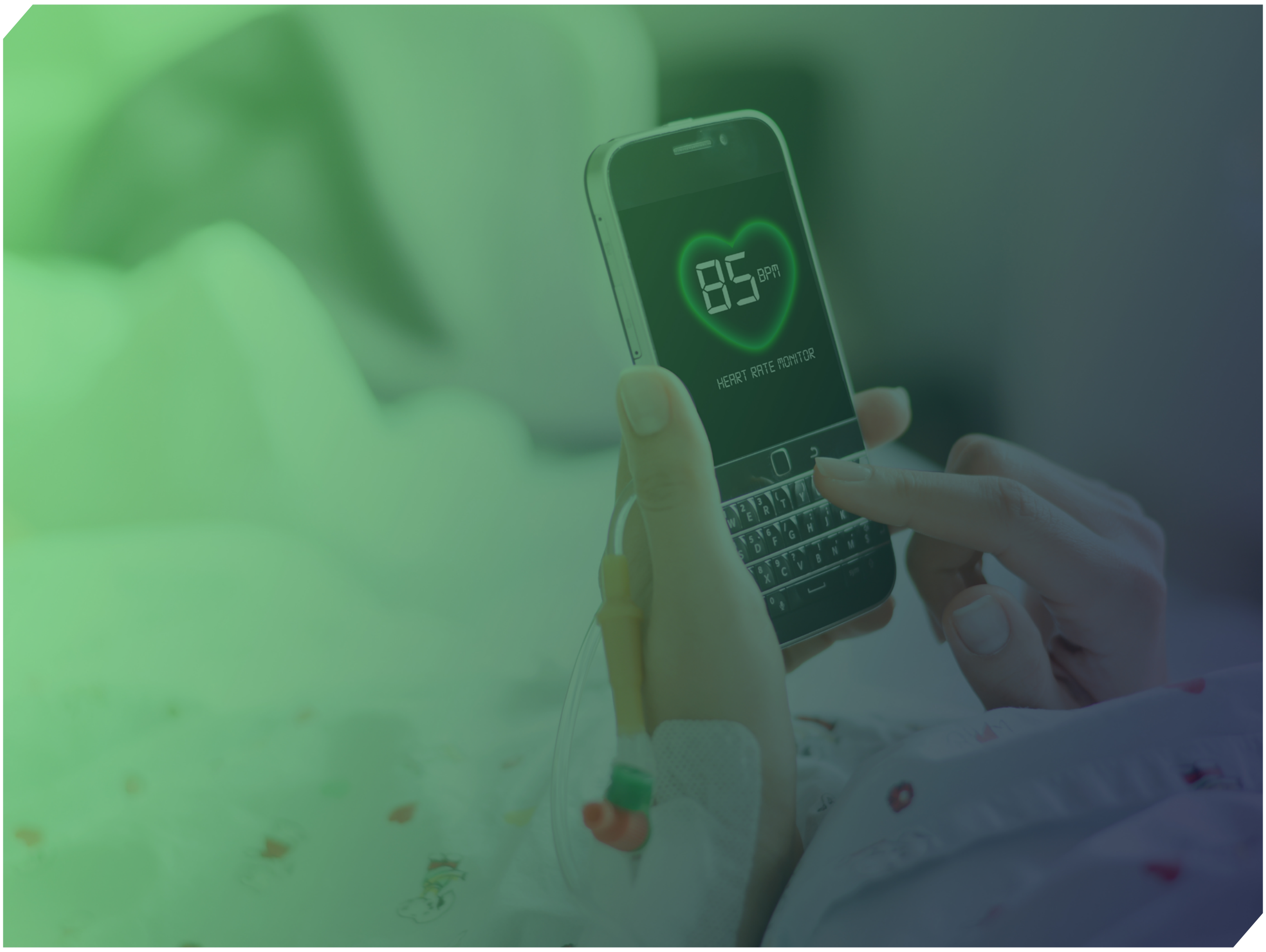
WHITEPAPER

Qt

MEDACUITY

Leveraging Human Factors Testing to Develop a Better MedTech Product



Contents

Overview

Usability and human factors testing are becoming requirements rather than “nice-to-haves” across a wide array of industries that make products we use every day in both our professional and personal lives. The pace at which technology is evolving is faster than ever and innovation is paramount to a company trying to differentiate itself from its competition and “the norm”. Social media platforms help drive the pace at which the newest innovations become known and accessible. The question that is often lost in this rush of technology is, is that innovative technology usable and scalable?

In unregulated industries, where there may be few risks to the safety of the user or subject, adoption of the new technology is the highest concern and functionality critical to the safety of the user or subject is a secondary concern. If, however, a user finds a new product difficult, clumsy, or non-intuitive to use, it will not be adopted no matter how technically innovative. Word of difficulties experienced using the product will spread quickly through the same social media channels that raised the initial interest and drive down the adoption of the new technology. This lack of adoption will be reflected in sales and the product itself will become unsustainable by the manufacturer.

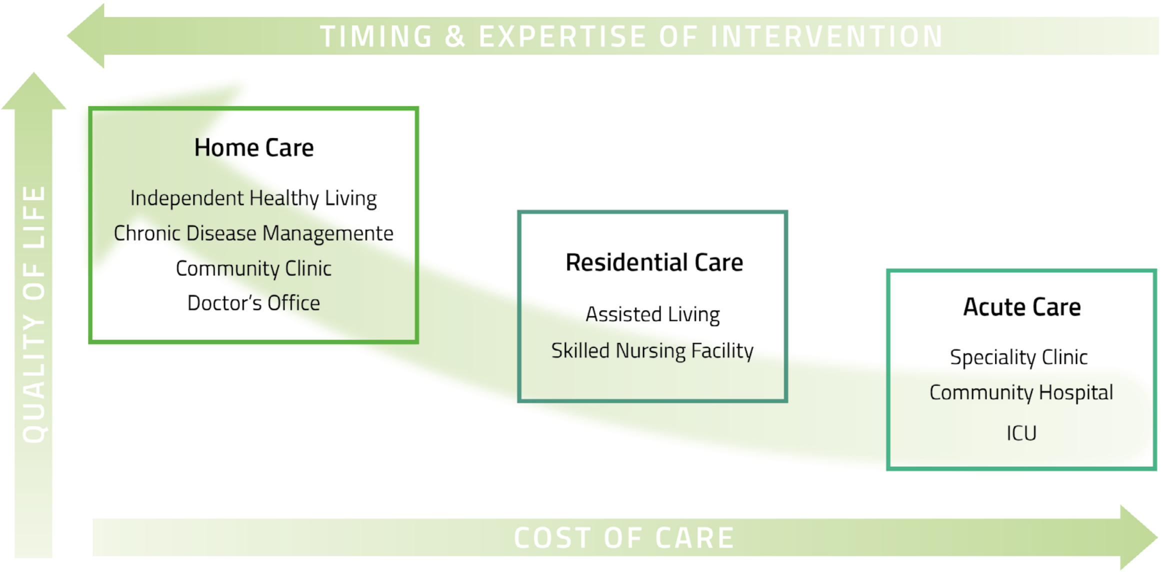
In a highly regulated industry like the medical device industry, this same concern of adoption exists but it is not the most critical one. In the medical device industry, it is the primary mission of the manufacturer to bring to the market products that are both safe and effective. Safe to both the patient and the user, and effective in diagnosing and/or treating a patient’s condition. Usability in this context is relevant to both the safety of the device (a non-intuitive device may lead to mistakes) and its effectiveness (a difficult feature may not get used).

The words safe and effective are, in and of themselves, very succinct, but executing upon this mission can in reality be quite complex. There is a multitude of standards, submissions, reporting requirements, and overlapping jurisdictions that seek to both guide and constrain the process of identifying and managing the safety and efficacy of a medical device product. There are three major types of risk that medical device developers need to consider:

- Clinical Risk - The risk of a device not functioning the way it is intended
- Cybersecurity Risk - The risk of device intrusion that jeopardizes its intended use, the accuracy of the data it contains, or the privacy of that data
- Human Factors Risk - The risk of the device being used incorrectly which jeopardizes patient and/or user safety

The goal of the device manufacturer is to mitigate these types of risks to their lowest possible levels. Historically, human factors risk in medical devices has taken a back seat to clinical and cybersecurity risks. This can, on the surface, seem like a reasonable approach since devices needed to operate and deliver the intended clinical results and be protected against security vulnerability. The industry and regulatory agencies have, however, come to realize that the usability of a device cannot be separated from the safety or security of the device and thus not be ignored or prioritized lower than other concerns. It's the numbers that have drawn this attention to the usability of medical devices and software. More than a third of medical device incidents involved a use error and more than half of all device recalls for design problems involve the user interface [1][2]. The fact that a device "works" is not enough if the user can't get the device to work. "Usable" medical devices must take into consideration safety, effectiveness, and adoption.

"Use" and "Useable" have already been referenced multiple times in this publication. A fair question is "Usable by whom?". Medical devices have traditionally been used in clinical settings like hospitals and medical offices by trained users. Today, however, a number of factors including the cost of hospitalization and an increasing demand for health services from an aging population is driving a trend toward the expanded use of medical devices outside of the hospital and clinical environments. Patients, insurance companies, hospital administrations, and device manufacturers all want to keep hospital stays as brief as possible, if needed at all, and readmittance to the hospital at a bare minimum.



Source: Eric Dishman, Intel Corporation (presented October 1, 2014, IOM-NRC Workshop on "The Future of Home Health Care").

Thus, healthcare is being driven out of the hospitals into alternative care centers and care in the home. With this shifting trend in the healthcare ecosystem, more types of users will have their hands on an increasing variety of medical devices and medical technology. The usability of these devices and technology will have to take into consideration human factors such as a person's age, education, clinical expertise level, and technical expertise level, just to name a few. More users and more user types increase the risk of a device being used improperly and improper use directly impacts a patient's safety.

Typically, improper use of a device is neither malicious nor is it intended. Improper use can stem from a device not being intuitive, being unpleasant to use, or being bothersome (think alarm fatigue) among other contributing factors. Picture yourself a nurse at a bedside with a device you find either difficult or bothersome to use. The device might not be intuitive or the process by which it is intended to be used is arduous. You, as the nurse, could choose not to use this device. Get too many nurses doing the same, the hospital will notice the lack of adoption, not recognize a return on their investment, and will not make a follow up purchase. If the reputation of the company is that the devices they develop are difficult to use, then the sale of other products might suffer as well. This could turn into a veritable slippery slope.

Often, however, adoption is not the issue if the device is the only option for the task at hand and non-use is not an option. This kind of scenario might cause a user look for an easier way to use the device than what has been prescribed; a "hack around". This occurs more than you might think. Anyone who has spent time in a hospital has at times heard nurses proclaim "Oh this device is so annoying. I don't know why it's doing that. Here, I just do this (press this button, type this command, turn off this function) to get it to stop". All of which beg the questions

- Is this device being used properly?
- Is this device functioning properly?
- Is my loved one's safety at risk?

Usability Risk

The focus needs to shift away from what a developer wants a technology to do and toward what the user and the patient need it to do. This is where a holistic approach to usability and human factors design comes in.

Software development teams become so familiar with their products that they don't see the difficulties someone else may have interacting with it. Assumptions are often made, either explicitly or implicitly, about the expertise and training of the users of the software or device. Software developers may imagine well-trained operators who have memorized the user's manual standing in front of their devices carefully entering information and reading every prompt and bit of text displayed. Reality is often very different. The device may be one of several dozen different devices from different manufactures on a floor, half of which are beeping and screeching alarms. An understaffed, overworked team that may be 6 months behind on its training is dashing from device to device and has perhaps 10-15 seconds to interact with the device before something else demands their attention. These scenarios represent two extremes. The real world lies somewhere in between, but it is likely closer to the latter. The risk of making poor assumptions and generalizations about a device's users become even more likely as the user community grows for MedTech devices outside of clinical settings, such as home healthcare.

Regulatory agencies across the world, who determine whether a medical device can be marketed and sold in their country/region, are increasingly providing guidance asking software developers to think about how their software will be used by people for which it is intended.

The FDA is seeking to focus attention on these concerns with its recent guidance on usability and human factors analysis in the development of medical software [3].

Under this guidance, mistakes made by operators using a software (use errors) are considered to be a problem with the software, not with the user. All too often "We'll address that in the user's manual" has become the work-around for potential usability problems. Issues seen in the field are often explained away as a mistake made by an incompetent or poorly trained user. With this latest guidance, however, blaming an injury or death on incompetent users of a device is no longer an option.   The new paradigm is to design in usability, design out use error.

The main tool to assess the usability of a software device and drive the process of "designing out use error" is the Human Factors Analysis. Manufacturers seeking FDA approval for new devices must submit evidence of systematic human factors analysis of use errors.

Human Factors Analysis - What does it mean?

Human factors analysis in the context of medical software development has three main goals:

- To gain understanding of how the software/ device will be used, including who will be using and in what environment.
- To drive a design that matches the capabilities and limitations of those human users of the software or device.
- To discover possible paths to use errors and eliminate them from the design

A key aspect of human factors analysis is that the analysis process must begin early enough to identify issues and incorporate remedies into the design. Waiting until post-implementation testing to identify usability issues will mean that the remedies will be much costlier to implement. They will be costly because the relatively easy, late-stage work-arounds currently used that involve instructions for use, warnings, and labelling are no longer considered sufficient to address risks posed by usability issues.

There are several complementary analysis types that, when performed over the lifecycle of the software, together constitute what the FDA refers to as "systematic" human factors analysis.

- **Use error and task analysis**
  Very early in the development process, before software designs are created or code is written, the design team works through paper and mental exercises with the goal of understanding how, by whom, and in what environment the software will be used. The experience gained by such exercises is essential to give the designers a basic understanding of the software from the perspective of the people who will use it.

- **Early prototype usability evaluation**
  Early prototyping to get users to identify bottlenecks in workflow and possible misunderstandings of instructions or displayed information. The value of such feedback is, however, directly related to how "real" the prototype is. While value can be obtained through storyboards and wireframe mockups, there is no substitute for seeing how an operator interacts with a functioning UI.

- **Formal usability testing**
  Provides feedback on not just user interface interactions, but also on the user's ability to execute tasks and perform required functions in near-real world environments.

- **Post-market usability studies or surveys**
  Usability concerns do not end with the deployment of a product to the field. Evaluation need to continue after the initial release in order to reveal any needs that were not met or could not have been anticipated. Post-market feedback is essential to drive both design improvements and development process improvements for future releases and products.

  Each of these types of analysis is necessary but, by themselves, are not sufficient to ensure the usability of the software. Together, however, they form the foundation for the development of a safe and usable system.

The Right Tools for The Job

Having the right tools for usability studies and human factors analysis is an important factor in their success. Selection of a UI tool will be driven by a number of factors including the target environment, performance requirements, platform constraints, cost, and other elements of the development tool chain with which the UI tools must work (IDEs, languages, etc.). Added to this list of drivers is the ability to effectively support human factors analysis and usability testing.

Ideally, developers want to perform as much patient and user research as possible and leverage this information during the different phases of the development process. In order to accomplish this, iterations of usability and human factors testing and the ability to quickly implement design changes based upon this testing are necessary. This ideal, however, often runs up against product release deadlines. The pressure of these deadlines results in many instances where developers "explain away" the human factors risks surfaced during usability testing rather than taking the feedback and creating a better product with a better overall user experience, effectively throwing away the value obtained from the investment in usability testing. That investment can be protected, and the value realized through a development framework that make it quicker and easier to develop and prototype UI/UX and, when something doesn't work, contain tools to make it easier to implement UI changes to ensure the best overall user experience.

The need for the UI tools to support usability testing is most critical in the early stages of the design process where quick prototype development and modification is essential to provide working user interface prototypes that can be put in front of representative users. There are a number of good tools available to provide quick UI mockups and prototypes. It is important to remember when looking at potential tools that human factors analysis doesn't end after the initial round of usability testing.

To be most effective, human factors analysis must be integrated into the development process. When human factors analysis is done in isolation, much can get "lost in translation" when the software team implements the actual user interface. The best tools will produce early prototypes of the actual product user interface that will evolve with the rest of the development effort into stable designs and eventually into the finished product. A "throw-away" prototype is a waste and the cost of producing and maintaining them will pressure teams into less frequent usability testing and human factors analysis.   When early prototypes are early implementations, the result is better feedback from testing (users are using real software) and more efficient development (there is no translation, reimplementation, or lost effort).

Another important aspect of the user interface tools is the ability to separate the presentation of task and workflow in the UI from the back-end code, logic, and algorithms. This separation facilitates updates and changes to the user interface throughout the design and implementation process. Usability issues (identified through the ongoing process of human factors analysis) can be corrected in the user interface with minimal disruption to the back-end implementations, even late in the development process.

Summary

The goal of software development teams is always to build great products and minimize the risk of failure. Medical software development adds the consideration of the safety of the product and the risks that it may pose to its users.

When developing new, innovative products, medical device developers need to prioritize the user experience (UX) to develop user friendly and intuitive designs for health technology that can be used properly and safely both today and in the future. Creating a safe, adoptable, and enjoyable medical device UX relies on feedback from the entire user ecosystem, including patients, doctors, nurses, technicians, maintenance providers among others, to drive positive user experiences. The more feedback the development team can get from numerous and varying user architypes, the better adoption a medical device will have and the safer and more properly the device will be used.

The right tools are essential to help drive this process and facilitate early and ongoing interaction of the software with users to gain insight and identify problems. Seamless evolution of early prototypes into productions software allows development teams to efficiently and effectively build, demonstrate, test, vet, and adapt designs to match the needs, capabilities, and limitations of those users. The Qt software is developed so that development teams can quickly and easily import graphical UI designs, done in graphics editors such as Photoshop and Sketch, and automatically generate the software code. Additionally, with Qt they can rapidly prototype on the target device or environment allowing them to evolve early prototypes into your final UI/UX design. This is an essential part of the process of developing a safe and effective software for medical applications.

About the Authors

Bruce Johnston, PhD

Senior Software Engineering Specialist at MedAcuity

Dr. Johnston has 25+ years' experience designing and developing complex software systems and 15+ years' experience working in regulated medical software development. Much of this experience has be focused on UI design and Human Factors aspects of Class II and Class III medical device programs.   Most recently Bruce has been providing strategic insight to a variety of clients on the intersection of modern software development methodologies, including HF/Usability and Agile practices, and the regulated medical environment.
Bruce first used Qt tools for UI development in 2006 and has worked with the Qt tools on a number of projects since.

Roger Mazzella, M.B.E

Sr. Product Manager- Medical, The Qt Company

In charge of Qt's cross platform UI/UX software for the medical industry. Roger has extensive experience in product and services management, regulatory law, compliance, and marketing for the medical device and diagnostics industry.
Prior to joining The Qt Company, Roger brought state-of-the-art regulatory and clinical software applications and services to the medical device and diagnostics industry and delivered quality engineering solutions and products for numerous multinational and Fortune 500 customers. Roger also worked as a quality and regulatory engineer, making significant contributions to the validation and verification of patient monitoring systems.

References

1. Medical Device Recall Report FY2003 to FY2012 Food & Drug Administration Center for Devices and Radiological Health Office of Compliance Division of Analysis and Program Operations

2. Analysis of Safety-Critical Computer Failures in Medical Devices, Homa Alemzadeh, Ravishankar K. Iyer, and Zbigniew Kalbarczyk (University of Illinois at Urbana-Champaign), Jai Raman (Rush University Medical Center), Copublished by the IEEE Computer and Reliability Societies, July/ August 2013

3. Applying Human Factors and Usability Engineering to Medical Devices, Guidance for Industry and Food and Drug Administration Staff; Food and Drug Administration, Center for Devices and Radiological Health, Office of Device Evaluation; February 3, 2016

MedAcuity is a specialized engineering firm that focuses on medical technology software development. Trusted and experienced since 2007, the firm was founded by a team of seasoned software engineers who identified opportunities to fill the gaps in engineering expertise and efficiency facing the medical technology industry. MedAcuity offers extensive experience across the MedTech industry with full life-cycle software development and subspecialties in other areas including tool validation, algorithm development, and cybersecurity. More information is available at www.medacuitysoftware.com.

The Qt Company develops and delivers the Qt development framework under commercial and open source licenses. We enable the reuse of software code across all operating systems, platforms and screen types, from desktops and embedded systems to wearables and mobile devices. Qt is used by approximately one million developers worldwide and is the platform of choice for in-vehicle digital cockpits, automation systems, medical devices, Digital TV/STB and other business critical applications in 70+ industries. With more than 250 employees worldwide, the company is headquartered in Espoo, Finland and is listed on Nasdaq Helsinki Stock Exchange. To learn more visit http://qt.io