שאלה מספר 2 מטלה 3 פרויקט ICM

במהלך הפרויקט עבדנו בצורה מקיפה ועקבית לפי התוכן הנלמד בקורס מבוא לבדיקות תוכנה, תוך שימוש בכל הכלים השונים והתמקדות בתהליכים של פיתוח תוכנה נכון, המלווה בבדיקות תוכנה מתאימות. דאגנו ליישם את כל עקרונות הפיתוח שנלמדו ולאחר כל שלב פיתוח דאגנו לבדוק את שלבי הפיתוח בהתאם לבדיקות השונות.

במהלך שלבי הפיתוח יישמנו מספר סוגי בדיקות תוכנה שהתאימו לשלבים השונים –

**Black Box Testing**

בדיקות Black box בדיקות אילו הן בדיקות התנהגות המבוססות על דרישות ופונקציונליות, שיטת בדיקה זו נעשית כאשר המבנה הפנימי לעיתים אינו ידוע לבודק, בדיקות אילו בדרך כלל בודקות את הפונקציונליות של המערכת, עוד בשלבים הראשונים של הפיתוח ביצענו מספר בדיקות מסוג זה כאשר מימשנו עבודה עם OCSF, לאחר שמימשנו תקשורת של שרת לקוח ושימוש במסד נתונים רצינו לבדוק תחילה שהתקשורת עובדת בצורה תקינה, כלומר בדקנו שלפי קלט ספציפי מצד הלקוח אכן מתעדכן מסד הנתונים בצד השרת ולאחר מכאן ציפינו לקבל פלטים ידועים מראש חזרה בצד הלקוח, תהליך זה היה הכרחי עבורנו בכדי לדעת שתעבורת התקשורת עובדת נכונה בכדי לבנות עליה את המשך תהליך הפיתוח.

עם התפתחות הפרויקט ופיתוח התוכנה חילקנו בין כל חברי הצוות מטלות ודפים למימוש, כל חבר צוות עבד על מספר דפים למשל (Committee page, Tester Page , Execution page), כאשר כל חבר צוות סיים לממש את הדפים שלו דאגנו כי כל חבר צוות יבצע White Box Testing.

**White Box Testing**

בדיקה זו מבוססת על הידע וההיגיון הפנימי של קוד היישום, ידוע גם כ " קופסת זכוכית ", בבדיקות מהסוג הזה יש להכיר את הקוד הפנימי ובדיקות אילו מתבצעות לרוב על ידי המפתח. כאשר כל חבר צוות סיים את מימוש הדפים שהוקצו לו , כל חבר צוות דאג לבצע בדיקות מהסוג הזה על הדפים שלו בכדי לבדוק את הנתיבים השונים, לדוגמא בדף Execution page כאשר המבצע צריך להצהיר על תאריך סיום צפוי לביצוע הבקשה יש מספר בדיקות שצריך לבצע כמו הסיטואציה בה הוא מזין תאריך שכבר עבר, או מזין תאריך ומחכה כעת לתשובה מן המפקח האם התאריך אושר או לא, כמובן כל זה על ידי מימוש שאילתות נכונות הבודקות את שלב הבקשה הנוכחי שמתעדכן בכל סיום שלב.

בהמשך שלב הפיתוח ולאחר שכל חברי הצוות ביצעו את המטלות השונות, תוך סיום מימוש הדפים השונים, בכדי לשלב את כל מרכיבי התוכנה ולבדוק האם המערכת עובדת בצורה תקינה כמערכת שלמה ביצענו Integration Testing.

**Integration Testing**

בדיקה זו מתמקדת בבדיקת מודולים משולבים, כלומר לאמת פונקציונליות משולבת לאחר אינטגרציה מודולי קוד ויישומים נפרדים, בדיקות אילו רלוונטיות במיוחד ללקוחות ושרתים ומערכות מבוזרות. לקראת סוף שלב הפיתוח ובכדי לבדוק כי המערכת עובדת נכונה בשלמותה ביצענו המון בדיקות מהסוג הזה על מנת לוודא כי חיבור הדפים ושילוב תנאים מדפים שונים עובדים בצורה נכונה, כלומר קלט מתוך מימוש של דף מסוים היווה פלט לתוך מימוש של דף אחר. לדוגמה כאשר כל תפקיד בביצוע הבקשה מבקש הארכת זמן , יש לעדכן את מסד הנתונים בצורה נכונה ולעבור לשלב של המפקח לאישור הזמן, בנוסף כאשר מסתיימים שלבים שונים בתהליך טיפול בבקשה יש לבדוק שהמעבר בין התפקידים השונים מתבצע נכונה תוך עדכון מסד הנתונים בשלב הנוכחי שבו הבקשה נמצאת ( דוגמה – כאשר מבצע הטיפול בבקשה מסיים את עבודתו כעת יש לעדכן את מסד הנתונים בכדי שבודק הבקשה יוכל לראות את הבקשה הספציפית בכדי להתחיל בשלב הבדיקה.)

**Usability testing**

בדיקות אילו הן בדיקות הבודקות ידידותיות למשתמש, נבדק בדרך כלל זרימת היישום, נשאלת השאלה – האם משתמש חדש יכול להבין את היישום ולתת מענה ועזרה נכונה בכל פעם שהמשתמש "תקוע" בכל נקודה בשימוש התוכנה. בדיקות מהסוג הזה ביצענו במהלך כל שלב הפיתוח, בסיום הדפים השונים וכמובן בסוף שלב הפיתוח כאשר המערכת שלמה, דאגנו לתת משוב וביקורת בונה עבור הדפים השונים של חברי הצוות, כלומר כל חבר צוות בדק דפים שונים שחבר צוות אחר מימש על מנת לבדוק שהGUI תקין ונוח לשימוש, השתדלנו לחשוב כמו אנשים חיצוניים שנתקלים לראשונה בתוכנה ולראות האם התוכנה ברורה, האם כל עמוד של בעל תפקיד בטיפול בבקשה ברור והאם מובן מה לעשות בדפים בכל נקודה, דאגנו לממש הסברים קצרים על כל דף כלומר על מה יש ללחוץ בכל חלק ובכל שלב בכדי לטפל בבקשה בצורה תקינה ותואמת לדרישות הלקוח.