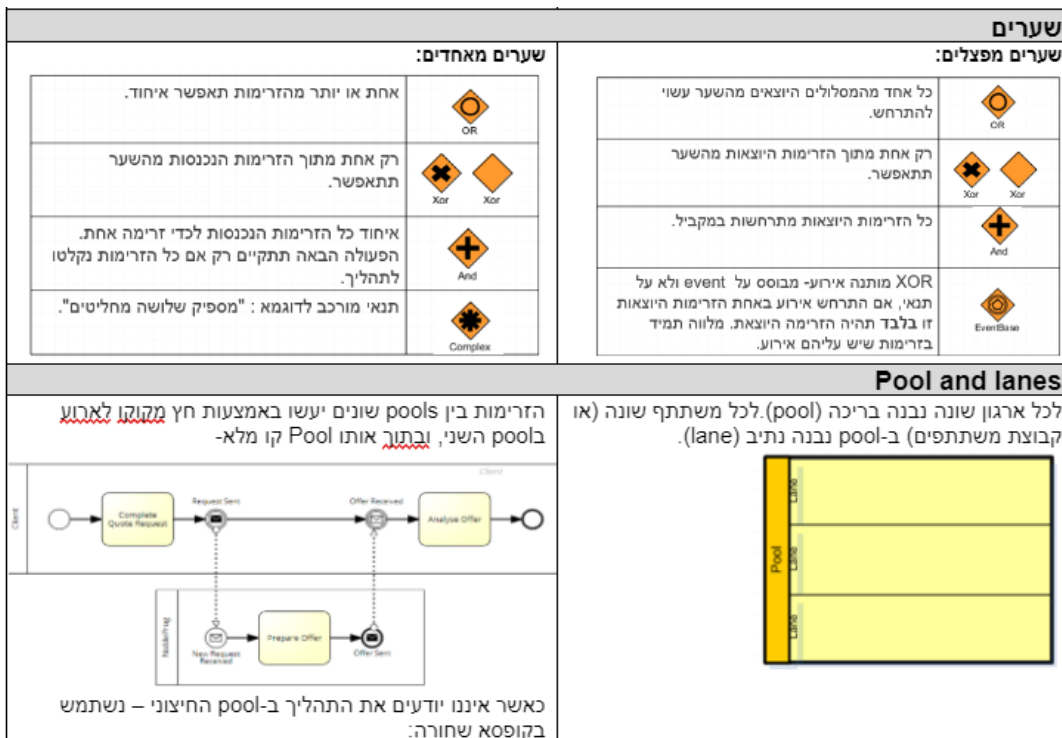
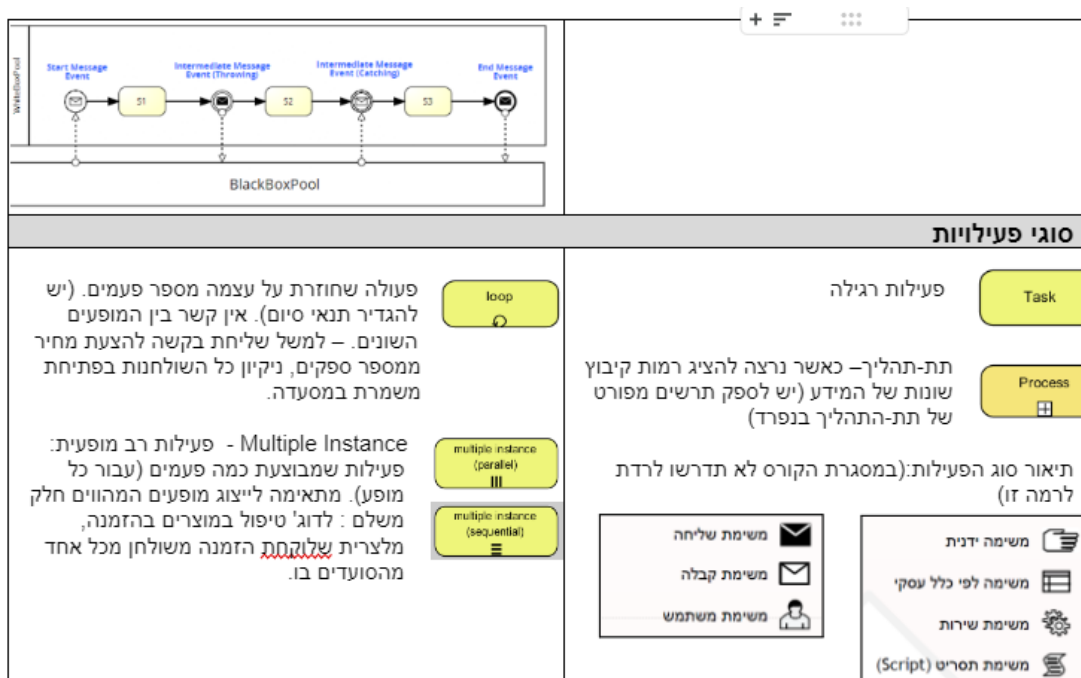


# תרגול 1- מידול תהליכים עסקיים שפת –BPMN

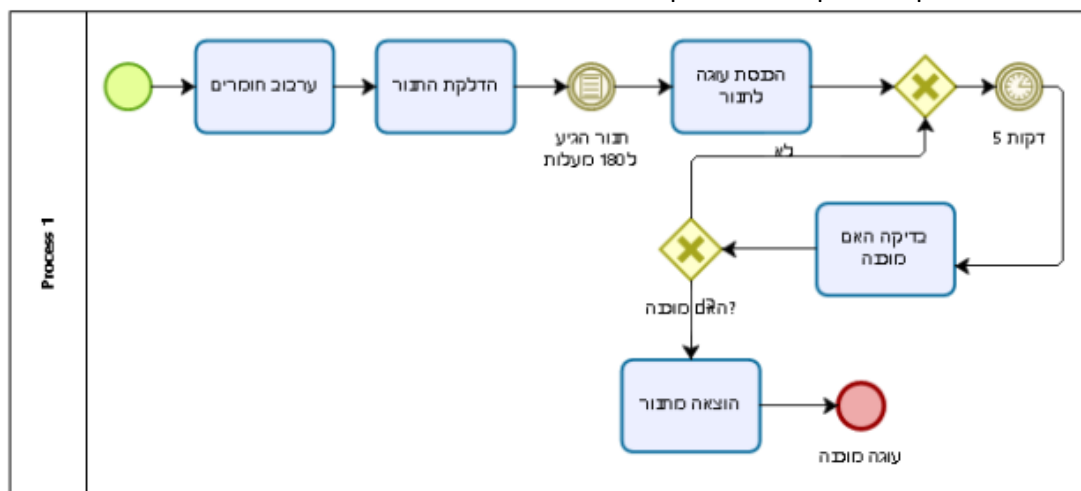
סיכום האלמנטים שנלמדו בהרצאה:





### תרגיל כיתה 1 – הכנת עוגה -חימום

תהליך הכנת עוגה מתבצע באופן הבא: תחילה מערבבים את כל החומרים, לאחר מכן מדליקים את התנור. לאחר שהתנור יגיע ל180 מעלות. נשים את העוגה בתנור. לאחר ששמשנו את העוגה בתנור נמתין 5 דקות ולאחר מכן נבדוק האם היא מוכנה, אם כן נוציא אותה מהתנור, ואם העוגה אינה מוכנה נמתין עוד 5 דקות ושוב נבדוק את העוגה.



### תרגיל כיתה 2 – גיוס מועמד

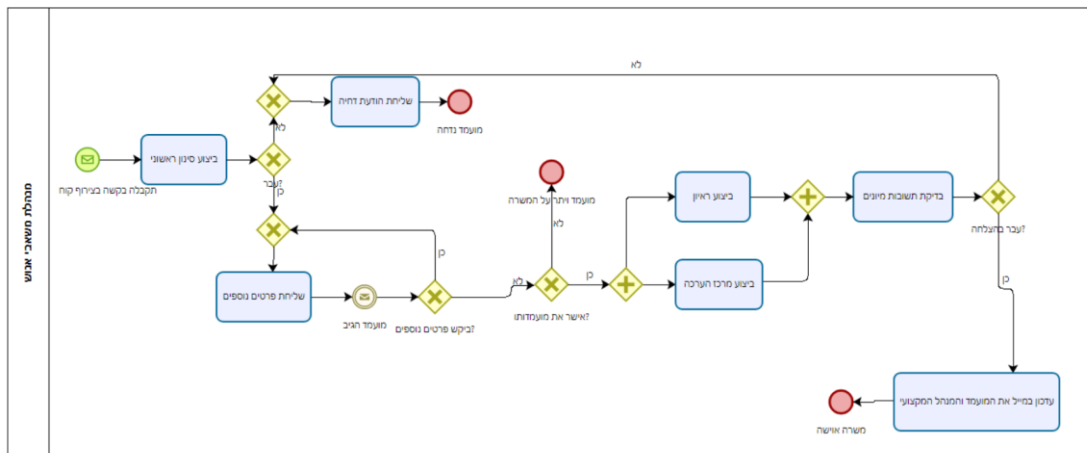
מנהלת משאבי האנוש (או בקיצור מנהלת משא"ש) של ארגון אקו"ם אחראית על תהליכי גיוס עובדים למשרות באקו"ם עבור המחלקות השונות. כיום, באקו"ם אין מערכת מידע לתמיכה בתהליכי גיוס עובדים, וכל העבודה ותיעודה נעשה באמצעות חשבון הדוא"ל האישי של מנהלת משא"ש. האינטראקציה מול המועמד נעשית באמצעות מיילים, כאשר מנהלת משא"ש מתעדת את השלבים השונים בתהליך על ידי שמירה של ההתקשרות במייל לתיקיות ייעודיות בהתאם לתפקיד ולסטטוס המועמד.

תהליך גיוס מועמד מתבצע באופן הבא: כאשר מתקבלת פנייה ממועמד בצירוף קורות חיים, מנהלת משא"ש מבצעת סינון ראשוני של התאמת המועמד על בסיס קורות החיים שלו.

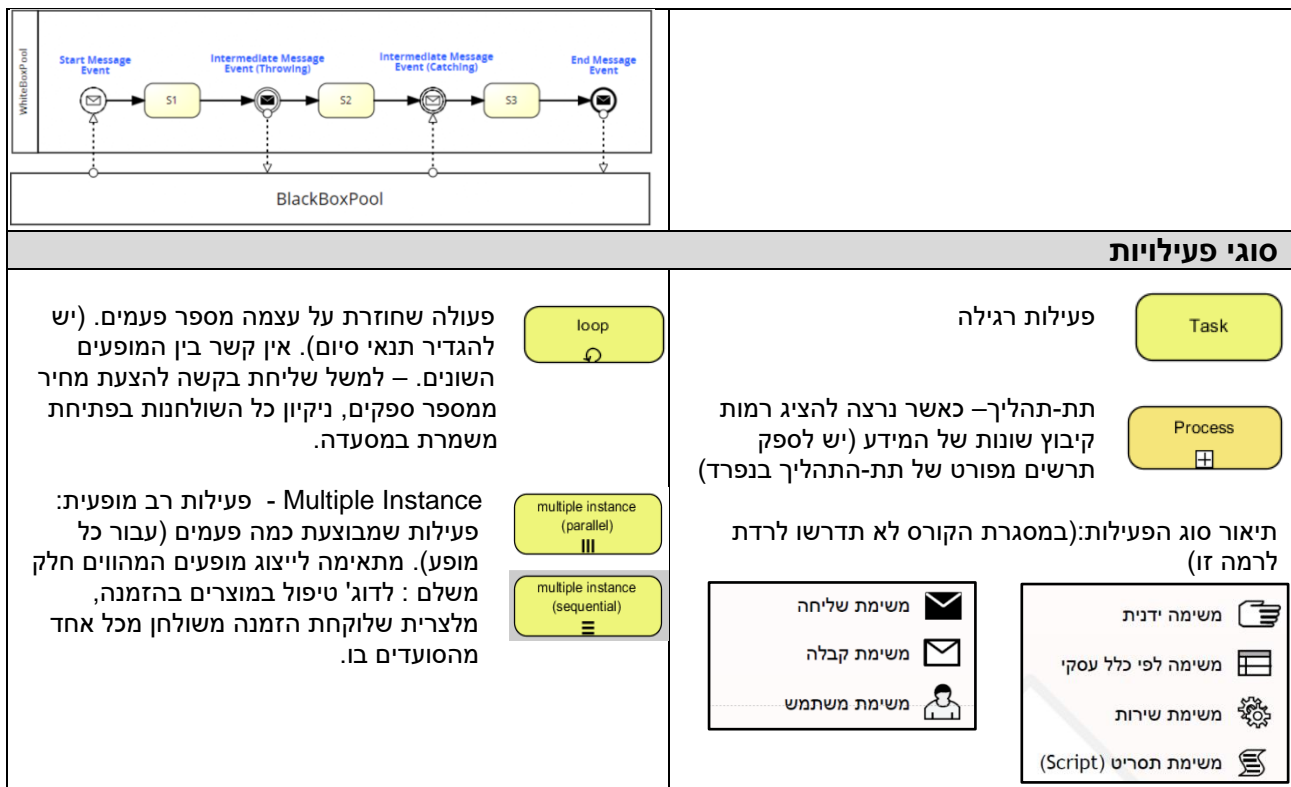
במידה והמועמד עבר את תהליך הסיון הראשוני, מנהלת משא"ש שולחת לו פרטים ראשוניים על המשרה וממתינה לתגובתו. במידה ותגובת המועמד כללה בקשה לפרטים נוספים, מנהלת משא"ש שולחת את הפרטים הנדרשים, ושוב ממתינה לתגובתו וכך על הלאה עד אשר תגובת המועמד מועמד תכלול אישור או דחייה של התפקיד ללא שאלות נוספות. עבור מועמד שלא נמצא מתאים, תשלח הודעת דחייה למועמד והתהליך יסתיים.

לאחר שהמועמד אישר את מועמדותו למשרה (לאחר קבלת הפרטים), מנהלת משאבי האנוש מבצעת עבורו 2 סוגי מיונים: א. ראיון שיבוצע על ידה, ב. מרכז הערכה קבוצתי תחת פיקודה.

לאחר סיום שלב המיונים, דהיינו לאחר קבלת התשובות מכל המיונים שאליהם המועמד נשלח, בודקת המנהלת את התשובות. מועמד מתקבל לעבודה בתנאי שעבר את בהצלחה את כל המיונים שתואמו לו. במידה ולא עבר את אחד המיונים, המועמד נדחה. עבור מועמד שהתקבלה החלטה לקבלו, מנהלת משא"ש שולחת הודעת דוא"ל על קבלתו למשרה, למועמד ולמנהלו המקצועי באק"ם.



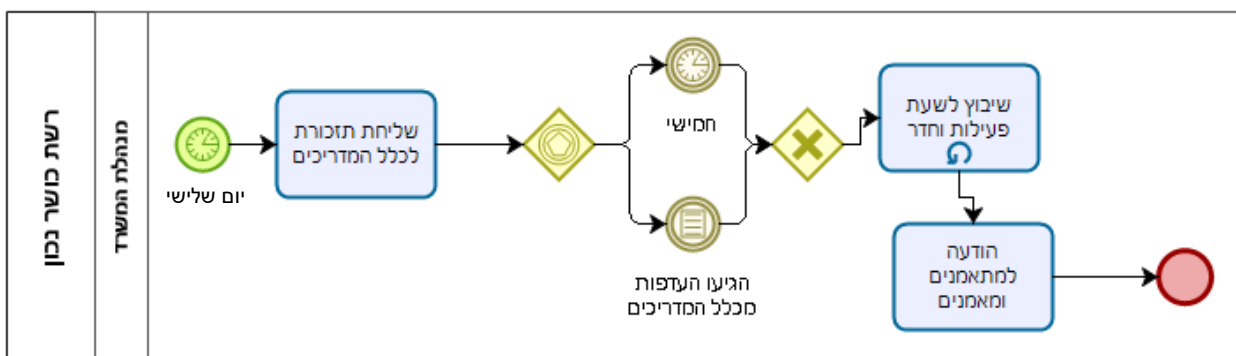




## תרגילי כיתה:

### תרגיל כיתה 1- סטודיו לפילאטיס:

**תכנון עבודה שבועי-** כל אחד מהסניפים השונים, מתנהל באופן עצמאי מבחינת תכנון העבודה השבועי, אך הרשת מחייבת את כלל המכונים להתנהל בצורה אחידה על מנת לוודא כי הפצת תוכנית השעורים השבועית תופץ ללקוחותיה לא מאוחר מיום חמישי בכל שבוע. מנהלת המשרד היא זו שאמונה על התהליך. בכל יום שלישי, מנהלת המשרד שולחת מייל תזכורת בתפוצת כלל המדריכים במכון בבקשה שישלחו את העדפותיהם לשיבוץ לשיעורים לשבוע הבא. המנהלת ממתינה לקבלת העדפות מכלל המדריכים עד ליום חמישי לכלל המאוחר (עובד שטרם הגיש את העדפותיו יחשב כחסום לסידור בשבוע זה והתהליך ימשיך כרגיל). במידה והתקבלו העדפות מכלל המדריכים עוד לפני יום חמישי, מתחילה מנהלת המשרד את שיבוץ המדריכים לכל שעת פעילות ולכל חדר פעילות של המכון (המכון פעיל כל יום בין השעות 07:30-21:00). לאחר סיום השיבוץ המנהלת שולחת את הסידור השבועי למדריכים ובנוסף שולחת הודעה למתאמני המכון על פתיחת רישום עבור שעורים אלו.

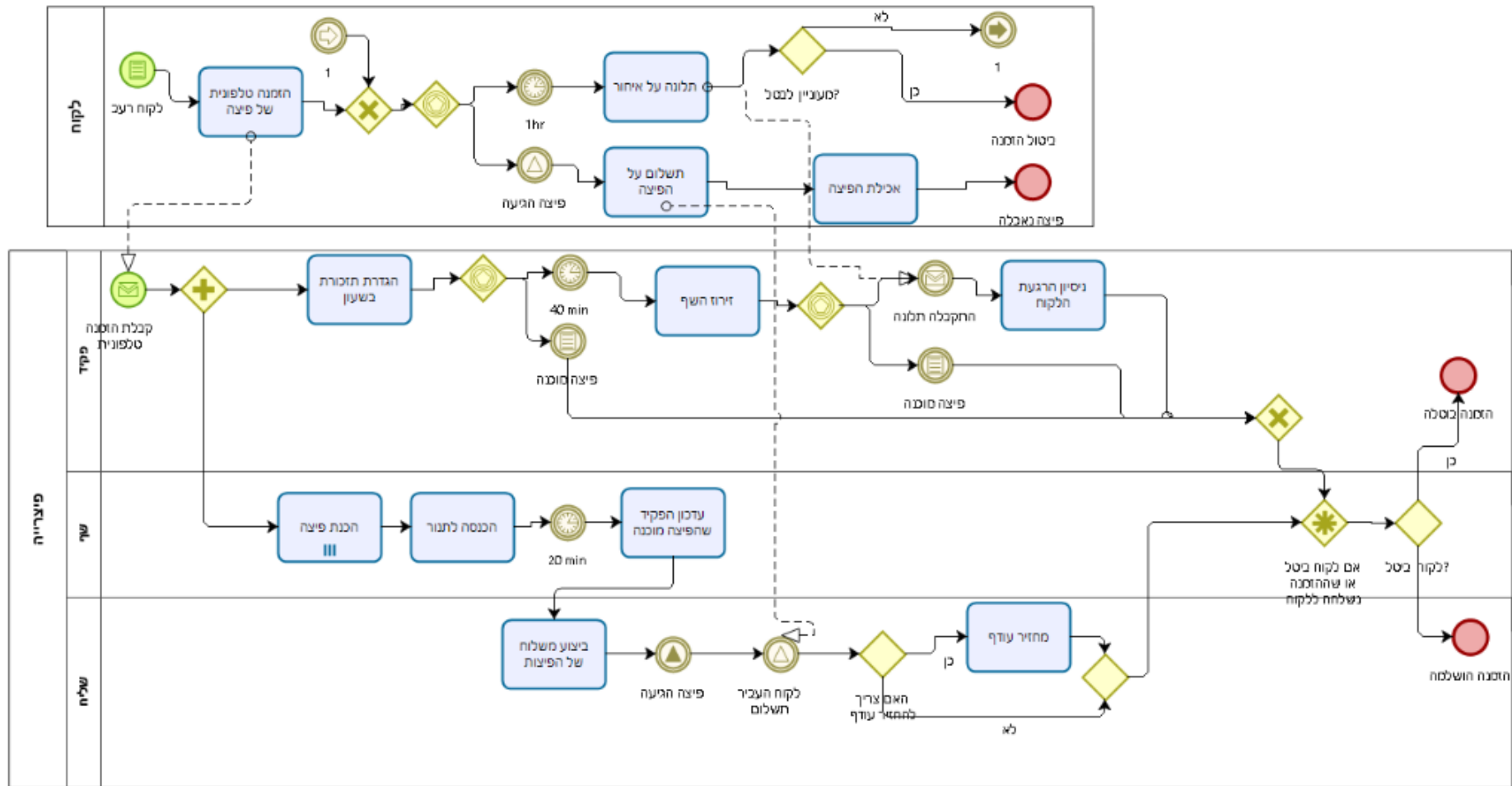


## תרגיל כיתה 2 – הזמנות פיצה

**תהליך הזמנת פיצה על ידי לקוח מתבצע באופן הבא:** כאשר לקוח רעב הוא מבצע הזמנה טלפונית של פיצה/ות. במידה ועברה שעה מרגע ההזמנה והפיצה טרם הגיעה, הלקוח מתקשר להתלונן על האיחור בגין הזמנתו וממשיך להמתין להגעתה שעה נוספת. לאחר הגעת הפיצה, משלם הלקוח על הפיצה ולאחר מכן אוכל את הפיצה.

לאחר שפקיד הקבלה בפיצרייה קיבל את ההזמנה, השף מכין כל אחת מהפיצות בהזמנה ומכניס אותם יחד לתנור. לאחר 20 דקות של אפייה, מעדכן השף את פקיד הקבלה שההזמנה מוכנה למשלוח והפיצה נשלחת ללקוח על ידי שליח. לאחר מסירת הפיצה הלקוח משלם והשליח מחזיר לו עודף.

הפיצרייה מתחייבת לספק ללקוח את הפיצה בתוך שעה ממועד ההזמנה, ועל כן **בזמן בו** הפיצה נכנסת לשלב ההכנה הפקיד מכוון תזכורת של 40 דקות בעבור ההזמנה, לאחר 40 דקות, אם הפיצה לא מוכנה הפקיד מזרז את השף וחוזר לעמדתו. בזמן שהפקיד ממתין שההזמנה תהיה מוכנה, ייתכן שהלקוח יתקשר להתלונן שהפיצה לא הגיעה, במקרה זה פקיד הקבלה מנסה להרגיעו טלפונית. במידה ובסיום השיחה בוחר הלקוח לבטל את העסקה ההזמנה מבטלת.







## תרגול 4- ניתוח דרישות

### סיכום הסיווגים שנלמדו בהרצאה:

דרישות מערכת נחלקות לשני סוגים עיקריים של דרישות:	
<p>דרישות לא פונקציונאליות - Non Functional Requirements</p> <p>מגדירות מאפיינים מדדים ואילוצים על אופן המימוש.</p> <p>דוגמאות: המערכת תואמת לתקנים וסטנדרטים של חברות ביטוח, על המערכת להיות זמינה 98% מזמן העבודה, שעות גיבוי המערכת יתבצעו רק בשעות הלילה</p>	<p>דרישות פונקציונאליות – Functional Requirements – דרישות המייצגות את התכולה של המערכת. מתארות מה המערכת צריכה לבצע. למשל: המערכת תאפשר הפקת דוח חריגים.</p>
תת חלוקה של דרישות פונקציונאליות:	
<ul style="list-style-type: none"> <li><b>דרישות תפעוליות (Operational requirements)</b> - דרישה המתייחסת לתפעול, לאינטראקציה או להתנהגות של המוצר / מערכת. מה המערכת צריכה לעשות (אילו אפשרויות זמינות למשתמש). מה המערכת עושה בעת קליטת פקודות. דוגמאות: בעת הזנת לקוח חדש, תבדוק המערכת האם הלקוח כבר קיים במערכת, בעת הזנת לקוח חדש, המערכת תקצה ללקוח מספר אוטומטי וסיסמא, הלקוח יוכל לבצע חיפוש עפ"י קטגוריות המערכת תאפשר סריקת מסמכים.</li> <li><b>דרישות מידע (Data requirements)</b> - דרישה המתייחסת למידע בו נדרשת התוכנה/מערכת לטפל (לקלוט, לאחסן, לאחזר, לעבוד, להפיק). דוגמאות: לכל לקוח יש <b>שמר</b> כרטיס לקוח ובו הפרטים הבאים:....., לקוח שלא רכש מוצר במשך שנה <b>יקבל סטטוס</b> של לקוח לא פעיל (לנסח דרישה נגזרת – "יש לשמור שדה סטטוס ללקוח).</li> <li><b>דרישות ממשק (Interface requirements)</b> דרישה המתייחסת לחילופי מידע או בקרה בין התוכנה לבין ישויות חיצוניות. ממשקי משתמש (כפתורי הפעלה, תצוגות, הפעלה קולית וכו') פרוטוקולים, סוגי מידע מועבר. דוגמאות: המערכת תקלוט את פרטי הלקוחות <b>מקובץ XML</b> שיועבר אחת לשבוע ממערכת X, המידע אודות הזמנות יועבר למערכת <b>דרך אתר האינטרנט</b> של החברה, המערכת תאפשר חזרה לאחור באמצעות הקיצור CNRL+Z, יש לתמוך ברזולוציות מסך 1020X1920 ומעטה.</li> </ul>	
תת חלוקה של דרישות לא פונקציונאליות:	
<ul style="list-style-type: none"> <li><b>דרישות ביצועים (Performance Requirements)</b> - פרמטרים שניתנים למדידה לגבי ביצועי המערכת. דוגמאות: המערכת תוכל לשמור לפחות כ 1000000 לקוחות, הזמן להפקת דוח יומי לא יעלה על 15 דקות, העבודה השוטפת הנה מול לקוחות בזמן אמת ולכן עיבוד והצגת הנתונים צריך להתבצע באופן מיידי – עד 4 שניות להתחברות ראשונית למערכת ועד שתי שניות למעברים בין מסכים ועוד..</li> <li><b>דרישות מאפייני איכות (Quality Attributes)</b> - תכונות המאפיינות את המוצר הכולל. מאפייני איכות כוללים: <ul style="list-style-type: none"> <li><b>אמינות (reliability)</b> - פעולה ללא תקלות לאורך זמן</li> <li><b>זמינות (availability)</b> - שרות רצוף, התאוששות מהירה מתקלות</li> <li><b>בטיחות (safety)</b> - שמירה על חיייהם ובריאותם (הפיזית והנפשית) של מפעילים ומשתמשים</li> <li><b>בטחון (security)</b> - הגנה על מרכיבי המערכת ועל המידע בה היא מטפלת</li> <li><b>אחזקתיות (maintainability)</b> - יכולת לערוך בקלות שינויים ותיקונים במוצר</li> <li><b>שימושיות (usability)</b> - תרומת המוצר למשתמש בביצוע משימותיו ובהשגת מטרותיו –ממדי שביעות רצון הלקוח</li> </ul> <p>דוגמאות: המערכת תבצע גיבוי יומי בכל יום בחצות (אחזקתיות), המערכת תבקש החלפת סיסמא מהמשתמש בכל חודש (בטיחות)</p> </li> <li><b>דרישות חמרה (Hardware Requirements)</b> - אילוצים ודרישות עבור החמרה של המערכת, כלומר מהי החומרה הנדרשת לשם הרבת המערכת. דוגמאות: כל אחד מאנשי התחזוקה יוכל להיכנס למערכת באמצעות כרטיס מגנטי, הלקוחות יוכלו לצפות במידע באמצעות עמדות PC שיוצבו באזור המלחחות <b>ועוד</b>.</li> </ul>	

- **דרישות אילוצי ניהול - (Management Constraints)** אילוצים ודרישות הנובעים מאפיינים ניהוליים למשל, תקציב, לוח, זמינות משאבים, התאמה לתקנים. דוגמאות: המערכת תותאם לתקן, ISO2000 גרסה ראשונה של המערכת תעלה לאוויר לא יאוחר מאפריל 2013, כל עבודות התחזוקה במערכת יעשו בחלון הזמן שבין חצות לשמונה בבוקר.
- **דרישות אילוצי מימוש (Design Constraint)** - הכתבת אופן מימוש ספציפי בתוכנה (אלגוריתם, מבנה נתונים, אופן הפעלה ספציפי, שימוש חוזר ברכיבים, שימוש בטכנולוגיה מסוימת וכו'). דוגמאות: כל רכיב במערכת יישום כ-קונטרול נפרד אותו ניתן להטמיע בכל מערכת אחרת, המערכת תעשה שימוש במסד נתונים SQL SERVER 2008 R2, בכל יום בשעה 17:00 המערכת תקלוט את נתוני מכירות כל הקופות.

**תרגיל כיתה:** קראו את סיפור המסגרת "מכרז פומבי מספר 12345 - הקמה ותפעול של מערכת PilatesMoodle", שנמצא בקבצי התרגול באתר הקורס ובנו טבלת דרישות על פי הסיפור.

עליכם להתייחס גם לדרישות משתמעות שלא נאמרו בצורה מפורשת.

- **דרישות פונקציונאליות** I - ממשק, D - מידע, O - תפעולית
- **דרישות לא פונקציונאליות:** C - אילוצי מימוש, M - ניהול, H - חומרה, Q - איכות, P - ביצועים.

הערות	סוג דרישה לא פונקציונאלית	סוג דרישה פונקציונאלית	נוסח הדרישה	
	C		1. הפיתוח העתידי יבוסס על פלטפורמת הפיתוח share-point של חברת מיקרוסופט	
	H		2. השירות המבוקש כולל אירוח של המערכת בשרת share-point	
Q - תפעול ותחזוקה. C - בכל יום בשעה 22:00..	C, Q		3. הכולל תפעול ותחזוקה שותפת הכולל גיבויים שיתבצעו באופן אוטומטי בכל יום בשעה 22:00 בלילה	
	H		4. אפשרות לשחזורים ונפח אחסון כולל בגודל 10 ג'יגה ביט	
זמינות	Q		5. הספק יתחייב לזמינות של המערכת בהיקף של 97% מהזמן	
	M		6. על הספק לספק מוקד תמיכה זמין בכל זמן שהמכון פעיל	
	M		7. זמן הטיפול בתקלה משביתה לא יעלה על חמש שעות.	
C - כל הקטע שהטכנולוגיה מבוססת עליו Q - תעוצב בהתאמה ורספונסיב..	C, Q		8. המערכת העתידית תהיה מערכת מבוססת WEB אשר תעוצב באופן responsive בהתאמה למכשירים סלולריים הנפוצים בשוק על ידי שימוש בטכנולוגיה של media quires	
		O	9. מנהלת המשרד, אחראית על קליטת פרטי המדריכים למערכת והגדרת סיסמא ראשונית לכניסת מדריך	

הערות	סוג דרישה לא פונקציונאלית	סוג דרישה פונקציונאלית	נוסח הדרישה	
נגזרת מ-9		O	יש לאפשר למנהלת המשרד לצפות ברשימת המדריכים במכון	10.
נגזרת מ-9		D	יש לשמור לכל מדריך את הסיסמה האישית שלו	11.
		D	לכל מדריך ישמרו שמו, תאריך התחלת עבודתו, מספר הטלפון שברשותו ומספר עובד	12.
		O	המדריכים יוכלו להיכנס למערכת באמצעות מספר עובד וסיסמא	13.
		O	יש לאפשר למדריך לשנות את סיסמתו בעת הצורך	14.
נגזרת מ-13	Q		השימוש במערכת מחייב הזדהות	15.
		O	המערכת מאפשרת צפייה, עריכה, הזנה של האימונים המתקיימים במכון	16.
		D	לכל שיעור ישמרו תאריך ושעת האימון, המדריך שמקיים את השיעור והחדר בו מתקיים מתוך רשימת חדרי המכון.	17.
נגזרת מ-16/17		D	יש לשמור את רשימת חדרי המכון	18.
נגזרת מ-16/17		O	יש לאפשר עדכון של רשימת חדרים	19.
נגזרת מ-16/17		O	יש לאפשר קליטה של פרטי אימון הכוללים תאריך, שעה, מדריך וחדר.	20.
		O	המערכת תאפשר למדריך לבנות את מערך השיעור עבור כל שיעור שנקבע במערכת: המדריך יוכל לבחור סדרות של תרגילים ולשייכם לשיעור	21.
נגזרת מ-21		O	המערכת תאפשר למדריך לצפות ברשימת סדרות, לכל סדרה לצפות ברשימת התרגילים שהיא כוללת.	22.
נגזרת מ-21		D	יש לשמור לכל שיעור את הסדרות ששובצו לו	23.
		D	כל סדרה מאופיינת על ידי מנח הגוף (מחזורת המתארת את מנח הגוף בסדרה, למשל "ישיבה זקופה על עצמות המושב עם רגלים מעתה")	24.
		D	יש לשמור רשימת סדרות. הסדרות השונות מובנות מסט של תרגילים ברמת קושי עולה ולכן לכל תרגיל י חשיבות למיקומו בסדרה.	25.

נוסח הדרישה	סוג דרישה פונקציונאלית	סוג דרישה לא פונקציונאלית	הערות
26. יש לאפשר להנהלת הרשת צפייה ברשימת סדרות וקליטת סדרה חדשה.	O		נגזרת מ-25
27. יש לאפשר להנהלת הרשת שינוי מיקום תרגיל בסדרה	O		נגזרת מ-25
28. לכל תרגיל המערכת תשמור את תיאור מילולי של התרגיל, כמו כן גם את משך זמן הביצוע ומספר החזרות לתרגיל זה. התרגילים ייחודים לסדרות שבהם הם נמצאים.	D		
29. לכל תרגיל יתכנו הגבלות למשל, "לא מתאים לאנשים הסובלים מבעיות גב תחתון", "לא מתאים לנשים בהריון"	D		
30. לכל מתאמן שיוגדר במערכת, יוגדרו ההגבלות שלו (במידה וקיימות) מתוך רשימת ההגבלות	D		
31. יש לאפשר צפייה, הוספה ועדכון של רשימת הגבלות	O		נגזרת מ-30
32. יש לאפשר צפייה/הזנה/עדכון הגבלות למתאמן	O		נגזרת מ-30
33. יש לאפשר להנהלה הגדרה של תרגילים חלופיים לתרגיל בסדרה על פי הגבלה.	O		
34. יש לשמור תרגילים חלופיים בסדרה	D		נגזרת מ-33
35. תרגילים אלו גם הם ייחודיים לסדרה (מיקומם בסדרה יתחיל מ-1000 על מנת למנוע בלבול עם סדר התרגילים בסדרה) תרגיל חליפי הוא תרגיל אשר נועד להחליף תרגיל עם מגבלה מסוימת. ייתכן שאותו תרגיל חליפי, ישרת מספר תרגילים בעלי מגבלות שונות באותה הסדרה. למשל עבור תרגיל הנקרא- "hundred" המבוצע בשכיבה בראש מורם ועובד על שרירי הבטן יוגדר תרגיל חלופי המבוצע במנחה אחר בעמידה, זהו פתרון גם עבור מתאמנים הסובלים מבעיות גב וגם עבור נשים בהריון.			
36. למדריך צריכה להיות גישה לצפייה ברשימת ההגבלות של המתאמנים הרשומים לשיעור	O		נגזרת מ-35

הערות	סוג דרישה לא פונקציונאלית	סוג דרישה פונקציונאלית	נוסח הדרישה	
		O	הסדרות המובנות ותרגיליהם מנוהלות על ידי הנהלת הרשת הכללית, על כן יש לאפשר צפייה ברשימת הסדרות, בנייה של סדרה חדשה או עריכת סדרה קיימת.	37.
		O	בעת הכנסת עובד חדש על מנהלת המשרד להקצות עבורו מפקח. המפקח ילווה את המדריך החדש בחודשיו הראשונים	38.
נגזרת מ-38		D	יש לאפשר שמירה של מפקח למדריך. המפקח הינו מדריך אחר מהמכון שאינו מדריך חדש.	39.
		O	כל מערכי השיעור של המדריך החדש יחשפו לצפייה ועדכון של המדריך המפקח. בגין כל שינוי במערך השיעור של מדריך על ידי מפקח, המדריך החדש יקבל התרעה.	40.
		O	המתאמנים הנרשמים למכון יקבלו סיסמא ראשונית אותה יוכלו לשנות מתי שיחפצו	41.
		D	לכל מתאמן במערכת יש לשמור את שמו המלא, מספר הטלפון שלו שישמש כשם המשתמש שלו, סיסמא, ויתרת מספר האימונים בכרטיסיה שרכש	42.
נגזרת מ-42		O	יש לאפשר למנהלת המשרד לעדכן את פרטי הכרטיסיה ברכישת כרטיסיה חדשה	43.
נגזרת מ-42		O	יש לבצע הפחתה של יתרת הכרטיסיה לאחר כל שיעור	44.
		O	המתאמנים יכולים להירשם לשיעורים שונים . בעת כניסה לאפליקציה, המשתמשים יוכלו לצפות בלוז השיעורים המוצעים לחודש ויראו את מספר המקומות הפנויים בשיעור	45.
נגזרת מ-45		D	יש לשמור רישום של מתאמן לשיעור	46.
		O	במידה והשיעור מלא, כלומר מספר המשתתפים שנרשמו שווה למספר המיטות בחדר, המשתתף יוכל להירשם כממתין לשיעור זה. לכל נרשם לרשימת המתנה, יוקצה מיקום ברשימת הממתנים על בסיס זמן הרישום	47.
נגזרת מ-47		D	יש לשמור מספר מיטות בחדר	48.

הערות	סוג דרישה לא פונקציונאלית	סוג דרישה פונקציונאלית	נוסח הדרישה	
נגזרת מ-47		D	יש לשמור אילו מתאמנים ממתנים לשיעור, לכל מתאמן שממתין לשיעור יש לשמור את מיקומו על בסיס זמן הרישום.	49.
		I, O	במידה ואחד ממשתתפי השיעור מבטל את הרשמתו המערכת תגרע אותו מהשיעור ולאחר מכן תפעיל את אלגוריתם חיפוש מחליף. האלגוריתם יבדוק אם קיימת רשימת המתנה לשיעור, במידה וקיימת רשימת המתנה תשלח הודעת SMS עבור המתאמן הבא (הראשון) ברשימת המתנה. המתאמן יוכל להשיב בהודעה חוזרת על החלטתו (0 – לא מעוניין, 1- מעוניין), במידה והמתאמן החזיר 0 המערכת תגרע אותו מרשימת הממתנים. ותיצור קשר עם הממתין הבא בתור. במידה והשיב 1 המערכת תקצה לו מקום בשיעור ותגרע אותו מרשימת הממתנים. וכך הלאה עד לכדי איש המקום הפנוי בשיעור.	50.
I- אינטגרציה עם שירותי ספק SMS. C- שליחת הודעה ב-8 בבוקר O- המערכת שולחת תזכורת	C	I, O	בכל יום בשעה 08:00 תשלח תזכורת לאישור הגעה לאימונים של יום למחרת.	51.
I- אינטגרציה עם שירותי ספק SMS. O- טיפול המערכת במענה המתאמן		I, O	על המתאמן להשיב בהודעה חוזרת כדי לאשר את הגעתו	52.
נגזרת מ-52		D	יש לשמור האם המשתתף אישר הגעה לשיעור	53.
C- תפעיל אלגוריתם... O- תפעול המערכת	C	O	בחצות (של אותו היום), המערכת תבדוק אילו משתתפים לא אישרו את הגעתם. במידה וקיימים משתתפים שטרם אישרו את הגעתם המערכת תפעיל את אלגוריתם חיפוש המחליף. במידה ולא נמצא מחליף בטווח זה המתאמנים שלא אישרו את הגעתם יחויבו מיתרת כרטיסייתם בגין השיעור.	54.

הערות	סוג דרישה לא פונקציונאלית	סוג דרישה פונקציונאלית	נוסח הדרישה	
נגזרת מ-54		O	למדריך צריכה להיות אפשרות לצפות בכלל השיעורים שלו במערכת ורשימת המשתתפים בשיעור וההגבלות לכל מתאמן	55.
		O	מספר עובד אוטומטי שיווצר על ידי המערכת	56.

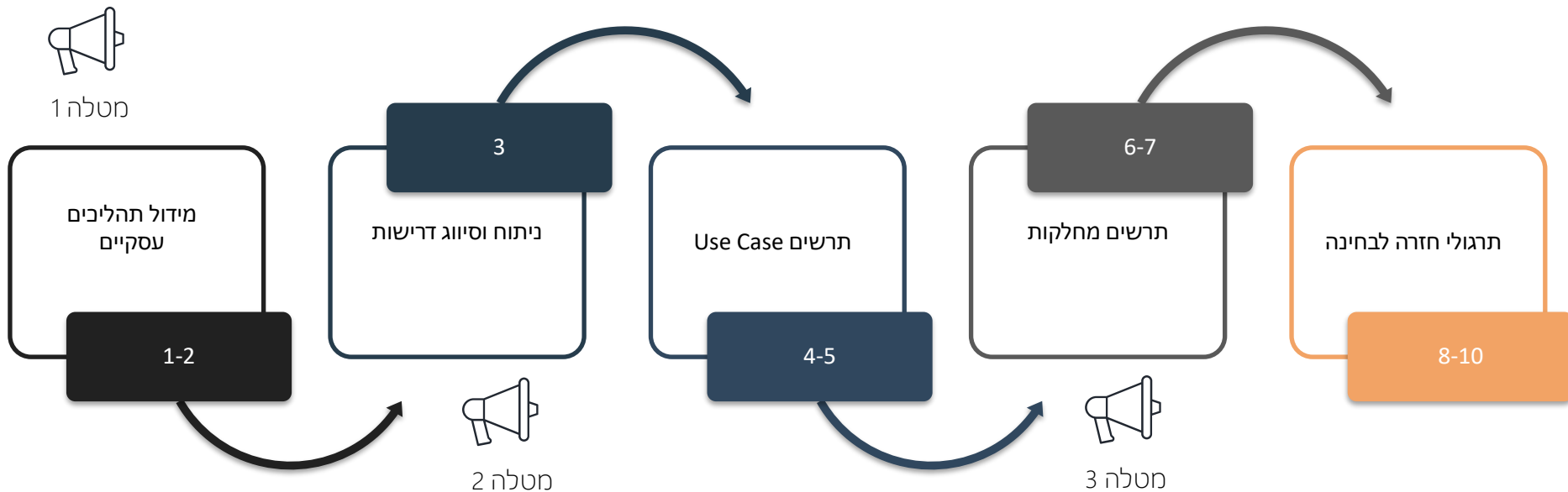
# ניתוח ועיצוב מערכות מידע

תרגול 4



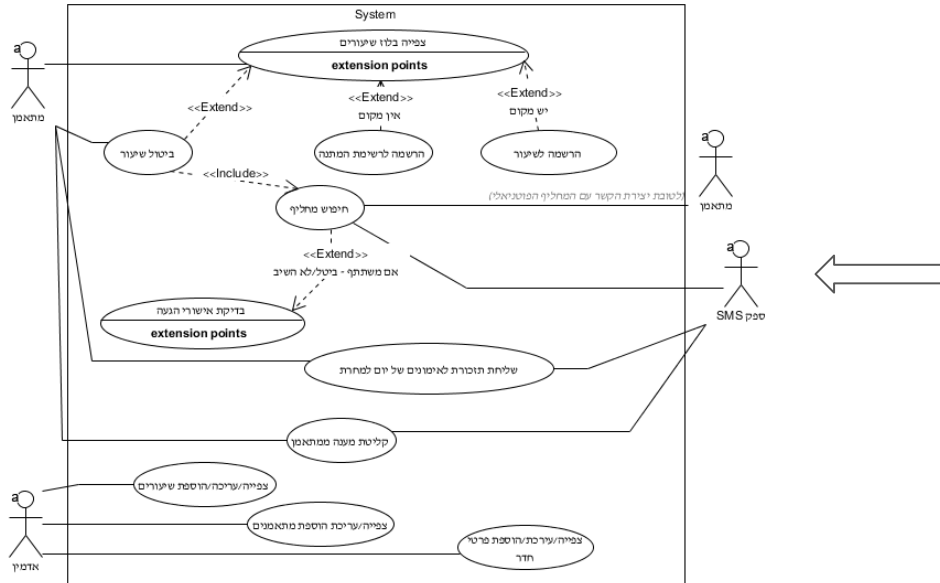


# מבנה הקורס



# USE CASE

תרשימים חוזרים ל-Visual Paradigm



תרחישים טבלה ב-Word

שם ה Use case	ביטול שיעור
תיאור קצר	מתאמן יוכל לבטל שיעור שהוא רשום אליו
שחקנים	מתאמן
תדירות	יומיומית/גבוהה.
Pre-conditions	המשתתף היה צריך להיות רשום לשיעור
Post-conditions	המתאמן נגרע מהשיעור
Triggers	המתאמן לחץ על כפתור ביטול שיעור.
MSS	<ol style="list-style-type: none"> <li>1. המערכת תבדוק את זמן השיעור- יש לפחות 3 שעות מהאימון</li> <li>2. המערכת תגרע מתאמן מהשיעור – גריעה הצליחה</li> <li>3. המערכת תפעיל את &lt;include&gt;אלגוריתם חיפוש מחליף – מחליף נמצא</li> <li>4. סיום תרחיש</li> </ol>
*הסתעפות א'	<p>חריגה לצעד 1 – זמן השיעור בטווח 3 השעות הקרובות</p> <p>1.1 המערכת תודיע כי לא ניתן לבטל</p> <p>סיום תרחיש</p>
*הסתעפות ב'	<p>חריגה לצעד 2 – גריעה לא הצליחה כתוצאה מנעילה של DB:</p> <p>2.1 שליחת הודעה למנהל וסיום תרחיש.</p>
*הסתעפות ג'	<p>חלופה לצעד 3 – מחליף לא נמצא</p> <p>3.1. חזרה לצעד 4.</p>



תרחישים

# נחזור לדרישות מתרגול 3



## דרישות פונקציונאליות

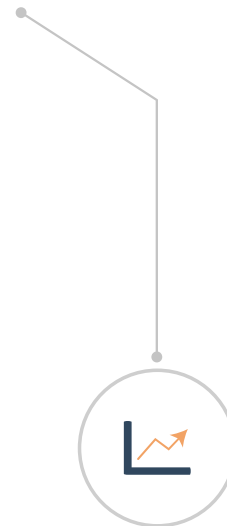
דרישות המייצגות את  
התכולה של המערכת.  
מתארות מה המערכת  
צריכה לבצע.

מה?

# התרחישים מורכבים מזרישות ה-O שלנו

מה המערכת צריכה או יכולה לעשות בעת קליטת פקודות

- בעת הזנת לקוח חדש, **תבדוק המערכת** האם הלקוח כבר קיים בה.
- בעת הזנת לקוח חדש, **המערכת תקצה** ללקוח מספר אוטומטי וסיסמא.
- **הלקוח יוכל לבצע** חיפוש עפ"י קטגוריות.
- **המערכת תאפשר** סריקת מסמכים.

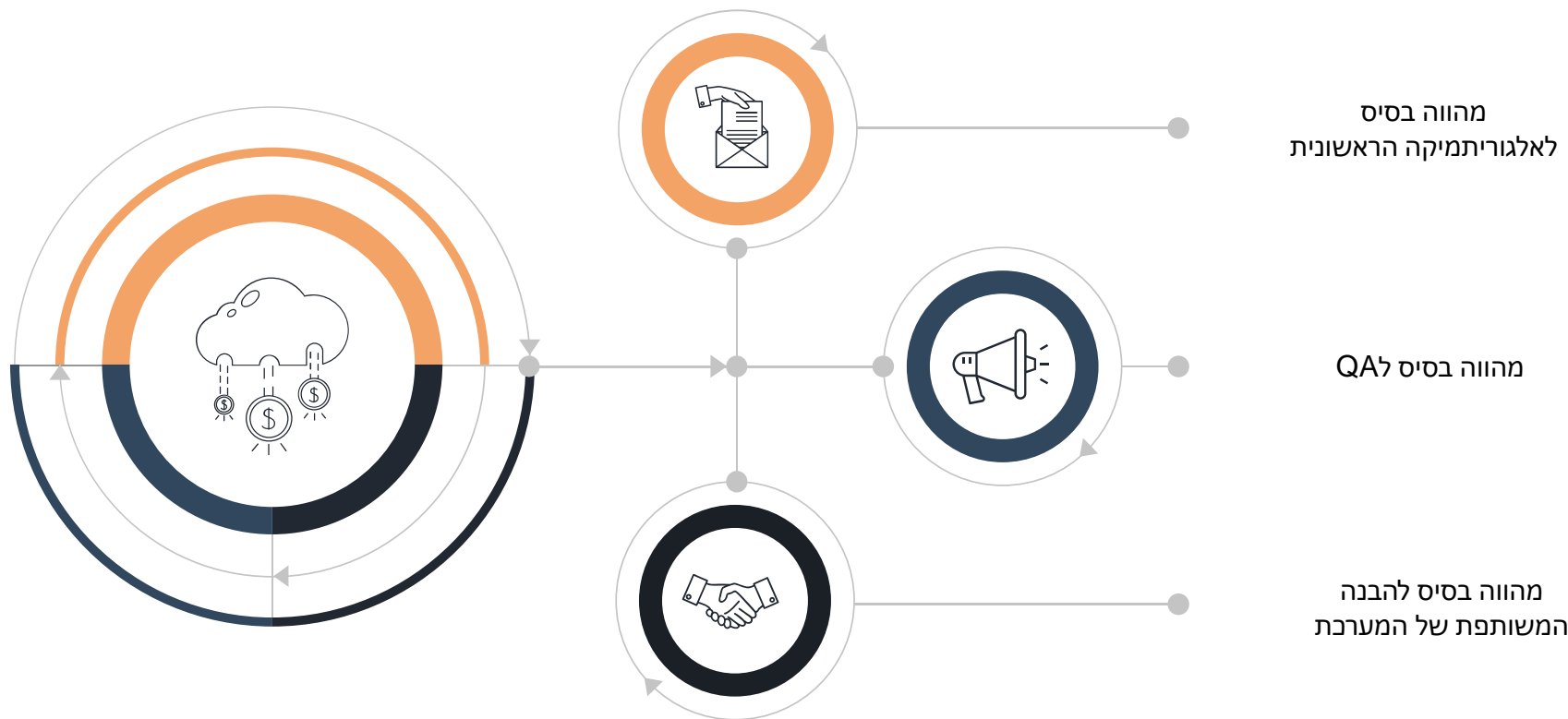


דרישות תפעוליות O

# מטרת ה-USE CASE



# למה צריך?



# הטבלה

שם ה Use case	
תיאור קצר	
שחקנים	
תדירות	
Pre-conditions	
Post-conditions	
Triggers	
MSS	
*הסתעפות א'	
*הסתעפות ב'	
*הסתעפות ג'	



# הטבלה

שם ה Use case	
תיאור קצר	כמה מילים פשוטות על ה-UC
שחקנים	מי בא באינטרקציה עם ה-UC שלנו, אנשים או מערכות אחרות שמתממשקים איתן <b>המערכת היא לעולם לא שחקנית לעצמה(!)</b> טיפ- דרישות ה-ו שהן לא גרפיות/וויזואליזציה
תדירות	לא תמיד יש. כמה פעמים אנחנו סוברים שה-UC הזה יתבצע
Pre-conditions	לא תמיד יש. תנאים הכרחיים ולא מובנים מאליהם לביצוע הפעולה, בלעדיהם זה לא יקרה לרוב נרשום- "המערכת תקינה ורכיביה תקינים".
Post-conditions	תמיד יש. מה יהיו ההשלכות על המערכת/העולם במידה ונסיבת השימוש תסתיים בהצלחה
Triggers	מה האירוע שיתחיל את הפעולה <b>לעולם לא יהיה חלק מהצעדים(!)</b> (למה? זה מה שהפעיל את זה בכלל)
MSS	התרחיש המוצלח- אם הכל היה עובד חלק בלי שגיאות ותקלות, השלבים לפיהם הוא היה מתבצע 1. 2.
*הסתעפות א'	הסתעפויות יכולות להיות בצורת חלופות או חריגות;
*הסתעפות ב'	חלופה- יצאתי מה-MSS אבל הצלחתי לחזור אליו חריגה- יצאתי מה-MSS ולא הצלחתי לחזור אליו. הפוסט קונדישן שלי לא התמלא, התהליך הסתיים ללא הגשמת המטרה

# קשרי הכלה <include>



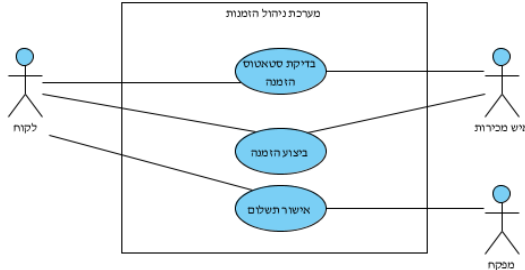
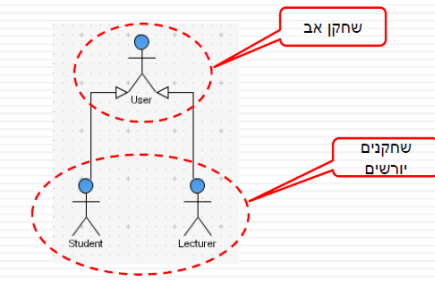
כשדרישה פונקציונאלית א' כוללת באופן מפורש את דרישה פונקציונאלית ב' בתוכה

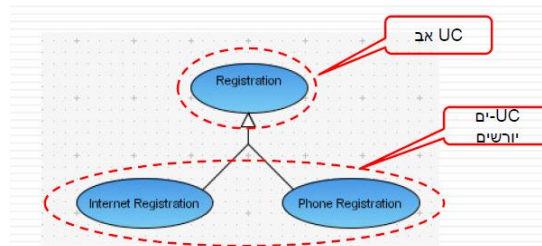
- למשל, תשלום בכרטיס אשראי **גורר** בדיקת תקינות כרטיס אשראי
- אין משמעות למה מגיע לפני מה אלא **מה מחייב את מה**
- UC יכול להיות מוכל במספר Use-Cases שונים



## תרגול UC-5

### סיכום האלמנטים שנלמדו בהרצאה:

	<p><b>Use-Case (או בעברית, נסיבת שימוש)</b></p> <ul style="list-style-type: none"> <li>סט של פעולות סדרתיות שהמערכת מבצעת</li> <li>פעולות אלו מהוות דרישה פונקציונלית הנדרשת מהמערכת</li> <li>כיצד נזהה Use-Case? <ul style="list-style-type: none"> <li>איזה פונקציות צריכים המשתמשים מהמערכת?</li> <li>האם המשתמשים צריכים לקרוא, ליצור, לעדכן, או לשמור סוג של מידע כלשהו במערכת?</li> <li>באילו מובנים המערכת "משרתת" את המשתמשים בה.</li> </ul> </li> </ul> <p><b>דוגמא – הבועה – משיכת כסף (מכספומט) מורכבת מסט הצעדים:</b></p> <ol style="list-style-type: none"> <li>1. Include &lt;ביצוע הזדהות&gt; - הזדהות הצליחה.</li> <li>2. המערכת תציג על המסך סכומים למשיכה או הגדרת סכום אחר.</li> <li>3. אם המשתמש בחר סכום אחר – המערכת תבקש מהמשתמש להקליד סכום.</li> <li>4. מערכת תחייב את חשבון הלקוח ותוציא כסף ללקוח.</li> </ol>
	<p><b>שחקן (Actor)</b></p> <ul style="list-style-type: none"> <li>מיצג ישות חיצונית הנמצאת באינטראקציה עם המערכת</li> <li>ישות חיצונית יכולה להיות אדם, מכשיר (חומרה) או מערכת אחרת</li> <li>שחקן הוא סוג (מחלקה ולא אובייקט) ולא מופע בודד</li> <li>כיצד לזהות שחקן? <ul style="list-style-type: none"> <li>מי משתמש בפונקציונליות המערכת?</li> <li>מי יזדקק לתמיכה מהמערכת כדי לבצע משימות?</li> <li>מי יצטרך לתחזק ולתפעל את המערכת?</li> <li>באילו התקני חומרה המערכת צריכה לטפל?</li> <li>עם אילו מערכות אחרות צריכה המערכת לתקשר?</li> <li>מי או מה מעוניינים בתוצאות שהמערכת תספק?</li> </ul> </li> </ul>
	<p><b>קשר אסוציאטיבי (Association)</b></p> <ul style="list-style-type: none"> <li>מקשר בין שחקן (Actor) לבין ה- Use-Case בו הוא משתתף</li> <li>לשחקן יש עניין / השתתפות ב use case אליה הוא מחובר</li> </ul>
	<p><b>קשרי הכללה (Generalization)</b></p> <p><u>הכללה בין שני שחקנים – שחקן הן יורש את התנהגות שחקן האב ויכול להרחיבה</u></p> <ul style="list-style-type: none"> <li>השחקן היורש, יורש את כל הקשרים של שחקן האב</li> <li>כל מה שהאבא יכול לבצע במערכת – גם הן יכול לבצע</li> <li>הן יכול לבצע פעולות משלו – האב אינו יכול לבצע</li> </ul> <p><b>מתי להשתמש בהורשה של שחקנים?</b></p> <ul style="list-style-type: none"> <li>כאשר לבן היורש יש יותר אפשרויות מאשר לאב</li> <li>כאשר רוצים להגביר את קריאות המודל</li> </ul>



### הכללה (Generalization)

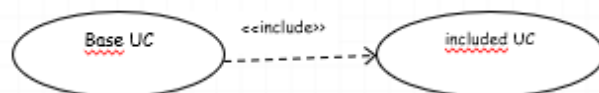
הכללה בין שני Use-Cases – ה- Use-Case הבן הוא מקרה ספציפי של Use-Case האב

- התנהגות הבן יורשת את התנהגות האב
- יורשת את הקשרים (עם שחקנים/ use cases אחרים)
- התנהגות הבן יכולה להחליף את התנהגות האב

דוגמא – לבועה תשלום יתכנו שלושה מקרים פרטיים: תשלום בכרטיס אשראי, תשלום בפייפל או תשלום במזומן.

### קשרי הכלה (Include)

- Use case Base (דרישה פונקציונלית א') כולל באופן מפורש את ה included use case (את דרישה פונקציונלית

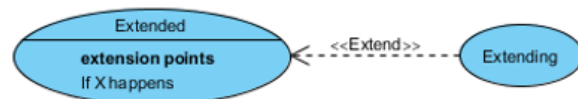


(ב')

- למשל, תשלום בכרטיס אשראי גורר בדיקת תקינות כרטיס אשראי
- אין קשר למה לפני מה אלא מה מחייב את מה
- Use-Case יכול להיות מוכל במספר Use-Cases שונים

### קשרי הרחבה (Extend)

Use case extended מוסיף ומרחיב את base use case בנסיבות



מסוימות.

למשל, בעת סיום ביצוע הזמנה ניתן (אך לא חובה) להדפיס הזמנה

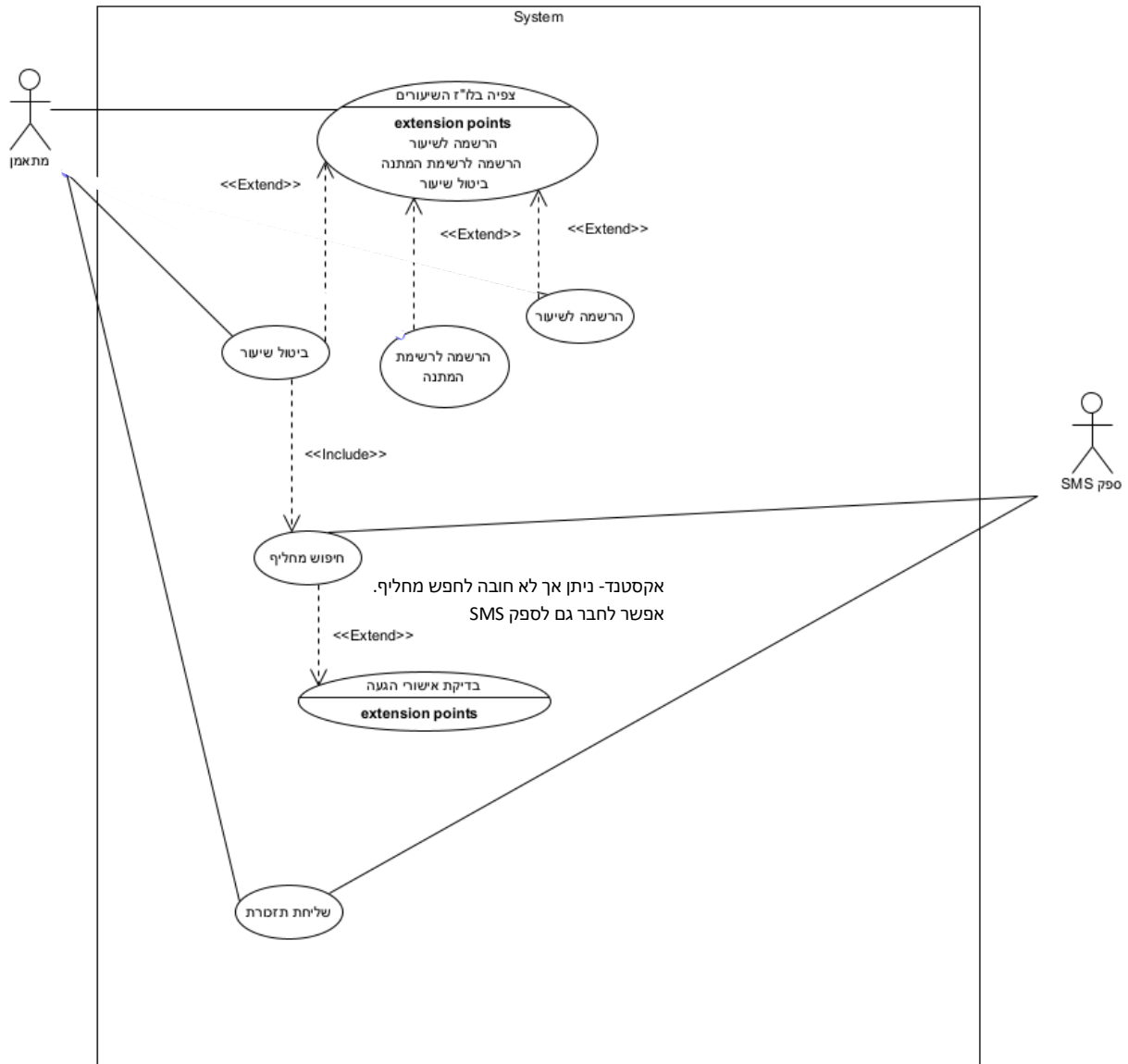
יש לשרטט עבור התרגילים הבאים תרשים UC:

**שאלה 1:**

סוג דרישה פונקציונאלית	נוסח הדרישה	
O	המתאמנים יכולים להירשם לשיעורים שונים. בעת כניסה לאפליקציה, המשתמשים יוכלו לצפות בלוח השיעורים המוצעים לחודש ויראו את מספר המקומות הפנויים בשיעור	1.
O	במידה והשיעור מלא, כלומר מספר המשתתפים שנרשמו שווה למספר המיטות בחדר, המשתתף יוכל להירשם כמתמין לשיעור זה. לכל נרשם לרשימת המתנה, יוקצה מיקום ברשימת הממתנים על בסיס זמן הרישום	2.
O, I	במידה ואחד ממשתתפי השיעור מבטל את הרשמתו המערכת תגרע אותו מהשיעור ולאחר מכן תפעיל את אלגוריתם חיפוש מחליף. האלגוריתם יבדוק אם קיימת רשימת המתנה לשיעור, במידה וקיימת רשימת המתנה תשלח הודעת SMS עבור המתאמן הבא (הראשון) ברשימת ההמתנה. המתאמן יוכל להשיב בהודעה חוזרת על החלטתו (0 – לא מעוניין, 1- מעוניין), במידה והמתאמן החזיר 0 המערכת תגרע אותו מרשימת הממתנים. ותיצור קשר עם המתמין הבא בתור. במידה והשיב 1 המערכת תקצה לו מקום בשיעור ותגרע אותו מרשימת הממתנים. וכך הלאה עד לכדי איוש המקום הפנוי בשיעור.	3.
O, I	בכל יום בשעה 08:00 תשלח תזכורת לאישור הגעה לאימונים של יום למחרת.	4.
O, I	על המתאמן להשיב בהודעה חוזרת כדי לאשר את הגעתו.	5.
D	יש לשמור האם משתתף אישר הגעה לשיעור	6.
I, O	בחצות (של אותו היום), המערכת תבדוק אילו משתתפים לא אישרו את הגעתם. במידה וקיימים משתתפים שטרם אישרו את הגעתם המערכת תפעיל את אלגוריתם חיפוש המחליף. במידה ולא נמצא מחליף בטווח זה המתאמנים שלא אישרו את הגעתם יחויבו מיתרת כרטיסייתם בגין השיעור.	7.

שם ה Use case	חיפוש מחליף
תיאור קצר	התרחיש מטפל במצב בו משתתף ביטל שיעור ומטרתו לאיש את השיעור באמצעות רשימת ההמתנה
שחקנים	מתאמן, ספק SMS
תדירות	יומיומית - גבוהה
Pre-conditions	קיים שיעור במערכת, המערכת תקינה ורכיביה תקינים.
Post-conditions	מחליף נמצא והמקום הפנוי בשיעור אויש
Triggers	משתתף ביטל שיעור
MSS	1. המערכת תבדוק אם קיימת רשימת המתנה לשיעור- קיימת 2. המערכת תשלח הודעת SMS עבור המתאמן הבא ברשימת ההמתנה- שליחה הצליחה 3. ממשק SMS יחזיר תשובה- המשתתף החזיר 1 (מעוניין בשיעור) 4. המערכת תקצה לו מקום בשיעור ותגרע אותו מרשימת הממתנים. 5. סיום תרחיש
*הסתעפות א'	חריגה ל- צעד - לא קיימת חלופת המתנה 1.1 סיום התרחיש.
*הסתעפות ב'	חלופה לצעד 2- שליחה לא הצליחה וכמות הניסיונות קטנה מ-3 2.1 ניסיון שליחה חוזר- שליחה הצליחה 2.2 חזרה לצעד 3
* הסתעפות ג'	חריגה לצעד 2.1- שליחה לא הצליחה וכמות הניסיונות גדולה מ-3: 2.1.1 שליחת הודעת שגיאה למנהל מערכת לצורך המשך טיפול
* הסתעפות ד'	חלופה לצעד 3- המשתתף החזיר 0 (לא מעוניין בשיעור) 3.1 גרעת מתאמן מרשימת המתנה ו-חזרה לצעד 2
* הסתעפות ה'	חלופה לצעד 3- עברה רבע שעה ללא מענה: 3.1 העברת מתאמן לסוף רשימת המתנה - חזרה לצעד 2.

שם ה Use case	ביטול שיעור
תיאור קצר	מתאמן יוכל לבטל שיעור שהוא רשום אליו.
שחקנים	מתאמן
תדירות	יומיומית/גבוהה.
Pre-conditions	המשתתף היה צריך להיות רשום לשיעור, המערכת תקינה ורכיביה תקינים.
Post-conditions	המתאמן נגרע מהשיעור
Triggers	המתאמן לחץ על כפתור ביטול שיעור.
MSS	1. המערכת תבדוק את זמן השיעור- יש לפחות 3 שעות מהאימון 2. המערכת תגרע מתאמן מהשיעור- גריעה הצליחה 3. המערכת תפעיל את <include>אלגוריתם חיפוש מחליף – מחליף נמצא 4. סיום תרחיש
*הסתעפות א'	חריגה לצעד 1 – זמן השיעור בטווח 3 השעות הקרובות 1.1 המערכת תודיע כי לא ניתן לבטל סיום תרחיש
*הסתעפות ב'	חריגה לצעד 2 – גריעה לא הצליחה כתוצאה מנעילה של DB: 2.1 שליחת הודעה למנהל וסיום תרחיש.
* הסתעפות ג'	חלופה לצעד 3 – מחליף לא נמצא 3.1 . חזרה לצעד 4.



## שאלה 2:

- בריכת השחייה "כמו דג במים" מעוניינת לרכוש מערכת אוטומטית האחראית לתחזוקת בריכת השחייה. המערכת המוצעת מורכבת ממחשב מרכזי, מרובוט אלקטרוני, חיישנים שמורכבים בדפנות הבריכה (לרבות מד טמפרטורה ממוחשב) וברדים.
- המערכת תומכת במספר תכניות ניקיון למשל ניקיון בריכת אימונים, ניקיון בריכת תינוקות, ניקיון בריכה ביתית, ציבורית וכו'. בעת אתחול המערכת, על מפעיל הבריכה לבחור את התכנית הרצויה מתוך רשימת התוכניות במערכת (ניתן אף לשנות את התכנית בעת הצורך). לאחר הגדרת התוכנית עליו להזין את גודל הבריכה ומס' משתמשים משוער ביום.
- על בריכת השחייה לעמוד בתקנים של משרד הבריאות על מנת לקבל אישור הפעלה ולכן בכל שעה עגולה נמדדים מדדים הקובעים את איכות המים (המערכת תשמור את מועדי הבדיקה ותוצאות המדדים). המדדים נמדדים ע"י החיישנים המותקנים בדפנות הבריכה ומועברים למערכת אחת לשעה. עבור כל מדד שנקלט המערכת מבצעת בדיקות תקינות מדדים:
  - מדד ראשון הוא רמת ה PH במים, רמת PH תקינה צריכה להיות בין 7.2 – 7.6. כאשר רמת ה PH גבוהה מידי (רמת כלור גבוהה) יפתחו ברזי המים ויזרמו מים חמים/קרים לבריכה על מנת לדלל את כמות הכלור, אם רמת ה PH נמוכה יזרם כלור לבריכה בכמות שתחושב ע"י המחשב עפ"י כמות ה PH שנמצאה וכמות המים בבריכה.
  - המדד השני הוא עכירות המים, אם המים עכורים ביחס לסף המותר המערכת שולחת הודעה למפעיל הבריכה. מפעיל הבריכה אחראי להוציא את האנשים מהבריכה, לאחר הוצאת האנשים מהבריכה, מפעיל הבריכה יעדכן את המערכת בדבר וכתוצאה מכך המערכת תפעיל את הרובוט לטיפול אקטיבי של ניקוי הבריכה.
  - המדד השלישי (נבדק ע"י מד החום) הוא טמפרטורת המים. הטמפרטורה המותרת בבריכות שחיה היא בין 24-29.5 מעלות (ניתן לקבוע על פי העונה), אם הטמפרטורה עולה או יורדת מטווח זה יזרמו מים חמים/קרים בהתאם לצורך.
- מכיוון שביצוע פעולה עבור תיקון אחד המדדים יכול להשפיע על תקינות מדד אחר (למשל, הזרמת מים חמים/קרים לבריכה יכולה להוריד את רמת ה PH במים), ברגע שמתבצע תיקון של אחד המדדים, כל המדדים יבדקו מחדש ביחס לערכים חדשים שידגמו.
- מפעיל הבריכה יוכל לעדכן מהי הטמפרטורה המומלצת לפי העונה בשנה.
- מפעיל הבריכה יוכל להפיק דוחות המתייחסים לתקינות המים עפ"י תאריכים ושעות או עפ"י מדדים.

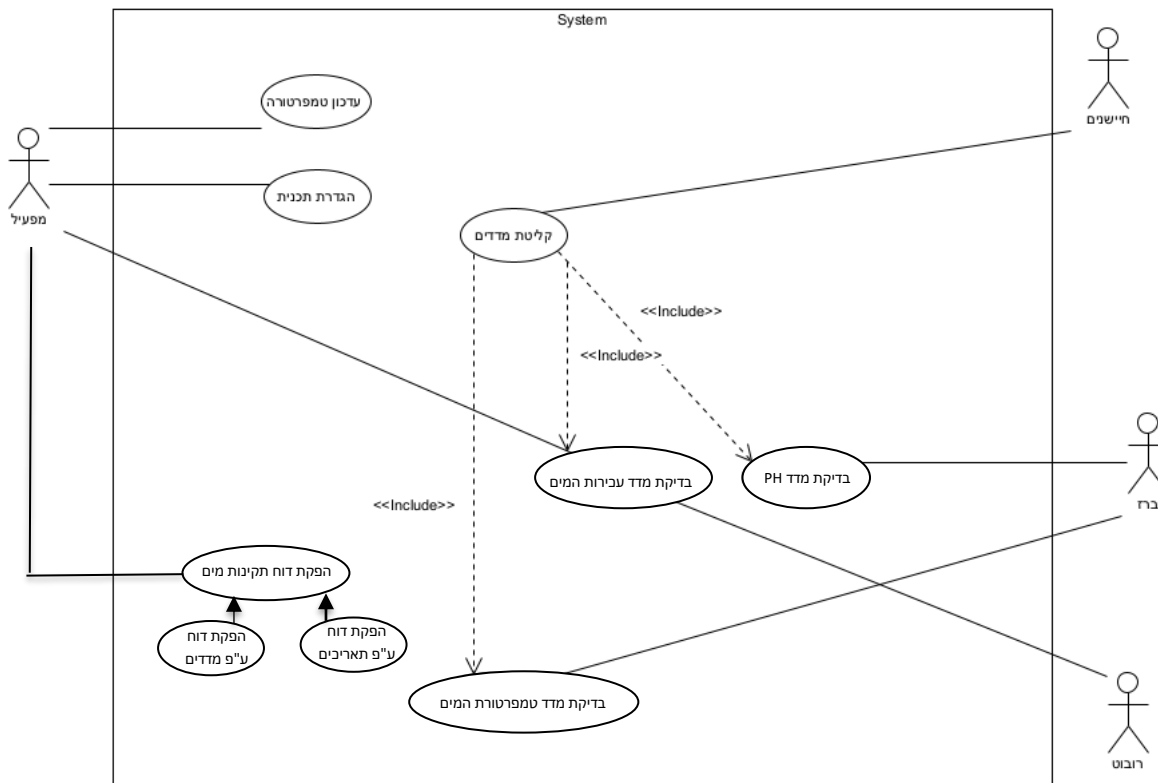


שם ה Use case	בדיקת רמת PH
תיאור קצר	התרחיש מטפל בבדיקת רמת PH במים וטיפול במידה והמדד אינו תקין
שחקנים	ברזים, חיישן
תדירות	יומיומית- לפחות אחת לשעה
Pre-conditions	ברזי מים תקינים, המערכת תקינה ורכיביה תקינים
Post-conditions	מדד PH תקין
Triggers	קליטת מדד חדש במערכת
MSS	1. בדיקה האם רמת PH בין 7.2-7.6 - נמצא בטווח 2. שמירת תוצאות בדיקה וסיום
*הסתעפות א'	חלופה לצעד 1 - רמת PH גבוהה מהטווח 1.1 חישוב כמות המים להזרמה ופתיחת ברז מים. 1.2 חזרה לצעד 2
*הסתעפות ב'	חלופה לצעד 1 - רמת PH נמוכה מהטווח 1.1 חישוב כמות הכלור להזרמה ופתיחת ברז כלור. 1.2 חזרה לצעד 2

שם ה Use case	בדיקת עכירות מים
תיאור קצר	מטרת השימוש הינה לבדוק את רמת עכירות המים ולטפל במידה והמים אינם צלולים
שחקנים	מפעיל, רובוט.
תדירות	יומיומית- לפחות אחת לשעה
Pre-conditions	רובוט תקין
Post-conditions	המים אינם עכורים
Triggers	קליטת מדד חדש במערכת
MSS	1.בדיקה האם המים עכורים ביחס לסף המותר- לא עכורים 2.שמירת תוצאות בדיקה וסיום
*הסתעפות א'	חלופה לצעד 1- המים עכורים: 1.1 המערכת שולחת הודעה למפעיל הבריכה על עכירות המים 1.2 מפעיל הבריכה עדכן כי סיים להוציא אנשים מהמים 1.3 המערכת תפעיל את הרובוט לטובת הניקיון. חזרה לצעד 2.
*הסתעפות ב'	1.2 חריגה לצעד 1- עברה שעה מהרגע בו התגלה מדד לא תקין: 1.2.1 המערכת תשלח הודעה למנהל הבריכה ותפעיל פעמון. סיום תרחיש.

שם ה Use case	בדיקת טמפרטורת המים
תיאור קצר	מטרת השימוש הינה לבדוק את טמפרטורת המים ולטפל במידה והמים בטמפרטורה הלא נכונה
שחקנים	ברזים, מד טמפרטורה (חיישנים)
תדירות	יומיומית- לפחות אחת לשעה
Pre-conditions	ברזי מים תקינים, מד חום תקין, המערכת תקינה ורכיביה תקינים.
Post-conditions	המים אינם עכורים
Triggers	קליטת מדד חדש במערכת
MSS	1.בדיקה האם הטמפרטורה בבריכות שחיה היא בין 24-29.5 מעלות- נמצא בטווח 2.שמירת תוצאות בדיקה וסיום
*הסתעפות א'	חלופה לצעד 1- הטמפרטורה לא בטווח: 1.1 הזרמת מים חמים/קרים לבריכה בהתאם לצורך. 1.2 חזרה לצעד 2.

שם ה Use case	קליטת מדדי תקינות מים
תיאור קצר	התרחיש מטפל בבדיקה של מדדי המים וטיפול במדדים לא תקינים
שחקנים	חיישנים
תדירות	אחת לשעה
Pre-conditions	המערכת תקינה ורכיביה תקינים
Post-conditions	כל המדדים תקינים
Triggers	הגיעה שעה עגולה
MSS	<p>1. ניסיון התחברות לחיישנים- ההתחברות הצליחה</p> <p>2. קליטת מדדים מחיישנים: PH, עבירות וחיישן חום- קליטת מדדים הצליחה</p> <p>3. הפעלת בדיקות תקינות:</p> <p>a. &lt;include&gt; הפעלת בדיקת תקינות רמת PH</p> <p>b. &lt;include&gt; הפעלת בדיקת תקינות עבירות מים</p> <p>c. &lt;include&gt; הפעלת בדיקת תקינות חום</p> <p>4. כל המדדים תקינים- תיעוד במערכת וסיום תרחיש</p>
*הסתעפות א'	<p>חלופה לצעד 1- ההתחברות לא הצליחה ומספר ניסיונות קטן מ3:</p> <p>1.1 חזרה לצעד 1</p>
*הסתעפות ב'	<p>חריגה לצעד 1- ההתחברות לא הצליחה ומספר ניסיונות שווה מ3:</p> <p>1.1 שמירת שגיאה בקובץ לוג ושליחת הודעת שגיאה למפעיל</p>
*הסתעפות ג'	<p>חלופה לצעד 2- קליטה לא הצליחה ומספר ניסיונות קטן מ3:</p> <p>1.1 חזרה לצעד 2</p>
*הסתעפות ד'	חלופה לצעד 4- מדדים לא תקינים- חזרה לצעד 1.



## סיכום האלמנטים שנלמדו בהרצאה:

מתוארת ע"י שלושה חלקים:

- **תכונות המחלקה**

- עבור כל תכ

- (String...

- פעולות המחלקה

- פעולה (מתן)

- ובינך לבין ה' ויבטלך.

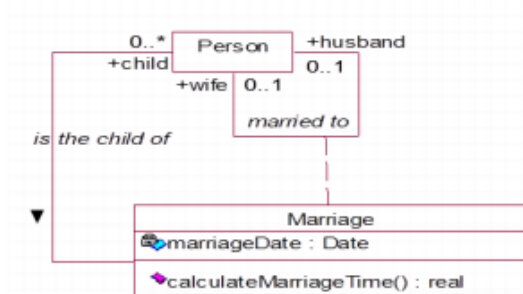
- יום כו מאלול:

קשרים בין מחלקות:

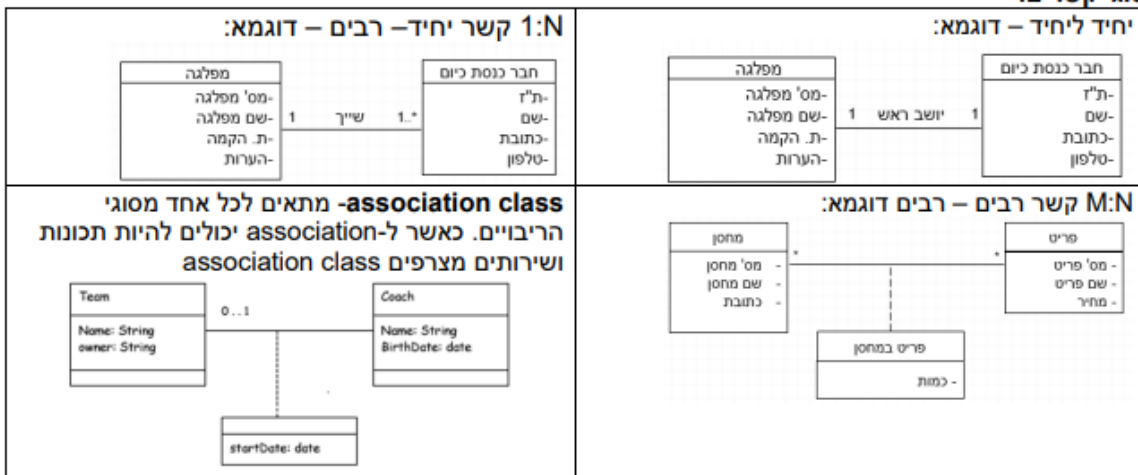
**Association** מתאר קשר זיקה ("היכרות") בין 2 מחלקות. מתואר בדרך כלל ע"י פועל. יתכן גם קשר בין המחלקה לבין עצמה (self-association).

על כל קשר יכולים להופיע:

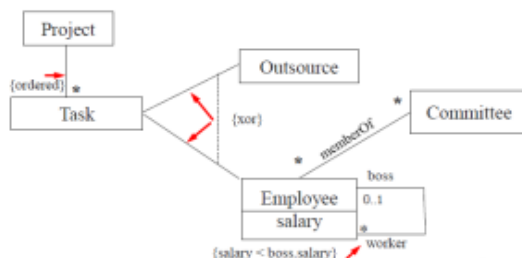
- שם הקשר (חד כיווני או דו כיווני).  
 ריבוי (multiplicity) מציינ כמה מופעים של המחלקה יכולים להיות מקושרים (link) למופע אחד של המחלקה השנייה. ברירת המחדל היא בדיוק 1.  
 תפקיד (role) מציינ מה תפקידו של כל אחד מהעצמים בקשר. שימוש ב-role נחוץ בעיקר כאשר הקשר מחבר בין המחלקה לעצמה (self-association) או כאשר קיימים מספר קשרים בין זוג מחלקות.



יחיד ליחיד – דוגמא:

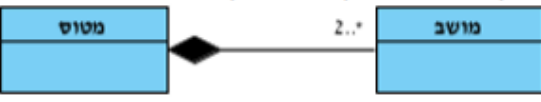

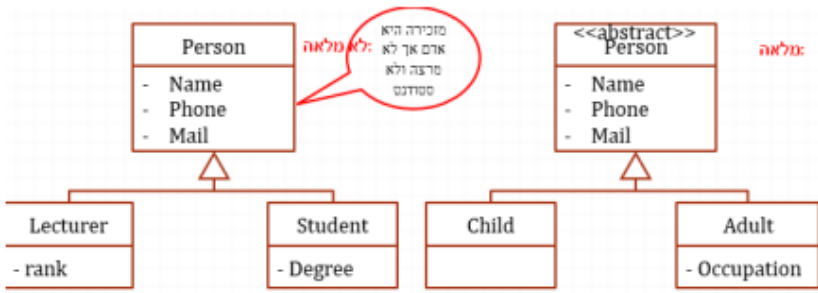
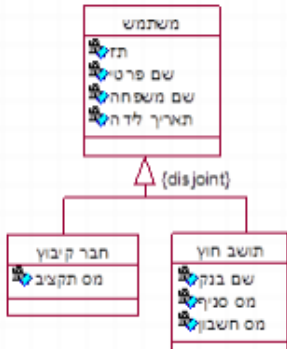
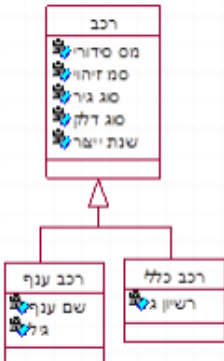


**אילוצים Constraints** - טקסט המייצג אילוץ סמנטי יסומן ב\} על קשר (אם מאלץ את הקשר) או תחת המחלקה (אם מאלץ את אחד השדות)



Window
length
width

$\{0.8 \leq \text{length}/\text{width} \leq 1.5\}$

<p><b>קשר הרכבה (Composition)</b></p> <p>0 ניתן להתייחס לעצמים של אחת המחלקות כחלקים של עצם במחלקה השנייה.</p> <p>0 קיומם של החלקים תלוי בקיומו של השלם.</p> 	<p><b>קשר הכלה (Aggregation)</b></p> <p>0 ניתן להתייחס לעצמים של אחת המחלקות כחלקים של עצם במחלקה השנייה.</p> <p>0 לחלקים יש קיום עצמאי משלהם.</p> 
<p><b>קשרי הורשה: הכללה (inheritance, generalization)</b></p> <p>0 הגדרת מחלקה על בסיס מחלקה אחרת.</p> <p>0 המטרה של הורשה הוא לצמצם את גודל הקוד הנוצר ע"י בניית היררכיה נכונה.</p> <p>0 תתי המחלקות יורשות את כל התכונות, המתודות והקשרים של מחלקת העל ויכולים להוסיף עליהם (ואף לשכתב אותם).</p> <p><b>סוגי הורשות – הורשה מלאה/לא מלאה</b></p> <p>0 בהורשה מלאה (Total), כל עצם ממחלקת העל משתייך לאחת מתתי המחלקות לפחות. נראות - &lt;abstract&gt; מעל שם המחלקה.</p> <p>0 בהורשה לא מלאה (Partial), קיים עצם ממחלקת העל שאיננו משתייך לאף אחת מתתי המחלקות. נראות – ברירת מחדל ללא סימון מיוחד.</p>  <p><b>סוגי הורשות – overlapping/disjoint</b></p> <p>0 בזרות (disjoint), כל עצם ממחלקת העל משתייך לכל היותר לאחת מתתי המחלקות.</p> <p>0 בחפיפה (overlapping), קיים עצם ממחלקת העל שמשתייך ל-2 או יותר מתתי המחלקות.</p>	
	

**מכרז פומבי מספר 12345**

**הקמה ותפעול של מערכת PilatesMoodle**

מנהלת המשרד, אחראית על קליטת פרטי המדריכים למערכת והגדרת סיסמא ראשונית לכניסת מדריך. לכל מדריך ישמרו שמו, תאריך התחלת עבודתו, מספר הטלפון שברשותו ומספר עובד אוטומטי שייווצר על ידי המערכת.

במערכת ינוהלו כלל השיעורים (אימונים) המתקיימים במכון. לכל שיעור ישמרו תאריך ושעת האימון, המדריך שמקיים את השיעור והחדר בו מתקיים מתוך רשימת חדרי המכון. המערכת תאפשר למדריך לבנות את מערך השיעור עבור כל שיעור שנקבע במערכת: המדריך יוכל לבחור סדרות של תרגילים ולשייכם לשיעור. כל סדרה מאופיינת על ידי מנח הגוף (מחרוזת המתארת את מנח הגוף בסדרה, למשל "ישיבה זקופה על עצמות המושב עם רגלים מעתה"). הסדרות השונות מובנות מסט של תרגילים ברמת קושי עולה על כן לכל תרגיל יש חשיבות למיקומו בסדרה. לכל תרגיל המערכת תשמור את תיאור מילולי של התרגיל, כמו כן גם את משך זמן הביצוע ומספר החזרות לתרגיל זה. התרגילים ייחודים לסדרות שבהם הם נמצאים. לכל תרגיל יתכנו הגבלות למשל, "לא מתאים לאנשים הסובלים מבעיות גב תחתון", "לא מתאים לנשים בהריון". על מנת לבצע התאמה בין תרגילים למתאמנים, לכל מתאמן שיוגדר במערכת, יוגדרו ההגבלות שלו (במידה וקיימות) מתוך רשימת ההגבלות. בכל סדרה ניתן להגדיר גם מספר תרגילים חליפיים, תרגילים אלו גם הם ייחודיים לסדרה (מיקומם בסדרה יתחיל מ-1000 על מנת למנוע בלבול עם סדר התרגילים בסדרה) תרגיל חליפי הוא תרגיל אשר נועד להחליף תרגיל עם מגבלה מסוימת. ייתכן שאותו תרגיל חליפי, ישרת מספר תרגילים בעלי מגבלות שונות באותה הסדרה. למשל עבור תרגיל הנקרא-"hundred" המבוצע בשכיבה בראש מורם ועובד על שרירי הבטן יוגדר תרגיל חלופי המבוצע במנח אחר בעמידה, זהו פתרון גם עבור מתאמנים הסובלים מבעיות גב וגם עבור נשים בהריון. הסדרות המובנות ותרגיליהם מנוהלות על ידי מנהלת הרשת הכללית.

יש לציין כי עבור מדריכים חדשים, (מדריכים אשר הותק שלהם במכון קטן משלושה חודשים), מחויבים בשיבוץ מפקח. המפקח הינו מדריך אחר מהמכון שאינו מדריך חדש. בעת הכנסת עובד חדש על מנהלת המשרד להקצות עבורו מפקח. המפקח ילווה את המדריך החדש בחודשיו הראשונים.

המתאמנים הנרשמים למכון יקבלו סיסמא ראשונית אותה יוכלו לשנות מתי שיחפצו. לכל מתאמן במערכת יש לשמור את שמו המלא, מספר הטלפון שלו שישמש כשם המשתמש שלו, סיסמא, ויתרת מספר האימונים בכרטיסיה שרכש.

המתאמנים יכולים להירשם לשיעורים שונים. בעת כניסה לאפליקציה, המשתמשים יוכלו לצפות בלוח השיעורים המוצעים לחודש ויראו את מספר המקומות הפנויים בשיעור. במידה והשיעור מלא, כלומר מספר המשתתפים שנרשמו שווה למספר המיטות בחדר, המשתתף יוכל להירשם כמתין לשיעור זה. לכל נרשם לרשימת המתנה, יוקצה מיקום ברשימת הממתנים על בסיס זמן הרישום.

### צרו תרשים מחלקות (Class Diagram) על בסיס הסיפור.

