# דף כריכה

# דף שער

**שם הפרויקט :**

**והשבות**   
חסד במחיר מציאה

**רקע**

* 1. **תיאור ורקע כללי**

מערכת מקוונת המיועדת בעיקר לרשת פנים ארגונית, ולקהל יעד ממוקד.

היא נועדה לשמש ככלי לניהול החזרת אבדות לבעליהם, באופן נגיש ומסודר.

המערכת דינאמית ומאפשרת התאמות פנים ארגוניות על פי צרכיו של כל לקוח אופציונלי.

* 1. **מטרות המערכת**

1. לספק כלי לפרסום מידי של מציאות במוסד לימודי או בכל מוסד ארגוני אחר, במטרה להקל על ניהולו השוטף של המוסד.
2. ליצור מערכת אמינה זמינה ופשוטה לשימוש שתאפשר מציאה לפריטים אישיים של יחידים במוסד ותהווה כתובת בלעדית לפרסום כל מציאה.

**סקירת מצב קיים בשוק**

כיום, מטפלים ארגונים ומוסדות רבים באבידות בדרך של פרסומים ספציפיים במקומות מרכזיים, או שלא מפרסמים כלל. במוסדות רבים אין ממש למי לפנות כדי להעזר במציאת ובפרסום אבדות,  
מערכת האבדות מהווה במוסדות רבים מטרד לא קטן, הן לצוות הניהול והן לבעלי אבדות, או למוצאים.  
הנהלת המוסד נדרשת לטפל בעשרות אם לא במאות ויותר מזה חפצים שאין לדעת אם ימצא להם דורש מתי והיכן, כל ההתנהלות מורכבת, לא מסודרת ויוצרת מצב לא אידיאלי בעליל.  
כשחבר במוסד מוצא חפץ יקר ככל שיהיה, אין לו דרך מסודרת ואחראית לידע את **כל** חברי המוסד האחרים במציאה במטרה למצוא את המאבד,  
קל וחומר שכשאדם מבין שאבד לו דבר מה, אין לו אפילו קצה חוט למציאת אבדתו, היא עשויה להתגלגל מיד ליד במשך זמן רב וקלוש הסיכוי למציאתו.

**מה הפרויקט אמור לחדש או לשפר**

הפרויקט מציע מערכת לניהול אבדות ומציאות, ידידותית ונגישה, המאפשרת פרסום של כלל המציאות במוסד, דפדוף מהיר וחיפוש יעיל שינגיש למאבד את אבדתו ויחסוך עשרות העברות ידיים וחיפושים עצמאיים מורכבים.

המערכת תהיה נגישה לכל חבר במוסד ותהיה גמישה להתאמות פנים ארגוניות ע"י הנהלת הארגון.

מערכת מקוונת זו עתידה להעלות משמעותית את אחוזי מציאת האבדות, הודות למאגר האחיד המוחלט והידיעה הברורה שכשמשהו אבד, המקום הטבעי שלו להיות – זה במאגר הזה.

כמו כן המערכת תקל משמעותית על ניהול מוסדות בכך שתסיר מעליהם את עול הטיפול באבדות בצורה מוחלטת.

**דרישות מערכת ופונקציונאליות**

**דרישות מערכת**מסד נתונים ריאלציוני שיושתת ברשת הEthrnet של אותו המוסד  
חיבור וסינכרון בין המחשבים, מערכת הפעלה Windows התומכת בשפות .net

**פונקציונאליות**

מנהל:

* 1. בוחר סיסמת מנהל
  2. מזין כתובת מייל רשמית למוסד
  3. מזין רשימת מיקומים על פי חלוקה מסודרת של המוסד
  4. מזין קטגוריות ראשיות ומשנה כפי הנצרך במוסדו

מוצא:

* + מזין פרטי אבידה מלאים כולל בחירת קטגוריה.
  + מזין פרטים אישיים לפניה ישירה של המאבד.

מאבד:

* + סוקר את לוח המציאות המעודכן
  + מסנן את המציאות לפי נתונים שונים: מיקום,קטגוריה וכו'
  + צופה בנתוני מוצא.

**בעיות צפויות במהלך הפיתוח**

* 1. הבעיות:

בעיה 1: התאמת המוצר לצרכים שונים של ארגונים, כמו מיקומים אפשריים למציאת אבידות (חלוקה על פי ערים / מדינות / קווי אוטובוס / אזורים בתוך מבנה וכדו), חלוקה לקטגוריות שונות ועוד.

בעיה 2: קושי בהטמעת המערכת בארגונים ומוסדות שמרניים שיקשו לקחת חלק במהלך הדיגיטציה

* 1. פתרונות אפשריים:

לבעיה 1:

* + פתרון 1: הכלת אפשרויות המתאימות לכל ארגון אפשרי.
  + פתרון 2 : יצירת חבילות מוכנות של נתונים לפי צרכים שונים ונתינת אפשרות בחירה בינהם.
  + פתרון 3: נתינת אפשרות עריכה אישית של מסד לכל לקוח מנהל.

לבעיה 2:

* + פתרון 1: ליווי בהטמעת המערכת בשלביה הראשונים.
  + פתרון 2: תמיכה טכנית, סיוע וזמינות לטווח הרחוק.
  + פתרון 3: חישוב תועלת ההשקעה שתופק ממערכת כזו לעומת הקושי הראשוני בשימושה.
  1. הפתרון הנבחר עבור כל אחת מהבעיות:

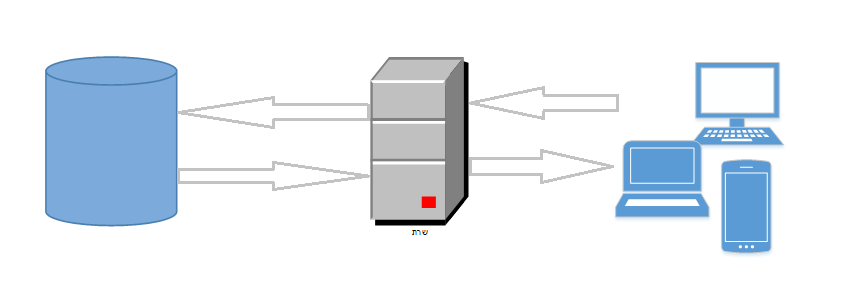
לבעיה 1: פתרון 3, כל ארגון ישתמש בחשבון המנהל בכדי לגשת בעצמו למסד הנונים ולערוך אותו, את הקטגוריות של האבידות, את רשימת המיקומים האפשריים וכו.

**פתרון טכנולוגי נבחר**

* 1. **טופולוגית הפתרון**

המערכת מורכבת משרת IISהמריץ את האתר בסביבת ה,server מסד נתונים – DB של SQL server

**דיאגרמה**



* 1. **טכנולוגיות בשימוש:**

מערכת זו מבוססת על עיקרון MVC, מחולקת לMODELS – מבנה לוגי של ישויות מסד הנתונים, VIEWS – הגדרת מראה המסכים שיוצגו למשתמש בדפי HTML. ו CONTROLLERS – שכבה שיוצרת את החיבור בין הנתונים למסך המשתמש, ומכילה את הפונקציונליות המתאימה לכל בקשה שמגיעה מהמשתמש.  
טכנולוגית AJAX לצורך טעינה מהירה של דפי האינטרנט, תוך עדכון החלקים המבוקשים באותה שליחה בלבד.

* 1. **שפות הפיתוח:** 
     1. בצד השרת: שפת C#
     2. בצד הלקוח: שפות javaScript, html5, css3

* 1. **תיאור הארכיטקטורה הנבחרת**

השימוש בטכנולוגית MVC מבטיח איכות בכל חלק ממרכיבי הישום, החל משכבת המידע הנשמר בצורה רציונאלית וממופה לטבלאות עם קישרי גומלין עובר דרך שכבת הבקרים החכמה שמופעלת אינטראקטיבית ומגיבה מידית לפעולות משתמש הקצה וכלה בממשק משתמש חדשני ונעים

* 1. **חלוקה לתכניות ומודולים:**
     1. **בצד השרת:**

ממומש במחלקות C# עם Actions לפועולות השונות ו Controllers המנהלים את הלוגיקה

* + 1. **בצד הלקוח:**

דפי HTML שונים.

עיון ברשימת מציאות וסינון לפי קריטריונים מסוימים, הסינון ישתמש בטכנולוגית טעינה חלקית של שונה לטבלת המציאות.

דף הוספת מציאה חדשה למאגר.

**סביבת השרת**

* 1. **ממשק המשתמש** יתקבל כדפי - HTML ויהיה מוצג באמצעות דפדפן אינטרנט מפענח V7 ומעלה.
  2. **ממשקים למערכות אחרות: אין**

.

* 1. **שימוש בחבילות תוכנה:**

התחברות עם EntityFramwork

שימוש בספרית Jquery.

שימוש בספרית Bootstrap.

שימוש בספרית mustache.

התממשקות למערכות טכנולוגיות לשליחת דואר אלקטרוני ושיחות טלפון

**מבני נתונים וארגון קבצים**

* 1. **שיטת האחסון**

הנתונים נשמרים במסד נתונים רלציוני בניהול של SQL Server

* 1. **מבני הנתונים**

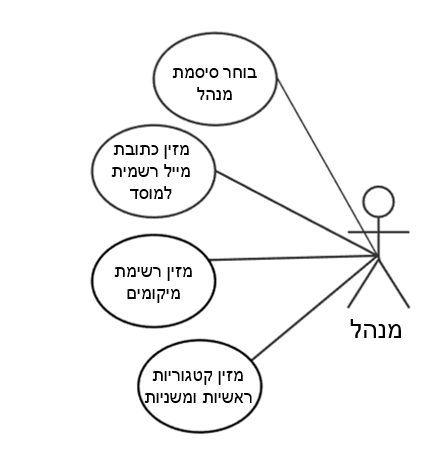
מאורגן במבנה טבלאי רלציוני, הטבלאות במבנה:

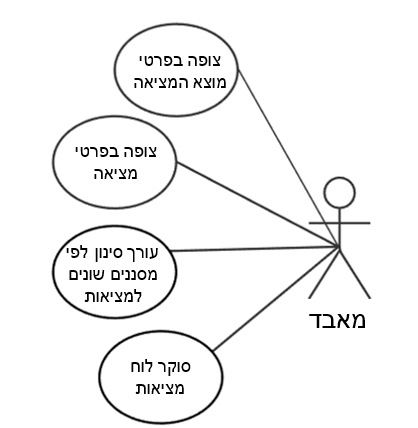
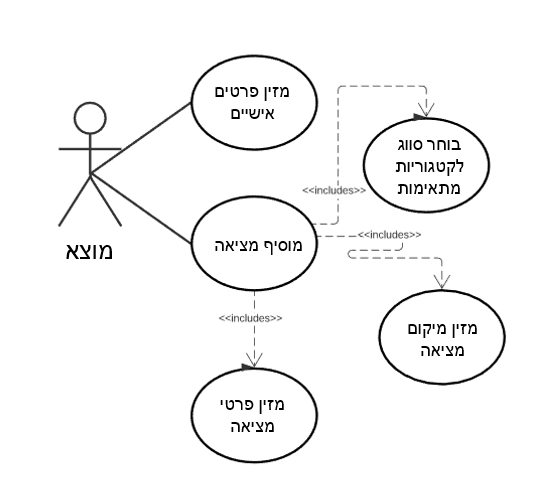
* טבלת מציאות
* טבלת הגדרות ארגון
* טבלת מיקומים
* טבלת קטגוריות
* טבלת תת קטגוריות
* טבלת ארכיון
  1. **מנגנוני התאוששות מנפילה/ קריסה/ תמיכה בטראנזקציות**

המידע נשמר בשרת SQL הבנוי עם מגנונים המתאימים למקרים אלו

**תרשימי מערכת מרכזיים**

* 1. Use Case





* 1. Sequence diagram

**תרשים פעילות UML  
עבור: מוצא מוסיף מציאה**

|  |  |  |
| --- | --- | --- |
| **מסד הנתונים** | **מערכת** | **מוצא** |
| מציאה נוספה למאגר  שמירת פרטים במאגר  החזרת תתי קטגוריה בהתאם לקטגורית האב  שליפת קטגוריות אב ומיקומים ממאגר | מציאה לא נוספה למאגר  הודעת שגיאה  בדיקת קריטריונים ותקינות פרטים  האם תקין?  כן לא  הצגת תתי קטגוריה בהתאמה  שליחת קטגורית אב שנבחרה  הצגת קטגוריות אב ומיקומים  שליחת בקשה להצגת שדות וקטגוריות \_ | מציאה לא מעודכנת במאגר  בוחר תת קטגוריה וממלא שדות נדרשים  בחירת קטגורית אב  בוחר אפשרות להוספת מציאה |

**תיאור המרכיב האלגוריתמי – חישובי**

* 1. סינון מציאות ע"פ הנתונים שמזין המאבד בדף החיפוש.
  2. פרישת פרטי מציאה ופרטי מוצא של המציאה אותה בוחר המאבד לבדוק.
  3. העברת המציאה לארכיון לאחר בירור טלפוני אוטומטי מול המוצא בענין סטטוס המציאה .

**תיאור/התייחסות לנושאי אבטחת מידע**

המידע מועבר לארכיון רק לאחר קלט אישי מהמפרסם

**משאבים הנדרשים לפרויקט:**

* 1. מספר שעות המוקדש לפרויקט: 720
  2. ציוד נדרש: מחשב, חיבור לאינטרנט.
  3. תוכנות נדרשות: visual Studio,SQL Server
  4. ידע חדש שנדרש ללמוד לצורך ביצוע הפרויקט: התממשקות לרשתות דואר אלקטרוני וקווים לוויניים
  5. ספרות ומקורות מידע:

**gitHub, stack overflow, tutorialsteacher.co, dotnet.microsoft**

**תכנית עבודה ושלבים למימוש הפרויקט:**

|  |  |  |
| --- | --- | --- |
| **שלב** | **משך זמן לעבודה זוגית** | **תאריכים משוערים** |
| * + ייזום הרעיון | **30 שעות** | **1.6-5.6** |
| * + ניתוח מערכת | **35 שעות** | **6.6-12.6** |
| * + ניתוח מבנה נתונים | **20 שעות** | **13.6-18.6** |
| * + אפיון UX - UI | **20 שעות** | **18.6-20.6** |
| * + כתיבת הלוגיקה העסקית | **150 שעות** | **21.6-10.10** |
| * + כתיבת ממשק המשתמש | **40 שעות** | **21.6-10.10** |
| * + עיצוב | **40 שעות** | **21.6-10.10** |

# הצעה לפרויקט גמר

# אישור הצעת הפרויקט ממה"ט

# תודות

# הצהרה

תאריך : 18/06/17

# תקציר

# תוכן העניינים

## מבוא

## מדריך למתכנת:

### אסטרטגיות טכנולוגיות:

### תיאור מבנה הפרויקט:

### עקרונות התכנון :

#### עקרונות תיאורטיים:

### תרשימים:

#### עץ תהליכים:

#### תרשים Uml:

#### תרשים מראה המחלקות:

### מבנה נתונים מאוכסנים:

#### תיאור מבנה קבצי XML:

#### מבנה קבצים ותיקיות:

##### תיקיית תמונות לעיצוב האתר-

##### קבצים המועלים לאתר ע"י המשתמש -

#### 2.6 תוכן הפרויקט:

#### 2.6.1 תיאור המחלקות:

**מחלקות ב- Dal:**

###### 1.6.2 תיאור הפונקציות:

## מדריך למשתמש:

### הוראות כלליות לשימוש באתר:

#### מדריך לאורח

#### מדריך למורה:

#### מדריך לתלמיד:

### מסכים:

#### מסך ראשי:

#### המסך השני שלך

#### המסך השלישי שלך

#### 3.3

## סיכום ומסקנות:

## נספחים:

## שונות:

## ביבליוגרפיה:

אתרים בנושא תכנות:

* stackoverflow.com
* webmaster.org.il
* w3schools.com