

Software prototyping

A case report of refining user requirements for a health information exchange dashboard

Scott D. Nelson¹; Guilherme Del Fiol²; Haley Hanseler²; Barbara Insley Crouch²; Mollie R. Cummins²

¹Vanderbilt University Medical Center, Nashville, TN, USA; ²University of Utah, Salt Lake City, UT, USA

Keywords

Software design, user-computer interface, human engineering/methods, medical informatics/methods

Summary

Background: Health information exchange (HIE) between Poison Control Centers (PCCs) and Emergency Departments (EDs) could improve care of poisoned patients. However, PCC information systems are not designed to facilitate HIE with EDs; therefore, we are developing specialized software to support HIE within the normal workflow of the PCC using user-centered design and rapid prototyping.

Objective: To describe the design of an HIE dashboard and the refinement of user requirements through rapid prototyping.

Methods: Using previously elicited user requirements, we designed low-fidelity sketches of designs on paper with iterative refinement. Next, we designed an interactive high-fidelity prototype and conducted scenario-based usability tests with end users. Users were asked to think aloud while accomplishing tasks related to a case vignette. After testing, the users provided feedback and evaluated the prototype using the System Usability Scale (SUS).

Results: Survey results from three users provided useful feedback that was then incorporated into the design. After achieving a stable design, we used the prototype itself as the specification for development of the actual software. Benefits of prototyping included having 1) subject-matter experts heavily involved with the design; 2) flexibility to make rapid changes; 3) the ability to minimize software development efforts early in the design stage; 4) rapid finalization of requirements; 5) early visualization of designs; 6) and a powerful vehicle for communication of the design to the programmers. Challenges included 1) time and effort to develop the prototypes and case scenarios; 2) no simulation of system performance; 3) not having all proposed functionality available in the final product; and 4) missing needed data elements in the PCC information system.

Correspondence to:

Scott D. Nelson, PharmD, MS
Principal Domain Specialist, EHR portfolio
Vanderbilt University Medical Center
Nashville, TN, USA
Email: Scott.Nelson@Vanderbilt.edu

Appl Clin Inform 2016; 7: 22–32

<http://dx.doi.org/10.4338/ACI-2015-07-CR-0091>

received: July 28, 2015

accepted: November 12, 2015

published: January 13, 2016

Citation: Nelson Scott D., Del Fiol Guilherme, Hanseler Haley, Crouch Barbara Insley, Cummins Mollie R. Software prototyping: A case report of refining user requirements for a health information exchange dashboard. Appl Clin Inform 2016; 7: 22–32
<http://dx.doi.org/10.4338/ACI-2015-07-CR-0091>

1. Background

Poisonings are a leading cause of unintentional injury deaths in the United States.[1] Poison control centers (PCCs) are 24-hour resources for poison information, clinical toxicology consultation, and poison prevention education. PCCs have been shown to reduce costs, unnecessary emergency department (ED) visits, and patient length of hospitalization [2, 3]. PCC specialists in poison information (SPIs) interface with healthcare professionals over the telephone and provide recommendations in the treatment of poisoned patients [4]. However, this reliance on telephonic communication can result in miscommunication, data loss, and error, especially in these settings of high communication loads, frequent interruptions, disaster scenarios [5], and/or routine surges in calls [6]. Health information exchange (HIE) has been proposed to improve information sharing between PCCs and EDs, thus improving decision-making, continuity of care, and reducing errors [7, 8]. Beyond just sending and receiving messages, this type of HIE requires careful consideration of the workflow and information needs of ED care providers and SPIs, as well as optimal display of the information to end-users, especially since system usability is a key factor in HIE use and success [9–12], and poor design can result in medical errors [13–16]. Currently, PCC information systems are not designed to facilitate HIE with EDs; therefore, we are developing specialized software to support the integration of HIE within the normal workflow of the PCC. While many development projects tend to design the technology at the programming code level, thereby removing the end user from the design process [17], we wanted to put the emphasis on the end user, in order to develop an attractive, effective, and efficient user interface combined with complex back-end functionalities [17]. This dashboard would then allow SPIs in the PCC immediate access to patient information from the ED, (e.g., general status, vital signs, progress notes, laboratory tests) through asynchronous communication. It would also allow SPIs to send consultation notes with patient assessment and recommendations to providers in the ED. The goal is to supplement telephonic communication and reduce the burden of communicating information that could be exchanged electronically.

2. Objective

The purpose of this report is to describe how we refined user requirements for a HIE dashboard to be used by SPIs in the PCCs through rapid prototype development, using a user-centered design approach.

3. Methods

In order to facilitate a user-centered design, we used rapid prototyping in iterative cycles that included feedback from target users on an interactive prototype of the proposed HIE software. A prototype is an example or a model of the intended end product, and ranges from low-fidelity quickly sketched drawings on paper, to high-fidelity interactive designs that look like the finished product. Low-fidelity prototypes are intended to show high-level design concepts, and do not represent details or what the final design of the product will look like. Conversely, high-fidelity prototypes are intended to closely resemble the user interface and functionality of the final product. There is a spectrum between low- and high-fidelity prototypes, which depends on how closely the prototype resembles the final product. In this project, we chose rapid prototyping with low-fidelity prototypes for the initial design process, then moved to usability testing with high-fidelity interactive prototypes. This rapid prototyping process helps to identify and solve major usability or interface problems quickly and cheaply, without having to rewrite the programming code [18], and allows us focus on the end-user design before spending efforts on the technology and programming code [17].

The participants were researchers and SPIs at an accredited regional PCC. Previously, we elicited user requirements for HIE via rigorous user interviews and analysis of PCC call recordings [4]. Using the list of requirements, we designed low-fidelity sketches by drawing ideas and designs out on paper with iterative refinement by the authors to define dashboard functionality. After a basic layout and functionality was established, we continued with iterative designs of an interactive high-

fidelity prototype using the web-based interactive prototyping software HotGloo (HotGloo GbR, Hamburg, Germany). This interactive design consisted of a set of screen design images, hyperlinked together in order to imitate the look, feel, and navigation of actual software (i.e. clicking on icons led to different screen images). While users could navigate through the screens with the appearance of using functional software, the design prototype was not functional software and users would not read or write any actual data. Next, we used expert-based methods of heuristics and usability guidelines to uncover potential usability problems with the design, such as simplifying and reordering the information displayed [19].

At the end of several more iterations, we conducted a formal scenario-based usability assessment with additional end users (experienced SPIs at the Utah PCC) that were not involved with the development process. A patient case vignette was developed based on a modified PCC case of snake envenomation with a subsequent ED encounter and inpatient stay [20, 21]. The case was then checked by SPIs to make sure that the case presented was representative of a typical poison control case and included the different tasks that are performed by SPIs (e.g., consultation, follow-up). SPIs were asked to think aloud while attempting to accomplish specific tasks related to the case vignette [22]. Sessions were recorded via screen capture with the usability testing software TURF [23]. In addition to the think-aloud techniques, a researcher (pharmacist and medical informatics fellow) observed the session, took field notes, and solicited additional open-ended comments and recommendations. After the session, the SPI filled out an online post-session interview using the System Usability Scale (SUS) [24]. SUS is a 10-question, 5-point Likert scale that provides subjective measurement of system usability. We anticipated that the input from approximately four users would be able to identify most usability problems during the iterative development [25]. A general overview of the methodology is presented in ► Figure 1.

4. Results

Overall, feedback from the users was very positive, and they were eager to contribute their ideas for dashboard improvement and additional functionality. Because prototype development was relatively simple, we were able to easily incorporate recommendations and feedback, and then continue to test with additional users. This type of iterative development helped users become engaged in the project during early stage design, and promoted willingness to assist with future testing and design.

We received three SPI survey results, showing better than average usability [26]. Overall, users strongly agreed that they could learn the system quickly, and had high satisfaction with accessing lab values, vital signs, medication lists, and progress notes in one location. The early rounds of testing showed the need to simplify the dashboard display by removing extraneous icons, excessive information, and unnecessary tabs. The users identified inconsistencies and missing data in the prototypes, along with preferences in how information was arranged on the screen, and defined additional information needs such as patient date of birth and medical record number (MRN) for patient matching. User feedback was then incorporated into the design, and helped make the dashboard more intuitive and user friendly [26]. ► Figure 2 and ► Figure 3 show an evolution of major revisions to the dashboard display and patient summary prototypes. Additionally, not shown, when users click on the outgoing message box, they are provided a prepopulated template to structure an outgoing message to the ED, using the HL7 clinical document architecture (C-CDA) consult note template [27]. After we came to a stable prototype through testing, we gave the interactive prototype to our developers to program the actual software and finish user-interface design details.

5. Discussion

In this paper, we describe how we used software prototyping to refine user requirements and design an HIE dashboard. We found this approach to be useful because it provided a method to engage end users early on in the project, and we could watch ideas quickly come to life, then test and refine the display with end users. It was also beneficial to have subject-matter experts heavily involved with the design and development of the prototype [11], as well as providing flexibility to make rapid changes

in iterative testing. Using the interactive prototype helped us to quickly identify usability shortcomings, and improve the dashboard design [28]. Then, the interactive prototype was used as an example of the system specifications to provide to the developers to program and work on the complex backend functionality. Typically, when programmers code, they may encounter problems, or be forced to make personal ad-hoc decisions on-demand about detail in the design and program code. An interactive prototype helps to communicate the final intended design and functionality, in a way that is quicker, easier, and more effective than supporting text descriptions. Even experienced programmers tend to rely more so on pictures in design guidelines than on the supporting text when making decisions on coding details [29]. Clear communication can help to save time and money by not having to rework or fix misunderstandings in the code later in development or usability testing [30]. If having a picture is worth a thousand words, then having an interactive picture is worth more. The prototype we developed facilitated rapid communication of ideas and established a common understanding of design and intended functionality, without needing to do software programming.

There are many options for developing prototypes or wire-frame designs, even for mobile devices [31, 32], and there exists many different methods for usability testing [22, 33, 34]. There does not appear to be one best answer for which prototyping software to use or usability testing method, and the selection of each depends on individual use cases, project requirements, and researcher experience. Successful use of prototypes really depends on properly eliciting user requirements early on, using appropriate tools, and being prepared to throw away designs as you work through several iterations [18, 35]. Additionally, rapid prototyping is intended to be short-term, so no substantial effort should be spent on each design cycle. Additional guidance for using high-fidelity prototypes are also described elsewhere [30].

5.1 Benefits

Benefits we observed by rapid prototyping include:

1. early construction in the development cycle with the ability to spread the development costs across the project;
2. no upfront large investment in time or money helps to finalize the system and user requirements [36];
3. early visualization of the possible designs promotes innovation [37];
4. and the prototype provided a power vehicle for communication of the interface design to the programmers [29].

5.2 Challenges

Some of the challenges and tradeoffs that we discovered relate to the development time required by high-fidelity prototypes and case scenarios. Low-fidelity prototypes are faster, cheaper, and could start out early with paper and pencil. There are many pros and cons related to using low- or high-fidelity prototypes [30], but we found it to be very useful to start with a low-fidelity prototype, and iteratively evolve it into a higher-fidelity prototype. Perhaps the greatest challenge is that the prototype appears to be a final product, which does not show system performance and some of the complex functions may not be feasible in the final product. Required changes from the prototype to the final product could affect usability or user satisfaction, as the absence of some functionality in the final product may disappoint users. We attempted to mitigate this by emphasizing with the users that the prototype represents the ideal software, and may not include all of the features and functionality represented. Additionally, we analyzed the current systems and message contents, and included domain experts to better understand the technical capabilities and limitations of the systems. Additionally, we found in this project that not all of the data elements needed by users are captured in the PCC information system (such as date of birth or MRN), and there is not currently a standard terminology for exchanging PCC data [38].

5.3 Study limitations

There are several limitations to this study that are worth mentioning. First, we had a limited sample size of 3 participants in the case-based usability testing. Yet, rapid and iterative design is not feasible with larger sample sizes. A full usability assessment is planned for once the software is developed. The second limitation has to do with dual process theory [39–41]. Automated thought process (“System 1”) uses a network of associations for rapid pattern-matching, while deliberate thought process (“System 2”) is a slower, rule-based, system for active reasoning. Experts tend to function with System 1 level thought, which is very difficult to articulate and verbalize. Additionally, most people stop talking when they are thinking or processing complex information, also making it difficult for experts to verbalize their goals and tasks [39, 42]. To address this limitation, we also used direct observation, screen capture, recorded PCC calls, interviews, and post-session questionnaires in obtaining system requirements and in the usability testing. Third, the usability testing sessions were carried out in the PCC; however, they were not done while the SPI was multitasking or taking other calls. Ideally, usability testing should be done in the intended healthcare environment, while the user is doing a challenging task [34]. Finally, usability testing included only one case vignette, a non-drug poison exposure. This served our goal of eliciting rapid feedback to support initial design. Use of multiple cases and different scenarios will help to identify additional usability problems and improve the system design [43]. Functional software is currently under development and formal usability testing using actual PCC cases is planned. Future usability testing will entail live documentation of PCC cases in the context of real-world operations. This will be accomplished by pairing a specialist taking live calls with a second, inactive specialist. The second specialist will participate only in usability testing of the software, using tandem headphones to follow and document live cases.

5.4 Context and future work

The rapid prototyping described in this case study was conducted as part of a larger effort to develop and evaluate a workflow-integrated HIE process for PCCs and Eds [27, 38, 44]. U.S. PCCs currently lack software and informatics tools to engage in standards-based health information exchange. We are developing this open source software to enable U.S. PCC HIE. The software is designed to facilitate both unidirectional and bidirectional HIE. In other words, PCCs can use the software to send a single document, an HL7 CCD consultation note to a recipient ED or regional HIE organization. With additional configuration and sufficient technical capability of ED information systems, it can be used to facilitate ongoing communication between the ED and PCC during active management of a poison exposure case, partially replacing and supplementing telephone communication. We are currently studying outcomes of the latter approach, which involves a high degree of workflow integration in both the ED and PCC setting, in collaboration with Intermountain Healthcare in Salt Lake City, UT.

6. Conclusions

We found the use of a rapid prototyping method for eliciting user requirements and for designing an information dashboard engaged users, and it helped design a better user-centered display. We started with low-fidelity prototypes and iteratively designed an interactive high-fidelity prototype that served as a software development specification for our programming team. The software is currently under development and more detailed usability testing is planned in order to both refine and evaluate the software.

Clinical Relevance Statement

This paper describes the benefits and challenges of using a software prototype for the development of health information technology in the poison control center setting. The iterative, user-centered method can provide a more user-friendly and intuitive program, better meeting the needs of the users. A good user interface design is critical to patient safety and end-user uptake.

Conflict of Interest

The authors declare that they have no conflicts of interest in the research.

Human Subjects Protections

The study was performed in compliance with the World Medical Association Declaration of Helsinki on Ethical Principles for Medical Research Involving Human Subjects, and was reviewed by our local Institutional Review Board.

Acknowledgments

We would like to thank Frank Drews, PhD for his help and review.

This study was funded by the US Agency for Healthcare Research & Quality (1 R01 HS021472-01A1).

At the time of this study, Dr. Nelson was supported by the VA Advanced Fellowship Program in Medical Informatics of the Office of Academic Affiliations, Department of Veterans Affairs. The views expressed in this article are those of the authors and do not necessarily represent the views of the Department of Veterans Affairs.

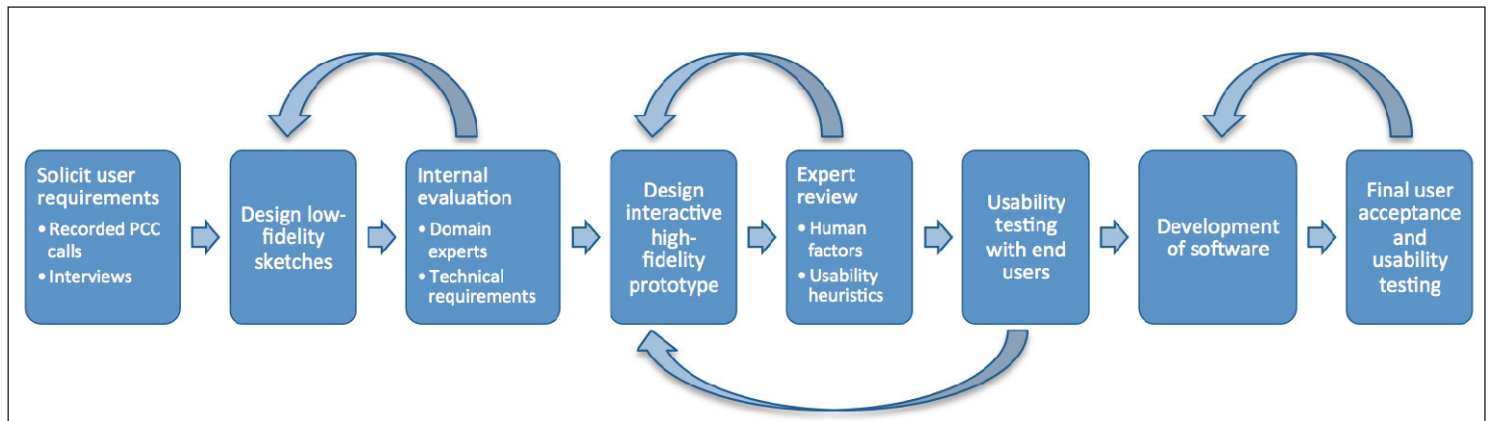
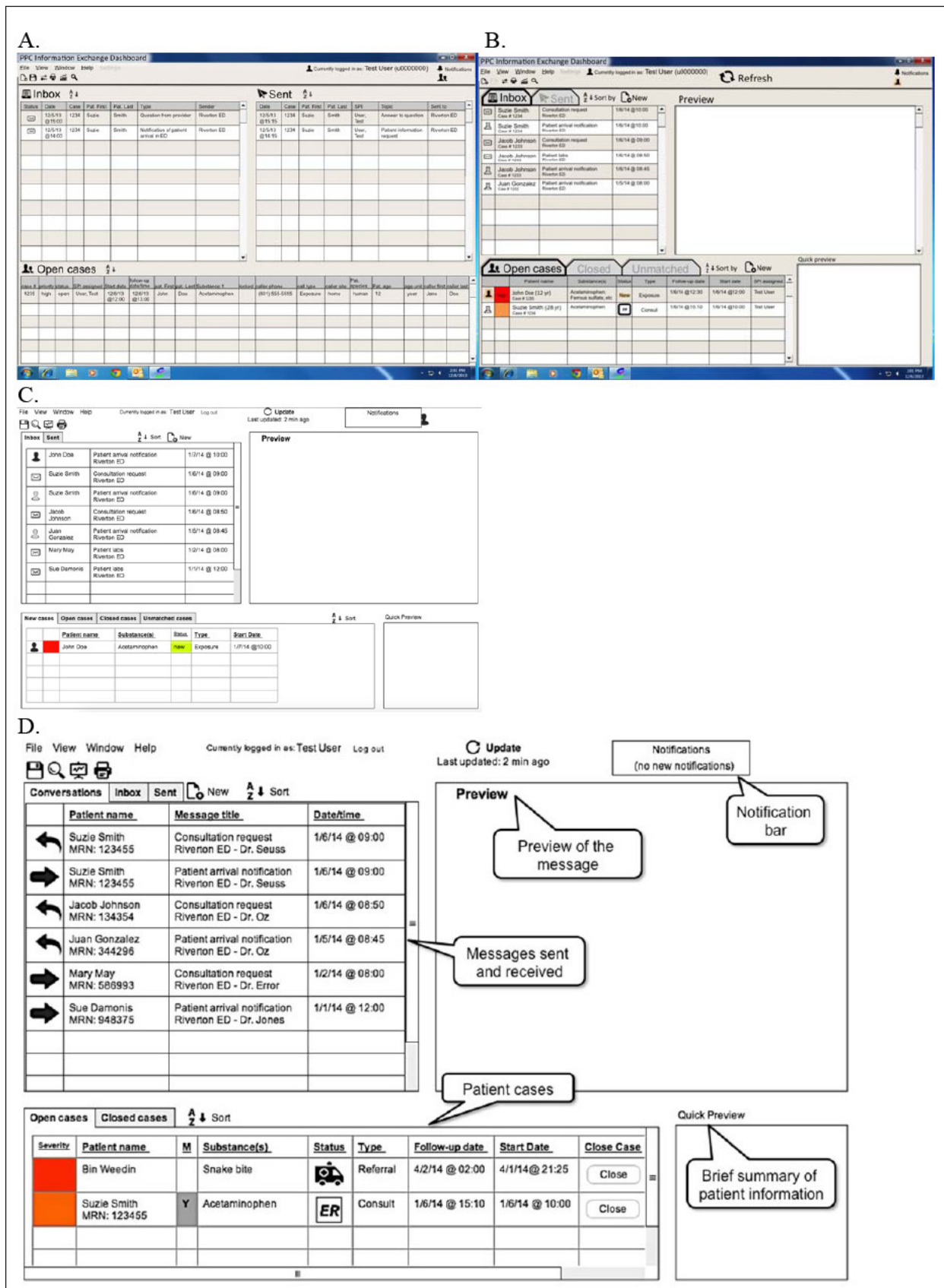


Fig. 1 Iterative development



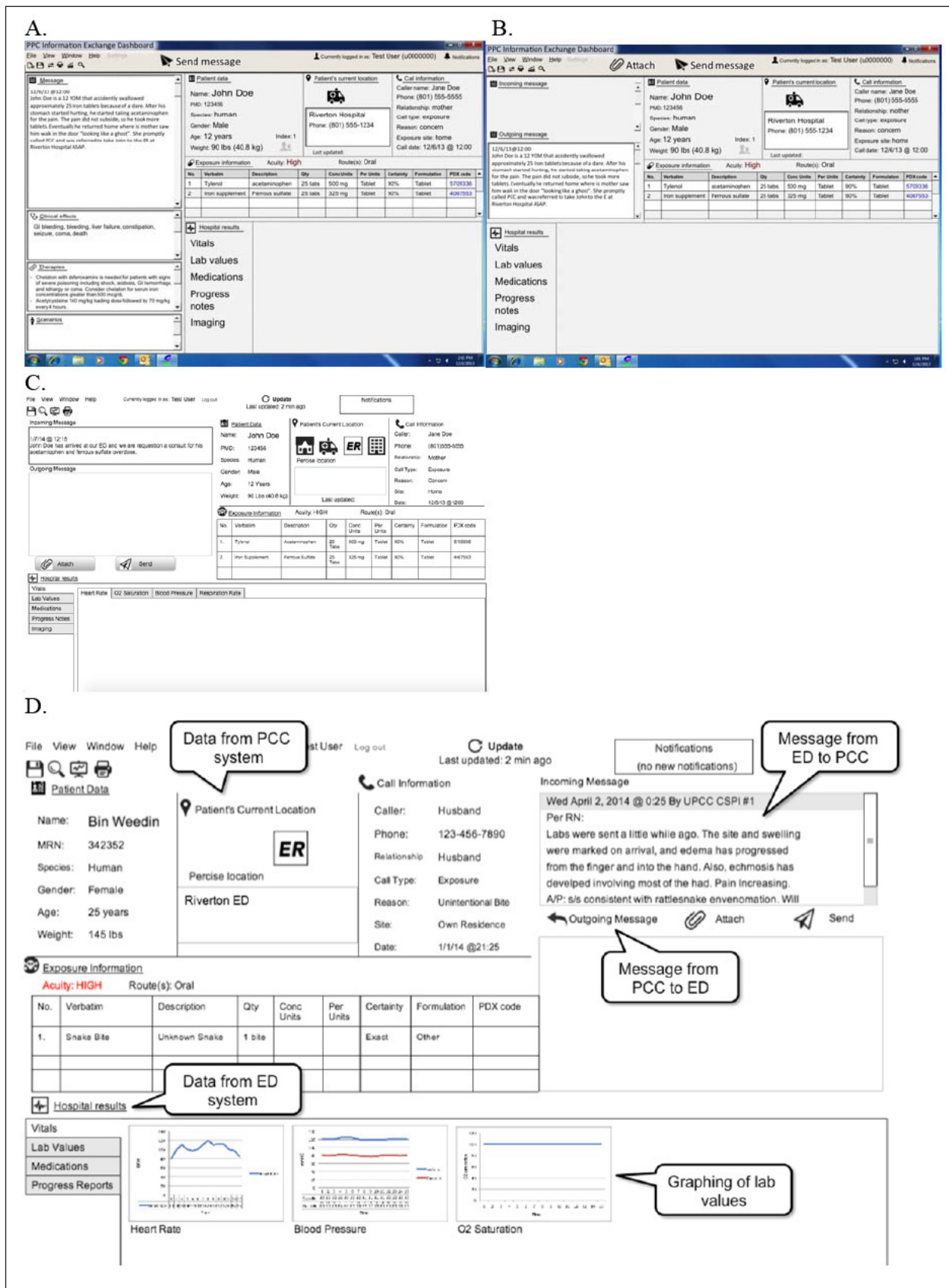


Fig. 3 Patient summary prototypes A through D. Figure D represents the final Patient summary prototype. Note: there were many other versions, but these represent the major revisions after user feedback.

References

1. Injury Facts, 2012 Edition. Itasca, IL: National Safety Council; 2012.
2. Friedman LS, Krajewski A, Vannoy E, Allegritti A, Wahl M. The association between U.S. Poison Center assistance and length of stay and hospital charges. *Clin Toxicol (Phila)* 2014; 52(3): 198–206. doi: 10.3109/15563650.2014.892125. PubMed PMID: 24580060.
3. LoVecchio F, Curry S, Waszolek K, Klemens J, Hovseth K, Glogan D. Poison control centers decrease emergency healthcare utilization costs. *J Med Toxicol* 2008; 4(4): 221–224. PubMed PMID: 19031372; PubMed Central PMCID: PMC3550108.
4. Cummins MR, Crouch B, Gesteland P, Wyckoff A, Allen T, Muthukutty A, Palmer R, Peelay J, Repko K. Inefficiencies and vulnerabilities of telephone-based communication between U. S. poison control centers and emergency departments. *Clin Toxicol (Phila)* 2013; 51(5): 435–443. Epub 2013/05/24. doi: 10.3109/15563650.2013.801981. PubMed PMID: 23697459.
5. Vassilev ZP, Kashani J, Ruck B, Hoffman RS, Marcus SM. Poison control center surge capacity during an unusual increase in call volume – results from a natural experiment. *Prehosp Disaster Med* 2007; 22(1): 55–58. PubMed PMID: 17484364.
6. Caravati EM, Latimer S, Reblin M, Bennett HK, Cummins MR, Crouch BI, Ellington L. High call volume at poison control centers: identification and implications for communication. *Clin Toxicol (Phila)* 2012; 50(8): 781–787. doi: 10.3109/15563650.2012.713968. PubMed PMID: 22889059.
7. Del Fiore G, Crouch BI, Cummins MR. Data standards to support health information exchange between poison control centers and emergency departments. *J Am Med Inform Assoc* 2014. doi: 10.1136/amiainf-2014-003127. PubMed PMID: 25342180.
8. Cummins MR, Crouch BI, Gesteland P, Staggers N, Wyckoff A, Wong BG. Electronic information exchange between emergency departments and poison control centers: a Delphi study. *Clin Toxicol (Phila)* 2012; 50(6): 503–513. Epub 2012/05/23. doi: 10.3109/15563650.2012.693183. PubMed PMID: 22612793.
9. Gould DJ, Terrell MA, Fleming J. A usability study of users' perceptions toward a multimedia computer-assisted learning tool for neuroanatomy. *Anat Sci Educ* 2008; 1(4): 175–183. doi: 10.1002/ase.36. PubMed PMID: 19177405; PubMed Central PMCID: PMC24532302.
10. Johnson KB. Barriers that impede the adoption of pediatric information technology. *Arch Pediatr Adolesc Med* 2001; 155(12): 1374–1379. PubMed PMID: 11732959.
11. Verhoeven F, Stehouder MF, Hendrix RM, van Gemert-Pijnen JE. Factors affecting health care workers' adoption of a website with infection control guidelines. *Int J Med Inform* 2009; 78(10): 663–678. doi: 10.1016/j.ijmedinf.2009.06.001. PubMed PMID: 19577956.
12. Weir CR, Crockett R, Gohlinghorst S, McCarthy C. Does user satisfaction relate to adoption behavior?: an exploratory analysis using CPRS implementation. *Proc AMIA Symp* 2000: 913–917. PubMed PMID: 11080017; PubMed Central PMCID: PMC2243831.
13. Fairbanks RJ, Caplan S. Poor interface design and lack of usability testing facilitate medical error. *Jt Comm J Qual Saf* 2004; 30(10): 579–584. PubMed PMID: 15518362.
14. Bates DW, Gawande AA. Improving safety with information technology. *N Engl J Med* 2003; 348(25): 2526–2534. Epub 2003/06/20. doi: 10.1056/NEJMsa020847. PubMed PMID: 12815139.
15. Horsky J, Kuperman GJ, Patel VL. Comprehensive analysis of a medication dosing error related to CPOE. *J Am Med Inform Assoc* 2005; 12(4): 377–382. doi: 10.1197/jamia.M1740. PubMed PMID: 15802485; PubMed Central PMCID: PMC1174881.
16. Koppel R, Metlay JP, Cohen A, Abaluck B, Localio AR, Kimmel SE, Strom BL. Role of computerized physician order entry systems in facilitating medication errors. *JAMA* 2005; 293(10): 1197–1203. doi: 10.1001/jama.293.10.1197. PubMed PMID: 15755942.
17. Brown S, Holzinger A, editors. Low cost prototyping: Part 1, or how to produce better ideas faster by getting user reactions early and often. *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction-Volume 2*; 2008: British Computer Society.
18. Heaton N, editor. What's wrong with the user interface: How rapid prototyping can help. *Software Prototyping and Evolutionary Development*, IEE Colloquium on; 1992: IET.
19. Johnson J. Designing with the mind in mind: Simple guide to understanding user interface design rules: Morgan Kaufmann; 2010.
20. Vincent CJ, Blandford A. Usability standards meet scenario-based design: Challenges and opportunities. *J Biomed Inform* 2014. doi: 10.1016/j.jbi.2014.11.008. PubMed PMID: 25460202.
21. Nelson S, Del Fiore G, Hanseler H, Crouch B, Cummins M. A dashboard for health information exchange between poison control centers and emergency departments: the poison control center view. 2014 Annual Meeting of the North American Congress of Clinical Toxicology (NACCT); Oct 2014; New Orleans, LA: *Clin Toxicol* 2014; 682–818.

22. Jaspers MW. A comparison of usability methods for testing interactive health technologies: methodological aspects and empirical evidence. *Int J Med Inform* 2009; 78(5): 340–353. doi: 10.1016/j.ijmedinf.2008.10.002. PubMed PMID: 19046928.
23. Zhang J, Walji MF. TURF: toward a unified framework of EHR usability. *J Biomed Inform* 2011; 44(6): 1056–1067. Epub 2011/08/27. doi: 10.1016/j.jbi.2011.08.005. PubMed PMID: 21867774.
24. Brooke J. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 1996; 189: 194.
25. Nielsen J. Estimating the number of subjects needed for a thinking aloud test. *International journal of human-computer studies* 1994; 41(3): 385–397.
26. Nelson SD, Del Fiore G, Hanseler H, Crouch B, Cummins M. A dashboard for health information exchange between poison control centers and emergency departments: the poison control center view. *Clin Toxicol* 2014; 52: 682–818.
27. Del Fiore G, Crouch BI, Cummins MR. Data standards to support health information exchange between poison control centers and emergency departments. *J Am Med Inform Assoc* 2015; 22(3): 519–528. doi: 10.1136/amiajnl-2014-003127. PubMed PMID: 25342180.
28. Nielsen J. Paper versus computer implementations as mockup scenarios for heuristic evaluation. *Proceedings of the IFIP TC13 Third International Conference on Human-Computer Interaction*. 725312: North-Holland Publishing Co.; 1990; 315–320.
29. Tetzlaff L, Schwartz DR. The use of guidelines in interface design. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*; New Orleans, Louisiana, USA. 108936: ACM; 1991. p. 329–33.
30. Rudd J, Stern K, Isensee S. Low vs. high-fidelity prototyping debate. *interactions* 1996; 3(1): 76–85.
31. Humayoun SR, Hess S, Kiefer F, Ebert A, editors. i2ME: a framework for building interactive mockups. *Proceedings of the 15th international conference on Human-computer interaction with mobile devices and services*; 2013: ACM.
32. Ko Y-j, Jung H, Kim E, Keum E. Suggestions of Prototyping Tool for Service Design Co-Creation. *Focusing on Mobile Application* 2013.
33. Saitwal H, Feng X, Walji M, Patel V, Zhang J. Assessing performance of an Electronic Health Record (EHR) using Cognitive Task Analysis. *Int J Med Inform* 2010; 79(7): 501–506. doi: 10.1016/j.ijmedinf.2010.04.001. PubMed PMID: 20452274.
34. Bhutkar G, Konkani A, Katre D, Ray GG. A review: healthcare usability evaluation methods. *Biomed Instrum Technol* 2013; Suppl: 45–53. doi: 10.2345/0899-8205-47.s2.45. PubMed PMID: 24111771.
35. Vijayan J, Raju G. Requirements Elicitation Using Paper Prototype. *Advances in Software Engineering*: Springer; 2010; 30–37.
36. Zhang J, Chung J-Y. Mockup-driven fast-prototyping methodology for Web application development. *Software: Practice and Experience* 2003; 33(13): 1251–1272. doi: 10.1002/spe.547.
37. Virzi RA, Sokolov JL, Karis D, editors. Usability problem identification using both low-and high-fidelity prototypes. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*; 1996: ACM.
38. Cummins M, Crouch B, Del Fiore G, Mateos B, Muthukutty A, Wyckoff A. Information Requirements for Health Information Exchange Supported Communication between Emergency Departments and Poison Control Centers. *AMIA Annu Symp Proc* 2014: 449–456.
39. Kahneman D. *Thinking Fast and Slow*. New York: Farrar, Straus and Giroux; 2011.
40. Smith ER, DeCoster J. Dual-Process Models in Social and Cognitive Psychology: Conceptual Integration and Links to Underlying Memory Systems. *Personality and Social Psychology Review* 2000; 4(2): 108–131.
41. Croskerry P. Clinical cognition and diagnostic error: applications of a dual process model of reasoning. *Advances in health sciences education* 2009; 14(1): 27–35.
42. Cooke L, Cuddihy E, editors. Using eye tracking to address limitations in think-aloud protocol. *Professional Communication Conference, 2005 IPCC 2005 Proceedings International*; 2005: IEEE.
43. Borycki E, Kushniruk A. Identifying and preventing technology-induced error using simulations: application of usability engineering techniques. *Healthc Q* 2005; 8 Spec No: 99–105. PubMed PMID: 16334081.
44. Cummins M, Crouch B, Del Fiore G, Greene T, Allen T, Naurs S. *Electronic Exchange of poisoning Information AHRQ 1R01HS021472-01A1*. Salt Lake City, UT 2013