

ステークホルダーのアクションと関心事に着目したレビュー観点導出手法

～今日からあなたも上級レビューア！ 『SAKE』の提案～

Methodology for extracting review perspectives focusing on stakeholders' actions and "kanshingoto"

2021年度 SQiP 研究会 第2研究コース（観抽 Hi グループ）

The 2nd section meeting teams "KAN-CHU-Hi" of SQIP Study Group in 2021

○茂木 郷志 樋口 雄基¹⁾ 宇根 勲²⁾ 蜂須賀 夏子³⁾○Satoshi Mogi Yuuki Higuchi¹⁾ Isao Une²⁾ Natsuko Hachisuka³⁾村田 健二⁴⁾ 濱田 航一⁵⁾ 児玉 敬⁶⁾Kenji Murata⁴⁾ Kouichi Hamada⁵⁾ Kei Kodama⁶⁾

研究概要

ソフトウェア開発において、品質向上を図るためにレビューの実施が重要かつ不可欠である。しかし、上級レビューアであれば検出できる欠陥も、上級レビューア以外では見逃すことが少なくない。その対策としてレビュー観点をあらかじめ用意することが多いが、レビューの実施効果は依然としてレビューアのスキルに依存する傾向にある。そこで筆者らは、レビューで検出すべき欠陥を上級レビューア以外でも検出できることを目的としたレビュー観点導出手法を考案した。実験により、本手法の有効性を確認することができた。

Abstract

In software development, it is important and indispensable to conduct reviews in order to improve quality. However, defects that can be detected by senior reviewers are often overlooked by non-senior reviewers. As a countermeasure, review perspectives are often prepared in advance. But the effectiveness of the review still tends to depend on the skill of the reviewers. Therefore, we have devised a method for deriving review perspectives, aiming at making it possible for non-senior reviewers to detect defects that should be detected by review. The effectiveness of this method was confirmed by experiments.

【所属】

パナソニック ITS 株式会社 室蘭開発室

Muroran Development Office, Panasonic ITS Co., Ltd.

神奈川県横浜市都筑区佐江戸町600番地 Tel: 080-8547-2319 e-mail: mogi.satoshi001@jp.panasonic.com

600 Saedochi, Tsuzuki Ward, Yokohama City, Kanagawa Japan

1) 三菱プレシジョン株式会社

Mitsubishi Precision Co., Ltd.

2) SCSK株式会社

SCSK Corporation.

3) 株式会社オーグス総研

OGIS Research Institute Co., Ltd.

4) 三菱総研DCS株式会社

Mitsubishi Research Institute DCS Co., Ltd.

5) IDEC株式会社

IDEC Corporation.

6) 旭化成株式会社

Asahi Kasei Corporation.

【キーワード】レビュー、レビュー観点、欠陥検出、品質特性、ステークホルダー

1. はじめに

1.1 背景

ソフトウェア開発では、品質向上を図るためにレビューの実施が重要かつ不可欠であるが、レビュー効果はレビューアのスキルに依存する傾向にある。知識や経験が豊富な上級レビューアは安定して欠陥を検出できるが、全てのレビューを上級レビューアが担当することは現実的には難しい。そこで、検出すべき欠陥を上級レビューア以外でも検出できるように、レビュー観点を準備する、等の工夫をして

いるが、欠陥を見逃してしまうケースが少なくない。SQuaRE 品質特性^[1]の活用やレビュー観点の設定は非常に有効な手段であり一定の効果は得られるものの、依然としてレビューアのスキルへの依存度が高い状態であることに変わりはない。

レビュー観点を設定してもスキルへの依存度が高い原因として、特に次の2つが重要と考える。

第1に、レビュー観点がプロジェクトやレビュー対象物の特性を踏まえたものになっていないケースが多いことが挙げられる。汎用的なレビュー観点にはレビュー対象に関係のない観点も含まれていたり、逆に検出すべき欠陥を検出するための観点が漏れていたり、必ずしも最適なレビュー観点とは言えないケースが多い。また最適なレビュー観点を自分達で設定しようと思っても導出方法がわからない。

第2に、レビュー観度の粒度をどの程度にすれば良いのか判断が難しいことが挙げられる。一般的に体系化された品質特性は抽象度が高く、いわゆる粒度の粗い観点となっていることが多いため、知識や経験が不足していると具体的に何をチェックすれば良いのか想起できない。だからと言って具体的な細かな観点を与えられるとその観点は見るが、数が増えるため時間もかかり、与えられた観点以外でレビューする時間的余裕がなくなる。レビューアの知識や経験の不足を補いつつも細かくなり過ぎない適切な観度の粒度があるはずだが、画一的な答えは存在しない。また、適切な観度の粒度を各々で設定しようと思っても決定方法がわからない。

1.2 解決すべき課題

以上の背景を踏まえ、筆者らは上級レビューアと同等レベルのレビュー活動を上級レビューア以外でも行えるようにすることを目的に、以下の点を解決すべき課題と捉えた。

RQ1：どうすればプロジェクトやレビュー対象物の特性を踏まえたレビュー観点を導出できるか。

RQ2：どうすれば上級レビューア以外でも欠陥を検出できる粒度のレビュー観点を導出できるか。

これらの課題の解を導出できれば、レビュー活動の改善が図れ、欠陥の流出を低減できるのではないかと考えた。

以降、2章では課題の解決策として筆者らが提案するレビュー観点導出手法を示す。3章で提案手法に対する実験と評価考察を行い、4章でまとめを示す。

2. 課題解決策の提案

2.1 提案手法が対象とする開発工程

提案手法が対象とする開発工程は、筆者らの担当開発領域を考慮し、ソフトウェア開発工程^[2]の要求定義、アーキテクチャ設計、詳細設計、実装とする。

2.2 提案手法の方針

課題解決に向けた手法の検討にあたり、レビュー対象物である成果物の性質に着目した。開発における各種成果物は、製品開発に関わる様々なステークホルダーの明示的、暗黙的な要求事項を反映したものであり、レビューはそれらが適切に反映されているかを検証する場に他ならない。そのため、製品に関わるステークホルダーを洗い出し、ステークホルダーの要求事項の源泉となる期待、要望、懸念、疑問といった『関心事』からレビュー観点を導出すれば、プロジェクトやレビュー対象物の特性を踏まえた観点導出が可能になると考えた。また、ステークホルダーの『アクション』に着目し、ステークホルダーの視点をより具体的に理解した上で、関心事を抽出する。さらに、関心事から段階的に『レビュー観点』に落とし込むことで、具体的な且つレビューアのスキルに応じた粒度の観点導出が可能になると考えた。

これらを踏まえたレビュー観点の導出方法を明確な手順として示すことで、スキルによる依存性を下げ、観点導出の確実性を高める。これにより、上級レビューア以外でも、有効な観点の導出が可能となり、上級レビューアに近いレベルで、欠陥を検出することができるように考えた。

2.3 関連するレビュー技法

レビュー技法の中で、ステークホルダーの視点をを用いた従来技法として Perspective Based Reading (以下、PBR とよぶ) や Role Based Review (以下、RBR とよぶ) が存在する。

どちらの技法もステークホルダーの視点をを用いて、プロジェクトやレビュー対象物の特性を踏まえたレビュー観点をレビュー実施可能な技法である。しかし、PBR は成果物を作成するため、大きな労力・時間が必要であり、RBR は具体的な手順が定義されていないため、レビューアのスキルによって欠陥検出結果にばらつきが生じやすい。大きな工数を必要とせず実業務への導入がしやすく、レビューアのスキ

ルへの依存性を克服できる技法は見つからなかった。

2.4 提案手法

筆者らが提案するステークホルダーのアクションからレビュー観点を導出する手法

『SAKE (Stakeholder Action Kanshingoto Extraction) Method』(以下, SAKE とよぶ) を以下に示す。

2.4.1 提案手法全体像

2.2 項の方針に基づき作成した, レビュー観点導出の手順を図1に示す。

ステークホルダーから段階的にレビュー観点を導出する「レビュー観点導出フェーズ」と導出した観点をレビュー観点特性表に当てはめ, 体系的に確認する「レビュー観点補充フェーズ」から成り, 計6つのStepを実施する。(詳細は2.4.2項に記載)

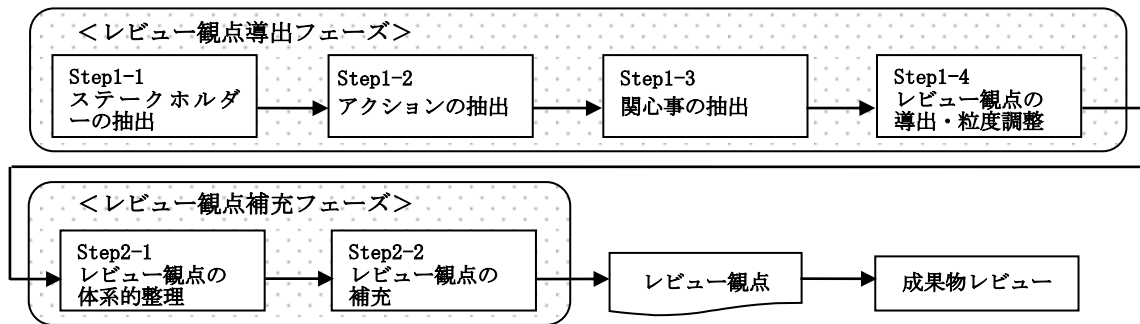


図1 実施フロー

本手法は, 開発計画時から実施可能である。ただし, 仕様不明確や仕様変更が生じることを考慮し, 仕様確定時や成果物レビュー前に適宜更新をかけることを推奨する。

本手法の実施者は, レビュー対象物のレビューアを想定している。ただし特に制限は無く上級レビューアや各ステークホルダーも可能であれば適宜参加することが望ましい。

なお, 類似/派生開発では, 過去作成した観点をベースに更新することでノウハウの継承と負荷軽減が期待できる。

2.4.2 提案手法の詳細

手法の実施手順詳細について述べる。

■■Step1 観点導出フェーズ

Step1にて, ステークホルダーとアクションに基づくレビュー観点を導出を行う(表1)。レビュー観点は各Stepを実施しながら連想することで導出するため, マインドマップ形式を用いることを推奨する。レビュー観点導出のイメージを図2に示す。

表1 Step1 レビュー観点導出フェーズの各ステップの実施事項

Step	実施事項/例/留意事項
Step1-1 ステークホル ダーの抽出 〈Stakeholder〉	<p>【実施事項】 開発対象の製品ライフサイクルをもとに, 製品に関わる全てのステークホルダーを抽出する。</p> <p>【例】 ユーザー, 企画, 営業, 設計, コーダー, テスター, 保守, 販売, 製造, 運送, サービス</p> <p>【留意事項】 組織図やプロジェクト体制表が存在する場合はこれらを参考にする。 ステークホルダーは, 悪意を持った人物や子供のような, 意図しない結果を生み出す可能性がある人物も含めて抽出することで, 多様な観点を導出することができる。 人以外の「物」(周辺システムや関連製品など)を挙げることで, 直接的に観点を導出しやすい場合もあるため, 必ずしも「人」でなくても良い。 当Stepの効果と効率性を高めるため, 一般的な製品開発に共通して登場すると想定されるステークホルダーの一部を筆者らが記載した「レビュー観点導出マインドマップ」(図3)をベースに, 対象製品に応じたステークホルダーを抽出しても良い。</p>
Step1-2 アクションの 抽出	<p>【実施事項】 各ステークホルダーが目的を達成するために行うアクションを抽出する。</p> <p>【例】設計者:「設計書を作成する」→「機能ブロックの抽出, I/F設計, リソース見積もり」</p> <p>【留意事項】 アクションを分解することで, より具体的なレビュー観点導出が期待できる。 専門性の高いステークホルダーのアクションについては, 開発プロセスや作業手順書等を参考</p>

<p><Action></p>	<p>にする。「物」の場合は、仕様や説明書を参考に、動作や開発対象製品にどのように関わるのかを抽出する。</p>
<p>Step1-3 関心事の抽出 <Kanshingoto></p>	<p>【実施事項】 各ステークホルダーのアクション実施時における、レビュー対象製品、または開発成果物に対するステークホルダーの関心事を抽出する。</p> <p>【例】 情報を入力する：「少ない手数で入力を完了したい（期待、要望）」「誤った値を入力できてしまわないか（懸念）」、「どんな文字形式も入力できるのか（疑問）」</p> <p>【留意事項】 関心事は、各ステークホルダーが製品や各開発成果物に対して持つ期待、要望、懸念、疑問などの要素であり、要求事項の源泉となるものである。製品や各開発成果物とその人との関係性の違いを考慮しながら、以下の点に留意して関心事を抽出すると良い。 ○期待・要望：各アクション達成のために、製品や成果物に対して許容できる、又はより良い状態を想定する。 (表現例：「xx であってほしい、xx あるべきだ」など.) ○懸念 : 各アクションの達成を阻害する要因を想定する。 (表現例：「xx となってしまうかないか」など.) ○疑問 : 各アクション達成のために確認すべき事項、不明点を想定する。 (表現例：「xx は何か、xx はどうであるか」など.) なお、関心事を想起することが難しい場合は、上級レビューアや、各ステークホルダーへのインタビューを行うことも有効である。</p>
<p>Step1-4 レビュー観点の導出・粒度調整 <Extraction></p>	<p>【実施事項】 関心事から想起されるレビュー対象物で確認すべき事項を検討の上、適切な粒度のレビュー観点を導出する。</p> <p>【例】 誤った値を入力できてしまわないか：「入力ミス検出機能はあるか」→「不正文字検出機能はあるか」</p> <p>【留意事項】 レビュー観点は、Step1-3 で抽出した関心事に対し、期待・要望を達成するために、懸念を回避するために、疑問を解消するために、レビュー対象物がどのような状態であるべきかを検討し、導出する。 また、レビュー観点の適切な粒度は、レビューアの知識、経験によって異なるため、レビューアの能力に応じて調整する。適切な粒度の目安としては、その観点の内容から、レビューア自身が『レビュー対象物の該当箇所』と『あるべき状態』を想起できる程度であることが望ましい。必要に応じて、上級レビューアや、該当するステークホルダー等に確認をとりながら行うことで、必要な観点の導出漏れを防ぐことができる。 Step1-3 で抽出した関心事がレビュー観点として十分に具体的に表現できていれば、関心事をそのまま観点としても良い。</p>

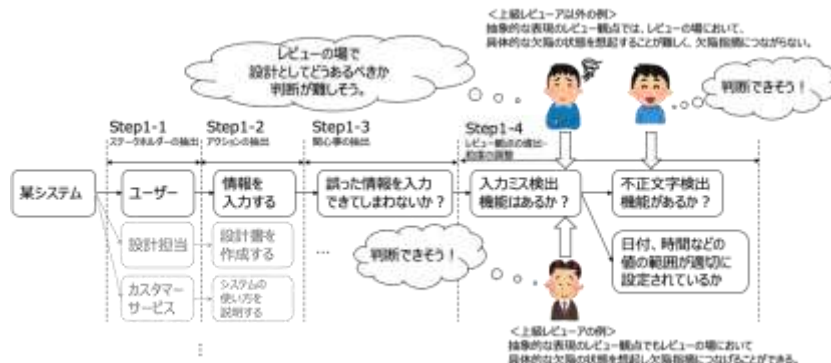


図2 レビュー観点の導出例



図3 レビュー観点導出マインドマップ

■■Step2 レビュー観点補充フェーズ

Step2にて、レビュー観点特性表でレビュー観点を体系的に整理した上で、観点を補充する。(表2) レビュー観点特性表は、多様なステークホルダーや多様な工程に於いての特性を含めることができるように、SQuaRE 品質特性をベースに、非機能要求グレードやドキュメント品質特性など一般的に体系化された多様な特性^{[1][3][4][5][6][7]}を追加・統合して、筆者らが作成した特性表である(図4)。なお、本手法が対象とする開発工程(2.1項に記載)を想定し、各工程に於いて関連する特性を収集して作成している。

表2 Step2 レビュー観点補充フェーズの各ステップの実施事項

Step	実施事項/留意事項
Step2-1 レビュー観点の体系的整理 <Systematization>	<p>【実施事項】 Step1で導出したレビュー観点をレビュー観点特性表に当てはめて体系的に整理する。</p> <p>【留意事項】 特性表全体を必ずしも利用する必要はなく、プロジェクトやレビュー対象物の特性に応じて必要な部分を取捨選択して利用すれば良い。</p>
Step2-2 レビュー観点の補充 <Replenishment>	<p>【実施事項】 レビュー観点特性表を俯瞰的に観察し、観点を補充する。</p> <p>【留意事項】 観点の分布がレビュー対象の特性を踏まえて適正だと言えるか、特性を視点に大局的に確認する。観点の分布の偏り等から、必要な観点の導出漏れに気づいた場合は、新たに観点を追加する。必要に応じて、追加した観点に対して、Step1-4に示す手順により適切な粒度に調整すると良い。(図5参照)</p>

機能適合性	完全性	正確性	機能適切性
性能効率性	効率性	リソース拡張性	
互換性	適応性		
使用性	理解性	習得性	快適性
	有効性	実用性	ユーザーエラー防止性
	アクセシビリティ	快感性	
信頼性	成熟性	可用性	耐故障性
	回復性		
セキュリティ	機密性	インテグリティ(完全性)	追跡性
	信用性	セキュリティ対策	
保守性	モジュール性	再利用性	解析性
	修正性	試験性	
リスク回避性	経済リスク緩和性	健康・安全リスク緩和性	環境適応性
固有の視点	一貫性	最新性	適合性
	精度		
その他の非機能要件	運用性	移行性	

図4 レビュー観点特性表

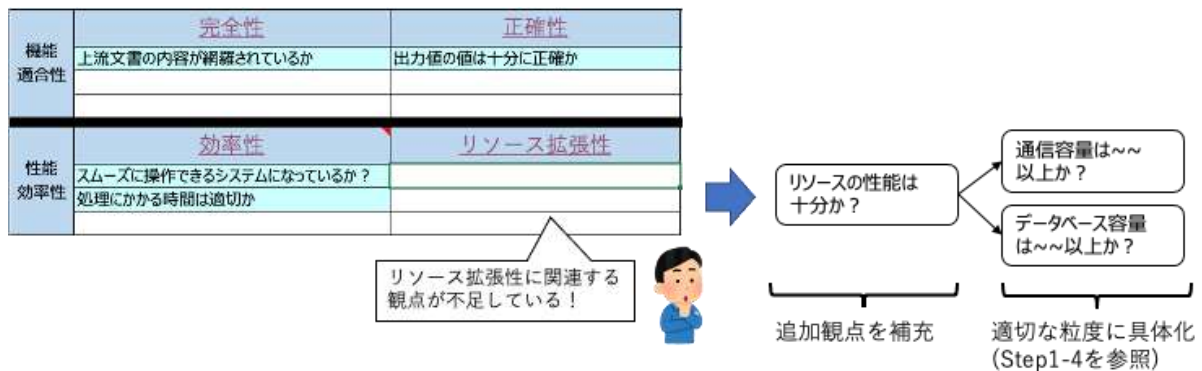


図5 レビュー観点特性表 観点当てはめ結果例 (某システムにおける実施例)

レビュー実施時の留意事項

- Step1, Step2で導出したレビュー観点を観察しながらレビューを実施する。
- 各レビュー観点を一つ一つ確認すると膨大な時間を要するため、ステークホルダー、アクション、関心事などのカテゴリから、複数の観点を観察しながら実施すると良い。

3. 解決策の検証

3.1 検証の目的

SAKEの方針で述べた「プロジェクトやレビュー対象物の特性を踏まえたレビュー観点を導出できる

こと」, 「レビューアのスキルに応じた粒度の観点を導出できること」, 及びその結果「上級レビューア以外が上級レビューアに近いレベルで欠陥を検出できること」を検証する。

3.2 検証手順

- 筆者らを除く所属各社の被験者を, レビューア経験年数が7年以上を上級レビューア, 7年未満を上級レビューア以外としてスキル別に分類する. また, レビューアはグループA/Bに分ける.
 - グループA: 上級レビューア 3人, 上級レビューア以外 9人
 - グループB: 上級レビューア 5人, 上級レビューア以外 7人
- 特殊なドメイン知識を必要としない同じ規模の仮想の仕様書を2つ用意する.
 - 仕様書X: 旅費精算システムソフトウェア要求仕様書(7ページ, 2883字)
 - 仕様書Y: ヘリコプター予約システムソフトウェア要求仕様書(6ページ, 3391字)
- 被験者は個人単位で, 表3に示す手順で計2回のレビューを実施する. この時, SAKEを経験したことによる優位性が出ないよう, RBR⇒SAKEの順とする. また, RBRの前にステークホルダー例を, SAKEの前にレビュー観点導出例を提示する.
- レビューで検出した欠陥は, 知見分析表^[8]に基づいて影響と緊急度を判定し, 重大欠陥/中程度欠陥/軽微欠陥の平均件数を比較する. なお, 判定基準を統一するため, 筆者の内2名を最終判定者とし, 他の筆者が一次判定した後, 最終判定者が再判定を行う.

表3 レビュー方法

		1回目: RBR	2回目: SAKE
レビュー対象物	グループA	仕様書X(旅費精算システム)	仕様書Y(ヘリコプター予約システム)
	グループB	仕様書Y(ヘリコプター予約システム)	仕様書X(旅費精算システム)
目安時間	準備	10分(ステークホルダー導出)	90分(レビュー観点導出)
	レビュー	30分	30分

3.3 検証結果

(1) プロジェクトやレビュー対象物の特性を踏まえたレビュー観点を導出できること

特性を踏まえたレビュー観点を導出できたか否かは, 異なる2つの仕様書に対する中程度以上の欠陥検出数の増減で確認する.

仕様書毎の検出した欠陥の平均件数を図6に示す. SAKEでは, 中程度以上の欠陥検出数が仕様書Xで2.02倍, 仕様書Yで1.70倍に増加した(①).

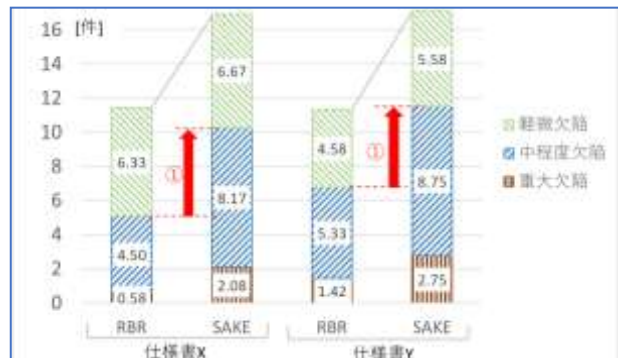


図6 検出した欠陥の平均件数(仕様書毎)

(2) ステークホルダーからレビューアのスキルに応じた粒度の観点を導出できること

SAKEで中程度以上の欠陥の検出に寄与した観点の粒度について検証した結果を図7に示す. 粒度の細かさは, 検出された欠陥と対応する観点の記述が, 同等レベルまで具体化されている(=粒度が細かい)か, されていない(=粒度が粗い)か, で測った(表4). 図7に示す結果から, 仕様書X, Yともに粒度は個々人で異なり, スキルによる傾向は見えなかった.

表4 観点の粒度の例

粒度	観点	検出された欠陥
細かい	社内システムと連携は問題ないか	社内システムとの連携確認はないのか?
粗い	機能要件は十分か?	明細欄には交通費精算データが入力できるとあるが, 入力する機能がない

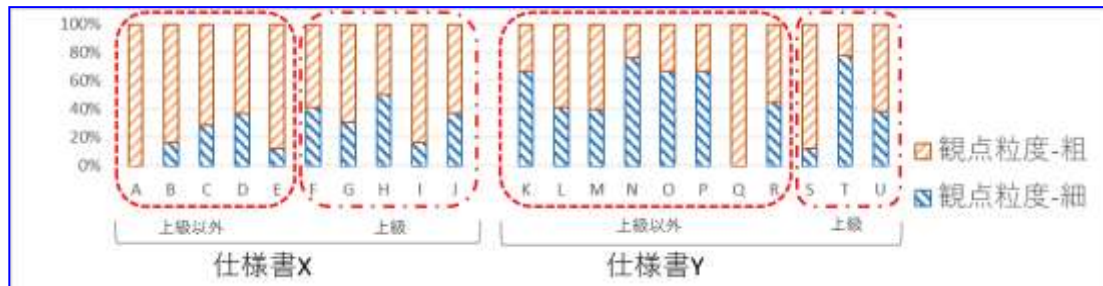


図7 観点粒度毎の欠陥検出数(比率)

(3) 上級レビューア以外が上級レビューアに近いレベルで欠陥を検出できること

上級レビューアと上級レビューア以外の結果を比較すると、どちらも SAKE の欠陥検出数が増えた(図 8). 特に上級レビューア以外の中程度以上の欠陥が大きく増えた(②). 最後に、中程度以上の欠陥において、上級レビューア以外の SAKE の件数(9.37 件/人)が、上級レビューアの RBR の件数(9.63 件/人)とほぼ同程度まで向上した(③).

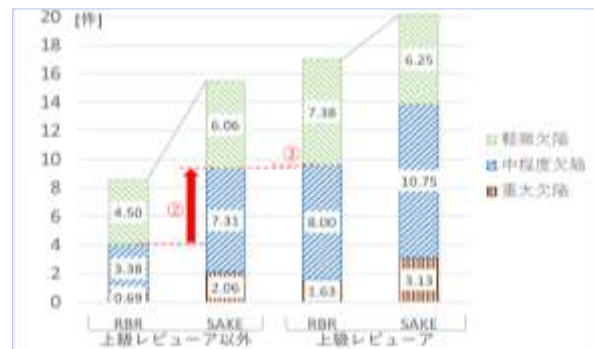


図8 検出した欠陥の平均件数(上級以外/上級毎)

3.4 考察

検証結果を踏まえて、本手法による課題の解決度合いについて考察する。

【考察1 プロジェクトやレビュー対象物の特性を踏まえたレビュー観点の導出について】

異なる 2 つの仕様書に対して、どちらも中程度以上の欠陥検出数が増加したという結果(図 6)から、SAKE を用いることでレビュー対象物の特性を踏まえたレビュー観点の導出が可能になると考える。実際に導出した観点から検出された欠陥の内容を見ると、RBR は主に正常系の事柄や仕様書の文言から想起しやすい欠陥だったが、SAKE は保守や例外事象、連携するシステム、利便性および機能の妥当性などレビュー対象の特性に応じた欠陥が増えた。

【SAKE の欠陥例】「予約画面で他の利用者の予約情報が見られるのは適切か？」

これは、ステークホルダーからアクションと関心事を明らかにしたことで、人やシステムの背景をより深く理解し、何が適切で何が不適切かを具体的に想起できたためと考える。

【考察2 レビューアのスキルに応じた粒度の観点の導出について】

上級レビューアの観点は粒度が粗くなると考えていたが、検証結果(2)より、傾向は見られなかった。これは、適切と思う観点の粒度は、スキルに因らず個人で異なることを示唆している。検証結果(1)(3)から、SAKE によって上級レビューア以外でも欠陥検出できる粒度の観点が出せた可能性はあるが、観点の粒度が結果にどう寄与するか更なる検証が必要である。

【考察3 上級レビューアに近いレベルで欠陥を検出できることについて】

上級レビューアのアンケートに「関心事の洗い出しまでは、ほぼ、通常のレビューにおいて脳内でやっている作業に近い」とあり、SAKE が上級レビューアの思考プロセスに近いことで図 8 の②、③の結果を得たと推察できる。このことから、SAKE は上級レビューア以外が上級レビューアの思考プロセスを学ぶ手段ともなり得る。また、SAKE の手順に沿ったことで、上級レビューアが脳内に持つ観点やノウハウの一部がレビュー観点として表出したと考える。すなわち、SAKE を繰り返し実践することで上級レビューアの思考プロセスが身につく、観点導出結果を参照することで、他のレビューアが持つ知見やノウハウを吸収できる。レビューア育成に有効な手法とも言えるのではないかと考える。

前述の課題に対する考察に加え、検証結果から得られたその他の考察について記載する。

【考察4 マインドマップについて】

SAKE の観点導出結果は、それ自体が異なる粒度の観点と相互の関連を表現したものである。SAKE では観点を一つ一つチェックすることに加えて、「マインドマップ」を俯瞰して、複数の観点を観察しながらレビューすることが可能であり、ある観点の周囲にある別の観点を見て気付きを得て欠陥数が増え

たとえており、今後検証していきたい。

【考察5 各手順(Step)について】

中程度以上の欠陥に着目すると、SAKEのStep1で得た観点による検出数(9.21件)は、RBRによる検出数(5.92件)よりも多かった(図9の①)。これは、レビュー観点特性表を用いずとも、一定の効果があることを示している。加えて、SAKEのStep2で得た観点でも1.79件(図9の②)を検出した。Step1で得た観点をStep2で特性の視点から見直すことで、大きな漏れや思考の偏りに気付けたためと考える。

【考察6 SAKEの準備時間について】

実験では、RBRと比較してSAKEの準備に多くの時間をかけたが、欠陥検出数を考慮すれば、開発における費用対効果が期待できる。今後はこれら時間と欠陥検出数の比較や、欠陥流出した時の手戻り時間も踏まえたSAKEの効果について、検証していきたい。



図9 Step毎の欠陥検出数

4. まとめ

4.1 結論

本研究では、上記で述べた課題について、ステークホルダー毎のアクションと関心事からプロジェクトやレビュー対象物に合わせたレビュー観点を導出し、SQuARE品質特性やドキュメント品質特性などの大きな視点への当てはめにより、重要な観点を漏れないかを確認する手法『SAKE (Stakeholder Action Kanshingoto Extraction) Method』を考案した。

筆者らの所属各社にて本手法を適用した実験を行い、上級レビューア以外でも上級レビューアと同等の欠陥を検出できたことから、本手法でプロジェクトやレビュー対象物の特性を踏まえたレビュー観点を導出が可能となったと想定できる。一方で、レビュー観位の粒度と欠陥検出数の相関関係を判断できるだけの明確な結果は得られなかった。

4.2 今後の課題

本手法を実際の開発現場に導入していくために取り組むべき課題を挙げる。

- ・開発する製品やレビュー対象物に応じて必要な特性が異なるため、レビュー観点補充フェーズでより大きな効果を得るには、事前に適切な特性を選ぶ必要がある。この選び方やレビュー観点特性表自体の利用方法については今後の検討が必要である。
- ・個々人のスキルに応じた観点を導出する際にどのような要因が関与するのか、それが粒度設定や欠陥検出結果にどのように関係するのかは、今後さらに検証が必要である。
- ・SAKEの時間短縮が課題である。そのために例えば、他者の「レビュー観点マインドマップ」を流用してレビューアに応じた粒度の再調整を行う方法や、状況に応じてSAKEの手順を簡略化する方法が考えられる。今後、これらの検討と検証が必要である。
- ・今回検証対象外の開発工程成果物、実プロジェクトの成果物に対する検証も必要である。

本研究で考案した手法「SAKE」について、検証により一定の効果を確認することができた。今後は開発現場での試行運用を通じて、より高い効果の得られる手法に改善していく。

参考文献

- [1] 「つながる世界のソフトウェア品質ガイド」, IPA, 2015
JIS X 25010 : 2013 (ISO/IEC 25010 : 2011), JIS X 25012 : 2013 (ISO/IEC 25010 : 2011)
- [2] 「組み込みソフトウェア向け開発プロセスガイド」, IPA-SEC, 2007
- [3] 「テスト技術者資格制度 Foundation Level シラバス Version2018 V3.1」, JSTQB, 2021
- [4] 「非機能要求グレード2018」, IPA, 2018
- [5] 「システム開発文書品質モデル Ver1.1」, システム開発文書品質研究会, 2018
- [6] 「Google's Engineering Practices Documentation」, <https://google.github.io/eng-practices/>
- [7] 「情報処理システム高度信頼化教訓集 (ITサービス編) 別紙: 障害事例一覧表」, IPA, 2020
- [8] 今村 陽介, 他, 「レビュー指摘を軽微欠陥から重大欠陥へシフトさせるレビュー時の新規役割「ハーベスタ」の提案」, SQiP研究会(2013年度)研究コース2(Kチーム), 2014