

組み込みシステム開発の設計者向け要件定義ガイドラインの提案と評価

Proposal and Evaluation of Requirements Definition Guidelines for Designers in Embedded System Development

研究員 : 寺村 幹夫 (株式会社デンソー)
 島田 真也 (株式会社日立ソリューションズ)
 橋本 達也 (NTTコミュニケーションズ株式会社)
主査 : 栗田 太郎 (ソニー株式会社)
副主査 : 石川 冬樹 (国立情報学研究所)
アドバイザー : 荒木 啓二郎 (熊本高等専門学校)

研究概要

従来, 新規要素の少なかった組み込み系開発では, 顧客の要求が詳細であることから, 設計者は過去の経験値に基づいて要求整理をせず, 詳細設計レベルから着手することが多い. 上流工程を簡略化したケースでは, 特に新人や途中参画者がシステムの用途を含む全体像, 仕様制約などを把握することが困難である. また, 昨今は顧客の要求が変化してきており, 例えば IoT 機器との接続といった高機能化の要求が増え, 新規性の高い開発案件が増加してきている. その結果, 欠陥の作りこみ数の増加や, 手戻り工数が膨らむ傾向がより顕著になってきた. 本研究では, これらの問題点を解消することを目的として, 新規性の高い開発案件の欠陥事例を分析し, 解決策を導出した. また, 解決策の評価実験において一定の効果があることを確認した.

1. はじめに

情報サービス業において, 要件定義工程から設計工程で作成するドキュメントの体系や内容は, 開発対象とするシステムによって大きく異なる.

例えば, 受託開発ソフトウェア業(以降, エンタープライズ系)では顧客の要求を仕様化する上でドキュメントを作成し, それを基にシステムの仕様について顧客との合意形成を図る文化が強い. そのため, 十分な時間をかけて顧客との折衝を繰り返し, 要件定義書としてシステムの仕様を明確化するプロジェクトが大半である. 一方, 組み込みソフトウェア業の請負開発(以降, 組み込み系開発)では, 顧客の要求を仕様化する, というプロセスは設計者が暗黙的に実施するものであり, 要件定義書の様な設計ドキュメントを作成する文化はあまり浸透していない. よって, プロジェクトの新規参入者はシステムの設計書やソースコードから仕様を読み取る必要がある. これは, 顧客の要求がハードウェアの制約によって定型化可能な範囲に留まっており, かつ納期も十分に長い期間が確保されていたことに起因している.

しかし, 近年の組み込み系開発における顧客の要求は, IoT 機器との接続を要求されるなど著しく高度化しており, それらを定型化することが困難となっている. さらに, 納期も従来より短い期間が要求されており, 暗黙的な要求の仕様が実行できない設計者による仕様レベルの欠陥の作り込み, その修正に伴う手戻り工数の増加が顕著である.

本研究では, この問題を解決するために, 仕様レベルの欠陥作り込み防止を目的とした要件解説書と要件定義書, および解説書を作成するためのガイドラインを提案し, それらが欠陥作り込み防止に効果があるか検証を行った. 結果, 欠陥の作り込み抑制への効果が確認できた.

以降, 2 章で本研究の課題を示し, 3 章で考案した解決策を考案する. 4 章では, 提案した解決策の評価結果と考察を示し, 5 章では, まとめと今後の展望を示す. なお, 組み込み系開発ソフトウェアを題材とした多種多様な研究や報告は多数存在するため, それらのいくつかを参考文献とした. ただし, 本研究は具体的なプロジェクトを対象と実践的な効果を狙っており, 先行研究としてプロジェクト状況が合致するものがないため, 先行研究は無いと判断した.

本稿で使用する用語を、表 1 のように定義する。

表 1 用語と意味

| 用語 | 意味 |
|----------|---|
| 上位設計者 | 顧客との折衝・設計・実装・テストの全ての経験を持ち、かつ実務経験が 20 年以上の開発従事者 |
| 設計者 | 上位設計者ではない、かつ実装担当またはテスト担当者ではない開発従事者 |
| 設計ドキュメント | 要件定義書, アーキテクチャ設計書, 詳細設計書を包含した呼称 |
| 要件 | 参考文献[1]中の仕様の一部機能, 制約といった実現すべき事柄のみを指す. 構造の実現手段については含まない |
| 欠陥 | 顧客に納入するまでに対処した問題のうち, 設計から開発段階では顕在化せず, テスト工程で発覚したもの |
| 不具合 | 顧客に納入後に問題が発生したもの |
| ECU | Electronic Control Unit の略称. 様々なセンサからの電気信号の取込み, 最適な計算を行いアクチュエータへ指示する役割を持つ制御装置 |

2. 課題設定

2.1 現状分析

エンタープライズ系のプロジェクトにおいて、顧客の要求の仕様化は要件定義の工程として規定され、要件定義書を成果物とすることは一般的である。これはシステムの仕様に対する顧客と設計者の認識齟齬を排除するとともに、勘案すべき多様な周辺システムとの関連性や制約事項の洗い出しがシステム設計をする上で重要であるとの認識による。

一方、組み込み系開発は、要求の仕様化に際して要件定義書を作成せず、詳細設計書の作成に着手してしまうプロジェクトも多い。組み込み系開発の一つであるオートモーティブ産業におけるシステム設計工程の例を図 1 に示す。

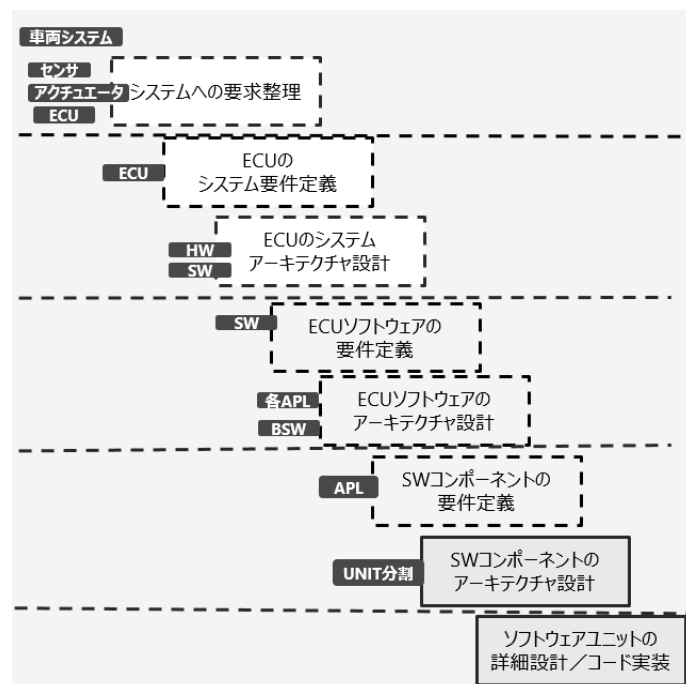


図 1 組み込み系開発の設計工程例

要求整理は、顧客の様々な要求から具体的な要件を抽出するとともに、システムの中で開発対象とするスコープを絞り込む工程である。また、要件定義では開発対象としたスコープの中で、システムに対して具体的にどういった改修をしていくか決定する工程である。

オートモーティブ産業の A 社では、顧客からの要求を受領後、コンポーネントのアーキテクチャ設計や詳細設計から着手するプロジェクトが大半である。これは、車への要求は自明であるものが多いことが一因である。例えば、アクセルを踏めば加速しブレーキを踏めば減速するなど、エンドユーザが車を運転する際の要求は明確であり、これを如何に実現するかが設計者として重要だからである。また、こういったエンドユーザの要求は普遍的なものが多いため、設計者はユーザ要求とそれを充足する仕様を自身の暗黙知として体系化し、仕様を実現するためのコンポーネントのアーキテクチャ設計や詳細設計から着手する前提で、組み込み系開発のプロセスを最適化してきた。さらに、プロジェクトの新規参入者であっても、詳細設計書から仕様やその上の顧客要求を獲得するだけの開発期間が存在した。

しかし、近年は顧客の要求は大きく変化している。例えば、コネクテッドカーなど次世代の機能に対する顧客の要求は従来にないものであり、こういった新規要求は今までの要求と比較して複雑かつ高度である。加えて、納期は年々短期化する傾向があり、新規参入者が仕様を把握する時間の確保が難しくなっている。結果、経験不足の設計者が要件の考慮不足や考慮漏れによる欠陥を作りこむ傾向が強くなっている。A 社のプロジェクトにおいて、新規性の高い機能追加時の欠陥の作り込み事例の概要を、表 2 に示す。

表 2 欠陥事例の概要

| # | 分類 | 作り込み箇所 | 欠陥内容 | 根本原因 |
|---|----------|--------------|---------------------------------|------------------|
| 1 | 機能要件考慮漏れ | SW アーキテクチャ設計 | 複数コンポーネント間で初期値設定が統一されていない | 前提条件/制限制約事項の認識齟齬 |
| 2 | 機能要件考慮漏れ | SW アーキテクチャ設計 | 複数コンポーネント間で停止 Duty 値設定が統一されていない | 前提条件/制限制約事項の認識齟齬 |
| 3 | 機能要件抽出漏れ | ECU システム要件 | 他 ECU との協調する機能考慮漏れ | 要件化視点の不足 |
| 4 | 機能要件抽出漏れ | ECU システム要件 | 強制駆動が 70%以上となった場合の駆動モードの機能考慮漏れ | 要件化視点の不足 |
| 5 | 機能要件抽出漏れ | ECU システム要件 | 通常時と強制駆動時の駆動 Duty 範囲切替の機能考慮漏れ | 要件化視点の不足 |

2.2 課題提起

一般的に、上流工程で作りこまれる欠陥は、下流工程で作りこまれた欠陥と比較して手戻り工数が多くなる。そのため、要件漏れの様な上流工程に起因する欠陥の防止は喫緊の課題である。本問題を解消する手段として、図 1 に示した工程全域の設計ドキュメントを作成していく方法があるが、この方法には以下 2 点の問題がある。

- 問題点 1. 現状の納期では設計工程全域のドキュメント作成及び保守が困難であること
- 問題点 2. 設計工程全域のドキュメントが作成可能な設計者が限られていること

問題点 1. は昨今の短納期化の中で、作成する設計ドキュメントを増やすことが難しいことに起因している。よって、作成すべき設計ドキュメントを峻別し、必要最低限にとどめる必要がある。

問題点 2. は、上流工程の設計に従事する設計者は、十分な経験や知識が必要となるからである。ドキュメントの作成方法についても設計者への教育を実施することが考えられるが、すべての教育を受講するまでに多くの工数がかかるため、こちらにも導入には課題が残る。

3. 解決策の考案

3.1 課題の解決方針

本稿では仕様レベルの欠陥の作り込み防止を目的とし、2.2 で挙げた 2 点の問題点を考慮した作成を優先すべき設計ドキュメントの選定、およびその作成方法のガイドラインを提案する。

3.2 解決策

問題点 1. への対処として、実プロジェクトの欠陥の作り込み要因を分析し、その要因を解消するためのドキュメントのみ作成すればよいと考えた。A 社のあるプロジェクトの欠陥分析結果を表 3 に示す。これは新規性の高い B 機能の設計書 250 ページを対象として業務経験が 20 年以上の上位設計者がレビューを行い、欠陥として抽出したものである。これらの欠陥はプロジェクトに従事していた業務経験が 10 年程度の設計者 13 人の、各設計工程のレビューでは抽出されなかった。

表 3 新規性の高い B 機能の欠陥分析結果

| # | 想定する欠陥内容 | 欠陥件数 | 想定する欠陥の作り込み要因 |
|---|-----------------------------|------|--|
| 1 | 共通の設定値がコンポーネント間で異なる設定となっている | 2 件 | ECU システム要件を実現するために関連する複数のコンポーネント間において ECU システム要件の認識がそれぞれで勝手な解釈をして異なる設定をしていた。 |
| 2 | 他コンポーネントへの影響考慮不足 | 1 件 | 下流のコンポーネントにおいて上流の ECU システム要件を意識していなかった。 |
| 3 | 特定状態での要件漏れ | 2 件 | ECU システムに対する要求分析不足 |

表 3 の欠陥の作り込み要因と 13 人の設計者の担当範囲及びレビュー内容から、下記の 3 点が記載されているドキュメントを作成すればよいと判断した。

- (a) システム全体の概要が整理されている
- (b) 顧客要求が記載されている
- (c) ECU システムの要件が記載されている

#1 の欠陥は ECU システムの全体概要を設計者が把握していないことが一因である。よって、全体を俯瞰できるドキュメントが必要であると考え、(a) の観点を設定した。#2 および#3 の欠陥は顧客の要求を意識せず、自身が担当する範囲のみ考慮して設計をしたことに起因している。よって、顧客要求、および顧客要求を実現するための ECU システム要件の双方を設計者へ提示することで、改修による他システムへの影響や顧客要求を実現できているか設計者が考慮するようになると考え、(b) と (c) の観点を抽出した。それぞれの観点の関連性を図 2 に示す。

(a)～(c) の観点を考慮すべき工程は、図 1 の工程例において、(a) と (b) は要求整理、(c) は要件定義となる。よって、2.2 で挙げた問題点 1. を解消するには、要求整理工程および要件定義工程のドキュメントを作成すればよいと考えた。以降、要求整理工程のドキュメントを「ECU システム要件解説書」、要件定義工程のドキュメントを「ECU システム要件定義書」と呼称する。図 1 に記載した工程例と、本稿で提案したドキュメントの関連性を図 3 および付録 1 に示す。

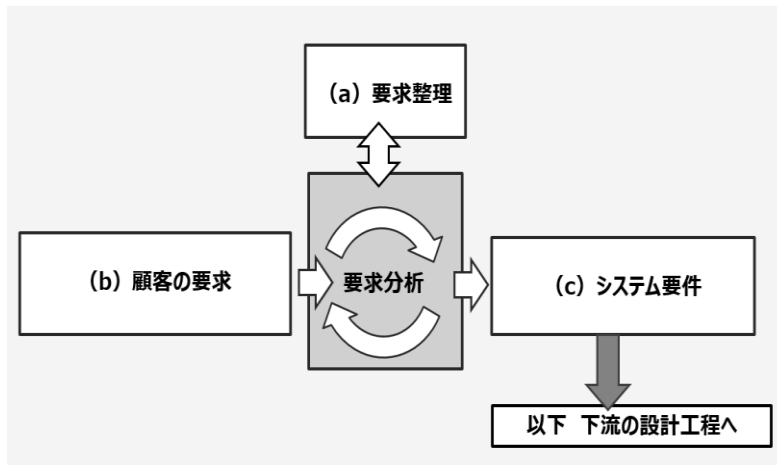


図 2 記載観点 (a) (b) (c) の関係性

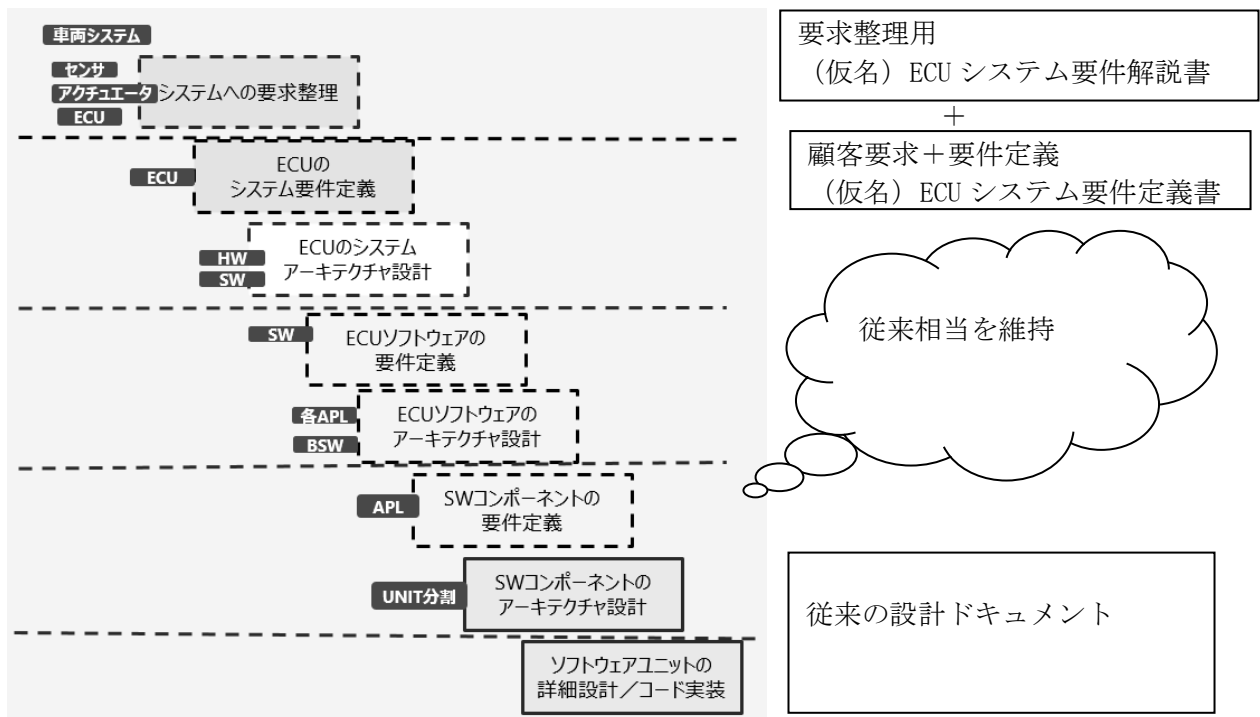


図 3 今回の解決策箇所

問題点 2. への対処としては、ECU システム要件解説書を作成するためのガイドラインを作成した。これは、設計者へ要求定義書の作成を指示するだけでは、記載内容が設計者の知識や経験に依存する可能性があるため、それを排除することを目的としている。ガイドラインは、システム共通の機能要件のパターンを暗黙的に理解している上位設計者 3 人と著者であるプロセス改善担当 1 人が共同で作成した。ガイドラインに記載した内容を付録 2 に示す。

なお、ECU システム要件解説書では、対象システム、機能のスコープの特定だけでなく、システムの外部インターフェースの特定方法、通常モードとフェイルセーフモードといった動作モードの切り替えを状態遷移図や状態遷移表を用いて整理する。一方、システム要件定義書では、システム要件解説書が取り扱っている範囲内で記載項目の範囲を機能、制約といった実現することのみに

限定している。これは実現手段となる設計検討時の選択肢の幅を拡げ、テスト設計し易いような下流工程に考慮した記載範囲としている。また、要求定義ガイドラインには、組み込み系開発で有効な要件の単文化表記にする手法などを記載している。具体的な内容を付録3に示す。

4. 解決策の評価および考察

4.1 評価方法

3.2 で述べた「ガイドライン」、および「ECU システム要件解説書」と「ECU システム要件定義書」を用いて、設計者が欠陥の作り込みを防止できるか評価する。

被験者 33 名に対して要求整理及び要件定義のトレーニングを 16 時間実施した。被験者は平均の業務経験が 12 年程度である。トレーニングに際しては、要件定義書や要件解説書の作成方法や、要件解説書のガイドラインを提示し、ドキュメントの作成方法のみをレクチャーした。なお、被験者 33 名は表 3 に挙げている B 機能の欠陥 5 件を作りこんだプロジェクトには参画しておらず、トレーニング中は欠陥への言及も行っていない。

トレーニングの完了後、被験者に対して B 機能の設計書をもとに、B 機能の要件解説書と要件定義書、および詳細設計書の一部を欠陥が出ないように開発設計する様に指示した。また、その過程で B 機能の設計書に欠陥を発見した場合は、指摘をするように指示した。表 3 で示している欠陥 5 件を基準として、各被験者が指摘した欠陥を「設計工程において作りこみを防止した欠陥」と見做して比較することで、欠陥の防止効果があるか評価する。

4.2 評価結果と考察

被験者 33 名が防止した欠陥数を表 4 に示す。

表 4 欠陥の防止結果

| 防止した欠陥の件数 | 被験者数 | 平均業務年数 | 業務年数の中央値 |
|-----------|------|----------|----------|
| 5 件 | 16 人 | 約 12.9 年 | 13 年 |
| 4 件 | 11 人 | 約 9.5 年 | 8 年 |
| 3 件 | 5 人 | 約 8.6 年 | 7 年 |
| 2 件以下 | 1 人 | 25 年 | 25 年 |

一人あたり平均で 4.3 件の欠陥を防止している。また、防止した欠陥件数が 3 件以上の被験者については、業務年数が 10 年前後の、比較的経験年数が浅い被験者であっても欠陥を防止している。次に、各々の欠陥の防止状況を表 5 に示す。

表 5 欠陥の防止項目の内訳

| 欠陥内容 | 欠陥を防止した被験者数 | 平均業務年数 | 業務年数の中央値 |
|-------------------|-------------|----------|----------|
| 共通設定のコンポーネント間の齟齬① | 23 人 | 約 11.5 年 | 12 年 |
| 共通設定のコンポーネント間の齟齬② | 26 人 | 約 11.7 年 | 11.5 年 |
| 他コンポーネントへの影響考慮不足 | 29 人 | 約 11.4 年 | 11 年 |
| 特定状態での要件漏れ① | 30 人 | 約 11.9 年 | 11.5 年 |
| 特定状態での要件漏れ② | 33 人 | 約 11.5 年 | 11 年 |

それぞれの欠陥を防止した被験者の業務年数に大きな差は見られないが、防止した被験者数には差が出た。共通設定については、コンポーネントの上流にあたるソフトウェア階層のアーキテクチャの特徴を知っているか否かの差が大きいことに起因すると考えられる。

他コンポーネントへの影響考慮不足、および特定状態での要件漏れは共通設定の欠陥と比較すると、より多くの被験者が欠陥を防止している。これら 3 つの欠陥をすべて防止した被験者は全体の 92.9%である。

以上より、要件定義書を作成した設計者は、少なくとも 2 件の欠陥を業務年数に依存せず摘出できていること、特に要件漏れや他コンポーネントへの影響考慮不足については 90%以上の被験者が欠陥の防止をしていることから、本提案手法は欠陥の防止に効果があると評する。他方、本提案手法によるプロジェクトの工数削減効果及び手戻り工数の削減効果については未検証であるため、今後の課題としたい。

5. おわりに

本研究では組み込みソフトウェア業における要求仕様化の属人性の排除、およびそれに伴う欠陥の作り込み防止を目的としたドキュメントとガイドラインの提案を行い、その有効性を検証した。結果、従来は業務経験が 20 年以上の上位設計者のみ防止していた欠陥を、業務経験が 10 年程度の設計者でも防止可能となることが確認できた。よって、提案内容は欠陥防止に有効であると考えられる。

一方、今回の実験では欠陥の作り込み防止による工数の削減効果については検証ができていない。本提案手法を実際のプロジェクトに適用した際に、手戻り工数がどの程度削減できるか、他のプロジェクト実績と比較することで有効性を検証したい。加えて、本稿では提案手法を複数プロジェクトに対して適用しての有効性検証はできていない。提案内容の一般的な有効性についても、引き続き検証を行っていくとともに、類似研究の調査も継続していく。

今後は、実験において欠陥を防止した件数が比較的少なかった共通設定のコンポーネント間の齟齬に対して、これを防止できる様な手法を模索していく。近年はシステムの多様化に伴い、従来には無いコンポーネントを勘案した設計が重要となっている。受託ソフトウェア業における要求の仕様化方法は、組み込みソフトウェア業と比較してより多くの周辺システムを加味した要求の仕様化を実施していることから、これを参考としてガイドラインのブラッシュアップを図る。

謝辞

本研究に対して有益なご助言をいただいた ソニー株式会社 栗田太郎氏、国立情報学研究所 石川冬樹氏、熊本高等専門学校 荒木啓二郎氏、南山大学 張漢明氏、第 35 年度ソフトウェア品質管理研究会研究コース 5 のメンバに感謝の意を表す。

本研究の実験にご協力いただいた 企業の皆様に感謝の意を表す。

参考文献

- [1] 山本修一郎, 「要求」の 40 年 1979-2019 ～過去・現在・未来～ ASDoQ 2019 基調講演
- [2] 山本修一郎, デジタルトランスフォーメーションに向けた要求管理知識, 第 20 回知識流通ネットワーク研究会, SIG-KSN-020-01
- [3] 福嶋慎一, 組み込み系ソフトウェア開発の課題分析と提言 2008 年度 JEITA ソフトウェア事業委員会セミナー
- [4] 組み込み系ソフトウェア向け設計ガイド ESDR [事例編], 独立法人情報処理推進機構 (IPA)
- [5] 貝瀬康利, 開発要件の完成度を高めるアプローチの検討, JASA 技術本部セミナー, JG-5
- [6] 要求工学知識体系 (REBOK) とユーザ指向要求工学の調査研究, 社団法人 情報サービス産業協会, 20-J007
- [7] 要求開発・管理ベストプラクティスとその体系化の調査研究, 情報サービス産業協会, 19-J004

- [8] 要求開発ベストプラクティスが示す成功パターンの調査研究, 情報サービス産業協会, 18-J008
- [9] 山田大介, ソフトウェア疲労をアーキテクチャ設計で ソフトウェア品質管理研究会 2019 年度第 8 回特別講義
- [10] IEEE, Requirement Boilerplates: Transition from Manually - enforced to Automatically-Verifiable Natural Language Patterns