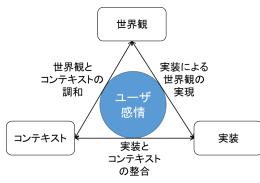


ゼロから始める 「テスト設計」の書 (ゲームテストの世界)

～ゲームテスト三銃士モデル～ …その後の世界は…

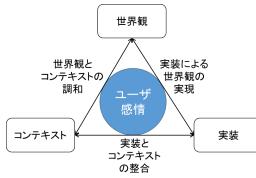


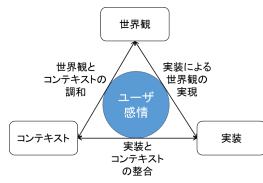
はじめに

はじめに

- 本セッションについての注意事項

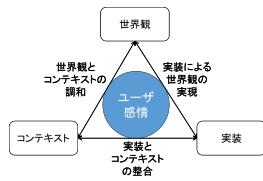
- 三銃士モデルを用いて、試行錯誤している最中の話が、主な内容になります。
- ISO/IEC/IEEE 29119 の話は、ほとんど出てきません。
- ただし…
活動を通して、プロジェクト・組織に「貢献」できそうな、幾つかの気付きがありましたので、是非とも皆さんと共有したいと思います。





改めて
本日の流れ
です！





本日の流れと登壇者紹介

本日の流れ

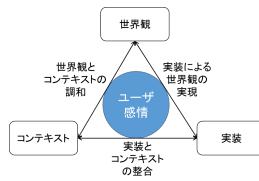
1. 改めて、三銃士モデルの紹介
2. 三銃士モデルの活用
3. プロダクトへの「貢献」（ゲームテストの世界）
4. 事例紹介

登壇者

山本 久仁朗 (YAMAMOTO KUNIO) : アカツキ

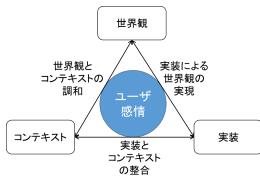


本日の流れと登壇者紹介



- 名前 : 山本 久仁朗(旧姓:村上)
- 最終学歴 : 一関高専 化学工学科 (~'89)
- 職歴 :
 - 日本プロセス株式会社 (Sier : SE、TE) (~'03)
 - アットネットホーム株式会社 (ISP : QA) (~'05)
 - ソニーデジタルネットワークアプリケーションズ株式会社 (PC／組込 : TE、SQA) (~'11)
 - NHN Japan 株式会社 (ゲーム : QA) (~'13)
 - DeNA 株式会社 (ゲーム／Web : QA) (~'15)
 - 株式会社 メルカリ (フリマアプリ : QA) (~'16)
 - 株式会社 mediba (Web／アプリ : QA) (~'17)
 - 株式会社 アカツキ (ゲーム／Web : QA) (現職)

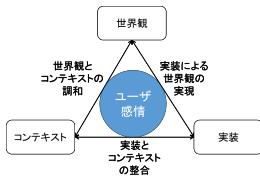




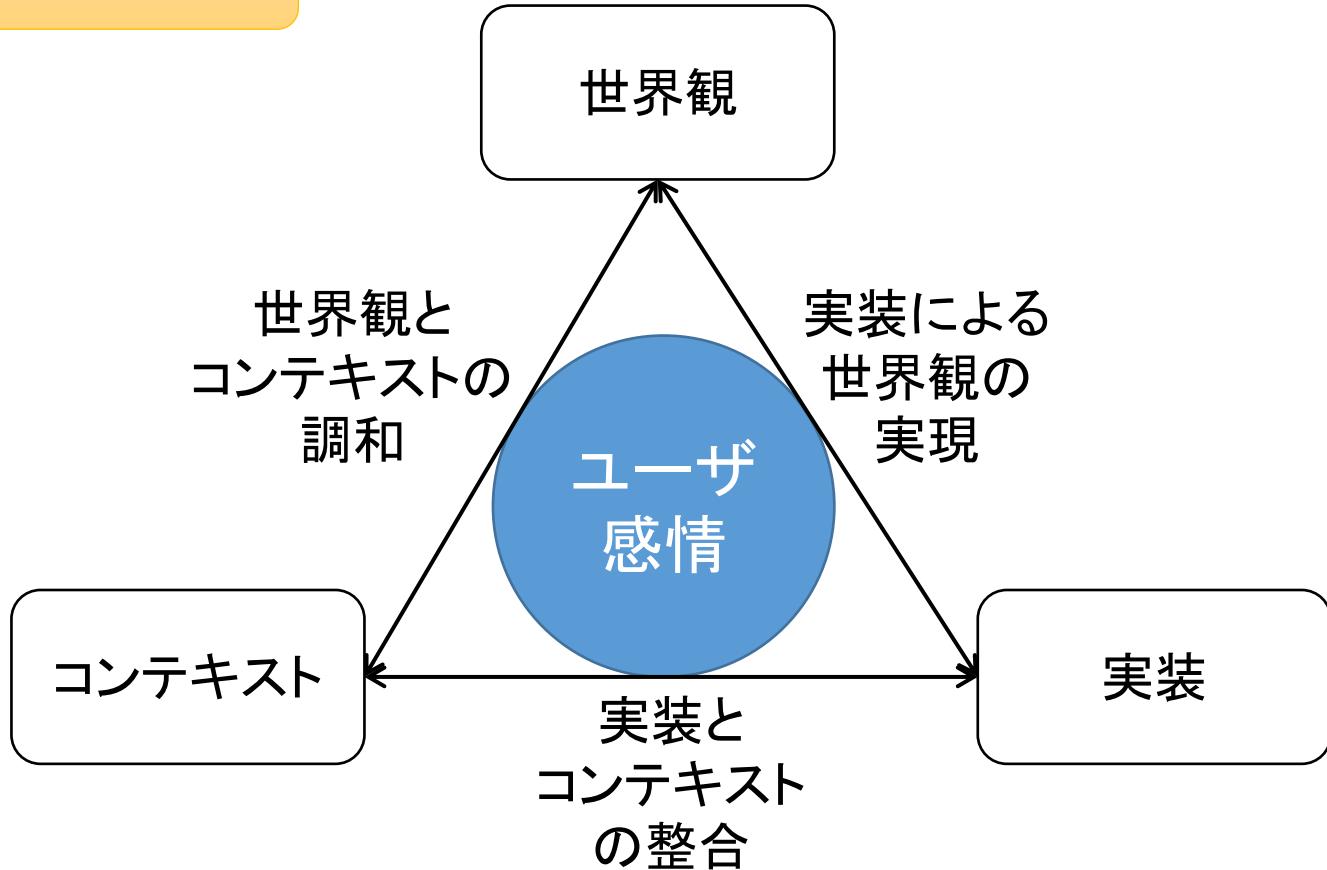
改めて 三銃士モデル の紹介



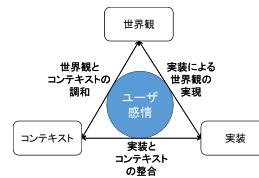
ゲームテストのためのテスト観点モデル



“三銃士モデル”



“三銃士モデル”に基づいたテスト観点の例



世界觀

- 全体のストーリー
 - レベルデザイン
 - 歴史考証/科学考証

実装

- 機能 / 操作
 - グラフィック / サウンド
 - 音楽 / ムービー
 - スピード

コンテキスト

- ビジネスモデル
 - マーケット / ユーザ
 - 競合との比較

実装による世界観の実現

- ## ➤ オブジェクト同士の めりこみの程度

世界観とコンテキストの調和

- 想定したユーザ層に対しての全体のストーリーの説明と理解のバランス

実装とコンテキストの整合

- 想定したマーケットのネットワーク環境におけるデータダウンロード時間

世界観に対するユーザ感情

- ## ➤ ワクワク感

実装に対するユーザ感情

- ## ➤ 没入感

コンテキストに対するユーザ感情

- ## ➤ 中毒性

実装による世界観の実現に対するユーザ感情

- ## ➤ 現実感

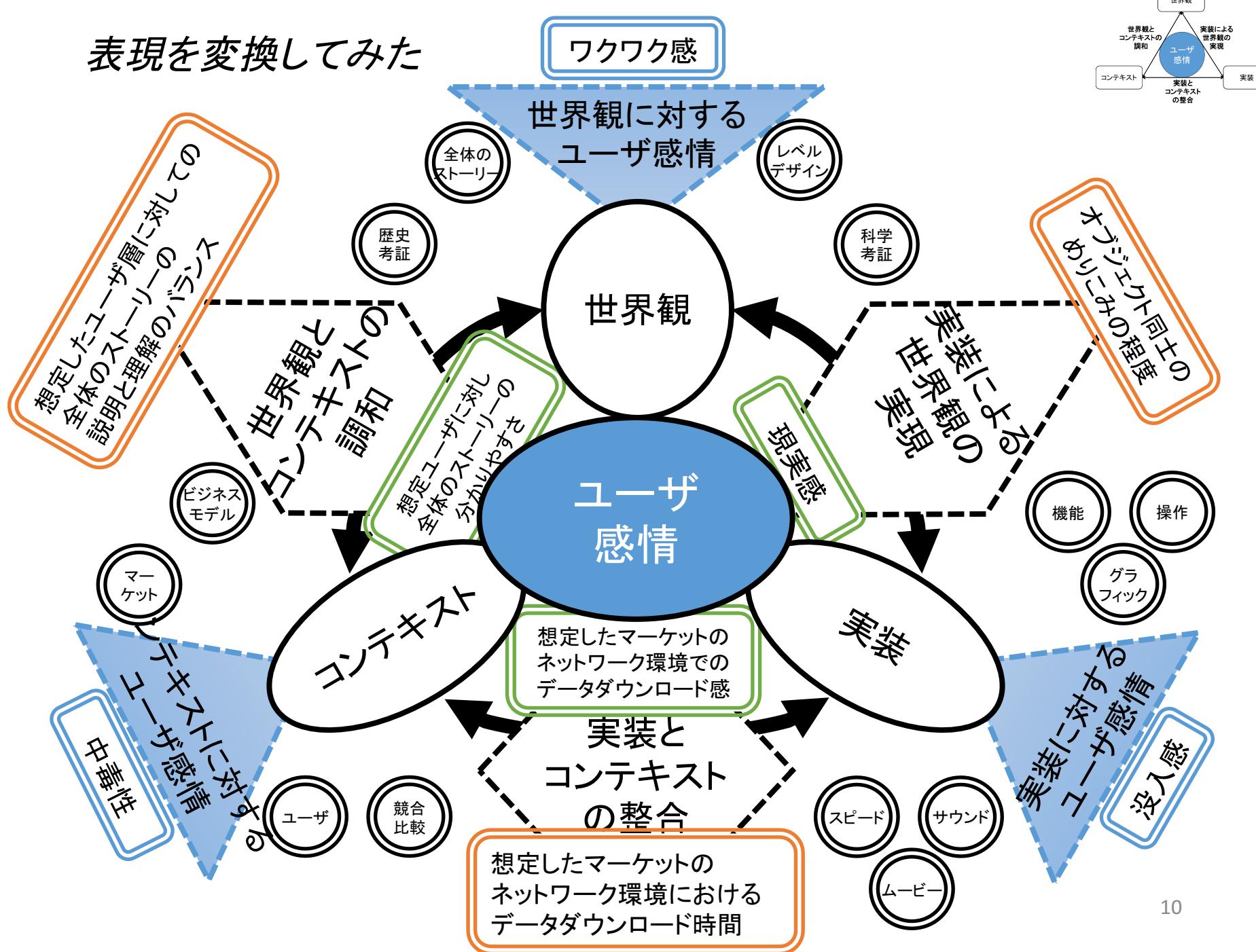
世界観とコンテキストの調和 に対するユーザ感情

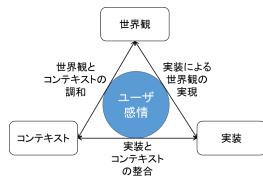
- 想定したユーザ層に対しての全体のストーリーの分かりやすさ

実装とコンテキストの整合に対するユーザ感情

- 想定したマーケットのネットワーク環境におけるデータダウンロードの際のイライラ感

表現を変換してみた

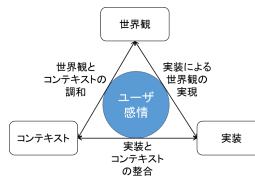




「三銃士モデル」 を活用して 何を目指したか？



三銃士モデルで、何を目指したか？



“ゲームテストをする人が 楽ができるようにしたい”

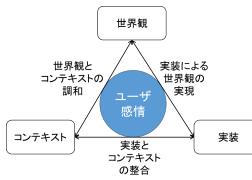
「2016年04月06日 研究会議事録」から抜粋

ゲームテストの現状から、より「難しく」「責任重大」になってしまっているが、その複雑さと厳しい条件下(期間・工数・テストベースの少なさ)でのテストが要求されていることは少なくない。

そのため、テスト設計・実施がベテラン・熟練テスター主導によるOJTでの人依存な業務実施・人材育成になっていた。

大規模・複雑化している案件に対応するためには、テスト設計方法を確立し、抜け漏れがなく効率的なテスト実施とより効果的な人材育成を促進したい。

三銃士モデルで、何を目指したか？



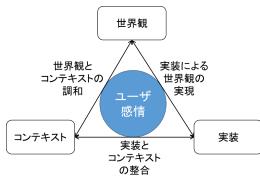
“ゲームテストをする人が 楽ができるようにしたい”

「2016年04月06日 研究会議事録」から抜粋

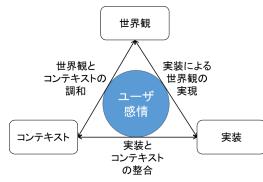
ゲームテストの現状から、より「難しく」「責任重大」になってしまっているが、その複雑さと厳しい条件下(期間・工数・テストベースの少なさ)でのテストが要求されていることは少なくない。

そのため、テスト設計・実施が**ベテラン・熟練テスター主導によるOJTでの人依存な業務実施・人材育成**になっていた。

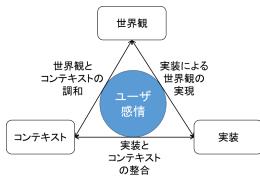
大規模・複雑化している案件に対応するためには、**テスト設計方法を確立し、抜け漏れがなく効率的なテスト実施**と**より効果的な人材育成を促進**したい。



ゲームテストは、
複雑化 & 責任重大
& (未成熟 or
人材不足) !!



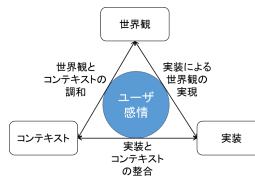
ここでの知見の
フィードバックが、
SW テスト業界に
「貢献」になる！？



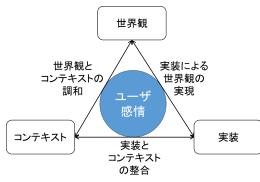
「三統一モデル」 で 何をしたいか？



三銃士モデルで、何をしたいか？

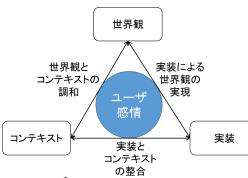


- テスト観点を構造化する
 - 暗黙的だった観点のカテゴライズを明確化
 - 観点構成を理解・分解・再構築しやすく
- テスト設計方法の提案
 - 観点の組合せを用いた、テスト設計方法の提案
 - 観点の組合せは、コンテンツに合わせてカスタマイズ
 - 観点・リスクから優先度付けして、アジャスト可能にしたい
 - まずはシンプルに、段階的な活用方法の提案
 - 観点・組合せの可視化で、ノウハウ伝授を促進
- テスト合否判定にユーザー観点を追加
 - ユーザー感情を判定材料とする、ユーザー観点でのテスト
 - ユーザー観点での判定を、テスター熟練度に依存せず実施
 - 多角的な観点でのテスト実施 & 設計の実施



予稿集の
資料と紐付け
てみましょう！

プロダクトへの「貢献」（ゲームテストの世界）



Step1：
単独テスト観点で実施
(機能・世界観等)

基本的なテスト実施により
不具合発見に貢献

Step2：
テスト観点の洗い出し

テスト観点の抽出により
不具合発見の抜け漏れ防止に貢献

Step3：
テスト観点のカテゴライズ

テスト観点のカテゴライズにより
ズームイン・ズームアウト可能
テストケースの最適化に貢献

Step4：
テスト観点を組合せて実施

テストフレーム(仮)の活用により
テスト観点の組合せ最適化し
テストケースの最適化に貢献

Step5：
テスト観点として
ユーザー感情の導入

ユーザー視点での改善提案による
施策の魅力向上に貢献

Step6：
総合的な観点での
テスト実施

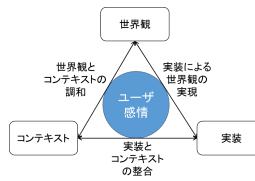
世界観・実装・コンテキストのバランスを
ユーザー感情を軸にテストすることにより
プロダクトの魅力向上に貢献

不具合発見
に貢献！

テスト最適化
に貢献！

魅力向上
に貢献！

三銃士モデルで、何をしたいか？



- テスト観点を構造化する

Step2-3

暗黙的だった観点のカテゴライズを明確化
観点構成を理解・分解・再構築しやすく

- テスト設計方法の提案

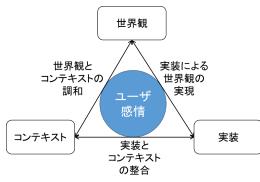
Step4

観点の組合せを用いた、テスト設計方法の提案
観点の組合せは、コンテンツに合わせてカスタマイズ
観点・リスクから優先度付けして、アジャスト可能にしたい
まずはシンプルに、段階的な活用方法の提案
観点・組合せの可視化で、ノウハウ伝授を促進

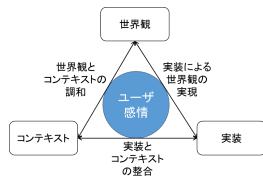
- テスト合否判定にユーザー観点を追加

Step5

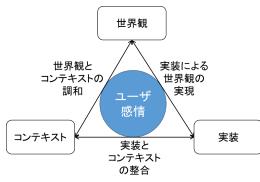
ユーザー感情を判定材料とする、ユーザー観点でのテスト
ユーザー観点での判定を、テスター熟練度に依存せず実施
多角的な観点でのテスト実施 & 設計の実施



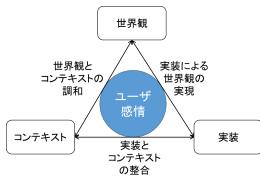
「貢献」について
後ほど予稿集
の資料を使って
お話しします！



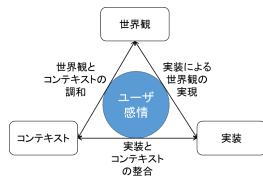
みなさんには
質問です。



テストで
設計して
いますか？

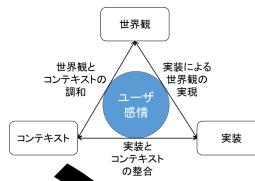


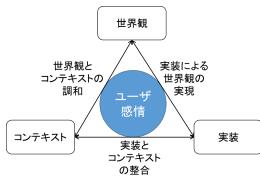
テスト設計で
どんな活動
しますか？



テスト設計の 成果物は なんですか？

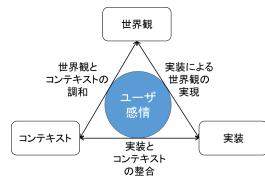
多くの組織が
人依存になり
設計方法が
不明確では？





改めて 設計とは 何でしょう？





改めて、三銃士モデルの紹介

・ 設計とは

- 建築物や工業製品等といったシステムの具現化のため、必要とする機能を検討するなどの準備であり、その成果物としては仕様書や設計図・設計書等、場合によっては模型などを作ることもある。
- 時間的なものとしては、製作開始時期・供用開始時期・想定使用期間・量産品の場合の製作継続期間などがある。
機能的なものとしては、使用目的・体積・面積・質量などがある。
社会的な状況・環境としては、法的規制・地球環境などがある。試作は、仕様の検討のための手段であり、納得または合意するまで試作を繰り返す。使用できる資源の組み合わせは多くある。

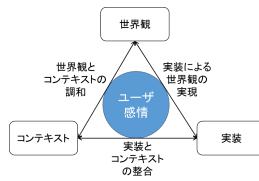
ライフサイクルコスト(生涯費用)：調達可能な金額であるか。

また、保守、廃棄を含めて、一番安く済む手法であるか。機械・器具・部品など：製作開始までに調達可能であるか。量産品の場合の製作継続期間中に途切れることなく調達可能であるか。

保守：想定使用期間の間、保守に必要な部品・器具・作業員などが確保できるか。製造・試験・保守・廃棄作業を考えると、使用できる組み合わせが少ないことがある。

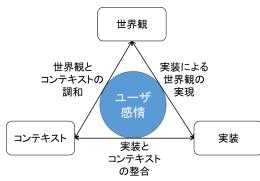
設計に使用される手法として、ウォーターフォールモデル、プロトタイプモデル、スパイラルモデル、コンカレントエンジニアリング、実験計画法、などがある。

改めて、三銃士モデルの紹介



・ ソフトウェア設計とは

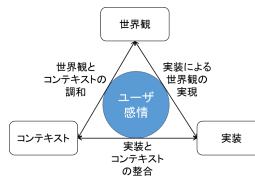
- ・ ソフトウェアのための問題解決と計画の工程である。ソフトウェアの目的と仕様が決定した後で、ソフトウェア開発者が設計をしたり、専門の設計者が開発計画を立てる。細かいコンポーネントやアルゴリズムの実装だけではなく、アーキテクチャ的観点での検討も行われる。
- ・ ソフトウェア開発工程での要求分析によって、ソフトウェア工学における仕様が確定する。そのソフトウェアがユーザーとの対話を必要とするものか、あるいはユーザー中心設計であれば、ソフトウェア設計にはユーザーエクスペリエンス設計も関わり、絵コンテなども仕様に含まれることになる。
- ・ ソフトウェア設計文書により、プログラミングを開始する前に制約条件、要求仕様などが明らかとなり、検討される。シミュレーションやプロトタイプの構築によって設計変更となる場合もある。要求分析や計画立案せずに、プログラミングしながら設計することも可能だが、複雑な大規模プロジェクトではそのような工程の省略は選択肢として考慮されないのが普通である。プログラミングの前に設計工程を置くことで、ソフトウェアの対象領域の専門家と設計者がプログラマと共同で作業し、ソフトウェアの利便性と技術的正当性を高めることができる。



三銃士モデルの活用



三銃士モデルの活用



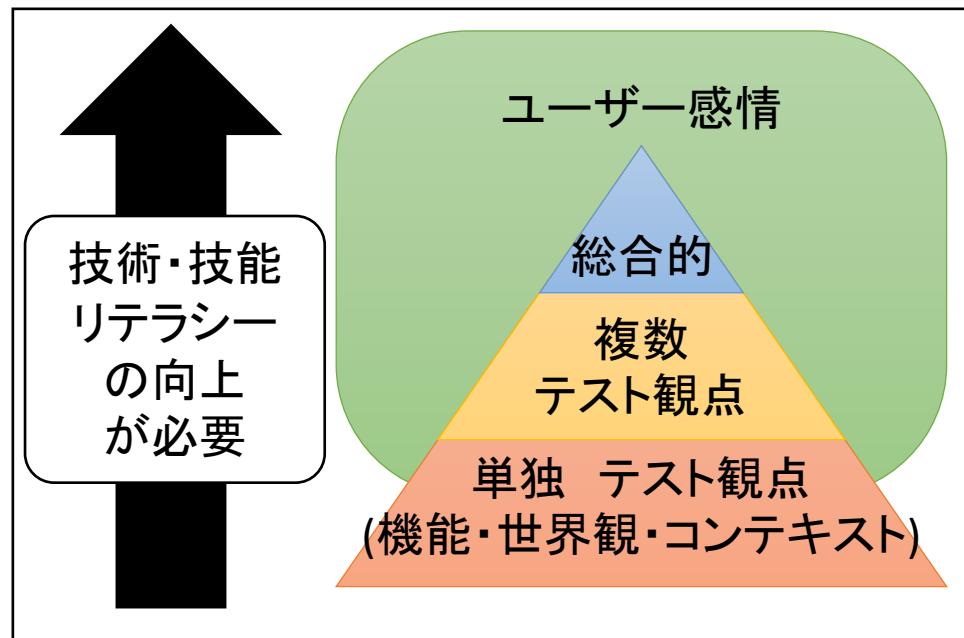
• オススメ Step

- Step1 : 単独テスト観点で実施(機能・世界観等)
- Step2 : テスト観点の洗い出し
- Step3 : テスト観点のカテゴライズ
- Step4 : テスト観点を組合せて実施
- Step5 : テスト観点としてユーザー感情の導入
- Step6 : 総合的な観点でのテスト実施

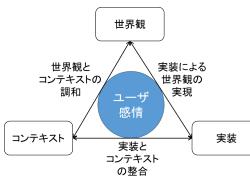
* テスト観点とは

テスト対象に対して、「何をテストすればいいか」という関心事を「テスト観点」とする

機能・仕様・パラメータ等の直接ソフトウェアに関連するものやプラットフォーム・構成・外部環境等の間接的なものがある



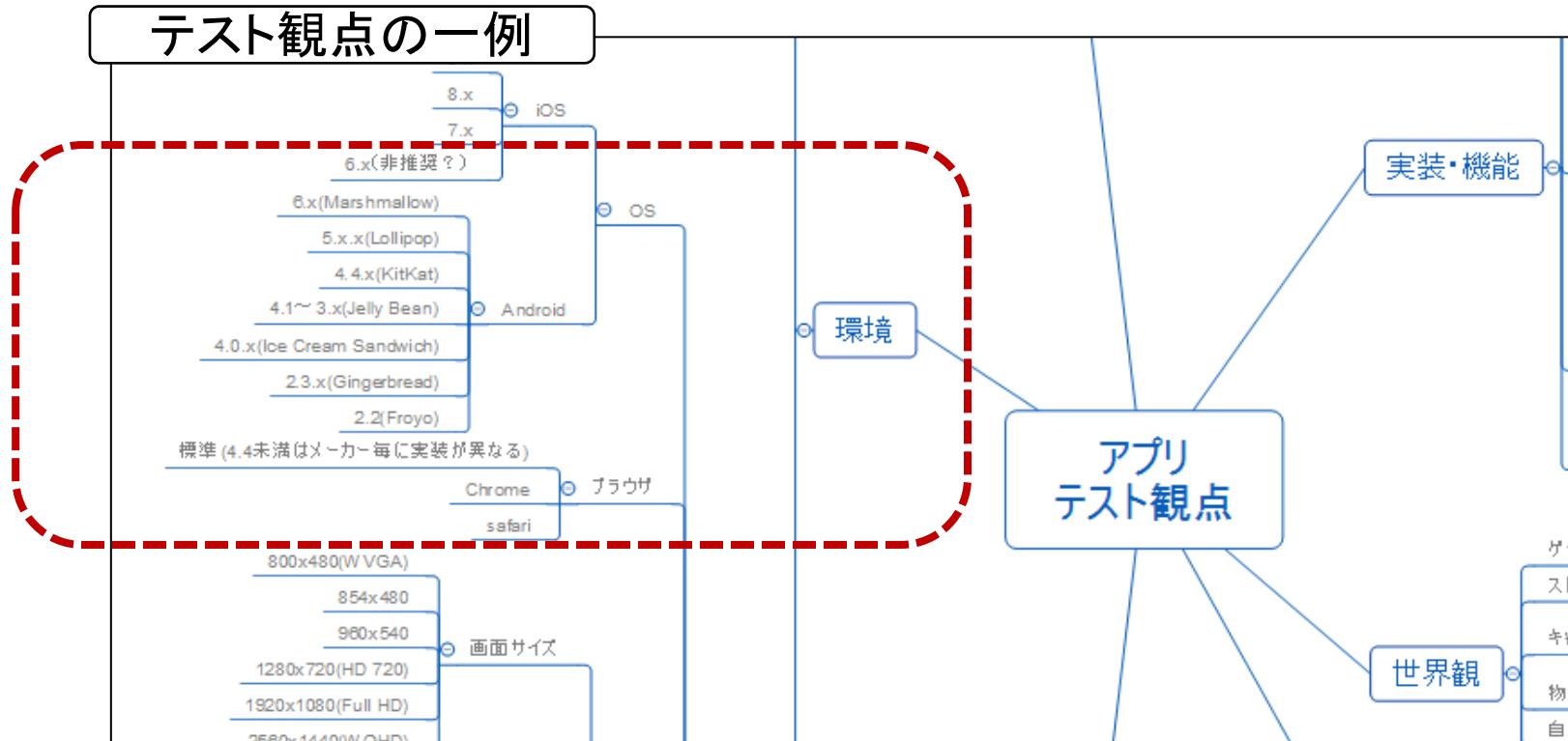
三銃士モデルの活用



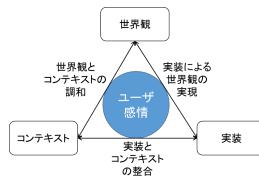
• Step2～3：テスト観点の構造化

• 観点構成を理解・分解・再構築しやすく

- 一例として、マインドマップ・構造ツリー等で構造化する
(各個人・組織によって、使いやすいツールでまとめる)
- 構造化すれば、分解(詳細化)・再構築(組換)が容易になる
- 体系的に分類することが、その後の再利用性を高めます



三銃士モデルの活用

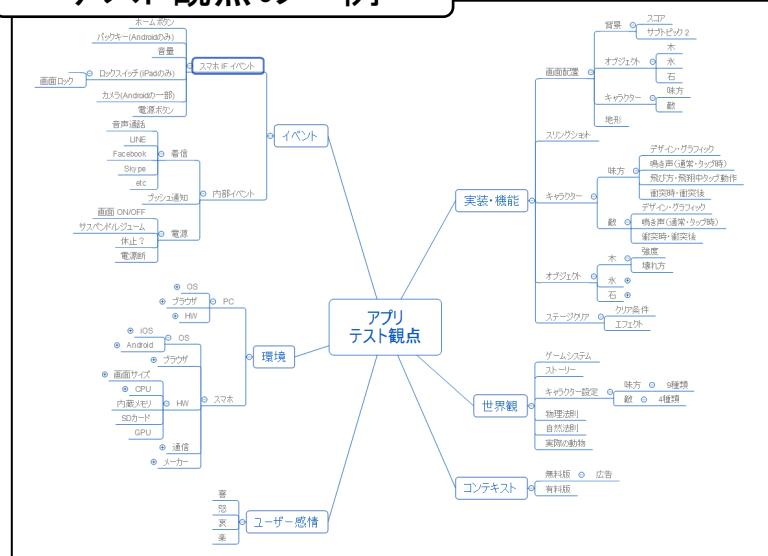


- Step4 : テスト観点を組合せて実施

- ・ 観点・組合せの可視化で、ノウハウ伝授を促進

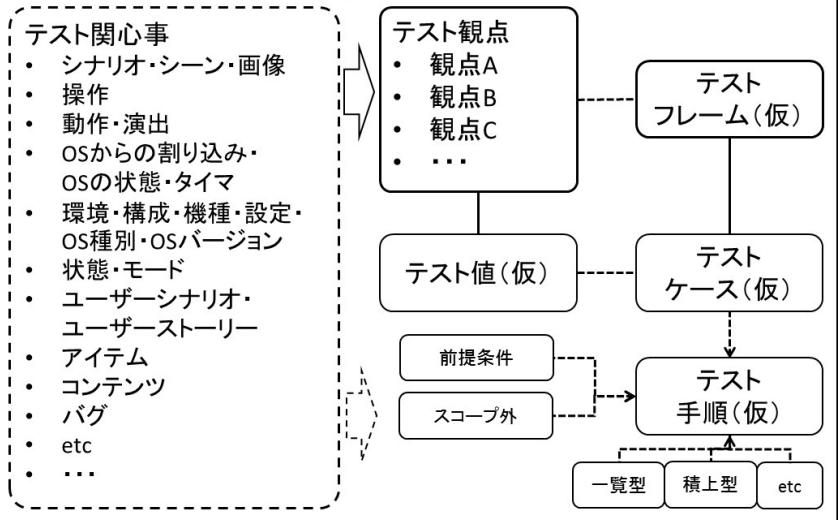
- テスト設計の活動を通じ、観点・組合せの可視化を行うことで、ベテランテスト設計者・実施者のノウハウを、若手のテストオペレーター・ビギナーテスト設計者への伝授を促進させるためのツールの一つしたい

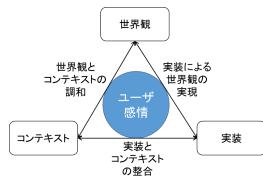
テスト観点の一例



- ・ テスト設計の流れ(例)

- テスト関心事
 - シナリオ・シーン・画像
 - 操作
 - 動作・演出
 - OSからの割り込み・OSの状態・タイマ
 - 環境・構成・機種・設定・OS種別・OSバージョン
 - 状態・モード
 - ユーザーシナリオ・ユーザーストーリー
 - アイテム
 - コンテンツ
 - バグ
 - etc
 - ...



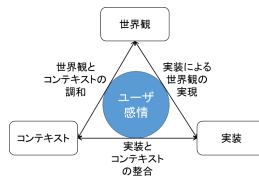


三銃士モデルの活用

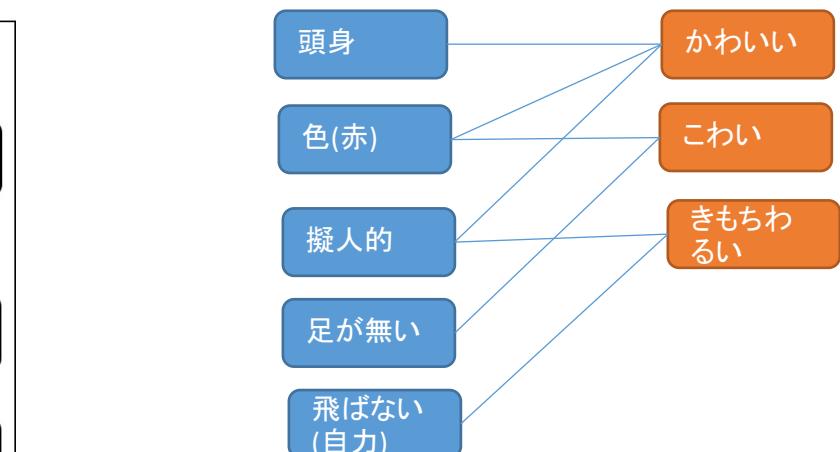
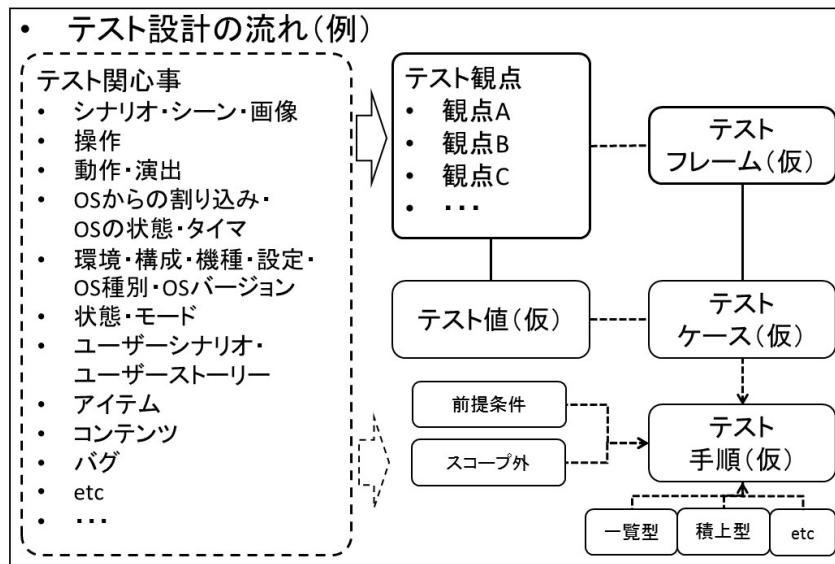
- Step5：テスト観点としてユーザー感情の導入
 - ユーザー感情を判定材料とする、ユーザー観点でのテスト
 - ゲームテストのシステム・総合テストにおいて、機能が正しくてもユーザー観点から見て、合格と判定しがたいものがあると思います。ユーザー感情を明確に分析し合否判定材料とすることでユーザー観点でのテストを増やし、フェーズに応じて優先度を高く実施することで、ユーザーファーストな、テストを促進したい

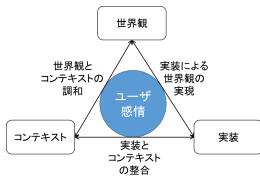
区分	フィーチャー	世界観	詳細	確認内容	ユーザー感情	
主人公が キャラ設定	デフォルメ (非リアル)		鳥であること	鳥と認識できること (くちばし、羽、鳴き声…)		
			頭身	1頭身であること	かわいい	丸いものはかわいい
			色(赤)	体が赤であること(くちばし・顔のバツは別)	かわいい	小さい子は赤ければ、かわいいと思う
					こわい	血の色を連想させる
			擬人的	顔が人間っぽい(眉毛・歯・表情) 仕草(腕組み)	かわいい	人間らしい
					きもちわるい	ぶさいく、人面的
			足が無い	足がないこと	かわいい	丸いものはかわいい
					きもちわるい	非現実的？
			丸い	丸いこと	かわいい	丸いものはかわいい
			飛ばない (自力)	飛び回っていない 重力に逆らっていない	かわいい	翻弄されている？面倒をみたくなる？

三銃士モデルの活用



- Step6：総合的な観点でのテスト実施
 - 多角的な観点でのテスト実施 & 設計を通じて、人材育成
 - ベテランのテスター・テストエンジニア・テストマネージャの、ナレッジ・ノウハウを効率的にトランスファーするために、下記のように共通認識を増やし、ロジカルな人材育成を促進
 - テスト観点の可視化・洗い出し・カテゴライズを通じて、知識を共有
 - テスト設計方法と実施を通じて、考え方を共有

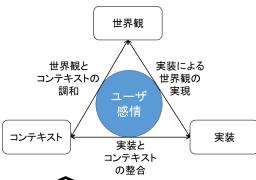




プロダクトへの「貢献」 (ゲームテストの世界)



プロダクトへの「貢献」（ゲームテストの世界）



Step1：
単独テスト観点で実施
(機能・世界観等)

基本的なテスト実施により
不具合発見に貢献

Step2：
テスト観点の洗い出し

テスト観点の抽出により
不具合発見の抜け漏れ防止に貢献

Step3：
テスト観点のカテゴライズ

テスト観点のカテゴライズにより
ズームイン・ズームアウト可能
テストケースの最適化に貢献

Step4：
テスト観点を組合せて実施

テストフレーム(仮)の活用により
テスト観点の組合せ最適化し
テストケースの最適化に貢献

Step5：
テスト観点として
ユーザー感情の導入

ユーザー視点での改善提案による
施策の魅力向上に貢献

Step6：
総合的な観点での
テスト実施

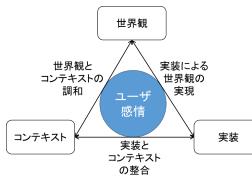
世界観・実装・コンテキストのバランスを
ユーザー感情を軸にテストすることにより
プロダクトの魅力向上に貢献

不具合発見
に貢献！

テスト最適化
に貢献！

魅力向上
に貢献！

プロダクトへの「貢献」（ゲームテストの世界）



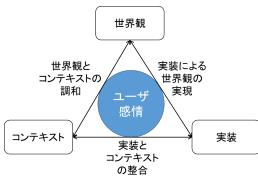
Step1：
単独テスト観点で実施
(機能・世界観等)

基本的なテスト実施により
不具合発見に貢献

- Step1：単独テスト観点で実施(機能・世界観等)
 - アプローチ：
 - 企画書・仕様書(機能・画面)・機能一覧・メニュー一覧等のテストベースをもとに、単独テスト観点でシステムテスト実施
 - 初歩的なテストアプローチだが、ゲームテストの現場では、いまだに多数実施されている
 - 貢献・効果：プロダクトの品質向上
 - 状況により、早い段階で、多くの不具合が発見される
 - QCDのバランスにより、成熟度が非常に若いプロジェクト・プロダクトでは、とても有効である
 - 基本的な動作確認をするだけならば、概ね有効である

…ゲームテストの現場では、納品物が動かないことも…
…少なくありません…

プロダクトへの「貢献」（ゲームテストの世界）



Step2：
テスト観点の洗い出し

テスト観点の抽出により
不具合発見の抜け漏れ防止に貢献

- Step2：テスト観点の洗い出し

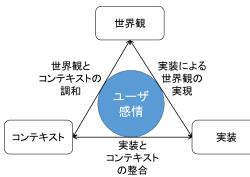
- アプローチ：

- テスト観点を複数メンバーで洗い出す
 - 企画・設計とのレビュー等でテスト観点の抜け漏れを確認
 - 市場バグ等も再発防止の一環としてテスト観点に追加

- 貢献・効果：プロダクトの品質向上

- テスト観点漏れのない状態で、考慮漏れを防ぐ
 - テスト観点でレビュー実施することで、テスト手順よりも全体を把握しやすいために、抜け漏れがわかりやすい

プロダクトへの「貢献」（ゲームテストの世界）



Step3：
テスト観点のカテゴライズ

テスト観点のカテゴライズにより
ズームイン・ズームアウト可能
テストケースの最適化に貢献

- Step3：テスト観点のカテゴライズ

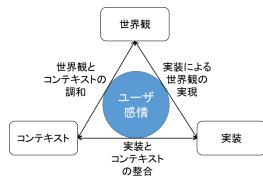
- アプローチ：

- Step2で洗い出した観点を機能・画面・属性・環境等の似たようなグループでカテゴライズする
- カテゴライズしたグループ内で、観点漏れがないか確認
- カテゴライズしたグループをベースに、網羅・一定基準で選定・代表値等、目的に応じて最適な粒度を選択

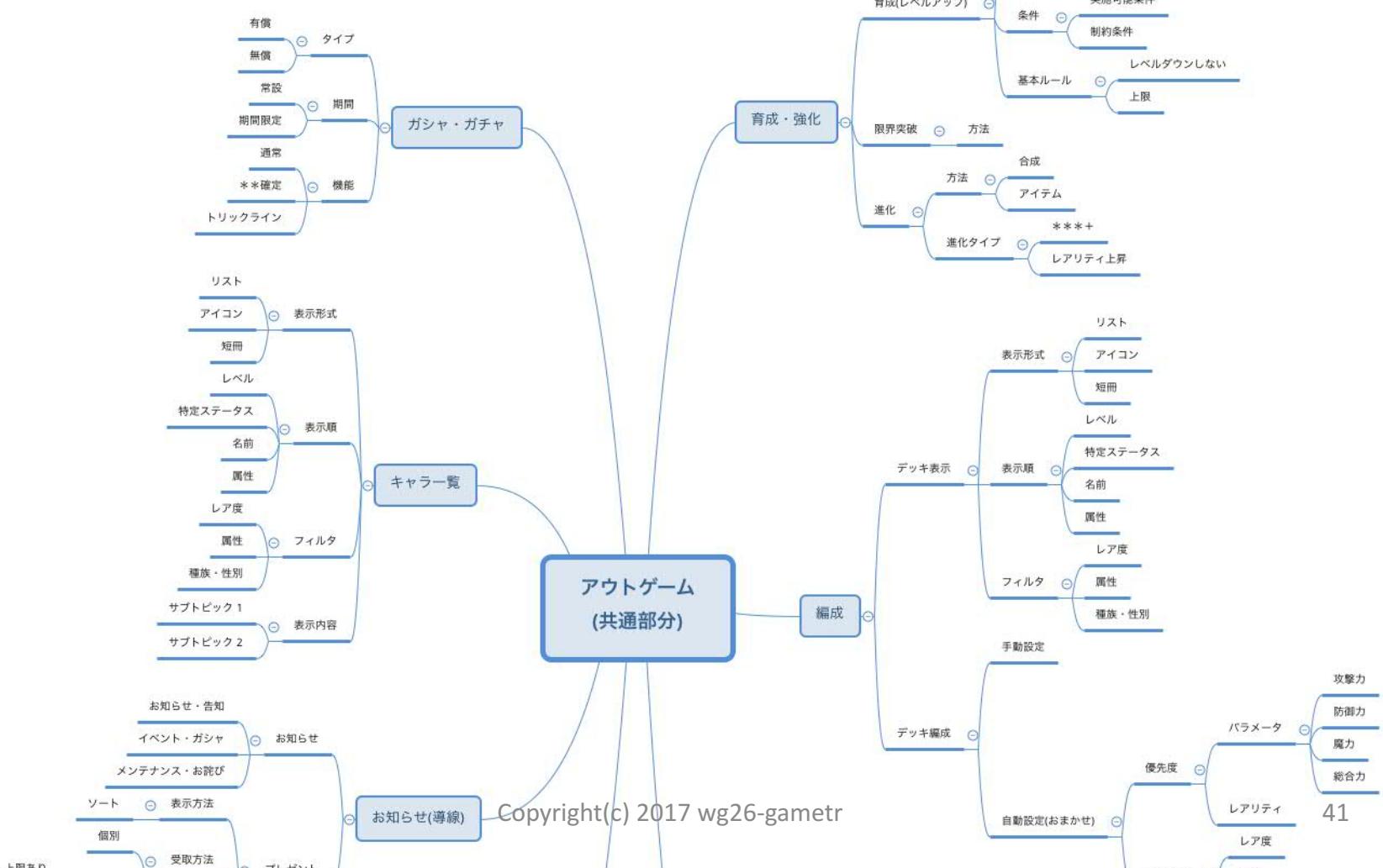
- 貢献・効果：テスト工数の最適化

- 各カテゴライズの内容を確認することで、抜け漏れを防ぐ
- カテゴライズ内から適切な値を選択することで、リスク・優先度が高い項目に注力することが可能になり、テストケースを最適化できる

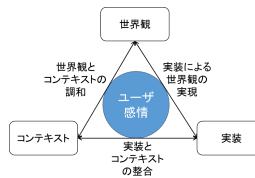
プロダクトへの「貢献」 (ゲームテストの世界)



テスト観点の一例



プロダクトへの「貢献」（ゲームテストの世界）



Step4：
テスト観点を組合せて実施

テストフレーム(仮)の活用により
テスト観点の組合せ最適化し
テストケースの最適化に貢献

- Step4：テスト観点を組合せて実施

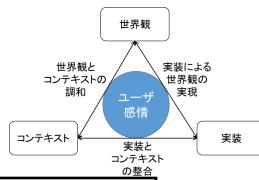
- アプローチ：

- テスト観点レベルでの組合せを検討
- テスト観点・組合せ表をテストフレームワーク(仮)として作成し
ベテランテスト設計者・実施者の考え方をメンバーに伝授
- テストフレームワーク(仮)での設計レビューを実施

- 貢献・効果：テスト工数の最適化、人材育成

- テスト対象への適用観点が抜け漏れを防止
- テストフレームワーク(仮)をベースに、ベテランテスト設計者・
実施者の考え方をメンバーに伝授することで、人材を育成
- テストフレームワーク(仮)でのレビューを実施により、手順書
ベースよりもレビュー工数が軽減(最適化)

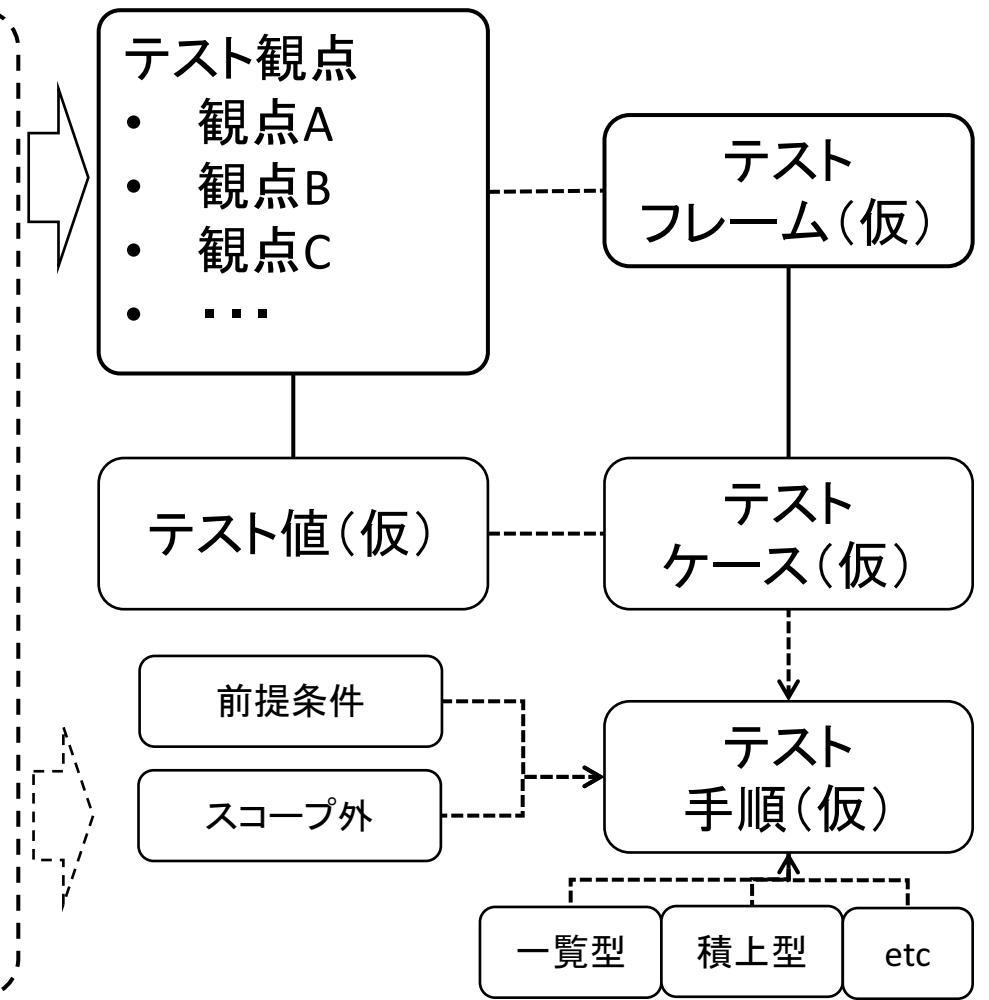
プロダクトへの「貢献」（ゲームテストの世界）

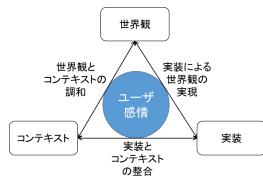


・ テスト設計の流れ(例)

テスト関心事

- ・ シナリオ・シーン・画像
- ・ 操作
- ・ 動作・演出
- ・ OSからの割り込み・OSの状態・タイマ
- ・ 環境・構成・機種・設定・OS種別・OSバージョン
- ・ 状態・モード
- ・ ユーザーシナリオ・ユーザーストーリー
- ・ アイテム
- ・ コンテンツ
- ・ バグ
- ・ etc
- ・ ...





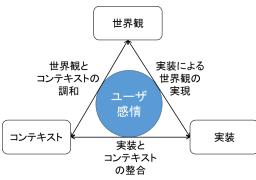
プロダクトへの「貢献」（ゲームテストの世界）

Step5：
テスト観点として
ユーザー感情の導入

ユーザー視点での改善提案による
施策の魅力向上に貢献

- Step5：テスト観点としてユーザー感情の導入
 - アプローチ：
 - ユーザー感情を分析して合否判定材料を明確化
 - ターゲットユーザーのポジティブ・ネガティブの感情を設定
 - ターゲットユーザー以外の感情を適時蓄積
 - 貢献・効果：サービスの魅力品質向上
 - ユーザー感情でのテストを実施することで、機能的として正しくても、目的・要求と異なっている場合は、改善提案を実施しプロダクト品質向上

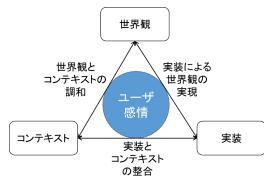
プロダクトへの「貢献」（ゲームテストの世界）



Step6：
総合的な観点での
テスト実施

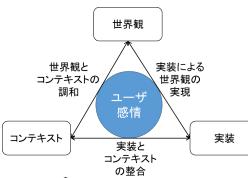
世界観・実装・コンテキストのバランスを
ユーザー感情を軸にテストすることにより
プロダクトの魅力向上に貢献

- Step6：総合的な観点でのテスト実施
 - アプローチ：
 - ユーザー感情を中心に、コンテキスト・世界観・実装のバランスを考えつつ、より良いサービスのために改善提案を前提にテストを実施(楽しい・感動・継続・課金)
 - 貢献・効果：サービスの魅力品質向上
 - 施策(イベント)全体のバランスから企画(レベルデザイン)の改善提案し、施策(イベント)の魅力品質向上(過去のイベントにおける、DAU・ARPU等から)



まとめ

プロダクトへの「貢献」（ゲームテストの世界）



Step1：
単独テスト観点で実施
(機能・世界観等)

基本的なテスト実施により
不具合発見に貢献

Step2：
テスト観点の洗い出し

テスト観点の抽出により
不具合発見の抜け漏れ防止に貢献

Step3：
テスト観点のカテゴライズ

テスト観点のカテゴライズにより
ズームイン・ズームアウト可能
テストケースの最適化に貢献

Step4：
テスト観点を組合せて実施

テストフレーム(仮)の活用により
テスト観点の組合せ最適化し
テストケースの最適化に貢献

Step5：
テスト観点として
ユーザー感情の導入

ユーザー視点での改善提案による
施策の魅力向上に貢献

Step6：
総合的な観点での
テスト実施

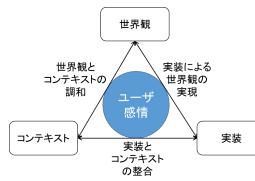
世界観・実装・コンテキストのバランスを
ユーザー感情を軸にテストすることにより
プロダクトの魅力向上に貢献

不具合発見
に貢献！

テスト最適化
に貢献！

魅力向上
に貢献！

三銃士モデルで、何をしたいか？



- テスト観点を構造化する

Step2-3

暗黙的だった観点のカテゴライズを明確化
観点構成を理解・分解・再構築しやすく

- テスト設計方法の提案

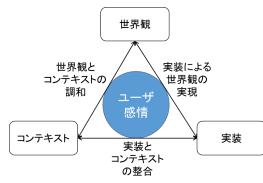
Step4

観点の組合せを用いた、テスト設計方法の提案
観点の組合せは、コンテンツに合わせてカスタマイズ
観点・リスクから優先度付けして、アジャスト可能にしたい
まずはシンプルに、段階的な活用方法の提案
観点・組合せの可視化で、ノウハウ伝授を促進

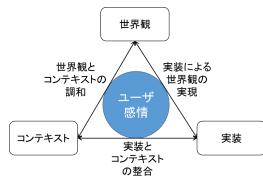
- テスト合否判定にユーザー観点を追加

Step5

ユーザー感情を判定材料とする、ユーザー観点でのテスト
ユーザー観点での判定を、テスター熟練度に依存せず実施
多角的な観点でのテスト実施 & 設計の実施



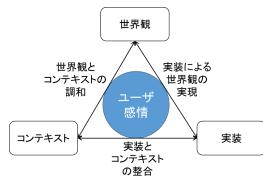
最後に、お知らせです



「三銃士モデル」 の実践は、 これからです！

活動状況

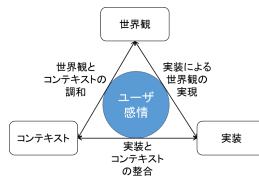
～これから～



「三銃士モデル」
一緒に育てる
仲間を
募集中です！

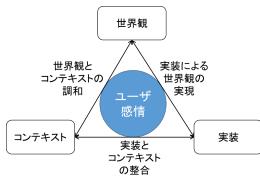
活動状況

～これから～



「三銃士モデル」
興味がある方
ご連絡ください！

※連絡先: ゲームテスト研究会
wg26-gametr@qualab.jp



ご清聴、
ありがとうございました。