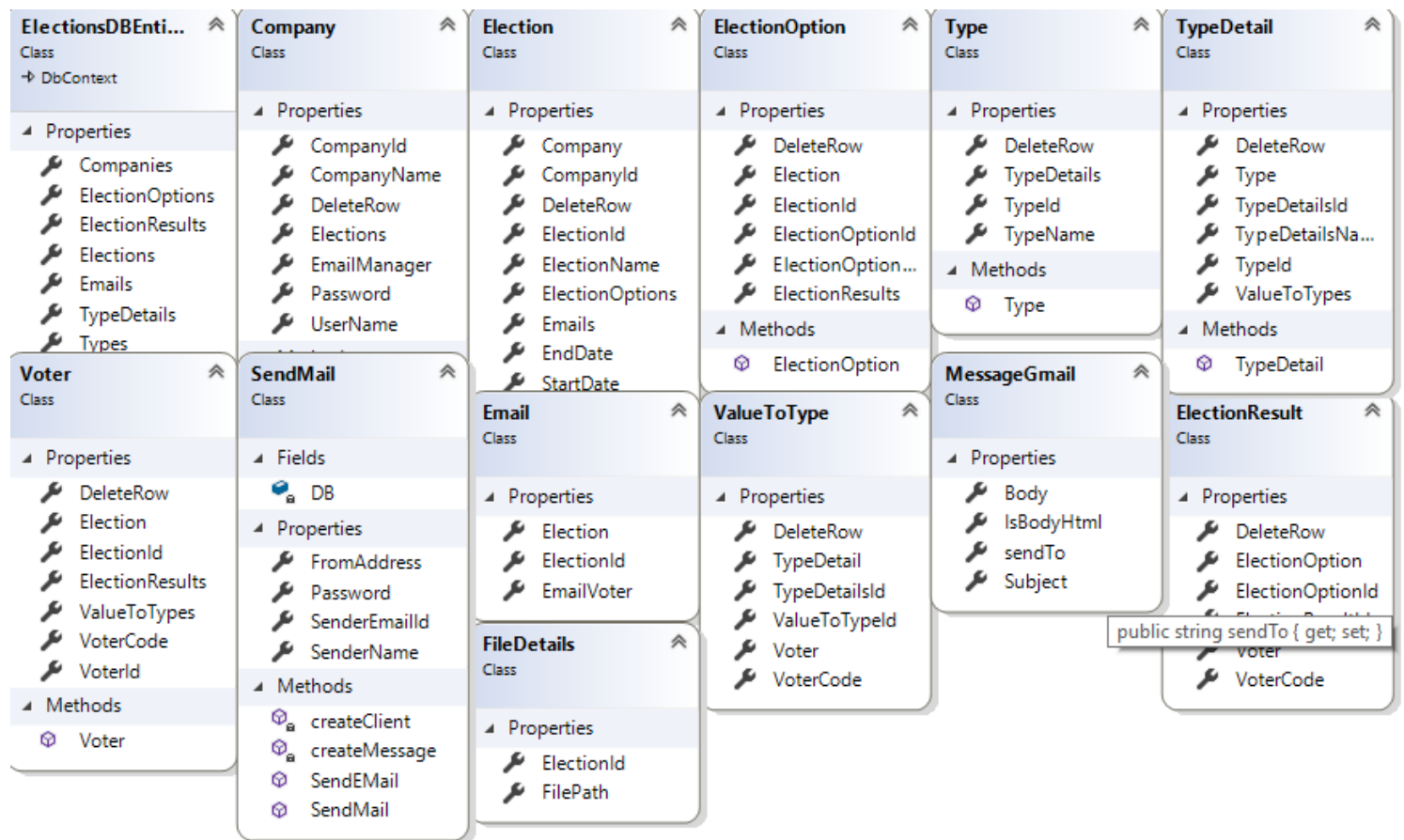
:DAL



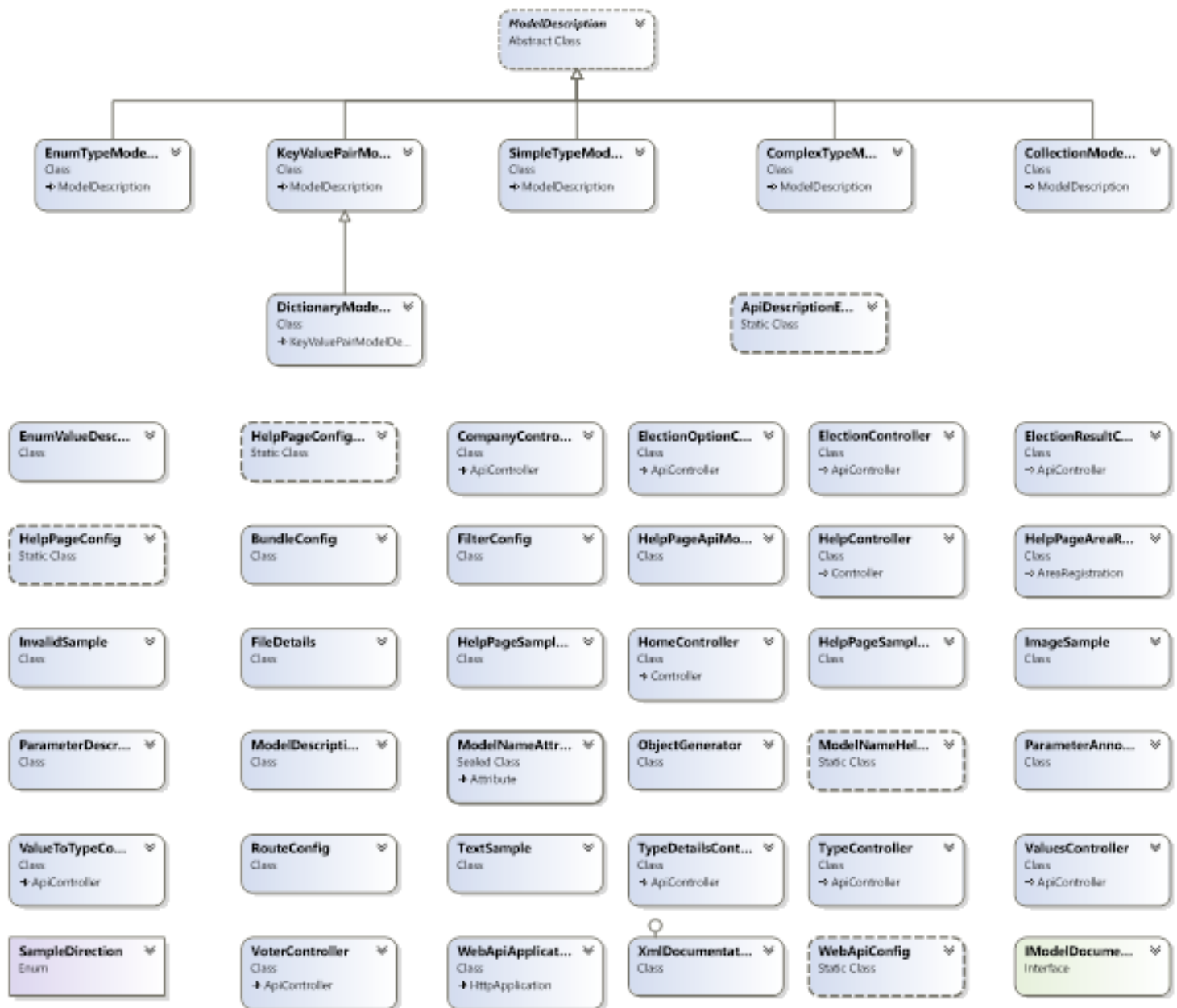
:BL



:Models

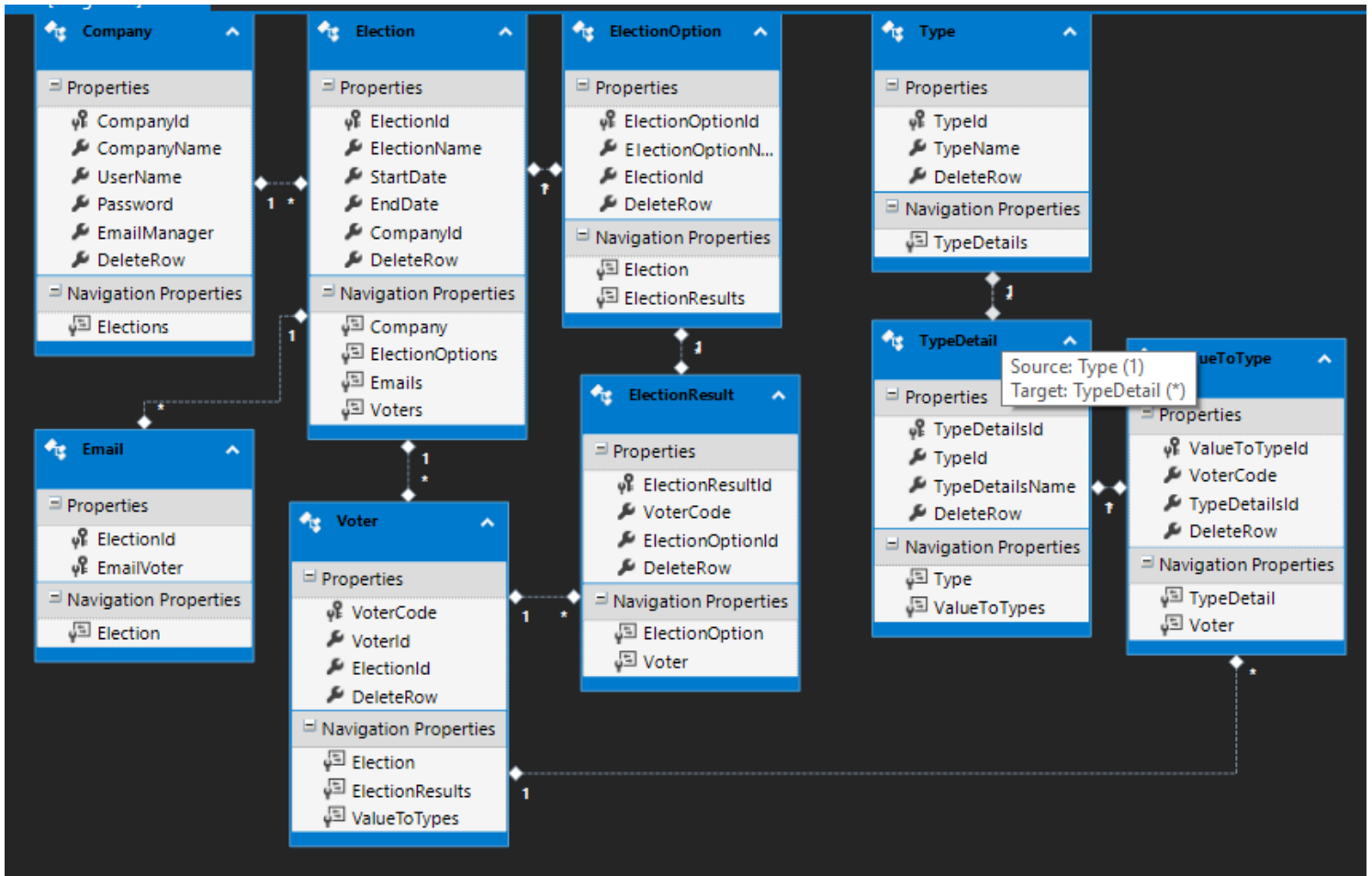


API:



2.5 מבנה נתונים מאוכסנים

2.5.1 תרשים SQL



2.5.2 תיאור הטבלאות

Company – מחלקת חברה המתארת חברה הנרשמת לאתר. באפשרות כל חברה לפתוח מס' בחירות, ובכל מערכת בחירות ניתן לתת רשימת משתתפים שונה.

Election – מחלקת בחירות. מתארת כל מערכת בחירות בנפרד עם כל פרטיה. כל מערכת בחירות שייכת לחברה מסויימת.

ElectionOption – מחלקת אופציות בחירה. בכל מערכת בחירות יש כמה אפשרויות בחירה. כל אפשרות בחירה שייכת למערכת בחירות מסויימת.

Voter – מחלקת בוחרים. מכילה את כל הפרטים של בוחר מסויים. כל בוחר שייך למערכת בחירות מסויימת.

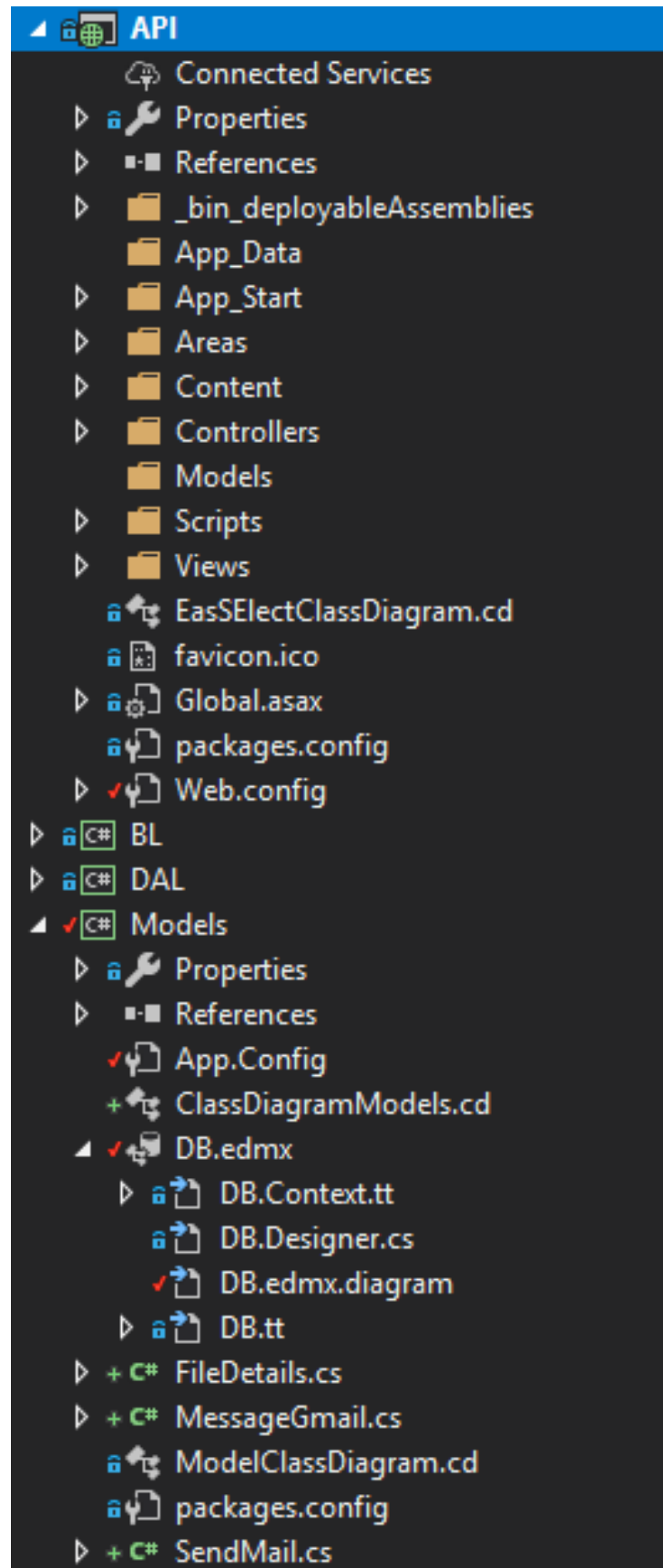
ValueToType – מחלקת ערכים לסיווגים – מכילה עבור כל אחד מבוחרים את כל פרטי הסיווג שלו. כל ערך לסיווג שייך לבוחר מסויים, ולכל בוחר יש כמה ערכים לסיווגים, ע"פ כמות הסיווגים שהגדיר מנהל הבחירות.

TypeDetail – מחלקת פרטי סיווג: פרטי האפשרויות של כל אחד מן הסיווגים. כל פריט סיווג שייך לסוג מסויים.

Type – מחלקת סיווגים – מכילה את כותרות הסיווגים שאותם הכניס מנהל הבחירות.

ElectionResult – מחלקת תוצאות בחירות. עבור כל מערכת בחירות מכילה את התוצאות – ע"פ קוד בוחר – את מה הוא בחר.

Emails – מחלקת כתובות אימייל הבוחרים, כדי לשמור על חשאויות הבחירות כתובת האימייל נשמרת בנפרד מפרטי המצביע, וכך לא נדע למי שייכת הבחירה.



2.6 תוכן הפרויקט

2.6.1 תיאור המחלקות

מחלקות ב-client:

Company - מחלקה המכילה פרטי חברה.

```
export class Company
{
    CompanyId:number
    CompanyName:string
    UserName:string
    Password:string
    EmailManager:string
    DeleteRow:boolean
}
```

Election - מחלקה המכילה פרטי בחירות.

```
export class Election{
    ElectionId:number
    ElectionName:string
    StartDate:Date
    EndDate:Date
    CompanyId:number
    DeleteRow:boolean
}
```

ElectionOption - מחלקה המכילה אופציות הבחירה.

```
export class ElectionOption{
    ElectionOptionId:number
    ElectionOptionName:string
    ElectionId:number
    DeleteRow:boolean
}
```

ElectionResult - מחלקה את תוצאות הבחירה: באיזה אופציה בחר הבוחר.

```
export class ElectionResult{
    VoterId:number
    ElectionOptionId:number
}
```

מחלקות ב-server:

SendMail - מחלקה המכילה פרטי שולח המייל, ומטפלת בשליחת מייל.

```
public class SendMail
{
    ElectionsDBEntities DB = new ElectionsDBEntities();
    2 references
    public string SenderName { get; set; }
    2 references
    public string SenderEmailId { get; set; }
    2 references
    public string Password { get; set; }
    3 references
    public MailAddress FromAddress { get; set; }
    1 reference
    public SendMail(string senderName, string emailId, string password) ...
    // פונקציה השולחת מייל
    1 reference
    public bool SendEmail(MessageGmail message) ...
    // פונקציה המיצרת את הודעת המייל
    1 reference
    private MailMessage createMessage(MessageGmail message) ...
    // gmail-התחממות
    // SmtP ע"י הפרוטוקול
    1 reference
    private SmtPClient createClient() ...
}
```

MessageGmail - מחלקה המכילה את פרטי הודעת המייל.

```
public class MessageGmail
{
    2 references
    public string sendTo { get; set; }
    2 references
    public string Subject { get; set; }
    2 references
    public string Body { get; set; }
    1 reference
    public bool IsBodyHtml { get; set; } = false;
}
```

```
public class FileDetails
{
    1 reference
    public string FilePath { get; set; }
    1 reference
    public long ElectionId { get; set; }
}
```

FileDetails - מחלקה המכילה את פרטי קובץ ה-XL: נתיב הקובץ ולאזה בחירות הוא מתייחס.

CompanyController - מחלקה
המטפלת בהוספת חברה/
מחיקה ועדכון, וכן הרשמה
לאתר.

```
public class CompanyController : ApiController
{
    CompanyBL CompanyBL = new CompanyBL();
    1 reference
    public long AddNewCompany(models.Company company)...
    [HttpGet]
    [Route("api/company/login/{userName}/{password}")]
    0 references
    public bool Login(string userName, string password)...
    [HttpPost]
    [Route("api/company/signUp")]
    // הוספת חברה
    0 references
    public long SignUp(models.Company company)...
    // עדכון פרטי חברה
    0 references
    public void Update(int companyId)...
    // מחיקת חברה
    0 references
    public void Delete(int companyId)...
    0 references
    public void Options()...
```

ElectionController - מחלקה
המטפלת בהוספת/ מחיקה/
עדכון של הבחירות.

מחלקה זאת מכילה את
פונקציית שליחת המייל לכלל
הבוחרים עם לינק לאתר
הבחירות.

```
public class ElectionController : ApiController
{
    ElectionBL ElectionBL = new ElectionBL();
    [HttpPost]
    [Route("api/election/addElection")]
    // הוספת בחירות
    0 references
    public long AddNewElection(models.Election newElection)...
    // מחיקת בחירות
    0 references
    public void Delete(int electionId)...
    // עדכון פרטי בחירות
    0 references
    public void Update(int electionId)...
    // פונקציה שולחת מייל לבוחרים עם לינק לאתר
    [HttpPost]
    [Route("api/election/sendMail")]
    0 references
    public bool SendLinkEmail(int electionId)
    {
        return ElectionBL.SendLink(electionId);
    }
}
```

ElectionController - מחלקה
המטפלת בהוספת/ מחיקה/
עדכון של אופציות הבחירה.

```
public class ElectionOptionController : ApiController
{
    ElectionOptionBL ElectionOptionBL = new ElectionOptionBL();
    [HttpPost]
    [Route("api/electionOption/addElectionOption")]
    // הוספת אופציית בחירה
    0 references
    public long AddNewElectionOption(models.ElectionOption electionOption)
    {
        return ElectionOptionBL.AddNewElectionOption(electionOption);
    }
    // מחיקת אופציית בחירה
    0 references
    public void Delete(int electionOptionId)...
    // עדכון אופצייה
    0 references
    public void Update(int electionOptionId)...
```


GeneralBL - מחלקה המכילה את אובייקטי כל מחלקות ה-BL, ופונקציות השייכות לכלל המחלקות ולא למחלקה מסויימת.

```
public class GeneralBL
{
    CompanyBL CompanyBL = new CompanyBL();
    ElectionBL ElectionBL = new ElectionBL();
    ElectionOptionBL ElectionOptionBL = new ElectionOptionBL();
    ElectionResultBL ElectionResultBL = new ElectionResultBL();
    TypeBL TypeBL = new TypeBL();
    TypeDetailsBL TypeDetailsBL = new TypeDetailsBL();
    ValueToTypeBL ValueToTypeBL = new ValueToTypeBL();
    VoterBL VoterBL = new VoterBL();

    // נתוני הבוחרים מקובץ האקסל, כל אחד לטבלה המתאימה, סוגים, פרטי סוגים, בוחרים וערכים לסווגים
    1 reference
    public int LoadDataVoters(string path, long electionId)
    {
        // בדיקה אם נתיב הקובץ חוקי
        bool possiblePath = path.IndexOfAny(Path.GetInvalidPathChars()) == -1;
        if (possiblePath==false)
            return 0;
        using (var reader = new StreamReader(path, Encoding.Default))...
        return 1;
    }

    // מציאת מילה מתוך המחרוזת המבטאת סוכ או פריט סווג
    2 references
    private string SearchWord(string[] values, int j,out int tmp)...
```

CompanyBL - מחלקה
המטפלת בחברות
הקיימות וביצירת חדשות.

```
4 references
public class CompanyBL
{
    CompanyDal CompanyDal = new CompanyDal();
    1 reference
    public long AddNewCompany(models.Company company)
    {
        return CompanyDal.AddNewCompany(company);
    }

    1 reference
    public bool Login(string userName, string password)
    {
        return CompanyDal.Login(userName, password);
    }
}
```

CompanyBL - מחלקה המטפלת בבחירות הקיימות וביצירת חדשות. וכן קוראת לשליחת המייל לבוחרים.

```

public class ElectionBL
{
    ElectionDal ElectionDal = new ElectionDal();
    1 reference
    public long AddNewElection(models.Election newElection)
    {
        return ElectionDal.AddNewElection(newElection);
    }
    // פוקציה הקוראת לפנקציה לשליחת המייל לבוחרים
    1 reference
    public bool SendLink(int electionId)

```

VoterBL - מחלקה המטפלת בבוחרים, מוסיפה, בודקת אם קיים וכו'.

```

2 references
public class VoterBL
{
    VoterDal VoterDal = new VoterDal();
    1 reference
    public void AddNewVoter(string voterId, long electionId)
    {
        int VoterId = int.Parse(voterId);
        Models.Voter newVoter = new Models.Voter()
        { ElectionId = electionId, VoterId = VoterId, Address="" };
        VoterDal.AddNewVoter(newVoter);
    }

    1 reference
    public bool IsVoterExists(int voterId, long electionId)
    {
        return VoterDal.IsVoterExists(voterId, electionId);
    }

    1 reference
    public long GetCodeVoterById(string voterId1, long electionId)
    {
        int voterId = int.Parse(voterId1);
        return VoterDal.GetCodeVoterById(voterId, electionId);
    }
}

```

2.6.2 תיאור הפונקציות

VoterBL - מחלקה
המטפלת בבחורים,
מוסיפה, בודקת אם
קיים וכו'.

```
public bool SendResults(int electionId)
{
    bool mailSend = true;
    Company company = ElectionResultDal.GetCompanyDetailsByElectionId(electionId);
    string companyName = company.CompanyName;
    string EmailManager = company.EmailManager;
    string password = company.Password;
    Models.SendMail sendMail = new SendMail(companyName, EmailManager, password);
    //להביא את רשימת המיילים של כל הבחורים
    List<Email> emailVoters = EmailsDal.GetAllEmails(electionId);
    for (int i = 0; i < emailVoters.Count; i++)
    {
        //מבצע את השליחה
        mailSend = sendMail.SendEmail(new MessageGmail(...));
        if (mailSend == false)
        {
            return false;
        }
    }
    return mailSend;
}
```

-LoadDataVoters

פונקציה המקבלת נתיב
לקובץ האקסל וקוד בחירות
המייצג את הבחירות אליהם
מתייחס הקובץ.

הפונקציה קוראת את הקובץ
בוחר אחר שורה, שכל
שורה מבטאת בוחר אחד,
ומכילה את פרטיו- ע"פ
הסיווגים שיש בטבלת
האקסל.

בסופה של הפונקציה
מתמלאות טבלאות: סוגים,
פרטי סוגים, ערכים
לסיווגים, בוחרים, וטבלת
המיילים.

```
public int LoadDataVoters(string path, long electionId)
{
    //בדיקה אם נתיב הקובץ חוקי
    bool possiblePath = path.IndexOfAny(Path.GetInvalidPathChars()) == -1;
    if (possiblePath == false)
        return 0;
    using (var reader = new StreamReader(path, Encoding.Default))
    {
        List<string> types = new List<string>();
        for (int i = 0, countWord = 0; !reader.EndOfStream; i++, countWord = 0)
        {
            var line = reader.ReadLine();
            var values = line.Split(';');
            //אם זה הכותרת שמביעה קריטריונים לסוגי סוגים
            if (i == 0)
            {
                for (int j = values[0].IndexOf(',') + 1; j < values[0].Length; j++)
                {
                    //אם זה השורות בטבלה שמביעות בוחר
                }
            }
            else
            {
                string typeDetailName = "";
                string voterId = "";
                for (int j = 0; j < values[0].Length; j++)
                {
                    //אם מדובר על קוד טביעת אצבע
                    if (j == 0)
                    {
                        //אם זה ערך של פרטי סוג
                    }
                    else
                    {
                    }
                }
            }
        }
    }
}
```

מדריך למשתמש

2.7 הוראות כלליות לשימוש באתר

מדריך למנהל בחירות:

המנהל נכנס למערכת ע"י שם משמש וסיסמא, באפשרותו ליצור בחירות חדשות, להכניס משתתפים לבחירות קיימות ולצפות בתוצאות בחירות קודמות. הוא יכול לשלוח לינק לכל משתתפי הבחירות המקשר אותם לבחירות האלו. בזמן בחירות הוא יכול לצפות בזמן אמת בתוצאות הבחירות – מי הצביע ומאילו סיווגים הוא. הוא לא יכול לצפות עד סיום הבחירות בתוצאות הבחירות עצמן.

מדריך לבוחר:

הבוחר לוחץ על הלינק שנשלח לו ע"י מנהל הבחירות, מזדהה באמצעות טביעת האצבע בקורא טביעת האצבע שלו, מחליט באיזו אופציה לבחור ומאשר/לא מאשר את הפרטים שאותם ביקש לדעת מנהל הבחירות. בסיום יום הבחירות ישלח אליו מייל עם תוצאות הבחירות.

מסך כניסה-login

שם משתמש

סיסמא

הכנס

מסך הרשמה-SignUp

שם חברה

שם משתמש

מיינ'

סיסמא

אימות סיסמא

הרשם

מסך בחירות חדשות- המנהל יכול להוסיף/ לעדכן בחירות, ולהוסיף/ לעדכן אופציות בחירה. המנהל טוען את קובץ פרטי הבוחרים, ושולח להם את הלינק לאתר הבחירות.

EasElect | בחירות חדשות | בחירות קיימות | בחירות עכשיו

- 1 הוספת בחירות**
 - שם בחירות
 - תאריך התחלה
 - שעת התחלה
 - תאריך סיום
 - שעת סיום
 - הבא
- 2 הוספת אופציות בחירה**
 - אופציה
 - הוסף אופציה
 - הבא
- 3 טען פרטי בוחרים**

פרטי בוחרים יופיעו בקובץ אקסל שישמר בשם קובץ csv. בעמודה הראשונה (A) יופיעו קוד סביעת האצבע של הבוחרים. בעמודה האחרונה יופיעו כתובות המייל שלהם. בעמודה השלישית יופיעו מסמכים שונים. שכליהם אחת מעניינים מסוג את הבוחרים. השורה הראשונה תכיל את כותרות הסיווגים.

לחץ כאן כדי לצפות בקובץ לדוגמה

שלח הודעה לבוחרים

במסך בחירות קיימות- המנהל יכול לעדכן בחירות כל עוד הם לא התחילו, ולהוסיף/ לעדכן אופציות בחירה כנ"ל. וכן להוסיף או למחוק בוחר מבחירות אלו.

EasElect | בחירות חדשות | בחירות קיימות | בחירות עכשיו

שומר שינויים

22 כנסת
23 כנסת
24 כנסת
25 כנסת

פרטי בוחרים:

מבחן משפחתי	גיל	כתובת	מין	מיקום
רווק	18	חיפה	נקבה	מיקום
נשוי	30	ירושלים	זכר	מיקום
גרדש	75	אליכין	זכר	מיקום
אלמן	45	תל אביב	זכר	מיקום
רווק	36	אופקים	נקבה	מיקום
נשוי	20	ירושלים	נקבה	מיקום
גרדש	28	באר שבע	זכר	מיקום
גרדש	89	נתיבות	נקבה	מיקום
אלמן	55	חיפה	זכר	מיקום
רווק	26	ראש העין	נקבה	מיקום
נשוי	74	כרמיאל	זכר	מיקום

פרטי הבחירות, לחץ לשינוי:

שם בחירות: כנסת 24

תאריך התחלה: 02/03/2020

שעת התחלה: 07:00

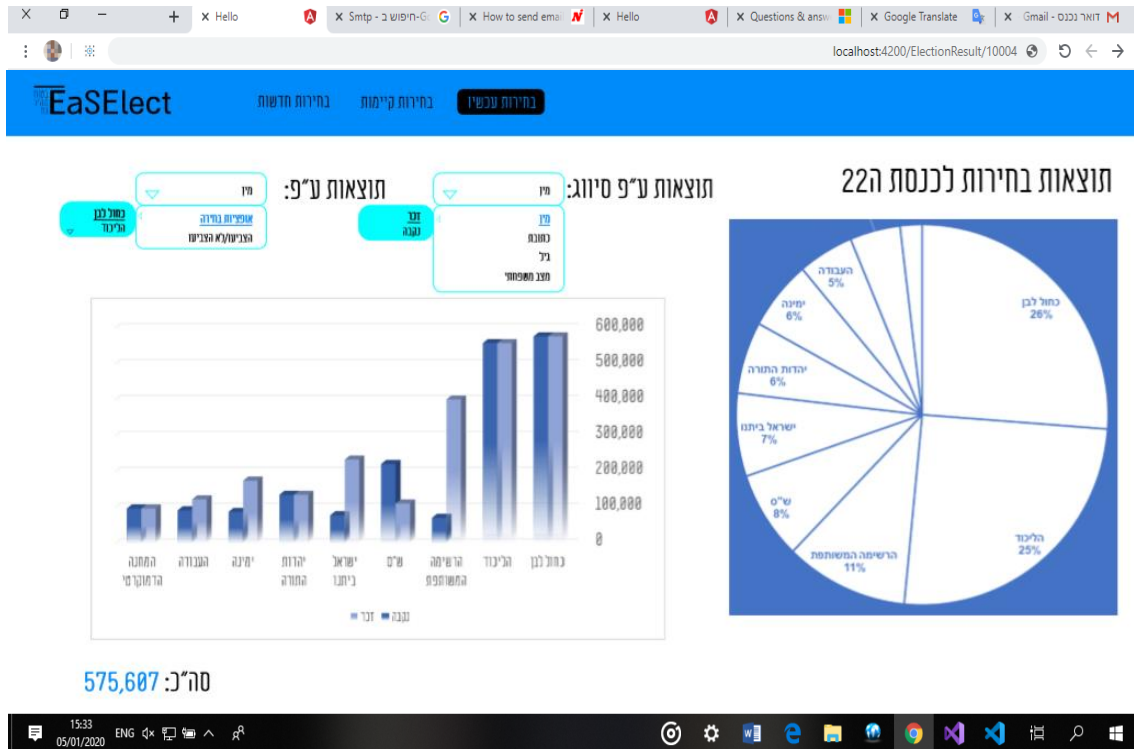
תאריך סיום: 02/03/2020

שעת סיום: 22:00

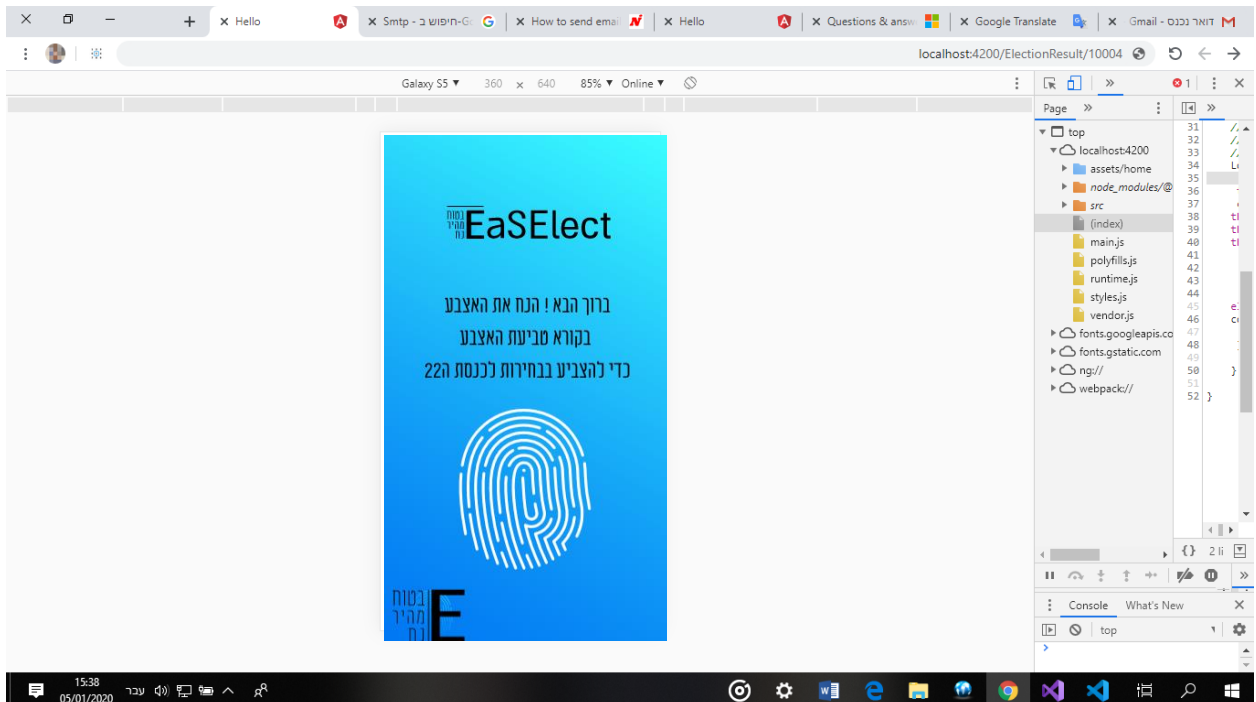
אופציות הבחירה:

- הליכוד בהנהגת בנימין נתניהו לרא
- כחול לבן בראשות בני גנץ ויאיר לפ
- התאחדות הספרדים שומרי תורה תנ
- יהדות התורה והשבת אגודת ישראל
- הרשימה המשותפת חדש, רע"ם, בר

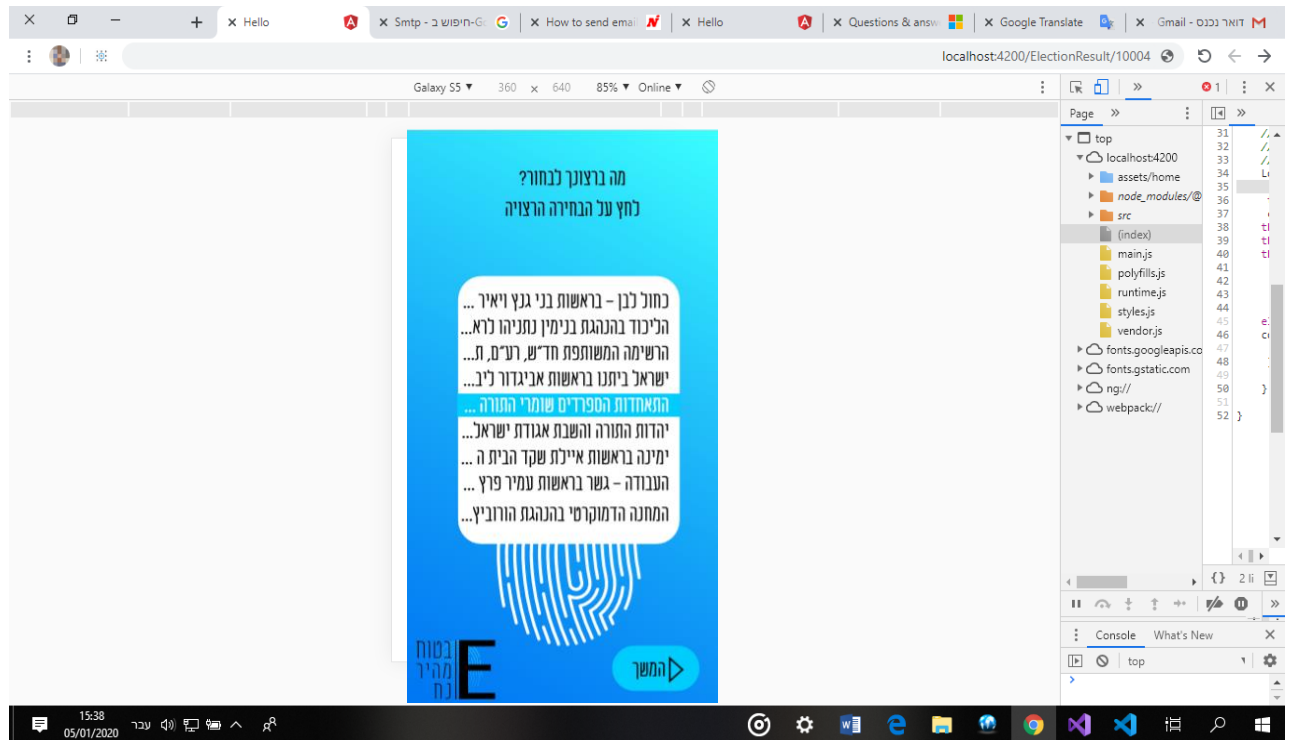
מסך בחירות נוכחיות- המנהל יכול לצפות בנתוני ההצבעה אך לא בתוצאות ההצבעה, על פי סוגים שונים של נתוני הבחורים, שהמנהל ביקש לראות.



מסך קורא טביעת אצבע- הבחור מניח את האצבע על קורא טביעת האצבע, נתוני הבחור נבדקים, האם הוא שייך לבחירות אלו והאם הגיע זמן ההצבעה, במידה שכן, עובר אוטומטי למסך הבא.

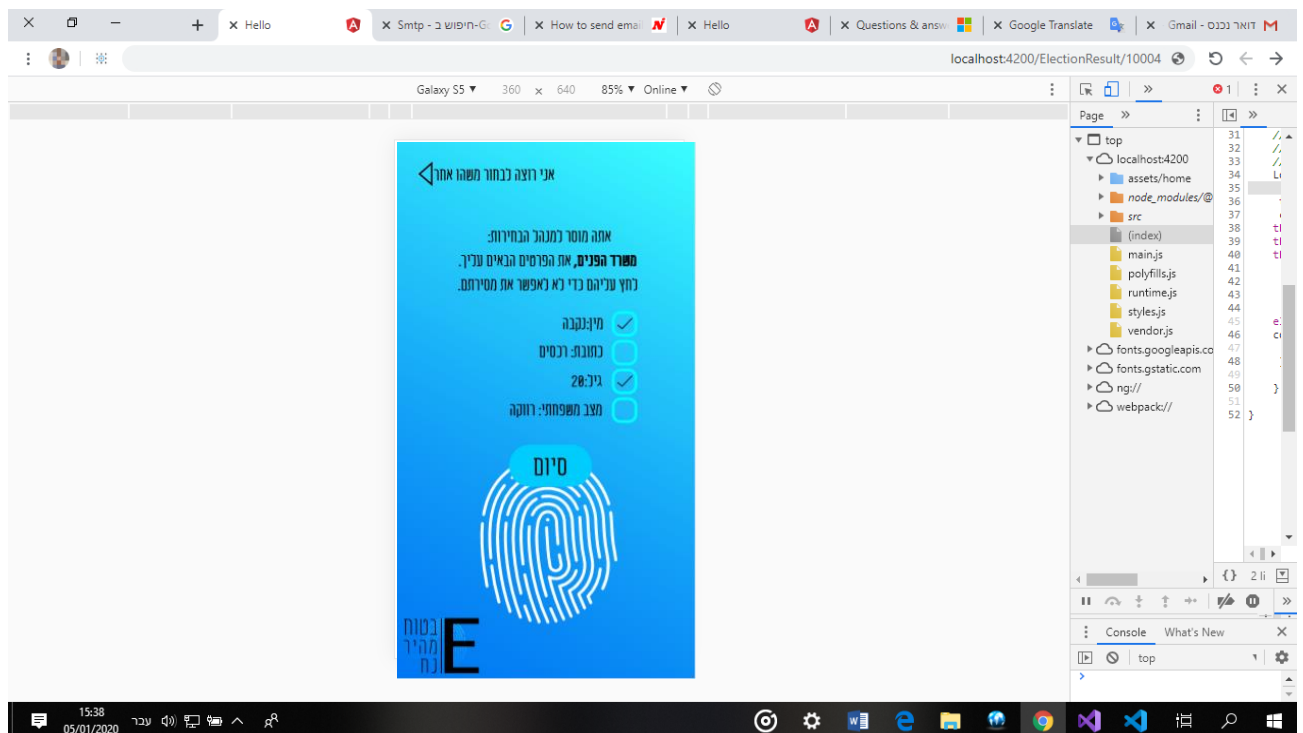


מסך בחירה- המצביע בוחר את בחירתו, ומאשר.



מסך אישור נתוני בוחר- הבוחר מאשר/ לא מאשר את הנתונים בהם יוכל מנהל הבחירות לצפות, ע"פ העדפתו האישית. ובכך מסתיים התהליך.

בסוף הבחירות מקבל הבוחר את תוצאות הבחירות.



3. סיכום ומסקנות

לאחר תקופה ארוכה ופורייה של שעות עבודה רבות ברצוננו להצביע על מספר הישגים ומסקנות שנתקלנו בהם.

פרויקט בהיקף נרחב שכזה העניק לנו ידע רב וניסיון מעשי בפיתוח, הן בצד ה-Server והן בצד ה-Client, תוך שילוב של הרבה מושגים וכלים הן אלו שלמדנו בשיעורים והן אלו שהתחדשו לנו בעקבות כתיבת הפרויקט.

פיתוח הפרויקט מהרעיון הכללי ועד למוצר המוגמר דרש השקעה רבה וידע נרחב, ואכן השקענו רבות בפיתוחו הן בלמידת כלים חדשים והן ביישום הכלים שרכשנו בשנות הלימודים ואכן הצלחנו בעזרת הבורא לבנות ממשק שיממש את הרעיון של ניהול בחירות בצורה ממוחשבת.

האתר נמצא בשלבי פיתוח אולם עדיין אינו גמור באופן סופי, בעקבות כך ייתכן שיתרחשו שינויים שונים ושיפורים נוספים עד להגשת המוצר המוגמר.

ביבליוגרפיה

4.

אתרים בנושא תכנות:

- stackoverflow.com
- docs.microsoft.com
- internet-israel.com
- w3schools.com
- material.angular.io
- freecodecamp.org
- npmjs.com
- github.com