

I T 検証標準工法ガイド

Ver. 1.1 (2014/4)



一般社団法人 I T 検証産業協会
標準化部会

IT 検証標準工法ガイド

はじめに

現在、IT に関するソフトウェアは、多くの人々が日常利用する様々なものに搭載されており、もはや人間が活動する上で欠かすことの出来ない基盤と言える状況です。また、ソフトウェアを活用して実現することの要求はより高度で複雑になっており、その動作の影響範囲も広がることで、ソフトウェアの品質はさらに高いレベルで求められています。

必然的に、ソフトウェアの開発やテストを行っている現場も無数に存在し、今このときにも多くの人々がソフトウェアの開発やテストに関わっています。しかしながら、ソフトウェアはその性質上、要求項目や実施項目が増えてそこに関わる関係者の数が増えると、各作業の依頼者と実施者が認識を一致させながら、漏れの無い作業を高品質で進めることが容易ではなくなります。

このような課題に対応するために、「IT 検証産業協会（以後、IVIA と表記する場合もあり）」（IT Verification Industry Association：略称 IVIA アイビア <http://www.ivia.or.jp/>）では、テスト技術の向上やテストの標準化、およびテスト技術者育成の部会活動を行っています。

IVIA の標準化部会では、ソフトウェア品質を高い水準で確保することを目的として、ソフトウェアテストの標準工法の作成を推進しています。

標準工法により、テストに関わる多くの技術者が、同じ認識を持ちながら各作業工程を漏れなく進められるようになり、テスト品質とソフトウェア品質を向上させられると考えています。

本書ではソフトウェアテストの各工程で実施すべき標準工法として、作業の概要説明と実施項目を抽出していますが、各項目の詳細解説や具体例の提示までは行っていないため、理解を深めるには他の資料を参照する必要があります。

ソフトウェアテストに関わる技術者が、テストの各工程の全体像を把握し、実施する作業項目の認識を合わせるための標準工法のガイドとして、本書を活用頂くことを想定しています。

教育研修シラバスとの関係

本書で挙げている各項目は、IVIA が発行する以下の IT 検証技術者 教育研修シラバスを基に抽出しています。

- ・ High-Level Syllabus (Level 5) Ver. 1.0.0 (2011/07)
- ・ ミドルレベル（レベル 3、レベル 4） Ver. 1.0.1 (2009/05)
- ・ エントリーレベル（レベル 1、レベル 2） Ver. 1.0.2 (2009/05)

著作に関する注意点

本書の著作権は、IVIA（一般社団法人 IT 検証産業協会）が管理、所有します。本書の全部ないし一部を承諾なしに複製し利用することは、著作権法の制限事項に基づき、禁じられています。

著作者：一般社団法人 IT 検証産業協会

執筆者（敬称略）：

飯野 智子	パナソニック システムネットワークス株式会社 システムソリューションズジャパンカンパニー
倉智 信之	日本プロセス株式会社
佐貫 賢治	パナソニック システムネットワークス株式会社 システムソリューションズジャパンカンパニー
柴尾 峰貴	日本プロセス株式会社
末田 信	株式会社富士通コンピュータテクノロジーズ
須山 浩克	日本プロセス株式会社
冬川 健一	株式会社ベリサーブ

改訂

バージョン	改訂内容	改訂日	備考
Ver. 1.0	初版	2011/4/1	
Ver. 1.1	一般公開版	2014/4/20	

目次

I T 検証標準工法ガイド	2
はじめに	2
教育研修シラバスとの関係	2
著作に関する注意点	3
改訂	4
1. I T 検証標準工法ガイドの構成	7
2. テスト要求分析	8
2.1. テスト対象の把握	8
2.2. 開発の背景、現状把握	8
2.3. テスト要求の獲得	8
2.4. テストへのインプットの確認	9
2.5. テスト要求の分析	9
2.6. テスト要件、方針の定義	9
2.7. テストアーキテクチャ概要	9
2.8. テスト要件定義の検証	10
3. テストアーキテクチャ設計	14
3.1. テスト概要の把握	14
3.2. テスト内容の分析	14
3.3. テスト詳細設計・手法の選択	14
3.4. テストリソースの計画	14
3.5. テストアーキテクチャ設計の検証	14
4. テスト詳細設計・実装	17
4.1. 仕様の把握	17
4.2. 仕様管理（トレーサビリティ）	17
4.3. 網羅性設計-①（テスト要素抽出）	17
4.4. 網羅性設計-②（テスト変数抽出）	17
4.5. テストケース-①設計（基本フロー）	17
4.6. テストケース設計-②（組み合わせ）	18
4.7. テストケース設計-③（期待値設計）	18
4.8. モニタ設計	18
4.9. テスト環境設計	18
4.10. プロジェクト管理	18
5. テスト実行	23
5.1. テスト実行計画	23
5.2. テスト環境準備	23
5.3. テストケース準備	24
5.4. テスト実行	24
5.5. テスト実行記録	24
5.6. 不具合報告	24
5.7. テスト実行報告	25
5.8. プロジェクト管理	25
6. テスト評価／報告	31
6.1. テスト報告書を準備する	31
6.2. テスト活動全般の情報を収集する	31

6.3.	テスト活動を分析する.....	31
6.4.	改善項目の達成率を評価する.....	31
6.5.	分析結果と評価指標を比較する.....	32
6.6.	改善点を抽出する.....	32
6.7.	改善策を検討する.....	32
6.8.	テスト評価報告書を作成する.....	32
6.9.	テスト評価報告書を開示する.....	32

1. IT検証標準工法ガイドの構成

本書では、ソフトウェアのテスト工程を以下の6つの工程に分類し、各工程で実施すべき作業内容と作業項目について記述します。

1) テスト要求分析

テストに対する要求を獲得、分析し、テスト要件として定義する。

テストの成果として求められる結果を得るために、テスト全体の方針を定め、テストアーキテクチャ設計以降で実施する概要やポイントをテスト要件として定義する。

2) テストアーキテクチャ設計

テスト要件定義で定めた方針や概要を具体的に進める計画を作成する。

テストを実際に進める上でのテスト内容の分類や、体制やスケジュール、コストやリスクを具体化することで、テストプロジェクトとして活動を進める計画を示す。

3) テスト詳細設計

テストする内容や範囲、実施方法の概要を定める。

テスト要件定義やテストアーキテクチャ設計に基づき、テスト内容や網羅する範囲を定め、どのようにテストを行うか実施方法の概要についてもここで設計を行う。

4) テスト実装

テストする内容の詳細と具体的な実施方法を定める。

テスト詳細設計に基づき、テストする環境や機能、値、データ・コンテンツ、テスト手順および期待結果を具体的に定め、テスト実行可能な内容にブレークダウンする。

本書ではテスト詳細設計とテスト実装をまとめて記述している。

5) テスト実行

テストを実行して、テスト結果の報告や不具合報告を行う。

テスト実装内容に従ったテストを実行して、期待結果との差異報告、不具合報告、およびテスト進捗状況の報告を行う。

6) テスト評価／報告

テスト要件定義やテスト計画との差異評価と報告を行う。

テストアーキテクチャ設計以降の工程を実際に行った結果として、得られた成果や発生した問題点を抽出、分析してその評価を行い、今後に活かすための報告を行う。

次章以降の各章（各工程）の本文中には、作業内容の概要を記述し、別表として各工程の作業項目とそのINPUTとOUTPUTについて記述しています。

各工程の実施内容の理解を進めるには、各章、各節の説明と対応する作業項目を照らし合わせながら読み進めてください。

2. テスト要求分析

テスト要求分析は、テストを行う場合には、その内容や規模に関わらず必ず実施する工程である。

テストを依頼する側からの要求が、システム全体に対する漠然としたテスト要求である場合も、既に存在するテスト手順書を、与えられた環境で実行して結果を報告するという明確なテスト要求であっても、テスト要求を獲得・分析してテスト要件を定義した上でテストを進めるという工程は必ず存在する。

もちろん、上記前者の漠然としたテスト要求を分析する方が、検討すべき項目が多く複雑になるが、本書ではそちらの場合を想定して要求分析工程の進め方を述べていくものとする。

テスト要求分析工程で検討や実施が必要と考えられる項目を抽出し、表 2.1 にまとめた。

2.1. テスト対象の把握

テスト要求分析を行う上で最初に必要となることは、テスト対象がどのようなものであるかを把握することである。

はじめにテスト対象がどのような用途で用いられるものか、システムドメインや製品ドメインを明確にして、そのシステムや製品が使用される環境や動作環境等、システム全体の概要構成を把握する。また、テスト対象となる範囲がシステム全体であるのか、その中で動作する製品であるのか、あるいは部分的なモジュールであるのか、その種別を明確にする必要がある（例として、サーバー・クライアントシステム、組み込み製品・システム、アプリケーション、ミドルウェア、OS 等の基本ソフトウェア等の種別）。もしテスト対象に求められる規格や基準等がある場合には、それも明確にしておく。

また、テスト対象を利用するターゲットユーザー（コンシューマ向、業務向、特定ユーザ向、海外向等）や、ビジネス上の狙い（差別化・特化、顧客獲得、利益獲得等）を把握することにより、テスト方針や重点項目等の要件定義に活用する。

2.2. 開発の背景、現状把握

テスト要求分析は、テスト依頼側からの明文化された項目や、目に見えるシステム構成からだけではなく、テスト対象の開発の背景や現状を考慮した上で行う必要がある。

例えばテスト対象が新規開発なのか、派生開発・バージョンアップなのか、また、今回開発した内容は何であるのか等は、テスト依頼側がそういった情報を予め提供してくれない場合でも、テストすべき内容を検討する上で重要な情報であるため、獲得・把握する必要がある。

また、開発プロジェクトの目的や目標やコンセプト、開発体制や開発規模、開発の現在状況等もテストの進め方やボリューム・規模の設定の仕方に影響を与えるため、把握しておいた方が良い内容である。

さらに、開発プロジェクトで利用したプロセスや技法・手法、ツール等についても把握し、テストでどのように採り入れるべきかの検討につなげる必要がある。

2.3. テスト要求の獲得

テスト依頼側からテスト要求に関する基本事項を獲得する。

テスト依頼側には、テストをリクエストすることになった背景や理由が必ず存在する。すなわち、現状のある状況に対して、テストを行うことで確認・改善したいことが必ず存在し、マスターテスト計画があればそこに記載されている場合があり、マスターテスト計画が無い場合には、テスト依頼側へのヒアリングに基づいて獲得するものとなる。

マスターテスト計画が存在し、それに基づくテストレベルを実施する要求である場合には、そのテストレベルを把握する。マスターテスト計画が存在しない場合には、テスト要求がマスターテスト計画であるか、マスターテスト計画には基づかない何れかのテストレベルの実施要求となる。

テスト要求を獲得する段階でマスターテスト計画またはテスト依頼側からもっと広範囲の要求が出されている場合には、その要求を取り込み、後述するテスト要求の分析において活用する。

2.4. テストへのインプットの確認

テストを行う場合に参照・活用可能なインプット情報を確認する。

参照・活用可能なインプット情報には、既存のテストドキュメントやテストケース、および過去のテスト結果や不具合情報からなるテスト資産、および開発プロジェクトからのテスト対象や設計に関する情報からなるテストベースが含まれる。

これらのインプットは、テスト要求分析工程およびそれ以降のテストアーキテクチャ設計やテスト詳細設計工程でも活用されるものである。

2.5. テスト要求の分析

前述までの工程で獲得、把握した情報から、テストに要求される項目や内容を分析し、テスト要件や方針の定義につなげる。

テスト対象把握や開発背景把握、テスト要求概要で得られた情報から、テストに求められる目的や、各種の品質特性におけるテストで確認する特性や品質目標を設定・定義する。

また、テスト対象のシステム構成や開発の背景等の情報から、システム構成に対するテスト対象機能や今回は対象外となる機能、あるいは開発により影響を受ける範囲をテスト範囲の概要として設定する。同時に、テスト範囲のテストを行うのに必要なテスト環境の概要やテスト環境構築のコストについても検討・定義する。

また、この分析工程でテスト対象の開発状況や品質状況から発生する課題やリスクの抽出も実施する。

2.6. テスト要件、方針の定義

分析したテスト要求に基づき、テスト要件や方針を定義する。

まずは実施するテストレベルを検討し、テストレベルに対するテスト内容の概要を設定する。その際、本テストプロジェクト以外で実施するテストレベルが（明示的にではなくとも）存在する場合には、テスト範囲が重複せず全体で効果的なテストとなるよう考慮が必要である。

また、テスト方針としてどのようなテストアプローチを採るかを検討・定義する。テストアプローチの種類としては、リスクベーステストなどの分析的アプローチや、品質特性ベースでテストを行う方法的アプローチ、開発側で採用しているプロセスに同期したテストを行うプロセス準拠アプローチなどがある。

また、テストアプローチとしてテストの狙いを主に不具合検出とするか、正しい動作を保証するための確認とするか、双方を効率良く組み合わせた最善型とするかという品質保証のタイプを定義する必要がある。

2.7. テストアーキテクチャ概要

テストアーキテクチャ設計工程で詳細を検討する内容であっても、テスト要求分析工程で検討を着手し、

事前に実現性の目処を立てておきたいことがある。その内容について概要を定義する。

いわゆるテストをどのように組み立てるかという点で、どのようなテスト（テスト種類やテスト方法やテストの難易度）をどれだけ（テストボリューム）、どのような順序で実施するか、あるいは自動化を検討するかどうか等によって、実現可能性の検討や必要な体制・期間・工数の概要が想定可能となる。その中から、テスト要求を満足し（品質、コスト、期間の要素で）、かつ実現性の高いテストを選択して組み立てる必要がある。

そこで検討した内容をテストアーキテクチャ設計にインプットすることにより、工程間の連携度合いが高まる。

2.8. テスト要件定義の検証

テスト要求分析の結果として作成・定義されたテスト要件をまとめ、不足や整合性について確認を行う。

前述のテスト要求分析の各工程のアウトプットは、単純に時系列に流れるように作成出来るものばかりではなく、後工程で検討した結果によって、前工程のアウトプットを調整する必要が発生する場合がある。可能であれば各作業で随時レビューを行いながら進めることが望ましいが、少なくともテスト要求分析工程の最後には、まとめとレビューを実施する。

< 表 2.1 テスト要求分析 (1/3) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト対象の把握	開発・テスト対象物	開発要件定義書: プロジェクト情報、プロダクト情報 開発アーキテクチャ設計書	テスト対象の用途 システム概要構成、使用環境、動作環境 システム概要構成に対するテスト対象範囲 適合すべき規格、基準等	要件定義書: テスト対象定義
	ターゲットユーザ	開発要件定義書: プロジェクト情報、プロダクト情報	テスト対象のターゲットユーザー(コンシューマ向、業務向、特定ユーザー向、海外向等)	要件定義書: テスト対象定義
	ビジネス戦略	開発要件定義書: プロジェクト情報、プロダクト情報	ビジネス上の狙い(差別化・特化、顧客獲得、利益獲得等)	コンセプト一覧
開発の背景、現状把握	開発の経緯	開発要件定義書: プロジェクト情報、プロダクト情報	テスト対象の生い立ち(新規開発、派生開発・バージョンアップ、OEM、出荷済製品・システム)	要件定義書: テスト対象定義
	開発内容	開発要件定義書: プロジェクト情報、プロダクト情報	今回開発された内容(新規開発、追加、変更機能) 過去プロジェクトから持ち越され修正された不具合	要件定義書: テスト対象定義
	開発プロジェクトの目的	開発要件定義書: プロジェクト情報	開発プロジェクトの位置付け、目的、目標、コンセプト	要件定義書: テスト対象定義 コンセプト一覧
	開発体制	開発要件定義書: プロジェクト情報	開発体制(内製、外部委託)、ステークホルダー、責任の所在、窓口担当	ステークホルダー: 役割の内容と基準
	開発規模	開発要件定義書: プロジェクト情報 開発アーキテクチャ設計書	機能数、ステップ数、開発者数、開発期間	要件定義書: テスト対象定義、開発スケジュール
	開発状況	開発スケジュール・WBS、進捗状況	開発の現在の工程、開発項目の進捗状況	要件定義書: 開発スケジュール
	利用した仕組み	開発要件定義書: プロジェクト情報	開発プロセス、仕組み、ツール、手法など	要件定義書: テスト対象定義
テスト要求の獲得	テスト要求概要	マスターテスト計画書 ヒアリングシート	テストが必要となった背景、理由。テストを行うことにより確認・改善したいこと	要件定義書: テスト要求概要
	テストレベル要求	マスターテスト計画書	マスターテスト(全体テスト)計画が存在する場合、その中でのテストレベル	要件定義書: テストレベル定義
	既存の課題要求等	過去のテスト報告書、課題一覧	過去プロジェクトからの受け渡し事項、課題事項	課題一覧

< 表 2.1 テスト要求分析 (2/3) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テストへの インプットの 確認	テスト資産	既存のテストドキュメント、テスト資産リスト	既存のテスト資産、活用可能なテストケース等の存在 既存のテスト結果、不具合情報	テスト資産リスト
	テストベース	開発要件定義書 開発アーキテクチャ設計書 打合せ議事録等	テスト対象の情報、開発・設計情報、テスト情報	テストベース一覧
テスト要求 の分析	テスト目的	要件定義書:テスト対象定義、テスト要求概要	テストを行う目的、テストを行うことによる狙い・効果	要件定義書:テスト目的
	品質目標	要件定義書:テスト対象定義、テスト要求概要	品質特性の分類と、テストで確認する特性や設定する目標レベル	要件定義書:品質目標
	テスト範囲概要	要件定義書:テスト対象定義、テスト目的、品質目標 テストベース一覧	システム構成に対するテスト対象となる機能・範囲	要件定義書:テスト範囲概要
			開発内容とテスト対象機能の関係(追加、変更による影響範囲) テスト対象外機能・範囲の有無とその理由	
	テスト環境概要	要件定義書:テスト対象定義、テスト範囲概要	テスト範囲のテストに必要なテスト環境の概要	要件定義書:テスト環境概要
			テスト環境構築に必要な資源、期間、コスト(実現可能性、代替方法:シミュレータ環境等の検討)	
テスト要件、方針の 定義	テスト対象の現在状況	要件定義書:開発スケジュール	テスト対象機能の開発状況や品質状況、課題	要件定義書:テスト対象の課題、リスク
	テストレベル定義	要件定義書:テストレベル定義	実施するテストレベルに対するテスト概要 他のチームで実施しているテストレベルと、実施するテストレベルのテスト範囲	要件定義書:テスト概要
	テストアプローチ	要件定義書:品質目標、テスト概要	テストアプローチの種類(分析的:リスクベース、方法論的:品質特性ベース、プロセス準拠、等) 品質保証のタイプ(不具合検出型、動作保証型、最善型)	要件定義書:テスト方針

< 表 2.1 テスト要求分析 (3/3) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テストアーキテクチャ概要	テスト内容の構成	要件定義書:テスト概要、テスト方針	テスト種類やテスト方法(機能要求のテスト方法、非機能要求のテスト方法・テストカテゴリ)、テストの難易度やボリューム 重要度や優先度の考慮	要件定義書:テストの種類・重要度・優先度
	テスト体制、規模	要件定義書:テスト概要、テスト方針、テスト内容の構成	テストの自動化に関する検討	
テスト要件定義の検証	要求分析検証	要件定義書	テストに必要な体制、期間、工数	要件定義書:テスト体制、テスト期間、テスト工数概要
			要求分析成果物のまとめ 要求分析成果物の不足項目の検討、レビュー	要求分析成果物一覧 要求分析成果物検証レポート

3. テストアーキテクチャ設計

テストアーキテクチャ設計工程で必要となる項目を抽出し、表 3.1 にまとめた。

各項目について、考慮すべき要素、検討内容、想定する成果物を整理し、テストアーキテクチャ設計における標準となる行動指標とした。

以下に、テストアーキテクチャ設計工程に沿って項目の説明および、内容、考慮・要素（例）・ポイントを説明する。

3.1. テスト概要の把握

テスト概要の把握工程では、テスト要求分析で作成された要件定義書を基にテスト対象物、ターゲットユーザー、ビジネス戦略等、また開発の背景・現状等の情報を入手する。

3.2. テスト内容の分析

テスト内容の分析工程では、テスト要求分析で作成された要件定義書を基にテストで必要になる情報を分析する。テスト要求内容、ターゲット、ビジネス戦略等からテストポリシー・テストの最終目的を定義する。テスト要求分析結果に沿うテスト種類をリストアップし、テストカテゴリ、テストタイプ型を検討する。テストエリアやテストの深度、テストサイクルについてはテスト対象の品質状態に合わせ検討する。また、テストで想定されるリスクについて検討し、影響範囲、発生確率を検討し対応策を準備する。

3.3. テスト詳細設計・手法の選択

テスト詳細設計・手法の選択工程では、テスト内容の分析工程で分析した結果を基に、テスト対象となる範囲（機能、機器）、テストスケジュール、必要となるテストサイクル（開始基準、終了基準）を定義する。

テストで使用するテストカテゴリ、テストモデル型を選定理由と共に定義する。また選定しなかったテストの理由を併せて記述し、テストの抜け漏れを防止する。

3.4. テストリソースの計画

テストリソースの計画工程では、テストに必要な環境（場所、テスト対象、機器）、人員やそれに伴う必要なトレーニングの計画を行う。

3.5. テストアーキテクチャ設計の検証

テストアーキテクチャ設計の結果として作成されたテスト計画書について、要件定義書との整合性や不足項目についてのレビューを行う。

< 表 3.1 テストアーキテクチャ設計 (1/2) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト概要の把握	テスト対象	要件定義書	開発・テスト対象物、ターゲットユーザー、ビジネス戦略等	テスト計画書 ・テスト目的 ・テストスコープ ・テストスケジュール ・テスト要員体制 ・リスク一覧等
	開発の背景、現状	・テスト目的 ・品質目標	開発の経緯、開発内容、プロジェクトの目的、開発体制、規模、状況	
	テストへの要求	・テストベース一覧 ・テスト範囲	テストへの要求概要、テストレベル要求、既存課題要求等	
	テストへのインプット	・テスト対象の現状 ・テスト対象の完成像 ・テスト環境	テスト要求分析にて検討したテスト資産リスト等	
テスト内容の分析	テスト目的	※不足している場合は、入手時期を明らかにする。	テストポリシー及び最終目的を定義する	
	テストレベル		対象となるテストレベルを定義する 他テストレベル実施部門との役割分担、重複するテスト項目の共有を実施する	
	テストカテゴリ、テストタイプ型		テストカテゴリ、テストタイプ型の素材から適応可能なものを比較検討する ・テストカテゴリ、テストタイプ ・テスト観点・テストモデル構築型 ・機能階層型 ・網羅型・ピンポイント型 ・4つのビュー型(ユーザー観点、仕様観点、設計・実装観点、バグ観点) ・品質特性型	
	テストリスク		テストで想定されるリスクの検討を実施する。抽出されたリスクに対して影響範囲、発生確率を検討し、対応策を準備する。	

< 表 3.1 テストアーキテクチャ設計 (2/2) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト詳細設計・手法の選択	テストスコープ	要件定義書 ・テスト目的 ・品質目標 ・テストベース一覧 ・テスト範囲 ・テスト対象の現状 ・テスト対象の完成像 ・テスト環境 ※不足している場合は、入手時期を明らかにする。	テスト対象範囲、範囲外の機能、機器を定義する	テスト計画書 ・テスト目的 ・テストスコープ ・テストスケジュール ・テスト要員体制 ・リスク一覧等
	テストスケジュール		開発スケジュール、出荷スケジュール等のマイルストーン及びテスト方針に基づきテストスケジュールを決定する。	
	テストカテゴリ、テストタイプ型		QCD分析等を活用し、テストカテゴリ、テストタイプ型を選択する。またそれぞれのテストにおける共通要素、個別要素を整理し、テストの相関関係を整理する	
	テストサイクル		テスト実行サイクル及びその内容をソフトウェアリリース計画との整合性を取り決定する また、リグレッションテストの開始タイミング/基準を検討する	
	テスト開始、終了基準		各テストレベル、サイクルにおける個別の開始基準、継続(中止/再開)基準、終了基準を定義する。また他のテストレベルとの関連性を明確にする	
テストリソースの計画	人員	テスト環境	人員の役割、配置、責任範囲を明確にする。またテストサイクルにおける人の配置を検討する。「概要」「テスト方針」「テスト範囲」で定義した内容と人の配置に整合性(総工数)が取れているか確認する	
	トレーニング計画		人員のスキルに合わせたトレーニングを計画する	
	テスト環境		テストで使用するテスト対象(ケーブル等を含め)を検討する テストで必要となる測定器類(各種アナライザー、治具等)を検討する テスト業務用のPC、LAN環境等の検討、テストスペース、人員分の座席等を検討する	
テストアーキテクチャ設計の検証	テスト計画の検証	要件定義書 テスト計画書	要件定義書に対する、テスト計画の考慮項目の整合性と不足項目の検討、レビュー	テストアーキテクチャ設計成果物一覧 テストアーキテクチャ設計成果物検証レポート

4. テスト詳細設計・実装

テスト詳細設計・実装工程で必要となる項目を抽出し、表 4.1 にまとめた。

標準とする項目は、IT 検証技術者 教育研修シラバス（ミドルレベル）で定義しているテスト工程、工程概要を基に抽出した。

また、各項目について、考慮すべき要素、検討内容、想定する成果物を整理し、検証業務のテスト設計における標準となる行動指標とした。

以下に、テスト設計工程に沿って項目の説明および、内容、考慮・要素（例）・ポイントを説明する。

4.1. 仕様の把握

テスト仕様の把握工程では、テスト計画書を基にテスト設計の実施を行い、テスト観点の漏れを防止するため、実施するテストの種類に応じたテスト対象の仕様（リファレンス）調査と入手を行う。

機能要求、非機能要求、コンプライアンス、運用、テスト環境等の観点から関係する情報を調査、入手し、その内容について、テスト設計で管理可能となるようドキュメント化するとともに、テスト対象の仕様の把握と漏れの防止対策としてテスト設計レビュー等を実施する。

4.2. 仕様管理（トレーサビリティ）

仕様管理（トレーサビリティ）工程では、テスト対象の仕様変更に対する整合性（テスト対象の仕様とテスト設計の整合性）やテストカバレッジ（テスト対象機能やテスト非対象機能の明確化、テスト対象機能の網羅性）を確保するため、テスト対象となる仕様の構成管理を行うための管理対象、方法を検討し、構成管理計画を明確化する。

4.3. 網羅性設計-①（テスト要素抽出）

網羅性設計-①（テスト要素抽出）工程では、機能構成、テストの網羅性を検討し、テスト要素の漏れがないことを確認するため、テスト対象の仕様（リファレンス）をベースに機能樹形図（Function Tree）等を利用して、テスト要素（テスト因子）を抽出する。

また、抽出したテスト要素とテスト対象の仕様（リファレンス）との関係付けを実施して、双方の未対応な項目について、レビューを実施しテスト要素の抽出漏れを防止する。

4.4. 網羅性設計-②（テスト変数抽出）

網羅性設計-②（テスト変数抽出）工程では、テスト要素に関係するコンディション（条件）とコンフィギュレーション（環境）などの入力情報を抽出し、各情報の有効／無効、保障範囲内／範囲外、および網羅性に関しての漏れがないことを確認する。

また、抽出したテストコンディションとテストコンフィギュレーションは、レビューを実施し漏れの防止に努めるとともに、要求元の確認を受けテスト設計内容の承認を得る。

4.5. テストケース-①設計（基本フロー）

テストケース設計-①（基本フロー）工程では、テスト要素と関係するテストコンディション（条件）、テストコンフィギュレーション（環境）について、正常／異常（障害）の条件を抽出し、テストケースの作成レベルを決定するため、網羅性設計で抽出したテスト要素とテストコンディション（条件）／テストコンフィギュレーション（環境）を組み合わせた基本フローのテストケースを作成する。

テストケースの作成レベルを決定し、テスト実行者のスキルや状況（工数や納期）を把握するとともに、第三者にテストケースの目的が理解できるようにテストケースを作成する。

4.6. テストケース設計-②（組み合わせ）

テストケース設計-②（組み合わせ）工程では、テスト要素と関係するテストコンディション（条件）、テストコンフィギュレーション（環境）について、有効なコンフリクト（衝突）の条件、競合するリソース等、有効なコンカレント（競合）の条件を抽出し、それらを組み合わせたテストケースを作成する。

テストケースの作成レベルを決定し、テスト実行者のスキルや状況（工数や納期）を把握するとともに、第三者にテストケースの目的が理解できるようにテストケースを作成する。

4.7. テストケース設計-③（期待値設計）

テストケース設計-③（期待値設計）工程では、期待値の有効範囲、期待値以外の出力結果の確認方法を決定するため、テストケース設計時に設計した期待値と、その確認方法を決定し、テストケースの文書（テスト手順書、チェックリストなど）に記述する。

4.8. モニタ設計

モニタ設計工程では、テスト実行結果（期待値確認）を確認するため、専用ツール、リソース（ネットワーク、ネットワークプロトコル、H/W資源など）を監視するツールを利用したモニタ監視の方法を決定する。

4.9. テスト環境設計

テスト環境設計工程では、テスト設計を反映したテスト環境を設計するため、テスト（設計）支援ツールの導入目的、効果、導入手順（アプローチ）、およびテスト設計に基づく、テスト対象のテスト実行を実現する環境を決定する。

4.10. プロジェクト管理

テスト設計の工程全般でのプロジェクト管理のポイントは以下である。

- ・チーム管理項目：スコープ管理、スケジュール管理、コスト管理、品質管理、要員管理、コミュニケーション管理、リスク管理、調達管理
- ・個人における作業管理方法：役割の確認、テスト設計の生産性の管理、テスト設計管理、不明点管理、報告・連絡・相談・質問

< 表 4.1 テスト詳細設計・実装 (1/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
仕様の把握	テスト対象の仕様(リファレンス)の調査と入手		テストの目的やテストアプローチを確認して、テスト設計するテストの種類に必要なテスト対象のドキュメントや情報を入手する	テスト仕様 ・テスト目的 ・テストアプローチ(手順、方針、環境) ・テスト対象範囲 過去の類似テストに関する資料 (テスト設計書、テスト報告書)
			テスト設計で必要となる(利用する)ドキュメントの一覧を作成する	
			テスト対象の機能要求を調査しドキュメントを入手する	
			テスト対象に関わるコンプライアンス(法規、標準(業界標準なども含む)、規格等)に係る情報を調査し入手する	
	仕様の把握と管理		テスト対象の非機能要求を調査する	
	テスト対象の仕様の把握と漏れの防止対策		テスト対象が運用(利用)される環境やリテラシー情報を調査する	
			過去のテストドキュメント(テスト仕様、テスト設計書(テストケースを含む)、テスト報告書等)を入手し調査する	
			調査した仕様をテスト設計文書として記録する	
			ブレンストーミングを実施して、仕様の把握とテスト観点の漏れを防止する	
仕様管理(トレーサビリティ)	仕様の構成管理	テスト仕様 ・テスト目的 ・テストアプローチ(手順、方針、環境) ・テスト対象範囲	テスト構成管理を計画する	テストに関する構成管理対象 ・管理対象範囲 ・管理単位(リリース単
網羅性設計-①(テスト要素抽出)	テスト要素の抽出	過去の類似テストに関する資料 (テスト設計書、テスト報告書)	テスト対象の仕様(リファレンス)をベースにテスト要素を抽出し、テスト要素の一覧を作成する	テスト要素 ・テスト機能構成 ・テスト要素 ・テスト条件
			機能樹形図(Function Tree)を利用して、機能構成を構造化する	
			機能樹形図(Function Tree)の最下層が、最小単位のテスト要素と一意になることを確認する	
			テスト要素にコンディション(条件)が含まれていないことを確認する コンディション(条件)にあたる場合は、一覧や表形式(マトリクス)にまとめて、テスト変数(テスト水準)とする	
	テストカバレッジ設計		機能樹形図(Function Tree)で別系統の同一機能を確認する	
			抽出したテスト要素の一覧とテスト対象の仕様(リファレンス)との関係付けを実施して、双方の未対応な項目を洗い出す	
	レビュー&承認		抽出したテスト要素は、レビューを実施して漏れの防止に努める	
			テスト詳細設計の結果(テスト要素、コンディション、コンフィギュレーションなど)は、要求元の確認を受けテスト設計内容の承認を得る	

< 表 4.1 テスト詳細設計・実装 (2/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
網羅性設計-② (テスト変数抽出)	テストコンディション(条件)の抽出	テスト仕様 ・テスト目的 ・テストアプローチ(手順、方針、環境) ・テスト対象範囲	テスト要素に与える入力情報として、テストコンディション(条件)を抽出する テスト要素の入力(起動される場合が多い)条件として、イベント(割り込み、プロセス等)を洗い出し、一覧表やイベント関係図を作成する	テスト網羅性 ・テスト条件の関連付け ・テスト条件の組合せ
	テストコンフィギュレーション(環境)の抽出	過去の類似テストに関する資料(テスト設計書、テスト報告書)	テスト対象を構成するH/Wコンフィギュレーション(環境)を抽出する テスト対象を構成するS/Wコンフィギュレーション(環境)を抽出する	
	テストカバレッジ設計		抽出したテストコンディション(条件)と仕様(リファレンス)との関係付けを実施して、双方の未対応な項目を洗い出す 抽出したテストコンフィギュレーション(環境)と仕様(リファレンス)との関係付けを実施して、双方の未対応な項目を洗い出す テストコンフィギュレーション(環境)での保証範囲外を検討し、抽出する 直交表(実験計画法)を用いて、コンディションの組み合わせの網羅性を検討する	
	レビュー&承認		抽出したテスト変数(コンディション、コンフィギュレーション)は、レビューを実施する テスト詳細設計の結果(テスト要素、コンディション、コンフィギュレーションなど)は、要求元の確認を受けテスト設計内容の承認を得る	
テストケース設計-① (基本フロー)	基本フローを構成する	テスト網羅性 ・テスト条件の関連付け ・テスト条件の組合せ	テスト要素と関係するテストコンディション(条件)を組み合わせ、テストケースを作成する テスト要素と関係するテストコンフィギュレーション(環境)を組み合わせ、テストケースを作成する テスト要素と例外のテストコンディション(条件)を組み合わせ、テストケースを作成する テスト要素と異常(障害)のテストコンディション(条件)を組み合わせ、テストケースを作成する	テストケース ・環境の組合せ ・条件を組合せ
	その他		テストケースの作成レベルを決定し、テスト実行者のスキルや状況(工数や納期)を把握してテストケースを作成する テストケース作成者以外の第三者にテストケースの目的が理解できるようにテストケースを作成する	

< 表 4.1 テスト詳細設計・実装 (3/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テストケース設計-② (組み合わせ)	組み合わせの検討	テストケース ・環境の組合せ ・条件を組合せ	コンディション(条件)やイベント、テスト要素など、有効なコンフリクト(衝突)の条件を抽出する 競合するリソース等、有効なコンカレント(競合)の条件を抽出する テストコンディション(条件)／テストコンフィギュレーション(環境)を抽出し、関連を文書(図)にまとめる テストケースの作成レベルを決定し、テスト実行者のスキルや状況(工数や納期)を把握してテストケースを作成する テストケース作成者以外の第三者にテストケースの目的が理解できるようにテストケースを作成する	テストケース ・実行レベルの決定 ・実行者のスキル、状況に応じたテスト条件
	その他			
テストケース設計-③ (期待値設計)	有効な期待値の設計	テストケース ・環境の組合せ ・条件を組合せ	テストケースの対象となるテスト要素の期待値および、期待値以外のテストケースを作成する リソース状況やプロトコルフレームのように動的に結果が変化する期待値を設計する場合には、期待値の部分にマスク(変化するフレームなど)や、期待値の有効範囲を設定する テストケースの文書(テスト手順書、チェックリストなど)に期待値を記述できない場合には、期待値と比較できるリファレンス(テスト基準)を用意する 自動テストツールなどの利用を前提とした場合には、テストケースのスクリプトに対応するコンペアデータを作成する	テスト結果(期待値) ・テスト基準 ・有効範囲 ・期待値以外(異常値) ・テストデータ
	期待値のリファレンスの準備			
モニタ設計	ツールによるモニタ設計	テストケース ・環境の組合せ ・条件を組合せ	専用ツールによるモニタ設計(または、ツールの調達)を行う リソース(ネットワーク、ネットワークプロトコル、H/W資源など)を監視するツールを利用したモニタ監視の方法を決定する 専用ツールによる出力分析ツール設計(または、ツールの調達)を行う	テストツール ・モニタリングツール
テスト環境設計	テスト(設計)支援ツールの検討	テストケース ・環境の組合せ ・条件を組合せ	テスト(設計)支援ツールの導入目的、効果、導入手順(アプローチ)を検討する	テスト環境 ・支援ツール
	テスト環境の設計		テスト設計に基づき、テスト対象のテスト実行を実現する環境を設計する	

< 表 4.1 テスト詳細設計・実装 (4/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
プロジェクト 管理(チーム 管理)	スコープ管理	テスト仕様 ・テスト目的 ・テストアプローチ(手 順、方針、環境) ・テスト対象範囲	テストの目的に合致したテストの種類 の設計およびテスト対象が設計要求範 囲に差異が無いこと、およびテストカバ レッジや目標件数(妥当性を含む)の管 理方法を明確にする	テスト設計書 ・テスト目的 ・テストアプローチ(手 順、方針、環境) ・テスト対象範囲 ・テスト制限事項 ・テスト環境(H/W、 S/W、ツール) ・テストケース(項目) ・テストスケジュール ・テスト体制(要員、ス キル) ・管理手法
	スケジュール管 理	過去の類似テストに 関する資料 (テスト設計書、テスト 報告書)	テスト設計のスケジュールに影響する 要因(リスク管理とも関係)を把握し、ス ケジュールを維持する方法(進捗管 理、定例会議、日報提出)を明確にす る	
	コスト管理	テストに関する構成管 理対象	テスト設計の見積りと実績の差異分析 方法を明確にする	
	品質管理	・管理対象範囲 ・管理単位(リリース 単位、時間単位 等) テストケース	テスト対象の仕様変更や仕様の不明 (曖昧、未定義など)に対応し、仕様と テスト設計のトレーサビリティ(テストの 構成管理)や不明な部分の仕様確認、 状況の管理方法を明確にする	
	要員管理	・実行レベルの決定 ・実行者のスキル、状 況に応じたテスト条件 テスト結果(期待値)	テスト設計の要員のスキル(テスト対象 の理解度)習得度の把握と作業への取 り組みに関する阻害要素(モチベーショ ンの低下など)点を考慮し、要員管理 方法を明確にする	
	コミュニケー ション管理	・テスト基準 ・有効範囲 ・期待値以外(異常 値)	仕様変更管理や質問表などのツール やチーム内・外の情報交換を阻害する 要素と対策管理方法を明確にする	
	リスク管理	・テストデータ テストツール ・モニタリングツール	①仕様不明等(曖昧、未定義など)の 問題の管理方法を明確にする ②想定外のリスクが発生したときの対 応(トラブルの解析や対応する担当者 の事前配置など)手順の立案方法を明 確にする	
	調達管理	テスト環境 ・支援ツール	テスト設計で調達したドキュメント管理 (機密、変更など)方法を明確にする	
プロジェクト 管理(個人 管理)	役割の確認		チーム内で割り当てられる役割を明確 にする	
	テスト設計の生 産性の管理		テストスケジュールに深く関係するテス ト設計工数の管理方法を明確にする	
	テスト設計管理		テスト設計エリア、件数などのテスト目 標やテスト設計スケジュール(個人のス ケジュール)などの管理方法を明確に する	
	不明点管理		テスト対象の不明点の確認方法(コミュ ニケーション)や問い合わせをしている 状況の管理方法を明確にする	
	報告・連絡・相 談・質問		テスト設計における、適切な報告とその 方法を明確にする	

5. テスト実行

テスト実行工程で必要となる項目を抽出し、表 5.1 にまとめた。

標準とする項目は、IT 検証技術者 教育研修シラバス（エントリーレベル）で定義されているテスト工程、工程概要を基に抽出した。

また、各項目について、考慮すべき要素、検討内容、想定する成果物を整理し、検証業務のテスト実行における標準となる行動指標とした。

以下に、テスト実行工程に沿って項目の説明および、内容、考慮・要素（例）・ポイントを説明する。

5.1. テスト実行計画

テスト実行計画工程では、テスト計画に基づいて設計されたテスト仕様とテストケースを用いて、実施するテスト実行の計画を立案する。

テスト実行計画工程で実施する項目を以下に列挙する。

- ・テスト対象の知識の習得
- ・テストの目標設定
- ・テスト実行スケジュール作成
- ・テスト組織図の作成
- ・テスト実行の要員計画作成
- ・コミュニケーション手段の計画
- ・テスト実行の見積り（試算）
- ・テスト実行時のリスク検討
- ・リスク一覧（表）と対策表の作成
- ・調達すべきテスト環境（必要機材、リソース、作業場所）計画

5.2. テスト環境準備

テスト環境準備工程では、テスト実行計画をもとにテスト実行に必要な環境の調達および設定を実施して、作業用環境、テスト環境、テスト対象物、データ等および、テスト実行に必要な連絡表、質問表、不具合報告書等のツール類を準備し、テスト実行で利用可能な状態とする。

テスト環境準備工程で実施する項目を以下に列挙する。

- ・作業用環境の準備
- ・テスト環境の準備
- ・テスト対象の準備
- ・技術者のアサイン
- ・動作実績の確認
- ・用語集の準備
- ・コミュニケーションツール（連絡表、仕様変更、質問表など）
- ・テスト実行組織図の作成
- ・テスト実行管理表の準備
- ・チェックリスト作成

5.3. テストケース準備

テストケース準備工程では、テスト対象となるものの不明点を整理し、テスト実行計画をもとにテストケースおよびテストツールを準備する。

テストケース準備工程で実施する項目を以下に列挙する。

- ・テストケース準備
- ・テスト対象の不明点整理
- ・確認リファレンスの準備
- ・チェックリスト作成

5.4. テスト実行

テスト実行工程では、テスト実行計画されたテストケースの実行やチェックリストで確認を行い、期待値に対して合否判定を行う。

テスト実行工程で実施する項目を以下に列挙する。

- ・テスト進捗の把握
- ・事象の発生（遅れ）
- ・事象の発生（インシデント）
- ・不明点の確認
- ・質問表の状況（ステータス）管理
- ・リリースバージョンの確認
- ・テストリソース（機材、要員）の管理
- ・テストケースの管理
- ・定例報告会への参加
- ・翌日の準備

5.5. テスト実行記録

テスト実行記録工程では、テスト実行の結果（合否結果、出力のテストエビデンス）やステータスおよび実行データについて管理シートを用いて記録する。

テスト実行記録工程で実施する項目を以下に列挙する。

- ・日次のテスト実行結果の記録
- ・テスト実行結果の中間報告
- ・不具合関連情報の日次記録
- ・優先度・重要度の管理

5.6. 不具合報告

不具合報告工程では、テスト手順書や仕様書などの期待値と異なる事象を検出し、リファレンスや仕様を確認した結果、不具合と認識された事象の報告と適切に修正されたことを不具合のステータスで管理する。

不具合報告工程で実施する項目を以下に列挙する。

- ・不具合報告書の作成
- ・プロジェクト固有の不具合報告書のルールを確認する
- ・不具合報告書のステータス（状況）管理
- ・修正確認の管理

- ・不具合報告書のステータス（状況）の報告

5.7. テスト実行報告

テスト実行報告工程では、テスト実行工程の結果をまとめ、工程内の事象に関する情報をトレース（追跡）できるように報告書にまとめる。また、テスト実行結果の他プロジェクトへ反映や次回テストへの反映を実現する方法を管理する。

5.8. プロジェクト管理

テスト実行の工程全般でのプロジェクト管理のポイントは以下である。

- ・チーム管理項目：テスト目標管理、スケジュール管理、テスト工数管理、品質管理、要員管理、コミュニケーション管理、リスク管理、調達管理
- ・個人における作業管理方法：役割の確認、テスト実行の生産性の管理、テスト実行管理、報告・連絡・相談・質問

< 表 5.1 テスト実行 (1/5) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト実行計画	テスト対象の知識の習得	テスト設計書 ・テスト目的 ・テストアプローチ(手順、方針、環境) ・テスト対象範囲 ・テスト制限事項 ・テスト環境(H/W、S/W、ツール)	マニュアル、(開発/テスト)仕様書、テストケース(テスト手順書など)、実機などからテスト対象の仕様や操作方法を確認する	テスト詳細工程 ・日割り工程表 ・進捗管理表 ・不具合管理、報告表
	テストの目標設定	・テストケース(項目) ・テストスケジュール ・テスト体制(要員、スキル) ・管理手法	旧バージョンが存在する場合、そのバージョンのテスト仕様を入手し差分を把握する 競合製品が存在する場合は、仕様の調査・分析を実施する	
	テスト実行スケジュール作成		テスト実行エリア、実行テストケースの抽出、優先順位の定義(優先順位の設定方法)などの具体的なテスト目標を設定する	
	テスト組織図の作成		テストサイクルやテスト種別、作業の振り分けなどの項目を抽出する	
	テスト実行の要員計画作成		テストチーム体制、開発チーム体制やプロジェクトの責任者を明確にする	
	コミュニケーション手段の計画		役割やアサインの期間、担当するテスト実行エリアを検討(考慮)する	
	テスト実行の見積り(試算)		仕様変更、レビュー、緊急報告や定例会議など具体的な手段を検討し、共有する	
	テスト実行時のリスク検討		テスト実行件数、不具合件数等によりテスト工数を算出する	
	リスク一覧(表)と対策表の作成		テスト実行を阻害する“要員のトラブル”や“テスト対象の品質”や“ブロックバグの存在”、“機器の不足や故障”、“コミュニケーションのミス”など具体的な要因を検討し、抽出する	
	調達すべきテスト環境(必要機材、リソース、作業場所)その他		過去に発生したリスク等を考慮し、ノウハウの蓄積方法を明確にするとともに、リスクに対する対応管理方法を明確にする	
			調達計画の立案に必要な項目(調達リスト、日付(期間)、担当者(管理者)など)を整理する	
			レビューの計画とレビューメンバーのアサインを実施する	
			二次派生物(コピーやメモ)などの廃棄方法を明確に規定しておく	

< 表 5.1 テスト実行 (2/5) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト環境準備	作業用環境の準備	テスト設計書 ・テスト目的 ・テストアプローチ(手順、方針、環境)	作業用PC やインフラ環境(mail アカウ ントや作業用ストレージ、およびテスト 管理用のツールなど)など、作業用環 境を準備する	テスト環境 ・H/W環境 ・M/W環境 ・コミュニケーション ツール(連絡表、仕様 変更届け、質問表等)
	テスト環境の準備	・テスト対象範囲 ・テスト制限事項 ・テスト環境(H/W、 S/W、ツール) ・テストケース(項 目)	テストツールやテストベンチ、テスト用 のインフラ環境やテストのログを収集す るツールやそれぞれのツールの操作説 明書(マニュアル)など、テスト環境を準 備する	
	テスト対象の準備	・テストスケジュール	必要なテスト対象と、十分なオプション (付属品)やサプライ用品を準備する	
	技術者のアサイン	・テスト体制(要員、 スキル) ・管理手法	テスト環境の設置や設定をする(必要 に応じてテスト環境を熟知している)技 術者のアサインを実施する	
	動作実績の確認		テスト実行に必要なリソースの動作実 績の確認をする	
	用語集の準備		専門知識に関する用語集(テスト対象 やプロジェクトの暗黙値)を準備する	
	コミュニケーションツール (連絡表、仕様 変更、質問表)		コミュニケーションツール(連絡表、仕様 変更届け、質問表等)と運用ルールを 規定する	
	テスト実行組織 図の作成		テスト実行にあたる組織図(開発担当 者、ステークホルダー、チーム体制等) を作成し、責任分解点を整理する	
テストケース準備	テスト実行管理 表の準備		機材管理、進捗管理、質問表管理、仕 様変更管理、不具合ステータス管理な どテスト実行で必要な管理項目を整理 し、管理手法を明確にする	テストデータ ・期待値(上下限值)
	チェックリスト作 成		テスト環境の準備物を一覧にし、漏れ を防止する	
	テストケース準備	テスト設計書 ・テスト目的 ・テストアプローチ (手順、方針、環境) ・テスト対象範囲 ・テスト制限事項	テスト手順書、テストデータ、チェック リストおよびテスト実行の結果を記録する テスト実行結果報告書(フォーマットや ツール)や不具合報告書(フォーマット やツール)を整備する	
	テスト対象の不明点整理	・テスト環境(H/W、 S/W、ツール) ・テストケース(項 目)	テスト対象に関する質問などをテスト実 行前に質問表で(テスト設計者または 開発担当者に)確認し、明確にする	
	確認リファレン スの準備	・テストスケジュール	仕様書やUI(画面)遷移図などのテスト 結果のリファレンスを準備する	
	チェックリスト作 成	・テスト体制(要員、 スキル) ・管理手法	テスト環境の準備物を一覧にし、漏れ を防止する	

< 表 5.1 テスト実行 (3/5) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト実行	テスト進捗の把握	テスト設計書 ・テスト目的 ・テストアプローチ (手順、方針、環境) ・テスト対象範囲 ・テスト制限事項 ・テスト環境 (H/W、S/W、ツール) ・テストケース (項目) ・テストスケジュール ・テスト体制 (要員、スキル) ・管理手法	割り当てられたテストケースに基づきテスト項目数をいつまでに実施するかを確認する手法を明確にする	テスト実行結果 ・正誤 ・不具合 ・進捗状況
	事象の発生(遅れ)		テストケース実行中に予定通りに完了しないことが判明した場合に、随時テスト実行管理者あるいはテストプロジェクト管理者に報告する	
	事象の発生(インシデント)		インシデントが発生した場合は、事象のテストエビデンス(記録)・現象の確認・操作手順の確認を直ちに行う	
	不明点の確認		質問表で確認する前に、テストケースに間違いがないか、仕様書に該当する記述が無いかを確認する	
	質問表の状況(ステータス)管理		不具合が多すぎてテストが進まない、機材が足りない、テスト手順が曖昧すぎて進行に支障をきたすなどの阻害要因を管理者へ報告する	
	リリースバージョンの確認		テスト実行の記録やインシデントの報告、テスト実行の制限などを確認する為に、リリースのバージョンと改修項目および制限事項を確認する	
	テストリソース(機材、要員)の管理		テスト対象や環境の故障・不足、要員の勤怠(傷病欠による欠員)などの状況を確認し必要に応じた対策案を検討する	
	テストケースの管理		テストケースの間違いを発見した場合の対応方法を検討し、管理者へ報告する	
	定例報告会への参加		必要に応じて朝礼、終礼、週間進捗報告会などに参加し、役割に応じた報告を行う	
	翌日の準備		原則として帰宅前に翌日のテスト実行の準備(テストケースの確認、テストデータや環境の準備など)を実施する	
テスト実行記録	日次のテスト実行結果の記録	テスト設計書 ・テスト目的 ・テストアプローチ (手順、方針、環境) ・テスト対象範囲 ・テスト制限事項 ・テスト環境 (H/W、S/W、ツール) ・テストケース (項目) ・テストスケジュール ・テスト体制 (要員、スキル) ・管理手法	テスト実行の記録を必ず日次で実施する ・実行したテストケースと数 ・テストケース単位の実行時間 ・テスト実行結果(OK/NG)、NGの場合の不具合番号 ・テストエリア ・テスト未実行数 ・実行不可能なテストケースとその理由など	テスト実行結果 ・実行したテストケースと数 ・テストケース単位の実行時間 ・テスト実行結果(OK/NG)、NGの場合の不具合番号 ・テストエリア ・テスト未実行数 ・実行不可能なテストケースとその理由など
	テスト実行結果の中間報告		テスト実行に影響する日次で解決すべき問題点と、週次報告書や中間報告書で報告する	
	不具合関連情報の日次記録		不具合関連の情報として、日々の変化を記録する ・不具合の総数 ・日次で新規に発見した数 ・修正対応数 ・修正確認対応数など	
	優先度・重要度の管理		不具合状況や優先度など、一度ランク付けされた優先度・重要度もテスト実行の状況により変更する	

< 表 5.1 テスト実行 (4/5) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
不具合報告	不具合報告書の作成	テスト実行結果 ・実行したテストケースと数 ・テストケース単位の実行時間 ・テスト実行結果(OK/NG)、NGの場合の不具合番号 ・テストエリア ・テスト未実行数 ・実行不可能なテストケースとその理由など	適切な言葉を使い不具合報告書を作成していく ①事象の発見 ②障害の特定 ③発生条件の特定 ④再現の手順化 ⑤不具合の評価 ⑥不具合報告書の作成: 不具合報告書の作成のポイントを理解する	不具合報告書の作成 ・事象の発見 ・障害の特定 ・発生条件の特定 ・再現の手順化 ・不具合の評価
	プロジェクト固有の不具合報告書のルールを確認する		プロジェクト固有のルールにあわせた不具合報告の管理を実施する	
	不具合報告書のステータス(状況)管理		優先度、重要度および抽出した時期などによるテスト実行への影響度などを考慮した不具合報告書のステータス(状況)を動的に管理する ①テスト実行に影響を与える優先度の管理 ②基本的なステータス(状況)の管理 ③不具合の傾向分析による不具合の分類	
	修正確認の管理		不具合の状況を常時確認する ①不具合が修正対応されるリリース(リリース番号、リリース日)を確認する ②テスト実行工程の進捗状況に合わせて、優先度を変更する	
	不具合報告書のステータス(状況)の報告		日次、週次など、状況にあわせた不具合のステータス(状況)を報告する	
テスト実行報告	作業記録の報告	テスト実行結果 ・実行したテストケースと数 ・テストケース単位の実行時間 ・テスト実行結果(OK/NG)、NGの場合の不具合番号 ・テストエリア ・テスト未実行数 ・実行不可能なテストケースとその理由など	テスト実行報告書は、テスト実行の記録のトレーサビリティ(追跡)を考慮して作成する	テスト実行報告書
			テスト実行中に発生したトラブルや問題点を、発生から対応までの状況を時系列で報告する	
			他のプロジェクトへ反映や次回のテストへの反映を実現する方法を管理する	

< 表 5.1 テスト実行 (5/5) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
プロジェクト 管理(チーム 管理)	テスト目標管理	テスト設計書 ・テスト目的	テスト実行の目標と完了基準の関係と 管理する	テスト実行結果
	スケジュール管理	・テストアプローチ (手順、方針、環境)	テスト実行のスケジュールの管理する	
	テスト工数管理	・テスト対象範囲	テストの実行時間(予測と実績、および 再分析)の管理する	
	テスト実行の品 質管理	・テスト制限事項	テスト実行に要求される品質の要素を 管理する	
	要員管理	・テスト環境(H/W、 S/W、ツール)	要員のスキル・作業状況を管理する	
	コミュニケーション管理	・テストケース(項目)	テスト実行中での仕様変更管理や質問 表などのツールやチーム内・外の情報 交換等のコミュニケーションの運用状 況を管理する	
	リスク管理	・テストスケジュール ・テスト体制(要員、 スキル) ・管理手法	テスト実行中に発生する想定・想定外 のリスクの把握と対策方法を管理する ①過去のテストで発生した問題やプロ ジェクト固有の想定される問題の予測 と発生の予兆の把握、発生時の対策 ②想定外のリスクが発生したときの対 応(問題の解析や対策の担当者など) 手順	
プロジェクト 管理(個人 管理)	調達管理		テスト実行で調達したリソースの管理 (故障、紛失、不足)方法を明確にする	
	役割の確認		チーム内で割り当てられる役割を明確 にする	
	テスト実行の生 産性の管理		テストスケジュールに深く関係するテス ト実行時間の管理方法を明確にする	
	テスト実行管理		テストエリア、テスト実行予定件数な どのテスト目標やテストスケジュール(個 人のスケジュール)などの管理方法を 明確にする	
	報告・連絡・相 談・質問		テスト実行における、適切な報告とその 方法を明確にする	

6. テスト評価／報告

検証業務におけるテスト評価工程で必要となる項目を抽出し、表 6.1 にまとめた。

標準とする項目は、IT 検証技術者 教育研修シラバス (High-Level Syllabus) で定義しているテスト評価工程を基に抽出している。

6.1. テスト報告書を準備する

テストが終了するまでに、テスト評価実施に向けた準備を始める。

テスト報告書作成に必要な以下の情報。

- ・テスト計画時の工程・工数と実績のデータ
また、それを比較して大きく異なっている部分の原因を分析するための、以下の情報を収集して分析を開始する。
- ・各アウトプットのレビュー結果で完了していない項目と理由
- ・前回のプロジェクトまたは他のプロジェクトの情報

6.2. テスト活動全般の情報を収集する

今までのテスト活動で作成した情報を収集し、テストが終了した時点で最終版の情報に差し替える。

テスト要求分析/テストアーキテクチャ設計段階で作成した以下のドキュメント類。

- ・テスト計画書
- ・テスト方針
テスト詳細設計段階で作成した以下のドキュメント類。
- ・テスト設計指針
- ・テスト実行スケジュール
テスト実行段階で作成した以下のドキュメント類。
- ・進捗報告書
- 他、適宜必要な情報を収集する。

6.3. テスト活動を分析する

収集した情報を分析する。

分析の基準とするテスト品質指標、過去や類似のプロジェクトの結果と照らし合わせ、テスト実行の活動に問題点が無かったか、また、今後のノウハウ（資産）とすることができるとような改善策の実績が有るか、などを工程ごとに詳細に分析する。

分析した結果についても定型フォーマットへの記述を行うことで、次回プロジェクトや他のプロジェクトでも参照しやすい形で保管する。

6.4. 改善項目の達成率を評価する

前回のプロジェクトで分析結果として提示した改善項目が存在する場合、または当初目標とした計全項目に対してどれだけの項目に取り入れ、その効果がどの程度あったかなどを可能な限り数値化して評価する。

以下のそれぞれについて分析する。

- ・テスト要求分析活動
- ・テストアーキテクチャ設計活動

- ・テスト詳細設計・実装活動
- ・テスト実行活動

6.5. 分析結果と評価指標を比較する

評価指標と分析結果を比較することで、活動が理想に近い活動となったかどうかを客観的に見ることができる。

乖離していた場合は、その障壁となった原因を分析することで今後の活動を改善する第一歩ととらえる。

6.6. 改善点を抽出する

ここでは、チームのメンバーとともに反省会や洗い出し作業を行うことで共通の意識を持たせ、より浸透した形で次回の活動につなげることを目的とする。

6.7. 改善策を検討する

前述の改善点の抽出に続けて、改善策の検討を行う。

チームのメンバーと共に改善すべき点の再発防止策を議論する。根拠の無い発想はアイデアとして一旦寝かせておき、具体的な取り組みに結びつくものを、実際のプロジェクトで取組む企画としてまとめるのが理想的である。

取組みには他部署の協力を必要とするケースもあるため、責任者会議やプロジェクト改善委員会の設置など、組織を全体で管理する風土作りも必要となる。

6.8. テスト評価報告書を作成する

テスト構造設計者は、各評価結果のトレーサビリティを確保出来るようにまとめ、文書化する。

6.9. テスト評価報告書を開示する

部署内あるいは社内での公開が主であるが、チーム以外のメンバーや他の部署のメンバーでも閲覧・参照が可能な形で保管することで、チームだけでなくグループや会社の財産とすることができる。

<表 6.1 テスト評価／報告 (1/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト報告書を準備する	終了基準クリア判定材料の収集とまとめ ・テスト結果の記録の収集とまとめ	・テスト報告書 ・スケジュール, 工数の予実データ ・各アウトプットのレビュー結果 ・参考にしてできる他プロジェクトの情報 ・前回プロジェクトの情報 ・マスターテスト計画書	・テストのサイクルごとの目標件数、適正不具合数、終了基準をあらかじめ決めておく(開発ステップ数、言語、分野などに基づく科学的指標による裏付けがあれば、尚良い) ・テスト報告書に「未記入」「未実施」「NG」のまま放置された項目が無いこと 「中止」の項目には妥当な理由が示されていること。 ・スケジュールや工数において正常な作業を阻害した要因を把握しておく	・進捗(スケジュール), 工数予実データの一覧 ・各アウトプットのレビュー結果の一覧 ・参考にしてできるデータの一覧 ・前回からの引継ぎ事項(前回の改善事項) ・承認されたテスト計画書
	テスト環境の整理・保存 ・テスト環境再構築の情報保管	テスト計画書(テスト環境)	第三者へ引き継ぐことや、時間を置いた後に市場障害が発生した場合に再テストすることを想定して環境構築手順書も整備する	インストール手順書 保守マニュアル
テスト活動全般の情報を収集する	テスト計画書(実績値)の分析 ・計画との差異	・進捗(スケジュール), 工数予実データの一覧 ・各アウトプットのレビュー結果の一覧 ・過去や類似のプロジェクトなど参考にしてできるデータ ・前回からの引継ぎ(前回の改善事項) ・承認されたテスト詳細設計の指針・原則が記載されたレベルテスト計画書	・計画値より少なくない場合、網羅率は妥当か ・計画値より大幅に多い場合、要求分析・テスト設計・仕様変更などの問題点を明らかにする	評価指標 (分析のベースとなるもの・予定・実績・過去の経験、顧客の評価についての一覧や、それらをまとめたもの。) ※実績については、全社で統計を作成することで業種別分野別の指標を構築できる
テスト活動を分析する	テスト活動のまとめと分析 ・過去の(類似)プロジェクトの結果との比較		・テスト構造設計者は、これまでに習得した「テスト活動における暗黙値・ノウハウ」などを活用して今回の各テスト活動について分析を行う。 ・その分析結果をステークホルダーへ報告するために、定量的・定性的な内容にして記録する	
	実施結果分析 ・テスト設計とのトレーサビリティ再確認	・承認されたテスト詳細設計の指針・原則が記載されたレベルテスト計画書 ・テスト報告書 ・開発仕様書のレビュー結果 ・テスト設計書 ・テストケース(テスト実行を実現するためのドキュメント群)	・テスト設計で示された内容を網羅しているか - 実施できなかったテストの件数と理由は妥当か - 完了できなかったテストの件数と理由は妥当か ・テスト結果は期待する値になっているか - 処理速度などのユーザビリティ項目も検討すること ・スケジュールや工数において正常な作業を阻害した要因を把握しておく	・テスト実行活動分析結果

<表 6.1 テスト評価／報告 (2/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト活動を分析する	実施状況分析 ・スケジュールの妥当性チェック	<ul style="list-style-type: none"> 承認されたテスト詳細設計の指針・原則が記載されたレベルテスト計画書(上記のリファレンス) テスト範囲の定義 自動化・再利用化の定義 テスト目的(目標値を含むテストスコープ) テストスケジュール テストタスク テスト要員体制 テスト範囲(テスト実施範囲と対象テスト工程) リスク一覧表 調達基準と技術要素トラ 	<ul style="list-style-type: none"> 計画より進んでいる場合、改善ポイントは何か。網羅率は問題ないか 計画より遅れた場合、問題は何か、改善策はあるか 	・テストアーキテクチャ設計活動分析結果
	・スコープとのバランスチェック		<ul style="list-style-type: none"> テストにかかった総工数は計画通りか 異常に工数を使っている項目の原因は何か 	同上
	テスト設計分析: 不具合、品質 ・品質指標との差異の妥当性	<ul style="list-style-type: none"> 承認されたテスト詳細設計の指針・原則が記載されたレベルテスト計画書 テスト報告書 テスト実行スケジュール(実績) テストケース(テスト実行を実現するためのドキュメント群) 各アウトプットのレビュー結果 スケジュールの遅延, 工数の予実差異 	<ul style="list-style-type: none"> テスト実施内容・品質が承認レベルに達しているか <ul style="list-style-type: none"> 欠陥の検出時期、傾向、インパクト 妨げの要因 	・テスト設計活動分析結果
	テスト設計分析: 網羅性、カバレッジ ・品質指標との差異の妥当性	<ul style="list-style-type: none"> マスターテスト計画書 テスト要求分析成果物(上記のリファレンス) テスト目的、テスト目標の詳細項目と、それぞれの構造を可視化したもの(確認された)テストベース一覧 テスト範囲の分析結果レポート テスト対象が持つ問題・課題一覧 テストの目的や目標との整合性一覧 (テスト環境を用意する開発への)テスト環境要求一覧 	<ul style="list-style-type: none"> 開発量に対するテスト項目数・不具合発生数が指標に比べて少ないか (納期と品質との優先度バランスを考慮し、潜在する欠陥のあぶり出しの必要性を検討する) 	・テスト要求分析活動分析結果

<表 6.1 テスト評価／報告 (3/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
前回の改善項目の達成率を評価する。	過去のプロジェクトで指摘された問題点の改善への取り組み ・不具合内容の解析	・前工程でのOUTPUT テスト実行活動分析結果 テストアーキテクチャ設計活動分析結果 テスト設計活動分析結果 テスト要求分析活動分析結果 ・前回の7-7のOUTPUT (過去のテスト活動評価で提案された改善案)	・前回のメトリクス分析やテスト項目書見直しで指摘された内容を反映したか ・効率化を目的とした項目整理とテストデータ改善を検討したか ・テストツール採用を検討したか、その効果はあったか。 ・次回に向けて今回のメトリクス分析やテスト項目書見直しを行ったか	達成率を評価した結果
分析結果と評価指標を比較する。	プロセス改善 ・終了報告書作成	・分析結果 テスト実行活動分析結果 テストアーキテクチャ設計活動分析結果 テスト設計活動分析結果 テスト要求分析活動分析結果 ・評価指標(テスト計画、業界標準類、類似プロジェクトの評価指標)	・カテゴリごとに工数や欠陥数が分類されているか ・他のプロジェクトと比較する情報が漏れなく収集できているか ・うまくいった点と理由も記載されているか	今回のテスト活動を評価指標と比較した結果 改善点(良かった点・悪かった点)
改善点(良かった点・悪かった点)を抽出する。	プロセス改善 ・終了報告書からの統計処理・ノウハウDBへの登録	・達成率を評価した結果 ・分析結果と評価指標を比較した結果	・効果的な改善策を次回に生かし、効果がなかった策については廃案にするか別の案を出す ・阻害要因の分析を行う(人的・環境的・設計的)	
改善策を検討する。	反省会 ・今後に向けての改善ポイントの洗い出し	・改善点(良かった点・悪かった点)	・テスト関係者およびプロジェクト支援機関や管理者など会社内で水平展開できる権限のあるメンバーを招集しているか ・アジェンダに添って議事を進め、議事録を記録しているか ※「アジェンダ例」参照 ・成功例と失敗例に対して原因をなぜなぜ分析しているか	改善案

※「アジェンダ例」

- ・テスト計画について
- ・品質基準について
- ・テストアイテム／ケースについて
- ・テストケースの失敗数／成功数
- ・テストスクリプトについて
- ・欠陥について
- ・バグ以外のインシデントについて
- ・テスト結果報告書について
- ・プロセス手順について
- ・スケジュールの計画ズレについて
- ・スケジュールの良否について
- ・コミュニケーションについて (プロジェクト内／外)

<表 6.1 テスト評価／報告 (4/4) >

項目	内容	INPUT	考慮・要素(例)・ポイント	OUTPUT
テスト評価 報告書を作成する。	テスト評価活動 の報告書作成 ・テスト評価活動全般に関する報告書のテンプレートを作成し、ルールに従って記述する	<ul style="list-style-type: none"> ・分析結果 <ul style="list-style-type: none"> テスト実行活動分析結果 テストアーキテクチャ設計活動分析結果 テスト設計活動分析結果 テスト要求分析活動分析結果 ・達成率を評価した結果 ・分析結果と評価指標を比較した結果 ・改善点(良かった点・悪かった点) ・改善策 	<ul style="list-style-type: none"> ・ホームページとリンクしたキャビネットの活用やデータベースなど可用性の高い形態で、共通化できる定型フォーマットやルールを策定しておくこと ・次回改修時(市場障害対応、OSバージョンアップ対応など)に必要な情報の記録作成 <ul style="list-style-type: none"> - 要求分析した仕様書・設計書のバージョン - テスト仕様書のバージョン - リリースしたアプリのバージョン - テスト環境の記録 ※出荷時のアプリバージョン、設計書バージョンに関しては、構成管理対象としているため、テスト成果物とは別の管理を行う場合も	・テスト評価報告書
テスト評価 報告書を開示する	プロセスの資産化 ・テスト実施の反省記録の資産化・ノウハウDB構築	テスト評価報告書	<ul style="list-style-type: none"> ・社内の技術発表会や外部団体(IVIA部会など)で発表を行ったり実施実績を収集したりするなどの活動ができれば、社内、グループ企業内、業界内で活用できる定型フォーマットやルールにすることが可能 	<ul style="list-style-type: none"> ・テスト評価報告書へのコメント、指摘事項 ・報告会(実施した場合)の議事録 ・次回プロジェクトへの引継ぎ資料

IT 検証標準工法ガイド

2014 年 4 月 20 日 Ver. 1.1

発行元

一般社団法人 IT 検証産業協会 (IT Verification Industry Association 略称 IVIA(アイビア))

〒160-0023 東京都新宿区西新宿 6-24-1 西新宿三井ビル 14F

E-MAIL : ivia_office@ivia.or.jp

URL : <http://www.ivia.or.jp>