

その開発、そのままテストまで 突き進んでいいんですか？

大西 建児

組み込みシステムに限らずソフトウェア・テストが難しい大きな理由は、テストで評価・検証する根拠となる「ユーザの真の要求」がとても分かりづらいからです。「テスト対象が何のために必要なのか」という目的(テスト対象の存在理由といってもよいだろう)がつかめていないと、テストをどこまでやればいいのか、その終了基準は見えてきません。たとえ「何のため」かが分かったとしても、じゃあ「どのような」ものであるべきかが分からなければ、やはり客観的なテストは行えません。

● テストの相手は仮想現実

特にソフトウェアの評価を難しくしているのは、ソフトウェアの要求や、その要求に基づいて定められる仕様が「仮想現実(まだ存在しないもの、仮に想定しているもの)」をベースに構築されるからです。仮想ですから、想定条件がはっきりしないと、そこから導かれる結果の妥当性は確保できません。しかし、「私の常識、あなたの非常識」などというように、第三者間で想定条件を明確にすることは簡単ではありません。世の中にソフトウェアという目に見えない製品が出てきてからというもの、このことは常に課題であり続けています。

この課題を克服し、仮想現実をできるだけ正確に表現するために、UMLをはじめとしたモデルを使った表記法が進化してきました。また、モデルによって抽象化した要求と実装ができるだけかい離しないように、MDD(モデル駆動開発)のような、開発手法を支援する環境も進化しています。しかしまだまだ、組み込みシステムの使い手(発注者)と作り手の思惑がすなり一致するとは限らないのが現実です。だからこそ、今だにソフトウェア・テストは一筋縄ではいかないのです。

● 筆者と妻とロール・ケーキ

あいまいな要求を満たすことの難しさを、日常生活の例から見てみましょう。実例の方が面白いでしょうから、ここでは我が家のエピソードを披露します。

とある朝のことです。朝食を食べながら妻が言いました。

「このところ、何もおいしいもの買って帰ってくれないわね～」

これは「何かおいしいものを買ってきなさい」との暗黙の要求といえますか、指令であることは、奥さまを持つ世の旦那さま方はよくお分かりでしょう。

「何か」と言っても、決して何でもよいわけではありませんよね。当然、妻が気に入るものでなければ、舌の肥えた彼女を満足させることなどできません。しかも、普段なら気に入るようなものでも、最近食べたものだったりストックがあるものだと、

満足感はかなり落ちてしまいます。

「え～、それあるのに～」(ちょっと逆切れ気味?!)

と、せっかく買ってもトホホな結果になってしまうことでしょう。

このような制約条件を考慮に入れた上であたりを付けたのが、妻の好物のロール・ケーキでした。そこで、その日は先ごろ改装された東京駅の地下街でロール・ケーキを買って帰ることにしました(図1)。

地下街をそぞろ歩くと、ロール・ケーキは3カ所で売っているようです。それぞれのお店は次のような感じでした。

- 1) 十数人の行列ができているお店。ただし行列が邪魔して、どんなロール・ケーキかは不明。
- 2) それなりに人はいる(行列とまではいかない)。プレーンなロール・ケーキとフルーツ入りのロール・ケーキを売っている。
- 3) 普通に人がいるが、2)の店よりは少ない。マロン・クリーム入りのロール・ケーキを売っている。

私は並ぶことが嫌いなため、1)は選択肢から落ちました。妻の好きなタイプのロール・ケーキだと分かっていたら並んだかもしれませんが、どんなタイプか分からないまま並んだ揚げ句、無駄足となるのが嫌でしたし。



図1 ロール・ケーキでも買って帰るか…



次に**2)**と**3)**ですが、この選択では妻と息子のどちらの機嫌を取るかがポイントになりました。

息子は栗好きです。ケーキだとモンブランが好物です。妻は栗好きですが、モンブラン好きではありません。「ぼそぼそ」しているからだそうですが、栗きんとんは好きだというからよく分かりません。ともあれ、今回は妻の機嫌を取ることが目的ですから、**2)**のお店としました。

次にプレーンかフルーツ入りの選択ですが、常日ごろの妻いわく「私はシンプルなのが好きな」ということなので、プレーンにしました。ただ、少々気になったのがケーキの表面に粒砂糖がまぶしてあったことです。表面にナッツなどがまぶしてあるものを妻は好まないのですが、ナッツじゃないから多分大丈夫だろうと思い帰宅しました。

これで妻のご機嫌取りができと思ったわけですが、ここまでやっても「TCS (Total Customer Satisfaction)」、もとい「完全なる満足」を得るに至りませんでした。妻のコメントは次のようなものでした。

「うーん、プレーンでまあまあおいしいけど、生クリームよりはカスタード・クリームが好き(バニラ・ビーンズ入りの本格派じゃないとNG)。それから表面が粒砂糖でぼそぼそしているのが気に入らない。あ、そうそう、カスタードならフルーツ入りでもよかったのに」

それでも満更ではなさそうだったのは、どうやら**2)**のお店がイタリアンの有名店だったからのようです。満更ではないようなので、とりあえず、当初の目的であった妻のご機嫌取り程度は果たせたようです。

このように10年以上連れ添った妻の要求ですら、正確に把握して満足させることは難しいものです。

また、「満足させる」といっても、そのレベルは前提条件によって異なります。この例では「ご機嫌取り」が目的でしたから、「まあまあおいしい」という言葉が妻の口から出ればOKでした。しかし、もし夫婦喧嘩が何かで私が謝るためならば、まあまあではだめで、妻の要求をほぼ完全に満たさなければならなかったでしょう。妻の誕生日だったなら、ロール・ケーキという選択そのものがきっとNGだったことでしょう。

身内ですら、このように要求を満たすことは難しいのですから、組み込みシステムの開発で赤の他人の要求を満たすことが簡単でないことは当たり前の話です。

特に今回の例題では、その目的を正確に把握しているかどうか要求を満たすための鍵となるのですが、どうも筆者にはうまくいっているようには見えません。

● 今回の事例の落とし穴

今回の特集は、開発の手順やそのアプローチ方法を解説することが主目的ですから、記事の内容はその目的に適ったものになっています。ですから、以下は記事そのものの信頼性を指摘するものではなく、違う角度からこの事例をとらえた場合に見えてくるものとしてお読みください。

例えば、本特集の目的が「“お客様”のニーズを満たすためのテスト計画のアプローチを知ってもらうこと」だったとしましょう。その場合なら、現実世界の落とし穴にこの事例がはまってしまっているため、まずそのことに気付かなければならないという趣旨の解説がされたはずですが。

どういうことか説明しましょう。冒頭で触れたように、テストは「テスト対象が何のために必要なか」が明らかでないと、終了基準が見えせんし、妥当な評価を行えません。言い換えると、テスト対象そのものがユーザの要求に沿って作られていないならば、どんなにテストを実施したとしても、開発者自身の満足にはつながれど、決して“お客様”の満足にはつながらないということです(図2)。

では、この記事の開発アプローチのどこに問題があったかを見ていきましょう。まず開発の始まりからして良くありません。顧客Kさんの話を営業Aさんが、技術者のGさんに丸投げしてしまっています。

多分、Gさんは技術者としては優秀な人なのでしょうが、プロダクト・マネージャとしてならその役割を果たせていません。なぜなら、Gさんは顧客Kさんの真のニーズを抽象度の高い段階でつかむことをしないまま、どんどん具象的な仕様の詰めを行ってしまっているからです。

先行事例のある開発ならば、顧客のニーズと開発者の認識は、仮想現実レベルでも実際の仕上がりである製品レベルでも大きく変わることはないでしょうから、この進め方で問題にはならないかもしれません。ですが今回の記事は「プロトタイプ」としての開発ですから、どこまで顧客のニーズを実現できれば満足していただけるのか、しっかりと要求を確認しなくてはならな



図2 このような「隠れた要求(条件)」がある可能性も...