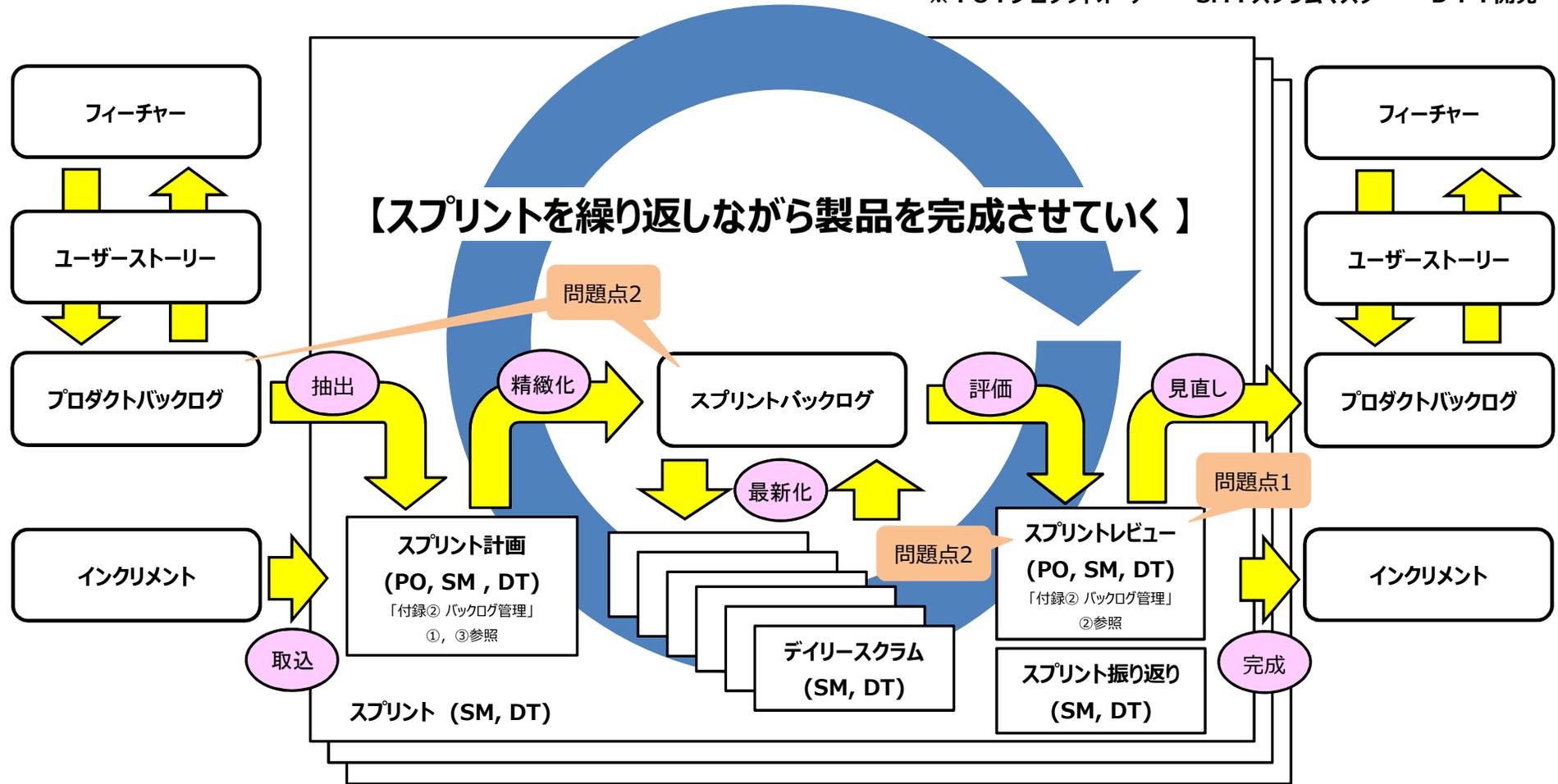


付録① スプリント運用（スクラム開発の全体像）

※ PO：プロダクトオーナー SM：スクラムマスター DT：開発



- 問題点1：スクラムは、スプリント単位で機能実装を繰り返していくプロセスであるため、ソフトウェア構造が乱れやすくなり、品質低下の一因となっている。これに対し、POは、スプリントレビューで、デモンストレーションによって動作確認に焦点を当てた評価を行っており、表面的な確認に留まることが多い。
- 問題点2：POは、プロダクトバックログでプロジェクト全体を進捗管理しているが、スプリントレビューにおいては、個々のスプリントの成果に焦点が当たっており、スプリントバックログにより進捗を確認する。
このような状況の中、スプリント内で発生した作業の増加やチームのベロシティの低下が全体にどの程度影響しているのかを、POが把握することは難しい。

上記2点の問題によって、開発プロジェクトの後半にQCD問題が顕在化することになる。その結果、全体計画との乖離から、後半のスプリントで負荷増加が発生する。

付録② バックログ管理（プロダクトバックログとスプリントバックログの更新運用例）

① 1回目のスプリント開始時（スプリント計画時）

プロダクトバックログ

総ストーリーポイント 16

No	内容	status	ストーリーポイント
1	ログインする。	実施中	3
2	買い物をする。	実施中	5
3	ログアウトする。	未実施	2
4	更新する。	未実施	5
5	表示する。	未実施	1

各スプリントのベロシティを8ストーリーポイントと想定。

1回目のスプリントにプロダクトバックログNo.1とNo.2を、

2回目のスプリントにプロダクトバックログNo.3～No.5を割り当てた。

スプリントバックログ（1回目）

task	status	ストーリーポイント
task 1 A	未着	1
task 1 B	未着	1
task 1 C	未着	1
task 2 A	未着	1
task 2 B	未着	1
task 2 C	未着	1
task 2 D	未着	1
task 2 E	未着	1

スプリントバックログ（2回目）

task	status	ストーリーポイント
task 3 A	未着	1
task 3 B	未着	1
task 4 A	未着	1
task 4 B	未着	1
task 4 C	未着	1
task 4 D	未着	1
task 4 E	未着	1
task 5 A	未着	1

※taskは実作業を表す。

例えば、上表にてtask1-Aは、プロダクトバックログ No.1の実作業 Aとなる。

② 1回目のスプリント終了時（スプリントレビュー時）

プロダクトバックログ

総ストーリーポイント 17

No	内容	status	ストーリーポイント
1	ログインする。	完了	3
2	買い物をする。	完了	5
2 a	買い物をする。（仕様追加）	未実施 追加	1
3	ログアウトする。	未実施	2
4	更新する。	未実施	5
5	表示する。	未実施	1

スプリント計画時に予定したtaskは全て終了したが、スプリントレビューでプロダクトバックログNo.2に仕様追加が必要であることが判明したため、その対応をスプリントバックログと（task 2-F）をプロダクトバックログ（No.2a）に反映した。

スプリントバックログ（1回目）

task	status	ストーリーポイント
task 1 A	完了	1
task 1 B	完了	1
task 1 C	完了	1
task 2 A	完了	1
task 2 B	完了	1
task 2 C	完了	1
task 2 D	完了	1
task 2 E	完了	1
task 2 F	未着	1

スプリントバックログ（2回目）

task	status	ストーリーポイント
task 3 A	未着	1
task 3 B	未着	1
task 4 A	未着	1
task 4 B	未着	1
task 4 C	未着	1
task 4 D	未着	1
task 4 E	未着	1
task 5 A	未着	1

③ 2回目のスプリント開始時（スプリント計画時）

プロダクトバックログ

総ストーリーポイント 16

No	内容	status	ストーリーポイント
1	ログインする。	完了	3
2	買い物をする。	完了	5
2 a	買い物をする。（仕様追加）	実施中 追加	1
3	ログアウトする。	実施中	2
4	更新する。	実施中	5
5	表示する。	未実施 取下げ	±

追加したバックログ②（No.2 a）（task 2-F）の優先度が高かったため、それを2回目のスプリントに割り当て、代わりにプロダクトバックログNo.5を取り下げることにした。

スプリントバックログ（1回目）

task	status	ストーリーポイント
task 1 A	完了	1
task 1 B	完了	1
task 1 C	完了	1
task 2 A	完了	1
task 2 B	完了	1
task 2 C	完了	1
task 2 D	完了	1
task 2 E	完了	1

スプリントバックログ（2回目）

task	status	ストーリーポイント
task 2 F	未着	1
task 3 A	未着	1
task 3 B	未着	1
task 4 A	未着	1
task 4 B	未着	1
task 4 C	未着	1
task 4 D	未着	1
task 4 E	未着	1

付録③ スクラム用語集

No.	分類	用語	説明	参考元
1	チーム	スクラムチーム (S T)	スクラムチームは、プロダクトオーナー・開発チーム・スクラムマスターで構成される。 スクラムチームは、フィードバックの機会を最大化するため、プロダクトを反復的・漸進的に作成し、動作するプロダクトを常に利用可能な状態にする。	①
2	チーム	プロダクトオーナー (P O)	開発チームの作業とプロダクトの価値の最大化に責任を持つ人。 プロダクトバックログの管理に責任を持つ唯一の人間。	①
3	チーム	スクラムマスター (S M)	スクラムの理解と成立に責任を持つ人。スクラムチームのサーバントリーダー (メンバーが成果を上げるために支援や奉仕をするリーダー)。 スクラムチームにスクラムの理論・プラクティス・ルールを守ってもらうようにする。	①
4	チーム	開発チーム (D T)	インクリメントを完成することができるメンバー。その専門家で構成されたチーム。 開発チームは、自分たちの作業を構成・管理する。	①
5	イベント	スプリント	スクラムでの開発単位。(イテレーションと呼ぶ場合もある) 「完成」したインクリメントを作るための、1か月以下のタイムボックス。	①
6	イベント	スプリント計画	スプリント開始時に、スクラムチームが行うスプリントの全体計画を作る。 プロダクトバックログを基に、当該スプリントで何を行うのかを計画し、スプリントバックログを作成する。	①
7	イベント	デイリースクラム	スクラムチームが次の24時間で何を行うのか、その計画を作る。 毎日1回実施する15分間のタイムボックス。前回のデイリースクラムから行った作業の検査と、次回のデイリースクラムまでに行う作業の予想を行う。	①
8	イベント	スプリントレビュー	スプリント終了時に、作成したインクリメントの検査と実施したスプリントの評価、デモンストレーションを行う。 スプリント計画時に作成したスプリントバックログのストーリーポイントや作業タスクを評価し、プロダクトバックログのストーリーポイントや作業タスクにフィードバックする。	①
9	イベント	スプリントの振り返り	スプリント終了時に、スクラムチームの検査を行い、次のスプリントの改善計画を作成する。	①
10	作成物	フィーチャー	顧客またはユーザーの視点からシステムに要求される機能。	②
11	作成物	ユーザーストーリー	フィーチャーより導き出されたシステムの要求仕様。プロダクトオーナーと開発チームとの対話を促進するプラクティス。 「○○として、○○したい。なぜなら○○だからだ。」というフォーマットで、要求(ユーザー、達成したいゴール、理由)をシンプルかつ簡潔に記述。	③
12	作成物	プロダクトバックログ	フィーチャー、ユーザーストーリー等、プロダクトの完成に必要な全ての要素が並べられた一覧。 プロダクトに対する要求の唯一の情報源。	①
13	作成物	スプリントバックログ	当該スプリントで対応するプロダクトバックログについて、そのインクリメントを完成し、スプリントゴールを達成するために必要な全ての作業タスクの一覧。	①
14	作成物	インクリメント	当該スプリントまでに完成したユーザーストーリーを合わせたもの。(ユーザーストーリーを実装したもの) スプリントの終わりには、新しいインクリメントが「完成」していなければならない。「完成」とはインクリメントが動作する状態であり、スクラムチームの「完成」の定義に合っていることを意味する。プロダクトオーナーがリリースを決定する/しないにかかわらず、インクリメントは常に動作する状態にしておかなければいけない。	①
15	その他	ストーリーポイント	ユーザーストーリーを実装するのに必要な作業量。(見積り) 相対見積りを行って見積もった値。相対見積りは、基準となるものとの相対的な規模の差異を見積りの数値として使う手法。相対見積りの数値はポイントと呼ばれる。	③
16	その他	ベロシティ	1スプリントでスクラムチームが実施出来る作業量。(生産性)	③
17	その他	プロダクトバーンダウンチャート	プロダクトバックログの総作業量(ストーリーポイント)に着目して、プロダクトの完成時期を類推するグラフ。 作業量を縦軸にとり、横軸に時間の流れをとる。	③
18	その他	スプリントバーンダウンチャート	スプリントバックログの総作業量(ストーリーポイント)に着目して、スプリントの完成時期を類推するグラフ。 作業量を縦軸にとり、横軸に時間の流れをとる。	③

※参考元

- ① Ken Schwaber and Jeff Sutherland, スクラムガイド スクラム完全ガイド :ゲームのルール, 2013年7月, <<http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-JA.pdf>>
- ② 株式会社アイネット, ユーザー・ストーリー, <<http://www.e-ainet.com/UserStory.html>>
- ③ 貝瀬岳志・原田勝信・和島史典・栗林健太郎・柴田博志・家永英治, スクラム実践入門 成果を生み出すアジャイルな開発プロセス, 技術評論社, 2015

付録④ メトリクス定義フレームワーク

【Goal】

測定を行う目的の定義。測定の対象，理由，観点，およびコンテキストを定めるもの。

カテゴリー

目的

【Questionnaire】

目的の達成のために評価すべき質問の定義。
ゴールが達成されたか否かを判断する具体的な基準。

質問項目

【Metrics】

前提となるモデルに基づいて測定可能なメトリクス。
上記の質問に対する答えを提供するのに必要な情報を提供する。

導出尺度

前提条件

判定基準値

判定時期

収集時期

基礎尺度

前提条件

入力時期

付録⑤ 評価項目(技術編) 「ISO25010品質特性」をもとにした確認項目の優先付け

①品質特性の観点より、品質問題の解決に必要なメトリクスを「品質メトリクスセット(METI ソフトウェアメトリクス高度化プロジェクト)」と「HIS Source Code Metrics」を参考に検討。
 ②各研究員が、検討されたメトリクスをPOの立場で重要性を採点。
 ③各研究員の採点結果を集計し優先付けを実施(集計欄のハンデング箇所)。

○非常に有効:3点
 △ある程度有効:1点
 ×有効でない:0点

参考情報「品質メトリクスセット(METI ソフトウェアメトリクス高度化プロジェクト)」

品質特性	品質新特性	メトリクス	Goal (目的)	Question (質問項目)	Metrics (測定指標)	研究員別の採点						集計					
						A	B	G	D	E	F	O	△	×	点数		
1 機能適合性	11 完全性	Fu-1 機能実装の完全性	明示された状況下で使用するときに、明示的ニーズ及び暗黙的ニーズを満足させる機能を、製品又はシステムが提供する度合い。	要求仕様書に記述された機能内容がどのくらい欠落なく正しく実装されているか。	評価中に検出された欠落機能数と、要求仕様書に記述された機能数を比較する。	△	△	○	○	×	×	○	3	2	1	11	
				作成したモジュールは要求を満足しているか?	評価者アンケートの結果 → 5段階評価で3未満の評価が全体の20%未満	×	×	×	×	×	×	×	0	0	6	0	
	12 正確性	Fu-4 マニュアル記述の正確性	Fu-5 期待に対する正確性	実際の動作内容に対してどのくらいマニュアルの記述内容が一致しているか。	例)対象となる全項目数と、利用者がマニュアルの記述と実際の動作との差異が許容範囲を超えていると判断した項目数を比較する。	△	△	×	×	×	△	×	0	3	3	3	
				全試験項目数に対して、どれくらい合理的に期待される結果が実際得られているか?	試験を実施し、妥当な期待された結果と受け入れ難い相違があると結果の数を比較する。	×	×	×	×	△	×	0	2	4	2		
	13 適切性	Fu-8 機能適切性	Fu-11 機能仕様の安定性	明示された状態(条件)で使用するときの資源の量に関する性能の度合い。	確認すべき機能数に対して、問題なく実装された機能はどれくらいか?	要件を満たす機能が適切に実装されているかを評価する。	×	○	△	×	×	×	1	1	4	4	
初期要件があまりでないか?					モジュール作成中に仕様変更の発生件数 モジュール作成中の課題の発生件数	○	△	△	×	○	○	3	2	1	11		
2 性能適合性	21 時間効率性	Pe-1 応答時間	明示された状態(条件)で使用するときの資源の量に関する性能の度合い。	レスポンス性能は出ているか?	オンラインレスポンス・パッチレスポンス実測値 → 目標値以内/目標値を大きく超えた機能については妥当性の了解が得られているか?	△	×	×	×	○	○	2	1	3	7		
				単位時間あたり、どれくらいタスク(作業)が処理できるか?	単位時間当たりのタスクの数	○	○	○	×	○	×	4	0	2	12		
	22 資源利用率	Pe-7 メモリ容量	Pe-8 (サーバ等の)ハードディスク容量	Pe-10 CPU利用率	Pe-19 (最大)同時アクセス数	特定の処理を行う際に必要とするコンピュータの記憶容量。	メモリ利用率	○	○	△	×	○	×	3	1	2	10
						特定の処理を行う際に必要とするサーバ等のハードディスクの容量。	ディスク利用率	×	×	△	×	×	×	0	1	5	1
	○	○	△	×	○	×	3	1	2	10							
○	×	×	×	○	○	3	0	3	9								
3 互換性	32 相互運用性	Co-3 インタフェース一貫性(プロトコル)	同じハードウェア環境又はソフトウェア環境を共有する際、製品、システム又は構成要素が他の製品、システム又は構成要素の情報を交換することができる度合い、及び/又はその要求された機能を実行することができる度合い。	仕様で定められたインタフェースプロトコルに対して、どれくらい正しい実装されているか?	X=A/B A=試験期間中に仕様通り実装された確認できたインタフェースプロトコルの数 B=仕様書に記述された実装すべきインタフェースプロトコルの数	○	×	×	○	△	○	3	1	2	10		
				入出力データの項目数に対して、どれくらいの項目を利用者が理解できるか?	X=A/B A=利用者がうまく理解できる入出力データ項目の数 B=インタフェースから利用可能な入出力データ項目の数	○	△	×	△	△	△	1	4	1	7		
				実装した機能が要求仕様を完全に満たしているか?	評価中に検出された欠落機能数 要求仕様書に記述された機能数	×	○	×	×	○	△	2	1	3	7		
				運転の開始、中断、終了などの運用条件がどのくらい明確にされているか?	△	△	×	×	○	○	2	2	2	8			
				要求事項を満たすために実施すべき試験に対して、実際にどれくらいの前で実施(もしくはレビュー)したか?	X=A/B A=試験期間中に操作シナリオに従った試験ケースを実際に実施した件数 B=要求事項を満たすために実施すべき試験ケース数	×	△	×	△	△	○	1	3	2	6		
○	△	○	×	×	○	3	1	2	10								
4 使用性	41 適切認識度	Us-3 理解可能な入出力	明示された利用状況において、有効性、効率性及び満足性をもって明示された目標を達成するために、明示された利用者が製品又はシステムを利用することができる度合い。	入出力データの項目数に対して、どれくらいの項目を利用者が理解できるか?	X=A/B A=利用者がうまく理解できる入出力データ項目の数 B=インタフェースから利用可能な入出力データ項目の数	○	△	×	△	△	△	1	4	1	7		
				実装した機能が要求仕様を完全に満たしているか?	評価中に検出された欠落機能数 要求仕様書に記述された機能数	×	○	×	×	○	△	2	1	3	7		
	43 運用性	Us-9 システムレベルの監視	Us-10 運用開始条件等の明確化の割合	運転の開始、中断、終了などの運用条件がどのくらい明確にされているか?	△	△	×	×	○	○	2	2	2	8			
				要求事項を満たすために実施すべき試験に対して、実際にどれくらいの前で実施(もしくはレビュー)したか?	X=A/B A=試験期間中に操作シナリオに従った試験ケースを実際に実施した件数 B=要求事項を満たすために実施すべき試験ケース数	×	△	×	△	△	○	1	3	2	6		
	○	△	○	×	×	○	3	1	2	10							
5 信頼性	51 成熟性	Re-2 障害除去	明示された時間帯で、明示された条件の下に、システム、製品又は構成要素が明示された機能を実行することができる度合い。	要求事項を満たすために実施すべき試験に対して、実際にどれくらいの前で実施(もしくはレビュー)したか?	X=A/B A=試験期間中に操作シナリオに従った試験ケースを実際に実施した件数 B=要求事項を満たすために実施すべき試験ケース数	×	△	×	△	△	○	1	3	2	6		
				個々のプログラム別のテスト実施時のステートメント単位のバリエーションはどのくらいか?	CO:命令網羅率(ステートメントカバレッジ) CI:分岐網羅率(ブランチカバレッジ) C2:条件網羅率(コンジャンクカバレッジ)	×	×	×	×	×	×	0	0	6	0		
				要求事項を満たすために実施すべき試験に対して、どれくらいレビュー、試験を実施したか?	×	△	×	×	×	×	0	1	5	1			
				ソフトウェアの単位規模(1KLOC等)あたりどれくらい欠陥を抽出したか?	単位量当たりのテストで抽出された欠陥の密度	○	△	○	○	×	○	4	1	1	13		
				レビューでどれくらい欠陥を抽出したか?	レビューで抽出された欠陥の数	×	△	×	×	×	×	0	1	5	1		
○	△	×	×	×	×	×	×	0	1	5	1						
6 セキュリティ	61 機能保持性	Sa-2 アクセス監査性	人間又は他の製品若しくはシステムが、認められた権限の種類及び水準に応じてデータアクセスの度合いを制御できるように、製品又はシステムが情報及びデータを保護する度合い。	不正アクセスを制御するための機能数に対して、どれくらい割合で機能が実装されているか?	X=A/B A=仕様書のとおり正確に実装されているアクセス制御機能の数 B=仕様書のアクセス制御機能の数 A=試験で検出された不正操作の種類の件数 B=試験として実施した不正操作種類の件数	△	×	×	×	○	×	1	1	4	4		
				Webアプリケーション特有の脅威、脆弱性に関する対策の内容や回数	×	×	×	×	×	×	0	0	6	0			
				クラスやパッケージ内の機能要素と情報要素間の関連性はどれくらい強いのか?	オブジェクト指向言語のソースコードの凝集度を図る指標として、LOCM(Lack of Cohesive Methods)がある。	×	×	×	×	×	×	0	0	6	0		
				コーディング規約標準として組織として定められた開発標準の関連する全項目数に対して、どれくらい割合で標準を適用したか?	適合項目数/適合要求項目数	×	×	×	×	×	×	0	0	6	0		
				開発容易性向上につながる保守ドキュメントについて、実際に用意できているドキュメントはどのくらいか?	例)機能仕様書、DBクrosリファレンス、データ項目クロスリファレンス、トランザクションリファレンス、変更手順書(組織変更、制度変更、限度額変更、等のドキュメント数	○	×	×	×	○	○	3	0	3	9		
7 保守性	71 モジュール性	Ma-1 凝集度	意図した保守者によって、製品又はシステムが修正することができる有効性及び効率性の度合い。	クラスやパッケージ内の機能要素と情報要素間の関連性はどれくらい強いのか?	オブジェクト指向言語のソースコードの凝集度を図る指標として、LOCM(Lack of Cohesive Methods)がある。	×	×	×	×	×	×	0	0	6	0		
				コーディング規約への適合率	適合項目数/適合要求項目数	×	×	×	×	×	×	0	0	6	0		
				保守ドキュメント充足	例)機能仕様書、DBクrosリファレンス、データ項目クロスリファレンス、トランザクションリファレンス、変更手順書(組織変更、制度変更、限度額変更、等のドキュメント数	○	×	×	×	○	○	3	0	3	9		
				一つのハードウェア、ソフトウェア又はシステムが修正することができる有効性及び効率性の度合い。	ユーザの指定によるオープンソース製品や第三者製品(OS/OS/OS)などの採用によりサポート内容への影響があるか。	例)特定製品の指定がない、一部に特定製品の指定がある、サポートが困難な製品の指定がある	○	×	×	×	×	0	0	6	0		
				各言語へどれくらい対応しているか?	言語数	○	○	×	×	△	×	2	1	3	7		

参考情報「HIS Source Code Metrics」

品質特性	品質新特性	メトリクス	Goal (目的)	Question (質問項目)	Metric (測定指標)	研究員別の採点						集計			
						A	B	G	D	E	F	O	△	×	点数
7 保守性	73 解析性	コメント密度 "COMF" "NOMV" "NOMVPR"	意図した保守者によって、製品又はシステムが修正することができる有効性及び効率性の度合い。	ステートメント数に対するコメント数の割合	コードのわかりやすさや明確さ。	×	×	×	×	×	×	0	0	6	0
				HIS Subset MISRA C 1.0.2のルール違反の総数	ソフトウェア開発プロセスにおける、HISルールの遵守適合	×	×	×	×	×	×	0	0	6	0
				HIS Subset MISRA C 1.0.2のルール毎の違反数	ソフトウェア開発プロセスにおける、HISルールの遵守適合	×	×	×	×	×	×	0	0	6	0
				非周期的なパス数(必要なテストケース数の最小値)	PATHを低減するための尺度、回数ごとに割って、サブ回数ごとに表示する。	×	×	×	×	×	×	0	0	6	0
				スタビリティインデックス "SI"	スタビリティインデックスは、ソフトウェア製品の2つの(バージョン)の変更数(変更/削除/追加)に対する尺度を提供する。	○	○	×	○	×	△	3	1	2	10
	74 変更性	パス数 "PATH" サイクロマティック複雑度 "V(G)" コールする関数数 "CALLING" コールされる関数数 "CALLS" 関数/パラメータ数 "PARAM" 関数毎の命令数 "STMT" リターンポイント数 "RETURN" ラングリップスコープ "VOCF"	サイクロマティック数の定義に従って	この関数がどれだけのサブ関数にコールされているか?	√(G)を減少させるための尺度、サブ関数へ、またサブ関数にベージングする関数全体に割り当て	△	○	×	△	×	×	1	2	3	5
				当該関数がどれだけのサブ関数にコールされているか?	システム全体を分析した際に、1%の範囲で意味を成すサブシステムとライブラリは0%5%の範囲が許容される。完成されたシステムの場合、main()はスタートアップコードによってコールされるので除外する。	△	○	×	△	×	△	1	3	2	6
				この関数は、どのように多くの異なるサブ関数を呼び出しているか? 同じサブ関数のコールは1つだけカウントする	関数の深さを評価する	△	○	×	△	×	△	1	3	2	6
				関数インタフェースがどれくらい複雑なのか?	関数の複雑度で、スタック構造や配列の複雑度が明確でない際にこの方法が必要となる	×	△	×	×	×	×	0	1	5	1
				関数内のリターンポイントの数	関数の複雑度と保守性で、関数のリターン箇所をカウントする。固有のリターンステートメントがない関数は0となる	×	×	×	×	×	×	0	0	6	0
75 試験性	Go toステートメント数 "GOTO" 再帰数 "sp.cq.cycle"	再帰数 "sp.cq.cycle"	再帰数 "sp.cq.cycle"	ラングリップスコープは、関数の保守/変更のコストを指標となる	高い値=似通ったまたは重複したコードがあるサブ関数のIn/Outのコールを考慮する必要がある	×	×	×	×	×	×	0	0	6	0
				Go toステートメントの数	Go toステートメントの数	×	×	×	×	×	×	0	0	6	0
				1つ以上の関数上の再帰数	1つ以上の関数上の再帰数	×	×	×	×	×	×	0	0	6	0

付録⑦-1 スクラム用PO勘所一覧表（技術編）

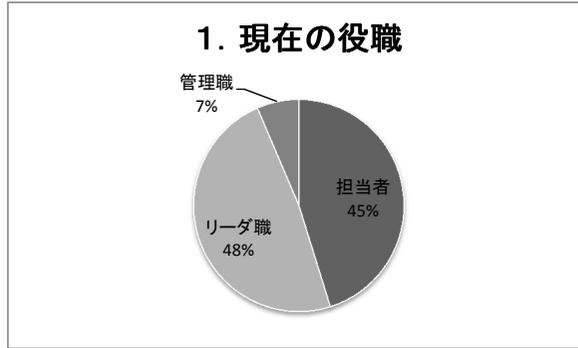
ID	メトリクス名	目的	質問項目	前提条件	測定尺度	判定基準値	収集時期	判定時期	基礎尺度
◆機能適合性－機能完全性・機能適切性									
DMs1	要件実装率	当該スプリントで実装すべき要件が、全て（欠落なく正しく）実装されていることの確認。	当該スプリントで実装すべき要件が、全て実装されたか？	当該スプリントで実装すべき要件がスプリントバックログとしてリスト化されていること。	要件実装率 [%] 「当該スプリントに割り当てられたスプリントバックログ数」のうち「当該スプリントで実装完了したスプリントバックログ数」	<100%> 当該スプリント内で実装できなかった要件がある場合は…POはプロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs1
DMs2	仕様変更の発生度	当該機能の要件が曖昧でないことの確認。（その1） 「必要なユーザー要求を漏れなく捕捉できているか。」	当該スプリントで完成した機能について、要件定義作業以降の仕様変更はどれくらい発生したか？	仕様変更の管理方法が確立していること。 当該機能に対する仕様履歴の管理が正しく運用されていること。 (例・スプリントバックログに関連する仕変更IDを設定したタスクを追加)	機能毎の仕様変更の発生件数	<基準値は P J 計画時に設定> 基準値を超える仕様変更がある機能については…POがその仕様見直しを要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs2
◆性能効率性－時間効率性・資源利用率									
DMs3	スループット	当該機能の処理能力が、非機能要件を満たしていることの確認。（単位時間当たりの処理量）	当該機能は単位時間あたり、どれくらいのタスク（作業）を処理できるか？	評価対象機能毎に期待する目標値（目標レベル）が明らかになっていること。	スループット・目標レベル スループットの実測値の平均をもとに、スループットの余裕率を6段階のレベルで整理する。（期待処理量の何倍処理できるか） レベル0：1.0倍（余裕無し） レベル1：1.2倍 レベル2：1.5倍 レベル3：2.0倍 レベル4：3.0倍 レベル5：10.0倍以上	<目標レベルは P J 計画時に設定> 目標レベルに満たない機能については…POがその性能改善の要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	対象機能の機能テスト完了時	スプリントレビュー時	BMs7
DMs4	メモリ利用率	当該機能が使用する主記憶容量が、非機能要件を満たしていることの確認。	当該機能はどれくらいメモリを占有するか？	評価対象機能毎に期待する目標値（目標レベル）が明らかになっていること。	メモリ利用率・目標レベル メモリ利用率の実測値の平均を4段階のレベルで整理する。 レベル0：80%以上 レベル1：50%以上 レベル2：20%以上 レベル3：20%未満	<目標レベルは P J 計画時に設定> 目標レベルに満たない機能については…POがその性能改善の要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	対象機能の機能テスト完了時	スプリントレビュー時	BMs8
DMs5	CPU利用率	当該機能が使用するCPU時間が、非機能要件を満たしていることの確認。	当該機能はどれくらいCPUを占有するか？	評価対象機能毎に期待する目標値（目標レベル）が明らかになっていること。	CPU利用率・目標レベル CPU利用率の実測値の平均を4段階のレベルで整理する。 レベル0：80%以上 レベル1：50%以上 レベル2：20%以上 レベル3：20%未満	<目標レベルは P J 計画時に設定> 目標レベルに満たない機能については…POがその性能改善の要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	対象機能の機能テスト完了時	スプリントレビュー時	BMs9
◆互換性－相互運用性									
DMs6	インタフェース実装率	当該機能に必要なインタフェース要件が、全て（欠落なく正しく）実装されていることの確認。（インタフェース＝貫性）	当該機能は仕様で定められたインタフェースプロトコル数に対して、どれくらい正しく実装されているか？	実装すべきインタフェースプロトコルが仕様書に記載されていること。	インタフェース実装率 [%] 「レビューで仕様通り実装された確認できたインタフェースプロトコルの数」/「仕様書に記載された実装すべきインタフェースプロトコルの数」	<基準値は P J 計画時に設定> 実装済がある場合は…POがその対応の是非を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs10
◆信頼性－成熟性									
DMs7	テストカバレッジ	当該プログラムの単体テストがテスト網羅性を満たしていることの確認。 「各プログラムのテストカバレッジはどのくらいか。」	当該プログラムはC1カバレッジ（ブランチカバレッジ）を満たしているか？ コード内の全てのブランチを一回以上実行しているか？	計測対象のプログラムが完成していること。	C1カバレッジ：分岐網羅率 [%] 「実行した分岐数」/「実行可能な分岐数」	<100%> 100%に達しない場合は…再テストを実施するが、もしくはソースコードの実行行を自目で確認する。	各スプリントバックログの実装完了時	スプリントレビュー時	BMs11
DMs8	バグ密度	当該機能における結合テストがテスト充足度を満たしていることの確認。 「想定される量の潜在バグを発見したか。」	当該機能のバグ密度はどれくらいか？ ソフトウェアの単位規模（1 KLOC等）当たり何件の欠陥を検出したか？	当該機能のテスト（発見バグの改修および回帰テスト）が完了していること。	バグ密度 [%] 単位量当たりのテストで検出された欠陥「バグ件数」/「規模(KLOC)」	<基準値は P J 計画時に設定> バグ密度が基準範囲内でない場合は…再テストを実施するが、もしくは発見バグの分析を行い、必要十分な品質が出ている事を確認する。	各スプリントバックログの実装完了時	スプリントレビュー時	BMs12, BMs13
◆保守性－モジュール性									
DMs9	スタビリティインデックス "SI"	当該コンポーネントのコードが複雑でないことの確認。（その2） 「スタビリティインデックス」	当該コンポーネントに、下記の条件にあてはまるモジュールがいくつあるか？ 「スタビリティインデックスの値が、0以上～1以下の範囲外」	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	スタビリティインデックス"SI"の値が「0 ≤ SI ≤ 1」の範囲外のモジュール数 スタビリティインデックスとは（ソフトウェア部品の2つの「-」間の変更数（変更/削除/追加）に対する尺度） 基礎尺度：(BMs3-(BMs4+BMs5+BMs6))/BMs3	<基準値は P J 計画時に設定> 該当モジュール数が基準値以上のコンポーネントは…POがSMと協議の上リファクタリングの要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs3, BMs4, BMs5, BMs6
DMs10	サイクロマティック複雑度 "v(G)"	当該コンポーネントのコードが複雑でないことの確認。（その1） 「サイクロマティック複雑度」 (Logiscope: ct_vg, VG, ct_cyclo, QA-C: STCYC)	当該コンポーネントに、下記の条件にあてはまるモジュールがいくつあるか？ 「サイクロマティック複雑度の値が、1以上～10以下の範囲外」	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	サイクロマティック複雑度"v(G)"の値が「1 ≤ v(G) ≤ 10」の範囲外のモジュール数	<基準値は P J 計画時に設定> 該当モジュール数が基準値以上のコンポーネントは…POがSMと協議の上リファクタリングの要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs14
DMs11	コールする関数数 "CALLING"	当該コンポーネントのモジュールが複雑でないことの確認。（その1） 「関数のコールされる数」 (Logiscope: dc_calling, NBCALLING, QA-C: STM29)	当該コンポーネントに、下記の条件にあてはまるモジュールがいくつあるか？ 「関数のコールされる数が、6以上」	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	他の関数から6以上コールされている関数の数	<基準値は P J 計画時に設定> 該当モジュール数が基準値以上のコンポーネントは…POがSMと協議の上リファクタリングの要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs15
DMs12	コールされる関数数 "CALLS"	当該コンポーネントのモジュール構造が複雑でないことの確認。（その2） 「関数をコールする数」 (Logiscope: dc_calls, DRCT_CALLS, QA-C: STCAL)	当該コンポーネントに、下記の条件にあてはまるモジュールがいくつあるか？ 「関数をコールする数が、8以上」	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	他の関数を8以上コールしている関数の数（同じサブ関数のコールは1つだけカウントする）	<基準値は P J 計画時に設定> 該当モジュール数が基準値以上のコンポーネントは…POがSMと協議の上リファクタリングの要否を決定し、プロダクトバックログ及び次のスプリントバックログを修正する	各スプリントバックログの実装完了時	スプリントレビュー時	BMs16

付録⑦-2 スクラム用PO勘所一覧表（技術編）
 基礎尺度（技術編）

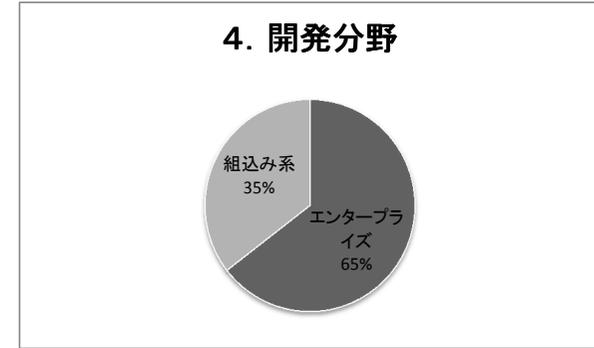
ID	基礎尺度	前提条件	入力時期
BMs1	スプリントに割り当てられた要件数(スプリントバックログ)	「スプリントバックログ」について以下の管理運用が実施されていること。 ① P Oは、「ユースケース」で製品要件を分析後、優先付けを行い、各スプリントで実装するべき「フィーチャ」を割り当て、その結果を「プロダクトバックログ」としてシフト化する。 ② S Mは、優先付けされた「フィーチャ」を、各スプリントで実装するようにチームを導き、実装した結果を「スプリントバックログ」に記録する。	各スプリントバックログの実装完了時
BMs2	モジュール毎の仕様変更発生数	「スプリントバックログ」について以下の変更管理運用が実施されていること。 ① S Mまたは S Tは、仕様変更が発生した場合、「スプリントバックログ」に関連するタスクを追加する。 ② S Mまたは S Tは、追加したタスクに関連するモジュールを特定する。	各スプリントバックログの実装完了時
BMs3	STMT 関数毎の命令数 (Logiscope: lc_stat, STMT, QA-C: STST3)	「スタビリティンデックス」について以下の管理運用が実施可能なこと。 ① S Mは、スプリント毎にソフトウェアのバージョンを設定し、管理している（構成管理をしている） ② S Mは、ソフトウェアのバージョン間の差分をソースコードのステートメント単位に抽出することができる ③ ソフトウェアの静的解析ツール (LogiscopeまたはQA-C) を導入している	各スプリントバックログの実装完了時
BMs4	Schange 現バージョンと1つ前のバージョン間で変更されたモジュールのステートメント数	「スタビリティンデックス」について以下の管理運用が実施可能なこと。 ① S Mは、スプリント毎にソフトウェアのバージョンを設定し、管理している（構成管理をしている） ② S Mは、ソフトウェアのバージョン間の差分をソースコードのステートメント単位に抽出することができる ③ S Mまたは S Tは、フィーチャを実現するソフトウェアを、ソースコードのモジュール単位に特定することができる ④ ソフトウェアの静的解析ツール (LogiscopeまたはQA-C) を導入している	各スプリントバックログの実装完了時
BMs5	Sdel 現バージョンと1つ前のバージョン間で削除されたモジュールのステートメント数	「スタビリティンデックス」について以下の管理運用が実施可能なこと。 ① S Mは、スプリント毎にソフトウェアのバージョンを設定し、管理している（構成管理をしている） ② S Mは、ソフトウェアのバージョン間の差分をソースコードのステートメント単位に抽出することができる ③ S Mまたは S Tは、フィーチャを実現するソフトウェアを、ソースコードのモジュール単位に特定することができる ④ ソフトウェアの静的解析ツール (LogiscopeまたはQA-C) を導入している	各スプリントバックログの実装完了時
BMs6	Snew 現バージョンと1つ前のバージョン間で追加されたモジュールのステートメント数	「スタビリティンデックス」について以下の管理運用が実施可能なこと。 ① S Mは、スプリント毎にソフトウェアのバージョンを設定し、管理している（構成管理をしている） ② S Mは、ソフトウェアのバージョン間の差分をソースコードのステートメント単位に抽出することができる ③ S Mまたは S Tは、フィーチャを実現するソフトウェアを、ソースコードのモジュール単位に特定することができる ④ ソフトウェアの静的解析ツール (LogiscopeまたはQA-C) を導入している	各スプリントバックログの実装完了時
BMs7	当該機能のスループット 単位時間に実施できた当該機能の処理数。（例：データエントリー数/時間）	評価対象機能毎に以下の事項が明らかになっていること。 ・評価するシステム環境 ・実行するプログラム ・投入データの内容と量 測定自体も、開発環境ではなく、評価するシステム環境で行う。	各スプリントバックログの実装完了時
BMs8	当該機能のメモリ利用率 当該機能を実行した時、当該機能を構成するプログラムが、単位時間にメモリを使用している割合。	評価対象機能毎に以下の事項が明らかになっていること。 ・評価するシステム環境 ・実行するプログラム ・投入データの内容と量 測定自体も、開発環境ではなく、評価するシステム環境で行う。	各スプリントバックログの実装完了時
BMs9	当該機能のCPU利用率 当該機能を実行した時、当該機能を構成するプログラムが、単位時間にCPUを使用している割合。	評価対象機能毎に以下の事項が明らかになっていること。 ・評価するシステム環境 ・実行するプログラム ・投入データの内容と量 測定自体も、開発環境ではなく、評価するシステム環境で行う。	各スプリントバックログの実装完了時
BMs10	当該機能のインタフェースプロトコルの数	当該機能のインタフェースプロトコルが仕様に記述されていること。	各スプリントバックログの実装完了時
BMs11	当該プログラムの分岐数	当該プログラムが完成していること。 当該プログラムで「実行可能な分岐数」が測定可能な事こと。 当該プログラムのテストで「実行した分岐数」が測定可能な事こと。	各スプリントバックログの実装完了時
BMs12	当該機能のバグ件数	当該機能の構成プログラムが完成していること。 当該機能のテストで検出したバグが管理されていること。	各スプリントバックログの実装完了時
BMs13	当該機能の規模(KLOC)	当該機能の構成プログラムが完成していること。 当該機能の構成プログラムの規模(ステップ数)が測定可能な事こと。	各スプリントバックログの実装完了時
BMs14	サイロマティック複雑度 Logiscopeの場合・・・ct_vg, VG, ct_cyclo QA-Cの場合・・・STCYC	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	各スプリントバックログの実装完了時
BMs15	関数がコールされる数 Logiscopeの場合・・・dc_calling, NBCALLING QA-Cの場合・・・STM29	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	各スプリントバックログの実装完了時
BMs16	関数をコールする数 Logiscopeの場合・・・dc_calls, DRCT_CALLS QA-Cの場合・・・STCAL	ソフトウェアの静的解析ツールを導入していること。（LogiscopeまたはQA-C） ソフトウェア構造上、モジュールをまとめたコンポーネントが定義されていること。	各スプリントバックログの実装完了時

付録⑨-1集計結果

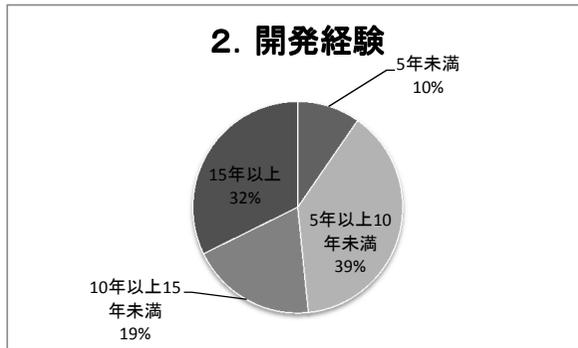
選択肢	ラベル名	一般質問 1
1	担当者	14
2	リーダー職	15
3	管理職	2
4	役員	0
5	その他	0



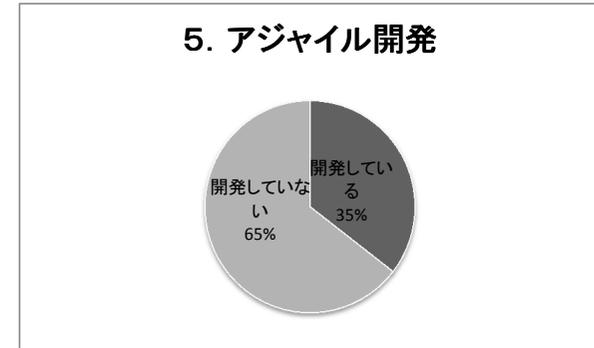
選択肢	ラベル名	一般質問 4
1	エンタープライズ	20
2	組込み系	11
3		0
4		0
5		0



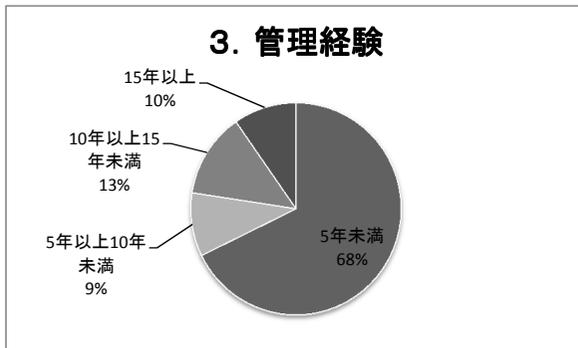
選択肢	ラベル名	一般質問 2
1	5年未満	3
2	5年以上10年未満	12
3	10年以上15年未満	6
4	15年以上	10
5		0



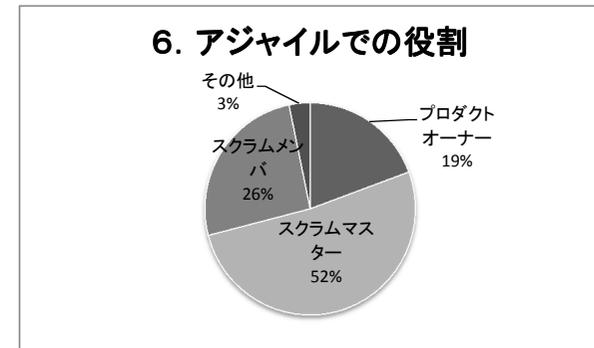
選択肢	ラベル名	一般質問 5
1	開発している	11
2	開発していない	20
3		0
4		0
5		0



選択肢	ラベル名	一般質問 3
1	5年未満	21
2	5年以上10年未満	3
3	10年以上15年未満	4
4	15年以上	3
5		0



選択肢	ラベル名	一般質問 6
1	プロダクトオーナー	6
2	スクラムマスター	16
3	スクラムメンバ	8
4	その他	1
5		0



付録⑨-2集計結果

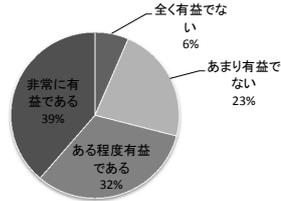
合格基準 70.00% 以上

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	1
2	あまり有益でない	7	実現が困難	7
3	ある程度有益である	10	実現できる	14
4	非常に有益である	12	容易に実現できる	9
5		0		0

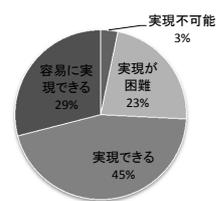
タイムマネジメント

有益である **70.97%** 実現可能 **74.19%**

DMp1 生産性(ペロシティー): 有益性



DMp1 生産性(ペロシティー): 実現可能性

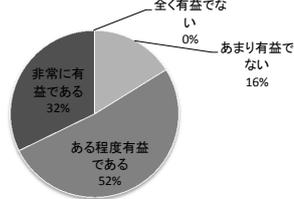


選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	5	実現が困難	8
3	ある程度有益である	16	実現できる	14
4	非常に有益である	10	容易に実現できる	9
5		0		0

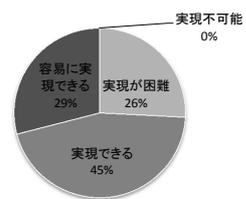
タイムマネジメント

有益である **83.87%** 実現可能 **74.19%**

DMp2 総ストーリーポイント: 有益性



DMp2 総ストーリーポイント: 実現可能性

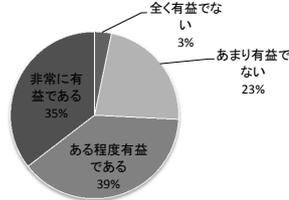


選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	5	実現が困難	9
3	ある程度有益である	12	実現できる	14
4	非常に有益である	11	容易に実現できる	8
5		0		0

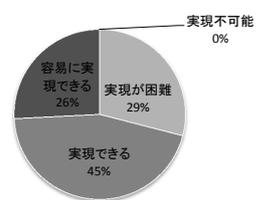
タイムマネジメント

有益である **74.19%** 実現可能 **70.97%**

DMp3 プロダクトの完成率: 有益性



DMp3 プロダクトの完成率: 実現可能性

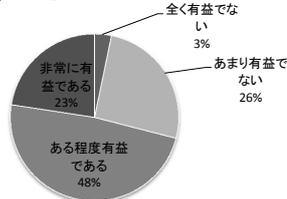


選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	8	実現が困難	9
3	ある程度有益である	15	実現できる	15
4	非常に有益である	7	容易に実現できる	7
5		0		0

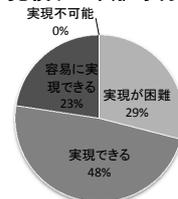
カスタマネジメント

有益である **70.97%** 実現可能 **70.97%**

DMp4 見積りの乖離: 有益性



DMp4 見積りの乖離: 実現可能性

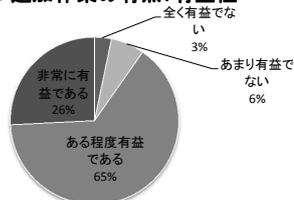


選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	2	実現が困難	3
3	ある程度有益である	20	実現できる	23
4	非常に有益である	8	容易に実現できる	5
5		0		0

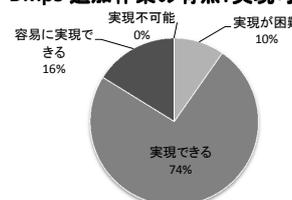
カスタマネジメント

有益である **90.32%** 実現可能 **90.32%**

DMp5 追加作業の有無: 有益性



DMp5 追加作業の有無: 実現可能性



付録⑨-3集計結果

合格基準 70.00% 以上

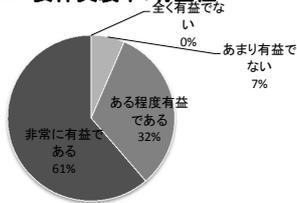
DMs1 要件実装率

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	2	実現が困難	0
3	ある程度有益である	10	実現できる	21
4	非常に有益である	19	容易に実現できる	10
5	0	0	0	0

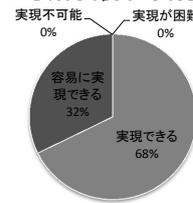
機能適合性

有益である **93.55%** 実現可能 **100.00%**

DMs1 要件実装率:有益性



DMs1 要件実装率:実現可能性



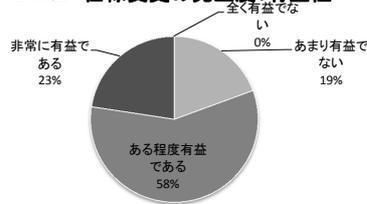
DMs2 仕様変更の発生度

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	6	実現が困難	11
3	ある程度有益である	18	実現できる	17
4	非常に有益である	7	容易に実現できる	3
5	0	0	0	0

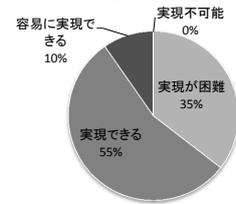
機能適合性

有益である **80.65%** 実現可能 **64.52%**

DMs2 仕様変更の発生度:有益性



DMs2 仕様変更の発生度:実現可能性



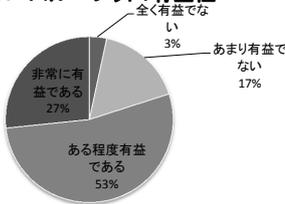
DMs3 スループット

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	5	実現が困難	17
3	ある程度有益である	16	実現できる	11
4	非常に有益である	8	容易に実現できる	2
5	0	0	0	0

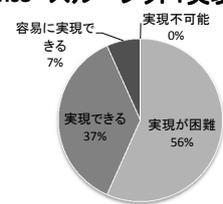
性能効率性

有益である **80.00%** 実現可能 **43.33%**

DMs3 スループット:有益性



DMs3 スループット:実現可能性



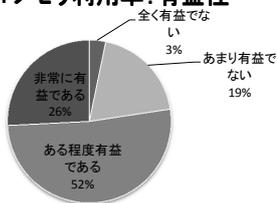
DMs4 メモリ利用率

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	6	実現が困難	13
3	ある程度有益である	16	実現できる	14
4	非常に有益である	8	容易に実現できる	4
5	0	0	0	0

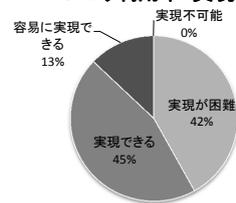
性能効率性

有益である **77.42%** 実現可能 **58.06%**

DMs4 メモリ利用率:有益性



DMs4 メモリ利用率:実現可能性



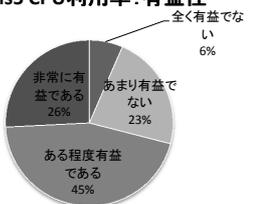
DMs5 CPU利用率

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	1
2	あまり有益でない	7	実現が困難	16
3	ある程度有益である	14	実現できる	11
4	非常に有益である	8	容易に実現できる	3
5	0	0	0	0

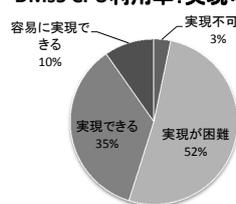
性能効率性

有益である **70.97%** 実現可能 **45.16%**

DMs5 CPU利用率:有益性



DMs5 CPU利用率:実現可能性



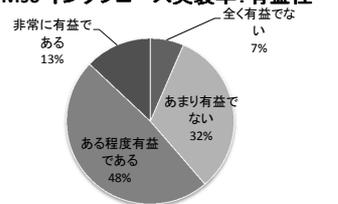
DMs6 インタフェース実装率

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	2
2	あまり有益でない	10	実現が困難	7
3	ある程度有益である	15	実現できる	15
4	非常に有益である	4	容易に実現できる	7
5	0	0	0	0

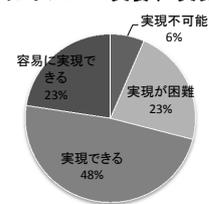
互換性

有益である **61.29%** 実現可能 **70.97%**

DMs6 インタフェース実装率:有益性



DMs6 インタフェース実装率:実現可能性



付録⑨-4集計結果

合格基準 70.00% 以上

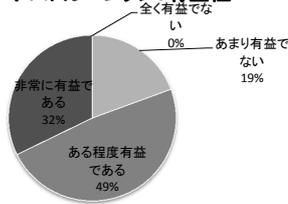
DMs7 テストカバレッジ

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	2
2	あまり有益でない	6	実現が困難	6
3	ある程度有益である	15	実現できる	18
4	非常に有益である	10	容易に実現できる	5
5		0		0

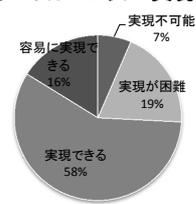
成熟性

有益である **80.65%** 実現可能 **74.19%**

DMs7 テストカバレッジ: 有益性



DMs7 テストカバレッジ: 実現可能性



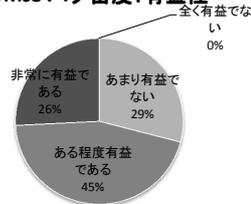
DMs8 バグ密度

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	9	実現が困難	6
3	ある程度有益である	14	実現できる	22
4	非常に有益である	8	容易に実現できる	3
5		0		0

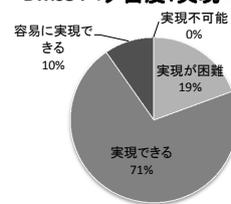
成熟性

有益である **70.97%** 実現可能 **80.65%**

DMs8 バグ密度: 有益性



DMs8 バグ密度: 実現可能性



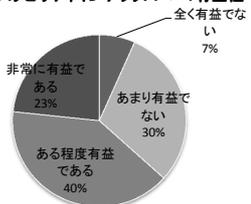
DMs9 スタビリティインデックス "SI"

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	0
2	あまり有益でない	9	実現が困難	6
3	ある程度有益である	12	実現できる	19
4	非常に有益である	7	容易に実現できる	3
5		0		0

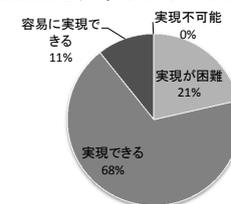
モジュール性

有益である **63.33%** 実現可能 **78.57%**

DMs9 スタビリティインデックス "SI": 有益性



DMs9 スタビリティインデックス "SI": 実現可能性



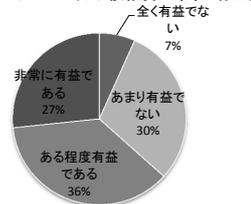
DMs10 サイクロマティック複雑度 "v(G)"

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	0
2	あまり有益でない	11	実現が困難	5
3	ある程度有益である	11	実現できる	18
4	非常に有益である	8	容易に実現できる	5
5		0		0

モジュール性

有益である **63.33%** 実現可能 **82.14%**

DMs10 サイクロマティック複雑度 "v(G)": 有益性



DMs10 サイクロマティック複雑度 "v(G)": 実現可能性



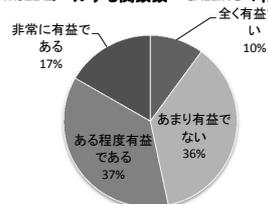
DMs11 コールする関数数 "CALLING"

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	3	実現不可能	0
2	あまり有益でない	11	実現が困難	7
3	ある程度有益である	11	実現できる	17
4	非常に有益である	5	容易に実現できる	4
5		0		0

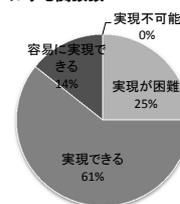
モジュール性

有益である **53.33%** 実現可能 **75.00%**

DMs11 コールする関数数 "CALLING": 有益性



DMs11 コールする関数数 "CALLING": 実現可能性



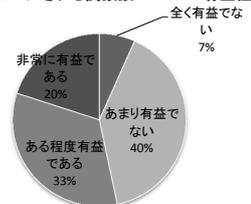
DMs12 コールされる関数数 "CALLS"

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	0
2	あまり有益でない	12	実現が困難	7
3	ある程度有益である	10	実現できる	17
4	非常に有益である	6	容易に実現できる	4
5		0		0

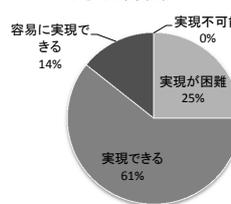
モジュール性

有益である **53.33%** 実現可能 **75.00%**

DMs12 コールされる関数数 "CALLS": 有益性



DMs12 コールされる関数数 "CALLS": 実現可能性

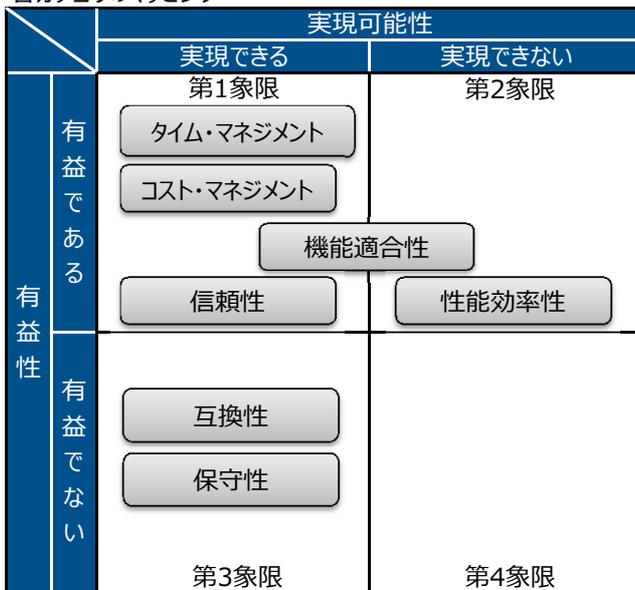


付録⑩分析結果（カテゴリ分けと傾向）

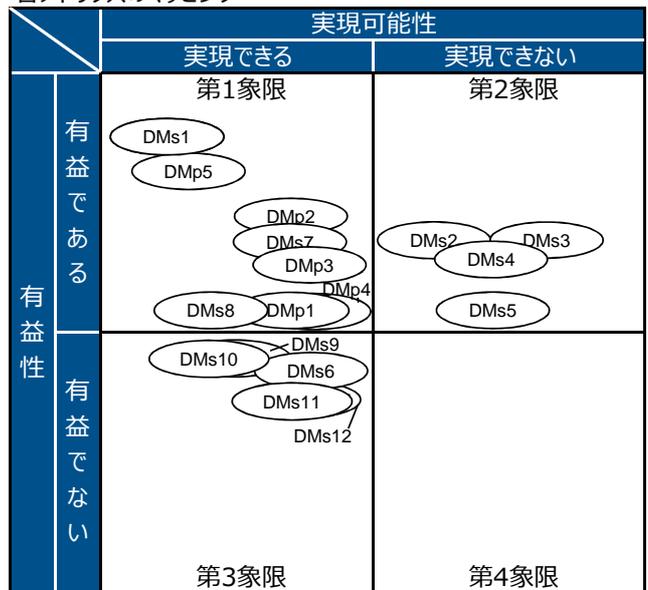
合格基準 70.00% 以上

ID	メトリクス名	カテゴリ	有益性	実現可能性	象限
DMs1	要件実装率	機能適合性	93.55%	100.00%	1
DMs2	仕様変更の発生度	機能適合性	80.65%	64.52%	2
DMs3	スループット	性能効率性	80.00%	43.33%	2
DMs4	メモリ利用率	性能効率性	77.42%	58.06%	2
DMs5	CPU利用率	性能効率性	70.97%	45.16%	2
DMs6	インタフェース実装率	互換性	61.29%	70.97%	3
DMs7	テストカバレッジ	信頼性	80.65%	74.19%	1
DMs8	バグ密度	信頼性	70.97%	80.65%	1
DMs9	スタビリティインデックス "Si"	保守性	63.33%	78.57%	3
DMs10	サイクロマティック複雑度 "v(G)"	保守性	63.33%	82.14%	3
DMs11	コールする関数数 "CALLING"	保守性	53.33%	75.00%	3
DMs12	コールされる関数数 "CALLS"	保守性	53.33%	75.00%	3
DMp1	生産性（ペロシティー）	タイム・マネジメント	70.97%	74.19%	1
DMp2	総スローポイント	タイム・マネジメント	83.87%	74.19%	1
DMp3	プロダクトの完成率	タイム・マネジメント	74.19%	70.97%	1
DMp4	見積りの乖離	コスト・マネジメント	70.97%	70.97%	1
DMp5	追加作業の有無	コスト・マネジメント	90.32%	90.32%	1

各カテゴリのマッピング



各メトリクスのマッピング



付録①分析結果（ヒストグラムと傾向分析）

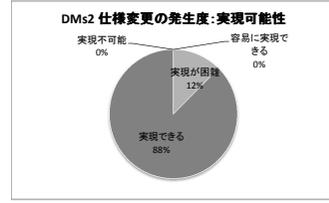
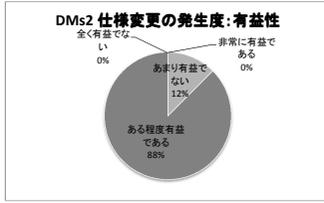
合格基準率 70.00% 以上

Dms2 仕様変更の発生度

■アジャイルでの役割：スクラムメンバ

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	1	実現が困難	1
3	中程度有益である	7	実現できる	7
4	非常に有益である	0	簡単に実現できる	0
5		0		0

機能適合性
有益である **87.50%** 実現可能 **87.50%**

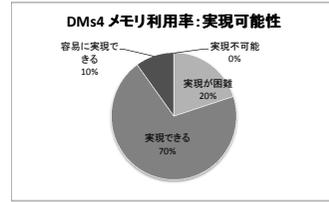
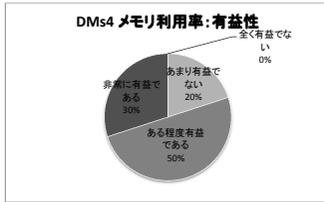


Dms4 メモリ利用率

■管理経験：5年以上

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	1	実現が困難	2
3	中程度有益である	5	実現できる	7
4	非常に有益である	3	簡単に実現できる	1
5		0		0

性能効率性
有益である **80.00%** 実現可能 **80.00%**

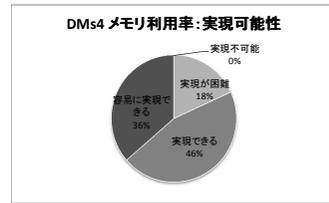
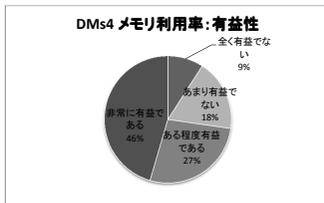


Dms4 メモリ利用率

■開発分野：組込み系

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	2	実現不可能	0
2	あまり有益でない	4	実現が困難	2
3	中程度有益である	3	実現できる	5
4	非常に有益である	5	簡単に実現できる	4
5		0		0

性能効率性
有益である **72.73%** 実現可能 **81.82%**

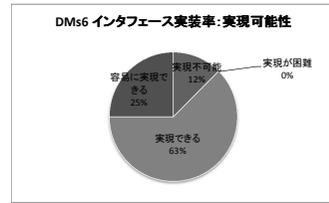
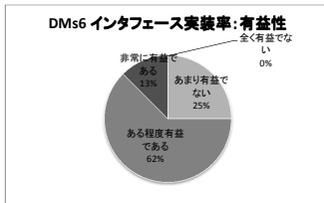


Dms6 インタフェース実装率

■アジャイルでの役割：スクラムメンバ

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	1
2	あまり有益でない	1	実現が困難	2
3	中程度有益である	5	実現できる	5
4	非常に有益である	1	簡単に実現できる	2
5		0		0

互換性
有益である **75.00%** 実現可能 **87.50%**

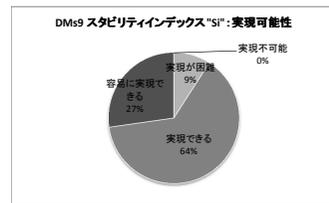
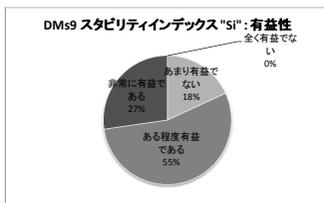


Dms9 スタビリティンデックス "Si"

■開発分野：組込み系

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	1	実現が困難	1
3	中程度有益である	6	実現できる	7
4	非常に有益である	3	簡単に実現できる	3
5		0		0

保守性
有益である **81.82%** 実現可能 **90.91%**

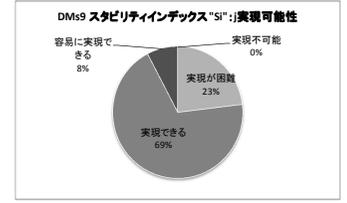
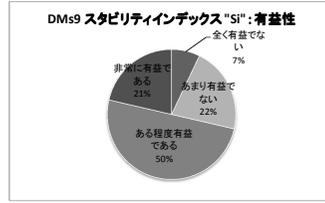


Dms9 スタビリティンデックス "Si"

■現在の役割：担当者

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	3	実現が困難	3
3	中程度有益である	7	実現できる	9
4	非常に有益である	3	簡単に実現できる	1
5		0		0

保守性
有益である **71.43%** 実現可能 **76.92%**

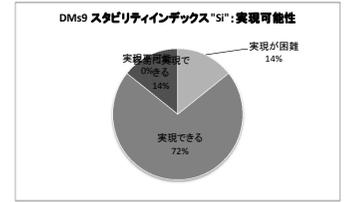
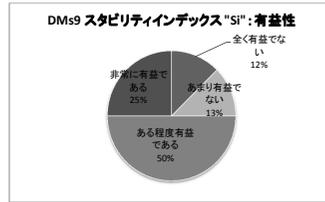


Dms9 スタビリティンデックス "Si"

■アジャイルでの役割：スクラムメンバ

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	2	実現が困難	1
3	中程度有益である	4	実現できる	5
4	非常に有益である	2	簡単に実現できる	1
5		0		0

保守性
有益である **75.00%** 実現可能 **85.71%**

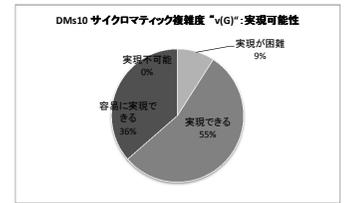
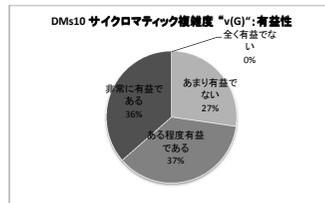


Dms10 サイクロマティック複雑度 "v(G)"

■開発分野：組込み系

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	0	実現不可能	0
2	あまり有益でない	1	実現が困難	1
3	中程度有益である	3	実現できる	6
4	非常に有益である	4	簡単に実現できる	4
5		0		0

保守性
有益である **72.73%** 実現可能 **90.91%**

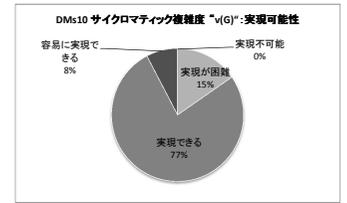
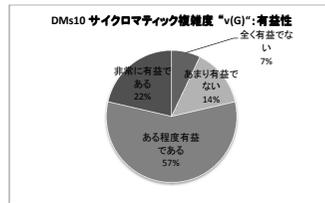


Dms10 サイクロマティック複雑度 "v(G)"

■現在の役割：担当者

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	2	実現が困難	2
3	中程度有益である	8	実現できる	10
4	非常に有益である	3	簡単に実現できる	1
5		0		0

保守性
有益である **78.57%** 実現可能 **84.62%**

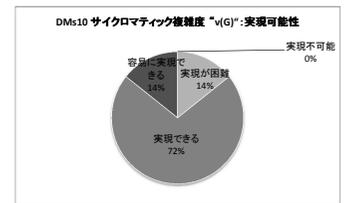
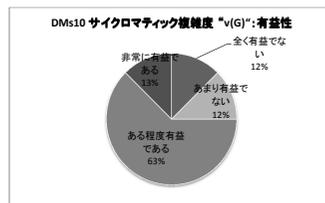


Dms10 サイクロマティック複雑度 "v(G)"

■アジャイルでの役割：スクラムメンバ

選択肢	ラベル名	有益性	ラベル名	実現可能性
1	全く有益でない	1	実現不可能	0
2	あまり有益でない	1	実現が困難	1
3	中程度有益である	5	実現できる	5
4	非常に有益である	1	簡単に実現できる	1
5		0		0

保守性
有益である **75.00%** 実現可能 **85.71%**



付録⑫分析結果（メトリクス毎の考察）

統計的に相関が強いもの（ピアソンのカイ2乗検定でp値≤0.05）を特定
ヒストグラムから見て関係が強そうな回答者のプロフィールを特定

ID	メトリクス名	カテゴリ	分析対象	統計的な相関	p 値	考察	
DMs2	仕様変更の発生度	機能適合性	実現可能性	なし	-	統計的な傾向	(なし)
						ヒストグラムから見た傾向	・管理者と現場に分けてみると、現場（スクラムメンバ）は「実現できる」感じている傾向にある。
						コメントからの考察	・仕様変更の規模や出所、管理するタイミング等、変更管理の仕組みの整備が前提として必要になる。
DMs3	スループット	性能効率性	実現可能性	一般質問 4	0.0403	統計的な傾向	・組み系系の人「実現できる」、エンタープライズ系は「実現できない」と感じている
						ヒストグラムから見た傾向	(なし)
						コメントからの考察	・組み系系 + エンタープライズ系：全てを計測することは難しく、対象を限定するべきである。 ・エンタープライズ系：計測する条件や範囲を事前に明確にするべきである。
DMs4	メモリ利用率	性能効率性	実現可能性	一般質問 4	0.0084	統計的な傾向	・組み系系の人「実現できる」、エンタープライズ系は「実現できない」と感じている。 ・管理経験5年以上の人は「実現できる」と感じている。
						ヒストグラムから見た傾向	・組み系系の人「実現できる」と感じている
						コメントからの考察	・目標値の設定等、前提条件を明確にする必要がある。 ・平均値でなくピーク値等、別の評価基準が適切である。
DMs5	CPU利用率	性能効率性	実現可能性	なし	-	統計的な傾向	(なし)
						ヒストグラムから見た傾向	(なし)
						コメントからの考察	・機能ごとにCPU利用率を評価するのではなく、全体で評価するべきである。 ・平均値でなくピーク値等、別の評価基準が適切である。
DMs6	インタフェース実装率	互換性	有益性	なし	-	統計的な傾向	(なし)
						ヒストグラムから見た傾向	・現場（スクラムメンバ）は有益だと感じているが、管理者は有益ではないと感じている。
						コメントからの考察	・インタフェースに特化した実装率ではなく、機能の実装率を評価すれば良い。
DMs9	スタビリティインデックス "Si"	保守性	有益性	なし	-	統計的な傾向	(なし)
						ヒストグラムから見た傾向	・組み系系の人「有益である」と感じている ・現在の役割が担当者である人は「有益である」と感じている
						コメントからの考察	・スクラムメンバは「有益である」と感じている ・保守性評価に対する意識向上が必要である。
DMs10	サイクロマティック複雑度 "v(G)"	保守性	有益性	なし	-	統計的な傾向	(なし)
						ヒストグラムから見た傾向	・組み系系の人「有益である」と感じている ・現在の役割が担当者である人は「有益である」と感じている
						コメントからの考察	・スクラムメンバは「有益である」と感じている ・保守性評価に対する意識向上が必要である。
DMs11	コールする関数数 "CALLING"	保守性	有益性	一般質問 5	0.0486	統計的な傾向	・アジャイル開発をしている人は有益であると感じている。
						ヒストグラムから見た傾向	(なし)
						コメントからの考察	・保守性評価に対する意識向上が必要である。
DMs12	コールされる関数数 "CALLS"	保守性	有益性	なし	-	統計的な傾向	(なし)
						ヒストグラムから見た傾向	(なし)
						コメントからの考察	・保守性評価に対する意識向上が必要である。