

15-A-8

大規模システム開発プロジェクトにおける ユーザーエクスペリエンス品質設計プロセスの 適用とその効果¹

1. 概要

本編では、大規模なシステム開発プロジェクトで UX (User eXperience) 品質設計プロセスを実践的に成功させた NEC ソリューションイノベータ株式会社の事例を紹介する。また、同社の UX 適用効果や、その後の運用についても紹介する。

顧客要求の高度化に伴い、システムは複雑になり、使いやすくわかりやすい操作が求められている。同社でも、スマートデバイスなどコンシューマ製品の業務浸透による顧客経験値向上が確認され、SaaS²などクラウド利用に伴う業務サービスが多様化し、結果として機能そのものよりもユーザーが使いたいと思う心地よさが期待されている。

同社では、このようなニーズの変化に対応するため、組織レベルで SI/ソフトウェア開発プロジェクトの改善活動に努めてきた[1]。特に、システムやサービスを人の視点に立ってユーザーの体験価値を高める「ユーザーエクスペリエンス (User eXperience: UX)」の考え方に注目し、それを大規模なプロジェクトに適用させるための新しい手法および枠組みの研究開発を進めている[2]。

2. 背景および課題

2.1. プロジェクトにおける UX の重要性

UX は、国際規格である ISO 9241-210³でも定義されており、製品・サービスの使い勝手や見栄えだけでなく、利用者の感情や反応までを含めた広い概念である。図 15-A-8-1 のように、いくら多くの機能がシステムに備わっていたとしても、操作の仕方がわからないのであれば、UX は低くなる。逆に、いくら見やすく分かりやすいシステムであっても、求めている機能が無ければ、UX は低くなる。そして、機能が多数揃っており、ユーザビリティにも

¹ 事例提供: NEC ソリューションイノベータ株式会社 イノベーション戦略本部/UX 戦略室
森口 昌和 氏

² Software as a Service

³ Ergonomics of human-system interaction -- Part 210: Human-centred design for interactive systems

優れていたとしても、本質的な業務の利用効率が上がらなければ、UX は低くなる。

すなわち、UX とは、製品・サービスの利用者品質を総合的に評価する基準であり、「ヒット」の目線で抜けや漏れなどのプロジェクト問題を解決するアプローチであると同社は捉えている。

UX は、これまでコンシューマ製品・サービスでは暗黙的に適用されてきた概念だが、業務システムでは、実際に利用するユーザーよりも、コストや機能に依ったプロジェクトが多くなりがちであるため、SI/ソフトウェアの品質設計プロセスに明示的に組み込む必要がある。



図 15-A-8-1 UX の概念図

2.2. プロジェクトで発生する問題の定義

本編では、個人やチーム、ステークホルダなど、「ヒット」の属性で発生する遅延や手戻りを、UX 問題と定義する (図 15-A-8-2)。この問題は、性格や経験の差、組織文化などの主観的な要素が主な要因となる。例えば、開発者によって成果物にバラつきが出たり、顧客から十分なヒアリングができなかったりすることが該当する。

一方、画面や機能など、「モノ」の属性で発生する遅延や手戻りを、UI⁴問題と定義する。この問題は、システムの動作や画面の見た目、わかりやすさなどの客観的な要素が主な要因となる。例えば、パフォーマンスが遅くて処理が終わるまで待たされたり、文字が小さくて見にくかったりすることが該当する。

なお、システムが動作しないといった根本的な問題は、バグと定義する。

⁴ User Interface

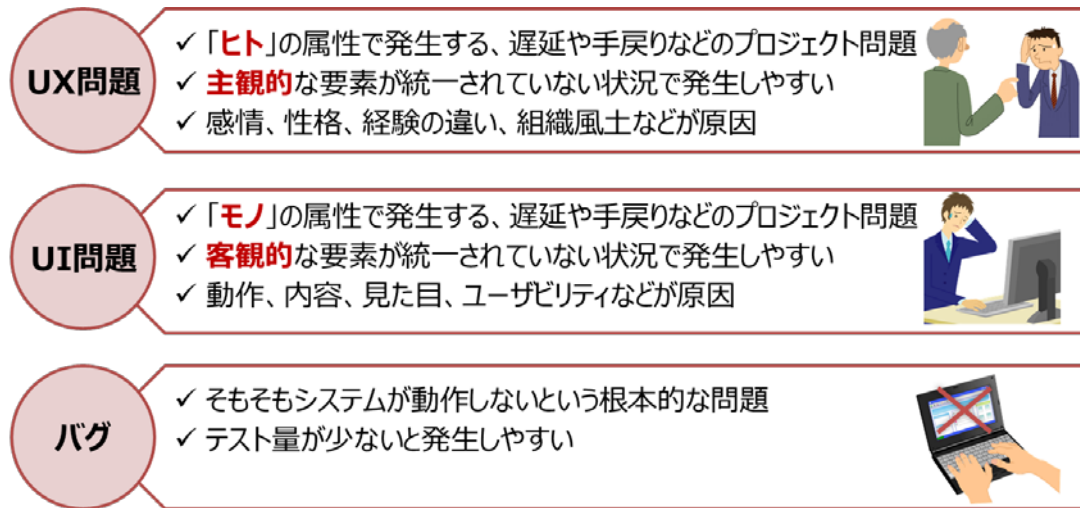


図 15-A-8-2 本編で扱う問題の定義

2.3. プロジェクトの失敗は主に UX 問題が原因

遅延や手戻りなどでプロジェクトがうまく回らない根本的な原因を探るため、これまで取り組んできた20個のプロジェクトの議事録やドキュメントを精査した(表15-A-8-1)。なお、20個のうち、半分以上が通信系・流通系で、BtoBtoC⁵の業務が多くを占めている。

表 15-A-8-1 分析した20個のプロジェクトの内訳

カテゴリ	件数	代表例
通信	6件	営放管理システム、など
流通	4件	POS管理システム、など
基幹	3件	総合運用管理システム、など
共通	3件	勤務管理システム、など
製造	2件	生産管理システム、など
海外	1件	
医療	1件	

20件

上記プロジェクトを精査した結果、合計619件もの問題点が抽出された(図15-A-8-3)。そのうち、前述した定義に従うと、UX問題は342件、UI問題は239件、そしてバグは38件となった。さらに、遅延や手戻りが発生しそうな重大な問題の割合を算出したところ、UX問題は50%であったのに対し、UI問題は37%、そしてバグは39%であった。このことより、

⁵ Business to Business to Consumer

UX 問題のほうが高いリスクを内包していることがわかる。なお、ここでの重大な問題とは、画面や機能が適切に実行できない可能性がある問題であり、一方の軽微な問題とは、ユーザーに迷いを与える可能性がある問題を意味する。

例えば、重大な UX 問題の一例として、システムを使っていて 1 つ前の画面に戻れないという問題があった。これはバグではなく、そもそも 1 つ前の画面に戻るという操作が想定されていなかったという「ヒト」の問題である。このように、後工程で手戻りの原因となる要件定義漏れも重大な UX 問題に該当する。

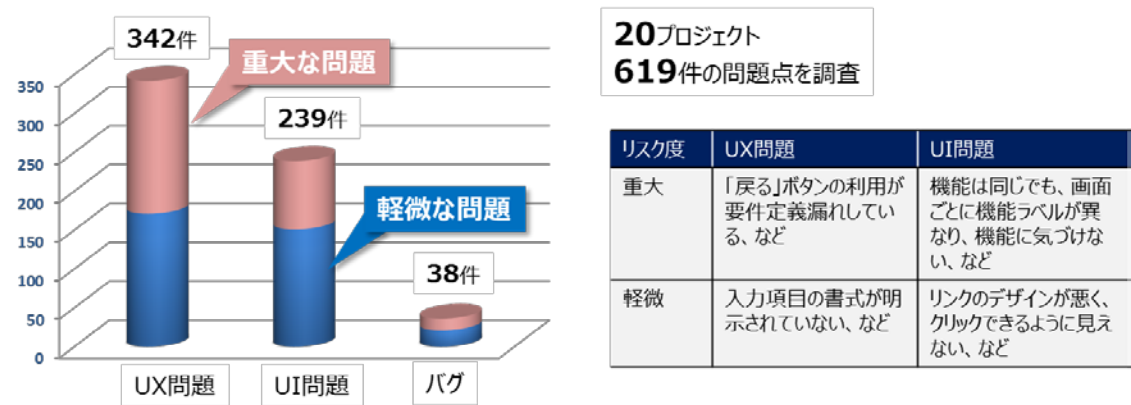


図 15-A-8-3 過去 20 プロジェクトからの問題抽出結果

さらに、重大な UX 問題を精査すると、大きな手戻りにつながる原因として、大きく 2 つに分類できることがわかった。それぞれ、「顧客ニーズの理解・抽出不足」と、「情報共有の不確かさ」である。

まず、「顧客ニーズの理解・抽出不足」とは、開発が機能目線になっており、実際にそのシステムを利用するユーザーのことが 1 つの枠組みとして考えられていないことを示す。その結果、顧客が本当に求めている潜在的なニーズが明確にならないまま開発を進めてしまい、後工程で手戻りが発生してしまう可能性が高くなる。例えば、「検索機能の追加」という要件定義があったとしても、「そもそもなぜ検索機能が必要なのか」という本質が明らかになっておらず、その後の工程で検索機能の必要性について問われ、トップページにショートカットリンクを備えるだけでいいという顧客から逆提案されることになったプロジェクトもあった。

次に、「情報共有の不確かさ」とは、プロジェクトメンバーによって意見や解釈がバラバラで、成果に統一感がないことを示す。その結果、リスクや開発の重要度などがうやむやになってしまい、後工程で手戻りが発生してしまう可能性が高くなる。例えば、「品目番号」「品目 ID」「製品コード」といった用語のブレがあるまま開発を進めてしまい、その後の工程でデータベースの整合が取れていないことが発覚し、予期せぬ調整が入ったプロジェクトもあった。

これらの問題は、特に大規模なプロジェクトで顕著に表れている。なお、ここで定義して

いる「大規模」とは、2年以上継続して実施され、開発者数が延べ100人を超えたプロジェクトを示す。

このような大規模プロジェクトでは、開発に関わるステークホルダーが増え、その分顧客の要求も多様化する。また、顧客側も一枚岩ではないため、役職や部署ごとに要求が異なり、それらを整理する必要性もこちらに求められる。さらに、開発部隊の人数も増加し、全体の管理が難しくなる。特に同社では、NECグループだけでなく様々な関係会社に関わることも多く、オフショアとして中国を利用することもあるため、より複雑化している。

2.4. 既存の UX 品質設計手法とその課題

UX 問題に対処するには、すでに価値が実証されている手法を用いるのが効果的である。既存の UX 向上手法としては、ヒアリングやフィールドワークなどの一般的な手法をはじめとし、ペルソナやストーリーボード、カスタマージャーニーマップなどの分析手法が知られている[3]。これらは、一般的に Web サイトや e-コマースなどのコンシューマ向け案件で利用されることが多い。

図 15-A-8-4 は、同社が実際に SI/ソフトウェア開発プロジェクトに適用し、効果のあった手法を挙げている。例えば、「ペルソナ」は、利用者の潜在的なニーズを洗い出し、1つのユーザーモデルとして具現化する手法である。実際のプロジェクトでは、ユーザーの置かれる立場や心情をステークホルダー間で共感し、方針や要求のズレを補正するために効果的だった。また、「ストーリーボード」は、ユーザーが満足する物語を絵コンテとして書き出し、本質的な価値を把握するための手法である。実際のプロジェクトでは、特にスクラッチ開発において、UX を向上させる新規アイデアを生み出すために効果的であった。

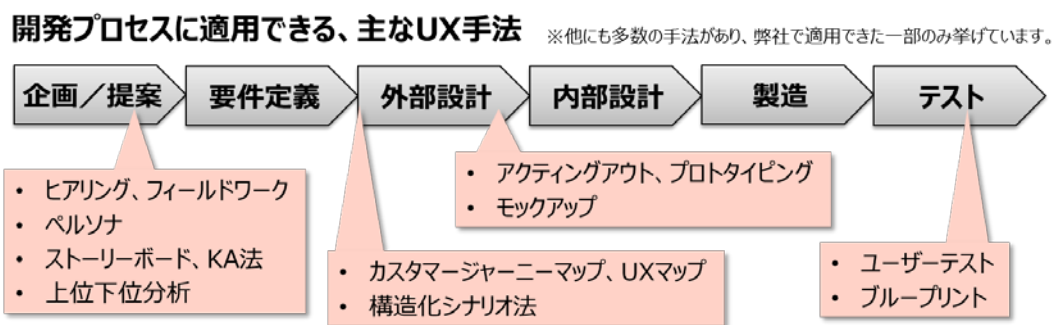


図 15-A-8-4 開発プロジェクトに効果的であった UX 品質設計手法の一例

しかし、図 15-A-8-4 のような手法をプロジェクトに適用するにあたり、大規模なプロジェクトになるほど、以下の3つの課題が顕著になった。

- ① 顧客側に、わざわざレビュー時間を割いてもらえない。
- ② 手法自体が顧客側や開発現場の経験になく、受け止められにくい。
- ③ 顧客側も開発現場も、工数や期間が限られている。

例えば、図 15-A-8-4 に挙げた構造化シナリオ法やカスタマージャーニーマップを適用すると、ユーザーの行動や心理が整理され、必要な画面や機能が明確になる（図 15-A-8-5）。しかし、そういったシナリオやマップは、作成に時間がかかるだけでなく専門的であるため、顧客側から理解を得るのが難しい。また、シナリオやマップが顧客現場に沿っているかどうかを、顧客側にレビューしてもらう必要があり、そういったレビュー時間を用意してもらうことも難しい。

また、ストーリーボードのような、絵コンテを用いた手法は、これまでの開発プロセスのドキュメントにないものであり、顧客側だけでなく、開発現場からも受け止められにくい。また、そもそもこういった活動を誰がやるのかという点で、工数や期間を設けにくいだけでなく、コストを切り詰めたプロジェクトでは活動費用そのものが得られないこともある。

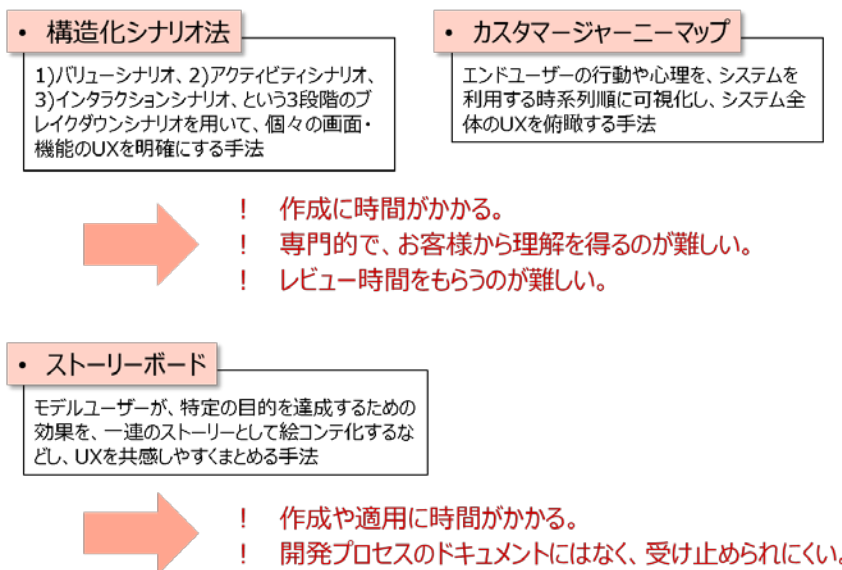


図 15-A-8-5 既存の UX 品質設計手法が現場に受け入れられない理由の例

特に一番の問題は、開発現場の機能目線という文化にある。例えば、エラーメッセージにおいて、「Error: 1011」のようにエラーコードを出力する状況がある。開発現場からは「エラーを出しており、エラーコードも載せているから問題ない」という言い分であるが、実際利用するユーザーからは「何が起きたか、そして何のエラーかわからない」というクレームが上がる。確かにシステムだけを見れば問題なく動作はしているが、ユーザー目線になっていないため、結局のところ使いにくくわかりにくいシステムになっている。例えばこの場合の正解は、「エラーが発生しました。品目番号の書式が間違っています。再度入力し直してください。」というように、ユーザーに次の行動を促すメッセージを設定すべきである。

このような機能目線をユーザー目線にするため、これまでも顧客要求やユーザーの立場などを開発現場に説明したり、UX やユーザビリティの重要性を教育したりしてきたが、根本

的な解決には結びつかなかった。その理由として、開発現場の作業者の入れ替わりが頻繁に発生し、教育しても去っていくことや、開発で精いっぱい時間が無いこと、そして作業者ごとに技術力が異なるため、全体に浸透しないということが、すでにわかっている。

3. 提案および実践

3.1. 大規模プロジェクトに順応する新たな2つのUX品質設計プロセス

前述したUX問題や、開発現場上の課題を解決し、大規模プロジェクトでUX品質設計プロセスを適用するために、2つのアプローチを提案し、実践した。1つ目のアプローチが、UXの基準となる代表画面・機能を予め策定し、顧客ニーズを確実に把握する「顧客体験型設計法」である。そしてもう1つのアプローチが、優先度や重要度などのUXを直感的に理解できるUI標準ドキュメントを策定し、開発現場での情報共有を確かなものとする「UX共有手法」である。

3.2. アプローチ1：顧客体験型設計法

顧客体験型設計法とは、画面や機能の要件を整理するに当たり、検索・一覧・詳細といった一般的な分類にまとめるのではなく、ユースケース・作業・行動といった、実際にシステム・サービスを利用するユーザーの体験価値で分類する手法である。特に業務向けシステムの場合は、コンシューマ向けに比べてユーザーの行動パターンが一定である点にも注目した。

図15-A-8-6は、顧客体験型設計法を用いて、生産管理システムの「製品一覧画面」を設計した例を示している。まず、業務内容や顧客要求をヒアリングすることで、製品一覧という1つの機能に、不良発見と進捗俯瞰という2つの体験が求められていることがわかった。次に、その2つの体験を活かせる機能を分析したところ、それぞれ数値一覧機能とグラフ表示機能という別々の機能が必要になることがわかった。なお、この機能分析では、顧客経験や開発能力に合っていない突拍子な画面・機能を選択しないようにするため、開発現場と打ち合わせをしている。また、体験は画一的ではないことにも注意したい。前述した製品情報一覧は、2つの体験で適合したが、いくら分析やヒアリングをしても、本当に顧客に適した体験分析は困難である。今回は体験分析をできるかぎり確実にするため、顧客先に常駐し、業務内容や課題を常に把握して対応していた。この工数をより効率化するのは今後の課題である。

負荷	手配方式	遅延	4/10		4/11		4/12	
			AM	PM	AM	PM	AM	PM
⊕ 工数		30	520	430	332	280	360	
⊕ 総値		200	250	400	420	300	360	
⊕ 手配数		1200	500	420	200	321	269	
⊖ 工数		5	12	6	22	27	18	
	標準品		5	2	6	8	5	
	数量品					10	5	

製品情報一覧画面において、工程上の不良製品を見つける作業では、個々の**数値**を重視する。

流通区分	手配方式	3/27		3/28		3/29		3/30	
		AM	PM	AM	PM	AM	PM	AM	PM
製造	製造								
製造	製造								
製造	製造								
輸入									
製造	製造								
製造	製造								
製造	製造								
製造	製造								

同じく製品情報一覧画面であっても、工程上の作業推移を俯瞰する作業では、数値ではなく**グラフ**を重視する。

図 15-A-8-6 顧客体験型設計法による体験分析例

一方、図 15-A-8-6 とは逆に、複数の機能が 1 つの体験に集約できる場合もある。図 15-A-8-7 は、生産管理システムにおいて、要件定義をベースに機能集約した結果をまとめたドキュメントの一部である。例えば、図 15-A-8-7 の No.4 は、資材手配業務の集約結果およびその理由を示している。この資材手配業務にはもともと、内示手配や確定手配、手動手配など、購買オーダーに基づいて調達する機能が複数あったが、これらは初めの検索から情報入力して登録するまで、一貫して同じユースケースであったため、1 つの発注情報管理画面として集約している。

No	LVI	集約	集約画面名	集約根拠 (ユースケース)	
1	10. モニタリング	調達	工程負荷	モニタリングは分析画面として独立で利用することが多いため、集約しない	担当が 画と集
2	10. モニタリング	現場	達成率	モニタリングは分析画面として独立で利用することが多いため、集約しない	パッケージ 画面と
3	3. 所要量計算	計画	生産計画(製品番号別)	特定の担当による独立な機能であるため、集約しない	担当は いる
4	7. 資材手配	調達	発注情報管理	購買オーダーを基本として同じ表示形態を踏むため、集約する	担当が 上記と
5	7. 資材手配	調達	輸入品確認	利用タイミングが異なるため上記画面と集約しない	担当が 集約し

図 15-A-8-7 機能集約を示すドキュメントの一部

このように、体験価値に重点を置く枠組みを用いて設計することで、持ち帰り検討を減らし、より少ないレビュー時間で顧客と意識合わせができるだけでなく、後工程での手戻りも抑えられた。実際、過去のプロジェクトでは、製品情報を一覧する画面として、1画面しか定義されておらず、顧客から「思っていた機能と違う」というクレームが上がり、後工程で別の画面が追加され、その分の手戻りが発生したこともあったが、このような手戻りも今回の手法で削減できることが確認できた。

なお、実際のプロジェクトでは、上記手法だけでなく、画面設計自体にもルールを設け、ユーザーの IT リテラシーに沿った使い勝手を設計し、画面を利用するユーザーの部署や役職に合わせて機能をグルーピングできるよう、ユーザビリティ要件を開発現場の UI 標準ドキュメントに記載し、開発現場の全フェーズで共有する取り組みも実施している。

3.3. アプローチ 2 : UX 共有手法

UX 共有手法では、顧客要求の優先度や重要度を一目でわかるようにするため、すでに開発プロセスで利用されている UI 標準ドキュメントのレイアウトを変更し、色やサイズで直感的に認識できるようにした。

図 15-A-8-8 は、UX 共有手法を適用した UI 標準ドキュメントに記載されているオートコンプリート機能の例である。なお今回、UI 標準に関わるステークホルダは、VALWAY テクノロジーセンター（同社の SI/ソフトウェア開発を UX 観点で支援している組織。以下、VTC と示す）⁶ と開発部隊、そして顧客の 3 つであったため、3 色の文字を利用して記述し

⁶ 2015 年 4 月、組織統合により「イノベーション戦略本部」に部署名変更

ている。図 15-A-8-8 では、青文字が VTC、黒文字が開発部隊、そして赤文字が顧客からの要求を示す。開発部隊からはオートコンプリートの一般的な機能説明が記述されており、それは標準サイズの文字となっている。一方、VTC や顧客からの要求は、UX やユーザビリティに関わる重要な要件であったため、その重要性に合わせて文字サイズを大きくし、さらに重要な要求については太文字にしている。図 15-A-8-8 では、手戻りを引き起こしそうなリスクの高い要求は「パフォーマンス」であったため、太文字で明示している。

このように、ドキュメントを俯瞰しただけであっても、開発に向けた重要なポイントを認識できるようにすることで、開発で精一杯で時間のない開発部隊のメンバーにも、顧客の想いが共有できるようになり、後工程での手戻りを抑えられることが確認できた。

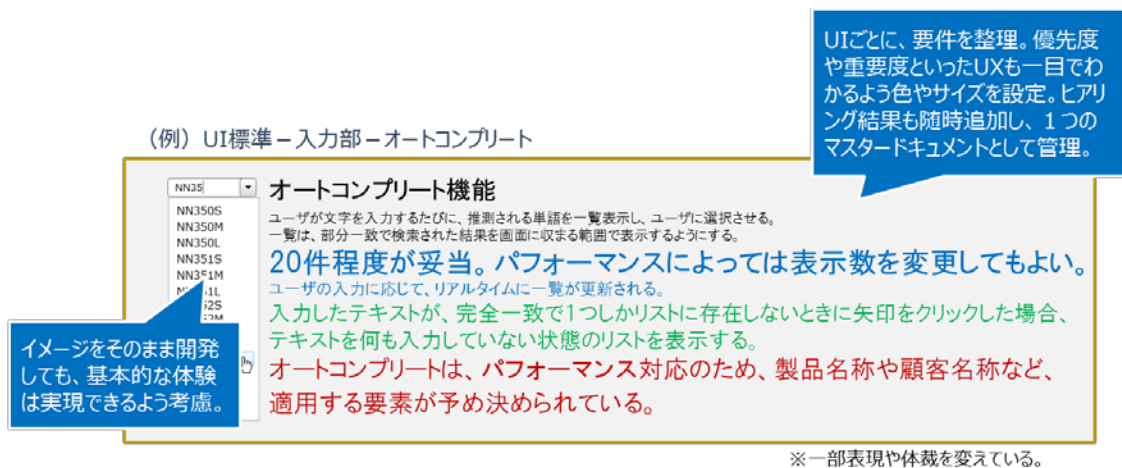


図 15-A-8-8 直感理解型 UI 標準ドキュメントの記載例

4. 結果

4.1. 大規模プロジェクトへの適用

今回、製造業務系の生産管理システムと、基幹業務系の営業管理システムの、2つの大規模プロジェクトに対し、前述した手法を適用した。

図 15-A-8-9 は、生産管理システムの開発スケジュールを示している。まず、VTC は画面 UI 設計フェーズからの参加であったため、そこで顧客体験型設計法を適用した。前述したように、画面や機能をユースケースや作業者などの行動別に集約することで、作業工数やレビュー時間を抑えられただけでなく、要求漏れも事前に発見できたため、画面 UI 設計フェーズでの手戻りも抑えられた。さらに、業務機能設計やサービス設計といった他のフェーズにおける検討漏れの早期発見にも貢献できた。また、設計標準策定フェーズにおいては、直感理解型 UI 標準ドキュメントを策定し、フェーズの推移によって作業者が入れ替わる場合やオフショアへの開発指示でも、顧客要件を維持するのに効果を発揮できた。

15-A-8 大規模システム開発プロジェクトにおける
ユーザーエクスペリエンス品質設計プロセスの適用とその効果

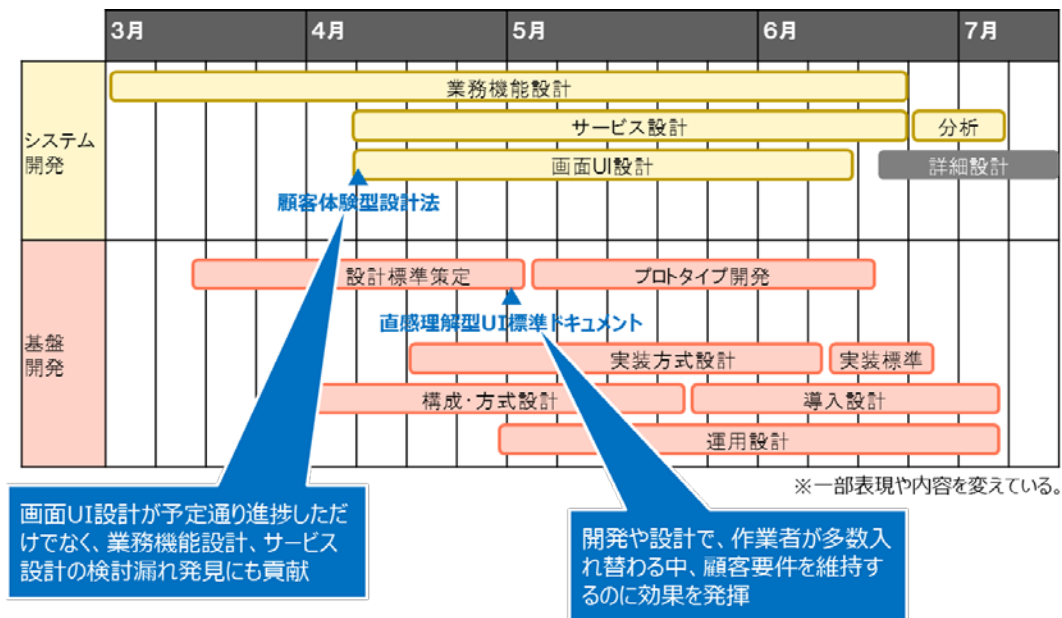


図 15-A-8-9 開発スケジュールから見る提案手法の効果

一方、営業管理システムでは、確認された 15 件の UX 問題のうち、11 件を解決または削減できることが確認できた（表 15-A-8-2）。なお、図中の『削減』とは、完全に解決することはできなかったが、進捗に影響が出るほどの手戻りは確認されなかったことを示す。例えば、『1. 個人の問題』や『2. チームの問題』による成果物の『バラつき』は、直感理解型 UI 標準ドキュメントにより、大きく削減された。また、『業務内容の把握が難しい』問題や『問い合わせが多発する』問題も、顧客体験型設計法により削減された。なお、こういった新しい手法を導入することで、現場の雰囲気が変わったことも、削減に影響したと思われる。

一方で、UI 標準ドキュメントに修正箇所が出ても、その修正がされないことがあった。これは、開発現場のメンバーには『修正の仕方がわからないこと』が主な原因であった。また、『ヒアリングが十分できない』問題なども、今回の手法では解決できないため、今後の課題となっている。

一部の問題については、プロジェクトとしてその問題を認識し、今回の手法を組織立って適用すると、より効果が增加することが確認された。例えば、原始的な方法ではあるが、理解不足や不整合が発生しないように、各フェーズのマネージャがメンバーに声掛け確認を定期的に行うようにしたことが比較的效果があった。大規模なプロジェクトになるほど、専門家やドキュメントだけでは対応できない要素も増加するため、メンバーに顧客体験を定期的に意識させる方法が効果的だったと考えられる。

表 15-A-8-2 営業管理システムにおける提案手法の効果

1. 個人の問題		適用結果	
(1)	画面デザインについてさまざまな意見が出て、なかなか決まらない。	△ 削減	※
(2)	ドキュメントのボリュームが大きく、設計担当者が一部しか読まない。	△ 削減	
(3)	ドキュメントの解釈の違いにより、設計者で画面にバラつきが出る。	△ 削減	
2. チームの問題		運用結果	
(4)	ドキュメントのボリュームが大きく、設計、標準化チームのレビュー時間が不足する。	△ 削減	※
(5)	画面活用方針が不明確のまま進み、画面標準の検討が遅れる。	○ 解決	
(6)	画面デザイン標準を適用したが、標準どおりにできていない。	○ 解決	※
(7)	チームごとに画面のバラつきがある。	○ 解決	
(8)	ドキュメントの修正が、徹底されない。(修正の仕方がわからない)	× 未解決	※
3. 全体の問題		適用結果	
(9)	組織的な制約などにより、既存システムの評価やヒアリングが十分できない。	× 未解決	
(10)	限られた時間で、複雑・広範囲な業務内容の把握が難しい。	△ 削減	
(11)	ドキュメントの説明不足により、設計部門から問い合わせが多発する。	△ 削減	
(12)	設計が進まないと決められず、仮決めたドキュメントで、のちに不整合が発生する。	△ 削減	※
(13)	業務背景や根拠の理解不足で、お客様の希望実現の可否が判断できない。	△ 削減	※
4. 制約による影響		運用結果	
(14)	設計が進んでいて、ドキュメントによる修正の影響範囲が大きくなる。	× 未解決	
(15)	全体方針に従うことで、効率の低下や遂行に支障をきたす業務がある。	× 未解決	

※プロジェクトが組織だて適用すると、より効果が増加。

また、前述した過去 20 プロジェクトから抽出された 342 件の UX 問題に対し、今回の提案で解決できた要件をピックアップした結果、148 件あることがわかり、プロジェクトの画面・機能上の UX 問題を、およそ 40%抑えられる見込みが算出できた（図 15-A-8-10）。

なお、この解決可能な 148 件の内訳は、例えば、「弱視者用の読み上げソフトで対応できていない。」「メイン作業である承認がどこまで完了したかわかりにくい。」といった、メイン作業および全体に関するコアな問題になっている。

一方、その他の 194 件については、「コード入力後に、登録キーを押す手順がわからない。」など、ニッチな作業および個別画面に関する問題となっている。これらについては、今回の手法だけでは解決できないため、別の手法を検討する必要がある。

画面・機能上の問題を、約 **40%** 抑えられる見込み

※ 前述した既存の20プロジェクトの342件のUX問題に対し、今回の提案で解決できた要件をピックアップした結果

解決可能	個別解決が必要	UX問題全体
148 件 (43%)	194 件 (57%)	合計 342 件

解決可能な問題の一例

- 「弱視者の読み上げソフトで対応できていない。」
- 「メイン作業である承認がどこまで完了したかわかりにくい。」
- 「戻るボタンについての規定がない。」
- 「一度メニューに戻るのは、作業上効率が悪い。」
- 「エラーメッセージのタイミングによって、誤読してしまう可能性がある。」

メイン作業、および全体に関するコアな問題については、削減および解決可能。

例えば、弱視者などのアクセシビリティは直感理解型UI標準で定義できる。また、承認作業といった特定の体験は顧客体験型設計法で分類できる。

個別解決が必要な問題の一例

- ！ 「コード入力後に、登録キーを押す手順がわからない。」
- ！ 「編集画面において、操作手順がわからない。」
- ！ 「一部の画面において、用語の意味がわからない。」
- ！ 「製造オーダと工程の入力値が連動するため、片方だけ修正する際に効率が悪い。」

ニッチな機能、個別画面については、これまでどおり個別に対応する必要がある。

図 15-A-8-10 提案手法による UX 問題の抑制率

4.2. 定常的な運用のためのフレームワーク化

同社は今回の結果を受け、UX 設計プロセスの大規模システム開発プロジェクトへの適用について、タスクフロー図にまとめ、個々の作業を定義し、フレームワーク化した。また、社内でも共有し、専門家でない一般 SE でも UX 設計プロセスが適用できる基盤を構築している。ただ、これまでの開発プロセスに導入するには、まだ抵抗のある現場も多いため、より浸透する仕組みへと改良していく必要がある。例えば今は、フレームワークを適用するに当たり、よく起こりうる問題点を留意事項として整理するなど、活用を促進する工夫に取り組んでいる[4]。

一方、UX 設計プロセスを適用すべきプロジェクトの重要度やリスクを判断するため、遵守状況も数値化して共有している。例えば、表 15-A-8-2 の『画面デザイン標準を適用したが、標準どおりにできていない。』という問題については、「91%もの画面で、標準に従っていない項目があり、画面単位では、4.03 個の機能が標準に従っていなかった」という実績値を得ている。これが特定の閾値を超えるようなリスクの高いプロジェクトの場合は、チームごとに UX 専門家をアサインしてすべての画面の遵守状況をチェックしたり、逆に値が低いプロジェクトの場合は、サンプリングチェックで対応したりするなど、UX 設計プロセスの適用基準として利用することを見込んでいる。今後は、こういった UX 設計プロセスの適用がより単純化できるよう、改良に取り組む予定である。

5. まとめ

本編では、大規模な SI／ソフトウェア開発プロジェクトにも対応できる新たな UX 設計プロセスとして、顧客体験型設計法と UX 共有手法の 2 つを紹介した。実際の大規模プロジェクトに適用したところ、画面 UI 設計フェーズだけでなく、業務機能設計やサービス設計などの他のフェーズの検討漏れ発見にも貢献し、予測されるリスクもおおよそ 40% 抑えられることが検証できた。今後は、フレームワーク化を通じて得られた知見を汎用化し、一般 SE でも利用できる手法として展開していきたい。

参考文献

- [1] 青木博之、三浦一成、月田逸郎、野田尚志、中井治男：SI/ソフトウェア開発におけるユーザー中心設計、NEC 技報 Vol.64 No.2、2011
- [2] 森口昌和：スマートデバイスの UX デザイン～事例から学ぶ失敗しないアプリ設計術～、日経 BP 社、2013
- [3] 川西裕幸、栗山進、潮田浩：UX デザイン入門、日経 BP 社、2012
- [4] 木下友見、大川裕行：大規模システム開発におけるユーザビリティ向上技術適用上の留意点、HCD 研究発表会、2013

掲載されている会社名・製品名などは、各社の登録商標または商標です。

独立行政法人情報処理推進機構 技術本部 ソフトウェア高信頼化センター（IPA/SEC）