

つよつよなりたいはんぺん

# テスト設計コンテスト'22 U-30クラス

2022/09/17



PARTNER  
SINCE 2010



- 1 | チームとコンセプトの紹介
- 2 | アピールポイントとテストプロセスの全体像
- 3 | 各プロセスの強み
  - ① テスト要求分析
  - ② テストアーキテクチャ設計
  - ③ テスト詳細設計
  - ④ テスト実装
- 4 | まとめ

1

# チームとコンセプトの紹介



## 構成

 TeamSpirit のQAエンジニア。

製品であるSaaSのテスト業務を担当している。

品質保証・テスト経験、職歴などバラバラの4人。



たちばな



かわにし



たなか



みき

## チーム名の由来

品質保証・テスト業務の経験が浅く（10ヶ月～3年）、今はまだ弱い（**白くて柔らかい**）ため、これから知識をぐんぐん**吸収**して、**強**くなりたい。

つよつよなりたいはんぺん



## テストプロセスを理解する

---

実務でしっかりとしたテストプロセスを経験できる機会はほとんどないため、一連のテストプロセスの流れを経験し、理解する。

## 得た知識を実務に活かす

---

作業をレビューし合うことで、チームメンバーからも知識を吸収する。再現性のあるテストプロセスを追求し、実務に活かせる知識や気づきを得る。

# 2

## アピールポイントとテストプロセスの全体像



## 再現性のあるテストプロセスの設計

---

各プロセスの手順や成果物の導出過程を詳細に言語化することで、システムの拡張に備えるだけでなく、テストベースが変わっても参考にできるものにした。

## 経験を活かした、テスト要求分析

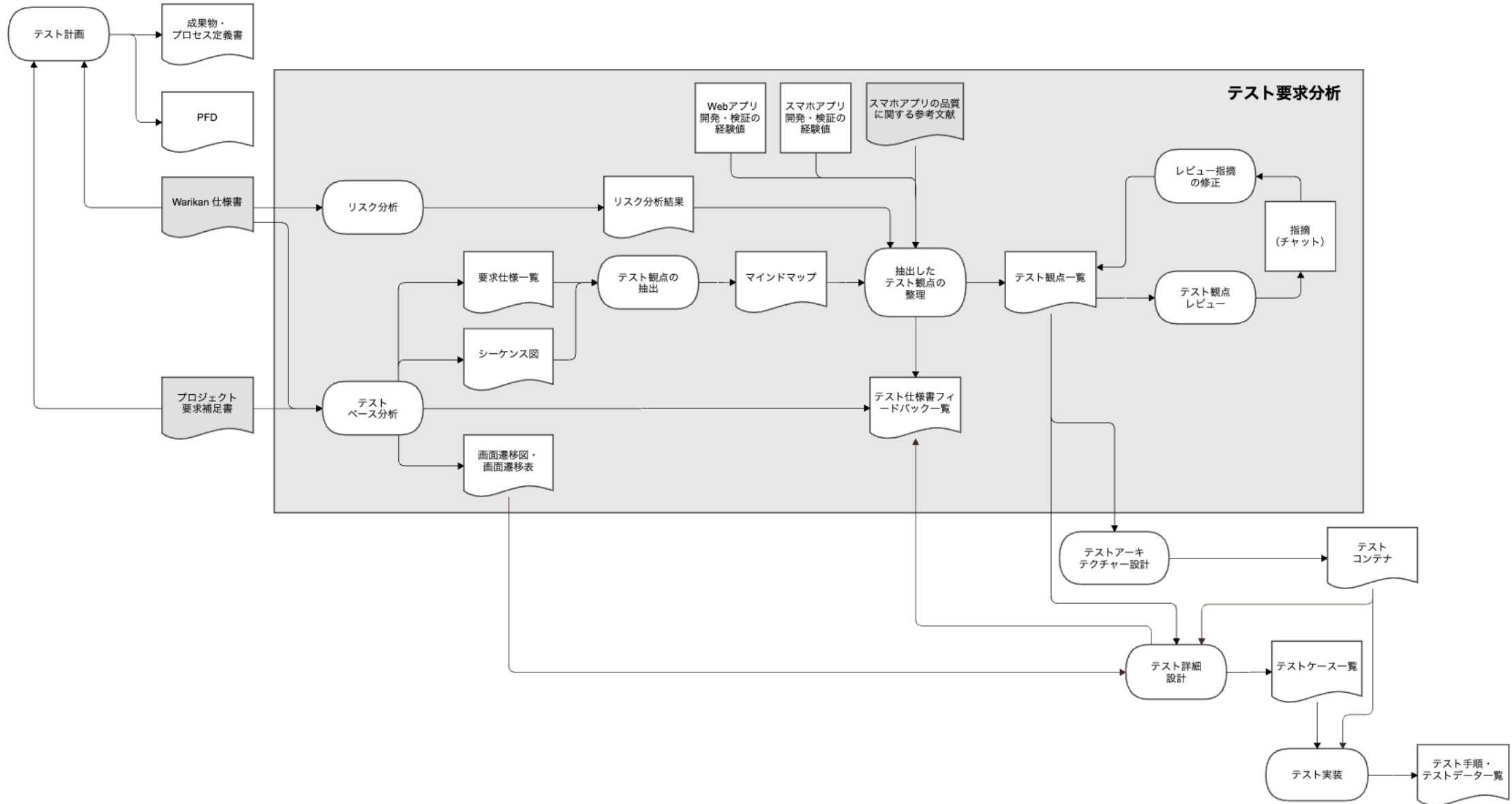
---

メンバーの経験を活かすことで、テスト観点を多面的に出すことができた。

さらに、テストベース仕様書に記載されていない観点を見つけ、

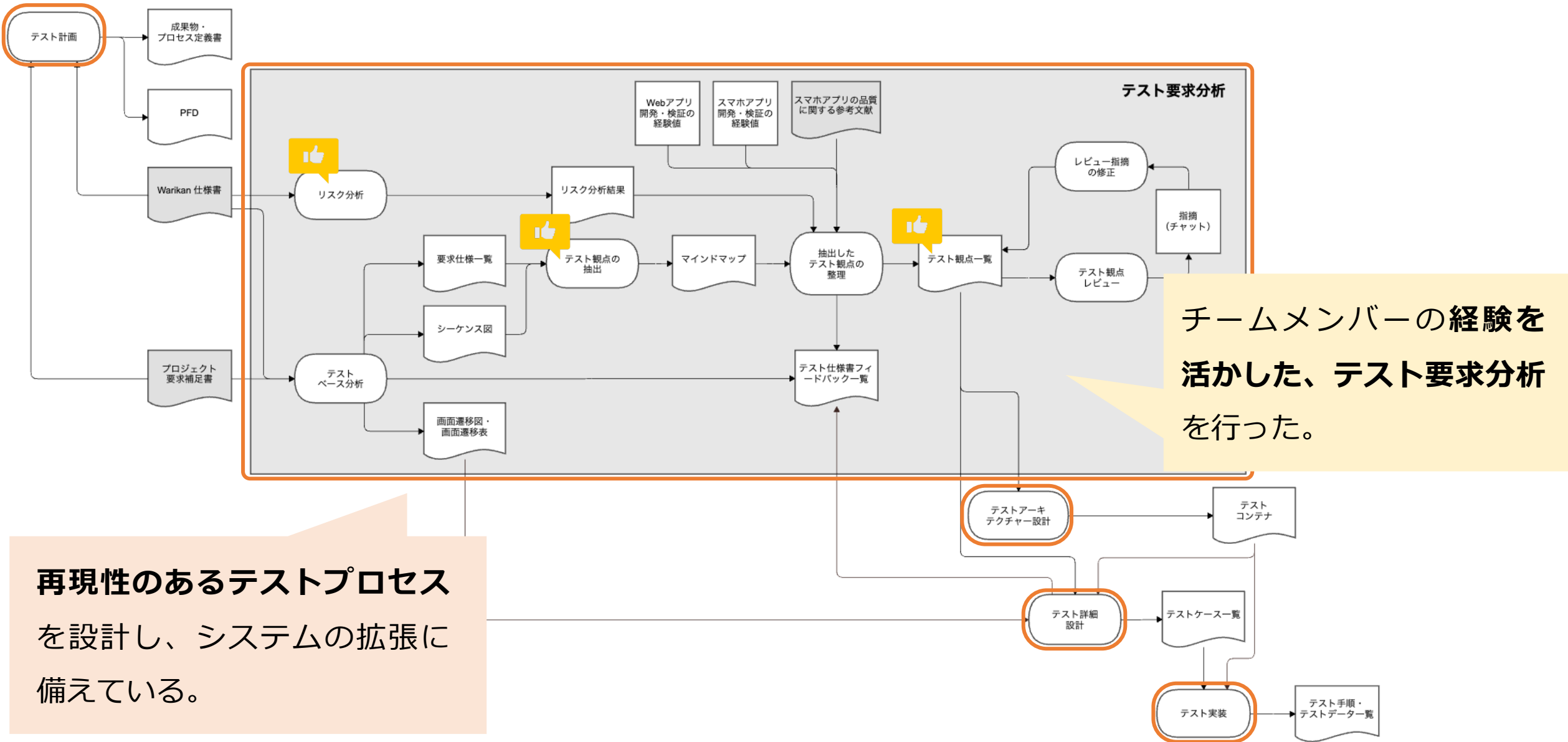
多くのフィードバックをしている。

# プロセスと成果物の全体像





# プロセスと成果物の全体像



# 3 各プロセスの強み

①

テスト要求分析

②

テストアーキテクチャ  
設計

③

テスト詳細設計

④

テスト実装

# 経験を活かした、テスト要求分析

---

メンバーの経験を活かすことで、**テスト観点を多面的に出すことができた。**

## 言語化

---

「わからない」を明確にし、テスト仕様書**フィードバック**一覧にまとめた。

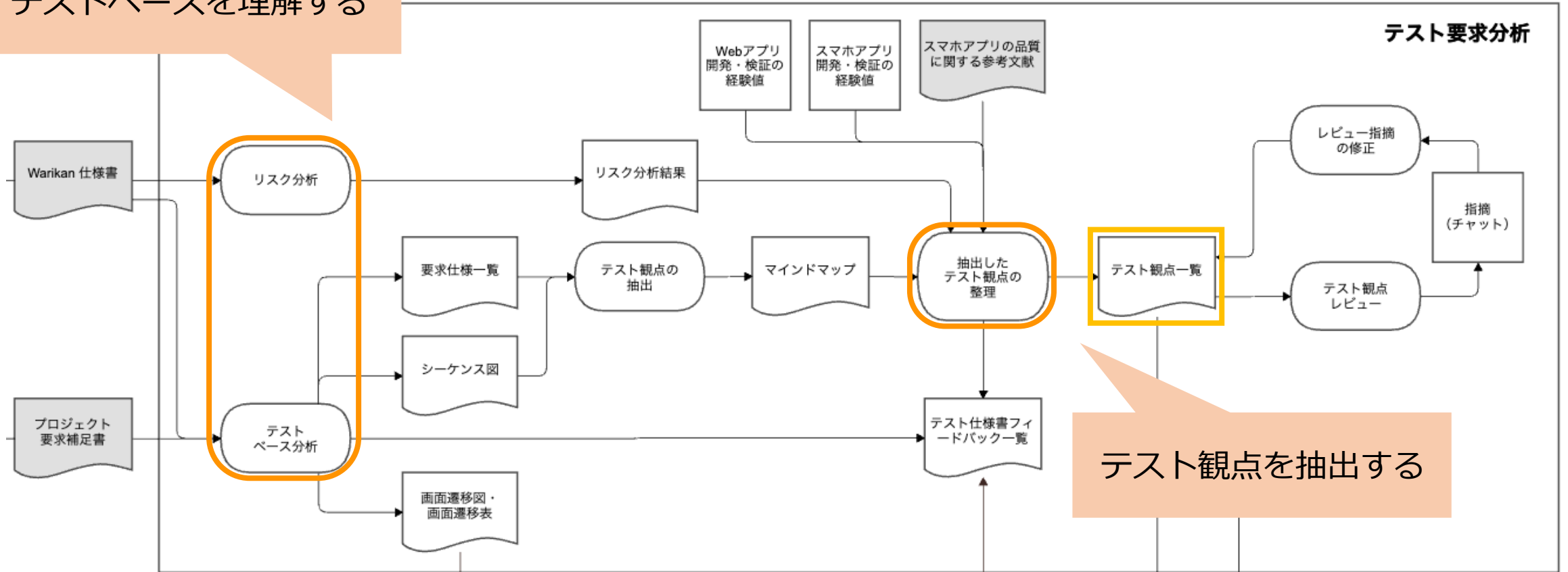
抽出したテスト観点をレビューし合い、意見を言い合うことで、

**網羅性・ピンポイント性**の高いテスト観点となった。



# ①テスト要求分析 の強み

テストベースを理解する



## テストベース分析の手順

1. Warikan仕様書、プロジェクト要求補足書 を読み込む
2. アプリケーションの特徴を整理する
3. Warikan仕様書から、要求仕様を理解する →
4. テストしない箇所を明確にする →
5. アプリケーションの理解を深める →
6. プロダクトリスクを分析する →

**要求仕様一覧**

**シーケンス図**

**画面遷移図・画面遷移表**

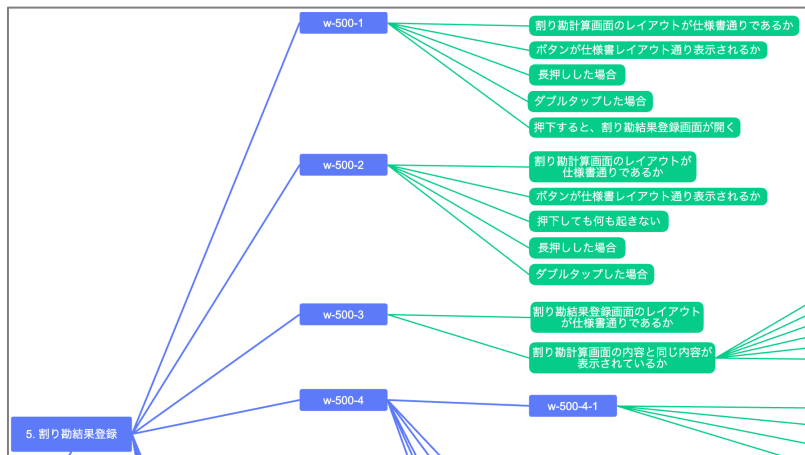
**リスク分析結果**

## 仕様ベースのテスト観点抽出

### 要求仕様一覧

↓ テスト観点の抽出

マインドマップで発散し、  
さらにディスカッションで発散・収束させた。



→  
整理

要求仕様Noと関連付けることで、  
**網羅性**を担保している

		要求仕様No	要求仕様詳細	テスト観点	テストパラメータ	
3.1. ログイン機能	3.1.1. 初期登録	w-311-1	初期画面のログイン画面で「新規登録」ボタンを押すと、新規登録画面へ遷移します。	アプリ起動時の初期画面がログイン画面である		
		w-311-1		ログイン画面で「新規登録」ボタンを押すと新規登録画面へ遷移する		
		w-311-1		ログイン画面のレイアウトが仕様書通りである		
		w-311-1		新規登録画面のレイアウトが仕様書通りである		
			w-311-2	新規登録画面では、アカウントの新規登録を行います。	N/A	
			w-311-3	新規登録画面は次の入力項目を持ちます。	N/A	
			w-311-3-1	「ユーザID」にはアカウントのIDを入力します。テキストフィールドの設定により15文字以内の半角英数字しか入力できません。	ユーザIDには、仕様書に定められたものが入力できる	文字種別 (半角英数字) 文字数 (1文字、15文字)
			w-311-3-1		ユーザIDには、仕様書に定められたもの以外が入力できない	文字種別 (全角文字、半角記号) 文字数 (16文字)
			w-311-3-2	「パスワード」「パスワード (確認用)」にはアカウントのパスワードを入力します。テキストフィールドの設定により20文字以内の半角英数字しか入力できません。	パスワードには、仕様書に定められたものが入力できる	文字種別 (半角英数字) 文字数 (1文字、20文字)
			w-311-3-2		パスワードには、仕様書に定められたもの以外が入力できない	文字種別 (全角文字、半角記号)

↑ テスト観点レビュー

## リスクベースのテスト観点抽出

### リスク分析結果

Webアプリ  
開発・検証の  
経験値

スマホアプリ  
開発・検証の  
経験値

スマホアプリの品質  
に関する参考文献

↓ テスト観点の抽出

リストアップとディスカッションで  
発散・収束させた。

整理  
→

仕様書に明記されている機能仕様以外にも、  
プロダクトリスクが存在しているのでは？

メンバーの経験値を用いて、バグを推測し  
ピンポイントなテスト観点を抽出する

テスト観点No	テスト観点	テストパラメータ
w-NF-TP1	プルダウン内の項目が画面から溢れていても、全てタップして選択できる。	
w-NF-TP2	テキスト入力時に、入力中の文字が見えて入力しやすい。	
w-NF-TP3	テキスト入力でSQLを入力し送信（登録）した場合に、予期せぬデータの閲覧・編集が行われない。	・ 「';delete from user_table;」 ・ 「'-'」
w-NF-TP4	テキスト入力でjavascriptを入力し送信（登録）した場合に、実行されない。	・ 「<script>alert('Bug!!!');</script>」
w-NF-TP5	スペース区切り無しの長文を表示する場合でも、全文が画面に表示される。	
w-NF-TP6	テキスト入力で改行が入力された場合に、予期せぬ表示にならない。	
...	...	...

↑ テスト観点レビュー



## 抽出したテスト観点の整理

テスト観点一覧（仕様ベース）

テスト観点一覧（リスクベース）

追記

追記

テストタイプ・  
テストカテゴリ

優先度

影響度

利用頻度/発生頻度

想定実行時間分類

テスト観点の時点で、想定実行時間とテスト実行範囲を見積ることが可能になる



# 全体像を俯瞰することで、意思決定をサポート

---

テスト観点ごとに定義した、いくつかの分類をコンテナ化。

どういった分類のテストがあるかを、**俯瞰**して確認できることで  
テスト実行対象の決定や、テストプロジェクトの見直しをサポートする。

### テストコンテナの作成

#### テスト観点一覧

テストタイプ・テストカテゴリ

↓ コンテナ化

テストタイプのコンテナ

影響度、機能、テストカテゴリ

↓ コンテナ化

影響度のコンテナ

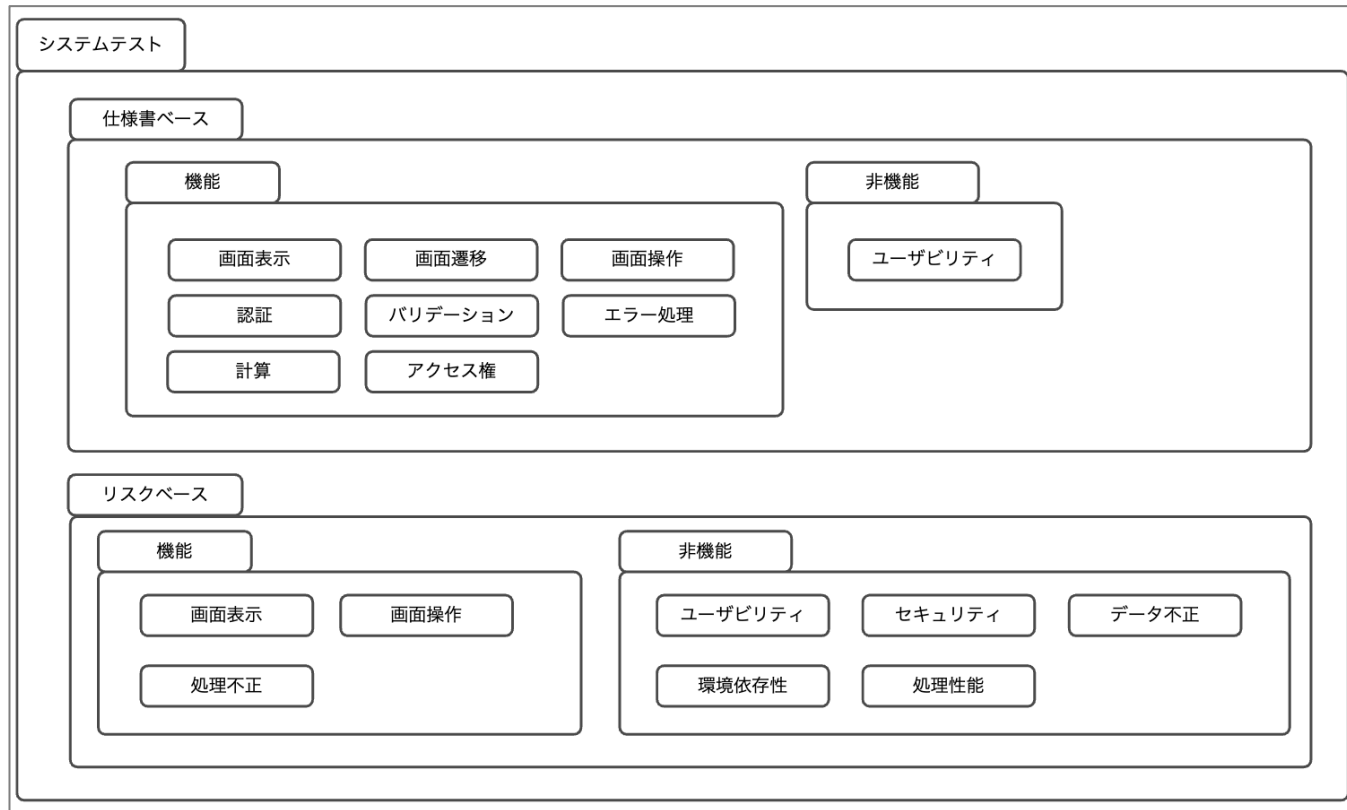
優先度、影響度、利用頻度・  
発生頻度、頻度、母数

↓ コンテナ化

優先度のコンテナ

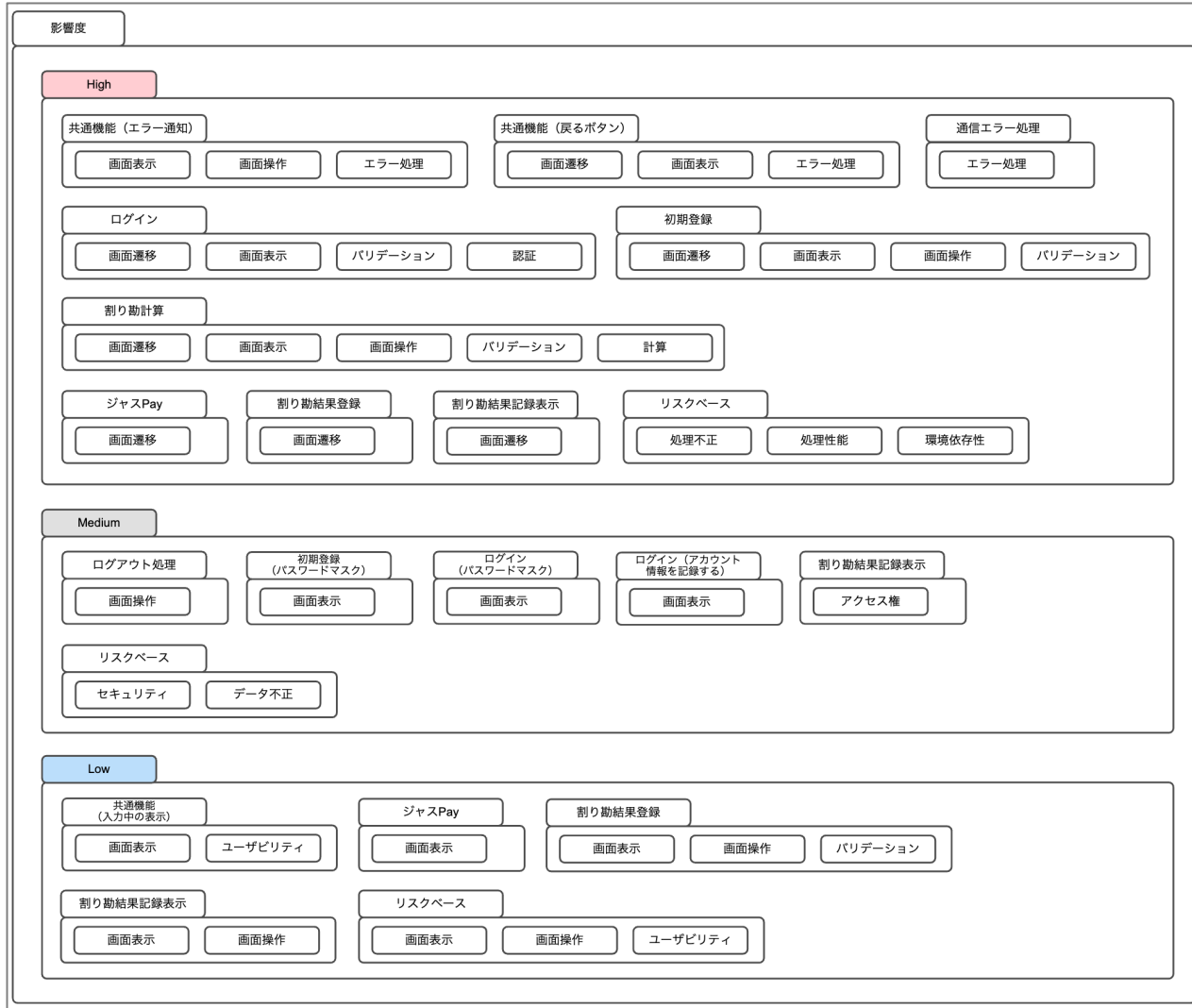
コンテナを分けることで、複雑さを軽減している

### テストタイプのコンテナ



本テストプロジェクトで、こういった  
観点のテストが用意されているかを  
俯瞰して確認できる

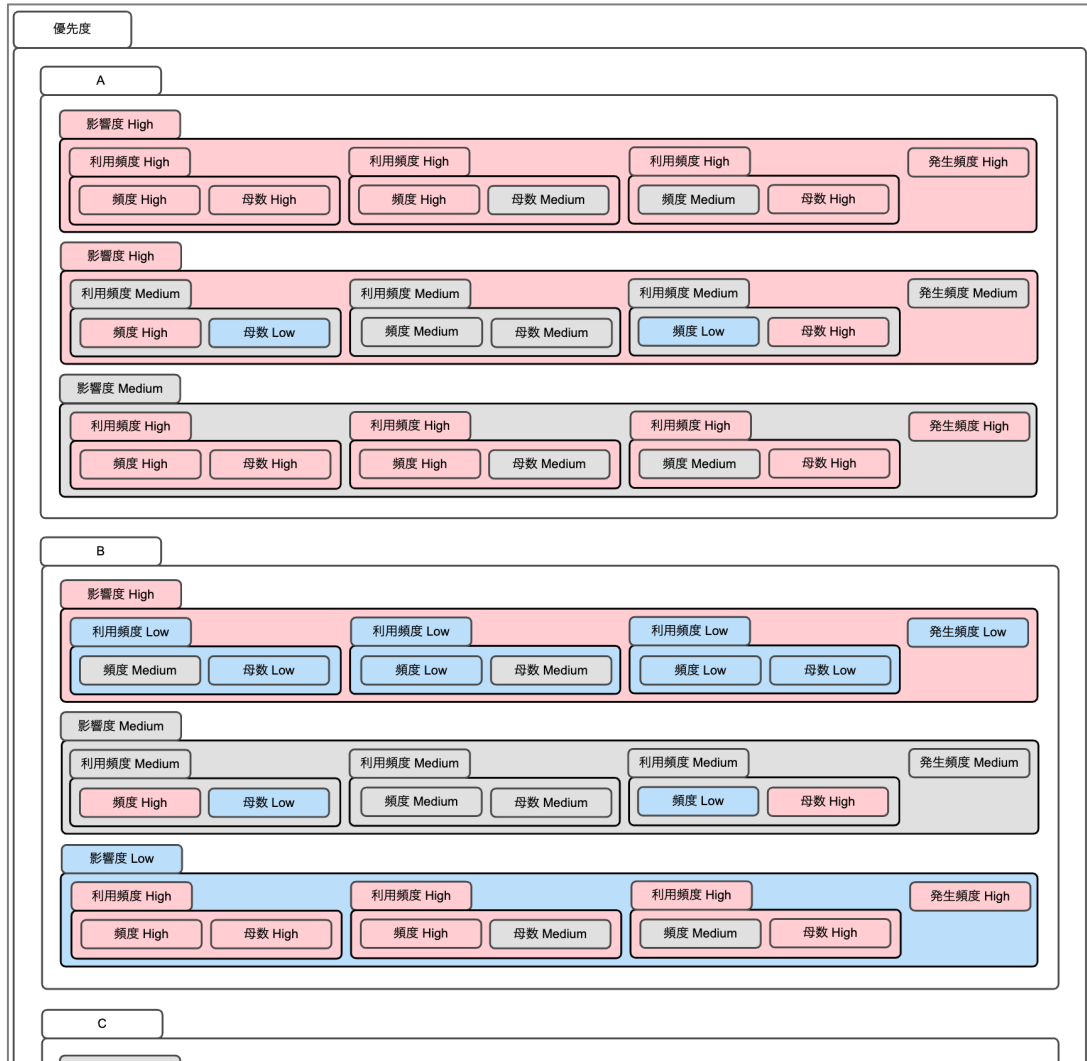
### 影響度のコンテナ



影響度の定義を俯瞰して確認できる。

影響度の見直しや、機能追加などによりテストプロジェクトを見直す際意思決定をサポート

## 優先度のコンテナ



優先度の定義を俯瞰して確認できる。

テスト実行範囲が限られる場合に、どのテストを実行対象とするかを検討する際意思決定をサポート

### 再現性のあるテストプロセスの設計

---

各プロセスの手順や成果物の導出過程を詳細に言語化。  
画面遷移表やデシジョンテーブルなどの、**テスト技法**も利用。

### トレーサビリティ

---

要求仕様Noやテスト観点Noと関連付けることで、**網羅性**の担保を継続した。  
さらに、テスト仕様書フィードバック一覧とのトレーサビリティを取ることで、  
不明点が明確になった際に、テストの見直しがしやすくなってる。

## テストケースの作成

テスト観点一覧

テストコンテナ

画面遷移表

↓ テスト詳細設計

テスト観点ごとに、内容を明らかにしテストケースを作成する。

優先度と想定実行時間を基に、想定実行時間とテスト実行範囲を再度検討する

No	要求仕様No	テスト観点No	テスト観点	条件	内容	予想結果	備考	優先度	想定実行時間 (分)
【概要】									
Warikan仕様書の「5. 割り勘結果登録」に記載の各機能について、各機能の振る舞いがWarikan仕様書と合致していることを確認する。									
TC5-1	w-500-1	w-500-1-TP-2	計算結果が表示された場合に、「結果を登録する」ボタンを押下すると割り勘結果登録画面に遷移する（別紙3のNo.22の実施）	(1)ログインし、割り勘計算画面に遷移する。 (2)計算結果が表示されている。	「結果を登録する」ボタンを押下する。	割り勘結果登録画面に遷移する		B	3
TC5-2	w-500-2	w-500-2-TP-3	計算結果が表示されない場合に、「結果を登録する」ボタンを押下しても何も反応しない（別紙3のNo.17の実施）	(1)ログインし、割り勘計算画面に遷移する。 (2)計算結果が表示されていない。	「結果を登録する」ボタンを押下する。	何も反応しない		B	3
TC5-3	w-500-3、 w-220-1	w-500-3-TP-1、 w-220-1-TP-1	割り勘結果登録画面のレイアウトが仕様書通りである 指定された画面に戻るボタンが表示されている（ここでは割り勘結果登録画面）（TC2-6の実施）	ログインし、割り勘結果登録画面に遷移する。	画面の表示を確認する。	レイアウトが仕様書通りである		C	3
TC5-4	w-500-3	w-500-3-TP-2	割り勘計算画面の内容と同じ内容が表示されている	ログインし、割り勘結果登録画面に遷移する。	表示されている、自分側の人数、相手側の人数、合計金額、支払の割合、自分側の金額/人と相手側の金額/人を確認する。	割り勘計算画面の内容と同じ内容が表示されている		C	3
TC5-5	w-500-4-1	w-500-4-1-TP-1	「日付選択」ボタンを押下すると、カレンダーが開く	ログインし、割り勘結果登録画面に遷移する。	「日付選択」ボタンを押下する。	カレンダーが開く		C	3
TC5-6	w-500-4-1	w-500-4-1-TP-2	カレンダーで今日より未来が選択できない	(1)ログインし、割り勘結果登録画面に遷移する。 (2)「日付選択」ボタンを押下する	今日より未来の日付を選択しようとする。	日付が選択できない		C	3
TC5-7	w-500-4-1	w-500-4-1-TP-3	カレンダーで2年前より過去が選択できない	(1)ログインし、割り勘結果登録画面に遷移する。 (2)「日付選択」ボタンを押下する	2年前の日付を選択しようとする。	日付が選択できない		C	3



## テスト技法の利用

画面遷移図・画面遷移表から導出したテストケースの一覧

	開始画面	入力	アクション	終了画面
1	ログイン画面	新規登録ボタン押し		新規登録画面
2	ログイン画面	ログインボタン押し		[入力情報の認証成功] 割り勘計算画面
3	ログイン画面	ログインボタン押し		[入力情報の認証失敗] ログインエラー画面
4	ログインエラー画面	アプリケーション画面のいずれかをタップ		ログイン画面
5	新規登録画面	戻るボタン押し		ログイン画面
6	新規登録画面	登録ボタン押し (新規登録画面)	[アカウント登録成功] ポップアップを1s表示	[アカウント登録成功] ログイン画面
7	新規登録画面	登録ボタン押し (新規登録画面)		[パスワード*とパスワード(確認用)*に同じ文字列が入力されていない] 新規登録エラー画面(同じ文字列)
8	新規登録画面	登録ボタン押し (新規登録画面)		[各入力項目が1つ以上空] 新規登録エラー画面(空)
9	新規登録画面	登録ボタン押し (新規登録画面)		[アカウント登録処理で入力されたIDが既に登録されていた場合] 新規登録エラー画面(登録済み)
10	新規登録画面	登録ボタン押し (新規登録画面)		[ジャス PayIDがジャス Pay システムサーバで認証されなかった場合] 新規登録エラー画面(ジャスPay)
11	新規登録エラー画面(同じ文字列)	アプリケーション画面のいずれかをタップ		新規登録画面

仕様ベースで抽出したテストケースと対応付けることで、画面遷移のテストケースに不足がないことを確認できる

割り勘計算の組み合わせを表現したデシジョンテーブル

表4 無効な組み合わせを除いたデシジョンテーブル

No		12	13	15	20	21	23	28	29	31
A	自分側の金額(自分側全員)	Y	Y	Y	N	N	N	N	N	N
B	切り上げによる桁数の変化	N	N	N	Y	Y	Y	N	N	N
C	相手側の金額(相手側全員)	Y	N	N	Y	N	N	Y	N	N
D	切り上げによる桁数の変化	N	Y	N	N	Y	N	N	Y	N
E	お釣り	N	Y	Y	N	Y	Y	N	Y	Y
	有効	Y	Y	Y	Y	Y	Y	Y	Y	Y

表5 再採番

No		1	2	3	4	5	6	7	8	9
A	自分側の金額(自分側全員)	Y	Y	Y	N	N	N	N	N	N
B	切り上げによる桁数の変化	N	N	N	Y	Y	Y	N	N	N
C	相手側の金額(相手側全員)	Y	N	N	Y	N	N	Y	N	N
D	切り上げによる桁数の変化	N	Y	N	N	Y	N	N	Y	N
E	お釣り	N	Y	Y	N	Y	Y	N	Y	Y
	具体例									
	自分側の人数	1	1	1	1	1	1	1	1	2
	相手側の人数	1	1	1	1	1	1	1	1	2
	金額	1000	125	1125	200	108	210	1100	210	19990
	支払いの割合	50	80	80	30	50	30	30	70	10
	自分側(切り上げ前)	500	100	900	60	54	63	330	147	1999
	自分側の切り上げ後金額	500	100	900	100	100	100	400	200	2000
	相手側(切り上げ前)	500	25	225	100	8	110	700	10	17990
	相手側の切り上げ後金額	500	100	300	100	100	200	700	100	18000
	自分側	500	100	900	100	100	100	400	200	1000
	相手側	500	100	300	100	100	200	700	100	9000
	お釣り	0	75	75	0	92	90	0	90	10

複雑な計算パターンをデシジョンテーブルで整理することで、網羅的かつ効率的なテストケースを作成することができる

### テスト実行可能なレベル

---

操作手順だけでなく、**テストデータの定義**なども具体的にすることで、  
テスト実行可能なレベルの成果物になっている。

また、テスト実行時の効率を考慮し、テストケースの**実行順序**を工夫した。

### 具体的なテスト実行の提示

---

テスト実行時間を計算し、テスト実行範囲を決定した。

すぐにテスト実行できるように、具体的なテスト実行の**分担**や**使用端末**を提示している。

## ④テスト実装 の強み

### テスト手順一覧・テストデータ一覧の作成

テストケース

テストコンテナ

↓ テスト実装

テストケース一覧に挙げられたテストケースごとに、  
前提条件や手順などの内容を明らかにし、テスト手順を作成する。

同値分割・境界値分析によって、効率の良い  
テストデータを選択している

No	テストケースNo	開始画面	前提条件	手順	予想結果	備考	優先度	想定実行時間 (分)
<b>【概要】</b> Warikan仕様書の「5. 割り勘結果登録」「6. 割り勘結果記録表示」のテストケースについて実行手順を記載する。以下に示す順序で行うことが望ましい。 なお、このシートのテストでは、テストデータは「テストデータ(5,6)」を使用する。								
1	TC5-2	割り勘計算画面	割り勘計算画面に、計算結果が表示されていない。	「結果を登録する」ボタンを押下する。	何も反応しない		B	3
2	TC5-1	割り勘計算画面	-	(1)各入力欄に D5-01 を入力する。 (2)「計算」ボタンを押下する。 (3)「結果を登録する」ボタンを押下する。	割り勘結果登録画面に遷移する		B	3
3	TC5-3	割り勘結果登録画面	-	割り勘結果登録画面のレイアウトを確認する。	レイアウトが仕様書通りである		C	3
4	TC5-4	割り勘結果登録画面	No2 実施後	表示されている、自分側の人数、相手側の人数、合計金額、支払の割合、自分側の金額/人と相手側の金額/人を確認する。	割り勘計算画面の内容と同じ内容が表示されている (D5-01 と一致する)		C	3
5	TC5-10	割り勘結果登録画面	-	表示されている開催日の日付を確認する。	実行環境OSの現在日付になっている		C	3
6	TC5-5	割り勘結果登録画面	-	「日付選択」ボタンを押下する。	カレンダーが開く		C	3
7	TC5-6	割り勘結果登録画面	No6 実施後	今日より未来の日付を選択しようとする。	日付が選択できない		C	3
8	TC5-8	割り勘結果登録画面	No6 実施後	3年前の、昨日と同じ日付を選択しようとする。	日付が選択できない		C	3

## ④テスト実装 の強み

### 実行時間の見積もりとテスト実行範囲の決定

<b>【実行時間とテスト実行対象】</b>
iOS 15およびAndroid 11でテストを実行することを考慮すると、現状の見積もりから想定される実行時間は17.20*2 = 34.40時間（実行時間が不明のテストを除く）であり、24時間を超過している。
そのため、以下のような工夫で実施効率を上げる。
・テスト実行者1人に対して、iOS 15・Android 11の両端末を用意して、同じテストケースを同時に実行する
・その場合の想定実行時間は、上記見積もりの2.0倍ではなく、効率化によって短縮された1.5倍とする
さらに、以下のようにテスト実行対象を絞り込む。
・優先度Cのテスト観点はiOS 15、Android 11のどちらか一方の端末でテストを実行する。
・優先度A,Bのテスト観点はiOS 15、Android 11の両方でテストを実行する。
この場合の想定実行時間は、 <b>23.82 時間</b>
となり、テスト実行者2人で2日（1日6時間相当）のテスト実行期間に収まる。

テスト手順ごとの想定実行時間を合計し、優先度で分ける。  
単純計算では全てのテストを実行することはできないため、  
テストの実施効率を上げる工夫を提示した。

具体的なテスト実行の提示をしている

実行者1		
実行対象	使用端末	想定実行時間(時間)
「3.各機能と操作の仕様」 全件	iOS 15・Android 11	11.03
「リスクベース」 優先度C	Android 11	0.62
	(合計実行時間)	11.64
実行者2		
実行対象	使用端末	想定実行時間(時間)
「5. 割り勘結果登録、6. 割り勘結果記録表示」 優先度A・B	iOS 15・Android 11	2.18
「5. 割り勘結果登録、6. 割り勘結果記録表示」 優先度C	iOS 15	3.35
「リスクベース」 優先度A・B	iOS 15・Android 11	6.65
	(合計実行時間)	12.18

# 4 まとめ



## 再現性のあるテストプロセスの設計

言語化

テスト技法の利用

トレーサビリティ

## 経験を活かした、テスト要求分析

メンバーの経験

不明点の言語化

テスト実行を意識



## テストプロセスを理解する

---

テストプロセスの流れを経験し、  
再現性のあるテストプロセスを追求することで、理解を深めた。

## 得た知識を実務に活かす

---

チームで成果物を作成し、それを評価いただくことで、多くの知識を吸収した。  
再現性のあるテストプロセスを追求し、実務に活かせる知識や気づきを多く得られた。

ご清聴頂きありがとうございました。

つよつよなりたいはんぺん

