

UX と UI のデザインスキル向上のための概念モデル学習システムの提案

松山 聡志^{†1} 千石 靖^{†2}

近年、Web やデバイス、ソフトウェア技術の発達により、誰もがクリエイターになれる時代になった。そして、映像作品や Web サービス、モバイルアプリケーションなどのプロトタイプや制作過程などを習得できる Web ページや動画サービスが多くリリースされている。だが、Web で技術を習得したクリエイターはサービス作りにおける概念設計が不十分なため、コンセプトやデザインのクオリティが低く、使うユーザに取ってわかりにくいサービスが多い。そこで本研究では、クリエイターのサービス制作における概念設計の技法を習得する教育システムを構築する。その第一歩として、UX の UI のデザインスキルを向上させるため、ドナルド・アーサー・ノーマンが提唱したシグニファイアを学習できる Web サービスを制作する。このシステムには効果的、効率的に技法が習得できる認知的徒弟制を適用する。そして、制作したサービスの有用性を学術的に評価実験し、UX と UI のデザイン教育における工学的な設計指針を構築する。

Proposal of conceptual model learning system for improvements of UX and UI design skills

SATOSHI MATSUYAMA^{†1} YASUSHI SENGOKU^{†2}

As technical improvements in Web, devices, and software, everyone gets opportunities to become a creator. A number of Web sites and video services have been released that help learning basics of video production, Web development, and mobile application. However, many creators who acquired their knowledge through Web tend to have a lack of concept design in creating service, which lead them to create many non-user-friendly services. In this research, we are going to develop a learning system for creators to learn conceptual design ability in service development. For the first step we are going to create a Web service that help improve creators' UX and UI design skills by understanding conceptual model which was advocated by Donald Arthur Norman. The system will have "Cognitive Apprenticeship Approach" which makes efficient and effective for creators to learn technical skills. Evaluation experiment will be conducted to see whether the system has academic usability and construct engineered design guide for education of UI and UX.

1. はじめに

近年、通信インフラストラクチャーの設備や、新型デバイスの開発により、インターネット利用人口が増加し、Web サービスやモバイルアプリケーションが普及した。それに加え、2002 年一般家庭のインターネット回線が ISDN から ADSL へと移行し、ISDN では不可能であった動画を利用したサービスが利用されるようになった[1]。また、2003 年には無料ブログが登場したことで、個人でも容易かつ気軽にブログを立ち上げることが可能になった。そのため、アクティブなブログ数が 2004 年から 2006 年にかけて急激に増加した[2]。それらの事象により、クリエイターによる YouTube[3]やブログでの Web サービス、モバイルアプリケーションの制作方法を紹介する動画や記事が多くなった。本論文のクリエイターとは「デザイナー」、「エンジニア」、「運営者」を含めた創り手を意味する。現在では知識の無い一般のユーザでも、3 分間の動画でプログラミングを学習できるサイト「ドットインストール[4]」やオンラインでプログラミングを学ぶ「CODEPREP[5]」などのプログラミ

ングの学習からプロトタイプまで制作できる Web サービスを利用することにより、誰もがクリエイターになれる時代となった。

しかし、クリエイターが増える一方でデザイン性が低く、コンセプトが分からないといったサービスやアプリケーションが数多くリリースされている。それにより、ユーザが求めているサービスやアプリケーションが埋もれ、探す時間や、比較、選定をする手間が増えた。そのため、現在新しいサービスやアプリケーションを制作するクリエイターには、ユーザが分かりやすく、使いやすいといった設計力が求められている。

現在 Twitter Bootstrap[6]など視覚的なデザイン性を高めるサービスが登場しているが、分かりにくいサービス、使いにくいアプリケーションは増加し続けている。このことから、ユーザが「分かりやすい」、「使いやすい」と感じるのは視覚的なデザイン設計だけでなく、「概念モデルを考慮する必要がある」と考えられる。

そこで本研究では、D・A・ノーマンの著書「複雑さと共に暮らす デザインの挑戦」をもとに、「複雑さについて」、「概念モデル」、「シグニファイア」、「構造」、「モジュール」、「自動化」、「UX (ユーザーエクスペリエンス)」、「UI (ユーザーインターフェース)」を学習できる Web サービスを

^{†1} 金沢工業大学大学院
Kanazawa Institute of Technology Graduate School
^{†2} 金沢工業大学大学院
Kanazawa Institute of Technology Graduate School

制作する。このシステムには効果的、効率的に技法が習得できる認知的徒弟制を適用する。そして、制作したサービスの有用性を学術的に評価実験し、UX と UI のデザイン教育における工学的な設計指針を構築する。

2. 学習内容について

本研究では、D・A・ノーマンの著書「複雑さと共に暮らす デザインの挑戦」の翻訳者であり、ノーマンの主張を考察した岡本らの研究[7]を参考に学習システムを構築する。

まず初めに、分かりにくさと複雑さについて説明し、クリエイターに複雑さは避けられないものであり、必要な物であることと、クリエイターはその複雑さを扱い易くしなければならぬことを理解してもらう。

次に、ユーザが操作やシステムを理解し、複雑さを管理して扱いやすくするために必要な「概念モデル」をメインで学習させる。

最後に、より複雑さをわかりやすくするためのコミュニケーション手段として、ユーザ体験の満足度を考え、混乱や面倒なしで顧客の的確なニーズを満たし、所有や使用する楽しさを生み出す「UX」、ユーザを適切な行動へ導く「シグニファイア」、複雑になってしまう状況を簡素化する「構造」、複雑な構造を小さな扱いやすいモジュールに分割する「モジュール化」、タスクを減らすための「自動化」を学習させる。

以上のように、提案する学習システムでは、複雑さについて理解してもらい、システムの理解や使いやすさを体感させる概念モデルと、複雑さをわかりやすくするためのコミュニケーション手段として、「UX」、「シグニファイア」、「構造」、「モジュール化」、「自動化」を学習させる。

2.1 複雑さについて

「複雑である」とことと「分かりにくい」ことは別であり、文献[8]では、“「複雑さ」という言葉は、世界の状態を表すために用いている。「分かりにくい」という言葉は、心の状態を表す。”つまり、分かりにくいことが問題であり、「分かりにくさ」は「困惑させる複雑さ」と言い換えることもできる。複雑さが管理されていない時に分かりにくさが発生し、その結果、ややこしく、混乱させられ、苛立たしく、使いにくさに繋がってしまう。複雑さを管理することによってサービスは扱いやすくなる。

これを文献[9]では、図 1 のような電卓アプリで例えている。およそ 50 個のキーが整然と並ぶ関数電卓であるが、全てのキーが理解できなくても使うことができる。これは、論理的なまとまりで組織化されているため、理解できないキーが存在しても、ユーザが使用できるのである。簡素化は装置の問題であるのと同じく、心の問題である。もし、キーがランダムに配列されていた場合、簡単だった電卓が

難しくて混乱するものになる。クリエイターは複雑さを減らすのではなく、複雑さを扱いやすくすることが重要である。

このように、「複雑さ」は外的な状態に基づき、「分かりにくさ」は内的な感情に基づく。複雑さを減らすことや、簡素化することは実際の問題解決にはならず、複雑さをわかりやすくすることでサービスの使いやすさが出てくる。



図 1 関数電卓

2.2 概念モデル

文献では、“概念モデルとは、モノがどう動くかについて人が持つ基本的な信念構造である[10]”と述べている。つまり、概念モデルとは人がシステムの動きに対して思い描くと考えられるイメージモデルである。システムを使うユーザのイメージモデルと実際のシステムのイメージが等しい場合、ユーザは思い描いた通りの操作ができるため使いやすい、分かりやすいと感じる。よって、使いやすさ、分かりやすさというのはユーザの概念モデルが実際のモノに表れているかに依存するものであるといえる。

文献[11]では、これをコンピュータのファイル構造で例えている。ユーザがコンピュータのファイルのあるフォルダから別の場所へ移そうとすると、コンピュータのファイル構造を見るが、本来ファイルは、コンピュータのシステム上のディスプレイで映るようにフォルダにまとめられて存在しているわけではない。「コンピュータの記憶装置の中に蓄積されているもの」を、デザイナーが「ユーザの概念モデルを読み解き、設計したもの」がディスプレイに映っているのである。

このようにユーザが思い描く操作イメージと実際のシステムのイメージが類似している場合、ユーザが操作やシステムを理解できるため、分かりやすく、使いやすいと感じることができるのである。

2.3 UX

UX とは、“製品やサービスに関わるあらゆるユーザ体験の満足度を総合的に評価しようという概念[12]”のことであり、D・A・ノーマンは、“UX の第一の要件は、混乱や面倒なしで顧客の的確なニーズを満たす事。第二の要件は、

所有する楽しさ、使用する楽しさを生み出す「簡潔さと優雅さ」である。真のユーザーエクスペリエンスは、顧客の要望に答えることや、機能を充実するのでは十分ではなく、「多角的な専門分野のサービス」のシームレスな結合が必要なのである。[13]”と述べている。サービスは多くの場合、複雑なシステムで成り立っている。複雑さをわかりやすくする唯一の手段は、すべての UX を全体的にデザインすることである。ノーマンは、“効果的なサービスのデザインは、テクノロジーではなく、人のマネジメントである[14]”と述べている。UX を全体的にデザインするために考え方の基盤となる方法の一つとして、人間支援のデザインがある。人間支援のデザインとは、機械（システム）に知性、礼儀、共感、理解などの社会的マナーを持たせることであり、これらを持たせることによって、UX が向上し、ユーザのサービスに対する満足度の向上につながる。

2.4 シグニファイア

文献では、“シグニファイアは、それが意図的なものであれ、意図的ではないものであれ、適切な行動への知覚的なサインである。[15]”と述べている。つまり、シグニファイアはユーザが知覚し解釈するものであるため、クリエイターは製品にシグニファイアを適切に取り入れることによりユーザとの有効なコミュニケーションが可能になり、すなわち、ユーザのとりうる行動や適切な使い方を導くことが可能となる。

これを文献[16]では、著者は塩と胡椒の容器に例えている。図 1 はそれぞれ塩と胡椒の容器である。区別するのは非常に難しい。この容器では中身は詰める人のやり方（文化）に依存してしまい、文化という複雑さを例示させるものである。そこで複雑さを解消する、つまり適切な使い方をしてもらうためには、クリエイターは、意図的にシグニファイアを使うべきなのである。シグニファイアを使う場合には、誰もが同じ知識を持っていること、「社会的同期」が必要となる。容器の中身を区別（社会的同期）する方法は様々である。塩と胡椒は見ればわかるという社会的同期があるため、図 2 のように中身が見えるような容器に変えるという方法も一つである。これによって、誰もが中身を把握することができる。

このように社会的同期によってユーザとのコミュニケーションが可能になるため、クリエイターはユーザの立場を考えるといった特別な努力が必要であるといえる。シグニファイアというのはユーザに正しく知覚されなければ、目的を達成することができない。よって、文化的な仕組み、社会的システム、社会的な解釈を熟知した上で使う必要がある。



図 1 塩と胡椒の容器[17]



図 2 中身を見せることで社会性を持たせた
塩と胡椒の容器[18]

2.5 構造

文献では、“複雑になってしまう状況を簡素化する一つの方法は構造を与えることである。一つひとつが簡単で学びやすい、取り扱いやすいモジュールへとタスクを構造化する。[19]”と述べている。

これらを銀細工の作業台を例に説明している。図 3 の作業台だけ見ると複雑そうに見えるが、それらは道具ごとに修得され、道具を学ぶことで使いこなすことができ、複雑さを理解できるタスクにモジュール化している。そのため、複雑そうに見える作業台も、銀細工師にとっては理解可能で多くの簡単な道具の意味のある集まりとして見える。

このように、一見複雑そうに見えるものであっても、構造を与えることによって、全体のシステムを理解しやすくなり複雑さをわかりやすくすることができる。

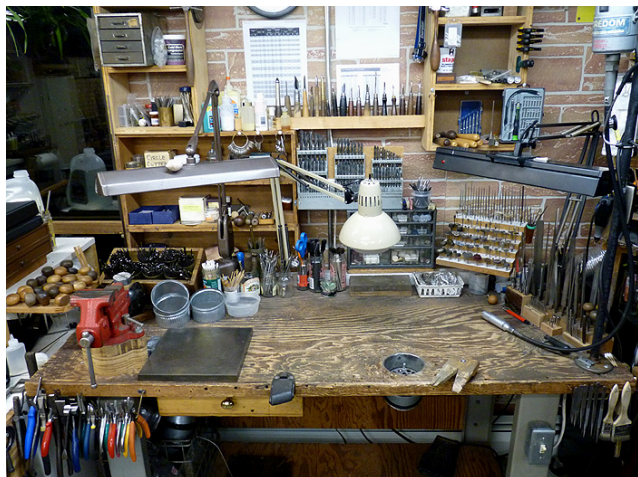


図 3 銀細工師の作業台[20]

2.6 モジュール化

文献では、“モジュール化とは複雑な構造を小さな扱いやすいモジュールに分割することである．[21]”と述べている．これを利用しているのが、図 4 のようなプリント、スキャン、コピー、ファックスの機能がついた複合機である．各機能はグループに分けられ、各機能をアイコンで表示している．このように、なにをしたいのかという活動を選択するという、活動を中心とした設計「活動中心設計」はユーザの実際の要求をモデル化しているので、複雑になりがちなボタンの集まりを適切にモジュール化し、比較的単純に扱える．



図 4 リコー複合機[22]

2.7 自動化

文献では、“自動化により、タスクを行う必要がなくなる[23].”と述べている．

最近のサービスでは、ユーザの行動を自動化することにより、操作が簡単になっている．例えば、ネットショッピングサイト Amazon での 1-Click 機能や、車の自動アイドリングストップ機能に、自動化が施されている．自動化はテクノロジーに根本的な複雑性を加えるが、その作業をするユーザの観点からすれば、行動を簡単にする．

しかし、簡単になるのは、自動化がうまく機能している限りにおいてである．自動化が破綻すると、全く自動化されていなかった時よりも、そのタスクは複雑になる．同様

に部分的な自動化は、完全な自動化や自動化なしに比べて問題が多い．なぜなら、自動状態と非自動状態の間を切り替えることから混乱と複雑さが生じるからである．

このように、複雑さの取り扱いにおいて重要な点は、“自動化は簡素化において最も有効な戦略だが、それは頑健で信頼できるシステムによって、機能が完全に自動化された場合に限る．[24]”ということである．

3. システム概要

本研究のシステムについて、Collins らによって示された、学習面において指導者が体験者に対して効率的に教えることができる認知的徒弟制[25]を使用する．使用するにあたって、張[26]、宮田[27]らの研究を参考に導入していく．文献[26][27]では、認知的徒弟制は以下の 6 つのメソッドからなると述べられている．

- モデリング：熟練者が作業を行っている様子を、初心者が観察する．
- コーチング：初心者が熟練者のガイドを受けながら実践する．
- スキャフォールディング：熟練者による支援を徐々に減らす．
- リフレクション：初心者が行った実践を内省する．
- アーティキュレーション：初心者が知識、推理または実践プロセスを明瞭に表現する
- エクスペロレーション：様々な場面において実践する．

本研究ではモデリング、コーチング、スキャフォールディング、リフレクション、アーティキュレーションに着目し、認知的徒弟制に基づく学習支援システムに必要な機能を提案する．学習者がシステムを利用し、学習するプロセスを図 5 に示し、図 6 に完成予想図を示す．

学習支援システムは「複雑さについて」、「概念モデル」、「UX」、「シグニファイア」、「構造」、「モジュール化」、「自動化」合計 7 章あり、各章は図 5 のように複数の節(動画)で構成する．

モデリング

文献で使用されている例や、世の中にある成果物や製作過程の一つのモデリングと捉える．これらを元に動画を制作し、学習者の教材として扱う．

コーチング

学習者が動画視聴時または演習時に問題や疑問に遭遇する場合があり、その際にコーチングを行わなければならない．学習支援システムは図 6 のように構成しているため一章ごとに Q&A(質疑応答)が投稿できるページを制作し、節ごとにリンクを設定する．そして、節の内容を文章化し視

覚的にまとめたスライド、節ごとのまとめページ、一節の内容が理解できているかを確認するクイズを提供する。クイズでは、各問にヘルプ機能をつけ、学習者にヒントを表示し、理解を深めてもらう。Q&A ページで回答するのは、制作者だけでなく、このサービスを利用した学習者であることもある。

スキファールディング

様々な知識を理解し、使いこなせるためには、課題演習を重ねる必要がある。学習者はモデリングプロセスを経て(各章を受講後)課題演習を行う。課題演習はヘルプ機能を付けない。課題演習で間違いが出た際、間違い部分の節へ移動させ、間違い部分を再度学習できるように設定する。

リフレクション

Q&A ページの学習内容を整理し、Q&A を投稿するあるいは、Q&A を閲覧することにより、学習者のリフレクションを促進する。

アーティキュレーション

Q&A ページで、質問者が投稿する際に自分の知識や考え、問題点を明瞭に表現させる。

3.1 開発環境

プログラムの開発エディタには Aptana Studio 3[28]を用い、開発言語には HTML5 を使用して CSS で Web ページを制作し、Web アプリケーションを制作する際に、PHP 5.4.22 と JavaScript を使用し、サーバは PHP が使用できるレンタルサーバを利用する。今後スマートフォンでも学習できるサービスにするため、レスポンスな Web ページを制作する。その際に、CSS フレームワークには Bootstrap 3.2.0[29]を使用する。YouTube の動画をメインに利用するため、動画再生のパフォーマンスがスムーズである Google Chrome[30]を推奨ブラウザとする。スライド部分の機能については、ユーザの持つ概念モデルに適している(ショートカットキー機能が利用できる)ことや、自分のドキュメントを Web ページに簡易に埋め込むことができることから SlideShare[31]を利用する。

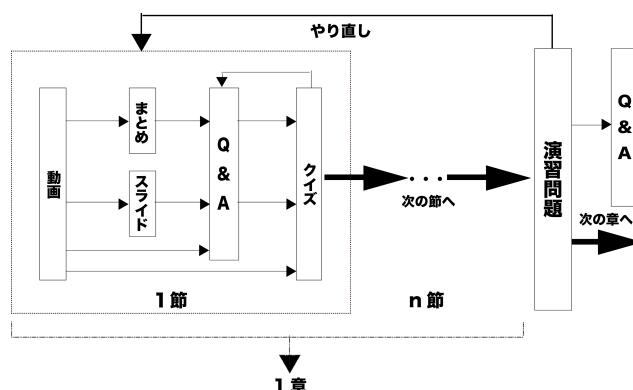


図 5 支援システムを用いた学習プロセス



図 6 完成予想図

4. 考察

本研究では、D・A・ノーマンの著書「複雑さと共に暮らす デザインの挑戦」をもとに、「複雑さについて」、「概念モデル」、「シグニファイア」、「構造」、「モジュール」、「自動化」、「UX (ユーザーエクスペリエンス)」、「UI (ユーザーインターフェース)」を学習できる Web サービスを開発する。本システムの教育サービスを行うことで、クリエイターの概念設計のスキルが向上すると考えられる。

近代化と生産性の向上を急ぎ、人間のエクスペリエンスの価値を見落としがちになっている今、知識の無い一般ユーザでも製品、Web サービスを創ることができる。これは、安価であるが大量生産で粗悪な製品も溢れていた産業革命と同じである。その時代、ウィリアム・モリスを主導とし、芸術と生活を一致させるという思想運動「アーツ・アンド・クラフツ運動」が行われ、産業界に大きな刺激を与え、20世紀のモダンデザインの源流にもなった。

だからこそ、現在、概念設計を学ぶことが重要であり、本研究はユーザをどれだけ考えられるかが重要になっている。本研究では認知的徒弟制を使用しているが、エクス

ロレーションの部分が足りていないのでその手法を考案して適切なアプローチを今後検討する必要がある。本教育システムが形になれば、様々なモノ作りに汎用できる。

5. まとめ

UX と UI のデザインスキル向上のための概念モデル学習システムの提案した。これにより、Web 界隈の作品のクオリティの向上と、ユーザの満足を得られると考える。

参考文献

- 1) プロバイダの歴史 / ADSL 時代
<http://www.hikari-navi.net/broadband/>
- 2) 総務省, ICT インフラの進展が国民のライフスタイルや社会環境等に及ぼした影響と相互関係に関する調査(2011).
- 3) YouTube <https://www.youtube.com/>
- 4) ドットインストール <http://dotinstall.com/>
- 5) CODEPREP <http://codeprep.jp/>
- 6) Twitter Bootstrap <http://getbootstrap.com/>
- 7) 岡本 明, 安村 通晃, 伊賀 聡一郎: 複雑さと共に暮らす: ドナルド・ノーマンの新著をもとに, 研究報告ヒューマンコンピュータインタラクション (HCI), 2011-HCI-144(2007).
- 8) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.2(2013)
- 9) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp. 12-13(2013)
- 10) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.37(2013)
- 11) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.37-40(2013)
- 12) Unix Design Office Inc. / ux-02. ユーザーエクスペリエンスの定義 <http://www.unix-d.co.jp/unix-hp/ux/102.shtml>
- 13) Unix Design Office Inc. / ux-02. ユーザーエクスペリエンスの定義 <http://www.unix-d.co.jp/unix-hp/ux/102.shtml>
- 14) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.177(2013)
- 15) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.253(2013)
- 16) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.104-110(2013)
- 17) デザインテクノロジーの最前線 / 複雑さとデザイン
http://www.daishinsha.co.jp/technology/column/design/design_n759.html
- 18) iwaki 塩・コショウ入れ KT5031BKSP
- 19) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.256(2013)
- 20) Jeffrey Herman / silver restoration & conservation
<http://hermansilver.com/shop1.htm>
- 21) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.257(2013)
- 22) RICOH / RICOH MP C8002/C6502,
https://www.ricoh.co.jp/mfp/mp_c/8002_6502/point/point2.html
- 23) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.264(2013)
- 24) D.A.ノーマン: 複雑さと共に暮らす デザインの挑戦, 新曜社, pp.265(2013)
- 25) Collins, A.: Cognitive apprenticeship, The Cambridge Handbook of the Learning Sciences, R. Keith Sawyer (Ed), Cambridge University Press (2006).
- 26) 張 龍明, 樋山 淳雄: 認知的徒弟制理論に基づくソフトウェア開発教育における強調学習支援システムの提案, 第 73 回全国大会講演論文集, pp. 545 - 547{2011}
- 27) 宮田 仁, 大隅 紀和, 林 徳治: プログラミングの指導方法

と問題解決能力育成との関連(3), 日本教育情報学会, 年會論文集 (12), pp.198-201, (1996)

- 28) Aptana Studio 3 <http://www.aptana.com/>
- 29) Bootstrap <http://getbootstrap.com/>
- 30) Google Chrome <https://www.google.co.jp/chrome/browser/>
- 31) SlideShare <http://www.slideshare.net/>