

筑波大学大学院博士課程

システム情報工学研究科特定課題研究報告書

テニススクール経営改革のための IT ソリューション提供
ーユーザビリティの向上とリスクマネジメントの実践ー

杜婧雨

修士（工学）

（コンピュータサイエンス専攻）

指導教員 三末和男

2013年3月

概要

本書は、2012 年度「研究開発プロジェクト」科目に関する特定課題研究報告書である。筆者が所属する研究開発プロジェクトは、つくば市内のあるテニススクールを顧客として、経営改革にむけた戦略案を作成した。それに基づいて本年度は、内部基盤の確立のため従業員を対象とした業務プロセス改善と IT ソリューションを提案した。レッスン管理と勤務評価を目的として、アカウント管理、スケジュール管理、代行管理、勤務評価管理の機能を開発した。また、マニュアルを作成し、講習会を行い、システム運用に支援した。さらに、業務プロセス改善とシステムの評価指標を測定し、利用者向けのアンケートを実施し、プロジェクトの成果を検証した。

筆者は本プロジェクトの中で、「ユーザビリティの向上」と「リスクマネジメントの実践」を担当した。「ユーザビリティの向上」では、ユーザビリティを定期的に評価し、継続的に改善した。「リスクマネジメントの実践」では、PMBOK のリスクマネジメントフローにしたがったマネジメントプロセスを推進し、その中から得られた知見を提言した。

本報告書では筆者が捉えた本プロジェクトの内容、及び筆者が担った役割とその成果について報告する。

目次

第1章	はじめに	1
1.1	背景	1
1.2	目的	1
1.3	基本方針	1
1.4	顧客情報	1
1.5	チーム体制	2
1.6	スケジュール	3
第2章	各フェーズの実施状況	5
2.1	戦略立案フェーズ	5
2.1.1	目的	5
2.1.2	方針	5
2.1.3	流れ	5
2.1.4	担当部分と振り返り	13
2.2	要件定義フェーズ	14
2.2.1	目的	14
2.2.2	方針	14
2.2.3	流れ	14
2.2.4	担当部分と振り返り	19
2.3	開発フェーズ	20
2.3.1	目的	20
2.3.2	方針	20
2.3.3	流れ	21
2.3.4	開発の実績	25
2.3.5	担当部分と振り返り	27
2.4	評価フェーズ	29
2.4.1	目的	29
2.4.2	方針	29
2.4.3	評価指標の測定結果	31
2.4.4	運用マニュアルアンケートの集計結果と分析	33
2.4.5	使い勝手アンケートの集計結果と分析	34
第3章	ユーザビリティの向上	38
3.1	ユーザビリティの定義	38
3.2	ユーザビリティの評価手法	39
3.2.1	ユーザビリティインスペクション	40
3.2.2	ユーザビリティテスト	42
3.3	考察	44
第4章	リスクマネジメントの実践	45
4.1	リスクマネジメントの目的	45
4.2	リスクマネジメント計画	45
4.3	リスク特定	46

4.4	定性的リスク分析	47
4.5	リスク対応計画	49
4.6	リスク監視とコントロール	50
4.7	考察	52
第5章	振り返り	54
5.1	プロジェクトの振り返り	54
5.2	チームの振り返り	56
5.3	個人の振り返り	57
5.4	終わりに	58
謝辞		59
参考文献		60
付録		61

図目次

図 1-1	本プロジェクトのスケジュール.....	3
図 2-1	マインドマップ（ビジョンの確認）.....	6
図 2-2	外部環境調査.....	7
図 2-3	マインドマップ（内部環境調査）.....	8
図 2-4	内部環境調査.....	8
図 2-5	生徒年齢分布図.....	9
図 2-6	SWOT 分析の結果.....	10
図 2-7	クロス SWOT 分析の結果.....	11
図 2-8	経営改革ロードマップ.....	11
図 2-9	BSC による評価指標.....	12
図 2-10	システム概要図.....	16
図 2-11	業務フロー図.....	17
図 2-12	紙プロトタイプ.....	19
図 2-13	画面イメージ.....	20
図 2-14	ユーザストーリーのタスクの洗い出し.....	23
図 2-15	KPT シート.....	23
図 2-16	システム構成概念図.....	24
図 2-17	バーンアップチャート.....	25
図 2-18	評価フェーズのスケジュール.....	30
図 2-19	7つの評価軸の得点.....	35
図 2-20	使い勝手アンケートの分析 1.....	36
図 2-21	使い勝手アンケートの分析 2.....	36
図 3-1	管理者向けの画面遷移図.....	41
図 3-2	コーチ向けの画面遷移図.....	41
図 3-3	ユーザエクスペリエンスのダイアグラム.....	42
図 4-1	リスクマネジメントフロー.....	45
図 4-2	リスクの影響度と発生確率のマトリックス.....	46
図 4-3	TTW を用いたリスク特定シート.....	46
図 4-4	リスクの特性要因図.....	47
図 4-5	各イテレーションのリスクマネジメント作業フロー.....	50
図 4-6	リスクスコア監視グラフ.....	51
図 4-7	要求変更時のマネジメントプロセス.....	53
図 5-1	顧客ミーティング時間の統計.....	54
図 5-2	easy step で顧客と共に作成したマインドマップ.....	55
図 5-3	Cacoo で共同作業の成果図.....	56
図 5-4	タックマンモデル.....	57

表目次

表 1.1	T1 の基本情報.....	2
表 1.2	役割分担表 (RAM)	2
表 2.1	T1 の企業理念, 経営ビジョン, ミッション.....	6
表 2.2	責任範囲と支援する範囲の明確化.....	15
表 2.3	ユーザストーリー内訳.....	18
表 2.4	開発環境のソフトウェア構成.....	21
表 2.5	開発で利用したツール.....	22
表 2.6	バーンアップチャートで用いる用語の定義	25
表 2.7	イテレーション 0 で終了したユーザストーリー	26
表 2.8	イテレーション 1 で終了したユーザストーリー	26
表 2.9	イテレーション 2 で終了したユーザストーリー	26
表 2.10	イテレーション 3 で終了したユーザストーリー	27
表 2.11	イテレーション 4 で終了したユーザストーリー	27
表 2.12	評価フェーズでの測定項目.....	29
表 2.13	IT-KPI の測定結果.....	31
表 2.14	評価フェーズのトラブルの詳細.....	33
表 2.15	KPI の測定結果.....	33
表 2.16	運用マニュアルに関するアンケートの結果.....	33
表 2.17	使い勝手アンケートの集計結果 1.....	34
表 2.18	使い勝手アンケートの集計結果 2.....	35
表 3.1	本プロジェクトでユーザビリティの定義.....	38
表 3.2	ユーザビリティ定量的評価内容.....	39
表 3.3	本プロジェクトにおける評価手法の比較.....	40
表 3.4	ユーザビリティチェックシート.....	40
表 3.5	発話と操作内容分析表.....	43
表 3.6	発話と操作内容の分析結果.....	43
表 4.1	リスク分析シート.....	48
表 4.2	リスクのレベルと対応策の関係.....	49
表 4.3	リスク登録簿 1.....	49
表 4.4	リスク登録簿 2.....	50

第1章 はじめに

1.1 背景

2011 年度の授業科目「PBL 型システム開発」で、つくば市内にある T1 インドアテニスクール（以下、T1 と呼ぶ）を顧客として、「テニススクールレッスン予約システム」というプロジェクトが立ち上げられた。しかし、提案まで実施したが、学生チームの解散により終了した。

後日、PBL 型システム開発の担当教員である山戸昭三教授が T1 を訪問した所、最後まで改革の提案をして欲しいとの要望を受けた。2012 年度の研究開発プロジェクトとして、再度仕切り直した。最初のミーティングの際に、顧客から T1 の真のニーズについて考える機会が欲しいという要望があった。そのため、経営環境を調査・確認することで顧客ニーズの再確認から始め、ニーズに合致したソリューションの提供を目指すこととした。

1.2 目的

本プロジェクトの目的は、経営者と共に検討することによって、現状と課題を共有し、顧客の IT ガバナンスのレベルに応じて、適切な IT ソリューションを提供し、経営基盤を確立することである。顧客に戦略の立案方法や推進方法を理解していただくことも含めて経営改革と考えている。学生チームは本プロジェクトを遂行することで、経営戦略の立案という超上流工程から、要件定義をし、システムの開発、納品、評価まで一連の活動を実践し、経営戦略立案と IT ソリューション提供の実践力を身につけることができる。

1.3 基本方針

本プロジェクトでは、顧客が戦略立案や IT 調達の経験不足から想定されるリスクを軽減するために、顧客と密なコミュニケーションを取ることを基本方針とした。戦略立案、要件定義フェーズでは 1 週間に 1 回程度、開発フェーズでは 2 週間に 1 回程度のミーティングを行い、顧客とプロジェクトの相互理解を促進することで、顧客の真の要求に適合した満足度の高い IT ソリューションの提供を目指した。

我々は、顧客とのミーティング以外にも、顧客の業務プロセスの理解のために、現場で働く人たちとコミュニケーションし、現場の人々の動きを観察した。

また、顧客への積極的な情報共有のため、Dropbox を用いて、議事録や合意した資料などを共有した。

1.4 顧客情報

T1 の基本情報を表 1.1 に示す。2003 年 4 月につくば市梅園地区に開業した。インドアコート 1.5 面、アウトドアコート 2 面を持ち、コーチが 17 人、生徒が 300 人程度の中小規模テ

ニススクールである。生徒は小学生から高齢者まで幅広く多岐にわたる。現在 10 クラスを持ち、初心者から熟練者まで指導を行なっている。

単にテニススキルの向上というだけでなく、「楽しいテニスレッスン」を目指しており、テニスを楽しむ場を提供することで地域住民に貢献している。学生コーチによるアットホームな雰囲気のレッスンを売りにしている。

表 1.1 T1 の基本情報

形態	特例有限会社	
所在地	つくば市梅園 2-17-8	
設立	2003 年 4 月	
経営陣	代表取締役 兼ヘッドコーチ	末満裕之 様
	マネージャ 兼事務員	末満ひろえ 様
従業員	コーチ	18 名
	事務員	2 名
生徒数	約 300 人	
クラス数	10 クラス	
インドアコート数	1.5 面	
アウトドアコート数	2 面	
その他設備	クラブハウス, 駐車場, 託児ルーム	

1.5 チーム体制

本プロジェクトは各フェーズの作業, 技術調査, マネジメントの三つの軸で分担し進めた。役割分担表(RAM 表)^[1]を表 1.2 に示す。我々のチームは, 対象とする IT ソリューションについてユーザ中心設計の観点で利用者満足度が重要であると考え, 表 1.2 の分担をして技術調査を行うこととした。またマネジメント分野については, 作業分担の理由により, 表 1.2 の分担とした。筆者は, ヒューマンインターフェース系の作業を中心に行った。

表 1.2 役割分担表 (RAM)

チームメンバ	各フェーズの作業	技術調査	マネジメント分野
白田良太	タスクボードを利用し, 対応可能のメンバが着手できる状態になったタスクにサインアップする	コーチ評価体制と管理機能	コミュニケーション
有田正信		テスト自動化	品質
永井達也		セキュリティ対策	進捗
井原淳平		開発環境・運用環境の構築	スコープ
杜婧雨		ユーザビリティの向上	リスク

1.6 スケジュール

プロジェクト期間は 2012 年 5 月～2013 年 1 月の 9 ヶ月間であった。プロジェクトスケジュールの初期計画，変更後計画と実績を図 1-1 に示す。大きく次の 4 フェーズでプロジェクトを進めた。

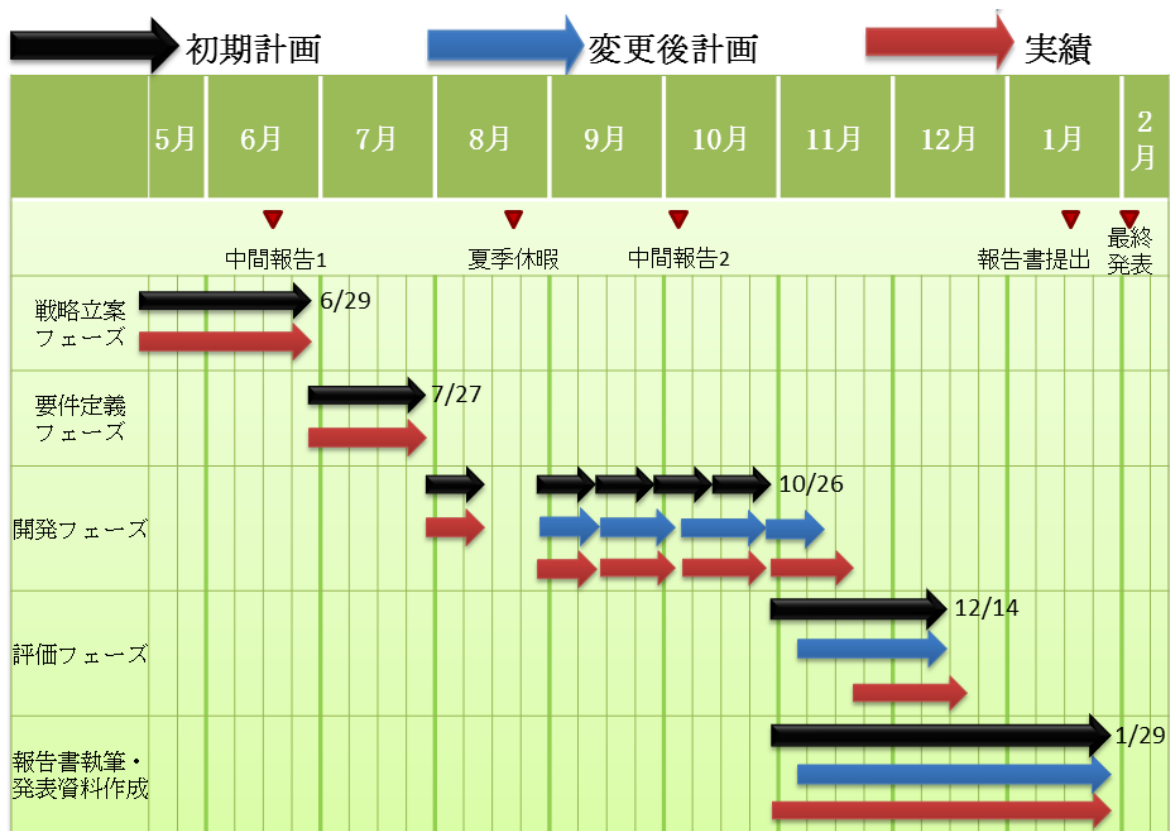


図 1-1 本プロジェクトのスケジュール

① 戦略立案フェーズ

戦略立案フェーズでは，市場動向や人口分布などの外部環境調査や内部環境調査としての経営陣やコーチへのヒアリングを行って情報収集した。その情報を元に SWOT 分析^[2]を用いた現状分析，クロス SWOT 分析を用いた戦略案(CSF)抽出，戦略決定を行った。また，バランス・スコアカード(BSC)^[3]を用いて経営改革を定量的に測定するための指標を抽出し，各指標に対し KGI と KPI の設定を行った。

② 要件定義フェーズ

経営戦略で定義された IT 戦略を元に，開発するシステムの要件を定義した。

要件定義フェーズでは，経営戦略を達成するために導入する IT ソリューションの要件の決定と，IT 導入の効果測定のための指標設定を IT-BSC^[4]を用いて行い，ITKGI と ITKPI を設定した。また，システム導入後の業務プロセスの改善を顧客と共に行った。

③ 開発フェーズ

要件定義フェーズで得られたシステム要件を基に，システムの設計・実装・テストを行う。開発手法として全 5 回のイテレーションによる反復開発を行った。

④ 評価フェーズ

システムを納品し、利用者に試用してもらった後、業務記録とアンケートにより戦略立案・要件定義フェーズで立てた目標が達成されているかを測定し、業務改革の効果、システムの品質、マニュアル等について、評価を行なっている。当初は12月末に終わる予定であったが、評価フェーズ中に、顧客のシステムへの習熟に予想より時間がかかることと、評価フェーズを短く設定したため、評価を行うためのデータが十分に集まらないことに気づき、2月末まで行う予定に変更したためである。

各フェーズの詳しく内容について、2章にて述べる。

第2章 各フェーズの実施状況

2.1 戦略立案フェーズ

2.1.1 目的

戦略立案フェーズの目的は、顧客の外部環境評価、内部環境評価、そして顧客の思いとあるべき姿を描き、認識を共有して、顧客が市場においてどのような戦い方をするのか、を決めること、すなわち、経営戦略を立案することである。顧客にとって最も効果のある改革を行い、使われない・使われなくなるシステムではなく、真に必要なソリューションを提供することが目的である。

2.1.2 方針

経営戦略を立案する際に、経営者の想いを共有し、経営者自身の発想を大切にしながら、支援を行う。また、コーチや事務員の問題意識を把握し、経営者と従業員の想いのギャップを埋められるように助言する。その後、経営者に、経営課題に気づいてもらい、経営者と共に経営戦略を作り上げる。

2.1.3 流れ

(1) ビジョンの確認

一般に、ビジョン（Vision）とは、会社としてやりたいこと、実現したいことを社会に説明したものがビジョン。他の会社でやっていない新しいこと、価値のあることをする必要がある。

当初、T1 経営陣は形式化されたビジョンを持っていなかったため、図 2-1 のようなマインドマップを経営陣の前で作りながらヒアリングを複数回行い、考えを掘り起こし、ビジョンの明確化を行った。企業理念・使命・経営陣の思いから構成されるビジョンを表 2.1 に示す。

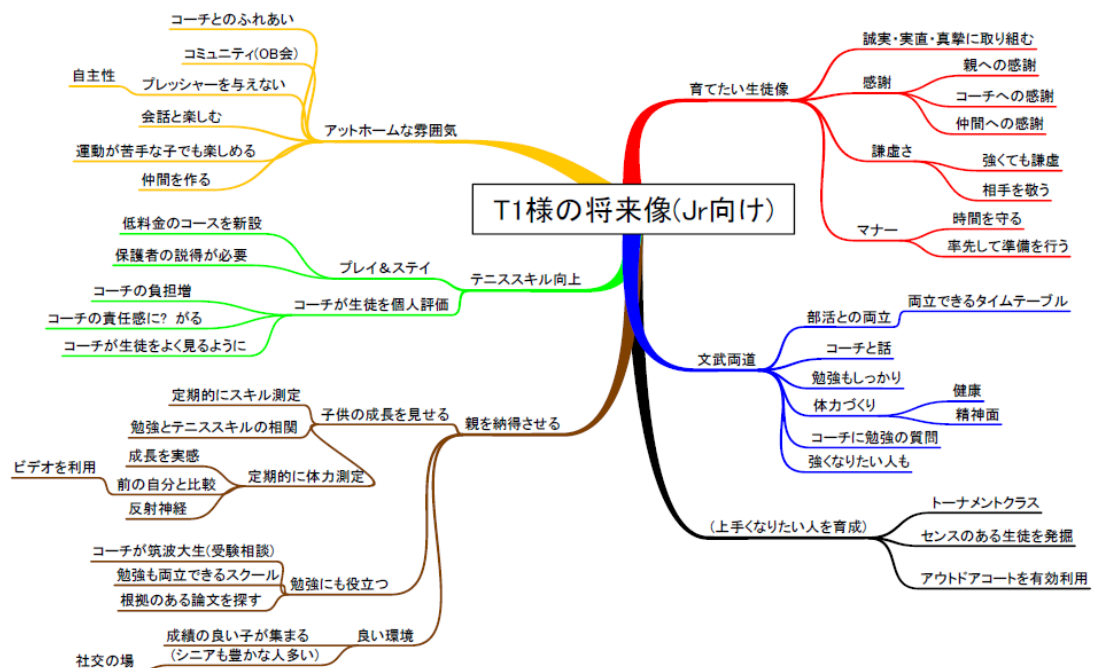


図 2-1 マインドマップ (ビジョンの確認)

表 2.1 T1 の企業理念, 経営ビジョン, ミッション

企業理念	
・来て下さる方を幸せにする ・ボールを打つこと, 試合をすることの楽しさを伝える	
経営ビジョン	ミッション
人生を豊かにするテニススクール	・ジュニアの生徒に対するミッション -テニスを通じ自分を考えることをサポートする -自分を見つめる力, 振り返る力を伸ばす ・すべての生徒に対するミッション -テニス仲間を作る場所を提供する -テニスをより楽しむために, 上達のヒントを提供する

以上の作業の結果, T1 の企業理念は「人生を豊かにするテニススクール」に決定し, T1 は「試合で勝つことを追い求める」よりも, 「テニスを通じて仲間と共に楽しむ」ことを重視するスクールであると定義した。

(2) 外部環境調査

外部環境調査では, テニススクール市場や他スクールの動向を調査し, T1 を取り巻く外部環境を洗い出した(図 2-2). IT コーディネータ実務ガイドを参考に, 規参入者, 社会動向, 代替サービス, 供給業者, 競争企業, 顧客ニーズの 6 つの視点に分類した。

新規参入者 ・N、C各テニススクール -インドアコートを設置 ・洞峰公園 -格安レッスンを開始	社会動向 ・錦織効果 ・健康志向 ・習い事増加 ・不景気	代替品・代替サービス ・フィットネス ・仕事を始める
供給業者 ・現コーチの紹介 ・大学サークル ・体育会を引退した学生 ・主婦	競合物 ・社会人サークル ・中学生の部活 ・進学塾	顧客ニーズ ・健康志向 ・技術向上 ・友達作り ・楽しみたい ジュニアの9割(経営者ヒアリングより) ・競技志向 ジュニアの1割(経営者ヒアリングより)

図 2-2 外部環境調査

T1 はつくば市内で初めて、2003 年にインドアコートを導入し、天候に関わらずレッスンが行えることを売りにしている。また、近年、小中学生の習い事の利用率が上昇している。特にスポーツ系の習い事は過去 10 年間で 10%程度増加しており^[5]、2011 年度では小学生の 54%、中学生の 14%がスポーツ系の習い事を利用している。茨城県の公立中学校では現在、硬式テニス部が存在していないが、設置を進める動きがあり^[6]、小中学生の生徒獲得の機会となっている。

しかし近年、市内の他テニススクールが相次いでインドアコートを導入した。そのため他スクールとの差別化を行い、より魅力のあるテニススクールへと改革することが急務となっている。

(3) 内部環境調査

内部環境調査では、網羅性の高く詳細な内部環境の情報を得るために、顧客にヒアリングをしながらマインドマップを使って行った。実際に作成したマインドマップの例をに示す。

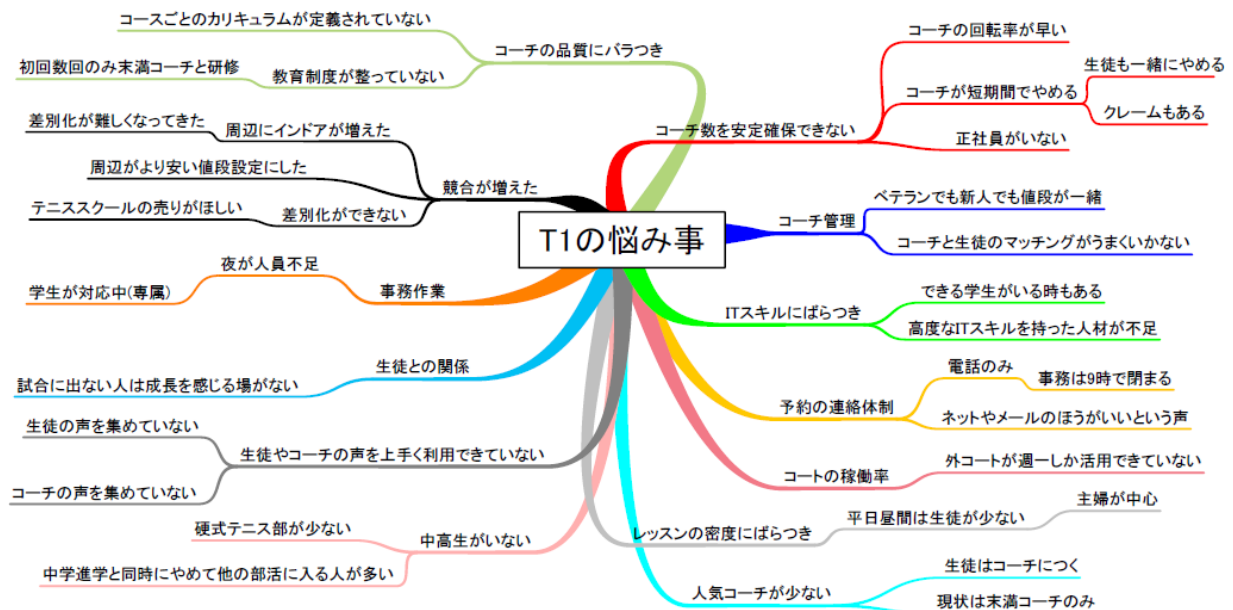


図 2-3 マインドマップ（内部環境調査）

マインドマップの情報を整理した結果を図 2-4 に示す。IT コーディネータ実務ガイド^[7]を参考に施設、人材、人材調達、宣伝方法、立地、社内システムの 6 つの視点に整理した。

施設	人材	人材調達
・インドアであること(非テント) ・合法的 - 某競合スクールは建築基準法に適合していない ・半面コートがある - 生徒1人当たりのコート面積広 - コーチと多く触れ合えるレッスン - 1人から開講 ・託児ルーム ・卓球台 ・コートの後ろが狭い ・老朽化(雨漏り等) ・アウトドアコートが遠い ・シャワー室が1つ ・1コマで1レベルレッスン - 顧客の要望に対応出来ない場合有	・学生コーチ - フレッシュ、さわやか - ある程度のミスでも生徒は許容 - スキルにばらつき - コーチの休みが多い ・未満ヘッドコーチ - コアなファンが存在 ・そのファンがいるコーチ	・学生コーチ - 人件費が抑えられる - 流動的 ・核となる正社員コーチがいない
宣伝方法	立地	社内システム
・ホームページ ・地域情報誌 - オープン直後は効果大 ・口コミ/紹介 ・チラシ(10年前) ・電柱広告 ・看板	・住宅地 - 地価が高い - 子供が比較的多い ・子供が段々少なくなっている - 近隣小学校が4→2クラス ・バス停が近い	・エクセルと紙 - 連絡先が見つからないことも - 出欠も紙 - 新人が台帳を覚えるのが大変 - 規模が増えたと不安 ・エクセルの顧客情報 - N社顧客管理ソフトを使いこなせなかった - 辞めた人はデータ化していない

図 2-4 内部環境調査

T1 は周囲に小中学校が複数ある住宅街の中に位置し、託児施設も備えていることが強みである。そのため、生徒の約 35%を小中学生、約 33%を 30 歳～45 歳の中年層で占めている。生徒年齢分布図を図 2-5 に示す。

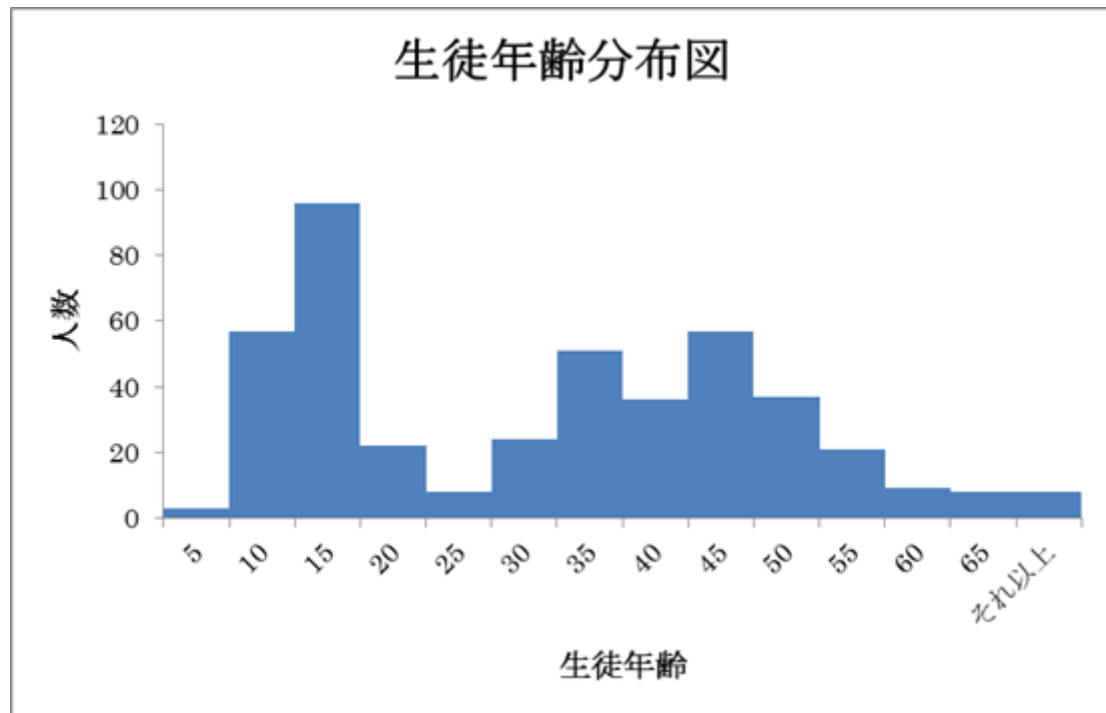


図 2-5 生徒年齢の分布図

17 人のコーチの内、14 人は学生コーチであり、売りである「アットホームな雰囲気」の要素の一つにもなっている。その反面、卒業と同時に辞めてしまう、急なシフト替えや欠勤が発生する等の問題が発生している。

現状 IT 化されているのは、Excel で管理されている顧客情報のみである。コーチの管理は紙ベースで行なっている。

(4) SWOT 分析

経営陣と共に SWOT 分析し、すなわち、集めた情報を強み、弱み、機会、脅威に分類した。SWOT 分析を行った結果の一部抜粋を図 2-6 に示す。

内部環境	S trengths (強み) ・住宅街に立地するため生徒に小中学生や主婦が多い ・インドアコート ・半面コート ・託児ルーム	O pportunities (機会) ・茨城県公立中で硬式テニス部を作る動き ・錦織圭効果によるテニスブーム ・習い事の増加 ・健康志向 ・TXによる商圈の広がり	外部環境
	W eaknesses (弱み) ・コーチ雇用期間が短い ・コーチのスキルにバラつき ・社内の業務プロセスが場当たりの ・社内システムがExcelと紙ベースで管理しづらい ・地価が高い	T hreats (脅威) ・他テニススクール ・フィットネス・社会人サークル等代替サービスの増加 ・少子化 ・中学高の部活・仕事復帰等による生徒の退会	

図 2-6 SWOT 分析の結果

(5) クロス SWOT 分析

SWOT 分析の結果からクロス SWOT 分析を顧客と共に行った。強みと機会、強みと脅威、弱みと機会、弱みと脅威をそれぞれ掛けあわせ、戦略案を抽出した。その結果の一部抜粋を図 2-7 に示す。

	機会(O)	脅威(T)
強み(S)	積極戦略(S×O) ・新コースの設置 －年配者,部活生向け等 －早朝,夜間コース ・口コミで生徒数増加 －生徒紹介サービス (他4項目)	差別化戦略(S×T) ・強みの宣伝 －大型看板の設置 －HP,SNSの利用強化 ・託児所の充実 －託児無料コースの設置 (他2項目)
弱み(W)	弱点改善戦(W×O) ・内部基盤の形成 －方針定義,文書化 －代行コーチ問題改善 ・指導方針,理念の共有 －コーチと意思共有 －生徒に思いを伝える (他3項目)	防衛戦略(W×T) ・生徒の意見収集 －辞める方へアンケート実施 －やめた理由の統計 ・コーチ採用の宣伝 －学生サークルとの繋がり創出 －コーチOBの同窓会開催 (他1項目)

図 2-7 クロス SWOT 分析の結果

(6) 戦略の決定

クロス SWOT 分析の結果, 生徒の要求に合致した新クラス設置により生徒満足度を上げる戦略や, 口コミにより生徒数を増やす戦略など, 強みと機会を活かした戦略が比較的多く抽出された。

しかし, 我々は顧客と共に, 弱みである内部基盤を固める事ができていなければ, 満足度向上や生徒数増加の戦略に伴う業務増に対応できないと判断した。

以上の理由により, 2012 年度は, 内部基盤強化の改革を行い, 満足度向上のための改革などは次年度以降に行うことを顧客と合意した。作成した経営改革ロードマップを図 2-8 に示す。

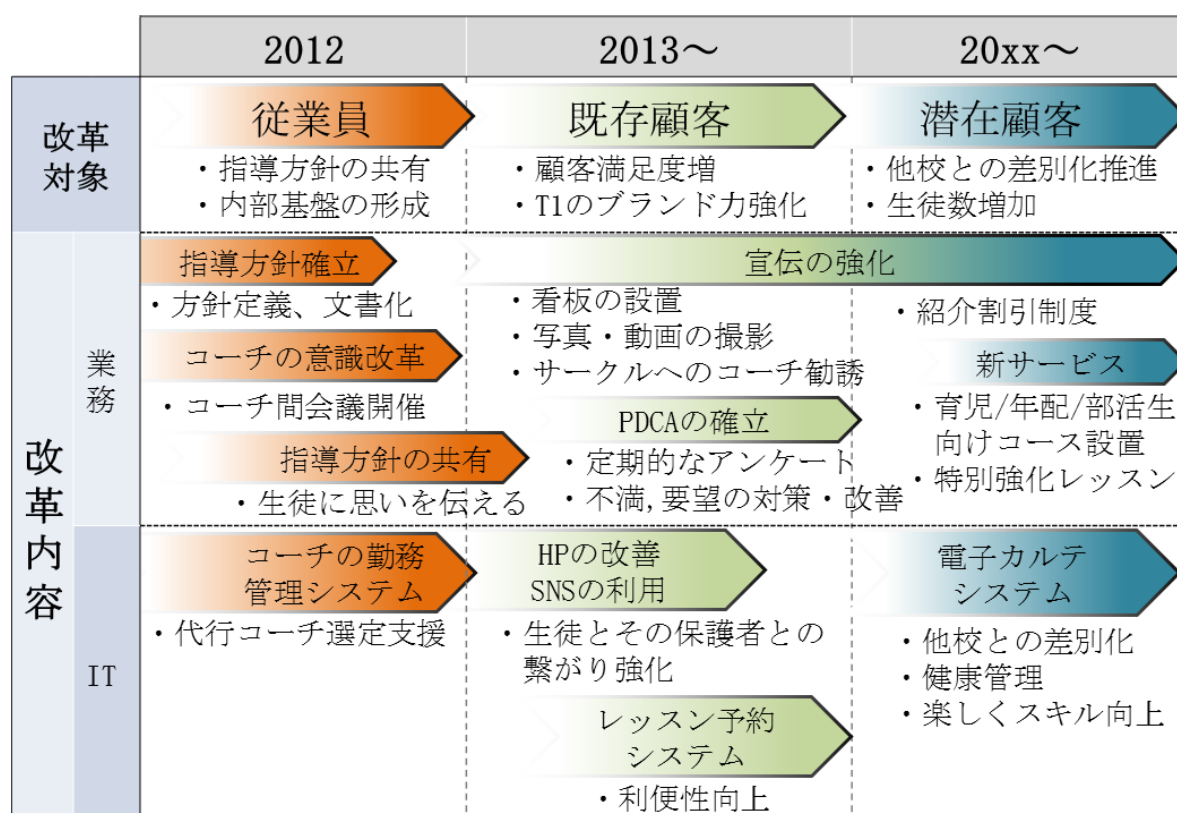


図 2-8 経営改革ロードマップ

(7) 評価指標の作成

今年度の目標は内部基盤の強化である。改革を実行する上での効果を定量的に測定する必要がある。本プロジェクトでは, それを把握する指標として KGI を設定し, それを達成するために, 適切に改革が実行されているかを中間的に計測する指標として KPI を設定した。

BSC を用いて財務の視点・顧客の視点・業務プロセスの視点・学習と成長の 4 つの視点から KGI と KPI を設定した。図 2-9 に BSC による評価指標を示す。

戦略マップ	CSF	KGI	KPI	アクションプラン
財務の視点 	・経営の安定	・正の営業利益と売上高営業利益率	・営業利益 ・売上高営業利益率	・低コストなIT運用
生徒の視点 	・退会者減	・生徒満足度 (総合、システム)	・無断欠席数 ・フリータイムの人の レッスンの消化率 ・レッスン振替回数	・レッスン予約 システム導入
業務プロセスの視点 	・事務の負担減	・レッスン開催 遅延0 ・事務の負担感	・代行選定一回当り 時間 ・事務作業時間 ・メール送信回数 ・コーチドタキャン 回数	・代行コーチ管理 システム導入
人材と変革の視点 	・理念の浸透 ・指導方針の 統一化	・従業員の意識と 満足度	・ミーティング 参加人数 ・ミーティング延べ 参加時間 ・アイデア提案回数 ・会議開催遅延率	・コーチにアンケート を取る ・ミーティングを 開催する ・アイデア提案を させる

図 2-9 BSC による評価指標

BSC は一般に、人材と変革の視点が改善されると業務プロセスが改善され、その効果により生徒の満足度が向上、結果的に財務状況も良くなるというインフルエンスダイアグラムを示している。

そのため、財務や顧客の KGI・KPI は遅効性を持ち、四半期から一年単位で測定することにより機能する。しかし、今回のプロジェクトでは、KGI・KPI の測定期間が一ヶ月弱しかなく、有意な値の測定が困難である。但し、本プロジェクトに限らず、とくに業務改革の指標では、評価期間が短くなる傾向が多い。

そのため、本プロジェクトの評価フェーズでは、IT ソリューションに強く関連する「業務プロセス」の視点の指標評価のみ、測定することにした。測定結果の詳細は、第 2 章第 4 節にて述べる。

KPI はシステムを運用している途中で、効果の途中経過が分かる。そして KGI は経営期間の期末の段階で、改革の効果がどの程度だったかが分かる。4 つの視点の KGI と KPI から改革方針・運用方針を少しずつ修正していくことで、PDCA サイクル(plan-do-check-act cycle)を回し、徐々に、組織成熟度向上の効果が発揮されることが期待される。

(8) 本年度アクションプランの策定

顧客にヒアリングした結果、T1 が抱える経営課題は次の 3 点である。

1 点目は経営者の持つ経営理念やレッスンの指導方針が現場コーチと共有できていない点である。現在は各コーチが独自に指導方針を持ちレッスンを行っているため、コーチによってレッスンの品質にばらつきが発生している。

2 点目は、レッスン管理業務の内、特に代行選定業務がプロセス化されていないという問

題である。スクールのコーチはほぼ全員が学生であり、学校の予定などにより不定期にレッスンを休む場合が多い。現在、代行選定業務が月に約 30 レッスン分発生しているが、属人的かつ場当たり的に行われている。コーチの安定的な割り当ては、スクール運営の基礎であるため、明確な仕組み作りを行う必要がある。

3 点目は、コーチの勤務評価が行われていないという問題である。コーチを評価する事は、スクールに有益な行いやコーチのモチベーションアップに繋がる重要な業務である。そのため、コーチに対して正当な評価を行うための業務プロセスが必要である。

本年度は以上の経営課題に対して業務プロセス改善の支援と IT ソリューションの提供を行う。1 点目の課題に対しては経営者とコーチの対話の機会を創出する事により、改善を図る。具体的にはコーチ全体ミーティングの見直しと、個人面談の実施に取り組む。2, 3 点目の課題に対しては、代行選定プロセスの見直しを行い、事務業務の不可軽減と代行コーチへの引き継ぎ改善に取り組む。業務プロセス改善は T1 経営陣が主体的に行い、IT システム導入はチーム SANITY が主体的に行うということを合意した。

2.1.4 担当部分と振り返り

戦略立案フェーズでは、筆者は内部環境分析を中心に、アンケートの提案、顧客情報分析、顧客アンケートの集計と分析などのタスクを担当した。そこで、アンケートの提案について反省したいと思う。

顧客の満足度を聞き取るために、筆者が生徒向けの満足度調査アンケートを作成し、アンケート作成ツール SurveyMonkey とアンケート分析手法 CS ポートフォリオ分析を提案した。SurveyMonkey とは Web アンケートやネット調査、フォームの作成ができるオンラインアンケートツールである。メール、Web サイト、Twitter、Facebook などに URL のリンクを貼り付け、手軽に回答を収集することができる。リアルタイムで結果を表示し、さまざまなグラフが選択できる。CS ポートフォリオ分析とは、項目別満足度と総合満足度から、重点改善領域を抽出する分析手法です。

さらに、顧客にレビューしてもらい、チームで修正を行った。しかし、アンケートの実施はしなかった。その原因を分析すると次のようになる。

まず一つ目は、筆者がアンケート調査に集中しすぎて、アンケート調査がプロジェクト全体における位置づけについて考えていなかった。本プロジェクトの目的は T1 の経営改革を支援することである。アンケート調査はあくまで情報収集の一つの手段になる。「なぜアンケートを実施するか」「アンケートで集めた意見に対して、改善できるか」について考えていなかった。

また、アンケート調査を実施することで、生徒に不満を思わせる可能性がある。例えば、コーチの言葉遣いに対する満足度を聞いたなら、そもそも気づいていない生徒も、それからコーチの言葉遣いについて気になり、不満に思うようになるかもしれない。アンケート調査の結果、満足度の現状を把握できたが、満足度がこれから下がる可能性もある。

さらに、アンケート調査の実施より、知りたい情報をより正しく早く手に入れられる手段がある。今回アンケートの提案から、チームで修正するまで、50 人時以上の工数がかかった。しかし、その後、過去アンケートの分析、コーチや事務員へのインタビュー、テニスレッスン見学などのことを行うことで、十分な情報を手に入れた。アンケートにかけた工数は無駄になった。

以上のことから、アンケート調査を行う際に、アンケートの位置づけを検討する上で、アンケート実施の影響とアンケートのコストを考慮すべきだということを教訓として残す。

2.2 要件定義フェーズ

2.2.1 目的

要件定義フェーズの目的は実装すべき機能すなわち機能要件と満たすべき性能すなわち非機能要件を明確にすることである。利用者である T1 経営陣，事務員，コーチがこのシステムで何をしたいのかを元に，それを実現するために，実装しなければならない機能や，達成しなければならない性能などを検討して明文化する。

2.2.2 方針

顧客はシステム発注経験がないため，システム開発の流れと各フェーズの内容に関してよく分からない。そこで，私たちは分かりやすく説明し，顧客の IT スキルに合わせて柔軟に対応する方針を決めた。顧客と一緒に考えることで，意識共有すること，顧客ニーズとシステム要件のぶれをなくすことを目指した。

2.2.3 流れ

(1) 取組む範囲の明確化

戦略立案フェーズで策定した戦略を実施するために，T1 とチーム SANITY それぞれの取組む範囲を表 2.2 のように合意した。

表 2.2 責任範囲と支援する範囲の明確化

T1 が取組む範囲	チーム SANITY サポートする範囲
◆ 業務改善案の立案と検討 スクールの理念，方針の明文化 コーチ間会議の開催と改善 継続的に指導方針を見直し，コーチ全員で共有していく組織づくり ◆ システムの発注 機能や使いやすさなどの要求 満たすべき達成条件の検討・合意 ◆ 導入・運用 システムを利用した業務の運用	◆ 業務プロセス立案のお手伝い どんな会議が効果的か？ 理念の共有はどのようにやったらいいか？ ◆ システム導入のお手伝い システム利用マニュアルの作成 システム利用者向け講習会の開催
チーム SANITY が取組む範囲	T1 をお願いしたい範囲
◆ 対象業務の分析 ヒアリング 各種分析 アンケートの作成，結果の分析 ◆ システム開発全般 要求整理，設計，実装，テスト プロジェクトマネジメント ◆ システム導入後の業務プロセスの提案 新しい業務プロセスの立案（コーチ管理業務） ◆ システムの評価 対象業務の評価指標の設定 アンケートの作成，結果の分析	◆ 開発期間のご協力 機能の詳細や画面設計へのご意見 開発期間，実装物に対してのご意見 利用者になる方のご意見 ◆ 情報収集のご協力 システム導入のためのアンケート実施 システム評価のためのアンケート実施 システム導入後の定量評価の記録（時間や回数など） ◆ 情報開示のご協力 分析のための資料の提供 現状についてのヒアリング

(2) システム概要の決定

システムの概要については顧客と協議を重ね，次のように決定した．全てのレッスン情報を管理し，コーチの代行業務の効率化と利便化を実現するためのシステムである．また，蓄積されたレッスンのデータから算出される定量値と日々記録する定性情報を元に，コーチ評価を行う事を支援するシステムである．利用者は「コーチ」と「管理者」，そして「経営者」である．各利用者に応じて適切な機能を提供する（図 2-10）．

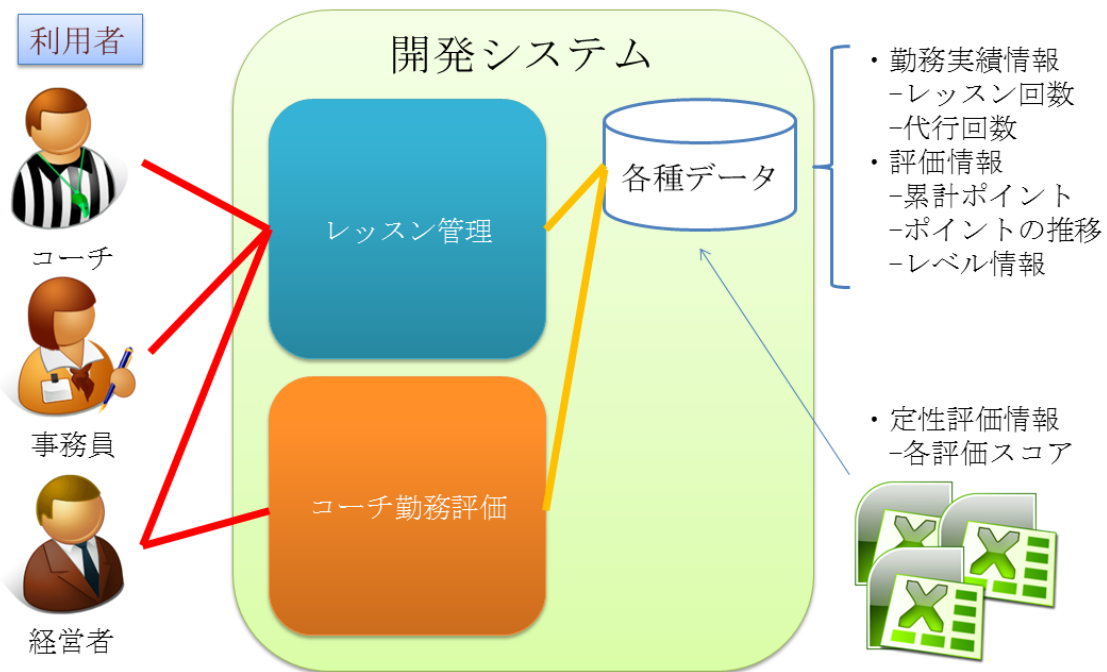


図 2-10 システム概要図

(3) 機能要件の抽出

システムの機能要件について、本プロジェクトでは利用者にとって価値のある機能を単位とするユーザストーリー^[8]として定義した。ユーザストーリーとは開発するシステムにおいて顧客が実現したいと考えている機能を簡潔に記述したものである。戦略立案フェーズで作成したマインドマップなどのドキュメントや実際の学生コーチ、管理者、経営者の意見や要求を元にユーザストーリーの抽出を行った。ユーザストーリー抽出の際には開発すべき機能を漏れなく抽出しなければならない。具体的には、システムの画面デザインのプロトタイプをMicrosoft PowerPoint を用いて作成し、画面ごとに必要な機能を洗い出し、機能の漏れがないことを確認しながらユーザストーリーを洗い出した。さらに顧客ミーティングにおいてチームで洗い出したユーザストーリーを元に顧客と内容を確認しながら提案とレビューを繰り返し行った。私たちのチームでは最終的に32個のユーザストーリーを抽出した。抽出したストーリーは主に4つの機能群に分割される。それぞれの概要を以下に示す。

① アカウント管理

利用者ごとのアカウントを管理する。利用者の氏名やメールアドレス、ログインパスワード、顔写真画像を取り扱う。また、ログイン認証とアクセス制御を行う。

② スケジュール管理

スクールで実施される全てのレッスンの情報を管理する。レッスンの実施日、クラス、担当コーチ、開始時間、コート等の情報を取り扱う。カレンダーとして閲覧できる。

③ 代行管理

スクールで不定期に発生するコーチ代行を管理する。代行依頼、代行立候補、代行コーチ選任までの一連のプロセスを扱う。メールによる通知や立候補が出来る。

④ 評価管理

コーチの評価情報を管理する。勤務回数、代行回数、出勤率などの定量評価値と、その他評価に加味すべき定性評価値を取り扱う。各コーチの勤務実績や評価値を、グラフとして閲覧で

スケジュール管理系と代行管理系の機能群は、レッスン管理に対応する。これらの機能は事務の負荷軽減と代行コーチへの引き継ぎを改善するための機能である。管理者はレッスン情報の管理ができ、コーチは欠勤時の代行依頼や、他コーチからの代行依頼に対して立候補を行うことができる。システム導入後の業務フローを図 2-11 に示す。

従来は代行依頼や立候補者の取りまとめを全て経営者が行っていたため経営者の作業負荷が多かった。本システム導入後、システム上で代行コーチの選定業務を行えるため、経営者の負荷を大きく軽減できる見込みである。

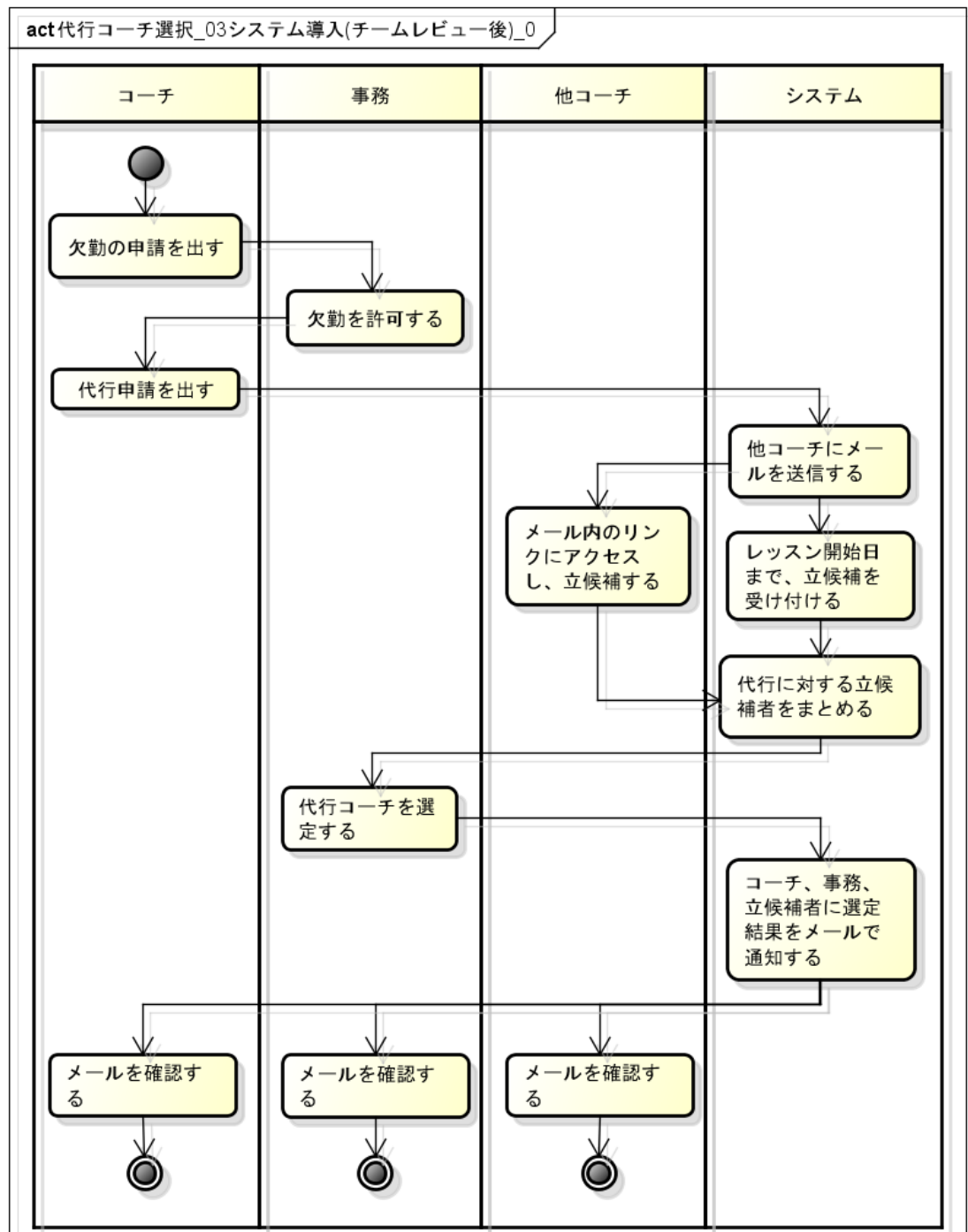


図 2-11 業務フロー図

また、代行申請を出す際にコーチはレッスンの様子や生徒の特長などを記述し、代行コーチがレッスンをしっかり引き継げるようにする。

評価管理系の機能群はコーチの人事評価を行うための機能である。レッスン管理機能により自動で蓄積されていくレッスンの勤務実績と、管理者が管理する定性的評価を基に評価ポイントを算出し、コーチごとに管理・閲覧できるようにする。

(4) 機能要件の受け入れ基準の設定

最終的な受け入れテストをするために、各ストーリーに対して受け入れ基準を設定した。ストーリーとして達成されるべきシステムの振る舞いや機能の制約事項などの詳細を記述した。

(5) ユーザストーリーポイントの見積り

ユーザストーリーポイント^[8]とは、ユーザストーリーの機能規模や複雑さを表す数値である。本プロジェクトではこのポイントをスコープマネジメントの指標として利用した。ポイントを見積もるための方法として、本プロジェクトではプランニングポーカー法^[9]を実施した。プランニングポーカー法は、ポーカーに似た要領で、チームの総意に基づいて各ユーザストーリーのポイントを決定していく見積もり手法である。ポイント決定の際にはユーザストーリーを他の複数のストーリーと比較した際の相対的なサイズで見積る。見積もりの大きさに使える数値は予め「1, 2, 3, 5, 8」に限定している。見積もりの大きさを限定した相対サイズでの見積りを行うことで大まかで現実的な規模の見積もりを素早く出すことが可能である。本プロジェクトでは、ユーザストーリーポイントの合計値でシステム規模を見積もった。

(6) ユーザストーリーの優先度

全てのストーリーに対して「コア機能」または「サブ機能」のどちらであるかを定義する。コア機能はシステム構成上必要不可欠な物や、業務プロセス改善の肝となるストーリーに対して設定する。サブ機能は必ずしも必須ではないが、実現することで補足的な価値を生み出すストーリーに対して設定する。以上のようにして洗い出したユーザストーリーの内訳を表 2.3 に示す。

表 2.3 ユーザストーリー内訳

機能群	コア機能数	サブ機能数
アカウント管理系	4	1
スケジュール管理系	5	2
代行管理系	6	0
評価管理系	7	2
合計	22	5

本プロジェクトではストーリー単位で機能を開発していく。そのため、顧客の要求や実装上の都合を加味し、全ストーリーに優先度を付け、順位を取り決めた。以上のようにして抽出されたユーザストーリーリストとその受け入れ基準を基に顧客と合意を取った。

(7) 非機能要件の検討

本システムの利用者はオペレーションスキルが低く、複雑な操作が困難な人を対象としている。そのため、ユーザビリティを特に重視する必要がある。また、本システムは利用者がいつでもどこでも利用できる Web システムとしての構築が求められている。そのため、外的脅威へのセキュリティを確保する必要がある。

2.2.4 担当部分と振り返り

要件定義フェーズでは、筆者は画面イメージの作成、業務フロー図の作成、顧客と資料共有の設定、ユーザインタフェースに関する技術調査などを担当した。以下、顧客とシステム要件を検討する際に工夫したことについてまとめる。

戦略立案フェーズで、顧客とシステムについて会話する際に、筆者は顧客の私たちが話していることに余りよく分かっていないような様子を感じ取った。それを見て筆者は、顧客はシステムのイメージがつかないのではないかと仮説を立てた。

なぜならば、二つの現象から分かる。一つ目は、現状業務がほとんど紙ベースで行われていることから、システムに触れる機会はあまりないということが分かる。二つ目は、顧客名簿の Excel ファイルを分析する際に、記入方法のばらつきとデータ利用率の低さから、顧客は IT にあまり詳しくないことが分かる。

この仮説に対して、言葉や文字で説明するより図や画面を用いて説明するという対策を提案し、次の二つのことを実施してみた。

まず、図 2-12 のように、紙プロトタイプを作成し、基本スケジュール、週カレンダー、月カレンダーについての要件を検討した。紙プロトタイプは顧客に見てもらって得たフィードバックを、ダイレクトに書き込むことができ、顧客からフィードバックをいち早く取り入れることができた。



図 2-12 紙プロトタイプ

次に、Powerpoint と Excel を使い、代行申請、代行への立候補、立候補者の選定、コーチ評価などの機能ごとの画面イメージを作成した。Powerpoint と Excel を利用することで、コードを書くより、短時間で画面のイメージを作成できた。また、画面遷移も再現でき、ユーザストーリーを検討する際に役に立った。

T1						
 管理者・経営者	← 先週 来週 →					
	7月9日	7月10日	7月11日	7月12日	7月13日	土日
	月	火	水	木	金	
	08:20~09:20	-	-	-	-	
★ 代行コーチ選定 1 📅 レッスン担当一覧 💬 メッセージ	09:30~11:00	初中級	中級	中級	中級	初中級
		榎村コーチ	末満コーチ	榎村コーチ	末満コーチ	榎村コーチ
	11:10~12:40	初中級	初級	初中級	初級	初中級
		榎村コーチ	末満コーチ	榎村コーチ	末満コーチ	榎村コーチ
	13:30~15:00	-		中上級	中上級	-
				末満コーチ	末満コーチ	
	15:10~16:10	-	-	-	-	
	16:30~17:30	Jr2	Jr1	Jr2	Jr1	Jr2
		知見コーチ	知見コーチ	渡コーチ	知見コーチ	渡コーチ
	17:40~19:00	Jr2-L	Jr2-L	Jr2-L	Jr2-L	
		知見コーチ	津田コーチ	渡コーチ	榎村コーチ	
	19:10~20:40	初中級	初中級	初級	初級	中上級
		奥山コーチ	津田コーチ	津田コーチ	榎村コーチ	末満コーチ
	20:50~22:20	初中級	中級	上級	初中級	中級

図 2-13 画面イメージ

2011 年度の授業科目「PBL 型システム開発」で、筆者が所属しているチームではウォーターフォール・モデルを採用し、顧客と要件定義書に合意した後、画面定義書を作成した。これに対して、本プロジェクトでは、順番を逆にして、要件がまだ決まっていない段階で、画面イメージを作成した。手直しが多くなり、コストが増えるではないかという心配があるかもしれない。確かに、開発フェーズで作成した画面はここで作成した画面を比べて、メニューバーの設定や色使いなどに関して変更があった。しかし、スケジュールのレイアウト、画面遷移などに関して、顧客と細かいところまで検討ができ、開発する際にも参考になった。

すなわち、経験や標準的な作業流れにこだわらず、顧客の反応を見ながら対策を考えることが大事だと気づいた。このフェーズでの作業を通じ、顧客とシステム要件を検討する際に、紙プロトタイプと画面イメージ図の利用が有効であることが分かった。

2.3 開発フェーズ

2.3.1 目的

開発フェーズの目的はユーザストーリーを元に、設計、実装、テストを通じて、システムを作り上げることである。イメージや期待との乖離を早期に解消するために、反復型の開発形態を取り、コミュニケーションの機会を創出する。

2.3.2 方針

本プロジェクトの顧客は、システムの発注経験がなく、情報システムに詳しくないため、「システム開発の初期段階で正確な要件を網羅できない」というリスクを想定した。このリスクを軽減するために、基本方針である「顧客との密なコミュニケーション」を軸に3つの方針を考えた。1 つ目は、開発段階でも積極的に顧客とコミュニケーションをとり、要望の

変化を常に確認していく。2つ目は、顧客がシステムを実際に操作する機会(これは顧客レビューと呼ぶ)を設け、顧客の気づきを促進する。3つ目は、要求を引き出すだけでなく、要件の変更やフィードバックなどにすぐに対応していく。もちろん、何でも受け入れるのではなく、議論の上、必要と判断すれば実行する。これらの方針をもとに、開発を行い、顧客にとって利用価値の高い、満足度の高いシステム構築を目指した。

開発フェーズでは、要件定義フェーズで定義した要件を基に、5回のイテレーションを行った。各イテレーションの初めに顧客とミーティングを行い、実装する機能の詳細化と確認を行った。また、イテレーションの終わりにもう一度顧客とミーティングを行い、実装した成果物のレビューを行い、フィードバックをもらった。これらの取り組みにより、利用可能なシステムを早期に構築し、継続的に改善することができた。

2.3.3 流れ

(1) 開発の準備

開発環境のソフトウェア構成を表 2.4 に示す。開発言語として Ruby を選択し、生産性の高さに定評のある Web アプリケーションフレームワーク Ruby on Rails^[10]を採用した。Web アプリケーションで用いられる典型的な処理をフレームワークに任せることで、短期間の開発ができた。RSpec は Ruby 言語プログラム用の総合テスト環境であり、テスト自動化に取り組むために採用した。また、システムの見やすさを考慮して、簡単な記述で統一されたデザインを実装できる CSS フレームワークである Twitter Bootstrap を採用した。

表 2.4 開発環境のソフトウェア構成

種類	名称	バージョン
OS	ubuntu 	12.04
DBMS	SQLite 	3.7.14
WEBサーバ	WEBrick HTTP Server	1.3.1
プログラミング言語	Ruby 	1.9.3
JavaScript フレームワーク	jQuery 	1.8.1
Webアプリケーション フレームワーク	Ruby on Rails 	3.2.6
テストツール	RSpec	2.10.1
CSSフレームワーク	Twitter Bootstrap 	2.1.0

開発で利用したツールを表 2.4 示す。Git を利用し、チーム内でソースコードの管理と共有を行った。Redmine を利用し、チームメンバーそれぞれの作業をチケットとして管理した。各チームメンバーの進捗状況と作業内容を Web 上で一括把握できた。顧客は、プロジェクトの最

終決定や作業の成果物が気に掛かると考え、顧客に公開できる情報は極力公開するという方針で、Dropbox を利用して顧客とドキュメントを共有した。顧客ミーティングで決定した事項や、顧客との合意を得たドキュメントを Dropbox 上で管理し、顧客がいつでも参照できる状態。このように顧客向けの情報開示に配慮することで、顧客の不安感の解消に努めた。

表 2.5 開発で利用したツール

種類	名称	バージョン
エディタ	Sublime Text 2 	2.0.1
ソースコード管理システム	Git 	1.7.12.1
プロジェクト管理ソフトウェア	Redmine 	2.0.0
開発ドキュメント共有	Dropbox 	1.4.7
	Google Docs 	
チーム内の メーリングリスト	Google Groups 	
顧客との ドキュメント共有	Dropbox 	1.4.7

(2) イテレーションの実施

開発フェーズでは5回のイテレーションに期間を分割し、反復に機能を作っていた。各イテレーションの前後に顧客とのミーティングを実施し、イテレーションで実現した機能のレビューや、次回イテレーションで取り組むストーリーの見直しを行った。

次に、イテレーションの流れについて述べる。

始めに、イテレーションで取り組むストーリーの選定と要件の確認を行った。顧客ミーティングで行い、当初からの変化がないかどうか留意して議論した。ユーザストーリーを印刷したものを机の上に並べ、優先度や要件の見直しを行った。開発フェーズで総ストーリー 32 件の中に、不要になったストーリー 3 件、優先度が変化したストーリー 5 件があった。

その後、選定されたストーリーを実現するために必要なタスクを洗い出した。具体的な例を図 2-14 に示す。一つのストーリーに対して、テストケースの洗い出し、UI 設計、UI 実装、クラス実装、テストなどのタスクを洗い出した。負荷を平準化するために、各チームメンバが着手できるタスクから担当し、作業時間の見積りをした。平日の 13:00 から 18:00 までをコアタイムとして設定し、チームメンバ全員が同じ部屋で作業した。そうすることによって、知識共有が効果的に行えた。また、あるチームメンバが困難な状況に遭遇しても、他のメンバがアイデアを出し合いチームの成長を感じることができた。

ストーリー	新規	
80 管理者は、コーチの定性的評価ポイントを個別に加算することができる 3.0	368 デバッグ 5.0	367 UI実装 3.0
	366 クラス実装 1.0	365 モデル実装 1.0
	364 テストコード実装 2.0	363 テストケース洗い出し 2.0
	362 UI設計 1.0	361 クラス設計 1.0
79 管理者はコーチの評価基準を設定できる 1.0	374 テストコード実装 1.0	373 テストケース洗い出し 1.0
	372 UI設計 1.0	371 クラス設計 1.0

図 2-14 ユーザストーリーのタスクの洗い出し

続いて、開発した機能を顧客にレビューしてもらった。、操作説明の後、顧客に操作してもらった。実際の使用感や見た目、表現などについて確認してもらった。開発フェーズで機能拡張3件、画面デザイン10件、文字表現3件の変更依頼に対応した。

最後に、チーム活動を向上させるために、KPT (Keep-Problem-Try) による振り返りを行った。それぞれのイテレーションで各チームメンバが気づいた事、感じた事を全体で共有した。実際に作成したKPTのシートを図2-15に示す。KPTを実施することにより、短時間のディリーミーティングの設置、レビュータイミングの変更、マネジメントの改善、リスクの洗い出しなどの成果が上がった。



図 2-15 KPT シート

(3) 納品

運用時におけるシステム構成概念図を図 2-16 に示す。戦略立案フェーズでの調査によると、T1 のホームページを業者に委託して運営しているが、IT 部門が存在せず、IT に詳しい人材がいない。そのため、システムの導入と運用に当って、可能な限り顧客の負担を少なくするために、手軽にサービスを設置、公開することができる heroku を採用した。heroku とは、元々 Ruby on Rails を対象とした、Amazon Web Services (AWS) の IaaS 上に構築された PaaS で、デプロイには分散リポジトリの Git を利用するなど、Web アプリケーションの開発から公開まで非常に簡単にできる優れたプラットフォームである。heroku 上では、Web サーバ Thin、DB サーバ PostgreSQL を利用した。利用者がシステムにアクセスする端末は、主に PC を想定したが、代行立候補など一部の機能に関しては携帯電話でも利用できるように配慮して開発を行った。11 月 19 日にシステムを納品した。

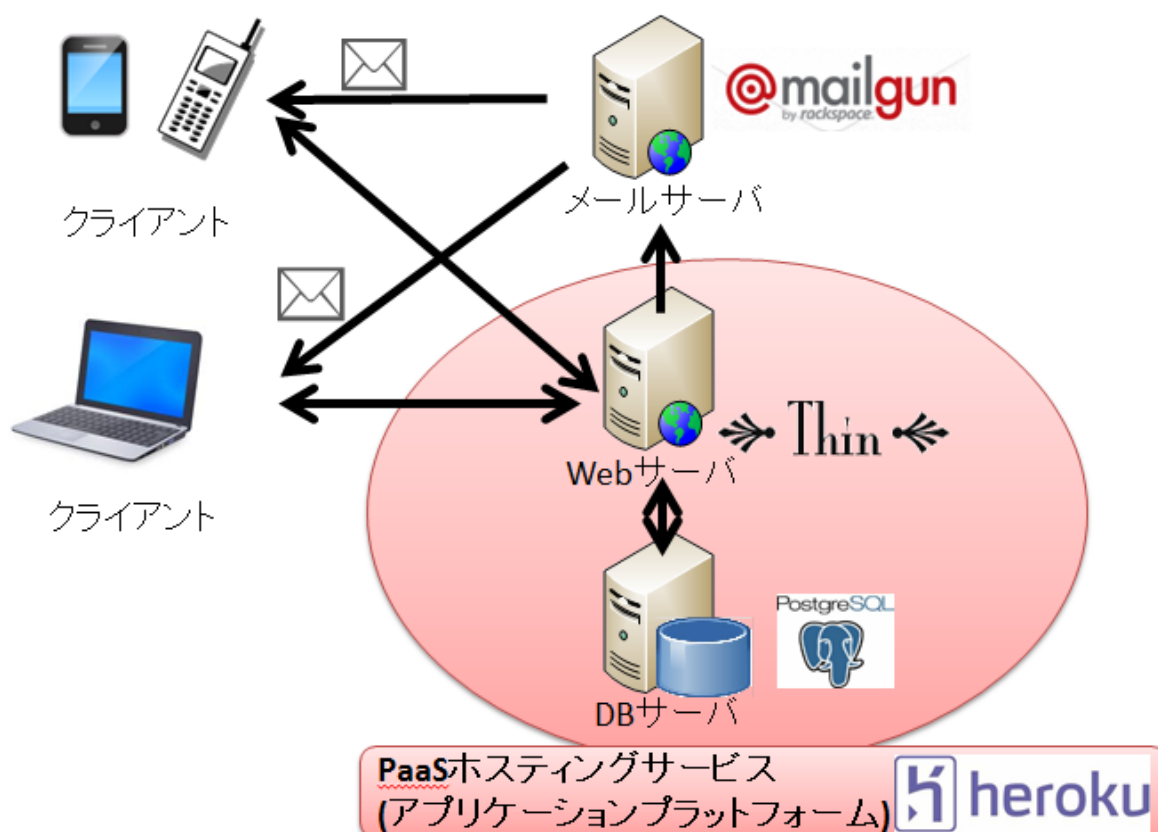


図 2-16 システム構成概念図

2.3.4 開発の実績

開発フェーズにおける開発の進捗状況をユーザストーリーポイントで表すバーンアップチャートを以下の図 2-17 に示す。また、用語の定義を表 2.6 に示す。

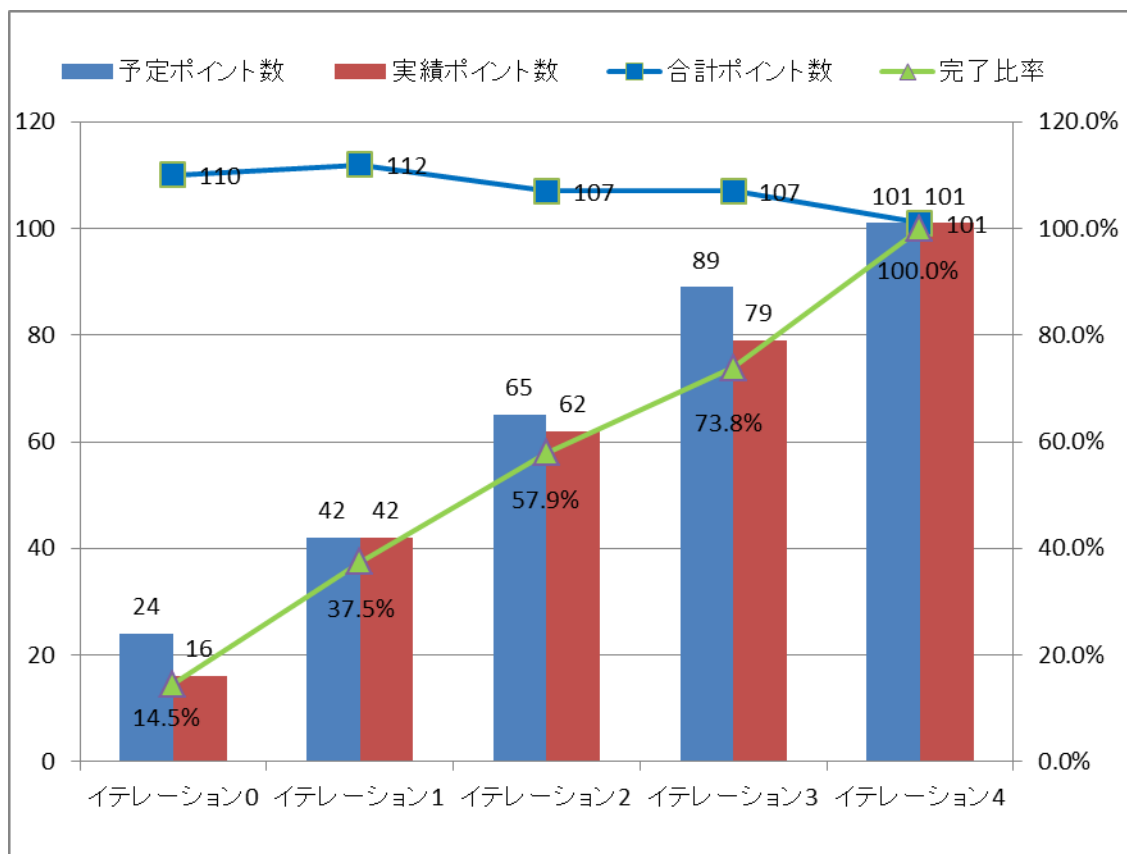


図 2-17 バーンアップチャート

表 2.6 バーンアップチャートで用いる用語の定義

用語	意味
予定ポイント数	各イテレーションの目標ストーリーポイント。
実績ポイント数	実際に終了したストーリーポイント。
合計ポイント	各イテレーション時点での予想合計ストーリーポイント。 ユーザストーリーの変更，ストーリーポイント見積りの修正等により変動する。
完了比率	各イテレーション時点での 実績ポイント/合計ポイント。

図に示す通り，本プロジェクトでは予定されたユーザストーリー全ての開発を終了している。イテレーション0で発生している進捗遅れは開発言語の習熟不足や開発環境に慣れない事が原因であった。その後，イテレーション2とイテレーション3においても進捗の遅れが発生している。イテレーション2については，開発終了箇所にて顕在化したバグ対応が主な原因であり，イテレーション3については開発メンバが体調不良などの理由で一時的に作業

時間が取れなかったことに起因するリソース不足が原因である。

また、合計ポイントの推移を見ると開発が進行するにつれて徐々に値が低下し、スコープが縮小している事がわかる。これは開発期間中に要求が変化し、優先順位が著しく低下した、或いは不要であると顧客によって判断されたユーザストーリーを開発スコープから外した為である。こうしたスコープ調整により、本プロジェクトでは予定した開発期間で顧客と合意した範囲のユーザストーリーの開発を完了することが出来た。各イテレーションで開発を終了したユーザストーリーの内訳を表 2.7 から表 2.11 に示す。

表 2.7 イテレーション 0 で終了したユーザストーリー

機能群	ユーザストーリー	コア / サブ	USP
アカウント	利用者がログイン ID とパスワードを用いることでシステムへのログインを制御することができる。	コア	8
アカウント	管理者は新規アカウントを追加することができる。	コア	2
アカウント	管理者はアカウント情報を変更することができる。	コア	2
アカウント	管理者はアカウントを削除することができる。	コア	2
アカウント	コーチは自分のアカウント情報の変更ができる。	サブ	2

表 2.8 イテレーション 1 で終了したユーザストーリー

機能群	ユーザストーリー	コア / サブ	USP
スケジュール	管理者は基本スケジュールの設定ができる。	コア	8
スケジュール	管理者は最新スケジュールを週カレンダーで閲覧できる。	コア	8
スケジュール	管理者は週カレンダーから担当コーチ名を変更することができる。	コア	3
スケジュール	コーチは最新スケジュールを週カレンダーで閲覧できる。	コア	2
代行	コーチは自分の週表示カレンダーから代行依頼の申請ができる。	コア	5

表 2.9 イテレーション 2 で終了したユーザストーリー

機能群	ユーザストーリー	コア / サブ	USP
代行	コーチは代行依頼一覧を閲覧できる。	コア	5
代行	コーチはシステム上で代行の立候補をすることができる。	コア	5
代行	管理者は代行依頼の一覧とそれぞれの立候補人数を閲覧できる。	コア	2
代行	管理者は代行依頼に対して代行コーチを選択できる	コア	5
スケジュール	管理者は基本スケジュールの変更開始日が設定できる。	コア	3

表 2.10 イテレーション 3 で終了したユーザストーリー

機能群	ユーザストーリー	コア / サブ	USP
代行	コーチは代行依頼のメール内のリンクをクリックすることで代行への立候補ができる。	コア	3
評価	経営者はコーチの定性評価ポイントを個別に加算することができる。	コア	3
評価	経営者はコーチの評価基準を設定できる。	コア	1
評価	経営者は月毎の各コーチ勤務内訳と出勤率をグラフで見ることができる。	コア	5
評価	経営者はコーチ毎のポイント数とレベルをグラフで閲覧できる。	コア	3
評価	経営者は HTTP ベーシック認証でコーチ評価ページにアクセスできる。	コア	2

表 2.11 イテレーション 4 で終了したユーザストーリー

機能群	ユーザストーリー	コア / サブ	USP
スケジュール	管理者は最新スケジュールを月カレンダーで閲覧できる。	サブ	5
スケジュール	コーチは最新スケジュールを月カレンダーで閲覧できる。	サブ	5
評価	経営者は全期間のコーチ全員のポイントをブロックチャートで見ることができる。	サブ	3
評価	経営者は四半期毎のコーチ全員のポイント伸び数ランキングを見ることができる。	コア	2
評価	管理者は全期間のコーチ全員のポイント数ランキング表を閲覧出来る。	コア	2
評価	管理者は四半期毎のコーチ全員の勤務回数とその内訳と欠勤率を比較することが出来る。	サブ	5

2.3.5 担当部分と振り返り

筆者が要件定義フェーズで CSS などの UI 設計に関する技術調査を担当したため、開発フェーズの前半は UI 設計や UI 実装のタスクを中心に開発した。後半では、クラス実装、テスト項目洗い出し、テストコード実装などのタスクも担当した。

本プロジェクトは 2011 年度の授業科目「PBL 型システム開発」のプロジェクトと比べて、以下の類似点がある。5 人のチームであり、プロジェクト期間は 9 ヶ月間、顧客は中小企業であり、提供したシステムは Web アプリケーションである。そのため、筆者は本プロジェクトでの開発経験はチーム開発を勉強するのに貴重な機会だと考えた。以下、チームで開発する際に、タスクの分担、見積りの方法、レビューのタイミング 3 つの面から筆者の経験について述べる。

(1)タスクの分担についての教訓

タスクを分担する際に、2011 年度は機能ごとに分担していたので、開発担当者は自分のところだけを見ていた。機能を実現できれば、実装方法はどうでもいい感じだった。それに対して、本プロジェクトは Redmine のタスクボードを利用し、対応可能なチームメンバが着手できる状態になったタスクにサインアップしていた。そうすることで、様々なタスクを担当でき、各チームメンバ間の繋がりが強くなり、実装方法の共有、コードの可読性が求められた。すなわち、実装方法が妥当かどうか、コードが分かりやすいかどうかお互いにチェックしていた。例をあげると、筆者は開発後半では、管理者ページの評価基準設定のテスト項目洗い出しとテストコード実装を担当した。RSpec の勉強ができ、クラス実装コードの可読性も認識した。また、Evaluations の Model 設計に quarter というカラムを追加すれば実装が楽になるという提案をした。このようなことから、様々なタスクを担当することで、勉強にもなるし、他チームメンバが気づいていない問題点も発見できた。

(2)見積り方法についての教訓

タスクを見積もる際に、2011 年度は WBS を用いて、予定工数を適度に割り振った。その後、作業時間が大きくずれても当然だと思い、気にしなかった。それに対して、本プロジェクトはタスクの担当者がタスクの内容と自分のスキルを考える上で決めていた。その後、作業時間を測り、見積りととの差を分析した。そうすることで、見積りの精度を上げていく。例えば、私は初めて代行立候補の UI 実装を担当する際に、1 時間と見積もったのに、実際に 5 時間もかかった。それは学習時間を考慮していなかったためだった。その後、タスクを見積もる際に、自分がどこまでできるか、何をについて勉強する必要があるかをきちんと分析してから決めた。自分の生産性を正しく認識でき、見積りと作業時間の差も減った。

(3)レビューのタイミングについての教訓

レビューする際に、2011 年度は機能を実現した後、チームでレビューした。その結果、修正するところが多い場合、かなり時間をかけた。それに対して、本プロジェクトはデイリーミーティングを設け、毎日進捗状況を共有していた。完成していないタスクもどこまでやっているか、どこで詰まっているかについて報告した。筆者が UI 全体をチェックする際に、同じコードを書いていたのに、ユーザページの Modal のボタン様式と管理者ページの違うというところで詰まっていた。最初は Twitter Bootstrap が効いていないと想定したが、筆者の報告を聞いたら、他のチームメンバが最初管理者ページの CSS を編集したと教えてくれた。この一言で、問題が解決した。

もちろん、これらの経験はすべてのチーム開発に適用し、必ず作業効率を上げるというわけではない。しかし、こうすることで、個人として成長でき、最終的にチーム力の向上と繋がると考えられる。

以上をまとめると、チーム開発する際に、様々な種類のタスクを担当すること、作業時間と自らの見積り時間を比較すること、一人で悩まずに他チームメンバと相談することが大事だと実感した。

2.4 評価フェーズ

2.4.1 目的

評価フェーズの目的は、業務プロセス改善と IT ソリューション導入の効果、ステークホルダーの満足度など本プロジェクトが顧客にもたらした効果を検証することである。要件定義フェーズで決定した KGI と KPI の測定と利用者向けのアンケートの実施を通じて検証する。

2.4.2 方針

評価フェーズで測定する項目を表 2.12 に示す。

表 2.12 評価フェーズでの測定項目

項目	測定対象	目的	測定方法
IT-KGI, IT-KPI	システム	IT ソリューションと IT ガバナンスの 2 つの視点から、システムが実際に活用されているかを測定する。	システム DB と台帳の比較
KPI	経営者 事務員	IT ソリューション導入前後の業務時間など、定量的な値の推移を測定する。	業務時間の記録
KGI	経営者 事務員 コーチ	システムの勤務管理機能と評価管理機能について、定性的な効果を測定する。	アンケート
使い勝手 アンケート	経営者 事務員 コーチ	システム全般の使いやすさ、見た目の印象などについて測定する。	
マニュアル アンケート	経営者	システムの運用マニュアルの読みやすさなどについて測定する。	

システム納品後、3 週間運用してもらい、評価指標の測定とアンケートの実施を予定していた。しかし、バグが発生し、システムが正常に動作しなかった。また、顧客の都合により、報告がバグ発生から数日後になった。その結果、実際の運用期間が短くなり、評価として意味のあるデータを集めることが困難になった。バグへの対応は迅速に行い、問合せを受けたその日に対応を完了している。現在は安定して稼働している。

そのため、今回収集したデータが足りず、業務プロセス改善と IT ソリューションについての評価は、安定稼働している状態での運用期間を確保し、再度評価を実施することにした。評価フェーズのスケジュールを次の図 2-18 に示す。2013 年 2 月も引き続き評価期間を継続し、2 月末に KGI により最終的なプロジェクトの効果を測定する予定である。

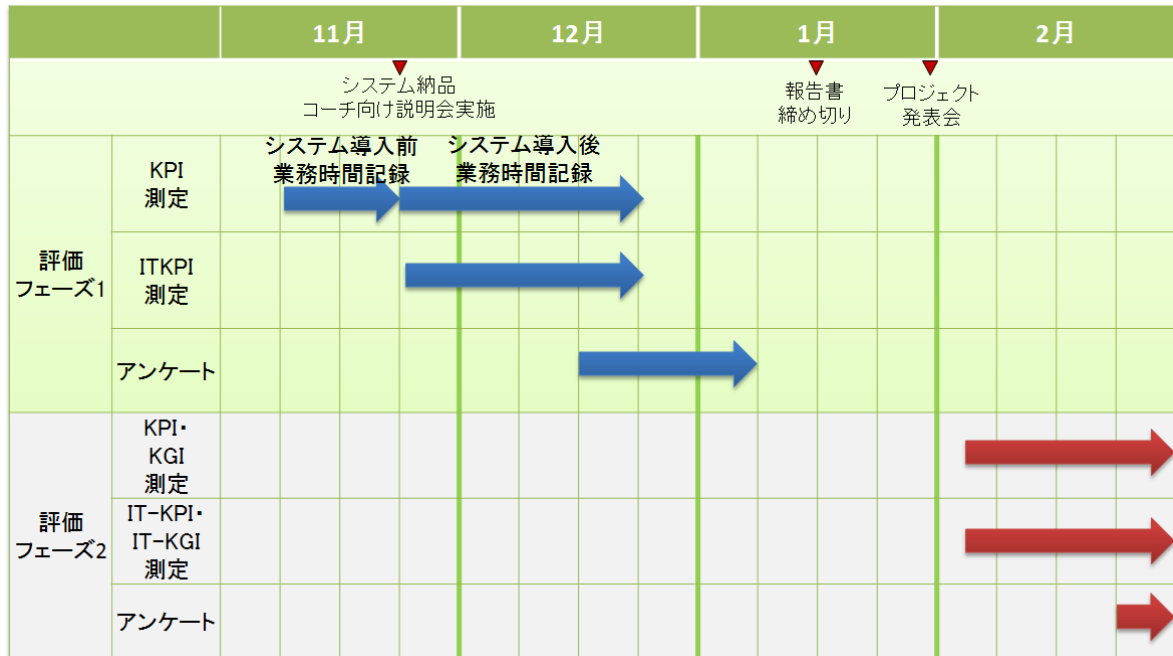


図 2-18 評価フェーズのスケジュール

2.4.3 評価指標の測定結果
IT-KPI の測定結果を表 2.13 に示す。

表 2.13 IT-KPI の測定結果

大分類	小分類	CSF	KPI					アクション・プラン (参考)
			測定項目	測定方法	3日後 (11/23～ 11/26)	10日後 (11/27～ 12/04)	20日後 (12/05～ 12/23)	
ITソリューション	データ化	コーチの情報がDB化されている	コーチ情報変更時 DB更新率	サーバ確認+T1通常記録	66.7% (12人/18人)	72.2% (13人/18人)	100% (18人/18人)	コーチへの教育 実施
		レッスンの情報がDB化されている	レッスン変更時 DB更新率	サーバ確認+T1通常記録	100%	不明	100%	事務員の情報更 新の徹底
	システム活用 AHA(運用と利用の 促進)	勤務管理機能が活用されている	欠勤申請時 システム利用率	サーバ確認+T1通常記録	100% (3回/3回)	15.3% (2回/13回)	13% (2回/15回)	コーチへの教育 実施
		人事評価機能が活用されている	定性評価入力回数	サーバ確認	0回	0回	0回	経営陣への説明
ITガバナンス	運用体制 DS6(サービススマ ンジメント管理)	コーチトラブル回数	SANITY 問い合わせ回数	SANITY記録	0回	0回	1回	マニュアルの作 成 事務員への教育 実施
		管理者トランブル回数	SANITY 問い合わせ回数	SANITY記録	0回	1回	1回	マニュアルの作 成 事務員への教育 実施
	セキュリティ DS6(システム・セ キュリティの保証)	不正アクセストラブル回数	SANITY 問い合わせ回数	SANITY記録	0回	0回	0回	自アカウント利用 徹底、 セキュリティ教育

IT-KPI を測定する際に、トラブルを記録した。トラブルの詳細を表 2.8 に示す。12 月 4 日に顧客から報告された管理者トラブルと 12 月 11 日に顧客から報告されたコーチトラブルはコア機能に関するトラブルであったが、顧客の多忙により発生日から数日後の報告となってしまう、評価期間の前半はシステムがほぼ使用不能になってしまった。しかし、トラブル報告後は全て 1 日で解決しており、迅速な対応ができた。システムが一時的に停止したが、12 月 11 日のトラブル解決後は全機能が正常に動作している。私たちも今回のトラブルから、SQLite と PostgreSQL で SQL 実行結果の違い、RFC に準拠していないメールアドレスの対策などを学んだ。

表 2.14 評価フェーズのトラブルの詳細

報告日	分類	内容	原因	解決方法	解決日
12/04	管理者 トラブル	基本スケジュールを変更すると、最新スケジュールの曜日がずれて破壊される。	開発環境:MySQL と本番環境: PostgreSQL の違いによる、実行結果の相違。	SQL 文修正	12/04
12/11	管理者 トラブル	アウトドアコースの基本スケジュール設定ページにアクセス出来ない。	データベース初期値の入力忘れ。	データベース初期値入力	12/12
12/11	コーチ トラブル	システムからのメールが 18 人中 7 人にしかメールが届かない。	RFC に準拠していないメールアドレスを使用しているユーザにより、Rails の処理が途中で停止した。	エラーが起きても処理が継続するように設定。また、RFC に準拠していないメールアドレスの登録を拒否。	12/12

トラブルの影響により、評価期間において実際にシステムを利用できるのは 1 週間程度となってしまう、低い利用率となった。しかし、1 月初旬の経営陣のヒアリングから評価期間終了後はトラブルも無く運用できていることを確認している。評価フェーズ 2 の KPI 測定では 100%近い利用率を測定できると考えられる。

業務に関する KPI の測定結果を表 2.15 に示す。導入前より、代行に掛かる時間が増えてしまった。これはシステムトラブルの影響により、システムを利用した場合も、正常にメールが送信されているか等の確認を取る必要があり、余計に業務が増えてしまった影響であると考えられる。

トラブル解決後、事務の方から「メール送信の負担が減り楽になった」旨のコメントをもらった。そのため、評価フェーズ 2 では業務時間が削減出来ていることを確認できると考えられる。

表 2.15 KPI の測定結果

大分類	小分類	CSF	KPI			
			測定項目	導入前結果 (11/14～22)	導入後結果1 (11/23～12/8)	導入後結果2 (12/9～12/21)
内部基盤の確立	業務プロセス の視点 ME1(IT成果のモニタリングと評価)	事務の負担減	代行一件あたり 事務作業時間 (分)	7.6分	10.2分	10分
			代行一件あたり メール送信回数 (回)	1.2回	1.4回	2回
	人材と変革の 視点 ME1(IT成果のモニタリングと評価)	代行立候補の手軽さ増				

2.4.4 運用マニュアルアンケートの集計結果と分析

本プロジェクトでは運用準備として運用マニュアルの整備と講習会の実施を行った。筆者は主に運用マニュアルの作成に携わった。作成にあたり、IT 全般の知識が無く、システム運用経験が無い顧客でも運用をしていく事が出来るよう、分かりやすい表現と構成、情報の網羅に配慮した。そこで作成した運用マニュアルについて、読みやすさ、情報に過不足が無いのか、運用の為に情報が載っているかの観点で評価を行う。アンケート対象は経営者とし、顧客のマニュアルに対する満足度を基に評価を行う。アンケートの結果について次の表 2.16 に示す。

表 2.16 運用マニュアルに関するアンケートの結果

運用マニュアルについて	評価方式	経営者 1	経営者 2
Q1. このマニュアルがあれば、今後システムを運用できそうだと感じるか	5段階	4	4
	理由(自由記述)	わかりやすく丁寧にまとめているから	さほど難しくないから
Q2. マニュアルの内容は容易に理解できたと思うか	5段階	5	4
	理由(自由記述)	構成が見やすい	簡単だから
Q3. マニュアルから求める情報を素早く得る事が出来ると思うか	5段階	4	3
	理由(自由記述)	目次や見出しが分かりやすい	
Q4. 誤字・脱字が無く、表記や表現に一貫性があったと思うか	5段階	5	3
	理由(自由記述)	非常にまとめている	
Q5. 問い合わせ先など困った時のための情報が明らかになっていると感じましたか	5段階	5	3
	理由(自由記述)	困った時の解決方法という項目でまとめているので	

この表から、全ての設問において3以上の値を取っており、高い評価を受けていることが分かる。特にQ1、Q2では共に4以上の評価をしており、運用していくために必要な情報を分かりやすい形で顧客に提供することができたと考えられる。

2.4.5 使い勝手アンケートの集計結果と分析

システムの利用者である経営者、事務員、コーチを対象として、使い勝手アンケート^[11]を実施した。13 枚を回収し、そのうち 12 枚を有効回答として集計した。集計結果を表 2.17 と表 2.18 に示す。「得点」は 5 段階評価で示しており、1 に近いほど悪い結果であることを示し、5 に近いほど良い結果であることを示している。▼は逆転項目であり、質問表現がその評価軸の方向と逆になっている。逆転項目の得点は得られた 5 段階結果を 6 から引いたものを使用することとした。

表 2.17 使い勝手アンケートの集計結果 1

大項目	小項目	結果	得点
操作の わかりやすさ	操作手順はシンプルで分かりやすい	3.8	3.8
	使い方はすぐに理解できる	3.8	3.8
	T1 System では、次に何をすればよいか迷わない	2.8	2.8
役立ち感	T1 System ではすぐに欲しい情報が見つかる	3.1	3.1
	T1 System には分からない言葉が多く出てくる▼	2.4	3.6
	T1 System を使用するのは時間の浪費である▼	2.3	3.7
高感度	T1 System の見た目は楽しい	3.2	3.2
	T1 System は印象に残る	3.2	3.2
	T1 System には親しみがわく	3.1	3.1
信頼性	T1 System に掲載されている内容は信用できる	2.6	2.6
	T1 System は信頼できる	2.5	2.5
	T1 System の文章表現は適切である	3.3	3.3
構成の わかりやすさ	T1 System には統一感がある	3.1	3.1
	メニューの構成が分かりやすい	3.4	3.4
	自分が T1 System 内のどこにいるのか分かりやすい	3.0	3.0
見やすさ	T1 System の文章は読みやすい(行間、レイアウト等)	3.5	3.5
	T1 System のアイコンなどのデザインは見にくい	2.5	3.5
	T1 System を利用していると、目が疲れる感じがする	2.5	2.5
反応の良さ	T1 System は操作に対してすばやい反応が返ってくる	2.7	2.7
	T1 System を利用しているときに、 画面が正しく表示されないことがある▼	3.2	2.8
	T1 System 利用している時に、表示が遅くなったり、 途中で止まってしまうことがある▼	2.9	3.1
総合評価	T1 System は使いやすい	3.1	3.1

表 2.18 使い勝手アンケートの集計結果 2

評価軸	総合評価との 相関係数	アンケートの 平均得点
操作の分かりやすさ	0.71	3.47
役立ち感	0.72	3.47
好感度	0.61	3.17
信頼性	0.74	2.80
構成の分かりやすさ	0.76	3.17
見やすさ	0.49	3.17
反応の良さ	0.25	2.87

集計結果を分かりやすくするために、図 2-19、図 2-20、図 2-21 のようにグラフ化した。図 2-19 からシステム全体のバランスが分かる。図 2-20 と図 2-21 は評価軸の得点と重要度を用いたマトリックス分析である。重要度は総合評価との相関係数で表す。

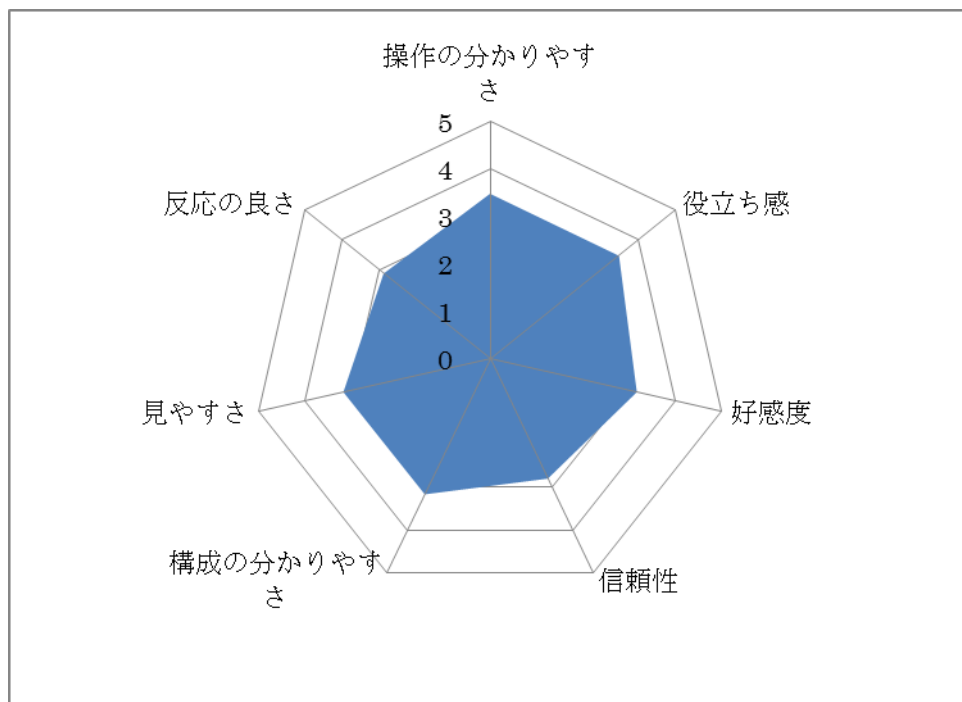


図 2-19 7つの評価軸の得点

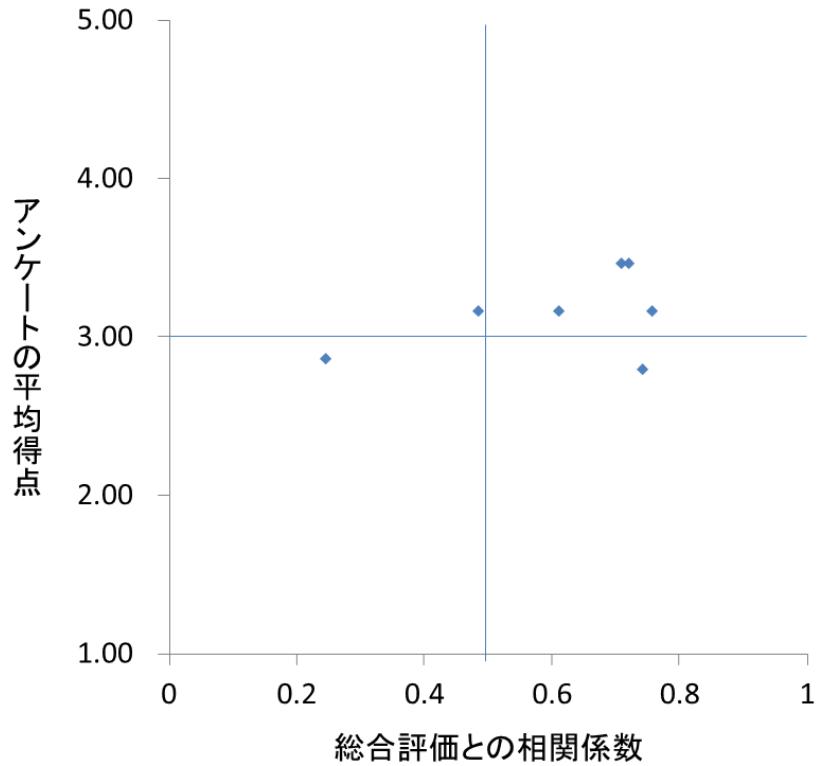


図 2-20 使い勝手アンケートの分析 1

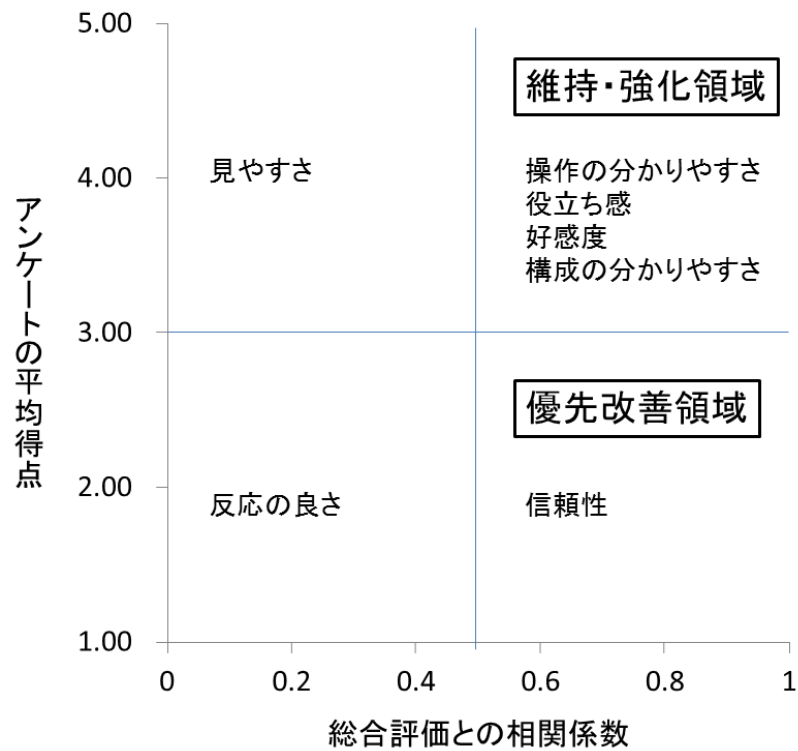


図 2-21 使い勝手アンケートの分析 2

図 2-20 と図 2-21 から操作の分かりやすさや役に立ち感などは維持，強化すべきである．それに対して，信頼性は優先改善すべきである．「信頼性」に関する項目以外は総じて高い評価を得られており，開発フェーズではイテレーションごとに試用してもらい，顧客の意見を随時取り入れるなど，顧客との密なコミュニケーションを実践した結果が出ていると言える．

「信頼性」については，自由記述欄では「システムが正常ではない」というコメントが散見されたため，システムトラブルの悪印象により評価が低くなったと考えられる．システム全体に関する自由記述欄では「正常に作動すれば良いシステムである」というコメントが複数存在し，システムの自体は評価されていることがわかった．

第3章 ユーザビリティの向上

3.1 ユーザビリティの定義

ユーザビリティとは一般的には使いやすさのことであるが、使いやすさは様々な属性が含まれる。この属性として、「Usability Engineering」^[12]の著者として知られる Nielsen による定義や、ISO9241 Part11^[13]における定義が代表的である。

Nielsen はユーザビリティを学習しやすさ(Learnability)、効率の良さ(Efficiency)、記憶しやすさ(Memorability)、エラーの少なさ(Errors)、主観的満足度(Satisfaction)の 5 つの属性で定義している。また ISO9241 Part11 はユーザビリティを有効性(Effectiveness)、効率性(Efficiency)、満足度(Satisfaction)、利用状況(Context of use)の 4 つの属性で定義している。

ISO9241 Part11 は広義な定義であるため、本プロジェクトが構築した Web アプリケーション形式のシステムを評価するのに難しい。そこで、本プロジェクトでは、多角的な構成要素を持つ Nielsen による定義を参考にし、ユーザビリティを表 3.1 のように説明した。各属性の定量的評価内容を表 3.2 に示す。評価結果は 2 章 4 節に示した。

表 3.1 本プロジェクトでユーザビリティの定義

属 性	説 明
学習しやすさ	システムは、コーチがそれを使ってすぐ代行申請、代行立候補できるようにしなければならない；管理者がそれを使ってすぐ最新スケジュールの閲覧、代行申請の選定、コーチ評価の閲覧、基本設定などできるようにしなければならない
効率の良さ	システムは、一度管理者がそれについて学習すれば、後は代行業務の効率を上げられるよう、効率的な使用を可能にすべきである
記憶しやすさ	システムは、不定期利用の管理者及びコーチがしばらく使わなくても、再び使うときに覚えなおさなくて使えるよう、覚えやすくしなければならない
エラーの少なさ	システムは、エラー発生率を低くし、もしエラーが発生しても簡単に回復できるようにしなければならない；致命的なエラーが起こってはいけない
主観的満足度	システムは、管理者及びコーチが個人的に満足できるよう、また楽しく利用できるようにしなければならない

表 3.2 ユーザビリティ定量的評価内容

属 性	定量的評価内容
効率の良さ	代行一回あたり事務作業時間 メール送信回数 立候補回数
エラーの少なさ	コーチトラブル回数 管理者トラブル回数
学習しやすさ	欠勤申請時システム利用率 立候補時システム利用率 定性評価入力回数
記憶しやすさ	
主観的満足度	(使い勝手アンケート) 操作の分かりやすさ 役立ち感 好感度 信頼性 構成の分かりやすさ 見やすさ 反応の良さ

3.2 ユーザビリティの評価手法

本プロジェクトはアジャイル開発手法を採用したため、各イテレーションで PDCA サイクルを回していた。そうすることで、各イテレーションでユーザビリティの評価を行い、次のイテレーションで反映し、継続的改善することができると考えた。

調査によると、様々なユーザビリティの評価手法が開発されたが^{[15][16]}、代表的な手法のカテゴリとしてユーザビリティインスペクション(usability inspection)^[17]及びユーザビリティテスト(usability test)^[18]が挙げられる。

ユーザビリティインスペクションは、システムの UI 設計を評価者が分析してその設計の良否を評価する手法の総称である。ユーザビリティテストは、システムを利用者に操作してもらうことで評価する手法の総称である。この 2 つの評価手法が本プロジェクトにおける利点と欠点について、筆者が表 3.3 のようにまとめた。

表 3.3 本プロジェクトにおける評価手法の比較

	ユーザビリティインスペクション	ユーザビリティテスト
利点	・設計段階でも評価できる，手直しが少ない ・短期間で実施でき，実施コストが小さい	・実際に利用者視点からの使いにくさの問題点を発見しやすい ・開発チームには思いもよらない問題点が見つかりうる
欠点	・利用者が直面する問題をうまく予測できない ・発見できる問題点の数は多いが重要度が小さい ・評価結果が開発者の経験や主観に強く依存する	・ある程度動作するプロトタイプが必要となる ・経営者はもちろん，事務員やコーチの参加が必要なので，実施コストが大きい

本プロジェクトでは，それぞれの評価手法の利点を獲得するために，双方の手法を適用した．利点を獲得すると共に，欠点も2倍になるではないかと思うかもしれない．しかし，ユーザビリティテストの採用により，ユーザビリティインスペクションの欠点を回避できた．それに，各イテレーションでのレビューの導入により，ユーザビリティテストの欠点を補った．

3.2.1 ユーザビリティインスペクション

ユーザビリティインスペクションでは，表 3.4 のようなチェックリストを用いて，UI 設計を分析してその良否を評価した．これは調査した3つのチェックリストをレビューし，本プロジェクトに合う項目を選び，本プロジェクトにおいて重要である5つの領域に分類したものである．

表 3.4 ユーザビリティチェックシート

リーダビリティ		
	・文字が読みやすい大きさであるか	
	・テキストカラーは背景に対して見やすいか(ラベルの文字色は白であるかどうか)	
	・文字はわかりやすく伝えているか (メールの件名に【T-1】があるかどうか)	
ナビゲーション		
	・グローバルナビゲーションが用意されており、カテゴリーの横渡りが可能か	
	・現在のページのサイト内での位置を判別することができるか	
	・ローカルナビゲーションが用意されており、下位階層の行き来が可能か	
リンク要素		
	・リンクのテキスト色はサイト内で統一しているか	
	・ボタンがクリックできるとわかるデザインであるか	
	・リンクにカーソルを当てるときマウスオーバーが適用されるか	
エラー		
	・Notfound (ページが見つからない) 用のページを用意しているか	
	・システムエラー (タイムアウトなど) 用のページを用意しているか	
	・エラーメッセージがエラーが起こる状況と合うかどうか	
フォーム		
	・必須項目がわかりやすいか	
	・入力エラー時にエラー項目がわかりやすいか	
	・入力例がかかれているか	

このチェックリストを使って、すべてのページをチェックした。ページ間の関連を図 3-1 と図 3-2 の画面遷移図に示す。各項目のチェック結果を「○：満たしている」、「×：満たしていない」、「－：関係ない」のいずれかになる。そうすることで、システムのユーザビリティ上の欠陥や開発基準を満たしていないものを見つけ出した。満たしていないものに対して、タスクとして登録し、該当イテレーションで完成するように努力した。

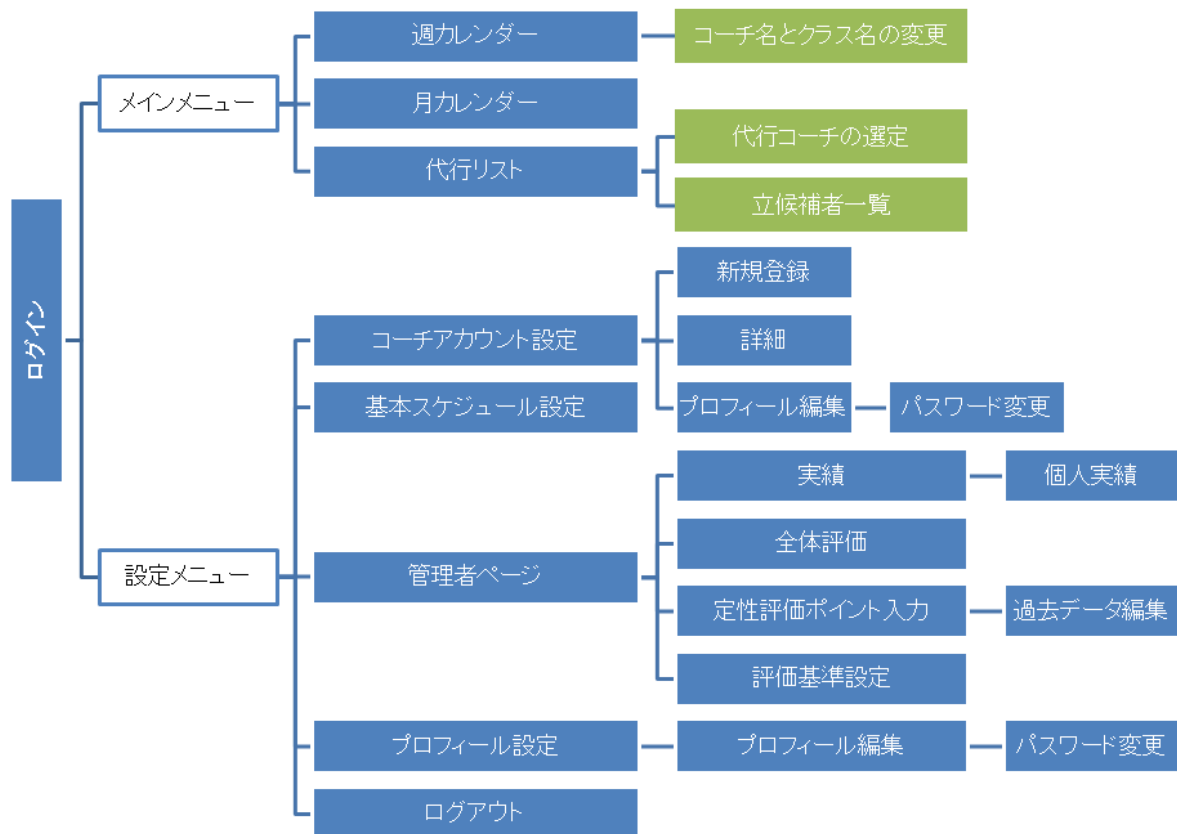


図 3-1 管理者向けの画面遷移図

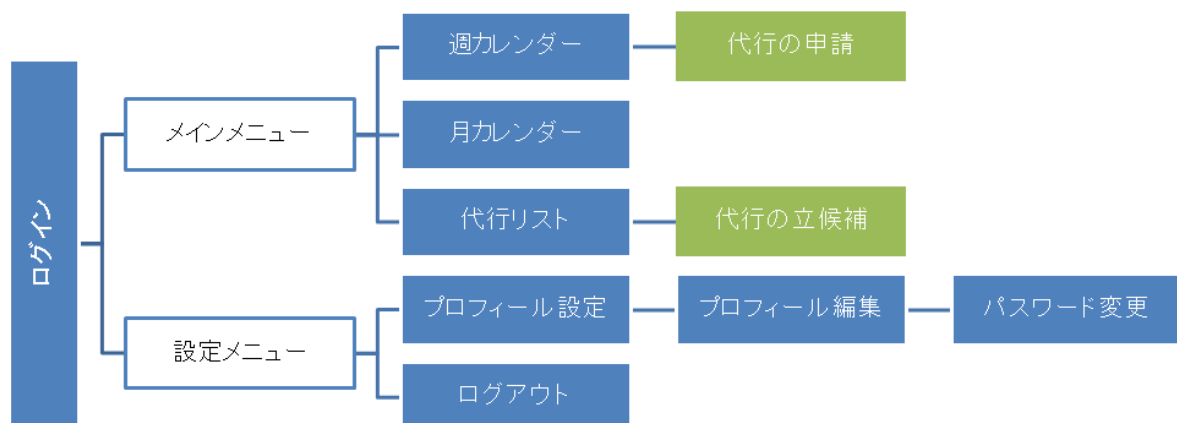


図 3-2 コーチ向けの画面遷移図

3.2.2 ユーザビリティテスト

アメリカでユーザーエクスペリエンスのコンサルティングを行うジェシー・ジェームス・ギャレット氏によると、ユーザーエクスペリエンス設計には戦略・要件・構造・骨格・表層という五つの段階(図 3-3)があり、全ての段階で意識と議論が必要とされている。しかし、ユーザビリティインスペクションがカバーできるのが表面的な部分表層、骨格、構造だけになる。そのため、ユーザビリティテストの導入が必要とされている。

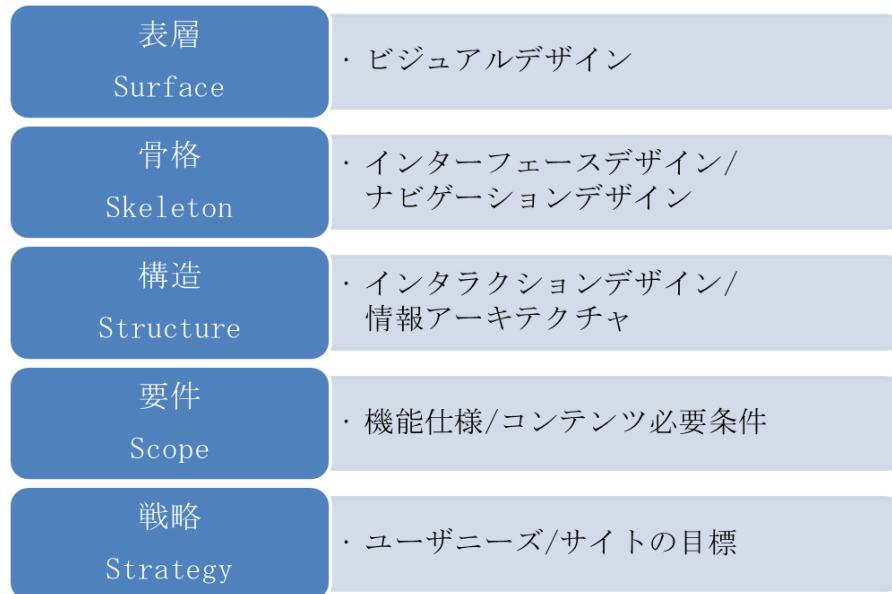


図 3-3 ユーザーエクスペリエンスのダイアグラム

ユーザビリティテストでは、利用者にシステムを実際に利用してもらい、操作の際に話す言葉と操作状況を分析して使いにくさの原因を特定する分析法^[20]を利用した。分析結果の一例は表 3.5 に示す。

表 3.5 発話と操作内容分析表

9月12日 顧客ミーティング13回目				
番号	発話者	発話内容	分類	改善案・対応策
1	経営者1	プルダウンメニュー押しづらい	UI	ノートPCのタッチパネルが原因と考えられ、次回からマウスで操作して頂く
2	経営者1	名前が暗くて見づらい	UI	文字色を白くする。該当配色案を失敗例としてチェックシートに登録する
3	経営者1	基本スケジュールを変更するのと最新スケジュールを変更するのはどう違うのか？	使い方	その場で説明した
4	経営者2	代行ではなく、レギュラーコーチを変えたい場合はどうすればいい？	使い方	その場で説明し、評価フェーズで教育を行う時やマニュアル作成する時も注意する
5	経営者1	代行申請者から代行者を決定する時の流れは？	使い方	アル作成する時も注意する
6	経営者1	メールのタイトルでT1からのメールと分かるようにした方がいい。	UI	すべてのメール件名の最初に【T-1】を追加し、分かりやすくする。該当項目をチェックリストに登録する
7	経営者1	メールを読んでくれたかどうか分かるようにして欲しい	機能	システムが煩雑になり、優先度が高い機能であれば考慮する
8	経営者2	代行者が決まった時の通知は他の立候補してくれたコーチにするべきか	機能	「受入れ基準どおり送る」と回答した
9	経営者1	代行依頼は一斉にメールが送られてしまうのか？	機能	「全部自動で送られる」と回答した
10	経営者1	誰が何回代行を担当してくれたかが分かるようになってたが、上記のような特別な場合は、どうシステムに入れればいいのか？打ち込めるようになっているのか？	機能	「管理者が人事評価の画面から変更できる」と回答した
11	経営者2	遅刻するから前半だけ他のコーチにお願いする時などはどうする？	使い方	その場で説明し、評価フェーズで教育を行う時やマニュアル作成する時も注意する
12	経営者1	初めてコースが1日だけ追加される時はどうするか	使い方	アル作成する時も注意する

開発フェーズでの発話と操作内容の分析結果は表 3.6 に示す。合計 36 件のフィードバックを受けた。

表 3.6 発話と操作内容の分析結果

日付	UI	機能	使い方	合計
8月9日	2	1	0	3
9月12日	3	4	5	12
10月5日	1	2	0	3
10月29日	3	8	3	14
11月9日	0	0	4	4
合計	9	15	12	36

利用者がシステムを利用する経験が少ないため、レビューする際に、機能に関する質問がわりと多かった。また、10月29日にレビューした後、事務員にもシステムのアカウント管理などの関連機能を利用してもらい、フィードバックを受けた。そのため、当日の合計件数がわりと多かった。

我々のチームでは、アジャイル開発^[8]で進めており、各イテレーション^[9]の最後に開発した機能を顧客がレビューしているため、この手法が顧客の意向を反映することに適合していると判断した。利用者による評価と改善を繰り返すことにより、顧客の意向を反映したシステム化に向けて収束させていった。

3.3 考察

本章では、主に開発フェーズでユーザビリティの評価手法とそれによるユーザビリティの向上について述べた。プロジェクトの最初から最後まで、ユーザビリティを向上するために工夫すべき点があると考ええる。以下、戦略立案フェーズ、要件定義フェーズと評価フェーズでユーザビリティを向上するために工夫したことについて述べる。

戦略立案フェーズで対象業務と選定と利用者の意識の理解が重要である..対象業務を選定するために、顧客と一緒に環境分析などを行った。利用者を把握するために、コーチや事務員へのインタビュー、レッスン見学などを行った。また、週に1回顧客とコミュニケーションを取り、マインドマップと使って顧客の想いを聞き出すことを行った。そうすることで、開発者が利用シーンを想像でき、利用者の立場からシステム機能を考えることができる。

要件定義フェーズで、システム要件を洗い出し、定義することが大切である。それを達成するために、モックアップやプロトタイプを作成し、顧客にレビューしてもらった。さらに、データベースやモダルなどの専門用語を使わない、ユーザストーリーという顧客視点そうすることで、顧客はシステムの基本機能、画面遷移、基本レイアウト等を確認することができた。

評価フェーズで、利用状況の把握が大事だと思う。本プロジェクトでは、まず利用状況と利用品質の継続的測定を提案した。また、利用者視点に基づき、講習会とマニュアル作成などの支援活動を実施した。さらに、未利用者を対象としたアンケート調査を行い、システムの利用拡大に向けた課題を整理した。そうすることで、システムの利用状況を把握でき、利用者の意見を集め、ユーザビリティ向上の検討に役に立った。

以上のことから、ユーザビリティの向上はプロジェクト全過程と関わることが分かった。ユーザビリティに関して、戦略立案フェーズの目的は、どこに、どのような問題があるか発見することである。要件定義フェーズと開発フェーズの目的は戦略どおりに問題を解決することである。評価フェーズの目的は戦略どおりに問題が解決されているかを確認することである。各フェーズの目的と合わせて工夫すべきだと考えられる。

第4章 リスクマネジメントの実践

4.1 リスクマネジメントの目的

本プロジェクトは筑波大学の特定課題研究として実施され、明確な期日が決められている。従ってこの期日以内にプロジェクトとしての成果を生み出すことは要件の一つであり、納期を超えることは許されない。そのため、プロジェクトにマイナスの影響を及ぼす可能性があるものをうまく対応して、影響を軽減する必要がある。失敗プロジェクトを回避するために、筆者はリスクマネジメントを担当した。リスクマネジメントの実施に関して、次節より説明する。

4.2 リスクマネジメント計画

本プロジェクトでは、リスクマネジメントを実践する際に、米国 PMI (Project Management Institute) の PMBOK (Project Management Body of Knowledge) で提示されているリスクマネジメントフロー(図 4-1)^[1]に準拠して行った。PMBOK はプロジェクトマネジメントフレームワークのデファクトスタンダードである。

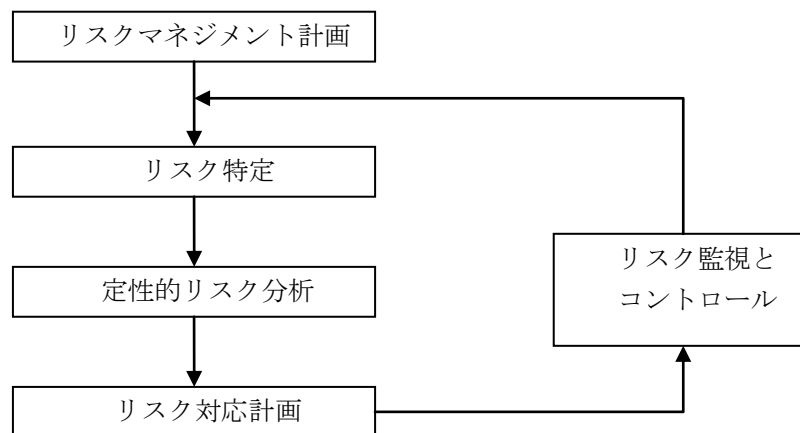


図 4-1 リスクマネジメントフロー

筆者の役割はプロジェクトに対して、リスクマネジメントフローにしたがったマネジメントプロセスの推進とそこから得られた知見を提言することである。プロジェクトの目標を阻害させる要因となるマイナスのリスク（以下、単にリスクと呼ぶ）を特定し、定性的リスク評価を行い、高いリスクスコアを持つリスクに対してリスク対応策を立案し、定期的にリスクスコアを監視しながらプロジェクトを成功に導くべく努力した。

リスクの影響度と発生確率の評価基準を図 4-2 に示す。なお、リスクの総合評価リスクスコアは「該当リスクの発生確率」×「該当リスクの影響度」で算出する

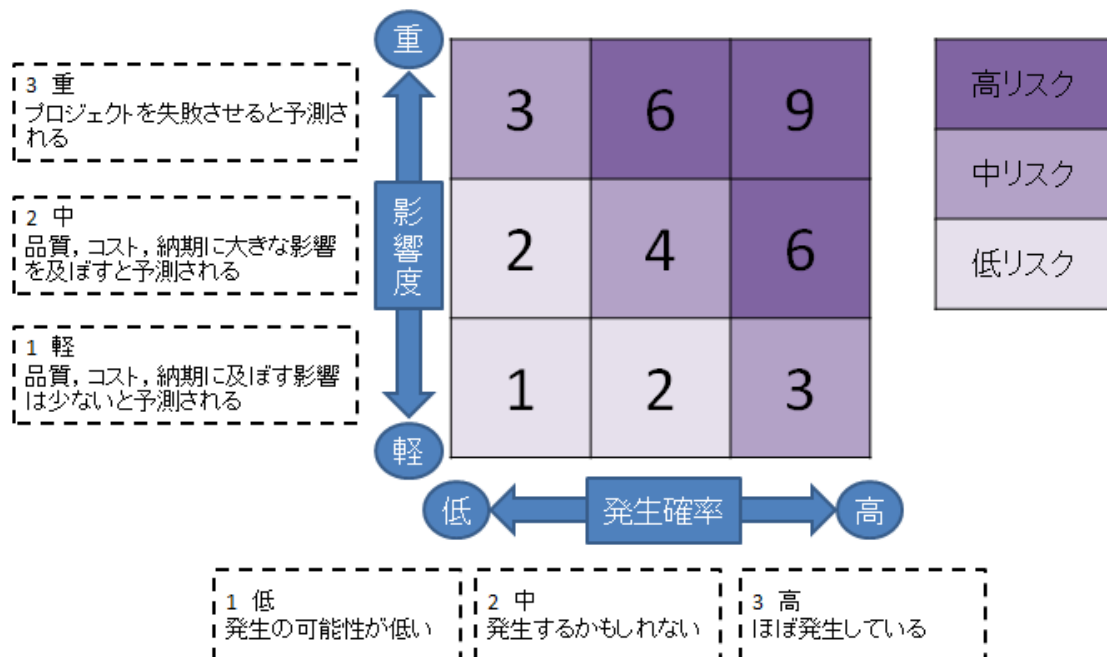


図 4-2 リスクの影響度と発生確率のマトリックス

4.3 リスク特定

予想されるリスクを洗い出すために、チーム内で問題解決手法 TTW「シンク・タンク・ウィンドウズ」(図 4-3)を用いて、ブレインストーミングを行った、漏れなくリスクを洗い出すために、リスク区分やリスクリストなどを提示した。

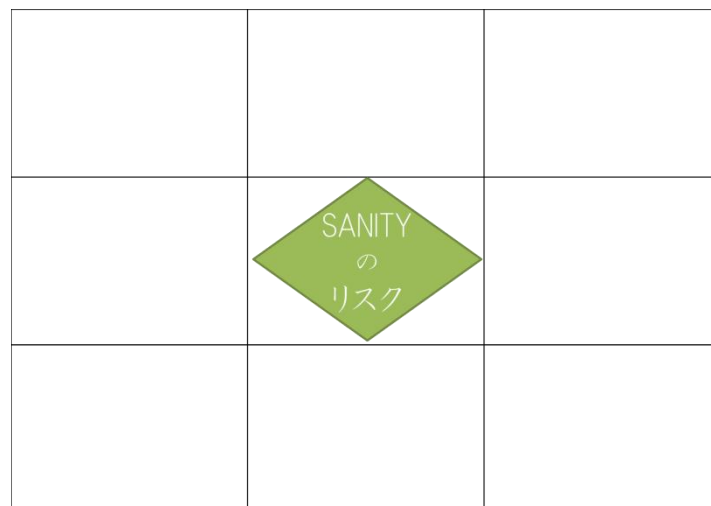


図 4-3 TTW を用いたリスク特定シート

TTW では、3×3 のマス目の中心に解決したいテーマ「チーム SANITY のリスク」を書き、さらに周辺の 8 マスにテーマから連想するキーワードを書いた。12 文字以内のキーワードを短期集中的に書くことで、論理的思考だけでなく、直感思考も併用して、リスクの洗い出しに役に立った。

その後、KJ法を使って整理し、図 4-4 の特性要因図にまとめた。特性要因図を利用することで、考えられる全てのリスクを一枚の用紙に書き出し、分類と体系化をすることができた。また、全体の構造を俯瞰できるので、重要な要因を推測することも可能となった。

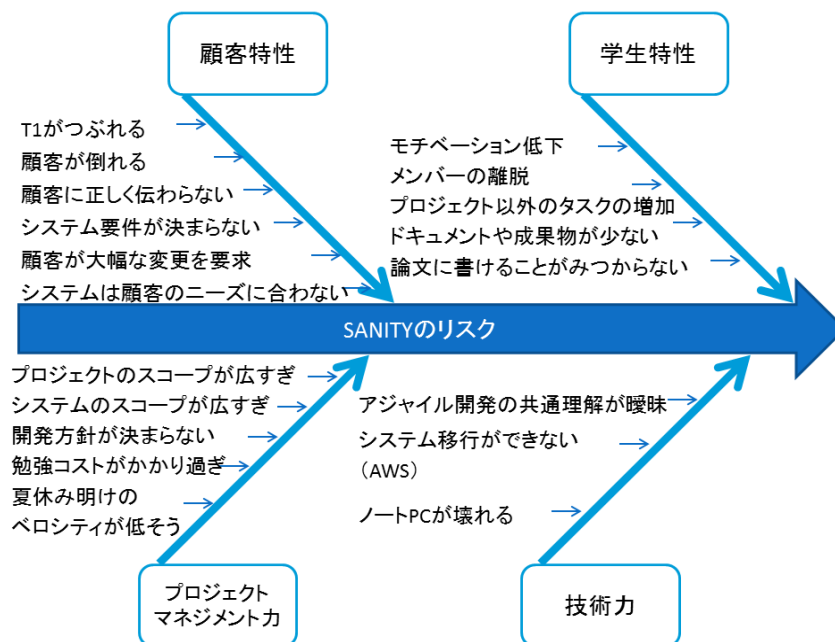


図 4-4 リスクの特性要因図

4.4 定性的リスク分析

すべてのリスクに対して、チームメンバがそれぞれ発生確率と影響度を評価した。その後、違いが大きいリスクを中心に検討した。考え不足や記入ミスによる違いを修正し、楽観的や悲観的な考え方による違いを保留した。続いて、リスクスコアを算出し、「5」以上のリスクを重点監視リスクとした。リスク分析シートを表 4.1 に示す。表の中の「S・A・N・I・T」は各チームメンバの名前の頭文字で、その列の値は該当チームメンバに記入された値である。「AVE」は平均値を表す。リスクスコアはリスク発生確率の平均値と影響度の平均値の積である。

表 4.1 リスク分析シート

番号	リスク事象	発生確率						影響度						リスク スコア
		AVE	S	A	N	I	T	AVE	S	A	N	I	T	
R4	夏休み明けのベロシティが低いので、作業効率が下がる	2.8	2	3	3	3	3	2.0	2	1	2	2	3	5.6
R15	プロジェクトのスコープが広すぎ、手が回らなくなる	2.2	2	2	2	2	3	2.4	2	1	3	3	3	5.3
R21	7月27日までシステム要件が決まらない	2.2	2	2	2	2	3	2.4	3	1	3	2	3	5.3
R9	10月まで論文に書けることがみつからない	2.0	2	1	2	2	3	2.6	2	2	3	3	3	5.2
R12	開発方針が決まらない	2.0	2	1	2	2	3	2.6	2	2	3	3	3	5.2
R16	システムのスコープが広すぎ、手が回らなくなる	1.8	2	1	2	2	2	2.8	3	2	3	3	3	5.0
R14	勉強コストがかかり過ぎ、期限内担当タスクを完成できない	2.2	2	2	2	2	3	2.2	3	1	3	2	2	4.8
R7	ノートPCが壊れる	1.6	2	1	1	2	2	2.8	3	3	3	2	3	4.5
R1	モチベーション低下による作業効率が低い	2.0	2	2	2	2	2	2.2	2	2	2	2	3	4.4
R10	ドキュメントや成果物が少なく、卒業要件を満たさない	2.0	2	1	3	2	2	2.0	2	2	1	2	3	4.0
R3	メンバーの離脱による作業が進まない	1.6	1	2	2	2	1	2.4	2	2	3	3	2	3.8
R20	提供したシステムは顧客のニーズに合わない	1.6	1	1	2	2	2	2.4	3	2	3	2	2	3.8
R11	システム移行ができない (AWS)	1.6	2	1	1	2	2	2.2	3	2	2	3	1	3.5
R6	地震で大学が壊れる	1.0	1	1	1	1	1	3.0	3	3	3	3	3	3.0
R17	T1がつぶれる	1.0	1	1	1	1	1	3.0	3	3	3	3	3	3.0
R18	顧客が倒れる	1.0	1	1	1	1	1	3.0	3	3	3	3	3	3.0
R5	プロジェクト以外のタスクが増える	1.8	2	1	3	2	1	1.6	1	1	2	2	2	2.9
R19	顧客が大幅な変更を要求	1.0	1	1	1	1	1	2.8	3	2	3	3	3	2.8
R22	顧客に正しく伝わらない	2.0	1	2	2	2	3	1.4	1	1	1	2	2	2.8
R8	高度ITサーバが止まる	1.0	1	1	1	1	1	2.6	3	3	2	2	3	2.6

4.5 リスク対応計画

リスク特定のプロセスで洗い出したリスクには、高確率で発生しそうなものや、発生確率の低いもの、発生したとしても大した影響を与えないものまでさまざまなものが含まれている。洗い出したリスクをすべて対応するのが難しいため、リスクスコアを評価の尺度として、リスクの種類に応じて対策を講じる必要がある。リスク分析プロセスで評価した発生確率と影響度は、リスクマネジメント計画プロセスで定義したリスクの発生確率と影響度のマトリックスと照らし合わせ、高リスク、中リスク、低リスクに分類した。

リスクの対応策は、リスクの発生を抑えるための「予防策」と、リスクが発生した場合の「発生時対策」の2種類になる。リスクの種類と対応策の関係を表 4.2 に示す。

表 4.2 リスクのレベルと対応策の関係

リスク種類	対応策
超高リスク	予防策と発生時対策を立て、監視を続ける
高リスク	予防策と発生時対策を立て、監視を続ける
中リスク	発生時対策を立て、監視を続ける
低リスク	特に対応策は立てずに、監視を続ける

これまでのプロセスで、リスクを詳細に分析し、発生確率、影響度、対応策などを決めてきた。それらの活動の結果はすべてリスク登録簿(表 4.3, 表 4.4)に登録した。リスク登録簿はリスク管理台帳として常に更新され、リスクマネジメントに活用されていた。

表 4.3 リスク登録簿 1

番号	リスク事象	発生確率(0,1,2,3)						影響度(1,2,3)						リスクスコア
		S	A	N	I	T	平均値	S	A	N	I	T	平均値	
1	システム移行ができない(AWS)	2	2	2	3	2	2.2	2	3	3	3	2	2.6	5.7
2	コートレビューのタイミングが遅い	2	1	2	2	1	1.6	2	1	2	2	2	1.8	2.9
3	プロジェクト以外のタスクが増加する	1	1	1	1	1	1.0	2	1	2	2	2	1.8	1.8
4	システムは顧客のニーズに合わない	1	1	2	2	1	1.4	3	3	3	2	2	2.6	3.6
5	プロジェクトのスコープが広すぎ	1	1	1	1	1	1.0	2	2	2	2	2	2.0	2.0
6	メンバーの離脱	2	3	2	2	1	2.0	2	2	2	2	2	2.0	4.0
7	システムのスコープが広すぎ	1	1	1	1	1	1.0	2	2	2	2	3	2.2	2.2
8	顧客が大幅な変更を要求する	1	1	1	1	1	1.0	3	2	2	2	3	2.4	2.4
9	チーム内のコミュニケーションが少ない	1	1	1	1	1	1.0	2	2	2	2	2	2.0	2.0
10	コーチとのコミュニケーション不足	3	3	3	2	3	2.8	2	2	2	2	2	2.0	5.6
11	事務員とのコミュニケーション不足	2	2	2	2	3	2.2	2	2	2	2	2	2.0	4.4
12	評価フェーズの作業見通しが立てていない	2	2	2	2	2	2.0	3	2	3	3	3	2.8	5.6
13	1/16まで報告書を完成できない	1	1	1	1	1	1.0	3	3	3	3	3	3.0	3.0
14	1/28まで発表の準備ができない	1	1	1	1	1	1.0	3	3	3	3	3	3.0	3.0

表 4.4 リスク登録簿 2

番号	リスク事象	予防策	発生時対策	リスク	担当者
				スコア	
1	システム移行ができない (AWS)	調査を行う イテレーション4で対策する 1回あげてみる AWSでも対応できるようにする	他のプラットフォームを探す	5.7	井原
10	コーチとのコミュニケーション不足	学校でやる (松山君と連絡) 次回の顧客ミーティングでやる	イテレーション4でやる	5.6	永井
12	評価フェーズの作業見通しが立てていない	イテレーション4でやる 担当者を決めて、計画を立てる 業務プロセスを明確にする	すぐとりかかる	5.6	有田
11	事務員とのコミュニケーション不足	次回の顧客ミーティングでやる 事前にメールで伝える	-	4.4	白田
6	メンバーの離脱	みんなで作業を分担する	他の人やる 業務フローを決める	4.0	全員

4.6 リスク監視とコントロール

リスク監視とコントロールはチームのイテレーション期間と合わせて、約2週間毎に行なった。リスクマネジメント作業フローを図4-5に示す。まず、リスクマネジメント担当の筆者がリスク項目の増減を提案した。前回、3人以上が発生確率を0と判定したリスクを削除し、KPTのProblemから新しいリスクを抽出した。次に、チームメンバ全員が参加したリスクマネジメントミーティングを行った。ミーティングでは、前回の振り返り、リスク項目の確認、リスクの分析、リスク対応策の検討などを行った。その後、筆者がミーティングの成果を整理し、リスク登録簿を更新した。さらに、各プロセスの改善点の考慮、次回ミーティングの準備などを行った。

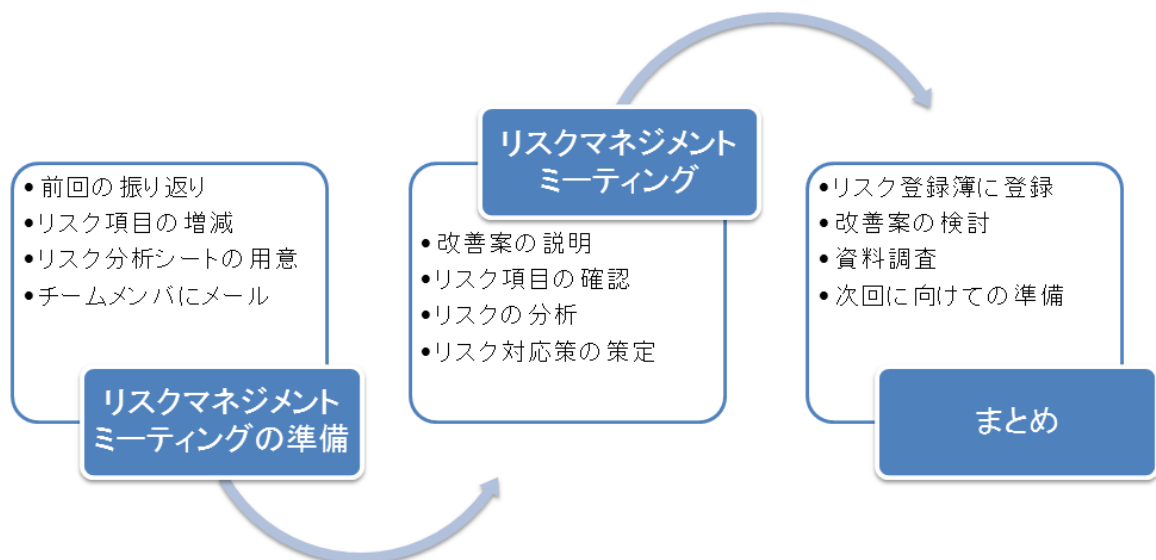


図 4-5 各イテレーションのリスクマネジメント作業フロー

本プロジェクトでは、全部 36 個のリスクを監視していた。リスクスコアの監視状況は図 4-6 に示す。以下、要件定義フェーズでリスクスコアが一番高かったリスク R12 と開発フェーズで重要なリスク R24 の監視状況について報告する。

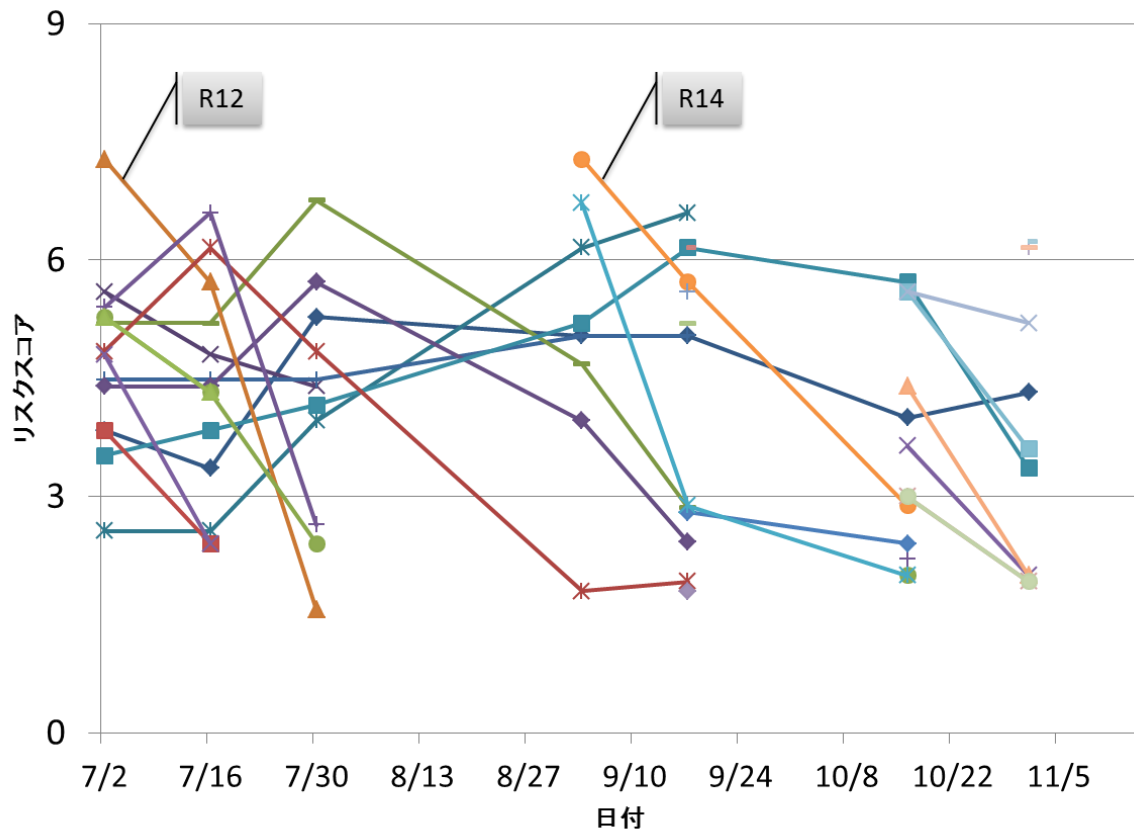


図 4-6 リスクスコア監視グラフ

まず、一つ目は「R12：開発フェーズに入るまで開発方針が決まらない」というリスクの状況を述べる。これは要件定義フェーズで発生したリスクである。その予防策として、科目「PBL 型システム開発」で経験ウォーターフォール型開発とアジャイル開発のメリットとデメリットを検討し、チームプロジェクトに合う開発手法を検討すると決めた。その結果、顧客はシステム発注経験がないなどの現状を考慮した上で、変更要求に柔軟に対応できるアジャイル開発手法を採用することにした。図 4-6 から、7 月 30 日のリスクスコアが低くなったことが分かる。対応策を立てることでリスクの発生を防止した。

二つ目は「R14：開発フェーズでレビューのタイミングが遅いことにより、品質が下がる」というリスクの状況を述べる。これはイテレーション 1 の後半で発生したリスクである。あるチームメンバーがレビューする際に、スキーマの設計に問題があるということが指摘された。しかし、締め切りが迫ってきて、できた実装コードを作り直す時間がなかった。そこで、上記のリスクを登録し、イテレーション 2 から設計段階からレビューするという発生時対策を立てた。図 4-6 から、イテレーション 1 からイテレーション 2 まで、リスクスコアが下がっていることが分かった。予想外のリスクが発生した際に、リスクとして登録し、発生時対策を立てることで、リスクの影響を軽減できた。

4.7 考察

本プロジェクトでリスクマネジメントを行うことで、19 個リスクの予防、と 8 個リスクの軽減することができた。本プロジェクトで採用したリスクマネジメントプロセスが有効であるという結論を得た。次に、リスクマネジメントを実践する際に、重要だと感じたことを三つ述べる。

(1) コミュニケーションマネジメントへの気配り

リスクマネジメントのプロセスには、チームメンバ全員参加すべきである。また、各チームメンバが自由に意見を言える環境を作るべきである。本プロジェクトでは、基本的にイテレーションの初日にチームメンバが集中作業する部屋にミーティングを行った。病気や用事で全員が揃わない場合は、Skype でミーティングをやるか、ミーティングの時間をずらすなどの手段で対応した。そして、KJ 法、ブレインストーミングなどの手法を用いて、各チームメンバの意見を聞き出すのに工夫した。

(2) プロジェクトの規模に応じたリスクマネジメントへの取り組み

リスクマネジメントのコストは、プロジェクトにとっての価値に見合ったものにするべきである。そのため、リスク分析手法、実施頻度、実施時間などは、プロジェクトの規模やチームの状況に適合させるべきである。本プロジェクトでは、プロジェクトの規模とチームメンバへの負担を考慮した上で、リスクの発生確率と影響度に対して 3 段階評価を採用した。プロジェクトの規模と保有工数を考慮した上で、リスク発生確率・影響度マトリックスに基づいたリスク分析手法を採用した。さらに、プロジェクトの期間とスコープを考慮した上で、約 2 週間 1 回、1 回 1 時間の頻度で行うことにした。

当初リスクマネジメントに投資する時間を、リスクマネジメント担当 10%、他チームメンバ 2%と予定した。実際に、リスク分析する際に、Google Docs を使うことで、チーム全員が共同作業でき、紙ベースやローカルで行うより効率が上がった。リスクマネジメントの作業時間は、リスクマネジメント担当は約 6%、他チームメンバは約 1.5%を投資した。この投資時間の割合は妥当であるという結論を得た。

(3) それぞれのマネジメントプロセスの連動

リスクマネジメントの成功には、他のマネジメントのプロセスの的確な実行が必要である。本プロジェクトでは、コミュニケーション、進捗、スコープ、品質、リスク 5 つのマネジメントを実施した。一つのマネジメントは、その他のマネジメントのプロセスから分離し孤立して存在することはない。例えば、顧客が要求変更をした際に、その要求変更を受け入れるかどうかの判断は、各マネジメント担当がそれぞれの視点から検討してから決めた。その作業のプロセスを図 4-7 に示す。

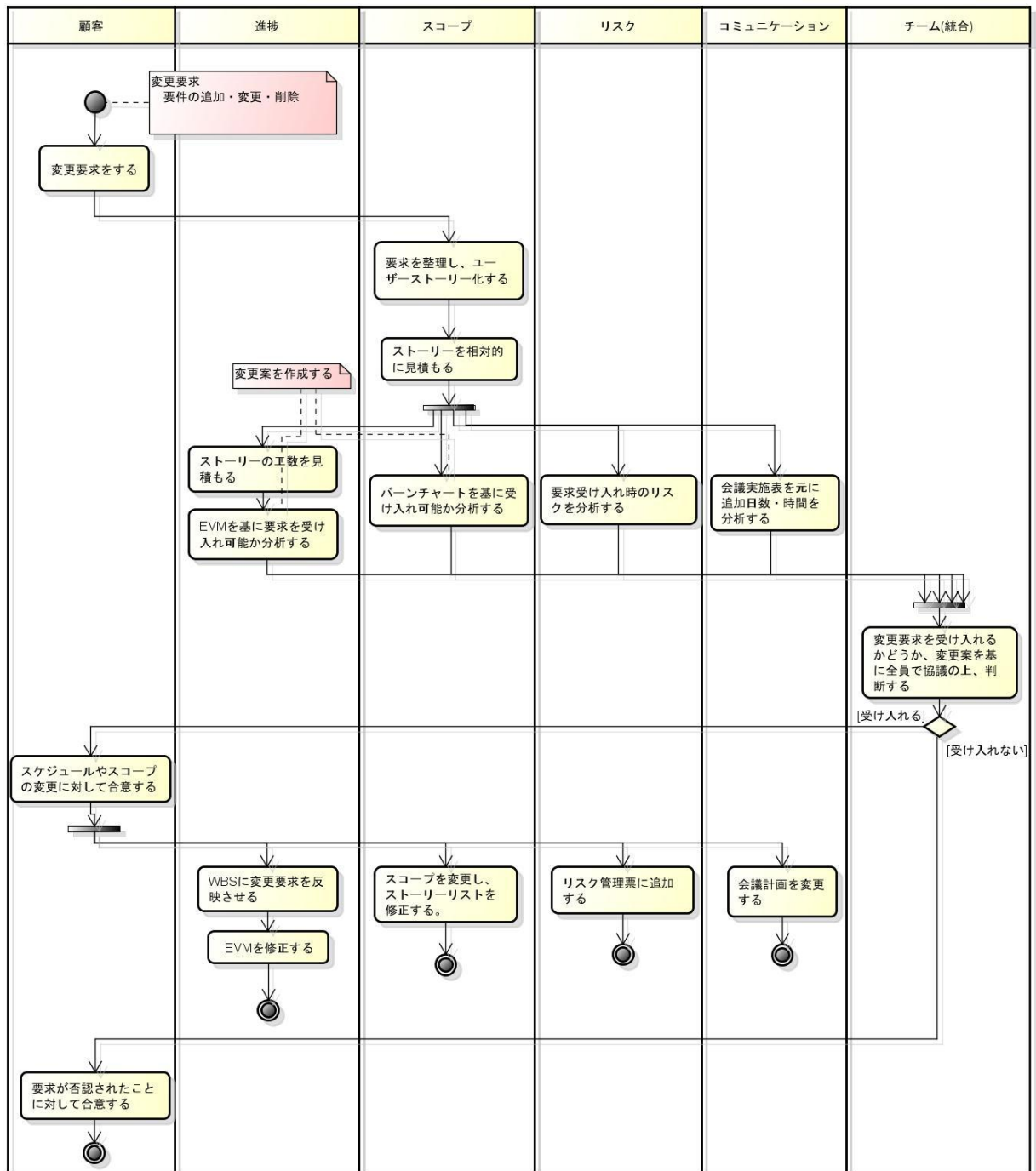


図 4-7 要求変更時のマネジメントプロセス

第5章 振り返り

5.1 プロジェクトの振り返り

(1) 密なコミュニケーション

本プロジェクトの基本方針は顧客と密なコミュニケーションを取ることである。実績として、図 5-1 に示したように合計 18 回 175 人時の顧客ミーティングを行った。棒グラフが実際に T1 に伺い、顧客とミーティングした時間を表している。グラフ下の人時表記はミーティング時間に参加人数を掛けた数値である。各フェーズのグラフを見てみると、ほぼ週に一回の高頻度で顧客とコミュニケーションをしていた。開発フェーズでも、イテレーション期間を合わせて、ほぼ二週間一回に顧客にできたシステムを見せて、フィードバックをもらっていた。特に事情がなければ、チームメンバ全員が出席していた。戦略立案フェーズでは、課題担当教員も同席していた。評価結果によると、顧客のニーズに近い、業務改善に役に立つ IT ソリューションを提供することができた。密なコミュニケーションの効果があった。

しかし、2 時間の顧客ミーティングを行うために、準備時間、移動時間、ミーティング時間とまとめ時間を合わせて、平均 30 人時かかった。そこで疑問に思うのは 1 回の顧客ミーティングで 30 人時に相当する成果が出せるか。参加人数やミーティングの内容について改善できるところがあると考える。

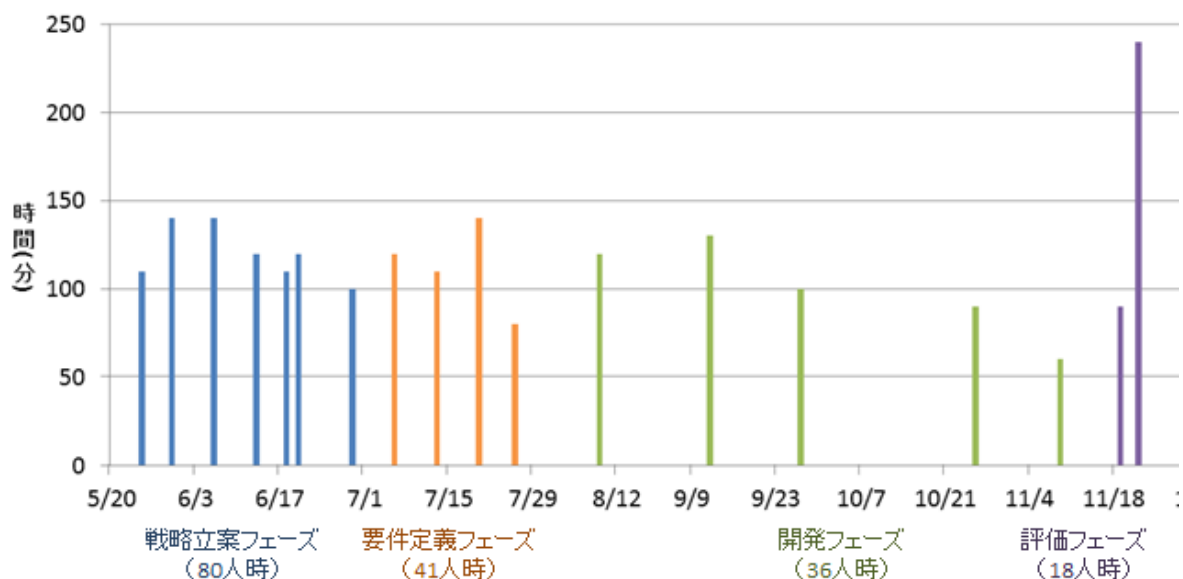


図 5-1 顧客ミーティング時間の統計

(2) ツールの利用による効率向上

本プロジェクトでは、積極的にツールを利用した。

easy step(図 5-2)を利用することで、顧客と共に、T1 の内部状況、現状問題点、将来像などの分析を行った。様々の意見を随時に記録でき、後で振り返る時分かりやすく、便利だった。

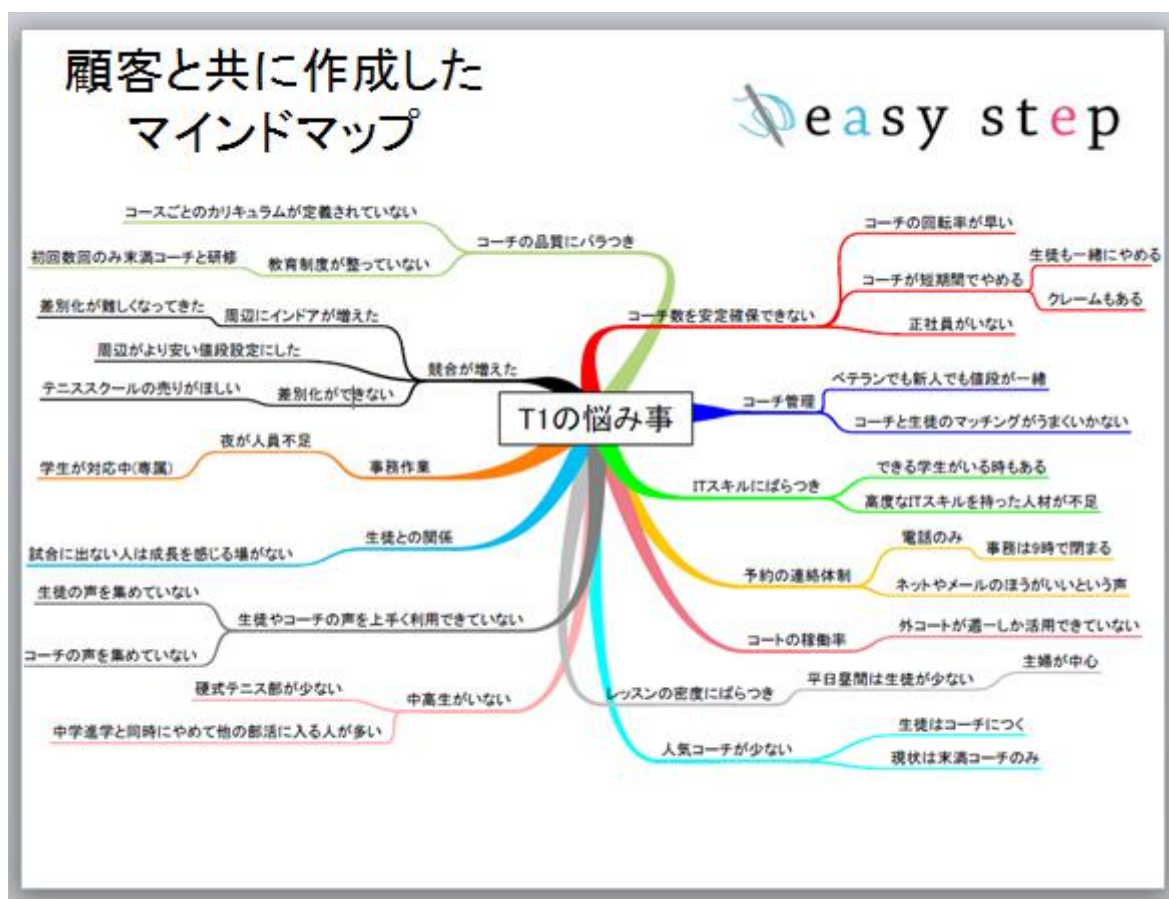


図 5-2 easy step で顧客と共に作成したマインドマップ

Cacoo(図 5-3)を利用することで、Web 上で図を共有したり、共同編集したりすることができた。Cacoo は Web 上で簡単にサイトマップやワイヤーフレーム、UML、ネットワーク図などの図が作成できるドローツールである。みんなの意見が可視化され、効率よく議論することができた。

について報告するようにした。そうすることで、作業が詰まっているメンバが他メンバから支援を受け、問題解決の時間が短縮した。また、問題解決方法を共有することで、同じ問題に時間がかからないようになった。以上のことから、コロケーション戦略の採用とディリーミーティングの設置は有効だった。

(2) タックマンモデルによるチームの分析

タックマンモデル^[26]を用いて、チームの振り返りについて述べる。タックマンモデルとは、チームビルディング(組織進化)モデルの 5 段階を指す。心理学者のタックマンが唱えたモデルである。チームは形成されただけで機能し始めることはない。チームは形成後、混乱を経て、期待通り機能するようになる。



図 5-4 タックマンモデル

本プロジェクトのチームメンバは、2011 年度の授業科目「PBL 型システム開発」を受け、お互いにある程度知っているため、成立期が短くて、1 週間くらいだった。その後、どの IT ソリューションを提供するか、ウォータフォール型開発手法とアジャイル開発手法のどちらを採用するかなどの問題で、各チームメンバの考えが違った。早く動乱期を通過するために、チームミーティングで考え方を共有する、チームが求めるものを検討するなどの対策をした。開発フェーズから安定期に入り、チームの状態が安定した。イテレーションの KPT により、チーム規範や作業方法を少しずつ改善し、チームとして機能を始め、機能期に入った。以上のことから、本プロジェクトを遂行することにより、チームとして順調に成長できた。

5.3 個人の振り返り

(1) 使い慣れた技術と新しい技術の戦い

本プロジェクトが開発フェーズに入る前に、実装言語は PHP にするか、それとも Ruby にするかという議論があった。PHP を利用する理由は、納期を重視する観点から、チームメンバ 5 人中の 4 人が経験者で、2011 年度の授業科目「PBL 型システム開発」での実装経験も利用できるからである。Ruby を利用する理由は、生産性が高いと言われた Ruby on Rails を利用できるからである。Ruby on Rails は Web アプリケーション作成のためのフレームワークである。

筆者は最初、リスクが比較的に低い PHP に賛成していたが、チームメンバの検討により、Ruby に決定した。その後、心配しながら、Ruby と Ruby on Rails の勉強を始め、「Ruby on Rails 3 アプリケーションプログラミング」「Rails によるアジャイル Web アプリケーション開発 第 4 版」「Rails3 レシピブック 190 の技」などの書籍を参考にした。開発フェーズのイテレーション 2 でシステムの代行リスト機能の実装などを担当した。一見学習コストがかかりそうだが、いろんなメリットが見つかった。要求変更に対応できた。自動リロードにより開発の効率を上げた。命名規則など Web アプリケーション開発の良い習慣も身につけた。新しい技術に挑戦して良かったと感じた。

(2) 研究のためのプロジェクト開発

本プロジェクトは顧客のニーズを聞き出し、IT ソリューションを提供すると共に、研究と

いう目的も忘れてはいけない。ただ機能を実現するだけでなく、どのように実現したのか、なぜその技術を選んだのか、結果はどうなったのかも十分に検討すべきである。また、PDCA サイクル(plan-do-check-act cycle)を回し、継続的に改善すべきである。筆者は本プロジェクトに参加することで、チームワーク、プロジェクトマネジメント、技術学習などの面で、個人として成長できた。

5.4 終わりに

本プロジェクトは、つくば市内のあるテニススクールを顧客として、経営改革のため、まず内部基盤に着目して改革を行い、満足度向上のための改革などは次年度以降に行うという戦略案を提案した。さらに、本年度の経営改革を支援するために、従業員を対象にした業務プロセス改善と IT ソリューションを提案した。提案した IT ソリューションはコーチへの評価を目的として、要件定義から、開発、納品、評価まで実施した。筆者はシステム開発、ユーザビリティ向上、リスクマネジメントを担当している。

筆者が担当したユーザビリティの部分で、Nielsen によるユーザビリティの定義を参考し、定量的評価指標を策定した。また、「ユーザビリティチェックリスト」と「発話と操作内容分析表」二つの評価手法を通じ、ユーザビリティの向上に努めた。さらに、評価フェーズで測定、アンケートなどによる評価結果から、ユーザビリティを確保できたという結論を得た。

筆者が担当したリスクマネジメントの部分で、PMBOK のリスクマネジメントフロー^[1]にしたがったマネジメントプロセスを推進した。マネジメントプロセスを実行しながら、少しずつ改善した。コミュニケーションマネジメントへの気配り、プロジェクトの規模に応じたリスクマネジメントへの取り組み、それぞれのマネジメントプロセスの連動の重要性を認識した。

謝辞

本報告書を執筆するにあたり，御指導，御協力頂いた皆様に対し，深く感謝の意を示します．指導教員の三末和男准教授は，プロジェクトや生活において多くの励ましを頂き，プロジェクトを進める上で大きな励みとなりました．ここに深く感謝致します．

プロジェクトの遂行にあたり，課題担当教員の山戸昭三教授は，御多忙の中，平日休日を問わず，熱心な御指導，御助言を頂きました．ここに深く感謝致します．

本プロジェクトの顧客となって頂いた T1 インドアテニススクールの末満様には，御多忙の中，戦略立案から評価まで御親身に御協力頂きました．ここに深く感謝致します．

本プロジェクトのチームメンバである白田良太氏，有田正信氏，永井達也氏，井原淳平氏には，本プロジェクト活動から生活に至るまで，広くお世話になりました．ここに深く感謝致します．

筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻高度 IT 人材育成のための実践的ソフトウェア開発専修プログラムの常勤教員である中沢研也教授には，御多忙の中，熱心な御指導，御助言を頂きました．ここに深く感謝致します．また，筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻高度 IT 人材育成のための実践的ソフトウェア開発専修プログラムの担当事務員である木村直美様には，本プロジェクト生活全般において大変お世話になりました．ここに深く感謝致します．その他，筑波大学大学院システム情報工学研究科コンピュータサイエンス専攻に在籍される皆様をはじめ，本当に多くの方々にお世話になりました．ここに深く感謝致します．

最後になりましたが，私を暖かく見守って下さいました両親並びに周囲の皆様に深く感謝致します．

2013 年 3 月 杜婧雨

参考文献

- [1] Project Management Institute:「プロジェクトマネジメント知識体系ガイド(PMBOK ガイド)第4版」, Project Management Institute, 2008.
- [2] IT コーディネータ協会 IT 経営研究所:「IT コーディネータプロセスガイドライン Ver2.0」, IT コーディネータ協会, 2011.
- [3] 手塚貞治:「戦略フレームワークの思考法」, 日本実業出版社, 2008.
- [4] IT ガバナンス協会:「IT ガバナンス導入ガイド第2版」, 2007.
- [5] 樋田大二郎, 櫻井茂男, 沓澤糸:「第4回子育て生活基本調査(小中版)」, Benesse 教育研究開発センター, pp.76-79, 2011.
- [6] 茨城県テニス協会:「茨城県中学校テニス連盟の茨城県中学校体育連盟への正式加盟支援について」, <http://www.ibta-tennis.jp/blog/jimu/2012/06/post-41.html>(access:28th.Sep.2012).
- [7] IT コーディネータ協会 IT 経営研究所:「IT コーディネータ実務ガイド Ver1.0」, IT コーディネータ協会, 2011.
- [8] Jonathan Rasmusson:「アジャイルサムライ」, 西村直人(監), オーム社, 2011.
- [9] Mike Cohn:「アジャイルな見積りと計画づくり」, マイナビ, 2009.
- [10] 山田祥寛:「Ruby on Rails 3 アプリケーションプログラミング」, 技術評論社, 2011.6.
- [11] 仲川薫, 須田亨:「ウェブサイトユーザビリティアンケート評価手法の開発」, 2004.
- [12] J. Nielsen: “Usability Engineering”, Academic Press(1993).
- [13] ISO 9241: Ergonomic requirements for office work with visual display terminals(VDTs) – Part 11: Guidance on usability(1998).
- [14] 岡本誠:「人間中心設計とユーザシナリオの活用」, デザイン学研究特集号, Vol.18-2, No.70, 2011.
- [15] 田平博嗣:「製品・システム・サービスの人間中心設計プロセス」, 電子情報通信学会誌, Vol.94, No.9, 2011.
- [16] 岡田英彦:「ユーザビリティとその評価手法」, システム/制御/情報, Vol.45, No.5, pp. 269-276, 2001.
- [17] J. Nielsen and R. L. Mack (Eds.): “Usability Inspection Methods”, John Wiley & Sons(1994).
- [18] J.S.Dumas and J.C Redish: “A Practical Guide to Usability Testing”, Ablex Publishing(1993).
- [19] 武井遠藤:「ユーザー中心ウェブサイト戦略,(株)ビービット」, 2006.
- [20] 池谷義紀:「WEB デザインユーザビリティ」, 2003.
- [21] 郷健太郎:「HCD シナリオのデザインヒューリスティクス」, HIS 研究報告集 Vol.9, No.4, p1-6.
- [22] 日本プロジェクトマネジメント協会:「IT プロジェクト実践リスクマネジメント・ガイドブック-II ～疑問に答える～」, 2010.9.
- [23] 日本プロジェクトマネジメント協会:「IT プロジェクト実践リスクマネジメント・ガイドブック」, 2007.8.
- [24] 日経 SYSTEMS:「プロマネやっではいけない(計画・管理編)」, 2011.12.
- [25] 岡村正司:「IT プロジェクトを成功に導くリスク・マネジメント大全」, 日経 BP 社, 2008.3.
- [26] 高木晴夫:「人的資源マネジメント戦略」, 慶応義塾大学ビジネス, 2004.

付録

番号	資料
A	戦略立案フェーズ
A1	マインドマップ
A2	経営者の想い
A3	SWOT分析
A4	戦略ロードマップ
A5	BSC
A6	生徒年齢分析図
A7	入会年度分布図
A8	レッスン台帳の分析
A9	生徒向けアンケートの分析
A10	商圈分析
A11	提案したシステム案
B	要件定義フェーズ
B1	インセプションデッキ
B2	ユーザストーリー
B3	システム画面イメージ図
B4	勤務評価シート
B5	業務フロー図
B6	変更管理フロー図
C	開発フェーズ
C1	画面遷移図
C2	クラス図・ER図
C3	利用者発話と操作内容分析表
D	評価フェーズ
D1	クイックマニュアル
D2	運用マニュアル
D3	利用者向けアンケート
E	その他
E1	ミーティング案
E2	顧客ミーティングの議事録
E3	EVM

※個人情報を保護するため、一部黒塗りをしています。