



**המכון הממשלתי להכשרה בטכנולוגיה ומדע**

**פרויקט גמר לתואר הנדסאי מחשבים**

**שם הפרויקט: "smartLists.Com"**

**smartLists.Com**

**מוגש ע"י**: אלישבע משה והדס בידני

**שם מנחה:** שרה אבער

פרטי הסטודנטים:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **תאריך סיום לימודים** | **כתובת** | **טלפון** | **ת.ז.** | **שם** |
| 2023 | העליות 20 הוד השרון | 0533156394 | 325751550 | אלישבע משה |
| 2023 | חפץ חיים 5 ראש העין | 0556795633 | 322346529 | הדס בידני |

שם המכללה: סמינר נתיבות הדסה – שלוחת המכללה למנהל ראשל"צ. סמל המכללה: 72395

מסלול ההכשרה: הנדסאים

מגמת לימוד: הנדסת תוכנה מקום ביצוע הפרויקט: בית

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **תואר** | **כתובת** | **טלפון** | **מקום עבודה** | **שם** |
| סמינר נתיבות הדסה | B.Ed | 052-7683535 | פנקס 6 בני ברק | לאה זיכרמן |

פרטי המנחה:

1. שם הפרויקט: יצירת רשימות וניהול תזכורות לאטרקציות.
2. רקע
   1. **תיאור ורקע כללי:**

האתר מקבל יעד. האתר מציג רשימות של "מה להביא" לאטרקציה עבור היעד שנקלט. האתר מנהל תזכורות לפני יציאה לכל אטרקציה. בנוסף לזה האתר מארגן את ימי הטיול בהתאם לאטרקציות הנקלטות כגון: יצירת רשימת "מה להביא" עבור כל אטרקציה, תזכורת ליציאה לפני האטרקציה, ישנה אפשרות ליצור רשימה עצמאית ואפשר גם להוסיף מרשימה הקיימת כבר באתר ולהוסיף פריטים שאינם נמצאים במאגר או ברשימות המוצעות.

* 1. **מטרת המערכת:**
* לארגן למשתמש תיק טיול.
* להוסיף אטרקציה.
* לאפשר יצירה של רשימת פריטים עצמאית.
* להעלות רשימה שיצרו.
* לתזכר לפני כל אטרקציה.
* לצפות ברשימות קיימות.
* להוסיף מרשימות קיימות.

**סקירת המצב הקיים בשוק:**

1. **מה הפרויקט אמור לחדש או לשפר?**

כיום ישנם המון אתרי אטרקציות אבל אין בהם רשימות של מה להביא לאטרקציה . האתר שלנו מכיל מגוון של אטרקציות ולכל אחת מהן מספר רשימות של מה להביא אליה- אותן יצרו מבקרים שביקרו באותה אטרקציה.

**דרישות מערכת, סביבות הטמעה ושימוש.**

* 1. **דרישות פונקציונאליות:**

רשימת דרישות המשתמש מהמערכת:

* המערכת מאפשרת הצגה למשתמש רשימת פריטים לאטרקציה.
* המערכת מאפשרת למשתמש להירשם ולהכניס את פרטיו.
* המערכת מאפשרת ניהול תזכורות שיסייע בזמנים למשתמש.
* המערכת מציגה למשתמש את כל האטרקציות הקיימות לפי הסינון.
* המערכת מאפשרת למשתמש פעולות כמו הוספה\מחיקה\עדכון על פריטים שברשימתו בלבד.
* המערכת מאפשרת לפרסם רשימה שיצר המשתמש.
* המערכת מאפשרת דיווח על חוות דעת פוגעניות.
* המערכת מאפשרת המערכת מאפשרת למשתמש סינונים ברמה גבוהה.
* הכנסת חוות דעת על הרשימות המוצעות.
* המערכת מאפשרת לבעל האתר להוסיף/ לעדכן / למחוק אטרקציה.

1. בעיות צפויות במהלך הפיתוח, ופתרונות תפעוליות, טכנולוגיות, עומס וכו' :
   1. **תיאור הבעיות-**
      * **בעיה 1-** כיצד נמנע פרסום רשימה שלא תקינה?
      * **בעיה 2-** כיצד נמנע גלישת אטרקציה אחת על השנייה בזמניהן?
   2. **פתרונות אפשריים:**
2. **לבעיה 1-**

* כל רשימה שתפורסם באתר תעבור בדיקה של צוות האתר ורק לאחר אישור הרשימה תפורסם.
* הרשימה תפורסם ללא בקרת צוות האתר. ואם יהיו מספר תגובות שהרשימה לא תקינה הרשימה תמחק מהאתר.

1. **לבעיה 2-**

* ניתן אזהרה למשתמש.
* לא לתת אפשרות כזו.

**הפתרון הנבחר עבור כל אחת מהבעיות:**

1. **לבעיה 1-** פתרון מס' 1
2. **לבעיה 2-** פתרון מס' 1
3. **פתרון טכנולוגי נבחר:**

# פתרון טכנולוגי נבחר

## טופולוגית הפתרון

## מימוש ע"י אתר אינטרנטי וע"י תוכנה המותקנת במכשיר חכם.

## דיאגרמה



## טכנולוגיות בשימוש:

צד הלקוח: React ליצירת אתר/אפליקציה רספונסיבי, מעוצב ונעים לעין.

צד השרת: שירות אינטרנט באמצעות Web Api- הפרויקט הינו אתר, וטכנולוגיה זו תואמת את הדרישות.

מסד נתונים באמצעות: SQL Server.

## 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# הוא תנאי הכרחי לשימוש במגוון טכנולוגיות.

### בצד הלקוח:

React – טכנולוגיה מתקדמת המבוקשת מאוד בתעשיה.

## **תיאור הארכיטקטורה הנבחרת**

תבנית הארכיטקטורה הנבחרת היא מודל שלוש השכבות – ארכיטקטורה בה הלוגיקה העסקית של היישום והגישה אל הנתונים מפותחים ומתוחזקים כמודולים נפרדים. מלבד היתרונות הרגילים שלה תהיה לנו כמפתחים האפשרות להפריד בין השכבות ולטפל בכל שכבה בנפרד מבלי לגעת בשכבות האחרות. שיטה זו תקל במהלך פיתוח הפרויקט.

## **חלוקה לתכניות ומודולים**:

### בצד השרת:

### שכבת הלוגיקה העסקית (**BL** - Logic Business) השכבה שאמונה על הלוגיקה של המערכת,

### עוסקת בעיבוד המידע, בחישובים שונים ושליחתו לשכבת התצוגה. בשכבה זו נממש את הפונקציונאליות של המערכת.

### שכבת הנתונים (**DAL** - Layer Access Data) שכבה זו מורכבת ממקור נתונים –מסד הנתונים שלנו, וממערכת תוכנה framework entity אשר תפקידה לקרוא את המידע הנדרש למערכת, לשמור את העדכונים, ולהוסיף מידע חדש או למחוק פרטי מידע קיימים.

### Service-מחלקה שמספקת מידע לשאר המחלקות ומתקשרת עם ה- web api

שימוש ב- רידקס להעברת מידע בין קומפוננטות רבות וסרביסים לחיבור לוואב אייפי חיצוני.

## ממשק המשתמש/לקוח – GUI

שכבת ה GUI מורכבת מדפי HTML שמוצגים למשתמש דרך הדפדפן.

## ממשקים למערכות אחרות / API

## מערכת של שליחת מיילים .

## שימוש בחבילות תוכנה

## Entity Framework, React

1. **שימוש במבנה נתונים וארגון קבצים:**
   1. שיטת האחסון  
      ע"י SQL Server – מסד נתונים
   2. **נפילה\ קריסה\ תמיכה-**

בנוי לתמיכה במס' גדול של קריאות בו זמניות ואין חשש לקריסה ברמה של MySQL Server

הפרויקט.

* 1. **מבנה הנתונים:**

**משתמש**



**אטרקציה**

****

**סוגי משתמש**

****

1. **תרשימי מערכת מרכזיים:**Use Case.

משתמש

מנהל האתר

1. **תיאור המרכיב האלגוריתמי\ חישובי:**

* חישוב מתי נתזכר את המשתמש לפני היציאה לאטרקציה.
* חשוב אורך הזמן שייקח למשתמש להגיע ממקומו הנוכחי עד לאטרקציה.

**איסוף מידע ונתונים סטטיסטיים- אנליטיקות:**

באתר יוצגו הנתונים הסטטיסטיים של: היעד (מדינה) הנבחר ביותר, אטרקציה נבחרת ביותר,

משתמש שהכי הרבה אנשים נעזרו ברשימותיו.

1. **תיאור התייחסות לנושא אבטחת המידע:**

**ציין מס׳ מקרים ותגובות להם ניתן מענה אבטחתי:**

* כניסת משתמש על ידי מייל ושם משתמש, כך שלכל משתמש אזור אישי ופרטי שאינו חשוף למשתמשים האחרים.
* שליחת אישור הרשמה למייל כדי לוודא שהמייל שהוזן אכן שייך לנרשם.
* במקרה שבעת כניסת משתמש הסיסמא אינה תואמת לשם המשתמש שהקיש, המערכת תציג לו הודעת שגיאה ולא תאפשר כניסה. במקרה שהוא שכח סיסמא המערכת תאפשר שינוי סיסמא על ידי שליחת מייל לאימות.
* למשתמש אורח אין אפשרות ליצור רשימה חדשה או להוסיף מקיימת אלא הצגת הרשימות הקיימות בלבד והדפסתן.

1. **משאבים הנדרשים לפרויקט:**
   1. **מספר שעות המוקדש לפרויקט:** 700
   2. **ציוד נדרש:** CPU i5, RAM 8GB, HD SSD ,חיבור אינטרנט.
   3. **תוכנות נדרשות:** Visual studio, Browsers, Visual Studio Code ,Microsoft Sql Server
   4. **ידע חדש שנדרש לצורך ביצוע הפרויקט:**

הרחבת הידע בשפת .React

לימוד בניית שרת בC# **.**

התמצאות רחבה יותר ב mui ובכליה.

* 1. **ספרות ומקורות מידע:**

/https://reactjs.org

<https://reactjs.org/tutorial/tutorial.html>

/https://stackoverflow.com

/https://github.com

/https://www.npmjs.com

/https://www.geeksforgeeks.org

/https://www.react.express

1. **תכנית עבודה ושלבים למימוש הפרויקט:**

ייזום הרעיון וניתוח המערכת- אפריל 2023

ניתוח מבנה נתונים ואפיון UX-UI – מאי 2023

כתיבת הלוגיקה העסקית -

כתיבת ממשק משתמש ועיצוב -

בדיקות התוכנה -

התקנה והטמעה -

1. **תכנון הבדיקות שיבוצעו:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **מספר בדיקה** | **מספר דרישה במסמך אפיון** | **מקרי הבדיקה** | **ידנית\ אוטומטית** | **חשיבות** | **הערות** |
|  |  | התחברות/כניסה | ידנית | גבוהה |  |
|  |  | הצגת אטרקציות לפי יעד | ידנית | גבוהה |  |
|  |  | תזכורות לפני אטרקציה | אוטומטית | גבוהה |  |
|  |  | מניעת גלישת אטרקציה אחת על השנייה | אוטומטית | גבוהה |  |

1. **בקרת גרסאות:** כרגע אין צורך הבקרה תתבצע ע"י **github.**

חתימת הסטודנט:

אלישבע משה

הדס בידני

חתימת המנחה האישי: שרה אבער

#### **הצהרת סטודנט**:

שם סטודנט: אלישבע משה

ת.ז: כ325751550

שם סטודנט: הדס בידני

ת.ז:000000

המכללה בה אני לומד: סמינר אור החיים.

אנו הח"מ מצהירות בזאת כי פרויקט הגמר וספר הפרויקט המצ"ב נעשו על ידינו בלבד.

פרויקט הגמר נעשה על סמך הנושאים שלמדנו במכללה ובאופן עצמאי.

פרויקט הגמר וספר הפרויקט נעשו על בסיס הנחייתה של המנחה האישית.

מקורות המידע בהם השתמשנו לביצוע פרויקט הגמר מצוינים בספר הפרויקט.

אנו מודעות לאחריות שהננו מקבלות על עצמינו ע"י חתמתינו על הצהרה זו שכל הנאמר בה הוא אמת ורק אמת.

חתימת הסטודנט: אלישבע משה

חתימת הסטודנט: הדס בידני

#### **אישור המנחה האישי:**

הריני מאשרת שהפרויקט בוצע בהנחייתי. קראתי את ספר הפרויקט ומצאתי כי הוא מוכן לצורך הגשת הסטודנט לגנה על פרויקט הגמר.

שם: שרה אבער

חתימה: שרה אבער

תאריך: 14.11.2023

תקציר: האתר מביא חדשנות ויצירתיות בנוגע לתכנון טיולים. האתר בעצם מתכנן תיק טיול למשתמש שכולל: רשימות של מה להביא לאטרקציה, תזכורות יציאה לפני כל אטרקציה. הרשימות בנויות מפריטים שאנשים לקחו איתם לאטרקציה. מה שמשקף את ההתארגנות לאטרקציה ונותן הכנה מושלמת ליציאה.

**תודות:**

בראש ובראשונה, תודה לבורא עולם על היכולת והכוחות להוציא מהכוח אל הפועל.

תודה רבה למרכזת לאה זיכרמן על העמל וההשקעה, על הרצון שנצליח ונגיע להכי טוב שאנו יכולות.

תודה רבה למנחה שרה אבער על ההכוונה והכלים, העצות המועילות והאוזן קשבת.

למשפחותינו היקרות שתמכו, עודדו ונתנו לנו כוחות לאורך כל הדרך.

לחברותינו היקרות על האכפתיות וההתעניינות בשלבי הפרויקט.

בהזדמנות זו אנו מודות אחת לשנייה על תהליך נפלא של עמל והשקעה לצד חוויה וסיפוק רב. ללא שיתוף הפעולה

ביננו והרצון להצליח לא היינו מגיעות לתוצאה זו.

**תוכן עניינים:**

הצעת פרויקט 1

הגדרת דרישות ותיאור כללי 13

תיאור כללי 14

ממשקים חיצוניים 17

ממשק אדם ומכונה 18

מבנה נתונים וארגון קבצים 21

תכנון 26

מה הקנה הפרויקט 29

ביבליוגרפיה 31

# פרק 1

## **הגדרת דרישות ותיאור כללי**

**1 הגדרת דרישות ותיאור כללי**

1.1.1 מטרות המערכת

המערכת מציגה מגוון אטרקציות על פי סינון המשתמש, ועוזרת לו להכין רשימת פריטים מה להביא לאטרקציה זו בכך שמציגה לו רשימות של משתמשים שכבר בקרו בה

היקף-

700 שעות. מבנה וארגון

הפרויקט מחולק לצד שרת וצד לקוח.

צד הלקוח כתוב בטכנולוגיית .react צד השרת כתוב ב – C# ומנוהל בצמוד למודל שלושת השכבות

משימות המערכת

 כניסה ורישום משתמשים.

 הכנסת רשימות.

 סינון אטרקציות בהתאם לדרישות המשתמש.

 הוספת אטרקציה חדשה.

 הוספה והצגה של חוות דעת.

 התכתבות בין משתמש למנהל.

 עדכון אטרקציה.

1.2 תיאור חומרת המערכת

מרכיבי המערכת

• כל מחשב או מכשיר חכם שמותקן בו דפדפן ו SERVER SQL

• שרת

1.2.1

1.3 תיאור תוכנת המערכת 1.3.1 כללי

התוכנה מכילה שני חלקים:

• שרת - מערכת השרת מריצה שרת WEB ובסיס נתונים - Sql .Server

• לקוח - מערכת עם דפדפן אינטרנט תיקני.

בבניית המערכת ניתן דגש על נוחות המשתמש.

• נוחות המשתמש – ממשק אחיד, ברור וקל לשימוש ללא ידע מקצועי והבנה רבה במחשבים .

1.3.2 כלי תוכנה לפיתוח המערכת

1.3.2.1 כלים למפתחים

Visual Studio 2022  Visual Studio Code  Sql Server 2019 

פיתוח

שרת

#C בטכנולוגיית API

SQL

לקוח

1.3.2.2

-

-

react - 3css -

- Html5 - דפדפן אינטרנט

1.4 תיאור הפונקציות

1.4.1 הוספת משתמש חדש

הוספת משתמש חדש מתבצע על ידי מילוי טופס רישום עלי ידי המשתמש , בדיקת תקינות הקלט

ושמירתו בטבלת המשתמשים. 1.4.2 כניסת משתמש רשום

הכניסה לאתר מתבצעת על ידי מילוי שם משתמש וסיסמה בטופס כניסה.

1.4.3 חיפוש אטרקציה שנעשה ע"י פונ' סינון ששולפת את האטרקציות ומציגה רק את הרלוונטים לסינון.

1.4.4 הוספת רשימה חדשה

אפשרות הכנסת רשימה ושליחה לאישור המנהל. בצד המנהל תוצג רשימה של הרשימות המחכות

לאישור, וכשהמנהל יאשר או ימחק הן יוסרו מהרשימה.

1.4.5 שליחת הודעה למשתמש.

אפשרות שליחת הודעה למשתמש אחר.

1.4.6 הצגת כל ההודעות.

אפשרות הצגת כל ההודעות של המשתמש.

1.4.7 הוספת חוות דעת

אפשרות הוספת חוות דעת למשתמש רשום

1.5 זרימת המידע במערכת

1.5.1 כללי

* + רישום משתמש חדש.
  + כניסה בתור משתמש קיים.
  + המשתמש יכול לערוך סינון ולקבל את האטרקציה המבוקשת.
  + המשתמש יערוך רשימה של פריטים לאטרקציות שאהב.
  + כניסה לאזור האישי (במידה והמשתמש רשום.(

# פרק 2

## **ממשקים חיצוניים**

## **לא רלוונטי**

# פרק 3

## **ממשק אדם ומכונה**

3.1 כללי

הקפדנו לשמור על ממשק משתמש נעים וברור ככל הניתן על מנת לספק למשתמש חוויה

נעימה, תחושה שאכן התוכנה ענתה על צרכיו, הקלה על עבודתו וחסכה לו זמן יקר.

3.1.1

* דף הבית – הדף הראשי של האתר ,כולל בתוכו קישורים לדף אטרקציות,סטטיסטיקות.
* חיפוש אטרקציה- אפשרות לחיפוש ספציפי והפעלת סינון של מוצרים רלוונטיים.
* סטטיסטיקות- מוצגות ללקוח מספר סטטיסטיקות מעניינות הקשורות לאתר.
* חוות דעת – מוצגות ללקוח חוות דעת של משתמשים אחרים וכן קיימת אפשרות למשתמש רשום להוסיף חוות דעת מעצמו.
* אודות- דף הסבר על האתר .
* הרשמה- דף רישום למשתמש חדש .
* כניסה- דף כניסה למשתמש רשום .
* הרשימות שלי- תצוגה של רשימות המשתמש.
* דף עדכון אטרקציה- אפשרות למנהל לעדכן את פרטי האטרקציה שהכניס לאתר.
* אזור אישי – דף אישי לכל משתמש המציג לו את אטרקציות שאהב, התכתבויות.
* הבקשות שלי– תצוגה של כל בקשות המשתמש.
* ניהול משתמשים– הצגת כל המשתמשים ומספר אופציות לפעולה בידי המנהל.
* הוספת אטרקציה- אפשרות להוספת אטרקציה.
* הוספת פריט ברשימה- אפשרות של משתמש רשום להוסיף פריט שאינו קיים.

3.1.2תרשימי מסכים

משתמש רשום

משתמש חדש

דף הבית

כניסה/הרשמה

יצירת קשר

אודות

סטטיסטיקות

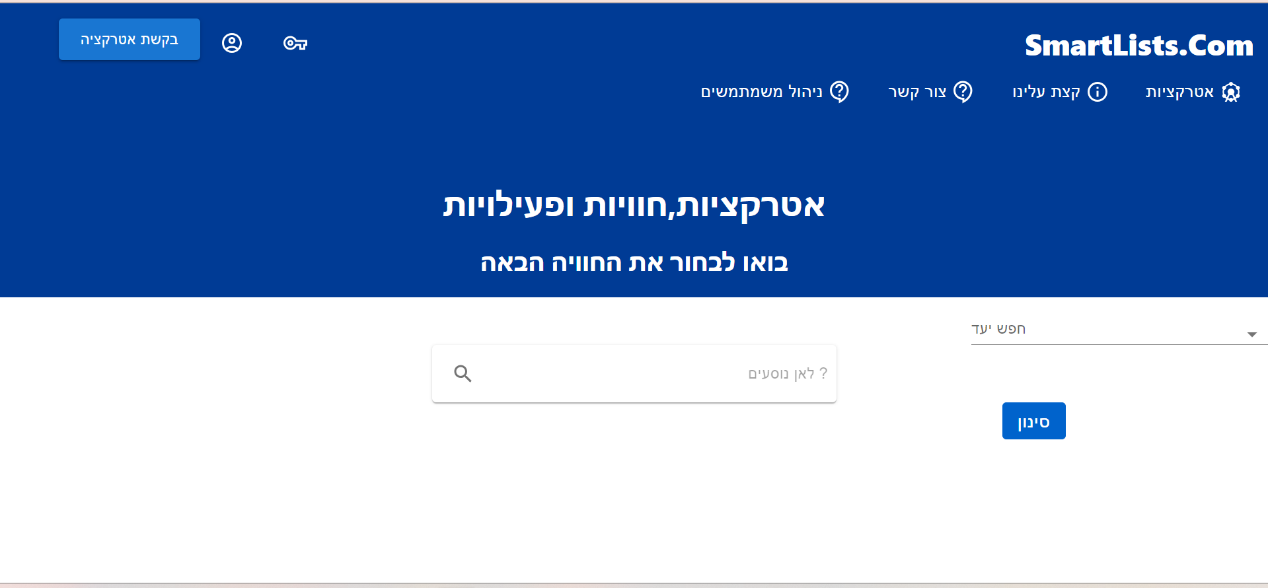
אטרקציות

אזור אישי

3.1.3 צילומי מסכים

3.1.3.1 דף הבית – כניסה למערכת

דף ראשי לאתר, כולל בתוכו קישורים לדף הכנסת אבדה או מציאה , סטטיסטיקות אודות טיפים וחוות דעת .



תמונה שמכילה טקסט, צילום מסך, גופן, מספר

התיאור נוצר באופן אוטומטי3.1.3.2 דף כניסה

# פרק 4

## **מבנה נתונים וארגון קבצים**

4 מבנה נתונים וארגון קבצים

4.1 ארגון קבצים

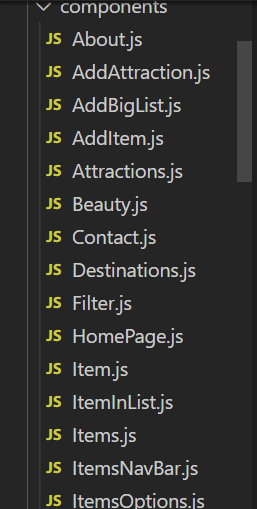
4.1.1 כללי

הפרויקט מחולק לצד שרת וצד שרת בפורמט JSON.

* + - 1. CLINET

השתמשנו בטכנולוגיית REACT.

תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטי4.1.1.1.1 מראה כללי של הפרויקט

:SERVER 4.1.1.2

השרת מכיל בסיס נתונים ,SQL בצד השרת מתבצעות השליפות.

השרת מתחלק ומנוהל בצמוד לארבע השכבות:

* DAL
* ENTITY
* DTO
* BLL

בעזרת השכבות בנינו פרויקט יעיל ומאורגן יותר.

פירוט על השכבות:

תמונה שמכילה טקסט, צילום מסך, תוכנה, גופן

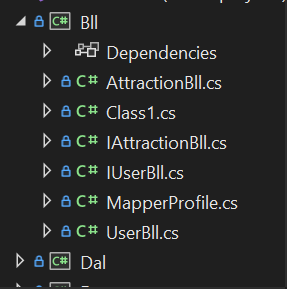
התיאור נוצר באופן אוטומטיClient Interface - 4.1.1.2.1 צילום

4.1.1.2.2 צילום מבנה הcontrollers-

תמונה שמכילה טקסט, צילום מסך, גופן, מספר

התיאור נוצר באופן אוטומטי

4.1.1.2.3 צילום מבנה ה – BLL

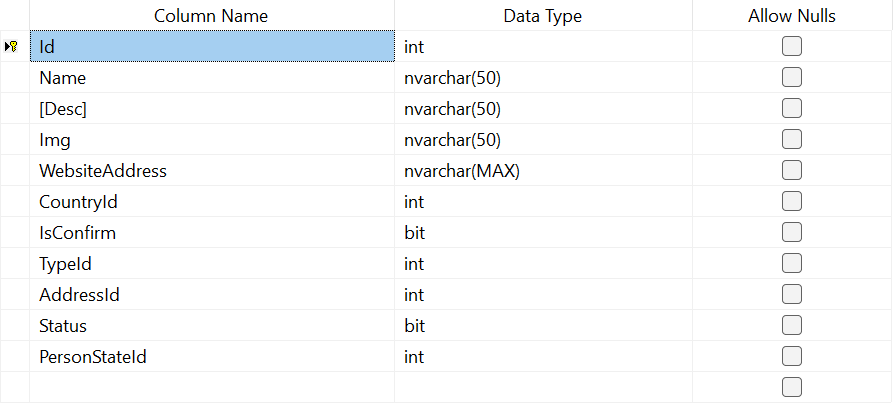


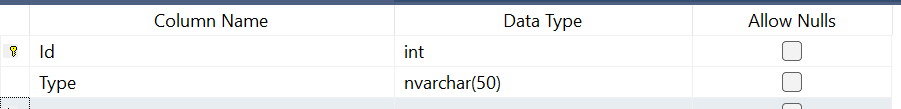
**טבלת משתמשים**

**תמונה שמכילה טקסט, צילום מסך, מספר, גופן

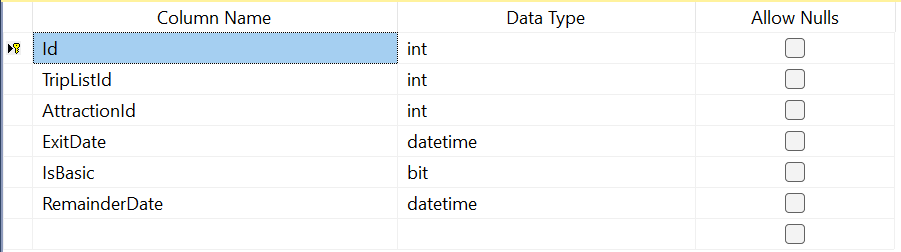
התיאור נוצר באופן אוטומטי**

**טבלת אטרקציות**

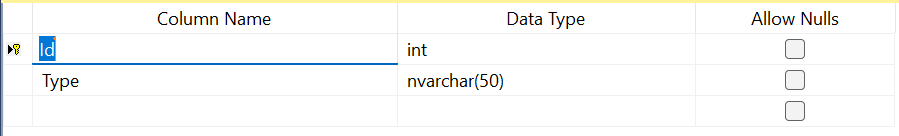
****

**תמונה שמכילה טקסט, קו, מספר, עלילה

התיאור נוצר באופן אוטומטיטבלת רשימה לאטרקציה**

****

**טבלת סוגי משתמש**

****

**תמונה שמכילה טקסט, צילום מסך, מספר, גופן

התיאור נוצר באופן אוטומטיטבלת פריטים**

**תמונה שמכילה טקסט, צילום מסך, מספר, קו

התיאור נוצר באופן אוטומטיטבלת חוות דעת**

**טבלת בקשה לאטרקציה**

**תמונה שמכילה טקסט, קו, גופן, מספר

התיאור נוצר באופן אוטומטי**

# פרק 5

## **תכנון**

תכנון

5.1 כללי

האתר מתחלק לצד שרת וצד לקוח, עם הפרדה ברורה כך שכל חלק נתיק ויכול להתקיים בנפרד.

צד השרת נכתב בפלטפורמת net.Asp בשפת C# . צד הלקוח נכתב בטכנולוגיית react – שפת javascript אלמנטים של HTML ועיצוב של CSS.

השתדלנו ככל האפשר להיצמד לעקרונות אותם מנחים הפלטפורמות הנ"ל, כדי להפיק את המרב שהן

מציעות ושמירה על ארכיטקטורת היכולות שלהן.

5.2 עקרונות תכנות בפרויקט זה עמדה לפנינו מטרה לכתוב קוד קצר ויעיל, תוך שימוש מקסימאלי בטכנולוגיות העומדות לרשותנו ובלי התוכנה שבחרנו.

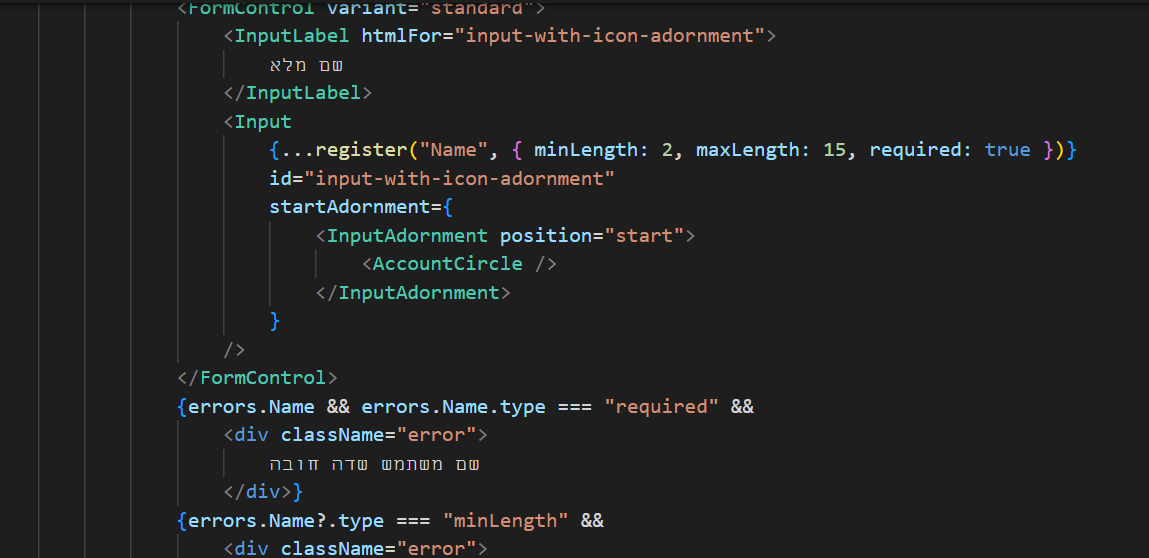
* + תכנות מונחה עצמים
  + שפת C# בנויה כולה מאובייקטים ומחייבת שפיתוח הקוד יעשה ע"י מחלקות.
  + חלוקה לשכבות

חלוקה ברורה בין ממשק המשתמש, הפקוד הלוגי של המערכת והתקשרות למערכות אחרות.

5.3 תיאור אלגוריתמים

הרשמה לאתר

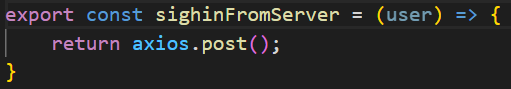
בדיקות תקינות לשדות.



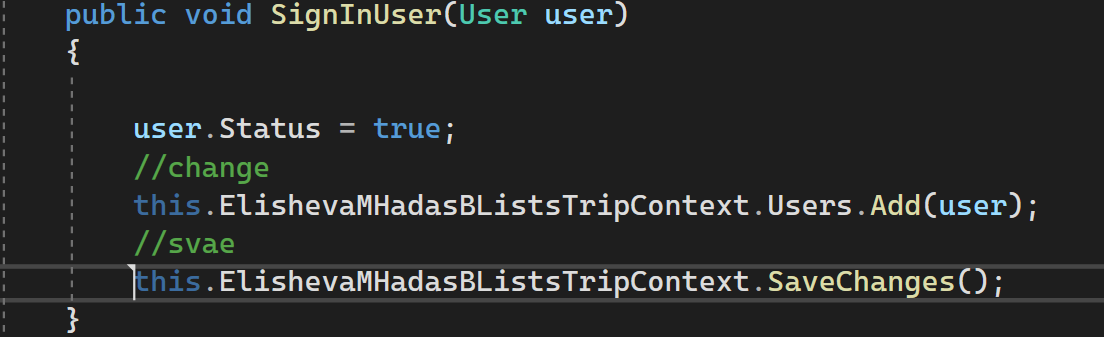
תמונה שמכילה טקסט, צילום מסך, גופן

התיאור נוצר באופן אוטומטילאחר בדיקת תקינות השדות – ניתן להירשם.

Action



Server



# פרק 6

## **מה הקנה הפרויקט**

**.6 מה הקנה הפרויקט**

* יכולת למידה עצמית.
* עבודה בצוות.
* קביעת לוח זמנים ועמידה בהם.
* פיתוח והרחבת הידע והשימוש בשפת C# (webapi) וב- .React
* פיתוח החשיבה ויצירתיות.
* התמודדות עם בעיות ופתרונם באופן עצמאי.

## **פרק 7**

## **ביבליוגרפיה**

#### 7 ביבליוגרפיה

אתרי אינטרנט לדוגמא:

* /https://stackoverflow.com
* [/https://www.w3schools.com](https://www.w3schools.com/)
* [/https://mui.com](https://mui.com/)
* /https://[www.react-israel.co.il/react-router](http://www.react-israel.co.il/react-router)
* ועוד