



פרויקט גמר 2020
מגישות: אהובי מייטס ואילה נהרי

תוכן העניינים:

3	שלמי תודה.....
4	מבוא
5	תקציר.....
6	תיאור הפרויקט
7	מטרות הפרויקט
8	תיחום המערכת
9	מבנה המערכת
10	מפרט טכני
11	תרשימי ניתוח מערכת
12	טבלאות מסד נתונים
13	פירוט מבנה ה-Data Base
19	ניתוח ה-Data Base
20	האלגוריתם המרכזי
22	זרימת הקוד במערכת
23	תצוגת הפונקציות המתוארות בזרימת המידע
25	מבנה הקוד
26	מדריך משתמש
27	מסכים
30	פירוט המחלקות בשכבת ה-service
32	מסקנות
33	סיכום
34	ביבליוגרפיה

שלמי תודה

בפתח דבר נודה לבורא עולם שהביאנו עד הלום וסייע לאורך כל הדרך.
בנוסף, נודה לכל אלו שנתנו רבות מכוחם ותבונתם למען הצלחת הפרויקט.
לרכזת המסלול, גב' שרה ברלין-
שטרחא ועמלה רבות להצלחתנו. על הדאגה והמסירות.
על תכנון עם המון מחשבה של מערכת לימוד המותאמת לצרכינו.
לגב' שולמית ברלין-
שייעצה, הדריכה, כיוונה וסייעה בפיתוח הפרויקט עד סיומו.
לגב' הבלין-
על הלימוד, התמיכה והסיוע הרב בבניית הפרויקט.
למורות לאנגולר-
על הלימוד העזרה והתמיכה לאורך כל הדרך .
ולכל הצוות על מקצועיות, השקעה ונכונות תמידית
לסייע תוך יחס חם ואכפתיות לאורך כל הדרך.
להנהלת סמינר "בית יעקב בנות אלישבע"-
על השקעה במשאבים הרבים שהועמדו לרשותנו.
ומעל הכל למשפחותינו היקרות שהיו חלק בלתי נפרד מ"הפרויקט".
על התמיכה, ההבנה, העידוד והאמון שנתנו בנו לאורך כל הדרך, ועל הכל בכלל...

מבוא

בבואנו לחשוב על נושא לפרויקט הגמר, רצינו לחקור נושא בתחום התכנה אשר יעשיר אותנו, ובמקביל יביא לידי ביטוי את הידע והכלים שרכשנו וההשקעה הרבה לאורך כל תקופת הלימודים. בנוסף רצינו לממש אותו באופן שיקנה לנו את הניסיון והפרקטיקה שבבניית פרויקט בעצמנו, משלב התכנון ועד ליישום הפרטים הקטנים.

התחבטנו הרבה בנוגע לבחירה שלנו, בסופו של דבר החלטנו לפתח פרויקט שייתן מענה לאנשים רבים העומדים לפני אירוע, אך מתקשים לנהל את התקציב שלהם בצורה אפקטיבית. פיתחנו אתר האחראי על ארגון האירוע תוך ניהול התקציב בצורה הכי יעילה ונכונה ללקוח.

האתר יקבל את תקציב הלקוח לחתונה, ויסיע למשתמש לחלק את התקציב בין התחומים השונים באופן המתאים ביותר בסגנון האישי של המשתמש וידאג לחלק אותו להוצאות החתונה. החלוקה תעשה עפ"י אחוזים להוצאה, שיקבעו ע"י סטטיסטיקה של משתמשים קודמים. יתכנו שינויים בחלוקה התקציב עפ"י העדפות אישיות של הלקוח. (לאחר שינוי האחוזים ע"י הלקוח המערכת תשקלל מחדש את הנתונים ותתבצע חלוקה מעודכנת של האחוזים).

לדוגמא:

זוג המעוניין לארגן חתונה, פעמים רבות נתקל בקושי לניהול התקציב ובקושי בחיפוש ובחירת בעל המקצוע/האולם המומלץ לאירוע שלהם. המערכת תקבל את התקציב מהלקוח ותחלק אותו באופן הכי נכון ומתאים לדרישותיו.

האתר יציג ללקוח מגוון אולמות איכותיים ומגוון של בעלי מקצוע מהטובים בתחום. לאחר חלוקת התקציב האתר יציג ללקוח רק אולמות/ בעלי מקצוע שסכומם קטן או תואם את התקציב, כך לא תתבצע חריגה מהתקציב.

במקרה והלקוח לא מצא שירות מסוים במסגרת התקציב- הוא יכול לשנות את חלוקת האחוזים ולהגדיל את התקציב לסוג שירות זה.

תקציר

1. כללי- המערכת נועדה לעזור בניהול תקציב הלקוח ולחסוך ללקוח טרחה בחיפוש ובחירת השירותים השונים.

כמו"כ המערכת תבצע הרשמה לנותני השירותים השונים.

2. אופן פעילות התוכנה מתחלק לשני חלקים:

שלב האתחול- המערכת קולטת את פרטי הלקוח ואת התקציב המיועד לאירוע.

שלב ההרצה - המשתמש בוחר את סוגי השירותים בהם מעוניין. המערכת תחלק את התקציב עפ"י אחוזים שנקבעו להוצאה עפ"י סטטיסטיקת משתמשים קודמים. תינתן ללקוח אפשרות לשנות את החלוקה על פי העדפות אישיות. המערכת תציג ללקוח רק נותני שירותים התואמים את התקציב (הלקוח יבצע סינון לנותני השירות עפ"י פרמטרים שונים).

3. הממשק- אופן התנהלות המערכת היא באמצעות אתר שיצרנו בו תתבצע הרשמת לקוחות ובמקביל תתבצע הרשמה גם לנותני השירות. המערכת תציג ללקוח סוגי שירותים- הלקוח יבחר את הסוגים בהם הוא מעוניין. הלקוח יכניס פרמטרים שונים לבחירת נותן השירות הרצוי, המערכת תציג את נותני השירות התואמים את התקציב והעדפות הלקוח.

4. הפיתוח- האתר פותח בסביבת העבודה NET.(בשפת C#).

צד השרת נכתב בשפות html i type script בטכנולוגית אנגולר.

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

5. הספר- המעיין בספר זה ימצא תיאור נרחב של הפרויקט הן בפן הלוגי והן בפן הגרפי החיצוני שישתלב במערכת הכללית. במהלך הספר ניתן להבחין בעקרונות התכנון של המערכת- השכבות השונות מדגם של אלגוריתמים הנכללים בפרויקט תהליכים שקיימים במערכת, תרשימים ועוד... ובנוסף מכיל הספר מדריך למשתמש בו נמצאים צילומי מסך והסברים כיצד ניתן להשתמש בתוכנה.

תיאור הפרויקט

עסקים נרשמים לאתר ומפרסמים את שירותיהם. הלקוח נכנס לאתר מכניס את פרטיו ואת תקציב החתונה שלו.

(השירותים השונים יחולקו לארגון משותף של החתונה, ארגון החתן וארגון הכלה, ויוצגו עפ"י רמת העדיפות שלהם.)

סוגי השירותים יוצגו ללקוח, הלקוח יבחר את סוגי השירותים שמעונין בהם.

התקציב יחולק לאחוזים על פי סטטיסטיקת הלקוחות הקודמים (כל לקוח משפיע על הסטטיסטיקה).

אלגוריתם יחלק את התקציב לשירותים עפ"י אחוזים מסוימים שיקבעו ע"י האתר, ויהיו נתונים לשינוי עפ"י דרישות והעדפות הלקוח.

האתר מיועד לארגון החתונה ולחלוקת משימות בצורה מדורגת לפי רמת העדיפות של השירות.

האתר כולל בתוכו שירותים שונים הנצרכים לארגון משותף של החתונה, לדוגמא: אולם, צלם, שירות קייטרינג, סידור פרחים וכו', ובמקביל שירותים המיועדים לחתן ולכלה.

תקציב החתונה יקבע ע"י הלקוח. האתר יציג ללקוח שירותים שונים התואמים את התקציב, והלקוח יבחר עפ"י העדפותיו.

אתר זה מפרסם את השירותים השונים, שכן כל לקוח שנכנס רואה את בעלי המקצוע.

בעלי מקצוע המעוניינים להתפרסם באתר נרשמים באתר ומשאירים את המייל שלהם ובו פרטי בעל המקצוע מאיזה תחום הוא, מה המחיר, מה הדרישות וכו'.

מטרות הפרויקט

- המטרה הראשית היא ללמוד להעמיק ולהתמקצע בAngular , c# , Web -i Api.
- כמו כן המטרה של האתר היא להקל על המשתמש במהלך ההכנות לחתונה בחיפוש אחר בעלי מקצוע אמינים.
- לנהל לו את התקציב בצורה האפקטיבית ביותר- למנוע גלישה מהתקציב
- לאפשר ללקוח להתכונן לחתונה בצורה רגועה ומסודרת.

המערכת תאפשר את התהליכים הבאים:

- הכנסת התקציב המאופשר למשתמש וקליטת התנאים.
- ניהול נכון של התקציב- חלוקת התקציב לסוגי השירותים השונים.
- הצעת שירותים התואמים את התקציב בלבד.

תיחום המערכת

נושאים באחריות המערכת:

- טעינת נתונים למסד הנתונים שבאתר.
- הצגת ספקי שרות העומדים על דרישות של הלקוח בהתאם לאירוע שבחר.
- חישוב ספקי השרות המתאימים ביותר לפי דרישותיו של בעל האירוע.
- חלוקת התקציב לסוגי השירותים עפ"י סטטיסטיקת משתמשים קודמים.

נושאים שאינם באחריות המערכת:

המערכת אינה מטפלת בסגירת אירוע מול השירותים השונים וכן בתשלום האירוע אלא רק ביעוץ לניהול התקציב והצעות לנותני שירותים התואמים את התקציב.

מבנה המערכת

בשלב תכנון המערכת ובניית מסד הנתונים התבקשנו לבנות זאת בצורה הנכונה, המדויקת והממוקדת ביותר תוך בניית היררכיה ברורה, על מנת למנוע שיבושים בנתונים ובצורת התצוגה.

במהלך השנים רוב מפתחי מערכות אינטרנטיות השתמשו ב .Net Framework- עם Web API בצד שרת ובצד קליינט .Angular, התוצאות המתקבלות הן די טובות והביאו לבנייה של מערכות טובות, ועדיין יש בה בעיה אחת שהפריעה, שמערכות אלו עובדות רק ב.Windows.

.NET Core היא פלטפורמה חגיגית מבית מיקרוסופט, מבוססת קוד פתוח וחוצה פלטפורמות המאפשרת פיתוח והרצת תוכנה, בין אם זה מערכות אינטרנטיות או מערכות שרצות על המחשב בצורה מקומית (הן נקראות Universal Windows Platform).

התוכנה מחולקת למספר שכבות:

- שכבת מסד הנתונים (SQL SERVER)
 - שכבת הישויות (Entity Framework) והגישה לבסיס הנתונים (Repository).
 - שכבת הקוד (service)
 - שכבת הקוד נעזרת בספריית ה-COMMON שמכילה את מבנה הנתונים כדי לתקשר עם שכבת ה-Repository- WebAPI.
 - שכבת הקישור c# (WebAPI).
 - ממשק משתמש angular –(UI)
- בשיטה זו קיימת הפרדת רשויות מוחלטת וכל שכבה עומדת באופן עצמאי לחלוטין ומתקשרת רק עם השכבה הסמוכה לה .

היקף הפרויקט

600 שעות.

מפרט הטכני

סביבת פיתוח:

עמדת פיתוח: מחשב lenovo

RAM 8 GB

Core i7

מערכת הפעלה: Windows 10

שפת תכנות: אנגולר, C#.

מסד נתונים: Sql Server

עמדת משתמש מינימלי:

חומרה: Core i3 or higher

או smartphone

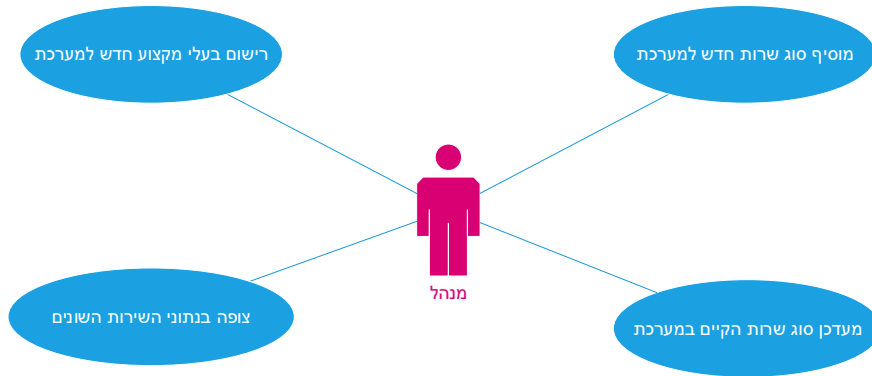
מערכת הפעלה: Windows XP/ Android/IOS

תוכנות: Sql Server, visual studio ,

חיבור לאינטרנט: נדרש

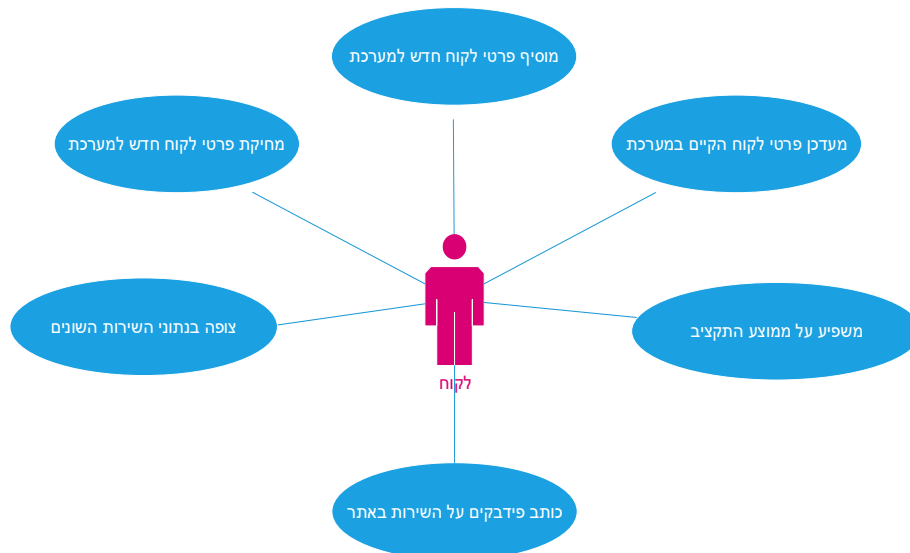
תרשימי ניתוח מערכת

ניהול המנהל

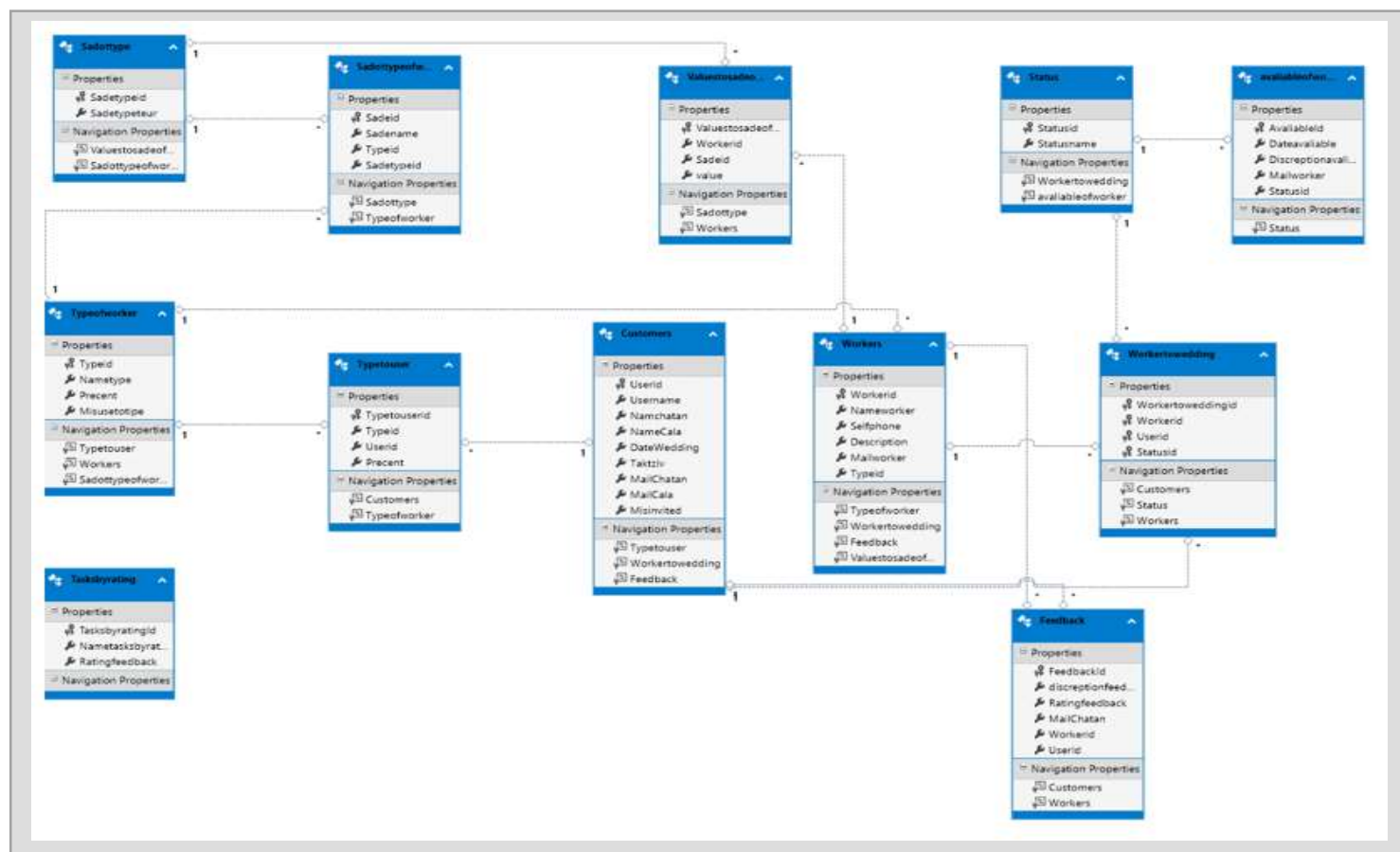


* האתר רוצה לשמור על אמינות נותני השירותים – לכן נותן השירות נדרש להגיש בקשה להרשמה למערכת והנהלת האתר תחליט האם להכניסו למערכת- עפ"י שיקולים שונים.

הלקוח



טבלאות מסד נתונים



פירוט מבנה ה Data Base

Customers – טבלת לקוחות

טבלה זו מייצגת את פרטי ההרשמה ללקוח, אזור אישי ללקוח.

שם	סוג	הסבר	
מפתח	int	קוד לקוח-סיסמא	UserId
	string	שם משתמש	Username
	string	שם חתן	Namchatan
	string	שם כלה	NameCala
	date	תאריך חתונה	DateWedding
	int	תקציב החתונה – מטבע ישראלי, שקלים.	Taktziv
	string	מייל חתן	MailChatan
	string	מייל כלה	MailCala
	int	מספר המוזמנים	Misinited

Typeofworker – טבלת סוגי בעלי מקצוע

טבלה זו מייצגת את סוגי השירותים השונים .

שם	סוג	הסבר	
Typeid	int	קוד סוג שרות	מפתח
Nametype	string	שם סוג שרות	
Precent	int	אחוזים לסוג שרות	
Misusetipe	string	מספר האנשים שהשתמשו בשירות מסוים	

Workers - טבלת בעלי מקצוע

טבלה זו מייצגת את בעלי המקצוע השונים.

שם	סוג	הסבר	
Workerid	int	קוד בעל מקצוע	מפתח
Nameworker	string	שם בעל מקצוע	
Selfphone	int	טלפון בעל מקצוע	
Description	string	תאור בעל המקצוע	
Mailworker	string	מייל בעל מקצוע	
Typeid	int	סוג השירות אליו שייך בעל המקצוע	

Sadotypeofworker – טבלת שדות לסוג לבעלי מקצוע

טבלה קישור בין Sadotype לבין Typeofworker.

טבלה זו מייצגת את השדות של סוג שרות מסוים.

שם	סוג	הסבר	
Sadeid	int	קוד שדה	מפתח
Sadename	string	שם שדה	
Typeid	int	קוד סוג שרות אליו השדה שייך	
Sadetypeid	int	קוד סוג שדה	

Sadotype – טבלת שדות לסוג לבעלי מקצוע

טבלה זו מייצגת את שדות לסוג בעלי מקצוע, זוהי טבלה של רבים לרבים!

שם	סוג	הסבר	
Sadetypeid	int	קוד סוג שדה	מפתח
Sadetypeur	string	תיאור השדה	

Valuetosadeofworker – טבלת ערכים לשדות לבעלי מקצוע

שם	סוג	הסבר	
Valuetosadeofworker	int	קוד ערך	מפתח
Workerid	int	קוד בעל מקצוע	
Sadeid	int	קוד שדה	
value	string	תיאור הערך	

Workertowedding – טבלת בעל מקצוע לחתונה

טבלת קישור בין Workers ל Customers .

טבלה זו מציגה את מצב הלקוח מול בעל במקצוע עפ"י הסטטוס.

האם הם סגרו ביניהם או שהלקוח שומר את בעל המקצוע בעדיפות ואז הסטטוס הוא תהליך בהמתנה .

שם	סוג	הסבר	
מפתח	Workertoweddingid	int	קוד בעל מקצוע לחתונה
	Workerid	int	קוד בעל מקצוע
	Userid	int	קוד לקוח
	Statusid	int	קוד סטטוס

Status- טבלת סטטוס

טבלה זו מייצגת את סטטוס פניית בעל מקצוע ואת סטטוס סגירת הלקוח עם בעל המקצוע.

שם	סוג	הסבר	
מפתח	Statusid	int	קוד סטטוס
	Statusname	string	תיאור סטטוס

Typetouser – טבלת סוגי שירות ללקוח

טבלת קישור בין Typeofworker ל Customers.
טבלה זו מציגה את השירותים שבעל המקצוע בחר.

שם	סוג	הסבר	
מפתח	Typetouserid	int	קוד סוג שירות ללקוח
	Typeid	int	קוד סוג שירות
	Userid	int	קוד לקוח
	Precent	int	אחוזים בהוצאה

Tasksbyrating – טבלת משימות עפ"י דירוג

שם	סוג	הסבר	
מפתח	Taskbyrating	int	קוד משימה
	Nametasks	string	שם משימה
	Ratingfeedback	int	מספר דירוג הפידבק

Feedback – טבלת פידבקים

טבלה זו מייצגת את פידבק הלקוח לבעל המקצוע.

שם	סוג	הסבר	
Feedbackid	int	קוד פידבק	מפתח
discreptionfeedback	string	תיאור הפידבק	
Ratingfeedback	int	מספר דירוג הפידבק	
MailChatan	string	מייל חתן	
Workerid	string	קוד בעל מקצוע	
Userid	int	קוד לקוח	

Workersregistration – טבלת פניות בעלי מקצוע

נותן השירות מבצע פניה להרשמה לאתר, מערכת האתר מחליטה אם לאשר אותו.

שם	סוג	הסבר	
Avaliableid	int	קוד פניה	מפתח
Dateavaliable	date	תאריך שליחת הפניה	
Discreptionavaliable	string	תיאור בעל המקצוע	
Mailworker	string	מייל בעל מקצוע	
Statusid	int	קוד סטטוס	

ניתוח Data Base

בתחילת הדרך הנושא והמושגים הקשורים בו היו זרים לנו לחלוטין, וכדי להקיף את הנושא כולו, להביאו על כתב ולהעלותו על מחשב, נדרשה מאתנו עבודה רבה וחשיבה ארוכת טווח,

איסוף המידע הדרוש, הבנת המושגים, ותפיסת כל מהלך הפרויקט עד לקבלת תמונה כוללת ורחבה, דרשו מאמץ רב והבנה מעמיקה.

אפיון בסיס הנתונים ארך שעות רבות והיווה חלק גדול ומשמעותי בפרויקט כולו ובאתגר שהוא העמיד בפנינו.

מבנה ה data base עבר גרסאות רבות ושונות עד לקבלת מבנה אמין ומדויק ככל האפשר.

כל תהליך בנייתו של בסיס הנתונים נעשה עם מחשבה לעתיד כך שמבנהו יהיה מותאם גם להמשך פיתוח.

במהלך הגדרת היעדים ותכנון בסיס הנתונים נתקלנו בקשיים שונים ולכן נאלצנו לשבת זמן רב בניסיון להבין הכל לפרטי פרטים, ובצורה המדויקת ביותר.

חקרנו ולמדנו מושגים חדשים.

השקענו מאמצים רבים במטרה לייעל ככל האפשר את התהליך- שיעשה בצורה הקלה, המובנת והנוחה ביותר.

בחרנו לבנות את הטבלאות בצורה כללית כך שנוכל להוסיף שירותים שונים בקלות ובפשטות. במקום לבנות טבלה לכל נותן שירות, לדוגמא: טבלת אולמות/זמרים/צלם וכו'.

בנינו את טבלת "typeofworker" - טבלה זו מכילה רשימה של סוגי שירותים שונים (כנ"ל - אולמות, זמרים וכו'),

קיישנו את טבלה זו לטבלה הנקראת "workers" - זוהי טבלה המכילה את בעלי המקצוע, לכל בעל מקצוע מציינים מאיזה סוג שירות הוא.

כמו כן קיישנו את טבלת typeofworker לטבלה הנקראת sadottype-טבלה זו מציינת את השדות של שירות מסוים לדוגמא לסוג השירות אולם יהיה את השדות גודל, מחיר, מיקום וכו'.

ולטבלה זו תקושר טבלת Valuestosadeofworker - המכילה ערכים שונים לשדות לדוגמא: לשדה מחיר 5000 - ₪, למיקום -ירושלים וכו'.

כך במקרה של עדכון/ הוספה/מחיקה לא נצטרך לבצע את הפעולה מספר פעמים (לכל נותן שירות באופן פרטי) אלא נבצע בטבלאות הכלליות והדבר ישתנה אצל כל נותני השירות המכילים ערך זה.

האלגוריתם המרכזי

חישוב הממוצע - כמה אחוזים מהתקציב אנשים בדרך כלל משקיעים בשירות מסוים. הממוצע מתעדכנת בכל פעם מחדש. חלוקת התקציב תתבצע עפ"י הממוצע - הלקוח יבחר את סוגי השירותים בהם הוא מעוניין והתקציב יחולק לאחוזים בין השירותים השונים. הלקוח יוכל לשנות את האחוזים שנקבעו לפי רמת העדיפות שלו - שאר האחוזים יתחלקו מחדש, במקרה והלקוח קבע עם נותן שירות מסוים - המצב יתעדכן בסטטוס, והאחוזים לסוג השירות שנקבע ינעלו. תינתן אף אפשרות לנעול שירות מסוים אם הלקוח לא מעוניין בשינוי האחוזים לסוג זה, כך שבמקרה ויתבצע שינוי באחוזים השינוי לא יכלול את הסוגים הנעולים. כמו"כ אם בבחירת שירות מסוים לא נוצל כל התקציב - שארית התקציב תחולק בין סוגי השירותים האחרים. כך תתבצע חלוקה מדוקדקת ויעילה תוך ניצול מרבי של התקציב. לדוגמא:

לקוח מכניס תקציב של 100,000 ₪ לארגון האירוע שלו. הלקוח בחר לאירוע שלו 4 סוגי שירותים: אולם, צלם, זמר, סידור פרחים. המערכת תחלק את התקציב שלו עפ"י הממוצע של המשתמשים הקודמים בצורה כזאת, לדוגמא: אולם - 50%, צלם - 15%, זמר - 25%, וסידור פרחים - 10%. כך יוצגו ללקוח רק אולמות עד 50,000 ₪, צלמים עד 15,000 ₪, זמרים עד 25,000 ₪ וסידורי פרחים עד 10,000 ₪. הלקוח החליט להשקיע יותר באולם ולהעלות אותו ל-60% מהתקציב. מתבצע שקלול מחדש של האחוזים והאחוזים יתחלקו מחדש עפ"י השינוי. אולם - 60%, צלם - 12%, זמר - 20%, סידור פרחים - 8%

אופן הביצוע:

צד Server:

פונקציה המחשבת את הממוצע עפ"י משתמשים קודמים - Avg()

פונקציה המחשבת את האחוזים לכל סוג שירות עפ"י ממוצע החלוקה של משתמשים קודמים.

```
public void Avg()//שירות סוג לכל האחוזים חישוב
{
    var result = iTypeR.GetAllTypeouser() //השונים השירותים הצגת
    typeid י"עפ בקבוצות
    .GroupBy(b => b.Typeid)
    .ToDictionary(
        g => g.Key.ToString(),// לפי הצגה key
        g => new
        {
            s = g.Select(b => b.Precent)
            .Sum(),
            c = g.Count()
        });
    var x = result.Select(g => new { g.Key, avg = g.Value.s / g.Value.c });
}
```

Client TX:

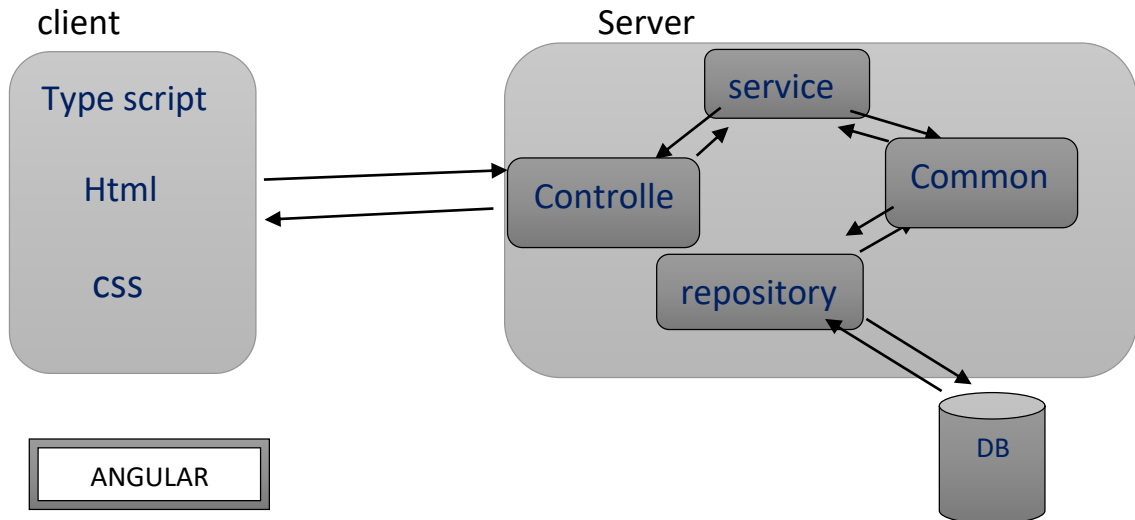
הלקוח יבחר את סוגי השירותים בהם הוא מעונין ואת האחוזים לכל סוג שירות. במקרה והלקוח לא בחר את כל סוגי השירות או שינה את האחוזים שניתנו לו כברירת מחדל (עפ"י ממוצע משתמשים קודמים), תתבצע חלוקה מחדש של האחוזים באופן הזה:

נסכום את האחוזים של כל סוגי השירותים (במקרה וסוג מסוים לא נבחר האחוזים מסוג זה יהיו 0), במקרה והסכום קטן מ100 נחלק את השארית בין סוגי השירותים האחרים כך:

סכום האחוזים/100, את סוגי השירותים השונים נכפיל ב-% התוצאה שתינתן, לדוגמא:

אם סכום האחוזים 80 נחשב: $1.25 = 100/80$ האחוזים יוכפלו ב1.25.

זרימת הקוד במערכת



דוגמא לזרימת מידע:

תהליך אישור סיסמת משתמש.

ברצוננו לבדוק האם הסיסמא שהוקשה ע"י המשתמש תואמת את הסיסמא הקיימת ב-

Data Base ואז לאשר את כניסתו למערכת.

יתבצעו השלבים הבאים:

המשתמש יקיש שם וסיסמא והם יועברו ל-Type script של הדף.

מה-Type script ישלחו הנתונים לפונקציה findCustomer הנמצאת

ב-Service . ה-service יבצע בקשת url לשרת.

השרת מקבל את הבקשה ומנווט ל controller הנמצא ב-Web Api.

ה-controller יזמן את הפונקציה Getcustomer הנמצאת ב-service .

ה-service מעוניין בקבלת נתונים מה-DB ולכן הוא פונה ל-repository .

ה-repository שואב את הנתונים הרצויים ממסד הנתונים וכעת מתבצע שלב החזרה.

בפונקציה Getcustomer שב-service מתבצעת הבדיקה על הנתונים.

מה-service התשובה חוזרת ל-controller ומשם היא חוזרת ל-service שמעביר את התשובה ל-Type script.

תצוגת הפונקציות המתוארות בזרימת המידע

Typescript - הפונקציה של ה-components:

```
connect()
{
  this.enterService.findCustomer(this.CustomerMember).subscribe(
    data => {
      debugger;
      if (data) {
        new alert("succses");
      }
      else {
        new alert("try agian");
      }
    },
    errors => {
      console.log("request failed");
    }
  );
};
```

הפונקציה ב-Service

```
findCustomer(id:number, name:text): Observable<Customer>
{
  debugger;
  return this.http.get(this.urlService+"Customers/Getcustomer/"+id+"/"+name);
}
```

הפונקציה ב-Controller

```
//סיסמא משתמש שם הכנסת יי"ע משתמש שליפת הפונקציה זימון
// GET api/<controller>/5

[HttpGet("Getcustomer/{id,name}")]

public CCustomers Get(int id, string name)
{
  return iCustS.Getcustomer(id,name);
}
```

הפונקציה ב-service

```
////שליפת שם משתמש וסיסמא
public CCustomers Getcustomer(int Id, string name)
{
```



```
{
    CCustomers customers =
    GetAllCustomer().FirstOrDefault(a.UserId == Id && a.Username
    == name);
    return customers;
}

//// (common כאובייקט) חזרת רשימת כל הלקוחות
public List<CCustomers> GetAllCustomer()
{
    List<CCustomers> customers = (from tmp in iCustR.GetAllCustomer()
    select new CCustomers
    {
        UserId = tmp.UserId, //סיסמה
        Username = tmp.Username, //שם משתמש
        Namchatan = tmp.Namchatan,
        NameCala = tmp.NameCala,
        DateWedding = tmp.DateWedding,
        Taktziv = tmp.Taktziv,
        MailChatan = tmp.MailChatan,
        MailCala = tmp.MailCala,
        Misinvited = tmp.Misinvited,
    }).ToList();

    return customers;
}
```

הפונקציה ב-repository

```
public List<Customers> GetAllCustomer()
{
    return context.Customers.ToList();
}
```


מבנה הקוד

הפרויקט מחולק לשני חלקים:

צד השרת וצד הלקוח.

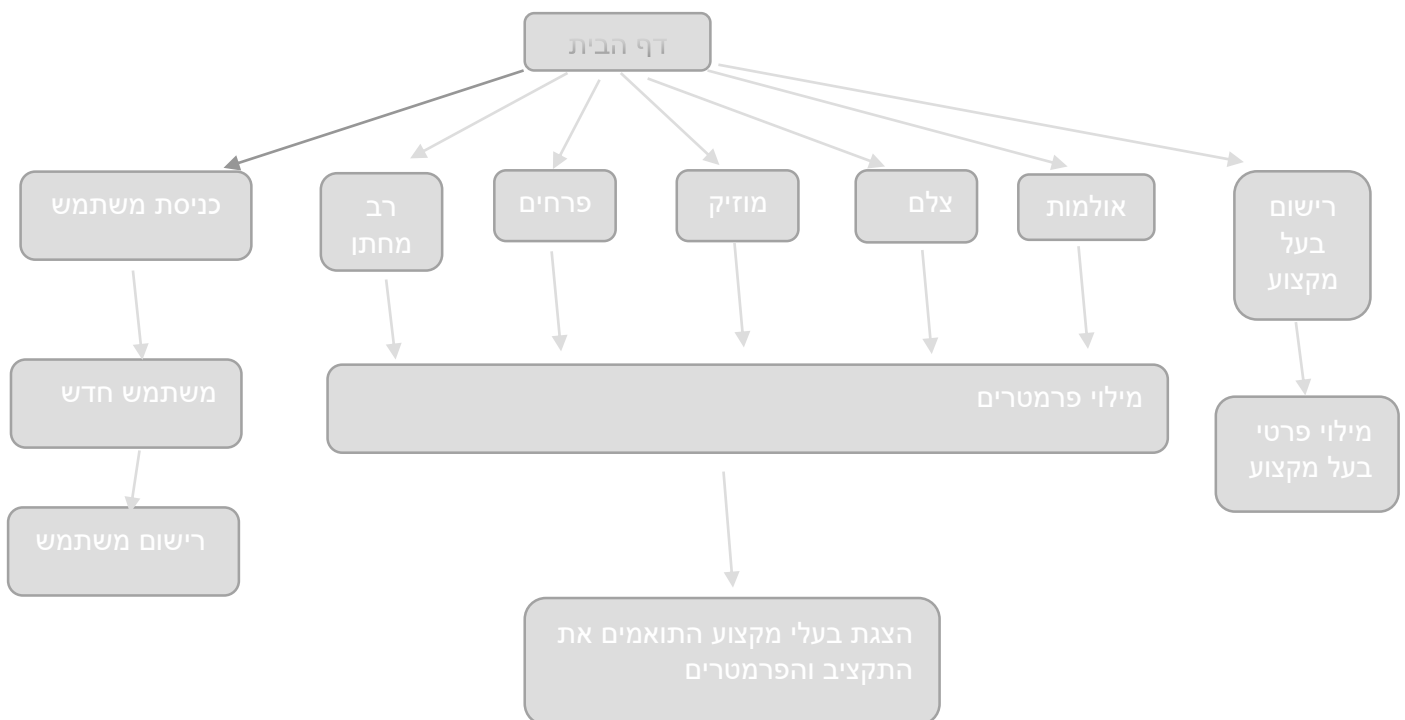
צד השרת נכתב בשפת C# בטכנולוגית WebApi.

ואילו צד הלקוח נכתב ב Angular הכולל: Html, Type Script.
וכן השתמשנו בספרייה: Material.

מדריך משתמש

משתמש המעוניין להיכנס לאתר יכול להיכנס להתעניין ולראות בעלי מקצוע.

משתמש המעוניין שהאתר ינהל לו את החתונה מבחינה תקציבית מבחינת פרמטרים ועוד חייב להירשם לאתר, ברישום הוא מגיע שוב לדף הבית ומתחיל בניהול החתונה.



מסכים

מסך ראשי:

כאשר פותחים את האתר יופיע מסך זה.

מזל טוב

אולמות צלמים+וידאו מוזיקה רב מחתן פרחים כניסת משתמש

מסך התחברות -כניסת משתמש:

במסך זה המשתמש קיים יכנס ע"י הזדהות וסיסמא.

מזל טוב

אולמות צלמים+וידאו מוזיקה רב מחתן פרחים כניסת משתמש

הכנס שם משתמש:

הכנס סיסמה:

הרשמת משתמשים חדשים:

מזל טוב

אולמות צלמים + וידאו מוזיקה רב מחמם פרחים כניסת משתמש

משתמשים חדשים

הכנס שם משתמש:

הכנס סיסמה:

הכנס סיסמה שוב:

אישור

הכנס שם חתן:

הכנס שם כלה:

תאריך חתונה:

תקציב החתונה

מייל חתן:

מייל כלה:

מספר מוזמנים:

במסך הבא יהיה על המשתמש לבחור באיזה שירותים הוא מעונין וכמה אחוזים
מכספו הוא נותן לכל שירות:

מזל טוב

[אולמות](#) [קייטרינג](#) [צלמים + וידאו](#) [מוזיקה](#) [סידור פרחים](#) [כנ](#)

ברכות על הצטרפותך!!!

בחירת האחוזים		בחירת השירותים:	
50%	<input type="range"/>	<input type="checkbox"/> אולם	
20%	<input type="range"/>	<input type="checkbox"/> צלם	
20%	<input type="range"/>	<input type="checkbox"/> זמר	
10%	<input type="range"/>	<input type="checkbox"/> סידור פרחים	
0%	<input type="range"/>	<input type="checkbox"/> רב מחתן	

חיפוש בעל מקצוע עפ"י פרמטרים שונים-

במסך זה אנו רואים שמחיר לזמר הוא עד 7000 שח עפ"י חישוב סכום התקציב באחוזים שהלקוח הכניס, רשימת הזמרים שהוצגו הם אלו שסכומם עד 7000 שח.



מזל טוב

[אולמות](#) [קייטרינג](#) [צלמים + וידאו](#) [מוזיקה](#) [סידור פרחים](#) [כנ](#)

מוזיקה:

מחיר לזמר: 7000

מחיר	אזורים:	ז'אנר:	קטגוריה:
<input type="checkbox"/> עד 5000	<input type="checkbox"/> כל האזורים	<input type="checkbox"/> מוזיקה חסידית	<input type="checkbox"/> זמר
<input type="checkbox"/> מעל 5000	<input type="checkbox"/> אזור ירושלים	<input type="checkbox"/> מוזיקה מזרחי	<input type="checkbox"/> Dj
<input type="checkbox"/> עד 9000	<input type="checkbox"/> הנהג	<input type="checkbox"/> מוזיקה קרליבך	<input type="checkbox"/> תזמורת
<input type="checkbox"/> מעל 9000	<input type="checkbox"/> מרכז	<input type="checkbox"/> מוזיקה אלטרנטיבית	
	<input type="checkbox"/> צפון		

למידע נוסף	4000	זמר חתונות ברמה גבוהה	527634221	אמיר דביר
למידע נוסף	8500	זמר חתונות	25387838	ליפה שמלצר
למידע נוסף	6000	זמר חתונות מספר 1 בארץ	25385726	אברהם מרד

פירוט המחלקות בשכבת ה-service

בפרק זה נרחיב על חלק מהמחלקות בשכבת ה-service וכן נפרט על חלק מהפונקציות הכתובות במחלקות אלו.

שליפת בעל מקצוע עפ"י פרמטרים מסוימים- מכיוון שטבלת worker אינה מכילה את כל הפרמטרים הנדרשים לחיפוש נאלץ לבצע כמה פונקציות על מנת לקבל המידע המבוקש:

1. פונקציה המחזירה רשימת שדות לסוג בעל מקצוע מסוים-

GetSadotType()

לדוגמא בעל מקצוע זמר יוחזר רשימת השדות קטגוריה, מחיר, ז'אנר וכו'

```
public List<CSadottypeofworker> GetSadotType(int Id)
{
    List<CSadottypeofworker> result = GetAllSadottypeofworker().Where(a
=> a.Typeid == Id).ToList();
    return result;
}
```

פונקציה המחזירה רשימת ערכים לפי קוד שדה-

GetValuestosadeofworkers()

לדוגמא שדה למחיר יוחזר רשימת ערכים 2000,4000,50000 וכו'

```
// שדה קוד י"עפ ערכים רשימת מחזיר
public List<CValuestosadeofworker> GetValuestosadeofworkers(int Id)
{
    List<CValuestosadeofworker> result = GetAllValuestosadeofworker().
Where(a => a.Sadeid == Id).ToList();
    return result;
}
```

2. פונקציה המחזירה רשימת בעלי מקצוע עפ"י סוג שירות מסוים-

GetworkerByName()

לדוגמא לשירות זמרים יופיע הרשימה חיים ישראל, אמירן דביר, אברהם פריד וכו'

```
// שם י"עפ מקצוע בעל שליפת
public CWorkers GetworkerByName(string name)
```

```

    {
        CWorkers result = GetAllWorkers().FirstOrDefault(a
=>a.Nameworker== name);
        return result;
    }

```

פונקציה המציגה רשימת בעלי מקצוע עם כל פרטיהם מסוג שירות מסוים:

```

public object[] GetDetailsWorker(int Id)///לישאול סוג על להחזרה
{
    // הערכים, השדות, מקצוע בעלי הטבלאות בין קישור
    var DetailsWorker = new[] {(from tmp in iWorkerR.GetAllWorkers()
join tmp1 in iValuesR.GetAllValuestosadeofworker()
on tmp.Workerid equals tmp1.Workerid join
tmp2 in iSadotypR.GetAllSadotypeofworker()
on tmp1.Sadeid equals tmp2.Sadeid
where tmp.Typeid == Id // המקצוע שבעל בדיקה)
מסוים שירות מסוג הוא

select new CWorkers
{
    Nameworker = tmp.Nameworker,
    Selfphone = tmp.Selfphone,
    Description = tmp.Description,
    Mailworker = tmp.Mailworker,
    Image = tmp.Image,
    price = tmp1.Value, //מחיר
}).ToList() };

    return DetailsWorker;
}

```

מסקנות

הדבר החשוב ביותר שלמדנו הוא שתכנון טוב הוא בסיס לפרויקט מוצלח. כאשר יש הכנה טובה, כל מהלך הפיתוח זורם. בבואנו לכתוב את הפרויקט נתקלנו בקשיים רבים: בתחילת תכנון המערכת לא ידענו מאיפה להתחיל, מהו המבנה ובמה להתמקד. היינו צריכות למצוא את האלגוריתם לפיענוח התקציב ע"י סטטיסטיקת האחוזים. קושי עיקרי נוסף היה המבנה של הפרויקט- המורכב מצד שרת וצד לקוח. כל צד דרש חשיבה והשקעה מרובה. צד הלקוח נכתב בטכנולוגית Angular על מנת שיהיה ממשק נוח למשתמש. לא היינו בקיאות כלל בטכנולוגיה זו. השימוש בספריות הנ"ל הקנו לנו ניסיון רב בחיפוש אחר חומרים שונים ובלמידה עצמית. לאחר שעות רבות של עמל ויזע, אנו חשות כי הפקנו תועלת רבה מפרויקט הגמר ומקוות שיסייע לנו בעתיד בעזרת ה'.

סיכום

ראשית, הפרויקט הקנה לנו ידע רב בשפות השונות: C#, Angular, WebApi.

האתר נכתב במשך השנה האחרונה בעבודה מאומצת של תכנון בעמל והשקעה, תוך שמירה על רמת תכנות גבוהה.

בהגיענו למועד סיום הפרויקט ובמבט אחורה על הדרך אותה עברנו ניתן לומר כי היא הביאה אותנו להבנה מעמיקה יותר ביסודות התכנות.

לידע כיצד ללמוד ולהכיר טכנולוגיות חדשות ולרצון לפתח תוכנה ברמה המקצועית ביותר.

ברמה האישית - לעבודה בזוגות ובצוות. דבר הדורש שיתוף פעולה מלא ותיאום בין חברי הצוות.

ניתן לומר שהפרויקט קידם אותנו באופן ניכר מבחינה מקצועית ואישית.

פרויקט זה אכן היווה אבן דרך משמעותית בהתמקצעות בתחום הנדסת התכנה.



ביבליוגרפיה

1. StackOverFlow

2. GitHub

3. ויקיפדיה