

チーム名 「紙印テスト倶楽部」

コミケにて**テスト設計本**を
70部完売!

新たなネタを披露しに来ました

メンバー紹介

小田部 テスコン5年目にして
安定のボッチ



◆ 目的

- ◆ 各ステークホルダにとっての『**価値**』の**最大化**

◆ 開発スタイル

- ◆ **短納期、繰返し、修正や変更が前提の開発スタイル**に対し、開発範囲やテスト状況を俯瞰したテストアーキテクチャと、それに連携したテスト実装方法を用いる事で、対応可能とする

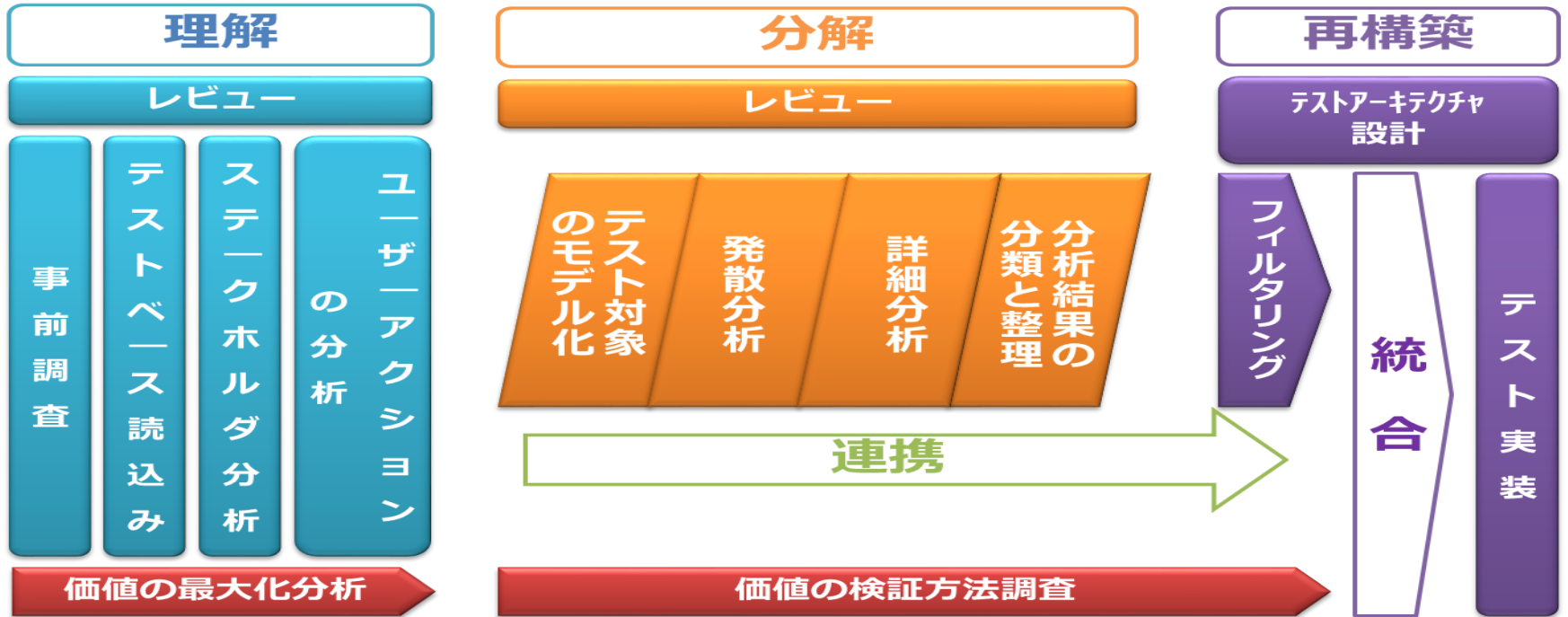
◆ 原則

- ◆ **テスト対象を極小まで分解**し、各工程を連携させたテスト設計を実施する事で、**軽量高速かつ柔軟なテスト設計が可能**とする

◆ 原理

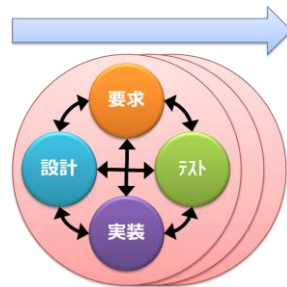
- ◆ **理解、分解、再構築**に加え**連携と統合**を意識し、上記内容に対応したテスト設計をデザインする

Fujiメソッド (テスト設計) 概要



- 以下の開発現場を想定
 - 自社開発
 - 短納期・繰返し
 - 修正及び変更が前提

- テスト設計方針
 - 軽量・高速
 - 極小化・柔軟



- トップダウンとボトムアップの摺り合わせ
 - ステークホルダの価値観からトップダウンで分析する
 - テストベースからボトムアップで分析する
 - 両者の分析結果を「統合」プロセスで摺り合わせ、各テストレベルの整合性を実現する

- 工程を厳選する
 - 「掛けたリソース < 得られる効果」であること
 - 「必要なとき、必要なだけ」成果物を作成する
 - 極小単位に「分解」したテスト対象に対して作業する
- 成果物の再利用性を高める
 - 既存の成果物に対して情報を追加、整理、又は削除して新たな成果物を作成する事で、短時間での作成を可能にする
 - 必要な情報のみを成果物から抽出可能とする事で、開発現場への柔軟な対応が可能となる
- 常に次の工程を考慮して作業する
 - 次工程での作業が効率よく進むように、現工程の成果物を作成する

事前調査と読み込み

ユーザアクションの分析



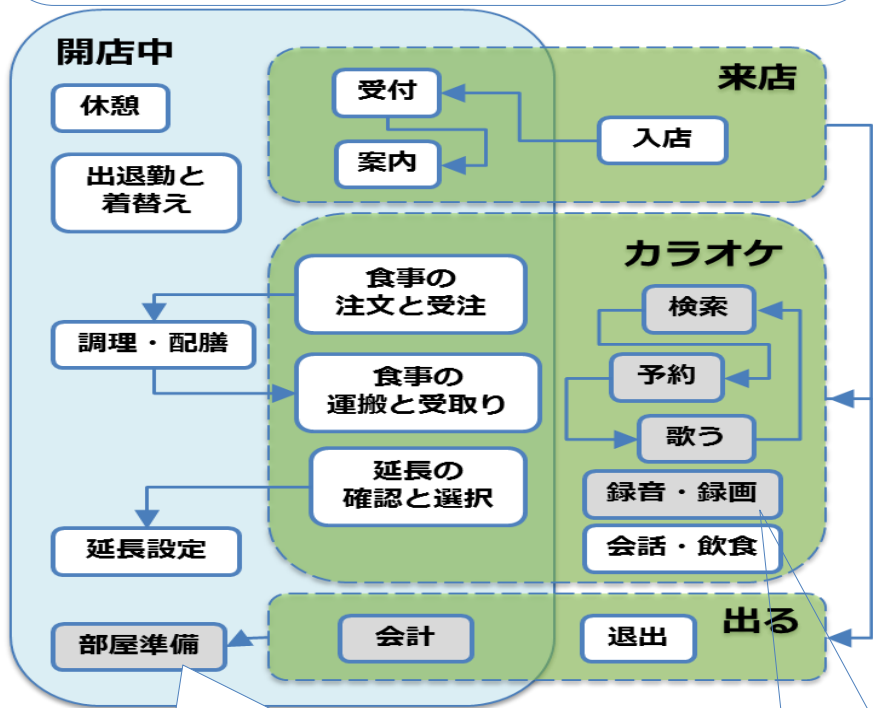
ステークホルダ分析

レビュー及びリスク分析

書籍「カラオケ上達100の裏ワザ」で調査

- ・ カラオケのルーツはスナックの流し演奏者
- ・ グループでワイワイ楽しむ
- ・ 1人でストイックに歌の練習
- ・ スナックで他の客の注目を浴びながら歌う
- ・ カラオケコンテストで腕試し
- ・ ネットで全国のカラオケ愛好者と交流

人物像と利用内容が具体化



早く終わらせるには、一括標準設定機能が欲しい

録画中に店員の割込みが入ったら嫌だろうな

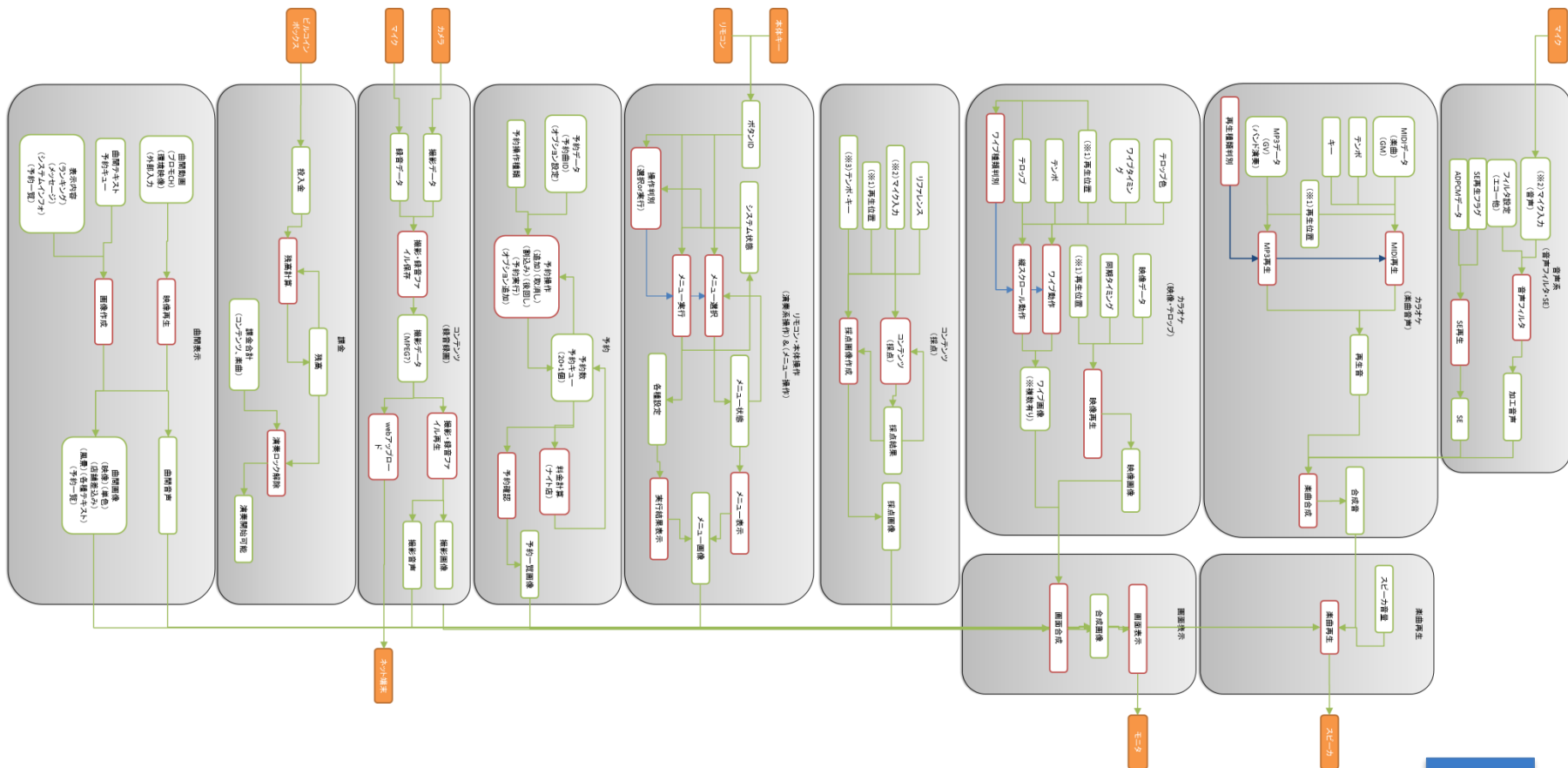
	当り前	魅力	反対	
ボックス店	個人	曲の調整 練習環境	歌唱評価 全国比較	誤評価 曲が少ない
	団体	簡単操作 見やすい画面	曲が充実 演出盛上がり	使いにくい 曲が少ない
	オーナー	一括設定 手間いらず	人気有料コンテンツ 専用の曲と映像	飽きられる
ナイト店	個人	課金明確 曲がある	周囲の注目 名音声フィル	課金トラブル 他の客が占有
	オーナー	一括設定 手間いらず	人気有料コンテンツ ワザで快適操作	課金トラブル 客同士トラブル
サプライヤ	機器の設定 しやすさ	自動接続 遠隔監視	不明な設定トラブル 解決しにくい	

- ・ 専用のプロセスは設けずにレビューを実施する
- ・ テスト設計の全プロセス中で疑問点が出てきたときに随時追加
- ・ 各プロセスは様々な視点で作業していたのでレビューを代用できた
- ・ 重点リスクの判断基準
 - 致命的な欠陥に直結
 - ステークホルダの要求に影響
 - 複雑な仕様

モデル化 (Fujiブロック図)



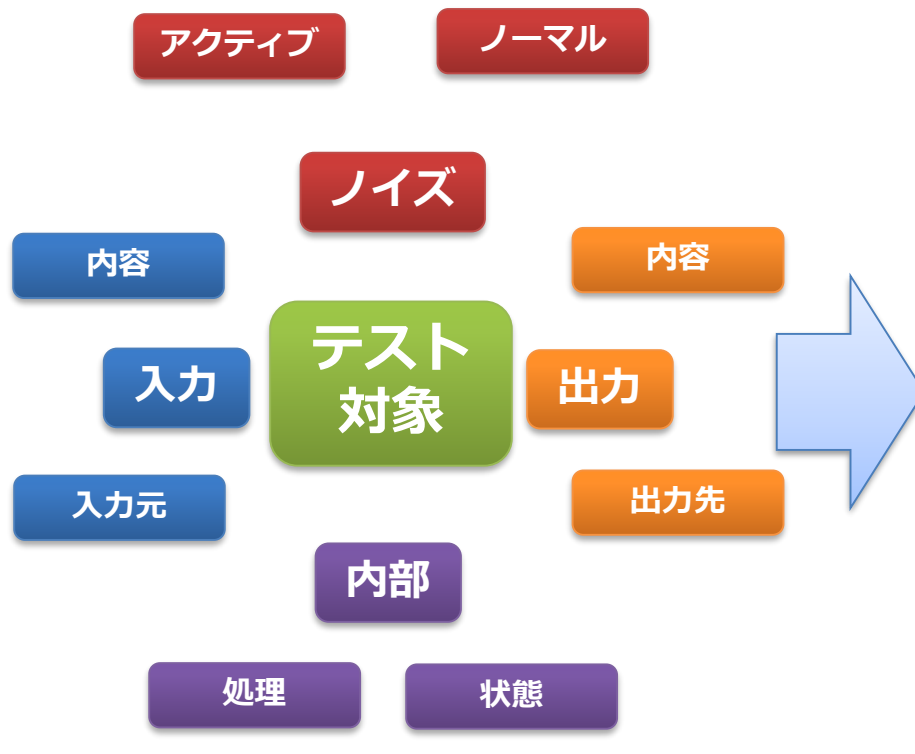
- テストベースを個々の機能とその入出力でモデル化
 - 個々の機能単位での軽量・高速・柔軟なテスト設計を可能にした





分類整理を前提としたマインドマップ

- ラルフチャートと論理的機能構造を参考に、メインブランチを設定



ツリー構造で分類整理

- 抜け漏れのチェック及び「操作・作業」「テストタイプ」「テストレベル」を追加し、テスト実装の対象を明確化



集約



全分析結果をフィルタリングで集約

- 全分析結果を「操作・作業」「テストタイプ」「テストレベル」でフィルタリングし、集約された情報を用いてテスト実装

機能

タイプ

レベル

整理結果A

整理結果B

整理結果C

整理結果D

整理結果E

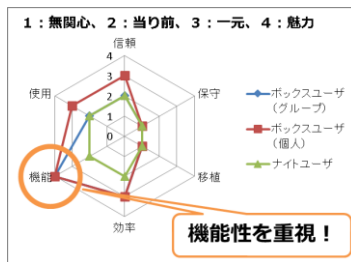
フィルタリング・集約

- ◆ **テスト実装に必要な情報のみを抽出**
- ◆ **最も知見が蓄積された時点の情報**を利用できる
- ◆ **重複した情報**もまとめてテスト実装

価値の最大化分析 と検証方法調査

統合と ビジネステスト

- 品質特性を用いた**価値の定量化**
 - 比較可能にすることで最も重視すべき価値を検討しやすくする

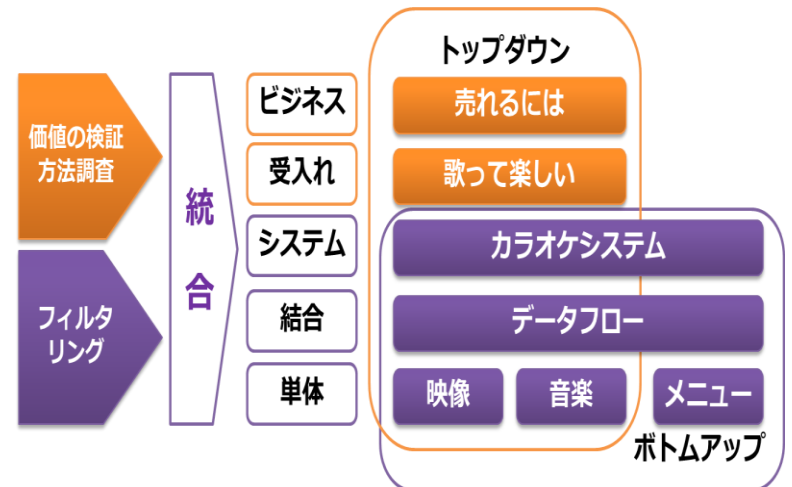
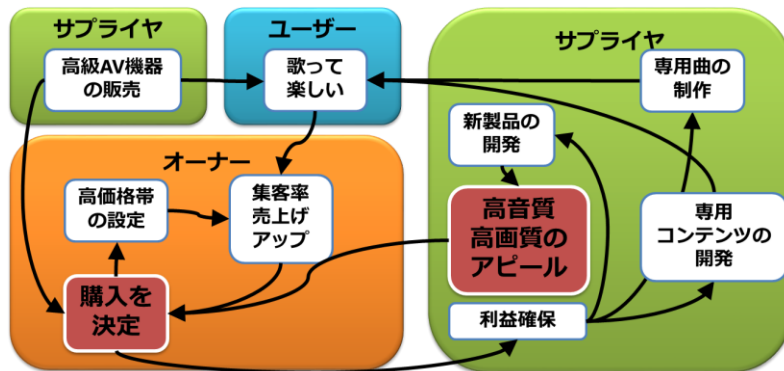


- ビジネスモデルの構築
 - カラオケシステムの**価値が最大になる条件**を模索する
 - 価値の最大化に必要な**テスト観点**を決定する
 - このカラオケシステムでのみ最も高音質、高画質であること

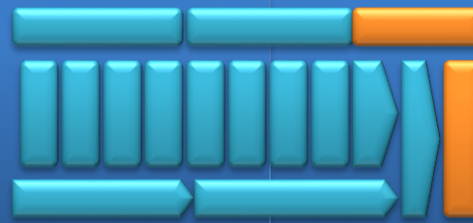
- トップダウンとボトムアップの分析結果を**組み合わせ**、各テストレベルのテスト観点を決定する

- トップダウンは主に**最終的な品質レベル**を決定する
- ボトムアップは主に**同時に確認すべき機能**を決定する

- 「**ビジネステスト**」をテストレベルに追加
 - テスト対象が**売れる**（ステークホルダにとって価値がある）ことを確認する
 - テスト設計自体の価値**も高める

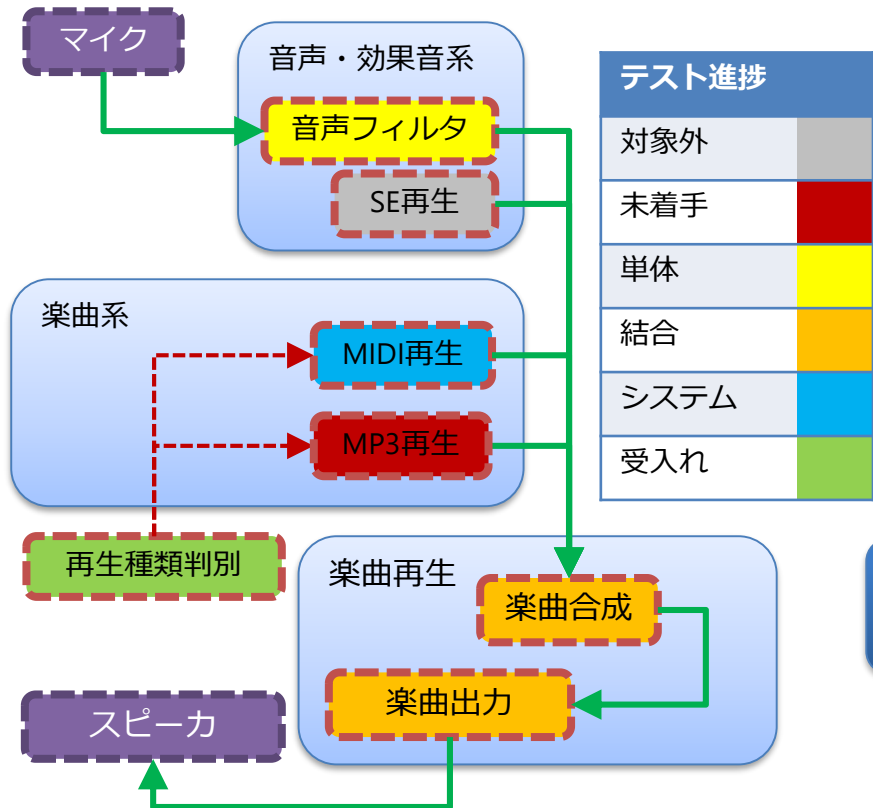


テストアーキテクチャ設計



連携型 テストケース

- 開発に合わせたテストアーキテクチャ
 - 開発範囲とテストの進捗を俯瞰可能



- 開発の進捗に合わせて実施可能なテストケースを実装する
 - 開発済みの機能と実施済みのテストレベルから次に実施可能なテストケースを判定し、必要な情報をテスト仕様書から抽出してテスト実装する
 - テストアーキテクチャと連携する事で、開発状況に合わせたテストケースの実装と実施が可能になる

