

テスト設計チュートリアル テスコン編 '21

テスト設計コンテスト'20 OPENクラス決勝戦 成果物の解説



ソフトウェアテスト技術振興協会 (ASTER)

テストベース

Quality Forward ユーザーマニュアル



Quality Forward 提案書

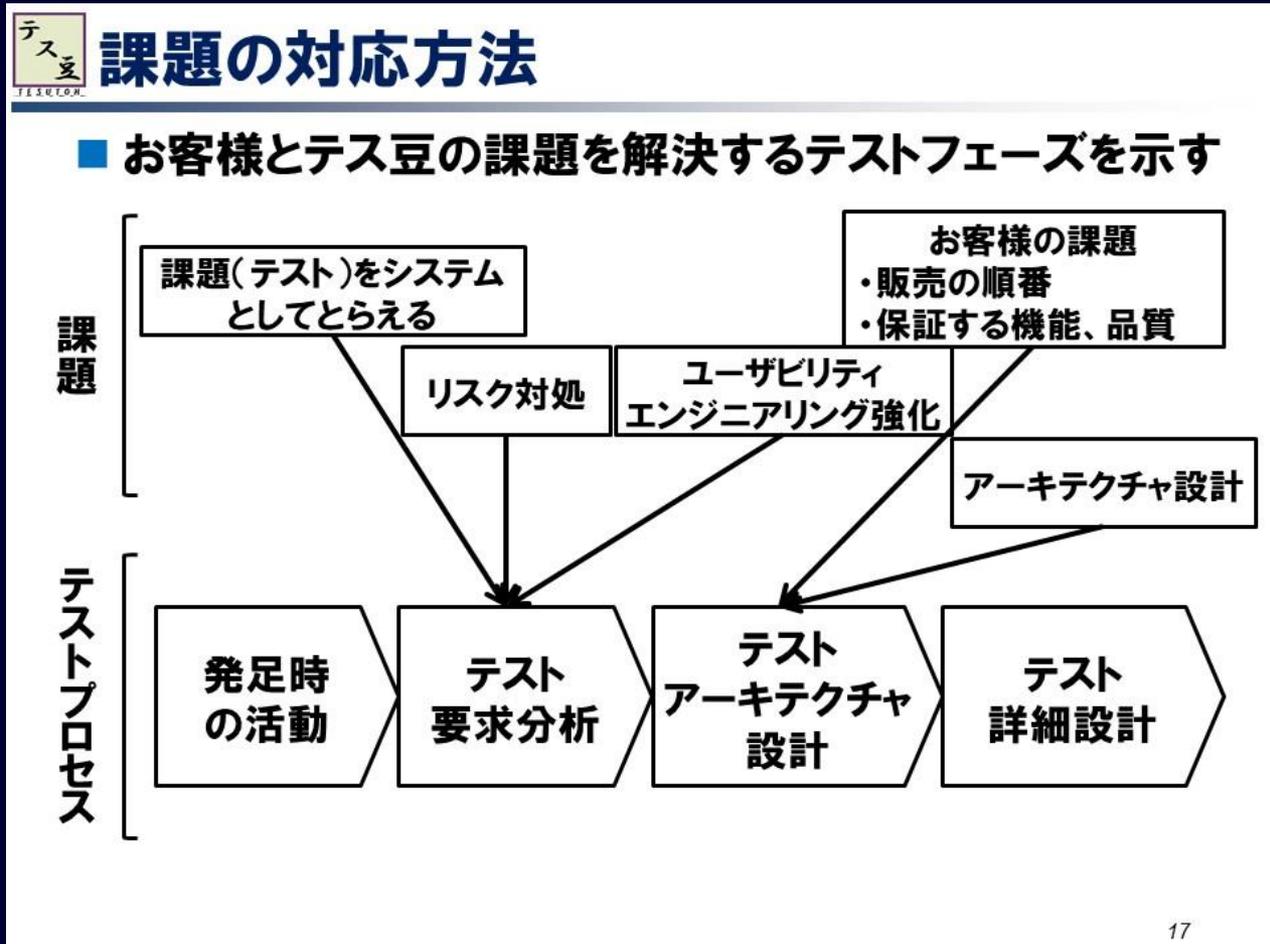


審査結果

順位	得点	チーム名
1	70	セクシーゴリラ
2	63	出席番号となり同士
3	58	ウイングアーク1st
4	56	ジョゼ
5	49	テス豆

- 成果物の一部は以下で公開されています。
<http://aster.or.jp/business/contest/contest2020.html>

テスト豆の成果物解説 ①全体像



テス豆の成果物解説 ②戦略

- **Keep**

- 「課題(QFのテスト)をシステムとして捉える」という考え方は独自性がある
てよい。

- **Problem**

- 一般的なテストの課題やチームの課題なども混在しており、整理しきれていない。

How 課題(QFのテスト)をシステムと捉える

- **テスト要求を抽出するため、以下を実施**
 1. QFのユースケース一覧を作る
 2. 各ユースケースの詳細として、アクタとQFとのやりとりを書く
 3. やりとりに出てきた活動等を機能としてまとめる
 4. 機能に求められる要求を書く

The diagram illustrates the process of extracting test requirements. It starts with an actor (stick figure) interacting with a 'テストシステム' (Test System) box. Inside the box, a specific use case is shown: '01. サインインしてPWを再設定する' (01. Sign in and reset password). This leads to a flowchart labeled 'アクタとQFのやりとり' (Actor and QF interaction). A large arrow labeled '要素分解' (Element decomposition) points to a '機能構造' (Functional structure) diagram. Another large arrow labeled '要求抽出' (Requirement extraction) points to a final diagram labeled '入力、処理、性能、人、情報等の要求' (Requirements for input, processing, performance, people, information, etc.).

- **利点**
 - ユーザとQFのやり取りからQFに対する**テスト要求を体系的網羅的に出せる**

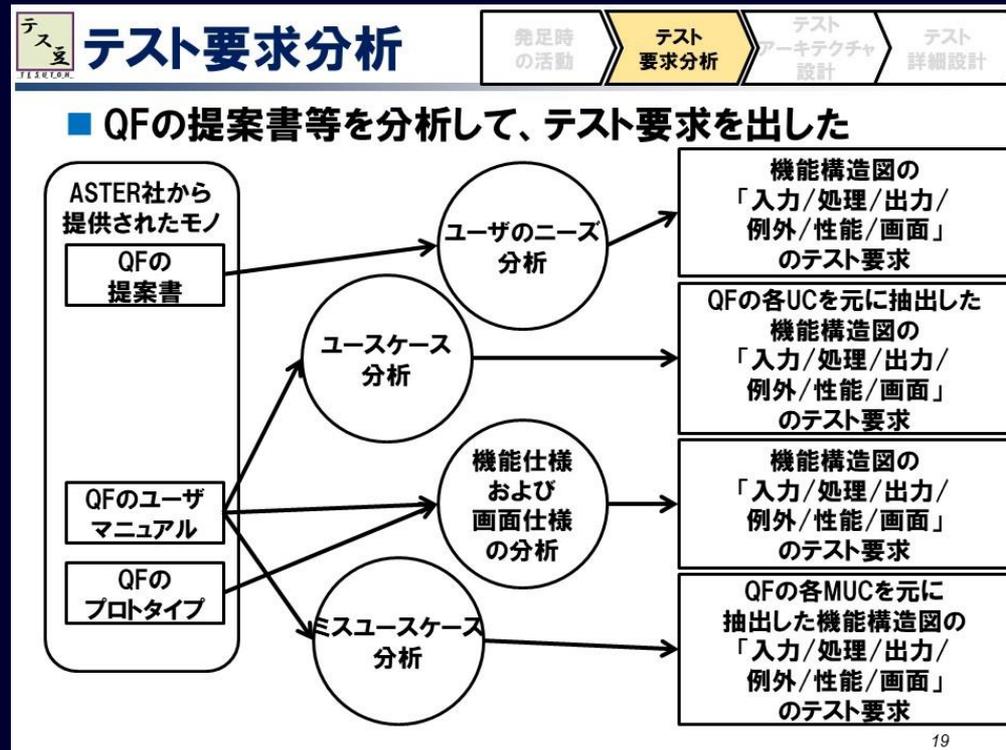
6

テスト豆の成果物解説 ③テスト要求分析

- Keep

- モデリングを積極的に活用している。
ミスユースケース※分析は独自性があってよい。

※ユースケースを脅かすユースケースのこと

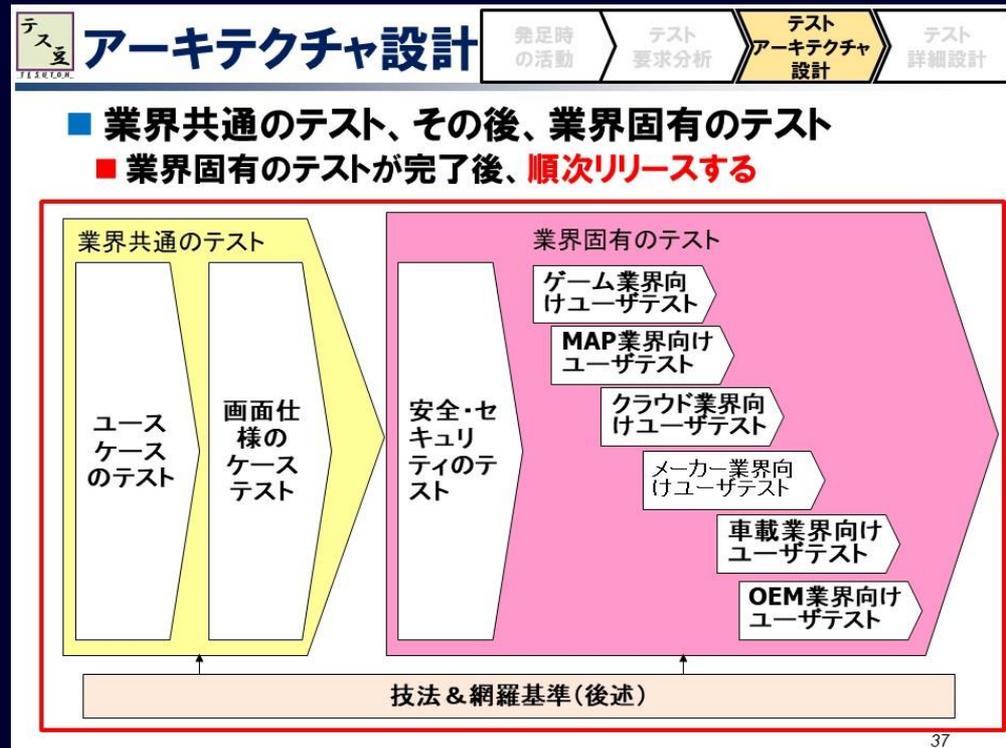


19

テスト豆の成果物解説 ④テストアーキテクチャ設計

- Problem

- テストアーキテクチャは流れを示しただけである。テスト要求分析で抽出したテスト観点との関係が見えない。



ジョゼの成果物解説 ①全体像

要求分析の手法 概要

テストアーキテクチャ検討の概要

機能、利用シーン、製品コンセプト、テストタイプの4つのPointを分析
分析結果を組み合わせてテストアーキテクチャを検討

テストアーキ検討のイメージ



ジョゼの成果物解説 ②テスト要求分析(コンセプト分析)

- **Keep**
 - 提案書にあった「コンセプト」を基に分析している点はよかった。

要求分析の手法 コンセプト分析

Input
提案書

Point
提案書に記載されているシステムコンセプトを分析し、「どういった部分で不具合や不便があると不満が出やすいか」を検討
機能分析結果をコンセプトごとに分類しテストタイプを決める

コンセプト

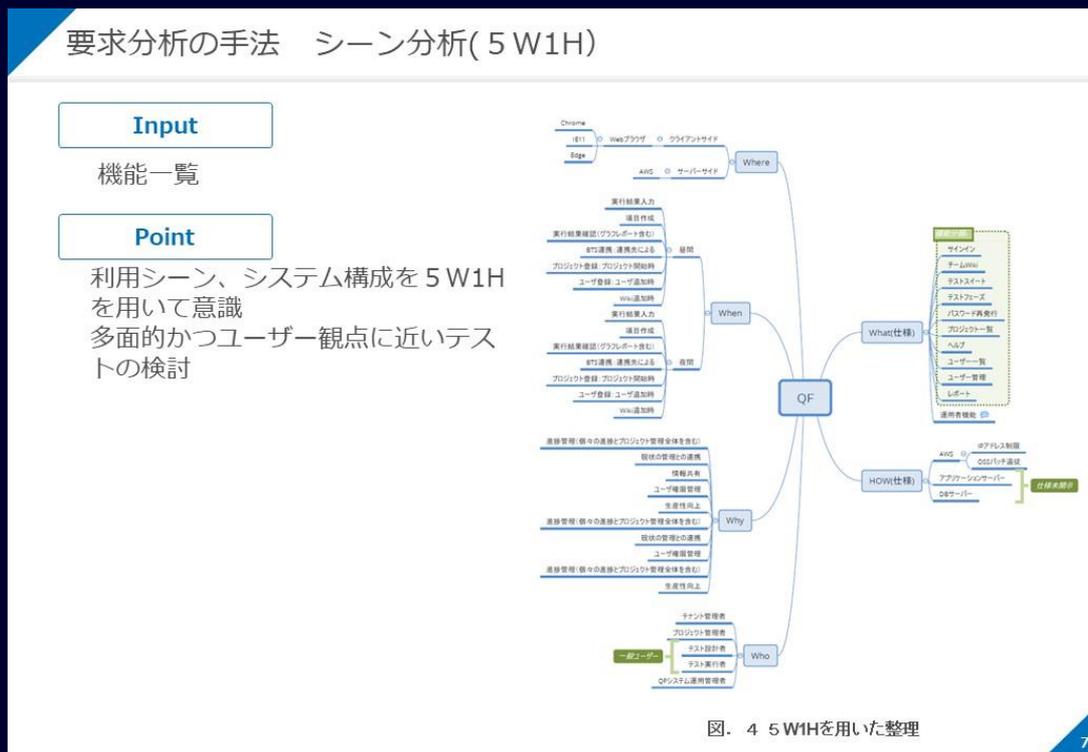
- 高度なテストマネージメント
- 大規模テストへの対応
- 超高速UI

8

ジョゼの成果物解説 ④テスト要求分析(シーン分析)

- **Problem**

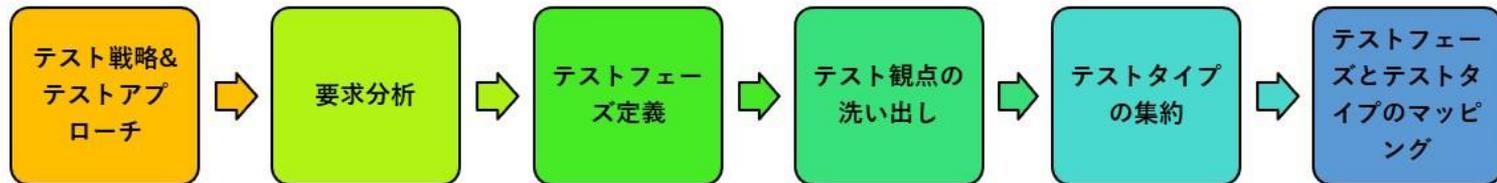
- シーン分析の結果がテスト設計に反映しきれていない。



ウィングアーク1stの成果物解説 ①全体像

テスト計画

- IEEE Standard 29119-3に基づいたマスターテストプランの作成
- 内部品質は開発チームで確保されているものとし、外部品質の実現に注力する
- ISO/IEC 25000シリーズで定義されている品質特性毎に品質目標の作成
- ソフトウェア製品品質、システム製品品質だけでなく、利用時の品質やマニュアル品質についても確保する
- 以下の流れでテストを計画する



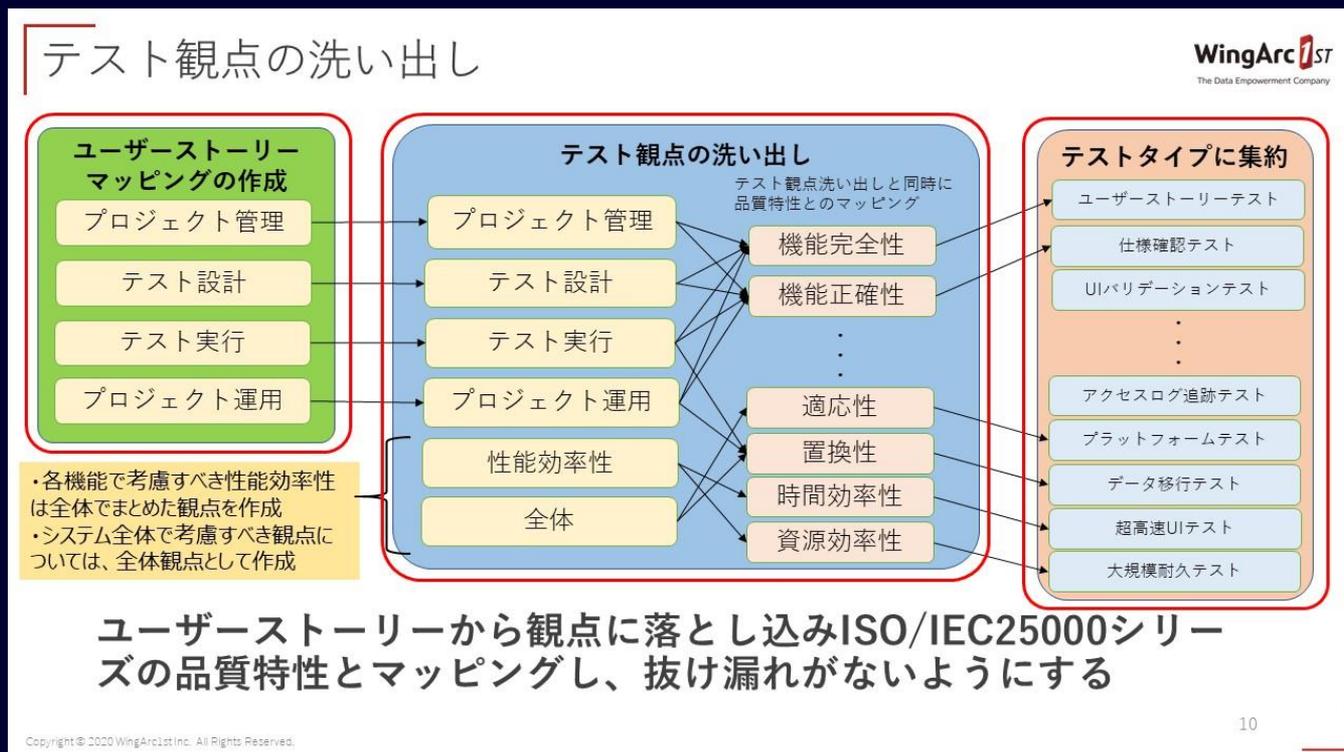
ウィングアーク1stの成果物解説 ②テスト観点

Keep

- さまざまなテスト観点を基に、テストケースに落とし込んでいる。

Problem

- テスト観点が単なる機能の羅列になっていて、何をどこまで深くテストすべきなのかが明示されていない。



ウィングアーク1stの成果物解説 ③テストタイプ

- **Keep**

- 品質特性格別テストタイプは、目新しさはないが手堅いやり方である。テストタイプの集約もよい考え方である。

- **Problem**

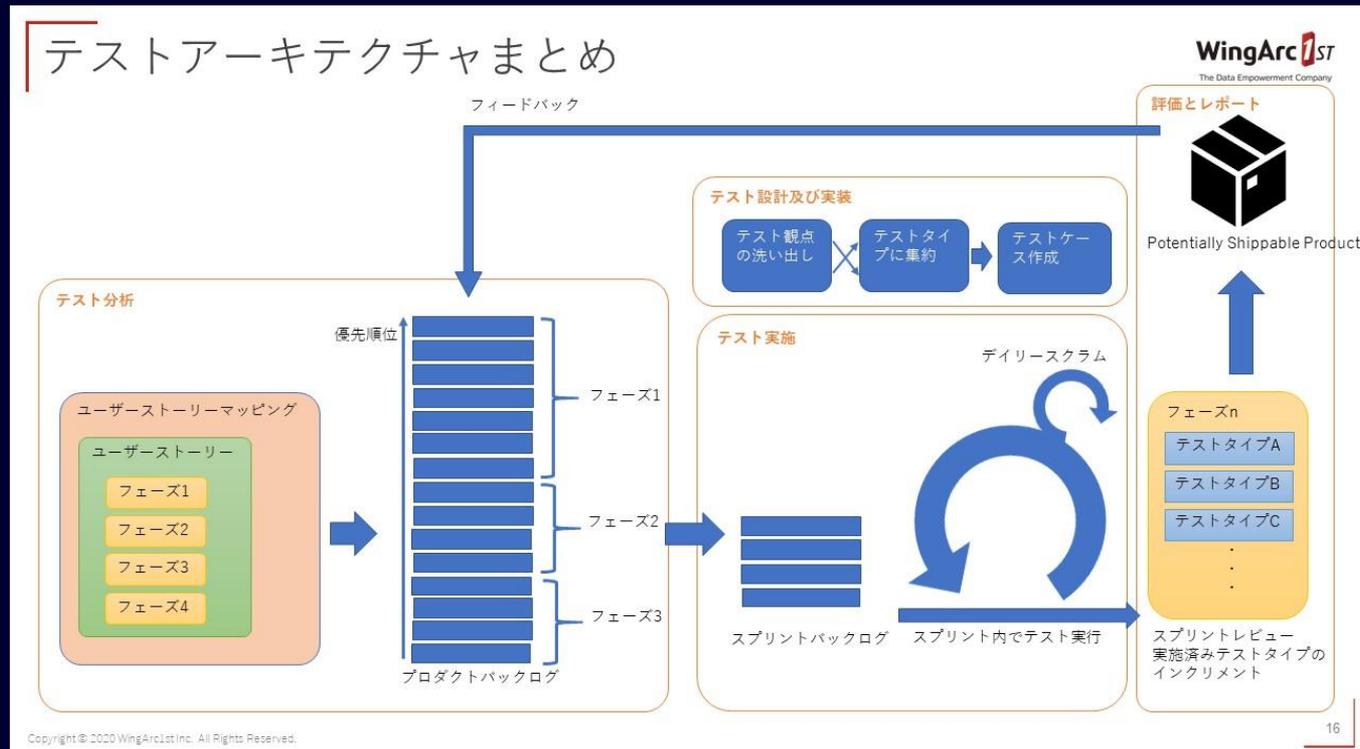
- テスト観点やテストタイプの抽出がSQuaREの品質特性に依存しすぎている。



ウイングアーク1stの成果物解説 ④テストアーキテクチャ

- **Problem**

- テストアーキテクチャが単なるプロセスになっている。



出席番号となり同士の成果物解説 ①全体像

6.まとめ

22

販売戦略 > テスト戦略 > 全体計画 > 要求分析 > 詳細設計 > まとめ

販売戦略を基にテスト戦略を立案

テスト戦略/マニュアル/提案書/SQuaREから、網羅的にテストタイプを抽出

抽出したテストタイプをリスク分析し
リスク高であるテストタイプに絞り込む
「シナリオテスト」「セキュリティテスト」「ロングランテスト」「探索的テスト」(※)

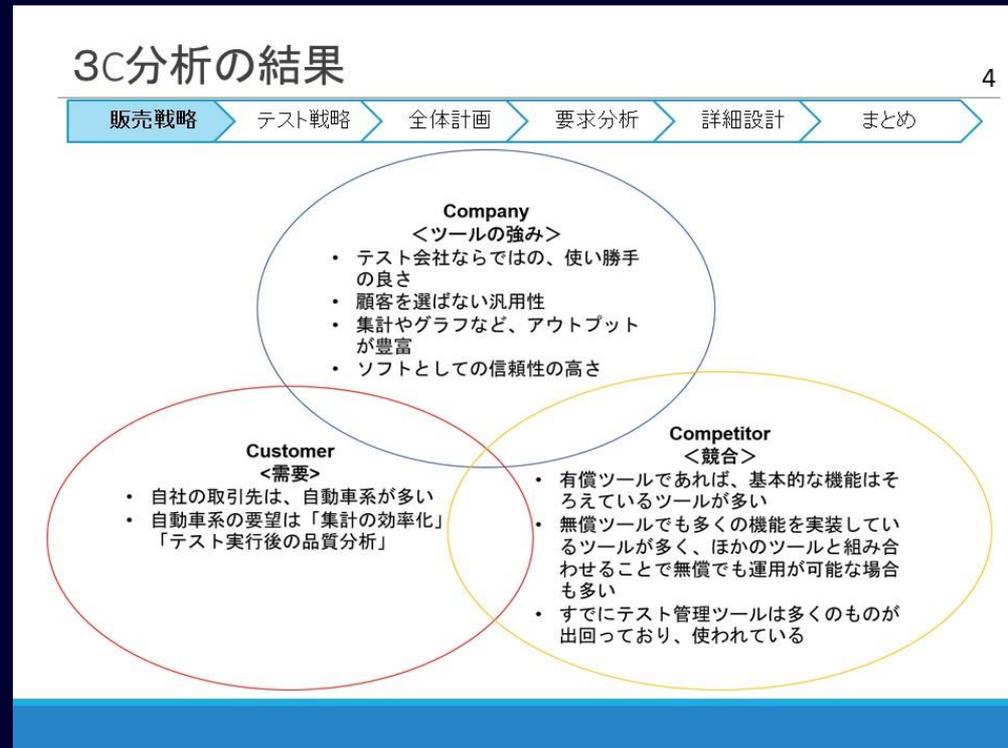
様々な視点から分析を行い、それを基に「シナリオテスト」を作成

機能、画面、状態遷移のカバレッジを80%以上確保

(※)セキュリティテストは外部ベンダーに委託
ロングランテストはシナリオテストをベースに自動テスト
探索的テストはテストチャータを準備

出席番号となり同士の成果物解説 ②販売戦略

- Keep
 - 販売戦略を考えているのはよかった。



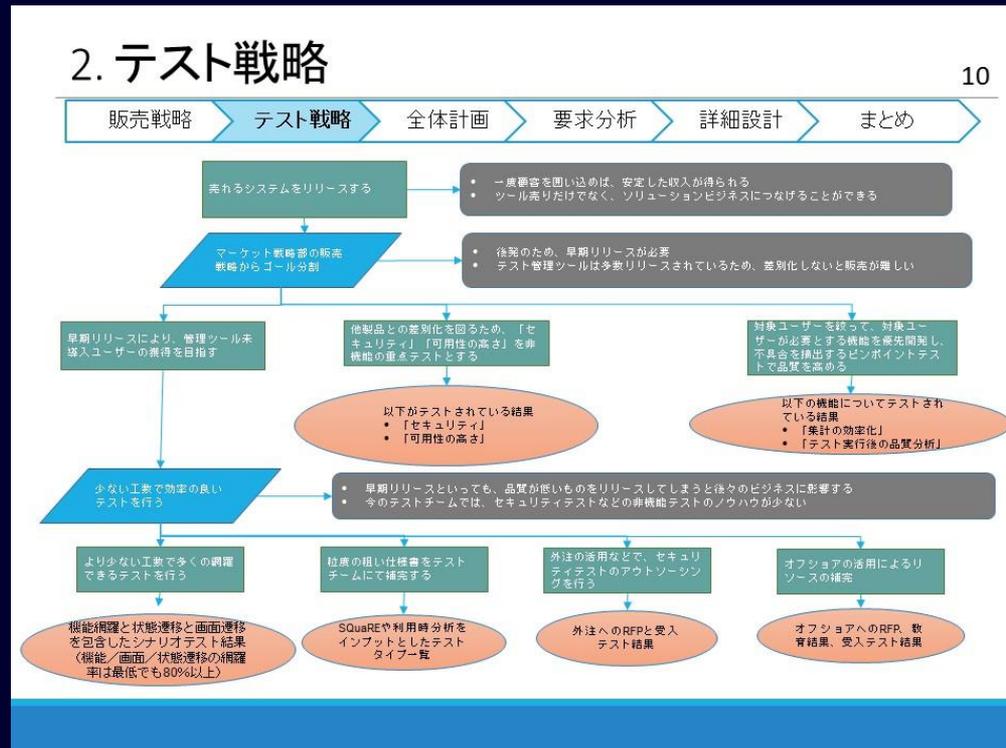
出席番号となり同士の成果物解説 ③テスト戦略

Keep

- D-Case (GSN) のような新たな技術を取り入れようとする姿勢はよかった。

Problem

- テスト戦略にD-Case (GSN) を使っているが、結論ありきのように見える。



出席番号となり同士の成果物解説 ④テストカバレッジ

Keep

- カバレッジ目標を設定して、計測、評価しているのはよかった。

Problem

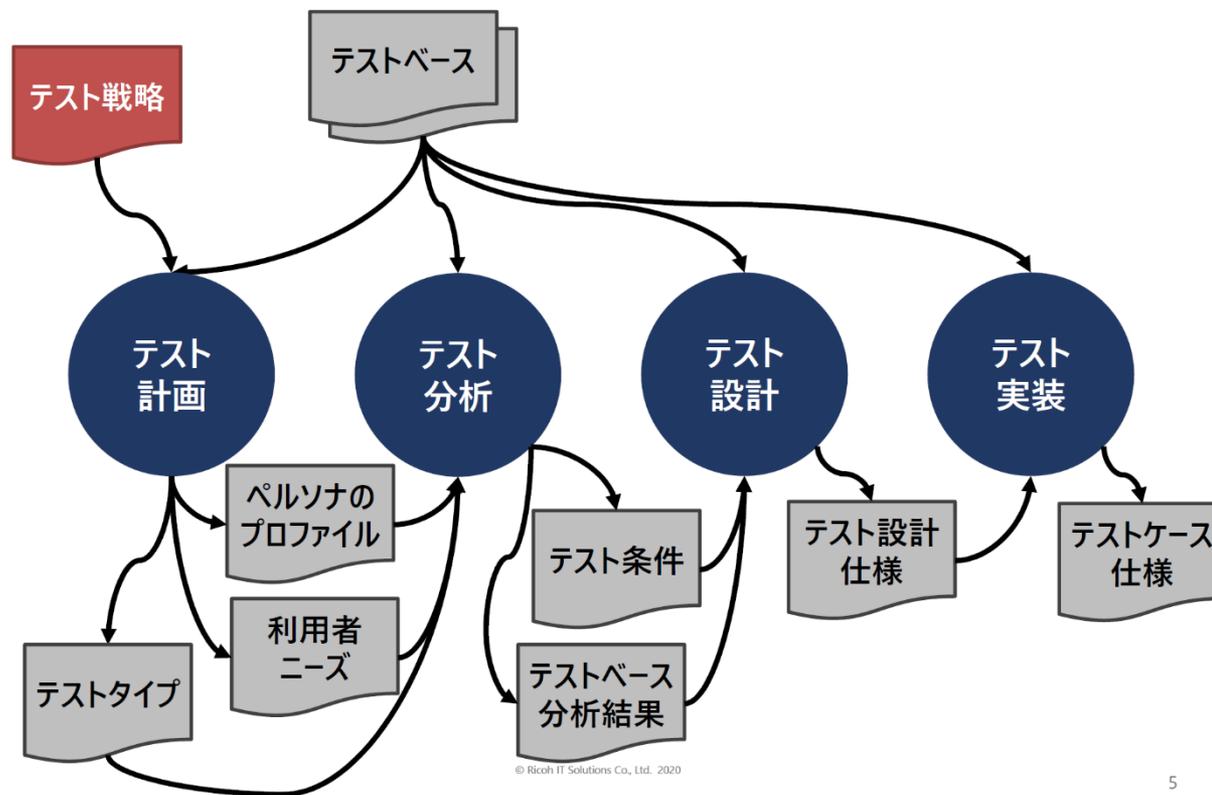
- カバレッジ目標を80%と設定することにより、重要なテストケースが漏れる可能性がある。



セキュリゴリラの成果物解説 ①全体像

■ テスト設計の全体的な流れ

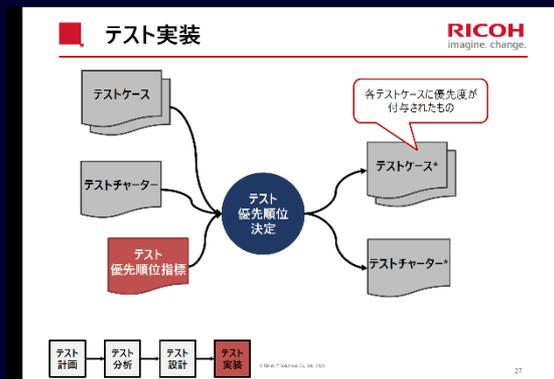
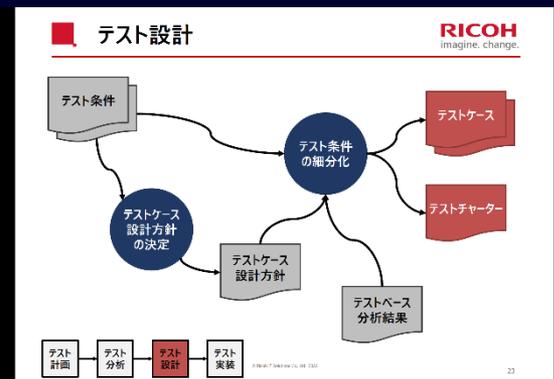
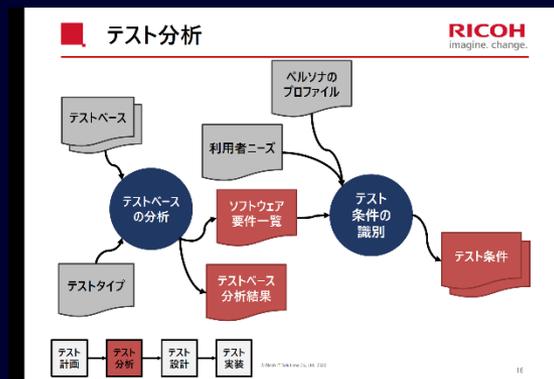
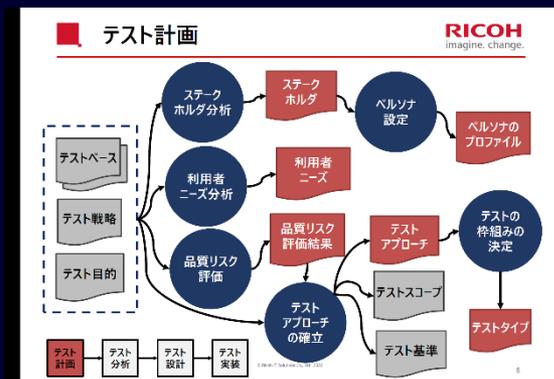
RICOH
imagine. change.



5

セクシーゴリラの成果物解説 ②テスト開発プロセス

- **Keep**
 - 全体プロセスはオーソドックスだが非常に分かりやすい。



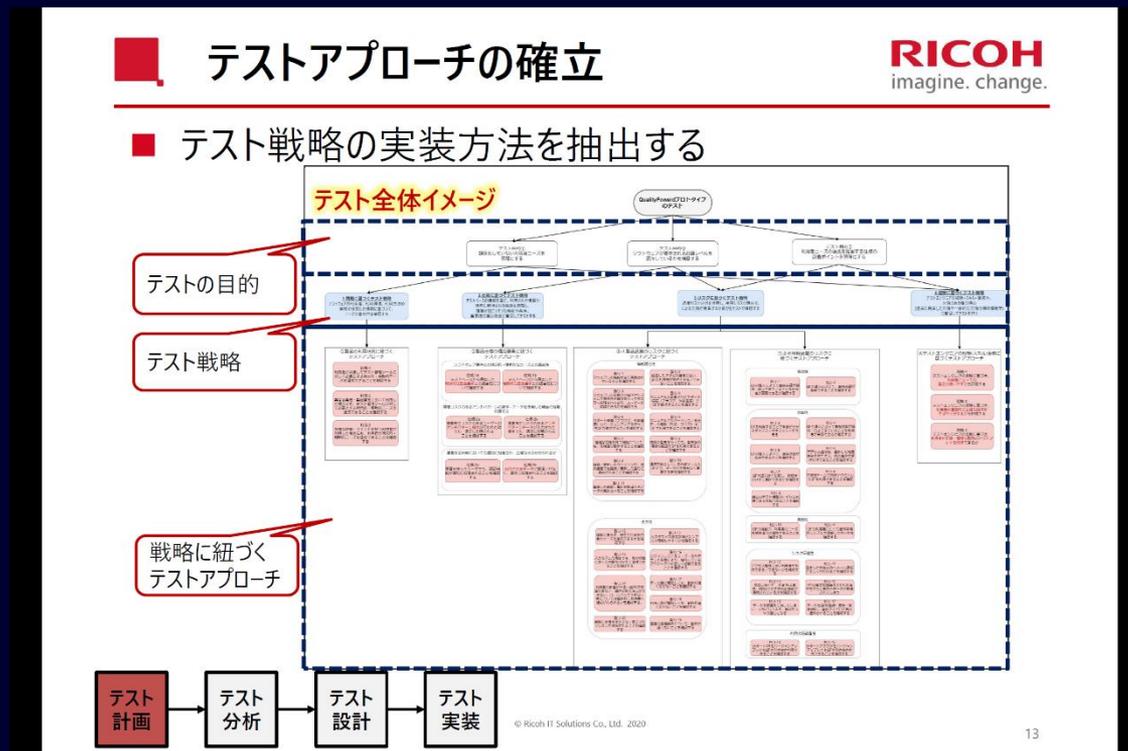
セキュシーゴリラの成果物解説 ③テストアプローチ

- **Keep**

- テストアプローチの確立、テストタイプマッピングがテストアーキテクチャとして分かりやすい。

- **Problem**

- 明確なテストアーキテクチャがない。



セクシーゴリラの成果物解説 ④テストタイプマッピング

- **Problem**

- テストタイプが3つに集約される根拠が弱い。



imagine. change.

■ テストの枠組みの決定

■ 組織で持つテストの枠組みにテストアプローチを紐づける

テストタイプマッピング

テストアプローチ	テスト範囲/リスク	テストアプローチと品質リスクの確認ポイント	テストタイプ	
① 製品の利用状況に基づくテストアプローチ	利用者が共通して使用する明示的・暗黙的なニーズとの適合性	利用-1 利用者が共通して使用するツールに対して必要とする明示的・暗黙的なニーズを達成できることを確認する	ユーザビリティテスト	
	特定の利用状況で利用が必要と感ずる明示的・暗黙的なニーズとの適合性	利用-2 異なる機種・製品開発において利用した場合にも、テスト管理ツールに対して必要とする明示的・暗黙的なニーズを達成できることを確認する	ユーザビリティテスト	
	特定の利用者が必要と感ずる明示的・暗黙的なニーズとの適合性	利用-3 多様な経験・マインドを持つ利用者が利用した場合にも、利用者が明示的・暗黙的なニーズを達成できることを確認する	ユーザビリティテスト	
	② 製品仕様の構成要素に基づくテストアプローチ	ソフトウェア要件との明示的・暗黙的なニーズとの適合性	仕様-1a テストベースから抽出した明示的な製品要件との適合性について確認する	機能テスト
		障害リスクのあるアンチパターン/データの操作・データを使用した場合の処理の正しさ	仕様-1b テストベースから抽出した暗黙的な製品要件との適合性について確認する	機能テスト
		悪意ある利用においても適切に処理され、正確な出力が行われるか	仕様-2a 障害発生リスクのあるユーザーのアンチパターン/操作が行われた時でも、適切に処理されることを確認する	機能テスト
③-1 製品品質のリスクに基づくテストアプローチ	製品品質のリスク	仕様-2b 障害発生リスクのあるアンチパターン/データが入力された場合でも、適切に処理されることを確認する	機能テスト	
	テスト活動の経験値が低い利用者が、QFを利用できない例が多発した	仕様-3a 悪意ある利用においても適切に処理され、正確な出力が行われるか	機能テスト	
	QFのカスタマイズのための設定方法が分からず、使われない機能がある	仕様-3b 出力されるデータに間違いがなく、適切に処理されることを確認する	機能テスト	
	認証機能が有効に機能せず、利用権限のないものがコンテンツを利用できる	制御-4 経験に限らず、想定される利用者のニーズを達成できるかを確認する	ユーザビリティテスト	
	QFをマニュアル記載外の操作や不適切な操作による誤動作が適切に発生しないことを確認する	制御-5 カスタマイズ設定手順がシンプルで理解しやすいかを確認する	ユーザビリティテスト	
		制御-1 設定したアクセス権限に従い、QFの利用者が利用できるべきでないことを確認する	機能テスト	
		制御-2 マニュアルに記載外の操作やシステムで想定外の操作や	機能テスト	

テストアプローチと品質リスク一覧

テストの枠組み(テストタイプ)である
①ユーザビリティテスト、②機能テスト、
③業務効率性確認テストに紐付け

テスト計画

→

テスト分析

→

テスト設計

→

テスト実装

© Ricoh IT Solutions Co., Ltd. 2020