

תאריך כתיבת המטלה - 4.4.20

liorwunsch@gmail.com

vladigr1@gmail.com

eldadkro@gmail.com

elroye77@gmail.com

liadvv@gmail.com

קבוצה 1

לי אור וונש

ולדיסלב ברקנס

אלדד קרונפלד

אלרואי כהנא

ליעד וקסמן

שאלות פתוחות

שאלה 1:

בעזרת מודל UC ניתן לראות באופן ויזואלי ופשוט את הפעולות שכל משתמש במערכת יכול לבצע, המודל גם מציג באופן כללי וקצר את מטרת המערכת. מהמדול שלנו ניתן להסיק כי המערכת מאפשר ללקוחות לרכוש דלק בתחנות הדלק של המערכת ובנוסף גם מתחזקה מלאי דלק.

דוגמאות פרטניות במערכת "myFuel":

1. יוצר היררכיה של שחקנים - אפשר לפרש מה כל שחקן יכול לבצע במערכת
• לכל צרכן של המערכת יש גישה להתנהגות מסוימת של המערכת, כלומר התנהגויות המערכת ממודרות ע"פ המשתמשים שלה למשל:
- המשתמש Client: לקנות/להזמין מוצר מהמערכת, לבחור סוג רכישה, לעקוב אחרי סטטוס של הזמנה מסוג "דלק ביתי"
- המשתמש Marketing Manager: לאשר שינוי של תעריף, לפרסם דו"ח שיווקי
Marketing Representative - : לרשום לקוח אל המערכת ולשייך אותו לתוכנית רכישה
• כל צרכן אשר רוצה לקבל גישה מהמערכת תחילה חייב להיות מחובר אל המערכת לכן ישנו שחקן בשם User אשר כל הצרכנים אשר הם בני אדם הם יורשים ממנה על מנת שיתבצע התחברות למערכת

2. מציג ויזואלית דרישות מורכבות ממסמך הדרישות

• הצרכן Fuel Station Manager יכול לבצע אישור הזמנה של מלאי חסר לכן כאשר מתבצע כזה אישור המערכת יוצרת הזמנה של מלאי חסר, משום שלא תמיד המנהל מאשר ניתן להבין זאת בעזרת קישור ה- extend.

3. מציג את השחקנים האינטראקטיביים לכל תרחיש פונקציונלי

• לדוגמה יש הרבה שחקנים במחלקת המכירות אבל אנחנו מיד יודעים שמנהל מכירות יוצר מבצע מכירות.

שאלה 2:

דרישות פונקציונליות שלא הצלחנו לבטא בעזרת UC

דרישות נסתרות: קיימות דרישות פונקציונליות הדורשות פעולה שאינה בהכרח מתוארת בסיפור. פעולות אלו שאינן מתוארות אין התייחסות בטא. לדוגמה:

1. התייחסות למחיר בתוכנית רכישה. היות ויש דרישה שמחיר בתוכנית מסוימת יהיה זול יותר מאחרת מצד שני תעריף מקבע בחלק אחר.
2. קיימים נתונים שצריך לשמור במערכת ואינם מצוינים כלל בטקסט.
3. באיזו צורה מקבל מנהל הרשת דוחות.
4. כל הספקים רשומים מראש במערכת?
5. שיוך חיישן NFC לרכב

דרישות חיצוניות: קיימות דרישות פונקציונליות שהמערכת שלנו אינה עוסקת בהן אלא מערכת חיצונית.
לדוגמה:

1. ניהול משתמשי העובדים.
2. חיוב לקוח על קנייה.

דרישות פונקציונליות אופן פעילות פנימיות: קיימות דרישות שהמערכת מבצעת ולא נכללות בדיאגרמה מהגדרת Use Case.
לדוגמה:

1. מערכת אנליטית - איך מערכת מנתחת אירועים ודירוגם.
2. רישום ראשוני של מלאי סוגי הדלק השונים הזמינים למכירה בתחנת דלק מסוימת.

שאלה 3:

במודל UML נראה כיצד להתמודד עם מוגבלויות אלו.
כדי להתגבר על מוגבלויות אלו ננקוט בכמה גישות:

1. נשתמש בדיאגרמות מידול נוספות שבנויות יותר לפתור את מוגבלויות של UML.
כל שיטת מידול שמה דגש על נושאים של UML חלש בו.
דיאגרמת activity, sequence : במידה ונרצה להתרכז על קשרי גומלין/ פירוט פעילויות (סדר פעולות).
דיאגרמת class : במידה ונרצה להתרכז במבנה מערכתי (מבנה סטטי של מערכת).
2. נעבוד צמוד לסיפור המעשה כלומר אפילו שלשלים הבאים נסתמך בUML תמיד הבדיקה סופית מבוססת על פי דרישות בקטע.
דיאגרמת UML אומנם מציגה ויזואלית את התרחישים הפונקציונליים אבל הם אינם תחליף לסיפור מעשה היות ו UML לא מכסה את כל דרישות (לדוגמה דרישות לא פונקציונליות).

גישות אלו מתייחסות למערכת "MyFuel"

באופן הבא:

1. המערכת מפיקה דוחות לפי ניתוח של מערכת פנימית (מערכת אנליטית) כדי לפרט איך היא עושה את זה נעזר בדיאגרמת activity שבאה יהיה פירוט מסודר של כל השלבים הפונקציונליים הדרושים וסדר פעולותם.
2. המערכת מאפשרת תדלוק מהיר. בתדלוק המהיר קיימים מספר שלבים המבוצעים ע"י תת מערכות ואפילו שחקנים שונים.
דיאגרמת sequence תאפשר להציג את השלבים לפי סדר הפעולות הנדרשות וגם מאילו מחלקות של המשתתפים יש לגשת על מנת למשוך נתונים או פעולות מסוימות.
3. מצוין בסיפור כי תוכניות רכישה שונות משפיעות על מחירי מוצרים ("זול יותר") אין התייחסות מלאה בסיפור על תרחיש זה. לכן גם אם אינו מצוין בUML ונצטרך לתאר זאת במפורש כמובן נצטרך להתייחס לזה ולחזור לסיפור המעשה.