

調査報告書

ソフトウェア開発現場における
工数見積もり誤りの防止に関する研究

2016年 3月

中央大学 理工学部 経営システム工学科

川平 陽明

目次

謝辞

第1章 研究目的・・・1

第2章 仮説の設定

2.1 工数見積もりのプロセスと方法・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・3

2.2 工数見積もり誤りを引き起こす原因に関する仮説・・・・・・・・・・・・・・・・・・・・4

第3章 ソフトウェア開発現場における工数見積もりに関する調査

3.1 調査の計画と実施・・・7

3.2 調査した項目と内容・・・7

第4章 「工数見積もり方法」と「工数見積もり誤り」に関する調査結果と解析

4.1 回答企業の概要・・・9

4.2 工数見積もり誤りの割合・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・12

4.3 工数見積もり方法・・・13

4.4 工数見積もり誤りとその原因・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・22

4.5 工数見積もり方法と誤りの関係・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・25

4.6 工数見積もり誤りの原因を改善する難しさとその克服策・・・・・・・・・・・・・・28

第5章 考察・・・37

第6章 結論と今後の課題・・・39

参考文献・・・40

付録 ソフトウェア開発における工数見積もり誤りの防止に関する調査

謝辞

本研究を進めるに当たり、「ソフトウェア開発における工数見積もり誤りの防止に関する調査」の実施に置きまして、貴重な時間を割いてご協力くださいました企業の方々に対して厚く御礼申し上げます。

第1章 研究目的

現代社会において、情報通信技術の発達により人々の生活に IT (Information Technology) が欠かせないものとなっている。それに伴い、あらゆるソフトウェアが必要となり、多くの IT 企業が顧客の要求に応じて必要なソフトウェアの開発を行っている。しかし、各々のソフトウェアの開発に必要な工数 (人・月) は様々な要因の影響を受けるため、その見積もりは容易ではない。結果として、ソフトウェア開発現場では、勤務時間外労働の増加による超過労働やトラブルの発生による不具合、納期の間近に完成してしまい品質チェックの不十分による品質の低下、開発コストに誤りが生じてしまうことなどが重要な問題となっている。

開発工数の見積もりは、ソフトウェア開発における総合的な費用管理、設定された工数の下での作業進捗管理、各工程に配分された工数投入による品質向上度の評価を行うために必要である。図 1 は、開発工数の見積もりにもとづく、ソフトウェア開発管理の全体像をまとめたものである。この中で、開発管理計画のベースとなるソフトウェア開発工数の見積もりについては、従来からも多くの研究・提案が行われてきている。しかし、開発という業務の性質上、考慮すべき要因が多く、またそれらが人の行動・能力に関連するため、十分な成果が得られていないのが現状である。

本研究では、工数見積もりにおける誤り (見積もりと実績との差が大きい事象) を減らすために IT 企業が行っている取り組みについて、どのような要因をどのように考慮しているのか、どのような要因の考慮が難しいのかという視点から調査・分析を行う。これにより、工数見積もり誤りを引き起こしている原因がどこにあるのか、誤りを減らす対策として今後どのような面に着目して検討を進めるのがよいのかを明らかにする。

第 2 章 仮説の設定

2.1 工数見積もりのプロセスと方法

ソフトウェアの開発管理工数を見積もる場合には、図 1 に示した通り、作業の分解・階層化、開発規模の見積もり、開発工数の見積もりというステップに従って進められることが多い。

このうち、作業の分解・階層化については、WBS (Work Breakdown Structure) などの方法が用いられる。

また、開発規模の見積もりでは、次のような代表的な方法が用いられている。

- (1) 経験に基づく方法：過去に行われたプロジェクトの経験をもとに、新規に開発する対象の規模を予測する方法である。類似したソフトウェアを開発する場合には高い精度で規模を見積もることができる。しかし、経験とは異なる開発を行う場合や、使用する言語が異なる場合には見積もりの精度は低くなる。
- (2) LOC 法：従来からよく用いられており、生成するソースコードの行数をもとに開発規模を見積もる方法である。LOC 法では、全体の生成できるコード数が分かれば、プロジェクトに参加できるプログラマの数から開発期間を見積もることができる。古い言語は 1 行が比較的単純であったが、現在の多くのプログラムは複数行でプログラムが組み立てられているものが多いので適切に測ることが難しい。
- (3) ファンクションポイント法(FP 法)：ソフトウェアの機能に関する基準を使って規模を見積もる方法である。この方法は LOC 法とは異なり、実装に用いるプログラミング言語による影響を受けない。項目に分けられている機能分類と、それぞれに対する複雑度をもとに数値を割り当てたものを積算することによって、全体のファンクションポイントを計算することで、開発規模を見積もる。

さらに、開発工数を見積もる代表的な方法としては以下のようなものがある。

- (1) 経験に基づく方法
- (2) COCOMO：ソースコードの行数をもとに開発規模を見積もり、それをもとに、開発規模と開発工数の関係を表す予測モデルに基づいて開発工数を求める方法である。
- (3) 共通モデル：開発規模の見積もり値に適切な生産係数を掛け合わせるにより工数を見積もる方法である。すなわち、開発工数 E と開発規模 L の関係を、 $E = pL^c$ で表し、これをもとに工数を見積もる方法である。ここで、 c は定数であり、過去の実績データから推定される。また、生産性調整係数 p は、ソフトウェア開発に関わる様々な要因を詳細化し、これらの要因をそれぞれ評価・指数化した上で、その積を用いて算出される。

2.2 工数見積もり誤りを引き起こす原因に関する仮説

工数見積もり誤りを引き起こしている原因を、文献[1]を参考に整理した。見積もり誤りが起こる原因は、様々な側面から考えられるが、顧客の曖昧な要求に基づいて一品一様の見積もりを行うソフトウェア開発においては、「考慮すべき要因を十分考慮していないこと」が主な原因になっていると考えられる。このため、どのような要因の考慮が不足していたのかという視点から仮説を立てた。立てた仮説を以下に示す。

A. 人的要因に対する考慮不足

- (1) ソフトウェアの開発の生産性は携わる個人の能力に大きく依存する。見積もりを行う際、あるいはプロジェクトに参加する人を決める際に、個人個人の能力が把握できておらず、これらを十分考慮できていない。
- (2) 開発に関わるリーダーのプロジェクトについての経験や、持っている知識の差によって違いが生じる。プロジェクトに関わるリーダーが経験者であるか初心者であるかの力の差を考慮することができていない。
- (3) 開発に関わるメンバーのプロジェクトについての経験や、持っている知識の差によって違いが生じる。プロジェクトに関わるメンバーが経験者であるか初心者であるかの力の差を考慮することができていない。
- (4) プロジェクトを一緒に行う企業間でコミュニケーションが上手く取れるかどうか、上手く取れないことによるタイムラグを考慮できていない。
- (5) リーダーとメンバー間のコミュニケーションが上手く取れるかどうか、上手く取れないことによるタイムラグを考慮できていない。
- (6) ソフトウェア開発においては開発期間が限られている場合が多くある。しかし、企業内における他プロジェクトの状況を把握できておらず、当該プロジェクトに関わる人数を集めることが困難な状況を考慮できていない。

B. 問題要因に対する考慮不足

- (7) 見積もりを行う際に、類似のプロジェクトが無く、特徴が異なるために予見することが困難である状況を考慮できていない。開発プロジェクトは似たようなものはあるが、一回限りのプロジェクトになりがちである。そのため、プロジェクトごとに特徴が異なる。さらに、情報通信技術の発達が早いことにより、経験豊富なプロジェクト管理者でも、以前の知識や経験をそのまま転用することはできず、プロジェクトの進行に応じて生じる問題を予見することが難しい。
- (8) ソフトウェア開発は、土木事業などと違い、開発の対象が目に見えない。そのため曖昧な部分が残されたまま、開発を行うことが少なくない。このような曖昧になっている部分が開発に与える影響について考慮できていない。
- (9) 顧客が要求を具体化できていないために、詳細に見積もることが困難な場合がある。

このような顧客が要求を具体化できていない部分を考慮できていない。

- (10) システムを開発する際に、前段階の工程が完成していないと次に行う工程に移ることが出来ない部分がある。このような部分の開発がスケジュール通り進まないことを考慮できていない。

C. プロセス要因に対する考慮不足

- (11) 機械の設計などについては、これまでに何度も実践され、その有効性が確かめられている標準的な開発プロセスが確立できている。しかし、ソフトウェア開発は実施する組織によってプロセスが大きく異なることにより、標準となる開発プロセスが十分に確立できていない場合が多い。標準的な開発プロセスが確立できていない、あるいは確立できていても当該のプロジェクトに活用できないことを考慮できていない。
- (12) 開発で使用するプログラム言語には、簡単に使用できるものとそうでないものがある。使用するプログラム言語の難しさを考慮できていない。

D. プロダクト要因に対する考慮不足

- (13) 顧客と開発者側とで必要なコストに対する認識が異なり、開発側が必要な工数を適切に見積もれない場合がある。開発にかかるコストに対する顧客と開発者側との認識の相違を考慮できていない。
- (14) 見積もりを行う際に、過去に使用したシステムのモデルやプログラムを改良して開発するように計画していたものが、開発が進むにつれ使うことが出来ないことが判明する場合がある。このような問題が生じることを考慮できていない。
- (15) 開発が始まった後に顧客から新たな要求が追加されることがある。このような追加が起こることを考慮できていない。

E. 資源要因に対する考慮不足

- (16) 開発予算の関係で開発にかけられる人数が限られているために、工数を少なめに見積もらざるを得ないことがある。開発予算の関係によって開発にかけられる人数が限られる状況を考慮できていない。
- (17) 開発に使用できるハードウェアに制限があり、自由に使えない場合がある。開発に使用できるハードウェアの制限を考慮できていない。

上記の仮説(1)～(17)を模式的に表すと図 2 の通りとなる。

表 1 ソフトウェアの生産性変動要因[1]

変動要因の種類	内容
人的要因	開発チーム、開発組織の大きさと熟練度
問題要因	対象とする問題の型と重要度、要求仕様の構成、問題解の制約
プロセス要因	要求仕様言語、設計言語、プログラミング言語、および開発技法
プロダクト要因	対象システムの信頼性、規模、効率、制御構造
資源要因	対象とするハードウェア、開発時間、開発予算

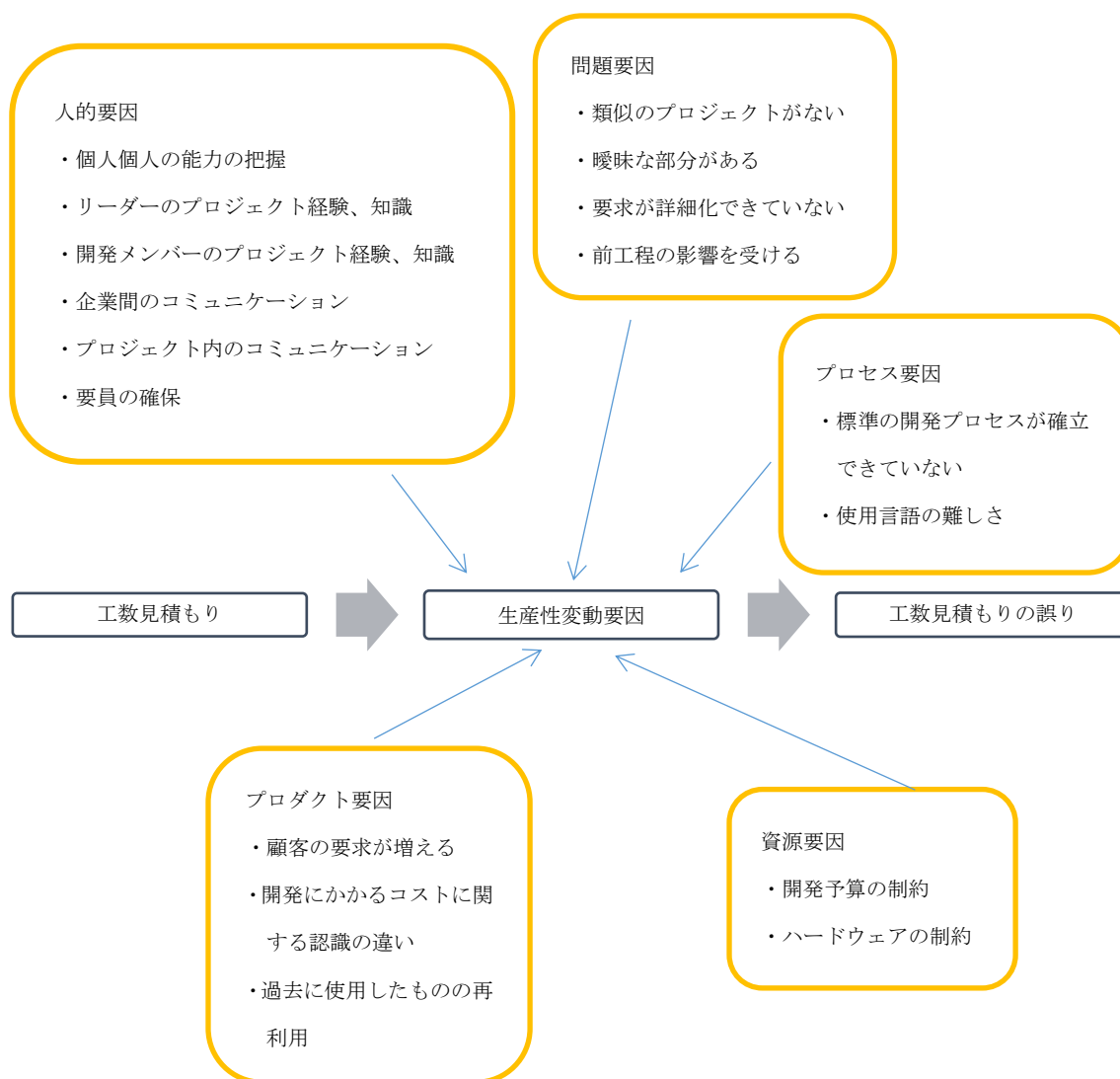


図 2 工数見積もり誤りを引き起こす原因に関する仮説

第 3 章 ソフトウェア開発現場における工数見積もりに関する調査

3.1 調査の計画と実施

3 章で述べた仮説を検証するために、ソフトウェア開発を行っている企業に対して郵送調査を行った。対象とした企業は、ISO9001 認証取得組織で産業分類が「情報技術」であるものの中から無作為に選んだ 260 社である。

回答に当たっては、ソフトウェア開発を行っており、工数見積もり誤りの防止・低減が重要な課題となっている職場を選んで答えて頂いた。調査期間は、2016 年 1 月～2 月である。結果として、28 社より回等を得た（回収率 10.8%）。

3.2 調査した項目と内容

調査した項目は以下の 4 項目である。

I. 職場の概要

(1) 開発しているソフトウェアの種類（組込ソフトウェア、システムソフトウェア、業務用アプリケーションソフトウェア、工業用アプリケーションソフトウェア、パッケージソフトウェア、コンテンツソフトウェアなど）、(2) 行っている開発業務の内容（要件定義、外部設計、内部設計、コーディング、レビュー、テスト、導入支援、保守など）、(3) 1つのプロジェクトを担当する人の人数、(4) 1つのプロジェクトの開始から終了までの期間を調査した。

II. 工数見積もりのプロセスと方法

(1) 工数見積もりをどのようなプロセスをどのようなステップ（作業の分解、開発規模の見積もり、開発工数見積もりなど）で行っているか、それぞれのステップをどのような方法で行っているかを聞いた。また、(2) 2.2 工数見積もり誤りを引き起こす要因の仮説をもとに、「工数見積もり誤りが発生する要因をどのように考慮しているか」を、選択式で回答してもらった。なお、列挙したもの以外で考慮している要因があれば、自由書式で回答してもらった。

III. 工数見積もり誤りとその原因

(1) 「工数見積もり誤り（実際の工数が見積もった値の±20%以上を超える事象）の発生割合」を選択式で回答してもらった。また、工数見積もり誤りが起きる原因について、(2) 「作業の分解・階層化」～「開発工数見積もり」の各プロセスが関係しているか、(3)

「個人個人の能力の差を考慮していないこと」～「開発予算の関係で開発にかけられる人数が限られる状況を考慮していないこと」のそれぞれが関係しているか、を5段階で回答してもらった（1. 原因になっているとは思わない～5. 原因になっていると思う）。また、他の原因としてどのようなものがあるかを自由書式で回答してもらった。

IV. 工数見積もり誤りを減らす取り組み

(1) 見積もりプロセスと(2) 見積もりにおいて考慮すべき要因に分けた上で、Ⅲで「どちらかといえば原因となっていると思う」または「原因になっていると思う」と回答したものについて、当該の原因を改善する難しさと克服のために実践している/しようとしていることをそれぞれ自由書式で回答してもらった。

第 4 章 「工数見積もり方法」と「工数見積もり誤り」に関する調査結果と解析

4.1 回答企業の概要

(1) 開発しているソフトウェアの種類

回答企業が開発しているソフトウェアの種類についてのデータを集計した。結果を表 2 に示す。

表 2 より以下のことが分かった。

- ①業務用アプリケーションソフトを開発している企業が大半である。
- ②ファームウェアなどの組み込みソフトウェア、OS などのシステムソフトウェアを開発している企業が一部ある。

表 2 開発しているソフトウェアの種類

開発しているソフトウェアの種類	回答企業
1.ファームウェアなどの組込ソフトウェア	2
2.OS などのシステムソフトウェア	2
3.業務用アプリケーションソフトウェア	24
4.工業用アプリケーションソフトウェア	1
5.パッケージソフトウェア	1
6.コンテンツソフトウェア	0
7.その他	1

注 1) 回答企業数 28 社。複数回答あり。

注 2) その他はアプライアンス機器のソフトウェア開発である。

(2) 行っている開発業務

回答企業が行っている開発業務の内容についてのデータを集計した。結果を表 3 に示す。

表 3 より以下のことが分かった。

- ①要件定義から保守まで全般的に行っている企業が多い。
- ②運用を行っている企業は少数である。

表 3 行っている開発業務

開発業務	回答企業
要件定義	23
外部設計	23
内部設計	23
コーディング	22
レビュー	21
テスト	24
導入支援・運用支援	18
運用	2
保守	20
その他	5

注 1) 回答企業数 28 社。複数回答あり。

注 2) その他は、カスタマイズ、開発、提案、設計、基本計画である。

(3) 1つのプロジェクトを担当する人の人数と開始から終了までの期間の関係

「1つの開発プロジェクトを担当する人数の数」と「1つのプロジェクトの開始から完了までの期間」についてのデータを集計した。ただし、回答頂いた人数や期間に幅がある場合には、該当するすべてのセルに1/該当するセルの数を入力した。結果を表4に示す。

表4から以下のことが分かった。

- ①人数で見ると、1人以上～20以下が多い。21人以上は少ない。
- ②期間としては、1ヶ月以上～15ヶ月未満が多い。15ヶ月以上は少ない。
- ③人数と期間は弱い関連があり、大人数で短期間の開発を行っている企業、少人数で長期間にわたる開発を行っている企業は少ない。

表4 1つのプロジェクトに掛かる期間と人数の関係

人数 \ 期間	1ヶ月以上～3ヶ月未満	3ヶ月以上～6ヶ月未満	6ヶ月以上～9ヶ月未満	9ヶ月以上～12ヶ月未満	12ヶ月以上～15ヶ月未満	15ヶ月以上～18ヶ月未満	18ヶ月以上～21ヶ月未満	21ヶ月以上～24ヶ月未満	24ヶ月以上	合計
1人以上～10人以下	2.71	4.10	2.69	1.44	3.44	0.45	0.45	0.27	0.27	15.81
11人以上～20人以下	0.55	0.70	0.78	0.53	0.53	0.19	0.19	0.16	0.16	3.81
21人以上～30人以下	0.28	0.31	0.32	0.24	0.24	0.12	0.12	0.09	0.09	1.81
31人以上～40人以下	0.22	0.24	0.25	0.17	0.17	0.12	0.12	0.09	0.09	1.48
41人以上～50人以下	0.08	0.11	0.12	0.12	0.12	0.12	0.12	0.09	0.09	0.98
51人以上～60人以下	0.06	0.09	0.10	0.10	0.10	0.10	0.10	0.07	0.07	0.78
61人以上～70人以下	0.06	0.09	0.10	0.10	0.10	0.10	0.10	0.07	0.07	0.78
71人以上～80人以下	0.06	0.09	0.10	0.10	0.10	0.10	0.10	0.07	0.07	0.78
81人以上～90人以下	0.05	0.07	0.08	0.08	0.08	0.08	0.08	0.06	0.06	0.65
91人以上～100人以下	0.05	0.07	0.08	0.08	0.08	0.08	0.08	0.06	0.06	0.65
101人以上	0.02	0.04	0.06	0.06	0.06	0.06	0.06	0.06	0.06	0.45
合計	4.15	5.90	4.69	3.02	5.02	1.52	1.52	1.09	1.09	28

注1) 回答企業数28社。複数回答あり。

注2) 黄色の部分には0.5以上の部分である。

4.2 工数見積もり誤りの割合

回答企業における工数見積もり誤りの発生の現状を把握するために、調査項目Ⅲ-(1)の「工数見積もり誤りの発生割合」で得られたデータをヒストグラムによりグラフ化した。結果を図3に示す。

図3より以下のことが分かった。

- ①30%未満の企業が大半で、30%以上の企業はほとんど無い。
- ②工数見積もり誤りの割合が10%未満、10%以上～20%未満、20%以上～30%未満の企業がほぼ均等に含まれている。

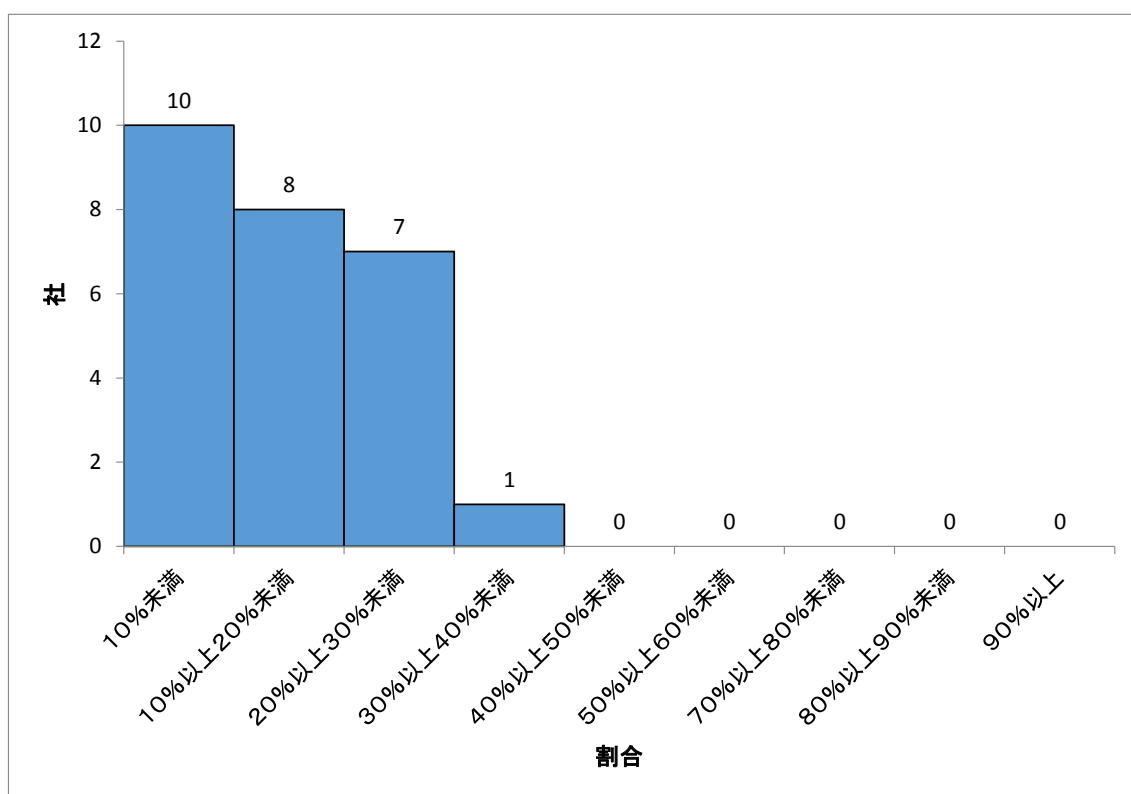


図3 工数見積もり誤りが発生するプロジェクトの割合
注) 回答企業数 26 社、未回答企業数 2 社。

4.3 工数見積もり方法

(1) 工数見積もりをどのようなプロセス/方法で行っているか

回答企業における工数見積もりのプロセスの現状を明らかにするために、調査項目Ⅱ-(1)の「工数見積もりをどのようなプロセス/方法で行っていますか」で得られたデータを度数分布表により整理した。結果を表5に示す。

表5より以下のことが分かった。

- ①作業の分解(WBS等)→開発規模見積もり→開発工数見積もりの順で行っている企業が大半である。
- ②開発規模見積もりを行っていない企業が少数ある。

表5 見積もりのプロセス

	見積もりのプロセス	回答企業
1	作業の分解(WBS等)→開発規模見積もり→開発工数見積もりの順に行っている	20
2	作業の分解(WBS等)を行った後、開発規模見積もりを行わないで、開発工数見積もりを行っている	4
3	作業の分解(WBS等)を行わず、開発規模見積もりを行った後、開発工数見積もりを行っている	1
4	作業の分解(WBS等)や開発規模見積もりを行わないで、開発工数見積もりのみを行っている	2
5	その他	1

注) 回答企業数 28 社。

(2) 工数見積もり誤りが発生する要因をどのように考慮しているか

回答企業において工数見積もり誤りに影響する要因が考慮されている現状を明らかにするために、調査項目Ⅱ-(2)「工数見積もり誤りが発生する要因をどのように考慮していますか」で得られたデータを度数分布表により整理した。結果を表6～表21に示す。

表6～表21より以下のことが分かった。

- ①個人個人能力の差の考慮の仕方は、「全く考慮していない」から「個人個人能力を定期的(年1回等)に評価し、その結果を考慮している」まで均等に分かれている。また、「経験年数や職位などを考慮している」と回答した企業が他に比べると少し多い。
- ②リーダーのプロジェクトの経験や、持っている知識の差については、「リーダーとして類似のプロジェクトに携わった経験があるかを考慮している」が多い。また、「リーダーとして携わったプロジェクトの経験に加え、リーダーとしてのマネジメント能力を評価し、考慮している」と回答した企業も多い。
- ③メンバーとしての類似のプロジェクトの経験や、持っている知識の差については、「メンバーとして類似のプロジェクトに携わった経験があるかを考慮している」が多い。

他方、「全く考慮していない」、「資格に加え、メンバーとしてのプロジェクトに携わった経験年数を考慮している」と回答した企業も多い。

- ④プロジェクトを一緒に行う企業間でのコミュニケーションについては、「過去にプロジェクトを一緒に行った結果(コミュニケーションの問題が合ったかどうか)を考慮している」が大半である。他方、「企業ごとの特性(規模、繁忙さなど)やマネジメント能力を評価し、考慮している」、「相手企業との契約形態やコミュニケーション方法の違い(対面による相談が容易かどうか)を考慮している」と回答した企業も多い。他方、「全く考慮していない」、「過去に一緒にプロジェクトを行ったことがあるかのみを考慮している」と回答した企業は無い。
- ⑤リーダーとメンバーの間でのコミュニケーションによる遅れの仕方は、「全く考慮していない」から「リーダーとメンバーのコミュニケーション方法(対面による相談が可能かどうかなど)によって考慮している」まで均等に分かれている。また、「プロジェクトの規模のみを考慮している」と回答した企業が他に比べると少し多い。
- ⑥他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況については、「プロジェクトの規模・期間(必要となる人数の多さなど)のみを考慮している」から「自プロジェクトの状況に加え、同時に行われる他のプロジェクトの状況を考慮している」まで均等に分かれている。他方、「全く考慮していない」と回答した企業は少ない。
- ⑦類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況については、「特徴が大きく異なるプロジェクトのみ考慮している」、「特徴の相違の大きさを段階評価し、考慮している」、「過去のデータをもとに、特徴の相違が工数に与える影響を定性的に評価し、考慮している」が大半である。他方、「全く考慮していない」と回答した企業は少ない。さらに、「過去のデータをもとに、特徴の相違が工数に与える影響を定量的に評価し、考慮している」と回答した企業も少ない。
- ⑧標準となる開発プロセスの適用性については、「標準の開発プロセスが適用できているかのみを考慮している」、「開発プロセスの中で、標準が適用できない部分の割合を明確にし、考慮している」が大半である。また、「開発プロセスの中で、標準と最も異なる部分を明確にし、評価・考慮している」と回答した企業も多い。他方、「全く考慮していない(標準を定めていない場合を含む)」、「開発プロセスのステップ毎に標準が適用できる度合いを評価付けし、考慮している」と回答した企業はほとんど無い。
- ⑨ソフトウェア開発の曖昧になっている部分については、「開発するソフトウェアの中で特に曖昧な部分を明確にし、その曖昧さを評価・考慮している」が多い。また、「全体的な曖昧さを段階評価し、考慮している」と回答した企業も多い。他方、「全く考慮していない」、「ソフトウェアを構成する各部分について曖昧さを評点付けし、考慮している」と回答した企業は無い。
- ⑩顧客が要求を具体化できていない部分については、「顧客が最も具体化できていない

要求を明確にし、評価・考慮している」が多い。また、「顧客が要求を具体化できていない状況を前提的に段階評価し、考慮している」と回答して企業も多い。他方、「全く考慮していない」、「顧客の要求をいくつかに分けた上で、各々の具体化できていない度合いを評点付けし、考慮している」と回答した企業はほとんど無い。

- ⑪開発が始まった後に顧客から新たな要求が追加される可能性については、「ソフトウェアの種類によって顧客から新たな要求が追加される可能性を評価し、考慮している」、「過去のデータに基づいて、新たな要求が追加される可能性の高い顧客かどうかを評価し、考慮している」が大半である。他方、「全く考慮していない」、「新規のソフトウェアかどうかのみを考慮している」と回答した企業はほとんど無い。
- ⑫再利用しようとしていたソフトウェアやモデルが活用できない可能性については、「特別の事情がある場合のみを考慮している」が多い。また、「再利用しようとしているソフトウェアやモデルのうち、活用できない可能性が最も高いものを明確にし、評価・考慮している」と回答した企業も多い。他方、「全く考慮していない」、「再利用しようとしているソフトウェアやモデルごとに活用できない可能性を評点付けし、考慮している」と回答した企業はほとんど無い。
- ⑬開発で使用するプログラム言語の難しさについては、「特に難しいプログラム言語のみを考慮している」が多い。また、「プログラム言語の難しさを段階評価し、考慮している」、「プログラム言語と開発環境などとの相性を考慮している」と回答した企業も多い。他方、「全く考慮していない」と回答した企業はほとんど無い。
- ⑭開発で使用するハードウェアの制限については、「開発で使用するハードウェアの制限が特に厳しい場合のみを考慮している」が多い。他方、「開発で使用するハードウェアの制限が特に厳しい場合のみを考慮している」以外の回答をした企業は少ない。
- ⑮開発に掛かるコストに対する顧客と開発者側との認識の相違については、「特に大きな認識の相違がある場合のみ考慮している」、「認識の相違の大きさを段階評価し、考慮している」、「ソフトウェアまたは開発プロセスを要素に分け、要素毎に、開発にかかるコストに対する認識の相違を評価し、考慮している」が大半である。他方、「全く考慮していない」、「ソフトウェアまたは開発プロセスを要素に分け、その中で、開発にかかるコストに対する認識の相違が最も大きい部分を明確にし、評価・考慮している」と回答した企業はほとんど無い。
- ⑯開発予算の関係で開発にかけられる人数が限られている状況については、「特に開発予算が限られている場合のみを考慮している」、「開発予算が限られている状況を段階評価し、考慮している」が大半である。他方、「全く考慮していない」、「開発予算の中で最も厳しい項目を明確にし、評価・考慮している」、「開発予算の各項目の厳しさを評点付けし、考慮している」と回答した企業はほとんど無い。

表6 個人個人の能力の差の考慮の仕方

	個人個人の能力の差の考慮の仕方	回答企業
1	全く考慮していない	3
2	一部の特別な人のみを考慮している	5
3	経験年数や職位などを考慮している	9
4	個人個人の能力を評価し、考慮しているが、能力の定期的な評価・更新は行っていない	4
5	個人個人の能力を定期的(年1回等)に評価し、その結果を考慮している	5
6	その他	2

注) 回答企業数 28 社。

表7 リーダーのプロジェクトの経験や持っている知識の差の考慮の仕方

	リーダーのプロジェクトの経験や持っている知識の差の考慮の仕方	回答企業
1	全く考慮していない	3
2	リーダーとしてプロジェクトに携わった経験年数を考慮している	1
3	リーダーとして類似のプロジェクトに携わった経験があるかを考慮している	13
4	リーダーとして携わったプロジェクトの成果(成功や失敗等)を考慮している	1
5	リーダーとして携わったプロジェクトの経験に加え、リーダーとしてのマネジメント能力を評価し、考慮している	9
6	その他	0

注) 回答企業数 27 社、未回答企業数 1 社。

表8 メンバーのプロジェクトの経験や持っている知識の差の考慮の仕方

	メンバーのプロジェクトの経験や持っている知識の差の考慮の仕方	回答企業
1	全く考慮していない	4
2	資格に加え、メンバーとしてプロジェクトに携わった経験年数を考慮している	5
3	メンバーとして類似のプロジェクトに携わった経験があるかを考慮している	16
4	メンバーとしてプロジェクトで残した成果を考慮している	3
5	取得している資格やいままでに受けた教育を考慮している	0
6	その他	0

注) 回答企業数 28 社。

表9 プロジェクトを一緒に行う企業間でのコミュニケーションによる遅れの考慮の仕方

	プロジェクトを一緒に行う企業の間でのコミュニケーションによる遅れの考慮の仕方	回答企業
1	全く考慮していない	0
2	過去に一緒にプロジェクトを行ったことがあるのみを考慮している	0
3	過去にプロジェクトを一緒に行った時の結果(コミュニケーションの問題があったかどうか)を考慮している	10
4	企業ごとの特性(規模、繁忙さなど)やマネジメント能力を評価し、考慮している	6
5	相手企業との契約形態やコミュニケーション方法の違い(対面による相談が容易かどうか)を考慮している	7
6	その他	5

注) 回答企業数 28 社。

表10 リーダーとメンバーの間でのコミュニケーションによる遅れの考慮の仕方

	リーダーとメンバーの間でのコミュニケーションによる遅れの考慮の仕方	回答企業
1	全く考慮していない	4
2	プロジェクトの規模のみを考慮している	8
3	リーダーとメンバーが過去に一緒にプロジェクトを行ったことがあるかを考慮している	6
4	過去にプロジェクトを一緒に行った時の結果(コミュニケーションの問題があったかどうか)を考慮している	4
5	リーダーとメンバーのコミュニケーション方法(対面による相談が可能かどうかなど)によって考慮している	3
6	その他	3

注) 回答企業数 28 社。

表 11 他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが
困難な状況の考慮の仕方

	他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが 困難な状況の考慮の仕方	回答企業
1	全く考慮していない	2
2	プロジェクトの規模・期間(必要となる人数の多さなど)のみを考慮して いる	4
3	プロジェクトの種類(特別な能力を持つ人の必要性など)によって人を 集める困難さを評価し、考慮している	7
4	プロジェクトの途中段階でメンバーを増減する可能性を評価し、考慮 している	6
5	自プロジェクトの状況に加えて、同時に行われる他のプロジェクトの状 況を考慮している	6
6	その他	3

注) 回答企業数 28 社。

表 12 類似のプロジェクトが無く、特徴が異なるために予見することが
困難な状況の考慮の仕方

	類似のプロジェクトが無く、特徴が異なるために予見することが困難な 状況の考慮の仕方	回答企業
1	全く考慮していない	1
2	特徴が大きく異なるプロジェクトのみ考慮している	9
3	特徴の相違の大きさを段階評価し、考慮している	6
4	過去のデータをもとに、特徴の相違が工数に与える影響を定性的に 評価し、考慮している	7
5	過去のデータをもとに、特徴の相違が工数に与える影響を定量的に 評価し、考慮している	2
6	その他	3

注) 回答企業数 28 社。

表 13 標準となる開発プロセスの適用性の考慮の仕方

	標準となる開発プロセスの適用性の考慮の仕方	回答企業
1	全く考慮していない(標準を定めていない場合を含む)	1
2	標準の開発プロセスが適用できるかのみを考慮している	10
3	開発プロセスの中で、標準が適用できない部分の割合を明確にし、考 慮している	11
4	開発プロセスの中で、標準と最も異なる部分を明確にし、評価・考慮し ている	6
5	開発プロセスのステップ毎に標準が適用できる度合いを評点付けし、 考慮している	0
6	その他	0

注) 回答企業数 28 社。

表 14 ソフトウェア開発の曖昧になっている部分の考慮の仕方

	ソフトウェア開発の曖昧になっている部分の考慮の仕方	回答企業
1	全く考慮していない	0
2	特別に曖昧な場合のみ考慮している	5
3	全体的な曖昧さを段階評価し、考慮している	8
4	開発するソフトウェアの中で特に曖昧な部分を明確にし、その曖昧さを評価・考慮している	15
5	ソフトウェアを構成する各部分について曖昧さを評点付けし、考慮している	0
6	その他	0

注) 回答企業数 28 社。

表 15 顧客が要求を具体化できていない部分の考慮の仕方

	顧客が要求を具体化できていない部分の考慮の仕方	回答企業
1	全く考慮していない	0
2	特に具体化できていない場合のみを考慮している	4
3	顧客が要求を具体化できていない状況を全体的に段階評価し、考慮している	7
4	顧客が最も具体化できていない要求を明確にし、評価・考慮している	14
5	顧客の要求をいくつかに分けた上で、各々の具体化できていない度合いを評点付けし、考慮している	1
6	その他	2

注) 回答企業数 28 社。

表 16 開発が始まった後に顧客から新たな要求が追加される可能性の考慮の仕方

	開発が始まった後に顧客から新たな要求が追加される可能性の考慮の仕方	回答企業
1	全く考慮していない	1
2	新規のソフトウェアかどうかのみを考慮している	1
3	ソフトウェアの種類によって顧客から新たな要求が追加される可能性を評価し、考慮している	9
4	過去のデータに基づいて、新たな要求が追加される可能性の高い顧客かどうかを評価し、考慮している	7
5	要求を顧客から得たプロセスによって新たな要求が追加される可能性を評価し、考慮している	2
6	その他	8

注) 回答企業数 28 社。

表 17 再利用しようとしていたソフトウェアやモデルが活用できない可能性の
考慮の仕方

	再利用しようとしていたソフトウェアやモデルが活用できない可能性の 考慮の仕方	回答企業
1	全く考慮していない	0
2	特別の事情がある場合のみを考慮している	13
3	再利用しようとしているソフトウェアやモデルが活用できない可能性を 段階評価し、考慮している	4
4	再利用しようとしているソフトウェアやモデルのうち、活用できない可 能性が最も高いものを明確にし、評価・考慮している	7
5	再利用しようとしているソフトウェアやモデルごとに活用できない可能 性を評点付けし、考慮している	2
6	その他	2

注) 回答企業数 28 社。

表 18 開発で使用するプログラム言語の難しさの考慮の仕方

	開発で使用するプログラム言語の難しさの考慮の仕方	回答企業
1	全く考慮していない	1
2	特に難しいプログラム言語のみ考慮している	10
3	プログラム言語の難しさを段階評価し、考慮している	6
4	過去のデータに基づいて、プログラム言語の難しさを評点付けし、考 慮している	3
5	プログラム言語と開発環境などとの相性を考慮している	6
6	その他	2

注) 回答企業数 28 社。

表 19 開発で使用するハードウェアの制限の考慮の仕方

	開発で使用するハードウェアの制限の考慮の仕方	回答企業
1	全く考慮していない	1
2	開発で使用するハードウェアの制限が特に厳しい場合のみを考慮し ている	14
3	開発で使用するハードウェアの制限の厳しさを段階評価し、考慮して いる	4
4	開発で使用するハードウェアのうち、最も制限が厳しいものを明確に し、評価・考慮している	2
5	開発で使用する各々のハードウェアについて、制限の厳しさを評点付 けし、考慮している	3
6	その他	3

注) 回答企業数 28 社、未回答企業数 1 社。

表 20 開発に掛かるコストに対する顧客と開発者側との認識の相違の考慮の仕方

	開発にかかるコストに対する顧客と開発者側との認識の相違の考慮の仕方	回答企業
1	全く考慮していない	0
2	特に大きな認識の相違がある場合のみ考慮している	8
3	認識の相違の大きさを段階評価し、考慮している	8
4	ソフトウェアまたは開発プロセスを要素に分け、その中で、開発にかかるコストに対する認識の相違が最も大きい部分を明確にし、評価・考慮している	2
5	ソフトウェアまたは開発プロセスを要素に分け、要素毎に、開発にかかるコストに対する認識の相違を評価し、考慮している	9
6	その他	1

注) 回答企業数 28 社。

表 21 開発予算の関係で開発にかけられる人数が限られている状況の考慮の仕方

	開発予算の関係で開発にかけられる人数が限られる状況の考慮の仕方	回答企業
1	全く考慮していない	1
2	特に開発予算が限られている場合のみを考慮している	9
3	開発予算が限られている状況を段階評価し、考慮している	7
4	開発予算の中で最も厳しい項目を明確にし、評価・考慮している	1
5	開発予算の各項目の厳しさを評点付けし、考慮している	1
6	その他	9

注) 回答企業数 28 社。

4.4 工数見積もり誤りとその原因

(1) 工数見積もり誤りがおきる原因：見積もりプロセス

回答企業が、どのプロセスが工数見積もり誤りを引き起こしていると考えているのかを明らかにするために、調査項目Ⅲ-(2)の「工数見積もり誤りが起きる原因：見積もりプロセス」で得られたデータを帯グラフにより整理した。結果を図4に示す。

図4より以下のことが分かった。

- ①「開発規模の見積もり」について、「原因になっていると思う」、「どちらかといえば原因になっていると思う」と回答した企業が約半数ある。
- ②他方、「作業の分解」と「開発工数の見積もり」については、「原因になっていると思う」、「どちらかといえば原因になっていると思う」と回答した職場が3～4割であり、相対的に少ない。

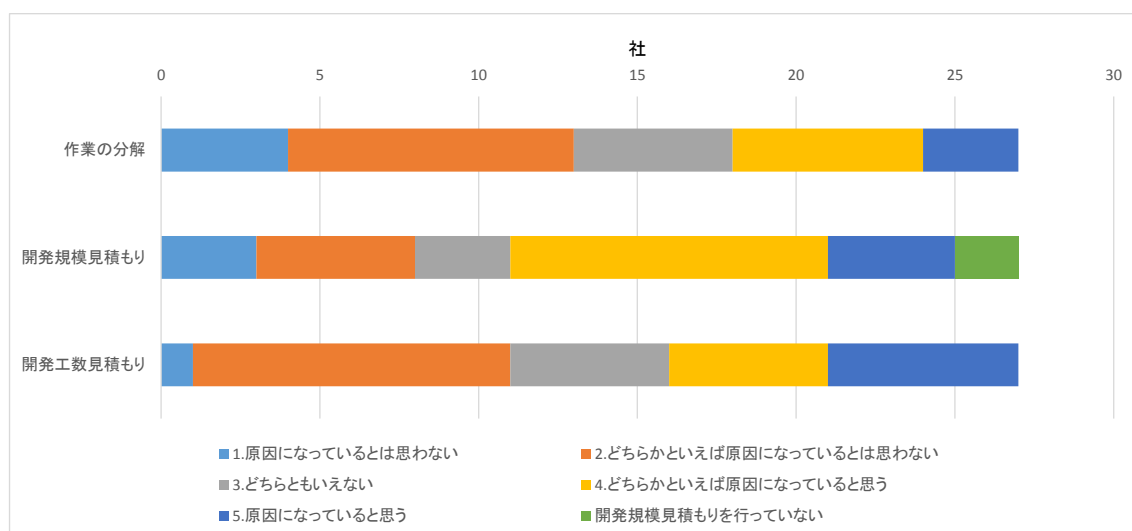


図4 工数見積もり誤りの原因：見積もりプロセス

注) 回答企業 27 社、未回答企業 1 社。

(2) 工数見積もり誤りが発生する原因：見積もりにおける要因の考慮不足

回答企業が、どのような要素を考慮できていないことが工数見積もり誤りの原因と考
えているのかを明らかにするために、調査項目Ⅲ-(3)の「工数見積もり誤りが発生する
原因：見積もりにおける要因の考慮不足」で得られたデータを帯グラフにより整理した。
結果を図5に示す。

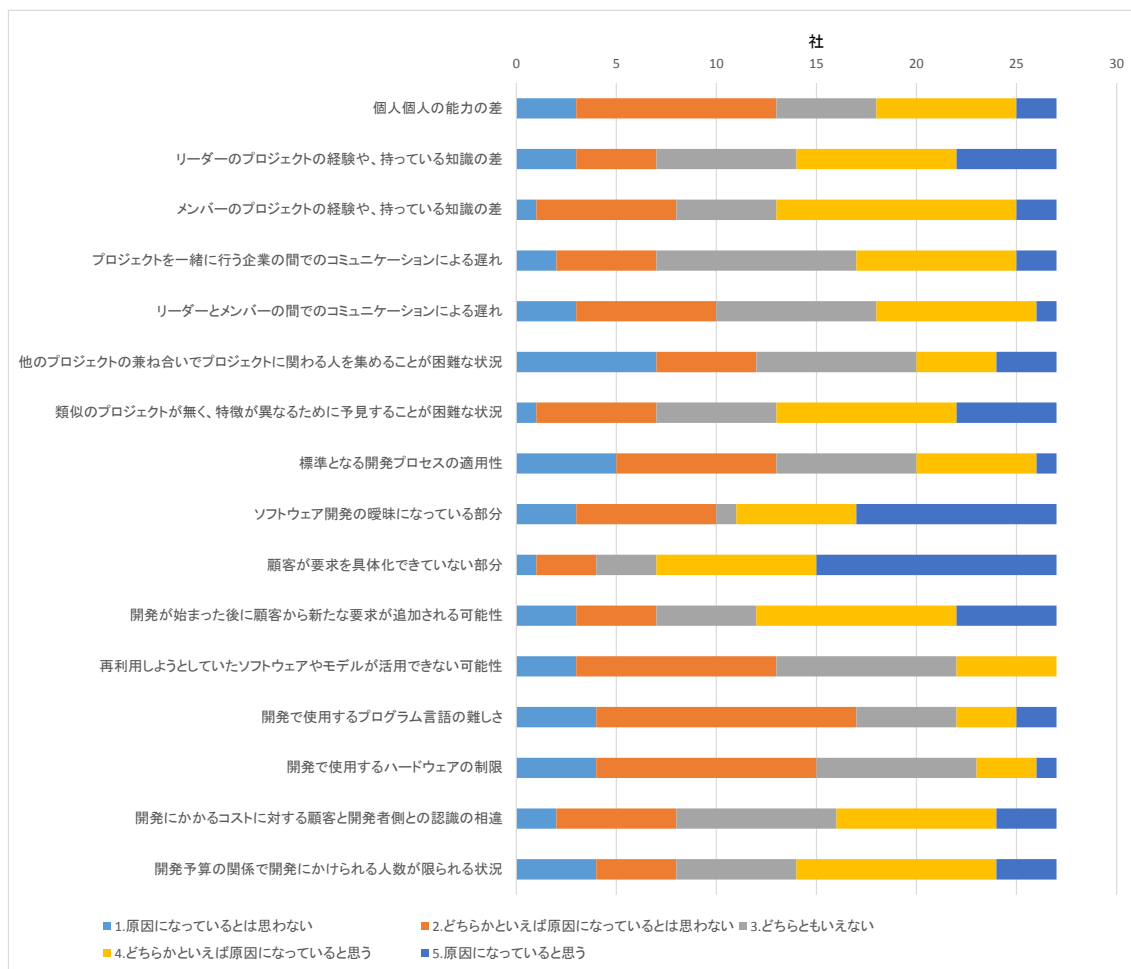


図5 工数見積もり誤りの原因：見積もりにおける要因の考慮不足
注)回答企業27社、未回答企業1社。

図5より以下のことが分かった。

- ①7割ほどの企業が原因と考えているものは、「顧客が要求を具体化できていない部分」を考慮できていないことが原因であると考えている企業が多く、約7割の企業が「原因となっていると思う」、「どちらかと言えば原因となっていると思う」と答えている。
- ②「リーダーのプロジェクトの経験や、持っている知識の差」、「メンバーのプロジェクトの経験や、持っている知識の差」、「類似のプロジェクトが無く、特徴が異なるため

に予見することが困難な状況」、「ソフトウェア開発の曖昧になっている部分」、「開発が始まった後に顧客から新たな要求が追加される可能性」、「開発にかかるコストに対する顧客と開発者側との認識の相違」、「開発予算の関係で開発にかけられる人数が限られる状況」を考慮できていないことが原因と考えている企業も多く、約半数の企業が「原因となっていると思う」。「どちらかと言えば原因となっていると思う」と答えている。

- ③多くはないが、「再利用しようとしていたソフトウェアやモデルが活用できない可能性」、「開発で使用するプログラム言語の難しさ」、「開発で使用するハードウェアの制限」を考慮できていないことが原因と考えている企業が約3割ある。
- ④「個人個人の能力の差」、「プロジェクトお一緒に行う企業の間でのコミュニケーションによる遅れ」、「リーダーとメンバー間でのコミュニケーションによる遅れ」、「他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況」、「標準となる開発プロセスの適用性」を考慮できていないことが原因と考えている企業はほとんどない。

4.5 工数見積もり方法と誤りの関係

どのような要因の考慮不足が工数見積もり誤りに影響を与えているかを明らかにするために、「工数見積もり誤りの発生割合」を目的変数、「工数見積もりのプロセス」および「工数見積もり誤りに影響を与える要因をどのように考慮しているか」を説明変数にして重回帰分析を行った。ここで、「工数見積もり誤りの発生割合」は値が大きいほど誤りの発生割合が高いことを、「工数見積もり誤りに影響を与える要因をどのように考慮しているか」は値が大きいほど考慮が徹底していることを表している。なお、F 値による減数減増法($F_{in} = F_{out} = 2.0$)を用いて変数の選択を行った。結果を表 22 に示す。

表 22 要因の考慮が工数見積もり誤りの発生割合に与える影響

説明変数名	偏回帰係数	t 値	P 値(両側)
定数項	7.723	4.333	0.001
見積もりのプロセス	-1.480	-4.296	0.001
リーダーのプロジェクトの経験や、持っている知識の差	0.435	2.140	0.052
プロジェクトを一緒に行う企業の間でのコミュニケーションによる遅れ	-0.449	-1.728	0.108
他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況	-0.929	-3.834	0.002
類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況	0.792	2.678	0.019
標準となる開発プロセスの適用性	-0.653	-2.174	0.049
ソフトウェア開発の曖昧になっている部分	-1.030	-2.596	0.022
開発が始まった後に顧客から新たな要求が追加される可能性	0.456	1.941	0.074
開発で使用するプログラム言語の難しさ	-0.259	-1.616	0.130
開発にかかるコストに対する顧客と開発者側との認識の相違	0.843	4.849	0.000

注) 重相関係数 0.873、残差自由度 13、残差標準偏差 0.849。

表 22 より以下のことが分かった。

- ①企業による工数見積もり誤りの割合の違いの大半は、見積もりプロセスの違いや、見積もり誤りに影響を与える要因の考慮の仕方の違いによって説明できる（寄与率 76.2%）。逆に言えば、見積もりプロセスや要因の考慮を適切に行うことで、工数見積もり誤りの発生割合を低減できると考えられる。
- ②「見積もりのプロセス」は見積もり誤りの発生割合に影響を与えている。作業の分解・構造化→開発規模の見積もり→開発工数の見積もりというステップを踏んでいる企業の方が、見積もり誤りの発生割合が低い。
- ③A.人的要因に関する考慮不足について見ると、「リーダーのプロジェクトの経験や、持っている知識の差」、「プロジェクトを一緒に行う企業の間でのコミュニケーション

による遅れ」、「他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況」の考慮が大きな影響を与えている。また、B. 問題要因の考慮不足について見ると、「ソフトウェア開発の曖昧になっている部分」、「類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況」の考慮が大きな影響を与えている。C. プロセス要因に対する考慮不足については、「標準となる開発プロセスの適用性」、「開発で使用するプログラム言語の難しさ」の考慮が、D. プロダクト要因に対する考慮不足については、「開発が始まった後に顧客から新たな要求が追加される可能性」、「開発にかかるコストに対する顧客と開発者側との認識の相違」の考慮が大きい影響を与えている。E. 資源要因に対する考慮不足は大きな影響を与えていない。

- ③上記のうち、「プロジェクトを一緒に行う企業間でのコミュニケーションによる遅れ」、「他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況」、「ソフトウェア開発の曖昧になっている部分」、「標準となる開発プロセスの適用性」、「開発で使用するプログラム言語の難しさ」の考慮については、より徹底して考慮している企業の方が、工数見積もり誤りの発生割合が低い。他方、「リーダーのプロジェクトの経験や、持っている知識の差」、「開発が始まった後に顧客から新たな要求が追加される可能性」、「開発にかかるコストに対する顧客と開発者側との認識の相違」については逆の傾向となっている。このような考慮を行っている企業ほど、これらの考慮が必要となる難しいプロジェクトを行っているためだと考えられる。

重回帰分析を行って得られた結果（工数見積もり誤りの発生割合に影響を与えていると統計的に判断できるもの）と、4.4の結果（回答企業が原因と考えているもの）の共通点・相違点をまとめた結果を、表 23 に示す。

表 23 より以下のことが分かった。

- ①重回帰分析で有意であり、原因と考えている企業が多いものは、「見積もりプロセス」と「リーダーのプロジェクトの経験や、持っている知識の差」、「類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況」、「ソフトウェア開発の曖昧になっている部分」、「開発が始まった後に顧客から新たな要求が追加される可能性」の考慮である。
- ②他方、重回帰分析で有意であり、原因と考えている企業が少ないものは、「プロジェクトを一緒に行う企業間でのコミュニケーションによる遅れ」、「他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況」、「標準となる開発プロセスの適用性」、「開発で使用するプログラム言語の難しさ」、「開発にかかるコストに対する顧客と開発者側との認識の相違」の考慮である。

- ③重回帰分析で有意ではなく、原因として考えている企業が多いものは、「メンバーのプロジェクトの経験や、持っている知識の差」、「顧客が要求を具体化できていない部分」、「開発予算の関係で開発にかけられる人数が限られる状況」の考慮である。
- ④重回帰分析で有意ではなく、原因とを考えている企業も少ないものは、「個人個人の能力の差」、「リーダーとメンバーの間でのコミュニケーションによる遅れ」、「再利用しようとしていたソフトウェアやモデルが活用できない可能性」、「開発で使用するハードウェアの制限」の考慮である。

表 23 企業の回答と重回帰分析の共通点・相違点

重回帰分析 企業の回答	有意である	有意ではない
原因と考えている 企業が多い	<ul style="list-style-type: none"> ・見積もりプロセス ・リーダーのプロジェクトの経験や、持っている知識の差の考慮 ・類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況の考慮 ・ソフトウェア開発の曖昧になっている部分の考慮 ・開発が始まった後に顧客から新たな要求が追加される可能性の考慮 	<ul style="list-style-type: none"> ・メンバーのプロジェクトの経験や、持っている知識の差の考慮 ・顧客が要求を具体化できていない部分の考慮 ・開発予算の関係で開発にかけられる人数が限られる状況の考慮
原因と考えている 企業が少ない	<ul style="list-style-type: none"> ・プロジェクトを一緒に行う企業間でのコミュニケーションによる遅れの考慮 ・他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況の考慮 ・標準となる開発プロセスの適用性の考慮 ・開発で使用するプログラム言語の難しさの考慮 ・開発にかかるコストに対する顧客と開発者側との認識の相違の考慮 	<ul style="list-style-type: none"> ・個人個人の能力の差の考慮 ・リーダーとメンバーの間でのコミュニケーションによる遅れの考慮 ・再利用しようとしていたソフトウェアやモデルが活用できない可能性の考慮 ・開発で使用するハードウェアの制限の考慮

4.6 工数見積もり誤りの原因を改善する難しさとその克服策

4.5 で特定できた工数見積もりの誤りの原因（原因と考えている企業が多くかつ重回帰分析で有意になったもの）をどのようにして改善するのがよいのかを明らかにするために、アンケートⅣ-(1)、(2)の「工数見積もり誤りの原因を改善する難しさと克服のために実践している/しようとしていること」で得られたデータを KJ法により整理した。結果の一部を表 24～表 37 に示す。

表 24～表 37 より以下のことが分かった。

- ①作業の分解を行うプロセスを改善する難しさは、主に、「作業の漏れ、予見できない作業が発生する」と「作業の分解に個人差が発生する」である。これらの克服策としては、「顧客、有識者によるレビューを行う」、「作業の分解をパターン化する」、「作業の分解を行える能力を育成する」ことなどが行われている。また、「見積もり期間が短い」という難しさもあり、これについては「見積もりの期間を顧客と調節する」ことが行われている。
- ②開発規模見積もりを行うプロセスを改善する難しさは、「新規性が高く既存の規模見積もりプロセスが適用できない」、「範囲・粒度の特定」、「実装方法の想定」、「規模見積もりに関する知識・スキル不足」である。これらの克服策としては、「顧客に範囲・粒度を確認する」、「事前に技術的な検証を行う」、「経験者によるレビューを行う」、「規模の見積もりを行う能力を育成する」、「標準化を進め、類似プロジェクトに関する情報を蓄積する」ことが行われている。
- ③開発工数見積もりを行うプロセスを改善する難しさは、「経験がないものの工数を見積もらなければならない」、「業界や客先、言語等による生産性の違いがあり、その見積もり方法が確立していない」、「工数見積もりに関する知識・スキル不足」である。これらの克服策としては、「経験者や仲間内によるレビューを行う」、「工数の見積もりを行う能力を育成する」、「層別、実績のフィードバック、複数手法の併用を行う」が行われている。また、「不適切な作業の分解」など、前のプロセスが適切に行われていないことも難しさになっている。
- ④リーダーのプロジェクトの経験や知識の差を考慮する難しさは、「能力を定量的に評価できない」、「リーダーの経験・知識の差が多種多様である」、「経験・知識が同じでも能力が異なる」である。また、「能力の差を考慮することに対する反対がある」や「必要な能力を持った人が確保できない」などもあがっている。これらの克服策としては、「経験・知識や能力を評価する工夫をする」、「教育研修を行う」、「途中でレビューを行う、支援者を付ける」、「情報を共有しメンバーがサポートする」が行われている。また、「予備費を持つようにする」という対策も見られる。
- ⑤類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮する難しさは、「未経験のものを見積もる手法がない」、「人の経験に大きく依存する」、「必要な余裕度が判断できない」、「内容を詳細化できない」、「見積もりが困難な状況を想

定できない」である。これらの克服策としては、「見積もり手法を教育する」、「リスクを明確にし、コンティジェンシープランを作成する」、「見積もりを大目に行う、プロジェクト期間を長めにとる」、「仕事を受けない」、「プロジェクトに関する情報を広く収集する」ことが行われている。

- ⑥ソフトウェア開発の曖昧になっている部分を考慮する難しさは、「曖昧な部分に気づけない、分からない」、「曖昧な部分による工数増を見積もれない」、「顧客に要求の具体化を求めることが出来ない」がある。これらの克服策としては、「コミュニケーション力を育成する」、「前提条件や曖昧な点を顧客とあらかじめ合意しておき、変更がある場合は再見積もり・追加見積もりを行う」、「曖昧さを客観的に評価し、見積もりに反映する」、「レビュー等により曖昧な部分を残さないようにする」ことが行われている。
- ⑦開発が始まった後に顧客から新たな要求が追加される可能性を考慮することの難しさは、「事前の予測が難しい」、「顧客の理解を確認できない」、「規模を見積もることが難しい」「開発が進まないと具体化されない」がある。これらの克服策としては、「類似のプロジェクトを参考にする」、「新たな要求は追加見積もりになることを顧客と合意しておく」、「要求を具体化・可視化・細目化したものを用いて顧客と打ち合わせを行う」ことが行われている。

表 24 作業の分解を行うプロセスを改善する難しさ

作業の分解を行うプロセスを改善する難しさ	作業の漏れ、予見できない作業が発生する	<ul style="list-style-type: none"> ・予見できない作業が発生するリスク ・作業の分解に漏れがある ・作業の漏れなく分解すること ・基本的な作業分解はガイド類の整備で担保できるが、顧客・業務・ソリューション等の各々の特性まで勘案した必要作業の網羅性
	作業の分解に個人差が発生する	<ul style="list-style-type: none"> ・作業を分解する粒度が分解を行う人によりまちまちである ・自身の経験に基づいて洗い出され、やったことの無い作業が抜けている ・見積期間が短い場合、「経験の豊富な人間」「作業を細分化できるSEとしての能力」「多くの人間の稼働の動きを把握できる能力」が必要となるため、人材が枯渇している。そのため、個人毎で見積レベルの差が発生している ・分解作業におけるリスク想定等のスキルアップ、リスク想定スキルアップ
	見積もり期間が短い	<ul style="list-style-type: none"> ・見積り期間、及び見積り費用が十分でない場合に、作業を詳細まで洗い出すことが出来ない

表 25 作業の分解を行うプロセスを改善する難しさの克服策

作業の分解を行うプロセスを改善する難しさの克服策	顧客、有識者によるレビューを行う	<ul style="list-style-type: none"> ・作業の分解結果を依頼者にレビューしてもらう ・見積り時に第三者組織＋有識者によるレビューを設けており、漏れのチェックをはじめ、多面的な助言を実施 ・フェーズの進行に伴って段階的に詳細化していき、有識者とのレビューを行う
	作業の分解をパターン化する	<ul style="list-style-type: none"> ・SOWの明確化を行う ・分解する粒度を1日や1週間などと決めて、作業を分解する ・設計やコーディングなどのプロセスでの共通作業をPJとして規定する ・過去の類似案件を参考にし、見積り作業をパターン化する
	作業の分解を行える能力を育成する	<ul style="list-style-type: none"> ・開発経験を積んできた人間(リーダーも経験した)を営業の立場としても教育し稼働(お金)の計算ができるようにする ・社内教育の徹底、レベルアップ、上位有識者のサポート
	見積り期間を顧客と調節する	<ul style="list-style-type: none"> ・見積り期間を十分とれるよう顧客と調整する

表 26 開発規模見積もりを行うプロセスを改善する難しさ

開発規模 見積もりを 行うプロセスを改善 する難しさ	新規性が高く既存の規模見積もりプロセスを適用できない	・新領域(業界・業務・技術・ソリューション等)は、当社の見積もりプロセスが適用しにくい
	範囲・粒度の特定	・顧客ごとに求められる成果物の粒度が異なること ・開発の対象(スコープ)の把握。既存システムの場合、修正に伴う影響範囲の特定。また、ステークホルダー(プロダクト担当者)の把握。
	実装方法の想定	・想定していた実装方法では実現できない場合がある
	規模見積もりに関する知識・スキル不足	・見積もりする人のスキル、経験に依存する ・規模見積り範囲の見極めに関するスキルアップ、リスク想定スキルアップ ・FP 法など見積もり手法の知識が乏しく、今までのやり方で見積もってしまうこと ・過去の生産性を考慮できること。見積り手法に適用させるための作成物の数をヒアリングと共に実施することは難しい。個人レベルで差が出てしまう。標準となる見積り基準の選定。 案件毎の開発手法や個別事情による工数の補正
	見積もり期間が短い	・見積り期間、及び見積り費用が十分でない場合に、作業内容を詳細まで詰め切ることが出来ない
	見積もりと実績の乖離の改善方法が分からない	・見積り結果と実績規模が解離しているということだけでは、どう改善すればいいのかわからない ・妥当性のある開発規模を見積もること

表 27 開発規模見積もりを行うプロセスを改善する難しさの克服策

開発規模 見積もりを 行うプロセスを改善 する難しさ を克服の ために実践 している /しようと していること	顧客に範囲・粒度を確認する	・弊社の標準的な成果物サンプルをベースに、どのレベルを求めているかを加減している ・見積書のレビューにおいて、スコープの確認、影響範囲の確認、ステークホルダーの確認と承認を実施。特に調査者と承認者を明確にし、責任を明確にする
	事前に技術的な検証を行う	・サンプルプログラム等による事前に技術検証を行う
	経験者によるレビューを行う	・経験者による第三者レビュー(このレビューもスキル、経験に依存する) ・規模見積り根拠の合理性をどこまでどのように検証しているかのチェックを実施 ・標準化した見積りプロセスに準拠するよう指導しているが、プロジェクト毎の個別性が高いため、見積り基準は画一化せず、個別設定させ、その根拠を確認するようにしている ・上位有識者のサポート、
	規模見積もりを行う能力を育成する	・FP 法など見積り手法の教育 ・社内教育の徹底、レベルアップ
	標準化を進め、類似プロジェクトに関する情報を蓄積する	・規模見積りに必要な要件から機能の洗い出し、データ洗い出しに焦点をあてている ・類似 PJ からの類推、既存規模からの類推、RFP で提供される機能一覧からの積上げを併用し、積上げ(LOC)か FP を採用するために、事例の蓄積・統計化を行っている ・類似する過去プロジェクトを参考にする ・過去の類似案件を参考にし、見積り作業をパターン化する ・開発規模見積りの標準化
	見積り期間を長く取る	・見積り期間を十分とれるよう顧客と調整する
	原因を分析し改善する	・実際に解離した事例から原因を分析し、プロセスの改善のポイントを洗い出す。再発しない為の施策を考え、プロセスを改善させる

表 28 開発工数見積もりを行うプロセスを改善する難しさ

開発工数 見積もりを 行うプロセスを改善 する難しさ	不適切な作業の 分解	・工程の後半になるほど WBS が荒くなり工数見積精度が落ち、FP 生産性はフェーズ別の工数と相性が悪い。
	経験がないもの の工数を見積も なければならない	・経験のあるもの(業務的、技術的)なものでない場合の工数の妥当性や開発の機能の重みづけが見誤る場合がある。
	業界や客先、言 語等による生産 性の違いがあり、 その見積もり方法 が確立していない	・業界や客先、言語等による生産性の違い ・作業の難易度とそれに合った生産性を見極め ・工数計算(規模/生産性)で使う生産性を見積もりプロセスが確立していない。類似 PJ 事例、過去 PJ 実績等を採用しているが、新分野適用におけるチェック方法がない ・妥当な計数(生産性など)を見積もりこと
	工数見積もりに関 する知識・スキル 不足	・見積もりする人のスキル、経験に依存する ・見積もり者の経験不足 ・COCOMO II 法など見積もり手法の知識が乏しく、今までのやり方で見積もってしまうこと ・作業には必ず待ち時間や遅れも生じる可能性があり、各作業(タスク)の切り替わりの際の時間の考慮等も経験則で加味する必要がある。リスク想定スキルアップ。
	見積もりと実績の 乖離の改善の方 法が分からない	・見積もり結果と実績規模が解離しているということだけでは、どう改善すればいいのかわからない

表 29 開発工数見積もりを行うプロセスを改善する難しさの克服策

開発工数 見積もりを 行うプロセスを改善 する難しさ を克服の ために実 践している /しようと していること	経験者や仲間内 によるレビューを 行う	・経験者による第三者レビュー(このレビューイもスキル、経験に依存する) ・複数様式での見積もり、見積もりレビュー ・複数メンバーでの工数見積もりの共有、検討、リスク箇所の洗い出し、工数を乗せる等を実施
	工数見積もりを行 う能力を育成する	・COCOMO II 法など見積もり手法の教育 ・経験・ノウハウの蓄積、情報共有 ・社内教育の徹底、レベルアップ、上位有識者のサポート、開発規模見積りの標準化
	層別、実績のフィ ードバック、複数 手法の併用を行 う	・作業難易度をレベル分けして工数を載せる ・外部設計以降の見積もりは、FP を必須として、実績をフィードバックし、係数精度(生産性精度)を上げていくことを検討中 ・WBS による積み上げ見積もりと LOC による見積もりと両面で見積もり比較検討する
	原因を分析し改 善している	・実際に解離した事例から原因を分析し、プロセスの改善のポイントを洗い出す。再発しない為の施策を考え、プロセスを改善させる 案件、顧客の特性を考慮(リスクとして)し工数に余力を持たせるように検討している

表 30 リーダーのプロジェクトの経験や知識の差を考慮する難しさ

リーダーのプロジェクトの経験や知識の差を考慮する難しさ	能力の差を考慮することに対する反対がある	<ul style="list-style-type: none"> ・ 工数見積においては、個人の能力差が工数に現れてはいけなくていいと考えているので、そこに難しさがある
	能力を定量的に評価できない	<ul style="list-style-type: none"> ・ リーダーの能力を定量的に測ること ・ 外部設計以降の見積もりは、FP を必須として、実績をフィードバックし、係数精度(生産性精度)を上げていくことを検討中
	リーダーの経験・知識の差が多種多様である	<ul style="list-style-type: none"> ・ リーダーにも「初心者」や「ベテラン」がいる ・ スキル、経験が同じ人はいないので、バラツキが出てしまう ・ 「経験と知識の差」は一律で無い ・ 経験年数や知識(業務知見、技術的知見)は各リーダーにより異なるため異なった見積が出てくる ・ 経験や知識の差の有無の判断基準や判断方法
	経験・知識が同じでも能力が異なる	<ul style="list-style-type: none"> ・ 経験年数や過去プロジェクトの成否は、見積りを加減する材料のひとつにはなる反面、その加減値の妥当性を見極めるの難しさ ・ リーダーの経験と実態が伴っていない場合がある
	必要な能力を持った人が確保できない	<ul style="list-style-type: none"> ・ 上位有識者の不足

注) その他の回答：リーダーによってチームの生産性が大きく変わる。機械的に作業しても結果は出ない

表 31 リーダーのプロジェクトの経験や知識の差を考慮する難しさの克服策

リーダーのプロジェクトの経験や知識の差を考慮する難しさを克服するために実践している/しようとしていること	経験・知識や能力を評価する工夫をする	<ul style="list-style-type: none"> ・ 経験や知識の差をレベル分けし、係数化する ・ 5段階で計数を掛けるなどして補正する ・ 過去の経験(活動)の見える化に努める。実体を把握する ・ 見積もり時は、PM・各リーダー経歴評価の徹底 ・ 技術者スキルの定量的把握
	教育研修を行う	<ul style="list-style-type: none"> ・ 社員研修(リーダー研修)の実施 ・ 管理者教育
	途中でレビューを行う、支援者を付ける	<ul style="list-style-type: none"> ・ 経験者による第三者レビュー(このレビューもスキル、経験に依存する) ・ 明らかに経験や知識が不足している場合や、経験上そのおそれがあると判断した場合には、PM/リーダーの支援者として経験者や有識者のアサインを指導しています ・ プロジェクト推進時は、推進状況のウォッチと軌道修正 ・ 見積り値自体の増減ではなく、プロジェクト体制におけるリーダー経験値に応じたリーダー補佐の設置や、PMO要員の増員などにより対応 ・ PMOからのサポート
	情報を共有しメンバーがサポートする	<ul style="list-style-type: none"> ・ チームの雰囲気作り。メンバーとのコミュニケーションを多くとる。 ・ プロジェクト報告会での情報共有 ・ 複数人数による情報の共有、チェック。リーダーが不足している経験箇所を他のメンバーから情報を貰い補う等
	予備費を持つようにする	プロジェクト予備費を持つ

表 32 類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮する難しさ

類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮する難しさ	未経験のものを見積もる手法がない	<ul style="list-style-type: none"> ・ 未経験プロジェクトへの対応体制の数値化 ・ 類似のプロジェクトがない場合の見積もり手法
	人の経験に大きく依存する	<ul style="list-style-type: none"> ・ 予見(想定)を行った人物の経験により変わるため ・ 見積基準の妥当性を評価できない
	必要な余裕度が判断できない	<ul style="list-style-type: none"> ・ 比較対象が無いために客観的評価が難しい上に、余裕度(予備費)を数値化して見積りに加える事の難しさ ・ 上積みレベルのバラツキをどう判断/認識して反映させていくか
	内容を詳細化できない	<ul style="list-style-type: none"> ・ 過去の知識が生かせないため、見積を細かくすることが難しい ・ 分からない場合に、細分化ができず曖昧に判断してしまることがある
	見積もりが困難な状況を想定できない	<ul style="list-style-type: none"> ・ 予見が困難なものへの見積もりはできないこと ・ 困難な状況が想定できないことがある ・ 経験が無いので影響を考慮できない ・ リスクの明確化

その他の回答：業務側(発注元)、システム側(当社)、パートナー(再委託先)のいずれかで業務知識が不足している場合、開発要件が不足する

表 33 類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮する難しさの克服策

類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮する難しさを克服するために実践している/しようとしていること	見積もり手法を教育する	<ul style="list-style-type: none"> ・ 社員研修の実施 ・ FP 法などの見積もり手法の教育
	リスクを明確にし、コンティンジェンスプランを作成する	<ul style="list-style-type: none"> ・ 要件不足のリスク、要件定義時の前提を業務側(発注元)と共有し、リスク顕在化の際の予算処置およびコンティンジェンスプランを共有する ・ 見積とは別にリスク項目を洗い出し、リスク額を評価、見積に合算する形でコストを算出する ・ リスクチェックシートによるリスク識別及びリスク度を、客観的に評価している
	見積もりを大目に行う、プロジェクト期間を長めに取る	<ul style="list-style-type: none"> ・ 過去、少しでも経験したメンバーにヒアリングし、不透明な部分には大目の見積もりを実施する ・ ある程度余力をもったプロジェクト遂行 ・ 見積期間を通常よりも長く取れるように交渉する。プロジェクトの期間および工数についてもバッファを設ける
	仕事を受けない	<ul style="list-style-type: none"> ・ あまりにも不透明な部分が多い場合には、仕事を受けない
	プロジェクトに関する情報を広く収集する	<ul style="list-style-type: none"> ・ データ活用/収集範囲の拡大、協調フィルタリング手法を活用し、統計的に参考になりそうなPJの自動抽出の提供 ・ 別の要員から情報をもらい、別途、内容を明確にもらえるように事前情報を集める等を行う ・ プロジェクト間情報共有(プロジェクト報告会実施)

表 34 ソフトウェア開発の曖昧になっている部分を考慮する難しさ

ソフトウェア開発の曖昧になっている部分を考慮する難しさ	曖昧な部分に気づけない、分からない	<ul style="list-style-type: none"> どこが曖昧かに気付けない 曖昧さを明確にすること システム方式設計の完成度に対する見極め 第三者による評価が難しい
	曖昧な部分による工数増を見積もれない	<ul style="list-style-type: none"> 曖昧な箇所に対する見積もりは困難であること 見積もり段階で曖昧な部分がどのように変動するか予測が難しい 曖昧な部分の見積もりをどうやって精度を上げていくか 工数への影響度合い測定が難しい 全量を把握することが困難 曖昧な部分は想定で見積もりを作成するが、規模が膨らむ 全く新規の開発となる可能性があること
	顧客に要件の具体化を求められない	<ul style="list-style-type: none"> 開発するソフトの曖昧さの原因は、顧客の要件定義の曖昧さに起因していることが多い。しかし顧客に要件の具体化を強く求めても限界があることの難しさ 曖昧さの排除 顧客の決定の遅れ 顧客の要件が曖昧な場合が多い

その他の回答：仕様が曖昧で開発着手後に想定外の実装が増えてしまう

表 35 ソフトウェア開発の曖昧になっている部分を考慮する難しさの克服策

ソフトウェア開発の曖昧になっている部分を考慮する難しさを克服するために実践している/しようとしていること	コミュニケーション力を育成する	<ul style="list-style-type: none"> コミュニケーションに関する講習の実施などの教育訓練 教育、経験の積み上げ。
	前提条件や曖昧な点を顧客とあらかじめ合意しておき、変更がある場合は再見積もり・追加見積もりを行う	<ul style="list-style-type: none"> 見積算出の前提とした条件を整理し、見積提示時に顧客と合意をとるよう指導 曖昧な部分がどこか、いつまでに明確にするかを顧客と合意し、それまでは概算見積もりとする。要件が決まった時点で、再度見積もりを行う 未確定部分の把握 曖昧な部分については、当方側の想定または実施条件を記載し、外れる場合は別途対応(コスト・スケジュール等)を協議し対応するようにしています 見積りの前提をハッキリ顧客に示しておき、課題管理のような方法で書面化しつつ、具体化した段階で、前提との差に応じて顧客に対して追加請求を行う 提案型で想定したものを事前に説明する。大きく開発に影響するものは無償で受けないこととスケジュール変更が必要なことを事前に確約する 未決定部分の別契約
	曖昧さを客観的に評価し、見積もりに反映する	<ul style="list-style-type: none"> アーキテクチャの明確度(システム方式設計の完成度)を客観的に評価し、リスクとして想定し見積もりに反映
	レビュー等により曖昧な部分を残さないようにする	<ul style="list-style-type: none"> 設計レビューを行い、曖昧な部分を残さないようにしている 第三者によるレビューなど 適用するソフトウェアについて事前に使用し、製造メーカーとの打合せを経て曖昧さを無くした状態で見積りようになっている

表 36 開発が始まった後に顧客から新たな要求が追加される可能性を考慮することの難しさ

開発が始まった後に顧客から新たな要求が追加される可能性を考慮する難しさ	事前の予測が難しい	<ul style="list-style-type: none"> ・ 事前に予測ができない ・ 何が発生するか分からないので困難
	顧客の理解を確認できない	<ul style="list-style-type: none"> ・ 実装仕様を提示・説明し了承をもらった場合でも追加変更はある ・ 顧客は要件定義に入れたつもりでも、記載に抜けや漏れが発生する可能性がある ・ 機能の粒度が大きすぎる場合に発生する。顧客の粒度と開発側に粒度があわないことで見積もと乖離することがある
	規模を見積もることが難しい	<ul style="list-style-type: none"> ・ 追加規模を予測するのが難しい ・ 要求の追加をどの程度見込むか ・ どれくらいの追加要求があると見込み、見積もりに積み上げるか ・ 追加要求がいつどれくらい生じるかを予見して数字上に盛り込むことは現実的には難しいこと ・ 追加されるボリュームを予測すること
	開発が進まないとい具体化されない	<ul style="list-style-type: none"> ・ 開発が進むにつれ顧客側のイメージが具現化されていき、目的レベルの要求範囲で見積もり範囲を超える要求が出る ・ 顧客の決定の遅れ

表 37 開発が始まった後に顧客から新たな要求が追加される可能性を考慮する難しさの克服策

開発が始まった後に顧客から新たな要求が追加される可能性を考慮する難しさを克服するために実践している/しようとしていること	類似のプロジェクトを参考にする	<ul style="list-style-type: none"> ・ 類似のプロジェクトの値を参考にする
	新たな要求は追加見積もりになることを顧客と合意しておく	<ul style="list-style-type: none"> ・ 変更管理の実施を見積もり前提条件として記載。プロジェクト開始時にプロジェクト計画内でそのプロセスを顧客と合意する。 ・ 実装仕様について了承を得た後の仕様変更・追加は別見積りとなることを見積条件に入れている ・ 追加要求に係る開発見積もりの顧客との共有化 ・ 追加要件の発生に対しては予め顧客との間で、いつまでなら受入れ可能か、但し受け入れ可能でも追加費用や期間延長が生じる可能性について提案に盛り込んでおく ・ 発生した段階で再見積りを実施するようにしている ・ 要求が追加された時点で、追加見積りを行う ・ 追加分は別見積りとする ・ 未決定部分の別契約など
	予備費を持っておく	<ul style="list-style-type: none"> ・ 新規顧客や、追加要求が見込まれているプロジェクトについては、リスクコンティンジェンシーとしてコストをあらかじめ計上しておく ・ プロジェクト予備費を持つ
	要求を具体化・可視化・細目化したものを用いて顧客と打ち合わせを行う	<ul style="list-style-type: none"> ・ 要求レベルを具体的にヒアリング(あまり考慮しすぎると見積額が高くなり受注できなくなる) ・ 開発するシステムの目的(業務課題)を明確化し、顧客の要求を図・表などでできるだけ視覚化することで仕様の抜けや漏れが内容に要件定義を作成している ・ 開発サイドは顧客に対して細分化した機能で見積りを説明するように努めている

第5章 考察

4.4 および 4.5 では、工数見積もり誤りの原因に関する企業の回答と、工数見積もり誤りの発生割合と工数見積もりの仕方に関する重回帰分析の結果より、「見積もりプロセス」、「リーダーのプロジェクトの経験や、持っている知識の差の考慮」、「類似のプロジェクトが無く、特徴が異なるために予測することが困難な状況の考慮」、「ソフトウェア開発の曖昧になっている部分の考慮」、「開発が始まった後に顧客から新たな要求が追加される可能性の考慮」が重要な原因になっていると考えられることがわかった。

4.3 の結果に基づいて、これらの原因についての回答企業における現状のレベルをまとめると、表 38 の通りとなる。この表より、工数見積もり誤りを改善するためには、「リーダーのプロジェクトの経験や、持っている知識の差の考慮」、「類似のプロジェクトが無く、特徴が異なるために予測することが困難な状況の考慮」に取り組むことが必要と考えられる。

表 38 工数見積もり誤りを改善するために強化すべき取り組み

重要な原因	見積もりのプロセスの現状	現状のレベル
見積もりプロセス	<ul style="list-style-type: none"> ・作業の分解(WBS等)→開発規模見積もり→開発工数見積もりの順で行っている企業が大半である ・他方で開発規模見積もりを行っていない企業が少数である 	全体的にレベルが高い企業が多い
リーダーのプロジェクトの経験や、持っている知識の差の考慮	<ul style="list-style-type: none"> ・「リーダーとして類似のプロジェクトに携わった経験があるかを考慮している」が多い ・「リーダーとして携わったプロジェクトの経験に加え、リーダーとしてのマネジメント能力を評価し、考慮している」と回答した企業も多い。 	全体的にばらつきがあり、考慮のレベルが低い企業が多い
類似のプロジェクトが無く、特徴が異なるために予測することが困難な状況の考慮	<ul style="list-style-type: none"> ・「特徴が大きく異なるプロジェクトのみ考慮している」、「特徴の相違の大きさを段階評価し、考慮している」、「過去のデータをもとに、特徴の相違が工数に与える影響を定性的に評価し、考慮している」が大半である。 ・「全く考慮していない」と回答した企業は少ない。 ・「過去のデータをもとに、特徴の相違が工数に与える影響を定量的に評価し、考慮している」と回答した企業も少ない。 	全体的にばらつきがあり、考慮のレベルが低い企業が多い
ソフトウェア開発の曖昧になっている部分の考慮	<ul style="list-style-type: none"> ・「開発するソフトウェアの中で特に曖昧な部分を明確にし、その曖昧さを評価・考慮している」が多い。 ・「全体的な曖昧さを段階評価し、考慮している」と回答した企業も多い。 ・「全く考慮していない」、「ソフトウェアを構成する各部分について曖昧さを評点付けし、考慮している」と回答した企業は無い。 	ばらつきがあるが、全体的に考慮のレベルが高い企業が多い
開発が始まった後に顧客から新たな要求が追加される可能性の考慮	<ul style="list-style-type: none"> ・「ソフトウェアの種類によって顧客から新たな要求が追加される可能性を評価し、考慮している」、「過去のデータに基づいて、新たな要求が追加される可能性の高い顧客かどうかを評価し、考慮している」が大半である。 ・「全く考慮していない」、「新規のソフトウェアかどうかのみを考慮している」と回答した企業はほとんど無い。 	ばらつきがあるが、全体的に考慮のレベルが高い企業が多い

4.6 の結果に基づけば、上記の2つの原因についての取り組みのレベルをあげる場合には、特に次の点に注意するのがよいと考えられる。

(1) リーダーのプロジェクトの経験や知識の差を考慮する場合には、能力を定量的に評

働できない、リーダーの経験・知識の差が多種多様である、経験・知識が同じでも能力が異なるなどの難しさを克服する必要がある。また、能力の差を考慮することに対する反対がある、必要な能力を持った人が確保できないなども状況を複雑にしている。したがって、経験・知識や能力を評価する工夫をするとともに、教育研修を行うのがよい。また、リーダーの経験・知識の差を補う意味で、途中でレビューを行う、支援者を付ける、情報を共有しメンバーがサポートするなども大切である。さらには、予備費を持つようにすることも考える必要がある。

- (2) 類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮する場合には、未経験のものを見積もる手法がない、人の経験に大きく依存する、必要な余裕度が判断できない、内容を詳細化できない、見積もりが困難な状況を想定できないなどの難しさを克服する必要がある。したがって、見積もり手法を教育するだけでなく、リスクを明確にしてコンティジェンシープランを作成する、見積もりを大目に行う、プロジェクト期間を長めにとる、さらには仕事を受けないことも考えるのがよい。また、プロジェクトに関する情報を広く収集し、共有することも大切である。

第6章 結論と今後の課題

本研究では、ソフトウェア開発を行っている企業に対して郵送調査を行い、開発工数見積もり実態を調査することで、工数見積もり誤りを減らすために今後どのような取り組みを行う必要があるのかを明らかにすることを試みた。結果として、「リーダーのプロジェクトの経験や知識の差を考慮」「類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況を考慮」の取り組みを行う必要があることがわかった。

このうち、リーダーのプロジェクトの経験や知識の差の考慮については、経験・知識や能力を評価する工夫をするとともに、教育研修を行うのがよい。また、リーダーの経験・知識の差を補う意味で、途中でレビューを行う、支援者を付ける、情報を共有しメンバーがサポートするなども大切である。他方、類似のプロジェクトが無く、特徴が異なるため予見することが困難な状況の考慮については、リスクを明確にしてコンティンジェンスプランを作成するとともに、見積もりを大目に行う、プロジェクト期間を長めにとることを考えるのがよい。また、プロジェクトに関する情報を広く収集し、共有することも大切である。

今後の課題としては、工数見積もりの原因に取り組む困難さおよびその克服方法についてより詳細な検討を行うこと、様々な克服方法を実際に適用しその有効性や限界を明らかにすること、IT 業界の他の分野について同様の調査を行い、分野の違いを考慮に入れた工数見積もりの方法を研究することなどが残されている。

参考文献

- [1] 山田茂・高橋宗雄 (1933) : 「ソフトウェアマネジメントモデル入門—ソフトウェア品質の可視化と評価法」、共立出版。
- [2] 保田勝道(1995) : 「ソフトウェア品質保証の考え方と実際」、日科技連出版社。
- [3] ロジャー S プレスマン(2000) : 「製品とプロセス/ソフトウェアプロジェクトの管理」、日科技連出版社。
- [4] 野中誠・小池利和・小室睦(2012) : 「データ指向のソフトウェア品質マネジメント—メトリクス分析による事実にもとづく管理の実践」、日科技連出版社
- [5] 神長裕明・郷健太郎・杉浦茂樹・高橋正和・藤田茂樹・渡辺喜道 (2012) : 「未来へつなぐデジタルシリーズ 13—ソフトウェア工学の基礎」、共立出版

付録

ソフトウェア開発における 工数見積もり誤りの防止に関する調査

ソフトウェア開発における工数見積もり誤りの防止に関する研究

調査の目的

現代の社会において、IT (Information Technology) は人々の生活や様々な組織の活動に欠かせないものとなっています。このため、多くの IT 企業が顧客の要求に応じて必要なソフトウェアの開発を行っています。しかし、各々のソフトウェアの開発に必要な工数 (人・月) は様々な要因の影響を受けるため、その見積もりは容易ではありません。結果として、予想以上に工数がかかり、十分なレビューやテストを行えないまま出荷することによる市場トラブルの発生、開発コストの増大による予算超過、開発担当者の勤務時間外労働の増加などの問題を引き起こしています。

ソフトウェア開発工数の見積もりについては、従来からも多くの研究・提案が行われてきていますが、開発という業務の性質上、考慮すべき要因が多く、またそれらが人の行動・能力に関連するため、十分な成果が得られていないのが現状です。

このような現状をふまえ、本調査では、工数見積もりにおける誤り (見積もりと実績との差が大きい事象) を減らすために IT 企業が行っている取り組みについて、どのような要因をどのように考慮しているのか、どのような要因の考慮が難しいのかという視点から調査・分析を行い、工数見積もり誤りを減らすために今後どのような取り組みを行う必要があるのかを明らかにしたいと考えています。

回答に当たっての注意

- 本調査は、I. 職場の概要、II. 適用している工数見積もりの方法、III. 工数見積もり誤りとその原因、IV. 工数見積もり誤りを減らす取り組み、の4つのパートから構成されています。
- 本調査は、ISO9001 適合組織で産業分類が情報技術の中から無作為に選んだ全 260 企業・組織にお送りしています。貴社・貴組織においてソフトウェア開発を行っており、工数見積もり誤りの防止・低減が重要な課題となっている職場を一つ選んでご回答ください。なお、回答いただくのは当該職場における工数見積もりの状況が分かる方であればどなたでも構いません。
- 選択式の質問と自由書式の質問とありますが、出来るだけ具体的にお答え下さい。なお、答えられない・答えにくい質問については空欄で結構です。
- 本調査用紙の電子ファイル (Microsoft Word) を下記のホームページからダウンロードできます。必要に応じてご活用ください

<http://www.indsys.chuo-u.ac.jp/~nakajo/man-month.html>

- ご回答いただきました調査用紙につきましては、2016年2月10日(水)までにE-mailアドレス: a12.ga7b@g.chuo-u.ac.jp (1は数字、-はマイナス) 宛てに電子ファイルでお送りください。下記担当者宛てに郵送いただいても構いません。
- 調査結果を取りまとめた報告書は2016年3月下旬にお送りする予定です。報告書の送付を希望される場合は下記に送り先をご記入ください。

住所又は e-mail アドレス :

会社名 :

所属 :

氏名 :

- 本調査に関してご不明な点、ご質問がありましたら下記までご連絡ください。

本件担当者 : 中央大学理工学部経営システム工学科 川平陽明

〒112-8551 東京都文京区春日 1-13-27

Tel 03-3817-1933 Fax 03-3817-1943

Tel 080-5072-8701(直通) E-mail a12.ga7b@g.chuo-u.ac.jp

- I 回答いただく「ソフトウェア開発を行っている職場」の概要についてお聞きします。
ソフトウェア開発を行っており、工数見積もり誤りの防止・低減が重要な課題となっている職場を一つ選んでください。

- (1) 当該職場が開発しているソフトウェアの種類として、最も当てはまると思うものを下記の選択肢より選び、番号でお答えください。なお、「7.その他」を選んだ場合には、開発しているソフトウェアの種類を簡潔に付記してください。

開発しているソフトウェア	1. ファームウェアなどの組込ソフトウェア 2. OSなどのシステムソフトウェア 3. 業務用アプリケーションソフトウェア 4. 工業用アプリケーションソフトウェア 5. パッケージソフトウェア 6. コンテンツソフトウェア 7. その他	
--------------	---	--

- (2) 行っている開発業務（要件定義、外部設計、内部設計、コーディング、レビュー、テスト、導入支援、保守等）の内容を簡潔にお書きください。

開発業務内容	
--------	--

- (3) 1つの開発プロジェクトを担当する人の数はおおよそ何人ですか。概数でお答えください。なお、人数に幅がある場合は〇〇～〇〇という形で記入してください。

1つのプロジェクトを担当する人の人数（概数）	人
------------------------	---

- (4) 1つのプロジェクトの開始から完了までの期間はどのくらいですか。概数（月）でお答えください。なお、期間に幅がある場合は〇〇～〇〇という形で記入してください。

1つのプロジェクトの開始から完了までの期間（概数）	月
---------------------------	---

Ⅱ 回答頂いている職場において行っている工数見積りのプロセス／方法についてお聞きします。

(1) 工数見積りをどのようなプロセス／方法で行っていますか。

① 工数見積りのプロセスとして最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「5.その他」を選んだ場合には、行っているプロセスについて簡潔に付記してください。

見積りのプロセス	<ol style="list-style-type: none"> 1. 作業の分解 (WBS 等) → 開発規模見積り → 開発工数見積りの順に行っている 2. 作業の分解 (WBS 等) を行った後、開発規模見積りを行わず、開発工数見積りを行っている 3. 作業の分解 (WBS 等) を行わず、開発規模見積りを行った後、開発工数見積りを行っている 4. 作業の分解 (WBS 等) や開発規模見積りを行わず、開発工数見積りのみを行っている 5. その他 	
----------	---	--

注) ここで言う「作業の分解」とは、開発規模見積りと開発工数見積りを行う前段階で作業を分解・階層化することです。また、「開発規模見積り」とは、開発工数見積りを行うために、最終的にまたは作業ごとに作成しなければならない成果物の量（ソースコードの行数や仕様書のページ数）を見積もることです。

② 「作業の分解」を行う際に使用している方法 (WBS 等) について簡潔にお答えください。なお、作業の分解を行っていない場合には回答していただく必要はありません。

作業の分解の方法	
----------	--

③ 「開発規模の見積り」を行う際に使用している方法 (類似のプロジェクトの値を修正する、LOC 法の使用、ファンクションポイント法の使用等) について簡潔にお答えください。なお、開発規模の見積りを行っていない場合には回答していただく必要はありません。

開発規模見積りの方法	
------------	--

④ 「開発工数見積り」を行う際に使用している方法 (類似のプロジェクトの値を修正する、COCOMO モデルの使用など) について簡潔にお答えください。

開発工数見積りの方法	
------------	--

(2) 工数見積り誤りが発生する要因をどのように考慮していますか。

① 見積りを行う際に、個人個人の能力の差を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

個人個人の能力の差	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 一部の特別な人のみを考慮している 3. 経験年数や職位などを考慮している 4. 個人個人の能力を評価し、考慮しているが、能力の定期的な評価・更新は行っていない 5. 個人個人の能力を定期的 (年 1 回等) に評価し、その結果を考慮している 6. その他 	
-----------	--	--

②リーダーのプロジェクトの経験や、持っている知識の差を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

リーダーのプロジェクトの経験や、持っている知識の差	<ol style="list-style-type: none"> 1. 全く考慮していない 2. リーダーとしてプロジェクトに携わった経験年数を考慮している 3. リーダーとして類似のプロジェクトに携わった経験があるかを考慮している 4. リーダーとして携わったプロジェクトの成果（成功や失敗等）を考慮している 5. リーダーとして携わったプロジェクトの経験に加え、リーダーとしてのマネジメント能力を評価し、考慮している 6. その他 	
---------------------------	---	--

③見積もりを行う際に、メンバーのプロジェクトの経験や、持っている知識の差を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

メンバーのプロジェクトの経験や、持っている知識の差	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 資格に加え、メンバーとしてプロジェクトに携わった経験年数を考慮している 3. メンバーとして類似のプロジェクトに携わった経験があるかを考慮している 4. メンバーとしてプロジェクトで残した成果を考慮している 5. 取得している資格やいままでに受けた教育を考慮している 6. その他 	
---------------------------	---	--

④見積もりを行う際に、プロジェクトを一緒に行う企業の間でコミュニケーションが上手く取れるかどうか、上手く取れないことによる遅れを考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

プロジェクトを一緒に行う企業の間でのコミュニケーションによる遅れ	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 過去に一緒にプロジェクトを行ったことがあるのみを考慮している 3. 過去にプロジェクトを一緒に行った時の結果（コミュニケーションの問題があったかどうか）を考慮している 4. 企業ごとの特性（規模、繁忙さなど）やマネジメント能力を評価し、考慮している 5. 相手企業との契約形態やコミュニケーション方法の違い（対面による相談が容易かどうか）を考慮している 6. その他 	
----------------------------------	--	--

⑤見積もりを行う際に、リーダーとメンバー間のコミュニケーションが上手く取れるかどうか、上手く取れないことによる遅れは考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

リーダーとメンバーの間でのコミュニケーションによる遅れ	<ol style="list-style-type: none"> 1. 全く考慮していない 2. プロジェクトの規模のみを考慮している 3. リーダーとメンバーが過去に一緒にプロジェクトを行ったことがあるかを考慮している 4. 過去にプロジェクトを一緒に行った時の結果（コミュニケーションの問題があったかどうか）を考慮している 5. リーダーとメンバーのコミュニケーション方法（対面による相談が可能かどうかなど）によって考慮している 6. その他 	
-----------------------------	---	--

⑥見積もりを行う際に、他のプロジェクトの兼ね合いでプロジェクトに必要な人を集めることが困難な状況を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況	<ol style="list-style-type: none"> 1. 全く考慮していない 2. プロジェクトの規模・期間（必要となる人数の多さなど）のみを考慮している 3. プロジェクトの種類（特別な能力を持つ人の必要性など）によって人を集める困難さを評価し、考慮している 4. プロジェクトの途中段階でメンバーを増減する可能性を評価し、考慮している 5. 自プロジェクトの状況に加えて、同時に行われる他のプロジェクトの状況を考慮している 6. その他 	
---------------------------------------	---	--

⑦見積もりを行う際に、類似のプロジェクトが無く、特徴が異なるために予見することが困難である状況を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特徴が大きく異なるプロジェクトのみ考慮している 3. 特徴の相違の大きさを段階評価し、考慮している 4. 過去のデータをもとに、特徴の相違が工数に与える影響を定性的に評価し、考慮している 5. 過去のデータをもとに、特徴の相違が工数に与える影響を定量的に評価し、考慮している 6. その他 	
------------------------------------	---	--

⑧標準となる開発プロセスが定まっていなかったり、定まっても適用できなかったりすると開発工数が大きくばらつきます。見積もりを行う際に、標準となる開発プロセスが適用できるかどうかを考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

標準となる開発プロセスの適用性	<ol style="list-style-type: none"> 1. 全く考慮していない（標準を定めていない場合を含む） 2. 標準の開発プロセスが適用できるかのみを考慮している 3. 開発プロセスの中で、標準が適用できない部分の割合を明確にし、考慮している 4. 開発プロセスの中で、標準と最も異なる部分を明確にし、評価・考慮している 5. 開発プロセスのステップ毎に標準が適用できる度合いを評点付けし、考慮している 6. その他 	
-----------------	---	--

⑨ソフトウェアは目に見えにくいものです。見積もりを行う際に、開発するソフトウェアについて曖昧になっている部分を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

開発するソフトウェアについて曖昧になっている部分	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特別に曖昧な場合のみ考慮している 3. 全体的な曖昧さを段階評価し、考慮している 4. 開発するソフトウェアの中で特に曖昧な部分を明確にし、その曖昧さを評価・考慮している 5. ソフトウェアを構成する各部分について曖昧さを評点付けし、考慮している 6. その他 	
--------------------------	---	--

⑩顧客が要求を具体化できていないために、詳細に見積もることが困難な場合があります。見積もりを行う際に、顧客が要求を具体化できていない部分を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているかについて簡潔に付記してください。

顧客が要求を具体化できていない部分	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特に具体化できていない場合のみを考慮している 3. 顧客が要求を具体化できていない状況を全体的に段階評価し、考慮している 4. 顧客が最も具体化できていない要求を明確にし、評価・考慮している 5. 顧客の要求をいくつかに分けた上で、各々の具体化できていない度合いを評点付けし、考慮している 6. その他 	
-------------------	--	--

⑪開発が始まった後に顧客から新たな要求が追加されることがあります。見積もりを行う際に、このようなことが起こる可能性を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているか簡潔に付記してください。

開発が始まった後に顧客から新たな要求が追加される可能性	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 新規のソフトウェアかどうかのみを考慮している 3. ソフトウェアの種類によって顧客から新たな要求が追加される可能性を評価し、考慮している 4. 過去のデータに基づいて、新たな要求が追加される可能性の高い顧客かどうかを評価し、考慮している 5. 要求を顧客から得たプロセスによって新たな要求が追加される可能性を評価し、考慮している 6. その他 	
-----------------------------	--	--

⑫過去に使用したソフトウェアやモデルを改良する計画をしていたが、開発が進むにつれて、使おうとしていたソフトウェアやモデルを使うことができないことがわかります。見積もりを行う際に、再利用しようとしていたソフトウェアやモデルが活用できない可能性を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「5.その他」を選んだ場合には、どのように考慮しているかについて簡潔に付記してください。

再利用しようとしていたソフトウェアやモデルが活用できない可能性	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特別の事情がある場合のみを考慮している 3. 再利用しようとしているソフトウェアやモデルが活用できない可能性を段階評価し、考慮している 4. 再利用しようとしているソフトウェアやモデルのうち、活用できない可能性が最も高いものを明確にし、評価・考慮している 5. 再利用しようとしているソフトウェアやモデルごとに活用できない可能性を評点付けし、考慮している 6. その他 	
---------------------------------	---	--

⑬開発で使用するプログラム言語には、簡単に使用できるものとそうでないものがあります。見積もりを行う際に、使用するプログラム言語の難しさを考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているかについて簡潔に付記してください。

開発で使用するプログラム言語の難しさ	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特に難しいプログラム言語のみ考慮している 3. プログラム言語の難しさを段階評価し、考慮している 4. 過去のデータに基づいて、プログラム言語の難しさを評点付けし、考慮している 5. プログラム言語と開発環境などとの相性を考慮している 6. その他 	
--------------------	---	--

⑭開発に使用するハードウェアに制限があり、自由に使えない場合があります。見積もりを行う際に、開発に使用するハードウェアの制限を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているかについて簡潔に付記してください。

開発で使用するハードウェアの制限	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 開発で使用するハードウェアの制限が特に厳しい場合のみを考慮している 3. 開発で使用するハードウェアの制限の厳しさを段階評価し、考慮している 4. 開発で使用するハードウェアのうち、最も制限が厳しいものを明確にし、評価・考慮している 5. 開発で使用する各々のハードウェアについて、制限の厳しさを評点付けし、考慮している 6. その他 	
------------------	--	--

⑮顧客と開発者側とで必要なコストに対する認識が異なり、開発側が必要な工数を適切に見積もれない場合があります。見積もりを行う際に、開発にかかるコストに対する顧客と開発者側との認識の相違を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているかについて簡潔に付記してください。

開発にかかるコストに対する顧客と開発者側との認識の相違	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特に大きな認識の相違がある場合のみ考慮している 3. 認識の相違の大きさを段階評価し、考慮している 4. ソフトウェアまたは開発プロセスを要素に分け、その中で、開発にかかるコストに対する認識の相違が最も大きい部分を明確にし、評価・考慮している 5. ソフトウェアまたは開発プロセスを要素に分け、要素毎に、開発にかかるコストに対する認識の相違を評価し、考慮している 6. その他 	
-----------------------------	---	--

⑯開発予算の関係で開発にかけられる人数が限られているために、工数を少なめに見積もらざるを得ないことがあります。見積もりを行う際に、開発予算の関係によって開発にかけられる人数が限られる状況を考慮していますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。なお、「6.その他」を選んだ場合には、どのように考慮しているかについて簡潔に付記してください。

開発予算の関係で開発にかけられる人数が限られる状況	<ol style="list-style-type: none"> 1. 全く考慮していない 2. 特に開発予算が限られている場合のみを考慮している 3. 開発予算が限られている状況を段階評価し、考慮している 4. 開発予算の中で最も厳しい項目を明確にし、評価・考慮している 5. 開発予算の各項目の厳しさを評点付けし、考慮している 6. その他 	
---------------------------	--	--

⑰上記の①～⑱以外で工数見積りにおいて考慮している要因があれば空欄にお書きください。

その他の要因	
--------	--

Ⅲ 工数見積もり誤りの現状およびそれらを引き起こしている要因についてお聞きします。

(1) 工数見積もり誤り（実際の工数が見積もった値の±20%以上を超える事象）が発生するプロジェクトの割合として、最も当てはまるものを下記の選択肢より選び、番号でお答えください。正確なものでなく、おおよその値で構いません。

工数見積もり誤り発生割合	1. 10%未満 2. 10%以上20%未満 3. 20%以上30%未満 4. 30%以上40%未満 5. 40%以上50%未満 6. 50%以上60%未満 7. 70%以上80%未満 8. 80%以上90%未満 9. 90%以上	
--------------	---	--

(2) 工数見積もり誤りが起きる原因について、見積もりプロセスの視点からお聞きします。

① 工数見積もりのプロセスのうち、「作業の分解」を行うプロセスが誤りの原因になっていると思いますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。

作業の分解	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
-------	--	--

② 工数見積もりのプロセスのうち、「開発規模見積もり」を行うプロセスが誤りの原因になっていると思いますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。

開発規模見積もり	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
----------	--	--

③ 工数見積もりのプロセスのうち、「開発工数見積もり」を行うプロセスが誤りの原因になっていると思いますか。最も当てはまると思うものを下記の選択肢から選び、番号でお答えください。

開発工数見積もり	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
----------	--	--

(3)工数見積もり誤りが発生する原因について、見積もりにおいて考慮すべき要因の視点からお聞きします。

①見積もりを行う際に、個人個人の能力の差を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

個人個人の能力の差を考慮していないこと	<ol style="list-style-type: none"> 1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う 	
---------------------	--	--

②見積もりを行う際に、リーダーのプロジェクトの経験や、持っている知識の差を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

リーダーのプロジェクトの経験や、知識の差を考慮していないこと	<ol style="list-style-type: none"> 1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う 	
--------------------------------	--	--

③見積もりを行う際に、メンバーのプロジェクトの経験や、持っている知識の差を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

メンバーのプロジェクトの経験や、知識の差を考慮していないこと	<ol style="list-style-type: none"> 1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う 	
--------------------------------	--	--

④プロジェクトを一緒に行う企業の間でのコミュニケーションによる遅れを考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

プロジェクトを一緒に行う企業の間でのコミュニケーションによる遅れを考慮していないこと	<ol style="list-style-type: none"> 1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う 	
--	--	--

⑤開発に関わるリーダーとメンバーの間でのコミュニケーションによる遅れを考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

リーダーとメンバーの間でのコミュニケーションによる遅れを考慮していないこと	<ol style="list-style-type: none"> 1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う 	
---------------------------------------	--	--

⑥他のプロジェクトの兼ね合いでプロジェクトに関わる人を集めることが困難な状況を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

他のプロジェクトとの兼ね合いで人を集めることが困難な状況を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
--	--	--

⑦類似のプロジェクトが無く、特徴が異なるために予見することが困難な状況を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

類似のプロジェクトが無く、予見することが困難な状況を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
-------------------------------------	--	--

⑧標準となる開発プロセスの適用性を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

標準となる開発プロセスの適用性を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
---------------------------	--	--

⑨開発するソフトウェアについて曖昧になっている部分を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

開発するソフトウェアについて曖昧になっている部分を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
------------------------------------	--	--

⑩顧客が要求を具体化できていない部分を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

顧客が要求を具体化できていない部分を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
-----------------------------	--	--

⑪開発が始まった後に顧客から新たな要求が追加される可能性を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

開発が始まった後に顧客から新たな要求が追加される可能性を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
---------------------------------------	--	--

⑫再利しようとしていたソフトウェアやモデルが活用できない可能性を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

再利しようとしていたソフトウェアやモデルが活用できない可能性を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
--	--	--

⑬開発で使用するプログラム言語の難しさを考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

開発で使用するプログラム言語の難しさを考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
------------------------------	--	--

⑭開発で使用するハードウェアの制限を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

開発で使用するハードウェアの制限を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
----------------------------	--	--

⑮開発にかかるコストに対する顧客と開発者側との認識の相違を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

開発にかかるコストに対する顧客と開発者側との認識の相違を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
---------------------------------------	--	--

⑯開発予算の関係で開発にかけられる人数が限られる状況を考慮していないことが原因になっていると思いますか。最も当てはまると思う番号を選んでください。

開発予算の関係で開発にかけられる人数が限られる状況を考慮していないこと	1. 原因になっているとは思わない 2. どちらかといえば原因になっているとは思わない 3. どちらともいえない 4. どちらかといえば原因になっていると思う 5. 原因になっていると思う	
-------------------------------------	--	--

⑰上記の①～⑯以外で、当該の要因を考慮していないことが工数見積り誤りの原因になっていると思われるものがあれば空欄にお書きください。

考慮していないことが原因となっていると思われるその他の要因	
-------------------------------	--

Ⅳ 工数見積もり誤りを減らす取り組みの難しさとその克服についてお聞きします。なお、Ⅲにおいて「5. 原因となっていると思う」または「4. どちらかと言えば原因となっていると思う」と回答された項目のみ、回答してください。

(1) 工数見積もり誤りの原因を改善する難しさとその克服のために実践している／しようとしていることについて、見積もりプロセスの視点からお聞きします。

①「作業の分解」を行うプロセスを改善する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

②「開発規模見積もり」を行うプロセスを改善する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

③「開発工数見積もり」を行うプロセスを改善する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

(2) 工数見積もり誤りの原因を改善する取り組みの難しさとその克服のために実践している／しようとしていることについて、見積もりにおいて考慮すべき要因の視点からお聞きします。

①見積もりにおいて「個人個人の能力の差」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ②見積もりにおいて「リーダーのプロジェクトの経験や、知識の差」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ③見積もりにおいて「メンバーのプロジェクトの経験、知識の差」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ④見積もりにおいて「プロジェクトを一緒に行う企業の間でのコミュニケーションによる遅れ」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑤見積もりにおいて「リーダーとメンバーの間でのコミュニケーションによる遅れ」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑥見積もりにおいて「他のプロジェクトとの兼ね合いで人を集めることが困難な状況」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑦見積もりにおいて「類似のプロジェクトが無く、特徴が異なり予見することが困難な状況」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑧見積もりにおいて「標準となる開発プロセスの適用性」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑨見積もりにおいて「開発するソフトウェアについて曖昧になっている部分」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑩見積もりにおいて「顧客が要求を具体化できていない部分」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑪見積もりにおいて「開発が始まった後に顧客から新たな要求が追加される可能性」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑫見積もりにおいて「再利用しようとしていたソフトウェアやモデルが活用できない可能性」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑬見積もりにおいて「開発で使用するプログラム言語の難しさ」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑭見積もりにおいて「開発で使用するハードウェアの制限」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑮見積もりにおいて「開発にかかるコストに対する顧客と開発者側との認識の相違」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑯見積もりにおいて「開発予算の関係で開発にかけられる人数が限られる状況」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

- ⑰見積もりにおいて「考慮していないことが原因となっていると思われるその他の要因」を考慮する難しさは何ですか。また、それを克服するためにどのようなことを実践していますか／しようとしていますか。

難しさ	
克服のために実践している／しようとしていること	

アンケートは以上です。ご協力ありがとうございます