

# 経費削減と品質向上を目的とした回帰テストの実施

【ケーススタディ・某ソフトウェア開発会社様：ソフトウェア Version UPに伴う回帰テストの実施・テストパターンの標準化】

設計からテストまでのすべての業務に携わっているケースが多く見受けられます。

しかしながら、近年、単価の高い開発者の経費削減が謳われる中、開発者が本業に集中出来るように、業務分担の改善が必要になってきております。某ソフトウェア開発会社様は年に数回の Revision UP、Version UP を行います。

ソフトウェアの品質と信頼性を保証する上で、回帰テストを正確に実施することは重要なステップの1つです。しかしながら、テスト業務に時間を要し、他の業務に時間を割けられない現状がありました。そこで弊社で、回帰テスト、並びにテストの標準化を実施いたしました。

## 目的

1. テスト業務を単価の高い開発者から、単価の安い外部要員に委託することで経費を削減する
2. テスト業務を外要因に委託することで、開発者を本業に専念させる
3. テスト項目表を標準化することにより、検証精度を向上及び均質化させる

## 作業概要

### (1)回帰テスト項目表の追加及び標準化

- ・ Revision Up、Version Up に伴うテスト項目の追加、並びに項目表を文書化して修正

### (2)回帰テストの実施

- ・ 単体テスト、結合テスト終了後の回帰テストの実施
- ・ 欠陥(バグ)発見時の早期連絡

### (3)回帰テスト検証結果報告書の作成

## 作業効果

### (1)経費削減

- ・ テスト業務を単価の高い開発者から、単価の安い外部要員に委託することで、今までの 1/3 経費を削減

### (2)属人化から標準化することにより、テストの効率と精度が向上

- ・ テスト項目表を文書化することにより、テストの精度が個人に依存せず、一定レベルの品質を保持

### (3)プロダクトの品質向上

- ・ 弊社(第 3 者)がテストを実施することで既存欠陥(バグ)を発見し、プロダクト品質が向上

## 備考

(注 1) 回帰テスト

プログラムを変更した際に、その変更によって予想外の影響が現れていないかどうか確認するテスト。

(注 2) Version UP

メーカーが従来のソフトウェア、ハードウェアに新機能の追加や既存機能の改良といった内容の変更を施して新版を作り、それを発表、販売すること

(注 3) Revision UP

細かな部分だけの改訂で、製品名などの変更も伴わないこと

### 修正前テスト項目表

24		「次へ」
25		「キャンセル」



### 修正後テスト項目表

修正前は、項目数が少ないが、修正後は項目数を増やし、単語を文章化することにより、誰がテストを実施しても分かりやすいように修正。

31	42	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■設定」画面へ遷移した時、「◆◆◆名」(G) が空白になっていること
32	43	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■グループ設定」画面へ遷移した時、「出力先ベースフォルダ」(H)のバグが「C:\Program Files\◆◆◆」になっていること
33	44	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■設定」画面へ遷移した時、「◆◆◆」(G) がグレースアウトしていること
34	45	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■設定」画面へ遷移した時、「◆◆◆」(T) がグレースアウトしていること
35	46	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■設定」画面へ遷移した時、「◆◆◆」(O) がグレースアウトしていること
36	47	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■設定」画面へ遷移した時、「◆◆◆」(O) がグレースアウトしていること
37	48	「○○○画面」- ロゴを新規作成する - ロゴ ON、「次へ」00 を押下、「■■■■設定」画面へ遷移した時、「◆◆◆」ボタンがグレースアウトしていること

## 弊社利用による効果

### (1)テスト画面の納品

- ・ テスト結果を、全て画面で納品することにより、修正が発生した場合、問題の動きを確認出来、迅速な対応が可能。

### (2)テスト項目表の標準化

- ・ 既存のテスト検証表を精査し、修正、追加を実施。誰がテストを実施しても同じテスト品質を保持、テストの効率と精度が向上。テスト担当者の変更に伴う引き継ぎ工数が大幅に削減。

### (3)複数名でのテスト検証

- ・ 膨大なテスト検証をダブルチェックで実施し、単独では見落としてしまうエラーの確認漏れを回避。