

ITプロジェクトのテスト設計

ITプロジェクトのテスト設計

(株)オフィス新



Copyright (C) Office Shin, Ltd. All Rights Reserved.

目次

- 1章 ソフトウェア品質の重要性
- 2章 ソフトウェアの品質とテスト
- 3章 テストに対する心構え
- 4章 テストライフサイクルとテスト計画
- 5章 テスト網羅性の指針
- 6章 テスト技法
- 7章 テスト設計・準備における留意事項
- 8章 単体テスト設計
- 9章 結合テスト設計
- 10章 総合テスト設計
- 11章 テスト管理
- 付 録



Copyright (C) Office Shin, Ltd. All Rights Reserved.

1章 ソフトウェア品質の重要性

- 1.1 高まるソフトウェア品質の重要性
- 1.2 品質問題の考察
- 1.3 不具合の原因
- 1.4 ソフトウェア品質を向上させるには

1.1 高まるソフトウェア品質の重要性

最近のITシステム事故より



社会問題、人身事故、組織の信用失墜・・・。
インフラを支えるソフトウェアの問題は、社会に大きな影響を与える。

2章 ソフトウェアの品質とテスト

- 2. 1 良い品質のソフトウェアとは
- 2. 2 品質の2つの側面
- 2. 3 ソフトウェアの品質尺度(メトリクス)
- 2. 4 ISO25010(ソフトウェアの品質特性)
- 2. 5 作業の品質尺度の例
- 2. 6 ソフトウェアのバグとは
- 2. 7 ソフトウェア品質を決定づける様々な要因
- 2. 8 問題解決コストは下流工程になるほど増大する
- 2. 9 品質管理の基本方針
- 2. 10 レビューとテスト

2. 1 良い品質のソフトウェアとは 1/2

品質の定義は、団体や個人によって様々な定義されてきたが、現在では世界的に顧客満足度の向上を最終目的とすることが一般的である。

ISO9000	「本来備わっている特性の集まりが、要求を満たす程度」
ISO 25000	「指定された特定の条件で利用する場合の、明示的または暗黙的ニーズを満たすソフトウェア製品の能力」
PMBOK	「本来備わっている特性がまとまって、要求事項を満たす度合い」

品質 = 顧客の要求事項への適合（顧客の満足度向上）

■役割毎の品質視点

プログラマ視点	開発（コーディング）バグの有無
SE視点	設計、開発バグの有無
PM、上級SE視点	顧客満足度

「品質は誰かにとっての価値である」

(共立出版(株)「ワインバーグのシステム思考法」より引用)

3章 テストに対する心構え

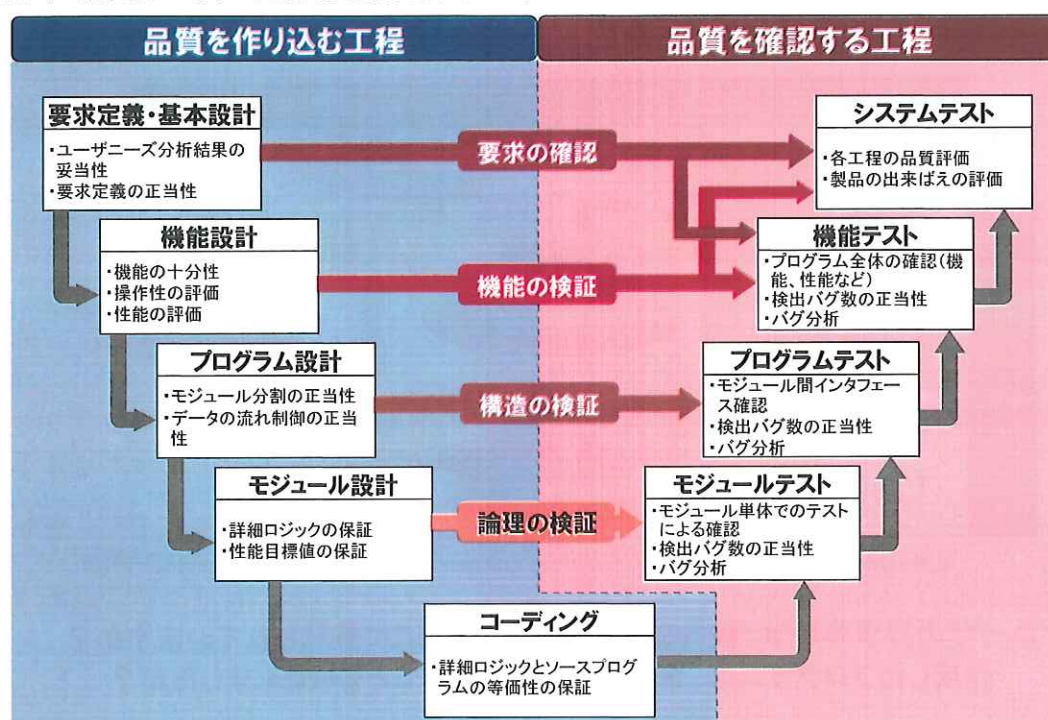
- 3. 1 テストの難しさ
- 3. 2 テストの目的
- 3. 3 良いテストとは
- 3. 4 ソフトウェアテストの7原則

4章 テストライフサイクルとテスト計画

- 4. 1 開発工程における品質保証の観点
- 4. 2 V&V(検証と妥当性確認)
- 4. 3 ソフトウェア品質保証の原点
- 4. 4 W字モデル
- 4. 5 テストフェーズ
- 4. 6 実施するテストの種類
- 4. 7 品質保証のタイプ
- 4. 8 テストライフサイクル
- 4. 9 テスト計画テンプレート
- 4. 10 テスト計画
- 4. 11 プロジェクト計画とテスト計画

4. 1 開発工程における品質保証の観点

ここでは、V字モデルでの品質保証の観点例を次に示す。品質を作り込む工程とそれに対応する確認工程の対応を確認されたい。

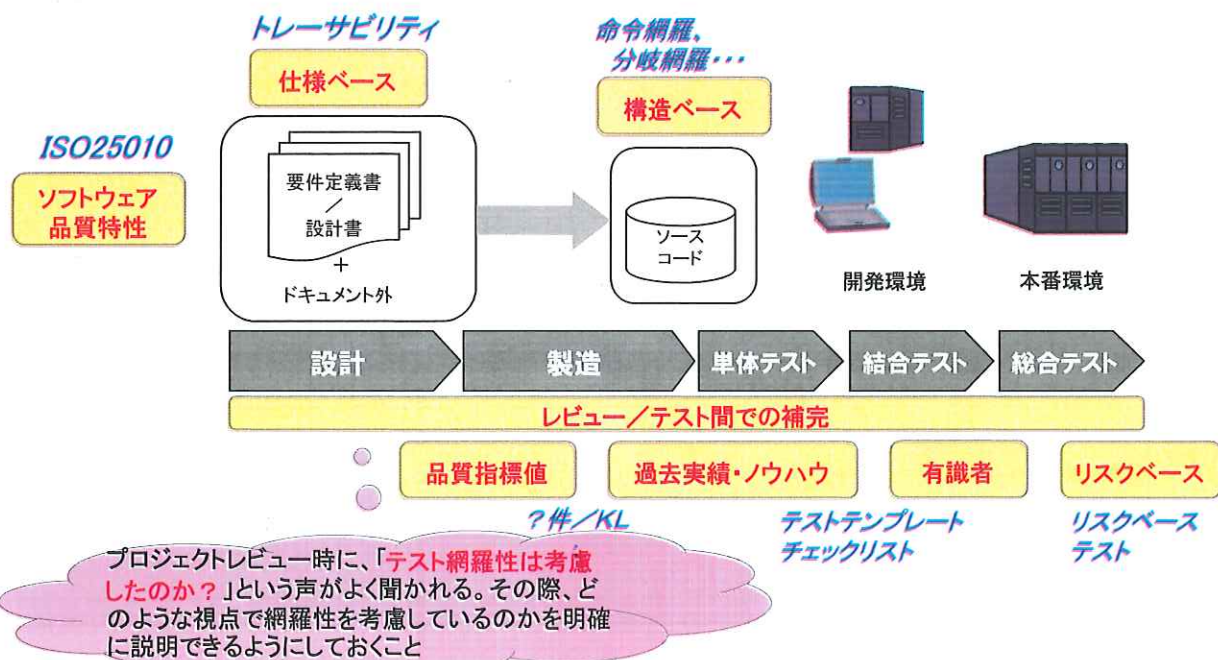


5章 テスト網羅性の指針

- 5.1 テスト網羅性の観点
- 5.2 ソフトウェア品質特性を利用した網羅性チェック
- 5.3 構造ベースの網羅性チェック
- 5.4 仕様ベースの網羅性チェック
- 5.5 設計者、有識者による網羅性の補完
- 5.6 品質指標値による網羅性チェック
- 5.7 四象限グラフによる網羅性チェック
- 5.8 リスクベーステスト観点による網羅性チェック
- 5.9 レビュー／テスト間での網羅性の補完
- 5.10 過去のバグ情報、ノウハウによる網羅性の補完(エラー推測)

5.1 テスト網羅性の観点

網羅率は「網羅を確認したい対象の実数÷網羅を確認したい対象の総数」で算出する。テスト対象をモレなく網羅するためには、**様々な観点からテスト項目の網羅性を確認**する必要がある。ここでは、その網羅性チェックの観点を示す。



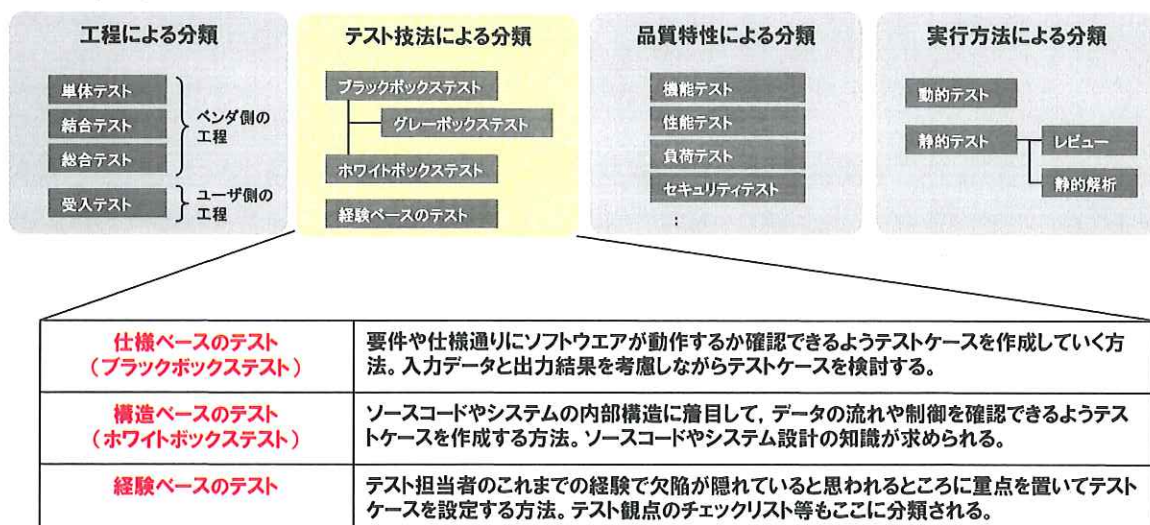
6章 テスト技法

6.1 テストの分類

6.2 テスト技法の種類

6.1 テストの分類

テストは様々な観点から分類されている。どのテスト工程でどのテストを実施するのかは、テスト計画段階であらかじめ決めておく。ここでは、テスト技法の分類に焦点を当て、様々なテスト技法を紹介する。



なお、プログラムの内部やシステム設計内容を考慮しながら、機能仕様から見たテストを行う手法を、**グレーボックステスト**と呼ぶ。

テスト設計

別冊2 テスト技法

(株)オフィス新



目次

- テスト技法の種類
- テスト技法の代表的な検討場面
- テスト技法① 制御フローテスト
- テスト技法② データフローテスト
- テスト技法③ 同値分割
- テスト技法④ 境界値分析
- テスト技法⑤ デシジョンテーブル
- テスト技法⑥ ドメイン分析テスト
- テスト技法⑦ 状態遷移テスト
- テスト技法⑧ ペア構成テスト
- テスト技法⑨ エラー推測
- テスト技法⑩ ユースケーステスト
- テスト技法⑪ リスクベーステスト
- テスト技法⑫ トランザクションフローテスト
- テスト技法⑬ アドホックテスト
- テスト技法⑭ 探索的テスト
- テスト技法⑮ 静的テスト

7章 テスト設計・準備における留意事項

- 7. 1 テスト設計における留意事項
- 7. 2 テスト実施の流れ
- 7. 3 テスト準備における留意事項
- 7. 4 テスト実施における留意事項

7. 1 テスト設計における留意事項

8章 単体テスト設計

- 8.1 単体テストの目的と内容
- 8.2 単体テスト設計の観点
- 8.3 単体テストの単位
- 8.4 単体テスト設計の手順
- 8.5 単体テスト設計の詳細手順
- 8.6 単体テストの準備と実施
- 8.7 単体テスト設計でのテスト技法一覧

8.1 単体テストの目的と内容

単体テストの目的は、**詳細設計書通りにソースコードが製造されているかどうか、ソースコードの妥当性を確認**することである。

システムは各モジュールの集合体であるため、各モジュールの細部までの網羅的なテストを実施し、品質確保を図る事が、後続の結合テスト、総合テストへの必要条件となる。

単体テストの観点と内容を大別すると以下の通りでなる。

モジュールの機能テスト	モジュールの機能(全ての入力値、出力値、オプション、エラーケース等)をテストする
モジュールのロジックテスト	一般的に、モジュールのロジック(経路)について、すべての分岐が真偽両方の側とも実行されるようにテストする。テスト対象モジュールに関連する上位/下位モジュールが必要な場合、ドライバ/スタブ(上位/下位モジュールに対する疑似モジュール)を作成し使用する

単体テスト設計時の留意点

- ・ 新規作成部分を確実にテストすること
- ・ 他のモジュールとの入出力条件が仕様通りであるか確認できること
- ・ 繰り返し処理が正常か確認できること(**少なくとも0回、1回、最大回についてテスト**する)
- ・ エラー処理が動作するか確認できること
- ・ 境界値、中間値の確認ができること

9章 結合テスト設計

- 9.1 結合テストの目的と内容
- 9.2 結合テスト設計の観点
- 9.3 結合テストの単位
- 9.4 結合テスト設計の手順
- 9.5 結合テスト設計の詳細手順
- 9.6 結合テストの準備と実施
- 9.7 結合テスト設計でのテスト技法一覧

9.1 結合テストの目的と内容

結合テストの目的は、概要設計書(機能設計)に基づいた一連の機能やコンポーネントが、設計書通りにシステム機能として実現されているかどうかという視点で不具合がないを確認することである。結合テスト設計は、概要設計書(機能設計)に基づいて、外部機能に着目して行う。また、基本設計書(システム構成)に基づいて、コンポーネント間のインタフェースに着目したテスト項目設計を行う。

結合テストの内容

- ・機能(プログラム間連携)、業務(機能間連携)、サブシステム(業務間連携)が、設計書通りの機能として実現されているかどうかを確認する
- ・他サブシステムとの連携インタフェース、他システムとの連携インタフェースが、設計書通りとなっているかを確認する
- ・機能間(JOB間、JOBとオンライン等)の排他制御が正しく動作するかを確認する
- ・総合テストで実施すべき、非機能テスト(負荷テスト、性能テスト、運用テスト等)についても結合テストの段階から確認することが望ましい

結合テストは、一連のテストの流れをパターン化したテストシナリオに基づいて行う。プログラム間連携、機能間連携、業務間連携等の確認には、ユーザが行う一連の業務に基づいたテストシナリオが必要となる。また、排他制御の確認や負荷テストでは、システムの利用や運用がどのように行われるのかを考慮したテストシナリオが必要となる。

結合テスト計画及びテストシナリオ、テスト仕様が確定した段階で、ユーザに報告し、結合テスト設計について合意を得る。

10章 総合テスト設計

- 10. 1 総合テストの目的と内容
- 10. 2 総合テスト設計の観点
- 10. 3 総合テスト設計の進め方の手順
- 10. 4 総合テストの準備と実施

10. 1 総合テストの目的と内容

総合テストの目的は、システム要件と基本設計書に基づき、システム全体が業務処理として確実に動作するかどうかという視点で不具合がないかどうかを確認することである。また、システムに求められる信頼性、性能、操作性等の各種観点から、納品するシステムに問題点がないかを確認することである。

総合テストでは下記のテストを実施する。

総合テストの内容

- 機能要件ベース
要件定義書／基本設計書に記載されている業務シナリオ(業務フローやユースケース等)を元に、機能要件を確認する
- 非機能要件ベース
要件定義書／基本設計書に記載されている品質要件を元に、非機能要件を確認する
- テスト観点ベース
様々なテスト観点(テスト分類、タイミング、機能組み合わせ等)を考慮したテストを実施する

契約上、総合テストは客先責任作業としているプロジェクトが多い。しかし、総合テストまでをベンダ側責任で実施し、運用テストを客先の検収作業としているプロジェクトも見受けられる。総合テストの作業責任が客先なのかベンダなのかにより、客先とベンダの作業内容は大きく異なるため、プロジェクト計画段階に作成する全体テスト計画において、総合テストの作業責任と役割分担を明確にしておく必要がある。

11章 テスト管理

- 11.1 品質管理の概要
- 11.2 品質データ分析の実施
- 11.3 テスト密度とバグ密度の関係
- 11.4 テスト密度と網羅率の関係
- 11.5 信頼度成長曲線
- 11.6 品質見解への留意点

11.1 品質管理の概要

品質基準を満たすことに焦点を合わせたプロセスであり、品質上の目標を達成するための取り組みを考え、作業(結果)を計測し、目標に近づけるために行う活動。品質管理ツールを利用する場合が多い。

品質管理ツールの例

ゴンペルツ曲線 (バグ管理グラフ)

QC 7つ道具

- チェックシート
- グラフ
- パレート図
- 特性要因図
- 散布図
- ヒストグラム
- 管理図

