

מתודולוגיות בדיקות תוכנה - חלק ב

בדיקות תוכנה שיעור 4

מחזור חיי פיתוח תוכנה ושילוב
הבדיקות

מה נלמד בשיעור?

אפיון ודרישות

מחזור חיי מערכת

מחיר התקלות בשלבי הפרויקט השונים

השתלבות הבדיקות

סוגי מערכות

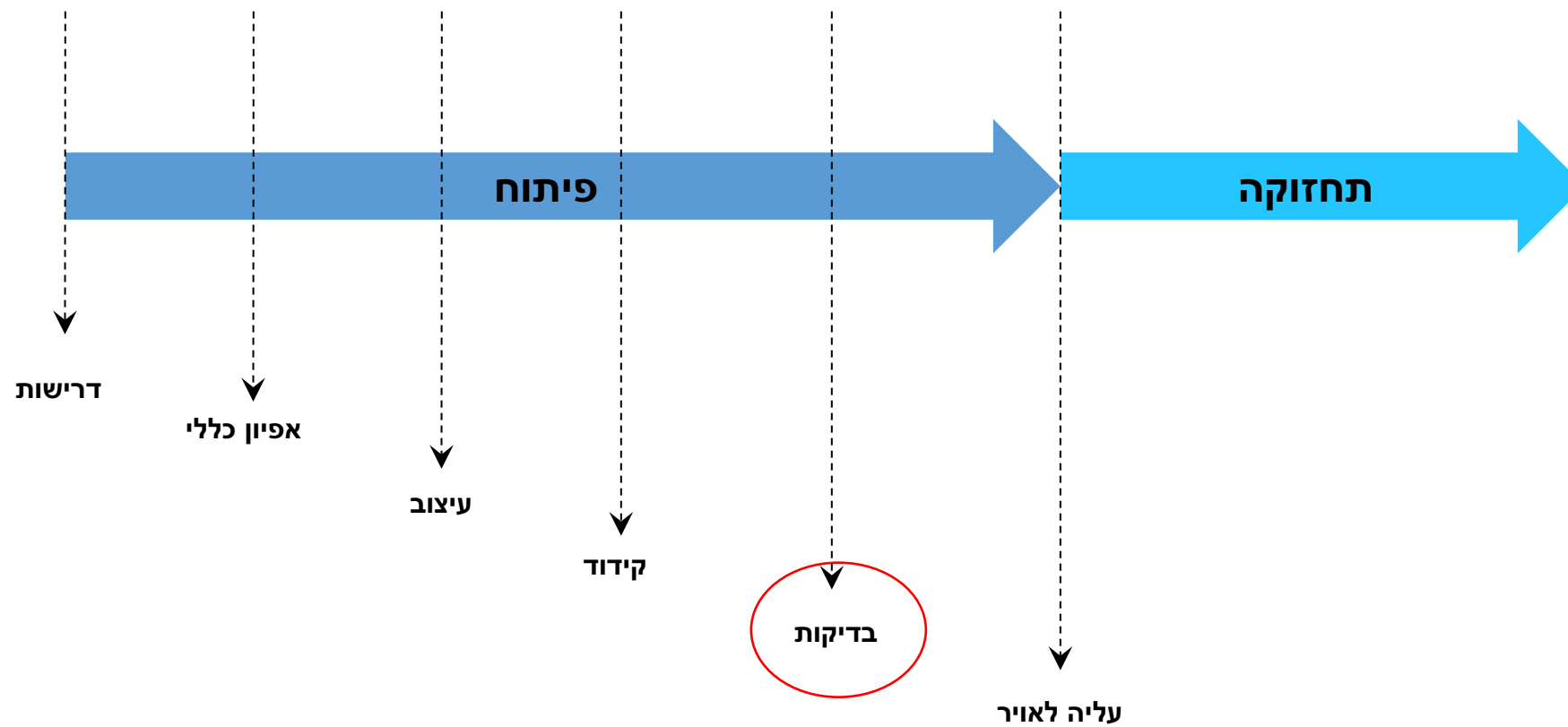
איך בודקים?



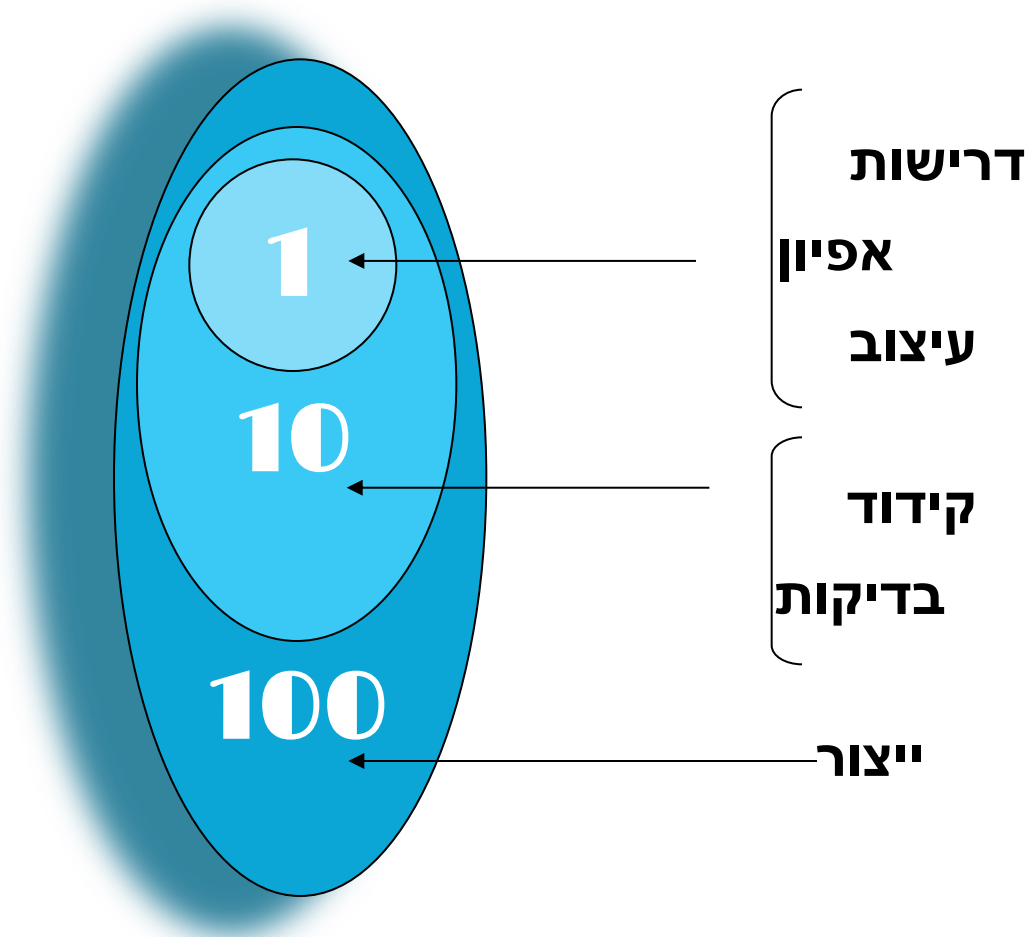
דוגמה – אפיון 1

מערכת דיווח שעות

מחזור חיי מערכת



מחיר התקלות בשלבי הפרויקט



מערכות מידע/עיבוד נתונים (IT- Information Technology)

- מערכות המבוססות על נתונים לצורך שליפה והצגת מידע וכן לצורך עדכון נתונים
- אפיון המערכות נעשה בראיה מרחבית להתאמתן לארגונים רבים
- לדוגמא: מערכות כספים, משאבי אנוש וכו'.

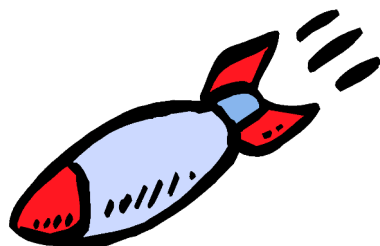


איך נבדוק?

- ההתמקדות היא בדיקת התהליכים העסקיים כפי שהוגדרו ע"י הארגון
ופחות בדיקת התוכנה עצמה!
- במקום מסמך אפיון – מסמך **Blue print**

מערכות זמן אמת/משובצות (Real-time\Embedded)

- מערכות מחשב שיש בהן דרישות לביצועים במגבלות זמן.
- תכונות מיוחדות:
- אינן מבוססות על מאגרי מידע אלא על מכשור.
- ציוד יקר ולעיתים מסוכן
- לדוגמא: מערכת השולטת על טילים וכו'.



איך נבדוק?

- הכנות מורכבות
- הוספת קוד זמני למערכת שיוציא פלט חשוב לצורך בקרה
- בניית סימולטורים

מערכות שו"ב – שליטה ובקרה (Command&Control)

- מערכות המאפשרות לדרג הניהולי:
- קבלת מידע ממקורות שונים
- הצגת תמונת מצב אחודה של המערכת בכל זמן נתון (פעילות השרתים, תקינות מסד הנתונים).
- שליטה על פעולות של הרכיבים השונים (הפעלת שרת, שליחת הודעות וכו')
- דוגמא: מערכת לשליטה ימית (עמית-טורף), מערכת לניטור שרתים.

איך נבדוק?

- בניית סביבת עבודה הכוללת את כל הרכיבים הנדרשים
- התמקדות בתקשורת בין הרכיבים ואמינות המידע המשתקף

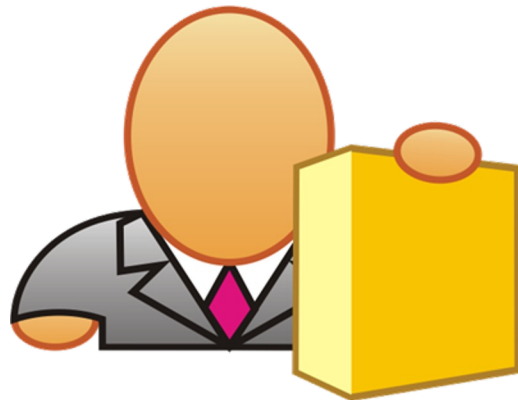
תוכנות מדף

- תוכנה המופצת לכל רוכש כפי שהיא.
- אין לקוח מוגדר
- לדוגמא: מערכת ניתוב שיחות.

איך נבדוק?

יש לשים דגש מיוחד על:

- בדיקות התקנה
- בדיקות תאימות



מערכות בתחזוקה

מערכת ששלב הפיתוח שלה הסתיים והיא הועברה לשלב הייצור ושימוש שוטף ע"י המשתמשים.



איך נבדוק?

נתייחס לשאלות הבאות:

- מה היו השינויים שבוצעו במערכת
- מה היו התוספות למערכת
- אילו תקלות תוקנו במערכת
- מהן ההשפעות של השינויים והתוספות על חלקי המערכת האחרים

תרגיל 1 – סוגי מערכות

אז מה למדנו?



- ✓ מה ההבדל בין מסמכי דרישות ואפיון?
- ✓ מהם השלבים במחזור חיי מערכת?
- ✓ מחיר התקלות בשלבים השונים
- ✓ על אילו סוגי מערכות דיברנו?
- ✓ האם קיים שוני בבדיקת מערכות שונות?

מה נלמד במפגש הבא?



✓ בדיקות יחידה

✓ בדיקות אינטגרציה

✓ בדיקות מערכת

✓ בדיקות מסירה

✓ בדיקות רגרסיה

✓ בדיקות שפיות

✓ בדיקות Monkey

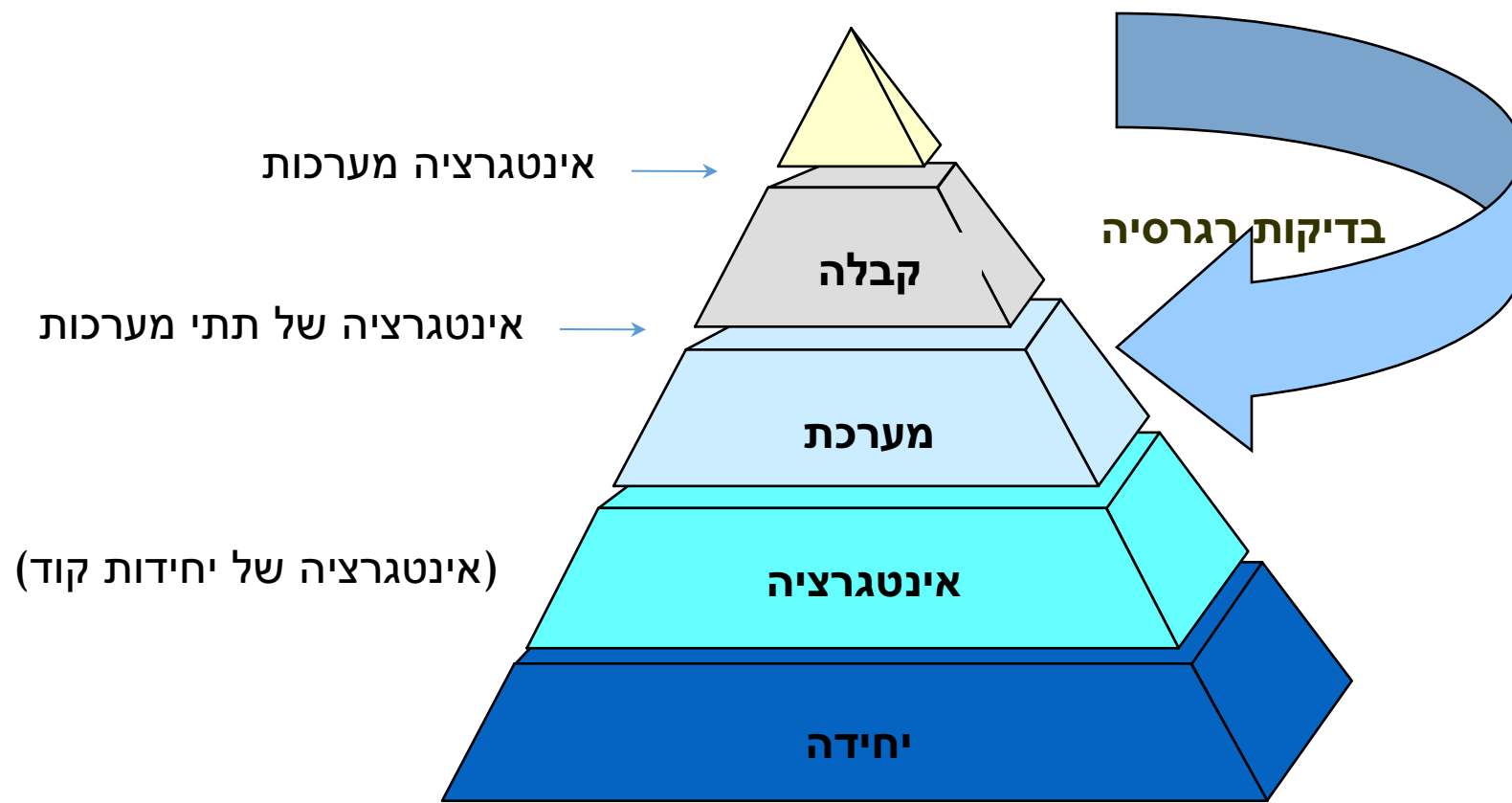
QA?

בדיקות תוכנה שיעור 5

מושגים בבדיקות תוכנה

מה נלמד בשיעור?

	בדיקות יחידה	
	בדיקות אינטגרציה	
	בדיקות מערכת/מסירה	
	בדיקות קבלה	
	בדיקות רגרסיה	
	בדיקות שפיות	
	Monkey testing	



סיכום רמות הבדיקה

מבחני קבלה	אינטגרציה (ברמת האפליקציה) / מערכת	יחידה / אינטגרציה (ברמת קוד)	
בודקים	בודקים	מפתחים	אחריות
להבטיח תפקוד המערכת בהיבט המשתמש	להבטיח תפקוד הרכיבים ביחד	להבטיח תפקוד כל רכיב תוכנה בנפרד	מטרה
קופסה שחורה	קופסה אפורה / שחורה	קופסה לבנה	שיטה
מסמכי דרישות	מסמכי דרישות ואפיון	מסמכי אפיון מפורט	על בסיס
דורש הכרות ברמת המשתמש וברמת התהליכים העסקיים של הארגון	דורש הכרות המערכת עד רמת קבצים / טבלאות	דורש הכרות ברמת שורות קוד	דרישות
לא מאפשרת זיהוי אי יעילות בקוד. מזהה אי-התאמות לדרישות לקוח.	לא מאפשרת זיהוי אי יעילות הקוד. מזהה אי-התאמות לדרישות לקוח.	מאפשרת זיהוי ליקויים בקוד. לא מזהה אי-התאמות לדרישות לקוח.	מעלות/מגרעות

רמות בדיקה נוספות

קיימות רמות בדיקה נוספות:

- בדיקות רגרסיה
- בדיקות שפיות
- Monkey Testing

Monkey	שפיות	רגרסיה	
בודקים	מפתחים / בודקים	מפתחים / בודקים	אחריות
אין	קופסה שחורה	קופסה לבנה / שחורה	שיטה
אין	בדיקות אחרות	בדיקות אחרות / תקלות	על בסיס
מאפשרת זיהוי תקלות שיתכן שלא יתגלו בבדיקות המתודיות. לא מכסה את כלל המערכת.	מאפשרת להבטיח את "בדיקתיות" המערכת ומעברה התקין לסביבת הבדיקות.	מאפשרת לזהות תקלות חדשות שנגרמו במהלך תיקון תקלות אחרות	מעלות/מגרעות

תרגיל 2 – רמות בדיקה

אז מה למדנו?

מה תוכלו לספר על:

✓ בדיקות יחידה?

✓ בדיקות אינטגרציה?

✓ בדיקות מערכת?

✓ בדיקות קבלה?

✓ בדיקות רגרסיה?

✓ בדיקות שפיות?

✓ בדיקות Monkey?

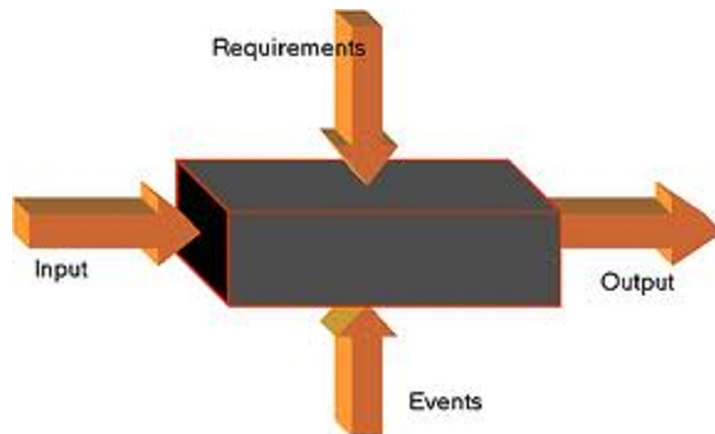


מה נלמד במפגש הבא?

✓ מהן בדיקות פונקציונליות?

✓ מהן בדיקות לא-פונקציונליות?

✓ טכניקות שונות לביצוע בדיקות



QA?

בדיקות תוכנה שיעור 6

סוגי בדיקה

מה נלמד בשיעור?

בדיקות פונקציונליות

בדיקות לא פונקציונליות

טכניקות לבדיקות

תרגיל

בדיקות פונקציונאליות – תקינות המערכת (Functional)

- **בדיקות ממשק משתמש - GUI (Graphical User Interface)** –
בדיקות חוקיות שדות, בדיקות מסכי המערכת אל מול הגדרות
האפיון והדרישות
- **בדיקות תקינות פונקציונאלית** – תקינות תפעול מסכי המערכת.
- **בדיקות תהליכיות (Process Tests)** – בדיקת תהליכים מבצעיים
וטכניים מלאים.

בדיקות פונקציונליות - המשך (Functional)

- **בדיקות התאוששות מנפילות (Failure & Recovery Tests)** – מתקלות כוח, תקשורת ותוכנה. ברכיבי המערכת ובסיס הנתונים.
- **בדיקת הרשאות** - בדיקת אכיפת פעולות למשתמשים מסוגי שונים.
- **בדיקת ממשקים** – בדיקות אל מול מערכות משיקות
- **במ"מ** – בדיקת עמידות המערכת בניסיונות חדירה

בדיקות לא פונקציונליות (Non-Functional)

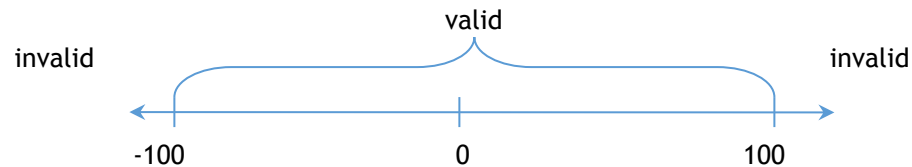
- **בדיקות גיבוי ושחזור (Backup & Recovery Tests)**
- **בדיקות התקנה (Installation Tests)** – תקינות ערכת ההתקנה
- **בדיקת תאימות (Compatibility Tests)** – תקינות התפקוד בסביבות עבודה שונות.
- **בדיקות הסבת נתונים (Data Conversion Tests)**
- **בדיקות עומסים וביצועים (Load & Stress Tests)** – מתבצע עפ"י מסק המשתמשים וסוגי הפעילויות. בדיקת יכולת גידול.
- **בדיקות שרידות – MTBF, אתר DRP**

תרגיל 3– זיהוי סוג בדיקה

מחלקות שקילות

זיהוי מחלקות של תנאי קלט ופלט לפי הכלל:
כל חבר באותה מחלקה יגרום לאותה תגובה במערכת.

המטרה - צמצום מקרי הבדיקה



Case 1: Invalid $X < -100$

Case 1: Valid $-100 \leq X \leq 100$

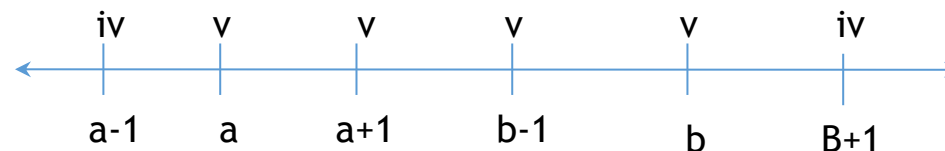
Case 1: Invalid $X > 100$

בדיקת גבולות/ערכי קצה (Boundary Testing)

בחינת הגבולות, התרכזות סביב ערכי הקצה ולא טווחי נתונים.

המטרה - בדיקת שדות שיכולים לכלול ערכים בטווח מסוים.

דוגמה: שדה שמקבל ערכים $a \rightarrow b$



טבלאות החלטה (Decision Table)

בניית טבלאות המכילות את כל הקומבינציות האפשריות במערכת.

המטרה -

כיסוי כלל האפשרויות, ביצוע מינימום בדיקות בדומה לשיטת מחלקות השקילות.



טכניקות לבדיקות - המשך

דוגמא:

נתונה מערכת אשר קובעת איזו הטבה מגיעה לעובד על פי קריטריונים של ותק, גיל ותפקיד. לשם הדוגמא, נתונים ההחלטות על שווי ההטבה בטבלה הבאה.

מצב 8	מצב 7	מצב 6	מצב 5	מצב 4	מצב 3	מצב 2	מצב 1	
-	-	-	-	+	+	+	+	2 שנות ותק ומעלה
-	-	+	+	-	-	+	+	מעל גיל 40
-	+	-	+	-	+	-	+	תפקיד ניהולי
								שווי הטבה
+				+				ארוחה זוגי
	+	+				+		שבוע נופש
			+		+		+	טיסה לחו"ל

על פי טכניקה זו נבצע סה"כ 8 בדיקות, אחת לכל מצב. לצורך הדוגמא, במצב 1, אנו יכולים לבדוק עפ"י אדם עם 4 שנות ניסיון, בן 42 בתפקיד ניהולי. התוצאה הרצויה הינה הטבה של טיסה לחו"ל. באותה מידה יכולנו לבצע בדיקה על אותו מצב עם אדם בעל 3 שנות ניסיון, בן 65 בתפקיד ניהולי בכיר.

טכניקות לבדיקות - המשך

מכונת מצבים

בניית דיאגרמה או טבלה בה מוצג תהליך הבדיקה. באופן זה, מוצגים באופן ויזואלי כל הקשרים בין התהליכים במערכת ומוצגים בבירור מעברים אפשריים ובלתי אפשריים.

המטרה - שימושי בעיקר לבדיקות אוטומטיות (ילמד בהמשך) ולבדיקת מערכות בעלות מס' מסכים כגון אפליקציות אינטרנט.



טכניקות לבדיקות - המשך

דוגמא: נדמה הרשמה לאתר אינטרנט.

	לחיצה על הרשמה	מילוי פרטים	לחיצה על אישור במייל- לקוח חדש	לחיצה על אישור ההרשמה
A- דף ראשי	B			
B- דף פתיחת לקוח		C		
C- שליחת מייל			D	
D- הודעת שגיאה			E	
E- לקוח חדש נפתח				A

ניתן לראות את ה-Flow של המערכת והמעבר בין האפשרויות השונות. על פי הטבלה ניתן לראות כי תהליך זרימת המערכת והמעבר לאחר שליחת המייל תלוי האם הלקוח קיים או שזהו לקוח חדש. כמו כן ניתן לראות כי לא קיימת אפשרות של מעבר מהדף הראשי להודעת שגיאה או לשליחת מייל.

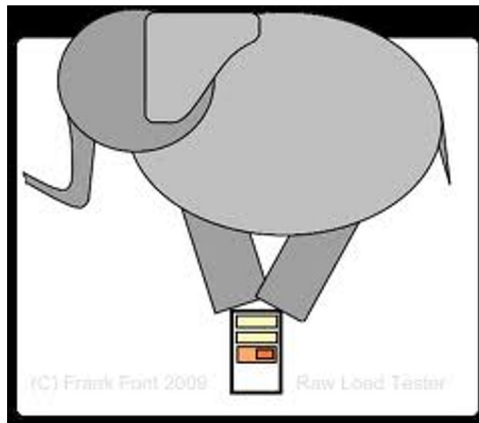
/בעזרת הטבלה תהליך הזרימה הרבה יותר מסודר ומאורגן לטובת תכנון הבדיקות

תרגיל 4– טכניקות בדיקה

✓ מהן הבדיקות הפונקציונליות? ציינו אותן.

✓ מהן הבדיקות הלא פונקציונליות? ציינו.

✓ על אילו טכניקות בדיקה דיברנו?



מה נלמד במפגש הבא?



✓ כיצד ניתן לנהל את האיכות

✓ מדידת איכות

✓ סטנדרטים

✓ סקר תוצרים

✓ ניהול סיכונים

✓ סקירת קוד

✓ בקרת שינויים

QA?

בדיקות תוכנה שיעורים 7-8

הבטחת איכות תוכנה

מה נלמד בשיעור?

	בעיות אפשריות במערך פיתוח התוכנה	
	מדידת איכות	
	סטנדרטים	
	סקר תוצרים	
	סקירת קוד	
	ניהול סיכונים	
	בקרת שינויים	
	ניהול גרסאות	

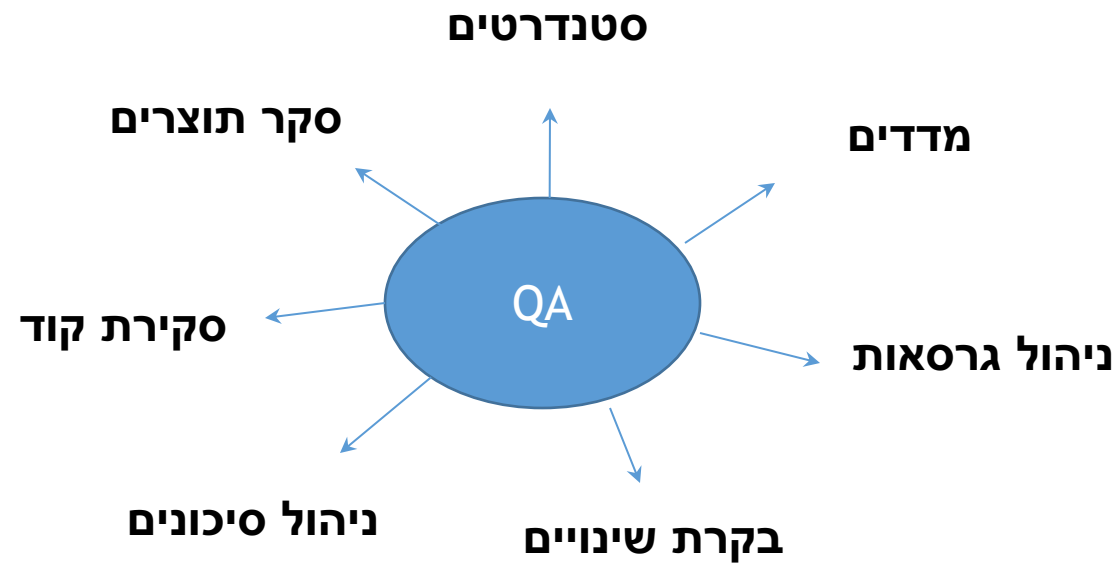
בעיות במערך פיתוח תוכנה

קיימות בעיות אפשריות רבות בכל אחד משלבי פיתוח התוכנה,
משלב הייזום, לפיתוח ועד לייצור והתחזוקה.

- ✓ בעיות תקשורת
- ✓ חוסר סטנדרטיזציה של מסמכים
- ✓ הבדיקות לא מתועדות
- ✓ אין מעקב אחר שינויים בגרסאות
- ✓ בעיות נוספות?

האם ניתן למנוע בעיות אלה?

אפשרויות פעולה לניהול איכות



מדד - ערך מספרי המייצג מצב או אירוע, לשם הערכה

שיטת GQM (Goal Question Metric) למציאת מדדים לשיפור תוצר/תהליך:

1. הגדרת יעדים
2. הגדרת שאלות להשגת היעדים
3. זיהוי מדדים שיתנו תשובות לשאלות
4. מעקב, ניתוח וטיפול

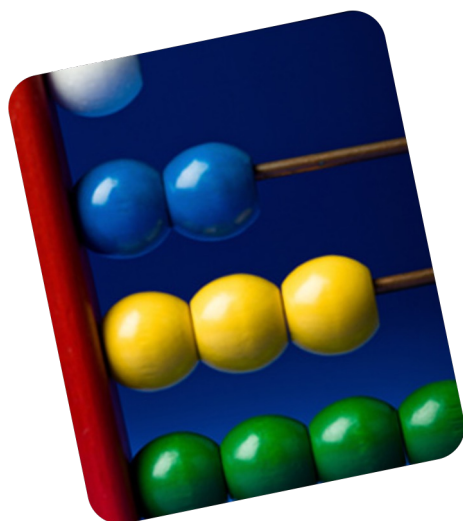


שיטה להגדרת מדדים לצורך החלטה:

1. הגדרת גבולות היעד
2. מדידת מצבי ה"אמת"
3. השוואת המדידות לגבולות שנקבעו

דוגמאות למדדים:

- מדדים לאיכות מסמכי הדרישות והאפיון
 - ✓ אחוז המשפטים הכלליים במסמך
 - ✓ זמן המושקע בתיקוני קוד בגין אי הבנת האפיון
- מדדים המציגים את איכות הקוד
 - ✓ אחוז העמידה בסטנדרטים
 - ✓ כמות הזמן המושקע בביצוע תיקון לקוד
- מדדים המציגים את איכות המערכת
 - ✓ צפיפות תקלות
 - ✓ מס' התקלות הפתוחות
- מדדים המציגים את איכות הבדיקות
 - ✓ אחוז כיסוי הבדיקות



למה צריך סטנדרטים?

- אחידות
- בהירות
- פשטות שימוש

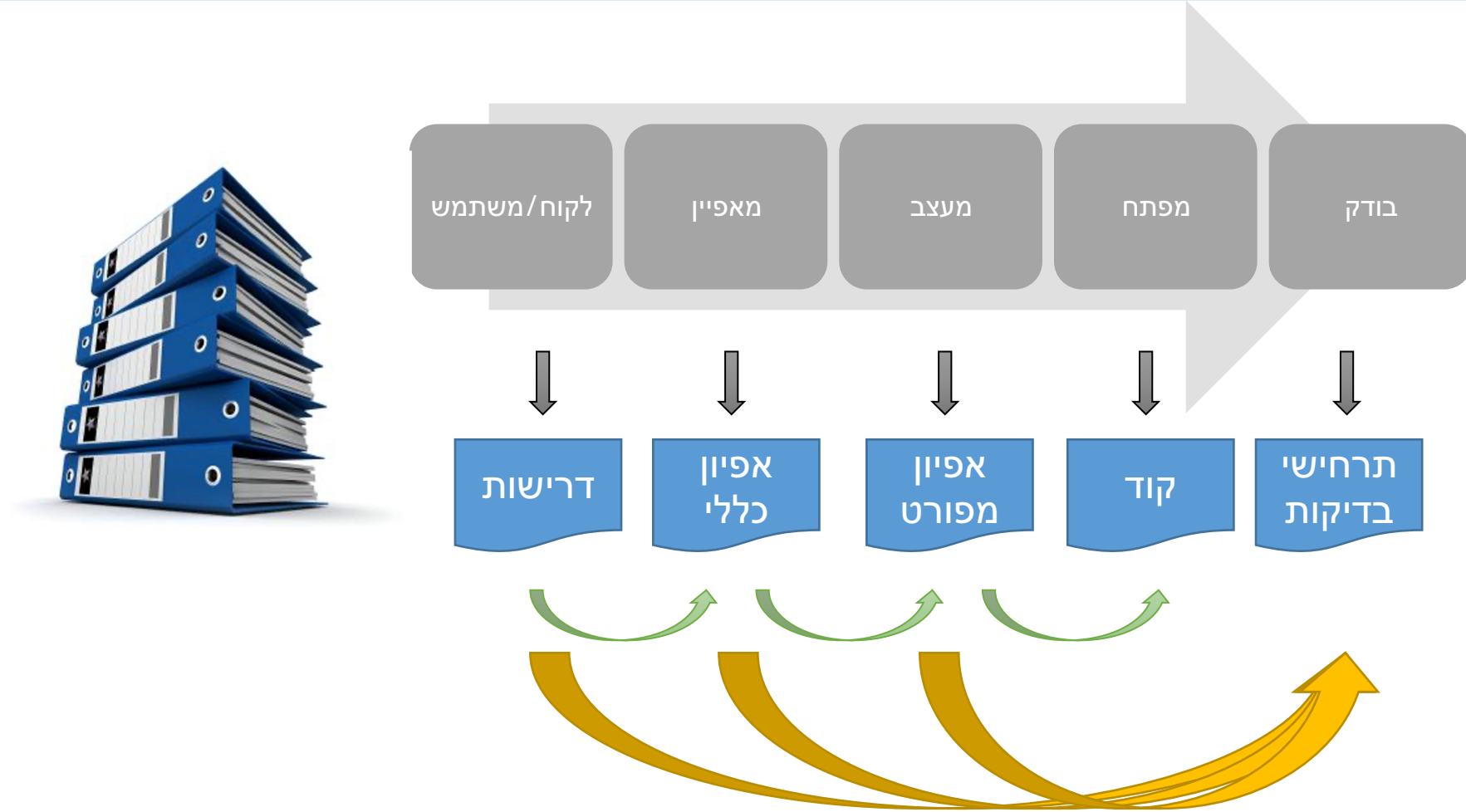
חסכון בזמן



היכן נשתמש?

- סטנדרטים בקוד
- סטנדרטים במסמכי המערכת (אפיון, בדיקות וכו')

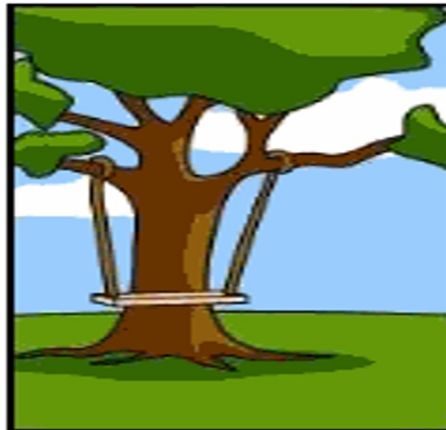
בקרת מסמכים



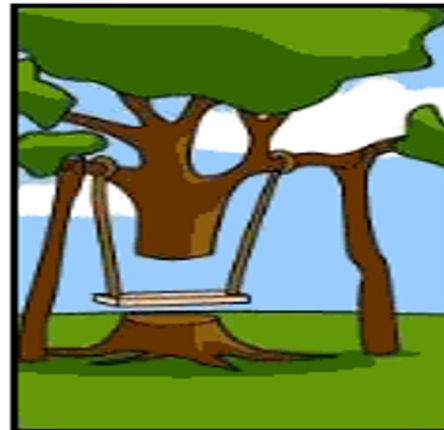
בקרת מסמכים



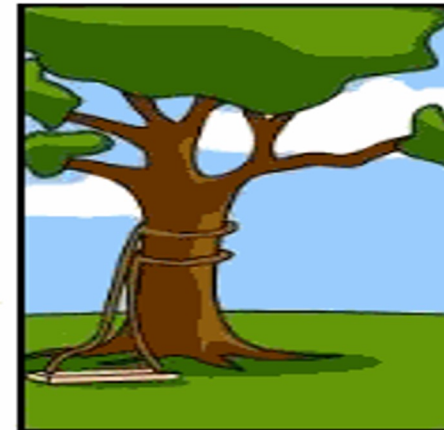
How the customer explained it



How the Project Leader understood it



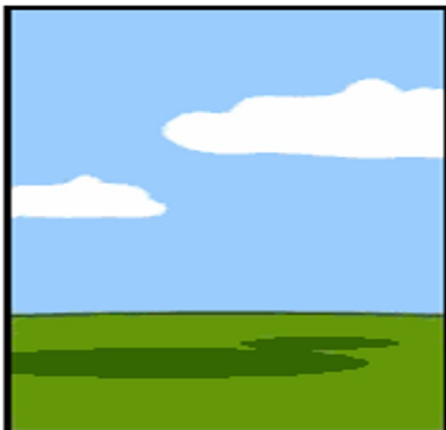
How the Analyst designed it



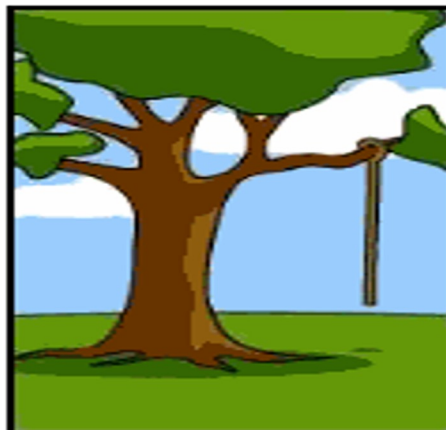
How the Programmer wrote it



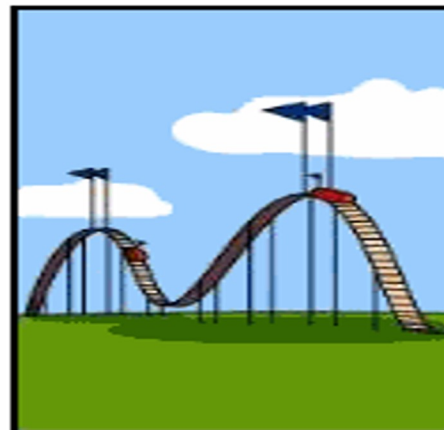
How the Business Consultant described it



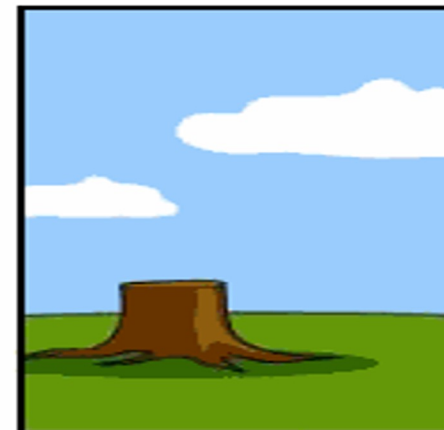
How the project was documented



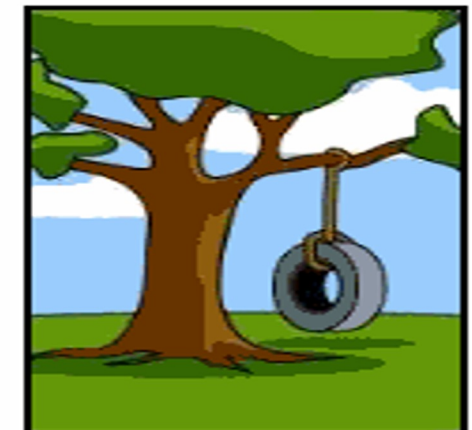
What operations installed



How the customer was billed



How it was supported



What the customer really needed

מהו סקר?

למה צריך סקר תוצרים?



סוגי תוצרים:

- System Requirements Review – **SRR** •
- Preliminary Design Review – **PDR** •
- Critical Design Review – **CDR** •
- Test Readiness Review – **TRR** •

	SRR	PDR	CDR	TRR
מי מנהל?	הנדסת מערכת	מנהל פיתוח / הנדסת מערכת	צוותי הפיתוח	מנהל הבדיקות
מי משתתף?	צוות אפיון, הנדסת מערכת, צוותי הפיתוח ומנהל הבדיקות			
מה עושים?	הצגת דרישות המערכת	הצגת מסמכי התכן ותהליכי המערכת	הצגת פיתרון המימוש של המערכת	הצגת תוכנית הבדיקות (STP)
באיזה שלב?	בעת התנעת הפרויקט	לאחר אישור הדרישות	לאחר אישור מסמכי התכן	לאחר אישור הדרישות

מהלך הסקר:

1. צוות העבודה הרלוונטי מציג בקצרה את התוצרים הנסקרים
2. הצוות הסוקר מעלה הערות
3. נערך דיון לאימות ולתיקוף ההערות שהועלו, כדי לקבוע את הנושאים לפעולה (action items)
 - תיקונים
 - שינויים
 - תוספות
4. החלטה לגבי התוצרים הנסקרים, הקובעת את המשך הפעילות בפרויקט
 - אישור מלא
 - אישור חלקי
 - דחייה



Code Inspection = Code Review = Walkthrough = סקירת קוד

רשימות תיוג

- רמת סטנדרטים של תוכנה (כתיבת הערות לפונקציה, סטנדרטים לשמות משתנים וכו')
- רמת שפת הפיתוח



שלבי הסקירה:

- הפצת רכיב קוד "נקי" למבקרים (צוות הפיתוח)
- הצגת מהות הקוד
- סקירת הקוד ע"י כל אחד מהמבקרים
- פגישת סיכום ותיעוד

מהו סיכון?



כל גורם העלול למנוע מהפרויקט להגיע לסיומו המוצלח ולעמוד במטרותיו. קיימים סיכונים ידועים ובלתי ידועים.

- חוסר ידע (טכנולוגי, מקצועי, מסחרי...)
- חוסר זמן (הגדרת סיום הפרויקט ע"י הלקוח בזמן קצר מיכולת נטו למימוש)
- חוסר בתקציב
- חוסר בכ"א (העדר מתכנתים, נתחי מערכות...)
- שינויים בתכולת הפרויקט תוך כדי עבודה (לקוח משנה דעתו במהלך הפרויקט) - סיכון לא ידוע

השפעות אפשריות של הסיכונים:



- אי מימוש בתכולה
- אי עמידה בזמנים
- אי עמידה בתקציב
- חוסר שביעות רצון לקוח
- אי עמידה במטרות נוספות שהוגדרו לפרויקט

התהליך:

- זיהוי וכימות הסיכונים
- קביעת פעולות לכל סיכון (גידור, ניטור, קבלה)
- מעקב אחר סיכונים שדווחו

דירוג הסיכונים:

- הסתברות למימוש הסיכון (סיכוי – ערכו בין 0 ל-1)
- מידת השפעתו על הפרויקט ומידה ויתממש (סיכון)
- **נזק = סיכון X סיכוי**

מס'	סיכון	סיכוי	נזק	רמת סיכון	תיאור הנזק	פעילות	תיאור	אחראי	תאריך יעד
1									
2									

ניהול סיכונים

דוגמאות לסיכונים:

סיכון	סיכוי	נזק	רמת סיכון	תיאור הנזק	פעילות	תיאור	אחראי	תאריך יעד
בודקת מנוסה יוצאת לחופשת לידה	1	6	6	איכות פחותה של הבדיקות, עיכוב בלוח הבדיקות	גידור	גיוס בודק נוסף לצוות והכשרתו בהתאם	עומר	13.4
המעבדה לא תהיה פנויה להרצת בדיקות בזמן	0.4	9	3.6	עיכוב בתחילת בדיקות האינטגרציה ובהמשך עיכוב הגרסה כולה	ניטור	מעקב אחר תפוסת המעבדה	קרן	

תרגיל 5– זיהוי סיכונים

בקרת שינויים

למה בקרת שינויים?

בכל שלבי הפרויקט חלים שינויים רבים, בתכולות, החלטות ניהוליות ועוד... שינויים אלה צריכים להיות מבוקרים על מנת להבטיח שכל הגורמים הנוגעים בדבר יקבלו בצורה מסודרת את כל המידע.

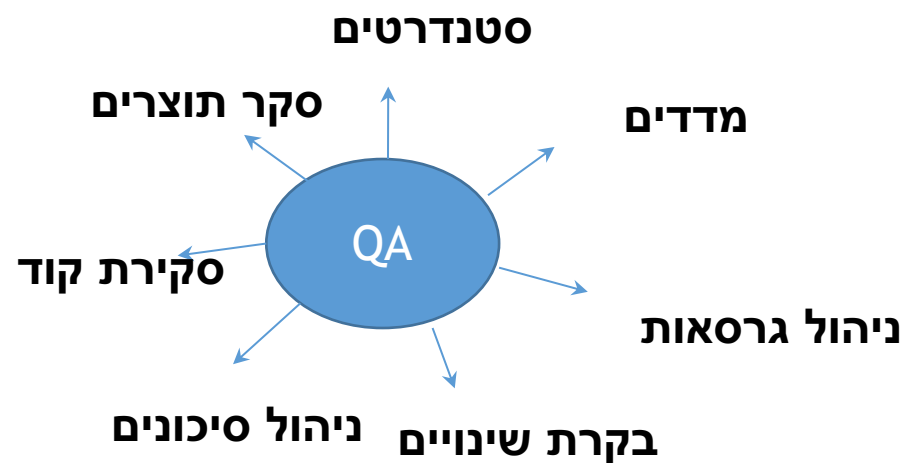
התהליך:

- הגדרת מקורות השינויים
- הגדרת מסגרת לאיסוף בקשות לשינויים ותוספות
- הגדרת נוהל לקביעת תכולת גרסה
- הגדרת נוהל למעקב ביצוע של תכולת גרסה
- הגדרת מסלול מיוחד לתיקוני חירום



✓ למה חשוב לבצע פעולות ניהול איכות?

✓ על אילו פעולות דיברנו?



מה נלמד במפגש הבא?

✓ מתודולוגיות לבדיקות

✓ מהו מסמך STP?

✓ איך כותבים מסמך STP?



QA?

THANK YOU