קבוצה מספר 1

לי חוגי - 311178552 – [leehugi93@gmail.com](mailto:leehugi93@gmail.com)

איתי דוד – 308021591 – [itaydavid22@gmail.com](mailto:itaydavid22@gmail.com)

רביב קומם - - [name@gmail.com](mailto:name@gmail.com)

ליאור קאופמן - - [name@gmail.com](mailto:name@gmail.com)

עידו קדוש - - [name@gmail.com](mailto:name@gmail.com)

תאריך כתיבת המטלה: 11.11.2019

**שאלה 1:**

תארו את אופן השימוש במודל Use Case בעבודתכם. הסבירו את מקומו ותרומתו של המודל לתהליך הפיתוח הכולל של המערכת בעזרת דוגמאות פרטניות (ספציפיות) מהמערכת "ICM" (לא Login ).

**תשובה:**

בעזרת מודל ה-use case ניתחנו את הסיפור בצורה מעמיקה, כלומר זיהוי השחקנים הפועלים במערכת ופעולותיהם השונות המתוארים כ- use cases. כלומר זיהנו את כל הפעולות שניתן לבצע במערכת ומי יבצע כל אחת מהפעולות.

בעזרת המודל אנו מבינים איך תראה המערכת בצורה טובה יותר, ומקבלים מבט על הכולל בתוכו את כל משתמשי המערכת, הפונקציות במערכת. כעת אנו כמתכנתים נדע לענות על הדרישות של המערכת ולממש את מגוון הפעולות הנדרשות בסיפור.

כאשר התחלנו לבנות את ה-use case, דבר ראשון עברנו על הדרישות בסיפור וניתחנו איזה דרישה היא פונקציונלית ואיזה דרישה לא פונקציונלית, לדוגמה כל שלב בטיפול הבקשה הוא דרישה פונקציונלית ולכן במודל כל שלב הוא use case. לאחר מכן ניתחנו את השחקנים המשתמשים במערכת ואת תת תפקידים שיש להם – למשל יש הרבה שחקנים שהם מ"מ ובטיפול הבקשה הם מקבלים תת תפקיד כמו בוחן או מעריך (כלומר ניתחנו כיצד השחקנים קשורים זה לזה באמצעות יחסי הורשה) .

לאחר מכן זיהנו את כל הפעולות שמתבצעות במערכת עם הבחנה בין פעולות שהשחקנים יכולים לבצע במערכת, לבין פעולות שמתבצעות על ידי המערכת. כאשר רק פעולות שמתבצעות על ידי שחקנים יהוו use case.

בעזרת המידול של ה-use case הצלחנו להבין טוב יותר את הפונקציונליות של המערכת ולתאר לנו מי יוכל להשתמש בכל פונקציה, לדוגמה בעזרת מידול של Use Case ניתן לראות בקלות כי בשלב בחינת הבקשה לשינוי משתתפים חברי וועדת השינויים, כאשר כל אחד יצטרך לבחון את הדו"ח ובסופו של התהליך לקבל החלטה אשר יכולה להיות חיובית, שלילית או לעיתים אף להגיע להחלטה שיש צורך בחומרים נוספים ומידע נוסף. כלומר הצלחנו לפשט תרחיש יחסית מורכב בעזרת הדיאגרמה ובכך אנו מבינים יותר את דרישות הלקוח כאשר נגיע לשלב המימוש בפרויקט זה.

**שאלה 2:**

תארו בפירוט איזה מרכיבים פונקציונליים ספציפיים של האופיין של מערכת "ICM" (כפי שמתואר במסמך ("Project Semester" לא הצלחתם לבטא בעזרת מודל ? UC מה הסיבה )או הסבר( לאי הצלחה זו?

**תשובה:**

כל שלב במודל הuse-case מבטא תהליך רחב הכולל פעולות שמבצע אחד השחקנים ומתקשר למערכת. לעיתים תהליכים אילו כוללים תתי תהליכים נלווים שמהווים חלק מהדרישות הפונקציונליות במערכת, אך לפעמים לא ניתן לבטא באמצעות המודל את כל הדרישות כuse-cases מכיוון שתהליך במודל כולל בתוכו מגוון של פעולות נלוות המבטאות את אותן דרישות פונקציונליות ונכללות בפעולה הנרחבת, חלק מתוך הדרישות הפונקציונליות שלא ביטאנו במודל הם לדוגמא :

- קיים במודל use-case עבור המעריך, שמבטא תהליך – "ביצוע הערכת משמעות לבקשה " ולפי הסיפור לאחרת ביצוע הערכת משמעות על אותו מעריך לבצע הפקת דו"ח , במודל עצמו כאשר ציינו את "ביצוע הערכת משמעות לבקשה" כ-use-case , use-case זה כולל בתוכו את כל תהליכי השלב הנלווים כמו הפקת הדו"ח ואותם אנו מבטאים בשלב ה description.

- קיים במודלuse-case עבור המפקח, תהליך שמבטא "צפייה בכל הבקשות " ולפי הסיפור לאחר שהמפקח צופה בבקשות הוא רשאי לבצע עדכון פרטים בבקשות שנמצאות בטיפול. במודל לא ביטאנו את "עדכון הפרטים בבקשות" כuse-case בפני עצמו מכיוון והכללנו אותו בתהליך של צפייה בבקשות, אך בשלב הdescription התייחסנו ופרטנו גם את תת התהליך הזה שהמפקח יכול לבצע ומהווה כדרישה פונקציונלית.

לסיכום , מודל הUse Case לא מאפשר לנו לתאר זרימה של פעילויות ותהליכים, בדוגמה שלנו של מערכת "ICM" ישנו את תהליך הגשת הבקשה. תהליך זה גדול וכולל שלבים ומשתתפים רבים, בפועל על פי דרישות המערכת התהליך עצמו קורה באופן רציף ולינארי שלב אחר שלב. עבור אירועים ששחקן אחד מבצע ולאחר מכן דרוש אישור משחקן אחר, לא הכללנו את זה במודל אבל כן תיארנו זאת ב-description, כיוון שזה לא תחת הגדרת include (כי לא אותו שחקן ביצע את האישור) ולכן את ההסבר לכך שצריך לבצע אישור ע"י שחקן אחר תיארנו ב-description. כלומר על מנת למדל את התהליך לדיאגרמת Use Case היה עלינו לפצל את כל הפעילויות בין השחקנים השונים ובכך איבדנו את האפשרות לראות את התהליך כולו ממבט על ובפרט לדעת בכל רגע נתון באיזה שלב נמצא התהליך.

נוסף על כך, במודל הUse Case מתארים רק פעולות של משתמשים במערכת, אבל בסיפור ישנן דרישות פונקציונליות לגבי פעולות שעל המערכת לבצע. לדוגמא על פי הסיפור על המערכת לבצע שאיבת והצגת מידע ממסד נתונים חיצוני בכל מיני פעולות שונות, אך לא ניתן להציג פעולה זאת על אף שהיא מרכזית וחשובה.

**שאלה 3:**

בתשובות 1 ו-2 תיארתם יתרונות ומגבלות שונות של מודל UC .הציעו דרכים להתגבר על המגבלות שציינתם, ונמקו למה הצעותיכם נותנות מענה למגבלות אלה. הסבירו את תרומתה של הגישה שאתם מציעים כאן לפתרון אותן המגבלות שתיארתם תוך התייחסות ישירה למערכת "ICM " ובסיוע דוגמאות פרטניות (ספציפיות) מהמערכת

**תשובה:**

כמו שציינו בסעיפים 1 ו-2 ישנם יתרונות וחסרונות במודל הUC לדוגמא שלא ניתן לבטא במודל את כל הדרישות הפונקציונליות של המערכת, אך כפי שהסברנו רוב הדרישות הפונקציונליות נכללות בשלבי תהליכים שמתבטאים כ use-cases ולכן הדרכים להתגבר על מגבלות אילו הן באמצעות שלבי הdescriptions, כאשר בשלב זה אנו מתארים use-case אנו כוללים בתוכו את תיאור כל השלבים המקדימים לאותו הuse-case ואת התוצרים הכוללים שלו שמהווים חלקים נרחבים מתוך הדרישות של המערכת, כמו בדוגמאות שהצגנו בשאלה 2 כאשר לא הכנסו את "הפקת הדו"ח" ואת "עדכון הפרטים בבקשות" לuse-case תיארנו אותם בשלבי הdescription של "ביצוע הערכת משמעות לבקשה" ו-"צפייה בכל הבקשות" וכך כאשר נבנה את המערכת נדע להתייחס לשלבים אילו כאל דרישות פונקציונליות שעבורן נממש פונקציות נלוות במערכת ICM.

כמו כן כפי שלמדנו בקורס "מבוא להנדסת תכנה", הדיאגרמה מציגה רק נדבך אחד בפרויקט ועם כל היתרונות שלה היא אינה מספקת בצורה מלאה. הפעולה המשלימה על מנת לתאר את כל האספקטים של המערכת היא שרטוט דיאגרמות UML נוספות כגון:

* דיאגרמת Activity שתאפשר לנו לראות זרימת תהליכים, אשר מהווה פתרון לחיסרון של דיאגרמת הUse Case שציינו בסעיף מס' 2.
* דיאגרמת Sequence שתאפשר לנו לראות את כל השלבים בתהליך מסוים וכיצד הם מתרחשים.
* דיאגרמת Class שתאפשר לנו לראות את כל הרכיבים השונים של המערכת ואת הקשרים ביניהם, החל מהשחקנים של המערכת ועד כל אובייקט שנשמר בתוכה.
* דיאגרמת State שתאפשר לנו לראות שינוי במצבים של אובייקט מסוים, לדוגמא בסיפורנו ניתן בצורה מאוד טובה להבין מה יהיה סטאטוס בקשת שינוי בכל רגע בתהליך.
* דיאגרמת OPM במספר רמות שתאפשר לנו "מבט על" על המערכת וכיצד כל הרכיבים עובדים ביחד ולאילו תהליכים הם קשורים.