האקטרון 2 קבוצה 9

דניאל רבינוביץ, דולב דדון, דניס דומברובסקי,איתי שטרית

<https://github.com/ddenis1994/team9-Hucktron-2>

הדרישות העסקיות שכוסו באקטרון זה :

הפקדת דוחות על בסייס התוצרת

דירוג המוצר למטרת תמכור

הפקת דוח עפ חקלאי בעל תבואה טובה יותר

חיסכון בכוח באדם בשל דירוג הפרי

בוצעו בדיקות יחידה לפונקציות הבאות :

1. SizeInPixsel
2. Diameter
3. Chackpassword

1.

הבטחת איכות תוכנה- הוא מכלול הפעולות הנדרשות להבטיח את איכותה של תוכנת מחשב כחלק מתהליכי הפיתוח והתחזוקה שלה.זוהי בעצם מציאת דרך מתוכננת ושיטתית להערכת איכות המוצר, בכדי להקנות ביטחון כי המוצר עומד בדרישות האיכות ויענו על מלוא צורכי המשתמש ,ז"א זה הוא שם כולל לכל הפעולות הננקטות בכדי ליצר מוצר ברמת האיכות הנדרשת.

הבטחת איכות התוכנה מכילה שני עקרונות:

א. המוצר חייב להתאים לייעודו המתוכנן.

ב. משגי הפיתוח והיצור חייבים להימנע (או לפחות להצטמצם(.

איכות המוצר או השירות נקבעת ע”י המשתמש או הלקוח, ולא ע”י הקהל הרחב!

תהליכים להבטחת איכות תוכנה :תהליך היצור, שלב היזום, כתיבת הקוד, עיצוב, פיתוח, ותחזוקה של המוצר הסופי,הגדרת דרישות, בדיקות אבטחת תוכנה, בדיקות מול דרישות ,זיהוי פגמים, עמידה בסטנדרטים כללים, עמידה בדרישות שלא הוגדרו אך הם בסטנדרט המקובל.

תהליכיים בפרויקט שלנו:

שלב היזום-השלב בו עולה הרעיון לפיתוח התכונה ,בוחנים את הבעיה ואת הצורף בפיתוח הפרויקט.

בשלב הייזום קבענו מי הוא הלקוח שלנו ?מה הוא צריך? ומה הדרישה שלו ?ומה היא המטרה של התוכנה בסופו של דבר.

זיהוי פגמים-בכדי לבדוק שהתוכנה עובדת בצורה הטובה ביותר ,ביצענו בדיקות תוכנה (unit -test),צילמנו תמונות של הפירות תפוח בננה ואפרסק ובדקנו האם התוכנה בודקת אותם בצורה נכונה ומביאה תוצאות כפי שנדרש.

2.

מדדים הם כלי ניהול ממדרגה ראשונה לשם השגת מטרות ולשם שיפור מתמיד.

* **מדדי תפוקה בפיתוח תוכנה-**המדד הנפוץ, ואולי הטבעי ביותר לתפוקה הוא מספר אלפי שורות תוכנה KLOC – K Lines of Code)) או מדידה חליפית של נפח התוכנה שפותחה.

**מטרת המדד היא שכתיבת הקוד תהיה חסכונית ,מה שיכול לגרום גם לחיסכון בזמן ריצה**

* **מדדי איכות הפיתוח תוכנה-**המדד הנפוץ הוא צפיפות הפגמים, הנמדדת במספר הליקויים המתגלים באלף שורות תוכנה (N / KLOC), כאשר N הוא מספר הליקויים שהתגלו.

**מטרת מדד איכות פיתוח התוכנה היא לגלות כמה שיותר ליקויים של התוכנה.**

**המאמר שבו השתמשנו :**

**http://www.leadersnet.co.il/%D7%94-sow-%D7%A9%D7%9C-%D7%94%D7%A4%D7%A8%D7%95%D7%99%D7%A7%D7%98-3/**

3.

**מערכות סוציו טכניות-**מערכות שכוללות מערכות טכניות אבל גם מעבדים ביצועיים ואנשים שמשתמשים ובעלי אינטראקציה עם המערכת הטכנית מערכות כאלו נשלטות ע"י החלטות ארגוניות לדוגמא :אינסטגרם.

**מערכות מבוססות מחשב -** מערכות הכוללות חומרה ותוכנה אך המפעילים ותהליכים תפעוליים בדר"כ לא נחשבים חלק מהמערכת למשל WORD .

ההבדל ביניהם בעצם הוא שבמערכות סוציו טכניות לאדם יש השפעה על התוכנה ,ואילו במערכת מבוססות מחשב לא .

4. "החורים" בגבינה, על פי Reason, מדבר בעצם על בעיות שיכולות להיווצר עקב טעויות אנוש, להלן בעיות שיכולות להיווצר בתוכנה שלנו עקב טעויות של העובד:

א. בחירת סוג פרי לא נכון לדוגמא: בחירת תפוח בזמן שבודקים אפרסק

ב. במקרה ונבחר קובץ שאינו פורמט של תמונה.(ז"א במידה ולא נצלם את הפרי אלא נבחר מתוך מאגר תמונה)

ג. טעות בבחירת הספק והסקת מסקנות שגויות לגבי התבואה של החקלאי הנבדק.

5. מערכת דטרמיניסטית היא מערכת שעבור רצף של קלטיים היא תחזיר רצף קבוע של פלטים.

דוגמא תכונה לא דטרמיניסטית המערכת שלנו היא: שלפני ביצוע הבדיקה האדם בוחר בעצמו על איזה סוג פרי מתבצעת הבדיקה ז"א:תפוח/בננה/אפרסק. (הבחירה של הפרי זה בעצם הקלט).

אופיים הבלתי צפוי של אנשים) הגורם האנושי (וכמו כן ,שינויי מערכת אשר אינם צפויים עקב שינויים תכופים בחומרה ,תוכנה ובמידע.

דוגמא כללית : מערכת סוציו טכנית מערבת גם מגע אדם בה וזה הופך אותה ללא דטרמיניסטית. מערכת סוציו-טכנית לא תמיד תפיק את אותה התוצאה.