

単体テスト仕様書の作成

作成日: 2014/04/23

作成者: 西村

更新履歴

更新日	更新概要	作業者
2014/04/23	・ 新規作成 (仮)	西村
2014/05/14	・ 注意点の追加	西村
2017/11/30	・ 公開用の体裁調整	西村
	・	
	・	

単体テストの目的

単体テストは「機能単体でプログラムが期待した通りに動作するか」を確認する目的で行います※1。

- ・ ソエルではおおむね画面単位で単体テストをお願いしています※2。また、ここでいう「期待通り」とは「期待通りにエラーになる」ことも含みます(例えば入力欄に範囲外の値を入れた時に想定したエラーメッセージがちゃんと出る等※3)
- ・ 単体テスト仕様書(テストケース)は「対象のプログラムがどのような動作を期待されているのか」「どこまで想定してプログラムが作成されているのか」を一覧で見ることができる資料としても活用できます。

※ 1 複数機能間でのテストは結合テストになります。

※ 2 単体テスト仕様書を作成する場合。単体テストをプログラムで書く(PHPUnit 等を使用する)場合はクラス単位になることが多いです。

※ 3 正常に動くかのテストケースは「正常系」、エラーが出るかなどのテストケースは「異常系」と呼ばれます。

単体テスト仕様書のフォーマット

基本的に過去の案件のものをコピーして使って頂ければ問題ないです。シートは現状では「テスト観点」「テストケース詳細」のシートのみ流用してもらえば OK です。

「テストケース詳細」の項目は下記の通りです。

No	テストケースの番号(連番)
テスト項目／条件	「初回起動時」等、どの条件でのテストかの分類
検証内容	テストする事柄
結果	テストした結果(○か×)

- ・ 他社指定のフォーマットがある場合は他社のフォーマットに従って下さい。

テストケース(テスト項目)の作成の目安

テストケースは自分以外の人間が実施することを想定して作成して下さい。

- ・ 作成者と実施者を別にする
- ・ ダブルチェック(他の人による再試験)をする
- ・ レビュー(テストか妥当かを複数人で検査する)をする

ということもあるので、他人が単体テスト仕様書だけを見て何を、どうテストすればよいのかわかるものが望ましいです。

また、網羅性が高い(= 漏れなくテストできている/できるようになっている)とよいです。

※ 時間の関係上「代表的なケースのみパスできていればよい」、としてテストを一部省略する場合があります。その場合は「なぜこのテストを省略しても問題ないか」の理由も記載するとよいです。

下記、できているとよいことを列挙します。

1. テストでの前提条件、使用するテストデータ(DB)、ファイルなどはすべて説明に残す
2. 動作条件でテストケース进行分类する
3. 入力(データや条件)によって出力がどうなるかがわかるようにテストケースを書く
4. 境界値をテストに含める
5. 異常値をテストに含める
6. 入力/出力のパターン(条件)を網羅する

イメージがつかない場合は次のページ以降を確認してみてください。

その他の注意点

- ・ テストケースは設計を元にして書いて下さい。実装からテストケースを書いた場合テストが通るのは当たり前で意味がありません。

1. テストでの前提条件、使用するテストデータ(DB)、ファイルなどはすべて説明に残す

他人がすぐテストに取り掛かれるよう、最初に前提条件を記述するとわかりやすいです(テストケースもその前提を元に作成できるので無駄な説明が省略できます)

【テストケース】	
<input type="checkbox"/>	(前提)
	<ul style="list-style-type: none"> ・ あらかじめ[テストデータ]シートのデータをDBに適用すること バッチキュー(TBL_BATQUE)は一度全データクリアしてください ・ 単体テストのため、バッチを直接起動して動作を確認すること Visual Studio上から実行するとB095実行→A500/A061確認→B095実行のように画面とバッチを切り替える時間がかかる(スタートアップ側はBatch#B095YSNCustCSVBatch#bin#Release# をコマンドプロンプトで開いて「 B095YSNCustCSVBatch.exe 3400021752 」のようにバッチキューIDを引数に与えて直接EXE起動することを推奨します。 また、バッチのテストデータはあらかじめ処理状態を処理中(cProcStat=2)としています(基盤側のバッチ処理条件を満たすため)

2. 動作条件でテストケースを分類する

「初回起動時」「○○ボタン押下時」のような動作を条件として分類分けすると仕様書の見通しがよくなり、テストもしやすくなります。

No.	テスト項目/条件	検証内容
3	メニューボタン押下時	・メニューウィンドウが表示されるか確認
4	リンクコピーボタン押下時	・画面URLがクリップボードにコピーされるか確認 ※メモ帳などに貼り付けてURLが表示されるか
5	閉じるボタン押下時	・画面が閉じるか確認
6	CSV出力ボタン押下時	<ul style="list-style-type: none"> ・出力先ファイル名[101] が空白の場合、下記メッセージを表示して処理が中断するか確認 「出力先ファイル名は必ず入力して下さい」(C001/C501) ・出力先ファイル名[101] にファイル名に使用できない文字・記号が使われていた場合、下記メッセージを表示 「ファイル名に使用できない文字・記号が含まれています」(C001/C524)

3. 入力(データや条件)によって出力がどうなるかがわかるようにテストケースを書く

入力 → 結果、と言う形の表にすると見やすくなりますしテストするときもテストしやすいです。

・出力先ファイル名[101]の拡張子が.csv(大文字も可)以外である場合、下記メッセージを表示して処理が中断するか確認 「ファイルの拡張子が正しくありません。csvのみ設定できます」(C001/C553)	
テストケース	
ファイル名	結果(×=エラー)
取引 先情報2014.csv	○
取引 先情報2014.CSV	○
取引 先情報2014.txt	×
取引 先情報2014.csvx	×
取引 先情報2014.xcsv	×

※入力されたファイル名の拡張子が「.csv」「.CSV」以外の場合はエラーとするテストケースの例

4. 境界値をテストに含める

例えば「身長入力欄は整数で 20～300 まで入力できるものとする」という仕様があった場合、境界値の 20、300 とその近辺(19、21、299、301)はバグが出やすいので特にテストするようにして下さい。

5. 異常値をテストに含める

例えば下記のようなものがテスト出来ているとよいです。

入力欄の種類	異常値の例
整数 (「100」等)	<ul style="list-style-type: none"> ・ 数字以外 (「A」等) ・ 大きすぎる値 (「10000000000000000000」等)
実数 (「1000.33」等)	<ul style="list-style-type: none"> ・ 数字以外 (「A」等) ・ 大きすぎる値 (「10000000000000000000」等) ・ ドットが連続する値 (「121.14.341」等)
日付 (例えば「YYYY/MM/DD」の形式)	<ul style="list-style-type: none"> ・ 数字以外(「AAAA/BB/CC」等) ・ 無効な日付 (「2001/02/29」「2014/15/99」等)
文字列	<ul style="list-style-type: none"> ・ 長すぎる文字列 (想定が 50 文字程度の所に 500 文字など) ・ 特殊な文字 (HTML だと<script>タグ、 SQL だと「'」や LIKE で使う「%」など)

文字列の「特殊な文字」の部分は特殊な文字が特殊なまま扱われていないか(= しっかりエスケープされているか)を確認して下さい。それができていないと予期せぬ動作を引き起こす原因になります。

(HTML の場合だと<script>を実行される※ 1、表示が崩れたりする、SQL の場合だと DB を不正に操作される※2)

※1「XSS」(クロスサイトスクリプティング)と呼ばれます。

PHP で「echo するときは h()で囲うように」とお願いしているのはこのためです。

※2「SQL インジェクション」と呼ばれます。

6. 入力/出力のパターン(条件)を網羅する

例えば入力項目として「性別(男性・女性の 2 パターン)」「身体活動 (低・中・高の 3 パターン)」というものがあり、組み合わせにより結果が変わる場合は 2 × 3 の 6 パターン分のすべてのテストケースを用意して下さい。

※ 組み合わせが膨大な場合は「なぜ省略しても問題ないか」を記載して一部省略するようにしてもよいです。

組み合わせのテストケースの例。「性別」「身体活動」が入力側、「結果」が出力側です。

性別	身体活動	結果
男性	低	1.4
男性	中	1.5
男性	高	1.6
女性	低	1.2
女性	中	1.3
女性	高	1.4

場合によってはマトリクス(2次元表)にするのもあります。

性別 / 身体活動	低	中	高
男性	1.4	1.5	1.6
女性	1.2	1.3	1.4