

専門的な装置を制御するソフトウェア向けのシンプルな UX 手法 の提案

Proposing simple UX methods for developing specialized machine control software

主査 : 金山 豊浩 (株式会社ミツエーリンクス)
副主査 : 三井 英樹 (Weblysts.com)
村上 和治 (東京海上日動システムズ株式会社)
リーダー : 清水 有子 (日本電子株式会社)
研究員 : 志賀 愛弓 (TIS 株式会社)
水野 智仁 (株式会社ヴィッツ)

研究概要

近年、UX 手法の重要性は徐々に認識されるようになってきた。しかし UX 手法を実施するには専門家の助けや多くの開発工数が必要となる場合があり、導入に二の足を踏む現場もある。また、専門的な装置を制御するソフトウェアでは使いやすさや操作性などよりも、機能面が重視される傾向にあるため、UX 手法の重要性が理解されにくい。そこで、本論文ではそのような分野でも UX 手法が適用されやすくなることを目指し、装置制御を行うソフトウェアを対象に、機能面を重視したまま、コストや時間をかけずに実施できる UX 手法を提案する。本提案手法は UX の知識があまり無い開発者でも導入できるようになっている。また、本提案手法を電子顕微鏡の制御ソフトウェア開発に適用して効果を確認したので、合わせて報告する。

Abstract

In recent years, the importance of UX techniques has become gradually recognized. But in many cases to implement the UX techniques, support from expert and a lot of development time are required. There is also hesitation for implementing the UX techniques at the software development site. Furthermore, in the fields of controlling the technical system since it tends to the functional aspect is more important than usability or operability of the software, the importance of UX approach is not well understood. Therefore, in this paper, we suggest the UX methods which can be carried out without the cost and time in such field while an emphasis on functional aspect. These methods can be introduced by the developers who do not have a lot of knowledge about UX. Also we confirmed the effect by applying to the control software development of the electron microscope.

1. はじめに

UX やユーザー中心設計の概念は以前から存在する。さらに近年では、スマートフォンをはじめとするモバイルツールが普及することに合わせて、UX 手法を活用するソフトウェア開発現場が増えてきている。そのため、特に Web 開発を対象にした UX 手法が数多く提案されている [1]。一方、専門性の高い装置制御ソフトウェアを見ると、ユーザーに装置の知識が必要であったり、ソフトウェアの動作に周囲のハードウェアや環境との連携が必要であったりするなどの事情から、Web 開発向けに提案された UX 手法をそのまま適用しにくい場

第4分科会 (H20 チーム)

合がある。また、専門性の高いソフトウェアの場合は、市場が機能を重要視することが多いため、ユーザーの使いやすさよりも機能面に主眼を置いて開発する傾向にある。そのため、使いやすさやユーザビリティがおざなりになりやすい。しかし、専門性の高いソフトウェアであっても、特に新しいユーザー層を開拓する場合や、ソフトウェアの作業効率を上げるための改良を行う場合などでは、UX手法が非常に有効であると考えられる。そこで、私たちは、装置制御の既存ソフトウェアの改良において、機能面に主眼を置いたまま、シンプルに実施できるUX手法を提案する。提案するのは、開発の企画段階においてユーザー調査を行う手法と、それにより作成した企画を評価するためのプロトタイプング手法である。以下にこれら二つの手法について詳細に説明する。

2. 提案手法

2.1 企画段階におけるユーザー調査

2.1.1 一般的なUXのユーザー調査手法

企画段階にて行うUXによるユーザー調査の手法はいくつかあるが、ここでは一般的な手法として、コンテクスチュアル・インクワイアリー[2]とヒューリスティック評価について簡単に紹介する。

コンテクスチュアル・インクワイアリーは、ユーザーとUX専門家が擬似的に師匠と弟子の関係を作ってユーザーの行動を観察する手法である。被験者であるユーザーに師匠となってもらい、UX専門家は弟子としてユーザーの作業方法を教えてもらう。UX専門家は表面上弟子であるが、実際は対象のソフトウェアを熟知しており、ユーザーがどのように使うかを事前に想定している。そして、想定していた使い方とユーザーの使い方が異なる場合には、なぜそうするのかを師匠であるユーザーに聞き、潜在的な要求や真の要望をあぶりだしていく。UX専門家は、ミスリードせず本来のユーザーの利用状況を把握できることが重要である。そのためにはユーザー調査の経験や知識が必要である。一般的に、対象のユーザーが他のユーザーの言動に誘導されないよう、個々のユーザーごとにヒアリングを行うのが良いとされる。

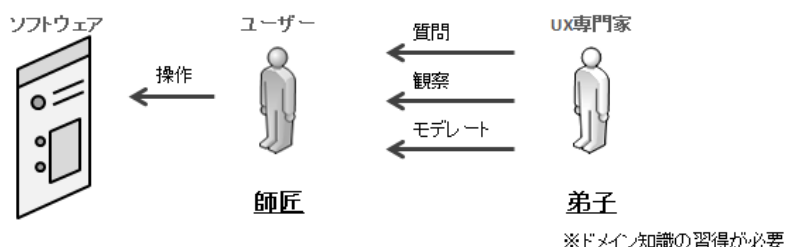


図1 コンテクスチュアル・インクワイアリー

また、コンテクスチュアル・インクワイアリーに限らず、実際のユーザーを被験者とする手法の場合、被験者を集めることにも時間とコストがかかる。テストに対して理解を示し、モチベーションを高く保って協力できる被験者を見つけることは容易ではない。そして、被験者と信頼関係を築いて本音を話してもらうためのテクニックも必要である。

一方、ヒューリスティック評価では開発者が評価を行う。ユーザビリティの専門的な視点でソフトウェアを操作し、ニールセンの「ユーザビリティに関する10のヒューリスティクス(問題解決に役立つ知見)」[3]などの指標に基づいて評価する手法である。そのためユーザビリティを適切に評価できるUXの専門家が必要である。

これらの手法では、一般的にUXの専門家が開発に加わるのが前提となっている。対象のソフトウェアのドメインが専門的である場合には、UXの専門家はドメインについての知識も必要となり、さらに時間やコストがかかる。

2.1.2 提案手法『お手軽ユーザー調査』

提案手法では、従来の UX 手法の形式にこだわらず、UX の知識があまりない開発者のみで実施できる。開発メンバーが UX について深く学習したり、開発メンバー以外の UX 専門家がドメインについて学習したりするコストは不要である。

ところで、本手法がターゲットとしている専門的なソフトウェアの場合、ユーザーは使い始めに利用方法を学習するのが一般的であるが、それでも熟練者と初心者には知識量の差があり、操作方法や要求が異なってくる。熟練者はカスタマイズできることを求め、初心者は自由度が無くても分かりやすいことを求めている。そのため、ユーザー調査は初心者ユーザーと熟練者ユーザーの両方に対して実施するべきである。

前述したとおり、ユーザー調査の被験者を集めるのは難しい。特に専門的な装置を制御するソフトウェアの初心者ユーザーについては、ドメイン知識はあるが、操作の経験が無い人でなくてはならない。そこで、本手法では、初心者ユーザーとして、対象のソフトウェア開発を担当していない別のソフトウェア開発者を利用する。初心者でありながら、モチベーションを維持してテストを行うことができ、ソフトウェアの専門家としての視点で評価できるメリットもある。社内から選べば、謝礼を用意したり秘密情報について考慮したりする必要がない。また本音を聞き出せるように信頼関係を構築する時間も不要になる。

本提案手法はコンテクスチュアル・インクワイアリーと同じ思想で行う。前述したように、コンテクスチュアル・インクワイアリーは、ユーザーを師匠とみ立て、開発者が弟子のようにユーザーのタスク処理方法をヒアリングする手法である。(図1参照)これをUXの専門家を加えずに実施するには、むしろ本当の初心者ユーザーが熟練者ユーザーに質問をする方が良い観察結果が得られると考えた。コンテクスチュアル・インクワイアリーの初心者が弟子の役を行うと、ユーザーに使い方を説明してミスリードしてしまったり本来のユーザーの使い方を観察できなくなったり、掘り下げるべきところで必要な質問ができなかったりする可能性があるためだ。本当の初心者ユーザーが師匠から使い方を教わる場合は、事前にソフトウェアの知識が無いために、そのような問題が起こりにくいと考えた。そこで、本手法のユーザー調査では、熟練者ユーザーと初心者ユーザーのヒアリングを同時に行い、弟子の役は初心者ユーザーに担当してもらう。これはヒアリングにかかる時間が短縮できるメリットもある。ソフトウェア開発者は書記として、熟練者ユーザーと初心者ユーザーのやりとりを観察して記録する。

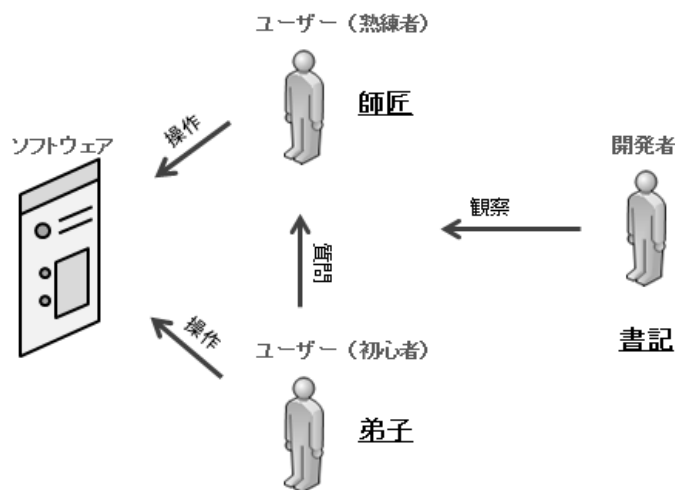


図 2 提案手法「お手軽ユーザー調査」

以下に提案手法の詳細な手順を述べる。各手順で参加者が注意すべき点をまとめて[付録A]とした。これをカードにして渡す仕掛けにすれば本手法を実施しやすい。

第4分科会 (H20 チーム)

(1) 熟練者ユーザーから初心者ユーザーへ説明

最初に、熟練者ユーザーが初心者ユーザーにソフトウェアの使い方を説明する。書記は熟練者ユーザーが説明したやり方や初心者ユーザーから出た質問を記録する。書記は質問などを行わずに観察するのみであるが、想定していた使い方と熟練者の使い方の違う部分や、初心者にとって分かりにくい部分が見えてくる。このステップでは、初心者ユーザーと熟練者ユーザーに対して、それぞれ以下の点に留意させる。初心者ユーザーには、説明を受けた後に一人で操作してもらうことを伝えておき、一人でも操作できるようになるつもりで説明を聞いてもらう。熟練者ユーザーには、操作中に何かしらの情報を確認する場合には、それを細かく説明するように念を押しておく。熟練者は操作方法については漏れなく説明するが多いが、現在の装置の状態を画面で見確認するような場合は一瞬の作業となるために説明から漏れがちであるためだ。

(2) 初心者ユーザーが使用

説明の後、初心者ユーザーに使ってもらう。書記は初心者ユーザーがどのように使うか、どこで戸惑うか、何が分からないかを記録する。大抵の場合、事前に説明を聞いていても初心者ユーザーは使い方が分からなくなるので、質問も自由に行ってもらい、書記はそれも記録する。ここでも書記は開発者側の視点で使い方を説明しないよう、質問に答えるのは熟練者ユーザーのみである。また、初心者ユーザーの考え方が分かるように、思っていることを口に出しながら操作してもらう。これは一般的に思考発話法と呼ばれており、人によって得意不得意があるが、被験者がソフトウェア全般に詳しい場合は、趣旨に沿って比較的上手に思考発話法を実施できることが期待される。また、もし書記に余裕があれば、初心者ユーザーが開発者の想定していなかった使い方や熟練者ユーザーの説明と異なる使い方をした場合には、なぜそのような使い方をしたのか、質問すると良い。ただし、ユーザーの操作を想定しているものに誘導しないよう、質問だけを行うことが重要である。

(3) 熟練者ユーザーが使用

その後、熟練者ユーザーに普段通りに装置を制御してもらい、書記はその手順を記録する。やはりここでも書記の方から質問は行わない。熟練者ユーザーは最初に初心者ユーザーに使い方を説明しているが、説明したやり方と実際のやり方が異なる場合がある。正しい使い方として理解している方法と、実施しやすい方法が異なることがあるためだ。そのため、熟練者ユーザーにも説明なしで普段通りの操作を実施してもらう。

(4) 要望や質問など

最後に、熟練者ユーザーと初心者ユーザーの双方から、感じたことや要望などを何でもざっくばらんに話してもらう。ここでは、書記も、ユーザーに対して疑問に感じたことなどを質問する。

2.1.3 『お手軽ユーザー調査』実施後の分析

ソフトウェア開発者は、ユーザー調査での議事録から現状のソフトウェアの問題点や要望などを洗い出す。まず、ユーザー調査を実施する前に、開発者が想定しているユーザーの使い方を元にカスタマージャーニーマップ[4]を作成しておく。カスタマージャーニーマップは、ユーザーの使い方をいくつかの目的ごとに分割し（これをタスクと呼ぶ）、タスクごとのユーザーの行動や使い方、感情をまとめ、その流れを図示する。その後でユーザー調査を実施し、ユーザー調査を実施した後に、実際のユーザーの使い方から再度カスタマージャーニーマップを作成する。いずれも初心者ユーザーと熟練者ユーザーのそれぞれにおいて作成する。これらを比較することで、開発時に想定していたタスクが全て必要であったか、想定のないタスクがなかったか、また、タスク実行時の感情の想定が間違っていなかったかを調べる。そして、ユーザー調査前に見落とししていた部分に着目し、新しいソフトウェアを設計する。

2.1.4 他手法との比較

師匠と弟子となった被験者を開発者が観察する手法は、Svetlantaらが2012年に子供向

第4分科会 (H20 チーム)

けソフトウェアの開発手法としても提案している[5]. Svetlanta らの方法では, 一人目の子供に対象のソフトウェアの使い方を説明した後に, 二人目の子供に対して一人目の子供から使い方を説明させる. モデレーターはそれを観察し, 一人目の子供が使えなかったり説明できなかったりしたときには誘導する手法である. ソフトウェアの使い方を子供がどう理解するかを観察することができて有用であることが示されている.

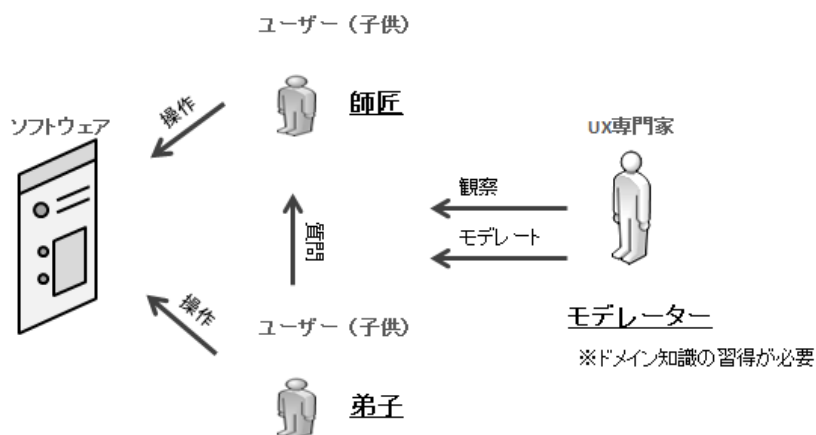


図3 Svetlanta らの手法

本提案手法とは, 師匠となる被験者がソフトウェアに慣れていないことや, モデレーターが被験者の操作方法を誘導すること, 被験者が子供のため思考発話ができないことなどが異なるが, 被験者に師匠と弟子を演じさせてユーザビリティを確認する部分は似ており, 提案手法の成果も期待できる.

これまでに上げた師匠と弟子の関係性を模してユーザー調査を実施する手法の比較を [付録 B] で一覧にまとめておく.

2.2 企画テスト段階におけるプロトタイプ手法

次に, 企画内容を評価するための提案手法を説明する.

2.2.1 一般的な企画テスト手法

UX 手法で大切なことは, 常にユーザーの体験を向上させるように設計することである. そのため, ユーザーが真に求めるものに近づいているかを評価しながら開発を行う. その方法として, プロトタイプやストーリーボーディングなどの手法が提案されている. プロトタイプでは, 提案するソフトウェアの UI だけを作成して評価を行うものが多い. ユーザーがプロトタイプを操作した際に, 開発者が影でコンピューターが行う作業を実施するオズの魔法使いという手法が提案されている[6]. この手法を使えば, プロトタイプが紙でできていても, ユーザーはソフトウェアの使用感を得ることができる. 本提案手法ではこの手法を応用する.

2.2.2 提案手法『二人羽織式オズの魔法使い』

ハードウェアを制御するソフトウェアの場合, 実際のハードウェアの動きが分からなくては評価を行っていくので, 絵コンテのように体験を表現するストーリーボーディングではイメージをつかみ難い. 一方で, 評価のために多くの時間やコストをつぎこむと, 評価回数を減らすことにつながり, 良い効果を得られない. そのため, 本提案手法ではプロトタイプを紙で作るペーパープロトタイプを実践する. ただし, プロトタイプの操作時に UI を切り替えることに加え, 装置の制御も行い, 実際にどう動くかをユーザーに理解させる必要がある. プロトタイプではハードウェアは既存ソフトウェアや手動で制御を行うしかないので, 私たちは以下の手法でプロトタイプを実践する手法を提案する. プロ

第4分科会 (H20 チーム)

トタイピングは、プロトタイプを熟知した開発者二人で実施する。一人目はソフトウェアのプロトタイプの操作を行い、二人目はその操作で行われる装置制御を手動か、あるいは既存ソフトウェアによって実行する。装置やソフトウェアの配置を工夫して、二人目の開発者が実施している内容はユーザーに伝わらず結果だけが目に触れるようにする。ユーザーがソフトウェア内部での動作を知ってしまうと、ブラックボックステストにならないためである。これにより、ハードウェアを制御するソフトウェアでもプロトタイプによる実地でのテストが可能となる。

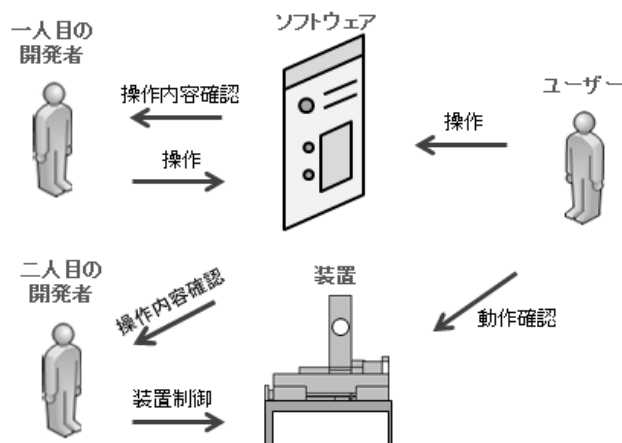


図4 提案手法「二人羽織式オズの魔法使い」

3. 適用結果

3.1 電子顕微鏡制御ソフトウェア

提案手法を実際プロジェクトに適用し、プロトタイプの作成を行って、ユーザー評価を実施した結果を示す。適用したプロジェクトは、電子顕微鏡を制御するソフトウェアの一機能である。この機能は、これまで専門的なユーザーを対象に開発されてきたが、これからは新たにこの分野に参入するユーザーも利用できるものとしたい。さらに、これまでのユーザーにも満足してもらえる機能を追加することを目的とする。ただし、電子顕微鏡を制御するソフトウェアのうちの一部の小さな機能であるため、大規模な時間やコストをかけることができない。そのため、本提案手法が適していると考えた。また、本提案手法を適用する前に、既に会議によって追加機能や改善UIの仕様を決め、プロトタイプを作成していたので、本手法を適用して作成したプロトタイプと比較を行うこととした。以後、分かりやすさのために会議によって仕様を決めたプロトタイプを「既存プロトタイプ」と呼び、本手法を適用して作成したプロトタイプを「提案プロトタイプ」と呼ぶ。

3.2 『お手軽ユーザー調査』の適用

一つ目の提案手法「お手軽ユーザー調査」は、熟練者ユーザー二人と初心者ユーザー三人に対して実施した。すると、開発者側が想定した使い方と熟練者ユーザーの使い方に違いがあることが分かった。熟練者ユーザーは一部の操作をソフトウェアからではなくハードウェアのオペレーションパネルから実施しており、その場合にはソフトウェアの画面を起動していなかった。そのため、現在の状況を確認しにくくなるという問題が生じていた。また、ソフトウェア開発者の方では便利になると考えて実施していた、別のハードウェアとの連動機能が熟練者ユーザーにとっては不要に感じる場合があり、その場合にはわざわざ連動を解除して独自に設定して使っている実態も分かった。初心者ユーザーが使った場合は、画面を表示してもどの画面を見て良いか分からない、と感じることも分かった。また、初心者ユーザーの場合は、オペレーションパネルよりもソフトウェアの画面の方が使いやすいことも分かった。その他にも、熟練者ユーザーは慣れてしまっているが、初心者

第4分科会 (H20 チーム)

ユーザーからすると分かりにくい表示や、初心者ユーザーなら疑問を持たずに使う機能でも、熟練者ユーザーからは内部の状態を知りたいという要求が出てくる操作があることも分かった。これらのヒアリング結果を元に、分析を行って既存機能の改良と新機能の追加を行い、提案プロトタイプを作成した。

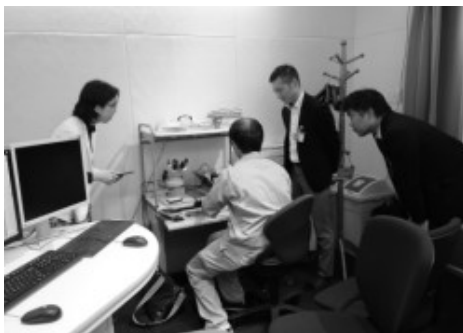


図5 ユーザー調査の風景

3.3 既存手法との開発工数の比較

既存プロトタイプの開発にはおよそ8人日かかり、提案プロトタイプの開発にはおよそ11人日かかった。これはヒアリング対象のユーザーや会議の参加者などの時間も全て含むものである。これより、本手法を採用しても開発工数はあまり増えないことが分かった。慣れればより短時間で実施できる可能性がある。

3.4 『二人羽織式オズの魔法使い』による評価結果

評価は熟練者ユーザーと初心者ユーザーに対して二つ目の提案手法「二人羽織式オズの魔法使い手法」を実施し、後述する項目を評価させた。初心者ユーザーは、対象ソフトウェアの分野に詳しくないソフトウェア開発者である。全くの初心者とするため、「お手軽ユーザー調査」を実施した際の初心者ユーザーとは別人を選んだ。

最初に、ソフトウェアの画面を見ただけで操作方法がイメージできるかどうかを確認するために、二人の熟練者ユーザーには説明無しで二つのプロトタイプのメイン画面をそれぞれ5秒間見てもらい、以下の質問を行った。

「操作方法がイメージできましたか？」

その結果、以下のようになった。

表1 5秒間だけ見た後の感想

	熟練者①	熟練者②
既存プロトタイプ	分かる	分かる
提案プロトタイプ	分かりそう	まあまあ分かる

次に、やはり説明のないまま、二人の熟練者ユーザーに二つのプロトタイプを使ってもらった。使った順番で結果が左右されないよう、一人には既存プロトタイプを先に評価してもらい、もう一人には提案プロトタイプを先に評価してもらった。

また、使い方が分からない場合は質問をしてもらい、その回数をカウントした。その結果、以下のようになった。

表2 説明なしで使った場合の質問回数

	熟練者①	熟練者②
既存プロトタイプ	1回	0回
提案プロトタイプ	2回	2回

さらに、全てのタスクを実施してもらった後に、以下の質問を行った。

①使いやすさ「どちらが使いやすかったですか？」

②操作性「どちらの操作に満足しましたか？」

③市場価値「どちらが欲しくなりましたか？」

この結果については表3に示す。

第4分科会 (H20 チーム)

初心者ユーザーの場合は、装置の使い方から説明が必要であるため、それらの説明と二つのプロトタイプの説明を最初に実施した。その後で、二つのプロトタイプを使ってもらい、熟練者ユーザーと同様に質問を行った。その結果は以下となった。

表 3 全タスク実施後の質問の回答

質問	熟練者①	熟練者②	初心者
① 使いやすさ	提案プロトタイプ	どちらも同じ	提案プロトタイプ
② 操作性	どちらも同じ	提案プロトタイプ	提案プロトタイプ
③ 市場価値	提案プロトタイプ	提案プロトタイプ	提案プロトタイプ

被験者となった熟練者ユーザーが既存プロトタイプ作成時の会議のメンバーであったため、利用時の質問回数などは提案プロトタイプの方が多結果となった。それでも提案プロトタイプのほとんどの操作は質問なしで実施でき、提案プロトタイプの方が欲しくなるとの回答を得られた。自分たちで仕様を決めたソフトウェアよりも本提案手法を適用したソフトウェアの方が欲しくなったというのは当初想定していた以上の成果であった。さらに、初心者ユーザーにおいては、どの質問においても提案プロトタイプの方が高い評価を得られた。これらのことから、提案手法の有用性を確認できた。

4. まとめ

本研究では、UXの専門知識のない開発者が、装置制御ソフトウェアに対してあまり工数をかけることなくUX手法を適用できることが示せた。また、提案手法を使えば熟練者と初心者の双方に有効な改良が行えることも分かった。本手法の適用で開発したソフトウェアの成功が、より本格的なUX手法を適用する動機付けになることが期待できる。今後、他プロジェクトへの適用によってコンテクスチュアル・インクワイアリーとの有用性を比較したいと考えている。さらに、より詳細な手順書の作成を検討している。

参考文献：

- [1] 時田 郷、吉武 良治、国内におけるUXデザインの動向調査、人間工学、Vol. 51、378-379、2015
- [2] J. Whiteside, J. Bennett, and K. Holtzblatt、Usability Engineering: Our Experience and Evolution、Handbook of Human Computer Interaction、M. Helander, (Ed.)、791-817、1988
- [3] Nielsen, J., and Molich, R.、Heuristic evaluation of user interfaces、Proc. ACM CHI'90 Conf.、Seattle, WA, 1-5 April、249-256、1990
- [4] Bruce Temkin、Mapping the Customer Journey、Customer Experience Professionals、February 5、2010
- [5] Svetlanta Ognjanovic., Jason Ralls., Pablo Gonzalez.、Testing by Teaching: Peer Tutoring, a Usability Evaluation Method for Children、User Experience Magazine、11(3)、2012
- [6] Kelley, J.F.、CAL - A Natural Language program developed with the OZ Paradigm: Implications for Supercomputing Systems、First International Conference on Supercomputing Systems、St. Petersburg, Florida, 16-20 December 1985、238-248、1985

付録A 提案手法の手順と参加者の役割

「お手軽ユーザー調査」	熟練者の注意点
1. 熟練者から初心者へ説明	情報取得方法についても忘れずに説明する
2. 初心者が使用 (思考発話)	初心者からの質問に答える
3. 熟練者が使用	何も気にせずいつものように操作する
4. 要望や質問など	

「お手軽ユーザー調査」	初心者の注意点
1. 熟練者から初心者へ説明	後で一人で使えるように意識する
2. 初心者が使用 (思考発話)	思考発話しながら操作する
3. 熟練者が使用	
4. 要望や質問など	

「お手軽ユーザー調査」	開発者の観察点
1. 熟練者から初心者へ説明	<ul style="list-style-type: none">・想定していた使い方と説明の違い・初心者が理解しにくい部分
2. 初心者が使用 (思考発話)	<ul style="list-style-type: none">・初心者の使い方・初心者が理解しにくい部分
3. 熟練者が使用	<ul style="list-style-type: none">・熟練者の実際の使い方 (初心者への説明と違うかも)
4. 要望や質問など	

第4分科会 (H20 チーム)

付録B 師匠と弟子の関係性を模する調査手法の参加メンバーの比較

		手法		
		コンテクスチュアル・インクワイアリー	Svetlanta らの手法	お手軽ユーザー調査 (提案手法)
UX 専門家	必要性	必要	必要	不要
	役割	弟子かつモデレーター	モデレーター	—
	ドメイン知識	必要	必要	—
	モデレーターとして操作の誘導	場合により実施	場合により実施	—
被験者	初心者ユーザーの募集	困難	困難	容易
	役割	師匠	にわか師匠 (子供) 弟子 (子供)	師匠 (熟練者) 弟子 (初心者)
	ドメイン知識	必要	不要	必要
	思考発話	可能	不可能	可能
	ヒアリング形態	個別	同時	同時