

# ניתוח מערכות מידע

כולל את מתודולוגיית ה-UML

יניב אליהו

עריכה לשונית ועיצוב: **שרה עמיהוד**

עיצוב עטיפה: **שרון רז**

## תודתנו למר רז הייפרמן על שנתן הסכמתו להשתמש בחומרים שלו בכתיבת הספר

### שמות מסחריים

שמות המוצרים והשירותים המוזכרים בספר הינם שמות מסחריים רשומים של החברות שלהם. הוצאת הוד-עמי ויניב אליהו עשו כמיטב יכולתם למסור מידע אודות השמות המסחריים המוזכרים בספר זה ולציין את שמות החברות, המוצרים והשירותים. שמות מסחריים רשומים (registered trademarks) המוזכרים בספר צוינו בהתאמה.

### הודעה

ספר זה מיועד לתת מידע אודות מוצרים שונים. נעשו מאמצים רבים לגרום לכך שהספר יהיה שלם ואמין ככל שניתן, אך אין משתמעת מכך כל אחריות שהיא.

המידע ניתן "כמות שהוא" ("as is"). הוצאת הוד-עמי ויניב אליהו אינם אחראיים כלפי יחיד או ארגון עבור כל אובדן או נזק אשר ייגרם, אם ייגרם, מהמידע שבספר זה, או מהתקליטור/דיסקט שעשוי להיות מצורף לו.

לשם שטף הקריאה כתוב ספר זה בלשון זכר בלבד. ספר זה מיועד לגברים ונשים כאחד ואין בכוונתנו להפלות או לפגוע בציבור המשתמשים/ות.

• טלפון: **09-9564716**

• פקס: **09-9571582**

• דואר אלקטרוני: **info@hod-ami.co.il**

• אתר באינטרנט: **www.hod-ami.co.il**

• דואר אלקטרוני של יניב אליהו: **yaniv@c4sa.com**

• אתר C4SA: **www.C4SA.com**

# ניתוח מערכות מידע

כולל את מתודולוגיית ה-UML

יניב אליהו





כל הזכויות שמורות

**הוצאת הוד-עמי**  
**לספרי מחשבים בע"מ**

ת.ד. 6108 הרצליה 46160

טלפון: 09-9564716 פקס: 09-9571582

info@hod-ami.co.il

וכן

© כל הזכויות שמורות

**יניב אליהו וחברת C4SA בע"מ**

c4sa@c4sa.com

אין להשאיל ו/או לעשות שימוש מסחרי ו/או להעתיק, לשכפל, לצלם, לתרגם, להקליט, לשדר, לקלוט ו/או לאחסן במאגר מידע בכל דרך ו/או אמצעי מכני, דיגיטלי, אופטי, מגנטי ו/או אחר - בחלק כלשהו מן המידע ו/או התמונות ו/או האיורים ו/או כל תוכן אחר הכלולים ו/או שצורפו לספר זה, בין אם לשימוש פנימי או לשימוש מסחרי. כל שימוש חורג מציטוט קטעים קצרים במסגרת של ביקורת ספרותית אסור בהחלט, אלא ברשות מפורשת בכתב מהמוציא לאור.

הודפס בישראל 11/2004

All Rights Reserved  
**HOD-AMI Ltd.**  
P.O.B. 6108, Herzliya  
ISRAEL, 2004

מסת"ב 965-361-364-2 ISBN

# תוכן עניינים מקוצר

---

|     |  |
|-----|--|
| 31  | חלק 1: אפיון ועיצוב מערכות מידע.....                               |
| 33  | הקדמה.....   |
| 35  | פרק 1: עבודת מנתח מערכות ומחזור חיי מערכת תוכנה.....               |
| 49  | פרק 2: ייזום.....  |
| 66  | פרק 3: חקר מצב קיים.....   |
| 97  | פרק 4: אפיון מוכוון תהליכים.....                                   |
| 101 | פרק 5: אפיון מוכוון אירועים.....                                   |
| 125 | פרק 6: עיצוב תוכנה.....  |
| 133 | פרק 7: אפיון מערכות מוכוון עצמים לפי UML.....                      |
| 223 | פרק 8: עיצוב מערכות מוכוון עצמים לפי UML.....                      |
| 235 | פרק 9: תבניות עיצוב – Design Patterns.....                         |
| 257 | פרק 10: עיצוב ממשק גרפי (GUI).....                                 |
| 301 | פרק 11: עיצוב ממשק באינטרנט.....                                   |
| 355 | חלק 2: אפיון ועיצוב בסיסי נתונים.....                              |
| 356 | הקדמה.....   |
| 359 | פרק 1: מסדי נתונים כחלק מניתוח המערכת.....                         |
| 365 | פרק 2: רקע ומושגי יסוד.....  |
| 381 | פרק 3: עקרונות מערכות לניהול בסיסי נתונים.....                     |
| 427 | פרק 4: אפיון בסיסי נתונים.....                                     |
| 477 | פרק 5: עיצוב בסיסי נתונים – המעבר לטבלאות.....                     |
| 509 | פרק 6: בסיסי נתונים מוכווני עצמים (Object Oriented Databases)..... |

|            |   |
|------------|---|
| 523        | פרק 7: שפת SQL                          |
| 607        | פרק 8: מחסני נתונים (Data Warehouse)    |
| 647        | פרק 9: כריית נתונים (Data Mining)       |
| <b>673</b> | <b>חלק 3: אבטחת איכות ובדיקות תוכנה</b> |
| 675        | הקדמה                                   |
| 677        | פרק 1: הנדסת איכות תוכנה                |
| 690        | פרק 2: מערכות לניהול איכות – לשם מה?    |
| 698        | פרק 3: ניהול איכות, סיכונים ופרויקטים   |
| 703        | פרק 4: תהליך הפיתוח הבסיסי              |
| 735        | פרק 5: עיקרי מערכת האיכות – לב המערכת   |
| 751        | פרק 6: שיפור תהליכים                    |
| 759        | פרק 7: בדיקות תוכנה                     |
| 791        | פרק 8: שיקוף – סקר (Review)             |
| 797        | פרק 9: תרגיל מסכם                       |
| <br>       |   |
| 799        | מילון מונחים                            |
| <br>       |   |
| 809        | אינדקס                                  |

# תוכן עניינים

---

**31 חלק 1: אפיון ועיצוב מערכות מידע.....**

**33 הקדמה.....**

**35 פרק 1: עבודת מנתח מערכות ומחזור חיי מערכת תוכנה.....**

36..... על ניתוח ומנתח

36..... ניתוח מערכת – הגדרה

37..... עבודת מנתח המערכת

37..... מחזור חיי מערכת

38..... ייזום

39..... חקר מצב קיים

41..... אפיון

41..... הגדרת דרישות

42..... אפיון הפתרון

45..... עיצוב

45..... עיצוב תפישתי

45..... עיצוב לוגי

45..... עיצוב פיסי

46..... מימוש

46..... בדיקות

47..... הדרכה והטמעה

47..... התקנה והסבה

48..... תחזוקה

48..... מה עם אבטחת איכות?

**49 פרק 2: ייזום.....**

50..... שלב הייזום

50..... מטרות שלב הייזום

50..... היזום

51..... סוגי יזמות במערכות מידע

52..... נושאים עיקריים בשלב הייזום

52..... תיחום ארגוני

54..... תיחום לוגי – תהליכים ואירועים

55..... מטרות ויעדים

55..... חזון

55..... מטרות

|         |                              |
|---------|------------------------------|
| 56..... | יעדים                        |
| 56..... | מטרות ויעדים של מערכת המידע  |
| 57..... | הגדרת הבעיות                 |
| 58..... | שלב ראשון – הגדרת הבעיה      |
| 58..... | שלב שני – הגדרת הסיבות       |
| 59..... | שלב שלישי – הגדרת התוצאות    |
| 60..... | סוגי בעיות                   |
| 60..... | עלות תועלת ו-ROI             |
| 61..... | אופק הזמן                    |
| 62..... | תכולת היישום – אילוצי הפתרון |
| 62..... | אילוצים טכנולוגיים           |
| 62..... | אילוצי מימוש                 |
| 62..... | אומדן עלויות                 |
| 63..... | שיווק                        |
| 63..... | סגירת השלב                   |
| 64..... | מסמך הייזום                  |
| 64..... | נפח העבודה                   |
| 64..... | מי כותב                      |
| 65..... | סיכום                        |

### **פרק 3: חקר מצב קיים.....66**

|         |                                       |
|---------|---------------------------------------|
| 67..... | שלב חקר מצב קיים                      |
| 67..... | בעלי התפקידים                         |
| 67..... | השלב כחלק ממחזור חיי פיתוח מערכת מידע |
| 68..... | מטרות שלב חקר מצב קיים                |
| 68..... | מתי מבצעים?                           |
| 68..... | אפשרויות בחקר מצב קיים                |
| 69..... | המתודולוגיה והנוטציה                  |
| 69..... | היסטוריה                              |
| 69..... | המתודולוגיה                           |
| 70..... | הנוטציה                               |
| 70..... | עברית טבעית                           |
| 70..... | תרשימי זרימה פונקציונליים             |
| 70..... | מאגר נתונים או מאגר מידע              |
| 71..... | תהליך או פעולה ממוחשבת                |
| 71..... | פעולה ידנית                           |
| 71..... | תיוק                                  |
| 72..... | קלט ידני למחשב                        |
| 72..... | החלטה                                 |
| 73..... | זרימת נתונים                          |
| 73..... | מסמך, טופס, דוח, ידני או ממוחשב       |
| 73..... | מחבר                                  |
| 73..... | בניית התרשים                          |
| 75..... | תיאור מאותגר טכנולוגי                 |

|         |                                       |
|---------|---------------------------------------|
| 75..... | DFD (Data Flow Diagram) – זרימת המידע |
| 75..... | סימונים                               |
| 75..... | ישות חיצונית                          |
| 76..... | מאגר נתונים                           |
| 76..... | תהליך                                 |
| 76..... | אפיק נתונים או אפיק מידע              |
| 77..... | ניהול תצורה                           |
| 78..... | בניית התרשימים                        |
| 78..... | תרשים תוכן                            |
| 80..... | DFD "0" פיסי                          |
| 81..... | עץ תהליכים                            |
| 82..... | DFD "0" לוגי                          |
| 83..... | תרשימי פירוק (DFD) ברמה 1 ומעלה       |
| 87..... | תיאור תהליכים                         |
| 87..... | מילון הנתונים                         |
| 87..... | יתרונות וחסרונות השיטה                |
| 88..... | נפח עבודה                             |
| 88..... | מסמך חקר מצב קיים                     |
| 89..... | סגירת השלב                            |
| 90..... | תרגיל מסכם                            |
| 90..... | בית המסחר "שטוזה"                     |
| 96..... | אז מה עלי לעשות?                      |
| 96..... | סיכום                                 |

## **97 ..... פרק 4: אפיון מוכוון תהליכים**

|          |   |
|----------|---|
| 98.....  | הקדמה                                     |
| 98.....  | היסטוריה                                  |
| 98.....  | ניתוח מוכוון תהליכים                      |
| 99.....  | בניית התרשימים                            |
| 99.....  | המעבר מחקר מצב קיים לאפיון המערכת העתידית |
| 100..... | יתרונות השיטה                             |
| 100..... | שילוב השיטה בנוהל מפת"ח                   |
| 100..... | סיכום                                     |

## **101 ..... פרק 5: אפיון מוכוון אירועים**

|          |                                     |
|----------|-------------------------------------|
| 102..... | בעיות בגישה מוכוונת תהליכים         |
| 102..... | היסטוריה                            |
| 103..... | הגדרה                               |
| 103..... | השתלבות גישת האירועים במערכות מחשוב |
| 103..... | סוגי אירועים                        |
| 103..... | אירוע חיצוני – External Event       |
| 103..... | אירוע פנימי – Internal Event        |
| 103..... | אירוע זמן – Temporal Event          |
| 104..... | מבנה אירוע                          |

|            |                                     |
|------------|-------------------------------------|
| 104        | שיטת הסימון (נוטציה)                |
| 104        | ישות חיצונית                        |
| 105        | שעון                                |
| 105        | מאגר נתונים                         |
| 105        | פלט                                 |
| 105        | אירוע                               |
| 106        | אפיק מידע                           |
| 106        | ניהול תצורה                         |
| 107        | אפיון אירועים                       |
| 107        | הפיכת המציאות לאירוע                |
| 111        | היררכיה                             |
| 111        | אירוע-על                            |
| 112        | דוגמה לאירוע-על                     |
| 113        | יתרונות הגישה מוכוונת האירועים      |
| 113        | שילוב מודל האירועים עם מודל הנתונים |
| 114        | שילוב מודל האירועים בנוהל מפת"ח     |
| 114        | תרגיל מסכם                          |
| 114        | סיפור המקרה                         |
| 115        | אז מה יש לעשות?                     |
| 115        | פתרון בשלבים                        |
| 118        | אירוע השאלה                         |
| 118        | אירוע החזרה                         |
| 119        | אירוע בקשה לרכש                     |
| 119        | אירוע אישור רכש                     |
| 120        | אירוע הזמנת ספרים מההוצאה לאור      |
| 120        | אירוע קבלת הזמנה                    |
| 121        | אירוע הפקת תזכורות                  |
| 121        | אירוע הפקת קטלוג                    |
| 122        | אירועי-על                           |
| 123        | עץ המערכת                           |
| 123        | סוף התרגיל                          |
| 124        | סיכום                               |
| <b>125</b> | <b>פרק 6: עיצוב תוכנה</b>           |
| 126        | הקדמה                               |
| 126        | עיצוב לוגי                          |
| 126        | עיצוב אירועים                       |
| 128        | עיצוב מסכים                         |
| 128        | עיצוב בסיס הנתונים                  |
| 128        | עיצוב פיסי                          |
| 129        | עיצוב אירועים (פיסי)                |
| 129        | עיצוב מסכים (פיסי)                  |
| 130        | עיצוב בסיס הנתונים (פיסי)           |
| 130        | בעלי תפקידים                        |

|            |   |
|------------|---|
| 132        | סיכום   |
| <b>133</b> | <b>פרק 7: אפיון מערכות מוכוון עצמים לפי UML</b> |
| 135        | הקדמה   |
| 136        | בעיות בפיתוח מערכות                             |
| 136        | מתודולוגיה חדשה                                 |
| 137        | מהו ניתוח מוכוון עצמים                          |
| 141        | היכרות עם UML                                   |
| 143        | מטרות UML                                       |
| 143        | מרכיבי UML                                      |
| 145        | תהליך הפיתוח                                    |
| 146        | הגדרת דרישות                                    |
| 146        | Use Case Diagram                                |
| 146        | פונקציונליות                                    |
| 147        | שיטת סימון (נוטציה)                             |
| 152        | Use Case Documentation                          |
| 157        | OOAD  |
| 157        | מבוא  |
| 158        | יסודות המתודולוגיה                              |
| 158        | מחלקה – Class                                   |
| 159        | אובייקט – Object                                |
| 159        | הפשטה – Abstraction                             |
| 160        | תכונות – Attributes                             |
| 161        | שיטות – Methods                                 |
| 161        | סיכום שיטות הסימון                              |
| 162        | מצב האובייקט – Object State                     |
| 162        | התנהגות האובייקט – Object Behavior              |
| 162        | שיטת סימון                                      |
| 163        | הכמסה – Encapsulation                           |
| 165        | סוגי שיטות – Method Types                       |
| 166        | שיטות גישה – Access Methods                     |
| 167        | בנאי – Constructor                              |
| 167        | בניית אובייקט – Object Construction             |
| 167        | מפרק – Destructor                               |
| 168        | פירוק אובייקט – Object Destruction              |
| 168        | תכונות מחלקתיות – Class Attributes              |
| 169        | תכונות סופיות – Final Attributes                |
| 169        | העמסה – Overloading                             |
| 170        | קשרים   |
| 170        | קשר הקשר – Association                          |
| 171        | כיוון – Navigation                              |
| 172        | מידת ריבוי מפורשת – Cardinality                 |
| 173        | מידת ריבוי – Multiplicity                       |
| 174        | שם האסוציאציה – Association Name                |

|            |   |
|------------|---|
| 174        | Association Class – מחלקת קשר                   |
| 175        | Aggregation – קשר הכלה                          |
| 176        | Composition – קשר הכלה חזקה                     |
| 177        | Self Association – קשר אסוציאטיבי עצמאי         |
| 178        | תרגיל ביניים                                    |
| 181        | Inheritance – הורשה                             |
| 183        | Protected – מוגן                                |
| 185        | Overriding – רמיסה                              |
| 186        | Polymorphism – ריבוי צורות                      |
| 187        | Multiple Inheritance – הורשה מרובה              |
| 188        | Interface – ממשק                                |
| 190        | Abstract Class – מחלקה מופשטת                   |
| 191        | תרגיל   |
| 195        | Open Close – גישת פתוח סגור (פ"ס)               |
| 195        | המודל הסטטי                                     |
| 195        | Class Diagram                                   |
| 196        | מציאת מחלקות                                    |
| 196        | Class Diagram ל- Use Case המעבר מתרשים          |
| 197        | CRC Cards                                       |
| 199        | Class Diagram ל- ERD המעבר מתרשים               |
| 200        | המודל הדינמי                                    |
| 200        | Sequence Diagram – תרשים רצף                    |
| 204        | Constraints – אילוצים                           |
| 204        | Remarks – הערות                                 |
| 205        | Create & Destroy – יצירה ופירוק                 |
| 206        | זיהוי התרשים                                    |
| 206        | דוגמה מסכמת לתרשימים                            |
| 207        | תרגיל   |
| 208        | Collaboration Diagram – תרשים שיתוף             |
| 210        | תרגיל   |
| 211        | State Machine Diagram – תרשים מצבים             |
| 214        | תרגיל   |
| 216        | Activity Diagrams – תרשים פעילויות              |
| 218        | תרגיל   |
| 219        | שימוש בשרטוטים במהלך מחזור חיי המערכת           |
| 222        | סיכום   |
| <b>223</b> | <b>פרק 8: עיצוב מערכות מוכוון עצמים לפי UML</b> |
| 224        | הקדמה   |
| 224        | המעבר לעיצוב בעזרת כלי CASE                     |
| 225        | מודל ה-MVC                                      |
| 226        | Stereotypes                                     |
| 227        | המעבר לבסיס הנתונים                             |
| 228        | עיצוב פיסי                                      |

|     |       |
|-----|-------|
| 229 | תרגיל |
| 234 | סיכום |

**235 .....פרק 9: תבניות עיצוב – Design Patterns**

|     |                                     |
|-----|-------------------------------------|
| 236 | הקדמה                               |
| 236 | מטרות תבניות העיצוב                 |
| 237 | סוגי תבניות עיצוב                   |
| 237 | Singleton                           |
| 241 | Composite                           |
| 244 | מתי נשתמש בתבנית זו?                |
| 244 | הבהרות במימוש תבנית עיצוב Composite |
| 245 | יתרונות וחסרונות                    |
| 246 | Abstract Factory                    |
| 248 | מתי נשתמש בתבנית זו?                |
| 248 | יתרונות וחסרונות                    |
| 248 | Template Method                     |
| 250 | מתי נשתמש בתבנית זו?                |
| 250 | יתרונות וחסרונות                    |
| 250 | Command                             |
| 252 | מילון תבניות (חלקי)                 |
| 252 | Singleton                           |
| 252 | Adapter                             |
| 252 | Composite                           |
| 252 | Decorator                           |
| 252 | Factory Method                      |
| 253 | Abstract Factory                    |
| 253 | Template Method                     |
| 253 | Strategy                            |
| 253 | Observer                            |
| 253 | Proxy                               |
| 253 | Visitor                             |
| 254 | Bridge                              |
| 254 | Façade                              |
| 254 | Command                             |
| 254 | Iterator                            |
| 254 | תרגיל מסכם                          |
| 254 | מערכת Sea&See                       |
| 256 | סיכום                               |

**257 .....פרק 10: עיצוב ממשק גרפי (GUI)**

|     |  |
|-----|--|
| 258 | הנדסת אנוש על קצה המזלג                |
| 259 | עקרונות בעיצוב עבודה שיתופית עם המשתמש |
| 259 | החשיבות של ממשק מעוצב כראוי            |
| 260 | הצורך בפשטות הפעלה של יישומים          |

|     |                                     |
|-----|-------------------------------------|
| 260 | תחומי מפתח לשיפור                   |
| 260 | תוכניות התקנה וביטול התקנה          |
| 260 | תוכנית התקנה – המפתח להצלחה         |
| 261 | ניסיון הפעלה ראשון חייב להיות מוצלח |
| 262 | תוכנית הסרה – אסור לזלזל בה         |
| 262 | שימוש נכון במערכת הקבצים            |
| 263 | תמיכה ב- My Documents               |
| 263 | אחסון קבצי תוכניות                  |
| 263 | תמיכה בנתוני יישומים                |
| 264 | אחסון זמני – הנחיות                 |
| 264 | שמות ברורים לקבצים                  |
| 265 | שימוש בממשק משתמש גרפי              |
| 265 | תפריט התחל                          |
| 265 | סמלים בשולחן העבודה                 |
| 266 | התחלה נכונה                         |
| 266 | שימוש באזור ההודעות שבשורת המצב     |
| 267 | עיצוב הודעות                        |
| 268 | הזדמנויות שאפשר להחמיץ              |
| 268 | המערכת אינה מקומית בלבד             |
| 269 | חשיבות תמיכת המקלדת                 |
| 269 | עיצוב חלונות משניים                 |
| 270 | תמיכה בהעדפות צבע וגופן של המשתמש   |
| 270 | שימוש ראוי בצבע                     |
| 270 | תמיכה במסך כפול                     |
| 271 | תמיכה בקבוצות של אותיות             |
| 271 | רשימת תיוג לבניית ממשק טוב          |
| 273 | עקרונות עיצוב ממוקד-משתמש           |
| 273 | שליטת המשתמש                        |
| 273 | גישה ישירה                          |
| 274 | עקביות                              |
| 275 | סלחנות                              |
| 275 | משוב                                |
| 275 | אסתטיקה                             |
| 275 | פשטות                               |
| 276 | מתודולוגיה של עיצוב                 |
| 276 | קבוצת עיצוב מאוזנת                  |
| 276 | מעגל העיצוב                         |
| 277 | עיצוב                               |
| 278 | אבטיפוס                             |
| 279 | בדיקה                               |
| 279 | בדיקה חוזרת של התהליך               |
| 280 | הערכת שימושיות בהליך העיצוב         |
| 280 | טכניקות לבדיקת שימושיות             |
| 281 | טכניקות הערכה אחרות                 |

|     |                              |
|-----|------------------------------|
| 282 | הבנת המשתמשים                |
| 283 | חלופות עיצוב                 |
| 283 | קלט – עקרונות בסיסיים        |
| 283 | קלט העכבר                    |
| 283 | מצביעי העכבר                 |
| 284 | צורות מצביע נפוצות           |
| 285 | פעולות עכבר                  |
| 285 | פעולות עכבר נפוצות           |
| 286 | קלט המקלדת                   |
| 286 | פעולות מקלדת נפוצות          |
| 287 | מקשי טקסט                    |
| 287 | מקשי גישה                    |
| 288 | מקשי מצב                     |
| 289 | מקשי קיצור דרך               |
| 290 | עיצוב ויזואלי                |
| 290 | תקשורת ויזואלית              |
| 290 | עיצוב וארגון                 |
| 291 | היררכיה של המידע             |
| 291 | מיקוד והדגשה                 |
| 292 | מבנה ואיזון                  |
| 292 | יחסים בין רכיבים             |
| 292 | קריאות וזרימה                |
| 292 | אחידות ושילוב                |
| 292 | צבע                          |
| 293 | צבע כצורה משנית של מידע      |
| 293 | שימוש בקבוצה מוגבלת של צבעים |
| 294 | אפשרויות לשינוי צבע          |
| 294 | גופנים                       |
| 295 | רב-ממדיות                    |
| 295 | עיצוב רכיבים ויזואליים       |
| 296 | קלט דרוש ואופציונלי          |
| 296 | שיקולי עיצוב מיוחדים         |
| 296 | צליל                         |
| 297 | נגישות                       |
| 298 | נושאים נוספים                |
| 298 | סיכום                        |

## **פרק 11: עיצוב ממשק באינטרנט** 301

|     |                                 |
|-----|---------------------------------|
| 303 | הקדמה – שימושיות ב-Web לשם מה?  |
| 304 | עיצוב דף                        |
| 304 | נכסי מסך                        |
| 306 | עיצוב לפלטפורמות מרובות         |
| 308 | מהיכן מגיעים המשתמשים?          |
| 308 | עיצוב שאינו תלוי ברזולוציית מסך |

|     |                                      |
|-----|--------------------------------------|
| 309 | שימוש בתוכן לא סטנדרטי               |
| 310 | ענייני שדרוג                         |
| 312 | הפרדת המשמעות מהתצוגה                |
| 313 | זמני תגובה                           |
| 314 | חיזוי זמני תגובה                     |
| 315 | הורדות מהירות, חיבורים מהירים        |
| 317 | צפייה במסך הראשון                    |
| 317 | קישורים                              |
| 317 | תיאור הקישורים                       |
| 320 | צביעת הקישורים                       |
| 321 | ציפיות מקישורים                      |
| 322 | קישורי חוץ                           |
| 324 | גיליונות סגנון                       |
| 325 | אחידות בעיצוב באמצעות גיליונות סגנון |
| 326 | בדיקת גיליונות הסגנון                |
| 327 | אמינות                               |
| 328 | עיצוב תוכן                           |
| 329 | כתיבה ל-Web                          |
| 330 | קצר את הטקסט                         |
| 330 | עריכה                                |
| 331 | סריקה                                |
| 331 | שפה ברורה                            |
| 332 | חלוקת דפים                           |
| 333 | כותרות דפים                          |
| 333 | כתיבת כותרות                         |
| 335 | קריאות (Legibility)                  |
| 336 | תיעוד מקוון                          |
| 338 | מולטימדיה                            |
| 339 | זמן תגובה                            |
| 339 | תמונות                               |
| 340 | הקטנת התמונה                         |
| 340 | הנפשה                                |
| 341 | רצף במעברים                          |
| 341 | ציון ממדים במעברים                   |
| 342 | הדגמת שינוי לאורך זמן                |
| 342 | ריבוב תצוגה                          |
| 342 | העשרת מצגות גרפיות                   |
| 342 | המחשת מבנים תלת-ממדיים               |
| 343 | משיכת תשומת לב                       |
| 343 | וידאו                                |
| 344 | קול                                  |
| 345 | גרפיקה תלת-ממדית                     |
| 346 | מתי להשתמש בתלת-ממד                  |
| 347 | תרגול והפנמה                         |

## 355 ..... חלק 2: אפיון ועיצוב בסיסי נתונים

### 356 ..... הקדמה

#### 359 ..... פרק 1: מסדי נתונים כחלק מניתוח המערכת

|     |  |
|-----|--|
| 360 | אינטגרציה עם מתודולוגיות האפיון הקלאסי |
| 360 | הקשר ל-UML                             |
| 361 | אפיון ועיצוב בסיסי נתונים              |
| 362 | בעלי התפקידים                          |
| 362 | ארכיטקט בסיס הנתונים                   |
| 362 | מנתח מערכות                            |
| 363 | מנהל בסיס הנתונים                      |
| 363 | מומחה טכנולוגי ספציפי                  |
| 364 | שיפור בסיסי נתונים קיימים – BPR        |
| 364 | סיכום                                  |

#### 365 ..... פרק 2: רקע ומושגי יסוד

|     |  |
|-----|--|
| 366 | מבוא                                       |
| 368 | רקע  |
| 371 | מערכת מידע                                 |
| 372 | רכיבי מערכת המידע                          |
| 372 | רכיב קליטת תנועות (Transaction Processing) |
| 373 | רכיב עיבוד הנתונים (Data Processing)       |
| 373 | רכיב הצגת הנתונים (Data Presentation)      |
| 373 | רכיב ניהול הנתונים (Data Management)       |
| 374 | יישום (Application)                        |
| 375 | תוכניות יישום (Application Programs)       |
| 377 | מערכות תפעוליות ומערכות מחסן נתונים        |
| 379 | סיכום                                      |
| 380 | שאלות חזרה ותרגילים                        |
| 380 | שאלות חזרה                                 |
| 380 | תרגילים                                    |

#### 381 ..... פרק 3: עקרונות מערכות לניהול בסיסי נתונים

|     |   |
|-----|---|
| 382 | מבוא  |
| 383 | ציוני דרך עיקריים בהתפתחות מערכות לניהול נתונים |
| 383 | מערכות בסיסיות לניהול נתונים                    |
| 384 | מערכות לניהול קבצים – FMS                       |
| 385 | מערכות לניהול בסיסי נתונים – דור ראשון          |
| 386 | מערכות לניהול בסיסי נתונים טבלאיים              |
| 387 | מערכות לניהול בסיסי נתונים מוכווני עצמים        |
| 387 | מערכות ניהול קבצים                              |
| 391 | מערכות לניהול בסיסי נתונים                      |

|     |   |
|-----|---|
| 394 | מודל העבודה של מערכת RDBMS                          |
| 395 | סכמה גלובלית (Global Schema)                        |
| 397 | תת-סכמה (View)                                      |
| 399 | סכמה פיזית (Physical Schema)                        |
| 401 | בסיס הנתונים (Database)                             |
| 401 | קטלוג המערכת (System Catalog)                       |
| 402 | תוכניות יישום (Application Programs)                |
| 403 | תפקידים שונים בסביבת בסיס נתונים                    |
| 403 | מנהל בסיס הנתונים                                   |
| 405 | מעצבי בסיסי נתונים (Database Designers)             |
| 405 | מהנדסי יישום (Application Engineers)                |
| 405 | משתמשי קצה (End Users)                              |
| 406 | עיקרון הפעולה של מערכות RDBMS                       |
| 406 | הגדרת בסיס הנתונים (Database Definition)            |
| 407 | גישה לבסיס הנתונים                                  |
| 407 | גישה אינטראקטיבית (Interactive Access)              |
| 410 | גישה מתוכנית יישום הכתובה בשפת תכנות מדור שלישי     |
| 412 | גישה מתוך תוכנית הכתובה בשפת תכנות מדור רביעי       |
| 412 | מחוללי שאילתות ודוחות (Query and Report Generators) |
| 412 | גישה משרת Web                                       |
| 413 | מודל העבודה של מערכת RDBMS                          |
| 416 | שירותים במערכת לניהול בסיסי נתונים                  |
| 416 | אחסון, עדכון ושליפה                                 |
| 417 | מבנה לוגי ופיסי של נתונים                           |
| 418 | אי-תלות לוגית ופיזית                                |
| 419 | ניהול קטלוג המערכת                                  |
| 420 | המשתמש במערכת המידע                                 |
| 420 | מנתח המערכות  |
| 421 | מהנדס תוכנה   |
| 421 | מנהל בסיס הנתונים                                   |
| 421 | תמיכה בעיבוד תנועות ועדכון מרובה משתמשים            |
| 421 | אבטחת נתונים  |
| 422 | יתרונות וחסרונות של טכנולוגיית בסיסי הנתונים        |
| 422 | יתרונות   |
| 423 | חסרונות   |
| 424 | סיכום   |
| 425 | שאלות חזרה ותרגילים                                 |
| 425 | שאלות חזרה  |
| 426 | תרגילים   |

## **פרק 4: אפיון בסיסי נתונים** 427

|     |                                 |
|-----|---------------------------------|
| 428 | מבוא                            |
| 429 | מודל הנתונים (Data Model)       |
| 430 | תהליך אפיון ועיצוב מודל הנתונים |

|     |       |  |
|-----|-------|--|
| 432 | ..... | המודל התפישתי (Conceptual Model)                   |
| 433 | ..... | המודל הלוגי (Logical Model)                        |
| 434 | ..... | המודל הפיסי (Physical Model)                       |
| 434 | ..... | אי תלות לוגית ופיסית                               |
| 435 | ..... | מודל ישויות-קשרים (Entity Relationship Data Model) |
| 436 | ..... | ישויות   |
| 437 | ..... | ישות חזקה וישות חלשה                               |
| 437 | ..... | תכונה (Attribute)                                  |
| 438 | ..... | תכונה פשוטה (אטומית) ותכונה מורכבת                 |
| 438 | ..... | תכונה מחושבת                                       |
| 439 | ..... | נתון מובנה   |
| 439 | ..... | נתון לא מובנה                                      |
| 440 | ..... | טיפוס נתונים (Data Type)                           |
| 441 | ..... | ערך של תכונה (Attribute Value)                     |
| 441 | ..... | ערך חובה – Mandatory                               |
| 441 | ..... | תכונה בעלת ערך בודד או מרובת ערכים                 |
| 442 | ..... | מרחב ערכים (Attribute Domain)                      |
| 444 | ..... | קבוצת ישויות (Entity Type)                         |
| 446 | ..... | מפתחות ישות (Entity Keys)                          |
| 448 | ..... | קשרים  |
| 449 | ..... | פונקציונליות הקשר                                  |
| 449 | ..... | קשר חד-חד ערכי (1: 1)                              |
| 450 | ..... | קשר חד-רב ערכי (1: M)                              |
| 450 | ..... | קשר רב-רב ערכי (M: N)                              |
| 450 | ..... | קרדינליות הקשר                                     |
| 451 | ..... | תלות קיומית  |
| 452 | ..... | קשר נושא מידע                                      |
| 454 | ..... | קשר רקורסיבי                                       |
| 455 | ..... | ישות על (SuperType) ותת-ישות (SubType)             |
| 457 | ..... | יחס בחירה בין קשרים                                |
| 458 | ..... | חיי ישות לאורך זמן                                 |
| 460 | ..... | העברת קשרים  |
| 460 | ..... | שרטוט תרשים ישויות-קשרים                           |
| 461 | ..... | נרמול נתונים (Data Normalization)                  |
| 461 | ..... | היסטוריה   |
| 461 | ..... | הגדרה  |
| 461 | ..... | רמות הנרמול  |
| 462 | ..... | נרמול נתונים – לשם מה?                             |
| 463 | ..... | אנומליה  |
| 464 | ..... | שימור מידע ושימור תלויות                           |
| 464 | ..... | תלות פונקציונלית                                   |
| 465 | ..... | רמות נרמול   |
| 465 | ..... | רמת נרמול ראשונה – 1NF                             |
| 466 | ..... | רמת נרמול שנייה – 2NF                              |

|     |                                |
|-----|--------------------------------|
| 467 | רמת נרמול שלישית – 3NF         |
| 468 | רמת נרמול נוספת – BCNF         |
| 469 | סיכום רמות הנרמול              |
| 470 | שילוב מודל הנתונים בנוהל מפת"ח |
| 470 | תרגיל מסכם                     |
| 471 | חברת הביטוח "פרוטקשיין" בע"מ   |
| 472 | פתרון בשלבים                   |
| 476 | תרגיל להערכה עצמית             |
| 476 | חברת השכרת רכב "רייס" בע"מ     |

## **פרק 5: עיצוב בסיסי נתונים – המעבר לטבלאות 477**

|     |  |
|-----|--|
| 478 | מבוא   |
| 478 | תהליך עיצוב מודל הנתונים                       |
| 479 | המודל הלוגי (Logical Model)                    |
| 479 | מודל הנתונים הטבלאי (Relational Data Model)    |
| 481 | נרמול נתונים בשלב העיצוב                       |
| 481 | הקדמה  |
| 481 | צורת נרמול רביעית – 4NF                        |
| 483 | צורת נרמול חמישית – 5NF                        |
| 485 | המעבר לטבלאות                                  |
| 485 | תרשים דוגמה לטבלה (TIC – Table Instance Chart) |
| 486 | שלבי עבודה במעבר לטבלאות                       |
| 486 | מיפוי ישויות פשוטות לטבלה                      |
| 487 | מיפוי תכונות                                   |
| 487 | הכנת נתוני דמה                                 |
| 487 | מפתחות זרים (Foreign Key)                      |
| 491 | מיפוי קשר רקורסיבי                             |
| 492 | מיפוי קשתות                                    |
| 495 | תרגום ישויות-על ותת-ישויות                     |
| 496 | תת-ישויות – טבלה מאוחדת                        |
| 498 | תת-ישויות – פיצול טבלאות                       |
| 500 | תת-ישויות – יישום באמצעות קשת                  |
| 502 | ההחלטה על היישום                               |
| 502 | תרגיל מסכם                                     |
| 503 | חברת הביטוח "פרוטקשיין" בע"מ                   |
| 503 | פתרון  |
| 508 | סיכום  |

## **פרק 6: בסיסי נתונים מוכווני עצמים (Object Oriented Databases) .... 509**

|     |  |
|-----|--|
| 510 | מבוא   |
| 513 | סקירת החסרונות העיקריים של המודל הטבלאי          |
| 515 | שפה לטיפול באובייקטים                            |
| 516 | מערכות ODBMS                                     |
| 517 | סקירת ההבדלים בין מערכות RDBMS לבין מערכות ODBMS |

|     |  |
|-----|--|
| 518 | מערכות Object-Relational DBMS            |
| 519 | טיפוס הנתונים (Binary Large Object) BLOB |
| 521 | סיכום                                    |
| 522 | שאלות חזרה ותרגילים                      |

## **פרק 7: שפת SQL 523**

|     |   |
|-----|---|
| 525 | פתח דבר – הרקע להתפתחות השפה                                |
| 527 | מבוא  |
| 528 | שפת SQL כשפה תקנית  |
| 532 | אי-תלות בנתונים בשפת SQL                                    |
| 533 | כיצד מפעילים פקודות SQL                                     |
| 534 | סוגי פקודות SQL   |
| 534 | פקודות להגדרת בסיס הנתונים (Data Definition)                |
| 535 | פקודות לטיפול בנתונים (Data Manipulation)                   |
| 535 | פקודות לבקרת גישה (Data Access Control)                     |
| 535 | פקודות לבקרת תנועות (Transaction Control)                   |
| 536 | פקודות מיוחדות לשילוב שפת SQL בשפה מארחת (Programmatic SQL) |
| 536 | אבני הבניין של פקודת SQL                                    |
| 539 | טיפוסי נתונים (Data Types)                                  |
| 540 | שיטה לתיאור מבנה פקודות SQL                                 |
| 541 | בסיס נתונים לדוגמה  |
| 544 | פקודות לטיפול בנתונים (Data Manipulation)                   |
| 544 | הפקודה Select   |
| 544 | מבנה כללי של הפקודה Select                                  |
| 545 | תרשים תחביר של הפקודה Select                                |
| 546 | שאליות לטבלה אחת (Single Table Queries)                     |
| 546 | שליפת כל העמודות וכל השורות                                 |
| 547 | שליפת שורות ועמודות מסוימות                                 |
| 548 | שינוי שם עמודה  |
| 549 | שליפת שורות ללא הצגת שורות כפולות                           |
| 550 | בחירת שורות (Row Selection)                                 |
| 551 | בדיקת השוואה (Comparison Test)                              |
| 552 | בדיקה השוואה עם תנאי בוליאני (Boolean Condition)            |
| 553 | בדיקת טווח ערכים רציף (Range Test)                          |
| 554 | בדיקת קיום ערך בתוך קבוצת ערכים (Set Membership Test)       |
| 555 | בדיקת מחרוזת (Pattern Matching)                             |
| 558 | עמודות מחושבות (Calculated Columns)                         |
| 560 | הוספת כיתוב קבוע בשורות המוצגות                             |
| 560 | פונקציות מובנות (Built-in Functions)                        |
| 563 | אריתמטיקה של תאריכים (Date Arithmetics)                     |
| 564 | מיון התוצאה (ORDER BY)                                      |
| 566 | שאליות מקובצות (Grouped Queries)                            |
| 569 | שאליות עם מספר טבלאות (Multi-table Queries)                 |
| 572 | דוגמאות לשאליות   |

|     |       |  |
|-----|-------|--|
| 576 | ..... | Outer Join על ידי טבלאות                     |
| 577 | ..... | (Self Join) צירוף טבלה אל עצמה               |
| 578 | ..... | (Cartesian Product) מכפלה קרטזית בין טבלאות  |
| 579 | ..... | (Sub Queries) תת-שאליות                      |
| 580 | ..... | (Subquery Comparison Test) בדיקת תנאי השוואה |
| 582 | ..... | (Set Membership Test) בדיקת קבוצת ערכים      |
| 583 | ..... | (Any) השוואת ערך בודד מול ערך כלשהו          |
| 584 | ..... | (All) השוואת ערך בודד מול כל הערכים          |
| 586 | ..... | (Correlated Sub Queries) תת-שאליות מתואמות   |
| 587 | ..... | (Existence Test) בדיקת קיום                  |
| 589 | ..... | HAVING תת-שאליות במשפט                       |
| 589 | ..... | (Union) איחוד תוצאות של שאליות               |
| 591 | ..... | (Intersect) חיתוך תוצאות של שאליות           |
| 592 | ..... | פקודות לעדכון בסיס הנתונים                   |
| 592 | ..... | (Single Row Insert) הוספת שורה בודדת         |
| 593 | ..... | (Multi-Row Insert) הוספת מספר שורות          |
| 594 | ..... | (Update) עדכון שורות                         |
| 595 | ..... | (Delete) ביטול שורות                         |
| 596 | ..... | (Null Values) טיפול בערכים חסרים             |
| 598 | ..... | (Three Valued Logic) לוגיקה תלת-ערכית        |
| 599 | ..... | טיפול בערכים חסרים על ידי פקודות SQL         |
| 599 | ..... | הוספת שורות עם ערכים חסרים                   |
| 600 | ..... | (Null Value Test) בדיקת ערכים חסרים          |
| 601 | ..... | בדיקת ביטוי לוגי                             |
| 601 | ..... | (Between Test) בדיקת טווח                    |
| 602 | ..... | (Join) צירוף טבלאות                          |
| 602 | ..... | (Built In Functions) פונקציות מובנות         |
| 603 | ..... | (Grouped Queries) שאליות עם הקבצות           |
| 604 | ..... | סיכום  |
| 605 | ..... | שאלות חזרה ותרגילים                          |
| 605 | ..... | שאלות חזרה                                   |
| 606 | ..... | תרגילים                                      |

**פרק 8: מחסני נתונים (Data Warehouse) ..... 607**

|     |       |   |
|-----|-------|---|
| 608 | ..... | הקדמה   |
| 610 | ..... | רקע ומושגי יסוד   |
| 610 | ..... | מבוא  |
| 613 | ..... | המגמות האסטרטגיות העיקריות בטכנולוגיית המידע                |
| 614 | ..... | המגמות העיקריות בתחום הארכיטקטורה היישומית                  |
| 617 | ..... | המגמות העיקריות בתחומי טכנולוגיית המידע                     |
| 617 | ..... | אפיון סוגי היישומים בארגון                                  |
| 617 | ..... | כללי  |
| 618 | ..... | (Operational Business Applications) יישומים עסקיים תפעוליים |
| 618 | ..... | (Decision Support Applications) יישומים תומכי החלטות        |

|     |   |
|-----|---|
| 619 | השוואת מאפייני שני סוגי היישומים  |
| 623 | המסקנה: שתי סביבות שונות, אבל משולבות.  |
| 624 | השילבים בגיבוש תפיסת מחסן הנתונים   |
| 624 | כללי  |
| 625 | השילבים בהתפתחות מערכות לאספקת מידע<br>מחוללי דוחות ושאלות לגישה לנתונים תפעוליים |
| 625 | (Report Writer and Query Generator)   |
| 626 | סביבות ייעודיות לניתוחים סטטיסטיים במחשבים מרכזיים                                |
| 626 | מערכות תומכות החלטה (DSS – Decision Support Systems)                              |
| 627 | מערכות מידע למנהלים (EIS – Executive Information Systems)                         |
| 628 | מרכזי מידע (Information Centers)  |
| 629 | סביבות תומכות החלטות מבוססות מחשבי אישיים   |
| 630 | מחסן הנתונים (Data Warehouse)   |
| 631 | ארכיטקטורת מחסן הנתונים   |
| 631 | מבוא  |
| 631 | מהו מחסן נתונים?  |
| 634 | הארכיטקטורה   |
| 635 | ארכיטקטורת מחסן נתונים ארגוני (Enterprise Data Warehouse Architecture)            |
| 638 | ארכיטקטורת מרכז הנתונים (Data Mart Architecture)                                  |
| 639 | מחסן נתונים רב-שכבתי (Multi Tier Data Warehouse)                                  |
| 641 | ארכיטקטורת מאגר נתונים תפעולי (ODS – Operational Data Store)                      |
| 642 | מחסן נתונים מדומה (Virtual Data Warehouse)  |
| 644 | סיכום   |

## **פרק 9: כריית נתונים (Data Mining)**

|     |  |
|-----|--|
| 648 | מבוא   |
| 648 | מהי כריית נתונים                                   |
| 651 | תחומי המחקר שתרמו להתפתחות כריית הנתונים           |
| 652 | סטטיסטיקה  |
| 652 | אינטליגנציה מלאכותית                               |
| 652 | מערכות לומדות (Machine Learning)                   |
| 653 | לימוד והיסק  |
| 653 | יישומים, טכניקות, אלגוריתמים ומוצרים לכריית נתונים |
| 654 | יישומים (Applications)                             |
| 656 | טכניקות (Techniques)                               |
| 656 | הצגה ויזואלית של נתונים (Visualization Techniques) |
| 657 | הסקת חוקים ויחסים בין הנתונים (Association Rules)  |
| 657 | ניתוח סדר הופעה (Sequence Based Analysis)          |
| 658 | ניתוח אשכולות (Cluster Analysis)                   |
| 658 | סיווג (Classification)                             |
| 659 | עצי החלטה (Decision Trees)                         |
| 662 | רשתות עצביות (Neural Networks)                     |
| 665 | אומדן (Estimation)                                 |
| 665 | טכניקות אחרות                                      |

|     |  |
|-----|--|
| 665 | טבלת סיכום היתרונות והחסרונות של הטכניקות השונות |
| 666 | אלגוריתמים ומודלים (Algorithms and Models)       |
| 666 | מוצרים לכריית נתונים (Data Mining Tools)         |
| 667 | תהליך כריית הנתונים                              |
| 669 | אינטגרציה בין מחסן הנתונים לכריית הנתונים        |
| 671 | בעיות עיקריות בכריית הנתונים                     |
| 672 | סיכום  |

## **673 חלק 3: אבטחת איכות ובדיקות תוכנה**

### **675 הקדמה**

#### **677 פרק 1: הנדסת איכות תוכנה**

|     |                                 |
|-----|---------------------------------|
| 678 | תוכנה, איכות והנדסה             |
| 678 | התוכנה מהי                      |
| 678 | האיכות מהי                      |
| 679 | מה ההקשר של ההנדסה לכאן         |
| 680 | שמונה עקרונות הנדסיים חשובים    |
| 681 | הנדסת תוכנה                     |
| 682 | שמונה עקרונות של הנדסת התוכנה   |
| 684 | איכות תוכנה                     |
| 684 | הגדרת דרישות                    |
| 685 | השגת איכות התוכנה               |
| 685 | תהליכי איכות                    |
| 686 | תהליכי הנדסת תוכנה              |
| 686 | מהו הדבר שמייחד את הנדסת התוכנה |
| 687 | מודלים המבוססים על מחזור חיים   |

#### **690 פרק 2: מערכות לניהול איכות – לשם מה?**

|     |                                     |
|-----|-------------------------------------|
| 691 | אתגר האיכות                         |
| 691 | איכות בתוכנה                        |
| 693 | מערכות איכות                        |
| 694 | מערכות לניהול איכות                 |
| 695 | סדרת התקנים ISO 9000                |
| 695 | מה מיוחד כל כך בתוכנה               |
| 695 | הקווים המנחים ISO 9000-3            |
| 696 | ISO 9000-3 – פירוט הסעיפים העיקריים |
| 697 | איכות תוכנה – עובדות החיים          |
| 697 | עקרונות איכות תוכנה                 |

#### **698 פרק 3: ניהול איכות, סיכונים ופרויקטים**

|     |                                  |
|-----|----------------------------------|
| 699 | תוכנית איכות                     |
| 701 | מה צריכה להכיל תוכנית איכות טובה |

|            |   |
|------------|---|
| <b>703</b> | <b>פרק 4: תהליך הפיתוח הבסיסי</b>               |
| 704        | מבוא  |
| 704        | מהו התהליך הבסיסי                               |
| 704        | מחזורי חיים                                     |
| 705        | מודל מפל המים                                   |
| 706        | מודל החילזון                                    |
| 707        | מודל מסוג V                                     |
| 707        | מתודולוגיות                                     |
| 709        | אימות ובדיקת תקפות                              |
| 711        | ניהול תצורה                                     |
| 712        | תהליך ניהול התצורה                              |
| 712        | זיהוי ויכולת מעקב                               |
| 713        | בקרת שינויים                                    |
| 715        | תכנון ניהול תצורה                               |
| 715        | תחזוקת תוכנה                                    |
| 715        | בעיות תחזוקת התוכנה                             |
| 716        | כיצד תומך תקן ISO 9000-3 ברעיונות מפתח אלה      |
| 716        | הנחייה בנושא מחזור החיים                        |
| 716        | הנחייה בנושא מתודולוגיות                        |
| 717        | הנחיות בנושא אימות                              |
| 717        | הגדרות של תקן ISO 9000-3 לאימות ולבדיקת התקפות  |
| 718        | הנחיות בנושא סקרים                              |
| 718        | הנחיות בנושא בדיקות והוכחת התקפות               |
| 719        | הנחיות בנושא ניהול תצורה                        |
| 719        | הנחיות בנושא תחזוקת התוכנה                      |
| 719        | נהלי פיתוח תוכנה                                |
| 720        | מתודולוגיות מעשיות                              |
| 721        | סיווג הפרויקטים                                 |
| 721        | כללי תיכון                                      |
| 722        | נהגי תכנות                                      |
| 723        | פיתוח המבוסס על טכנולוגית 'הדור הרביעי'         |
| 723        | פיתוח מבוסס חבילות יישומים סטנדרטיות או מותאמות |
| 723        | בעיות אופייניות לחבילות יישומים                 |
| 724        | הבחירה והשימוש בכלי תוכנה                       |
| 724        | הערכה ובחירה של כלים                            |
| 725        | קריטריונים לבחירת כלים                          |
| 725        | ההכנה וההרצה של כלי התוכנה                      |
| 726        | ניהול הכלים                                     |
| 726        | פיתוח כלים בארגון                               |
| 727        | ניסוי   |
| 727        | רמות הבדיקה                                     |
| 727        | תכנון הבדיקה                                    |
| 728        | מה אמורה תוכנית בדיקה להכיל                     |

|     |  |
|-----|--|
| 729 | מפרט הבדיקה ותיכון הביצוע                                    |
| 731 | בקרה ורישום של בדיקות  |
| 732 | קבלת המוצר   |
| 733 | כיצד משתלב תהליך הפיתוח הבסיסי במערכת ניהול איכות (QMS) שלנו |
| 733 | אלמנטים מרכזיים של תהליך פיתוח                               |

## **פרק 5: עיקרי מערכת האיכות – לב המערכת ..... 735**

|     |  |
|-----|--|
| 736 | מבוא   |
| 736 | מהו עיקר מערכת איכות                             |
| 737 | פעילויות עיקריות במערכת האיכות                   |
| 739 | תקן ISO 9000-3 ומה שמעבר לו – עיקרי מערכת האיכות |
| 740 | מדיניות האיכות                                   |
| 741 | שש כותרות למדיניות האיכות                        |
| 742 | ארגון ותחומי אחריות                              |
| 742 | נציג ההנהלה                                      |
| 742 | הדרכה  |
| 744 | מבדקי איכות פנימיים                              |
| 746 | פעולה מתקנת                                      |
| 747 | עשרה צעדים בדרך לפעולה מתקנת יעילה               |
| 748 | סקר חוזר של ההנהלה                               |
| 748 | מדידות ומדדים                                    |
| 749 | בקרת התייעוד                                     |
| 749 | רשומות האיכות                                    |
| 749 | רכש  |
| 750 | שיפור התהליך                                     |
| 750 | עשרה צעדים – מתחזוקה לשיפור מתמשך                |

## **פרק 6: שיפור תהליכים ..... 751**

|     |  |
|-----|--|
| 752 | הצבת מטרות מציאותיות לשיפורים          |
| 752 | שבעה צעדים קלים כדי להיות 'הטוב ביותר' |
| 752 | מחזור PDCA                             |
| 753 | טכניקות וכלים                          |
| 758 | מה בדבר איכות המוצר?                   |

## **פרק 7: בדיקות תוכנה ..... 759**

|     |                     |
|-----|---------------------|
| 760 | מבוא                |
| 760 | סוגי הבדיקות        |
| 761 | תכנון הבדיקות       |
| 762 | בדיקות פונקציונליות |
| 762 | תנאים               |
| 762 | דרישות סביבתיות     |
| 762 | קלט                 |
| 762 | שיטה                |
| 763 | פלטים               |

|     |                                   |
|-----|-----------------------------------|
| 765 | מטריצת כיסוי                      |
| 766 | בדיקות אינטגרציה                  |
| 766 | תנאים                             |
| 766 | דרישות סביבתיות                   |
| 766 | קלט                               |
| 766 | שיטה                              |
| 767 | פלטים                             |
| 768 | בדיקות מערכת                      |
| 768 | תנאים                             |
| 768 | דרישות סביבתיות                   |
| 768 | קלט                               |
| 768 | שיטה                              |
| 769 | ביצועים                           |
| 769 | קצב נתונים                        |
| 769 | עומס ונפח נתונים                  |
| 769 | אבטחת המידע                       |
| 770 | גיבוי והתאוששות                   |
| 770 | פלטים                             |
| 771 | בדיקות קבלה                       |
| 771 | תנאים                             |
| 771 | דרישות סביבתיות                   |
| 772 | קלט                               |
| 772 | שיטה                              |
| 772 | פלטים                             |
| 773 | תוכנית בדיקות קבלה                |
| 773 | טבלת גרסאות                       |
| 774 | זיהוי תוכנית הבדיקות              |
| 774 | מבוא                              |
| 774 | רכיבי בדיקה                       |
| 775 | רכיבים לא לבדיקה                  |
| 776 | פונקציות ותהליכים לבדיקה          |
| 776 | פונקציות ותהליכים לא לבדיקה       |
| 776 | גישה                              |
| 776 | כללי                              |
| 777 | בדיקות פונקציונליות               |
| 777 | בדיקות תהליכים                    |
| 777 | בדיקות הסבה                       |
| 777 | בדיקת ממשקים חיצוניים             |
| 777 | בדיקות ביצועים                    |
| 778 | כלים בשימוש                       |
| 778 | תנאים להשלמת הבדיקות              |
| 778 | אילוצים לבדיקות                   |
| 778 | קריטריונים להצלחה / כישלון        |
| 779 | קריטריונים לעיכוב וחזרה על בדיקות |

|     |   |
|-----|---|
| 779 | תוצרי הבדיקות                             |
| 780 | פעילויות מכינות                           |
| 780 | דרישות סביבתיות                           |
| 780 | אחריות                                    |
| 780 | מנהל הפרויקט                              |
| 780 | אחראי בדיקות                              |
| 781 | צוות הבדיקות                              |
| 781 | הלקוח                                     |
| 781 | צוות תשתיות פרויקטלי                      |
| 781 | צוותי הפיתוח                              |
| 782 | מנהל אבטחת איכות                          |
| 782 | משאבים ודרישות הדרכה                      |
| 783 | לו"ז                                      |
| 783 | ניתוח סיכונים                             |
| 783 | תחלופת משתמשים עיקריים                    |
| 783 | השלכות                                    |
| 783 | ניהול הסיכון                              |
| 783 | אין תשתית מסודרת בגוף הפיתוח              |
| 783 | השלכות                                    |
| 783 | ניהול הסיכון                              |
| 784 | שינוי בהגדרת הדרישות בשלב מתקדם של הפיתוח |
| 784 | השלכות                                    |
| 784 | ניהול הסיכון                              |
| 784 | כוח אדם לא מיומן בביצוע הבדיקות           |
| 784 | השלכות                                    |
| 784 | ניהול הסיכון                              |
| 784 | לוח זמנים קצר לבדיקות                     |
| 784 | השלכות                                    |
| 784 | ניהול הסיכון                              |
| 784 | אימות כיסוי הבדיקות                       |
| 785 | אישורים                                   |
| 786 | סיכום בדיקות קבלה                         |
| 786 | טבלת גרסאות                               |
| 786 | זיהוי מסמך סיכום בדיקות קבלה              |
| 787 | סיכום                                     |
| 787 | תמצית מנהלים                              |
| 787 | סקירת רכיבים שנבדקו                       |
| 787 | שונות                                     |
| 788 | הערכת שלמות הבדיקות                       |
| 788 | שלמות הבדיקות לעומת הקריטריונים לשלמות    |
| 788 | פונקציות / תהליכים שלא נבדקו כראוי        |
| 788 | סיכום תוצאות                              |
| 788 | תוצאות הבדיקות                            |
| 788 | זיהוי התקלות שנפתרו ודרך פתרון            |

|     |   |
|-----|---|
| 788 | זיהוי התקלות שלא נפתרו                      |
| 789 | הערכה                                       |
| 789 | סיכום פעילויות                              |
| 789 | סיכום עיקר פעילויות הבדיקה והארועים הנלווים |
| 789 | סיכום שימוש במשאבים                         |
| 790 | אישורים                                     |

## **791 ..... פרק 8: שיקוף – סקר (Review)**

|     |                          |
|-----|--------------------------|
| 792 | מבוא                     |
| 792 | הגדרה                    |
| 792 | מטרות השיקוף             |
| 792 | זיהוי כשלים              |
| 792 | שיפור                    |
| 793 | עמידה בסטנדרטים          |
| 793 | אחידות פיתוח             |
| 793 | שיתוף מידע וידע          |
| 793 | שיטה                     |
| 793 | הכנות                    |
| 794 | משתתפים                  |
| 794 | ביצוע                    |
| 795 | סיכום השיקוף             |
| 795 | ביצוע חוזר (במקרה הצורך) |
| 795 | משך                      |
| 795 | תוצרים                   |

## **797 ..... פרק 9: תרגיל מסכם**

|     |            |
|-----|------------|
| 797 | תיאור מצב  |
| 797 | חיים קשים  |
| 797 | איכות ונחת |

## **799 ..... מילון מונחים**

## **809 ..... אינדקס**

# הקדמה

## סוף מעשה במחשבה תחילה

אתה חייב להכניס לארגון שלך מערכת מחשוב. ממש חייב! כי אם לא תעשה זאת – לא תשרוד בשוק. הרי כבר לכל המתחרים יש מערכת מחשוב חדשה שמנהלת את העסק. מנהלת לבד, כך הם טוענים. הנה המנכ"ל של החברה המתחרה נמצא בחופשה, ואיך הוא יכול להרשות זאת לעצמו עם כל הלחץ שיש בעבודה? אין ספק שאתה מנכ"ל טוב, אז גם אתה רוצה לנהוג כך.

כדי להשיג את מטרותך הבאות שני תוכניתנים מומחים! הרי יש להם תואר במשהו. אז מה אם הם נראים ילדים? הקדשת בשבילם ארבע שעות. מה זה לא הרבה? בטח הרבה. חצי יום עבודה. מה קרה? יש להם ניסיון, מה הם לא הבינו את הצרכים? שילכו לעבוד, גם כן. וחיכית... וחיכית...

בינתיים המתחרים פותחים פער. הלקוחות שלהם נדבקים אליהם כמו מסטיק. כמו אזולירבנד, כמו אפוקסי. ושלך? נו טוב, הם היו קשישים גם ככה. אה...כן, גם הזוג הזה, ממילא לא הרווחתי עליהם, סתם נטל. קבוצת ליצנים. נו טוב... וחיכית עוד... ועוד חיכית...

ואז, זה קרה. קארין, המזכירה השרמנטית, היפהפיה, שעולה לך כמו שני מהנדסים וחבילת ביסלי, נכנסה. חשבת ששמעת אותה אומרת ש"הילדים" התקינו אצלה את המערכת החדשה, ובכלל אין לה מה לעשות עם זה. זה לא מה שהיא צריכה.

בעודך בוהה בקארין, צלצול הטלפון: ג'ורג' מהאספקה. נדמה לך שצעק משהו על מערכת לא מוצלחת. אה...כן, גם שוש מהכספים מלמלה היום בישיבת דירקטוריון משהו על בזבוז כסף משווע. החלטת ודי.

נתת ל"ילדים" הזדמנות שנייה, והפעם החלטת שאתה הולך עד הסוף, עד הקצה. הולך לעשות את זה כמו שצריך. לאחר שיחת הבהרה עם ה"ילדים" החלטת לתת להם שבוע נוסף לתיקון המעוות. הסכמת להגדיל את תקציב המערכת ב-20%. פינית לעצמך שעתיים נוספות לשאלות ה"ילדים". נו טוב... מפה רואים את קארין, ומהבית לא. וחיכית.

משפּשֶׁף יֵדִיךְ זֶה בְּזוֹ בְּבִטְחָה וְחוֹלֵם עַל פְּגִישַׁת הַדִּירֶקְטוֹרִיוֹן הַבֵּאָה. אוֹהוּ, מֵה שְׁיוֹשֵׁב הַרֶאֶשׁ יֵגִיד עֲלֶיךָ. אֵילוּ שְׁבַחִים, וּוְאֵי וּוְאֵי וּוְאֵי. לֹא עוֹד "בְּזָבוּז מְשׁוּעָה", לֹא עוֹד מַעֲרַכַת לֹא מוֹצֵלַחַת, לֹא עוֹד לְקוֹחוֹת לֹא מְרוֹצִים. נִהְפֹךְ הוּא...

בְּעוֹדֶךָ חוֹלֵם, חֹזֶרֶת לַהֲכֵרָה, וְרֵאִיִּתְךָ הַתְּפַקְסָה עַל שֵׁלֵט הַ"נֶחֱמָה נֶחֱמָה נֶחֱמָה מֵאוֹמֵן". לַפְתָּע הַבַּחֲנָת כִּי בְרִדְיוֹ מִנְגֵן שִׁיר: "מְשִׁיחַ לֹא יָבוֹא... מְשִׁיחַ גַּם לֹא יִצְלָל".

אַתָּה יִכּוֹל לְהַמְשִׁיךְ וּלְחַכּוֹת.

לְחַכּוֹת לְ"יֵלָדִים". לְחַכּוֹת לְצִלְצוֹל מֵהִיוֹ"ר. לְחַכּוֹת לְכֶסֶף שֶׁל הַלְקוֹחוֹת. לְחַכּוֹת לְמַעֲרַכַת חֲדָשָׁה. דְּרֶק. לֹא מַעֲרַכַת וְלֹא נַעֲלִים. אֲלֵא מֵה חֲשַׁבְתָּ?

כֶּסֶף יִכּוֹל לְעֹזוֹר, גַּם חֲצֵי יוֹם עִם הַמִּנְכָּ"ל. גַּם קָאָרִין יִכּוֹלָה לְעֹזוֹר. אֲבֵל אִם אַתָּה לֹא רוֹצֵה שֶׁנֶחֱמָה נֶחֱמָה יַעֲזוֹר, אַתָּה צָרִיךְ לְשִׁנוֹת גִּישָׁה!

נִכְנַסְתָּ לְאַלְפָּא 166 הַשְּׁחוּרָה שֶׁלְךָ. חֲצֵי סִיבּוּב בְּסוּוִיץ' וְהוֹפֵן נִדְלַקְתָּ. בְּאוֹפֵן אוֹטוֹמָטִי כֶּסֶף הַנְּהַג מִסְתַּדֵּר בְּתַנּוּחָה הַרְצוּיָה לְךָ. הָאוֹרוֹת בְּרַכֵּב כְּבִים אֵט אֵט, הַרְדִּיּוֹ נִדְלַק בְּדִיוֹק בְּדִיֶּסֶק אוֹתוֹ אַתָּה נוֹהֵג לְשִׁמוּעַ כְּשֶׁאַתָּה מְצוּבְרָח. כֵּאֵילוּ שֶׁהַבּוֹנֵבּוֹן מְרַגֵּשׁ אוֹתְךָ. הַהֵגָה מַחְלִיק כֵּאֵילוּ אֵין מִתְחַתֵּי אִסְפֵּלֵט. כְּפַתּוּרֵי הַשִּׁיּוֹט הָאוֹטוֹמָטִי נִמְצָאִים בְּדִיוֹק בְּמִקּוֹם הַנְּכוּן. אַתָּה לוֹחֵץ פּוֹל גֹּז, וְנוֹתֵן לְתַאֲוָצָה לְהַחֲזִיר אוֹתְךָ לְעוֹלְמוֹת אַחֲרִים, טוֹבִים יוֹתֵר.

שׁוֹב אַתָּה חוֹלֵם. הַפַּעַם אַתָּה חוֹלֵם עַל מַפְעַל הָאַלְפָּא בְּאִיטְלִיָּה. אַתָּה רוֹאֵה אוֹתָם בּוֹנִים אֵת הַרְכֵב שֶׁלְךָ, עֶשְׂרוֹת שָׁל אֲנָשִׁים, כּוֹלֵם בְּחִלּוּקִים לְבָנִים. כִּמְהָ נִהְגִים יוֹשְׁבִים וּמְדַבְּרִים עִם הַהֲנַהֲלָה. אַה, הֵנָּה אֲלִבְרָטוֹ מַחְבַּרְת הַסְּקָרִים. מַעֲנִיִּין, בְּכָל מִקּוֹם הוּא תִּקְוֵעַ. עַד אִיטְלִיָּה הִגִּיעַ הַתּוֹלַעַת. אַתָּה מִסְתַּכֵּל בְּשַׁעוֹן וְרוֹאֵה כִּי הַשְּׁנָה הִיא 1999. מוֹזֵר, אַתָּה אוֹמֵר לְעֶצְמְךָ, הִרִי אֵת הָאוֹטוֹ הַזֶּה הוֹצִיאֵו לְשׁוֹק רֶק ב־ 2003. מוֹזֵר עוֹד יוֹתֵר שֶׁהַשַּׁעוֹן מֵרֵאָה אֵת הַשְּׁנָה, אֵךְ זֶה לֹא מְטָרִיד אוֹתְךָ כְּרַגַע. אַתָּה עֵכְשִׁיו דּוֹאֵג לְאִיטְלָקִים. אֵיךְ הֵם רוֹצִים לְהַרְוִיחַ כֶּכֶה? מֵה קוֹרָה לְאִיטְלָקִים הָאֵלוּ? 4 שָׁנִים לְפָנַי הַשְּׁקֵת רַכֵּב מִתְחִילִים לְעֹבֹד עֲלֵיוֹ?

הַרְמִזוֹר הָאָדוּם מַחֲזִיר אוֹתְךָ לְמִצִּיאוֹת. אַתָּה חוֹשֵׁב שֶׁהַבְּנֵת מְדוּעַ אוֹמְרִים בְּכָל הַכְּנַסִּים הַמְשַׁעֲמָמִים הַלָּלוּ שֵׁשׁ לְהַקְדִּישׁ כ־ 25% מִזְמַן הַפְּרוִיֶּקְט עֲבוּר נִיתוּחַ הַמַּעֲרַכַת. הִרִי בְּסוֹפוֹ שֶׁל דָּבָר אַתָּה בְּעֶצְמְךָ הוֹצֵאת פִּי שְׁנַיִם כְּדִי לְרַכּוֹשׁ אֵת הָאַלְפָּא הַזֶּה. "נְכוּן" – אַתָּה אוֹמֵר לְעֶצְמְךָ, וְהִרִי הַרְכֵב הַזֶּה שׁוֹוֶה כָּל אֲגוּרָה, תְּרֵאָה, כָּל דָּבָר בְּמִקּוֹמוֹ, כָּל דָּבָר בְּשִׁלְמוֹתוֹ, מִמֶּשׁ הַתְּגַלְמוֹת הַבְּרִיאָה.

הַבְּנֵת מֵשִׁהוּ. אַתָּה מְרַגֵּשׁ כְּאֲבִים עֲזִים בִּלְבָב, זַעַם קָרָה וְצַמְרָמוֹרֵת. נִדְמָה כֵּאֵילוּ נִסְתַּם אֲבִי הַעוֹרֶקִים.

אוֹר אָדוּם וְצָהוּב מִשְׁתַּלְבֵּים בִּיחַד וְסִיבּוּב פְּרִסָּה חֵד, מְשַׁמֵּיעַ חִיכּוֹךְ צְמִיגִים מְצַמְרֵר. אַתָּה חוֹזֵר בְּשֵׁאֲטָה לְמִשְׂרָד. שֵׁם יֵשׁ לְךָ אֵת מְסַפֵּר הַטֵּלְפוֹן שֶׁל מִנְתַּח הַמַּעֲרַכַת שְׁפִיטְרַת לְפָנַי חוֹדְשִׁים.

מִי יוֹדַע, אוֹלֵי בְּנִסְיָה זֶה קְנִיתָ עוֹלָמְךָ.

# עבודת מנתח מערכות ומחזור חיי מערכת תוכנה

---

1. על ניתוח ומנתח
2. ניתוח מערכת – הגדרה
3. עבודת מנתח המערכת
4. מחזור חיי מערכת
5. ייזום
6. חקר מצב קיים
7. אפיון
8. עיצוב
9. מימוש
10. בדיקות
11. הדרכה והטמעה
12. התקנה והסבה
13. תחזוקה
14. מה עם אבטחת איכות?

## על ניתוח ומנתח

אמא רצתה שתהיה רופא? שתהיה מנתח? אין בעיה. אתה במקצוע הנכון. מהו מקצוע ניתוח המערכת? מהי עבודתו של מנתח המערכת? מהי מערכת בכלל? מהם השלבים השונים בניתוח מערכת מידע? זאת ועוד בפרק זה.

## ניתוח מערכת – הגדרה

המושג "ניתוח מערכות" (Systems Analysis) מורכב משתי מילים. כל אחת לכשעצמה מוכרת, אך מה משמעות השילוב? תחילה נגדיר:

|                        |   |
|------------------------|---|
| <b>מערכת</b><br>System | <b>מערכת</b> היא קבוצת יחידות (או מרכיבים) המקיימות קשרי גומלין או תלות הדדית קבועה ומתמשכת לאורך זמן, יוצרות שלמות ופועלות יחד לשם השגת מטרה משותפת. |
|------------------------|---|

ההגדרה כוללת בתוכה את הדברים הבאים: **מערכת** הינה אוסף יחידות המקיימות קשרי גומלין ותלויות זו בזו. קשרי הגומלין קבועים ומתקיימים לאורך זמן. יש יחידות רבות המתפקדות יוצא מן הכלל זו עם זו, אך השלמות לא תושג, אלא בצירופן. לאחר הצירוף אין משמעות ליחידות הבודדות כי אם למערכת שלמה. לאחר שהגדרנו את אוסף היחידות כמערכת, אנו יכולים לדון בהשגת המטרה המשותפת. מערכת עשויה באופן שניתן יהיה לשלב אותה עם מערכות אחרות, דבר הנובע בעקיפין מההגדרה, ודבר זה יוצר היררכיה אשר מתקיימת במערכות.

מערכת מורכבת מרכיב קלט, עיבוד ופלט. למעשה, כל יחידה במערכת מורכבת בעצמה מרכיבים כאלה. כך ניתן להמשיך הלאה, כי כל חלק מיחידה מורכב אף הוא מ...

עתה נעבור למילה השנייה: **ניתוח**. מהו ניתוח? כאן אין צורך להביא הגדרה מילונית, אלא נשתמש במשל: פציינט נכנס לרופא אשר בודק אותו ומגיע למסקנה שיש בעיה מסוימת אשר עדיין לא ניתן לעמוד על היקפה, אך מחייבת ניתוח דחוף. מגיע זמן הניתוח, הרופא פותח את הבטן ומנסה לאמת את חשותיו, הוא חוקר הנה והנה, מזיז מעט את הכבד, עולה לריאות, יורד לכליות ולבסוף, לאחר חקר מדוקדק, מחליט מה וכיצד לעשות את הטיפול. עתה מגיע השלב המעשי של תיקון הבעיה. המנתח חותך מעט, עוקר פה, תופר שם, מוסיף ומשפץ ולבסוף – תופר הכל, אורז וסוגר ומעביר את החולה לחדר ההתאוששות. עד כאן המשל.

ניתוח, אם כן, הינו שלב הכולל חקר מצב קיים ואפיון מצב חדש. אמור מעתה, אפיון מערכת עוסק במערכת חדשה. ניתוח מדבר על המצב הקיים ועל המערכת החדשה כאחד.

שימו לב שהמקצוע, או הנושא, הוא **ניתוח מערכות**. כי כך היא המציאות. בפועל, **מנתח המערכות** עוסק בפיתוח מערכת כולל חקר מצב קיים, אפיון ועיצוב המערכת החדשה.

## עבודת מנתח המערכת

המרכיב החשוב ביותר עבור מנתח מערכת ומקצוע ניתוח המערכות בכלל הוא **הלקוח**. הלקוח הוא זה שעבורו נוצר מקצוע ניתוח המערכות. מנתח המערכת חוקר את המצב הקיים של הארגון ושל הלקוח, מבין את הבעיות והקשיים במצב הקיים, עוזר ללקוח להגדיר את דרישותיו מהמערכת העתידית, ואז פונה למלאכת אפיון המערכת החדשה.

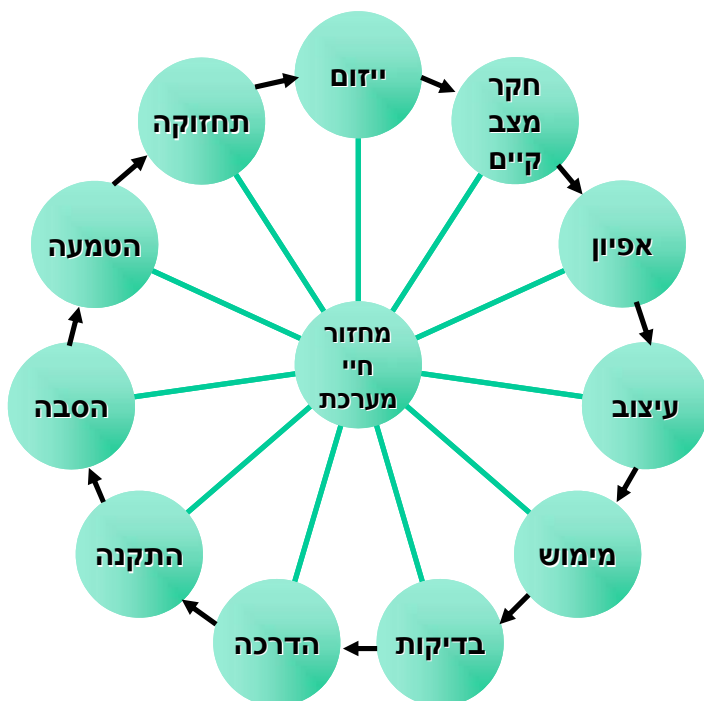
מנתח המערכת מגשר בין הלקוח (או הארגון המעוניין בפתרון מחשוב) לבין התוכניתנים הכותבים את הקוד (מימוש המערכת עצמה). מנתח המערכת מבין את צרכי הלקוח וכותב את תיק האפיון המתאר את היישום עצמו, את המערכת, את חלקיה ואת הקשרים בין מרכיביה השונים. המערכת מתוארת בעזרת שרטוטים רבים ומלל. לפני שניתן להעביר את המערכת לשלב המימוש על ידי התוכניתן, יש לעצב את הפתרון הסופי. בעבודת העיצוב מכינים את תבנית הדוחות, את מבני המסכים ומפרקים את המערכת כולה ליחידות עבודה קטנות: שגרות ופונקציות. מוסיפים על כך את הטכנולוגיה, הן החומרה והן התוכנה, ועתה הכל מוכן לביצוע על ידי התוכניתן. האחרון לוקח את תיאור היחידות הקטנות ומטפל בהן זו אחר זו. הוא לא זקוק לראייה מערכתית כי עתה אין צורך בכך, למרות שאין בכך פגם כמובן, וכותב את תוכנת היישום עצמה.

מנתח המערכת הוא הגוף המקצועי הגבוה ביותר במערכת, אם יורשה לי להתעלם מהמנהל ששייך לגוף הניהולי יותר מאשר המקצועי. אמנם, תמצא במקומות רבים כי תפקיד מנהל הפרויקט ותפקיד מנתח המערכת מבוצעים על ידי אותו אדם, במיוחד בארגונים קטנים או בפרויקט קטן. מנתח המערכת מלווה את המערכת משלב הרעיון והייזום, הגדרת המסגרת והגדרת המרכיבים, דרך שלבי הפיתוח וההטמעה אצל המשתמשים, עדכונים שוטפים עד לפיתוח מערכת תוכנה חדשה וצפייה בגסיסתה של המערכת הקודמת. מנתח המערכת לעיתים יהיה גם ראש הצוות, ובמקרים בהם הגיע מעולם התוכנה – גם מעצב התוכנה.

למנתח המערכת דרושים כישורים רבים, החל ממיומנויות של מגע אישי ויכולת לדבר, לשאול, לראיין ולהכין סקרים. הוא צריך ליזום, להכין ולהגיש מצגות, להדריך ברמות הניהול והרמות הטכניות, לדעת להטמיע את המערכת בסביבה עסקית או מבצעית פעילה. הוא צריך להיות בעל חשיבה יוצרת, כושר יזמות, ראייה מערכתית, ואחרון אך לא פחות חשוב – בעל חלומות שיודע להגשימם.

## מחזור חיי מערכת

מחזור חיי מערכת כולל את השלבים הבאים: ייזום, חקר מצב קיים, אפיון, עיצוב, מימוש, בדיקות, הדרכה, התקנה, הסבה, הטמעה ותחזוקה. נציג כל שלב ונלמד על הקלטים הזורמים אליו, על התוצרים שלו ועל הפלטים לשלב שאחריו. גם נלמד מי אחראי לביצוע כל שלב.



## ייזום

ייזום הוא השלב הראשון במערכת מידע, שבו הרעיון קורם עור וגידים ויוצא לדרך כפרויקט. מטרת השלב הם העלאת הרעיון לתהליך, לפיתוח המערכת ולהגדרתה באופן כללי של הבעיה. לעיתים שלב זה משמש להצגת הצורך להתחיל בתהליך. יש שתי נקודות פתיחה לשלב הייזום: קיימת מערכת ממוחשבת במצב הקיים שיש לבצע בה תהליך של שיפור או החלפה, ונקודת פתיחה שנייה: לא קיימת מערכת מחשוב במצב הקיים ויש להתחיל בתהליך ניתוח מערכת מן היסוד.

יש היבטים שונים לשלב הייזום:

- **ייזום לצורך שינוי ארגוני** – הייזום מתרכז בשינוי ארגוני ובהשלכותיו. ייתכן שיהיה צורך בתמיכה של מערכת מחשוב, או שהתהליך עצמו קורה עקב הכנסת מערכת מחשוב אשר עשויה להחליף עבודתם של אנשי ארגון מסוימים.
- **ייזום לפיתוח מערכת חדשה** – בין אם קיימת מערכת במצב הקיים ובין אם לא קיימת, הייזום לפיתוח מערכת חדשה מתחיל מלימוד הארגון (והמערכת, אם יש כזו) דרך הגדרת הצרכים למערכת חדשה וכלה במימוש המערכת.
- **ייזום לצורך שיפור מערכת קיימת** – קיימת מערכת בארגון ואין כוונה להחליפה. כל שנדרש הוא ניתוח פערים (Gap Analysis) בין המצוי לרצוי כדי לשפר את המערכת הקיימת. גישה זו היא זולה ומהירה. טובה? רק במקרים מסוימים.

- **ייזום לצורך התאמות מוצר מדף** – הארגון צריך מערכת מסוימת אך אין ביכולתו לפתח ולתחזק אותה בעצמו, או שהוא רוצה לחסוך בעלויות. הארגון גילה שיש חברה שפיתחה מוצר מדף שעונה על הצרכים. כל שיש לעשות הוא להתאים את מוצר המדף לארגון עצמו. מקובל לומר כי התאמה של עד 20% הינה כלכלית.

היזם הינו הגורם בארגון או מחוצה לו, אשר לוקח על עצמו להוביל את תהליך הייזום עד לקבלת ההחלטה אם עוברים לשלב הבא: האפיון. היזום יכול להיות כל אחד מאלה: הלקוחות של הארגון, הצרכנים הפוטנציאלים, אנשי תעשייה וניהול, מנהלי הארגון, מנהלים זוטרים, עובדים, גורמים מיחידת המחשוב בארגון.

הקלט לשלב הייזום הוא קושי או בעיה שיש רצון עז לפתור, או צורך מסוים לדבר מה אשר ניתן יהיה לשווקו. הקלט עשוי להגיע תוך כדי התבוננות מעמיקה במצב ובהתנהלות הארגון, או דרך תלונות לקוחות, גורם פנים או, ולעיתים גם זה נראה לעין וגלוי, גניבת רעיון מחברה מתחרה. ניתן לבצע בשלב ההתחלתי סקר שוק או משאל צרכנים או משתמשים, כדי להבין אם אכן יש עתיד לרעיון או שמא נסחפנו יתר על המידה.

התוצר של שלב הייזום הינו מסמך הייזום. מסמך זה נועד למנהלים בארגון שרעיון היזמות מופנה אליו, ולכן עליו להיות קצר ותמציתי. מסמך הייזום צריך לתאר את הבעיה הקיימת ואת הפתרון המוצע, באופן כללי כמובן. כמו כן יש לתת הערכה כספית ולחשוב על אופק הזמן מרגע האישור ועד לגירסה פועלת של המערכת. מקובל לומר, כי ההערכה הכספית בשלב זה נעה בטווח שגיאה של 300 אחוז. כלומר, קשה מאוד להעריך במדויק את עלות הפרויקט בשלב הייזום ואסור להסתמך על כך. לכן לעיתים רבות נמצא כי בשלב זה מעריכים משאבים נצרכים ולא עלות בפועל כמו גם תיחום מקסימלי של עלות המערכת.

סופו של שלב זה הינו בקבלת החלטה ניהולית אם ממשיכים לעסוק בפרויקט ועוברים לשלב הבא, או גונזים את הפרויקט ועולם כמנהגו נוהג. לעיתים מחליטים להקפיא את הפרויקט ל-18 חודשים כדי שניתן יהיה להוזיל את עלות החומרה לחצי, שכן, G.Moore כבר אמר וצדק: החומרה תכפיל עצמה כל שנה וחצי. פועל יוצא של כך, ניתן יהיה לרכוש חומרה בחצי מעלותה, לאחר שנה וחצי.

## חקר מצב קיים

לאחר שהארגון מאשר את תוצר שלב הייזום ומחליט לעבור לשלב הבא, יש לעצור לרגע. לפני שמאפיינים מערכת חדשה רצוי מאוד ללמוד את המצב הקיים. לימוד המצב הקיים יעזור למפתחי המערכת וגם להנהלת הארגון להבין ולהגדיר בפירוט את הבעיות הקיימות וללמוד את הארגון וצרכיו באופן יסודי. לימוד זה הינו חיוני להצלחת המערכת החדשה.

חברת המכרזים באינטרנט הראשונה בישראל היתה 4SALE. אם זכרוני אינו מטעה אותי, היתה חברה אחרת לפנייה – אבל לא הצליחה לעלות יפה. היכן היא כעת? הכיצד? הרי בקורס ניהול השיווק באוניברסיטה לומדים כי חברה שיש לה דריסת רגל ראשונה בשוק, בריחוק עצום מהחברות המתחרות, סופה שתיטול אחוז נכבד מציבור הצרכנים. לא זו אף זו, לשמור על לקוח קיים עולה חמישית מלהביא לקוח חדש לארגון, אולי אפילו פחות. כלומר, זהו עוד יתרון של החברה שהיתה הראשונה בשוק. אם כן, כיצד נכשלה החברה?

# אפיון מערכות מוכוון עצמים לפי UML

---

1. הקדמה
2. מתודולוגיה חדשה
3. מהו ניתוח מכוון עצמים
4. היכרות עם UML
5. מטרות UML
6. מרכיבי UML
7. תהליך הפיתוח
8. הגדרת דרישות
9. Use Case Diagram
10. Use Case Documentation
11. OOAD
12. גישת פתוח סגור (פ"ס) – Open Close
13. המודל הסטטי

**Class Diagram .14**

**.15 המודל הדינמי**

**Sequence Diagram – תרשים רצף .16**

**Collaboration Diagram – תרשים שיתוף .17**

**State Machine Diagram – תרשים מצבים .18**

**Activity Diagram – תרשים פעילויות .19**

**.20 שימוש בשרטוטים במהלך מחזור חיי המערכת**

**.21 סיכום**

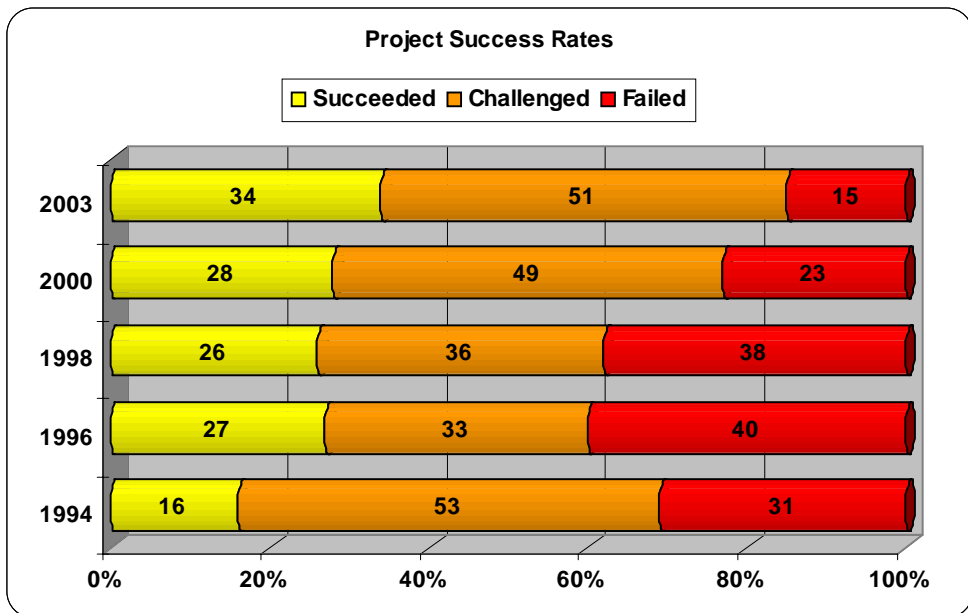
## הקדמה

עד עתה ראינו שיטות שונות לאפיון מערכות. שיטה אחת עסקה באפיון המידע תוך הסתכלות על התהליכים בארגון. שיטה אחרת עסקה באפיון המידע תוך הסתכלות על האירועים בארגון. בכל שיטה היינו צריכים לתאר את מודל הנתונים עצמו על ידי אפיון מודל הנתונים ולקשור אותו למערכת העתידית. שיטות אלו הוכחו כטובות, ועובדה היא שבמשך כשלושה עשורים זו היתה הדרך לאפיון מערכות.

למעשה, פעלנו על פי שיטות אלו מכיון שלא היתה מתודולוגיה אחרת עדיפה. האנושות ממציאה דבר טוב יותר רק כאשר היא מגלה חיסרון במצב הקיים. החיסרון לא נראה תחילה, מכיון שהמערכות שפותחו בשנות ה-70 היו קטנות יחסית, והטכנולוגיה לא היתה משוכללת כמו היום. אך כיום, כאשר עשויים לפתח מערכת במשך מספר שנים, על ידי מאות ואלפי אנשים ובטכנולוגיות חדשות שרק יצאו לשוק, הדברים סבוכים יותר.

אנו מוצאים את עצמנו עוסקים בבאגים אשר לא חשבנו כי יכולים לקרות. באגים אלה לעיתים עולים לנו בלקוח עצבני ובמקרים חמורים וקיצוניים – עלולים לעלות אף בחיי אדם (באג בתוכנה למטוס). האם יש דרך להימנע מבאגים אלה? כנראה שלא. אך יש דרכים להפחתת כמותם וחומרתם.

הבה נראה מה אומר העולם על פרויקטים מוצלחים. השנים אינן מטיבות עימנו, ואנו נוכחים בפרויקטים שנכשלים. כמה? זאת נראה על פי הגרף הבא:



ניתן להבחין בכך שפחות משליש הפרויקטים שמתחילים מסתיימים בהצלחה. מה זאת "הצלחה"? פרויקט שעמד בלוחות הזמנים, שצרך לא יותר מכמות המשאבים שהקצו לו ושהסתיים עם המאפיינים (התכולה) שהוגדרו מראש. כל שאר הפרויקטים נכשלו! מה זה Challenged? בתרגום חופשי זה "אותגרו". כלומר נכשלו, אך כדי לא להפסיד את הקרן,

מנהלי הפרויקט הסכימו להכפיל את כמות המשאבים, להאריך את לוח הפרויקט ובסופו של דבר, הסכימו לקבל רק מחצית מתכולת הפרויקט. אין ספק שפרויקט מעין זה הוא כישלון. הכותרת של התרשים מעניינת במיוחד: "Project Success Rates" – שיעור הצלחת פרויקטים. כמובן שהכותרת צריכה להיות: "שיעור כישלון פרויקטים". זו לא ממש הצלחה.

## בעיות בפיתוח מערכות

עיקר הבעיות בפיתוח פרויקטי תוכנה אשר הובילו בין השאר לכישלונות הפרויקטים הם:

- **שימוש חוזר בקוד** – כמעט ואין. תוכניות נוספות לכתוב כל קטע קוד מחדש, על אף שכבר נכתב. גם מנתחי מערכות אינם משתמשים בחלקים אשר עשויים להתאים להם ממערכות שכבר פותחו, אלא מפתחים מחדש.
- **מציאת באגים** – תהליכי הבדיקה מביאים בסופו של דבר לגילוי הבאגים. אמנם לא כולם נמצאים, אך אלה שמתגלים, מתגלים בשלב מאוחר יחסית. ככל שמתקדמים במחזור חיי המערכת, תיקון באג עולה יותר. בשלב האפיון, כל שיש לעשות זה לשנות את המסמך. בשלב הפיתוח, יש לערב את מנתח המערכת, את המעצב ואת התוכניתן ולשנות קוד. בשלב התחזוקה, מעורבים בכל תהליך התיקון כל צוותי העבודה של הפרויקט, לעיתים אף הלקוח, והבעיה חמורה בהרבה.
- **ריבוי שיטות סימון (נוטציות)** – כל אחד משתמש בשיטת סימון "יחודית" לו, בשרטוטים שלו ובחוקים שלו. לא פלא, אם כן, שאין שימוש חוזר בקוד.
- **המערכות גדלות ונעשות מורכבות יותר ויותר** – פיתוח מערכות בסדר גודל ענק, דורש מאמץ נוסף למאמץ הבנייה של המערכות החלקיות.

## מתודולוגיה חדשה

דוח מבקר המדינה לשנת 2001, קובע כי מכלל פרויקטי התוכנה במגזר הציבורי, רק 13 אחוז מסתיימים בהצלחה. רק 13%!! לא נדון מי משלם על שאר ה-87% שירדו לטמיון, אך אין ספק כי חייבים לעשות משהו בנידון.

מה יש לעשות?

האנושות החליטה כי מתודולוגיית הפיתוח הינה הצעד הראשון. יש לשנות את המתודולוגיה ולהתאימה לפרויקטים בני זמננו. כמובן, שמנהל פרויקט מוצלח יכול לתרום, כמו גם גורמי חוץ שלא בהם ענייננו בספר זה. נפנה אם כן לשימוש במתודולוגיה חדשה.

## מהו ניתוח מוכוון עצמים

לא עוד תהליכים, לא עוד אירועים. אמור מעתה **עצמים** או **אובייקטים**.

הגישה החדשה שמה את **הלקוח** (סוף סוף) במרכז. כאשר הלקוח מדבר איתנו בשפתו, ולא משנה אם אנו עוסקים בתהליכים או באירועים, מנתחי המערכת "אונסים" אותו ומגישים לו מערכת מחשוב אשר עברה הסבה מתהליכים ואירועים ל**נתונים**. מה הבעיה? כאשר הלקוח מגלה תהליך שאינו תקין אצלו בארגון ומשפר אותו, יש לתקן זאת גם במערכת המחשוב, כדי שהאחרונה תתמוך בשיפור החדש. במקרה זה מנתח המערכת צריך לבצע הסבה (או המרה) מהמערכת לעולם האמיתי של הלקוח, לשנות את הדרוש שינוי, להסב חזרה למערכת המידע, ורק אז (בתקווה גדולה) השינוי יופעל במערכת המציאותית. לא עוד!

גישת האובייקטים עוסקת ישירות בעולם האמיתי. לא עוד הסבות. על המערכת להיראות ולהתנהג כמו העולם האמיתי. כלומר, אנו שואלים את עצמנו מה קיים בעולמו של הלקוח, כיצד מתנהג עולם זה, והתשובות שתקבלנה תהיינה הבסיס למערכת המחשוב. בצורה זו, אם הלקוח מבצע שינוי ארגוני בעולמו, אותו שינוי בדיוק, ללא הסבה או המרה, יתבצע במערכת המחשוב. אם ללקוח יש בעיה מסוימת בתהליך שקורה בעולמו, אותה בעיה קיימת בתהליך המתקיים במערכת עצמה. כאשר הלקוח ימצא פתרון מעשי לבעיה זו, אותו פתרון כמות שהוא צריך להתאים גם למערכת המחשוב.

הזכרנו קודם "אובייקטים". מהו **אובייקט**? כל דבר שקיים בעולם הוא אובייקט. אני, אתה, הספר הזה בו אתה קורא, הציפורים בשמיים, ואפילו החדר בו אתה שוהה. עולמו של הלקוח בנוי מאובייקטים. חלקם פיסיים ואפשר לחוש בהם, חלקם לוגיים כמו חשבון הבנק שלי. בוודאי הוא בנמצא. יש לו יתרה (זכות אני מקווה...), אבל לא ניתן לראות אותו ולחוש בו. חשבון הבנק שלי הוא אובייקט, אך לא פיסי. כל מה שיש בעולם אלה אובייקטים. אובייקטים יודעים להתנהג בצורות מסוימות זה עם זה.

אם הלקוח ידבר באותה שפה עם מנתחי המערכת, המעצבים והמפתחים, או יותר נכון, אם כל אנשי הפיתוח ידברו עם הלקוח ועם עצמם בשפת הלקוח, בשפת העולם, אזי תהליך הפיתוח יהיה אינטואיטיבי יותר, קל יותר וברור יותר לכולם. ההמחשה הטובה ביותר לחוסר התקשורת בין הגורמים השונים הקשורים למערכת, מובאת בתרשים-סיפור הבא.

הלקוח תיאר את רצונו כך :



מנתח המערכת, הבין כך ולכן אפיין :



המעצב, הפליא לעצב :



# עיצוב בסיסי נתונים – המעבר לטבלאות

---

1. מבוא
2. תהליך עיצוב מודל הנתונים
3. נרמול נתונים בשלב העיצוב
4. צורת נרמול רביעית
5. צורת נרמול חמישית
6. המעבר לטבלאות
7. תרגיל מסכם
8. סיכום

## מבוא

הפרק הקודם עוסק באפיון מודל הנתונים, או במינוח אחר, עוסק בעיצוב המודל התפישתי. ראינו כי המעבר מרצונות הלקוח, הארגון, למודל שניתן לשרטטו על נייר לצורך הפשטת הבעיה, מונה מספר שלבים ובעל כללים אחדים. כעת משיש לנו את המודל התפישתי בתבנית של תרשימים (ERD) המשמשים לתיאור המידע שהארגון צריך, אנו צריכים להעביר את המודל, ולמעשה התרשימים, לטבלאות שהן הבסיס לבניית בסיס הנתונים.

## תהליך עיצוב מודל הנתונים

שאלה מרכזית הנשאלת בעת שמנתחים ומקימים מערכת מידע היא: כיצד לעצב את מבנה בסיס הנתונים, כך שישורת בצורה הטובה ביותר את תוכניות היישום השונות? ראוי לזכור שבסיס נתונים אינו משרת יישום אחד בלבד, אלא אמור לשמש את כלל היישומים הפועלים בארגון כעת ובעתיד.

המציאות שעבורה אנו מקימים את המודל יכולה להכיל מאות, ולעיתים אלפי אלמנטים של נתונים (Data Elements). אחת הבעיות המרכזיות והקשות היא כיצד לארגן אותם במבנה לוגי אחד. מבנה לוגי זה חייב להיות יציב ככל הניתן ועליו לייצג בצורה הטובה ביותר את המבנה הפנימי של הנתונים. המבנה צריך להיות גמיש מספיק, כדי לאפשר ביצוע שינויים בעת הצורך ללא זעזוע קשה במערכות המידע שכבר פועלות.

נחזור ונעיין בשלבים השונים המתייחסים לתהליך הניתוח, התכנון והעיצוב של בסיס הנתונים:

- **שלב העיצוב התפישתי (Conceptual Design)** עוסק בעיצוב מבנה המידע "הטהור", המשוחרר מכל האילוצים הנובעים משימוש בחומרה ובתוכנה. הקלט לשלב זה הוא אוסף כל הדרישות, כפי שבאו לידי ביטוי במהלך אפיון הדרישות. הפלט הוא תרשים ישויות קשרים. זהו למעשה שלב אפיון מסד הנתונים.
- **שלב העיצוב הלוגי (Logical Design)** עוסק בתרגום המבנה התפישתי שהתקבל בשלב הקודם לסכמה של מערכת DBMS מסוימת (ברוב המקרים, לטבלאות). בשלב זה נלקחים בחשבון האילוצים של מערכת זו.
- **שלב העיצוב הפיסי (Physical Design)** עוסק בתרגום הסכמה למבנה פיסי, הלוקח בחשבון אילוצי ביצוע ויעילות, נפחי אחסון, שיטות גישה לנתונים (אינדקסים, מצביעים), הקבצת נתונים על הדיסק (Clusters) ועוד.

עיצוב מודל הנתונים, נושא הפרק הנוכחי, מקביל לשלב העיצוב הלוגי. לאחר שהעלינו על הנייר את תרשימים המידע והקשרים בין חלקיו, אנו מסבים את התרשימים לטבלאות בדרך לשלב העיצוב הפיסי. הטבלאות ישרתו אותנו בהקמת בסיס הנתונים בצורה היעילה והאיכותית ביותר. מערכת ניהול בסיס הנתונים תיבחר על ידנו בדרך שתתאים לדרישות ניהול המידע ותתאים למערכות החומרה והתוכנה שבהן היא תפעל.

## המודל הלוגי (Logical Model)

המודל הלוגי נגזר מהמודל התפישתי. מודל זה מותאם למערכת ניהול בסיסי נתונים (DBMS) מסוימת ומכיל אילוצים שונים הנובעים מהצורה שבה מפתחי המערכת בחרו לייצג את המודל. מקובל לכנות מודל זה גם בשם **סכמה** (Schema), על פי ההגדרה של ANSI/SPARC. המודל הלוגי הוא למעשה הסכמה של המציאות הרלוונטית, כפי שהיא מיוצגת במערכת ניהול בסיסי נתונים מסוימת.

במהלך השנים הוגדרו מודלים לוגיים שונים, אולם רק שלושה מהם זכו לתפוצה נרחבת ויושמו במערכות DBMS מסחריות: **המודל ההיררכי**, **המודל הרשתי והמודל הטבלאי**. המודל הטבלאי הפך עם השנים לנפוץ ביותר. חשוב לציין שבמקביל למערכות DBMS שהתבססו על אחד משלושת המודלים הקלאסיים, התפתחו מערכות מסחריות המשתמשות בשילוב כלשהו בין המודלים האלה או רק בחלק ממודל מסוים.

המודל הלוגי עומד בבסיסה של כל תוכנה לניהול בסיסי נתונים. על כן באופן טבעי חלים עליו אילוצים מסוימים ומגבלות, אשר נובעים בעיקר מהצורך לטפל במבני הנתונים בצורה יעילה, נוחה ומובנת למחשב. המעבר מהמודל התפישתי למודל הלוגי דורש ביצוע המרות מסוימות. המעבר ממודל תפישתי למודל טבלאי קל באופן משמעותי לעומת המעבר לכל אחד מהמודלים האחרים.

קיימות שיטות סימון ורישום (Notation methods) שונות בשוק בסיסי הנתונים ולמעשה, כל חברה שמפתחת ומשווקת בסיסי נתונים משתמשת בשיטות סימון ורישום ייחודיות לה. בהמשך תוצג שיטה של חברת Oracle משתי סיבות: האחת, מכיון שבחלק האפיון פעלנו לפיה והשנייה, מכיון שזו שיטת הסימון והרישום השלטת בשוק בעת כתיבת שורות אלו.

## מודל הנתונים הטבלאי (Relational Data Model)

המודל הטבלאי מאפשר תיאור הנתונים בצורה טבעית ביותר על ידי אוסף של טבלאות. אין כל צורך להוסיף מבנים מיוחדים שמטרתם העיקרית היא סיוע לייצוג הנתונים במחשב. בכך הוא מאפשר אי-תלות נתונים מרבית. המודל הטבלאי מייצג את כל הנתונים באמצעות טבלאות, ובו כל טבלה מתאימה לקבוצת ישויות, כל שורה לישות וכל תכונה מיוצגת על ידי עמודה. ייצוג הקשרים בין הישויות מתאפשר באמצעות עמודות זהות בטבלאות שונות. הקשרים ב-ERD אינם ניתנים לציור כאשר אנו עוברים לטבלאות, שכן, המעבר לטבלאות מתבצע כדי שניתן יהיה לייצג את המודל "בשפת בסיס הנתונים". לכן, קשר בין טבלה אחת לאחרת אינה ניתנת לייצוג על ידי קשר, אלא יש להמיר את הקשר לעמודה בטבלה. בגדול, התהליך פשוט: תכונת המפתח בטבלה אחת משוכפלת לטבלה אחרת, שם נקראת העמודה "מפתח זר". כך, ניתן לחבר בין נתונים הנמצאים בטבלה אחת, יחד עם נתונים בטבלה שנייה. פירוט לנושא זה תמצא בהמשך הפרק.

**יחס (Relation)**, כלומר קשר בין שורה לעמודה בטבלה, מציין ערך בתכונה עבור ישות נבחרת מתוך קבוצת ישויות:

| מספר ת.ז. | שם פרטי | שם משפחה |
|-----------|---------|----------|
| 033663360 | כוחי    | דל       |
| 056782211 | יגאל    | גלון     |
| 056771340 | גיל     | גול      |

ערכי התכונות של הישות השנייה בטבלה פרושות על פני שורה שלמה. הקשר בין השורה השנייה לעמודה מסוימת (יחס) הוא ערך התכונה הנבחרת עבור הישות הזו.

המודל הטבלאי הופך כל קבוצת ישויות (Entities) לטבלה נפרדת. המימוש של הקשרים בין קבוצות הישויות במודל הטבלאי נעשה על ידי עמודות זהות המופיעות בטבלאות שונות, ותוך התבססות על העובדה שערכים זהים מופיעים בשתי העמודות. למשל, הקשר החד-רב-ערכי בין טבלת מחלקות לבין טבלת עובדים ממומש על ידי הוספת העמודה מספר מחלקה לטבלת עובדים. את הקשר הרב-רב-ערכי בין עובדים ופרויקטים מממש המודל הטבלאי על ידי הוספת טבלת קשר חדשה, המכילה את העמודות משתי הטבלאות (רק אם טרם נרמלנו את מודל הישויות-קשרים). צורת מימוש זו היא ישירה ופשוטה מאוד להבנה גם על ידי משתמשים שאינם מקצועני מחשב. המודל מבוסס על שפת SQL.

בטרם נראה כיצד מעבירים את מודל הישויות-קשרים לטבלאות, נרענן את זיכרונו בצורות הנרמול השונות. מקובלות 3.5 צורות נרמול:

- 1NF – הורדת תכונות היכולות להכיל קבוצות ערכים והורדת שדות מחושבים.
- 2NF – יצירת מצב שבו כל תכונה שאינה מפתח תלויה בכל המפתח.
- 3NF – יצירת מצב שבו כל תכונה שאינה מפתח תלויה אך ורק במפתח, כלומר שתכונה לא תהיה תלויה בתכונות נוספות פרט למפתח.
- BCNF – מניעת תלות בתוך המפתח (מפתח מורכב). צורת נרמול זאת נחשבת לצורת נרמול 3.5 מכיון שהיא נמצאת בין צורת נרמול שלישית לרביעית.

בשתי צורות נרמול מקובלות נוספות, 4NF ו-5NF, נעסוק בהמשך הפרק. צורות נרמול אחרות אינן ישימות בדרך כלל בעולם המעשה ועניינן בתחום האקדמי בלבד.

# נרמול נתונים בשלב העיצוב

## הקדמה

בשלב העיצוב אנו ממשיכים במלאכה שהתחלנו בשלב העיצוב התפישתי – האפיון. שלב העיצוב התפישתי מסתיים בהבאת מודל ישויות-קשרים לרמת נרמול בסיסית ואינטואיטיבית. למעשה, התהליך המסובך יותר מתחיל עכשיו. עתה יש לקבוע את המראה הסופי של בסיס הנתונים המסוים שיש להקים. שיקולי יעילות וביצועים צריכים להילקח בחשבון, כמו גם מצבים פרטניים הקשורים לאופי היישומים, ביצועים ועוד. רמות נרמול 4 ו-5 עוסקות במצבים מיוחדים אלה.

## צורת נרמול רביעית – 4NF

|                         |  |
|-------------------------|--|
| רמת נרמול רביעית<br>4NF | צורת נרמול רביעית קובעת כי אסור שקבוצת ישויות תכיל תלויות רב-ערכיות, כלומר, אסור שתקיים תלות בה תכונה A קובעת קבוצת ערכים B. |
|-------------------------|--|

רמת נרמול רביעית מזכירה את רמת הנרמול הראשונה. ברמה הראשונה קבענו כי אסור לאפשר לתכונה להכיל מספר ערכים בו-זמנית. ברמת נרמול רביעית אנו קובעים שאסור לקיים תלות רב-ערכית. תלות רב-ערכית מוגדרת על ידי תכונה אחת אשר קובעת קבוצות ערכים לתכונה אחרת. נלמד זאת באמצעות **דוגמה**:

בית הספר מחזיק מספר יועצים, שלכל אחד מהם יש כישורים ותחביבים. למשל:

| מספר יועץ | כישורים         | תחביבים       |
|-----------|-----------------|---------------|
| 1000      | Design Patterns | מקרמה         |
| 1000      | Design Patterns | סריגה         |
| 1000      | Object Oriented | מקרמה         |
| 1000      | Object Oriented | סריגה         |
| 1001      | C++             | פיסול בקרטון  |
| 1001      | C++             | טחינת תבלינים |

לפי ההגדרה:

