

テスト設計書を書こう！

- 抜け漏れの少ないテスト設計書テンプレートの提案 -

NEC SWQCテスト技術コミュニティ

杉田 正実(日本電気通信システム株式会社)
吉澤 千佳子(日本電気株式会社)

SWQCテスト技術コミュニティの紹介(1)

NECグループのSWQC活動として、テーマ別コミュニティ活動中！
テスト技術コミュニティは、特にソフトウェアテストに興味を持ったメンバーで構成されています。

テストに悩む技術者 集まれ！

- こんな方を募集します
 - テストは人海戦術だと信じている
 - そう信じる上司に困っている
 - いつもテスト終盤に出続けるバグに悩む
 - どこまでやったらテストが終わったと言えるのかわからない
 - テストに技術があるなんて聞いたことがない

狙い 人海戦術テストからテストエンジニアリングへ

- こんな活動をします～
 - ★ テストを根本から変える
 - 実践的なテスト技術の勉強
 - 現場のテストの改革
 - ★ 「テストのプロ」作り
 - 技術には専門家が重要、コミュニティのメンバーは、引っ張りだこのテスト専門家...を目指したい
 - ★ テストの疑問は、ここにきたら解ける！
 - テストの悩みなんでも相談室として、社内のテスト技術のハブの役目を果たす

SWQC活動:NECにおけるソフトウェアの総合的品質管理活動(1981年に開始)
現在は組織を超えてSW開発技術者が切磋琢磨する場として活動中

SWQCテスト技術コミュニティの紹介(2)

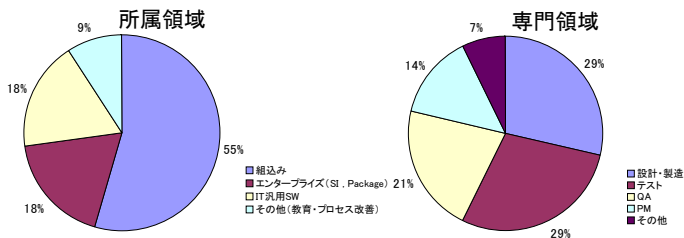
活動状況とメンバ構成

・活動状況

- 月1～2回の交流会(FtoF)、ML・Wikiも活用

・メンバー構成

- 組込み、エンタープライズ、教育など、様々な領域出身者で構成
- 専門領域はテストだけでなく設計、QAなど多岐に渡る



問題点

テスト設計書を導入したが、効果小

従来はテストプロセス未定義⇒今回、テスト設計を導入しプロセス定義



テスト設計書<IEEE829-1998より>
・章立てと記述内容を定義

・テストで摘出すべきバグが流出

- ・章立てだけでは、何を書くのかわからない
- ・技術者のスキル差異により、テスト設計もれが発生

- ・テスト設計⇒テスト項目のリンク確認が困難
(項目数が多すぎて、テスト設計内容が正しくテスト項目へ展開されているか確認できない)

<テストプロセスと発生した問題点>

* あるコミュニティメンバーの事例に基づく

課題

<原因分析結果>

- ◆ 境界値・既定値・Nullなど当たり前のテスト設計もれ
- ◆ テスト設計⇒テスト項目へ展開時のテスト項目作成もれ
- ◆ 「テスト設計書」の章立て規定では、テスト設計内容の統一困難



- ◆ 結果として、出荷後に欠陥が流出

<課題>

- ◎ 当たり前レベルのテスト設計もれ防止
- ◎ テスト設計⇒テスト項目へのリンク付け
- ◎ 人によるテスト設計内容のばらつき軽減



解決策

<課題>

- ◎ 当たり前レベルのテスト設計もれ防止
- ◎ テスト設計⇒テスト項目へのリンク付け
- ◎ 人によるテスト設計内容のばらつき軽減



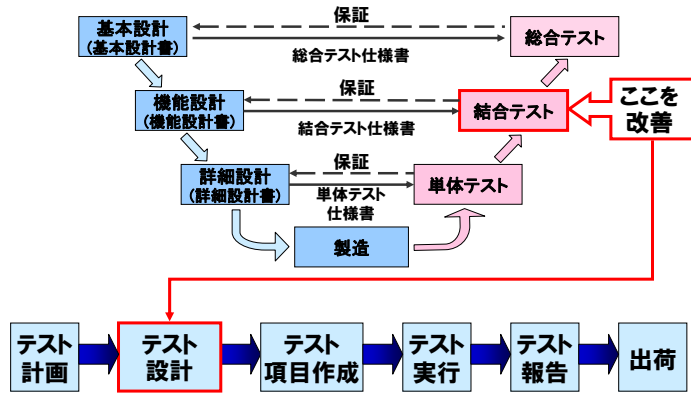
<施策>

テスト設計書へ「テスト設計テンプレート」を導入

1. テスト観点を網羅した「テスト設計テンプレート」を導入
 - ・ 機能・非機能テストの2つに分け、テスト観点をリストアップ
2. 「テスト設計テンプレート」に沿ってテスト設計を実施
 - ・ テスト設計者は、テンプレート記載のテスト観点に沿って実施すべきかを検討し、実施する場合はその詳細を設計
 - ・ 上記をテスト観点毎にIDを付け、テスト項目へリンク付け

改善対象範囲

■結合テストのテスト設計を改善対象として選定

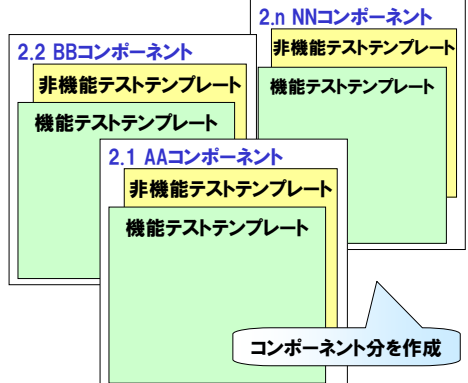


施策1:テスト設計書の構成

XXシステム 結合テスト設計書

※目次※

1. はじめに
 - 1.1. ドキュメント識別子
 - 1.2. スcope
 - 1.3. リファレンス
 - 1.4. テスト成果物
2. 結合テスト設計の詳細
 - 2.1. AAコンポーネント
 - 2.2. BBコンポーネント
 - ...
 - 2.n NNコンポーネント..
3. 一般
 - 3.1. 用語集
 - 3.2. 変更手順及び履歴



コンポーネントごとに、2種類テンプレートを書く

機能テストと非機能テスト

機能テスト:機能要件の確認

コンポーネントやシステムの機能仕様を分析して実施するテスト

非機能テスト:非機能要件の確認

コンポーネントやシステムで、機能に関係しない特性。例えば、信頼性、効率性、有用性、保守性、移植性のテスト

JSTQB ソフトウェアテスト標準用語集(日本語版)より
http://www.jstqb.jp/dl/JSTQB-glossary.V1.3.pdf

施策2-1:「テスト設計テンプレート」機能テスト観点の抽出

機能テスト観点を文献・書籍から調査

抜け漏れを防ぐ
観点の整理と再分類

メンバの経験より、結合テストに必要なテスト観点を抽出

- ◇バグを作り込みやすい設計箇所は?
- ◇結合テストで検出したいバグは?
- ◇ドメイン固有バグは、一般化して議論

基本はコンポーネント連携
加えて、結合テスト以降でテストが困難な観点を選択

- ・機能
- ・I/F
- ・エラー処理
- ・境界値
- ・計算ミス・誤差
- ・競合・タイミング

理解を促すために
観点の説明を記述

施策2-2:「テスト設計テンプレート」非機能テスト観点の抽出

非機能テスト観点を文献・書籍から調査

抜け漏れを防ぐ
観点の整理と再分類

列挙されたテスト観点を、議論のうえ分類し、グループ化を行った

- ◇ドメイン毎に非機能観点をリストアップ
- ◇文献・書籍から観点をリストアップ
- ◇類似した観点を吟味し、分類・整理

わかりやすく具体的なテストをイメージできる観点を選択

- ・操作性テスト
- ・運用テスト
- ・性能テスト
- ・信頼性テスト
- ・セキュリティテスト
- ・障害対応テスト
- ・インストールラビリティテスト
- ・移植性テスト
- ・保守性テスト
- ・機器構成テスト

理解を促すために
観点の説明を記述

テスト観点抽出ステップ(操作性の例)

JSTQB, SQuBOK, SWEBOK, Davisその他の書籍、論文、社内標準に出現するテストの種類、観点、目的など:約100項目

操作性観点の抽出

使用性テスト.....	5
アクセシビリティテスト....	2
ユーザビリティテスト.....	2
操作性	3
ユーザインタフェーステスト	1
ルックアンドフィール	1

(数字は出現数)

議論 & 知見の導入



操作性	ソフトウェア製品が、指定された条件下で理解され、学びやすく、使用しやすく、ユーザにとって魅力的である能力を判定するためのテスト
操作のしやすさ	ユーザ視点で、使い勝手に問題がないか、覚えやすいかどうかを検証するテスト
操作の一貫性	ソフトウェアの操作(メニュー、キーボード入力、他)に一貫性があるかを検証するテスト。国際標準や業界標準に準拠しているか、バージョンアップ時に操作性が変わらないか、などについても考慮が必要
アクセシビリティテスト	ソフトウェアを使用することが想定される身体的な制約を持つ人を含む全てのユーザ(子供や高齢者、障がい者、など)が、どの程度容易にコンポーネントやシステムを利用できるか判定するテスト

テンプレートの補強

課題として抽出された項目の必須事項を、テンプレートへ

実施の有無		ID	備考
○/×	詳細な観点		

IEEE829
による補強

テストの概要

入力条件	環境	特別な要求	依存関係
テストの実行に必要な入力(値、テーブル名、ファイル名など)を指定	テストを遂行する上で必要な環境(HW、SWなど)を指定	前提条件、事後条件、その他特別な手続き等	テストケースとテストの実行順序

施策2-3:「テスト設計テンプレート」のIEEE829による補強

10.2.2 (LTD セクション 2.2) アプローチの詳細化<抜粋>

Summarize the common attributes of any test cases. This may include input constraints that must be true for every input in a set of associated test cases, any shared environmental needs, any shared special procedural requirements, and any shared case dependencies. Sets of associated test cases may be identified as scenarios (also commonly called scripts or suites). Test scenarios should be designed to be as reusable as possible for regression testing, revalidation testing for changes, and training new employees who must either use or support the system over time.

テスト項目の一般的な属性を要約する。これには、関連するテスト項目の集合中のすべての入力に対し真でなければならない入力制約、共通の環境ニーズ、共通の特別な手順の要求、及びテンプレートの補強、共通の項目間の依存関係が含まれる場合がある。関連するテスト項目の集合は、シナリオ(一般に、スクリプトあるいはスイートとも呼ばれる)として特定される場合がある。回帰テスト、変更のための妥当性再確認テスト、及び、いずれシステムを使ったりサポートしたりしなければならない新入社員のトレーニングのために、テストシナリオはできる限り再利用可能のように指定するべきである。

機能テスト:

テスト設計テンプレート

対象とするコンポーネント名を記述

機能テスト テスト観点	実施の有無	詳細な観点	入力条件	環境	特別な要求	依存関係	ID	備考
機能 対象となるコンポーネントの機能を実行し、期待される動作を確認する。対象となるコンポーネントの機能を実行し、期待される動作を確認する。対象となるコンポーネントの機能を実行し、期待される動作を確認する。	○/×	機能の実行結果が期待される動作と一致していることを確認する。	対象となるコンポーネントの機能を実行するための入力値、テーブル名、ファイル名など。	対象となるコンポーネントの機能を実行するための環境(ハードウェア、ソフトウェアなど)。	対象となるコンポーネントの機能を実行するための特別な要求(前提条件、事後条件など)。	対象となるコンポーネントの機能を実行するための依存関係(他のテストケースなど)。		
非機能 対象となるコンポーネントの非機能を実行し、期待される動作を確認する。対象となるコンポーネントの非機能を実行し、期待される動作を確認する。対象となるコンポーネントの非機能を実行し、期待される動作を確認する。	○/×	非機能の実行結果が期待される動作と一致していることを確認する。	対象となるコンポーネントの非機能を実行するための入力値、テーブル名、ファイル名など。	対象となるコンポーネントの非機能を実行するための環境(ハードウェア、ソフトウェアなど)。	対象となるコンポーネントの非機能を実行するための特別な要求(前提条件、事後条件など)。	対象となるコンポーネントの非機能を実行するための依存関係(他のテストケースなど)。		

テスト実施の有無を記述

結合テストで実施すべきテスト設計事項を明記する

機能テストの観点

非機能テスト:

テスト設計テンプレート

対象とするコンポーネント名を記述

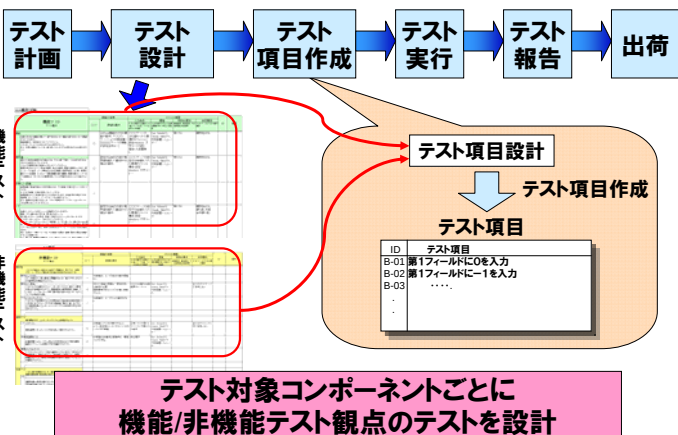
非機能テスト テスト観点	実施の有無	詳細な観点	入力条件	環境	特別な要求	依存関係	ID	備考
非機能 対象となるコンポーネントの非機能を実行し、期待される動作を確認する。対象となるコンポーネントの非機能を実行し、期待される動作を確認する。対象となるコンポーネントの非機能を実行し、期待される動作を確認する。	○/×	非機能の実行結果が期待される動作と一致していることを確認する。	対象となるコンポーネントの非機能を実行するための入力値、テーブル名、ファイル名など。	対象となるコンポーネントの非機能を実行するための環境(ハードウェア、ソフトウェアなど)。	対象となるコンポーネントの非機能を実行するための特別な要求(前提条件、事後条件など)。	対象となるコンポーネントの非機能を実行するための依存関係(他のテストケースなど)。		
結合テスト 対象となるコンポーネントの結合テストを実行し、期待される動作を確認する。対象となるコンポーネントの結合テストを実行し、期待される動作を確認する。対象となるコンポーネントの結合テストを実行し、期待される動作を確認する。	○/×	結合テストの実行結果が期待される動作と一致していることを確認する。	対象となるコンポーネントの結合テストを実行するための入力値、テーブル名、ファイル名など。	対象となるコンポーネントの結合テストを実行するための環境(ハードウェア、ソフトウェアなど)。	対象となるコンポーネントの結合テストを実行するための特別な要求(前提条件、事後条件など)。	対象となるコンポーネントの結合テストを実行するための依存関係(他のテストケースなど)。		

テスト実施の有無を記述

非機能テストの観点

結合テストで実施すべきテスト設計事項を明記する

テスト設計テンプレートの使用方法



テスト対象コンポーネントごとに
機能/非機能テスト観点のテストを設計

適用例:機能テスト設計テンプレート

機能テスト テスト観点	実施の有無	詳細な観点	入力条件	環境	特別な要求	依存関係	ID	備考
機能 対象となるコンポーネントの機能を実行し、期待される動作を確認する。対象となるコンポーネントの機能を実行し、期待される動作を確認する。対象となるコンポーネントの機能を実行し、期待される動作を確認する。	○/×	機能の実行結果が期待される動作と一致していることを確認する。	対象となるコンポーネントの機能を実行するための入力値、テーブル名、ファイル名など。	対象となるコンポーネントの機能を実行するための環境(ハードウェア、ソフトウェアなど)。	対象となるコンポーネントの機能を実行するための特別な要求(前提条件、事後条件など)。	対象となるコンポーネントの機能を実行するための依存関係(他のテストケースなど)。		
非機能 対象となるコンポーネントの非機能を実行し、期待される動作を確認する。対象となるコンポーネントの非機能を実行し、期待される動作を確認する。対象となるコンポーネントの非機能を実行し、期待される動作を確認する。	○/×	非機能の実行結果が期待される動作と一致していることを確認する。	対象となるコンポーネントの非機能を実行するための入力値、テーブル名、ファイル名など。	対象となるコンポーネントの非機能を実行するための環境(ハードウェア、ソフトウェアなど)。	対象となるコンポーネントの非機能を実行するための特別な要求(前提条件、事後条件など)。	対象となるコンポーネントの非機能を実行するための依存関係(他のテストケースなど)。		

- ・オブジェクトの生成/更新時、各属性に指定できる値の上限/下限、文字数の上限/下限、特殊値の指定/パターンを確認すること。
- ・ツリー/テーブルのページ切替について、表示件数の既定値、設定値と表示件数の確認を行うこと。
- ・検索件数の上限値の確認を行うこと。
- ・検索結果のオブジェクトの自動解放時間の設定値と変更値の確認を行うこと。
- ・属性ごとのセルの表示桁数の既定値、設定値、下限/上限値の確認を行うこと。

特殊値(文字)をリストアップすること(エスケープが必要な文字、エスケープ文字自身、HTML/XMLのシーケンスで意味のある文字、他)

[illegible]

試行し確認がとれました

1. 基本から学ぶソフトウェアテスト テストのプロ、を目指す人のために
Cem Kaner, Jack Falk, Hung Quoc Nguyen 著 バグ技術者交流会誌
日経BP社 ISBN978-4-8222-8113-7
2. ソフトウェアテスト技法 自動化、品質保証、そしてバグの未然防止のために
ボリス・ハイザー 著 小野間彰, 山浦恒矢訳
日経BP出版センター ISBN978-4-8227-1001-9
3. はじめて学ぶソフトウェアのテスト技法
リー・コーブラント 著 奈雅彦訳
日経BP社 ISBN978-4-8222-8251-6
4. 現場の仕事がバリエーション進む ソフトウェアテスト手法
堀江 孝 著
技術評論社 ISBN978-4-7774-1271-8
5. 「インターネットアプリケーションのためのソフトウェアテスト」
Hung O. Nguyen, Michael Hackett, Bob Johnson 著 村松淳次訳
ソフトバンククリエイティブ ISBN978-4-7973-2206-6
6. ソフトウェアテスト標準用語集(日本語版) Version 1.1 (2006年12月22日)
国際ソフトウェアテスト資格認定委員会 用語集作業班
http://www.jsitp.jp/dt/JSTQC-glossary/V1.1.pdf
7. 「ソフトウェアテスト」の用語集カタログ http://www.veriserve.co.jp/r_and/d/
8. ソフトウェア品質管理辞典 第2版 The QualityBOK Guide
オーム社 ISBN978-4-274-50162-3
9. IT用語辞典 (e-Words) <http://e-words.jp/>
10. ソフトウェアエンジニアリング基礎知識体系—SWEBOOK 松本吉弘訳
オーム社 ISBN978-4-274-94676-9
11. 要求工学 第23回: 非機能要求 山本修一郎著
ビジネスコミュニケーションズ 月刊ビジネスコミュニケーション 2006年9月号

NEC