

## הרגשת בטן

לפני מספר חודשים יצא לי לפגוש חבר ותיק שהוא גם חבר למקצוע בחתונה של מכר משותף. אני יודע שזה לא הזמן ולא המקום, אבל מפה לשם התחלנו לדבר על ענייני המקצוע, והוא חלק איתי את מצבו העגום:

"מחר יש פגישת שחרור לגרסה האחרונה של המוצר שלנו ואני עומד להמליץ שלא לשחרר את הגרסה. אבל יתעלמו ממני".

"ולמה ההמלצה שלא לשחרר?"

"כי יש לי הרגשת בטן ברורה שהגרסה האחרונה שקיבלנו - זו שרוצים לשחרר ללקוחות – דפוקה לגמרי. אבל מנהלי המוצר עובדים לפי רשימה של קריטריונים ליציאה (exit criteria), המוצר עומד בקריטריונים והרגשת בטן זה לא data".

שאלתי מאיפה תחושת הבטן מגיעה, ואז קיבלתי את כל הסיפור:

לפני שבועיים הקבוצה שלו קיבלה Build לסבב בדיקות אחרון לפני שחרור רשמי של הגרסה. תיאורטית הכל היה אמור להיות חלק ונקי, אבל תיאוריה לחוד ומעשים לחוד: הבודקים גילו כבר ביום הראשון שני באגים קריטיים.

צוות הפיתוח מיהר לתקן את הבאגים, ושחרר גרסה חדשה.

מכאן והלאה, במשך שבועיים, הדבר חזר על עצמו: צוות הבדיקות מצא באג או שניים קריטיים, וצוות הפיתוח שחרר גרסה חדשה.

והנה הבעייה שעמדה לפני בן שיחי:

"על פי קריטריוני היציאה שלנו, אין לשחרר מוצר שיש בו באגים קריטיים. אבל צוות הפיתוח מיהר לתקן כל באג שמצאנו, ולכן אין אף באג קריטי בקוד, נכון לגרסה ששחררה היום. בשבועיים האחרונים קיבלתי 8 גרסאות שונות. נראה לך שהספקנו לבדוק את המוצר כמו שצריך? בהסתמך על איכות הגרסאות בשבועיים האחרונים ברור לי שגם בגרסה של היום מתחבאים לא מעט באגים קריטיים נוספים ואם יתנו לי עוד כמה ימי בדיקות אמצא אותם. אבל תאריך השחרור הוא מחר, ועל הנייר הכל תקין, גם אם הבטן שלי אומרת אחרת".

ובאמת – מה עושים?

במצב אידיאלי, הדרך הנכונה היא להיות פרואקטיביים ולהכניס לרשימת הקריטריונים ליציאה עוד כמה תנאים מעבר לרק ספירה פשוטית של מספר הבאגים בגרסה ורמת החומרה שלהם. חייבים לבחון מגמות (trends): לבדוק מה קצב הבאגים החדשים שמדווחים ומה קצב סגירת הבאגים. מה הצפי לעוד באגים, בהסתמך על מספר הבדיקות שעוד נשאר להריץ. הנה דוגמאות אפשריות לקריטריונים אלה:

- מספר הבאגים הקריטיים שנפתחו בסבב הבדיקות האחרון קטן מ-X
- כמות הבאגים שנמצאו בשבועות האחרונים הולכת ויורדת כל שבוע בקצב Y
- לפחות X ימי בדיקות על הגרסה האחרונה ללא באגים קריטיים וללא יותר מ-Y באגים בחומרה "גבוהה".

וכו'.

חיפשתי קצת אם יש ברשת דוגמאות לרשימות של קריטריוני יציאה. לא ממש מצאתי. הנה אחד שחשבתי שהוא רלוונטי:

[http://www.pmtraining.com.tw/article/5phase/Phase3-Approval/Completion%20Criteria,Checklists/sw\\_criteria.doc](http://www.pmtraining.com.tw/article/5phase/Phase3-Approval/Completion%20Criteria,Checklists/sw_criteria.doc)

כיוון שכך, החלטתי לייצר בעצמי דוגמה... את הוורסיה שלי אפשר להוריד מהקישור הבא:

<http://bit.ly/QRCTables>

הוספת קריטריונים כאלה מראש היתה מציבה שלט "עצור" ברור במקרה שלנו, ובן שיחי היה יכול להנות מהחתונה.

אבל מה לעשות במקרה שמתואר כאן, שבו לא נקבעו מדדים כאלה מראש?

הצעתי לחבר "לתרגם" את הרגשת הבטן שלו לנתונים שהביאו להרגשה זו ולהציג אותם בצורה חדה וברורה: יותר מידי באגים קריטיים נמצאו בקוד ולא היה מספיק זמן לבדוק את הגרסה. אלו **עובדות**, לא תחושות.

המסר הוא: "זה המצב. על פי נתונים אלה אני מסיק שהגרסה לא באיכות מתאימה לשחרור". לא הרגשה, לא אינטואיציה, לא תחושה – אלא דעה מקצועית המבוססת על עובדות, הבנה בתהליכי פיתוח תוכנה, ונסיון מצטבר.

התקשרתי אחרי כמה שבועות לשמוע איך הלך. לא רע, מסתבר. חברי הציג שלושה גרפים פשוטים בישיבה:

- כמות הבאגים הקריטיים שנמצאו כל יום (1-2 כל יום, ללא מגמת שיפור)
- הזמן שהיה לבודקים לכל גרסה בשבועיים האחרונים (יום-יומיים עד שקיבלו גרסה מתוקנת)
- אחוז הבדיקות שהגרסאות האחרונות עברו (בין 20% ל-30% מהבדיקות)

זה הספיק לגרום לכולם להרגשת בטן לא טובה. תאריך השחרור נדחה.