



# Requirements

דרישות

Moshe Davidian



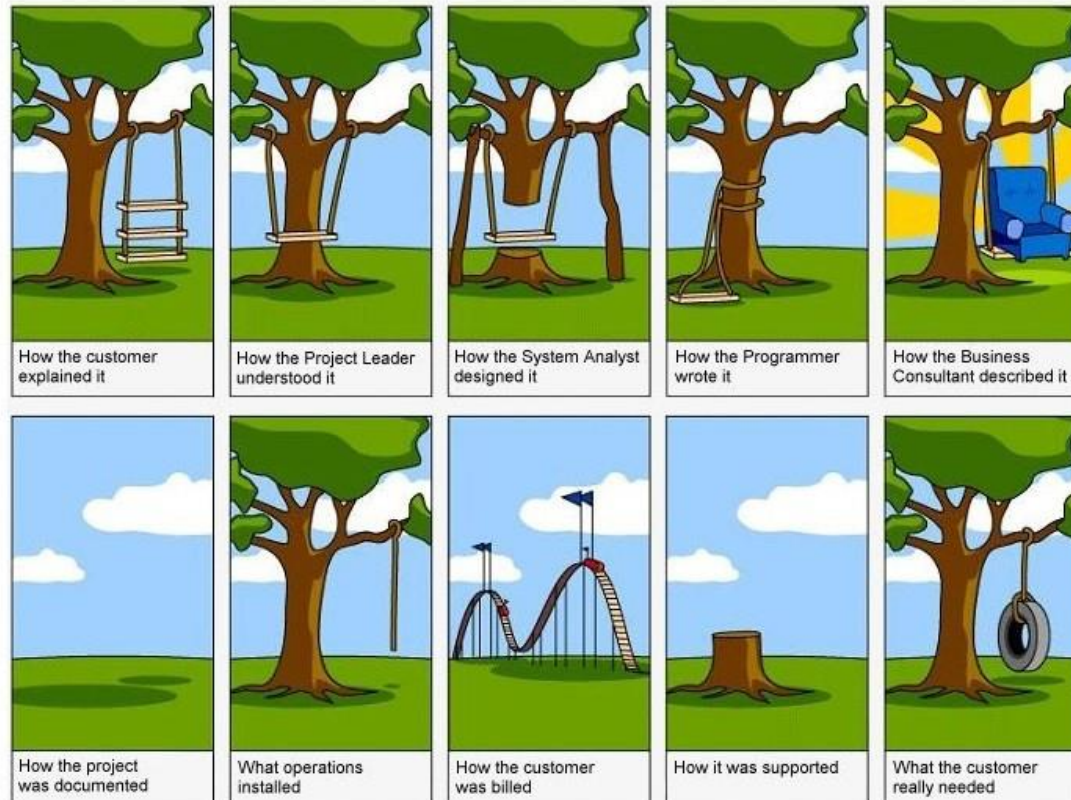


## מה זו דרישה?

- ▶ דרישה היא הגדרה של מאפיין או אילוץ של התכנן והוא חיוני לקבלת המוצר או התהליך.
- ▶ דרישה שאינה כתובה לא מאפשרת בקרה ומעקב, ולכן נדרוש שהדרישות יופיעו כמשפטים.
- ▶ מאפיין הוא תכונה שהמערכת צריכה לאפשר, או צורת התנהגות של המערכת במצב כלשהו.
- ▶ אילוץ הוא הגבלה כלשהי על המערכת - עקב מצב קיים (למשל קיום בסיס נתונים שעל המערכת להשתמש בו).
- ▶ מטרת שלב הדרישות הוא יצירת תשתית למפרט התוכנה על פי צורכי הלקוח כגון הגדרת צורכי הלקוח, הגדרת יכולות המוצר והתנאים בהם הוא נדרש לעמוד והבנה משותפת בין הלקוח למפתח.
- ▶ רשימת הדרישות מספקת כלי למדידה ולמעקב אחרי התקדמות תהליך הפיתוח.

## מה זו דרישה?

מטרת הדרישות הוא יצירת תשתית למפרט התוכנה על פי צורכי הלקוח המובילה להבנה בין הלקוח למפתח. ►



# הנדסת דרישות

- ▶ הנדסת דרישות היא תהליך שבו אנו אוספים ומגדירים את הפעולות שהמערכת צריכה לספק.
- ▶ בשלבים הראשונים של התהליך, רוב המאמצים יושקעו על הבנת דרישות עסקיות ברמה גבוהה - מה באופן כללי על המערכת לבצע.
- ▶ בהמשך התהליך, יבוצעו מאמצים להפקה והבנה של דרישות מערכת מפורטות - דרישות ספציפיות יותר, מתוך הדרישות הכלליות.
- ▶ ניהול דרישות הוא מרכיב קריטי בהצלחת פרויקטים, ועל כן, הוא מהווה נדבך מרכזי בכל תקן או מתודולוגיה של ניהול פרויקטים.
- ▶ מערכת איכותית היא מערכת שהצליחה לענות על הקריטריון של מתן מענה לדרישות מפורשות ומשתמעות.



## הנדסת דרישות

- ▶ צרכי בעלי העניין (למשל לקוחות, משתמשי קצה, ספקים, מפתחים, בודקים, יצרנים ואנשי תמיכה לוגיסטית) הם הבסיס לקביעת הדרישות.
- ▶ צרכי הלקוח, ציפיותיו, האילוצים, הממשקים והרעיון שעומד מאחורי המוצר עוברים ניתוח, התאמות ופירוט לצורך תרגומם לדרישות.
- ▶ לעתים קרובות, צרכי בעלי העניין, הציפיות, האילוצים והממשקים מזוהים בצורה גרועה או מתנגשים. מאחר והם צריכים להיות מזוהים ומובנים בבירור, משתמשים בטכניקות שונות להשגת מטרה זו.
- ▶ כדי להקל על האינטראקציה הנדרשת עם הלקוחות, נציג של משתמש הקצה או הלקוח מעורב לעתים קרובות כדי לייצג את צרכיו ולסייע בפתרון קונפליקטים.

## הנדסת דרישות

- ▶ הפקת הדרישות לא מתבצעת רק בשלב איסוף הדרישות אלא לכל אורך מחזור החיים של המוצר, על ידי זיהוי של דרישות נוספות שלא סופקו במפורש ע"י הלקוחות והבנת השפעותיהן על המוצר.
- ▶ דוגמאות לטכניקות להבנת הצרכים הם:
  - ▶ שאלונים, ראיונות ותרשימים מבצעיים המתקבלים ממשתמשי קצה.
  - ▶ אבטיפוס ודגמים.
  - ▶ סיעור מוחות.
  - ▶ סקרי שוק.
  - ▶ תצפית על מוצרים קיימים, סביבות ודפוסי זרימת עבודה.



# הנדסת דרישות

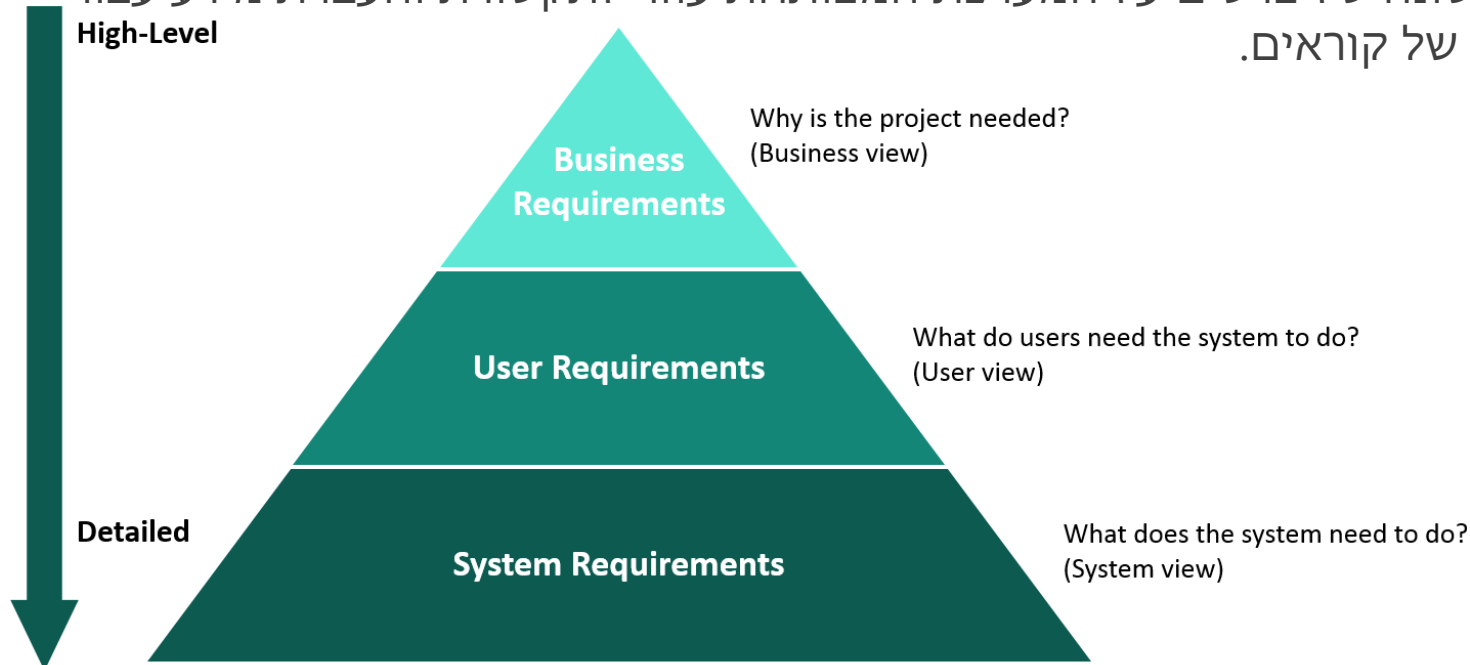
▶ דוגמאות למקורות של דרישות שעשויות שלא להיות מזהות על ידי הלקוח כוללות:

- ▶ מדיניות עסקית.
- ▶ תקנים.
- ▶ דרישות סביבתיות עסקיות (למשל, מעבדות, בדיקות ומתקנים אחרים ותשתיות טכנולוגיות מידע).
- ▶ טכנולוגיה.
- ▶ מוצרים מדור קודם או רכיבי מוצר (שימוש חוזר ברכיבי מוצר).



# רמת פירוט דרישות

- ▶ בדרך כלל, מציגים דרישות בשלוש רמות של פירוט - דרישות עסקיות, דרישות המשתמש ודרישות המערכת.
- ▶ המשתמש צריך הצהרות ברמה גבוהה של הדרישות, בעוד מפתחי מערכת צריך מפרט מערכת מפורט יותר.
- ▶ השימוש ברמה שונה של פרטים על המערכת המפותחת עוזר לתקשורת והעברת מידע עבור בין סוגים שונים של קוראים.





# דרישות משתמש

- ▶ דרישות המשתמש נכתבות עבור המשתמש, בדרך כלל בשפה פשוטה וללא פרטים טכניים.
- ▶ צורכי בעלי העניין, הציפיות, האילוצים והממשקים הם אלו שמגדירים את דרישות המשתמש.
- ▶ דרישות המשתמש מתארות את השירותים שהמערכת צריכה לספק ואת האילוצים שבהם היא צריכה לפעול.
- ▶ יש לאחד את הרצונות השונים של בעלי העניין הרלוונטיים, לקבל מידע חסר ולפתור קונפליקטים בתיעוד של דרישות הלקוח המוכרות.



## דרישות מערכת

- ▶ דרישות המערכת נכתבות עבור המפתחים, מפורטות בהן הדרישות הפונקציונאליות והלא פונקציונליות, נכתבות בבירור ובאופן קפדני.
- ▶ דרישות המערכת משמעותן תיאור מפורט יותר של שירותי המערכת והאילוצים התפעוליים, כגון האופן שבו המערכת תשמש, ואילוצי הפיתוח, כגון שפות התכנות.
- ▶ דרישות המערכת מתייחסות לצרכים הקשורים לכל שלב מחזור חיי המערכת.
- ▶ דרישות אלו נובעות מאילוצים, התחשבות בסוגיות שלא צוינו במפורש בקו הדרישות של הלקוח, בגורמים התלויים בארכיטקטורה שנבחרה, בעיצוב ובשיקולים העסקיים של היזם.
- ▶ רמה זו של פירוט נדרשת על ידי המעורבים בפיתוח המערכת, כמו מהנדסים, אדריכלי מערכת, בודקים וכדומה.



# דרישות פונקציונליות ולא פונקציונליות

- ▶ דרישות התוכנה מסווגות לדרישות פונקציונליות ולדרישות לא פונקציונליות.
- ▶ דרישות פונקציונליות
- ▶ הפונקציות העיקריות שיש לספק על ידי המערכת. כמו שהם באים לידי ביטוי בדרישות המשתמש.
- ▶ דרישות ספציפיות המתארות את פונקציות המערכת, מה הקלט, תהליך העיבוד ומהו הפלט הצפוי.



# דרישות פונקציונליות ולא פונקציונליות

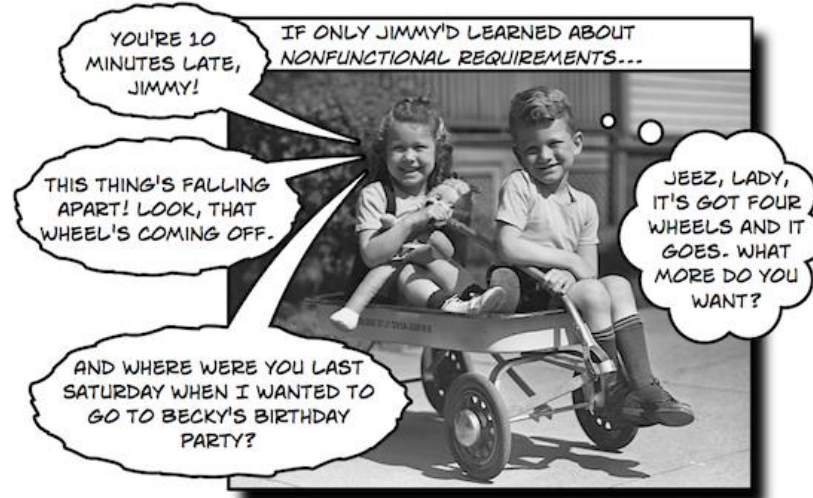
## ▶ דרישות לא פונקציונליות

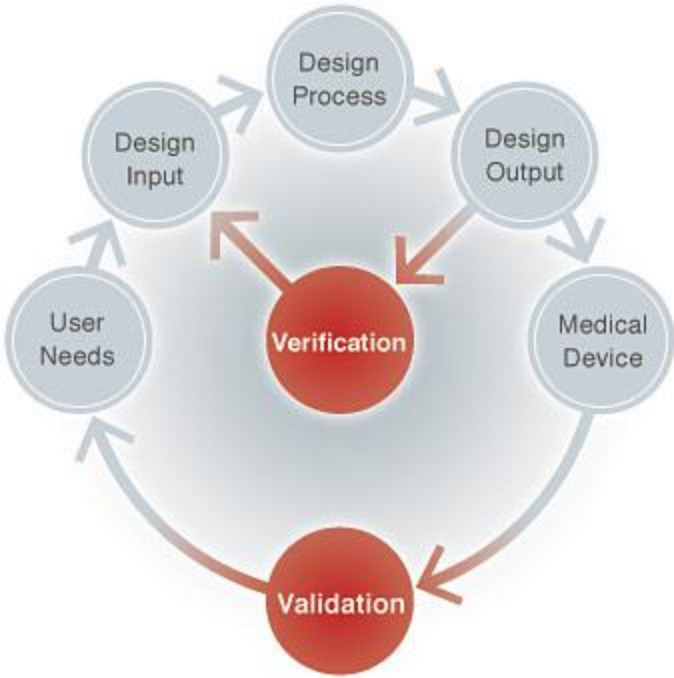
- ▶ אלה הם המגבלות על הפונקציות שמספקת המערכת.
- ▶ אילוצים, כמו למשל, עם כמה תהליכים המערכת יכולה להתמודד (ביצועים), מה הם בעיות האבטחה בהם המערכת צריכה לטפל.
- ▶ שיעור הכשל (מהימנות), מה הן השפות והכלים שישמשו לפיתוח, מה הם הכללים שאחריהם צריך לעקוב כדי להבטיח את המערכת פועלת במסגרת החוק של הארגון (חקיקה).



# דרישות פונקציונליות ולא פונקציונליות

- ▶ דרישות שאינן פונקציונליות הן לעתים קרובות קריטיות יותר מאשר דרישות פונקציונליות בודדות.
- ▶ משתמשים יכולים בדרך כלל למצוא דרכים לעקוף את פונקציית המערכת, במקרה והיא לא עונה במדויק על הצרכים שלהם. עם זאת, אי עמידה בדרישה לא פונקציונלית יכולה לגרום לכך שהמערכת כולה אינה שמישה.
- ▶ לדוגמה, אם מטוס אינו עומד בדרישות האמינות, הוא לא יהיה בטוח לפעולה, או אם מערכת בקרה מוטמעת אינה עומדת בדרישות הביצועים שלה, פונקציות הבקרה לא יפעלו כהלכה.





## אימות ותיקוף דרישות

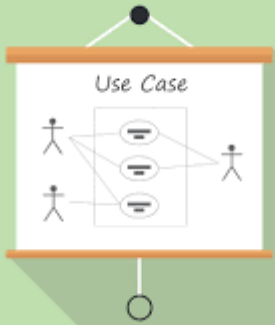
- ▶ אימות (Verification) האם אנו מפתחים נכון את הדרישה?
- ▶ • האם תהליך הפיתוח נעשה בצורה נכונה?
- ▶ תיקוף (Validation) האם אנו מפתחים את הדרישה הנכונה?
- ▶ האם הדרישה המפותחת תתאים ליעודה הצפוי?
- ▶ אימות ותיקוף הדרישות באים כדי להבטיח שהמוצר המתקבל יבצע את עבודתו כמתוכנן בסביבת המשתמש.
- ▶ התהליך מבוצע בשלב מוקדם של הפיתוח עם משתמשי הקצה.
- ▶ דוגמאות לטכניקות המשמשות לאימות הדרישות כוללות: ניתוח, סימולציות, אב טיפוס והדגמות.

# שיטת כתיבת דרישות

- ▶ ישנם מספר דרכים / שיטות לכתיבת דרישות, בין השאר:
- ▶ סיפור משתמש - Story User
- ▶ תרחישי שימוש - Use Case
- ▶ תבניות רישום דרישות.
- ▶ שימוש בנוהל מפת"ח (מתודולוגית פיתוח ותחזוקה) לרישום דרישות.



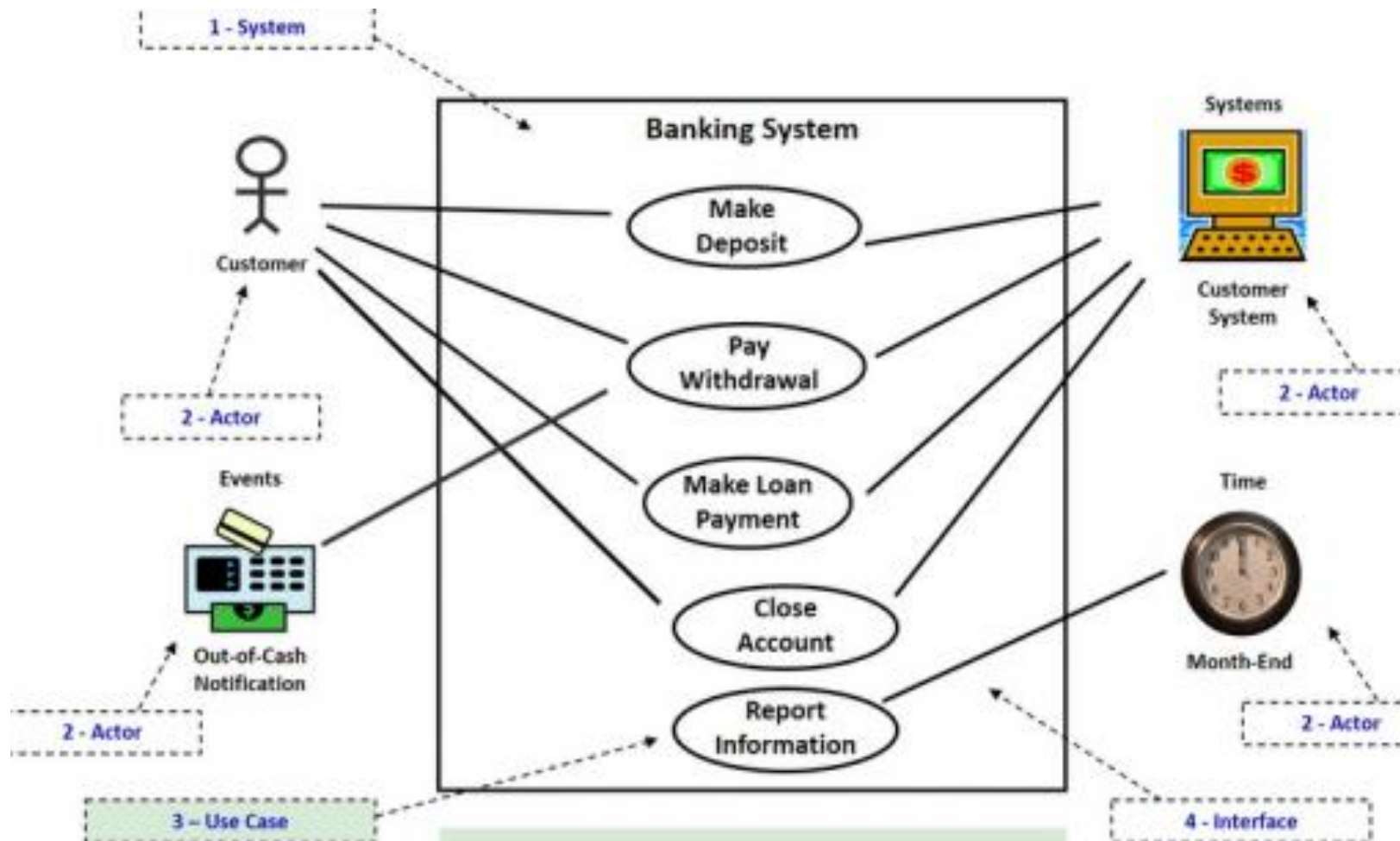
# Use Case



- ▶ תרחיש שימוש (באנגלית: Use Case) היא טכניקה לאיסוף וניתוח הדרישות הפונקציונליות של המערכת.
- ▶ תרחיש שימוש מורכב מרצף אירועים אחד או יותר, המתאר כיצד המערכת מתקשרת עם משתמשים (הקרויים "שחקנים") כדי להשיג פונקציה מסוימת.
- ▶ השחקנים בתרחיש שימוש יכולים להיות משתמשי קצה או מערכות אחרות.
- ▶ לרוב, תרחישי שימוש נכתבים בשפה פשוטה המובנת למשתמש הקצה או למומחה היישום, ולא בניב טכני.
- ▶ במקרים רבים תרחיש השימוש נכתב במשותף על ידי מנתחי המערכת ומשתמשי-קצה.
- ▶ נהוג לכתוב את התרחיש כמסמך ברור וקל להבנה בעיצוב פשוט.



# דוגמה לתרחיש שימוש



# תבנית לתרחיש שימוש

▶ אין תבנית תקנית לתיעוד תרחישי שימוש מפורטים, ולמעשה מקובלות בענף מספר תבניות מתחרות.

▶ בין השאר מקובל לכלול את הסעיפים הבאים:

▶ מספר זיהוי ושם תרחיש השימוש

▶ זיהוי האיטרציה

▶ תקציר

▶ תנאי התחלה - סעיף זה משמש להצהרה על תנאים החייבים להיות נכונים כשמשתמש מתחיל את תרחיש השימוש.

▶ אירועים מזניקים - האירועים או תנאי ההתחלה הגורמים להפעלת תרחיש השימוש.

▶ מסלול עיקרי - תיאור הרצף העיקרי או הטיפולוסי של האירועים בתרחיש השימוש.

▶ מסלולים חלופיים

▶ תנאי סיום

▶ הערות



# דוגמה לתרחיש שימוש

מספר זיהוי ושם התרחיש - 1, משוך כסף.  
זיהוי האיטרציה - מסלול עיקרי יפותח כחלק ממהדורה 1 של התוכנה.  
תקציר - תרחיש שימוש זה מאפשר ללקוח הבנק המצויד באמצעי זיהוי למשוך כסף מכספומט.  
תנאי התחלה - המערכת מקוונת, יש לפחות סכום מינימלי בקופת המכונה.  
מסלול עיקרי -

1. המשתמש מזדהה למערכת באמצעות אמצעי הזיהוי.
  2. המערכת מציגה תפריט אפשרויות מותאם-אישית.
  3. המשתמש בוחר "משיכת מזומן" מהתפריט ומזין את הסכום שברצונו למשוך.
  4. המערכת רושמת את הפעולה.
  5. המערכת סופרת את הסכום באופן ממוכן.
- וכולי...

## מסלולים חלופיים -

- הסכום המבוקש גדול יותר מסה"כ הכסף בקופה.
1. המערכת מודיעה למשתמש ומבטלת את הפעולה.
  2. המערכת מציגה את התפריט הראשי.
- וכולי...



## תבנית כתיבת דרישה

שם הדרישה	שינוי צבע הנקודה בגרף בעל שני צבעים	מס' מזהה	9
שם המערכת	S.Algo	תאריך בקשה	16/10/2016
שם הדורש	ד"ר עבד אלכרים אבו עפאש	טלפון	
דחיפות ביצוע	עליונה	אסמכתא	100
	ת. סיום נדרש	01/01/2017	

### הגרם המנתח

שם	יעל וקסלר	תפקיד	ראש צוות פיתוח	תאריך	27/12/2016
שיוך ארגוני	צוות פיתוח	טלפון	054-6399884	Email	<a href="mailto:Veksler.yael@gmail.com">Veksler.yael@gmail.com</a>

### ניתוח היישום

תאור מפורט לאופן היישום	<p>המשתמש יבחר נקודה מתוך רשימת הנקודות או ע"י בחירת נקודה מהגרף. המשתמש יכול לשנות את צבעה של הנקודה. מאדום לכחול וההפך.</p> <p>פעולה זו תעדכן את נתוני הנקודה ברשימת הנקודות של הגרף והצבע הנקודה ישתנה בהתאם.</p> <p>במידה והמשתמש יבחר בצבע הקיים הנתונים לא ישתנו.</p>
-------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## מהי דרישה טובה?

- ▶ על פי נוהל מפת"ח דרישה טובה, חייבת לעמוד בכמה קריטריונים:
  - ▶ ברורה ופשוטה להבנה
  - ▶ ברת בחינה (Testability) ניתנת לאימות ותיקוף בבדיקה, בהדגמה, באנליזה או בבחינה שהיישום אכן עונה על הדרישה.
  - ▶ חד משמעית - הדרישה אינה ניתנת לפרשנויות שונות.
  - ▶ מלאה - הדרישה כוללת את כל הפרטים.
  - ▶ עקבית - הדרישה אינה סותרת דרישות אחרות ואינה חוזרת על דרישה אחרת.
  - ▶ חד ערכית- כל דרישה עוסקת ביכולת אחת של המערכת.

Done 