テスト方針書

2016/11/22作成

作成者　長田裕介

目次

1. テストの目的・範囲
   1. 目的
   2. テスト種類と範囲
   3. 各局面と各テストの関係
2. テスト局面
   1. テストの考え方
      1. 単体テスト
      2. サブシステム内統合テスト
      3. サブシステム間統合テスト
      4. システムテスト
   2. テスト局面の定義と目的
3. 開始・完了基準
   1. 単体テスト
   2. サブシステム内統合テスト
   3. サブシステム間統合テスト
   4. システムテスト
4. テストスケジュール
5. 実施手順
6. 障害管理方針
7. テストの目的・範囲
   1. 目的

　本方針書は、売上管理システムの各テスト局面におけるテストの位置付けと実施目的、および範囲を定義することにより、テスト作業効率、システムの品質確保、スケジュール通りのテスト実施を目的とする。

　本方針書では、プロジェクト全体における計画とし、各テスト局面における作業内容は、各テスト開始までに各テスト計画書に記述することとする。

　尚、実施に際しての方針・方法・取り決めの事項が策定される都度、これに追加していくものとする。

* 1. テスト種類と範囲

　本プロジェクトにおいて実施をするテストとその範囲を以下に示す。

* 単体テスト

範囲：モジュール内関数

* サブシステム内統合テスト

範囲：コンポーネント内入出力

* サブシステム間統合テスト

範囲：コンポーネント間入出力

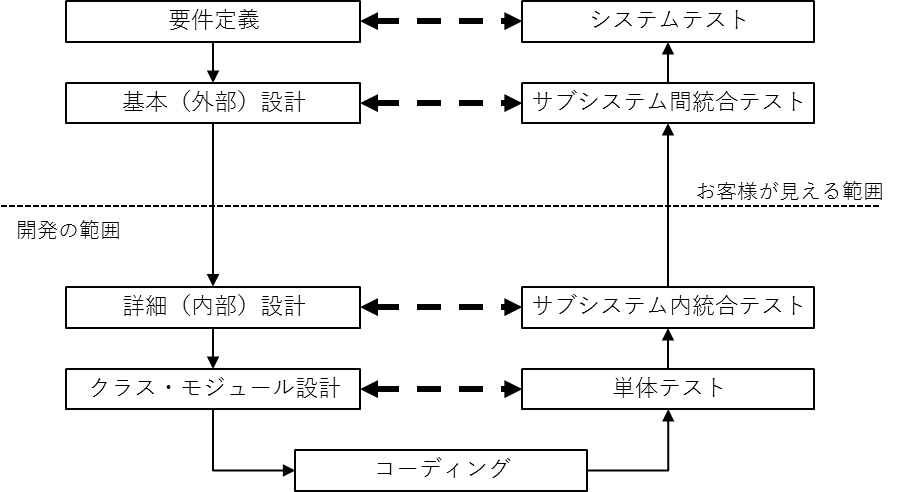
* システムテスト

範囲：システム全体

　ここで、「モジュール」とはプログラムの最小構成単位あるいは１つのコンパイルの単位を、「コンポーネント」とはモジュールによって構成される単位を表す。モジュールはプログラム内のメソッドや関数、APIが該当し、コンポーネントはサブシステムにおけるトランザクション処理や複数のトランザクション処理からなるコントロールが該当する。

* 1. 各局面と各テストの関係

　本プロジェクトにおける各局面と各テストの関連（概要図）を以下に示す。



1. テスト局面
   1. テストの考え方

　以下にアプリケーション開発から見た場合のテストに関する考え方を示す。



* 1. テスト局面の定義と目的
     1. 単体テスト

　単体テストの目的は、各モジュールの欠陥除去と機能検証にある。

テストに使用するデータは「モジュール仕様書」「メソッド記述書」をもとに作成する。このとき、モジュール機能の全てを検証できるようなテストケースを作成し、テストデータを準備しなければならない。また、単体テストはコンパイル済モジュールに対して行い、モジュール仕様を満たすことの確認をする。

単体テストはモジュール単位で、全方向分岐の確認を行う。ここで全方向分岐の確認とは、全てのロジックに対して確認を行うことである。すなわち、以下の条件を満たす確認を行えばよい。

・全ての分岐ロジックに対して何らかのテストケースで一度以上通っていて、通ったことのないロジックがない。

・ひとつのif文で分岐先が複数の場合は、その分岐先を全て確認している。

・別のif文との、全ての分岐の組み合わせを確認する必要はない。

　単体テストは全方向分岐のホワイトボックステストにより行う。下位モジュールへの実リンクは行わず、ドライバを使用する。なお、共通部品モジュールやデータベースとの結合は統合テストにて実施する。

* + 1. サブシステム内統合テスト

　サブシステム内統合テストの目的は、サブシステム内の各コンポーネントの欠陥除去と機能検証と、サブシステム内のコンポーネント間インターフェイスの検証にある。

　テストに使用するデータは「外部仕様書」「内部仕様書」をもとに作成する。サブシステム内の各コンポーネントの機能とインターフェイスの検証を行うためのテストケースを作成し、テストデータを準備する。その後、統合順序に従って統合テストを行う。

　サブシステム内統合テストはコンポーネント単位での結合テストを行う。また、コンポーネント間の結合テストを行う。このとき、以下の条件を検証する。

　　・実リンクによるモジュール間のインターフェイス確認

　　・ページ遷移の確認

　　・トランザクション（アクション）・データベースJOB単位の機能の確認

　　・複数トランザクション（アクション）間の機能の確認

　　・複数JOB間の機能の確認

　　・サブシステム単位の機能の確認

　サブシステム内結合テストはボトムアップ＋インクリメンタル方式およびビッグバン方式を併用して行う。オンラインに関しては、共通部分、データベースを結合して実施する。ただし、バッチの起動および終了に関しては基盤との結合はサブシステム間結合テストで行う。

* + 1. サブシステム間統合テスト

　サブシステム間統合テストの目的は、サブシステムの欠陥除去と機能検証と、サブシステム間インターフェイスの憲章にある。

　テストに使用するデータは「外部仕様書」をもとに作成する。サブシステムの機能とインターフェイスの検証を行うためのテストケースを作成し、テストデータを準備する。その後、統合順序に従って統合テストを行う。

　サブシステム間統合テストは、ブシステム間インターフェイス単位での結合テストと、複数のサブシステム間の結合テストを行う。

　サブシステム間結合テストは、下の2フェーズに分割して行う。

　　・フェーズ1　データベース側テスト

起動・接続テスト

ホストとの接続

機能テスト（セッション維持）

監視通知

障害テスト

　　・フェーズ2　アプリ側テスト

2つのサブシステムの結合テスト（ラウンド1）

2つ以上のサブシステムの結合テスト（ラウンド2）

　（ラウンド1と同一のテストシナリオにて、一貫テストを実施）

アプリとデータベースの結合

* + 1. システムテスト

　システムテストの目的は、コンポーネント間・サブシステム間の統合テスト済モジュールを、システム全体としてテストし、システム要件通りに機能し、稼働することを確認することにある。

　テストに使用するデータは「要件定義書」「外部仕様書」をもとにテストケースを作成する。システムテスト計画書に基づきテストケース、テストデータを設定した後、システムテスト仕様書に基づきシステムテストを実施する。

　システムテストは以下のカテゴリに分割される。

1. ファシリティ（機能部分）テスト

要件定義書で定義した機能及び外部設計書で定義した機能を実施し、要件の網羅性を確認する。

エラー処理など正常、異常ケースの動作を確認する。

ロジックの最終確認を行う。

1. ユーザビリティ（使用性）テスト

入力時のエラーチェックが正常に機能しているか。

不正な操作を事前に禁止することができるか。

1. セキュリティテスト

運用面でのセキュリティを確認する。

1. リカバリテスト

アプリケーションエラー、ハードウェアエラー、データエラー等が発生したときの障害時システム回復処理の確認を行う。

1. 大容量テスト

大容量のデータに対するトランザクションを実行し、パフォーマンス測定を行う。

1. ストレステスト

システムに対して処理負荷を与えて、パフォーマンス測定を行う。

1. 開始・完了基準

各テスト局面における開始・完了基準を下記に示す。

* 1. 単体テスト

　■開始基準

(1) 単体テスト計画書が完成していること

(2) エラー・フリーのコンパイル済モジュールができていること

(3) テストケースの設定がされていること

(4) 単体テスト環境が作成済であること

　■完了基準

(1) 設定したテストケースが全て完了していること

(2) 全方向分岐をすべて網羅していること

　　ただし、網羅できていない部分が発生した場合は別途アクションプランが

作成されていること。（騎乗検証・後続の局面での実施）

* 1. サブシステム内統合テスト

　■開始基準

(1) サブシステム内統合テスト計画書ができていること

(2) 対象モジュールの単体テストが全て完了していること

(3) テストケースの設定がされていること

　■完了基準

(1) サブシステム内の機能、モジュール間・コンポーネント間のインターフェイスが

検証済みであること

(2) 全テストケースの95%以上が検証済み、残ケースのアクションプランが作成され

ていること

* 1. サブシステム間統合テスト

　■開始基準

(1) サブシステム間統合テスト計画書ができていること

(2) 結合アプリのコンポーネントのサブシステム内統合テストが完了していること (3) テストケースの設定がされていること

　■完了基準

(1) サブシステムの機能、サブシステム間のインターフェイスが検証済みであること

(2) 全テストケースの95%以上が検証済み、残ケースのアクションプランが作成され

ていること

(3) 未解決の障害が管理、リストされており、重要度1(システム全体に影響するもの)

の障害が0件であること、もしくはシステムに障害が及ばないようなアクション

プランがあること

* 1. システムテスト

　■開始基準

(1) システムテスト計画書ができていること

(2) システム単位の機能及びシステム間インターフェイスが確認されていること

(3) テストケースが設定されていること

　■完了基準

(1) 設定したテストケースが全て完了していること

1. テストスケジュール
2. 実施手順
3. 障害管理方針

　障害管理手順については、個々のテスト計画書の中で記述する。