

# תיק (מסמך) אפיון

## גלופת לימוד

מסמך זה הוא גלופת לימוד של תיק אפיון, תורתית ומלאה (רמה 3 ולעתים קרובות גם רמה 4 ויותר של עץ המערכת) וכולל את כל רכיבי עץ המערכת בהיבטים של השלב הנוכחי – שלב האפיון. הנחיות מפורטות לשימוש בגלופה זו נמצאות במדריך המלא בקיט אפיון מערכת.

שימוש בגלופה זו מותנה ברישוי מפת"ח.

## תוכן העניינים

2	0. מנהלה
4	1. יעדים
13	2. יישום - מהות המערכת
46	3. טכנולוגיה ותשתית
64	4. מימוש
78	5. עלות – משאבים
85	נספחים

## 0. מנהלה

פרק זה משמש לניהול ובקרה של שלב האפיון. גורמים מעורבים, תכנית העבודה, מעקב ביצוע מול תכנון, ניהול תצורה ומעקב שינויים של התיק עצמו, אישורים, אבטחת איכות וכו'.

שים לב להבדל בין פרק זה ובין פרק 4 מימוש להלן. פרק זה מיועד לניהול שלב האפיון עצמו, פרק 4 מתאר את תכנית הפיתוח הכוללת של המערכת (מעבר לשלב הנוכחי) ואת אופן תפעול המערכת ותחזוקתה. עם סיום השלב הנוכחי, פרק 0 כבר אינו רלוונטי. פרק 4, לעומת זאת, הוא חלק מרכזי של תוצרי המערכת. (בשלב הבא יתעורר פרק 0 מחדש לתיאור אותו שלב).

בשלב בקשה להצעות הופך פרק זה להיות פרק המנהלה ויש לו מבנה מחייב מיוחד. ראה הקיט בקשה להצעות RFP בכרך יסודות \ מחזור חיים.

### רמת הפירוט של פרק זה ומבנהו המדויק נתונים לשיקול דעתו של הפרויקט \ הארגון.

ביחידות צה"ל יש להקפיד על אישורים לפי הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ ובפרט אישור אחראי במ"מ.

בהקשר עם האישורים, ובהקשר עם המנהלה של שלב האפיון כולו, חשוב לחזור ולהדגיש את **תמצית המנהלים**. לעיתים קרובות יאשר דרג מסוים את תמצית המנהלים ודרג אחר את תיק האפיון המלא. ראה פרק תמצית מנהלים בקיט תיעוד בכרך נושאים תומכים. תמצית המנהלים הוא מסמך נפרד, מחוץ לתיק זה, ומוצג לפניו. בכל דיון/שיקוף, בפרט בדיונים עם הצוות המנהלי, יש להקפיד על הגשת תמצית מנהלים (מסמך אפיון מקוצר לדרג הניהולי) בצמוד לתיק האפיון המלא. עם זאת, יש להקפיד על כך שהאישור הסופי, ע"י הצוות המנהלי לפרויקט, יהיה על תיק האפיון המלא ולא רק על תמצית המנהלים.

## 0.0 כללי

תיאור מקוצר של השלב הנוכחי בו נמצאת המערכת (ושבמהלכו מופק תיק זה).

### 0.1 גורמים מעורבים

- הצוות המקצועי (פנימי או חיצוני) המבצע (מיועד לבצע) את אפיון המערכת.
- צוות ההיגוי המנהלי (לאפיון)

### 0.2 תכנית עבודה

ראה הצעה בסעיף פעילויות (תכנית עבודה) במדריך המלא.

- במקרים מורכבים: תרשים גאנט, Pert\CPM
- ברוב המקרים מספיק: טבלת רשימת פעילויות ראשיות, מועדי סיום, מבצע ראשי וכו'.
- מעקב ביצוע מול תכנון (על התרשים או בתוך הטבלה)

### 0.3 כלים ונהלי עבודה

- כלים ונהלי עבודה של השלב הנוכחי

#### 0.4 ניהול תצורה ומעקב שינויים

להלן טבלת מעקב שינויים (ניהול תצורה) של השלב (התיעוד) הנוכחי:

מאשר	תיאור השינוי	מס' רכיב	מהדורה \ בסיס	תאריך

#### 0.5 אישורים

בדרך כלל, החתימות שבעמוד השער ובטבלה שבסעיף הקודם מספיקות. אם בכל זאת יש צורך באישורים ספציפיים, יש להשתמש בטבלה הבאה:

תאריך	שם	מייצג (מחלקה)	הערות	חתימה

## 1. יעדים

### 1.0 כללי – הבהקים

אפשר בשלב ביניים, לכלול בסעיף זה נושאים ודגשים כלליים של יעדי המערכת (עקרונות יסוד, אילוצים, מטרות כלליות וכו'), אשר במהלך האפיון חשוב לזכור אותם ושמקומם הסופי לא ברור. במהלך בניית התיק יועברו נושאים אלה למקומם הטבעי באחד מהרכיבים בפרק זה (1.X) או בפרקים הבאים. בשום פנים ואופן אין להפוך רכיב זה לאפיון המערכת כולה! לכל מידע המופיע כאן יש מקום טבעי וברור בפרקים וברכיבים דלהלן.

בצורתו הסופית, יכיל סעיף זה אך ורק עקרונות יסוד, תפיסה כללית וכו', נושאים אשר חשוב להדגישם, במבנה של הבהקים (Highlights) של יעדי המערכת, לפי הדוגמא להלן. זכור שסעיף זה יופיע בד"כ בתמצית מנהלים שתופק מתיק זה. ראה קיט תיעוד בכרך נושאים תומכים.

#### דוגמא להבהקים:

1. אפיון המערכת מגדיר שתי מהדורות. מהדורה 1 למימוש מיידי ומהדורה 2 למימוש בהמשך. בכל סעיף להלן, יש התייחסות ברורה לשתי המהדורות.
2. במערכת שתי תת-מערכות מרכזיות, תת-מערכת א' ותת-מערכת ב'. ראה חלוקה פונקציונאלית של המערכת בסעיף 2.3 להלן. למרות חלוקה ברורה זו, נחשבת המערכת לאחת (בשלב זה).
3. למערכת ישנם ממשקים חשובים ביותר עם מערכות חיצוניות, הן ממשקי קלט והן פלט. ראה פירוט ממשקים אלה ברכיבים 2.2 ו- 2.22 להלן.
4. המערכת מושתתת על הגדרה ברורה של ישויות המידע הבאות:
  - חוזה
  - מספר קטלוגי
  - חשבון ספק סופי.ראה הגדרה מדויקת של ישויות מידע אלה ואחרות ברכיבים 2.13 ו- 2.14 להלן.
5. קיימות שתי חלופות שונות לפתרון הטכנולוגי בעזרתו תמומש המערכת:
  - מסופים המחוברים למחשב מרכזי
  - רשת מחשבים אישיים.חלופות אלה מתוארות באופן כללי בסעיף 3.0 להלן ובשאר רכיבי פרק 3 במפורט.
6. למערכת זו דרישות אמינות ואבטחת מידע גבוהות ומחמירות. ראה פירוט דרישות אלה ברכיבים 2.19 ו- 2.23 להלן.

#### סוף דוגמא

### 1.1 לקוח\מומחה יישום

סביר שהמערכת מיועדת בעיקר ליחידה מסוימת בארגון. יחידה זו היא בעלת המערכת בכל המשמעויות העסקיות והניהוליות הרלוונטיות, כולל אחריות על המידע, מועדי הפעלת המערכת וכו'. תאר יחידה זו: תפקידיה, סמכויותיה, תחומי פעילותה וכו'.

סביר גם שיחידה זו היא אשר תעמיד את מומחה היישום אשר ישתתף באופן פעיל באפיון ובבדיקות המערכת. ציין מי מומחה היישום אשר ילווה את האפיון ומי המומחה אשר ילווה את שלב הבדיקות והתחזוקה השוטפת.

שים לב שכלל המשתמשים במערכת מוגדרים ברכיב 2.2 והגורמים המעורבים בפרויקט, ברכיב 4.1 להלן.

מומלץ מאד שמספר מומחי היישום יהיה נמוך, 1-2 אנשים. מספר גבוה מזה, מעורר שאלה לגבי מי בעצם מוסמך להחליט לגבי תכני המערכת ולאשר את הפרויקט. עם זאת, לעתים אין מנוס מהקמת נציגות משתמשים רחבה, כתוצאה ממבנה ארגוני מסוים (ארגון מבוזר, למשל). במקרה זה, יש להקפיד על היררכיה ברורה בין מומחי היישום ונציגות המשתמשים, הן בליווי הפרויקט והן באישורים.

### 1.1.1 לקוח \ משתמש עיקרי

היחידה בארגון שהיא הנשוא העיקרי של המערכת (היחידה הנושאית) והבעלים שלה. ציין את:

- שם היחידה
- יחידת האם
- מבנה פנימי
- יעדים ומטרות
- סמכויות ותחומי אחריות.

מבנה כללי של הארגון (המשרד, החברה) מוגדר בסעיף 1.4 להלן.

### 1.1.2 מומחה(י) היישום

לכל מומחה-יישום המעורב בפרויקט, ציין:

- שם
- תפקיד ושיוך ארגוני
- כתובת, טלפון, פקס, e-Mail
- מידת המעורבות: באילו שלבים, באילו רכיבים. מומחים אחרים (אינם מעורבים רשמית, אך אפשר להסתייע בהם)
- היו מעורבים בעבר
- מומחי יישום במערכות שכנות.

### 1.1.3 צוותי משתמשים

בפרויקטים גדולים, בהם מעורבים לקוחות שונים ובעלי עניין, יש לעתים צורך להרחיב את מעגל מומחי היישום ולצרף להם צוותי משתמשים. יש להגדיר מפורשות כפיפויות בהתאם לסעיף 1.1.2 לעיל.

## 1.2 יעדים ומטרות

יש להבחין בין יעדים - שהם ברמה עקרונית, לבין מטרות - שהן ברמה מעשית הניתנות לכימות ומדידה. להגדרה מדויקת. בהגדרת המטרות יש להיזהר מכפילות וחזרה על רכיבים מעשיים יותר כגון 2.3 ו-2.6. אם טרנזקציות המערכת, למשל, מגדירות בברור גם את המטרות, יש לקצר בסעיף זה ולהפנות מכאן לסעיף 2.6! (ובסעיפים שם, יש להפנות חזרה לסעיף זה ע"מ לוודא שכל המטרות והיעדים ממומשים, ראה שם).

הגדרה ברורה של יעדי המערכת ומטרותיה עשויה לשמש בסיס לבניית המערכת במספר מהדורות! יש להדגיש מטרות מינימום למהדורה הראשונה. נסה להגדיר גם יעדים ומטרות שמראש לא ייכללו במהדורה זו. נושא המהדורות עשוי להיות תקף גם לגבי האפיון עצמו: מה יאופיין כעת במפורט ומה יאופיין באופן כללי בלבד.

### 1.2.1 יעדים כלליים

1. רשימה ותיאור קצר
2. סיווג היעדים לפי קריטריונים, כגון:
  - יעדים ראשיים ומשניים
  - יעדים לטווח הקצר/הבינוני/הארוך
  - יעדים שימושו במהדורה זו ויעדים למהדורות הבאות.

### 1.2.2 מטרות מעשיות

- רשימה ותיאור קצר
1. סיווג המטרות לפי מספר קריטריונים, כגון:
    - חשיבות ותועלתיות
    - המדד ושיטת המדידה
    - הערכת המאמץ ואפשרויות המימוש בטווח הקצר והבינוני
    - חיתוך בין תועלת וחשיבות למאמץ ולמידת האפשרות המימוש; עדיפות ברורה למטרות חשובות וקלות לביצוע.
  2. מטרות בעדיפות שניה. מטרות שאינן כלולות במהדורה הנוכחית (זו שמאופיינת כאן), אך הן מועמדות לביצוע במהדורה/גרסה המיידית הבאה (שתבוא מיד לאחר הנוכחית).

### 1.2.99 מטרות עתידיות

- לביצוע במהדורות עתידיות (מעבר לנוכחית + הבאה)
- לאפיון במהדורות עתידיות (מעבר לתיק אפיון זה)

## 1.3 בעיות

מערכת מידע טובה היא מערכת המקיימת את התנאים הבאים:

1. פונקציונאלית - עונה לדרישות המשתמש/הארגון

2. **יעילה** - איכותית ובנויה נכון מבחינה הנדסית
3. **כלכלית** - עומדת בגבולות המשאבים ולו"ז שהוקצו
4. **חוקית** - עומדת בדרישות החוק, מינהל תקין וחוקת הארגון.  
להסבר מלא ראה הקיט מודל מפת"ח, סעיף מדדי איכות.

גם ה**בעיות** שהמערכת מיועדת לפתור נופלות באחד או יותר מהתחומים הנ"ל. עם זאת, יש הבדל מהותי בין התחום הראשון לבין השאר, 2 ו-3 (4 הוא מקרה מיוחד). בעיות בתחום 2 ו-3 הן בעיות פנימיות טכניות של המערכת ואופייניות למהדורות שיפור למערכות קיימות. בעיות מתחום א' הן המטרה העיקרית, הן היעד המרכזי של מערכות מידע בכל ארגון! בעיות בתחומים 2 ו-3 הן חשובות בעיקר במידה שהן גורמות לבעיות בתחום הראשון.

בעיות בתחום ה**פונקציונאלי** (1) הן בעיות בתפקוד הארגון כגון:

- מתן שירות (במונחי זמינות, איכות וכו')
- כ"א מיומן
- סרבול תהליכים
- אמינות
- הפסד כספי
- חוסר תקשורתיות (אישית)
- חוסר בקרה וכו'.

הטענה הקלאסית בדבר חוסר מידע זמין אינה בעיה, אלא במידה וחוסר זה גורם לאחת מהבעיות הנ"ל, היינו, לתפקוד לקוי של הארגון.

בעיות **טכניות/הנדסיות** פנימיות (ויעילות כנ"ל) במערכת עצמה, כולל בעיות תפעוליות, יכולות להיות במגוון רחב כגון:

- נפילות והתאוששות לקויה
- ביצועים לקויים ודרישות משאבים גבוהות
- תפעול כבד ומסורבל
- כ"א רב בתחזוקה (בעיה או סימפטום של בעיה אחרת?)
- קושי בהכנסת שינויים ובלימוד המערכת
- חוסר במילון נתונים
- חוסר תקשורתיות (טכנית)
- ארכיטקטורה מיושנת (יש לנמק ולאושש במספרים!).

בעיות בתחום זה הן, כאמור, משניות. חשיבותן עולה ככל שהן משליכות על פונקציונאליות המערכת או אם הן גורמות להטיה משמעותית בעלות/תועלת המערכת.

בסעיף זה יש נטייה מוצדקת במידת-מה להתייחס למצב הקיים. אך אין להגזים ואין להפוך סעיף זה לתיאור המערכת הקיימת. ככלל, תיאור מצב קיים איננו מטרה לעצמה, אלא במידה שהוא תורם לאפיון המערכת (המהדורה) החדשה. במקרה זה, מקומו הוא ברמת כל רכיב ורכיב להלן. הערה זו

כונה בפרט במקרה שהאפיון הוא למהדורה  $1+N$ , וקיימת כבר מהדורה  $N$  עובדת. מהדורה  $N$  היא המצב הקיים ותיק התחזוקה שלה הוא תיעוד המצב הקיים!  
אחרי כל ההנחיות הנ"ל, זכור שלעתים קרובות בעיות ומטרות הן שני צדדים של אותה מטבע. מנע אפוא כפילות וחזרה על הרכיב הקודם.

### 1.3.0 תמצית הבעיות במצב הקיים

- סיווג לפי פונקציונאלי, יעילות (הנדסי/טכני), כלכליות, חוקיות
- סיווג לפי חריפות הבעיה, היקף הנזק
- מידת ישימות הפתרון: קל/בינוני/קשה. במונחים של: משאבים, לויז', השתלבות בארגון ועוד, לפי המקרה הנדון. אין הכוונה כאן להגדרת מימוש המערכת במלואה. לכך מיועד רכיב 4 להלן.
- בעיות החוזרות על עצמן.

### 1.3.1 בעיות שהמערכת פותרת/אמורה לפתור

סיווג לפי סעיף 1.3.0 לעיל.

### 1.3.2 בעיות שהמערכת יוצרת/עשויה ליצור

סיווג לפי סעיף 1.3.0 לעיל.

### 1.3.99 בעיות שיידחו

בעיות שהמערכת (מהדורה זו של המערכת) אינה אמורה לפתור:

- סיווג לפי סעיף 1.3.0 לעיל.
- הסיבה מדוע לא מטופלת במהדורה זו.

## 1.4 הקשר ארגוני \ עסקי

הקשר בין מערכת זו (ויעדיה) לבין יעדי הארגון ומטרותיו הוא חשוב ביותר, אך אין להפוך רכיב זה למסמך יעדי הארגון. יש לצרף תקציר של יעדי הארגון ולהפנות למסמך מפורט יותר (מקור). אם אפיון זה הוא אפיון מהדורה חדשה למערכת קיימת, בדוק היטב אם לא חלו שינויים בסעיף זה מזמן אפיון המהדורה הקודמת או מזמן שהמערכת החלה בפעולתה.

לגבי המבנה הארגוני, רצוי שתשרים זה ירשם כבר בקודיפיקציה מתאימה (משתמשים וכו'), ראה הפרק סימול ישויות - קודיפיקציה בקיט כתיבת תיעוד בכרך נושאים תומכים. ודא שבתרשים מופיעים בבירור:

- הלקוח והיחידה הנושאת (רכיב 1.1)
- כלל המשתמשים (רכיב 2.2)
- יחידת המחשוב.

שים לב שהקשר לתכנית אב למחשוב (תכנון אסטרטגי) הוא ברכיב זה (1.4.1). קשר זה הוא רשות, בעוד שהקשר עם תכנית העבודה השנתית המופיע ברכיב הבא (1.5) הוא חובה.



#### 1.4.1 יעדי הארגון, אסטרטגיה

תקציר ו/או הפניה למסמך המפרט את יעדי/חוקת הארגון וליחידה הארגונית עבורה נבנית המערכת (מחלקה, מדור וכו').

אם נערכה לאחרונה תכנית אב למחשוב, מה הקשר למערכת זו?

#### 1.4.2 תרשים ומבנה ארגוני

רשימה פשוטה (ספר טלפונים) ו/או תרשים מבנה ארגוני. סימון בצבע על התרשים איפה נוגעת המערכת.

#### 1.4.3 השלכות או"ש

נושאי או"ש עקרוניים הנוגעים למערכת זו. נושאי או"ש פרטניים הקשורים לאופן הכנסת המערכת לפעולה והטמעתה, מפורטים בסעיף 4.7 להלן.

### 1.5 תכנית עבודה שנתית

במשרדי הממשלה יש הבחנה ברורה בין תוכנית עבודה שנתית, לבין תוכנית אב למחשוב - תכנון אסטרטגי, שהם מונחים זהים במפת"ח

- **תוכנית עבודה שנתית** (תע"ש) למחשוב נגזרת מתקציב המשרד (תקציב המדינה) והיא חובה. תע"ש מתוארת במפורט בקיט מיוחד בשם זה בכרך ניהול כולל. כיון שתע"ש מתארת פרויקטים, כולל פרויקטים רב-שנתיים, יש צורך, במקרים כאלה, (בממשלה) לקבל, בנוסף לתקציב השוטף, גם הרשאה להתחייב. אך בעקרון תע"ש היא תכנית שנתית.

- **תוכנית אב** למחשוב מתבצעת אחת למספר שנים (בד"כ 5 שנים) והיא רשות. תכנית אב היא רב-שנתית ומתוארת בקיט מיוחד בשם זה בכרך ניהול כולל.

כאמור, תכנית עבודה שנתית היא חובה וחייבת להתבצע פעם אחת (לפחות) בשנה במסגרת הכנת התקציב השנתי, בעוד שתכנית אב למחשוב היא רשות. לפיכך רכיב זה הוא חובה ואילו הרכיב הקודם (1.4) הוא רשות. עם זאת, בארגונים שסיימו לאחרונה ביצוע תוכנית אב למחשוב חובה למעשה להצביע על קשר ברור בין מערכת מידע זו לבין תכנית האב. חשוב לבדוק אם בתכנית העבודה השנתית הוגדרו מערכות שיש להן קשר עם מערכת זו ואם קשר זה יוצר תלות הדדית שיש להתחשב בה.

#### 1.5.1 אישור (סימוכין) תקציבי / עסקי

**ליחידות מחשב**: קישור ברור לתכנית העבודה השנתית המאושרת ולתקציב של יחידת המחשוב. במערכות מידע ממשלתיות - **סעיף זה הוא חובה**. במידה שמערכת המידע איננה תואמת את תכנית העבודה השנתית המאושרת לארגון, יש להביא סימוכין לנחיצות המערכת מוועדת ההיגוי למחשוב.

**לחברות מחשבים ובתי תוכנה**: חוזה או הזמנת עבודה מהלקוח ואישור הנהלת החברה. בפרויקטים גדולים, יש להקפיד על תקצוב רב שנתי (הרשאות), בנוסף לחלק המתוקצב בשנה הנוכחית. (בתי תוכנה: התחייבות רב שנתית של הלקוח).

## 1.5.2 תלות במערכות אחרות

מערכות אחרות (מידע ותשתית) שמופיעות בתוכנית העבודה השנתית או כבר בפיתוח שהמערכת הנדונה תלויה בהן. אין לכפול תת רכיב זה עם רכיבים 2.2.2, 2.22 ו- 3.33. כאן הכוונה לתלות תקציבית, תלות הנובעת מתכנית הפיתוח הכוללת (תכנית העבודה) של הארגון.

## 1.6 ישימות ועלות/תועלת

נסה לכמת את התועלות והחסכוניות הצפויים. תועלות נמדדות ביחידות שירות שהמערכת מפיקה. חסכוניות נמדדים במשאבים של: כסף, כ"א, זמן, חומרים כלים וכו'. ציין את שיטת המדידה המומלצת למדידת תועלות/חסכוניות אלה. במידה שקשה לכמת את התועלות, הנח הנחות אשר יאפשרו כימות בשלב מאוחר יותר.

בדוק אילו השלכות יש למידע הנכתב בסעיף זה על הערכת עלות המערכת (פרק 5 להלן), בפרט בהקשר עם חסמים - עליון ותחתון - לעלות המערכת. היעזר בהרחבה לרכיב 1.6: ישימות ועלות/תועלת בגלופת הלימוד של עץ מערכת אוניברסלי בקיט עץ מערכת. (עיין גם בקיט ניתוח חלופות שבכרך נושאים תומכים).

כתוב סעיף זה בסוף האפיון. סעיף זה הוא דוגמא מובהקת לכך שסדר הופעת הרכיבים בתיק (בעץ המערכת) איננו בהכרח סדר כתיבתם, בוודאי לא סדר העבודה.

### 1.6.1 סיכונים - ישימות הפרויקט

- בחן את המערכת/הפרויקט מול פרויקטים דומים בארגון ומחוצה לו
  - סקר סיכונים שנתי הוא חובה בפרויקטים גדולים ומתמשכים.
- ראה קיט ניתוח סיכונים בכרך נושאים תומכים.

### 1.6.2 עלות/תועלת – ישימות עסקית

עלויות (תמצית מרכיב 5.0 או רכיב 5.5)

1. תועלות וחסכוניות הניתנים למדידה (כימות של רכיב 1.2):

- התועלת או החיסכון
- יחידת המדידה
- כיצד ניתן למדוד לפני ואחרי מימוש המערכת או הכנסת שינוי
- תצפית בפועל.

2. תועלות וחסכוניות בלתי ניתנים למדידה (מתחרים, שינויים צפויים).

3. עלות מול תועלת: טבלה או נוסחה

## 1.7 אופק הזמן

ברירת המחדל לטווח הזמן בו מיועדת המערכת לפעול, היא 5 שנים. בפרק זמן זה יש לקחת בחשבון שנים עד שלוש מהדורות או יחידות מסירה נוספות, בנוסף לתחזוקה שוטפת של תיקון שגיאות ושינויים כפויים.

יש להימנע מפירוט יתר ברכיב זה. לתכנית עבודה מפורטת ומועדים מדויקים מיועד רכיב 4.2 (ו- 4.3) להלן. ברכיב זה (1.7) יצוינו שלושה תאריכי יעד, כמפורט בהמשך, במטרה להתוות מסגרת זמן כללית לפרויקט.

### 1.7.1 תוצרים

- מהדורה \ יחידת מסירה ראשונה (הבאה) של המערכת מותקנת ועובדת
- מהדורה \ יחידת מסירה שנייה (עוקבת) של המערכת מותקנת ועובדת
- סיום הפרויקט, השלמת כל המערכת

### 1.7.2 מועד נטישה

מועד קריטי שמעבר לו אין טעם בהתקנת המערכת ועדיף לנטוש את הפרויקט.

### 1.7.3 משך חיי המערכת

התקופה התפעולית של המערכת עד שסביר שתוחלף.

## 1.98 יעדים פתוחים (חלופות)

סעיף זה מטרתו לרכז את כל הנקודות שנשארו פתוחות ברכיב היעדים.

נקודות פתוחות הן לעיתים קרובות חלופות (אלטרנטיבות) אפשריות ולפיכך גם החלופות השונות תתועדנה כאן. ראה קיט ניתוח חלופות בכרך נושאים תומכים. יש לבדוק היטב את שני המקרים הבאים:

- נקודות פתוחות (וחלופות) ברמת רכיב משני (1.X)
- חלופה מקיפה לרכיב היעדים בכללותו.

שני מקרים אלה, שהם השכיחים יחסית, לא יתועדו למעשה ברכיב זה. נקודות פתוחות וחלופות ברמת רכיב משני - מקומן ברכיב המשני (1.X.98). חלופה מקיפה לרכיב היעדים כולו - תתואר ע"י **תיעוד חוזר ומלא** של רכיב 1 כולו (1.0 - 1.7).

רכיב זה מיועד אם כך, למקרי ביניים, היינו, מקרים בהם החלופות והנקודות הפתוחות הן כלליות ואינן משנות את הגדרת היעדים כפי שתועדה ברכיבים 1.1 - 1.7 לעיל.

בכל מקרה, חיוני לסגור סופית את הנקודות הפתוחות ולהכריע בין החלופות השונות ברכיב היעדים, אחרת המשך אפיון המערכת כולו תלוי על בלימה. חשוב ביותר להשאיר בתיק תיעוד גם לנקודות שנסגרו ולחלופות שנדחו, על מנת לסייע בדיונים מחודשים בעתיד. הנקודות הפתוחות יזוהו באופן הבא:

- הנקודה-הפתוחה/חלופה : שם או זיהוי אחר
- תאריך פתיחה
- תאריך סגירה
- הגורם שסגר
- הפניה לסיכום דיון / מסמך המתאים

## 1.99 יעדים עתידיים

סעיף זה מטרתו לרכז את כל הדרישות העתידיות ברכיב היעדים שאינן כלולות במהדורה הנדונה, אך סביר שיכללו במהדורות עתידיות. רישום דרישות עתידיות אלה, הוא במטרה כפולה:

- לא לאבד מידע קיים
  - להדגיש מה לא כלול במהדורה הנדונה ולסייע בכך לניהול תצורה של המערכת.
- מיקומן של דרישות עתידיות אלה ברכיב ראשי (פרק 1) של עץ המערכת, הוא מכוון. אם יש מספיק מידע וצורך לרשום תחזיות אלה ברמה מפורטת יותר, היינו, ברכיבי יעדים משניים (1.X), יש לבדוק היטב אם אין לפנינו בסיס להגדרה של המהדורה הבאה של המערכת. הגדרה כזו פירושה **תיעוד מלא** של רכיב 1 כולו ומיקומו בתיק אפיון נפרד (של המהדורה הבאה).
- לרכיב זה יש השפעה רבה על רכיבים מקבילים (X.99) ולכן יש ללבן אותו ולסגור אותו בהקדם האפשרי. אופן תיעוד הדרישות:
- שם או זיהוי אחר.
  - תיאור הדרישה או הפניה לתיעוד אחר (תכנון אסטרטגי למשל)
  - קשר לרכיבים X.99 אחרים
  - הפניה לסיכום דיון.

## 2. יישום - מהות המערכת

המטרה העיקרית של רכיבי פרק זה היא לוודא שהיישום, מהות המערכת, מכוסה על כל היבטיו. סדר הרכיבים: 2.1, 2.2, 2.3 וכו', הוא במידת מה שרירותי ובודאי שאין הוא קובע את סדר הגדרת הרכיבים ובנייתם. כל פרויקט יתחיל ברכיב שנוח לו, יזהה את הרכיב המוביל, יעבור משם לרכיבים האחרים וינוע ביניהם הלוך וחזור בשיטה של הגדרה סיבובית והדרגתית. לרכיב 2.20 חיתוכים והצלבות יש חשיבות רבה, לא רק כרכיב מסכם ומאבטח איכות, אלא כרכיב אשר מלווה את אפיון המערכת בזמן אמת ושומר כל הזמן על איזון ויחסי גומלין בין הרכיבים השונים. ישנן מתודולוגיות פורמליות שהולכות בשיטה זו שנקראת Cross reference Analysis. שיטה זו יכולה להצליח בעיקר בשילוב עם כלי ממוחשב. קשה מאד לנהל אותה ידנית.

למרות החלוקה השרירותית מה, אפשר לזהות תת קבוצות ברורות בתוך רכיב היישום:

- **מסגרת ותיחום כללי של המערכת (חיצוני ופנימי) - רכיבים 2.1 - 2.3**
  - **ממשק המשתמש - רכיב 2.4** מגדיר את האופן בו המשתמש רואה את המערכת ועובד אתה.
  - **כללים עסקיים - רכיבים 2.5 - 2.10** מגדירים את הכללים לפיהם פועלת המערכת. חלק זה נקרא גם המודל הדינמי של המערכת או החלק האלגוריתמי. ביחד עם רכיב 2.4 לעיל, הקבוצה הכוללת, 2.4 - 2.10, מגדירה את אופן פעולת המערכת ואת צורת העבודה איתה.
  - **מידע שהמערכת מנהלת - רכיבים 2.10 - 2.16** מגדירים זאת וכן מידע תומך שהמערכת נעזרת בו. חלק זה נקרא גם המודל הסטטי של המערכת או בסיס המידע. רכיב 2.10 מופיע פעמיים גם בקבוצת המודל הדינמי וגם בקבוצת המידע – המודל הסטטי. יש לרכיב זה אכן נשמה כפולה. ראה הסבר נוסף ברכיב עצמו להלן.
  - **פרמטרים אורתוגונליים - רכיבים 2.19 - 2.23** מגדירים פרמטרים החותכים את הקבוצות הקודמות ומתארים תכונות כלליות של היישום, כגון: ממשקים, אבטחת מידע ועומסים.
- חלוקה זו, בעיקר שלוש קבוצות הביניים: ממשק משתמש, כללים עסקיים ומידע, תואמת את רוב המתודולוגיות והשיטות לניתוח מערכת המקובלות ומאפשרת קישור לוגי ופיסי של פרק היישום עם כלים ומתודולוגיות מתקדמים להנדסת תוכנה (Software Engineering, CASE).

### 2.0 ארכיטקטורה כללית – הבהקים

אפשר, בשלב ביניים, לכלול ברכיב זה (בדומה לרכיב 1.0 לעיל), נושאים ודגשים כלליים של היישום, שחשוב לזכור אותם אך מקומם הסופי לא ברור.

במהלך בניית התיק יועברו נושאים אלה למקומם הטבעי - אחד הסעיפים בפרק זה (2.X) או בפרקים הבאים. בשום פנים ואופן אין להפוך רכיב זה לתיאור המערכת כולה! לכל מידע המופיע כאן יש מקום טבעי וברור בפרקים ובסעיפים שלהלן.

בצורתו הסופית, יכיל רכיב זה אך ורק עקרונות יסוד, תפיסה כללית וכו', נושאי יישום אשר חשוב להדגישם, במבנה של הבהקים (Highlights) ולפי הדוגמא להלן. זכור שסעיף זה יופיע בד"כ בתמצית מנהלים שתופק מתיק זה. ראה קיט תיעוד בכרך נושאים תומכים.

**דוגמא: (בלבד!)**

5. המערכת המאופיינת היא המשך של מערכת קיימת. המצב הקיים לא תועד בנפרד, אך אפיון זה מתחשב בכל הידע שהצטבר במערכת הקיימת, פרט בממשקים עם מערכות מידע חיצוניות. ראה פירוט בסעיפים 2.1 ו-2.22 להלן.

4. חלוקת המערכת ל- 2 תת-מערכות מרכזיות, תת-מערכת א' ותת-מערכת ב', מודגשת מאד לאורך תיק האפיון. מומלץ ביותר להשתמש בחלוקה זו לבניית המערכת בשתי מהדורות. ראה חלוקה פונקציונלית של המערכת ברכיב 2.3 שלהלן.
5. לקובצי המערכת יש שתי תכונות חשובות:
  - מידת הפירוט, ראה רכיב 2.11 להלן.
  - נפח רב, ראה רכיב 2.21 להלן.בעיצוב המערכת ובנייתה יש לקחת בחשבון שתי תכונות אלה.
6. לממשקי אדם-מחשב חשיבות רבה במערכת. המלצה ברורה לשלב העיצוב היא בניית המסכים לפי תקן X.11 ראה פירוט ברכיב 2.4, להלן.
7. המערכת מושתתת על הגדרה ברורה של ישויות המידע הבאות:
  - חוזה
  - מספר קטלוגי
  - חשבון ספק סופי.ראה הגדרה מלאה של ישויות אלה ואחרות בסעיפים 2.13, 2.14 להלן.

## סוף דוגמה

### 2.1 מאפיינים כלליים

#### 2.1.1 מצב קיים

- בתיעוד רכיב זה יש מספר אפשרויות, מהקל אל הכבד:
- הפנייה לרכיב 1.3 בעיות. התיעוד שם, ברמת 1.3.0 או סעיף 1.3 כולו, מספק. אין צורך להרחיב כאן.
  - סקירה קצרה של המצב הקיים כאן. התיאור ברכיב 1.3 לעיל איננו מספק. אפשרות זו היא המקרה הקלאסי של שימוש ברכיב זה.
  - סקירה יסודית ונרחבת של המצב הקיים כאן. יש לצרף לתיק נספח 2.1.1: מצב קיים. ראה גלופת לימודי ועבודה בלשונית תוצרים בקיט זה.
  - תיק מצב קיים: במקרים קיצוניים יותר בהם נדרש תיק מצב קיים, יש לפעול כמוסבר במדריך המלא ולהוציא תיעוד עץ מערכת מלא של המצב הקיים.
- למידע נוסף, בעיקר בהקשר עם שני המקרים האחרונים, ראה הרחבה לרכיב 2.1.1 מצב קיים ב בקיט עץ מערכת אוניברסלי בכרך מחזור חיים.

#### 2.1.2 אופי המערכת וסוגה

- ציין את **אופי המערכת** המתוכננת, באופן כללי, האם היא:
- חדשה לגמרי, ללא כל זיקה למערכת קיימת (ידנית או ממוכנת)
  - חדשה, אך בזיקה חלשה למערכת קיימת (ידנית או ממוכנת)
  - הרחבה של מערכת קיימת (מעבר למהדורה חדשה)

- חדשה אך דומה למערכת אחרת: במשרד, במשרדים אחרים וכו'.
- מה הוא **סוג המערכת** המתוכננת, באופן כללי, האם היא:

- מערכת רישומית
- מערכת תפעולית
- מערכת בקרה
- מערכת תכנון ומעקב
- מרכז מידע (information center)
- שירות ישיר למשתמשי קצה
- מערכת משובצת מחשב
- מערכת זמן אמת
- אחר.

יחידות צה"ל יסווג את המערכת עפ"י סיווגים נוספים המוגדרים בהוראות צה"ל/מקשר"ר/מצו"ב/במ"מ.

מה **סוג הפעילויות** הצפויות במערכת. האם רוב הפעילויות תהיינה מסוג:

- Batch (אצווה)
- OLTP (טרנזקציות מקוונות – Online Transaction Processing)
- Data entry (הזנת נתונים)
- פעולות Housekeeping (פתיחת/סגירת שנה)
- שימוש בכלים למשתמש קצה (DSS, גיליון אלקטרוני וכו')
- ממשקים בזמן אמת (סיגנלים) עם מכשור אחר
- אחר.

### 2.1.3 אילוצים

אילוצים ומגבלות יישומיים כלליים המשפיעים על אופיה הכללי וסיווגה של המערכת, כגון:

- אילוצי חוק וחוקת הארגון,
- תקינה ומדיניות.
- אתיקה ונורמות מקצועיות
- דרישות סקטוראליות
- מערכות קיימות בארגון וחוצה לו,
- סביבה טכנית קיימת או מתוכננת,
- אילוצים כלכליים,
- ביחידות צה"ל: אילוצי במ"מ. לפי הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ.

אילו צים אחרים בתחומי הטכנולוגיה, היישום, כוח אדם, עלויות, מימוש (לו"ז) וכו' יצוינו להלן בסעיף המתאים!

#### 2.1.4 מילון מונחים

במילון מונחים יש לקטלג מונחים ומושגים כלליים שהם שפת הארגון (שפת המערכת) ושהדרך הנוחה להסבירם ולהגדירם הוא במלל חופשי.

סביר שבשלבים המוקדמים של האפיון יכיל רכיב זה הגדרות שבשלבם מאוחרים יותר (של האפיון!) אכן יועברו לרכיבים קונקרטיים יותר בקבוצה הני"ל 2.9 - 2.13.

נקודות לשיקופים ולאבטחת איכות:

- כמה מונחים מכיל רכיב זה בסוף האפיון?
- העברת מונחים מרכיב זה לרכיבים הקונקרטיים 2.9 - 2.13 בשלב העיצוב - התרעה!  
בתיק האפיון הסופי, יכול רכיב זה להיות במספר תצורות:
- תצורה מינימלית: רק מונחים שלא מצאו את ביטויים ברכיב קונקרטי.
- תצורה מקסימלית: כל המונחים והמושגים של שפת המערכת. אלה שיש להם ביטוי קונקרטי תצוין לידם הפניה מתאימה.
- תצורת Hypertext: אם יוחלט להפוך את תיק האפיון למסמך Hypertext.  
זכור שרכיב זה עשוי גם להילקח, ברמה זו או אחרת, לתמצית המנהלים:  
הגדרת מונח תכיל:
  - טקסט חופשי.
  - קישור לרכיב קונקרטי

#### 2.2 תיחום היצוני

רכיב זה חשוב ביותר לתיחום גבולות המערכת: עם מי יש למערכת קשר, היכן מתחילה ונגמרת המערכת, מי ישתמש בה באופן שוטף.

רשימת המשתמשים תהיה מלאה גם אם זו חזרה על המשתמש העיקרי ברכיב 1.1.

לכל משתמש/מערכת מידע יש להגדיר: האם הוא ספק מידע (Source), צרכן מידע (Destination) או שניהם? מידת חיוניות המערכת לעבודתו השוטפת, נימוק! תיאור קצר של המידע, כולל תדירות הגישה ובאילו תקופות?

שים לב לקשר של רכיב זה עם רכיב 2.19 - אבטחת מידע. לכל משתמש הרשום כאן, צריך שתהיה גם הגדרה של הרשאה ומידור מתאימים שם.

הקבצה של משתמשים ומערכות מידע תיתכן במספר רמות, בהתאם להבנת המאפיין, המצב בשטח ומידת המעורבות של מומחה היישום. למשל, אפשר לציין מחלקת הנהלת חשבונות, או לרדת לרמה מפורטת יותר (מדור, רמה אישית). או אפשר לציין מערכת המידע של מנהל הכנסות המדינה, או מערך מע"מ, או לרדת לרמה מפורטת יותר.

זיהוי המשתמשים יכול להיעשות ע"י סימול (מספרי) או ע"י זיהוי שמי (או צרוף של שניהם). סימול הוא בהחלט רצוי. אם קיים סימול ברור של משתמשים בארגון, יש לפעול לפיו. אם אין, יש ליצור



סימול כזה. היעזר בפרק סימול ישויות - קודיפיקציה בקיט תיעוד שבכרך נושאים תומכים. רצוי לבחור בזיהוי לפי מספר רץ ולא ליפול במהמורה של היררכיות ומבנה ארגוני. אפשר גם להסתפק בזיהוי שמי, אך הוא חייב להיות ברור. בכל מקרה, הזיהוי יהיה פונקציונאלי לפי תפקיד ומיקום בארגון ולא אישי. בכל שיטה שתבחר, חשוב שיהיה אפשר להצליב משתמשים עם ישויות חשובות אחרות (ראה רכיב/סעיף 2.20 להלן).

## 2.2.0 תיחום כללי

דוגמא לכלי לשימוש ברכיב זה : DFD Context Diagram.

### 2.2.1 משתמשים

- משתמשי פנים, לפי מבנה ארגוני
- משתמשים חיצוניים לארגון

### 2.2.2 מערכות משיקות

- מערכות (מידע ותשתית) פנימיות בארגון
- מערכות (מידע ותשתית) חיצוניות לארגון

אזכור המערכות המשיקות כאן הוא למטרת השלמת התיחום החיצוני של המערכת. פירוט הממשקים של המערכת עם מערכות אחרות, פנימיות וחיצוניות, מקומו ברכיב 2.22 ממשיקים וקישורים להלן.

## 2.3 תיחום פנימי

שים לב שסעיפים 2.3 - 2.7 הבאים הם חטיבה לוגית אחת, המתייחסת לחלק האלגוריתמי של המערכת, כאשר :

- רכיב 2.3 מיועד לפירוק פונקציונלי ולתיחום כללי (בהמשך לרכיב 2.2)
  - רכיבים 2.4 (מסכים) ו-2.6 (טרנזקציות) הם המשך לוגי של רכיב 2.3 (והם מקבילים לרכיב 2.11 המייצג את צד הנתונים)
  - רכיב 2.5 (תהליכים) הוא שרשרת של טרנזקציות
  - רכיב 2.7 (מודולים) הוא המימוש הפיסי (בעיקר של רכיב 2.6). מכאן שכל הרכיבים הנ"ל משולבים זה בזה ובהגדרת כל רכיב חשוב להציג את קשריו עם הרכיבים האחרים.
- לעתים, הקשר בין הרכיבים (למשל הקשר של רכיב 2.2 - משתמשים, עם רכיב 2.3) הוא הדוק במידה כזו שמוצדק לאחד את הגדרתם. איחוד כזה מותנה במהות המערכת הספציפית, אך גם בכלי ובטכניקה שבשימוש.

קיימות שיטות וטכניקות שונות כיצד להגדיר חלוקה פונקציונאלית (פירוק פונקציונאלי) של מערכת מידע. מפת"ח, כנוהל-מסגרת פתוח, איננו מכתוב שיטה וטכניקה זו או אחרת. בהגדרת רכיב זה ותיעודו אפשר, אפוא, להשתמש בכל טכניקה מוכרת ומקובלת לפירוק פונקציונלי, בפרט אם היא נתמכת בכלי ממוכן. אין עדיפות לטכניקת Top/down או Bottom/up, להצגה בתרשים או במלל. הצגת המערכת Top/down, למשל, אין פירושה שהמערכת הוגדרה - או חייבת להבנות - Top/down.

שים לב שטכניקות מסוימות מגדירות יחד עם הפירוק הפונקציונאלי גם רכיבים נוספים, דוגמת: משתמשים, קבצים, ממשקי העברת מידע וכו'. אין מניעה מלהשתמש בטכניקות אלה ובלבד שהרכיבים הנוספים יופיעו גם ברכיב המתאים לפי עץ המערכת. הפתרון הטכני לכפילות זו הוא כמובן השימוש ב-repository (ראה רכיב 3.12) ובמחולל תיעוד מתאים, המאפשרים להציג את תכני ה-repository לפי סרגל תיעוד המוכתב לכלי מבחן.

הדגש ברכיב זה הוא על פירוק פונקציונאלי כללי! אין להיסחף, ברכיב זה, לפירוק מפורט המגיע עד רמת טרנוקציה ו/או מסך. לשם כך מיועדים רכיבים 2.4, 2.5 ו-2.6 הסמוכים. באשר למידת הפירוט של רכיבים אחרונים אלה, בשלב האפיון, ראה הנחיות שם.

שים לב שבהגדרת רכיב זה אפשר כבר להניח את היסודות לניהול תצורה של המערכת, היינו הגדרת מהדורות לבנייתה ואפשר גם לאפיונה. כל תת מערכת תהיה יחידה אוטונומית לניהול תצורה (CSCI, במונחי A2167-DOD).

הקפד על ציון הקשרים והתלויות בין תת-המערכות בחלוקות שהוגדרו. ציין אלו מהחלוקות הנ"ל (לתת-מערכות) ניתנות לביצוע כפרויקטים נפרדים, במקביל או בשרשרת ובאילו עדיפויות וסדרי קדימויות.

### 2.3.0 תיאור כללי של המערכת

1. הסבר כולל בתרשים כגון DFD 0 ומקרא קצר
2. חלוקה לתת מערכות ראשיות:
  - דרוג לפי: מורכבות, קדימויות ועדיפויות,
  - חלוקה למהדורות \ יחידות מסירה \ לתת-פרויקטים

#### 2.3.1 תת-מערכת \ יחידת מסירה 1

- הסבר כולל בתרשים כגון DFD 1 ומקרא קצר
  - פונקציות ותהליכים ראשיים
- שים לב, פירוק ופירוט נוסף של תת המערכת יהיה באמצעות סעיפים 2.5 ו-2.6 להלן.

מקובל במפת"ח להקדיש את תת המערכת הראשונה – 2.3.1 (לעתים גם השנייה, אם יש צורך) לתת המערכת המנהלת, המספקת שירותי שליטה ובקרה וכן מנגנונים רוחביים משותפים (טיפול בשגיאות וכדומה) לשאר תת המערכות.

#### 2.3.2 תת-מערכת \ יחידת מסירה 2

- הסבר כולל בתרשים כגון DFD 1 ומקרא קצר
- פונקציות ותהליכים ראשיים

#### 2.3.N תת מערכת \ יחידת מסירה N

### 2.4 ממשק משתמש

יש לבדוק היטב אם קיימים תקנים ארגוניים ותבניות מוכנות למסכי תפריט ופעולה בארגון. ראה הרחבה לסעיף 2.4.0: ממשק אדם מחשב בקיט עץ מערכת אוניברסלי בכרך מחזור חיים.

רצוי מאד להיעזר בכלי ממוכן. אם אפשר, כבר בכלי הפיתוח הסופי (מחולל מסכים) שבעזרתו תיבנה המערכת.

קיים הבדל המהותי שבין שלשת תת-הסעיפים להלן :

- רכיב 2.4.0 מגדיר כללי הנדסת אנוש ותקן לעיצוב המסכים במערכת,
- רכיב 2.4.1 מגדיר את מסכי התפריט ואפשר לראותו כהמשך לרכיב 2.3 לעיל
- רכיב 2.4.2 מגדיר את מסכי הפעולה ואפשר לראותו כחלק מרכיב 2.6 להלן.

הדגש בשלב האפיון יהיה על סעיפים 2.4.0 ו- 2.4.1 ופחות על 2.4.2. הערה זו נכונה בפרט לפרויקטים הבוחרים לבנות אבטיפוס.

רכיב 2.4.1 מסכי תפריט, אשר נקרא גם עץ המסכים, מתאר את עץ המסכים הכללי והסטטי. עצי מסכים דינמיים וחלקיים, המתארים את המסלולים השונים לפיהם אפשר לטייל בעץ או לעבור ממסך פעולה אחד לשני על מנת לבצע סידרה של פעולות, מתואר ברכיב 2.5 תהליכים הסמוך.

בהקשר עם סימול המסכים ראה פרק סימול ישויות - קודיפיקציה בקיט תיעוד שבכרך נושאים תומכים. בשלב ראשון רצוי לתת למסכים סימול עצמאי משלהם :

- M001, M002 וכו' ולקשר בשלב העיצוב עם החלוקה לתת-מערכות
- עם החלוקה לטרנזקציות (ובעתיד לאובייקטים), אפשר להשתמש בקידומת: 2.4.1, 2.4.2, לסימול המסכים.

במערכות חלונאיות יש העברת דגש לכיוון מסכי הפעולה ושילוב הניווט והפעולות האפשריות (היינו פונקציות תפריט) בתוכם. בכלל, ההבחנה בין שני סוגים אלה, קצת מתערפלת. אך עדיין יש חלוקה כללית לתפריטים ולמסכי פעולה, בעיקר ברמה הכללית של חלוקת המערכת והניווט בה בגדול וזה מה שחשוב באפיון.

## 2.4.0 כללי הנדסת אנוש

כללים ותקנים בממשקי אדם-מחשב: HCI (Human-Computer Interface). מבנה אחיד ותקני ברמת הארגון (שלדים מוכנים).

מבנה תקני והשתלבות עם רכיבי המערכת האחרים לפי תקן :

- בינ"ל/ישראלי כגון: x11 .
- תקן דה-פקטו של יצרן מוביל

## 2.4.1 מסכי תפריט - עץ המסכים

**מסך ראשי – Home Page / Main Menu :**

- מבנה, פונקציות, תכנים \ נתונים, בחירה \ קישורים
- שרטוט, או הפניה לאבטיפוס \ מודל מוחשי

**מסכים משניים עיקריים (תפריטי משנה)**

- מבנה, פונקציות, תכנים \ נתונים, בחירה \ קישורים
- שרטוט או הפניה לאבטיפוס \ מודל מוחשי

## עץ המסכים - site map

### 2.4.2 מסכי פעולה

לכל אחד ממסכי \ חלונות הפעולה:

- מבנה, פונקציות, תכנים \ נתונים, בחירה \ קישורים
- שרטוט או הפניה לאבטיפוס \ מודל מוחשי

### 2.5 תהליכים

תהליכים מוגדרים כשרשרת טרנזקציות שביצוען ברצף (sequence) מבצע תהליך ארגוני/עסקי אחד. אם טרנזקציה היא האטום, אזי תהליך הוא המולקולה. קיים אפוא קשר הדוק ביותר בין רכיב זה ורכיב הטרנזקציות הסמוך ורצוי להגדירם ביחד. (ההחלטה על הגדרה Top/down או Bottom/up, היינו, קודם תהליכים ומהם לגזור טרנזקציות, או ההפך, קודם טרנזקציות ואח"כ לבנות מהם את התהליכים - היא בידי המשתמש והכלי שבו הוא עובד).

מבין שניהם, יש עדיפות לטרנזקציות (לאטומים) ואם הן מוגדרות היטב ובאופן ברור (כולל קשר למסכים, קבצים וכ"ו), יש לשקול היטב את מידת ההשקעה ברכיב זה. זאת, בתנאי שניתן להצביע על התהליך שהטרנזקציה היא חלק ממנו. הסיבה למתן העדיפות לטרנזקציות היא שבמערכת ממוכנת תהליכים ממומשים ע"י טרנזקציות. תהליך הוא רצף לוגי של טרנזקציות המבוצעים תוך התניות וניווט. הרצף, היינו התהליך, נוצר באמצעות סינכרוניזציה בבסיס הנתונים. במלים אחרות, כל טרנזקציה בודקת בתחילת פעולתה את מצב הקבצים עליהם היא מתכוננת לפעול, כולל flags למיניהם, ובהתאם למצב זה מחליטה אם וכיצד לפעול.

תרומה עיקרית של רכיב זה היא בכיסוי של טרנזקציות ידניות. במלים אחרות, בתהליכים מודגשות גם פעילויות שאינן ממוכנות והן חלק מהתהליך. תרומה חשובה אחרת היא תיאור המסלולים הדינמיים בין המסכים, היינו אפשרויות הניווט בין המסכים, לעתים גם ישירות ממסך פעולה אחד לשני שלא דרך מסכי התפריט, על מנת לבצע רצף לוגי של פעולות. מסלולי פעולה אלה הם התהליכים.

טכניקה מקובלת לתיאור תהליכים היא Block Diagram ומומלץ מאד להשתמש בכלי גרפי התומך בטכניקה זו. טכניקה אחרת היא דיאגרמת ELH: Entity Life History המראה את מחזור החיים של ישות מידע מסוימת.

רכיבים 2.3, 2.5, ו-2.6 מגדירים, ביחד, את החלק האלגוריתמי - המודל הדינמי - של המערכת ברמות שונות של פירוק (decomposition). חשוב לאזן ביניהם. המלצת מפת"ח היא לייחד את 2.3 לרמת הפירוק העליונה של המערכת, היינו החלוקה לתת מערכות ומנגד, את 2.6 לרמת הפירוק הנמוכה ביותר, היינו לפעולות הבסיסיות ביותר של המערכת - האטומים. רכיב 2.5 מיועד לכל רמות הפירוק הביניים - תהליכים ותת תהליכים. בפועל, האיזון בין שלושה רכיבים מרכזיים אלה ייתכן במספר אפשרויות, מהקל אל הכבד, כדלהלן:

- מערכת פשוטה מאד. רכיבים 2.3 ו-2.5 מיותרים למעשה. יש קשר ישיר בין רכיב 2.4 ממשק המשתמש (תת רכיב 2.4.2 בפרט) ורכיב 2.6 טרנזקציות. שני רכיבים אלה מכסים למעשה את המודל הדינמי (החלק האלגוריתמי) של המערכת. מקרה מיוחד זה מכוסה בערכה מערכות קטנות שבכרך ניהול פרויקטים.
- מערכת פשוטה עד בינונית עם חלוקה ברורה לתת מערכות: מרכז הכובד הוא רכיבים 2.3 ו-2.6. יש קשר ישיר ביניהם (וביניהם ובין 2.4). רכיב 2.5 מנוון (מפנה אליהם או לא קיים כלל).

- מערכת פשוטה עד בינונית בלי חלוקה לתת מערכות: מרכז הכובד הוא רכיבים 2.5 ו- 2.6. רכיב 2.3 נותן הסבר כללי ותמציתי בלבד ומפנה מיד אל רכיב 2.5. אפשר בעצם להבין את המערכת ישירות מתוך רכיב 2.5 אשר מתחבר לרכיב 2.4 מצד אחד ולרכיב 2.6 מצד שני.
  - מערכת בינונית עד מורכבת: כל שלשת הרכיבים, 2.3, 2.5 ו- 2.6 חשובים כמוסבר להלן.
  - מערכת מורכבת המפותחת במספר מהדורות \ יחידות מסירה: רכיב 2.3 הוא מרכז הכובד (בתיק הראשי של המערכת). הוא מפנה לנספחים 2.3.1, 2.3.2 וכו' או לתיקי תת מערכות (מהדורות, יחידות מסירה) שם נמצא פרק היישום של כל תת מערכת. רכיבים 2.5 ו- 2.6 כמו כל שאר רכיבי היישום (בתיק הראשי) מתארים רק את הרכיבים המשותפים לכל תת המערכות. מקרים מיוחדים אלה מכוסים בערכה פיתוח במספר יחידות מסירה ובערכה מערכות גדולות ותת מערכות שבכרך ניהול פרויקטים.
  - בכל מקרה, יש לשמור על קשר ברור בין רכיב 2.5 ו- 2.3 לעיל. רכיב 2.5 מתחיל במקום שבו רכיב 2.3 הסתיים. השורשים של 2.5 הם העלים של 2.3 (לפי תת הרכיבים \ תת המערכות: 2.3.1, 2.3.2 וכו'). תהליך 2.5.X הוא המשך ישיר של תת מערכת 2.3.X. בהתאם לכך, רכיב 2.5.1 יפרט את כל התהליכים התומכים במנגנונים הרחוביים של המערכת (ראה תת מערכת 2.3.1 לעיל).
- פירוק (decomposition) של התהליך ייתכן במספר אפשרויות (רמות). הפירוק יכול להסתיים כמוצג לעיל ברמה 3 (2.5.1.1), להמשיך לרמה נמוכה יותר (2.5.1.1.1) או להיעצר ברמה גבוהה יותר (2.5.1). הכל בהתאם למורכבות וסביכות התהליך. בכל מקרה, הרמה הנמוכה ביותר, העלה של עץ התהליכים, תקושר לרכיב 2.6 ודרכו לרכיבים 2.4 ו- 2.11 או ישירות אליהם.
- הפירוק לא חייב להיות סימטרי לרוחב עץ התהליכים. ענף מסוים יכול להסתיים ברמה 2 (להיעצר ברכיב 2.5.1 למשל), בעוד שענף אחר יכול להסתיים ברמה 4 (רכיב 2.5.4.1.1 למשל).

## 2.5.0 אינדקס כללי

רשימה \ טבלה של כל התהליכים במערכת. לכל תהליך (שורה בטבלה) המאפיינים הבאים:

- שם וסימול (קישור לרכיב 2.3)
- סוג
- רמת פירוק (סופי, בר-פירוק)

## 2.5.X תהליך X

- שם וזיהוי
- סוג: BATCH, אוטומטי, ידני וכו'
- שיוך פונקציונאלי (קשר ל- 2.3),
- תיאור כללי: תרשים (תרשים זרימה, Block diagram, תרשים היררכי וכו')
- הגורמים המשתתפים (התחנות בארגון)
- פירוט ופירוק - הטרינזקציות המשתתפות: ידניות או ממוכנות (קשר לרכיב 2.6!).
- תיאור מפורט: מלל מובנה, קשר לקבצים, קשר לאחור למטרות.

## 2.5.Y.X תת תהליך Y

תיאור מפורט של תת תהליך Y עפ"י המפורט בסעיף לעיל.

- תיאור במלל
- תרשים כגון DFD ברמה 3 אשר אינו מתפרק יותר
- הפנייה לטרנזקציות המתאימות המרכיבות את תת התהליך

## 2.6 טרנזקציות

ראה תחילה ההנחיות ברכיב 2.3 לעיל וכן רכיב 2.5 לעיל. רכיב 2.5 הדן בתהליכים מכיל גם הגדרה לטרנזקציה.

קיימות שיטות וטכניקות רבות להגדרת טרנזקציות במערכת. מפת"ח, בהיותו נוהל-מסגרת פתוח, איננו מכתוב שיטה וטכניקה זו או אחרת. יש לראות אפוא את ההנחיות הבאות כהמלצות בלבד, בחלקן **המלצות חמות**, כגון שכל טרנזקציה תקבל סימול ברור.

חלוקה לטרנזקציות יכולה להיות במספר שיטות:

- ע"י פירוק פונקציונלי בהמשך ל- 2.3 לעיל
  - ע"י פירוק תפעולי בהמשך ל- 2.4 (עץ המסכים)
  - ע"י המשך הפירוק לתהליכים מהרכיב הקודם (2.5)
  - ע"י פירוק התואם את החלוקה לקבצים לוגיים (רכיב 2.11)
  - ע"י פירוק התואם את החלוקה לישויות המידע (רכיב 2.13)
  - בהתאם לחלוקה למשתמשים (2.2)
  - ע"י שילוב חלקי או מלא של הנ"ל: 2.2, 2.3, 2.4, 2.5 ו- 2.11/2.13
  - בלי שום היררכיה, כווקטור רציף, טרנזקציה אחר טרנזקציה
  - בכל סדר ברור אחר.
- כך או כך, ברור שכל חלוקה יוצרת, לכתחילה או בדיעבד, גם הקבצה של הטרנזקציות לקבוצות כגון: קבוצת טרנזקציות המופעלת ממסך תפריט מסוים; קבוצת טרנזקציות השייכת לתהליך מסוים; קבוצת טרנזקציות המטפלת בקובץ מסוים וכו'. הקבצה זו נובעת ישירות משיטת החלוקה שנבחרה. מפת"ח, כאמור, איננו מכתוב שיטה מסוימת, אלא רק מניח שיש חלוקה כלשהי לקבוצות.
- חלוקה נוספת בתוך כל קבוצה תהיה בשיטה המתאימה למערכת הספציפית, אך תדגיש את הסיווג לפי:

- טרנזקציות קלט ועדכון
- אוסף טרנזקציות לאחזור מידע
- טרנזקציות Batch
- טרנזקציות יוצרות/מקבלות ממשקים למערכות מידע אחרות
- טרנזקציות לתחזוקת/תפעול המערכת (HouseKeeping)

ההקבצה איננה ישות נוספת בעץ המערכת, אלא קשר בין רכיבים קיימים ועם כל הקושי (לכאורה), יש בכך הקלה ופישוט. בשלב האפיון תהיה הקבצת הטרנזקציות עם רכיב מוביל מסוים, כנזכר בהנחיה לעיל. בעיצוב המערכת, תתחדד ההקבצה ותתלכד עם יותר מרכיב אחד, למשל, עם התהליכים (2.5) והמסכים (2.4), או עם המסכים והקבצים (בגישת האובייקטים, ראה קיט בשם זה בכרך נושאים תומכים).

יש להקפיד על זיהוי ברור של הטרנזקציה הבדידה (שם וספרור). זיהוי זה יתאים לשיטת החלוקה (ההקבצה) שנבחרה. (ראה גם הפרק סימול ישויות - קודיפיקציה בקיט תיעוד בכרך נושאים תומכים). רצוי מאד להקפיד על קשר לאחור (trace back) של הטרנזקציות (או התהליכים) עם המטרות. היינו, לציין ליד כל טרנזקציה (או תהליך) את המטרה (מטרות) אותה(ן) היא מממשת.

## 2.6.0 אינדקס כללי

רשימה \ טבלה של כל הטרנזקציות במערכת. לכל טרנזקציה (שורה בטבלה) המאפיינים הבאים:

- שם וסימול
- סוג (קלט, פלט, עדכון וכו')
- הקבצה (קשר לתהליכים או מסכים, או קבצים)

## 2.6.X טרנזקציה X

### תיאור טרנזקציות קלט ועדכון

- מסכי הפעולה (לפי הפורמט שצוין ב- 2.4), קלט ופלט
- הדק (trigger) ההפעלה
- אלגוריתם הפעולה: מה עושה טרנזקציה זו. אפשר לציין זאת במלל מובנה, Pseudo code, תרשים זרימה או כל צורה ברורה אחרת
- קשר לרכיב 2.16 קלטים
- קבצים ורכיבים אחרים משתתפים: בראייה החלקית (partial view) של טרנזקציה זו, בהתייחסות ברורה לרכיבים הספציפיים: 2.9 שגרות, 2.10 טבלאות, 2.11 קבצים לוגיים וכו'
- קשר לאחור (who uses) Where used למטרות בקרת שלימות

### תיאור טרנזקציות לאחזור מידע: שאילתות ודוחות

אין טעם לפרט כל דו"ח וכל טרנזקציה היוצרת אותו. יש לקבץ את כל הדוחות והשאילתות הפועלים על אוסף קבצים מוגדר, לקבוצה אחת ולתארם במרוכז. **הקבצה זו תתואם עם רכיב 2.15** (דוחות המערכת) להלן. גישה זו תואמת גם את השימוש שבוודאי יעשה במחוללי שאילתות ודוחות (ראה 3.14).

### תיאור טרנזקציות batch:

- מסך השיגור
- אלגוריתם הפעולה
- קבצים משתתפים
- מסך הפלט (שהוא לעתים הדו"ח).

- היבטים תפעוליים.

#### תיאור טרנזקציות יוצרות/מקבלות ממשקים:

- בדומה לטרנזקצית קלט/עדכון בתוספת קשר לרכיב 2.22 ממשקים

#### תיאור טרנזקציות לתחזוקת/תפעול המערכת ((housekeeping):

- בדומה לטרנזקצית batch

פירוט הסעיפים מעבר לסעיף 2.6.0, יהיה כמוצע לעיל (בהקבצות המסומנות (x), או ללא הקבצה אלא במספר רץ בלבד 2.6.n, או בסימול אחר (...t) ראה פרק סימול ישויות – קודיפיקציה בקיט תיעוד) תחת סעיף 2.6.1 פירוט הטרנזקציות, או בכל דרך מסודרת אחרת.

## 2.7 מודולים (תכניות)

הגדרת רכיב זה איננה בד"כ חלק מאפיון המערכת ושייכת לשלב העיצוב. ההנחיות והפירוט שלהן ניתנים משתי סיבות עיקריות:

- **עבור העיצוב.** העיצוב יעשה כהמשך ישיר לאפיון ותיק העיצוב הוא למעשה השלמה וכתובה סופית של תיק זה.
- **עבור האפיון.** במקרים מיוחדים בהם כבר בשלב האפיון ידוע שהמערכת צפויה לשאול מודולים קיימים וזמינים בארגון (במערכות מידע אחרות). במקרה זה, תיעוד המודולים יהיה פשוט ומייד, ע"י הוצאת תדפיסים מהמערכת המשאילה או ע"י הפניה למערכת ניהול ספריות מקור מסודרת.

מבנהו המדויק של רכיב זה מושפע מאד מכלי הפיתוח והתחזוקה, כולל כלים לניהול תצורה וגרסאות. המבנה המוצע להלן יותאם ע"י הפרויקט לסביבת ושיטת הפיתוח הרלוונטיות. בפתוח (במקום טקסט זה) יוסבר המבנה שאומץ. בכל מקרה, הפירוק למודולים ישמור על קשר עם טרנזקציות (רכיב 2.6) או ישירות עם תהליכים (רכיב 2.5).

תיעוד רכיב זה יכול להיעשות בגישה ריכוזית או ביזורית בגישה הריכוזית, כל המודולים של המערכת, כולל אלה שמממשים מסכים, ממשקים יתועדו כאן. בגישה הביזורית, רכיב זה יכיל את המודולים המממשים תהליכים וטרנזקציות בלבד, כהמשך ישיר של רכיבים 2.5 ו-2.6 לעיל. מודולים המממשים מסכים וממשקים יתועדו במקומם (רכיבים 2.4 ו-2.22 בהתאמה). **ההחלטה באיזו גישה לבחור היא בידי הפרויקט.**

לרכיב זה קשר עם רכיבים רבים אחרים, אך בעיקר עם רכיב הטרנזקציות (2.6), שכן טרנזקציות (ותהליכים ופונקציות) ממומשים, בסופו של דבר, ע"י מודולים פסיים.

אם קיימות בארגון תכניות תיק תכנות תקניות, יש להיצמד אליהן. גם אם אין, כדאי שבפרויקט יהיה תקן כזה.

תיעוד מודול הוא תיק תכנות. בפועל, יהיו תיקי התכנות בתוך המודול הפיסי, ישוכנו ישירות בספריות המודולים וייבנו ע"י כלי הפיתוח של שלב העיצוב, הבנייה והקידוד! ההסבר כאן מובא לצורך הסבר והמחשה והיא מיועד עבור מקרים בהם אכן בונים תיקי תכנות מסורתיים.

ראה דיון בתיק תכנות בקיט עיצוב ובניה שבכרך מחזור חיים.



### 2.7.1 תכניות מקור – SOURCE MODULES

רשימה \ טבלה של כל המודולים במערכת, לכל מודול המאפיינים הבאים :

- שם וסימול
- הפנייה לתיק תכנות
- הפנייה לספרייה

### 2.7.2 תכניות ביצוע – EXECUTABLE MODULES

רשימה \ טבלה של כל המודולים במערכת, לכל מודול המאפיינים הבאים :

- שם וסימול
- הפנייה לספרייה
- קישור לתוכניות המקור

## 2.8 מהלכים (פרוצדורות בקרה)

רכיב זה מיועד לתיעוד מהלכי הבקרה של המערכת (Control procedures). מהלכים אלה נקראים גם : scripts, פקודות מערכת ההפעלה וכו', בהתאם למערכת ההפעלה הספציפית. תיעוד הרכיב יהיה בדומה לרכיב 2.7 ע"י הפנייה לספריות המתאימות.

רכיב זה מתאר באופן סטטי את ספריית המהלכים ובדומה לרכיב 2.7 לעיל הוא בא לידי ביטוי בפרט משלב עיצוב ובנייה והלאה. אופן הפעלת המהלכים הוא ברכיב 2.4 ממשק המשתמש לעיל וברכיב 4.4 תפעול המערכת להלן.

## 2.9 שגרות (אובייקטים משותפים)

היחס בין שגרה לוגית לשגרה פיסית הוא, בד"כ, 1:1 ולכן רכיב זה מכיל הן הגדרה לוגית והן הגדרה פיסית של שגרות המערכת. בשלב האפיון יש להתרכז בהגדרות הלוגיות ולהשאיר את ההגדרות הפיסיות לשלב העיצוב והבנייה.

שימוש בשגרות משותפות הוא גורם חיובי וחשוב ביותר, גם מנקודת ראות צרה של המערכת הספציפית, משום :

- ההסתמכות על קטעי מערכת מנופים ובדוקים
- החסכון בכתיבה
- הגברת אחידות ותקינה. התשלום, לעומת זאת, הוא בצורך לוודא באופן שוטף שכל הכנסת שינויים בשגרות אלה נעשית תוך התחשבות במערכת הספציפית (שלנו).

שגרות מטפלות במגוון אלגוריתמים משותפים ומרכזיים כגון : טפול בתאריכים במבנים שונים, כולל בדיקות לוגיות והסבה ממבנה למבנה. פריטי מידע וטבלאות ; נוסחאות מתמטיות ; תרגום פורמטים ; הודעות על שגיאות (שליחת messages) ; כתיבה, כתיבה ל- log, טיפול באמצעי קלט/פלט או קבצים מיוחדים וכו'.

שגרות משותפות תתכנה במספר רמות :

- ברמת הפרויקט, היינו, שיתוף בין טרנזקציות ומודולים של המערכת עצמה

• ברמת הארגון, היינו שימוש בספריית שגרות מרכזית המשותפת לכל מערכות המידע בארגון.  
הרמה הארגונית היא כמובן עדיפה. לפני שמגדירים ספריית שגרות פרטית למערכת, יש לבדוק קיומה של ספרייה מרכזית בארגון ולהשתמש בה. במידה שאין כזו, אין מנוס מאפיון מלא של רכיב זה (ובנייתו בהמשך), אך יש לוודא שלפחות ברמת המערכת הספרייה משותפת לכל הטרנזקציות והמודולים.  
מההיבט הטכני, שגרה היא מודול לכל דבר. מבנה שגרה (תיק תכנות) יהיה אפוא בדומה למבנה מודול. ראה 2.7 לעיל.

כאמור, היחס בין שגרה לוגית לשגרה פיסית הוא, בד"כ, 1:1. זה יתרון ברור לעומת מודול רגיל המיישם טרנזקציה. התשלום הוא בכך שיש להקפיד על רישום של where used ולזכור את כל המשתמשים בשגרה. (רצוי כמובן שהרישום ייעשה בעזרת כלי ממוחשב - ניהול תצורה - ראה רכיבים 3.12 ו-3.13 להלן). כדאי מאד לזכור ששגרה משותפת תיתכן:

- ברמה פיסית, היינו, שימוש בקוד משותף
- ברמה לוגית, היינו, אלגוריתם משותף המקודד בשפות שונות.  
חשוב מאד להדגיש את הקשר של רכיב מרכזי זה עם רכיבים מרכזיים אחרים:
- טבלאות מרכזיות (רכיב 2.10 הסמוך!)
- טרנזקציות (2.6) ומילון השדות (2.13).

### 2.9.1 שגרות מקומיות

רשימה \ טבלה של כל השגרות המקומיות ופרטיות שנכתבות כחלק מפיתוח המערכת ושייכות רק לה. לכל שגרה או שורה בטבלה המאפיינים הבאים:

- שם וסימול
- תיאור קצר
- הפנייה לספרייה

#### 2.9.1.X שגרה X

תיאור מפורט של שגרה X הכולל:

- תרשים זרימה – Flowchart
- קשר של Where used עם הרכיבים האחרים המשתמשים בשגרה
- ספריית מקור וספריית ביצוע.

### 2.9.2 שגרות ארגון

רשימה \ טבלה של כל השגרות הארגוניות המשותפות למערכות אחרות. לכל שגרה או שורה (בטבלה) המאפיינים הבאים:

- שם וסימול
- תיאור קצר
- הפנייה לספרייה מרכזית

### 2.9.3 שגרות צד שלישי

רשימה \ טבלה של כל השגרות החיצוניות המסופקות ע"י יצרן חיצוני. לכל שגרה או שורה בטבלה המאפיינים הבאים:

- שם וסימול
- תיאור קצר
- הפנייה לספרייה מרכזית

### 2.10 טבלאות קודים

שים לב להבדל בין רכיב זה ורכיב 2.11 הסמוך. ברכיב זה הכוונה לטבלאות מרכזיות, טבלאות במובן של קודים ופרמטרים אשר ניתנים בד"כ לכוונון חיצוני. טבלאות במינוח של בסיסי נתונים טבלאיים, שהם קבצי מידע לכל דבר, מקומם ברכיב 2.11 הסמוך. טבלאות קודים מקומיות (פרטיות) השייכות רק למערכת הנדונה ואשר משתנות במהלך פעילות המערכת, דינן כקובץ לכל דבר ויכללו ברכיב 2.11 להלן.

היחס בין טבלה לוגית לטבלה פיסית הוא, בד"כ, 1:1 (לאחר נרמול) ולכן רכיב זה מכיל הן הגדרה לוגית והן הגדרה פיסית של טבלאות המערכת. בשלב האפיון יש להתרכז בהגדרות הלוגיות ולהשאיר את ההגדרות הפיסיות לשלב העיצוב והבנייה.

שימוש בטבלאות משותפות הוא גורם חיובי וחשוב ביותר, גם מנקודת ראות צרה של המערכת הספציפית, משום:

- הגברת אחידות ותקינה
- ההסתמכות על קטעי מערכת מנופים ובדוקים
- החיסכון בכתיבה.

התשלום, לעומת זאת, הוא הצורך לוודא באופן שוטף שכל הכנסת שינויים בטבלאות אלה נעשית תוך התחשבות במערכת הספציפית (שלנו).

הגדרה מלאה ומפורטת של טבלאות המערכת, עשויה להיות קשה ומפרכת, אך "כגודל הצער, כך השכר". להשגת מטרה חשובה זו, אין להוציא מכלל אפשרות שימוש בשמות נרדפים, synonym, או טבלאות המרה, ובלבד שמוסכם על מהות הישות. יש כמובן להימנע משמות זהים למשמעויות שונות, homonym.

טבלאות משותפות תתכנה במספר רמות: ברמת הפרויקט, היינו, שיתוף בין טרנזקציות ומודולים של המערכת עצמה ברמת הארגון, היינו, שימוש בספריית טבלאות מרכזית המשותפת לכל מערכות המידע בארגון. הרמה הארגונית היא כמובן עדיפה. לפני שמגדירים ספריית טבלאות פרטית למערכת, יש לבדוק קיומה של ספרייה מרכזית בארגון ולהשתמש בה. במידה שאין כזו, אין מנוס מאפיון מלא של רכיב זה (ובנייתו בהמשך), אך יש לוודא שלפחות ברמת המערכת הספרייה משותפת לכל הטרנזקציות והמודולים.

כאמור, היחס בין טבלה לוגית לטבלה פיסית הוא, בד"כ, 1:1 וזה יתרון ברור. עם זאת, יש להקפיד על רישום של where used ולזכור את כל המשתמשים בטבלה. (רצוי כמובן שהרישום ייעשה בעזרת כלי ממוחשב - ניהול תצורה - ראה רכיבים 3.12 ו-3.13 להלן).

חשוב מאד להדגיש את הקשר של רכיב מרכזי זה עם רכיבים מרכזיים אחרים:

- שגרות (רכיב 2.9): במקרים רבים הגישה לטבלה היא באמצעות שגרה, שגרה וטבלה מתלכדים לעתים קרובות!
  - קבצים (2.11): יש לשים לב למקרים בהם טבלה היא בעצם קובץ נתונים לכל דבר.
  - מילון פריטי מידע (2.13): הימנע מכפילויות!
  - מילון מונחים (2.1.4): הימנע מכפילויות!
- יש לעשות שימוש מרבי ב**טבלאות חיצוניות** ולדרוש שהיישום יהיה בכלי ממוכן. טבלאות חיצוניות (ומשותפות) המאפשרות שינוי פרמטרים חיצוני מבלי לשנות תוכניות, הם רכיב חשוב ביותר באפיון המערכת ובנייתה.
- ביחידות צה"ל (וכן בכל ארגון/גורם המקפיד על חיסוי המערכות וסיווגן הביטחוני או העסקי), יש גם לפרט מי הגורמים הרשאים לגשת לטבלאות, תוך ציון לאילו צרכים, מטרות ומה הפעולה המותרת לכל גורם. ראה גם הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. גופים הכפופים להנחיות רב"מ/במ"ת יפעלו בסעיף זה עפ"י הנחיות אלה.
- בכל שלושת רמות הטבלאות שלהלן: מקומי, ארגוני וחיצוני, יש להישען על הגדרות תקניות במירב האפשרי.

#### 2.10.1 טבלאות מקומיות

רשימה/אינדקס בשיטת מיון ברורה של כל הטבלאות המקומיות/פרטיות שנכתבות כחלק מפיתוח המערכת ושייכות רק לה. לכל טבלה או שורה בטבלת האינדקס המאפיינים הבאים:

- שם הטבלה
- סמל הטבלה
- תיאור קצר
- סיווג ביטחוני/עסקי

#### 2.10.1.X טבלה X

- שם הטבלה וסימולה
  - שמות נרדפים
  - מבנה פנימי: שם הישות, קוד, שדות נוספים(!)
  - מרחב הערכים (Domain) ובדיקות סבירות
  - מקור (סוג) הטבלה
  - תקן ישראלי/בינ"ל
  - תקן ממשלתי
  - טבלה פרטית של המערכת, טבלאות מיוחדות לתת-מערכות מסוימות
  - סיווג, מידור והרשאות: מי רשאי לגשת ולצורך מה.
- השווה עם התיאור המפורט של קבצים, ב- 2.11 ו- 2.12 להלן.

## 2.10.2 טבלאות ארגון

רשימה \ אינדקס של כל הטבלאות המשותפות למערכות אחרות בארגון. כמפורט בסעיף 2.10.1 לעיל.

### 2.10.2.X טבלה X

ראה 2.10.1.X לעיל

## 2.10.3 טבלאות חיצוניות

רשימה \ אינדקס של כל הטבלאות החיצוניות גלובליות באחריות ארגון אחר כמפורט בסעיף 2.10.1 לעיל.

### 2.10.3.X טבלה X

ראה 2.10.1.X לעיל

## 2.11 קבצים לוגיים

רכיב זה מגדיר את בסיס הנתונים הלוגי - מודל הנתונים.

תמונה ברורה של הקבצים הלוגיים במערכת היא חיונית וגם אפשרית, כבר בשלב האפיון (הגדרת המערכת). ברוב מערכות המידע, אפיון מלא של רכיב זה, הוא חלק נכבד מאפיון המערכת כולה, רכיב היישום בפרט.

הגדרת קבצים לוגיים כוללת:

- הגדרה תמציתית של כל קובץ
- הגדרה פרטנית של כל קובץ (ראה 2.11.x להלן)
- סכימה כוללת של עץ (או רשת) הקבצים, היינו, הקשרים בין הקבצים
- סכימות חלקיות (פרטיות) לתת-מערכות או לטרנזקציות, לפי הקבצות. שים לב, שרכיב זה הוא **בסיס הנתונים הלוגי** של המערכת, או כפי שהוא גם מכונה, **מודל הנתונים**, לא בסיס הנתונים הפיסי. מודל זה צריך לשקף את כל המידע הדרוש ואת כל דרכי הגישה השונים (המסלולים) הנחוצים.

הגדרת הסכימות תיעשה בשיטה מקובלת (דוגמת ERD, IA, נרמול טבלאי וכו'), רצוי תוך שימוש בכלי ממוכן. יש להדגיש את סוג הקשר: אחד לאחד (1:1), אחד לרבים (1:M) ושדה-מפתח מקשר.

חיוני ביותר שהגדרה זו תישען על מילון פריטי מידע מוסכם וברור. מכאן הקשר ההדוק עם רכיב/סעיף 2.13 להלן.

קשרים עם רכיבים מרכזיים אחרים הם:

- טרנזקציות והמודולים (סעיפים 2.6 ו-2.7)
- מילון המונחים (רכיב 2.1.4) שגרות (רכיב 2.9)

במקרים רבים הגישה לקובץ היא באמצעות שגרה! הקשר העיקרי הוא כמובן עם רכיב הקבצים הפיסיים (2.12). הנחיות לגבי קשר זה נמצאות ברכיב 2.12 להלן. יש להיזהר מכפילות עם רכיב הטבלאות (2.10) וממקרים בהם קובץ ניתן להגדרה פשוטה יותר כטבלה.

קבצים ייתכנו במספר רמות :

- ברמת הפרויקט, היינו, קבצים המשרתים רק מערכת ספציפית
- ברמת הארגון, היינו קבצי מאסטר, משותפים למספר מערכות בארגון
- קבצים זמניים (קבצי ביניים) קבצי ממשק.

חשוב לציין הבחנות אלה, כולל אופן הגישה של המערכת לקבצים (קריאה בלבד או גם כתיבה).

בהגדרת קבצים לוגיים יש להקפיד על כללי הנרמול, אך אין להגזים. נרמול מהרמה השנייה (כל קובץ לוגי מכיל הגדרה של ישות אחת ברורה, שדה יחידני המזהה אותה וכל שאר השדות מתארים אותה), הוא בד"כ מספק.

לגבי סימול הקבצים, ראה המלצה בפרק סימול ישויות - קודיפיקציה בקיט תיעוד בכרך נושאים תומכים.

ביחידות צה"ל (וכן בכל ארגון/גורם המקפיד על חיסוי המערכות וסיווגן הביטחוני או העסקי), יש גם לפרט מי הגורמים הרשאים לגשת לקבצים (הלוגיים!), תוך ציון לאילו צרכים, מטרות ומה הפעולה המותרת לכל גורם. ראה גם הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. גופים הכפופים להנחיות רב"מ/במ"מ יפעלו בסעיף זה עפ"י הנחיות אלה.

## 2.11.0 כללי – מודל הנתונים

### סכמות

• סכמה כללית: ERD, DSD או Class Diagram

• סכמות משנה – Partial views

**אינדקס/טבלה של הקבצים הלוגיים, כולל המאפיינים הבאים:**

• שם וסימול

• שדה מפתח

• תיאור כללי

### 2.11.x קובץ לוגי X

• שם הקובץ וסימולו

• שמות נרדפים

• תיאור תמציתי

• שיטת הארגון (סדרתי, גישה ישירה, רב-מפתחי)

• מבנה פנימי: השדות המרכיבים את הקובץ

• בהפניה למילון, רכיב 2.13

• דגש על שדה-מפתח ראשי (יחידני) ושדות מפתח משניים.

• אופן גישה עיקרי (קריאה, כתיבה/קריאה)

• משתמשים עיקריים (קשר לאחור לרכיב 2.2)

- קשר לאחור לטרנזקציות ותהליכים
- דרישות אבטחת מידע: סיווג, מידור והרשאות: מי רשאי לגשת ולצורך מה.
- מרחב הערכים (domain) ובדיקות סבירות
- היקפים (גודל, כמויות) וקצב התחלפות הנתונים (volatility)
- רמת פירוט, עומק הנתונים, היסטוריה
- רמת עדכניות
- קשרים ותלויות בקבצים אחרים, מידת שיתוף נתונים
- רמת ביזור לוגי ותפעולי
- דרישות גיבוי ואמינות.

## 2.12 קבצים פיסיים – DATA BASE

הגדרת רכיב זה איננה בד"כ חלק מאפיון המערכת ושייכת לשלב העיצוב. ההנחיות והפירוט שלהלן ניתנים משתי סיבות עיקריות:

- **עבור העיצוב.** בד"כ העיצוב נעשה כהמשך ישיר לאפיון ותיק העיצוב הוא למעשה השלמה וכתובה סופית של תיק האפיון. אפשר להשתמש בתיק האפיון גם לצורך העיצוב.
  - **עבור האפיון.** המערכת צפויה לשאול קבצים פיסיים קיימים וזמינים בארגון. במקרה זה, תיעוד הקבצים יהיה פשוט ומייד, ע"י הוצאת תדפיסים מהמערכת המשאילה או ע"י הפניה לבסיס נתונים קיים, בפרט אם יש לבסיס הנתונים data dictionary מסודר.
- הצגת הקשר בין קבצים פיסיים ללוגיים היא חיונית ביותר לבדיקת שלימות מודל הנתונים ונכונות האפיון ואיננה בהכרח דרישה לתוספת מאמץ. אדרבא, היא יכולה לחסוך עבודה רבה ברכיב זה (או בקודם) ולמנוע כפילות עם הרכיב הקודם (קבצים לוגיים). זאת, בעיקר במקרים בהם הקשר בין קובץ לוגי לקובץ פיסיה הוא 1:1.

הגדרת קבצים פיסיים כוללת:

- הגדרה פרטנית של כל קובץ
- סכימה כוללת של עץ (או רשת) הקבצים, היינו, קשרים בין הקבצים
- סכימות חלקיות (פרטיות) בחלוקה פיסית משנית של בסיס הנתונים
- קשר עם הקבצים הלוגיים.

שים לב שרכיב זה מכיל את ההגדרות לבסיס הנתונים הפיסי של המערכת וקשור קשר הדוק עם רכיב 3.11, להלן.

הגדרת הסכימות תיעשה תוך הדגשת מודל בסיס הנתונים שבשימוש:

- קבצים עם אינדקס (index-sequential)
- קבצים בגישה ישירה (direct, random, hashing וכו')
- בסיס נתונים היררכי
- בסיס נתונים רשתי

- בסיס נתונים טבלאי
- בסיס נתונים של אובייקטים (בסביבת שרת/לקוח) וכו'.
- שים לב שבסיס הנתונים עצמו (ה-DBMS) מתואר ברכיב 3.11!
- רצוי להשתמש בכלי ממוכן להצגת הסכימות, כלי המאפשר:
  - דילול הסכימות הפיסיות מתוך הסכימות הלוגיות
  - הגדרה (אוטומטית) של הסכימות הפיסיות (שפת DDL)
  - הכנסת שינויים בסכימות הפיסיות.
- יש להדגיש את סוג הקשר - אחד לאחד (1:1), אחד לרבים (1:M), את שדה המפתח המקשר, את המפתחות המשניים ואת המסלולים האפשריים.
- חיוני ביותר להראות כיצד מקושרים הקבצים הפיסיים עם מילון פריטי המידע הפיסי (רכיב 3.12).
- יש להדגיש את הקשרים עם רכיבים מרכזיים אחרים כמו:
  - טרנזקציות (רכיב 2.6) ומודולים (רכיב 2.7)
  - מילון פריטי מידע (רכיב 2.13)
  - שגרות (רכיב 2.9). במקרים רבים הגישה לקובץ היא באמצעות שגרה!
- הקשר העיקרי הוא, כמובן, עם רכיב הקבצים הלוגיים (2.11) מחד ועם בסיס הנתונים הפיסי (3.11) מאידך.
- קבצים ייתכנו במספר רמות:
  - ברמת הפרויקט, היינו, קבצים המשרתים רק מערכת ספציפית
  - ברמת הארגון, היינו קבצי מאסטר, משותפים למספר מערכות בארגון
  - קבצים זמניים (קבצי ביניים)
  - קבצי ממשק.
- חשוב לציין הבחנות אלה, כולל אופן הגישה של המערכת לקבצים (קריאה בלבד או גם כתיבה).
- בהגדרת הקבצים הפיסיים, בשלב העיצוב, ייתכן De-normalization הנובע משיקולי ביצועים.
- ביחידות צה"ל (וכן בכל ארגון/גורם המקפיד על חיסוי המערכות וסיווגן הביטחוני או העסקי), יש גם לפרט מי הגורמים הרשאים לגשת לקבצים הפיסיים, תוך ציון לאילו צרכים, מטרות ומה הפעולה המותרת לכל גורם. ראה גם הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. גופים הכפופים להנחיות רב"מ/במ"ת יפעלו בסעיף זה עפ"י הנחיות אלה.

## 2.12.0 מודל כללי

1. סכמה כללית: Physical ERD או Storage Diagram
2. סכמות משנה
3. אינדקס הקבצים הפיסיים – טבלה, מאפיינים עיקריים:
  - שם הקובץ (גנרי ושם במערכת ההפעלה)



- סמל הקובץ (זיהויו במערכת ההפעלה)
- תיאור קצר
- סיווג ביטחוני או עסקי

#### 2.12.x קובץ פיסי X

- שם הקובץ וסימולו
- שמות נרדפים
- תיאור תמציתי
- שיטת הארגון (סדרתי, גישה ישירה, רב-מפתחי)
- הגדרות היוצרות את הקובץ (DDL)
- הגדרות המאפשרות גישות שונות לקובץ, כולל אבטחת מידע
- מבנה פנימי: השדות המרכיבים את הקובץ
- בהפניה למילון, רכיב 2.13
- דגש על שדה-מפתח ראשי (יחידני) ושדות מפתח משניים.
- אופן גישה עיקרי (קריאה, כתיבה/קריאה)
- משתמשים עיקריים (קשר לאחור לרכיב 2.2)
- קשר לאחור לטרנזקציות ותהליכים (דרך הקשר הלוגי, או ישירות)
- הגדרות (מסלולי) אבטחת מידע: מידור והרשאות: מי רשאי לגשת ולצורך מה.
- מרחב הערכים (domain) ובדיקות סבירות
- היקפים (גודל, כמויות) וקצב התחלפות הנתונים (volatility)
- רמת פירוט, עומק הנתונים, היסטוריה
- רמת עדכנות
- קשרים ותלויות בקבצים אחרים, מידת שיתוף נתונים
- רמת ביזור לוגי ותפעולי
- רמת השיתוף: קובץ פרטי למערכת, שיתוף בקריאה בלבד וכו'
- דרישות גיבוי ואמינות.

#### 2.13 מילון פריטי-מידע (שדות)

פריטי מידע (שדות) מוכלים במסכים, דוחות, קבצים, מסרים וממשקים פנימיים וחיצוניים. פריטי מידע ייתכנו בארבע רמות שונות: פרטיים (מודול), מקומיים (מערכת), ארגוניים וגלובליים. רכיב זה עוסק בשלושת הרמות האחרונות, היינו, בכל הרמות להוציא את רכיבי המידע הפרטיים (זמניים) שבשימוש מודול מסוים ואינם נמצאים בשום ממשק או מסר שיוצא ממנו או נכנס אליו. בכל שלושת הרמות שלהלן: מקומי (בין מודולים במערכת אבל לא מחוצה לה), ארגוני(בין מערכות בארגון אבל לא מחוצה לו) וגלובלי (בין ארגונים), יש להישען על הגדרות תקינות במירב האפשרי.

בדומה לשגרות וטבלאות, גם בפריטי מידע, היחס בין הישות (פריט המידע) הלוגית לפיסית הוא, בד"כ, 1:1 ולכן רכיב זה מוגדר במפת"ח כרכיב אחד. בשלב האפיון יש להתרכז כמובן בהגדרות הלוגיות ולהשאיר את ההגדרות הפיסיות לשלב העיצוב והבנייה.

רכיב זה איננו מילון הישויות הכולל של המערכת, אלא מילון שדות (פריטי מידע - data items) בלבד. מילון נתונים במשמעות של כלי קיטלוג מרכזי של כל ישויות המערכת (repository) או ככלי הצמוד לבסיס הנתונים (data dictionary), מקומו ברכיב 3.12.

הגדרה מסודרת של פריטי המידע היא בסיס לכל ניסיון ליצור אחידות, תקינה ושפה משותפת בין כל הגורמים המעורבים בפרויקט. הגדרה זו היא התשתית ל:

- הגדרת קבצי המערכת (2.11, 2.12)

- הגדרת טפסים ומסכים (2.4)

- הגדרת דוחות (2.15) וקלטים (2.16) ועוד.

הגדרה זו עשויה להיות קשה ומפרכת, אך "כגודל הצער - כך השכר". להשגת מטרה חשובה זו, אין להוציא מכלל אפשרות שימוש בשמות נרדפים (synonym) או דומים (לקוח-1, לקוח-2, ריבית-א, ריבית-ב וכו'). יש להימנע משם זהה למשמעויות שונות (homonym).

מילון פריטי מידע יתכן במספר רמות:

- ברמת הפרויקט, היינו, מילון מקומי (פריטי)

- ברמת הארגון, היינו, מילון פריטי מידע מרכזי משותף לכל מערכות המידע בארגון.

הרמה הארגונית עדיפה, כמובן ויש לעשות כל מאמץ להשתמש במילון מרכזי, אם קיים כזה. במידה שאין, יש להקים מילון פריטי, אך יש לוודא שלפחות ברמת המערכת יהיה המילון משותף לכל הטרנזקציות, הקבצים וכו'. השילוב של חוסר מילון ברמת הארגון יחד עם ממשקים חיצוניים (המחייבים הגדרת פריטי-מידע) ברמת הפרויקט, הוא חמור ויש להתריע על כך לצוות ההיגוי של הפרויקט!

לרכיב מרכזי זה קשרים עם רכיבים מרכזיים אחרים. יש לציין זאת באופן ברור ולהימנע מכפילויות:

- טבלאות המערכת (רכיב 2.10)

- מילון המונחים (רכיב 2.1.4)

- במקרים מסוימים גם שגרות (רכיב 2.9).

יש להיזהר מכפילות, בפרט עם הגדרה מפורטת של קבצים (2.11 ו- 2.12).

יש להבחין בין פריטי מידע שהם אטום בסיסי, לבין פריטים בעלי מבנה מורכב הנחלק לתת-שדות (מולקולה). דוגמאות למולקולות: כתובת, תאריך ועוד. במיוחד יש לשים לב לפריטים מורכבים מאד כגון: פריט עובד, פריטים מזהים וכו'. תיתכן אפוא הגדרה רקורסיבית במילון בו פריט-מידע מסוים מוגדר בעזרת פריטי מידע אחרים.

יש לעשות שימוש מירבי אפשרי בכלי ממוכן, כמוגדר ברכיב 3.12 באפיון רכיב זה (ובעיצובו ובנייתו בהמשך). כלי כזה הוא הדרך הסבירה היחידה לקבל תמונה של where used ולזכור מי משתמש בפריט מידע והיכן. כלי זה ישמש גם רכיבים אחרים כגון: טבלאות, שגרות וכו'.

יחידות צה"ל (וכן בכל ארגון/גורם המקפיד על חיסוי המערכות וסיווגן הביטחוני או העסקי), יבדקו אילו פריטי מידע יש צורך לסווגם. לגבי אותם פריטים בהם יש צורך, יש גם לפרט מי הגורמים

הרשאים לגשת אליהם, תוך ציון לאילו צרכים, מטרות ומה הפעולה המותרת לכל גורם. ראה גם הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ.

### 2.13.0 אינדקס כללי

רשימה \ אינדקס של כל פריטי המידע והשדות שאינם פרטיים (מדרגת מקומי ומעלה). לכל שדה (שורה בטבלת האינדקס) המאפיינים הבאים:

- שם וסימול
  - סוג: מקומי (מערכת), ארגון, גלובלי
  - מבנה תמציתי (תווים, גודל/אורך, תווים מיוחדים)
  - הפנייה למילון נתונים (מערכת מתעדת)
- שיטה זו של ריכוז כל פריטי המידע בטבלה/אינדקס אחד, תוך הבחנה בין הסוגים, מתאימה למערכות קטנות ובינוניות. ככלל, מספר פריטי המידע המעניינים אינו עולה על 50. שיטה אחרת, המתאימה יותר למערכות בינוניות פלוס, מוצעת להלן.

### 2.13.1 שדות מקומיים

רשימה \ אינדקס של כל השדות המקומיים שנמצאים רק בממשקים הפנימיים בין המודולים, בתוך המערכת ובאחריותה. לכל שדה או שורה בטבלת האינדקס המאפיינים הבאים:

- שם פריט-המידע וסימולו
- שמות נרדפים
- הפניה לשמות דומים (לוגית ו/או פיסית)
- מבנה פנימי: אורך, סוג ייצוג, נומרי/אלפא-נומרי
- מרחב ערכים (domain) ובדיקות סבירות
- מקור/אחראי בארגון (או חיצוני לארגון).
- where used, קשר לרכיבים אחרים: 2.11, 2.12, 2.10 ועוד.
- הפנייה למילון נתונים (מערכת מתעדת)

### 2.13.2 שדות ארגוניים

רשימה \ אינדקס של כל השדות הארגוניים (משותפים עם מערכות אחרות בארגון). לכל שדה (שורה בטבלת האינדקס) המאפיינים כמפורט בסעיף 2.13.1 לעיל.

### 2.13.3 שדות גלובליים

רשימה \ אינדקס של כל השדות הגלובליים המשותפים עם ארגונים אחרים. לכל שדה (שורה בטבלת האינדקס) המאפיינים כמפורט בסעיף 2.13.1 לעיל.

## 2.15 דוחות (ושאילתות)

מטרת רכיב זה היא לרכז את כל דוחות ושאילתות המערכת ברכיב אחד ולפרט אותם. עם זאת, ניתן לפשט רכיב זה. בחלוקה פונקציונאלית נכונה של המערכת, מוגדרים דוחות כחלק מהגדרת תת-

מערכות וטרנזקציות (2.3 ו-2.6 לעיל) ולא כרכיב עצמאי. במקרים רבים דוחות הם למעשה שאילתות (מסכי פלט) הנשלחות ל-hard copy. בנוסף, אם בסיס הנתונים הוגדר נכון ואם קיים מחולל דוחות גמיש ונוח, אפשר להפיק כמעט כל דו"ח, לעתים קרובות ע"י המשתמש עצמו. מצב זה הוא בהחלט רצוי ויש לשאוף אליו. מכאן שחשיבות רכיב זה יורדת אם רכיבים פונקציונאליים דוגמת 2.3, 2.5 ו-2.6 מכילים שאילתות ודוחות כחלק אינטגרלי מהגדרתם ואם רכיבים טכניים דוגמת 2.12, 2.13, 3.11, ו-3.12 מאפשרים הפקת דוחות נוספים בקלות יחסית.

לעתים קרובות, ניתן גם להגדיר דוחות ברמה של **קבוצת דוחות**. קבוצת דוחות היא אוסף דוחות בעלי מכנה משותף של גישה לאותם קבצים (מקורות קלט), השונים זה מזה בפרמטרים משניים שקל יחסית לשנותם במחולל הדוחות, כגון:

- תת-קבוצות אוכלוסייה בקבצים
- חיתוכים
- רמות מיון וסכום
- כותרות.

מקורות קלט, היינו מקור הנתונים המופיעים בדו"ח, היא בדיקה חשובה המוודאת שאין בדו"ח המבוקש (בקבוצת הדוחות) דרישה לנתונים שכלל לא מוגדרים בבסיס הנתונים או בטרנזקציות הקלט והעדכון.

דו"ח הוא כל פלט סופי מהמחשב שאיננו ע"ג מדיה המאפשרת עיבוד נוסף. טפסים, תרשימים גרפיים וכו', גם הם בכלל דוחות. אין זו הגדרה מדויקת, אך הכוונה היא לדוחות במובן הפשוט והמקובל.

ביחידות צה"ל (וכן בכל ארגון/גורם המקפיד על חיסוי המערכות וסיווגן הביטחוני או העסקי), יש גם לפרט מי הגורמים הרשאים לגשת לאילו דוחות, תוך ציון לאילו צרכים, מטרות, תדירות וכו'. ראה גם הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. גופים הכפופים להנחיות רב"מ/במ"ת יפעלו בסעיף זה עפ"י הנחיות אלה.

## 2.15.0 אינדקס ורשימה כללית

רשימה \ טבלה של כל דוחות המערכת או השאילתות המייצרות אותם. לכל דו"ח (שורה בטבלה) המאפיינים הבאים:

- שם וסימול
- מען עיקרי
- קבוצה

**המשך פירוט הדוחות יכול להיעשות באחת משתי שיטות:**

- בקבוצות
- ישירות כל דו"ח, בלי הקבצה

### 2.15.X קבוצת דוחות X

#### 2.15.X.0 אינדקס

ראה פירוט בסעיף 2.15.0 לעיל.

## 2.15.X.N דו"ח N בקבוצה X

- שם הדו"ח וסימולו
- למי מיועד הדו"ח: משתמש עיקרי
- דרישות אבטחת מידע: סיווג, מידור והרשאות: מי רשאי לגשת ולצורך מה.
- מבנה פנימי: רמות/מפתחות מיון, שבירה (break) וסיכום
- מרחב ערכים (domain) ובדיקות סבירות
- מקורות (קלט, בסיס נתונים): קבצים
- תקופתיות ותדירות הפקה
- היקף, גודל
- אמצעי/מדיה (אם שונה מנייר).
- דוגמא לדו"ח מודפס.

## 2.15.X דו"ח X

ראה פירוט בסעיף 2.15.X.N

## 2.16 קלטים (טפסים)

במרבית המערכות המודרניות והממוחשבות רכיב זה מיותר. טפסים הם חלק ממסכי המערכת ומקומם ברכיב 2.4.2 לעיל. השימוש ברכיב זה יהיה, בכל זאת, כתוצאה מאחת הסיבות הבאות:

- מדובר בטפסים שהם מחוץ למערכת, היינו שממולאים ידנית ומוזנים בשלב מאוחר יותר למערכת.

- חיוני להציג במרכז את כל הקלטים למערכת (בדומה לדוחות לעיל)

במקרים בהם הוחלט להשתמש ברכיב זה, יהיה הפירוט שלו בדומה לרכיב 2.15 דוחות לעיל. תת רכיב 2.16.0 ישמש אינדקס של כל הקלטים ורכיבים 2.16.1 ואילך יפרטו את הטפסים בהקבצות המתאימות. קבוצת קלטים היא אוסף קלטים בעלי מכנה משותף של גישה לאותם קבצים (מקורות קלט), השונים זה מזה בפרמטרים משניים שקל יחסית לשנותם במחולל המסכים מבלי לשנות את בסיס הנתונים.

בחלוקה פונקציונאלית נכונה של המערכת מוגדרים הקלטים כחלק מהגדרת מסכים וטרנזקציות (2.4 ו-2.6 לעיל). במקרים רבים קלט או טופס הוא בעצם מסך וגם טופס מודפס (Hard Copy) אשר מופק ישירות ממערכת המסכים. רכיב זה (2.16) הוא אפוא רכיב אורתוגונלי (לא-פונקציונאלי) שתפקידו העיקרי הוא לרכז את כל הקלטים במקום אחד. מבחינת מפת"ח אין זה רכיב חובה ומידת ההקפדה על קיומו נתונה לשקול דעתו של הארגון (הפרויקט).

יעדים (destinations), היינו, הקבצים המתעדכנים בעקבות קלט זה, הוא מידע חשוב המאפשר חיתוכים ואימותים, חישוב עומסים, אבטחת מידע, עריכת בדיקות מערכת וכו'.

עבור כל טופס יפורטו הנתונים הבאים:

- שם הקלט וסימולו
- למי מיועד הקלט (מי אחראי למילוי)

- מבנה הטופס : שדות
- מרחב ערכים (domain) ובדיקות סבירות
- יעדים (קבצים בבסיס נתונים)
- תקופתיות ותדירות
- היקף, כמויות
- אמצעי/מדיה קצה (אם שונה מנייר).
- דוגמא לקלט (טופס) מודפס.

## 2.19 אבטחת מידע

במערכות ממשלתיות/צבאיות יש לוודא תחילה באמצעות הממונה על הביטחון בארגון (קצין הביטחון), אם המערכת כפופה להנחיות רב"מ/במ"ת. מומלץ גם לעיין תחילה בקיט אבטחת מידע בכרך נושאים תומכים.

ההחלטה עד כמה להשקיע בניתוח סיכונים (תת רכיב 2.19.1) או לגשת ישירות להגדרת האמצעי והדרישות (2.19.2), צריכה להתקבל בכל ארגון/פרויקט לעצמו. יחידות צה"ל יפעלו בסעיף זה, בנוסף להנחיות מפת"ח, גם לפי הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. ראה פירוט צה"לי של רכיב זה בקיט דמ"צ/צה"ל - תת שלב אפיון בכרך מחזור חיים.

ראה הרחבה לרכיב 2.19.1 ו- 2.19.2 בקיט עץ מערכת אוניברסלי.

באפיון רכיב זה יש לשים לב לא להיתפס, מחד גיסא, לאבטחת מידע מוגזמת. מימוש כל ההמלצות הוא בלתי מציאותי. מאידך גיסא, אין להיתפס לשאננות ולהזניח נושא חשוב זה, המעוגן בחוק ובתקנות. יש למצוא את שביל הזהב ולציין רק דרישות מציאותיות ורלוונטיות למערכת הנדונה. אם הדרישות ברכיב זה רציניות, כדאי לבדוק אם אין כאן גלישה לפרויקט תשתית של אבטחת מידע בארגון. אם כן, יש לפעול על פי הקיט תשתית אבטחת מידע שבכרך מערכות תשתית.

לרכיב אבטחת מידע קשרים חשובים ומורכבים עם רכיבים אחרים. קשרים אלה יכולים להיות משני סוגים:

- **רכיבים משלימים**, היינו, רכיבים שהדרישות שלהם ויישומם בפועל, מסייעים לאבטחת מידע, ולהפך. לדוגמא: רכיבים 2.2, 2.4, 2.10, 2.11, 2.12, 2.15, 2.22;
  - **רכיבים סותרים**, היינו, רכיבים שקיים trade-off בין הדרישות שלהם והדרישות לאבטחת מידע. לדוגמא: רכיבים 2.9, 2.10, 2.21.
- רכיב זה הוא בחלקו אדמיניסטרטיבי (נהלים, הוראות וכו'), היינו חיצוני למערכת ובחלקו ממומש במערכת עצמה (ביישום או בטכנולוגיה). המימוש יכול להיות ישיר או באמצעות רכיבים אחרים (אורתוגונליות רכיב זה). אפיון רכיב זה, עיצובו ובנייתו כוללים אפוא את ההתייחסויות הבאות:
- רכיבי אבטחה ישירים במערכת עצמה (הוספת משתמשים, שינוי סיסמאות, מבנה Log המערכת וכו').
  - אבטחה ע"י רכיבים קונקרטיים: במסכים, קבצים, טרנזקציות וכו'.
  - נהלי אבטחה המותקנים במקומות שונים, אך מרוכזים ברכיב זה (או בנספח 2.19 בתיק זה).

יחידות צה"ל יפעלו, בסעיף זה, עפ"י הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. גופים הכפופים להנחיות רב"מ/במ"מ יפעלו בסעיף זה עפ"י הנחיות אלה.

## 2.19.0 כללי – הבהקים

הגדרה כללית של רגישות המידע במערכת והסיווג הביטחוני/עסקי שלה.

### 2.19.1 סיכוני אבטחת מידע (במ"מ)

ניתוח סיכוני במ"מ מבוסס על מתן תשובה מדורגת לשאלות הבאות:

- מי הגורמים העשויים לנסות לפרוץ למערכת?
- איזה סוג פריצה/איום מעניין אותם: קריאה, הריסה, שינוי (עדכון)?
- באיזו תדירות?
- מה מידת הנזק הצפוי לכל צירוף גורם / סוג פריצה?
- מה יכולת השיפוי (תיקון, התאוששות) מנזק זה?
- אילו רכיבים במערכת מועמדים לסוג פריצה / תדירות / נזק / התאוששות?

### 2.19.2 אמצעי אבטחת מידע (במ"מ)

פירוט אמצעים (דרישות) לאבטחת מידע (אם אזכור תקן או תשתית בארגון אינם מספקים)

- זיהוי המשתמש, סיסמאות והרשאות
  - מידור הגישה למערכת
  - רישום כניסות ויציאות למערכת
  - ניהול רישום גישה לקבצים
- אמצעי אבטחת נוספים

### 2.19.3 ניהול האבטחה

אמצעים המיועדים למנהל האבטחה עצמו – System security administration, כגון:

- ניהול המשתמשים: הקצאת סיסמאות והרשאות
- ניהול רישום כניסות ויציאות למערכת
- הגדרת מסלולי מידור

## 2.20 הצלבות וחיתוכים

רכיב זה איננו רכיב פונקציונאלי (קונקרטי), אלא מסכם (אורתוגונלי). מטרתו לבדוק ולאמת רכיבים אחרים (פונקציונאליים) ולהציג את המערכת מנקודות מבט שונות. עם זאת, מתודולוגיות וכלי Case רבים מתייחסים לרכיב זה כאל רכיב מרכזי ביותר באפיון מערכת מידע, המסייע לא רק לאמת אלא אף להגדיר את הרכיבים הקונקרטיים. טכניקת המטריצות (או cross reference) מסייעת גם בבדיקות שלמות, איתור חסרים, ישויות לא מטופלות וכו'.

אפקטיביות רכיב זה תלויה מאד בשימוש בכלי ממוכן, כגון:

- כלי עצמאי ad-hoc למטרה זו
- כלי הקשור במילון הנתונים (רכיב 3.11)
- כלי CASE לפיתוח תוכנה (רכיב 3.13).

המטריצות (החיתוכים) הבסיסיים הם דו-ממדיים ומציגים ישות אחת מול ישות שנייה. ע"י הכפלת (הצלבת) המטריצות עצמן אפשר כמובן ליצור מטריצות רב-ממדיות (שלושה, ארבעה ממדים וכו'), או מטריצות דו-ממדיות שהן פשוט הכפלה של שתי מטריצות. למשל, ע"י הכפלת מטריצת משתמשים/טרנזקציות עם מטריצת טרנזקציות/קבצים, אפשר לקבל את המטריצה משתמשים/קבצים. חיתוכים אלה קשה בוודאי לבצע ללא כלי ממוכן.

מטריצות הצלבה אינן טכניקה אפשרית יחידה ברכיב זה. ייתכנו טכניקות וכלים אחרים כגון: מחזור חיים של ישות במערכת - Entity Life History.

דוגמאות למטריצות:

- משתמשים/פונקציות ראשיות (קשור ברכיב הקודם - 2.19)
- פונקציות ראשיות/טרנזקציות
- משתמשים/תהליכים
- תהליכים/קבצים לוגיים
- טרנזקציות/קבצים לוגיים
- קבצים/אפריטי מידע
- קבצים לוגיים/קבצים פיסיים
- תת מערכות/ממשקים

## 2.21 נפחים עומסים וביצועים

רכיב זה הוא רכיב מסכם (אורתוגונלי), במידה שהדרישות לנפחים וביצועים מוגדרות בברור ברכיבים קודמים (טרנזקציות, קבצים וכו'). אם דרישות אלה לא הוגדרו שם, יש להגדירן כאן ברמה של כלל המערכת. בכל מקרה, ברכיב זה תובא הצגה כוללת של דרישות המערכת לנפחים, ביצועים והמצב בפועל.

הערכת היקף המערכת חשובה ביותר בשלב האפיון. זהו מעין תכנון קיבולת (capacity planning) של משאבי המחשוב שהמערכת עתידה לצרוך. לקיבולת זו יש משמעות גם בהגדרה הלוגית של המערכת, לא רק בהגדרה הפיסית (בעיצוב), למשל לצורך חקר ישימות המערכת.

ביציאה למכרז, היינו, בעת הפיכת מסמך האפיון למפרט, רצוי לבנות מדדים לעומסים וביצועים בטבלאות ברורות ולמלאן בערכים כמותיים. אם קשה לתת מדד מוחלט, רצוי לציין לפחות **חסמים** - עליון ותחתון - נוסף (או כתחליף) להערכת המדד עצמו.

אם יש דרישה שהמערכת תמדוד את עצמה תוך כדי התפעול השוטף ותוציא סטטיסטיקות שוטפות - יש לציין זאת בבירור. דרישה כזו יכולה להקל מאד על תחזוקת רכיב זה ובעקבותיו על תחזוקת המערכת כולה ויכולת הרחבתה.

הפרמטרים בעזרתם נמדדים נפחי מערכת ועומסים צפויים הם:

**גדלי קבצים**



- הפרדה בין קבצי O/L (דיסקים) לקבצי ארכיב וגיבוי (קלטות).
- גודל הקובץ נמדד ביחידות הבאות: סה"כ Mb, מס' קבצים, מס' שדות ומפתחות, מס' רשומות, קצב גידול (שנתי) צפוי (באחוזים), קצב התחלפות המידע בקבצים (Volatility).
- המורכבות הכוללת של מודל הנתונים (הסכימה הגלובלית). מורכבות זו נמדדת במס' היחסים (קשרים) בין הקבצים (הישויות), למשל, לפי מודל ERD.

#### מס' תחנות עבודה

- סה"כ בחיבור למערכת (למחשב המרכזי או לשרתים)
- כללי, בו-זמני (מס' sessions הפתוחים בו זמנית)
- מזוהים (משתמשים), לא מזוהים (גולשים)
- בעבודה אקטיבית בו-זמנית (מס' הטונזקציות המנוהלות בו-זמנית)

#### טרנזקציות ליחידת זמן בזמני תגובה נדרשים

- טרנזקציות מסוג קלט, עדכון ושאלתה, XX% בזמן תגובה שלא יעלה על y שניות.
- מס' מסרים (Messages)

#### הפקת דוחות כבדים

כמות, עיתוי, זמני סבב נדרשים.

#### מהלכי Batch מיוחדים וניהול המערכת (Housekeeping).

- כמות, עיתוי
- זמני סבב נדרשים עבור: עדכונים מרכזיים, ניהול המערכת (Housekeeping).

## 2.22 ממשקים וקישורים

רכיב זה הוא רכיב מסכם (אורתוגונלי) בחלקו, שכן קשרים עם מערכות אחרות כבר מודגשים (או צריכים להיות) ברכיבים קודמים: 2.6 - טרנזקציות, 2.11 - קבצים וכו'. ריכוז כל הקשרים של מערכת זו עם מערכות אחרות - כולל קשר עם תת-מערכות שלה-עצמה, אם אופיינו בתיק נפרד - הוא מטרת רכיב זה.

כל קשר ו/או תלות במערכת אחרת, גם אם פורטו לעיל, יוצגו ברכיב זה. קשר עם מערכת מידע אחרת הוא באחד מהאופנים הבאים:

- העברת נתונים (File transfer)
- קשר מקוון (מסוף למערכת אחרת)
- קשרים מיוחדים הנובעים משימוש ברכיבים דומים/משותפים

הקשר הראשון, היינו העברות מידע, הוא החשוב ביותר. הקשר השני הוא טכני בעיקרו והשלישי הוא נדיר יחסית. ייתכנו גם קשרים משולבים. למשל, קשר מקוון לצורך הפעלה יזומה של העברת מידע.

במערכות רבות אשר באפיון ראשוני נראות סגורות ואוטונומיות מתברר יותר ויותר הצורך והתועלת שביצירת קשר עם מערכות אחרות. למשל, מערכות מקוונות, היינו מערכות הקולטות מידע ראשוני בעצמן, מגלות שמידע חשוב וזמין נמצא במערכות אחרות ואפשר לקלוט אותו ע"י העברת קבצים (File transfer) פשוטה יחסית. חשוב ביותר לברר נקודה זו במהלך האפיון!

שים לב לקשר ההדוק של רכיב זה עם רכיבים 2.2 – משתמשים ומערכות משיקות ו- 2.6 - טרנזקציות (קבוצת הטרנזקציות המטפלות בהעברות מידע). המנע מחזרות ומכפילויות. קשר חשוב ביותר של רכיב זה הוא עם רכיב 2.13.

ביחידות צה"ל (וכן בכל ארגון/גורם המקפיד על חיסוי המערכות וסיווגן הביטחוני או העסקי), יש גם להגדיר היטב את הסיווג של כל ממשק ואת מנגנוני אבטחת המידע בשכבות השונות של הממשק, כולל השלכות הנובעות מכך שהמערכת המתממשקת היא בעלת סיווג נמוך/גבוה מהמערכת הנדונה. ראה גם הנחיות צה"ל/מקשר"ר/מצו"ב/במ"מ. גופים הכפופים להנחיות רב"מ/במ"ת יפעלו בסעיף זה עפ"י הנחיות אלה.

## 2.22.0 אינדקס ורשימה כללית

- שם המערכת וסימולה
- סוג הקשר: העברת נתונים, קישור מקוון וכו'
- סוג המערכת: חיצונית, פנימית בארגון, תת-מערכת
- סיווג המערכת וסוגה
- הקבצה (קשר לתהליכים או מסכים, או קבצים)

## 2.22.X ממשק X

תיאור מפורט של ממשק X הכולל:

- שם וסימול לוגי ופיסי
- תיאור קצר
- סוג הממשק: העברת מידע, מסר (Messaging), בקשה/אישור לקבלת שירות, קישור ליישום אחר
- אחריות ומקור/יעד: יוצר/מקבל, דחיפה/משיכה
- אופן ההפעלה: אצווה, מקוון, Queuing
- שירות/פעולה נדרשים
- נתונים: תיאור המידע המועבר
- רשומת בקרה/פרוטוקול פנימי
- פרוטוקול למימוש הממשק
- תקן
- תדירות, מועדים
- דרישות אבטחת מידע ונתונים: בשכבות השונות של הממשק.
- שדות ומבנה מפורט (בדומה למבנה קובץ, ראה 2.11.3)
- הפניה מתאימה לסעיפים אחרים בהם נמצא תיאור משלים של:
  1. מסך תפעולי להפעלת הממשק (2.4)

2. טרנזקציה שולחת (2.6.4)

3. קובץ עיקרי (2.11.3).

## 2.23 דרישות מיוחדות

רכיב זה מיועד לציין כל דרישה/תכונה מיוחדת של המערכת ברמת היישום, שלא צוינה עד כאן. אפשרות נוספת היא שרכיב זה יציג סיכום של תכונות נדרשות מהמערכת שאמנם פורטו ברמת הרכיבים הקודמים, אך יש צורך/טעם להציגן במרוכז. במילים אחרות, **תכונות אורתוגונליות**.

דרישות מיוחדות כוללות את הנושאים הבאים:

- גמישות
- יבילות
- שפות

בפועל רכיב זה מתייחס בעיקר לגמישות המערכת ויכולת התאמתה לשינויים, היינו, למדדים המציינים את יכולת המערכת לעמוד בשינויי תפעול, בשינויי force majeure (שינויי חוק ותקנות) וכו'. **אין הכוונה כאן להרחבת המערכת** - זאת יש לעשות ע"י מהדורה חדשה - אלא ליכולת המערכת להתאים עצמה לדרישות הנובעות מהתפעול השוטף, לשינויי תחזוקה proper (לא שו"ש) ובעתיד גם למעבר למהדורה חדשה.

רכיב זה הוא **סופר-אורתוגונלי**, שכן גמישות ויכולת הרחבה (וכן דרישות מיוחדות אחרות - אם נרשמו ברכיב זה) הן תכונה של רכיבים קונקרטיים בעץ המערכת, כגון: קבצים, טבלאות, טרנזקציות, מודולים וכו'. לפיכך יש להיזהר ביותר מכפילות, או גרוע מזה, מאזכור דרישות כאן במקום אזכור ברכיב המתאים (הקונקרטי). רכיב זה ישמש נקודת ריכוז בלבד.

בכל מקרה, רצוי להגדיר מדדים ברורים לדרישות המיוחדות (לגמישות). האידיאלי הוא מדדים שהמערכת מפיקה תוך כדי פעולתה ושניתן לסכמם, כגון: מס' השינויים בתפריטי המשנה של המערכת, מס' השינויים בטרנזקציות, התפלגות השימוש בטרנזקציות ועוד.

גם באורתוגונליות וגם בדרישה להגדיר מדדים, דומה רכיב זה לרכיב נפחים וביצועים (רכיב 2.21 לעיל).

הנושא השני הנזכר כאן, יבילות (Portability) - נקרא לעתים גם Interoperability, הוא בוודאי אורתוגונלי על עץ המערכת כולו, והכוונה כאן ליכולת העברת נתונים וקוד על פני פלטפורמות שונות - מערכות הפעלה ותשתיות מחשוב.

הנושא השלישי הוא בעל ערך במערכות המפותחות עבור לקוחות בישראל - שפות: תמיכה בעברית יחד עם רב-לשוניות.

בהתאם לתכונות המערכת המאופיינת יתכנו דרישות יחודיות לגמישות.

## 2.98 נקודות פתוחות (וחלופות)

סעיף זה מטרתו לרכז את כל הנקודות שנשארו פתוחות ברכיב היישום. נקודות פתוחות הן לעתים קרובות חלופות (אלטרנטיבות) אפשריות ולפיכך גם החלופות השונות שתועדנה כאן. עם זאת, יש לבדוק היטב את שני המקרים הבאים:

- נקודות פתוחות (וחלופות) ברמת רכיב משני (2.X)

- חלופה מקיפה לרכיב היישום בכללותו.

שני מקרים אלה, שהם השכיחים יחסית, לא יתועדו למעשה ברכיב זה. נקודות פתוחות וחלופות ברמת רכיב משני - מקומן ברכיב המשני (2.X.98). חלופה מקיפה לרכיב היישום כולו - תתואר ע"י **תיעוד חוזר ומלא** של רכיב 2 כולו (2.0 - 2.23).

רכיב זה מיועד אם כך, למקרי ביניים, היינו, מקרים בהם החלופות והנקודות הפתוחות הן כלליות ואינן משנות את תמונת היישום הכללית, או שעדיף לבחון אותן ברמה זו כיוון שהן משפיעות רק על קבוצה חלקית של רכיבי היישום (2.1 - 2.23).

להרחבה נוספת ולהסבר מקיף בנושא חלופות, ראה קיט ניתוח חלופות בכרך נושאים תומכים.

בכל מקרה, חיוני לסגור סופית את הנקודות הפתוחות ולהכריע בין החלופות השונות ברכיב היישום, אחרת אפיון המערכת כולו תלוי על בלימה. חשוב ביותר להשאיר בתיק תיעוד גם של נקודות שנסגרו וחלופות שנדחו, על מנת לסייע בדיונים מחודשים בעתיד.

יש לפרט ברשימה כוללת ותמציתית את רשימת הנקודות הפתוחות:

- הנקודה-הפתוחה/חלופה: שם או זיהוי אחר
- תאריך פתיחה
- תאריך סגירה

ולתאר במפורט עבור כל נקודה-פתוחה/חלופה:

- תיאור הנקודה-הפתוחה/חלופה
- תאריך פתיחה
- תאריך סגירה
- קשר לרכיבי יישום (ורכיבים אחרים בעץ המערכת)
- הגורם שסגר
- הפניה לסיכום דיון.

## 2.99 דרישות עתידיות

סעיף זה מטרתו לרכז את כל הדרישות העתידיות ברכיב היישום שאינן כלולות במהדורה הנדונה, אך סביר שיכללו במהדורות עתידיות. רישום דרישות (תחזיות) עתידיות אלה, הוא במטרה כפולה:

- לא לאבד מידע קיים
  - להדגיש מה **לא** כלול במהדורה הנדונה ולסייע בכך לניהול תצורה של המערכת.
- מיקומן של דרישות עתידיות אלה ברכיב ראשי (רמה שניה) של עץ המערכת, הוא מכוון. אם יש מספיק מידע וצורך לרשום דרישות אלה ברמה מפורטת יותר, היינו, ברכיבי יישום משניים (2.x), יש לבדוק היטב אם אין לפנינו הגדרה מלאה של המהדורה הבאה של המערכת. הגדרה כזו פירושה **תיעוד מלא** של רכיב 2 כולו ומיקומה בתיק אפיון נפרד (של המהדורה הבאה).

רכיב זה משמש בסיס לרכיבי (X.99) מקבילים בטכנולוגיה, במימוש ובעלות. מצד שני, רכיב זה איננו עומד בפני עצמו והוא מושפע מהרכיב המקביל ביעדים (1.99). במקביל אליו יש להגדיר את הטכנולוגיה (3.99), המימוש (4.99) והעלות (5.99) המשלימים רכיב זה.

יש להציג רשימה כוללת של תחזית עתידית: שם או זיהוי אחר. לכל סעיף יש לציין:

- תיאור הדרישה או הפניה לתיעוד אחר (תכנון אסטרטגי למשל)
- קשר לרכיבי X.99 אחרים
- הפניה לסיכום דיון.

### 3. טכנולוגיה ותשתית

פרק הטכנולוגיה נחלק לשלוש קבוצות, באופן הבא:

1. **מחשבים מרכזיים:** רכיבים 3.1-3.19 (לא כולם בשימוש) – תיאור המחשבים המרכזיים התומכים במערכת, חומרה, תוכנה וכלים, כולל שרתים קדמיים (Front-end servers), שרתי קבצים (Backend servers), מחשבים מרכזיים (מחשבי הארגון) שהם חלק מהמערכת ועוד.
2. **מחשבי קצה:** רכיבים 3.20-3.29 (לא כולם בשימוש) – תיאור מחשבי הקצה (Clients) השונים איתם ניתן לעבוד מול המערכת.
3. **רשתות תקשורת:** רכיבים 3.30-3.32 (לא כולם בשימוש) – תיאור רשת(ות) התקשורת, (Communications)

בהקשר עם **קבוצה מס' 1**, במקרה של ריבוי סוגי מחשבים מרכזיים מומלץ לנהוג באחת משתי האפשרויות הבאות:

- חלוקה לפי סוג משאב (האפשרות המומלצת): השתמש בחלוקת הרכיבים כפי שהיא מופיעה להלן: 3.1 חומרה, 3.2 אחסנת נתונים, 3.10 מערכת הפעלה וכו'. אם יש מספר שרתים, **בתוך** כל רכיב, יש לפרט לפי סוגי השרת: 3.1.1 חומרה שרת א', 3.1.2 חומרה שרת ב', 3.10.1 מערכת הפעלה בשרת א', 3.11.2 בסיס נתונים בשרת ב' וכו'.
- חלוקה לפי תצורות: רכיבים 3.1-3.19 יחולקו לפי תצורות גנריות (סוגי מחשבים/שרתים): 3.1 מחשב א, 3.2 מחשב ב, 3.3 מחשב ג וכו'. **בתוך** כל סוג מחשב/שרת תהיה חלוקה פנימית בהתאם לסוג המשאב, היינו: 3.1.1 חומרה שרת א, 3.1.2 אחסנת נתונים שרת א, 3.2.10 מערכת הפעלה שרת ב וכו'.

לגבי **קבוצה מס' 2**, מחשבי הקצה (3.20-3.29), מומלץ מאד להיצמד לחלוקה שלהלן: 3.20 חומרה, 3.21 תוכנות תשתית וכו'. במקרה מיוחד בו יש יותר מסוג אחד של מחשב לקוח (קצה), ניתן לבחור גם באפשרות 2 לעיל לגבי מחשבי השרתים.

### 3.0 ארכיטקטורה כללית – הבהקים

אפשר, בשלב ביניים, לכלול ברכיב זה (בדומה לרכיב 2.0 לעיל), נושאים ודגשים כלליים של הטכנולוגיה אשר חשוב לזכרם ומקומם הסופי לא ברור. במהלך בניית התיק יועברו נושאים אלה למקומם הטבעי באחד הרכיבים להלן.

בצורתו הסופית, יכיל רכיב זה אך ורק עקרונות יסוד, תפיסה כללית וכו', נושאים, שחשוב להדגישם ברמת הבהקים (Highlights), לפי הדוגמא להלן. בשום פנים אין להפוך רכיב זה לתיאור כללי של הטכנולוגיה או גרוע יותר, של המערכת כולה! לכל מידע המופיע כאן יש מקום טבעי וברור ברכיבים להלן.

טכנולוגיה במערכות מידע היא בד"כ טכנולוגית מדף. לכן אפשר לפשט מאד את תיעוד רכיב זה, ע"י הפניה לתיעוד קיים כגון: נהלי סיוע טכני, manuals טכניים ועוד. הערה זו נכונה בפרט למערכת מידע הנשענת על טכנולוגיה קיימת בארגון. במערכת מידע הנשענת על טכנולוגיה ייעודית וייחודית משלה, יש להקפיד על אפיון מובן וברור באמצעות פירוט מלא בתיק זה או ע"י הפניה לתיעוד טכני מקובל וידוע (תקינה!).

חשוב להדגיש את יחסי הגומלין הצפויים (והקיימים?) בין היישום לבין הטכנולוגיה. מי מוביל את מי? האם המערכת היא ביסודה פתרון טכנולוגי ליישום ברור (חבילת תוכנה?), האם היישום יסתדר עם טכנולוגיה קיימת? האם היישום יכול להכתיב טכנולוגיה (חומרה ותוכנה) המתאימה לו?

הצג תרשים ארכיטקטורה ורכיבי הטכנולוגיה המרכזיים של המערכת: מחשבים, פריסה ורשת. הוסף הסבר במלל של התרשים המפרט את עיקרי הטכנולוגיה שבשימוש המערכת. תוכן תרשים זה ומידת מורכבותו עשויים להשפיע על בחירת השיטה לפיה יתועדו שאר רכיבי הטכנולוגיה (ראה אפשרויות 1 ו-2 לעיל). זכור שסעיף זה יופיע בד"כ בתמצית מנהלים שתופק מתיק זה. ראה קיט כתיבת תיעוד בכרך נושאים תומכים.

### דוגמא: (בלבד!)

1. המערכת המוצעת היא המשך של מערכת קיימת. יש להתחשב בהשקעות בטכנולוגיה שבמערכת הקיימת, בפרט ציוד קצה גרפי וציוד לאחסנת נתונים. ראה פירוט ברכיבים 3.2 ו-3.3 להלן.

2. טכנולוגיה חלופית שתוצע חייבת להציג:

- שמירה על ההשקעה הקיימת
- שיטות הסבה פשוטות

• שמירה על ההשקעה החדשה לטווח של 5 שנים.

3. הטכנולוגיה שתוצע חייבת להתאים לתקינה בינ"ל וישראלית, בפרט בתחומים הבאים:

- תקשורת מקומית, ראה רכיב 3.30 להלן בפרק זה.
- תקשורת רחבה, ראה רכיב 3.31 להלן בפרק זה.

4. הטכנולוגיה שתוצע חייבת להתחשב ביותר בטכנולוגיות שכנות בארגון אתן תהיה לה אינטראקציה גבוהה. ראה בפירוט רכיב 3.33 להלן.

### סוף דוגמא

## 3.1 חומרה מרכזית

רכיב זה מיועד לתאר את המחשוב המרכזי (ה-server) סביבו פועלת המערכת, החל מ-PC בודד (אם זה המקרה) וכלה ב-Mainframe, או אוסף מחשבי שרת ו/או מחשבי מיני, בכל קונפיגורציה אפשרית. PC או תחנות עבודה אישיים **בנוסף** לחומרה המרכזית המתוארת כאן, מקומם ברכיב 3.3 (ראה הנחיות שם). המונח חומרה מרכזית כוונתו לחומרה העיקרית סביבה תפעל המערכת ולא שום כוונה לגישה ריכוזית לעומת מערכת מבוזרת. גם במערכות מבוזרות יש תמיד מרכז; גם אם הוא מורכב ממספר מחשבים שווי ערך (peers) המחלקים את העבודה ביניהם באופן דמוקרטי לחלוטין.

ציין מערכות אחרות, בארגון ומחוצה לו, הפועלות על חומרה זו.

אם יש דרישה מראש לציוד מסוים או **לסוג ציוד** מסוים, ציין זאת בברור כולל התייחסות לציוד חלופי (תואם). נמק מדוע דווקא ציוד זה.

במקרים בהם המערכת הנדרשת חייבת להשתמש בציוד מרכזי קיים, תאר ציוד זה בקצרה. אפשר גם להפנות לתיעוד היצרן.

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

במשרדי ממשלה יש להיעזר במערכת מצאי ענ"א הממשלתית (צמר"ת). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

הגדרת החומרה של כל השרתים התומכים במערכת, כולל מחשבים מרכזיים, מחשבי Backend ומחשבים מרוחקים:

- מעבד
- דגם
- זיכרון, כולל מטמון
- אמצעים נתמכים
- מודולריות ואפשרויות הרחבה

לתקנים בנושא זה, כולל תקני בטיחות (safety), ראה קיט תקינה ותקנים בכרך נושאים תומכים

### 3.2 אחסנת נתונים מרכזית

חשוב להפריד בין סוגי אחסנה שונים:

- אחסנה פעילה ושוטפת (Online)
- אחסנה לשם שמירה/ארכיב/גיבוי
- אחסנה במדיום המיועד להעברות מידע (קלטות, דיסקים נתיקים).

ציין מערכות אחרות - בארגון ומחוצה לו - הפועלות על ציוד דומה.

אם יש דרישה מראש לציוד מסוים או **לסוג ציוד** מסוים, ציין זאת בברור, כולל התייחסות לציוד חלופי (תואם). נמק מדוע דווקא ציוד זה.

במקרים בהם המערכת הנדרשת חייבת להשתמש בציוד אחסנה קיים, תאר ציוד זה בקצרה. אפשר גם להפנות לתיעוד היצרן.

וודא שהדרישות ברכיב זה תואמות את הדרישות ברכיב 2.21.

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה (בכ"א מהסעיפים להלן).

במשרדי ממשלה יש להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

הגדרת אחסנת הנתונים המרכזית של המערכת:

- אחסנת נתונים RAID – Online (דיסקים)
- אחסנה נתיקה (קלטות, דיסקים נתיקים, CD's/DVD's)
- אמצעי גיבוי והעברה
- אחסנה לטווח ארוך (ארכיב)

לתקנים בנושא זה, כולל תקני בטיחות (safety), ראה קיט תקינה ותקנים שבכרך נושאים תומכים.



### 3.3 ציוד קצה

הכוונה כאן לציוד קצה הנלווה לחומרה המרכזית, לא לתחנות הקצה שמקומן בסעיפים 3.20-3.29 להלן. חשוב להפריד בין ציוד קצה תקני ושווה לכל נפש, לבין ציוד קצה מיוחד כמו מדפסת צבע, צג גרפי וכו'. ציין מערכות אחרות - בארגון ומחוצה לו - הפועלות על חומרה זו.

אם יש דרישה מראש לציוד מסוים או **לסוג ציוד** מסוים, ציין זאת בברור כולל התייחסות לציוד חלופי (תואם). נמק מדוע דווקא ציוד זה.

שים לב שתחנות עבודה (כולל PC) יכולות להופיע ברכיב זה או ברכיב 3.1, לעיל. הדבר מותנה בגודל תחנת העבודה ומקומה במערכת. אם היא החומרה המרכזית, או חלק ממנה, מקומה ב- 3.1. אם תחנת קצה לשימוש אישי המחוברת למחשב מרכזי, מקומה כאן, ב- 3.3. במערכות שרת/לקוח, יתועדו (ו) השרתים) ב- 3.1 והלקוחות כאן.

ההתייחסות לתחנות-עבודה ברכיב 3.1 או כאן, עשויה להשתנות בעקבות שינויים טכנולוגיים תכופים. כך או כך, לא לשכוח רכיב חשוב זה ואם צריך, להפנות מרכיב לרכיב.

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

במשרדי ממשלה: יש להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים: בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

#### צגים - תכונות בסיסיות נדרשות:

- גודל מסך (אלכסון)
- רזולוציה וגודל נקודה
- כמויות
- מיקום ופריסה בארגון.

#### מדפסות קצה - תכונות בסיסיות נדרשות:

- טכניקת הדפסה בסיסית
- מהירות
- רוחב נייר
- סוג נייר
- תמיכה בפונטים וגדלי אותיות
- תמיכה בגרפיקה
- עמידה בתקן postscript
- רזולוציה
- תכונות נוספות אופציונאליות (צבע, דופלקס).

### 3.4 ציוד מיוחד

ציוד מיוחד המחובר לחומרה המרכזית (לא למחשב הקצה):

- קורא אופטי (OCR)

- סורק (Scanner)
- מדפסת/קורא בר-קוד וכו'
- קורא כרטיסים מגנטיים
- חיבור למכשור זמן אמת אחר.

ציין מערכות אחרות - בארגון ומחוצה לו - הפועלות על חומרה זו.

אם יש דרישה מראש לציוד מסוים או ל**סוג ציוד** מסוים, ציין זאת בברור כולל התייחסות לציוד חלופי (תואם). נמק מדוע דווקא ציוד זה.

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

במשרדי ממשלה יש להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

### 3.5 ציוד מתכלה

ציוד מתכלה כגון:

- קלטות וסרטים מגנטיים
- נייר (רגיל, מיוחד)
- ציוד מתכלה הקשור למדפסות
- טפסים מיוחדים (לקלט ולפלט)
- עזרים למשלוח

השגח לא לערב רכיב זה עם רכיב 3.9 תשתית סביבתית להלן.

בדוק באיזו מידה ציוד מתכלה המיועד למערכת זו, תואם גם מערכות אחרות כך שאפשר להחזיק מלאי מרכזי ולהחליף ביניהם. האם יידרש שטח-אחסון מיוחד לציוד זה או שיש אחסון מרכזי בארגון?

חובה לציין כמויות לתקופה של 5 שנים ברכיב זה.

### 3.9 תשתית סביבתית

ברכיב זה, חשוב להגדיר בברור אם יש למערכת דרישות תשתית פיזית מיוחדות או שהיא נשענת על תשתית כללית בארגון. אם רכיבי חומרה שהוזכרו לעיל - בעיקר 3.1 ו- 3.2 - אינם עצמאיים ונשענים על חומרה קיימת בארגון, סביר שכך גם בתשתית הפיזית/סביבתית.

לעתים יש למערכת **ציוד ייעודי שאינו ייחודי**, היינו, ציוד עצמאי המיועד למערכת זו בלבד, אך תקני ומקובל בארגון (במערכות אחרות). במקרה זה, סביר שיש למערכת גם תשתית ייעודית שיש להגדירה, אלא שההגדרה היא יחסית פשוטה.

הגדר בברור באחריות מי תהיה אחזקת התשתית: באחריות יחידת המחשוב, המערכת עצמה, או גורם אחר בארגון (מי הוא)?

חשוב לבדוק התאמת הדרישות לתשתית לשינויים צפויים ברכיבים אחרים (ברכיבים האחרים נדרש לציין תחזיות ומגמות לתקופה של 5 שנים). בכל מקרה, רכיב התשתית יוגדר כך שההשקעה בו תישמר לתקופה של 10 שנים לפחות.

לתייעוד רכיב זה היעזר בהרחבה לרכיב 3.9 תשתית סביבתית בקיט עץ מערכת אוניברסלי ובקיט תשתית פיזית – חדר מחשב בכרך התמחויות/מערכות תשתית.

אם יש סחף ברכיב זה והוא תופח, בדוק היטב אם אין לפנינו מערכת תשתית. ראה הכרך מערכות תשתית.

רכיב משנה מיוחד, 3.9.3, מיועד לריכוז נושא הבטיחות, למערכות שאינן מסתפקות באזכור דרישות אלה ברכיבי 3.X המתאימים.

### 3.9.1 אתר ראשי

- מבנה: חדר מחשב, מרכז תקשורת
- כוח ומיזוג אויר
- כבלים ותשתית תקשורת
- אל-פסק - UPS
- אמצעי גיבוי
- הגנה פיזית

### 3.9.2 אתר גיבוי

יש להיעזר בתת רכיב זה במקרים בהם אמצעי גיבוי שבסעיף 3.9.1 לעיל אינם מספקים והמערכת דורשת אתר גיבוי מלא. עם זאת, יש לזכור שברוב המקרים הפתרון לדרישת תשתית נרחבת כזו הוא ברמה של הארגון כולו ולא של המערכת הבודדת. סעיף זה יפרט איפוא רק את הדרישות הנוספות של המערכת מאתר הגיבוי המרכזי של הארגון.

### 3.9.3 דרישות בטיחות (SAFETY)

אפשר לרכז כאן את כל הדרישות לבטיחות (safety) כולל תקני בטיחות של המערכת (במקום לפזר מידע זה בכל אחד מהרכיבים).

## 3.10 מערכת הפעלה

בהגדרת הדרישות לרכיב זה יש לציין, בד"כ, תקן או תיאור כללי ביותר. ראה הרחבה לרכיב 3.10 מערכת הפעלה בקיט עץ מערכת אוניברסלי.

במקרים בהם מערכת המידע הנדונה תפעל בסביבת מערכת הפעלה קיימת, ציין את שם מערכת ההפעלה והגרסה המדויקת. אפשר גם להפנות לתייעוד היצרן ולנוהלי הסיוע הטכני בארגון. שים לב למקרים בהם ידועות (מוכרזות) כבר גרסאות עתידיות של מערכת ההפעלה.

התייחס לקיומה או אי קיומה של פונקציות סיוע טכני (System) בארגון, ומידת הצורך במעורבותה במערכת מידע זו (הקפד לרשום זאת גם ברכיב 4.1).

אפשר להיעזר במערכת מצאי לבדיקת נפישות מערכת ההפעלה וניסיונם של ארגונים אחרים אתה. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

תיאור כללי של מערכת ההפעלה

אם מערכת הפעלה קיימת:

- שם וזיהוי, כולל גרסה מדויקת
  - יצרן/ספק
  - אחראי לעדכונים ותחזוקה.
- אם מערכת הפעלה נדרשת:

- תקן
- דרישות כלליות ביותר.

מערכת הפעלה חלופית: יש דרישה? מספיק שהמערכת היא פתוחה ותקנית?

תמיכה, שירות ותחזוקה

- תמיכת סיוע טכני נדרשת
- שירות ותחזוקה חיצוניים?

פירוט נוסף (אם צריך!): ראה הרחבה לרכיב 3.10 מערכת הפעלה בקיט עץ מערכת אוניברסלי.

הדגמות והפעלה ניסיונית (אם צריך!)

### 3.11 בסיס הנתונים - DBMS

רכיב 3.11 מטפל בהיבטים הפיזיים של בסיס הנתונים:

- ניהול בסיס הנתונים
- ניהול מילון הנתונים (Meta Data)

#### ניהול בסיס הנתונים

מקטע זה מוקדש למערכת לניהול בסיס הנתונים - Data Base Management System עבור כל סוגי בסיסי הנתונים והתוכנות לניהול סוגי המידע השונים, לפי סוגי המחשבים המרכזיים והשרתים.

בהגדרת הדרישות לרכיב זה אפשר, לעתים קרובות, להסתפק בציון תקן או בתיאור כללי ביותר של שיטת (מודל) בסיס הנתונים (או שיטת ארגון הקבצים) הנדרש. לדוגמא: בסיס נתונים טבלאי בתמיכה של שפת SQL תקנית, בסיס נתונים רשתי בתקן Codasyl וכו'.

במקרים בהם מערכת המידע הנדונה תפעל תחת בסיס נתונים קיים, ציין את שם בסיס הנתונים והגרסה המדויקת. אפשר גם להפנות לתיעוד היצרן ולנוהלי הסיוע הטכני בארגון. שים לב למקרים בהם ידועות (מוכרזות) כבר גרסאות עתידיות של בסיס הנתונים.

אם מתבקש פירוט מלא, היעזר בהרחבה לרכיב 3.11 בסיס נתונים בקיט עץ מערכת אוניברסלי.

התייחס לקיומה או אי קיומה של פונקצית DBA (Data Base Administrator) ומידת הצורך במעורבותה זו במערכת מידע זו (הקפד לרשום זאת גם ברכיב 4.1).

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

אפשר להיעזר במערכת מצאי לבדיקת נפיצות בסיס הנתונים וניסיונם ארגונים אחרים אתו. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

תיאור כללי של בסיס הנתונים

אם בסיס נתונים קיים :

- שם וזיהוי, כולל גרסה מדויקת
- יצרן/ספק
- אחראי לעדכונים ותחזוקה.

אם בסיס נתונים נדרש :

- תקן
- המודל הבסיסי: רשתי, היררכי, טבלאי, אובייקטים וכו'.

תמיכה, שירות ותחזוקה

- תמיכת סיוע טכני נדרשת.
- שירות ותחזוקה חיצוניים?

פירוט נוסף - תכונות בסיס הנתונים (אם צריך!): ראה הרחבה לרכיב 3.11 בסיס נתונים בקיט עץ מערכת אוניברסלי

הדגמות והפעלה ניסיונית (אם צריך!)

### ניהול מילון הנתונים (META DATA)

מילון-נתונים משמעו **כלי ממוכן מרכזי** המיועד לתעד את כל ישויות המערכת (משתמשים, טרנזקציות, מודולים, מסכים, קבצים, פריטי-מידע) ואת הקשרים ביניהן. בבסיסי נתונים מודרניים פונקציה זו מוכללת בתוך ה-DBMS. ההסבר להלן, מיועד למערכות ייחודיות לניהול נתונים.

בהגדרת הדרישות לרכיב זה אפשר, לעתים קרובות, להסתפק בציון תקן או בתיאור כללי ביותר של שיטת (מודל) מילון הנתונים הנדרש. במקרים בהם מערכת המידע הנדונה תפעל תחת מילון נתונים קיים, ציין את שם מילון הנתונים והגרסה המדויקת. אפשר גם להפנות לתיעוד היצרן ולנוהלי הסיוע הטכני בארגון. שים לב למקרים בהם ידועות (מוכרזות) כבר גרסאות עתידיות של מילון הנתונים.

אם מתבקש פירוט מלא, היעזר בהרחבה לרכיב 3.11 בסיס נתונים בקיט עץ מערכת אוניברסלי.

מעצם מרכזיות כלי זה, נובע שחשוב מאד לוודא אילו כלים אחרים יודעים לפעול בשילוב אתו. ראה רכיב 3.13 הסמוך. קביעה זו חשובה לשני סוגי המילון, הן ל- data dictionary לצורך שילוב כלים בשלבי העיצוב והבנייה והתחזוקה והן ל- repository לצורך שילוב כלים לשלב האפיון. אם בשלב האפיון עצמו כבר נעשה שימוש במילון נתונים מסוג repository (אנציקלופדיה), חשוב לציין:

- שם וגרסה מדויקת

- דרישות למעבר המידע ממילון זה הלאה.

התייחס לקיומה או אי קיומה של פונקציית (Data Administrator) DA בארגון, ומידת הצורך במעורבותה במערכת מידע זו (הקפד לרשום זאת גם ברכיב 4.1).

שים לב לדרישה (הרצויה!) שהמילון יסייע בניהול תיק האפיון עצמו (והתיעודים שיבואו אחריו: תיק עיצוב, תיק תחזוקה ואחרים).

אפשר להיעזר במערכת מצאי לבדיקת נפיצות מילון הנתונים וניסיונם של ארגונים אחרים אתו. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

מילון נתונים data dictionary (קיים בארגון, או נדרש)

- שם וזיהוי, כולל גרסה מדויקת
- יצרן/ספק
- אחראי לעדכונים ותחזוקה
- קשר לכלי פיתוח ותחזוקה (ברכיב 3.13 להלן)
- על איזה בסיס נתונים בנוי
- ישויות מקוטלגות.

מילון נתונים מסוג repository (קיים בארגון, או נדרש)

- שם וזיהוי, כולל גרסה מדויקת
- יצרן/ספק
- אחראי לעדכונים ותחזוקה
- קשר לכלי פיתוח ותחזוקה (ברכיב 3.13 להלן)
- על איזה בסיס נתונים בנוי
- ישויות מקוטלגות
- איך עובר המידע הלאה.

תמיכה, שירות ותחזוקה

- תמיכת סיוע טכני נדרשת
- שירות ותחזוקה חיצוניים?

הדגמות והפעלה ניסיונית (אם צריך!)

### 3.13 כלי פיתוח ותחזוקה

רכיב זה יציין טכניקות וכלים (CASE) הנדרשים לפיתוח המערכת ותחזוקתה. יש לזכור שמפת"ח הוא נוהל מסגרת אשר איננו מכתב שימוש בכלים (טכניקות ומתודולוגיות) מסוימים, אלא מעודד שימוש בטכניקות וכלים קיימים. זאת, בתנאי שהם יודעים לדבר בשפת מפת"ח ומשלימים אותו.

בדומה לרכיב 3.11 הקודם, יש להתייחס הן לכלים וטכניקות קדמיים (Uppercase) והן לכלים אחוריים (Lowercase).

דוגמאות לכלים וטכניקות קדמיים (Uppercase):

- כלים גרפיים לשרטוט דיאגרמות
- ניתוחים מטריציוניים,
- כלים לעיבוד תמלילים, תיעוד ו- Hypertext.

דוגמאות לכלים אחוריים (Lowercase):

- מחולל יישומים (שפת פיתוח GL4, למשל)
- ניהול ספריות, Editor.
- עריכת בדיקות מערכת (טסטים)
- ניהול תצורה.

סביר שבשלב האפיון עצמו כבר נעשה שימוש בכלי CASE (Uppercase). במקרה זה חשוב לציין:

- את הכלים המדויקים בעזרתם אופיינה המערכת.
- הדרישות למעבר מכלים אלה הלאה לכלים האחוריים.

במקרים בהם מערכת המידע הנדונה תפעל עם כלי CASE קיימים, ציין את שמות כלים אלה כולל גרסה מדויקת. אפשר גם להפנות לתיעוד היצרן ולנוהלי הסיוע הטכני בארגון. שים לב למקרים בהם ידועות (מוכרזות) כבר גרסאות עתידיות של כלי ה-CASE.

לפירוט נוסף, היעזר בהרחבה לרכיב 3.13 כלים לפיתוח ותחזוקה בקיט עץ מערכת אוניברסלי. שימוש מלא בהרחבה זו הוא רק עבור מקרים מיוחדים בהם מערכת המידע הנדונה חייבת לבוא עם כלים מלאים משלה, אם בגלל שהמצב הקיים בארגון איננו משביע רצון, אם בגלל שלמערכת יש באמת דרישות מיוחדות. עם זאת, שים לב לא לגלוש לפרויקט תשתית. אם המצב בארגון מחייב טיפול יסודי בתשתית כלי הפיתוח והתחזוקה, יש לפעול על פי הפרק המתאים בכרך תשתיות בנוהל.

אפשר להיעזר במערכת מצאי לבדיקת נפיצות כלים לפיתוח ותחזוקה וניסיונם של ארגונים אחרים אתם. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

התייחס לקיומן או אי קיומן של פונקציות אבטחת איכות או הנדסת תוכנה בארגון ומידת הצורך במעורבותן במערכת זו (הקפד לרשום זאת גם ברכיב 4.1).

פירוט כלי ה-Case יכלול:

כלים לשלבים הקדמיים (UpperCASE) **בפרט** כלים בעזרתם אופיינה המערכת.

1. דרישות לכלים לשלבים הבאים (LowerCASE)

- לפי שלבי מחזור החיים: בקשה-להצעות, עיצוב ובניה וכו'
- לפי רכיבי עץ המערכת (בעיקר תת-רכיבים ברכיב 2-יישום)
- כלים כלליים כגון: ניהול תצורה, אבטחת איכות, אמידת עלויות.

2. פירוט כלים וטכניקות ידועים וקיימים בארגון:

- שם וזיהוי, כולל גרסה מדויקת
- יצרן/ספק
- אחראי לעדכונים ותחזוקה
- על איזה חומרה פועל.

### 3.14 תוכנות מדף

סעיף זה יפרט את כל תוכנות המדף (תוכנות צד שלישי), הן תוכנות תשתית והן תוכנות יישום, הנדרשות במחשבים המרכזיים ובשרתים, להוציא מערכות הפעלה, בסיסי נתונים, כלי פיתוח וכלי תפעול וייצור שכבר פורטו בסעיפים 3.10-3.13 לעיל.

#### 3.14.1 תוכנות שירות

הכוונה כאן היא לתוכנות כגון:

- מנועי חיפוש
- ניהול הודעות ומסרים
- אבטחת מידע: למשל, הצפנת מידע בין שרת למשתמש (לתאם עם הדרישות שברכיב 2.19.2. לאילו דרישות נותנת תוכנת השירות מענה).
- תוכנה לכיווץ נתונים
- תוכנה לניהול מסכי העזרה - Help
- קישוריות בין שרתים

#### 3.14.2 תוכנות יישום

### 3.15 כלי תפעול וייצור

רכיב זה שייך לקבוצת הרכיבים המגדירה כלים ממוכנים לפיתוח מערכת מידע ותחזוקה (3.13), אך בשל חשיבות הנושא מוקדש לכלים המסייעים ישירות לניהול **תפעול שוטף** של המערכת והייצור רכיב מיוחד.

חשוב מאד להתייחס לעובדת קיומו או אי קיומו בארגון של מדור הפעלה וייצור המופקד על תפעול כלל המערכות ומידת מעורבותו במערכת זו (היינו, יכולתו להקדיש משאבים לשם כך). ראה קיט ארגון יחידת המחשוב בכרך ניהול כולל.

במקרים בהם נדרש שהמערכת הנדונה תשתמש בכלי התפעול והייצור הקיימים בארגון, ציין כלים אלה בתמצית, כולל גרסה מדויקת. אפשר גם להפנות לתיעוד היצרן ולנוהלי התפעול בארגון. שים לב למקרים בהם ידועות (מוכרזות) כבר גרסאות עתידיות של כלי תפעול וייצור אלה.

אם מתבקש פירוט נוסף, היעזר בהרחבה לרכיב 3.15 כלי תפעול וייצור בקיט עץ מערכת אוניברסלי.

אפשר להיעזר במערכת מצאי לבדיקת נפיצות כלים מסוג זה וניסיונם של ארגונים אחרים אתם. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

#### 3.15.1 כלים למפעיל ואחראי ייצור

- פתיחה \ סגירה (העלאה \ הורדה) של המערכת
- מעקב שוטף אחרי פעילות המערכת, כולל הודעות על אירועים חריגים
- לקיחת גיבויים
- ביצוע שחזור והתאוששות



- הרצת מהלכים תקופתיים : דיווחים סטטיסטיים וניקוי

### 3.15.2 כלי שליטה ובקרה למנהל המערכת

- מי נמצא כעת במערכת
- ניטור אבטחת המערכת (במ"מ)
- משלוח הודעות מתפרצות
- מעקב ביצועים : מוניטורים וכלי מדידה – כלי ניטור
- תוכנת ניתוח סטטיסטיקה ומעקב שימוש
- רענון המערכת

### 3.20 חומרה - מחשב לקוח

- מחשב (מעבד מינימלי דרוש)
- יכולת חיבור והרחבות
- אחסנת נתונים
- תכונות צג נדרשות : גודל, רזולוציה
- רכיבי תקשורת : מודם, כרטיס תקשורת, רוחב פס, יציאה מול רשת מקומית
- תשתית סביבתית (אם שונה מסביבה משרדית רגילה)
- ציוד מיוחד
- ציוד קצה
- אמצעי גיבוי

### 3.21 תוכנות מדף תשתית – מחשב לקוח

עבור על סעיפים 2.5.1, 2.9, 2.19, 3.10-3.15 ובדוק מה מתוך אלה (מקבילים לאלה) נדרש במחשב הלקוח.

1. מערכת הפעלה

- מנהלי התקן (drivers)
- תוכנות שרות
- תוכנת אמולציה

2. בסיס נתונים

3. דפדפן

4. Plug-ins, כל תוכנה מקבילה לתוכנה בשרת

### 3.22 תוכנות מדף יישומיות – מחשב לקוח

עבור על סעיפים 2.5.2, 2.9, 2.19, 3.10-3.15 ובדוק מה מתוך אלה (מקבילים לאלה) נדרש במחשב הלקוח. היעזר בהרחבה לרכיב 3.22 כלים למשתמש קצה בקיט עץ מערכת אוניברסלי.

- 10 -

חבילות תוכנה לשימוש ישיר של משתמש הקצה :

- תוכנת Office
- מחולל דוחות מקומי
- קבוצות דיון
- רשימות תפוצה (Newsletters)
- ניהול שיחה – Chat
- תוכנה לעיבודים סטטיסטיים והכנת מצגות

### 3.30 תקשורת פרטית מקומית

תקשורת מקומית איננה רק LAN במובן הטכני, כי אם כל אמצעי המאפשר תקשורת ברמה של בנין או אוסף בנינים סמוכים (קמפוס). קישור לרשתות ה-LAN של הארגון, כולל רשת האינטראנט הפנימית, (כולל יציאה משם לרשת ציבורית) – Extranet.

חשוב מאד להתייחס לעובדת קיומה או אי קיומה בארגון של פונקציות אחראי תקשורת ו/או Helpdesk לניהול ובקרת רשת תקשורת מקומית, ומידת מעורבותם במערכת זו (הקפד לרשום זאת גם ברכיב 4.1).

אם קיימת רשת תקשורת מקומית בארגון, סביר לדרוש שהמערכת הנדונה תשתלב ברשת הקיימת. שילוב זה יתכן בשני אופנים :

- ברמה העקרונית, היינו, שימוש באותה טכנולוגיה
  - שימוש ברשת הפיסית הקיימת, בתנאי שיש קיבולת פנויה או שניתן להגדילה.
- במקרה זה, תאר בתמצית את הרשת הקיימת, כולל גרסה מדויקת. אפשר גם להפנות לתיעוד היצרן ולנוהלי הסיוע הטכני בארגון. במקרה האחרון (ב), יש לקבל מהאחראי גם נתוני קיבולת, עומס וכו'.
- לפירוט מלא (דרישות מלאות לרשת חדשה, או הרחבה ניכרת), היעזר בהרחבה לרכיב 3.30 תקשורת מקומית (LAN) בקיט עץ מערכת אוניברסלי. הרחבה זו מרחיבה, לעומת הפירוט שלהלן, אך גם היא עדיין מיועדת להגדרה של דרישות **מערכת מידע** בתחום תקשורת מקומית. ההרחבה מניחה שקיימת רשת, או מתוכננת רשת כזו בארגון. אם אין, ומערכת המידע הנדונה חייבת להגדיר רשת כזו לצרכיה, או שהמצב הקיים בארגון איננו משביע רצון, יש מקום לפירוט נוסף ברכיב זה. ראה קיט רשתות תקשורת בכרך התמחויות/מערכות תשתית.
- עם זאת, שים לב לא לגלוש לפרויקט תשתית. אם המצב בארגון מחייב טיפול יסודי בתשתית רשת תקשורת מקומית, יש לפעול על פי הקיט הנייל. לתקנים ותקינה ברכיב זה ראה קיט תקינה ותקנים בכרך נושאים תומכים.

הקפד גם על ציון הקשר עם רשתות אחרות, רחבה ו/או ציבורית, רלוונטיות.

רצוי לאמת הדרישות ברכיב זה מול הדרישות לתחנות עבודה ברכיב 3.3 או 3.1 לעיל.

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

אפשר להיעזר במערכת מצאי לבדיקת נפיצות הרשת המקומית המוגדרת כאן, לרכיביה השונים, וניסיונם של ארגונים אחרים אתה. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

תיאור רשת התקשורת יכלול את המרכיבים הבאים :

1. תחנות ברשת המקומית

- מס' כולל : מידי ועתידי
- סוג התחנה
- מיקום (פריסה בארגון) : מידי ועתידי
- רמת אבטחת המידע הנדרשת
- משתמש עיקרי.

2. תיאור כללי של הרשת (הקיימת)

- שם, יצרן, תקן
- קיבולת, עומס קיים וביצועים
- יציאות לרשתות אחרות
- מנהל/אחראי הרשת.

3. תיאור מפורט של טכנולוגית הרשת המקומית (הנדרשת)

### 3.31 תקשורת פרטית רחבה

ברכיב זה הכוונה לרשת תקשורת פרטית רחבה, בפריסה עירונית וארצית המיועדת לתת למשתמשים רחוקים גישה למערכת. אם הכוונה/הדרישה היא לממש את רשת התקשורת הרחבה ע"י רשת ציבורית, הפנה לרכיב הבא.

במקרים רבים, רכיב זה מיותר ואפשר להסתפק ברכיב 3.30 לעיל, שכן הרשת הפנימית של הארגון היא רשת משולבת (MAN) LAN/WAN.

חשוב מאד להתייחס לעובדת קיומה או אי קיומה בארגון של פונקצית אחראי תקשורת ו/או Helpdesk לניהול ובקרת רשת תקשורת רחבה, ומידת מעורבותם במערכת זו (הקפד לרשום זאת גם ברכיב 4.1).

אם קיימת רשת תקשורת רחבה בארגון (סיכוי גבוה שכן!), סביר לדרוש שהמערכת הנדונה תשתלב ברשת הקיימת. שילוב זה יתכן בשני אופנים :

- ברמה העקרונית, היינו, שימוש באותה הטכנולוגיה
  - שימוש ברשת הפיסית הקיימת, בתנאי שיש קיבולת פנויה או שניתן להגדילה.
- תאר בתמצית את הרשת הקיימת, כולל גרסה מדויקת. אפשר גם להפנות לתיעוד היצרן ולנוהלי הסיוע הטכני בארגון. במקרה האחרון (ב), יש לקבל מהאחראי גם נתוני קיבולת, עומס וכו'.
- רשתות תקשורת, רחבה ומקומית, הולכות ומתקרבות זו לזו. הקפד על ציון הקשר עם רשתות אחרות, מקומית ו/או ציבורית, רלוונטיות.

יש לאמת הדרישות ברכיב זה מול הדרישות לתחנות עבודה וציוד קצה אחר ברכיב 3.3 לעיל.

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

אפשר להיעזר במערכת מצאי לבדיקת נפיצות הרשת הרחבה המוגדרת כאן, לרכיביה השונים, וניסיונם של ארגונים אחרים אתה. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

לתקנים בנושא זה, כולל תקני בטיחות (safety), ראה קיט תקינה ותקנים בכרך נושאים תומכים.

תיאור רשת התקשורת יכלול את המרכיבים הבאים:

#### 1. תחנות ברשת הרחבה

- מס' כולל: מידי ועתידי
- סוג התחנה
- מיקום (פריסה בארגון): מידי ועתידי
- רמת אבטחת המידע הנדרשת
- משתמש עיקרי.

#### 2. תיאור כללי של הרשת

- שם, יצרן, תקן
- קיבולת, עומס קיים וביצועים
- יציאות לרשתות אחרות
- מנהל/אחראי הרשת.

#### 3. תיאור מפורט של טכנולוגיית הרשת הרחבה (הנדרשת)

### 3.32 רשת ציבורית

אם אחת האפשרויות לשירותי רשת תקשורת רחבה, עבור מערכת מידע זו, היא שימוש ברשת (מיתוג- מנות) ציבורית (דוגמת ישראלנט), או קישור לרשת האינטרנט העולמית, יש לציין זאת ברכיב זה.

הדגש אם חיבור זה הוא נוסף או תחליף לרשת רחבה פרטית וכיצד חיבור זה משתלב עם רשתות מקומיות ורחבות רלוונטיות. מה הקשר לרשת הממשלתית.

בחיבור לרשת ציבורית, יש להדגיש במיוחד את נושא אבטחת המידע והגנה מפני שימוש בלתי מורשה במשאבי המחשב (לאור הדרישות ברכיב 2.19).

רשימת התכונות לרשת ציבורית (גי' להלן), אין מטרתה הגדרת מפרטים לרשת תקשורת ציבורית. קשה מאד ואין בד"כ גם צורך להכתיב דרישות לרשתות אלה. מטרת הרשימה היא:

- לסייע בבחירת אופציות שירות שהרשת מציעה
- לסייע למאפיין המערכת לבדוק אילו שירותים הרשת הציבורית נותנת ואילו לא, ולהחליט אם צריך להקים רשת פרטית אם לאו.

**חלק מדרישות אלה ניתן לדחיייה לשלב העיצוב ואין מקומו באפיון.**

חובה לציין תחזיות ומגמות לתקופה של 5 שנים ברכיב זה.

לתקנים בנושא זה, כולל תקני בטיחות (safety), ראה קיט תקינה ותקנים בכרך נושאים תומכים.

#### 1. תחנות מהן ניתן להתקשר

- מס' כולל (מקסימלי, מינימלי, ממוצע)
  - סוג תחנה
  - מיקום (פריסה בארגון)
  - רמת אבטחת המידע הנדרשת
  - משתמש עיקרי.
2. תיאור כללי של סוג השירותים הנדרשים
- סוג הנתונים, מהירויות, אמינות וכו'
  - העברות מידע, קשר מסוף-יישום, APPC וכו'.
3. תיאור מפורט של הדרישות מרשת ציבורית
- שירותי הרשת: העברת נתונים, מיתוג מסוף
  - שיטת המיתוג (circuit, message, packet)
  - תמיכה בפרוטוקול ומשטר תקשורת נדרשים
  - שימוש ברכיבי תקשורת קיימים (בקרים, מודמים)
  - מהירויות
  - דחיסת נתונים
  - אבטחת נתונים
  - חיבור לרשתות מקומיות (LAN)
  - סוגי ציוד נתמכים
  - דרישות בתחנות הקצה: חומרה, כרטיסי תקשורת, תוכנה
  - שיטת החיוב וההתחשבות
  - רמות שירות וזמני שירות
  - אמינות (זמני MTBF, MTTR)
  - יציאות לרשתות אחרות
  - מנהל/אחראי על הקשר לרשת בארגון/בפרויקט.
4. קישור לרשת האינטרנט הכללית
- קישור ל-ISP
  - כתובות IP
5. קישור למאגרי מידע
6. קישור לרשת ציבורית אחרת על בסיס סקטור כלכלי, גופים בינ"ל, קונסורציום וכו'.

### 3.33 טכנולוגיות משיקות

לסכום רכיב הטכנולוגיה, ציין טכנולוגיות שכנות, בארגון או מחוצה לו, אתן יהיה לטכנולוגיה הנדונה קשר משמעותי. לטכנולוגיה שכנה יכולה להיות השפעה על מערכת המידע הנדונה במספר אופנים:

- העברות מידע
- מבני נתונים (מילון שדות)
- מיתוג מסופים וקבלת שירותי מידע
- מבני נתונים (מילון שדות)
- החלפת מידע מקצועי
- שיתוף משאבים
- משתמשים משותפים.

אפשר להיעזר במערכת מצאי לבדיקת קיום ומצב טכנולוגיות שכנות, לרכיביהן השונים, וניסיונם של ארגונים אחרים איתן. במשרדי ממשלה אפשר להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים - בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

ציין כאן כל טכנולוגיה שכנה שיתכן שיש לה השלכות על המערכת:

- שם ותיאור כללי קצר
- מחשבי Backend ואחרים איתם יש למערכת קשר חזק, אך לא נכללו בסעיפים 3.1-3.19 לעיל
- מאגרי מידע מרכזיים
- ספק ראשי
- תכונות חומרה
- תוכנה: מערכת הפעלה, בסיס נתונים, מילון נתונים
- תקשורת WAN ו-LAN. דגש על Gateways
- טכנולוגיות אחרות שכבר קשורות אתה
- שיטה קימת להעברת מידע על מדיה מגנטית (לא תקשורת).

### 3.98 נקודות פתוחות (וחלופות)

סעיף זה מטרתו לרכז את כל הנקודות שנשארו פתוחות ברכיב הטכנולוגיה. נקודות פתוחות הן לעתים קרובות חלופות (אלטרנטיבות) אפשריות ולפיכך גם החלופות השונות שתועדנה כאן. עם זאת, יש לבדוק היטב את שני המקרים הבאים:

- נקודות פתוחות (וחלופות) ברמת רכיב משני (3.X).
- חלופה מקיפה לרכיב הטכנולוגיה בכללותו.

שני מקרים אלה, שהם השכיחים יחסית, לא יתועדו למעשה ברכיב זה. נקודות פתוחות וחלופות ברמת רכיב משני - מקומן ברכיב המשני (3.X.98) חלופה מקיפה לרכיב הטכנולוגיה כולו - תתואר ע"י **תיעוד חוזר ומלא** של רכיב 3 כולו (3.0 - 3.33). רכיב זה מיועד למקרי ביניים, היינו, מקרים בהם

החלופות והנקודות הפתוחות הן כלליות ואינן משנות את תפיסת הטכנולוגיה הכללית, או שעדיף לבחון אותן ברמה זו כיוון שהן משפיעות על קבוצה חלקית של רכיבי הטכנולוגיה (3.1 - 3.33).

להרחבה נוספת ולהסבר מקיף בנושא חלופות, ראה קיט ניתוח חלופות בכרך נושאים תומכים.

מקובל להניח שסגירת הנקודות הפתוחות והכרעה בין החלופות השונות ברכיב הטכנולוגיה אינם כה חיוניים כמו ברכיב היישום, בפרט אם בהמשך מתוכננת יציאה למכרז. עם זאת, לא רצוי להשאיר מרחב פתרונות גדול מדי. אפיון פתוח מדי של רכיב הטכנולוגיה והסתמכות על הספקים שהם כבר ימצאו פתרון טכנולוגי טוב, יביא לאי בהירות של המפרט, המפ"ל והיכולת לקבל החלטה.

חשוב ביותר להשאיר בתיק תיעוד גם לנקודות שנסגרו ולחלופות שנדחו, על מנת לסייע בדיונים מחודשים בעתיד.

רכז כאן את הנקודות הפתוחות המרכזיות בפרק הטכנולוגיה, שפתרון חיוני למהדורה הקרובה של המערכת:

- רשימה כוללת ותמציתית
- תיאור מפורט לנקודה-פתוחה/חלופה

### 3.99 טכנולוגיות עתידיות

סעיף זה מטרתו לרכז את כל הדרישות והטכנולוגיות העתידיות ברכיב הטכנולוגיה שאינן כלולות במהדורה הנדונה, אך סביר שיכללו במהדורות עתידיות. רישום דרישות (תחזיות) עתידיות אלה, הוא במטרה כפולה:

- לא לאבד מידע קיים
  - להדגיש מה לא כלול במהדורה הנדונה ולסייע בכך לניהול תצורה של המערכת.
- מיקומן של דרישות עתידיות אלה ברכיב ראשי (3) של עץ המערכת, הוא מכוון. אם יש מספיק מידע וצורך לרשום דרישות אלה ברמה מפורטת יותר, היינו, ברכיב טכנולוגיה משניים (3.X), יש לבדוק היטב אם אין לפנינו הגדרה מלאה של המהדורה הבאה של המערכת. הגדרה כזו פירושה **תיעוד מלא** של רכיב 3 כולו ומיקומו בתיק אפיון נפרד (של המהדורה הבאה).

רכיב זה איננו עומד בפני עצמו ומושפע מהרכיבים המקבילים ביעדים (1.99) ו/או ביישום (2.99). במקביל אליו יש להגדיר את הרכיבים המקבילים במימוש (4.99) ובעלות (5.99) המשלימים רכיב זה.

תמיכה בטכנולוגיות עתידיות:

- חומרה וציוד מרכזי
- מערכות הפעלה ובסיסי נתונים
- תוכנות תשתית
- מולטימדיה
- גרסאות מתקדמות של דפדפנים
- כלי פיתוח
- תקשורת, שילוב עם טלפוניה
- תמיכה בציוד קצה (לקוח) מתקדם, כגון: מכשור סלולרי, מחשבי PDA

## 4. מימוש

### 4.0 כללי – הבהקים

בשלב ביניים אפשר לכלול ברכיב זה (בדומה ל-2.0 ו-3.0 לעיל), נושאים ודגשים כלליים במימוש המערכת שחשוב לזכורם ושמקומם הסופי לא ברור. במהלך בניית התיק יועברו נושאים אלה למקומם הטבעי באחד הרכיבים להלן בפרק זה (4.X) או בפרקים האחרים. בשום פנים ואופן אין להפוך רכיב זה לאפיון מימוש המערכת! לכל מידע המופיע כאן יש מקום מוגדר ברכיבים להלן.

בצורתו הסופית, יכיל רכיב זה אך ורק עקרונות יסוד, תפיסה כללית וכו', נושאים שחשוב להדגיש ברמת הבהקים (Highlights) של המימוש ולפי הדוגמא שלהלן. זכור שסעיף זה יופיע בד"כ בתמצית מנהלים שתופק מתיק זה. ראה קיט כתיבת תיעוד בכרך נושאים תומכים.

מימוש הוא מרכיב נכבד באפיון המערכת. אין טעם להשקיע באפיון היישום והטכנולוגיה - וודאי לא בבנייתם או רכישתם - אם לא ברור כיצד אפשרי או מוצע להטמיע את המערכת, לתפעל אותה וכו'. במימוש, כמו בטכנולוגיה, חובה להישען על תשתית וכלים תקינים הקיימים בארגון ואין הצדקה להמציא את הגלגל.

#### דוגמא: (בלבד!)

1. על מנת להבטיח תפעול ותחזוקה תקינים של המערכת, נדרשת מעורבות שוטפת ופעילה של \_\_\_\_\_ . ראה פירוט ברכיב 4.1 להלן.
2. מומלץ שהשלב הבא לאחר האפיון יהיה שלב הבקשה להצעות, בו יושם דגש על פניה לבתי תוכנה (החומרה היא משנית) ופתרון של חבילת תוכנה מוכנה והתאמות מועטות. ראה פירוט נוסף ברכיב 4.3, להלן.
3. מומלץ גם שהספק שיבחר (בית התוכנה) יבצע את התפעול והתחזוקה השוטפים, כמפורט ברכיבים 4.4 ו-4.6 להלן.
4. צפוי תהליך ממושך של שילוב המערכת בארגון, בפרט לאור ההמלצה להתבסס על חבילת תוכנה מוכנה. ראה הרחבה ברכיב 4.7 להלן.
5. תכנית הבדיקות המוגדרת בשלב זה היא חלקית ויש לשקול השלמתה לפני הפניה לספקים. ראה 4.8.1 להלן.

#### סוף דוגמא

### 4.1 גורמים מעורבים

לאחר שמומחה היישום והלקוח העיקרי הוגדרו ברכיב 1.1 לעיל, ומשתמשי המערכת הוגדרו ברכיב 2.2, רכיב זה מגדיר כלל הגורמים המעורבים בפרויקט:

- **הצוות המנהלי** האחראי להחלטות הניהוליות/כספיות העיקריות
  - **הצוות מקצועי** המבצע בפועל את העבודה - בראשו **מנהל הפרויקט**.
  - **גורמים חיצוניים** (אם צפויים) - ספקי חומרה/תוכנה
- ראה הגדרה בסעיף ניהול ובקרה בקיט מודל מפת"ח שבכרך מבוא ובהוראות התכ"ם (למשרדי ממשלה).



רצוי כמובן לוודא, במידת האפשר, שאותם גורמים ילוו את הפרויקט לכל אורך מחזור החיים, כולל השלבים המאוחרים. שים לב במיוחד לגורמים שמעורבותם היא:

- באישור שינויים ותיקונים במהלך תחזוקת המערכת
- באישור התקשרויות חיצוניות ובקרה תקציבית
- אישור היבטים של חוק ומינהל תקין.

נכון שבשלב האפיון קשה עדיין לסגור את כל הגורמים המעורבים, בפרט אם ההמשך הוא שלב הבקשה-להצעות. בה במידה נכון שסגירת כל הגורמים המעורבים היא עצמה חלק מאפיון המערכת. במהלך האפיון, בוודאי בסיומו, צריכים גורמים אלה להתברר סופית.

#### 4.1.1 ניהול

- ועדת היגוי (צוות מנהלי) לפרויקט
- מנהלת הפרויקט
- לכל גורם יש לפרט:
- שם, טלפון, פקס, e-Mail
- תפקיד ומיקום בארגון (ארגון חיצוני?)
- אחריות מיוחדת בצוות
- מעורבות בעבר (בפיתוח) או גם בתחזוקה
- מעורבות בכלל המערכת או בתת-מערכת מסוימת.

#### 4.1.2 צוותים מקצועיים – צוותי הפיתוח

- צוות אפיון
- צוות עיצוב ובנייה
- צוות ממשק המשתמש, אבטיפוס \ דגם
- צוות טסטים וכו'

לפי תפקידים: מנהל הפרויקט, מנתח המערכת, תכניתן ראשי וכו'. מומחי יישום וצוותי משתמשים המעורבים בפרויקט מוגדרים ברכיב 1.1 לעיל.

לכל גורם יש לפרט:

- שם, טלפון, פקס, e-Mail
- תפקיד ומיקום בארגון (ארגון חיצוני?)
- אחריות מיוחדת בצוות
- מעורבות בעבר (בפיתוח) או גם בתחזוקה
- מעורבות בכלל המערכת או בתת-מערכת מסוימת
- מעורבות בשלבים (מסוים) במחזור החיים.

#### 4.1.3 סיוע טכני

- תשתיות, מערכת הפעלה, בסיסי נתונים, תקשורת
- אבטחת איכות
- הנדסת אנוש וממשק משתמש
- אבטחת מידע – Security
- רכש ותקציב, מעקב עלויות
- מרכז תמיכה – Call center

#### 4.1.4 ספקים וגורמי חוץ

- חברות מחשבים ובתי תוכנה
- יועצים ומומחים

למידע נוסף העשוי לסייע בהגדרת תיעוד רכיב זה, ראה הרחבה 4.1.4 ספקים וגורמי חוץ בקיט עץ מערכת אוניברסלי

## 4.2 תכנית עבודה

מרכיב מרכזי באפיון מערכת ממוחשבת בכלל, מערכת מידע בפרט, הוא האופן בו מוצע/מומלץ לבצע את הפיתוח המלא של המערכת: ע"י פיתוח עצמי בבית, רכישת חבילת תוכנה, הסתייעות בקבלן חוץ וכו'. ובכלל, האם המערכת היא בעיקרה פתרון חומרה: הסבת מערכת קיימת, או מימוש של תוכנה חדשה על ציוד קיים בארגון. כך או כך, בסיום שלב האפיון, צריך שתינתן תשובה לשאלות אלה ותשורטט התוכנית והדרך למימוש כולל של המערכת.

שים לב למלכוד הבא: אם יוחלט על פניה לספקים, יידרש מהספק להציע ברכיב זה את תוכנית העבודה הנראית לו. הערכת ההצעות תהיה, בין השאר, על סמך תשובת הספק ברכיב זה. במקרים כאלה, אין להשקיע זמן רב מדי באפיון רכיב זה, אלא להתרכז בהגדרת אילוצים, אבני דרך וכו', אשר ינחו את הספק בתשובתו. הנחיות חשובות לספק:

- יש לכלול בעלות הפיתוח גם תיקונים והתאמות בעקבות תוצאות בדיקות המערכת
- הצעתו תהיה Turnkey fixed price.

במקרים של מימוש פנימי (פיתוח בבית), אין להסתפק באילוצים ואבני דרך ויש להציג ברכיב זה תכנית ברורה אשר תאפשר לדרג המנהלי של הפרויקט לקבל החלטות מתאימות. היעזר בקיט ניהול פרויקטים ותוכניות עבודה שבכרך נושאים תומכים.

#### 4.2.0 שיטת הפיתוח

1. שיטת הפיתוח וניהול הפרויקט שנבחרה: סדרתי, סבבים, יחידות מסירה וכו'. ראה הסבר בקיטים המתאימים בתת-כרך ניהול פרויקט תוכנה שבכרך ניהול.
2. סוג הפתרון הכללי הנדרש. שים לב לקשר עם רכיבים 1.7 ו- 2.1:

- חומרה בעיקרו או תוכנה בעיקרו
- חבילה מוכנה או פיתוח תוכנה

- מערכת חדשה או המשך מערכת קיימת (הסבה, הגירה)

#### 4.2.1 תכנית פיתוח כללית

- אבני דרך ראשיות ותוצרים
- תכנית עבודה כללית: תרשים גאנט בגדול

#### 4.2.2 תכנית פרטנית

- תרשים (תרשימי) גאנט ולו"ז פרטניים
- קישור לרשימות משימות ותוכניות עבודה פרטניות של הצוותים השונים

### 4.3 השלב הבא \ המיידית

ברכיב הקודם (4.2) שורטטה תוכנית עבודה כוללת לפיתוח מלא של המערכת, תוכנית שבשלב האפיון היא עדיין כללית ולא יכולה להיות לגמרי מדויקת. רכיב זה מתייחס **לצעד הבא** בלבד. ברכיב זה צריכה להיות מוגדרת ברור התחנה הבאה של המערכת, כולל הערכת עלויות, לו"ז, גורם מבצע וכו'. בקיצור, תוכנית עבודה ברורה שאפשר להביאה לפני הגורמים המעורבים (4.1 לעיל) לדיון וקבלת החלטות.

הבדל נוסף בין רכיב זה לרכיב 4.2 הוא, שבעוד שברכיב 4.2 נדרשת תוכנית כוללת ובמבט כללי על המערכת לאורך זמן רב, ברכיב זה נדרשת התייחסות קונקרטית לשלב הבא **במהדורה X של המערכת**.

- אם השלב הבא הוא בקשה להצעות: רכיב זה (4.3) אמנם שונה מרכיב 4.2
- אם השלב הבא הוא עיצוב ובנייה אפשר למעשה לאחד את שני רכיבים אלה.

יש לזכור שבנוהל מפת"ח כל שלב מכיל פעילות 0 בה מוגדרת סופית תוכנית העבודה לאותו שלב. פעילות 0 זו נעשית ע"י הגורם המיועד לבצע אותו שלב, בתחילתו. האחריות על תוכנית העבודה הסופית היא בסופו של דבר בידי הגורם שיבצע. התוכנית שברכיב זה, היא המלצה. אפשר להיעזר בפעילויות 0 אלה. ראה הקיטים העוסקים במחזור החיים בכרך מחזור חיים. תכנית זו נרשמת בפרק המנהלה של תיק המערכת המלווה את השלב (ראה לעיל בתחילת תיק האפיון). היעזר גם בקיט ניהול פרויקטים וניהול משימות שבכרך נושאים תומכים.

רכיב זה יכול תיאור של מה מתוכנן להתבצע במיידית, על פני הרבעון הקרוב:

- מטרה \ תוצר עיקריים - תכולה
- גורם מבצע (מומלץ)
- תכנית עבודה: רשימת משימות – Task List

### 4.4 תפעול שוטף

רכיב זה הוא רכיב עיצוב ברובו. עם זאת, כבר בשלב האפיון חשוב מאד להגדיר, במירב האפשרי, כיצד וע"י מי בדיוק תתופעל המערכת באופן שוטף ולהציב דרישות ברורות בנושא.

ביחידות מחשוב בהם קיימים נהלי ייצור ותפעול כלליים וכלי ייצור ותפעול ממוכן (רכיב 3.15 בעץ המערכת) יש לפעול לפיהם.

בעיצוב, יש להגדיר ברכיב זה תיק תפעול מסודר למערכת. מבנה התיק יהיה בהתאם לנוהלי התפעול הקיימים בארגון או כמוגדר בהרחבה לרכיב 4.4 תיק תפעול המערכת בקיט עץ מערכת אוניברסלי. באפיון, יש להתוות מסגרת כללית.

שים לב לקשר שבין רכיב זה ורכיב 3.15 המגדיר את **כלי התפעול**.

מיכון תיק התפעול והפעלתו Online בעזרת כלי ממוחשב הם יתרונות ברורים, אשר יכולים לחסוך מאמץ רב בתיעוד הסופי של רכיב זה ובתחזוקתו השוטפת. אלא שמיכון זה חייב להתאים לפתרון הטכני שיבחר ברכיב 3.15.

חשוב להבחין בין רכיב זה ובין תיק תפעול. תיק התפעול של המערכת הוא התיעוד המלא והמחייב של תפעול המערכת, בפרט מסוף שלב עיצוב ובנייה והלאה. במערכות רבות תיק התפעול הוא מסמך אלקטרוני חי המשולב עם כלי התפעול והייצור של המערכת. רכיב זה (4.4) יכיל, בשלבים המוקדמים של חיי המערכת: ייזום ואפיון, את דרישות תפעול המערכת. בהדרגה, עוברות דרישות אלה לתיק התפעול ורכיב זה הופך להיות רכיב מצביע.

ראה קיט תפעול בכרך יסודו/מחזור חיים.

דרישות תפעול המערכת – קרי תיק התפעול – יכללו, בין השאר, את הנושאים הבאים:

- הגורם המתפעל
- גורמים מסייעים ומפקחים: סיוע טכני, הלקוח, הנהלת מרכז המחשבים
- זמני הפעלה (משמרות)
- משאבים
- פתיחה \ סגירה של המערכת
- השבתות לצרכי תחזוקה יזומה, רה ארגון וכדומה
- בקרת שגיאות
- גיבויים ושחזורים
- תפעול במצבי חירום ומשאבים חלקיים
- אבטחה ובטיחות

#### 4.5 אינדקס התיעוד

רכיב זה משמש אינדקס (Reference) מרכזי לכל התיעוד המלווה את המערכת. מטרת רכיב זה היא **לרכז את כל ההפניות למסמכי המערכת** במקום אחד. כל התיעוד הנזכר ברכיב זה נוצר, באופן טבעי, במהלך בניית המערכת - ברכיב אחר בתיק זה או בחוץ. אין הכוונה לייצר תיעוד ברכיב זה ואף לא להעביר פיסית לרכיב זה את כלל תיעוד המערכת. מי שעובד מסודר לא נדרש ליצור כאן שום תיעוד חדש, אלא רק לעקוב ולבדוק היכן נמצא (יימצא) כל תיעוד ולרשום זאת כאן.

רכיבי האינדקס הראשיים הם אפוא:

- תיעוד עץ המערכת - תיעוד תפעולי
- תיעוד מחזור החיים – תהליך הפיתוח

בהמשך, ניתנת חלוקה נוספת כולל טבלה המציעה איך בדיוק יראה האינדקס. אך בפשטות, מה שנדרש ברכיב זה הוא לפרק אותו לשני תת-רכיבים:

- 4.5.1 יכיל הפניה לתיעודים הנדרשים לתפעול ולאחזקת המערכת: תיק תחזוקה, מדריך למשתמש, חוזי שירות, תיקי תכנות ועוד
- 4.5.2 יכיל הפניה לתיעודים המשמשים לבניית המערכת: מסמך ייזום, תיק אפיון, (מפרט והצעת ספק), תיק עיצוב וכו', והפניה לתיעודים המשמשים לניהול ובקרת השלבים השונים במחזור החיים: סיכומי דיון, תכניות עבודה וכו'.

התיעוד התפעולי הולך ומתכנס לתוך תיק זה עצמו ומוצמד לרכיבים המתאימים: תיק תפעול המערכת הוא ברכיב 4.4, מדריך למשתמש הוא ברכיב 4.7.4, תיקי התכנות ברכיב 2.7 ועוד.

מה מכל זה צריך להיות מוגדר בשלב האפיון? הדגש העיקרי, בשלב האפיון, הוא כמובן על מסמך האפיון עצמו (הנך נמצא כאן!). עם זאת, חשוב שכבר בשלב האפיון תתגבש תמונה ראשונה באשר לשאר התיעודים של המערכת: מי אחראי להם, מתי ייכתבו, האם יש להם בסיס בתיעוד קיים וכו'.

חלק מהתיעוד, בפרט התיעוד התפעולי, ניתן להעתיקה מנהלי הארגון או מערכות אחרות. לכל התיעודים יש גלופות לימוד ועבודה בקיטים השונים בנוהל מפת"ח.

#### 4.5.1 תיעוד תפעולי

שם המסמך	רכיב פנימי	הפנייה לתיעוד	סטטוס כללי
תיק מערכת (תחזוקה)	----	ספריהשם מסמך	
מדריך למשתמש	רכיב 4.7.4 להלן	ספריהשם מסמך	
תיק תפעול	רכיב 4.4 לעיל	ספריהשם מסמך	
חוזי שירות, הפעלה ותחזוקה	רכיב 4.6 להלן	ספריהשם מסמך	
תיעוד יצרן	----	ספריהשם מסמך	

#### 4.5.2 תיעוד תהליך הפיתוח

שם המסמך	קיט \ גלופה	הפנייה לתיעוד	סטטוס כללי
מסמך ייזום	ייזום מערכת	ספריהשם מסמך	
אפיון (על, מלא)	אפיון מערכת	ספריהשם מסמך	
בקשה להצעות – RFP	בקשה להצעות – RFP	ספריהשם מסמך	
מפ"ל, הצעות הספקים	בקשה להצעות – RFP	ספריהשם מסמך	
חוזה לפיתוח המערכת	חוזים	ספריהשם מסמך	
תיק עיצוב	עיצוב ובנייה	ספריהשם מסמך	
תיק בדיקות	בדיקות מערכת – Testing		
תיק ממצאים	בדיקות מערכת – Testing		
תכתובות			
סיכומי דיון ושיקופים		בנספחי התיקים	

## 4.6 שירות ותחזוקה

רכיב זה הוא העוגן לניהול תצורה. כבר בשלב האפיון יש לתת את הדעת על הדרישות לשירות ותחזוקת המערכת.

יש להגדיר בברור את מטרות התחזוקה: תיקוני שגיאות, התאמות לשינויי חוק ושינויים כפויים אחרים. לא שינויים ושיפורים (שו"ש). כמו כן, יש להעריך את עלות התחזוקה ולהגדיר חסמים למשאבים שיוקדשו לתחזוקה: כ"א וכו'. תזכורת: ברירת המחדל למשך חיי מערכת הוא 5 שנים.

שים לב שבמונח "תיק זה" בהנחיות להלן, הכוונה בד"כ לא לתיק האפיון אלא לתיק התחזוקה שהוא המשך ישיר וטבעי לתיק האפיון והוא הרלוונטי לשלב התחזוקה. רכיב זה (4.6) הופך, בשלב התחזוקה, לעוגן התחזוקה הראשי דרכו ניתן להגיע לשאר הרכיבים. כל דרישות השירות והתחזוקה והשינויים המרכזיים שאירעו במערכת, מרוכזים ברכיב זה וממנו אפשר להגיע לשאר הרכיבים. בשלב האפיון (בו אנו נמצאים כעת), יש להכין עוגן זה ולבססו לקראת תפעול המערכת ותחזוקתה.

תחזוקת המערכת תלויה, כמובן, בסוג הרכיב:

1. תחזוקת רכיב 1 - יעדים, פירושה עדכון מפעם לפעם של רכיב 1 בתיק זה.

2. תחזוקת רכיב 2 - היישום, פירושה ביצוע עדכונים במערכת עצמה:

- תחת נהלי בקרת תצורה בסעיף 4.6.2 להלן

- תוך עדכון רכיב 2.X המתאים בתיק זה.

3. תחזוקת רכיב 3 - טכנולוגיה, פירושה ביצוע עדכונים במערכת עצמה:

- תחת נהלי בקרת תצורה בסעיף 4.6.2 להלן

- תוך עדכון רכיב 3.X המתאים בתיק זה.

4. תחזוקת רכיב 4 - מימוש, פירושה עדכון מפעם לפעם של רכיב 4 בתיק זה.

5. תחזוקת רכיב 5 - עלות, פירושה עדכון מפעם לפעם של רכיב 5 בתיק זה.

מרכז הכובד הוא אפוא, ברכיבי היישום והטכנולוגיה. אך בהבדל ברור שהטכנולוגיה מטופלת בד"כ ע"י גורם מקצועי מרכזי, ספק חומרה לרכיבי חומרה וסיוע טכני בארגון ו/או ספק תוכנה לרכיבי תוכנות-תשתית.

רכיב זה כולל גם את נושא האחריות ולכן בכל מקום בו נאמר שירות ותחזוקה, יש לכלול גם את נושא האחריות, היינו מה הוא סוג השירות והתחזוקה שנדרש בתקופת האחריות (אם שונה מהשירות והתחזוקה הכלליים) ולמשך איזו תקופה.

רכיב זה כולל גם את נושא ניהול תצורה, ברמה הכללית לפחות, בשל הקרבה הטבעית של נושא זה לתחזוקת המערכת. ראה קיט ניהול תצורה בכרך נושאים תומכים..

להסבר מפורט יותר, ראה קיט תפעול ותחזוקה בכרך מחזור חיים. ראה גם הרחבה לרכיב 4.6 שירות ותחזוקה בקיט עץ מערכת אוניברסלי

### 4.6.1 מרכז תמיכה – HELPDESK (CALL CENTER)

- איש, נהלים ותפעול (זמני פעילות, זמני תגובה)

- משאבים

- רישומים : קבציי מעקב
- סיוע חיצוני

תיעוד רכיב זה (ומימוש) יכולים להיות ברמות שונות. החל מרשימת דרישות פשוטה להוספת המערכת למרכז תמיכה קיים, דרך פירוט בנספח 4.6.1 בפרק הנספחים בסוף התיק, עבור מקרים של צורך בשינוי והרחבה משמעותיים של מרכז קיים, וכלה בבנייה של מערכת תמיכה עצמאית אשר תפותח כפרויקט (תשתית) נפרד עם מחזור חיים ועץ מערכת מלא משלו! בכל מקרה, חשוב לזכור שמרכז תמיכה טוב הוא חלק אינטגרלי מהמערכת ומשובץ ברכיבי המערכת האחרים: ביישום, בטכנולוגיה ובמימוש של כלל המערכת. מערכת טובה מנהלת את עצמה!

#### 4.6.2 תחזוקת היישום

- בקרת תצורה, ניהול שינויים
- גורם אחראי \ מבצע
- שינויי תכנים (מערכות אינטרנט)
- שינויי תוכנה (קוד)
- שינויים במבנה קבצים וטבלאות

#### 4.6.3 תחזוקת תשתית וטכנולוגיה

- בקרת תצורה, ניהול שינויים
- ספקי חומרה, תוכנה, ספקי שירות חיצוניים. בהיבט סוג השירות שהם מספקים. ההסכם (החוזה) עצמו מכוסה ברכיב 4.5.1 לעיל.
- זמני קריאה ושירות
- תחזוקה מונעת
- מעקב פנימי של אגף התפעול והייצור בארגון

#### 4.6.4 מימוש שוטף

ניהול המימוש השוטף של המערכת יכול להיעשות ברכיב זה, במרכז עבור כל פרק 4 מימוש, או ברכיבים הספציפיים של פרק המימוש (למשל 4.7 השתלבות בארגון וכו'). החלט ומנע כפילויות. ככלל יכיל סעיף זה פרוט של הדרכות שוטפות, עדכון המדריך למשתמש, שיווק שוטף וכו'.

#### 4.6.5 עלויות שוטפות

מעקב עלויות שוטפות, כולל חוזים עם ספקים יכול להיעשות ברכיב זה או ברכיב 5.2 להלן. החלט ומנע כפילויות.

### 4.7 השתלבות בארגון – הנעת המערכת

שים לב להבדל חשוב שבין רכיב זה לרכיב 1.4 לעיל. רכיב 1.4 דן באופן בו יעדי המערכת משתלבים ביעדי הארגון: בתכנית העבודה הכוללת של הארגון, בתכנון האסטרטגי, בחוקת הארגון וכו'. ברכיב זה הכוונה מעשית ופשוטה: כיצד תוטמע המערכת, כיצד יונעו האנשים בארגון להשתמש בה? איך תתבצענה ההדרכות? האם יש צורך בהסבות? מי אחראי למדריך למשתמש ועוד.

יש להכין בשלב האפיון תוכנית הטמעה כוללת למערכת אשר תסייע בהנחלת המערכת בארגון, תוך התחשבות בנהלים קיימים, גורמי אנוש, או"ש וכו'.

רכיב מרכזי בהטמעת המערכת הם **מערכי הדרכה ותוכניות הדרכה**. חשוב שמערכים ותוכניות אלה ייבנו בהתאמה לסוגי המשתמשים במערכת.

ציין בעיות צפויות בהפעלת המערכת, הנובעות מאופן השתלבות המערכת בפעילות השוטפת של הארגון.

היעזר בהרחבה לרכיב 4.7 השתלבות בארגון בקיט עץ מערכת אוניברסלי ובקיט הטמעת מערכת בכרך נושאים תומכים.

#### 4.7.1 הטמעת המערכת

- פנימית: תכנית הדרכה והטמעה בארגון, מערכי ההדרכה
  - חיצונית: שיווק, הפצה ופרסום
- היעזר בהרחבה לרכיב 4.7.1: הדרכה והטמעה בקיט עץ מערכת אוניברסלי.

#### 4.7.2 הסבות

אין להיסחף בסעיף זה. רוב ההסבות מקומן בתהליך הפיתוח. הכוונה כאן רק להסבות נתונים (קבצים וטבלאות) שהן חלק מהמעבר למערכת החדשה, חלק מהתקנת המערכת.

- הסבת קבצים, טבלאות וממשקים
- הסבת תוכנות ושגרות

#### 4.7.3 או"ש

השתלבות בנהלי הארגון, בעבודת הארגון ותפקודו. ההשלכות העקרוניות על תהליכי האו"ש בארגון, שינויים נדרשים וכו', הוגדרו, תועדו והוסכמו ברכיב 1.4 (תת רכיב 1.4.3) לעיל. רכיב זה, 4.7.3 מטפל במשמעות המעשית של השלכות אלה ובפעולות שיש לבצע הלכה למעשה, כחלק בלתי נפרד מהטמעת המערכת.

#### 4.7.4 מדריך למשתמש

היעזר בהרחבה לרכיב 4.7.4: מדריך למשתמש בקיט עץ מערכת אוניברסלי ובתבנית מדריך למשתמש בקיט הטמעת מערכת.

### 4.8 חוסן ואמינות

תכנית לבדיקת מערכת היא גורם מרכזי אך לא יחיד במכלול הרחב יותר של חוסן המערכת ואמינותה. פעולות רבות המתבצעות לאורך שלבי הפיתוח תורמות לאמינות המערכת וחוסנה, כגון: שיקופים, חקר ישימות, ניתוח חלופות, אבטחת מידע וכמובן הגדרת שיטת גיבוי והתאוששות כמפורט להלן.

קיומה של תכנית לבדיקת המערכת, הוא חלק חיוני משלב האפיון. סביר ביותר שבהמשך הפרויקט יתווסף גורם עצמאי שיתמחה בבדיקת המערכת. גורם זה עשוי לקחת את תוכנית הבדיקה שהוגדרה בשלב האפיון (כאן) ולהרחיבה באופן משמעותי. אך הבסיס הראשון לתוכנית הבדיקה **חייב** להיכתב ע"י הגורם המאפיין את המערכת! ראה הרחבת נושא זה בקיט בדיקות מערכת - Testing בכרך מחזור חיים.



בגישת מפת"ח, ברור שהתוכנית לבדיקת המערכת תיבנה גם היא, בעקרון, בהתאם לעץ המערכת ותכיל התייחסות, ברמה זו או אחרת, לכל רכיבי עץ המערכת. אך הדגש העיקרי צריך להיות על תכנית לבדיקת רכיבים מסוימים, כמפורט להלן, אשר תדירות השינויים בהם גבוהה ביותר והם לב המערכת (רכיבי יישום בעיקר). תכנית לבדיקה יכולה להיות בשתי רמות (לפחות):

- רמה פרטנית, המציינת בדיוק אילו רכיבים יש לבדוק, בלתי מותנה בסוג השינוי שבוצע.
- תוכנית שלדית, המציינת בראשי פרקים אילו סוגי רכיבים יש לבדוק. התוכנית הפרטנית הסופית תיקבע לפי כל שינוי שבוצע.

היקפה של תכנית הבדיקה מותנה בהיקף המערכת. מערכות מסוג ג1 יכתבו בסעיף זה תכנית בסיסית ויעזרו בנספח 4.8.1 התמציתי. מערכות מסוג ג2 יכתבו תכנית מורחבת ויעזרו בנספח 4.8.1 המורחב. תכנית בדיקה זו תתועד בנספח 4.8.1 בסוף תיק האפיון על מנת לא להעמיס על רכיב 4.8.1 שלהלן (מהרכיב תהיה הפניה לנספח). מערכות מסוג ג3 יבנו תיק בדיקות המלווה כצל את תיק האפיון ויעזרו לשם כך בגלופות הלימוד והעבודה שבקיט בדיקות מערכת בכרך מחזור חיים.

בכל מקרה יש לזכור שמדובר עדיין באפיון, ותכנית הבדיקה איננה צריכה להכיל את כל נתוני הבדיקה (test data) והתסריטים (scripts) המדויקים. אלה יוגדרו בעיצוב. הדגש באפיון הוא **מה ייבדק ולא איך בדיקת ייבדק**, היינו, רשימה של איזה תהליכים, טרנזקציות, קבצים, דוחות וכו' ייבדקו ולא תהליך הבדיקה המדויק. גם את פירוט השיטות והכלים אפשר להשאיר לעיצוב. הסיבה שפירוט זה מוזכר להלן היא בעיקר עבור יציאה למכרז, היינו, במקרים בהם תיק האפיון הופך למפרט.

#### 4.8.1 תכנית בדיקה

רכיב זה (בדומה לכל שאר רכיבי עץ המערכת) הולך ומתעבה במהלך הפרויקט. תחילתו (בשלב האפיון) ברשימה תמציתית של בדיקות המערכת הבסיסיות הנדרשות, המשכו (בשלב עיצוב ובנייה) בהפניה לנספח 4.8.1 תכנית בדיקה המרחיב את תיאור בדיקות המערכת וסופו (בשלב בדיקות המערכת) הפנייה לתיק בדיקות מלא. היעזר בהרחבה לרכיב 4.8.1 תכנית בדיקה שבקיט עץ מערכת אוניברסלי.

תמצית בדיקות המערכת:

- בדיקות פונקציונאליות: מסכים, תהליכים, קלטים, דוחות וכו'
- בדיקות טכנולוגיות: זמינות \ ביצועים של חומרה, תוכנת תשתית, תקשורת וכו'
- בדיקות מימוש: הסבות, מוכנות להטמעה, שירות ותחזוקה וכו'

#### 4.8.2 זמינות ושרידות

##### גיבוי והתאוששות

1. גיבוי והתאוששות כנגד אילו סיכונים/בעיות
2. נפילת המערכת כולה
3. נפילה חלקית (לפרט ולהסביר)
4. תהליך ההתאוששות והשחזור
5. דרישות גיבוי והתאוששות במונחים של:

- זמנים

- משאבים
  - נוחות ההפעלה (איך יילקחו הגיבויים וע"י מי)
  - 6. השלכות על תפעול שוטף של המערכת
  - 7. ניהול ארכיב הגיבויים
  - 8. זמנים : MTTR (Mean Time To Recover)
  - 9. עבודה במתקן גיבוי
- יש להיזהר בהיסחפות בסעיף זה. בד"כ אין מערכת מידע מסוימת מגדירה מתקן גיבוי, אלא רק כיצד היא תתפעל במתקן גיבוי. וגם זה מקומו בד"כ בתיק התפעול ברכיב 4.4 לעיל.

#### זמינות

- קריטיות המערכת לפי מועדים, שעות, סוגי משתמשים וכו'.
- זמנים : Uptime ,MTBF (Mean Time Between Failures)
- יכולת לעבודה חלקית

#### אמינות והישרדות

- דרישות אמינות כללית (לא של פריט מסוים) ויכולת הישרדות.
- עמידה בתקנים

### 4.9 תצורות

ברכיב זה תצורה אינה במובן של ניהול תצורה למערכת (ניהול מהדורות), אלא הכוונה לפירוט התקנות שונות של המערכת, בקונפיגורציות וגדלים דומים אך שונים, בעקבות:

- ביזור המערכת (התקנות דומות במקומות שונים)
- התקנת המערכת בשלבים (אילוץ זמן, תקציב וכדומה).

ניהול תצורה למערכת (ניהול מהדורות, configuration management) מוסבר בהרחבה בקיט ניהול תצורה בכרך בנושאים תומכים.

בפרקים 2,3 לעיל, מוגדרים היישום והטכנולוגיה הבסיסיים אשר בעיקרון אינם משתנים מתצורה (התקנה) לתצורה (התקנה). אם יש הבדלים מהותיים בטכנולוגיה וביישום בתצורות השונות, אז אולי מדובר במערכות מידע שונות או בתת-מערכות שאז יש לפעול כמוסבר בפרק מערכות גדולות ותת מערכות. ההבדלים בין התצורות, להם מיועד רכיב 4.9 זה, מקורם העיקרי בלוח זמנים למימוש ומקצב ההתקנה(ות). כתוצאה מהבדלי מימוש אלה, נוצרים גם הבדלים משניים בטכנולוגיה וביישום ומכאן התצורות השונות.

רכיב זה, 4.9, משמש כתב כמויות של הקונפיגורציות המדויקות באתרים השונים בהם מותקנת המערכת. עם זאת, במקרים פשוטים יחסית, יכול פרק 3 להספיק ואין צורך ברכיב זה. בכל מקרה, מחשב הלקוח מכוסה ברכיבים 3.20-3.22 ולא כאן! שים לב שלרכיב זה קשר הדוק עם רכיבים: 2.21 (נפחים ועומסים), 4.2 (פיתוח כולל) ו- 5.3 (עלות לפי תצורות).

#### 4.9.0 רשימת תצורות (התקנות)

תיאור קצר הכולל:

- מקום
- מועד
- אחראי מקומי
- תצורה כללית.

#### 4.9.1 תצורת הפיתוח (והניסוי)

- חומרה, תוכנות, ספריות המערכת, שטחי עבודה, כלי פיתוח

#### 4.9.2 תצורה עיקרית (מרכזית, שרת ראשי)

1. חמרה ותשתית פיסית (בדומה לרכיבים 3.1 - 3.9)
2. תוכנת תשתית (בדומה לרכיבים 3.10 - 3.20)
3. תקשורת (רכיבים 3.30 - 3.32)
4. תוכנות יישומיות הפועלות עליה (ראה רכיב 2.3)
5. מימוש :

- תכנית התקנה מפורטת
- הדרכות, הטמעה מקומית
- תיק תפעול מקומי (אם שונה מ- 4.4 הכללי לעיל)
- שירות ותחזוקה מקומיים (4.6)

#### 4.9.x תצורות נוספות (ביזור, אתרים)

- פירוט התצורה / תכולה
- תכנית רכש והתקנה לאתר (לפי סוג אתר)

### 4.98 נקודות פתוחות (וחלופות)

סעיף זה מטרתו לרכז את כל הנקודות שנשארו פתוחות ברכיב המימוש. נקודות פתוחות הן לעתים קרובות חלופות (אלטרנטיבות) אפשריות ולפיכך גם החלופות השונות שתועדנה כאן. עם זאת, יש לבדוק היטב את שני המקרים הבאים :

- נקודות פתוחות (וחלופות) ברמת רכיב משני (4.X)
- חלופה מקיפה לרכיב המימוש בכללותו.

שני מקרים אלה, שהם השכיחים ביותר, לא יתועדו למעשה ברכיב זה. נקודות פתוחות וחלופות ברמת רכיב משני - מקומן ברכיב המשני (4.X.98) חלופה מקיפה לרכיב המימוש כולו - תתואר ע"י **תיעוד חוזר ומלא** של רכיב 4 כולו (4.0 - 4.9).

רכיב זה מיועד אם כך, למקרי ביניים, היינו, מקרים בהם החלופות והנקודות הפתוחות הן כלליות ואינן משנות את תמונת המימוש הכללית, או שעדיף לבחון אותן ברמה זו כיוון שהן משפיעות רק על קבוצה חלקית של רכיבי המימוש (4.1 - 4.9).

להרחבה נוספת ולהסבר מקיף בנושא חלופות, ראה קיט ניתוח חלופות בכרך נושאים תומכים.

בכל מקרה, חיוני לסגור סופית את הנקודות הפתוחות ולהכריע בין החלופות השונות ברכיב המימוש, אחרת אפיון המערכת לוקה בחסר. חשוב ביותר להשאיר בתיק תיעוד גם לנקודות שנסגרו ולחלופות שנדחו, על מנת לסייע בדיונים מחודשים בעתיד.

רכז כאן את הנקודות הפתוחות המרכזיות, ברכיבים 4.1 – 4.9 לעיל, שסגירתן היא חיונית להצלחת המהדורה הקרובה של המערכת:

1. רשימה כוללת ותמציתית

- הנקודה-הפתוחה/חלופה: שם או זיהוי אחר
- תאריך פתיחה
- תאריך סגירה.

2. תיאור מפורט של כל נקודה-פתוחה/חלופה (אם נדרש, או הפניה לסעיף 4.X.99 לעיל)

- תיאור הנקודה-הפתוחה/חלופה
- תאריך פתיחה
- תאריך סגירה
- קשר לרכיבי מימוש (ורכיבים אחרים בעץ המערכת)
- הגורם שסגר
- הפניה לסיכום דיון.

#### 4.99 תכניות עתידיות

סעיף זה מטרתו לרכז את כל הדרישות העתידיות ברכיב המימוש שאינן כלולות במהדורה הנדונה, אך סביר שיכללו במהדורות עתידיות. רישום דרישות (תחזיות) עתידיות אלה, הוא במטרה כפולה:

- לא לאבד מידע קיים
- להדגיש מה לא כלול במהדורה הנדונה ולסייע בכך לניהול תצורה של המערכת.

מיקומן של דרישות עתידיות אלה ברכיב ראשי (4) של עץ המערכת, הוא מכוון. אם יש מספיק מידע וצורך לרשום דרישות אלה ברמה מפורטת יותר, היינו, ברכיבי מימוש משניים (4.X), יש לבדוק היטב אם אין לפנינו הגדרה מלאה של המהדורה הבאה של המערכת. הגדרה כזו פירושה **תיעוד מלא** של רכיב 4 כולו ומקומה בתיק אפיון נפרד (של המהדורה הבאה).

תיעוד רכיב זה איננו עומד בפני עצמו ויבוא בד"כ רק אם הוגדרו הרכיבים המקבילים ביישום (2.99) ובטכנולוגיה (3.99). ציין כאן כל מידע מרכזי, ברכיבים 4.1 – 4.9 לעיל, שמשמעותו המשך המימוש במהדורות הבאות של המערכת.

1. רשימה כוללת ותמציתית:

- דרישה עתידית: שם או זיהוי אחר.

2. תיאור מפורט לדרישה עתידית

- תיאור הדרישה או הפניה לתיעוד אחר (תכנון אסטרטגי למשל)
- קשר לרכיבים x.99 אחרים

- הפניה לסיכום דיון.

## 5. עלות – משאבים

### 5.0 תמצית העלויות - הבהקים

בשלב ביניים אפשר לכלול ברכיב זה (בדומה ל-2.0, 3.0 ו-4.0 לעיל), נושאים ודגשים כלליים בעלות המערכת שחשוב לזכורם ומקומם הסופי לא ברור. במהלך בניית התיק יועברו נושאים אלה למקומם הטבעי באחד הרכיבים להלן (5.X). בשום פנים ואופן אין להפוך רכיב זה להערכת העלות המפורטת של המערכת! לכל מידע המופיע כאן יש מקום מוגדר ברכיבים להלן.

בצורתו הסופית, יכיל רכיב זה אך ורק עקרונות יסוד, תפיסה כללית וכו', נושאים שחשוב להדגישם ברמת הבהקים (Highlights) של העלות ולפי הדוגמא שלהלן.

עלות איננו רק רכיב שמטרתו להראות שאין ארוחות חינם ושכל דבר עולה בסופו של דבר כסף. עלות הוא גם רכיב אורתוגונלי אשר מסכם ועושה סדר. הגדרה נכונה של רכיב העלות היא בקרת איכות פנימית אשר חוזרת ובודקת שכל הרכיבים אופיינו נכון ונלקחו בחשבון.

בשלב האפיון נעשית לראשונה אמידת עלויות רצינית של המערכת. עם כל הקושי שבדבר, יש לבצע אמידת עלויות לפי מיטב השפיטה הקיים ברגע זה. בכל רכיבי העלות להלן ובפרט 5.1, 5.2, 5.5, העזר בקיטים חישוב עלויות ואמידת עלויות בכרך נושאים תומכים

#### דוגמא: (בלבד!)

1. ההערכה לעלות הקמת המערכת היא \_\_\_\_\_ . ראה פירוט ברכיב 5.1 להלן. ההערכה לעלות השנתית השוטפת לתפעול ולתחזוקה לתקופה של 5 שנים היא \_\_\_\_\_. ראה פירוט ברכיב 5.2 להלן.

2. בעוד שעלות הקמת המערכת נראית סבירה, עלויות התפעול והאחזקה השוטפים נראים גבוהים למדי. סיבה עיקרית לכך היא שאין בארגון תשתית מספקת לטיפול במערכות מידע מסוג זה.

3. הערכת העלות נעשתה ע"י חיתוך של הערכה ישירה לפי רכיבי עץ המערכת עם הערכה לפי התפלגות מחזור החיים. ריכוז העלויות נעשה לפי מודל ריכוז העלויות של מפת"ח.

#### סוף דוגמא

הצגה תמציתית של עלות המערכת, לדרג הניהולי, מבוססת על רכיב 5.5 להלן. סעיף זה, או תמצית רכיב 5.5 להלן, ישמש קלט לתמצית המנהלים.

### 5.1 עלות הקמה (פיתוח והתקנה)

רכיב זה יסכם את כל עלויות הפיתוח וההתקנה (עלויות הקמה) של מהדורות המערכת הצפויות, מהדורה ראשונה של המערכת וליחידות מסירה (מהדורות) נוספות.

עלויות הפיתוח וההקמה נחלקות לשתי קבוצות עיקריות:

- עלויות ישירות - בד"כ רכיבי יישום (2) וטכנולוגיה (3) שיש לבנותם, לרוכשם ולהתקינם.
- עלויות עקיפות - בד"כ רכיבי מימוש (4) שיש לבנותם, להנחילם ולהתקינם.

עם כל הקושי שבאמידת עלויות, בגמר האפיון יש כבר מידע לא מועט המאפשר אמידה לפי מיטב שפיטה ויש דרכים ושיטות. לעתים אין גם מנוס מהערכה פרטנית לפי רכיבי המערכת. אם עדיין קשה לתת הערכות לעלויות השונות, חובה לציין לפחות:

- חסם תחתון (סכום מינימלי לרכיב)
  - חסם עליון (סכום מקסימלי סביר שהארגון מוכן להוציא לרכיב).
- הטענה הנשמעת לעתים קרובות, בפרט במקרים בהם לאחר האפיון יש יציאה למכרז, שהספקים כבר יתנו הערכת מחיר, איננה מוצדקת משתי סיבות עיקריות:
- הארגון (המשרד) רוצה לדעת, לפני אישור היציאה למכרז, על איזה היקף פרויקט מדובר בכלל,
  - ללא הערכה פנימית של הארגון (המשרד) יקשה מאד להעריך את הצעות הספקים.

### 5.1.1 מהדורה ראשונה (קרובה)

במסמכי ייזום, תיקי אפיון ועיצוב: המהדורה הראשונה של המערכת. בתיקי תחזוקה: מהדורת הפיתוח הבאה.

### עלות פיתוח (כולל בדיקות ותיקונים)

1. הוצאות ישירות (בפרויקט): כ"א פנימי וחיצוני, הוצאה כספית. אופן הפירוט:

- הערכה כוללת
- הערכה כוללת ליישום
- הערכה כוללת לטכנולוגיה
- פירוט לפי רכיבים: רכיבי היישום, רכיבי הטכנולוגיה, אחר.

2. הוצאות עקיפות (בפרויקטים אחרים, בארגון):

- הדרכה והכשרת עובדים: מס' העובדים, שעות הדרכה לעובד
- לימוד תוך כדי עבודה (on-the-job training)
- הסבות (נתונים, שיטות עבודה)
- השלמת התייעוד התפעולי והפצתו
- התארגנות כוללת במשרד (שעות ניהול ובקרה).

### עלויות התקנה והטמעה

- הוצאות ישירות (בפרויקט): כ"א פנימי וחיצוני, הוצאה כספית
- הוצאות עקיפות (בפרויקטים אחרים, בארגון): כ"א, הוצאה כספית?

### 5.1.2 יחידות מסירה ומהדורות נוספות

ברכיב זה יש לתת הערכה, המיטבית האפשרית, של עלויות המשך הפיתוח: מהדורות ויחידות מסירה נוספות באופן הזמן המוגדר (רכיב 1.7 לעיל). ניתן לתעד רכיב זה באחד משני האופנים הבאים:

#### אופציה א:

במרוכז לכל היחידות והמהדורות הצפויות באופן הזמן המוגדר. הפירוט הבא ייכתב ישירות באופן הבא:

### 5.1.2 יחידות מסירה ומהדורות נוספות

## 1. עלות פיתוח

- הוצאות ישירות (בפרויקט) : כ"א פנימי וחיצוני, הוצאה כספית
- הוצאות עקיפות (בפרויקטים אחרים, בארגון ומחוצה לו) : כ"א, הוצאה כספית?

## 2. עלויות התקנה והטמעה

- הוצאות ישירות (בפרויקט) : כ"א פנימי וחיצוני, הוצאה כספית
- הוצאות עקיפות (בפרויקטים אחרים, בארגון) : כ"א, הוצאה כספית?

## אופציה ב:

לפי יחידות מסירה צפויות. חלוקה לתת-רכיבים באופן הבא :

### 5.1.2.1 יחידת מסירה א

- חלוקה לעלויות פיתוח, התקנה והטמעה, ישירות ועקיפות כמפורט לעיל.

### 5.1.2.2 יחידת מסירה ב

- חלוקה לעלויות פיתוח, התקנה והטמעה, ישירות ועקיפות כמפורט לעיל.

## 5.2 עלות שוטפת

ברכיב זה ירוכזו כל העלויות הצפויות בתפעול ובייצור השוטף של המערכת (running costs) ובתחזוקתה (maintenance cost), בהתאמה להערכה שניתנה לגבי אופק הזמן של המערכת, אופן המימוש הכללי, מהדורות עתידיות וכו'. ברירת המחדל היא תקופה של 5 שנים!

עם כל הקושי שבהערכת עלות התחזוקה, בגמר האפיון יש כבר מידע לא מועט המאפשר אמידה לפי מיטב שפיטה ויש דרכים ושיטות. יש גם כללי אצבע ידועים, ראה קיט תפעול ותחזוקה בכרך מחזור חיים. היעזר גם בקיט אמידת עלויות בכרך נושאים תומכים. לעתים אין גם מנוס מהערכה פרטנית לפי רכיבי המערכת. אם עדיין קשה לתת הערכות לעלויות השונות, חובה לציין לפחות :

- חסם תחתון (סכום מינימלי להוצאה שוטפת לרכיב או לכלל המערכת)
  - חסם עליון (סכום מקסימלי סביר שהארגון יקציב להוצאה שוטפת לרכיב או לכלל המערכת)
- דרישה רצויה ביותר היא שהמערכת תמדוד ותתמחר את עצמה באופן שוטף (מערכת חשבונאות או תמחיר פנימית), אך יש לשקול כבר באפיון אם דרישה כזו היא סבירה למערכת הנדונה וגם אם כן, יש לבדוק אם אין מערכת כללית כזו בארגון שאפשר לקבל ממנה את החלק היחסי של המערכת.
- עלויות שוטפות, תפעול, תחזוקה ותיקון תקלות שוטף לאופק הזמן שנקבע, בהתאם לרכיב 1.7 לעיל. ברירת המחדל של מפת"ח היא תקופה של 5 שנים.

### 5.2.1 מהדורה ראשונה (קרובה)

בשני סוגי עלויות אלה, תפעול ותחזוקה, יש לכלול לא רק כ"א (לסוגיו השונים), מחשב, תוכנה, תקשורת, מקום וכו', אלא גם פקס/טלפון, ציוד מתכלה, צרכי משרד, נסיעות וכו'. העלויות יחולקו עפ"י עלויות תפעול ועלויות תחזוקה מתקנת ומונעת, כשתת החלוקה היא לפי הוצאות שוטפות ועקיפות.



במקרים בהם יש תפעול מרכזי בארגון, עלויות התפעול הן התוספת שהמערכת (המהדורה במערכת) גורמת לתפעול המרכזי. תחזוקה מבוצעת בד"כ ע"י יחידה ייעודית ולפיכך ההוצאה היא עצמאית.

#### **הוצאות שוטפות ישירות (תחזוקה ישירה)**

1. הערכה כוללת
2. פירוט לפי רכיבים:
  - רכיבי היישום
  - רכיבי הטכנולוגיה (אחזקה חיצונית?)
  - רכיבי מימוש (!)
  - חומרים כלים, הוצאות ניהול ובקרה כלליות, אחרות.

#### **הוצאות שוטפות עקיפות**

- הכשרת עובדים נוספים: כמה, סוג הכשרה
- לימוד מתמיד של המערכת
- ניהול שוטף
- יצירת תיעוד תפעולי נוסף והפצתו

#### **5.2.2 יחידות מסירה ומהדורות נוספות**

ברכיב זה יש לתת הערכה, המיטבית האפשרית, של עלויות התפעול והתחזוקה של כל המהדורות ויחידות מסירה נוספות באופק הזמן המוגדר (רכיב 1.7 לעיל).

דרישת מפת"ח הבסיסית היא להציג את כל עלויות התפעול והתחזוקה במרוכז לכל היחידות והמהדורות הצפויות באופק הזמן המוגדר, בדומה לאפשרות א ברכיב 5.1.2 לעיל. אם ניתן לחלק רכיב זה לפי יחידות מסירה בדומה לאפשרות ב ברכיב 5.1.2 לעיל – מה טוב.

#### **5.3 עלות לפי תצורות**

רכיב זה ייבנה בתיאום מדויק עם רכיב 4.9 לעיל ויכלול, בחלוקה לתצורות, את כל עלויות ההתקנה וההטמעה (מתוך רכיב 5.1) ותפעול ותחזוקה (מתוך רכיב 5.2), כך שתתקבל עלות לתצורה. בתוך כל תצורה יש להביא בחשבון גם מהדורות עתידיות.

- תצורה
- מהדורה
- עלויות התקנה והטמעה
- עלויות תפעול ותחזוקה

רכיב זה איננו מוסיף עלויות בהשוואה עם רכיבים 5.1 ו-5.2, אלא רק מוציא מתוכן את העלויות בגין תצורות שונות של המערכת, בפרט מערכות המותקנות בהדרגה באתרים שונים. כל העלויות האלה נמצאות כבר ברכיבי 5.1 (התקנה והטמעה) ורכיב 5.2 (תפעול ותחזוקה).

## 5.4 מחירון

רכיב זה נכתב במקור לשלב הבקשה להצעות, קרי למפרט, ומיועד לספקים לשם פירוט עלויות לפי רכיבים בדידים (מחירון!), אך יש לו חשיבות גם לשלב האפיון.

בשלב האפיון יש לבנות ברכיב זה רשימה של רכיבי מערכת שאפשר לראותם כחלקי חילוף או upgrades או סביר שיסופקו ע"י ספקים שונים. בקיצור, כל רכיב שיש טעם לדרוש את עלותו בנפרד.

יש לקחת בחשבון תחזיות ומגמות במערכת לתקופה של 5 שנים.

ברכיב זה במיוחד, יש להיעזר במערכת מצאי שבארגון. במשרדי ממשלה: יש להיעזר במערכת הממשלתית (מרכבה). בארגונים אחרים: בדוק קיומה של מערכת מצאי דומה ונסה להיעזר בה.

בשלב הפיכת מסמך האפיון למפרט (RFP) ישוכתב רכיב זה באופן שיבהיר את דרישות המערכת בנושא המחירון. ראה פירוט מוצע והנחיות נוספות ברכיב זה בקיט תפעול ותחזוקה בכרך מחזור חיים.

פריט	מחיר יחידה	כמות	מחיר כולל
רכיבי חומרה			
תוכנות מדף			
רכיבי תקשורת			
אחסון ותחזוקה			
הוספת דו"ח			
הוספת קובץ \ דף			
ש"ע מנהל			
ש"ע מנתח מערכות			
ש"ע מתכנת			
ייעוץ כללי			
ייעוץ ניהולי			
ייעוץ טכני			

## 5.5 עלות כוללת ופריסה

קרא תחילה בעיון את הקיט אמידת עלויות בכרך נושאים תומכים. שים לב להבדל המהותי של רכיב זה (למעשה רוב רכיבי 5.X) בין שלב האפיון לשלב בקשה להצעות. באפיון, מטרת הרכיב היא לתת לארגון הערכת מחיר של המערכת (טבלאות מלאות). בשלב בקשה להצעות, יש להנחות בבירור את הספקים איך להגיש את ריכוז העלות (טבלאות ריקות).

### 5.5.1 סה"כ עלות – COST OF OWNERSHIP

- נומינלית
- מהוונת

### 5.5.2 פריסה

- לפי שנות תקציב

- לפי צפי הכנסות \ תועלות

## 5.98 נקודות פתוחות (וחלופות)

רכיב זה מטרתו לרכז את כל הנקודות שנשארו פתוחות ברכיב העלות. נקודות פתוחות הן לעיתים קרובות חלופות (אלטרנטיבות) אפשריות ולפיכך גם החלופות השונות שתועדנה כאן. עם זאת, יש לבדוק היטב את שני המקרים הבאים:

- נקודות פתוחות (וחלופות) ברמת רכיב משני (5.X)
- חלופה מקיפה לרכיב העלות בכללותו.

שני מקרים אלה, שהם השכיחים יחסית, לא יתועדו למעשה ברכיב זה. נקודות פתוחות וחלופות ברמת רכיב משני - מקומן ברכיב המשני (5.X.98). חלופה מקיפה לרכיב העלות כולו - תתואר ע"י **תיעוד חוזר ומלא** של רכיב 5 כולו. רכיב זה מיועד אם כך, למקרי ביניים, היינו, מקרים בהם החלופות והנקודות הפתוחות הן כלליות ואינן משנות את תמונת העלות הכללית.

חשוב ביותר להשאיר בתיק תיעוד גם לנקודות שנסגרו ולחלופות שנדחו, על מנת לסייע בדיונים מחודשים בעתיד. היעזר בקיט ניתוח חלופות שכרך בנושאים תומכים.

אופן רישום הנקודות הפתוחות:

### רשימה כוללת ותמציתית

- הנקודה-הפתוחה/חלופה: שם או זיהוי אחר
- תאריך פתיחה
- תאריך סגירה.

### תיאור מפורט לנקודה-פתוחה/חלופה

- תיאור הנקודה-הפתוחה/חלופה
- תאריך פתיחה
- תאריך סגירה
- קשר לרכיבי עלות (ורכיבים אחרים בעץ המערכת)
- הגורם שסגר
- הפניה לסיכום דיון.

## 5.99 צפי עלויות עתידיות

סעיף זה מטרתו לרכז את כל התחזיות העתידיות ברכיב העלות שאינן כלולות במהדורה הנדונה, אך סביר שיכללו במהדורות עתידיות. רישום תחזיות (דרישות) עתידיות אלה כאן, הוא במטרה כפולה:

- לא לאבד מידע קיים
- להדגיש מה **לא** כלול במהדורה הנדונה ולסייע בכך לניהול תצורה של המערכת, **גם דרך רכיב העלות.**

מיקומן של תחזיות עתידיות אלה ברכיב ראשי (5) של עץ המערכת, הוא מכוון. אם יש מספיק מידע וצורך לרשום תחזיות אלה ברמה מפורטת יותר, היינו, ברכיבי עלות משניים (5.X), יש לבדוק היטב

אם אין לפנינו הגדרה מלאה של המהדורה הבאה של המערכת. הגדרה כזו פירושה **תיעוד מלא** של רכיב 5 כולו ומקומה בתיק אפיון נפרד (של המהדורה הבאה).

תיעוד רכיב זה איננו עומד בפני עצמו ויבוא בד"כ רק אם הוגדרו הרכיבים המקבילים ביעדים (1.99), ביישום (2.99), בטכנולוגיה (3.99) ובמימוש (4.99).

שים לב לרכיב 5.2 לעיל, בפרט תת-רכיב 5.2.2. הפנה לרכיב זה וציין כאן רק מידע נוסף באשר לעלויות עתידיות שעשויות להתעורר בהמשך חיי המערכת, מעבר לאופק הזמן שברכיב 5.2 הנ"ל.

אופן רישום עלויות עתידיות:

#### **רשימה כוללת ותמציתית**

תחזית עתידית: שם או זיהוי אחר

#### **תיאור מפורט לתחזית עתידית**

- תיאור הדרישה או הפניה לתיעוד אחר (תכנון אסטרטגי למשל)
- קשר לרכיבים X.99 אחרים
- הפניה לסיכום דיון.

## נספחים

בדומה להנחיה בראש כל פרק (רכיבים 2.0, 3.0 ו-4.0 וכ"ו), יש להיזהר מאד מלהכליל בנספחים נושאים שמקומם בתיק האפיון עצמו. זכור: עץ המערכת הוא מקיף ביותר ולכל נושא יש כתובת! לנספחים יש שתי מטרות עיקריות:

- תיוק המסמכים (סכומי דיון, למשל) מהם ניתן ללמוד מדוע אופיינה המערכת כפי שמתועד בתיק זה, ומדוע הועדפה חלופה מסוימת על פני חברותיה. זאת, על מנת שבעתיד לא יוצעו (ויבוצעו!) רעיונות שכבר נידונו ונדחו.
- לקצר את רכיבי התיק עצמו ע"י כך שכל סעיף X.Y שמתנפח יכיל תמצית בלבד (אינדקס וכ"ו"ב) והפניה לנספח X.Y מפורט.

במערכות בהן תיק התחזוקה מנוהל בשיטת מפת"ח והאפיון הוא למהדורה חדשה, ייתכן וחלק מהנספחים נמצא כבר בתיק התחזוקה ואין צורך בכתבתם. ההחלטה אילו נספחים יש להעביר לתיק זה ואילו להשאיר במקומם, היא בידי הפרויקט.

ככלל, לכל רכיב בעץ המערכת ייתכן נספח, עם אותו שם וסימול מספרי, אשר מרחיב ומפרט את אותו רכיב. גוף המסמך (עמוד השדרה של עץ המערכת) יהיה בעל נפח סביר (עד 100 עמודים) שניתן לקרוא אותו ברצף ואשר מפנה לנספחים במקומות המתאימים.

**במפת"ח אין נספחים א', ב' וכ"ו. כל נספח הוא הרחבה של רכיב בעץ המערכת.**

להלן רשימת נספחים שכיחים:

### נספח 1.6.1 - ניתוח סיכונים וחקר ישימות

היעזר בקיט ניתוח סיכונים שבכרך נושאים תומכים.

### נספח 1.6.2 - עלות\תועלת

### נספח 2.7.1 - פירוט תוכניות מקור

נספח זה מובא כאן לצורך הסבר והמחשה והוא מיועד עבור מקרים בהם אכן בונים תיקי תכנות מסורתיים. בפועל, יהיו תיקי התכנות בתוך המודול הפיסי, ישוכנו ישירות בספריות המודולים וייבנו ע"י כלי הפיתוח של שלב העיצוב, הבנייה והקידוד!

להסבר בנושא תיק תכנות וגלופות לימוד ועבודה, ראה קיט עיצוב ובניה בכרך מחזור חיים.

### נספח 4.2 - פירוט תכנית העבודה

1. תרשימי גאנט מפורטים

2. רשימות משימות:

• לצוותים

• אישיות

3. עומס \ ניצול משאבים

### **נספח 5.1 - אמידת עלויות הקמה**

פירוט הטבלאות והחישובים מהם סוכמו העלויות ברכיב 5.1 לעיל.

### **נספח 98 - נקודות פתוחות - ניתוח חלופות**

ראה קיט ניתוח חלופות בכרך נושאים תומכים.

### **נספח 99 - ריכוז דרישות עתידיות**

### **נספחי Y.X**

פירוט נוסף לכל רכיב שמתפוצץ בגוף התיק. לדוגמא, נספח 2.21 המפרט טבלאות עומסים ונפחים.