

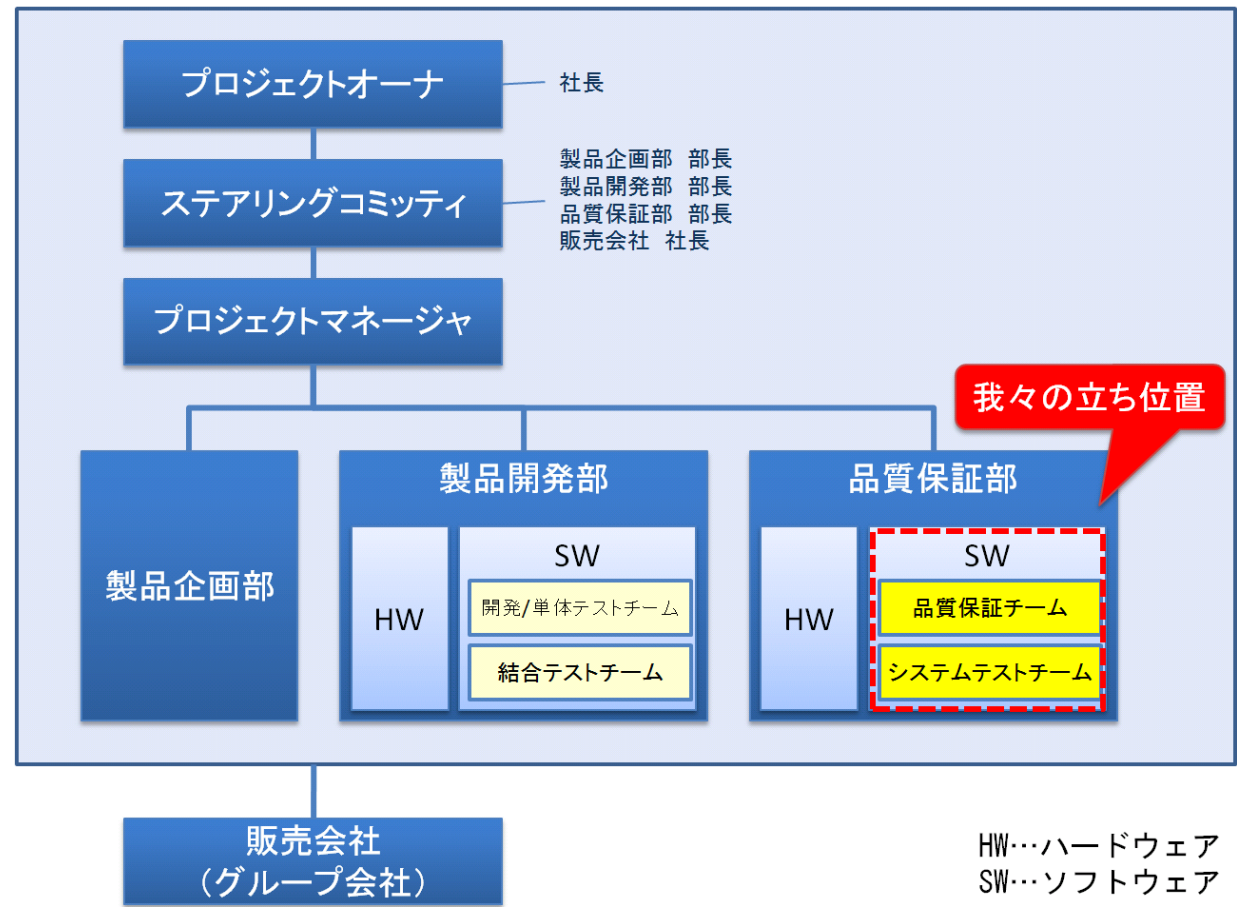
品質保証部門「モモテツ」のコンセプト

コンセプト

ユーザーが安心して製品を買ったり、使ったりできることを意識したテスト設計は普段から取り組んでいますが、販売部門や開発部門など**内部のステークホルダーがテストの進め方や内容について、安心してもらえるような取り組み**は出来ていませんでした。そのため、この機会を利用して、「**ステークホルダー全員を安心させるテスト設計**」がしたいと考え、各ステークホルダーの分析や安心させるための施策をテスト設計に盛り込みながら、活動を進めました。**特に内部ステークホルダーを意識した施策を重点的に**取り組みました。

位置づけ

普段の業務では、製品開発時の一部分のテストを担当していますが、今回は「**品質**」を**広い視点で捉えてみたい**と考え、製品開発の品質保証に早い段階から携わるカラオケシステム開発会社の**品質保証部門**という立ち位置にしました。設計・開発/テストプロセス全体の品質を計測・分析し、必要に応じて是正を促すことを盛り込んだテスト設計をしました。



方針

通信カラオケシステムに対し 検証と妥当性確認 (Verification and Validation) の視点でテストを設計・実施し、以下の全てを確認することを目的としています。

- 機能要件が満たされていること。
- ソフトウェア品質目標が満たされていること。
- 外部ステークホルダー各々に対するユーザビリティが考慮されていること。



「モモテツ」の役割

品質保証部

I 品質目標の設定および達成のための活動推進

市場ニーズの高い製品を開発するために、品質目標を設定し、達成するための活動をプロジェクト全体を巻き込んで推進する。

```

    graph LR
      A[目標を設定する] --> B[目標の確認手段や確認担当を決める]
  
```

I II III IV 品質保証チームが担当

V システムテストチームが担当

II 全体テスト計画の策定

テスト網羅を確保（漏れや意図しない重複を回避）するために、全体テスト計画（=テスト戦略。実施するテストレベルと各テストレベルでのテスト内容）を高位レベルで策定する。

```

    graph LR
      A[テスト全体で確認すべきテスト観点] -- 割り振り --> B(单体テスト)
      A -- 確認担当 --> C(結合テスト)
      A -- 振り当り --> D(システムテスト)
      E[テスト全体で確認すべき品質目標] -- 割り振り --> B
      E -- 確認担当 --> C
      E -- 振り当り --> D
  
```

V システムテストの実施

品質の最終確認のために、自らシステムテストを実施する。

全体テスト計画に沿ったテストを設計し、実施する

- 品質状況に応じてテストアーキテクチャを選択できるよう、4パターン準備しておく。
- 品質目標がクリアできているか確認を行う。

リスクベースの探索的テストを実施する

ODC分析結果から 狙うべき「テスト対象」「テスト条件」を特定し、テストを実行する。

III 開発プロセス全体の品質測定および是正勧告

品質をより上流で作りこむために、テストだけでなく開発プロセス全体の品質を測定し、必要に応じて是正勧告を行う。

品質測定の対象は **開発プロセス全体**

```

    graph LR
      A[要件定義] --> B[基本設計] --> C[詳細設計] --> D[実装] --> E[单体テスト] --> F[結合テスト] --> G[システムテスト]
  
```

実装工程の例

プロダクトメトリクス	プロセスメトリクス
FP規模	工数の予実比率 (実績工数、見積工数)
SLOC	レビュー指摘率 (レビュー工数、指摘数)
コメント率	
ネストの深さ (最大)	
サイクロマティック複雑度	

例

- 特定の担当者のソースコードが複雑だ → 担当者の教育を勧告
- レビュー工数が著しく少ない → レビュー実施を勧告
- レビュー指摘率が低い → レビュー体制や運営を確認し、是正勧告

IV 品質に応じたテスト内容の変更

プロジェクトの制約（コスト、スケジュール、スコープ）の中で品質を可能な限り高めるために、テスト設計時に「品質状況に応じてテスト内容を変更できる仕組み」を構築し、テスト実行時に制御する。

```

    graph LR
      A[ODC分析結果] --> B[読み取れる情報]
      B --> C[品質状況に応じてシステムテストの内容を分岐]
      C --> D[事前に4パターンのテスト内容を設計しておく]
  
```

読み取れる情報

- テストの網羅度
- バグの収束具合

品質状況に応じてシステムテストの内容を分岐

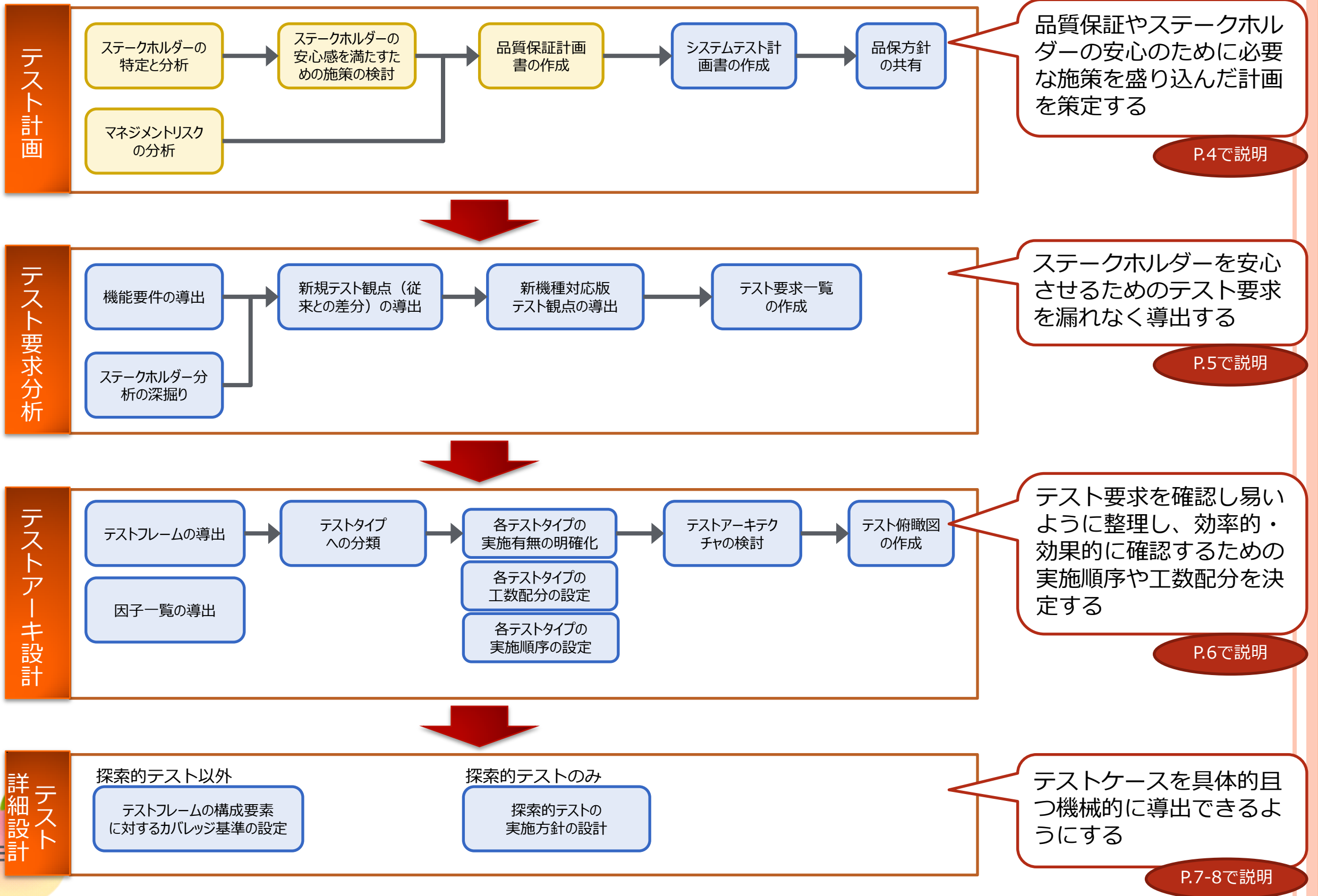
- テスト優良・バグ収束型
- テスト優良・バグ非収束型
- テスト不良・バグ収束型
- テスト不良・バグ非収束型

事前に4パターンのテスト内容を設計しておく

テスト設計の流れ (主なプロセス)

…品質保証チーム担当のプロセス

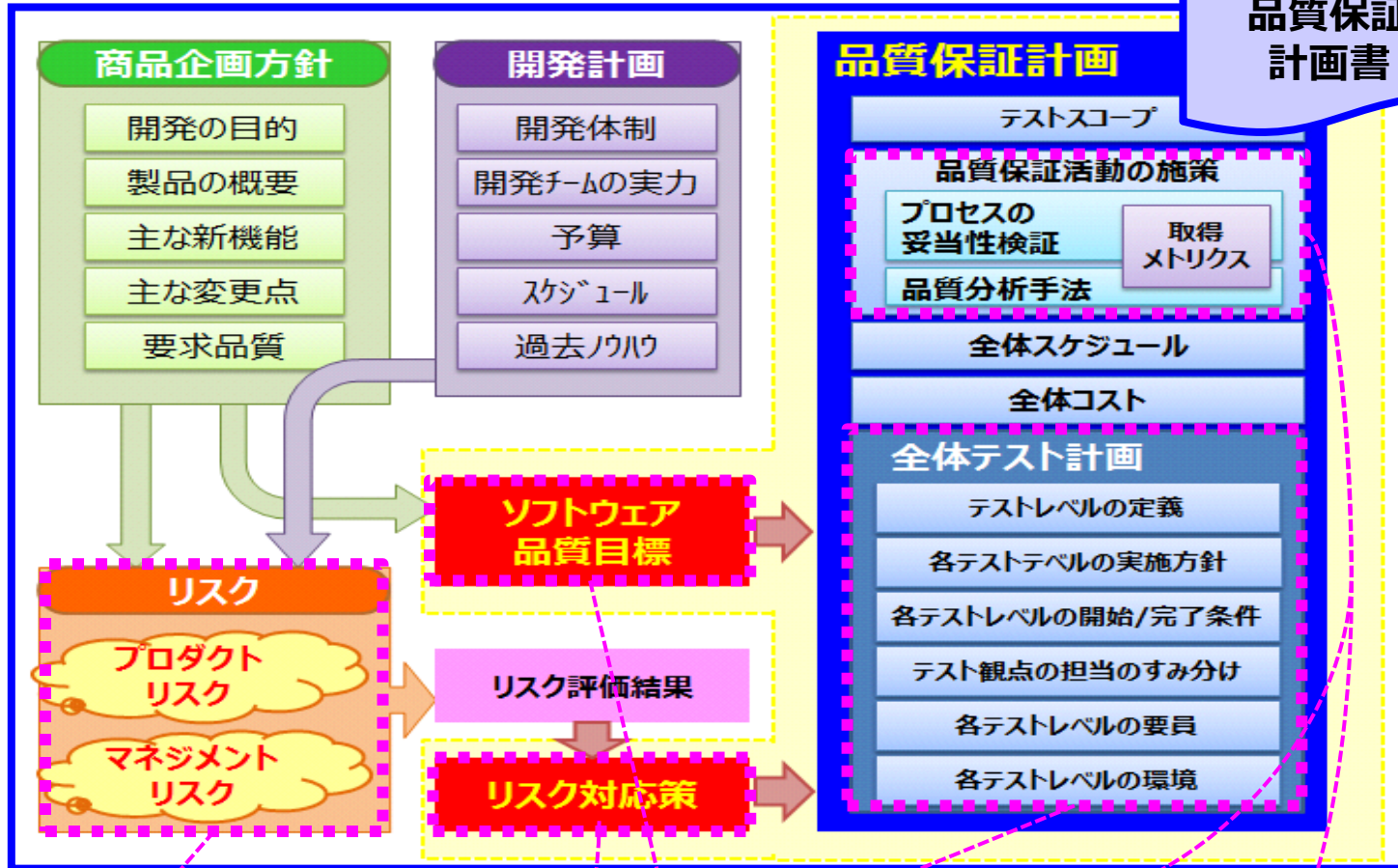
…システムテストチーム担当のプロセス



テスト計画

品質目標や潜在リスクを考慮し、品質保証計画を策定

策定した計画を、各テストレベル担当チームと共有



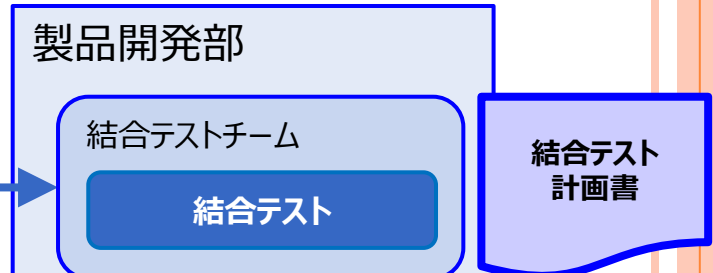
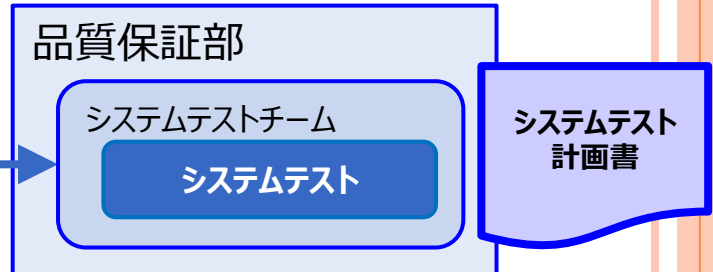
ソフトウェア品質保証計画書

各担当チームに品質保証計画に準拠したテスト計画を策定させることで...

ポイント
テスト全体での網羅性を確保

ポイント
共通手法による品質分析が可能

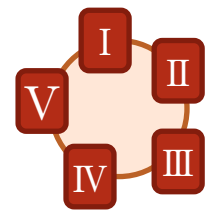
共有・準拠



ポイント
PMBOKの知識エリアを切り口とすることで、リスクの抽出漏れを防止

ポイント
品質保証部としての役割 (P.2参照) を果たすための施策を反映

ポイント
ステークホルダーに安心を与える施策を反映



ポイント
対応が必要なリスクは、対応策を反映

テスト要求分析

新機種対応版のテスト観点を導出

ポイント
ステークホルダーに安心を与える
テスト設計（コンセプト）に対応

従来機種テスト時に用いた
テスト観点

ステークホルダーの
安心感に繋がるテ
スト観点

機種差分（新規
要件）から導出し
たテスト観点

ポイント
機種のバージョンアップに対応

マージ

従来機種との差分

テスト要求一覧を作成

新機種対応版テスト観点

新機種対応版テスト観点から
テスト要求一覧を作成

テスト要求一覧

横軸： テスト対象

ポイント
「●●のテスト対象で●●の振る舞いを確認するには、どんなテスト条件が必要か」を
全ての組み合わせに対して検討

ポイント
確認する必要がない「テスト対象×（かける）振る舞い」
については、空欄では無く「×(バツ)」を入力することで、「テ
スト対象×振る舞い」の組み合わせに対する検討漏れを
防止



テスト詳細設計

テスト詳細設計書

率(使いやすさ)のテスト / sf (satisfaction) : 満足度のテスト

マスター基準
 ・従来機種に対するテストで確認実績がない水準を Must、確認実績がある水準を Option とする。

マスター基準と同じ基準を採用する場合は「J」を、異なる基準を採用する場合には具体的な基準を記述する。

テストケース	対応テストフレーム ID	分類	因子		全水準	テスト対象水準			
						選定基準	Must / Option		
USR_ev03 ユーザーが操作するにあたり、各所に備えられたコントロールの統一性があるか確認する	TF17	テスト条件	UI		コントロールの種類	-	なし	<ul style="list-style-type: none"> ・フォーム ・ボタン ・テキストボックス ・テキストラベル ・アイコン ・プログレスバー 	
			ユーザー視点テストの方式		システムテスト担当者によるOK/NG判定方式	<ul style="list-style-type: none"> ・システムテスト担当者が1名で実施する ・システムテスト担当者が複数名で実施する 	<ul style="list-style-type: none"> ・定量的な評価ができずテスト担当者の主観的な判断になってしまうリスクがあるため、複数名の議論による判断結果を採用する。 ・尚、この水準は従来機種テストで確認実績があるため Option に設定する。 	なし	<ul style="list-style-type: none"> ・システムテスト担当者が複数名で実施する
			機能関連利用時の品質モデル	画面満足性	画面快速性	可読性	表現に一貫性がある	<ul style="list-style-type: none"> ・《画面一覧を参照》 ・機能/画面内で表現が統一されていること。 ・製品内で表現が統一されていること。 ・シリーズ内で表現が統一されていること。 ・全製品内で表現が統一されていること。 ・客観的に見て、見やすく統一されていること。 	-

テストアーキテクチャ設計で導出したテストフレームおよび因子水準一覧を使用し、**機械的に**入力

テストフレーム一覧

因子水準一覧

しかし、全水準を確認するのは現実的ではない

選定基準により、テストする水準を絞り込み

ポイント
 選定基準により、テストケースの爆発を防ぎ、効果のあるテストケースの実現

因子水準の入力

テスト対象選定基準の設定



テスト詳細設計 (探索的テスト)

方針の策定

- 探索的テストの工数ボリュームを、アーキテクチャ設計で定めたパターンに応じて決定
目的：品質状況に応じた探索的テストの使い分け
- リスクベース且つピンポイント性の高いテストを実行
目的：市場バグのすり抜けを低減
- 30分から2時間程度の細かなセッションに分割して、テスト実行
目的：テスト進捗の管理/効率的な進行
- 重点項目は複数回、同様のテストを実施する
目的：発生頻度が少ないバグの流出防止
- 各自が行ったテストの内容は一覧化し、確認できるようにする
目的：テスト履歴の管理、テスト内容を共有による探索箇所の発想拡大
- テスト担当者の専任化はせずに、複数メンバーで分担して、テストする。
また、重点項目はテストスキルを考慮した上で、担当者を選定する
目的：個人差による探索視点の発想拡大、バグ検出効率の最大化を狙う

ポイント

- ・セッションを決めることで、1つの機能/観点の**テストの冗長化を防ぎ、メリハリのあるテストを実施**
- ・セッションによる、**テスト進捗の管理**が可能になる

ポイント

- ・**同様のテストを複数回行う**ことで、発生頻度の低いバグなどの流出リスクを低減
- ・探索的テストは確認内容に個人差が生じやすいテストであるため、複数名で確認することにより信頼度を向上

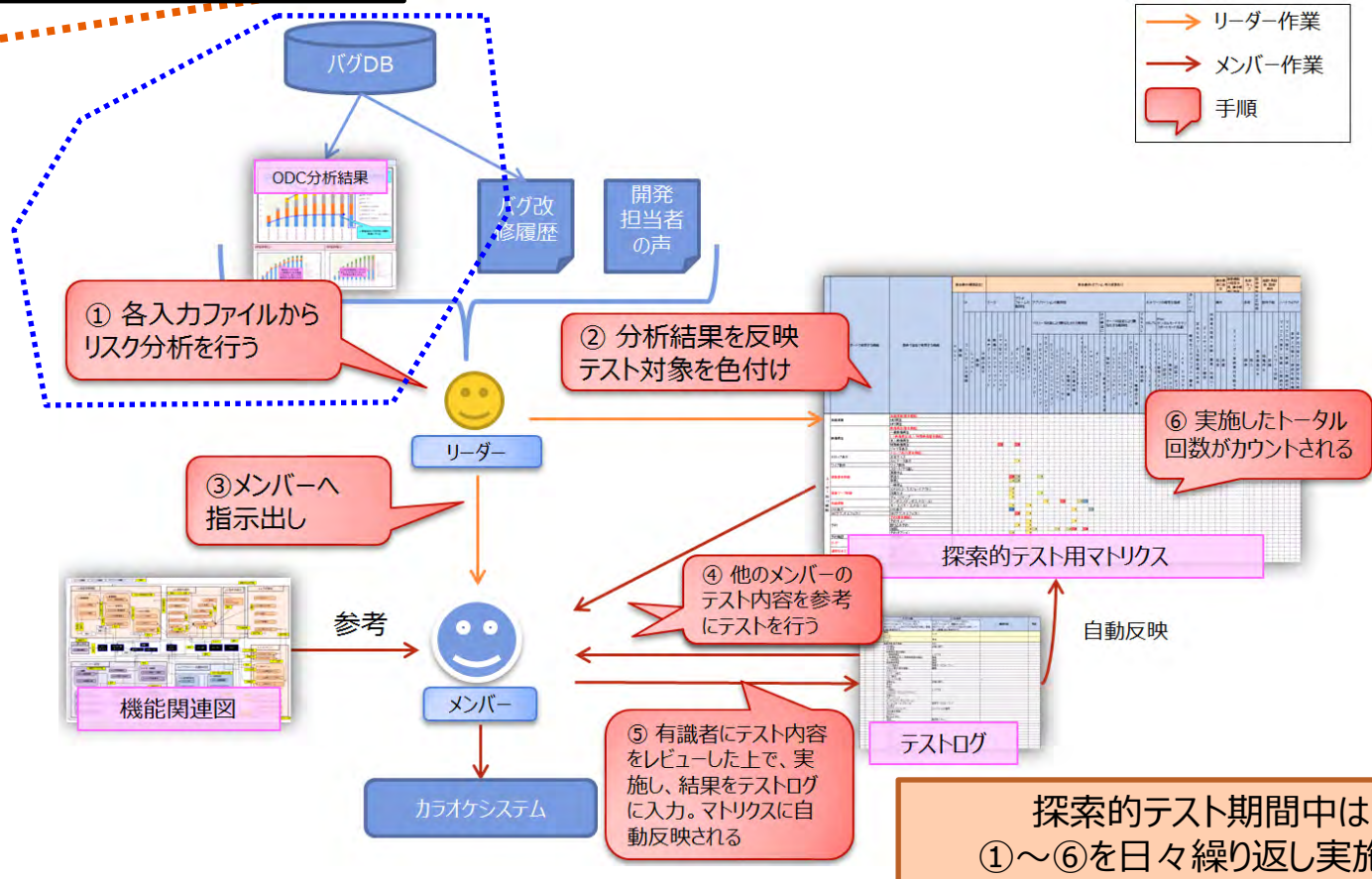
ポイント

- ・他メンバーのテスト内容を共有することで、**新たな探索視点を発想**
- ・**探索的テストの成果物**として、履歴を残す

フローの策定

ポイント

ODC分析によって、狙うべき「テスト対象」「テスト条件」をピンポイントで特定可能



探索的テスト期間中は、①～⑥を日々繰り返し実施する

