

**RentingGown**

שמלה מושלמת במרחק נגיעה

חוה ברמן ותמר פרישמן | סמינר הרב וולף | אוקטובר 2018



**שמות המגישות**: תמר פרישמן וחוה ברמן

חתימה:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

חתימה:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**שם המנחה**: גב' מורן ערוסי

חתימת המנחה:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**שם רכז המגמה**:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

חתימה:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## תודות

## כותרת 2

אם התמונה שבחרנו לעמוד השער אינה מתאימה לדוח שלך, תוכל להחליף אותה בקלות בתמונה משלך.

פשוט מחק את תמונת מציין המיקום. לאחר מכן, בכרטיסיה 'הוספה', לחץ על 'תמונה' כדי לבחור אחד מהקבצים שלך.

# **תוכן ענינים**

תודות---------------------------------------------------------------------------------------------- 2

תוכן ענינים-----------------------------------------------------------------------------------------3

הצהרה---------------------------------------------------------------------------------------------4

הצעה לפרויקט גמר--------------------------------------------------------------------------------5

אישור הפרויקט-----------------------------------------------------------------------------------

גוף העבודה---------------------------------------------------------------------------------------

# **הצהרת התלמידים/ות**

**שם**:תמר פרישמן ת.ז. 209301290

**שם**: חוה ברמן ת.ז. 207324401

החתומים/ות מטה, מצהירים/ות בזאת כי כל הפרויקט המוגש בספר זה, הינו פרי עבודתנו בלבד, על בסיס הנחייתה של המנחה מורן ערוסי ותוך הסתמכות על לימודינו במכללה והמידע אשר רכשנו תוך כדי העבודה.

חתימת התלמידים/ות:

חוה ברמן: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

תמר פרישמן: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

חתימת המנחה: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

תאריך: 17/07/2018

לכבוד

יחידת הפרויקטים מה״ט

הצעה לפרויקט גמר

א. פרטי הסטודנטים

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| תאריך סיום הלימודים | טלפון נייד | כתובת | ת.ז. 9 ספרות | שם הסטודנט |
| תמוז תשע"ח | 0583295951 | אליעזר 5 בני ברק | 209301290 | תמר פרישמן |
| תמוז תשע"ח | 0548413614 | בן זכאי 45 בני ברק | 207324401 | חוה ברמן |

שם המכללה: סמינר הרב וולף ע"י המכללה למנהל ראשון לציון

סמל המכללה:72395

מסלול ההכשרה: הנדסאים

מגמת לימוד: תכנות מחשבים

מקום ביצוע הפרויקט: בסמינר

ב. פרטי המנחה האישי

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מקום עבודה/תפקיד | תואר | טלפון נייד | כתובת | שם המנחה \* |
|  |  | 054222628 |  | מורן ערוסי |

חתימת הסטודנט חתימת המנחה האישי חתימת הגורם המקצועי מטעם מה״ט

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

דרך מנחם בגין 86 תל אביב ת.ד. 36049 מיקוד 67138 טלפון :03-7347521 פקס: 03-7347644

1. שם הפרויקט : Rentingown
2. רקע:
   1. תיאור ורקע כללי:

הפרויקט הינו אתר אינטרנט רספונסיבי המיועד לניהול מאגר שמלות להשכרה.

הפרויקט יתווך בין משכירים, שוכרים ובעלי עסקים המעונינים לקבל מידע אודות היצע וביקוש בשוק זה.

האתר יאפשר גלישה חופשית עבור כלל המשתמשים לחיפוש שמלות עפ"י קריטריונים נבחרים.

כמו"כ, התוכנה תזהה או תאפשר להזין מיקום, ותדרג לגולש את השמלות עפ"י מרחק ואזורים.

בכדי להוסיף שמלה/ות למאגר או לשכור שמלות מן המאגר, יקושר המשתמש לדף תשלום, ורק לאחר מכן יקבל את המידע המתבקש.

האתר יכיל גם "*backoffice*"- ניהול אחורי של המערכת עבור מנהליה, דפי סטטיסטיקות וכו'.

* 1. מטרות המערכת:
* המערכת תתווך בין המעונינים להשכיר שמלות לבין השוכרים.
* המערכת תקל על איתור השמלה המתאימה ביותר עבור הגולש עפ"י קריטריונים מפורטים(גם מבחינת המרחק הפיזי).
* המערכת תספק מידע לבעלי עסקים אודות סגנון השמלות המבוקשות בשוק.
* המערכת תאפשר הסקות סטטסטיות אודות השימוש, מצב הביקוש והשמלות.

1. סקירת מצב קיים בשוק, אילו בעיות קימות.

אתרים דומים קיימים אמנם בשוק, אך ממשק המשתמש שלהם אינו מספק נתוני מרחק בין מיקום הגולש לשמלות המבוקשות, דבר שמקל מאד על איתור השמלה המבוקשת. כמו"כ האתר יספק מידע עבור בעלי עסקים, וינהל מאגר סטטיסטיקות.

1. מה הפרויקט אמור לחדש או לשפר.

האתר יאפשר לבעלי עסקים להתרשם מהמצב הקיים בשוק, ולפעול עפ"י המידע, בהתאם לביקוש והיצע.

האתר יחזיק גם מידע לגבי מצבה הטכני של כל שמלה ושמלה (מספר שימושים, נקיון, מספר הצפיות בכל שמלה וכו')

האתר ינהל בדייקנות את זמני ההשאלה וההחזרה בין משכירים לשוכרים (יגבה קנס עבור איחור )

האתר ינהל סטטסטיקות עבור מנהלי המערכת.

1. דרישות מערכת ופונקציונאליות
   1. דרישות מערכת, סביבת הטמעה ושימוש.

* המערכת תעבור קומפילציה והפצה בסביבת Visual Studio
* בצד השרת המערכת אמורה לרוץ בסביבת שרת אשר מריץ IIS לקבלת בקשות לתצוגת דפי אינטרנט.
* המשתמש יוכל להריץ את האתר בכל מכשיר אשר מותקן עליו דפדפן אינטרנט.
  1. דרישות פונקציונאליות

רשימת דרישות המשתמש מהמערכת:

* בדף הכניסה תהיה למשתמש אפשרות להכנס כמנהל/כמשכיר/כשוכר או כבעל עסק המתענין במידע כללי.כלומר, משכירים ובעלי עסק המעונינים להצטרף למאגר ידרשו להרשם בפעם הראשונה ולשלם, ובפעמים הבאות יאלצו לעבור זיהוי בלבד, שוכרים יצטרפו למאגר רק לאחר שיבחרו שמלות ויעברו את הליך התשלום.
* משתמש המזוהה כמנהל יוכל לצפות ברשימות המשכירים/השוכרים/מצב ההשאלות סטטיסטיקות וכו',וכן לנהל את מאגר המידע (הוספה, הסרה וכו').
* משתמש המזוהה כמשכיר יועבר לדף הוספת שמלות או לדף בקשה להסרה מן המערכת, שתשלח אל מנהל המערכת.
* משתמש רגיל- שוכר יוכל לגלוש ולחפש באופן חופשי תוך שהמערכת תחשב את דרוג המרחקים ממיקומו, ללא הזדהות מוקדמת, רק לאחר בחירת שמלה, יצטרף למאגר השוכרים עד להחזרתה.
* משתמש המזוהה כבעל עסק יועבר לדפי מידע ודיאגרמות בהתאם למידע אותו הוא מעונין לקבל. (עפ"י דרישה).

1. בעיות צפויות במהלך הפיתוח ופתרונות (תפעוליות, טכנולוגיות, עומס ועוד ):
   1. תיאור הבעיות- הללו כפועל יוצא של דרישות המשתמש מהתוכנה.

בעיה 1: עפ"י מה לקבוע את מיקום השוכר בכדי לחשב דרוג מרחק האזורים.

בעיה 2: איך לעדכן את טבלת ההשאלות (ע"י מי יבוצע עדכון החזרת פריט מושאל)

* 1. פתרונות אפשריים. (נא ציין פתרונות אפשריים וחלופות ארכיטקטוניות)

לבעיה 1:

פתרון 1: באופן אוטומטי ע"י זיהוי מיקום המכשיר ממנו מתבצע החיפוש.

פתרון 2: המשתמש יזין את מיקומו באופן ידני.

לבעיה 2:

פתרון 1: כאשר חולפים מס' ימים מתאריך ההשכרה, המערכת תשלח הודעה למשכיר האם חזר, ותעדכן עפ"י תשובתו באופן אוטומטי את הטבלה.,כמו"כ תחייב את המשכיר בקנס במקרה של חריגה.

פתרון 2: לאפשר למשכיר לעדכן את סטטוס השמלה בכוחות עצמו, ובזמן הרצוי לו.

הפתרון הנבחר עבור כל אחת מהבעיות:

הפתרונות אינם סופיים, ויתגבשו במהלך הפיתוח.

לבעיה 1: שני הפתרונות.

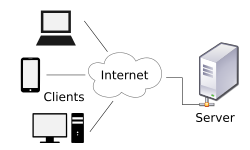
לבעיה 2: פתרון 2

7. פתרון טכנולוגי נבחר:

1. טופולוגית הפתרון-

המערכת מורכבת משרת [IIS], ממשק משתמש בצד הלקוח, DB’s, טווח תקשורתי-אינטרנט. המערכת תיושם ברשת האינטרנט

ממשק משתמש בצד הלקוח: דפדפן אינטרנט כלשהו: chrome, firefox , internet explorer



1. טכנולוגיות בשימוש:

כתיבת צד הלקוח: Html +Css +JavaScript / MVC Razor

בשיתוף bootstrap4 ליצירת אתר רספונסיבי, מעוצב ונעים לעין.

כתיבת צד השרת: שירות אינטרנט באמצעות MVC

מסד נתונים באמצעות Sql-Server

1. שפות הפיתוח:

שפות הפיתוח בצד השרת: #C, MVC

שפות הפיתוח בצד הלקוח: JAVA SCRIPT

CSS

HTML5

1. תיאור הארכיטקטורה הנבחרת- הסבר בכמה מילים מדוע.

הפרויקט נכתב בשפת C# וMVC כיון ששפות אלו נוחות לשימוש-למפתח ולמשתמש.

1. חלוקה לתכניות ומודולים.

BL – לוגיקה

DAL – פונקציות הגישה למסדי הנתונים

Entities - ישויות המוגדרות מול מסד הנתונים.

Client- מסכים להצגת הנתונים.

1. סביבת השרת

לצורך הפרויקט נשתמש בשרת מקומי – IISExpress , המסופק עם סביבת העבודה של Visual Studio .

אם האתר יירכש על ידי לקוח, נעלה אותו לשרת ארוח כלשהו או – Microsoft azure.

1. ממשק המשתמש/לקוח – GUI.

ממשק המשתמש יתקבל כ- HTML ויהיה מוצג באמצעות דפדפן אינטרנט .

1. ממשקים למערכות אחרות / API :

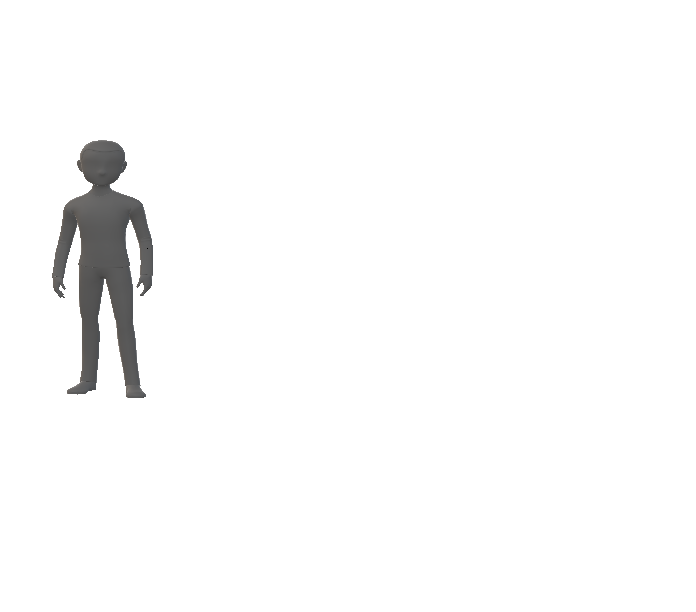
Googlemaps api - לצורך זיהוי מיקום הגלישה.יתכנו התתממשקויות אחרות.

1. שימוש בחבילות תוכנה.
   * Entity framework
   * Bootstrap
   * Mvc
2. שימוש במבני נתונים וארגון קבצים
   1. נא פרט את מבני הנתונים.
      1. **טבלת קטגוריות**: קוד, שם קטגוריה
      2. **טבלת מטרות**: קוד, שם מטרה
      3. **טבלת ערים**: קוד, שם עיר
      4. **טבלת משתמשים**: קוד, ת.ז., שם פרטי ,שם משפחה , עיר, רחוב, מספר בית, טלפון, תאריך ארוסין, תאריך חתונה, סטטוס (חתן/כלה)
      5. **טבלת מטלות**: קוד, סטטוס(חתן/ כלה) קטגוריה, מטרה, תאור המטלה, מס' ימים לביצוע לפני החתונה.
      6. **טבלת מטרות למשתמשים**: קוד, משתמש, מטלה, תאריך יעד לביצוע, נדרש,, הושלם, מחיר עלות, תאריך ביצוע
      7. **טבלת עסקים**: קוד, שם עסק, עיר, רחוב, מס' בית, טלפון, תמונה, סטטוס(גמ"ח/מכירה/חנות), מטלה, תאור
      8. שיטת האחסון

הנתונים ישמרו במסד נתונים של sql-server.

תמונות המוצרים יועלו לשרת קבצים יעודי של הלקוח.

1. תרשימי מערכת מרכזיים



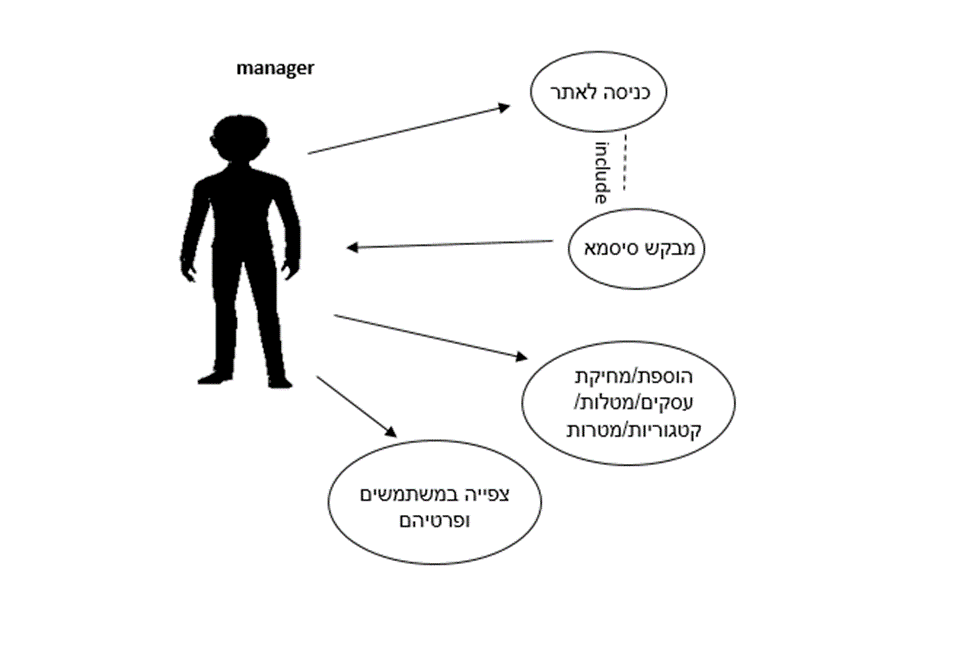
משתמש

נכנס לאפליקציה

מזין נתוני מסלול וזמנים

מקבל מסלול

מבטל נקודה/ מסמן מטלה כמבוצעת



1. תיאור המרכיב האלגוריתמי – חישובי.
   1. איזה בעיה בא לפתור, איך יפתור?

האלגוריתם יקצר למינימום מבחינת זמן, ויביא למקסימום מבחינת יעילות באיתור הפריט המבוקש, באזור המבוקש (לכל גולש יחושב המרחק של כל פריט ממיקומו הנוכחי),כאמור הודות להרחבת הקריטריונים לחיפוש והמידע המפורט על כל שמלה. החיפוש יעשה בהדרגתיות, למן התאריך שהוא הפרמטר החשוב ביותר, ועד לקריטריונים הנוספים, הרצויים על המשתמש.

כמו"כ האלגוריתם יבצע סטטיסטיקות ופילוחי מידע מדויקים ביחס לדרישות האוכלוסיה, גם עפ"י אזורים.נקודה זו תועיל מאד לבעלי עסקים באיסוף מידע על הרגלי ביקוש וצריכה בקרב הצרכנים (עפ"י תחום הענין של המשתמש, המשתמש יזין למערכת את התחום בו הוא מתענין והמערכת תשלוף מידע שיהא עליה לחשב).

* 1. איסוף מידע וניתוחים סטטיסטיים (אנליטיקות)

סטטסטיקות יבוצעו עבור שתי מטרות:

1.עבור מנהל המערכת לצורך תחזוק האתר (מידע על מספר צפיות, היצע וביקוש ועוד)

2 . עבור בעלי עסקים לצורך התרשמות מהמצב הקיים בשוק, (מהם הצבעים או הדגמים המבוקשים ביותר, מהו אחוז האנשים באזור מסוים המעונינים בפריטים מסוימים וכו')

כמו כן, ניתן יהיה לבצע סטטסטיקות על מידת שביעות הרצון של משתמשי האתר עבור ייעול ושיפור.

11. תיאור/התייחסות לנושאי אבטחת מידע

נא לציין אזורים הדורשים אבטחה, כגון: שרת, בקרת גישה לאתר, חשבונות משתמשים, מאגרי מידע וכיצד ניתן מענה.

יש לדאוג לאבטחת השרת מעומסים מופרזים הנגרמים על ידי גורמים זדוניים,

ואת שרת ה- sql server מ- sql injection.

\*יש לאבטח את האתר ממכונה שתירשם בתור משתמשים מדומים,

לכן תהיה לאתר הרשמות מורכבת.

\*משתמשים יזוהו בכניסתם למערכת על ידי ת.ז. וסיסמא, בכדי לשמור על רווחיות האתר, לא יהיה ניתן להוסיף פריטים או לצפות בסטטיסטיקות מבלי לעבור הליך זיהוי זה.

נא ציין מס׳ מקרים ותגובות להם ניתן מענה אבטחתי.

דף סטטיסטיקות יהיה זמין למנהל מערכת בלבד.

לשאר הדפים תהיה גישה עבור כלל המשתמשים.

1. משאבים הנדרשים לפרויקט:
   1. מספר שעות המוקדש לפרויקט, חלוקת עבודה בין חברי הצוות

720 שעות.

חברי הצוות עובדים במשותף.

* 1. ציוד נדרש

מחשב

חיבור לרשת האינטרנט כל משך העבודה.

* 1. תוכנות נדרשות

Sql-server

Visualstudio 2017

ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט

שימוש בweb services חיצוניים

* 1. ספרות ומקורות מידע

Angular Cli

StackOverflow

CodePen

3wSchools

מידע על webservices באמצעות חיפושי גוגל והורדת חומר.

בלוגים ואתרים נוספים.

1. תכנית עבודה ושלבים למימוש הפרויקט:

|  |  |
| --- | --- |
|  | * 1. ייזום הרעיון |
|  | * 1. ניתוח מערכת |
|  | * 1. ניתוח מבנה נתונים |
|  | * 1. איפיון UX – UI |
|  | * 1. כתיבת הלוגיקה העסקית |
|  | * 1. כתיבת ממשק המשתמש |
|  | * 1. עיצוב |
|  | * 1. בדיקות התוכנה |
|  | * 1. התקנה והטמעה |

1. תכנון הבדיקות שיבוצעו
   1. נא פרט בטבלה, בדיקות תהליכיות ברמת משתמש בהן נדרשת המערכת לעמוד .(full Flow)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מספר בדיקה | מקרי הבדיקה | ידנית/ אוטומטית | חשיבות | הערות – תוצאת הבדיקה הרצויה |
| 1. | משתמש נכנס | אוטומטית | גבוהה מאד | אם המשתמש מעונין בהוספת שמלה או בצפיה בסטטיסטיקות הוא יופנה להרשמה ותשלום |
| 2. | משתמש מבקש לבצע השאלה ולתקשר עם השוכר | אוטומטית | גבוהה מאד | יש לחייב רישום מצד במשתמש ולבדוק שההפניה לדף תשלום בוצע והתשלום עבר. |
| 3. | עברו מספר הימים המוקצבים לאחר תאריך ההשאלה של פריט מסוים | אוטומטית או ידנית | גבוהה | יוצר קשר עם המשכיר ויגבה קנס מפרטי החשבון של השוכר השמורים במאגר |
| 4. | משתמש המזוהה כמשכיר שולח בקשה להסרה מן המערכת | אוטומטית או ידנית | גבוהה מאד | יש לבדוק את סטטוס שמלותיו (האם יש להם תאריך כלשהו תפוס) ורק לאחמ"כ להסיר מכל מאגר הנתונים את פרטיו ופרטי השמלות |

1. בקרת גרסאות (version control)

ללא בקרת גרסאות.

חתימת הסטודנט חתימת המנחה האישי

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ג. הערות ראש המגמה במכללה



ד. אישור ראש המגמה:

שם: חתימה: תאריך:

ה. הערות הגורם המקצועי מטעם מה״ט:

ו. אישור הגורם המקצועי מטעם מה״ט:

שם: חתימה: תאריך: .

# **גוף העבודה**

כאשר אדם זקוק לשרותי השכרת שמלות הוא חפץ למצוא את ההתאמה הקרובה ביותר לחלומותיו שתענה במקסימליות על דרישותיו הטכניות, ויכולתיו, תוך דגש על נוחות גבוהה כבר בשלב האיתור, רבים מתייאשים כבר בשלב האיתור, בלי כל קשר לממצאים הקיימים בשוק.

נוסף על כך בכדי לשפר עבורו את נתוני האיתור וההתאמה, יהיה מן הראוי לנהל מאגר שיאסוף מידע מחיפושים ואיסופים קודמים.

ועל סמך ידיעות אלו יוכל הלקוח לבטח לבחור ולקבל את ההתאמה הטובה ביותר עבורו.

אך על מנת שאכן כל דרישותיו תתמלאנה עליו להשקיע זמן רב בבירורים מה –שלא כל אדם יוכל להרשות לעצמו, ומפאת חוסר אותו זמן יזדקק להתפשר ולוותר על חלק מדרישותיו והמה כמובן יבואו לידי ביטוי בהתאם, נוסף על כך, חרף הבירורים הרבים לא יוכל להגיע למירב התוצאות משום שחיפושיו הינם רק בקרב מכריו ותו לא.

אי לכך ובהתאם לזאת ראינו צורך לתכנת אתר שבזמן מינימלי מפנה לתוצאה המתאימה מקסימלית.

התוכנה תכיל ממשקים שונים עבור הרשאות שונות:

• ממשקים לאורח מזדמן האורח יכול לבצע חיפוש והמידע שיופק מחיפושו – יכלול את הנתונים שהזין. ויתאפשר לו לשכור פריט וכן יוכל להרשם למערכת. לא תנתן אופציה להעלות שמלות למאגר או לצפות בנתוני המשכיר, וכן לא לצפות במידע על נתח השוק) אלו יינתנו רק בתנאי שהמשתמש הינו – רשום במערכת(.

• ממשקים למשתמש הרשום במערכת המשתמש רשאי לבצע חיפוש, לצפות בנתוניו האישיים, להעלות שמלות למאגר, לבצע תשלום בנוסף יוכל המשתמש לשנות ולעדכן פרטים אישיים שלו במערכת.

• ממשקים למשתמש הרשום במערכת כבעל עסק למשתמש זה תנתן הרשאה לצפיה בפילוחי מידע וסטטיסטיקות, בנוסף לשאר הגישות הקיימות למשתמש רשום. (למעט העלאת שמלות למאגר).

• ממשקים למנהל מערכת המנהל יוכל לבטל השכרה, למחוק משתמשים, תהיינה לו הרשאות גישה לצפיה במאגר, לחיפוש, לצפיה בפילוחי מידע ואפשרות לשינוי המידע.

האתר מעניק למשתמש חיפוש שמלה על פי קריטריונים לבחירתו ומבחר קטגוריות לסינון,

השוכר-משתמש יכול ליצור עם המשכיר קשר טלפוני\ דרך מייל המשכיר.

יצירת אתר המכיל מידע למשתמשים ובעלי עסק בצורה מאורגנת, מקוטלגת, ממוינת ונוחה

לחיפוש.

האתר יעוצב בצורה נוחה וברורה לשימוש המאפשרת למשתמש איתור שמלות לפי מספר קריטריונים רב ומיון תוצאות החיפוש עפ"י מרחקם ממיקום המשתמש, הוספת שמלות למאגר.

# **1 . מדריך למתכנת:**

1.1 אסטרטגיות טכנולוגיות:

השתמשנו בשפות הבאות בכתיבת הפרויקט:

**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.

**: Html (hypertext markdown lang)** •

זוהי השפה הטבעית ליצירת דפי אינטרנט ברשת. זו שפת תגיות 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 .

JavaScript: **JavaScrip**t • היא שפת תכנות של HTML והאינטרנט. מאפשרת

למפתחים להטביע קטעי סקריפט )"תוכניות קטנות"( בתוך מסמכי HTML . ובכך

להפוך את הדפים לדינמיים יותר. ניתנת לקריאה ע"י דפדפן.

CSS: **CSS (casding style sheets)** •היא שפת עיצוב המגדירה תבנית העיצוב של מסמכי HTML. CSS

למשל מטפלת בגופנים, צבעים, גבולות, שורות, גובה, רוחב, תמונות רקע,

מיקום מתקדם ודברים רבים נוספים.

JQuery: **jQuery** •היא תשתית פיתוח(ספריה) עבור JavaScript ההופכת פעולות

נפוצות ב JavaScript לקלות הרבה יותר, עוזרת לשפר את חווית המשתמשים

באתרים ומטפלת בהבדלים באופן המימוש של JavaScript בין דפדפנים שונים.

כתבנו את הפרויקט בטכנולוגיית **Asp.net MVC .**

**Asp.net MVC** היא הטכנולוגיה המתקדמת ביותר מבית מיקרוסופט לפיתוח אפליקציות

WEB. הטכנולוגיה שמה דגש על פיתוח מהיר, שימוש בארכיטקטורה נכונה והפרדת

שכבות, שילוב ספריות קוד פתוח ותמיכה מובנית בבדיקות אוטומטיות.

**Asp.net MVC** הינה תשתית אשר משמשת לבניית יישומי אינטרנט, ע"י החלת

העקרונות של תבנית מודל –תצוגה- בקר( Design Pattern Model-View-Controller) על

תשתית ASP.NET .

תבנית MVC מהווה כבר שנים רבות תבנית ארכיטקטית חשובה בתחום מדעי

המחשב. התבנית מייצגת דרך יעילה להפרדה בתוך היישום )למשל הפרדת רכיב

הגישה לנתונים מרכיב הטיפול בתצוגה( והיא יעילה לעיצוב יישומי אינטרנט.

ההפרדה המוחלטת של המרכיבים מוסיפה מידת מה של מורכבות ליישום, אולם

התועלת הרבה שמתקבלת בתמורה מצדיקה את המאמץ.

תבנית MVC מפרידה את היישום לשלושה היבטים עיקריים:

11

Model .1 קבוצה של מחלקות המתארות את הנתונים עליהם פועלים. המודלים -

יהוו ברוב המקרים מעין שכבת גישה לנתונים, תוך שימוש בכלים כגון Entity Framework .

View .2 הגדרת אופן ההצגה של ממשק המשתמש. -

. Controller .3 השכבה האחראית על היחסים בין ה Model לבין ה View. מגיב

לקלט מהמשתמש, - מדבר עם המודל, וקובע איזו תצוגה צריך לממש.

השתמשנו בסביבות העבודה הבאות:

✓ :Visual Studio 2017

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 מייקרוסופט ..NET Framework תשתית זו מכילה מספר עצום של ספריות קוד

וטכנולוגיות המקלות עלינו בתהליך יצירת התוכנה.

✓ (localDB)SQL Server 2014 :

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 המרבי 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 שולחנים ותוכנות מבוססות רשת העושות שימוש בבסיס

נתונים המנוהל תחת שרת SQL Server במהלך העבודה עימן.

**1.2 תיאור מבנה הפרויקט:**

ה-Solution שלנו מכיל את הפרויקט העיקרי ממנו מובנה האתר : **RentingGown**.

הפרויקט מחולק לשלוש שכבות עיקריות:

**Models:**

קבוצה של מחלקות הכתובות בשפת C# המתארות את הנתונים עליהם מתבסס המידע באתר.

כל מחלקה כוללת מאפיינים לפי הצורך השימוש בפרויקט על סמך סנכרון הנתונים מתוך ה-Database

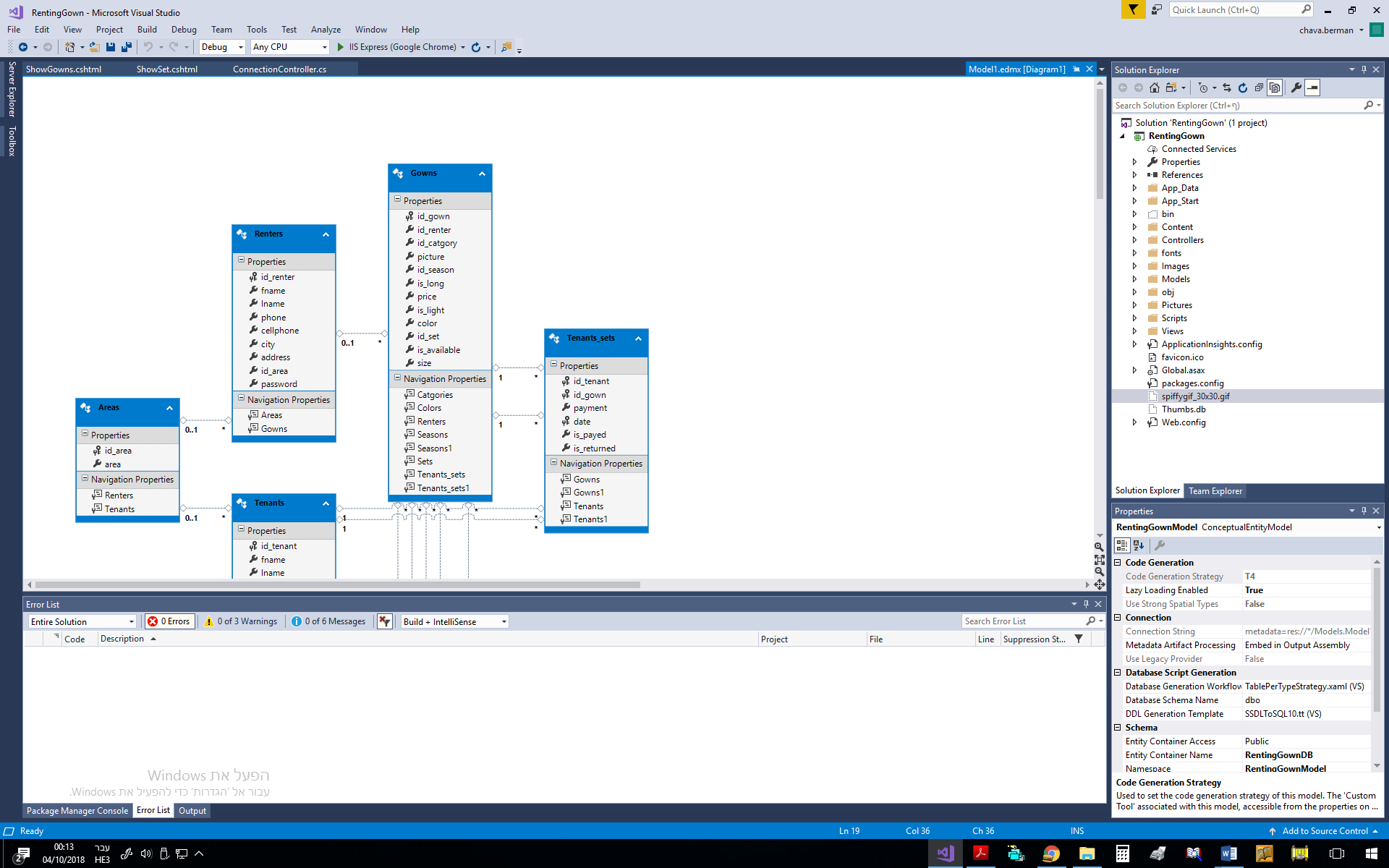
לדוגמא: מחלקת Sets המתארת נתונים עבור סטים של שמלות ומכילה את המאפיינים הבאים: מזהה סט,

כמות מהסט, תמונה.

**Views:**

הגדרת אופן ההצגה של ממשק המשתמש. כל View מציין דף Html אחר בדפדפן, וב-View מגדירים את תוכן וצורת התצוגה.

**Controllers:**

השכבה האחראית על היחסים בין ה-Model לבין ה- View מגיב לקלט מן המשתמש, "מדבר" עם ה- Model, וקובע איזו תצוגה צריך לממש. ה-Controller מקבל נתונים מה-View , עושה חישובים שונים עם שימוש ב-Database וב- Model, ומחזיר נתונים לתצוגת ה- View הרצוי.

**1.3 עקרונות התכנון/הבניה/הניתוח:**

1.3.1 עקרונות תיאורטיים:

* **הורשה:**

בהורשה משתמשים בתכנות מונחה-עצמים, ירושה היא דרך לבסס יחס "סוג של" (is-a ) בין עצמים. המימוש הנפוץ של הגדרה זו הוא בעזרת מחלקות. מחלקה יכולה לקבל בירושה תכונות והתנהגות של מחלקת-האם שלה (נקראת גם "מחלקת-על", "מחלקת אב" או "מחלקת בסיס"). היחסים בין מחלקות מגדירים היררכיה של מחלקות. מנגנון הירושה מסייע במידול של תחומים בדרך דומה לזאת המוגדרת באופן "טבעי" ע"י העוסקים בהם.

לדוגמה: אם ריבוע הוא סוג-של (או מקרה פרטי של) מצולע, ניתן להגדיר יחס ירושה בין המחלקה מרובע למחלקה מצולע; על פי רוב, אובייקט של המחלקה היורשת (מרובע) יתמוך בכל הפעולות המוגדרות על מצולע כלשהו, ובנוסף יתמוך בפעולות נוספות, המיוחדות למרובע. בדומה, ניתן

להגדיר יחס ירושה בין המחלקה ריבוע למחקה מרובע.

*שימושים:*

ירושה מאפשרת להשיג מספר מטרות:

1. שימוש חוזר בקוד. בעזרת ירושה ניתן לכתוב מחלקות בעלות טווח התנהגות נרחב, תוך כתיבת קטעי קוד קטנים יחסית שמרחיבים מחלקה קיימת.
2. מידול של העולם האמיתי באופן דומה לזה המוגדר על ידי האדם. למשל, ניתן 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. למשל משתנה מהמחלקה בעל חיים שנדרש לבצע פעולה מסוג "השמע קול" יבצע פעולות שונות לחלוטין במקרה שהאובייקט הוא מהמחלקה כלב או חתול. ניתן לבצע זאת גם ללא ירושה (בעזרת פקודות תנאי), אך ירושה מאפשרת מימוש מודולרי, גמיש וקל להרחבה.

בפרויקט השתמשנו בהורשה במקרים רבים:

* מחלקה ה-RentingGownDB

יורשת ממחלקת ה DbContext המקושרת ל-System.Data.Entity

ממחלקה זו נוצר מאגר הנתונים ע"י שיטת –Dbfirst עם חיבור ל-entity framework

public partial class RentingGownDB : DbContext

{

public RentingGownDB()

: base("name=RentingGownDB")

{

}

* מחלקות ה-views:

יורשות ממחלקות ה- models

לדוגמא: ה-view של ShowGowns יורש ממחלקת ה-Gowns

@model RentingGown.Models.Gowns

ז

@{

ViewBag.Title = "showGown";

}

<!-- Styles -->

<link href="~/Content/bootstrap.min.css" rel="stylesheet" />

<!-- Bootstrap -->

<link href="~/Content/line-awesome.min.css" rel="stylesheet" />

<!-- Icons -->

<link href="~/Content/style.css" rel="stylesheet" />

<!-- Style -->

<link href="~/Content/responsive.css" rel="stylesheet" />

<!-- Responsive -->

<link href="~/Content/colors/colors.css" rel="stylesheet" />

<!-- color -->

<section>

<div class="block gray ">

<div class="container">

<div class="row">

<div class="col-md-6 col-md-offset-3">

<div class="account-popup-sec static-popup">

<div class="acount-popup register-popup">

<h3>פרטי השמלה</h3>

<form dir="rtl">

<div class="field-form">

<img src="~/Images/@Model.picture" />

</div>

<div class="field-form">

<p><b> מחיר: </b>@Model.price</p>

</div>

<div class="field-form">

<p><b>מידה:</b> @Model.size</p>

</div>

<div class="field-form">

<p><b>עונה:</b> @ViewBag.season</p>

</div>

<div class="field-form">

<p><b>קטגוריה:</b> @ViewBag.catgory</p>

</div>

<div class="side-search-form">

<div class="field">

<a href="/Gowns/AddToTheBasket?id=@Model.id\_gown" class="filter-btn">הוסף לסל הבגדים</a>

* **הפרדת שכבות:**

כל תוכנה/אתר שנפגוש בעולם מבוססת על ארכיטקטורת שכבות הנקראת בשם Three-Tier-Application.

ז"א שלש שכבות במבנה שך DAL-BI-UI.

זוהי תבנית עיצוב בסיסית שמגדירה הפרדת האפליקציה ל:שכבת נתונים, שכבת לוגיקה ושכבת ממשק משתמש.

לתבנית עיצוב זו יתרונות רבים:

1. *תחזוקה:*

ניתן להחליף או לתקן מימוש פנימי של שכבה אחת בארכיטקטורה בלי לשנות שכבה אחרת.

1. *נוחות פיתוח:*

אדם אחד עובד על רכיב בתוכנה, אדם אחר עובד על רכיב אחר, כל עוד שהחתימות זהות ניתן לשלב כוחות ולייעל נתוני פיתוח.

1. *בדיקות:*

תקלה כלשהיא מבודדת בכל שכבה בנפרד, לדוגמא אם לא קיבלנו רשימת נתונים לתצוגה נבדוק קודם את שלב הביניים (שכבת הBL ) אם הנתונים שם תקינים נדע בוודאות שגם ברמת שכבת הDAL הנתונים תקינים וכל שנותר הוא לפתור את התקלה ברמת הUI

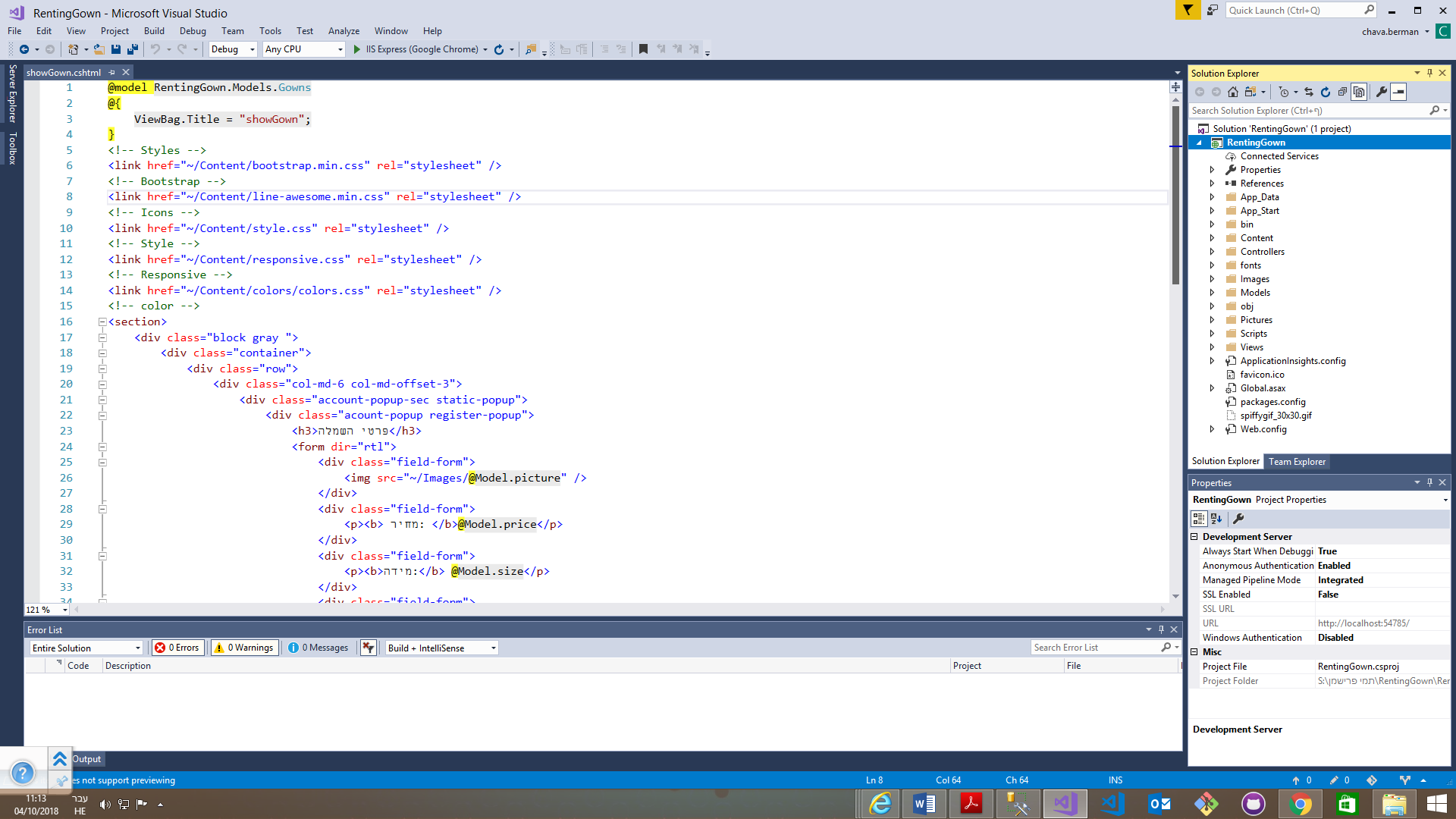
1. *שימוש חוזר:*

נניח שהרכיבים BL + DAL אהובים עלינו ועובדים היטב, ניתן להחליף את פלטפורמת ה UI לטכנולוגיה אחרת במינימום מאמץ.

1. *אבטחה:*

נוח יותר להגדיר Interface-API לשכבה מסוימת בלי לחשוף מבני נתונים, logic או DATA שאינם רלוונטיים למשתמש, כמו גם חסימה בפני האקרים (רלוונטי יותר בטכנולוגיות WEB ) ברמות שונות.

הפרויקט נכתב בשפת C# ומיושם בטכנולוגית mvc. טכנולוגיה זו בנויה על שלושת השכבות:



MVC

שלושת השכבות

* Dal-models
* BI-controllers
* Ui-views

1.3.2 עקרונות תכנותיים:

* **Session:**

שמירת מידע בצד שרת-

הוא משאב אישי ללקוח, המידע אינו משותף ואינו נגיש ללקוחות אחרים. כשאר הלקוח מבקר באתר, השרת מקצה לו אובייקט Session ושומר אותו בזכרון. Session האובייקט נשמר בזכרון כל עוד המשתמש נשאר באתר. אם במשך זמן (מוגדר מראש) האתר ננטש, אובייקט הSession ימחק, הוא ימחק גם כאשר סוגרים את הדפדפן או כאשר נותנים הוראה למחיקתו.

יישום:

לאחר איתור השמלות הנבחרות, מאפשר למשתמש להוסיף את השמלות שיאהב לסל הבגדים., סל זה הוא כמובן אישי לכל גולש ואין אפשרות לשיתוף בחירות בין הגולשים, כמו"כ הסל ישמר גם במעבר בין הדפים.

Gowns gown = db.Gowns.FirstOrDefault(p => p.id\_gown == id);

if (Session["listOfGowns"] == null)

{

Session["listOfGowns"] = new List<Gowns>();

(Session["listOfGowns"] as List<Gowns>).Add(gown);

}

else

{

var q = (Session["listOfGowns"] as List<Gowns>).FirstOrDefault(p => p.id\_gown == gown.id\_gown);

if (q == null)

(Session["listOfGowns"] as List<Gowns>).Add(gown);

}

* **Localstorage:**

שמירת מידע בצד לקוח-

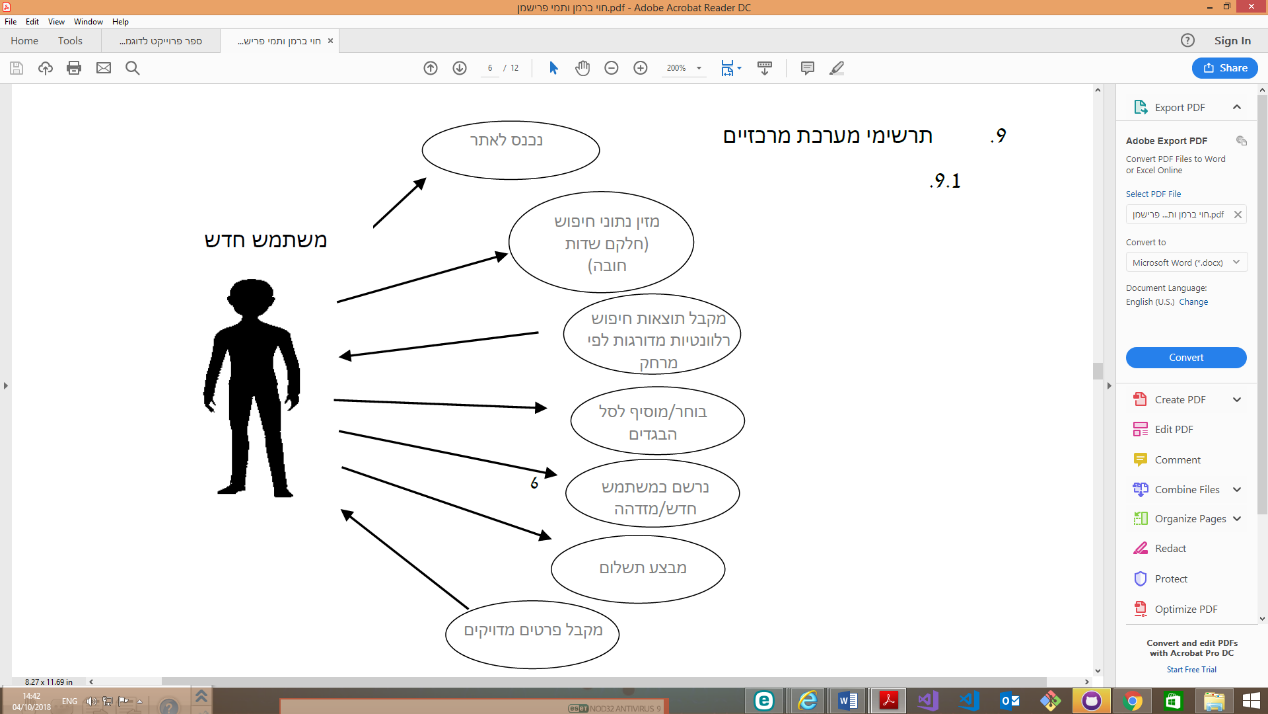
נתונים שנשמרים באמצעות ה- localStorage לא נמחקים עד שנמחוק אותם שהמשתמש ימחוק אותם ידנית, כלומר, גם כאשר הוא יכבה וידליק את המחשב הנתונים האלה ישמרו לנו.

מידע זה נשמר לאורך זמן רב (אפילו שנים).

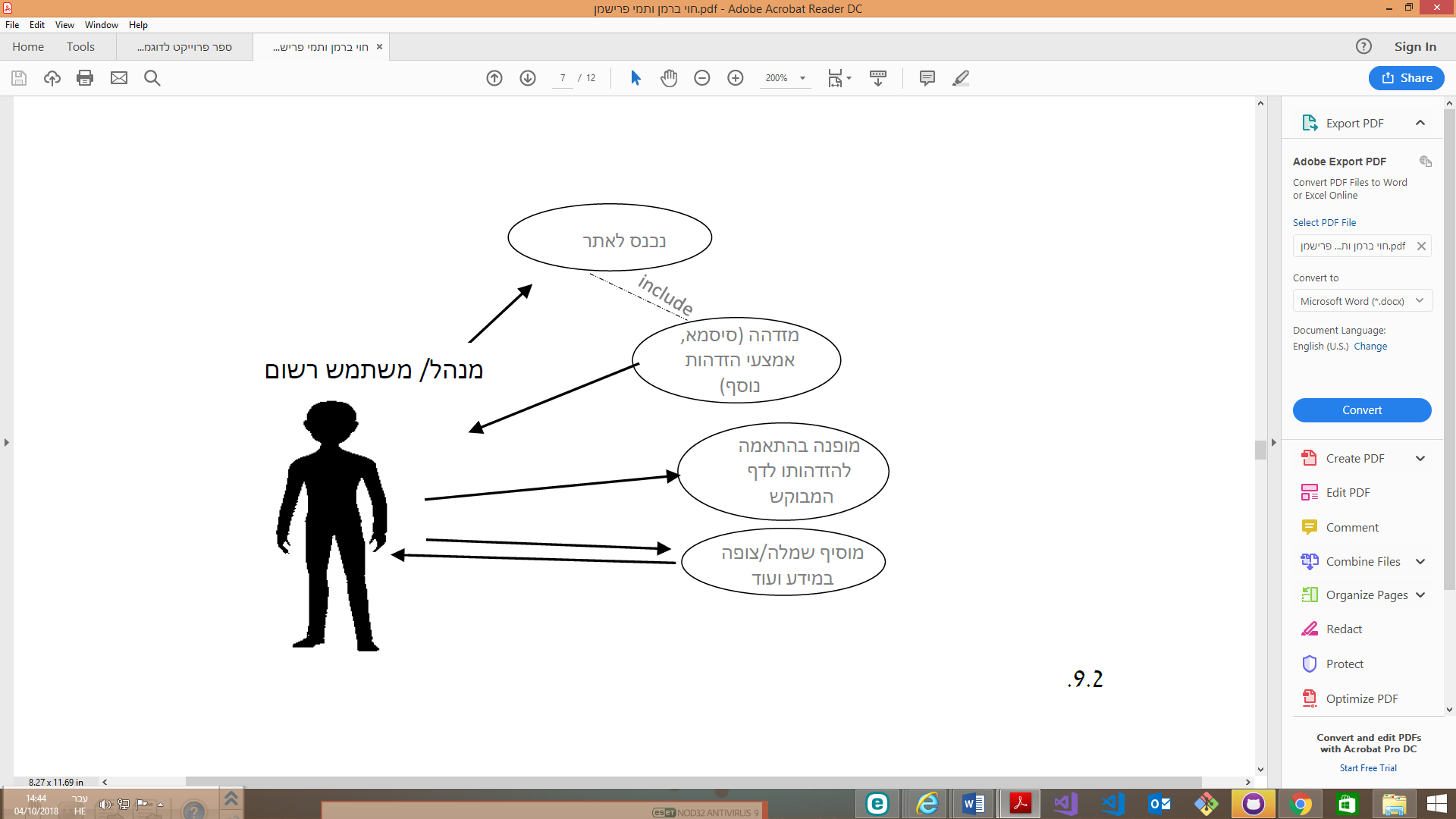
יישום:

בכדי לדרג את השמלות עפ"י מרחק ממיקום הגולש, שמרנו את המידע החוזר מmaps api ב-localstorage עם מספור עוקב, בכדי שנוכל למיין תוצאות.

הנתונים נמחקים מהאחסון בגמר תפקוד הקוד.

**1.4 תרשימים:**

1.4.1תרשימי uml:



1.4.2עץ תהליכים:

משתמש יכול למחוק או לערוך פרטי שמלה במאגר

המשתמש נכנס לאתר

מנהל מזדהה עם סיסמה

המשתמש יכול לערוך חיפוש שמלה עפ"י קריטריונים וקטגוריות.

המנהל מקבל גישה לצפיה ושינוי המאגר, המשתמשים ועוד.

משתמש נכנס ומזדהה עם סיסמה

בעל עסק נכנס ומזדהה עם סיסמה

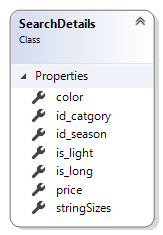
בעל עסק יכול לצפות במידע וניתוחי סטטסיטקות

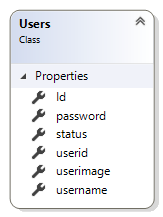
המשתמש יכול להוסיף את בחירתו,לבצע תשלום ולקבל דרכים ליצירת קשר עם המשכיר.

משתמש יכול לערוך שינויים בפרופיל שלו.

משתמש יכול להוסיף שמלה למאגר.

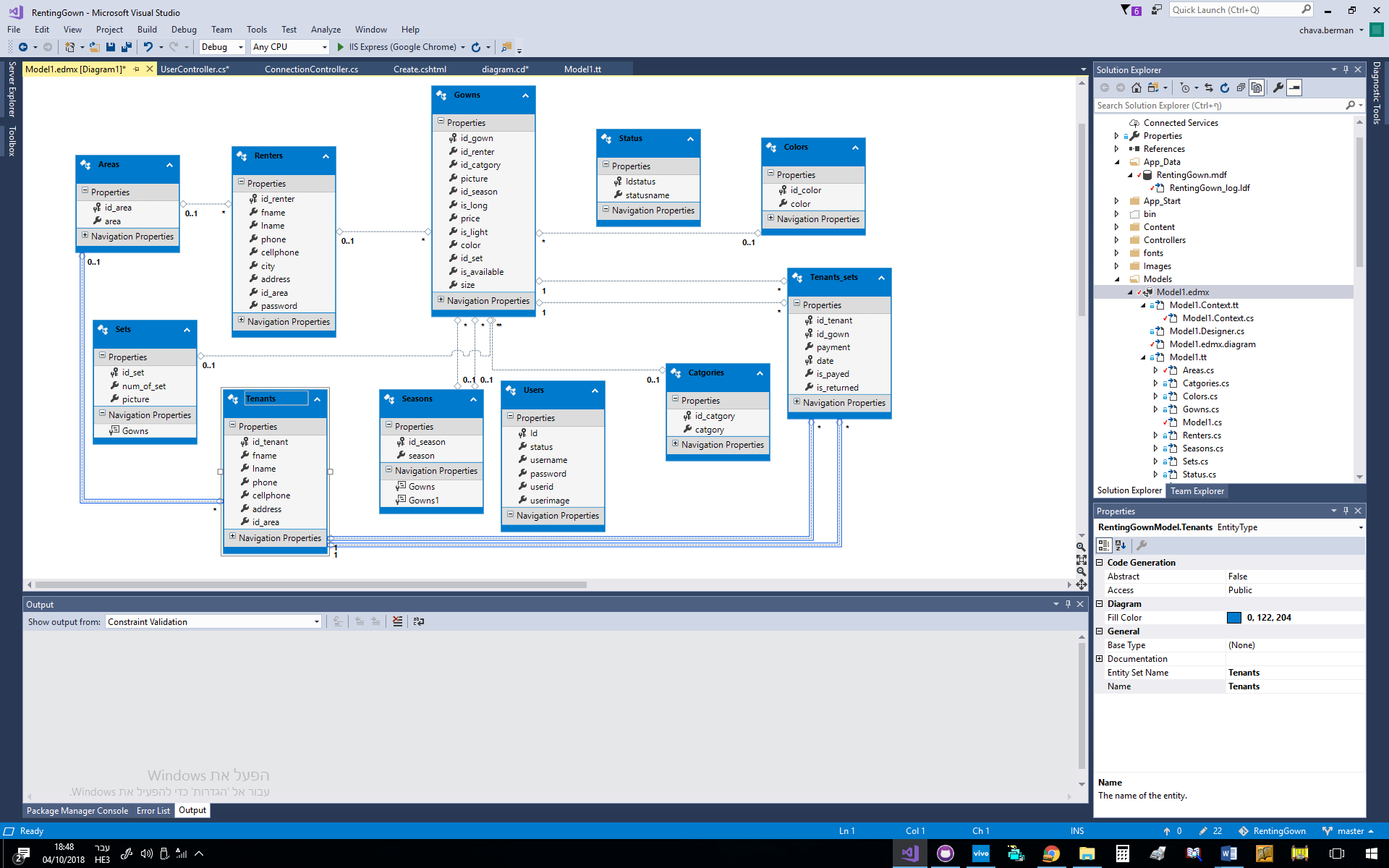
משתמש יכול לערוך שינויים בפרופיל שלו.

1.4.3 תרשים מראה המחלקות:



1.5 מבנה נתונים מאוכסנים:

1.5.1 תרשים SQL:



1.5.2 מבנה קבצים ותיקיות:

1. **קבצי תמונה המועלים ע"י המשתמש-משכיר עבור תצוגה-**

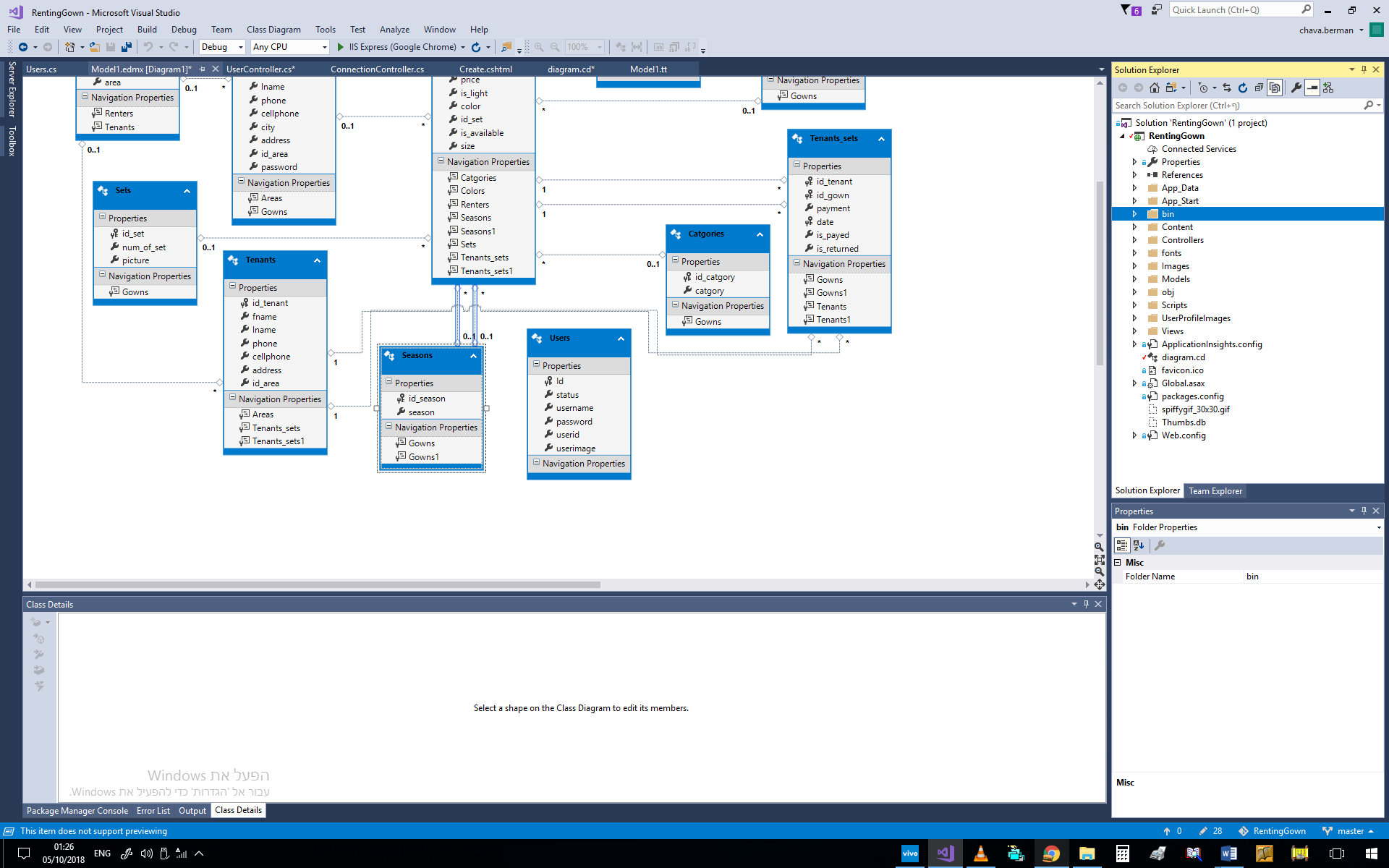
תיקיה המכילה, תמונות של שמלות במאגר שהועלו ע"י המשתמש.

2**. תיקיית תמונת פרופיל המשתמש**-

כאשר משתמש נרשם לאתר יש באפשרותו להוסיף תמונה לפרופיל האישי שלו, תמונות אלו כאמור, נכנסות לתיקיה זו.

3**. תיקיותContent ו- Scripts** -

בתוך תיקיות אלו נמצאים קבצי העיצוב, והסקריפט עבור ממשק המשתמש הויזואלי.



1.קבצי תמונה עבור המשתמש

2. תיקית תמונת פרופיל המשתמש

3. קבצי עיצוב

3. קבצי סקריפט

תקציר

כאשר אדם נצרך להשכרת שמלה הוא חפץ למצוא מוצר שיתאים בצורה המקסימלית לדרישותיו.

אך על מנת שאכן כל דרישותיו תתמלאנה עליו להשקיע זמן רב בבירורים- מה שלא כל אדם יוכל להרשות לעצמו, ואף לא ודאי שאכן יעלו מאמציו פירות. אי לכך ובהתאם לזאת ראינו צורך לפתח אתר – שבזמן מינימלי מפנה לפריט המתאים בצורה מקסימלית.

באתרנו 4 משתמשים אופציונאליים:

* אורח מזדמן- באורח יוכל לבצע חיפוש והמידע שיופק מחיפושו יהיה בהתאם לקריטריונים שהזין. לא תנתן בידו אופציה להעלות שמלות למאגר או לצפות בפילוח מידע, תנן בידו אפשרות לבצע תשלום ולקבל מידע אודות משכירי השמלות בהם הוא מתענין.
* משתמש רשום במערכת- משכיר