

2021テスト設計コンテスト
発表資料

チーム： ジョゼ

- チーム紹介
- テスト設計の背景
- アーキテクチャ設計
- テスト詳細設計
- まとめ

0. チーム紹介

ジョゼの概要

普段は、システムテストを生業としている、とある企業の社員たち
有志で集まって日夜のテスト設計を勉強中
2021シーズンは新たにメンバーを加えチームを刷新

ジョゼの由来

ヨーロッパサッカー界で有名な監督「ジョゼ・モウリーニョ」から拝借
テストの世界で「Special One」なものをいつか生み出せればという思いから
たとえ彼がチームを解雇されたとしても、新たなチームの監督に就任しても、
我々の思い、目標は変わらず…

ジョゼの構成員

齊藤 小山 遠藤 松井 小笠原 大久保 越湖 須藤

1.テスト設計の背景

テストの背景

システム開発：A社



テスト開発：ジョゼ社



A社からの依頼はテスト管理ツールの「システムテスト設計」

- テスト実施状況：単体・結合・総合テストは実施済み
- 提供されるもの：ユーザーマニュアルと提案書

Point

ユーザー視点のテストによる品質向上

1.テスト設計の背景

背景

- QFがリリースされ、多くの企業で導入されている中で、機能アップデートの機会が増加
- そのような状況から、開発の効率化も必要となってきた

前提

ソフトは世の中にリリース済み。一連の評価は完了している前提（2020テストコン）

- 企業からの問い合わせも多くなり、機能アップデートの機会が増えている
- 開発効率向上のために自動化も検討すべき
- ユーザーとしてのテスト設計をしてください

に着目し…

**ユーザー視点のテストによる品質向上
効率的にアップデート部分を評価**

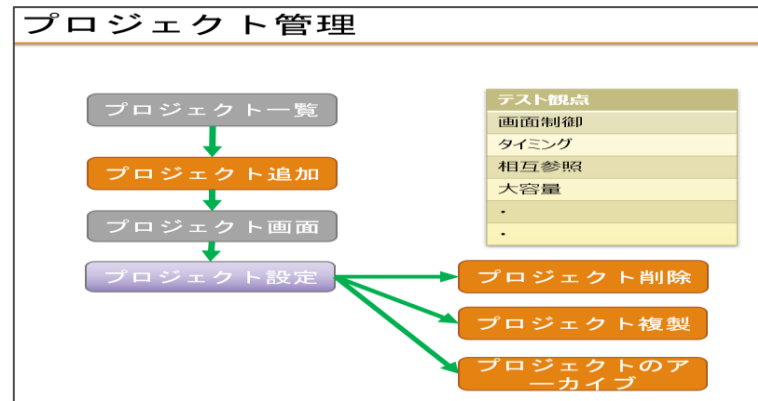
1. テスト設計の背景 保証する内容

ジョゼの保証前提

- QF本体の基本品質は、2020テストコンで保証済みとする
- アップデート部分の機能と、アップデートに伴う基本機能の評価を検討

テストアーキの検討

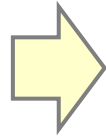
- ユーザー視点で、対象機能が含まれるユースケースを検討
- ユースケースに該当する観点をテンプレート化
- これらを掛け合わせることで、「ユーザー視点で効率的な評価」を実現



ユースケース×観点を、効率的かつ観点の漏れなく
アップデート部分の評価

1.5. 全体のプロセス

テスト設計の進め方



要求分析

- ・ 要求分析
- ・ 画面遷移図、機能一覧作成

アーキ 設計

- ユースケース分類
- ユーザー分類
- ユースケースコンテナ検討
- テスト観点、テストタイプ検討

詳細設計

- テスト目的の設定
- 詳細観点、関心毎、水準洗い出し



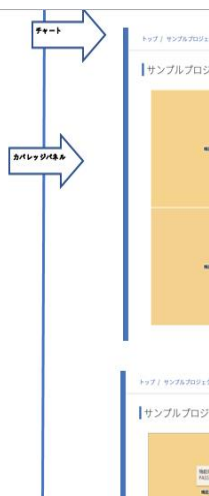
2. 要求分析の手法(機能分析)

Input

ユーザーマニュアル 提案書 現行ソフト

Point

提案書、ユーザーマニュアル、ソフト動作から、画面遷移図を作成
 操作・部品を整理
 遷移図から機能をリストアップし、一覧を作成



機能1	機能2	機能3	機能4	機能5
プロジェクト一覧	プロジェクト一覧表示	「参加ユーザ」リンク	ユーザ追加	選択したユーザを追
プロジェクト一覧	プロジェクト一覧表示	「参加ユーザ」リンク	ユーザ追加	設定変更
プロジェクト一覧	プロジェクト一覧表示	「参加ユーザ」リンク	ユーザ追加	プロジェクトから外す
プロジェクト一覧	プロジェクト一覧表示	「参加ユーザ」リンク	お知らせリンク	
チームWiki	削除	削除完了	削除	
チームWiki	プレビュー	「更新」リンク	保存	
レポート	プロジェクト全体のレポート	収束曲線グラフ	総数	
レポート	プロジェクト全体のレポート	収束曲線グラフ	変動	
レポート	フェーズ一覧	進行状況棒グラフ	「カバレッジパネル」リンク	
レポート	フェーズ一覧	進行状況棒グラフ	「チャート」リンク	
レポート	フェーズ一覧	進行状況棒グラフ	「実績推移表」リンク	
レポート	チャート表示	ラベル設定	更新	
レポート	チャート表示	バグ情報のアップロード	ファイルを選択	登録

遷移図と機能分類一覧の例

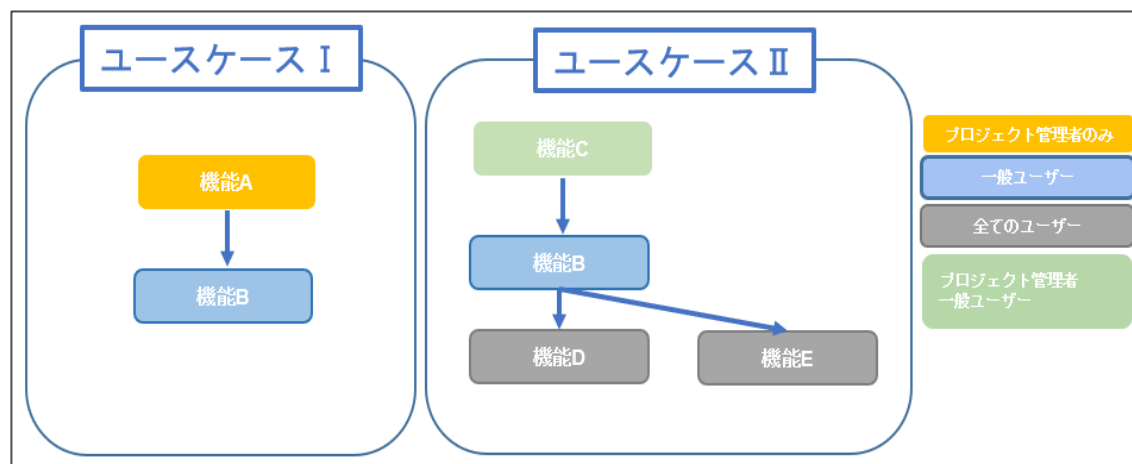
3. アーキテクチャ設計 ユースケースコンテナについて

ユースケースコンテナとは

ユースケースごとに、「誰が」「どの機能を」「どのタイミングで」使用するかを配置し構造化したモデル図

Point

「ユーザーとしてのテスト設計」という前提から、「テスト対象の機能が実ユースケースで使用に耐えうること」をカバーできるテスト構造として検討



ユースケースコンテナの例

3. アーキテクチャ設計 ユースケース分類

Input

機能一覧

Point

基本ユースケースで利用される機能を分類
複雑なケース、イリーガルケースなどは考慮しない
→あくまでユースケースに対するテスト観点

機能1	機能2	ユースケース分類
サインイン	「パスワード再発行」リンク	サインインできない場合
	「プロジェクトトップ」リンク	サインインできる場合/サインインできない場合
	アカウントロック	サインインできない場合
	サインイン成功ONS表示	サインインできる場合
	パスワード入力	サインインできる場合/サインインできない場合/パスワード変更
	メールアドレス入力	新規ユーザー招待/サインインできない場合
チームWiki	Wikiトップページ表示	Wiki閲覧/Wiki編集
	フルビュー	Wiki編集
	ページ一覧表示	Wiki閲覧/Wiki編集
	削除	Wiki編集
	内容表示	Wiki閲覧/Wiki編集
テストスイート	excelインポート	テストケース取り込み
	excelシート整形	テストケース取り込み
	ファイル選択	テストケース取り込み
	excelエクスポート	テスト結果のエクスポート
	カラム設定	カバレッジパネル表示
	カバレッジパネル表示	カバレッジパネル表示
	カラム設定	カバレッジパネル表示
	ステータス表示	ステータス設定変更、ステータス設定更新

ユースケース一覧の例

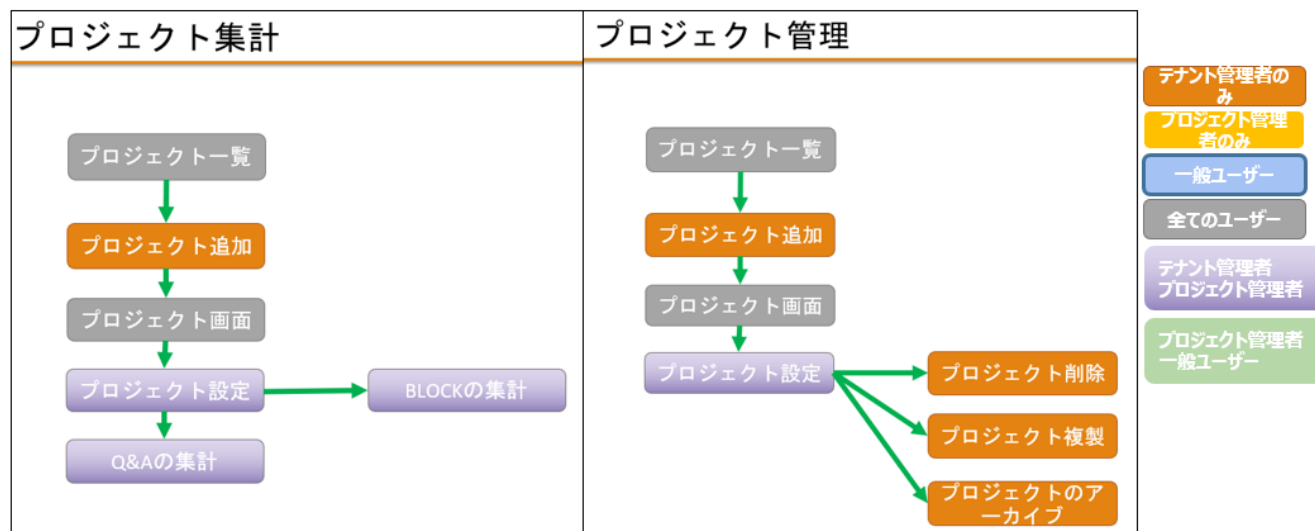
3. アーキテクチャ設計 ユーザー分類

Input

ユーザーマニュアル 機能一覧

Point

ユースケースに配置された機能に対して想定ユーザーを割り当て
実ユースケースとして「誰がよく利用するか」という視点で分類する
固まりの中で分類してコンテナ化完了



ユースケースコンテナの例

3. アーキテクチャ設計 テスト観点ツリー

Input

観点テンプレート

Point

テスト観点テンプレートを活用し、経験則に頼らない観点の検討
テスト観点テンプレート+QFの機能、特性を发散させ漏れなく洗い出し



観点ツリーの例

4. テスト詳細設計

Input

ユースケースコンテナ テスト観点ツリー

Point

各テスト観点ごとに以下を設定し、評価内容を検討

- 評価すべき目的（テスト目的）
- 具体的な確認観点（詳細観点）
- パラメータ（水準）テストの関心ごと

No	観点分類1	観点分類2	テスト目的	詳細観点	関心ごと	水準
1	環境	HW	クラウドシステム構成を確認する	可用性としての製品品質を担保していることを確認する。 冗長構成によりHW故障時にサービス継続できることを確認する。	SLA稼働率 99.9% 稼働実績 99.99% セキュリティ・データ破損にまつわる市場不具合0件 (Quality Forward 提案書に規定値)	IPアドレス制限によるアクセス制御 モニタリングとロギングによる不正アクセス確認 ログイン方式による第三者不正利用の防止
2	環境	ミドルウェア	クラウドシステム構成を確認する	可用性としての製品品質を担保していることを確認する。 ミドルウェア障害時にサービス継続できることを確認する。	SLA稼働率 99.9% 稼働実績 99.99% セキュリティ・データ破損にまつわる市場不具合0件 (Quality Forward 提案書に規定値)	セキュリティテストによる脆弱性の防止

5. まとめ

テストで保証されること

評価対象機能をユースケースごとにコンテナ化
評価観点をテスト観点テンプレートをベースにツリー化
コンテナ×観点ツリーで以下を達成

- ユースケースを意識した評価
- 評価方針に従ったテスト対象とテスト観点を効率よくピックアップ

今後の展望

自動化を含めた効率化の検討
→自動化対象の絞り方とその妥当性