We are IntechOpen,
the world's leading publisher of
Open Access books
Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Human-Computer/Device Interaction

Dagmawi Lemma Gobena

Abstract

Any interaction refers to the communication between two or more entities (be it abstract/conceptual or physical entity). Successful interaction is equated from the properties of each entity involved in the interaction as well as the capabilities of the interacting entities. With the diversified use and application of computers and specialized devices for specific tasks, such as biomechanical and biomedical devices, interaction design needs to further study the context of the tasks as well. Moreover, with the inclusion of embedded systems and smart devices, instead of focusing only on the hardware performance, the computer architecture needs to consider the opportunities. Especially, HCI can be improved as the current technologies are giving an opportunity for building smart interaction where the user interacts with devices implicitly and in less obtrusive way. In light of this, the design and architecture of an engineered product need to strive for making the product usable and used while making it useful to the user. And this can be achieved if interaction design is dictated by scrutinizing the user model with respects to the usability attributes in view of the context of its task as well as the platform capabilities and constraints as discussed in this chapter.

Keywords: interaction design, user-centered design, user interface, implicit interaction, HCI, CHI

1. Introduction

It used to be the case, in the mid-1990s, where people use computers only as a means to analyze and calculate sophisticated mathematical, statistical, and/or scientific problems. By then, the design and architecture of computers used to focus on the computational performance and/or storage capacity. By then, computer architecture was more concerned with the set of rules and methods to describe the functionalities of various components and how these components are organized in the implementation of computer systems to achieve better performance.

In view of "better performance," often computer engineers consider improved processing speed, storage capacity, and information retrieval. Otherwise, interaction performance, associated with the user, used to be less considered. This perspective cannot persist after the introduction of personal computers and especially with the proliferation of mobile technologies like smartphones. If the design and architecture consider only the internal architecture of the functionalities of the components, aiming to address data processing and storage requirements only, but not the user profile nor the context of the users' tasks, usability could be compromised.

Often, computer architecture is much concerned with the performance of the device so as to effectively and efficiently execute users' tasks regardless of the user itself. Instead, users are expected to learn how to use the device architecture and comply with the limitation of the platform while executing their task/s. Such consideration is, of course, rooted to the assumptions of the known user profile of the first-generation computers, where a single group of people used to design, build, program, operate, and maintain each machine [1].

After the introduction of personal computers, and when computers become an indispensable and useful machine for everyone, the functions that can be performed by computers and devices were abstracted at the software layer so as to mediate the interaction language between the machine and the human-user. Particularly, the user interface of software (i.e., application software) determines how users can communicate with the device so as to access a particular service in a machine.

As shown in **Figure 1**, though users actually interact physically with a device, they actually require it to execute a use case to accomplish their need. Hence, users are interacting logically with the service. Software engineers define the service as a use case that is realized by a certain subsystem/component in the software, while the interface is considered as boundary class during analysis and as the user interface during the design and implementation stage.

As the performance of the user is also important to benefit the improved performance of the computer/machine, the user interface design process (usability engineering) takes due consideration to understand the machine. As a result, one of the important tasks in usability engineering lifecycle is scrutinizing the platform capabilities and constraints [2–4]. The platform capabilities are used to meet users' requirements [5] but are limited as specified in the respective architecture of the computer/device. Similarly, the constraints are the result of the architectural organization of the components in the device.

A software that is designed to run on a certain machine (platform), as it benefits from the capabilities of the computer/devices, also could be limited in having some desired functionalities because of the constraints associated with the device architecture. Thus, the human-computer interaction (HCI) further needs to identify and/create interaction modality to complement the limitations while complying with the architectural description of the machine. Such approaches may enable and support users to effectively and efficiently execute their use cases (tasks/functions) on the constrained platform.

With the diversified use and application of computers and specialized devices for specific tasks, such as biomechanical and biomedical devices, the interaction design needs to further study the context of the tasks as well. Moreover, with the
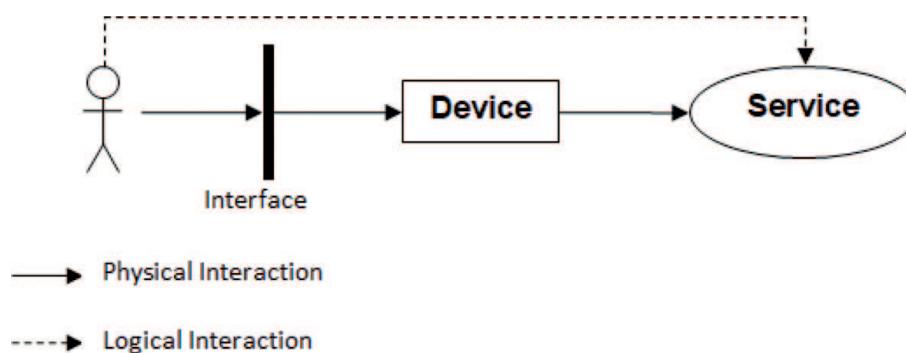
Figure 1.
Interaction between user, device, and service.

inclusion of embedded systems and smart devices, instead of focusing only on the hardware performance, smart technologies shall be considered as opportunities in the design of HCI, so that user interface/interaction designers can easily and systematically assimilate user interaction modalities in the devices.

In this chapter, we provide HCI basics first and then discuss the process of effective HCI development. Finally, we introduce the broad categories of HCI, explicit and implicit, in view of recent computational models and embedded systems, which are an integral part of specific purpose computer system as in vehicles and industrial, biomechanical, and biomedical devices/machines.

2. HCI basics

Lately, when computers of the fourth generation evolve and as application software became indispensable resource and widely used, software engineers use the term "user-friendly" in connection with the interaction and to denote how the software is convenient for users. However, this term is not appropriate for various reasons as indicated in [3], and it is not measurable. Technically, it may imply the consideration of a single dimension of user profile and users' needs, while indeed users' profile and needs are heterogeneous.

When supporting the user performance became equally important to computer performance, alternative terms like usability, user experience, end-user criteria, etc. are coined. In line with user interface design, the field evolved as HCI studying the interaction between the computer/devices and the human-user.

HCI primarily focuses on improving the user performance and satisfaction during interactions the users encounter with the computer/device. Though the "C" in the abbreviation "HCI" refers to the word "computer," technically it must mean both the hardware and software. And, by hardware, we mean any device or machine, or it may simply refer the device, which is an engineered product. But, since software is also an integral part of a computer system, in the following sections, by "engineering product" we mean the device or the software—or both.

2.1 Goals of HCI

Whether the interaction is with the hardware part of computer (machine or device) or the software, the goal of HCI design is to improve the usefulness, usability, and/or usedness of the engineered product [6]. It is also important to provide an intuitive and natural interaction experience for the user [7], so as to make the product acceptable by the user.

The acceptability of the product can have technical or social aspect [7] or the combination of both [3]. Though such classification of acceptability could be useful for simplifying the understanding, some of the attributes considered to measure the practical acceptability could be closely related to social acceptability.

A product's acceptability is related to usefulness of the product [8]. Nielsen [3] considers usefulness as "practical" acceptability, which in turn is divided into two categories, "utility" and "usability."

While utility is associated with meeting the functionality of the product, usability can further take a multidimensional concern, and it could be important to support more than single user, and this forms community subjected to social value [9]. Yet, satisfaction, which is considered as one of the practical usability metrics, could greatly affect the social acceptability. For instance, users could just simply dislike the product.

2.1.1 Usefulness and usedness

A useful engineered product (hardware/software) needs to meet the user requirement. That is, the product shall help the user to accomplish its task/s.

Users primarily desire to accomplish their task, which is the functional requirement, through the interaction they undertake with the engineered product. Having the notion of utility then, which is also, among the end-user design goal criteria [10], HCI focuses on the construction of effective interaction so as to make the computer useful to the user.

Usefulness is related to the perceived value of the product. That is, how users interact with the engineered product is not an inhibitor to the adoption of an engineered product if its value is high [11]. It has to be noted that if a product is useful, the user might need to learn how to use it, even if learning the product could be difficult. Also, even after learning the "useful" product, which has high utility value, the interaction with the product might not be pleasing or safe or both. Hence, the goal of HCI includes production of usable products—making the useful product usable.

2.1.2 Usability

In the case where an engineered product is useful but not usable, usability could often be compensated through trainings and technical supports. When a useful product is possible to affect the operation or safety of the user, it is common to use precaution mark or dialogs, such as warning or disclaimer, so that the user may use the product with care but with his/her own risk.

If the motor capability inhibits the user from operating a product that is perceived useful (e.g., due to disability), then supplementing the user's physical limitation through additional accessories could be vital. For example, due to stroke, a user might be unable to control his/her fingers as usual. Such people, therefore, will experience difficulty with interactions that may require keyboard use and typing due to their limited motor skills of the hands and fingers. Hence, it is important to consider assistive typing or Typing aid, as shown in **Figure 2**, for improving the performance of the user.

Construction of a device and computer system needs to take into consideration on how users interact with it. While performing their task using the engineered product as utility, it is preferred to support users to interact with the engineered

Figure 2.
Use of typing aid for improving typing accuracy for people with limited motor skills of the hands and fingers [12].

product easily, naturally, safely, and securely [6]. Hence, producing a usable product is one of the HCI goals.

Usability is not a one-dimensional attribute of an engineered product; rather it comprises several components that can sometimes conflict; hence, interaction designers need to trade off between the various quality attributes. In order to trade off between important usability attributes and/or construct usable product, it is important to study and understand the human needs as well as ability. Hence, understanding the physical, cognitive, perceptual, personality, and cultural differences [9] of users to whom the product is engineered is important.

While computer design and architecture focus on improving the performance of the hardware, nowadays, it is important to consider the performance of the user as well who would use the product. Such consideration will make the product usable; and usability engineering strives to attain it.

The user performance is associated with various usability attributes. Models have been suggested to understand and represent the user performance so as to apply it during the construction of usable product and latter to qualitatively and quantitatively evaluate the engineered product.

The level of the focus bestowed to the various usability attributes may overlap and/or vary in scope. In fact, as usability is getting universal acceptance, the attributes considered in various models are increasing. For example, the usability consideration in the ISO 9241-11 covers three attributes (effectiveness, efficiency, and satisfaction) in 1998. Then, in 3 years, in the ISO 9126 model, extended model has been proposed to include and clarify the concern into understandability, learnability, operability, attractiveness, and even compliance to standard.

The usability consideration is also evolving along with the improving performance of engineered products and is becoming diversified. Those models which are proposed to simplify usability concern by grouping usability attributes are documented in [13]. These attributes can be useful to systematically capture the physical, cognitive, perceptual, personality, and cultural differences of users.

The usability attributes are useful to objectively measure the interaction design from motor capability, psychological makeup, perceptual and cognitive model, and experience and expectation perspectives of users. Yet, as interaction is not one directional, it is possible that the engineered product could affect the user performance. Following, we have summarized the attributes, shown in **Figure 3**, in alignment with psychological, motor capability, and perceptual and cognition model of the user.

Perceptual and cognitive model of the user: Aligning the design and architecture of engineered product toward the perceptual and cognition model (or mental model) of the user would help to easily engage the user; and the more a user is engaged and utilizes the product, the more he/she learns and understands it.

A product that is easy to understand and learn would improve the user experience [14]. Users' performance, as well, will improve as they learn the product and when time passes by using the product. Thus, learning and understanding the product is important to improve the performance and experience of the user.

The important attributes considered in various usability models are grouped and described as shown in **Table 1**. We took learnability, from Nielson, Shackel, ISO 9126, and Eason model [3, 15, 16], understandability from the ISO 9126 standard, and memorability from Nielson's model [3] as attributes that belong to the perceptual and cognitive dimension.

Psychological makeup of the user: Even if users are physically able to interact with an engineered product, it may not be always the case that they will be willing or interested in using it. Why users might dislike the product or be anxious about it, why users are not motivated to use the product or attracted to, and why users are not eager to use the product are not necessarily related to the usefulness of the
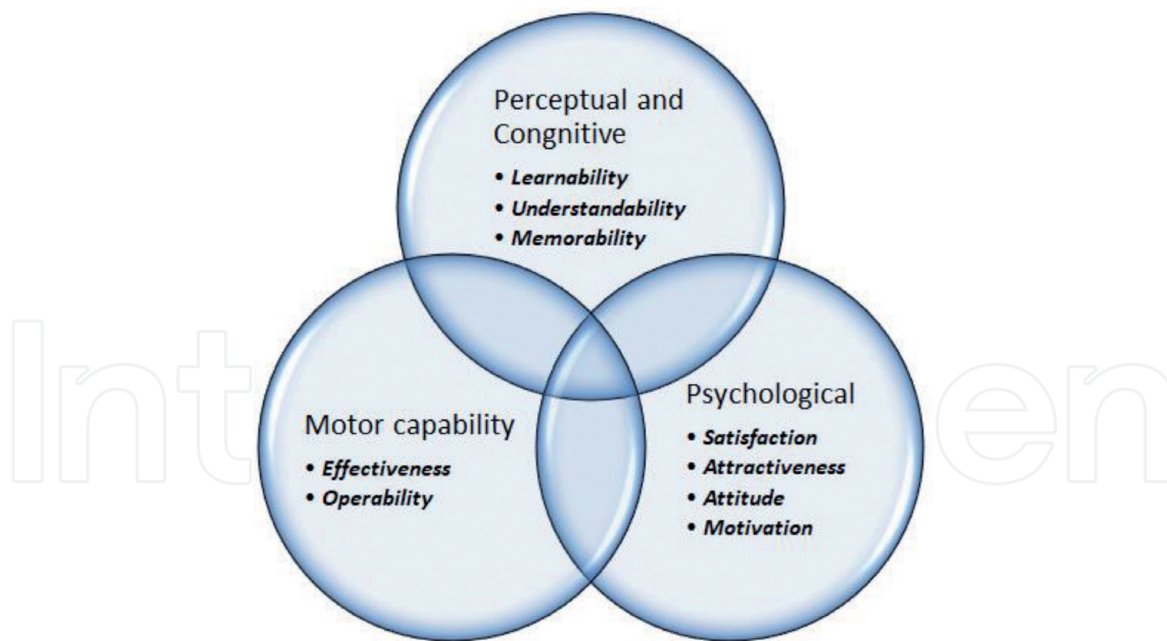
Figure 3.
Usability dimensions and related attributes.

Attribute	Description
Learnability	As an engineered product becomes usable, it would have the benefit of reducing training and supports to the end user. This usability attribute can also be considered as a scale for measuring the effort the end user needs to learn the system
Understandability	This attribute is a supplement to learnability as it determines the ease of which the systems' functions can be indeed understood by the user
Memorability	This is a more desired attribute and primary usability goal in areas where users are casual types. That is, when users are expected to use the system casually, then the system should be easy to remember

Table 1.
Usability attributes related to perceptual and cognition model of the user.

product. Even people, who enjoy using computers, may have very different preferences for interaction styles during usage of specific application in the computer [9].

Rather some usability attributes, which can affect the user performance (directly or indirectly), could be related to psychological makeup of the user. The satisfaction attribute used in Nielson's model [3] and ISO 9241-11 [17], attractiveness from (ISO 9126), attitude from the Shackel's definition of usability [15], and motivation from Eason et al. [16, 18] can be grouped into psychological scale as detailed in **Table 2**.

Motor capability of user: Motor capability refers to the physical orientation of a user. The user's engagement depends on his/her motor capability toward operating the machinery. This encourages the user to improve the understanding about the engineered product with which he/she interacts with [19]. Therefore, it is useful to understand and model user's ability as well as disability. From the engineered product's perspective, the operability of the product and the effectiveness of the user in performing a task are the major attributes considered from the motor capability dimension, as discussed under **Table 3**.

2.2 The building blocks

Having a multidimensional perspective of HCI, such as the mental, psychological, and physical dimensions, profiling the user is important since users can have

Attribute	Description
Satisfaction	This attribute is associated with acceptability of the engineered product by the user as the product is pleasant to use, in a specified context of use, so that the user is subjectively satisfied with it
Attractiveness	This attribute is related to the look and feel of the engineered product. As "beauty is in the eye of the beholder," meeting this requirement could be challenging, especially when the personality of users is diverse. Hence, applying multiple interaction styles/methods could be vital
Attitude	Shackel [15] defines this psychological attribute as "the acceptance of users with their levels of discomfort, tiredness, frustration and personal effort." It has to be noted that users do not always operate the engineered product in a silent and comfortable environment. Some environments might create additional stress and distraction, while the frequency of the distraction may also vary
Motivation	This refers to the determination of the user in using the engineered product for completing the task. Some projects deliver useful products, but users might "just" disregard it and may prefer to stick to old trends. In some cases, attractive engineered products (e.g., smart phone) might improve the motivation of the user, which is associated with design solution. Yet, other solutions, such as rewarding the user might be useful. For example, Ethiopian Airlines, when it first introduced online flight booking, offered 2% discount for those who book online, and this could be one strategy to motivate using the system—on which the company invested. Otherwise, if the system is not well used, the entire investment on the production of the given system could be a waste

Table 2.
Usability attributes related to the psychological makeup of the user.

Attribute	Description
Operability	Engineered product shall enable the user to operate and control it. But how much the user can control the product and the degree of usage is associated with the motor capability of the user
Effectiveness	Associated with the user's accomplishment of a specified task or goal successfully timely, it is also performance measurement of specific user groups. The effective utilization of the engineered product is directly related to the user's physical capability; hence, we associate it with motor capability

Table 3.
Usability attributes related to motor capability of user.

their own capabilities and constraints. Once the user profile is defined, setting goals to be achieved in the construction of the interaction modality and in alignment of the users' profile is important. For example, visually receiving information could be a capability of the user yet can be constrained by the visual acuity.

A successful design solution of HCI shall also be associated with the platform (which can be hardware or software) and need to consider the context of the user task. Hence, platform capabilities and constraints are determinants to meet the goal of HCI. The platform capabilities considered in the design of a certain engineered product shall be in alignment to the capability of the user. Likewise, the identified user capability shall help the design decision during the selection between various competitive platforms. For example, an industrial machine to be operated with touch screen interface is usable for users with hand/s and who are not visually impaired.

The user and platform are two of the building blocks that need due consideration to meet the goals of HCI; and the task is the third block. By "task," we mean the use case that is meaningful in the system under construction. The measurement of a particular task accomplishment is related to the context of the environment

in general. For example, performing a task in a destructive environment and silent environment requires different design consideration in building successful HCI. Hence, HCI needs to take into account the context of the task as well as the environment where the task should be executed.

Furthermore, a user might be able to accomplish a particular task effectively and efficiently if it knows how to initiate the task, how the task used to be known in one environment (e.g., real-world environment) is represented in the newly developed product (e.g., computer/virtual environment), and how the engineered product gives feedback when the task is completed. In this regard, interaction design shall apply appropriate metaphor selection and application.

3. Command and feedback

The goals of HCI can be meet by undertaking engineering process that may result with effective and usable two-way communication between the user and the engineered product. This two-way communication often appears as a command issued by the end user and feedback responded by the product.

The mode of command determines how the user easily issues the instruction to the engineered product while using it. This is further associated with the motor capability of the user as discussed above. For example, in the design of biomechanical systems, as biomechanics focus on the structure and function of the mechanical aspects of biological systems, the motor capability of the user would be vital in the design consideration, among other usability factors discussed above. However, since the interaction is a two-way communication, it is also important to properly design the feedback of the engineered product, in such a way the feedback can be well received and understood by the end user.

Users expect adequate feedback about their actions and ways of easily checking the status [9] of the task they are executing. Feedback has traditionally been seen as a responsive facility to be a communication of immediate system state to the end user [20].

The effectiveness of the feedback from the system is equated to the appropriate selection from tones of interaction styles, where the selection is based on the user, task, and platform model. That means:

- The persona needs to allow the user to receive and understand the feedback.

- The feedback needs to be produced in the context of the task and with the consideration of the task environment.

- The platform (the engineered product) needs to have the capability to produce the feedback.

Renaud and Cooper [20] classify feedback as archival and immediate. Archival feedback, primarily, is related to information about past activity such as a history of both the user's actions, together with the system's response. It can also be considered "similar to an instruction manual or a system overview that allows users to preview system operations" [21]. Whereas, immediate feedback is used to communicate the present state of the system and involves real-time prompts [21]. In both types of feedbacks presented in [20], the interaction design considers modeling the user so that the user can receive the feedback well, while the assumption is that the user interacts with the system explicitly.

4. Explicit versus implicit interaction

In the customary computing models, users are expected to interact with the system (engineered product) explicitly—hence explicit HCI (e-HCI). In the case of e-HCI, the interaction design aims to design and present the engineer product in such a way users know the presence of the product, and they should learn and understand how to interact with the product. Hence, it is required to design the computer-to-the-user.

However, as the result of smart computing models like ubiquitous computing, where the interaction is desired to be implicit, the notion of the command and feedback design might requires to designing the user-to-the-computer as well so as to enable the computer perceive the context of the user [22]—leading to implicit HCI (i-HCI) [6].

Weiser's vision of ubiquitous computing, which demands that computer to be an invisible servant [23, 24], can be realized if the user interacts with the engineered product less obtrusively. This is also pertinent in designing smart devices that improve the efficiency and automation of industrial devices.

Also, with regard to HCI, invisibility of computers can be achieved, partly through i-HCI [22] and context-aware systems. On the other hand, in addition to providing natural interaction (such as speech and gesture), the e-HCI development for interactive ubiquitous systems requires consideration of capabilities and constraints of heterogeneous platforms and users. Hence, the designers and developers are often compelled to configure and integrate heterogeneous platforms to meet needs of ubiquitous computing, such as mobility and implicit interaction. Therefore, both considerations of designing the-user-to-computer and the computer-to-the-user might be required.

In interactive systems where the user interacts with the engineered product explicitly, the user needs to have the model of the system. Thus, the interaction design focuses on crafting the computer/device, assuming the user will understand the presence of the engineered product, and it can operate it with its motor capability. Thus, the focus is more about the human-to-computer (H2C) interaction.

For example, consider a user who conducts quality inspection in a manufacturing process while the item is being manufactured, the item might need to pass over a conveyor belt. The user (inspector) needs a control over the conveyor—from turning on/off the conveyor to speed control. Thus, the user needs to explicitly access the control panel. In such case, the important usability consideration for the interaction design is positioning the control panel using the right metaphors for the interface. Otherwise, the user would have the knowledge on the existence of the control panel and the associated buttons/switches. Yet the design of the control panel shall consider various usability attributes discussed in Section 2.

Alternatively, the interaction could be computer-to-human. For example (considering the example in the preceding paragraph), instead of requiring the user to directly manipulate the control panel, the conveyor could be designed to be smart and know the absence/presence of the user (inspector) might take actions autonomously or advise the user on favorable actions. This approach makes the interaction implicit.

In i-HCI, the presence of the user in the computing model is not primarily to interact with the product. But, the presence can be sensed by the computer, and the computer shall take actions (give feedback to the user) based on previously or dynamically modeled user's context. Implicit interaction is based on the assumption that the engineered product, with which the user interacts, has a certain understanding of users' behavior and action in a given situations being a user-aware [6]. Thus, the design and architecture of the engineered product need to consider

additional components (such as sensors and actuators) other than the components useful to attain the functional operations.

5. Conclusion

Any engineered product is designed and constructed with the intention of solving user's problem, often, through improvement of the user performance and capacity. And the users need to interact with the product in order to utilize it for accessing the service/s or functionality/functionalities provided by the respective product.

The interaction between the user and the product is primarily for addressing the user's problem which is associated with the usefulness of the engineered product. But the usefulness of the product is only one aspect. Otherwise, the acceptability of the product is associated with the usability and usedness of the product as well. Therefore, considering computer/device as an engineered product, the design and architecture of any engineered product shall give due consideration for the design of human-computer/device interaction.

First and foremost, as a user is one who operates and interacts with engineered product; it is important to properly profile the user considering the user's motor capability, psychological makeup, as well as the perceptual and cognitive model. Secondly, as the engineered product, which eventually interacts with the user, would be constructed over a certain platform and set of technologies, it is worth to consider the capabilities and constraints of the platform over which the product is designed or the product itself. Therefore, the understanding of the user as well as the platform capabilities and constraints would help to consider the right types of interaction modalities.

Also, in the design and architecture of an engineered product, in today's technology and with the emerging needs of smart computing, interaction design might lead to the design of implicit interaction between the product and the user. This, in turn, requires integration of various technologies such as RFID, sensors, and actuators to build smartness in the engineered product. Hence, the design and architecture of engineered product need to take into account the specification and organization of additional components beyond the ones used for processing the functional requirements and storing the information produced in relation to the task.

Author details

Dagmawi Lemma Gobena
Addis Ababa University, Ethiopia

*Address all correspondence to: dagmawi.lemma@aau.edu.et

IntechOpen

References

[1] Andrew S. Tanenbaum Modern Operating Systems. 4th ed. New Jersey: Pearson Education; 2015

[2] Mayhew D. The Usability Engineering Lifecycle. San Francisco: Morgan Kaufmann Publishers; 1999

[3] Nielsen J. Usability Engineering. London, San Diego: Morgan Kaufmann; 1993

[4] Ratner J, editor. Human Factors and Web Development. 2nd ed. New Jersey: Taylor & Francis e-Library; 2009

[5] Magno Q , Paul T, Tim C, Rajeev S. Corporate knows best (maybe): The impact of global versus local IT capabilities on business unit agility. In: Proceedings of the 51st Hawaii International Conference on System Sciences (HICSS); 2-6 January 2018. Hawaii, USA: HiCSS; 2018. p. 5212-5221

[6] Posland S. Ubiquitous Computing: Smart Device, Environment, and Interactions. Chicester, UK: John Wilesy & Sons Ltd; 2009

[7] Dong Wu, Jianhua G, Yi Li. Interaction technology based on 3D printing topographics and table for emergency management. In: Proceedings of the 2018 2nd International Conference on Big Data and Internet of Things (BDIOT 2018); 24-26 October 2018; Beijing, China: ACM; 2018. p. 100-104

[8] Giovanny A, Auguste R, Eric B. Improving users' product acceptability: An approach based on Bayesian networks and a simulated annealing algorithm. International Journal of Production Research. 2016;**54**(17):1-18

[9] Shneiderman B, Plaisant C, Cohen M, Jacobs S, Elmqvist N, Diakopoulos N. Designing the User Interface: Strategies for Effective Human-Computer Interaction. 6th ed. Harlow: Pearson Education; 2018

[10] Bruegge B, Dutoit A. Object Oriented Software Engineering: Using UML, Patterns and Java. 3rd ed. Harlow: Prentice Hall; 2014

[11] Derrett N. Heckel's law: Conclusions from the user interface design of a music appliance—the bassoon: Pers Ubiquit Comput. 2004;**8**(3-4):208-212

[12] www.sportaid.com. [Online]. https://www.sportaid.com/typing-aid.html

[13] Madan A, Kumar S. Usability evaluation methods—A literature review: International Journal of Engineering Science and Technology (IJEST). 2012;**4**(2):590-599

[14] Allison G, Michael A. When cognition interferes with innovation: Overcoming cognitive obstacles to design thinking; research. Technology Management. 2018;**61**(4):45-51

[15] Shackel B. Usability – Context, framework, definition, design and evaluation. Interacting with Computers. 2009;**21**(5-6):339-346

[16] Eason K. Towards the experimental study of usability. Behaviour & Information Technology. 1984;**3**(2):133-143

[17] Nigel B, James C, Susan H. ISO 9241-11 revised: What have we learnt about usability since 1998? In: Human-Computer Interaction: Design and Evaluation. Vol. 9169. Cham: Springer; 2015

[18] Tomayess I, Pedro I. Usability and human computer interaction (HCI). In: Sustainable Design. London: Springer; 2015

[19] Vladimir R. The old media in the new media: A comparative. International Journal of Performance Arts and Digital Media. 2019;**15**:23-24

[20] Renaud K, Cooper R. Feedback in human-computer interaction—characteristics and recommendations. 2000;**26**:105-114

[21] Chiang P, Chang H, Chang Y. PotteryGo: A virtual pottery making training system. IEEE Computer Graphics and Applications. 2018;**38**(2):74-88

[22] Page T, Gisli T. Emerging technologies in interaction with mobile computing devices—A technology forecast. i-manager's Journal on Mobile Applications and Technologies. 2018;**5**(1):1-14

[23] Adam G. Everyware: The Dawning Age of Ubiquitous Computing. Berkeley, USA: New Riders Publishing; 2006

[24] Weiser M. The computer for the 21st century. Scientific American. 1991;**263**(3):94-104